

Santa Clara University

Scholar Commons

Computer Science and Engineering Senior
Theses

Engineering Senior Theses

6-3-2026

IAJES Website Redesign

Justin Fan

Nicolas Gibson

Isabelle Hu

Anisha Malani

Follow this and additional works at: https://scholarcommons.scu.edu/cseng_senior



Part of the [Computer Engineering Commons](#)

SANTA CLARA UNIVERSITY
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Date: June 3, 2026

I HEREBY RECOMMEND THAT THE THESIS PREPARED UNDER MY SUPERVISION BY

Justin Fan
Nicolas Gibson
Isabelle Hu
Anisha Malani

ENTITLED

IAJES Website Redesign

BE ACCEPTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

BACHELOR OF SCIENCE IN WEB DESIGN AND ENGINEERING

Silvia Figueira

Silvia Figueira (Jun 4, 2026 14:55:48 PDT)
Thesis Advisor [Silvia Figueira]

Silvia Figueira

Silvia Figueira (Jun 4, 2026 14:55:48 PDT)
Department Chair [Silvia Figueira]

IAJES Website Redesign

by

Justin Fan
Nicolas Gibson
Isabelle Hu
Anisha Malani

Submitted in partial fulfillment of the requirements
for the degree of
Bachelor of Science in Web Design and Engineering
School of Engineering
Santa Clara University

Santa Clara, California
June 3, 2026

IAJES Website Redesign

Justin Fan
Nicolas Gibson
Isabelle Hu
Anisha Malani

Department of Computer Science and Engineering
Santa Clara University
June 3, 2026

ABSTRACT

The International Association of Jesuit Engineering and Science Schools (IAJES) connects more than sixty Jesuit institutions worldwide to promote collaboration and shared learning in STEM research. Santa Clara University is one of more than sixty member universities whose faculty and staff participate in conferences and task forces to address priority issues identified by the Society of Jesus. IAJES' existing website, built on a static no-code platform, limited the organization's ability to manage content, support member interaction, and scale with its growing global network. Users faced unclear navigation options, a basic design, and no effective tools for managing roles, resources, or communication. These challenges directly hindered IAJES' mission of fostering international collaboration.

This project addressed these limitations through the design and development of a dynamic website that improves usability while remaining easy for non-technical administrators to maintain. The platform supports secure member authentication, role-based editing privileges, centralized announcements, newsletter creation, task force management, and interactive galleries for meetings and resources. Guided by human-centered design, stakeholder feedback, and iterative prototyping, the resulting system provides IAJES with a scalable, maintainable, and professional digital hub that strengthens communication and supports future growth across its global community.

Table of Contents

1	Introduction	1
1.1	Problem Statement	1
1.2	Background or Related Work	2
1.3	Objectives	3
1.4	Our approach	3
2	User Research	5
2.1	Methods	5
2.2	Stakeholder needs	6
2.3	User stories	6
3	Design and Rationale	8
3.1	Design	8
3.2	Functional requirements	9
3.3	Non-functional requirements	9
3.4	Rationale	10
3.5	Use Case Overview	11
3.6	Use Case Descriptions	11
3.7	Use Case Diagram	17
3.8	Class Diagram	18
3.9	Sequence Diagram	19
3.10	Activity Diagram	21
4	Design Overview	22
4.1	System Components	22
4.2	General Design	22
4.3	Prototypes	23
4.3.1	Sign-up, Login, and Logout	23
4.3.2	University, Page, and Profile Search	26
4.3.3	Profiles	29
4.3.4	Landing	32
4.3.5	About Us and Organizational Structure	34
4.3.6	Biennial Meetings (International Meetings) and Regional Meetings	38
4.3.7	Webinars and Projects	44
4.3.8	Participatory Activities, Conferences, Contests, and Research	48
4.3.9	Task Forces	52

5	System Evaluation	58
5.1	Internal Testing	58
5.1.1	Experience	58
5.1.2	Functionality	59
5.2	External Testing	60
5.3	Unit Testing	61
5.4	Integration Test(s)	62
5.4.1	Overview	62
5.4.2	Test 1: Signup-to-Profile Data Flow	62
5.4.3	Test 2: Profile Field Mapping Verification	63
5.4.4	Test Execution	63
5.4.5	Summary	63
6	Implementation Plan	64
6.1	Timeline	64
6.2	Agile software development	67
6.3	Project Risks	67
7	Constraints and Standards	69
7.1	Constraints	69
7.1.1	Integration with Existing IAJES infrastructure	69
7.1.2	Accessibility and Frugal Innovation Constraint	69
7.1.3	Security and Privacy Constraint	69
7.1.4	Time Constraint	70
7.2	Standards	71
7.2.1	Unified Modeling Language (UML) Standards	71
7.2.2	General Data Protection Regulation (GDPR) - European Union Privacy Law	71
7.2.3	OWASP Top 10 Web Application Security Risks - OWASP Foundation	71
7.2.4	California Consumer Privacy Act (CCPA) - California Privacy Law	71
8	Societal Issues	72
8.1	Ethical	72
8.2	Social	73
8.3	Political	73
8.4	Economic	74
8.5	Health and Safety	74
8.6	Manufacturability	75
8.7	Sustainability	75
8.8	Environmental Impact	76
9	The Final Product	77
9.1	Differences Between Prototype and Final Product	77
9.2	Final Pages	77
9.2.1	Sign-up and Login	77
9.2.2	Profile Search	79
9.2.3	Profile Page	81
9.2.4	Landing Page	84
9.2.5	About IAJES	87
9.2.6	Organizational Structure	89
9.2.7	International Meetings (Biennial Meetings)	91
9.2.8	Regional Meetings	93
9.2.9	Video Resources	98

9.2.10	Webinars	102
9.2.11	Task Forces	105
9.2.12	Newsletter and Newsletter Editing	110
10	Conclusion	114
11	Acknowledgments	116
	Bibliography	117
A	Code Files	120
A.1	root.jsx	120
A.2	styleguide.jsx	121
A.3	app.css	130
A.4	search.jsx	143
A.5	video-resources.jsx	149
A.6	video-resource.jsx	155
A.7	menu.jsx	162
A.8	popup.jsx	168

List of Figures

3.1	Use Case 1: User Login Process	12
3.2	Use Case 2: Edit Profile	13
3.3	Use Case 3: Filter Results	14
3.4	Use Case 4: Add Task Force	15
3.5	Use Case 5: Edit Task Force	16
3.6	UML Use Case Diagram: IAJES Website Redesign	17
3.7	Class Diagram of the IAJES Website	19
3.8	Sequence Diagram: Account Creation and Verification Logic	20
3.9	Activity diagram for account creation functionality	21
4.1	Color Scheme	23
4.2	Sign-up	24
4.3	Login	25
4.4	Logout	26
4.5	University Search Results	27
4.6	Profile Search Results	28
4.7	Page Search Results	29
4.8	Profile	30
4.9	Profile (Editing View)	31
4.10	Profile (Settings View)	32
4.11	Landing	33
4.12	About Us	35
4.13	About Us Announcement Editing Popup	36
4.14	Organizational Structure	37
4.15	Biennial Meetings (Meetings List View)	39
4.16	Biennial Meetings (Detail View)	40
4.17	Regional Meetings (Region List View)	41
4.18	Regional Meetings (Meetings List View)	42
4.19	Regional Meetings (Detail View)	43
4.20	Regional Meeting Gallery Edit Popup	44
4.21	Webinars (List View)	45
4.22	Webinars (Detail View)	46
4.23	Projects (Detail View)	47
4.24	Participatory Activities	49
4.25	Conferences	50
4.26	Contests	51
4.27	Research	52
4.28	Task Forces (List View)	54
4.29	Task Forces (Detail View)	55
4.30	Task Forces "What We Do" Edit Popup	56
4.31	Task Forces "Who We Are" Edit Popup	57

6.1	A Gantt Chart depicting the project process	65
6.2	A detailed spreadsheet Gantt Chart with more precise dates.	66
9.1	Sign-up	78
9.2	Login	79
9.3	Profile Search	80
9.4	Profile	82
9.5	Profile Editing (1)	83
9.6	Profile Editing (2)	83
9.7	Landing	85
9.8	Landing Editing (1)	86
9.9	Landing Editing (2)	86
9.10	About IAJES	88
9.11	Organizational Structure	90
9.12	International Meetings	92
9.13	Regional Meetings (Region List)	94
9.14	Regional Meetings (Meeting List)	95
9.15	Regional Meetings Detail	96
9.16	Regional Meetings Editing (1)	97
9.17	Regional Meetings Editing (2)	97
9.18	Regional Meetings Editing (3)	98
9.19	Video Resources	99
9.20	Video Resources Detail	100
9.21	Video Resources Editing (1)	101
9.22	Video Resources Editing (2)	101
9.23	Webinars	103
9.24	Webinars Archive	104
9.25	Task Force	106
9.26	Task Forces Detail	107
9.27	Task Forces Editing (1)	108
9.28	Task Forces Editing (2)	108
9.29	Task Forces Editing (3)	109
9.30	Task Forces Editing (4)	109
9.31	Newsletter	111
9.32	Newsletter Draft Selection	112
9.33	Newsletter Draft Article Editing	113

Chapter 1

Introduction

1.1 Problem Statement

The International Association of Jesuit Engineering and Science Schools (IAJES) serves as a central hub for collaboration among institutions dedicated to Jesuit values in STEM education. Despite its importance as a global network, the current IAJES website faces significant usability and functionality challenges that limit its effectiveness in engaging stakeholders, facilitating collaboration, and providing intuitive access to resources. Users often encounter difficulties navigating the site, accessing content across different devices, and managing interactions across multiple time zones. The existing user interface is outdated, lacks responsiveness, and provides limited backend capabilities for content management and user administration. These issues collectively reduce the organization's ability to maintain effective communication and resource sharing among member institutions worldwide.

To address these challenges, we propose a comprehensive redesign of the IAJES website that focuses on improving both the user experience (UX) and the backend functionality. The proposed redesign is guided by human-centered research, stakeholder feedback, and international accessibility standards. Specifically, the project aims to enhance navigation, improve content organization, implement responsive design, and introduce scalable backend features such as content management systems, user roles, and resource-sharing capabilities. This approach ensures that the redesign is grounded in data-driven insights rather than speculative assumptions about user needs.

We achieve these goals by conducting structured stakeholder interviews, analyzing user navigation patterns, and evaluating the website's accessibility and usability using both quantitative and qualitative metrics. This framework allowed us to identify critical pain points that hinder efficient interaction and engagement. By integrating best practices in UI/UX design with modern backend development technologies, the project provides a holistic solution that balances technical feasibility with user-centered principles.

This redesign is particularly important for the global Jesuit STEM community because it directly impacts the or-

ganization’s capacity to foster collaboration, share resources, and support international stakeholders. A well-designed, functional, and accessible website can significantly improve communication and engagement, ultimately advancing IAJES’s mission of global academic collaboration. Through this project, we aim to demonstrate the value of combining human-centered design thinking with robust technical development to create a digital platform that effectively serves both organizational and user needs.

1.2 Background or Related Work

Our website was made to address the specific needs of IAJES. No previous project has addressed our exact problem, as our website was the first one coded from scratch for our client.

Before this project, the original website used by IAJES was made using Google Sites. As a no-code editor, this allowed IAJES to achieve the bare minimum: storing resources, announcing upcoming events, and providing details of their organization to the public. Using an editor like Google Sites allows users to create “[h]ighly engaging sites without programming skills” (5). It allowed IAJES to adapt their website through quick drag-and-drop actions, and provided a serviceable, although generic, styling to the content. Due to many IAJES members having a lack of web development experience, switching to an editor that required code seemed unnecessary, especially as it would require much effort from the organization to adjust and delegate new website managers for.

However, this website was difficult to scale, difficult to customize, and lacked important connectivity features that IAJES desired. Google Sites also made it impossible to create dynamic functionalities, with no way to code connections between the front-end and back-end. As their organization started to expand and grow, the no-code builder was not growing with their needs, and something needed to be done. That is why our team was asked to step in. Made from scratch, our custom website was made through IAJES feedback, with their needs guiding our process every step of the way. The final website is to be owned entirely by IAJES, providing them with full control, while still providing quick and accessible editing features—designed with the “no-code” mentality of Google Sites that they are familiar with.

All members of the IAJES organization will likely see and interact with the website, if only to create a profile. This website will be important to spark conversations between IAJES task force groups, and even forming new Task Forces by connecting those with similar research interests. Announcements found on the front page are as easily accessible as before, only now with engaging layouts and design elements. By enhancing the user experience and showcasing IAJES professionally, we believe our website will be found and used much more by the public and the organization themselves.

1.3 Objectives

In order to fulfill our goal of making a new official website for IAJES, our objectives are the following:

- Communicate with IAJES through the whole project to understand their actual requirements, desires, and also to receive feedback on existing work.
- Develop a style guide and a proposal of pages, features, and timeline for IAJES to review to ensure our goals align with theirs before starting development.
- Design the Figma screens of the website to guide the final design of the website. This will also be used to present to IAJES and receive design feedback before coding starts.
- Decide on the technology used and establish a GitHub repository and best practices for managing Git.
- Fully code both the front and back end of the website, while referencing the site proposal, styleguide, Figma, and IAJES feedback.
- Deploy the website through Firebase App Hosting for demo purposes. Later, deploy the website on a custom domain through IAJES' own servers.
- Integrate the website content with IAJES' databases and continue site maintenance and updates as necessary.
- Once updates are no longer needed, we will hand off the website to IAJES web managers.

1.4 Our approach

Our team had a structured approach that encouraged robust development practices and iterative collaboration to produce a polished final product ready for deployment. At a high level, our approach took the following form:

- We first interacted with our client to determine their needs.
- Then, we translate these needs into actionable deliverables that took the form of website functionality, pages, and content.
- After determining our requirements, the development phase started. Throughout this process, we provided constant proposals and updates to our client to get feedback that guided what we implemented and prioritized. To start this phase, we fully planned out the design of the website through Figma prototypes.

- Then, we used all approved proposals, prototypes, and other design decisions to collaboratively code the final website using GitHub. By using a system of pull request reviews and iterative coding, we developed the front-end and back-end alongside each other while ensuring working integration.
- Finally, we host the final website and continue to maintain it for IAJES until it is ready to be fully transferred to them.

By implementing our approach, our work resulted in a system that met our client's needs directly, accommodated new desires on the fly, and structured work well—allowing us to finish in a timely manner. While website development is made easy through websites like Google Sites, Wix, and Wordpress, our site provides a custom experience designed to be as easy to maintain as Google Sites. Functionalities we provide—such as authenticated roles, task forces, and newsletter generation—are tailor-made to match IAJES' mission of member connectivity, easy site management, and professional public presence. Unlike an external team or AI system, our team was invested in the success of our client, as SCU is a part of IAJES itself. This encouraged us to provide IAJES with more than the bare minimum, leading to more ambitious yet creatively valuable ideas being introduced for the website's design and function.

Chapter 2

User Research

2.1 Methods

Due to the nature of our project, our team had direct access to client feedback in order to revise our plans in real-time. At weekly virtual meetings, we talked directly with IAJES's president and vice-president to present our current progress, understand their needs, and adapt accordingly. This constant communication allowed us to customize our features to IAJES' specific requests, and adapt to changes more easily. Using WhatsApp, we also would check-in with IAJES to determine small fixes quickly when needed.

Furthermore, our team wrote and presented multiple documentation items early on, such as a full site proposal with all roles and their permissions, and all the pages to be on the website. This was instrumental to double check that our plan for the website was the same as IAJES envisioned. Similarly, our team designed and presented all website pages in a Figma prototype before development started. By creating these prototypes that we could give to our client, IAJES could see a complete picture of what the final product would look like, and give us comments outside of weekly meetings. For example, our team wanted to know the best color choices for the homepage, so we designed three color drafts for our stakeholders to choose from during a weekly meeting. This process allowed us to realize a fundamental problem in how we designated different sections as different colors. IAJES' feedback told us that none of those three drafts were sufficient, because IAJES wanted a more professional designed look with less blocky color sections. This led us to create another color draft that ended up being the final version.

Throughout the development process, our team wrote down all feedback and observations about IAJES's needs in weekly meeting notes and a master to-do list. During the first two quarters of our senior year, our team would also meet bi-weekly only with our senior design advisors (without IAJES) to discuss our progress. This was useful if further analysis of the comments from IAJES was needed. Plus, if even more analysis was needed, our team met intermittently outside of our weekly meetings in order to understand what we need to work on.

During the summer of 2026 during our final deployment phase, we plan to present our website at an IAJES conference to receive general feedback from IAJES members.

2.2 Stakeholder needs

The stakeholders for this project were: the leaders of the International Association for Jesuit Engineering Schools (IAJES), IAJES members, and the general public. Our team was working directly with IAJES leadership: President Carlos Rivas, and Vice President Raúl García. Their needs for the project included the following: a professional, clear, and truthful presentation of IAJES to the public, allowing members to find and connect with other members, especially through similar research interests, easy editability for those inexperienced with web development, and finally, the secure and long-term storage of IAJES information.

As another stakeholder in the project, certain members of IAJES may have their own specific needs. However, our team is optimizing our project to be applicable to IAJES as an organization as a whole. This is because we believe that the needs expressed by the leaders of IAJES are in alignment with those within the organization.

Finally, the public audience has a stake in this project, as our website will be visible to them, and they will use this as a source of truth about IAJES. Although generalized, we assume that the public prioritizes clarity and truthfulness of information, and easy access to information they want to know. Similar to IAJES members, these desires are also already held by IAJES leaders, and they are enforced by industry web design standards as listed in Section 7.2.

2.3 User stories

General Users

- As a user, I want to filter IAJES profiles by different criteria such as task force, location, research interest, etc., so that I can find and connect with people who share my interests and are nearby to possibly start research projects or learn more about IAJES.

Registered Members

- As a registered member, I want to be able to edit my profile picture, demographics, career details, and contact information, so that my information can be up-to-date.

Task Force Members

- As a member of an IAJES task force, I want to register as an IAJES member on the IAJES website so that I can gain a profile page and allow both task force members and the public to contact me.

Task Force Managers

- As a Task Force Manager, I want to be able to edit and update the information of my associated task force, so that the task force's information can be up-to-date.

Super Admin and Website Managers

- As a Super Admin or Website Manager, I want to be able to create a new Task Force section, so that when a new IAJES task force is established, their associated info, projects, and members can be reflected in the website for the public to learn about.

Chapter 3

Design and Rationale

3.1 Design

Our website is a centralized system that allows IAJES to perform the following functions:

- Signing up, logging in, and logging out a user.
- Storing a database of all authenticated profiles and email-validated IAJES members, alongside their relevant user-submitted information.
- Connecting IAJES members through contact forms and member search with applicable filters.
- Showcasing and storing projects, regional and international meetings, video resources, webinars, and other highlights.
- Drafting and sending out newsletters to all subscribers, keeping them up-to-date on current news about the organization.
- Allowing users to customize or delete their profiles, and edit submitted information such as contact details, industry, job/position, industry, country, major, research interest, and task force affiliations.
- Allowing website administrators to edit website sections such as task forces, meetings, projects, videos, webinars, and the landing page without code.

3.2 Functional requirements

After discussions with IAJES, the following functional and nonfunctional requirements were determined.

1. **The website must allow admins to create, edit, and send newsletters to a predefined mailing list.** The website must support a simple HTML editor to create text and images formatted to fit into an email, regularly and automatically sent out to subscribers on the mailing list. Only admins must be able to use this functionality.
2. **The website must support editing information known to be need of regular updates without accessing the code or database.** There are several pages on the website that contain information that will be updated or edited over time due to changes in the organization. These pages and information must be able to be updated by users unfamiliar with code by only interacting with the interface.
3. **The website must allow users to sign in, with different capacity of control over the site depending on what role the user has.** Pages in which information can be edited can only be edited by admins. Prompts on the webpages for editing capability must be hidden from all other users.
4. **The user profiles must allow a user to send an emailed message through the website.** Each user profile has a section in which the current user can send an email using a form on the website to the person to which the profile belongs to. This form must be accompanied by security measures to prevent bots from spamming users with emails.

3.3 Non-functional requirements

1. **The website design shall be appealing to visiting users on any screen size.** The website layout shall be responsive to different screen sizes and rearrange content to fit smaller or larger sizes in a way that remains accessible, consistent, and appealing to visiting users.
2. **The website shall serve as a platform where IAJES members can discover and connect with one another.** General information about IAJES members on the website shall allow other IAJES members and site visitors to get to know other members of the organization and provide a way to connect.
3. **The website shall be developed in a frugal manner and resources used in development shall be easily affordable and accessible.** For we work in collaboration and under guidance of the Frugal Innovation Hub, development of the website shall also be conducted in the manner of the Frugal Innovation Hub.

4. **Updates to the website shall be reflected for all users using the website.** Parts of the website can be edited through various webpage interfaces, thus for accuracy and to prevent conflict of information, information on the website must be the most recent update for all users unless indicated otherwise.

3.4 Rationale

The system is designed as a single, centralized website in order to make it easy to store all of IAJES' resources in one place. It was also the original proposal, and makes sense for IAJES' fundamental goal of replacing their old website with a new one.

Various functionalities were implemented to address certain goals from IAJES, and other functionalities were implemented due to being standard web development practice. Many were a mix of both reasons. Functionalities such as the newsletter and profile lookup, were implemented to align with IAJES' vision of connectivity. An alternative our team considered was creating a full on-site messaging system, similar to LinkedIn. While this would help with connecting users more easily, this functionality was not chosen due to time constraints (Section 7.1), and an easier alternative of using email was adopted. Users already have multiple messaging platforms, so having them keep track of a whole separate one would have also reduced its usage and therefore not been as valuable. The modular storage of meetings, task force projects, video resources, and more were implemented to allow IAJES to upload new content and manage the existing content they show to the public. Allowing these modular sections to be easily editable through on-site inputs aligned with IAJES' goals to make the website "as easy to edit as Google Sites". Authentication, login, and role permission systems are common practice in website development, and were especially applicable to our editing systems. In order to prevent those not in IAJES from editing the website, a distinction between those with and without permission to make edits had to be established.

React, TailwindCSS, and JS were chosen as the code languages because our team were familiar with the three. Furthermore, React router allows for an easy navigation system. TailwindCSS allows for simple styling changes with less time spent looking between files. And JS is a popular choice that was compatible with Supabase's database back-end. Notably, regular CSS is also used on the website but only for certain specialized elements. However, we prioritized using TailwindCSS for general elements, because it allows us to make quicker changes to styling content, and keep those changes within the same file as the HTML.

The system is designed without an excess of plugins or additional code languages, as more plugins are not necessary for the scope of our chosen functionalities. Furthermore, we wanted to keep our code simple for those who may develop it in the future after we pass the site off to IAJES.

3.5 Use Case Overview

After defining the functional and nonfunctional requirements, the next step for us was a comprehensive visualization and analysis of the website's functionalities. This section showcases our project's Use Case Descriptions and multiple Unified Modeling Language (UML) diagrams (see Section 7.2), including Use Case, Class, Sequence, and Activity diagrams. These diagrams helped translate functional requirements into actionable implementations. By making explicit the interactions between our users and the site, they offered us a systematic representation of the website's behavior, structure, and relationships. Many diagrams look at specific complex scenarios that we needed a robust understanding of before implementation, and others provide a bird's eye view of our whole architecture.

3.6 Use Case Descriptions

The use case descriptions in Figures 3.1 through 3.5 provide details for hypothetical scenarios that both IAJES and our team wanted our website to successfully address. For each case, we determined dependencies, the case description, preconditions, the main and alternate flows, and postconditions. Dependencies included info on what functionalities were needed for this process to be addressed. The description provides a plain-text overview, and the preconditions told us what happened for a given process to start. The main flow is a chain of actions that we intend to happen if everything is successful, and the alternate flow defines actions to take if unforeseen errors occur. Finally, the postconditions are what the user has achieved by taking the main flow.

These descriptions were useful to reference during the development phase, as they provided structure for which actions should come next in the user flow, and reminded us of features we intended to create. Together, they provided a baseline goal for the functionalities of the website.

Use Case 1: User Login Process

Actor: General User

Dependencies:

- Integration with university email authentication
- Secure password hashing and validation service
- Error-handling module for invalid login attempts

Description: The User logs into the IAJES platform using their university email and password. Upon successful authentication, they gain access to features based on their role permissions.

Preconditions:

- User has a registered account with a valid university email.
- The authentication service and database are functional.
- User is on the Login page.

Main Flow:

1. The User selects the “Login” button from the header or landing page.
2. The system displays the login form requiring university email and password.
3. The User enters their login credentials and selects “Submit.”
4. The system validates the format of the entered email and password.
5. The system authenticates the credentials against stored account information.
6. Upon successful authentication, the system starts a secure session and redirects the User to their dashboard.

Alternate Flow:

- If email/password is incorrect, system displays an error message and prompts retry.
- If authentication system is unavailable, user receives a temporary access error.

Postconditions:

- User is successfully authenticated and redirected.
- User session is created and stored securely.

Comments: This login process ensures secure, role-based access across all IAJES functionalities while protecting user data through university email authentication.

Figure 3.1: Use Case 1: User Login Process

Use Case 2: Edit Profile

Actor: Registered Member

Dependencies:

- User profile database
- Image upload service for profile picture
- Input validation utilities for text fields

Description: The user updates their personal profile information, including contact details, job/position, task force affiliations, and visibility settings.

Preconditions:

- User is logged in.
- User has navigated to their “Profile” page.
- Profile data is retrievable from the database.

Main Flow:

1. The User selects the “Profile” option from the header or navigation menu.
2. The system displays the User’s profile page with their current information.
3. The User clicks the “Edit Profile” button.
4. The system shows an editable form with fields such as name, industry, job/position, task force, contact information, and location.
5. The User updates one or more fields with new information.
6. The User selects the “Save Changes” button.
7. The system validates all inputs (e.g., required fields, proper formatting).
8. The system writes the updated profile information to the database.
9. The system reloads the profile view with the updated information and a confirmation message.

Alternate Flow:

- Invalid fields (e.g., special characters, missing required data) trigger error messages.
- If image upload fails, system reverts to previous profile picture.

Postconditions: Updated profile information is stored and displayed according to visibility rules.

Comments: Editing the profile ensures member information remains current and accurate, supporting effective networking and task force collaboration.

Figure 3.2: Use Case 2: Edit Profile

Use Case 3: Filter Results

Actor: General User

Dependencies:

- Search index of member profiles and task forces
- Dynamic filter engine
- Database query optimization utilities

Description: The user filters search results using criteria such as task force, industry, job/position, country, major, or research interest.

Preconditions:

- User is logged in.
- Search results are already displayed following a user query.

Main Flow:

1. The User performs an initial search or navigates to a results page (e.g., members, task forces).
2. The system displays a list of search results.
3. The User selects the “Filter/Sort” option on the results page.
4. The system displays available filter criteria such as name, task force, industry, job/position, country, major, and research interest.
5. The User chooses one or more filter values (e.g., by task force and country).
6. The User applies the selected filters.
7. The system queries the database using the selected filter criteria.
8. The system refreshes the results list, showing only items that match the applied filters.
9. The User optionally adjusts or clears filters to view a different subset of results.

Alternate Flow:

- If no items match selected filters, system displays “No Results Found.”
- If filter service is unavailable, user is shown the unfiltered list with a notice.

Postconditions: Results list reflects the applied filters.

Comments: Filtering improves discoverability by allowing users to quickly refine large datasets into highly relevant results.

Figure 3.3: Use Case 3: Filter Results

Use Case 4: Add Task Force

Actor: Website Manager

Dependencies:

- Administrative role verification
- Task Force database
- Member directory for initial assignments

Description: An authorized administrator creates a new task force by entering its name, description, and initial member list.

Preconditions:

- User is logged in as a Superadmin, Website Manager, or Task Force Admin.
- User is on the Task Force Management page.

Main Flow:

1. The Admin (Superadmin, Website Manager, or Task Force Admin) navigates to the Task Force Management page.
2. The system displays a list of existing task forces and management options.
3. The Admin selects the “Add Task Force” option.
4. The system presents a creation form requesting information such as task force name, description, region, and initial members.
5. The Admin fills in the required details and selects initial members from the member directory (if applicable).
6. The Admin selects the “Create Task Force” button.
7. The system validates the provided information, including checking for a unique task force name.
8. The system stores the new task force record in the database.
9. The system updates the Task Force list and displays the new task force with a confirmation message.

Alternate Flow:

- Missing required fields trigger error messages.
- If name already exists, system prompts user to choose a unique name.

Postconditions: New task force is created and visible to administrators and members.

Comments: This function ensures scalable organization growth by allowing admins to create new collaborative groups as IAJES expands.

Figure 3.4: Use Case 4: Add Task Force

Use Case 5: Edit Task Force

Actor: Task Force Manager / Website Manager

Dependencies:

- Task Force database
- Admin role authentication
- Notification system for member updates (optional)

Description: An authorized administrator modifies an existing task force's name, description, or member list.

Preconditions:

- User is logged in with appropriate administrative rights.
- The task force already exists in the system.

Main Flow:

1. The Admin (Superadmin, Website Manager, or Task Force Admin) navigates to the Task Force Management page.
2. The system displays a list of existing task forces and management options.
3. The Admin selects the "Edit Task Force" option.
4. The system presents the current view of the website with form fields holding information such as task force name, description, region, and initial members.
5. The Admin edits details and deletes or adds members from the member directory (if applicable).
6. The Admin selects the "Save Changes" button.
7. The system validates the provided information.
8. The system updates the task force record in the database.
9. The system displays the updated task force page with a confirmation message.

Alternate Flow:

- Removing required members (e.g., the only admin) triggers a warning.
- Invalid or missing fields prompt correction before resubmission.

Postconditions: Edited task force details are saved and displayed.

Comments: Editing ensures accuracy and flexibility as membership and responsibilities evolve within IAJES task forces.

Figure 3.5: Use Case 5: Edit Task Force

3.7 Use Case Diagram

The use case diagram in Figure 3.6 provides a look at our early plan for the roles and permissions assigned to users of our website. While public users and registered members have access to most of the same functions, profiles and profile editing are only given to registered members. Admins have all the permissions of non-admins, along with additional permissions to edit various sections of the website.

Notably, our current site now has more granular permissions, where administrators can be given access to only their region or task force. Furthermore, we now include a validation system, and make a distinction between validated and invalidated accounts. Specifically, validated accounts will always have access to contact other users, while invalidated accounts can only contact a user if they have checked a certain option in their profile settings.

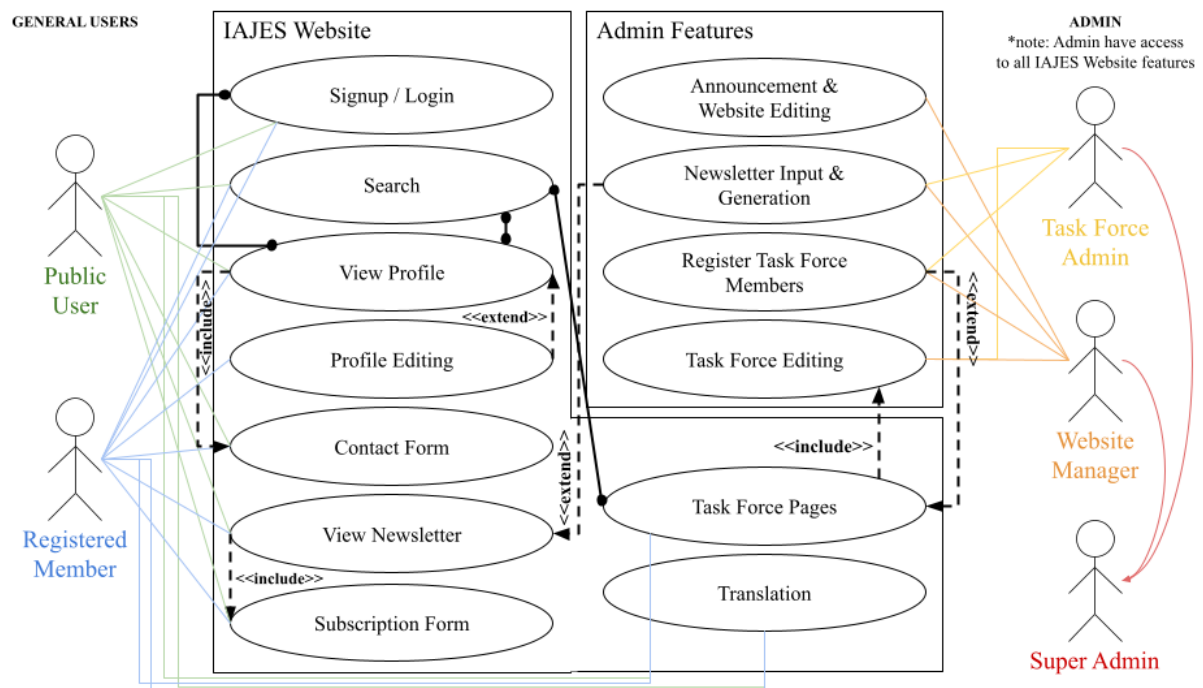


Figure 3.6: UML Use Case Diagram: IAJES Website Redesign

3.8 Class Diagram

The class diagram in Figure 3.7 provides a structural overview of the IAJES website redesign by modeling the main system entities, their attributes, and how they interact. User information is captured through the **Profile** class, which stores key fields such as name, school, major, job position, task force affiliation, and research interests. Role-based access control is represented using **RoleType**, **Role**, and **Permission**, enabling different platform capabilities for admins, task force managers, members, and general users. Site content is organized under the general **Content** class and specialized into informational page types such as **Conference**, **Contest**, **RegionalMeeting**, **BiennialMeeting**, and **Activity**, supporting consistent data handling across the site. Publishing and communication features are modeled through **Newsletter** and **Article**, including subscription and authorship relationships. Together, these classes and associations ensure the system supports membership management, content publishing, and structured browsing and search functionality.

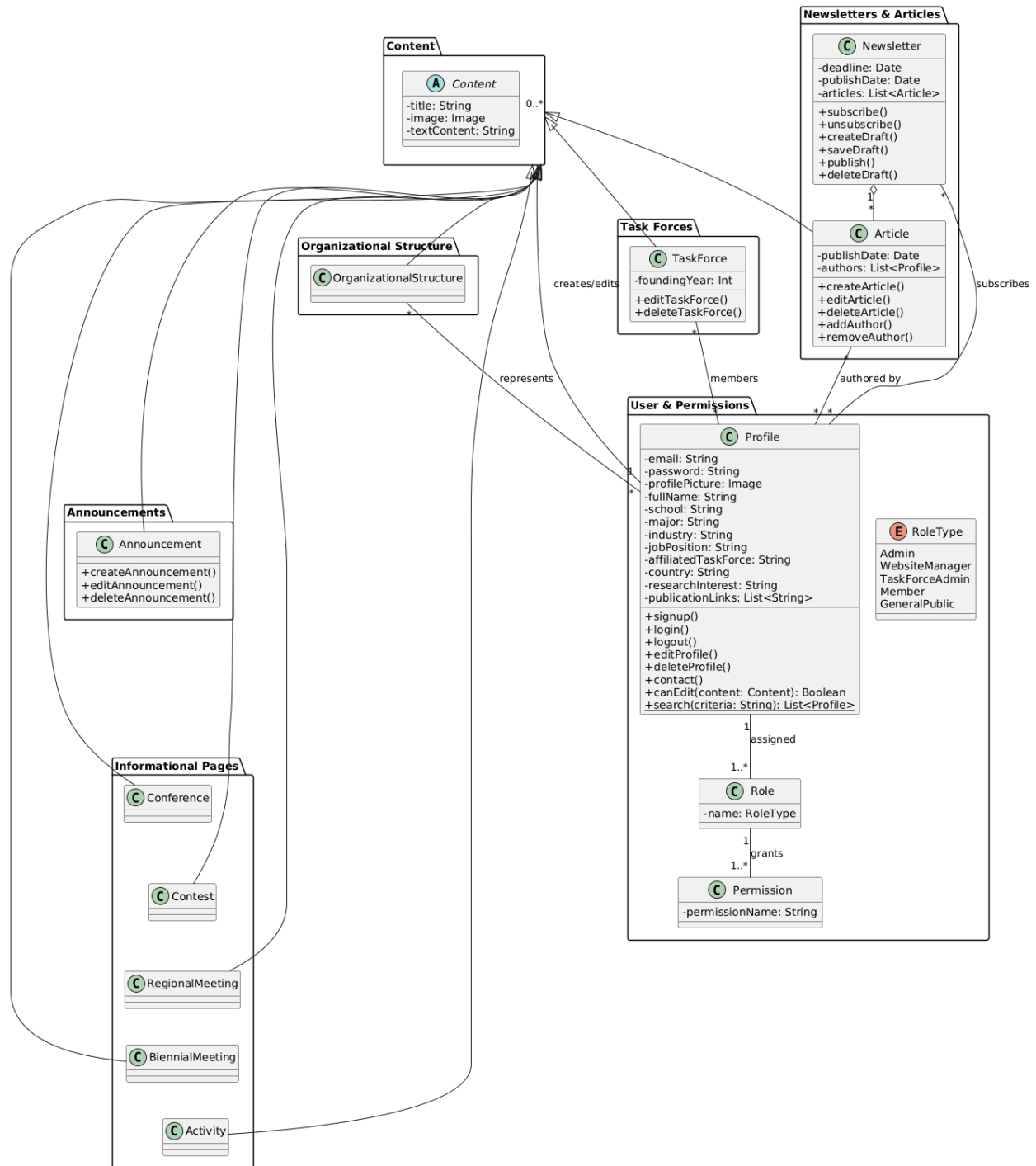


Figure 3.7: Class Diagram of the IAJES Website

3.9 Sequence Diagram

The sequence diagram in Figure 3.8 illustrates the end-to-end account creation process and verification logic as interactions between the user interface and backend services. The user first submits account information (email, pass-

word, and profile details), after which the application creates authentication credentials using Firebase Authentication. Once the account is created, a corresponding **Profile** record is stored in Firebase Firestore with an initial tag of **unverified** to represent membership status. The diagram also includes an optional (**alt**) path where the user can subscribe to the IAJES newsletter, which triggers a backend action to add the email to a mailing list. Verification logic then checks whether the user email belongs to the IAJES domain: if it does, the system updates the tag to **verified** and confirms successful membership; otherwise, the user is notified as ineligible and the system deletes both the authentication record and the associated Firestore profile entry to maintain valid membership constraints.

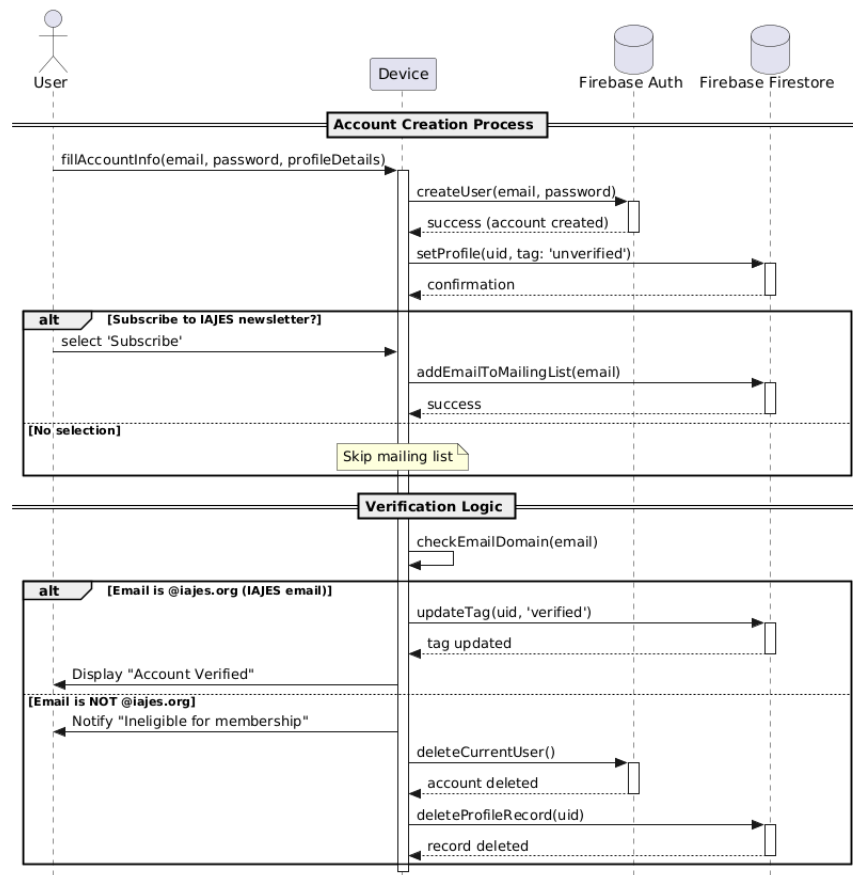


Figure 3.8: Sequence Diagram: Account Creation and Verification Logic

3.10 Activity Diagram

The activity diagram in Figure 3.9 describes the control flow of account creation, optional newsletter signup, and membership verification from a high-level behavioral perspective. The process begins when the user enters account information and includes a validation loop to ensure inputs are correct before proceeding. After successful validation, the system creates the account and assigns an **unverified** tag to the new profile, ensuring membership status is tracked consistently. The diagram then branches based on whether the user opts to subscribe to the newsletter: selecting “yes” adds the account email to the newsletter mailing list, while selecting “no” skips this step and continues. Finally, the system evaluates whether the account email matches the IAJES domain. If the email is valid for IAJES membership, the system removes the **unverified** tag to mark the account as verified; if the email is not eligible, the system notifies the user and deletes the account. This activity flow matches the verification and cleanup logic shown in the sequence diagram and ensures only qualified accounts remain active.

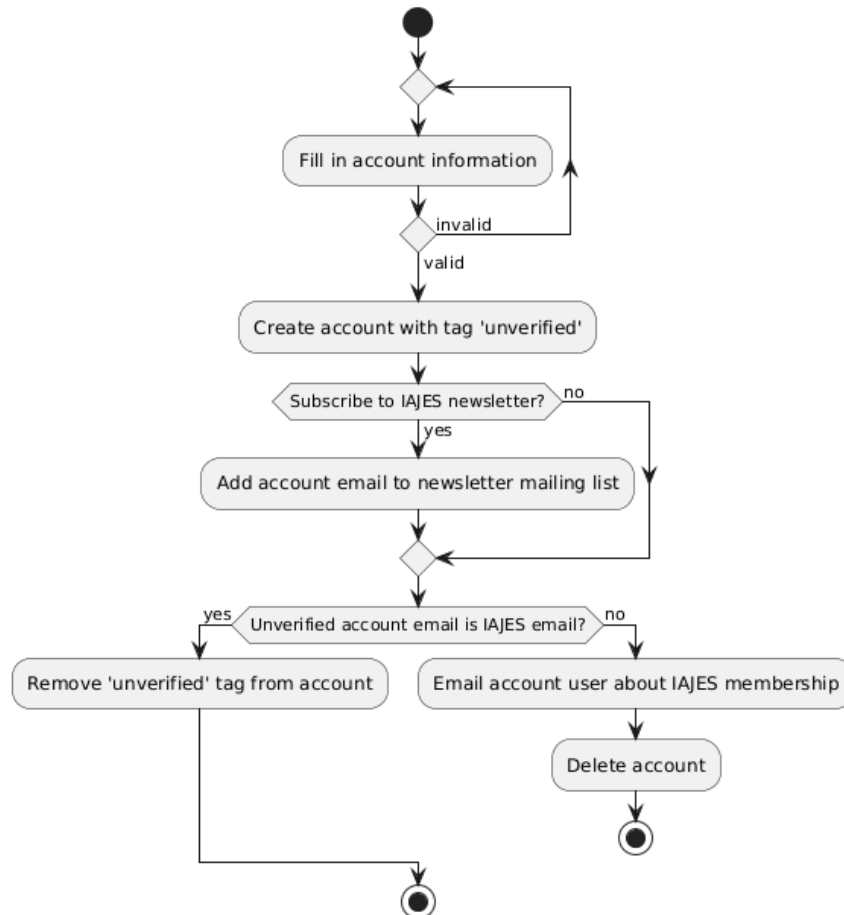


Figure 3.9: Activity diagram for account creation functionality

Chapter 4

Design Overview

4.1 System Components

Our system is built using a simple, modern tech stack chosen for scalability, maintainability, and quick development. The front-end uses React operating in framework mode, with React Router to allow fast page navigation and modular component design. Styling is done through a combination of Tailwind CSS and custom CSS, which allows for both quick styling changes and more custom elements. JavaScript is used to manage dynamic interaction with the database. For this back-end database, we used Supabase, which provided a PostgreSQL database and real-time API services that are integrated with the front-end.

During development and demonstration, we deployed the application using Firebase App Hosting and Google Cloud, providing a public access link that easily updated and scaled with us during development. For the final deployment phase, the system will be migrated to IAJES servers and containerized using Docker to allow the website to sit within IAJES' own system, allowing for easier maintenance and scalability.

Project development and collaboration tools used included: GitHub for version control, Figma for interface design and prototyping, and Google Drive and Google Docs for documentation and team coordination.

4.2 General Design

When developing this website, the user experience and design choices were especially important for us. This is why our team chose to create extensive hi-fi prototypes detailing almost every proposed page of our website. By developing in grayscale, we were able to focus solely on the content and layout of each page. This approach also allowed us to rapidly make changes without worrying about adjusting the colors each time. Also, by using Figma, we were able to transform our pages into interactive prototypes, which was especially useful for showcasing to IAJES and getting feedback on the user experience of our site. The ability to use components also allowed many designs to be made

modular and adapt responsively.

Later on, we defined the color scheme, shown in Figure 4.1. We drew inspiration from IAJES' current logo, using their existing colors in order to keep a consistent brand identity that we could expand upon. The cool colors also helped to provide a calming and professional tone to the website, ensuring users feel comfortable during their interaction with the platform (8). Using Figma, we also developed multiple color variations on existing prototype pages, to receive feedback on the desired color scheme and application.

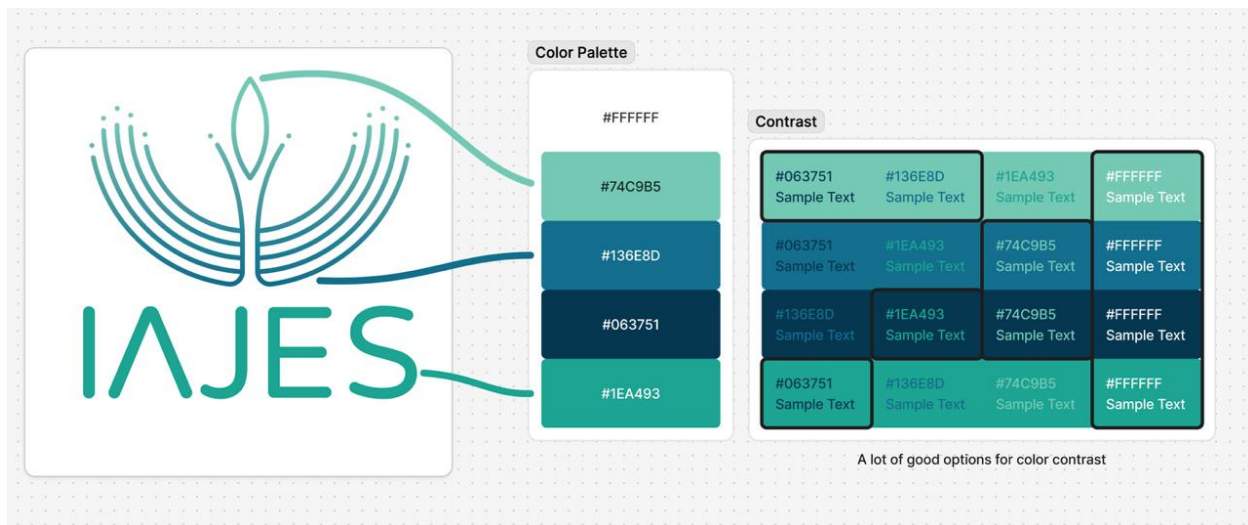


Figure 4.1: Color Scheme

Alongside the colors, our team aimed to create a professional feel by using clean, minimalist design. using clear size and color hierarchy to direct the user's attention to the most important elements. We also sought to feature much of the same content as found on the original website, so dividing up information into understandable sections with engaging images was also a major priority—helping to keep the user engaged for longer.

4.3 Prototypes

In this section, we show our initial prototype designs.

4.3.1 Sign-up, Login, and Logout

The Sign-up, Login, and Logout pages, shown in Figures 4.2 through 4.4, consist of simple forms that are standard in many webpage login systems. We kept this standard flow in order to keep the familiarity users have with these systems. Adding a version of the IAJES logo to the logout page helped to add visual interest and build the brand across the whole website.

IAJES

Search

Create your IAJES account

First name

Last name

Email

Password

☐

 I agree to the Terms and Privacy Policy

☐

 Subscribe to IAJES Weekly

Create account

Already have an account? [Sign in](#)

(Footer)

Figure 4.2: Sign-up



Sign in to IAJES

Email

Password

☐ Remember me

Sign In

New here? [Create account](#)

(Footer)

Figure 4.3: Login

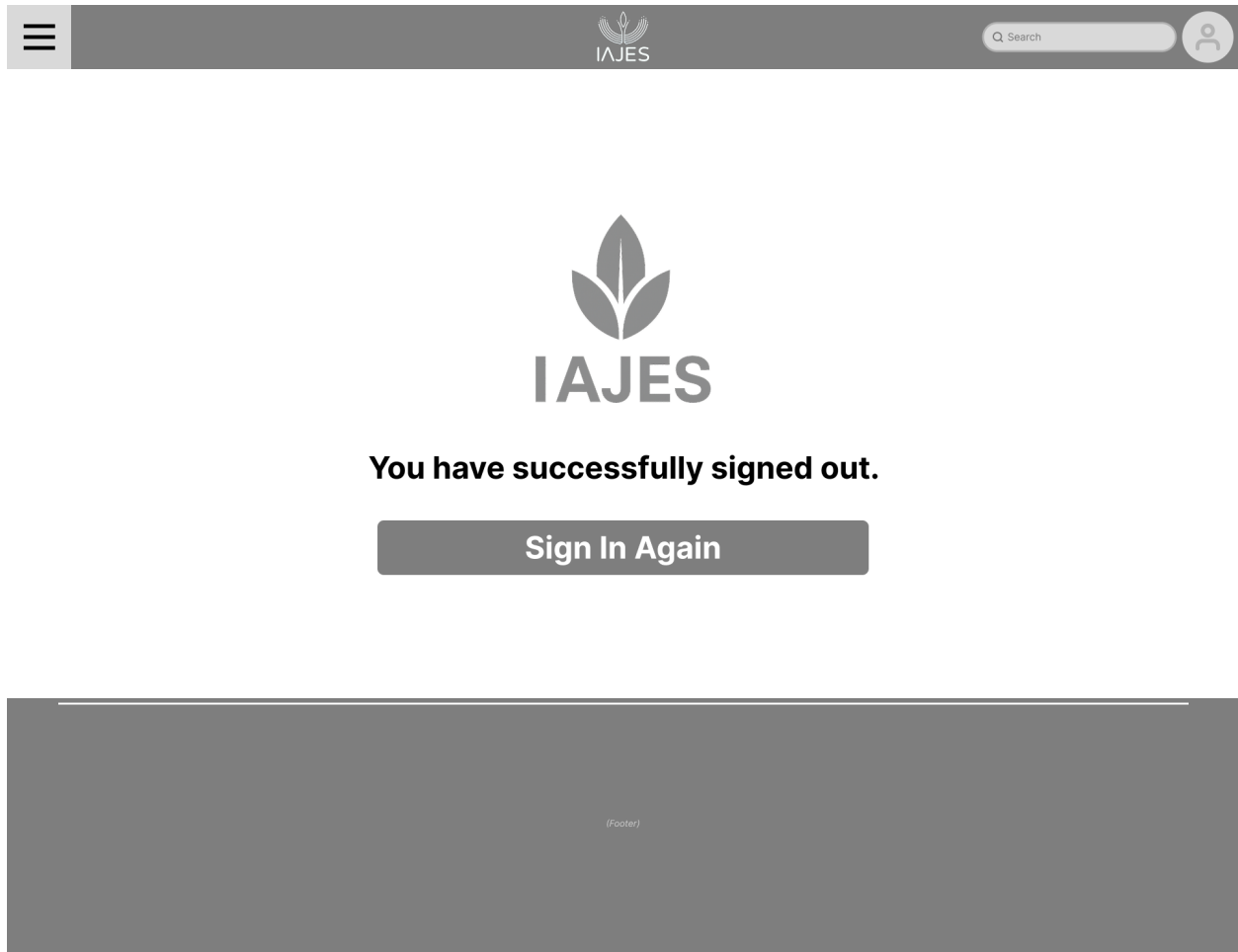


Figure 4.4: Logout

4.3.2 University, Page, and Profile Search

The search pages, shown with different results for universities, profiles, and pages in Figures 4.5 through 4.7, show how information is presented after searching. Based on the search terms and filters, the appropriate results are presented modularly in different layouts. Each result is presented simply in a card, holding all important information at a glance and making it quick to scan through. Each card is displayed differently based on its type. For example, university cards show a large picture of the university, the name, location, and description of the university, and the associated members before showing a link to the official site (as the universities themselves were not planned to have a page on the IAJES website). This is different from cards showing profiles, which have a smaller profile picture, their name, contact, and demographic information, before showing the affiliated task force and a large contact button. By emphasizing the contact button, we incentivize users to engage with others on the site, contributing to IAJES' goal of connectivity.

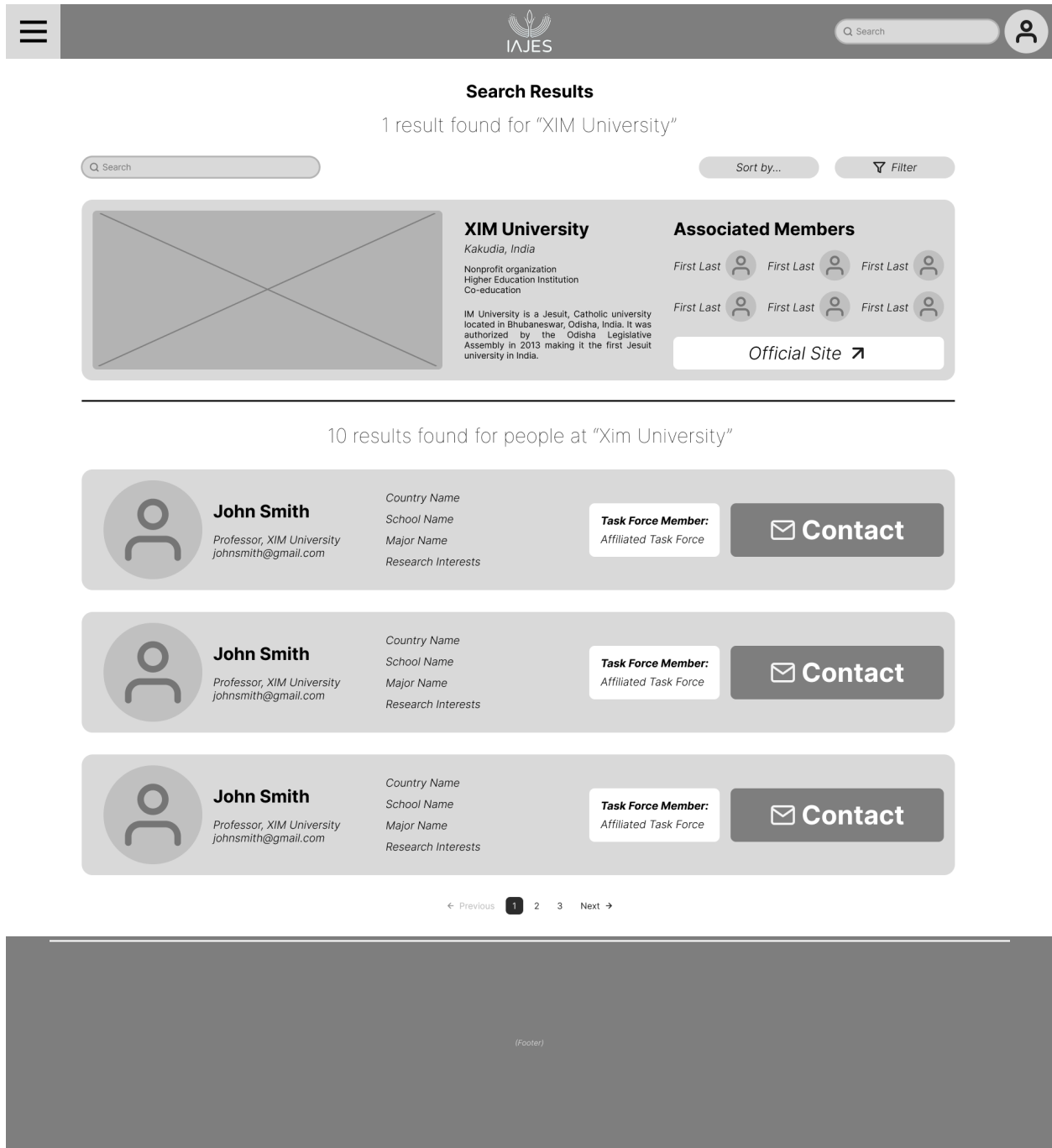


Figure 4.5: University Search Results

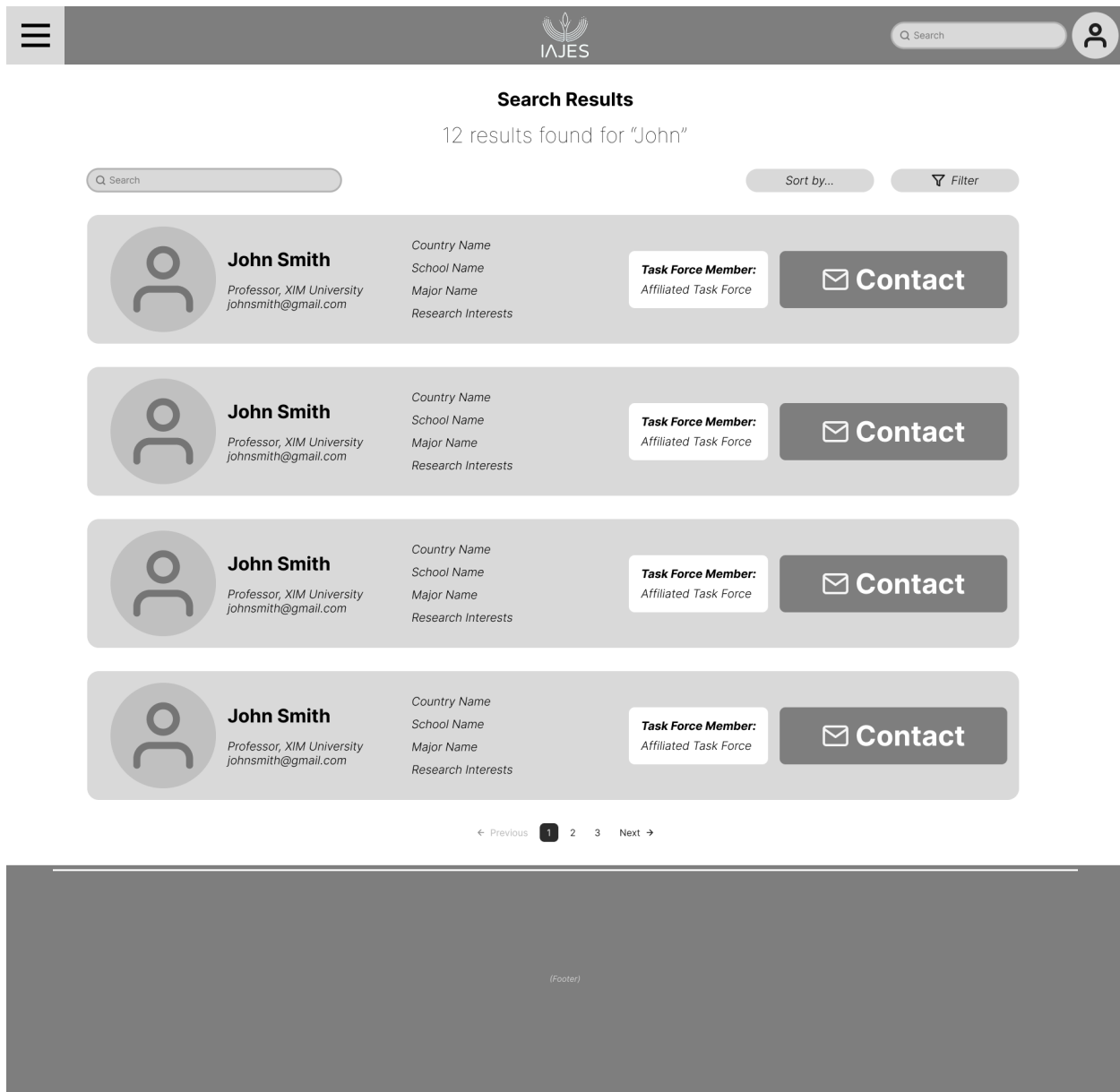


Figure 4.6: Profile Search Results

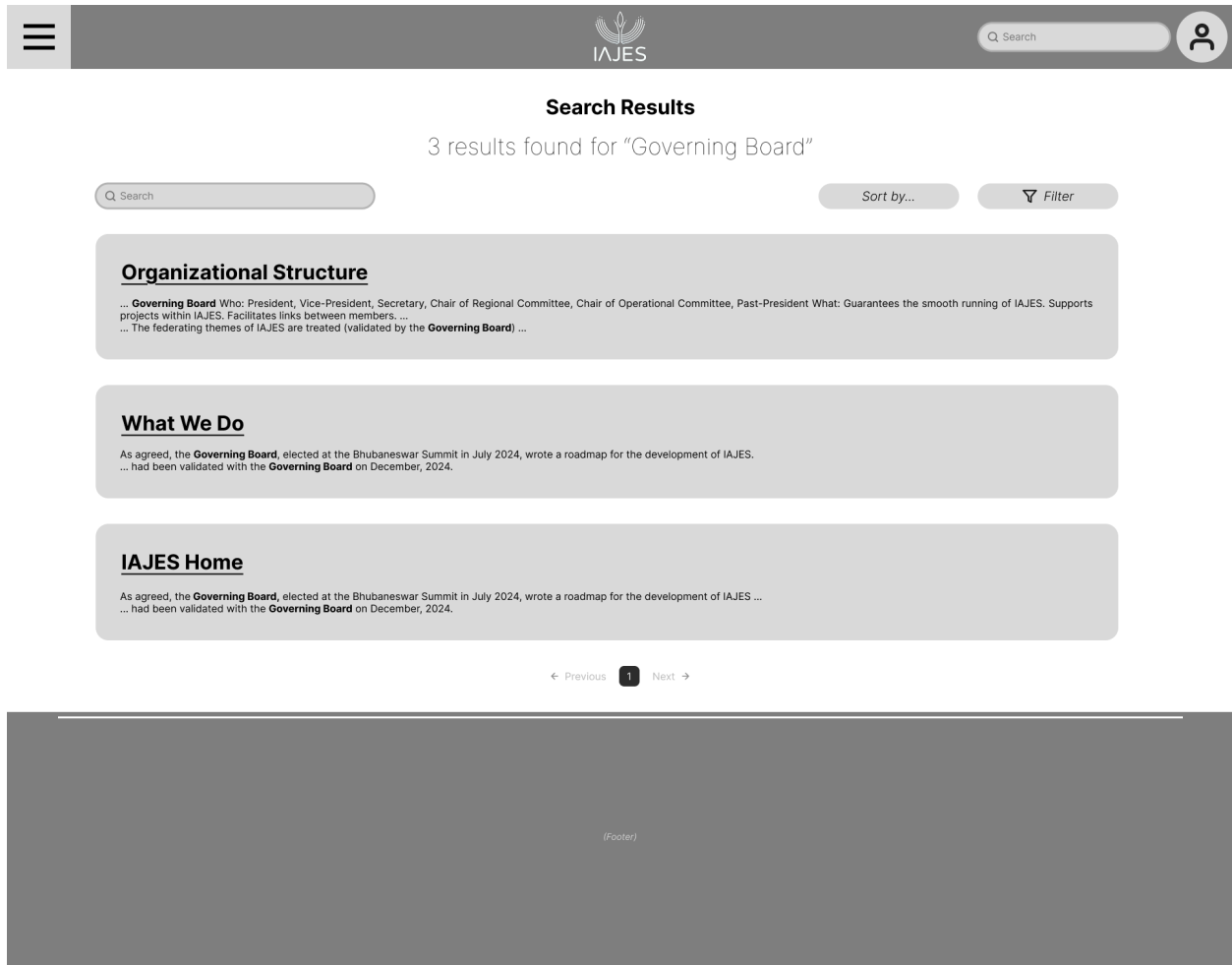


Figure 4.7: Page Search Results

4.3.3 Profiles

The Profile page, with different views for viewing a profile, editing a profile, and showing settings, are shown in Figures 4.8 through 4.10. Similar to profile cards in the search view, the most prominent information is shown in bold colors and larger fonts, such as the contact button, profile picture, and social media links. By clicking the edit button in the top right, a user can edit their own profile. When editing your own profile, we kept the profile page's structure, while transforming all details into editable fields. This allows the user to stay familiar with how their profile might look after exiting edit mode. The use of input text fields, file uploads, and modular link cards makes editing simple and easy to understand, drawing on the familiarity IAJES users have from Google Sites and Google Forms. For more general profile settings, a settings view was created. Creation of this page was especially helpful in determining what a given user could customize, and how it would be laid out.

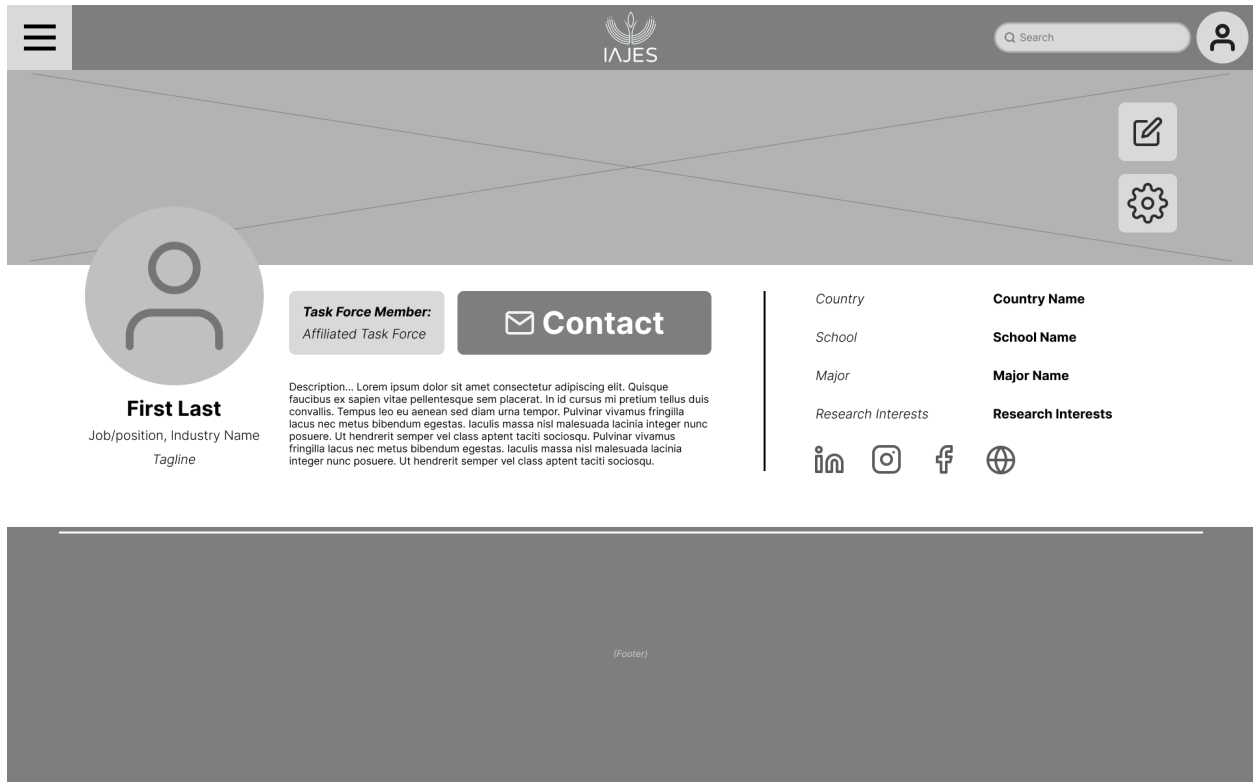


Figure 4.8: Profile

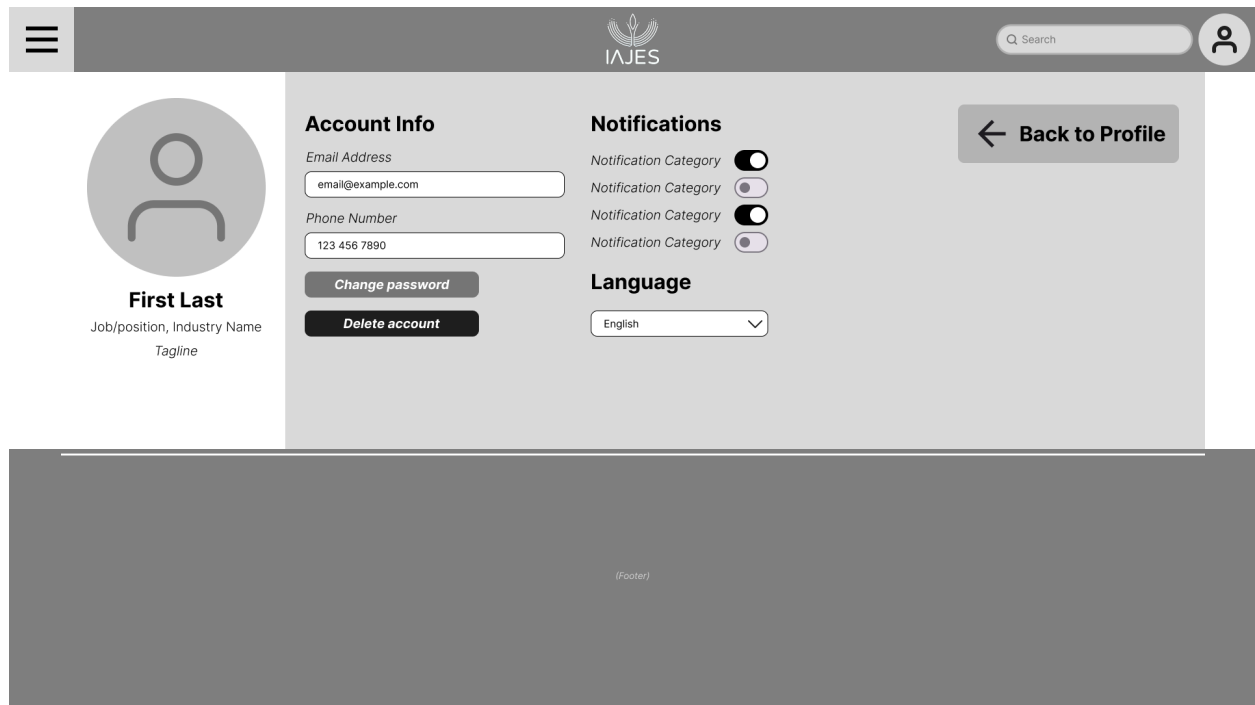


Figure 4.10: Profile (Settings View)

4.3.4 Landing

The Landing page, shown in Figure 4.11, was the first page created for the website. This view is essential for guiding new users and helping them get acquainted with IAJES and what information they can expect to find on the website. Although long, the sections were made to be modular, simple, and visually appealing, allowing the user to feel comfortable upon their first visit. Unlike other prototype pages, we used real images from the IAJES website to gauge how they would look in the final version.

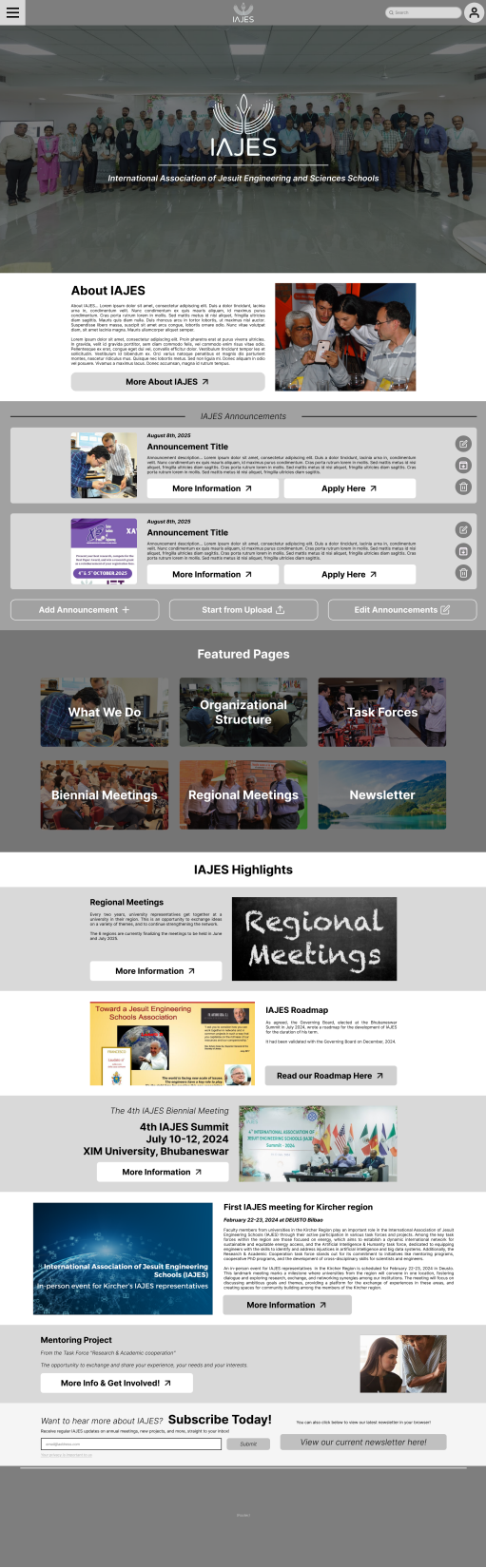


Figure 4.11: Landing
33

4.3.5 About Us and Organizational Structure

The About Us and Organizational Structure pages, shown in Figures 4.12 and 4.14, are two pages that show more detailed information about the organization. Similarly to the landing page, information is structured in modular cards and sections alongside relevant images and diagrams. For example, the use of an interactive map allows more interactivity to navigate more complex information. The infographic diagram showcasing the structure of IAJES also presents complex information in a readable way. Most information on pages such as these were taken directly from IAJES' original website, but presented in a minimal yet engaging manner to provide a fresh new experience for users. Furthermore, the announcements section on About Us is designed for editability. Figure 4.13 shows the layout of the edit popup, which was immensely helpful for determining what the announcements section would need, and acted as a guide for the editing popups of other pages as well.

About IAJES... Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis a dolor tincidunt, laoreet urna in, condimentum velit. Nunc condimentum ex quis mauris aliquam, id maecenas purus condimentum. Cras porta rutrum lorem in moris. Sed mattis metus id nisi aliquet, fringilla ultricies diam sagittis. Mauris quis diam nulla. Duis rhoncus arcu in tortor lobortis, ut maximus nisl auctor. Suspendisse libero moris, suscipit sit amet arcu congue, lobortis ornare odio. Nunc vitae volutpat diam, sit amet lacrima magna. Mauris ullamcorper aliquet semper.

Nunc rutrum sem eget risi varius convallis. Aenean eget ex lobortis, hendrerit orci et, ultricies dolor. Pellentesque pellentesque enim mauris, in tempor justo molestie tempor. Curabitur viverra sit amet, aliquam vel venenatis. Donec suscipit nisi lacus, ac pulvinar felis varius sit amet. Praesent quis lacus leo. Aliquam feugiat ipsum et magna pulvinar dignissim. Suspendisse vulputate nibh ac risi. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis a dolor tincidunt, lacinia urna in, condimentum velit. Nunc condimentum ex quis mauris aliquam.

Id maximus purus condimentum. Cras porta rutrum lorem in mollis. Sed mattis metus id risi aliquet, fringilla ut tristique diam sagittis. Mauris quis diam nulla. Duis rhoncus arcu in torbor lobortis, ut maximus nisi auctor. Suspendisse libero massa, suscipit sit amet arcu congue, lobortis ornare odio. Nunc vitae volutpat diam, sit amet lacina magna.

The diagram consists of four rectangular boxes arranged in a 2x2 grid. Each box has a light gray background and a dark gray border. The boxes are labeled as follows:

- Top-left box: **Task Forces**
- Top-right box: **Regional Meetings**
- Bottom-left box: **Webinars**
- Bottom-right box: **Projects**

To join the engineering schools in Jesuit Universities worldwide, in order to promote the development and growth of our society through academic activities, research and social action.

To seek a more humane and just world through the use of technology and innovation under the guiding principles of the Society of Jesus.

- To know, to share and to learn from positive experiences in education, research, innovation, and service within our universities, for the purpose of forming engineers inspired by the Jesuit tradition.
- To promote the care of the whole person in all our interactions and experiences.
- To propose and work on interdisciplinary and multi-campus projects addressing the needs of marginalized individuals and communities, using the knowledge of science and engineering.
- To sustain a worldwide community of engineering educators, which is driven by a living sense of fellowship, belonging and solidarity.

Make IAJES a worldwide leader and reference organization in the promotion of a new generation of engineers, scientists, and researchers deeply rooted and trained in the spirit of the Jesuit mission and Universal Apostolic Preferences.

(Map Embed)

General, Arturo Sosa SJ, at the assembly in Boston, August 2022

La Asociación Internacional de Universidades Jesuitas encuentra su razón de ser y da sentido a lo que hace al vivir e impulsar la colaboración y la solidaridad dentro y desde las instituciones universitarias que la conforman. Se trata de aprovechar lo mejor posible el enorme potencial de colaboración y solidaridad existente en las universidades que conforman esta red. Poco a poco hemos ido emprendiendo esa ruta. A medida que avanzamos reconocemos las ventajas de la colaboración y la solidaridad. Vamos aprendiendo mejores formas de aprovechar los recursos que

Traducción española del discurso del P. Sosa

"The International Association of Jesuit Universities finds its purpose and gives meaning to what it does by living and promoting collaboration and solidarity within and from the university institutions that make it up. It is a matter of making the best possible use of the enormous potential for collaboration and solidarity that exists in the universities that make up this network. Little by little we have been embarking on this path. As we move forward we recognize the advantages of collaboration and solidarity. We are learning better ways to take advantage of the resources we

English Translation of Fr. Sosa's Speech

As agreed, the Governing Board, elected at the *Bhubaneswar Surririki* in July 2024, wrote a roadmap for the development of IJES. This road map had to find the right balance to allow the development of the association, the realization of the actions, the right rhythm of the events, the necessary involvement of the members, and the close relationship with IAUJ. It had been validated with the Governing Board on December, 2024.

[View the IAJES 2025-2026 Roadmap ↗](#)

The Origin of IAJES

2015



Our initiative is both a response to a call from Pope Francis and Society of Jesus, especially through the assembly for higher education, Fr. Michael Gervais and the expression of a shared strong desire to foster fruitful links of communion and cooperation among our Jesuit engineering schools.

The concept was launched in July 2015 in Melbourne during the Assembly of Presidents and Rectors of Jesuit universities.

2017 / 2018

A brief prospectus in the fall 2017 led to the constitution of the first task force with worldwide representatives. Its main task was to prepare and promote a first founding conference for the IALES International Association of Jesuit Engineering Schools.

Foreseen in **DEUSTO BILBAO in July 2018**, this event would find a meaningful place just before the Assembly of Presidents and Rectors of Jesuit Universities where the IALES International Association of Jesuit Universities would solemnly be founded.

Figure 4.12: About Us



Figure 4.13: About Us Announcement Editing Popup

4.3.6 Biennial Meetings (International Meetings) and Regional Meetings

In Figures 4.15 and 4.16, the Biennial Meetings (later renamed International Meetings) hub and detail views are outlined. Figures 4.17 through 4.19 show the region hubs, regional meetings for a given region, and the detailed view for each meeting. These pages both store nested information and make it easy for users to find what they are looking for quickly. Furthermore, these pages are customized to the needs of their content. For example, the gallery for both regional and international meetings was desired to allow a user to see standout moments of community, celebration, and research taking place at these IAJES events. An edit popup is also shown in Figure 4.20. Since regional meetings happen more frequently than international meetings, we thought it pertinent to allow easy addition, editing, and deletion of regional meetings for administrators. Due to this editability, all content in these meeting detail views was planned to also be modular and adaptable to display the content provided—leaving no empty sections. The decision to include these modular resources, such as meetings and the following set of pages, were defined by the categories present on IAJES' original website.

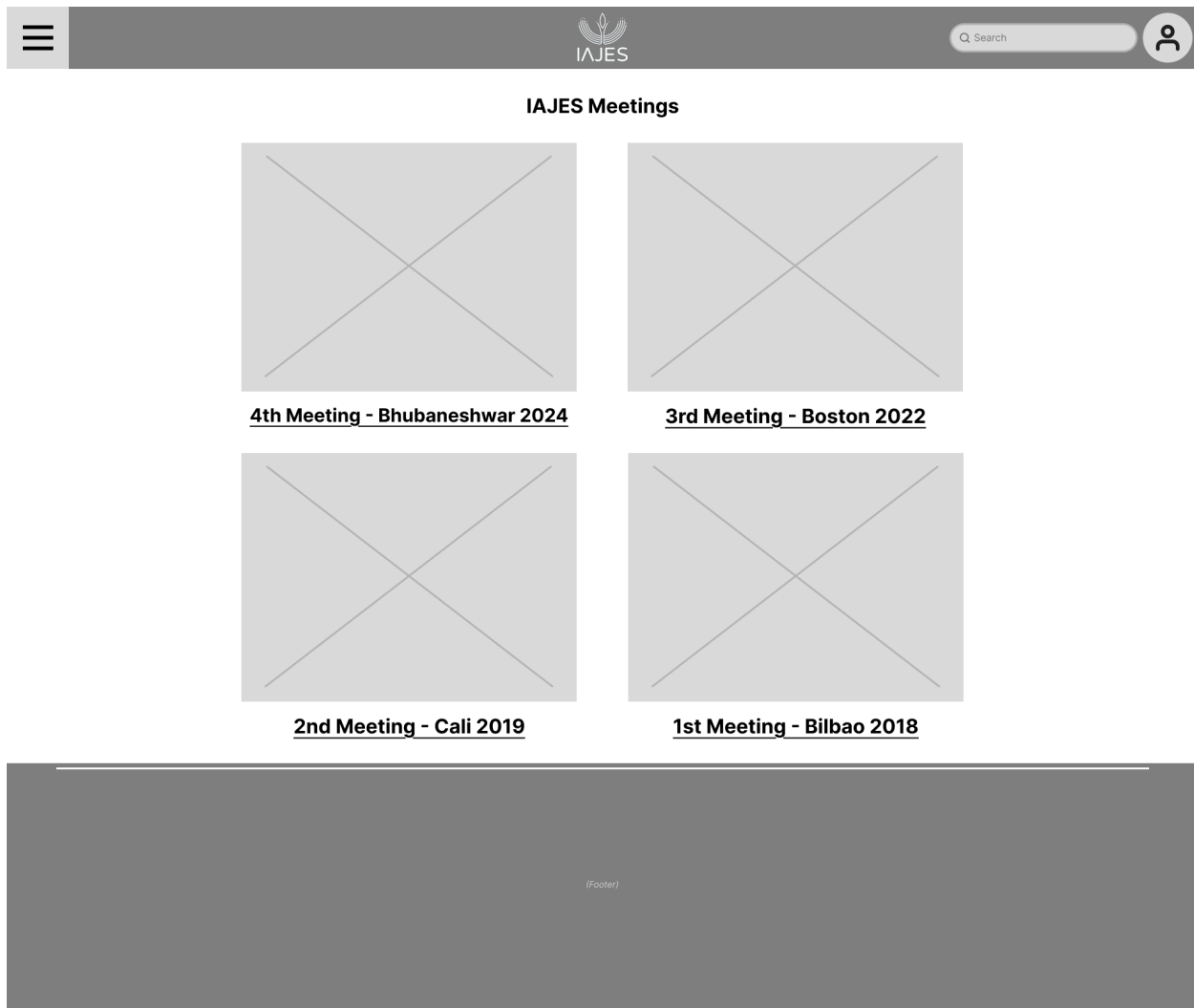


Figure 4.15: Biennial Meetings (Meetings List View)

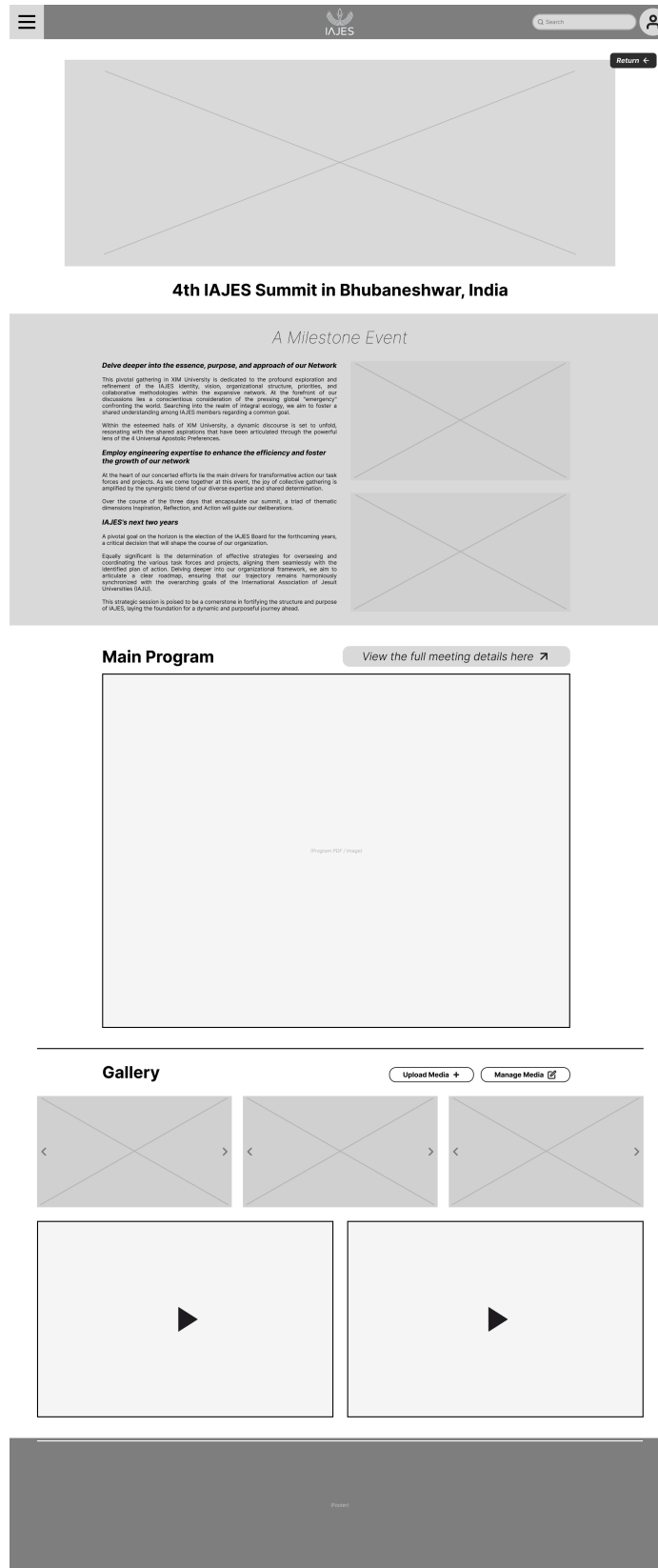


Figure 4.16: Biennial Meetings (Detail View)

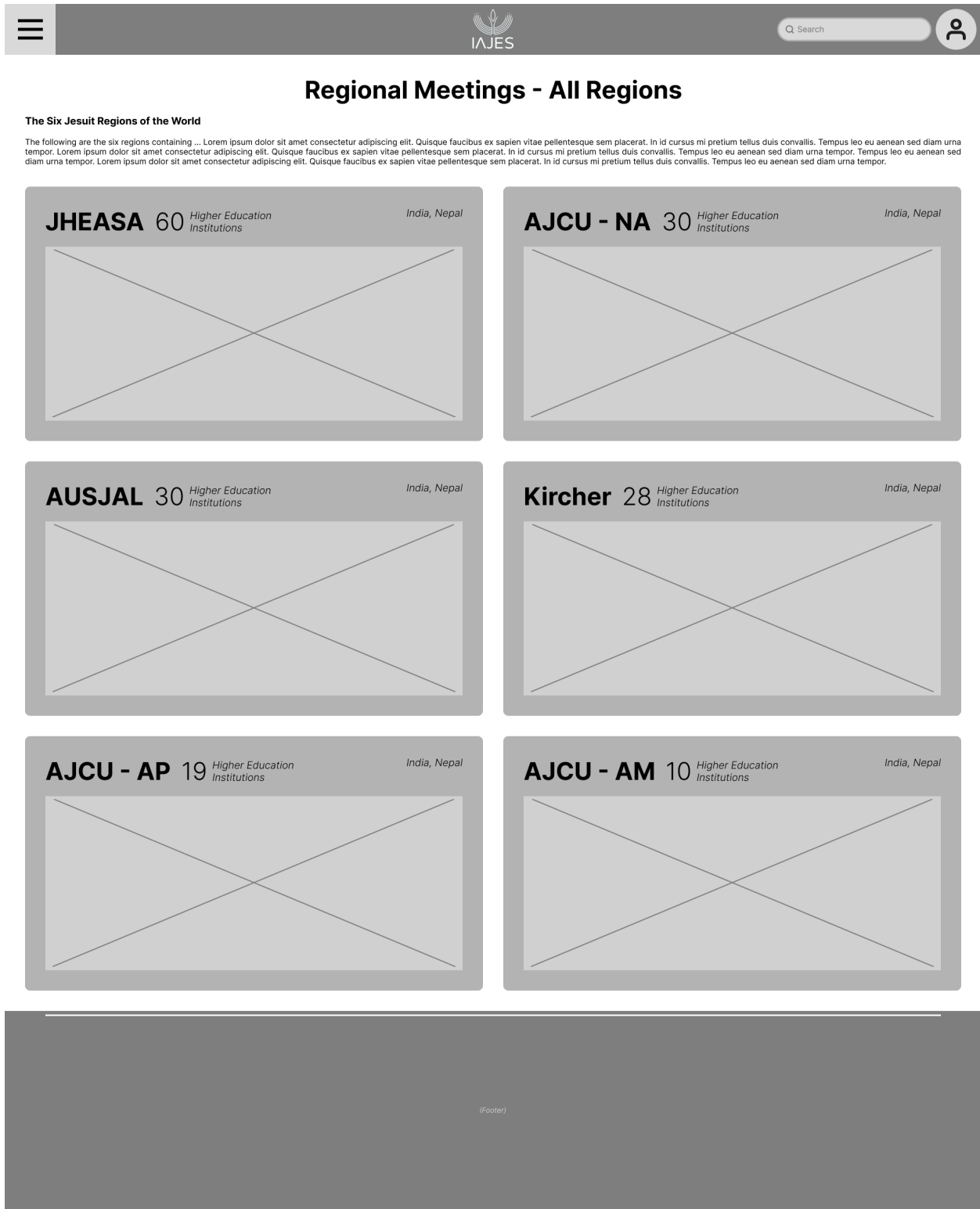
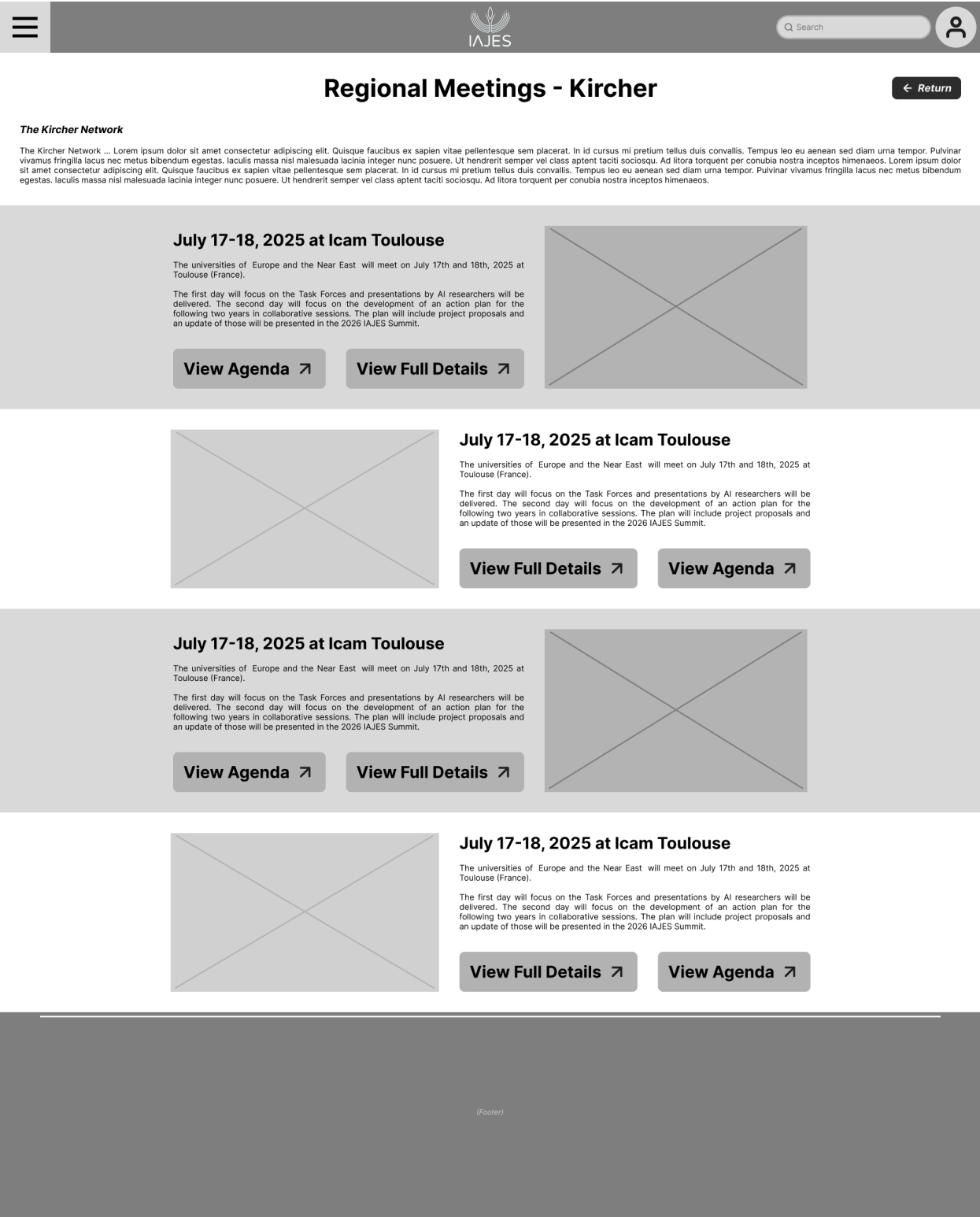


Figure 4.17: Regional Meetings (Region List View)



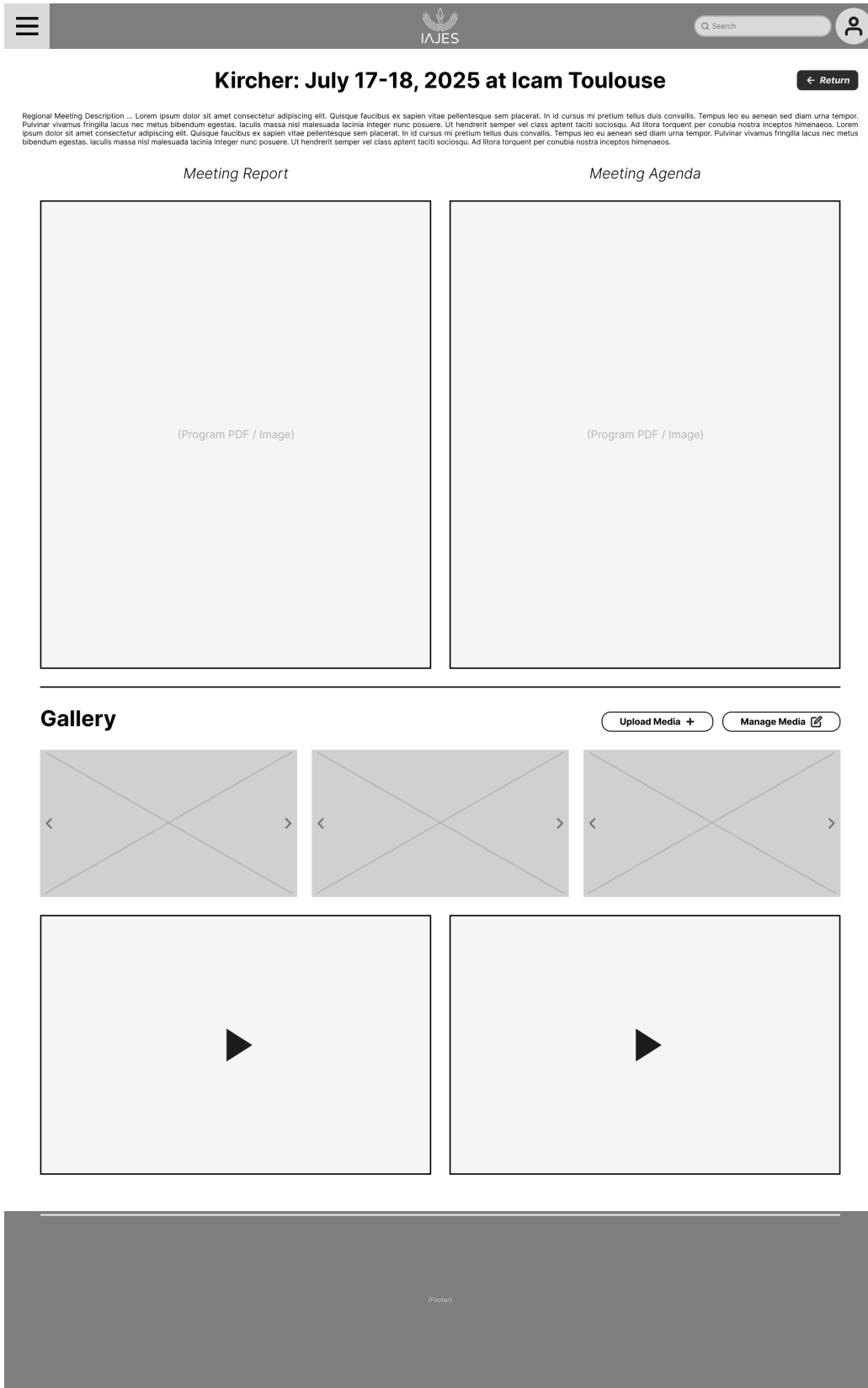


Figure 4.19: Regional Meetings (Detail View)

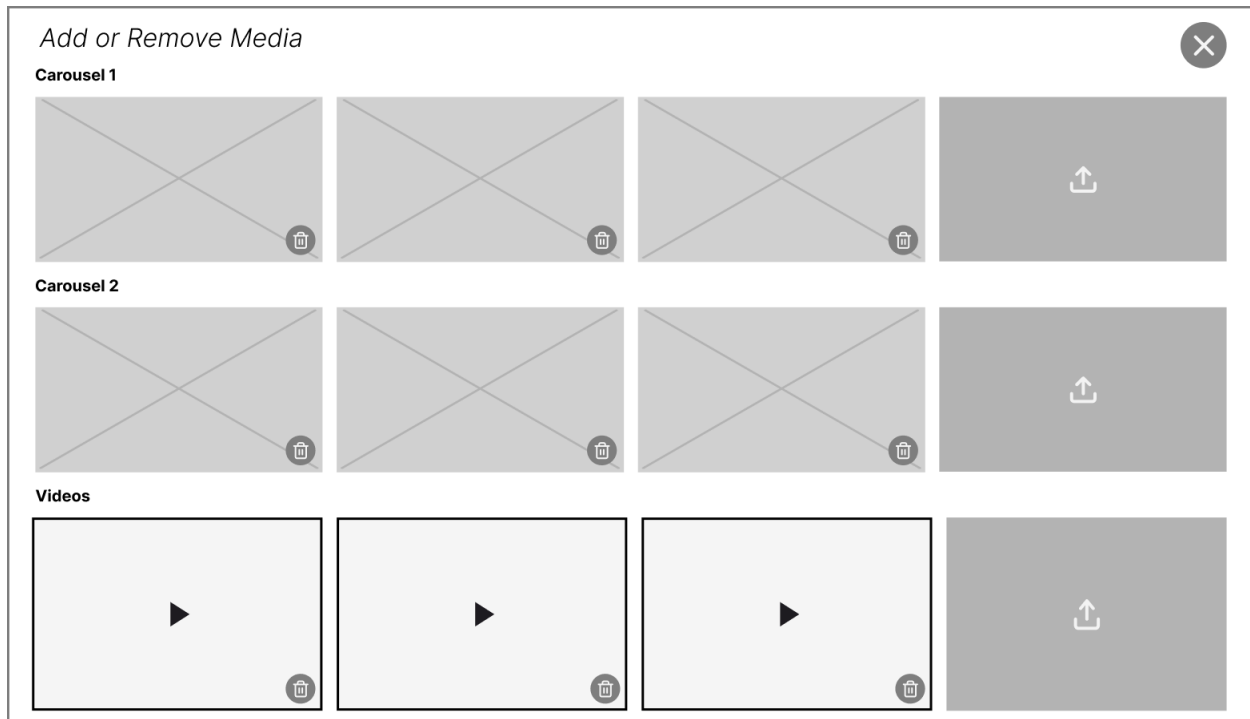


Figure 4.20: Regional Meeting Gallery Edit Popup

4.3.7 Webinars and Projects

In Figures 4.21 and 4.22, the webinars hub page and detail view are presented. Figure 4.23 shows a Project detail page, and although not prototyped, a similar hub page to Webinars was originally planned. Like regional meetings, both pages were planned to host editable content and were therefore designed modularly and customized to their set of relevant information.

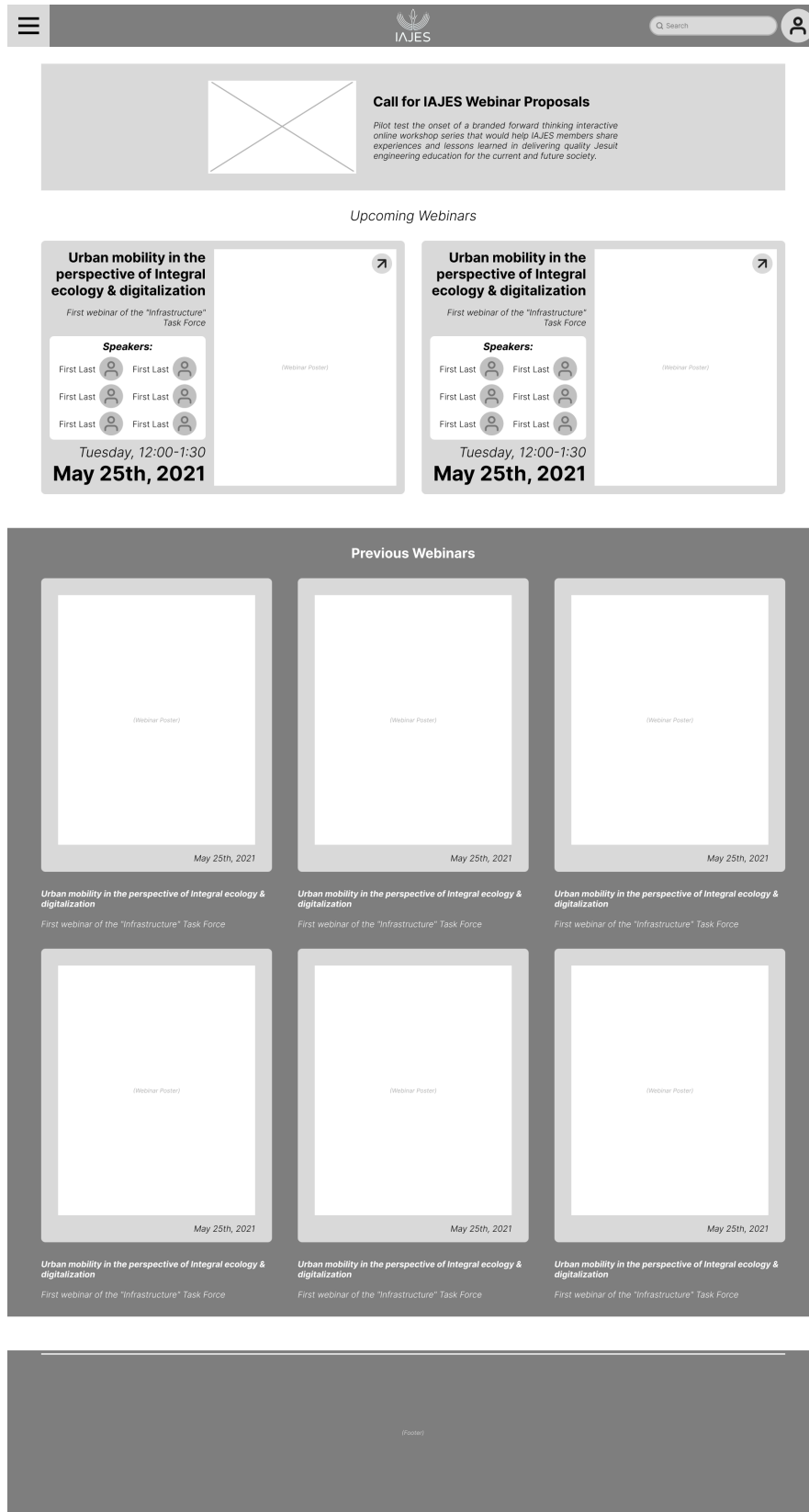


Figure 4.21: Webinars (List View)

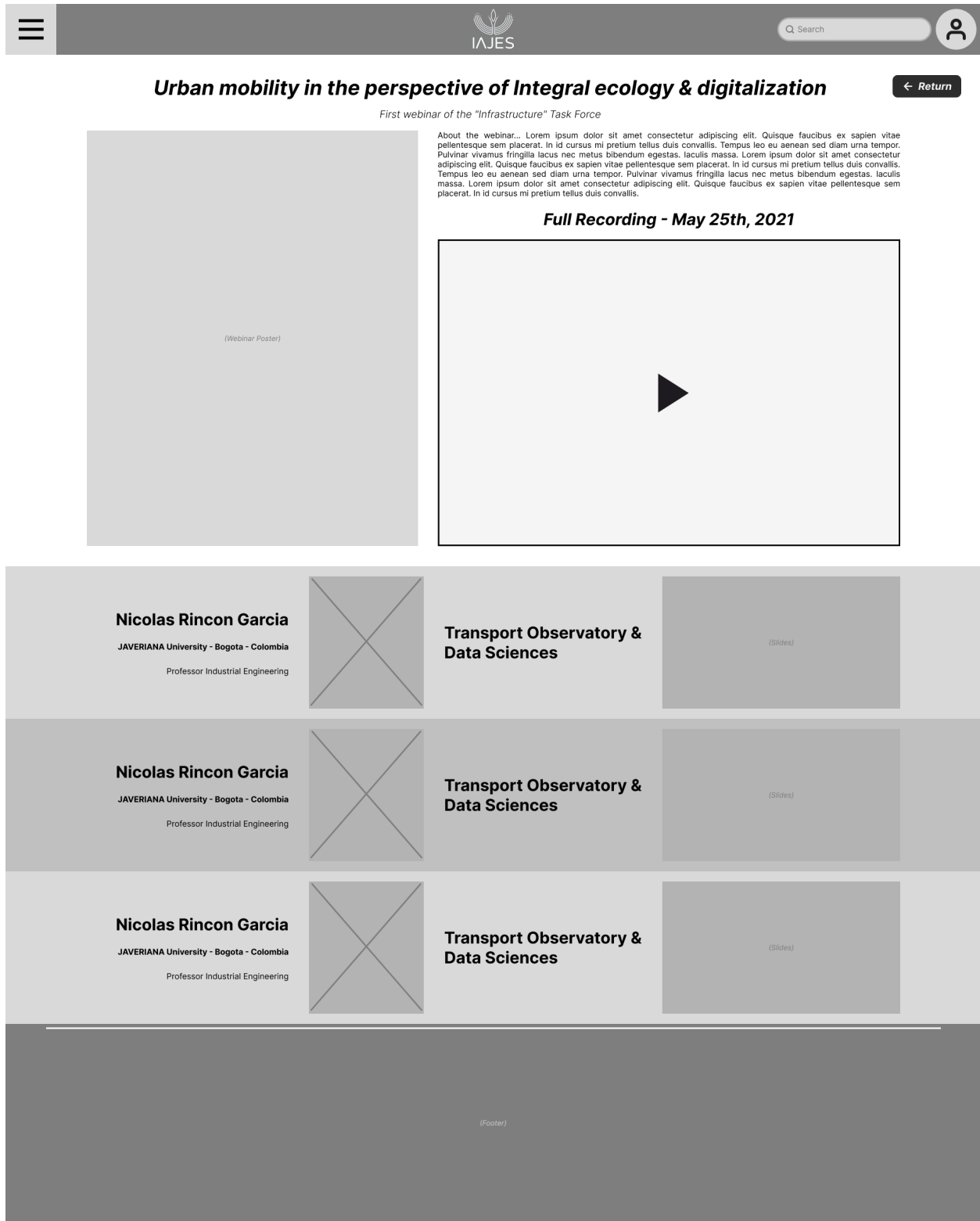


Figure 4.22: Webinars (Detail View)

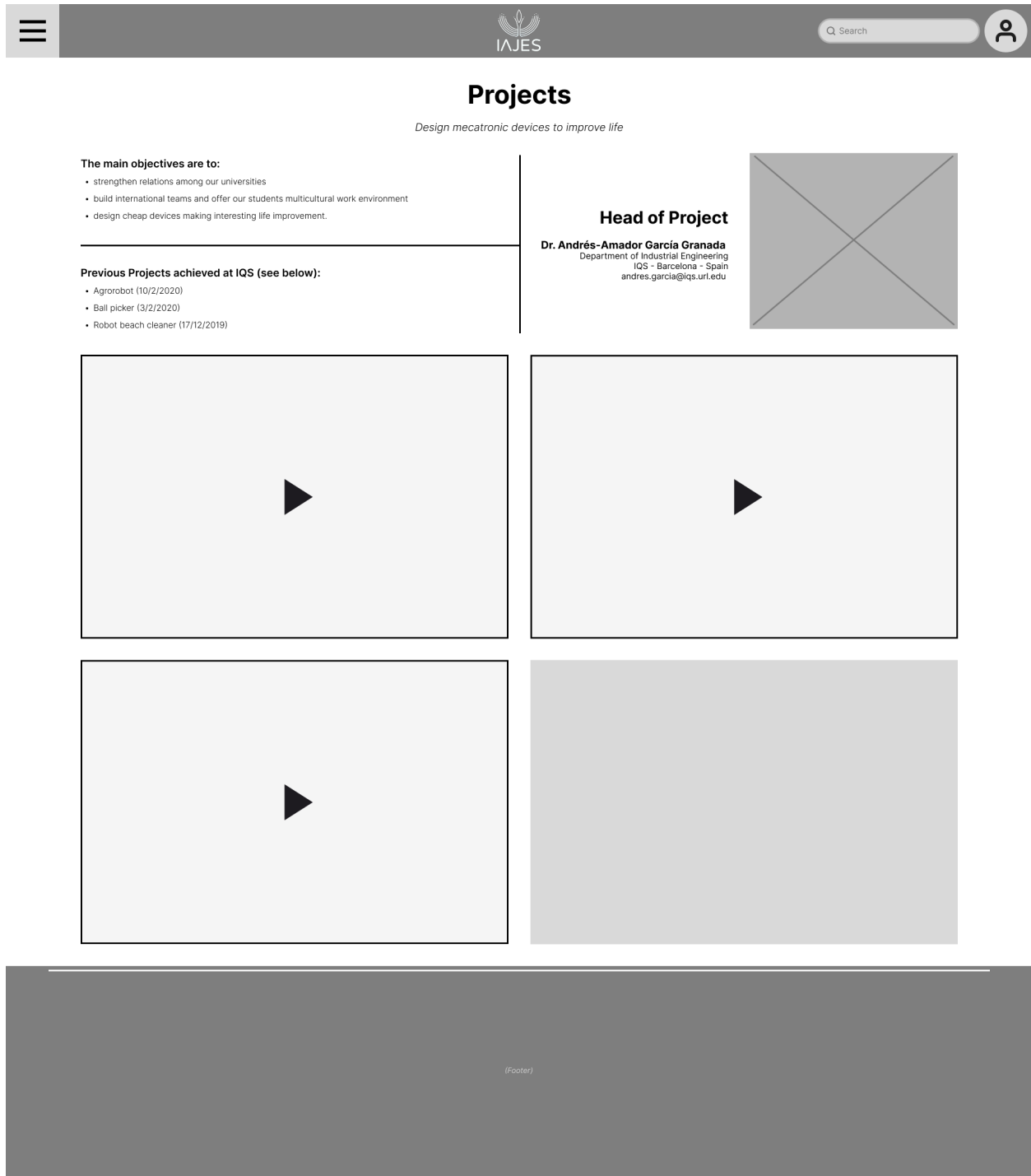


Figure 4.23: Projects (Detail View)

4.3.8 Participatory Activities, Conferences, Contests, and Research

Figures 4.24 through 4.27 show pages that were made to hold various types of modular resources. These resources include IAJES activities, conferences, contests, and research. These pages were made to be editable, modular, and hold resources for long-term storage and reference. As especially important to IAJES, we made sure that all relevant information was made clear and would be collected during the creation process.



Figure 4.24: Participatory Activities

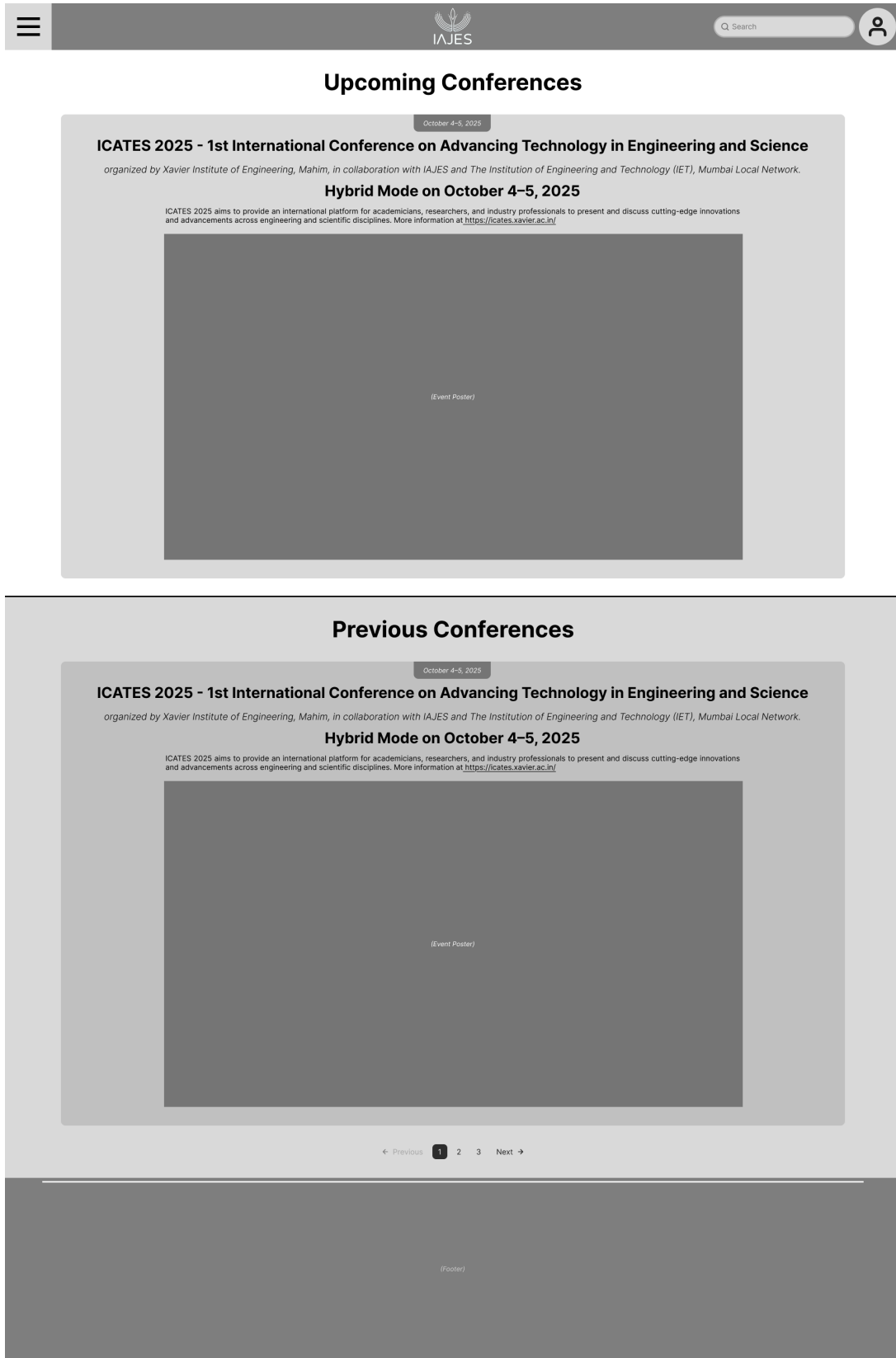


Figure 4.25: Conferences
50

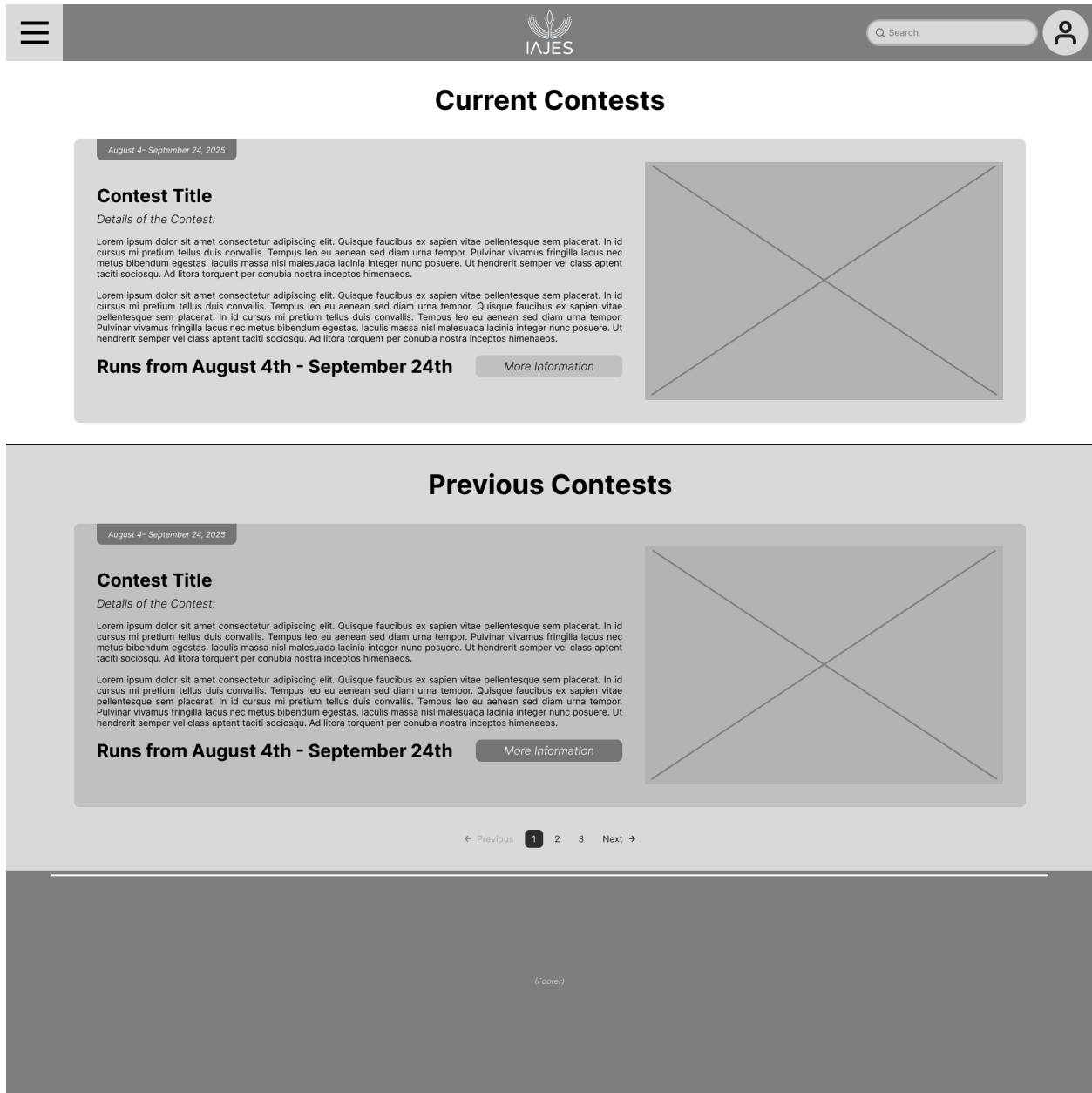


Figure 4.26: Contests

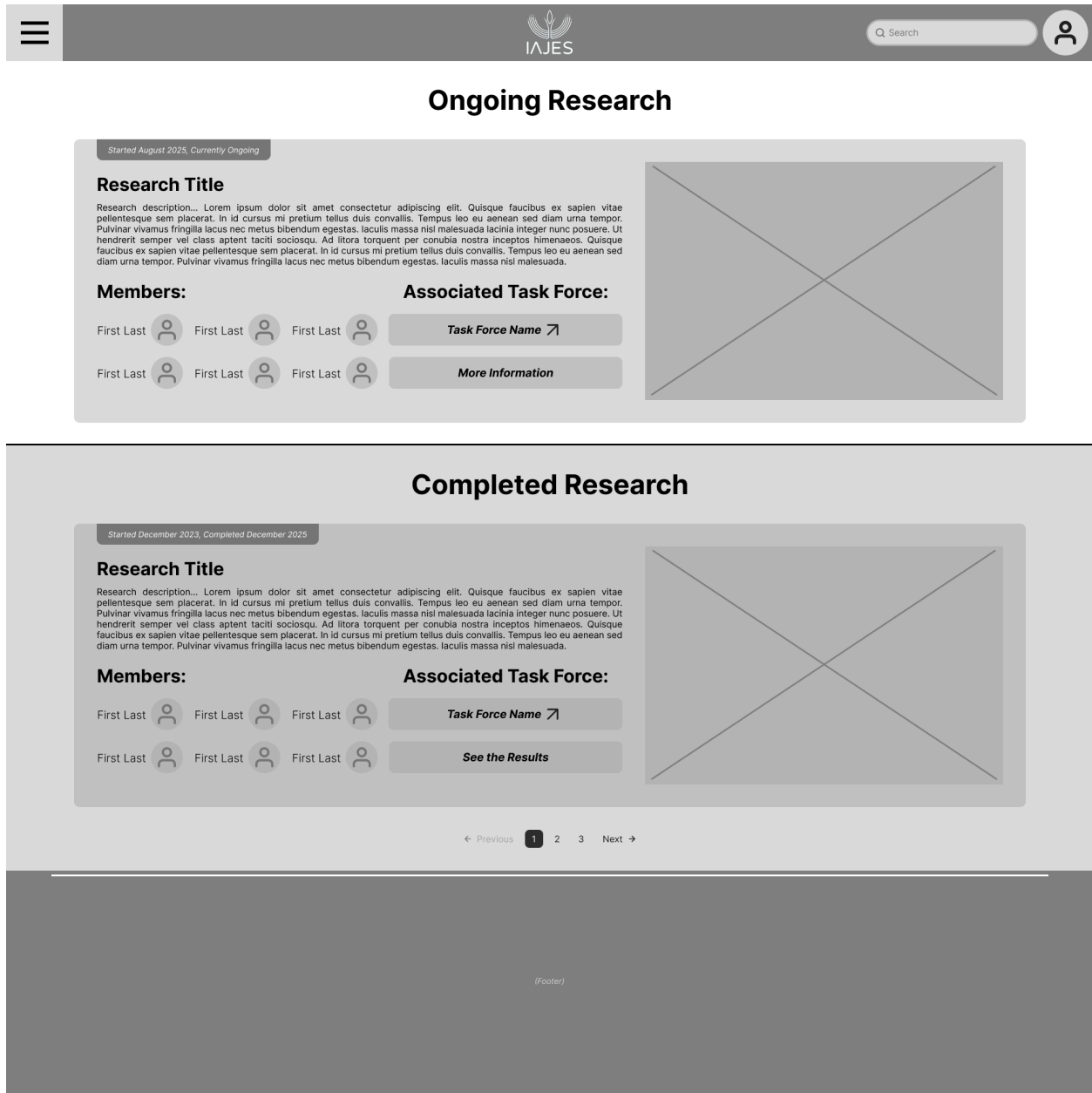


Figure 4.27: Research

4.3.9 Task Forces

The Task Forces hub and detail views are pictured in Figure 4.28 and 4.29, with popups for editing the various sections of the detailed view shown in Figure 4.30 and 4.31. As the heart of IAJES, task forces are especially important to get right, and is one of the most customizable sections of the website. For each task force, administrators would define the modular sections that made up each task force's detailed view. They could enter in entirely new sections of various

types, such as a text block, or list of members. This allowed flexibility to handle multiple different types of content, which admins could tailor to the needs of each task force individually.

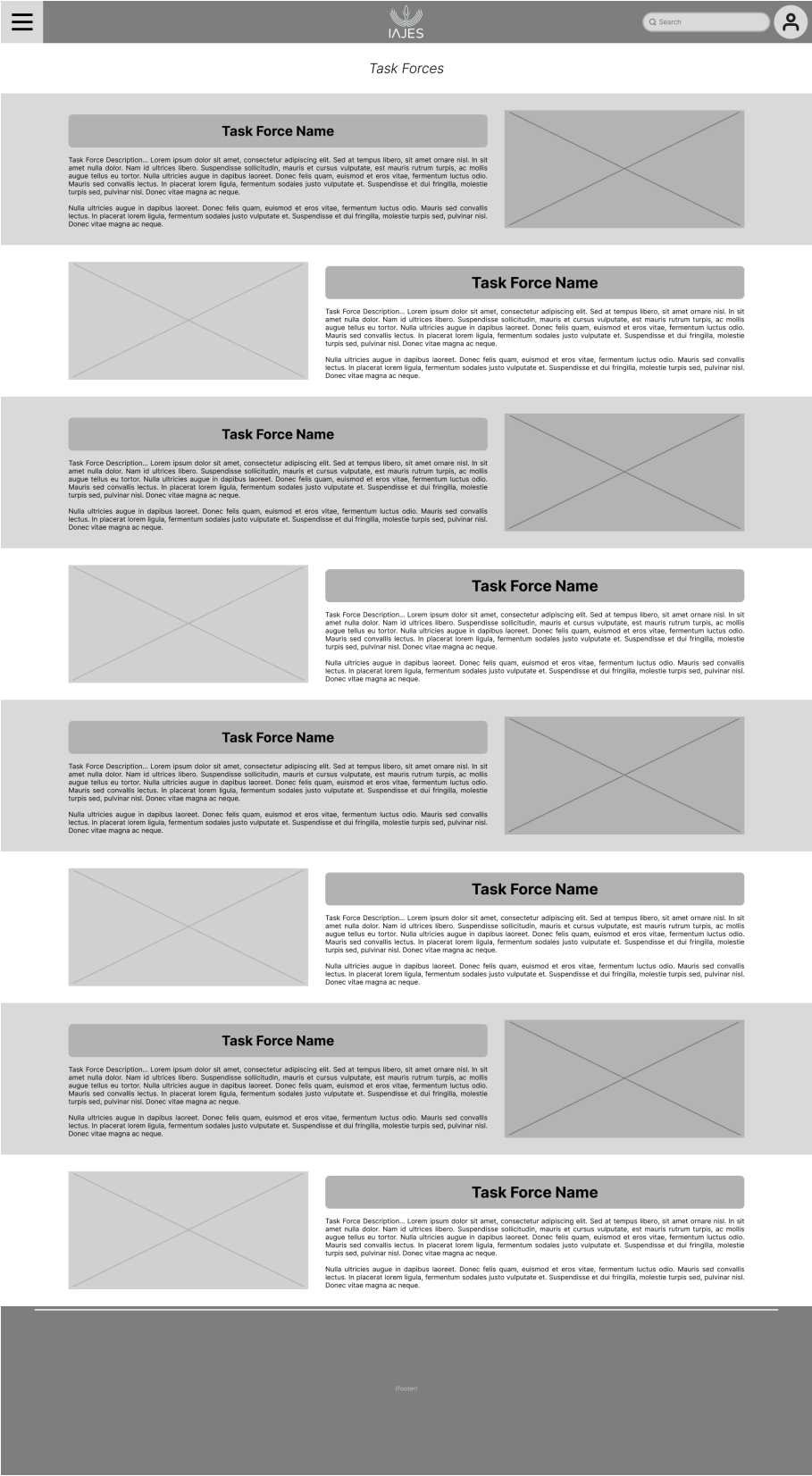


Figure 4.28: Task Forces (List View)

Task Force Name

Who We Are

[Edit](#)

Description Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus porta, sem ut dignissim tincidunt, metus odio lacinia eros, at tincidunt nisi massa ut ipsum. Vivamus at tellus ut massa pellentesque blandit. Vestibulum ut maximus orci, quis semper nisi. Fusce id ante blandit, hendrerit neque sit amet, ultricies turpis. Nulla non dignissim diam. Ut non turpis quis tortor sagittis convallis. Maecenas in sollicitudin leo. Nam laoreet interdum ipsum non aliquet. Ut tristique finibus nisi, id rhoncus velit mattis vitae. Aliquam quis faucibus augue, id faucibus leo. Curabitur tincidunt, nulla id commodo tristique, odio erat lobortis ante, ac tempor lorem enim id urna.

(Member Card)

(Member Card)

(Member Card)

(Member Card)

What We Do

[Edit](#)

Section

(Description Card)

(Footer)

Figure 4.29: Task Forces (Detail View)



Figure 4.30: Task Forces "What We Do" Edit Popup

IAJES

Search

Editing "Who We Are"

SaveReturn

Description

Description Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus porta, sem ut dignissim tincidunt, metus odio lacinia eros, at tincidunt nisi massa ut ipsum. Vivamus at tellus ut massa pellentesque blandit. Vestibulum ut maximus orci, quis semper nisi. Fusce id ante blandit, hendrerit neque sit amet, ultricies turpis. Nulla non dignissim diam. Ut non turpis quis tortor sagittis convallis. Maecenas in sollicitudin leo. Nam laoreet interdum ipsum non aliquet. Ut tristique finibus nisi, id rhoncus velit mattis vitae. Aliquam quis faucibus augue, id faucibus leo. Curabitur tincidunt, nulla id commodo tristique, odio erat lobortis ante, ac tempor lorem enim id urna.

B

I

U

Members

Edit Section NamesAdd Section

Section

Add Member

(Member Card)

(Member Card)

(Member Card)

(Member Card)

(Footer)

Figure 4.31: Task Forces "Who We Are" Edit Popup

57

Chapter 5

System Evaluation

Our tests for a successful website focus mainly on two areas: experience and functionality. Experience refers to the experience of using the website in terms of aesthetics, accessibility, and intuitiveness. While aesthetics are ultimately subjective, tests can be done to ensure that aesthetics are not detracted from on different screen sizes. For instance, illustrative patterns, features, and motifs should maintain their presence on the webpage without intruding on the page content as the aspect ratio of the screen is changed. As the basic idea of responsive design, it is worth testing this area so that users do not have their immersion broken while using the website. Accessibility and intuitiveness are equally as important because there will be many users on the website with differing levels of web awareness and accessibility. Furthermore, some parts of the site can be changed by certain users. To ensure no user frustration, accessibility and usability intuitiveness should be tested. Functionality refers to the actions that the site allows users to perform. As established in the clients' requirements, the website should support information editing on various dynamic pages, and this information should be reflected across the website. Page and user information would thus need to be stored in a database, so fetching and editing from the database must have no issues. Users can also have different roles that determine what actions they can perform on the site. Tests regarding the speed, newsletter generation, markup parsing, and user privileges should be taken into account before final deployment.

5.1 Internal Testing

5.1.1 Experience

To ensure our website had a responsive design that ensured aesthetics were preserved across screen sizes, we made sure to view our website at all major breakpoints during development and before final deployment. These breakpoints were not only tested through different size tabs, but also on any available devices running different operating systems of iOS, Windows, Linux, and others. We looked to see if any page elements exceeded page boundaries, were resized incorrectly, or were otherwise affected negatively by screen size and ratio. To enforce accessibility, we checked

our website against the accessibility guidelines of the WCAG 2.1 (7), to achieve a Level AA rating for the diverse audience that IAJES will encounter. We did this through online website checking services, as well as manual checks that we performed by cross-referencing the WCAG guidelines with our own site. In a similar vein, we tested the accessibility and limitations of the editing features present on our site. As members with certain role permissions are allowed to edit parts of the site, we ensured that any content provided would not break the aesthetics, responsiveness, and accessibility of our website. To test this, we input several common categories of error-prone inputs, such as the following: null/space/unusual unicode characters, a very large amount of text, one or more very large images, an invalid file, code injections. This way, we ensured that content could be limited or rejected by our editing system when necessary. We also mitigate inappropriate content, although this is currently indirectly handled by the permission granting system, and we expect only trustworthy actors to have gained role permissions.

5.1.2 Functionality

Our website has multiple core features that must be fully tested before deployment. Specifically, the data storage and retrieval of many dynamic pages must be robust and long-lasting. Examples of pages that continually store and retrieve data include: Profiles, Newsletters, Conferences, Contests, Biennial Meetings, and Regional Meetings. We tested these pages by storing multiple categories of possibly error-prone inputs and checking their outputs. These categories include: null/space/unusual unicode characters, large amounts of text, large amounts/sizes of images, invalid image file input, and code injections. Determining the outputs of these edge cases helped us to code constraints on the system that disallow certain inputs, still providing a smooth experience for users that now does not break our system. Similarly, we tested our website to ensure it runs smoothly during times of high traffic volume. We emulated this by having an external service to simulate high traffic. We also emulated this by having a group of people all join and interact with the site, although this may be a smaller traffic load than what the website will receive after deployment. In either case, if our website performs poorly under high traffic, we should research and implement optimization measures to counter slowdowns or crashes. Another aspect of data storage is authentication of user profiles. To test this, we created multiple test profiles throughout development, and used them to make sure that profiles have correct error handling during account creation and login, credentials were safely and privately stored in the database using encryption, and credentials were safely retrieved from the database for authentication against login inputs. This whole process was managed by our database management system, but it was still in our best interest to double-check to ensure the security of IAJES members. A smaller feature that also needed to be double-checked included the subscription and generation of newsletters. Newsletters are made of articles, which combine text content, image content, and profiles. Newsletters then need to be generated as an HTML email, before being sent to anyone subscribed. Before deployment,

we made test newsletters and test accounts that subscribe to the newsletter. We will first make sure that all accounts that subscribe are stored safely and securely in the database by checking the code/our database dashboard after performing the subscription action with these test accounts. For the newsletters themselves, each input of text, images, and profiles will need to be checked for accurate retrieval, deletion (if applicable), email generation, and email sending. To do this, we will make, delete, and send out test newsletters to our own emails, ensuring that they are able to be fully created and sent. Another smaller feature to check is the parsing of markup textboxes within editable sections. The ability to have font sizes, font weights, and other text decoration is possible with markup, but we need to ensure that it is parsed and shows up correctly on our site, for all types of devices. Testing of this will be done manually on multiple different browsers and devices, all cross-referenced with each other to notice any differences in any of Markdown's text decoration.

5.2 External Testing

Since our website will be viewed by users across the globe with different needs (and devices), we know that external testing is needed outside of our smaller scope internal methods. Luckily, our team is already heavily involved with the major stakeholder of IAJES leadership, and has been receiving direct feedback from them throughout the entire development process. This feedback has been especially valuable to understand the needs of IAJES members and the problems they may encounter. Having them involved so early in the process, starting all the way from ideation, has been instrumental in creating a website that is accessible and addresses problems in simple and elegant ways. We could design around a set of core features from the beginning, rather than implementing ones we missed later on. Of course, we also plan to gather feedback from actual IAJES members, as a seamless and effective experience for them defines the success of our website. The perception of the general public will be reviewed as well, as allowing others to know what IAJES does in an effective way may incentivize them to join or contribute to the organization. In particular, we may review our website through focus groups, where individuals are monitored as they view the website and provide feedback after their experience. In this test, we would look for how often they were able to find what they were looking for, how easy they found certain features such as editing, and the usefulness of accessibility features for those with disabilities. We may also perform questionnaires without direct monitoring if we do not have time. Finally, if we have access through our database dashboard, we will monitor the website's metrics for user creation, subscription rates, and page views, and determine if any updates are necessary for the site to optimize those metrics.

5.3 Unit Testing

Unit testing for authentication (sign up and sign in) is simple but still useful. We decided to use Jest because of its compatibility with React as well as Javascript. Presently, we have implemented five simple unit tests to ensure that the asynchronous functions interacting with the Supabase's authentication systems are accurate in result and interpretation.

```
test('Sign In with existing user', async () => {
  const testSignIn = {
    email: "ihu@scu.edu",
    pwd: "testdev"
  }
  const result = await signIn(testSignIn);
  expect(result).toBe(true);
});

test('Sign In with unknown email', async () => {
  const testSignIn = {
    email: "unknown@test.com",
    pwd: "testdev"
  }
  const result = await signIn(testSignIn);
  expect(result).toBe("invalid");
});

test('Sign In with wrong password', async () => {
  const testSignIn = {
    email: "ihu@scu.edu",
    pwd: "wrongPwd"
  }
  const result = await signIn(testSignIn);
  expect(result).toBe("invalid");
});

test('Sign Up as existing user', async () => {
  const testSignUp = {
    email: "ihu@scu.edu",
    pwd: "testdev"
  }
  const result = await signUp(testSignUp);
  expect(result.success).toBe(false);
  expect(result.message).toBe("An account with this email already exists. Please sign in instead.");
});

test('Sign Up as new user', async () => {
  const testSignUp = {
    email: "test@email.com",
    pwd: "testdev"
  }
  const result = await signUp(testSignUp);
  expect(result).toBe(true);
});
```

```
});
```

5.4 Integration Test(s)

5.4.1 Overview

Our integration tests verify that data flows correctly between the major components of the IAJES website: the React frontend, the Supabase Authentication service, and the Supabase PostgreSQL database (the custom users table). Rather than testing any single function or component in isolation, these tests exercise the full chain that a real user triggers:

1. Entering information on a form.
2. API calls to an external auth provider.
3. Database writes.
4. Database queries that the profile page depends on.

We used AI assistance to design these tests using the Node.js built-in test runner (`node:test`), which required zero additional dependencies beyond what the project already had installed.

5.4.2 Test 1: Signup-to-Profile Data Flow

Components under test: Signup page (`signup.jsx`) → Supabase Auth API → custom users database table → profile page query (`profile.jsx`)

This test validates the most critical user-facing behavior: when a user creates an account, the information they enter (first name, last name, and email) must be correctly stored and retrievable.

Procedure

- **Generation:** The test generates a unique email address using a timestamp (`test-<timestamp>@test.com`) to prevent collisions.
- **Auth Call:** It calls `supabase.auth.signUp()` with a test name, email, and password, mirroring `signup.jsx`.
- **Database Insert:** It inserts a row into the custom users table with the user's ID, matching the production insert logic.
- **Data Fetch:** It queries `supabase.from("users").select("*").eq("id", userId).single()`, mirroring the `profile.jsx` logic.

- **Assertion:** It asserts that fields (`fname`, `lname`, `email`, `id`) round-trip correctly.
- **Cleanup:** An `after()` hook deletes the test row to prevent data accumulation.

What this verifies: This test crosses three system boundaries and confirms that the user ID returned by Auth is a valid foreign key for our custom table and that column names match between write and read operations.

5.4.3 Test 2: Profile Field Mapping Verification

Components under test: Database schema → `mapUserRow()` function in `profile.jsx` → rendered profile UI fields.

Procedure

- Create an account through the live signup page (`/signup`) with known values.
- Navigate to `/profile/<user-id>` in the browser.
- Verify UI rendering: names in the header, email in the `mailto:` link, and ensure optional fields are empty rather than `undefined`.
- Use the edit profile popup to update fields (title, biography) and confirm they render correctly.

What this verifies: Confirms the frontend's data-shape assumptions match the database. It specifically validates that `mapUserRow()` handles null/empty fields gracefully without crashing.

5.4.4 Test Execution

The automated test runs via `npm test`, which invokes `node -test tests/*.test.mjs`. The test connects to the live Supabase instance and executes against real infrastructure.

5.4.5 Summary

These tests validate the data layer end-to-end (automated) and the frontend rendering (manual). Together, they cover the full path from user input to screen output across the React frontend, Supabase Auth, and PostgreSQL database.

Chapter 6

Implementation Plan

6.1 Timeline

To ensure the smooth progression of our project, our team created two Gantt Charts in the figures below to provide structure and estimate roughly how long each process should take. Figure 6.1 shows an overview of the process at a high level, blocking out the various stages of development. The stages included: Requirement Determination, Figma and Design Development, Initial Software Development, Software Development Finalization, as well as Rebase and Management. Senior Design Requirements were to be addressed throughout the year, and important dates were marked at the bottom three rows. Figure 6.2 has more specific dates and tasks, breaking down each of these phases clearly. Both Gantt Charts were useful, as they helped our team visualize the future and plan ahead accordingly.

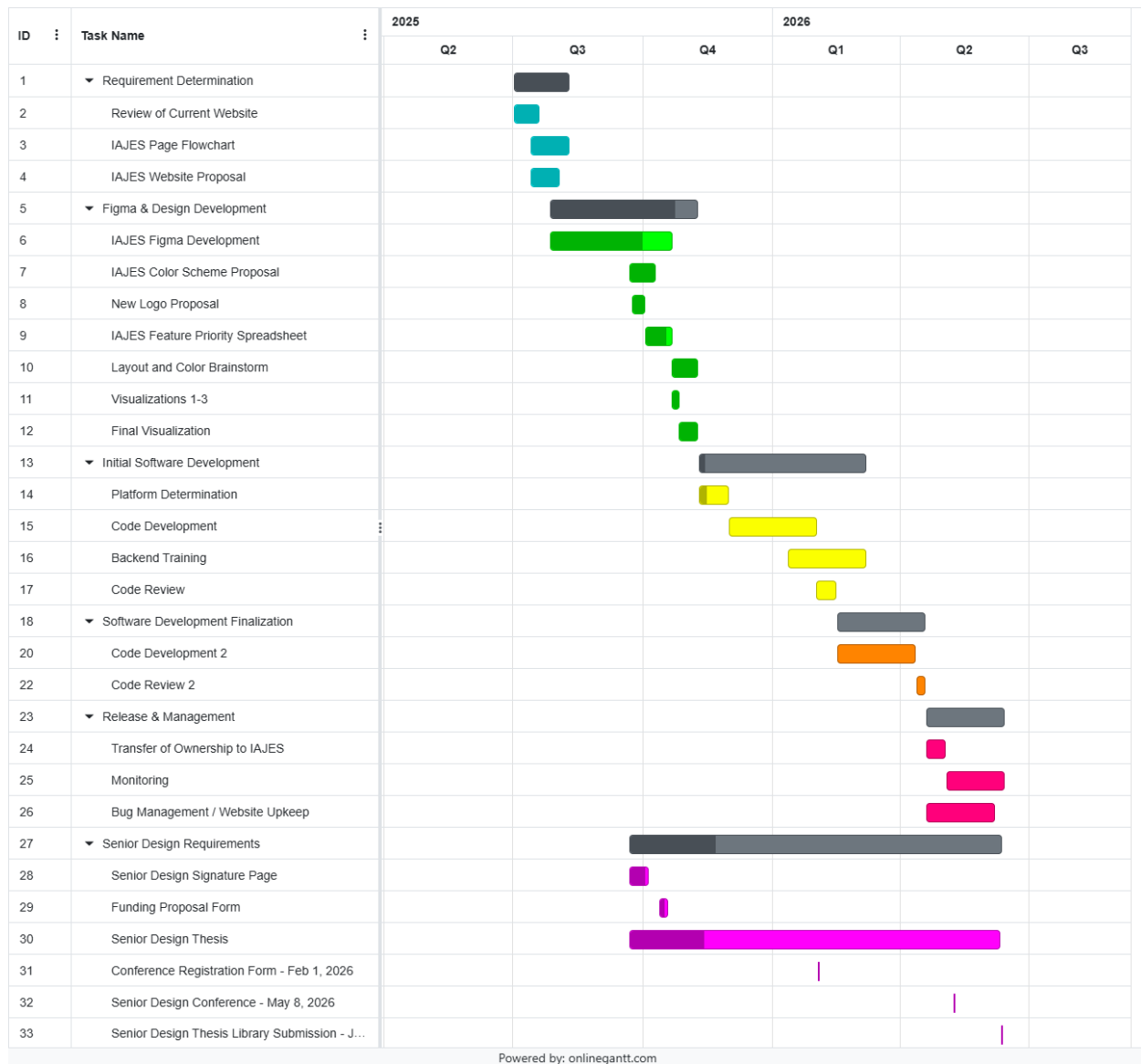


Figure 6.1: A Gantt Chart depicting the project process

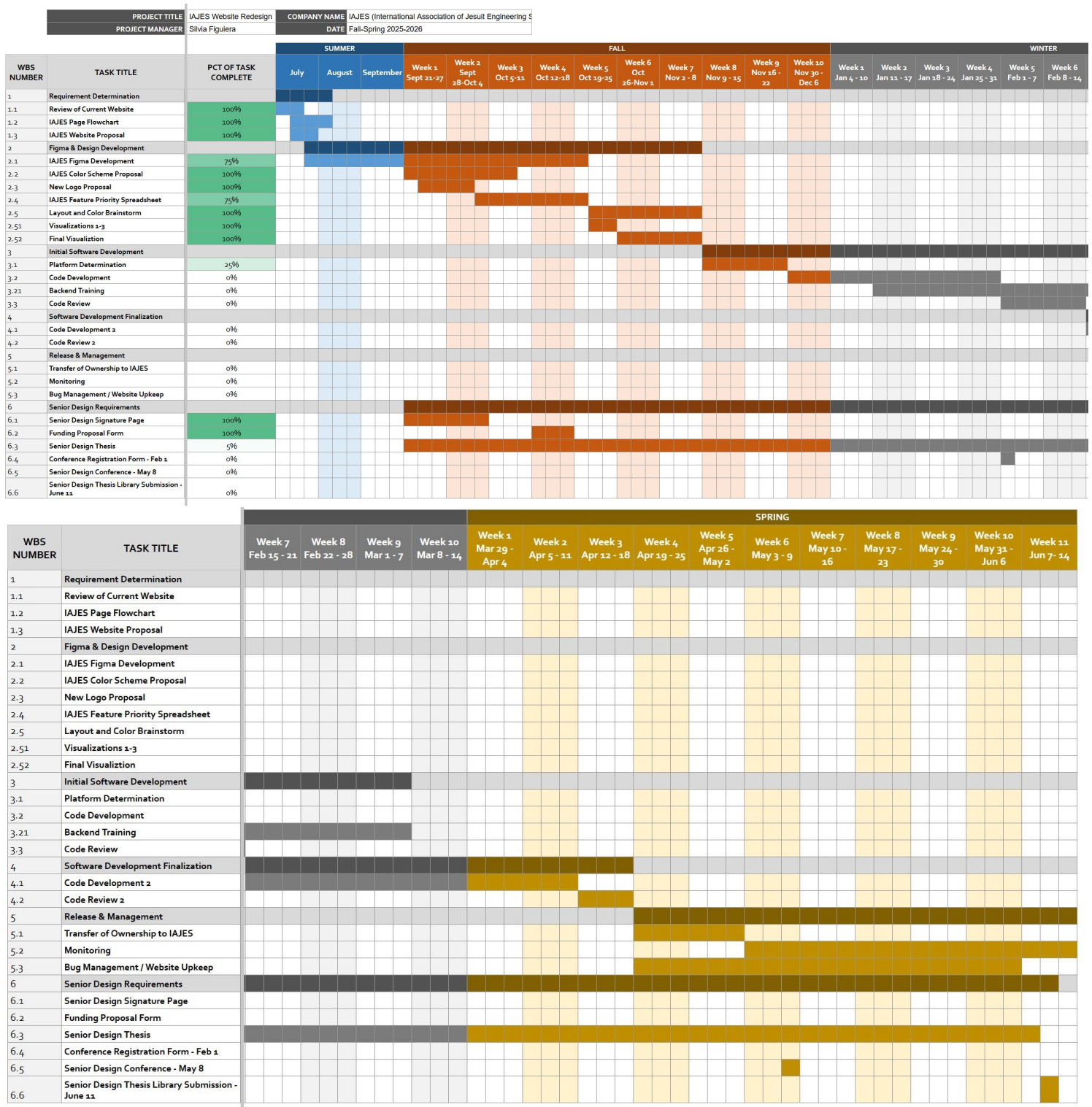


Figure 6.2: A detailed spreadsheet Gantt Chart with more precise dates.

6.2 Agile software development

Our team used agile software development methodologies during the development of this project. During our fall and winter quarters, we met with IAJES leadership bi-weekly, and our senior advisors weekly. In each two-week sprint, we worked towards different goals set at the beginning of that sprint.

Our goals were determined both by feedback from IAJES, as well as our user stories—scenarios that outlined required functionalities for their intended audience. For example, we implemented user profiles early on in development to address a high priority user profile that applied to the whole audience. Anyone who logged in on the website would gain access to the newsletter subscription. For validated IAJES members, this also meant a custom profile that allowed them to connect with others by showcasing their presence and interests online.

In each sprint, we manage a master “to-do” list that dictates tasks and who is assigned to each task. This allowed our team to see if anything was unassigned, who had completed their tasks, and what still needed to be done. For example, the use case requiring user profiles was split into multiple tasks: signup authentication, login validation (and logout), storing users onto our database, the creation of the profile front-end, hooking up the profile to the corresponding database table, the creation of a search page front-end, and the profile search functionality.

6.3 Project Risks

Table 6.1 shows project risks determined after discussions with IAJES and within our team. Here, we define the risks present with our project, the consequences if a risk goes unaddressed, and actions we can take to mitigate the identified risks.

Risk	Consequence	Mitigation
Our real IAJES data may be incomplete or delayed.	Limits functionality of search and profiles.	We will communicate early and often with IAJES engineers.
The implementation of user roles (admin, faculty, student) may be complex due to a large amount of possible interactions. The complexity of roles may cause ambiguity between permissions for users.	May result in confused users and discourage them from using the site, as well as feature delays and access issues.	Extensive testing with each user role and clear documentation for each role, its purpose, and its capabilities.
Too many features too early for frugal development. Planned features may take up more resources and funds that detract from the goal of developing frugally.	Incomplete or unstable deliverables and end product.	Careful research of the scope of planned features, as well as communicate with IAJES and FIH of what is possible and what is not.
Legal concerns about the security of user account information and privacy laws may limit sending newsletters.	User data is unsecure and able to be leaked, newsletter functionality may be limited or scrapped.	Use secure authentication, encrypt user data, follow privacy best practices and limit data access based on user roles.
Implemented editing systems and codebase being too complex for IAJES to take over after senior design finished.	Discourages IAJES members from interacting or updating pertinent information on the site.	Test with general IAJES members on the ease of editing through in-site forms. Test with IAJES engineers on the ease of editing the codebase. Revise systems based on feedback gathered.

Table 6.1: Project Risks

Chapter 7

Constraints and Standards

7.1 Constraints

7.1.1 Integration with Existing IAJES infrastructure

A key technical constraint was the requirement to integrate the redesigned website with IAJES's existing infrastructure and data systems. While the development utilized Supabase for backend functionality, the final deployment needed to align with the organization's domain, data management practices and administrative workflows. This constraint influenced architectural decisions, database schema design and deployment strategies to ensure long term maintainability by IAJES administrators.

7.1.2 Accessibility and Frugal Innovation Constraint

Through collaboration with the Frugal Innovation Hub, the project was constrained to use minimal resources while maximizing accessibility and impact. This required the team to prioritize lightweight technologies and intuitive user interfaces that function effectively even in low bandwidth environments. These constraints ensured that the platform remained inclusive and sustainable for stakeholders across diverse global regions. This emphasis on accessibility aligns with WCAG 2.1 standards. (7)

7.1.3 Security and Privacy Constraint

The need to protect sensitive user information imposed strict constraints on system design. The platform required secure authentication mechanisms and encrypted data transmission. These requirements limited design flexibility but ensured compliance with a recognized security framework that safeguards the confidentiality and integrity of user data.

7.1.4 Time Constraint

As a senior design project, development was bound by the academic calendar, including milestones for proposal, prototyping, implementation, testing and final deployment. This limited timeframe necessitated prioritization of essential features such as dynamic pages, user accounts, and newsletters while deferring non-critical enhancements to future development phases.

7.2 Standards

7.2.1 Unified Modeling Language (UML) Standards

Unified Modeling Language (UML) standards provide an essential framework for representing and communicating the structural and behavioral components of a software system through standardized visual notation. UML establishes a common language that enables developers, analysts, and designers to collaboratively conceptualize, design, and document complex systems. These standards include a range of diagram types, such as class diagrams, sequence diagrams, and use case diagrams, each intended to capture different aspects of system architecture and functionality (9). In developing the diagrams and plans for this project, UML conventions were followed to ensure clarity and readability within the software engineering community.

7.2.2 General Data Protection Regulation (GDPR) - European Union Privacy Law

The IAJES website manages data according to GDPR law. GDPR governs how personal data must be collected, stored, processed, and deleted for users in the European Union (3). However, organizations outside the EU often adopt GDPR as a best-practice model for privacy engineering, especially when serving an international audience such as the members of IAJES. It establishes requirements such as informed consent, transparency through a privacy policy, the right to access personal data, and the right to be forgotten. These requirements directly influence our implementations of account deletion, data storage, and user consent.

7.2.3 OWASP Top 10 Web Application Security Risks - OWASP Foundation

The OWASP Top 10 is an industry-recognized list of the most critical security vulnerabilities affecting web applications. It highlights risks such as broken authentication, injection attacks, sensitive data exposure, and security misconfiguration (6). Our project is impacted by this standard because OWASP guidance helps ensure secure implementation of our authentication and password resets, API communication with Supabase, and role-based access permissions.

7.2.4 California Consumer Privacy Act (CCPA) - California Privacy Law

CCPA establishes privacy rights for California residents regarding how their personal data is collected, used, and deleted. It requires clear disclosure of data collection practices, the ability for users to request deletion of their data, and transparency in privacy policies (2). Since our project is developed at Santa Clara University and may serve California users, CCPA directly applies to the account deletion and privacy policy features of our website.

Chapter 8

Societal Issues

8.1 Ethical

While working on the IAJES website, there were multiple ethical questions that we had to consider. Firstly, what data do we collect from IAJES members? To address this question, our team only used data that was necessary to achieve IAJES' main goal for the site: facilitating communication between members. Data points that did not serve to work toward this goal—or that did not contribute to the function of important website features such as authentication—were not collected.

Secondly, how do we handle privacy and the handling of IAJES members' data? For this concern, our team ensured that our chosen database had a long history of secure storage, that data was stored, encrypted, and retrieved correctly, and that members could delete their profile and their data at any time.

Finally, how do we decide the permissions for each role, and who gets access to what? Similar to data, our team ensured that all roles were essential to IAJES' central mission, and that no role had overextended permissions that did not apply to them. For example, while task force administrators would have the ability to edit task force pages, they may not have access to the editing of the general site. This is to ensure that power is not abused easily. Furthermore, a role approval process was drafted, ensuring that roles are deserved and applied to real IAJES members.

These are just some of the ethical questions our team discussed. Throughout the project, we were guided by a philosophy very similar to Principle Five of the ACM Software Engineering Code of Ethics and Professional Practice: the idea that software engineers should “subscribe to and promote an ethical approach to the management of software development and maintenance” (1). This philosophy, discussions with IAJES, and previous experience in web development were all crucial to help our team determine ethical solutions to these questions.

8.2 Social

The development of a new website for IAJES has a significant social impact because the functionality and user experience of our website are intended to enhance and supplement the experience of the IAJES community directly. Our project has a dedicated client—the IAJES organization—and the primary audience was their community members. Although our team is not affiliated with IAJES, we had an obligation to benefit our client with whom we developed a professional relationship and agreement. This obligation was derived from Principle Two of the ACM Software Engineering Code of Ethics and Professional Practice, which states that software engineers should act in the interests of their client and employer while also remaining consistent with the public interest.

If we had intentionally developed a site that was difficult to use and did not facilitate communication between IAJES members, we would have wasted our client's time and implied that their needs were not important. Therefore, our team took the time to truly empathize with IAJES and understand their needs and how best to address those needs through the project. For example, when making design decisions, we carefully considered what they valued most: accessible navigation, easy editing, effective communication between members, and a simple yet professional design.

The website also has social impacts outside the IAJES community, as the website represents their organization to the public. As the first principle of ACM's code states, "[s]oftware engineers shall act consistently with the public interest" (1). Therefore, our team considered how not only IAJES members would use the site, but anyone learning about the organization for the first time. For example, the site immediately provides useful information about IAJES in a clear, structured manner that facilitates easy understanding by the general public.

8.3 Political

The political impact of this project lies in information equity and global representation. IAJES serves a diverse network of institutions across different nations, each with its own educational standards and digital infrastructures.

The project facilitates "soft power" by creating a unified platform where Jesuit institutions in the Global North and Global South can collaborate as equals. By improving the UI/UX, we are essentially democratizing access to STEM resources, ensuring that a student or faculty member in a resource-constrained region has the same ease of access as one in a well-funded institution.

Navigating the "will" of the general public in this context refers to the diverse needs of the international Jesuit leadership board, faculty, and students. The project requires balancing the organizational hierarchy (the Chilean leadership) with the practical needs of the global user base.

By implementing structured backend functionalities and user roles, the project impacts how academic information

is governed and disseminated. This serves a greater societal good by promoting transparent, cross-border scientific collaboration, which is a core tenet of the Jesuit mission to address global challenges through education.

8.4 Economic

In an engineering context, economic considerations for this project revolve around Total Cost of Ownership (TCO) and the Frugal Innovation mandate of the project's sponsor.

Unlike a commercial product, the IAJES website does not have a direct revenue stream. Therefore, our design decisions must prioritize low-cost, high-efficiency hosting and maintenance. We must consider the "cost of money" by evaluating the long-term hosting fees of different backend frameworks (e.g., Python-based vs. JavaScript-based) and cloud services.

A significant economic decision involves choosing between custom-coded solutions and existing Content Management Systems (CMS). While a custom React/Next.js frontend offers high performance, we must weigh that against the future economic cost of hiring specialized developers to maintain it once our senior design team graduates.

Working with the Frugal Innovation Hub, our "pricing" is measured in human capital and time. We must account for the opportunity cost of our technical choices, selecting technologies like Firebase and Supabase, must be justified by their ability to scale without requiring massive infrastructure investments.

The "economic" success of the project is defined by delivering a professional-grade global platform on a student-led budget, essentially providing the IAJES board with a high-value asset that would otherwise cost tens of thousands of dollars in a commercial agency setting.

8.5 Health and Safety

For a global organization, accessibility is a requirement, as we must adhere to WCAG 2.1 (Web Content Accessibility Guidelines). Failing to do so "endangers" the inclusivity of the platform, effectively barring users with visual, auditory, or cognitive impairments from essential academic resources.

Our UI/UX design must account for ergonomic visual principles. This includes ensuring proper color contrast ratios to prevent eye strain and designing a responsive interface that minimizes repetitive strain (excessive clicking/scrolling) for users navigating the site over long periods.

While the site may not handle medical records, it manages user roles and resource sharing. We must implement secure backend protocols to protect the "digital safety" of stakeholders. In a global Jesuit network, data breaches could expose sensitive institutional information, making robust authentication a primary safety concern.

We must consider international data privacy laws, such as the GDPR (General Data Protection Regulation) in Europe and the LGPD in Brazil, which dictate how safe a user's personal data must be and the legal ramifications of failing to protect it.

8.6 Manufacturability

A significant manufacturability consideration is the use of modular components (e.g., React components). By building a reusable design system in Figma and translating it into code, we ensure the site can be "manufactured" more quickly and updated without rebuilding the entire architecture. The project is highly manufacturable using modern frameworks like React and Python. However, we must balance complexity with our senior design timeline. Using containerization tools like Docker improves manufacturability by ensuring the "product" runs identically on any developer's machine and in the final cloud environment, reducing "it works on my machine" bottlenecks.

A major problem that arises in software "manufacturing" is technical debt. If we build a backend that is too complex, the IAJES staff may not be able to maintain it. To address this, we are prioritizing a user-friendly Content Management System (CMS) interface, ensuring the "end-user" can "manufacture" new content without needing a software engineer.

The primary "raw materials" for our project are developer hours and cloud credits. The risk to manufacturability arises if the backend integration takes longer than the allotted academic year. We mitigate this by using an Agile development cycle, allowing us to produce a "Minimum Viable Product" (MVP) before adding secondary features.

8.7 Sustainability

Sustainability was reflected in this project not only in the long-term maintainability of the platform, but also in the project's alignment with principles of frugal innovation. Through our work with the Frugal Innovation Hub, we were encouraged to think carefully about how to create an effective solution while using as few resources as possible. This meant prioritizing practical features, avoiding unnecessary complexity, and designing a website that would be accessible to a broad range of users. Rather than relying on costly or resource-heavy approaches, our team focused on building a platform that IAJES could realistically manage and sustain over time. In this way, sustainability for our project meant creating a solution that was efficient, adaptable, and inclusive, while still meeting the organization's needs.

8.8 Environmental Impact

Although a web-based system has a smaller environmental footprint compared to physical products,our project still required careful consideration of its environmental impact. The project minimized resource consumption by leveraging cloud-based infrastructure, which benefits from energy-efficient data centers and optimized resource allocation. Efficient coding practices, optimized database queries, and responsive design further reduced server load and energy usage, contributing to lower carbon emissions. Additionally, by digitizing communication and collaboration among IAJES members, the platform reduces the need for printed materials and physical meetings, thereby decreasing paper consumption and travel-related emissions. Selecting scalable and maintainable technologies also helps prevent unnecessary redevelopment and electronic waste over time.

Chapter 9

The Final Product

9.1 Differences Between Prototype and Final Product

During the transition from prototype designs to the implemented system, several structural changes were made to better align the website with IAJES' actual needs and to simplify long-term maintenance. Early prototype pages for contests, participatory activities, conferences, and research were removed after discussions with our client revealed that these categories would either overlap significantly with existing task force content or was planned to be scrapped anyway. Similarly, the concept of “projects” was dissolved into the Task Forces pages, where ongoing initiatives are now naturally found within the groups responsible for them. The originally planned webinars hub and detail view structure was condensed into a single, streamlined archive page, and a new Video Resources archive was introduced to store recorded content in a more flexible format. Additionally, the Biennial Meetings section was renamed to International Meetings and also simplified into a one-page archive for easier browsing. Finally, the original design included university search and general page search features, but these were removed in favor of a focused profile search system that better supports IAJES' primary goal of member connectivity.

9.2 Final Pages

9.2.1 Sign-up and Login

These pages implement secure university-email authentication and role-based access control, closely reflecting the prototype design while integrating real validation, session management, and database interaction. A small change from the prototype was the removal of a dedicated “logout” page, now being delegated to a single click in the dropdown that appears when hovering the top right profile icon. (Figures 9.1 through 9.2)



Create an IAJES account

First name: Last name:

Email:

Create Password:

Re-enter Password:

☐ Subscribe to IAJES News

Already have an account? [Sign in here.](#)

Figure 9.1: Sign-up



Sign in to IAJES

Email:

Password: [Forgot password?](#)

[Sign In](#)

New here? [Create an account.](#)

Figure 9.2: Login

9.2.2 Profile Search

The final profile search page focuses exclusively on discovering members through filters such as task force, location, and research interests, replacing the broader university and page search concepts from the prototype. (Figure 9.3)

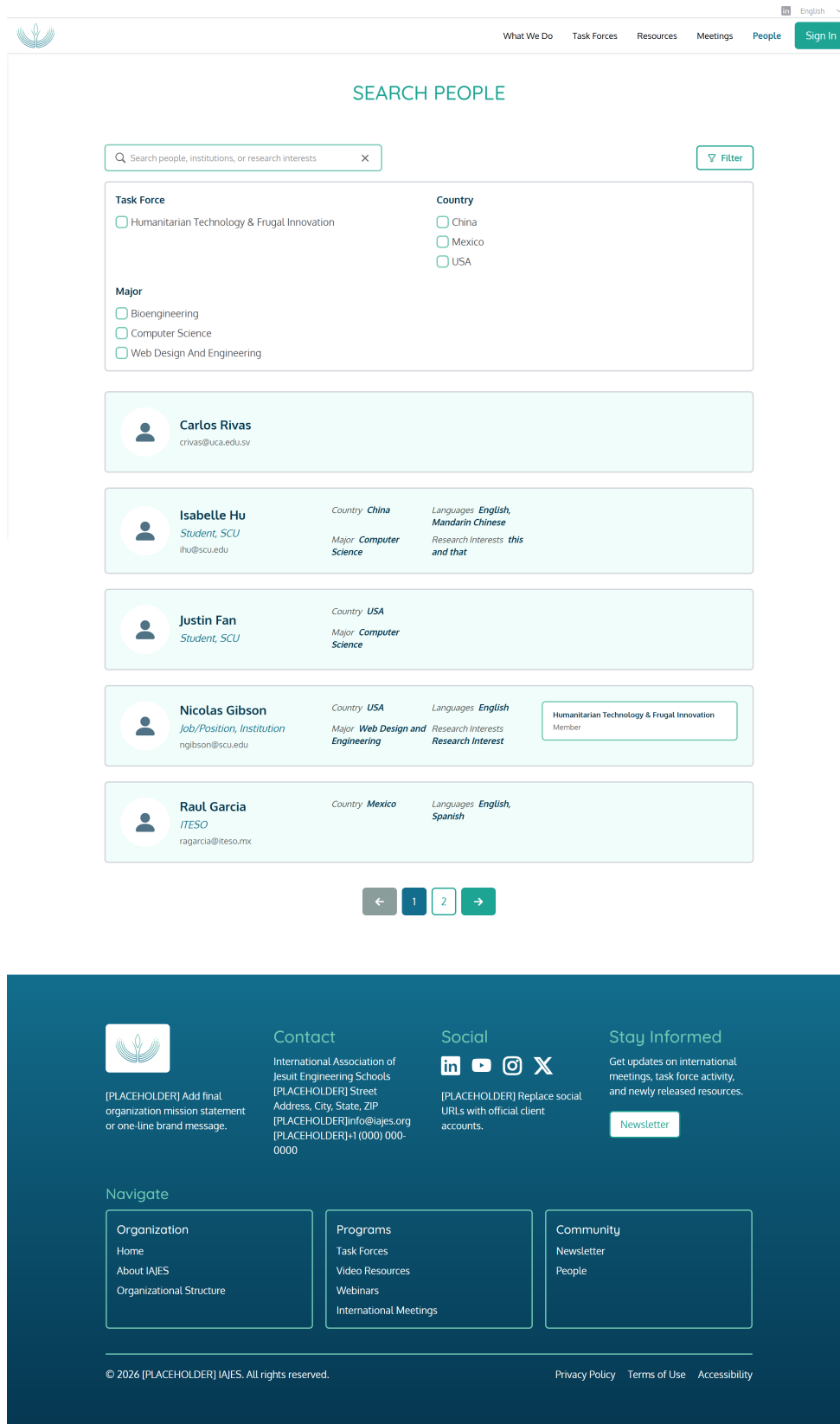


Figure 9.3: Profile Search

80

9.2.3 Profile Page

The profile page allows members to view and edit their personal information through an interface that closely mirrors the prototype while now being fully connected to live database updates and visibility rules. (Figures 9.4 through 9.6)

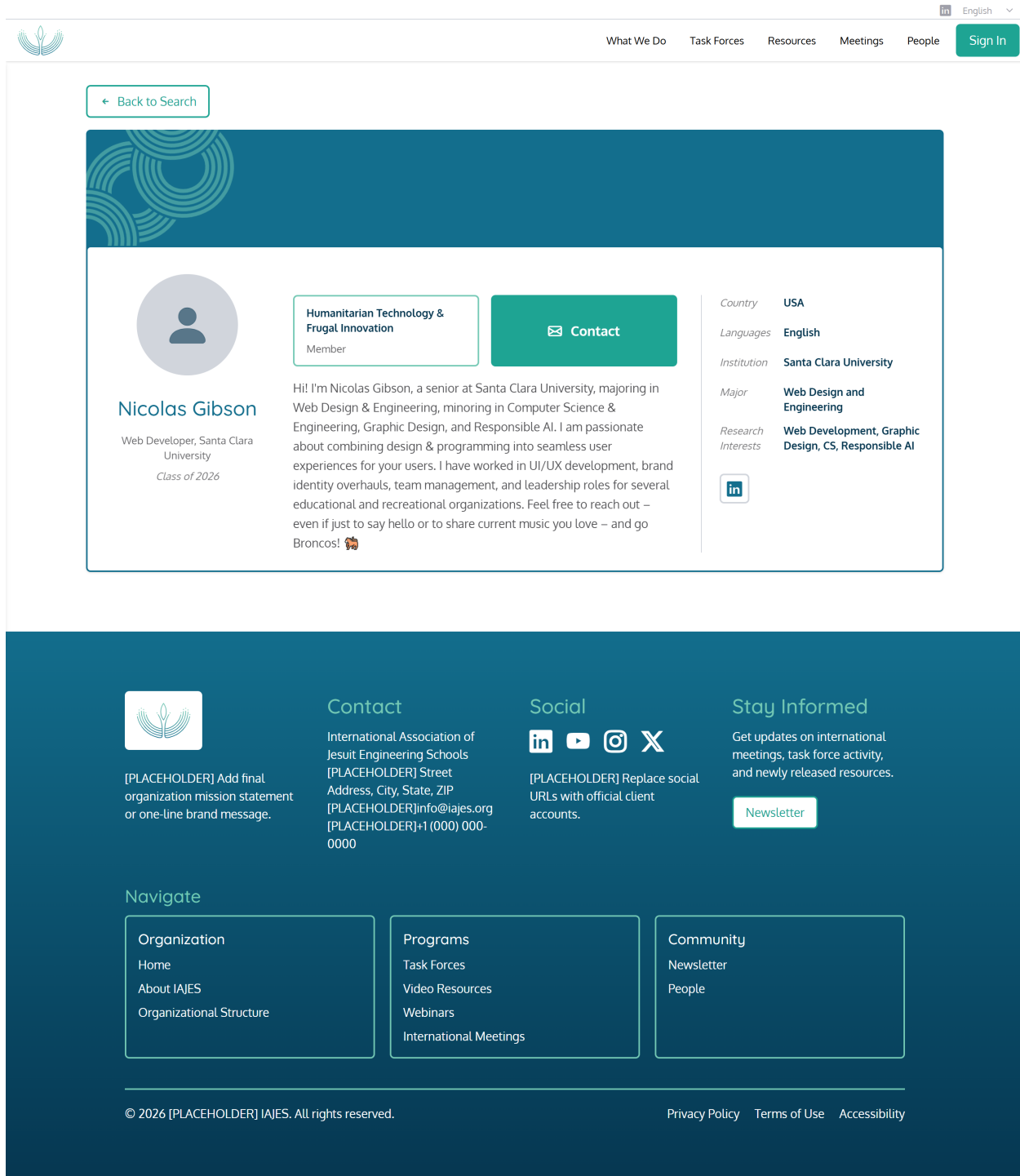


Figure 9.4: Profile

Edit Profile

Personal Information

First Name: Last Name:

☒ Allow site visitors to contact you?

Languages:

Banner Type:

Tagline:

Biography:

Hii I'm Nicolas Gibson, a senior at Santa Clara University, majoring in Web Design & Engineering, minoring in Computer Science & Engineering, Graphic Design, and Responsible AI. I am passionate about combining design & programming into seamless user experiences for your users. I have worked in UI/UX development, brand identity overhauls, team management, and leadership roles for several educational and recreational organizations. Feel free to reach out – even if just to say hello or to share current music you love – and go Broncos! 🐾

Academic Information

Job/Position: Institution:

Figure 9.5: Profile Editing (1)

Academic Information

Job/Position: Institution:

Country: Major:

Research Interests:

Task Force

Task Force: Task Force Role:

Social Links

LinkedIn: Instagram:

X (Twitter): Facebook:

Website:

Figure 9.6: Profile Editing (2)

9.2.4 Landing Page

The landing page maintains the modular, welcoming layout from the prototype while incorporating finalized branding, real content, and dynamic announcements, now called highlights, which are editable by administrators. (Figures 9.7 through 9.9)

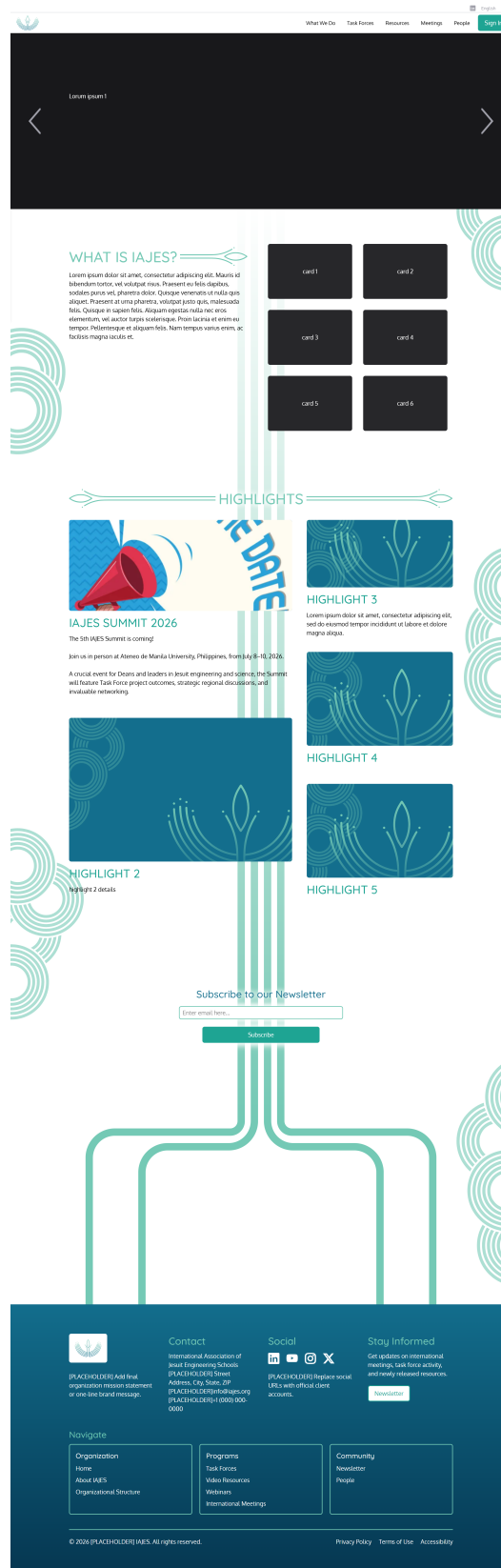


Figure 9.7: Landing
85

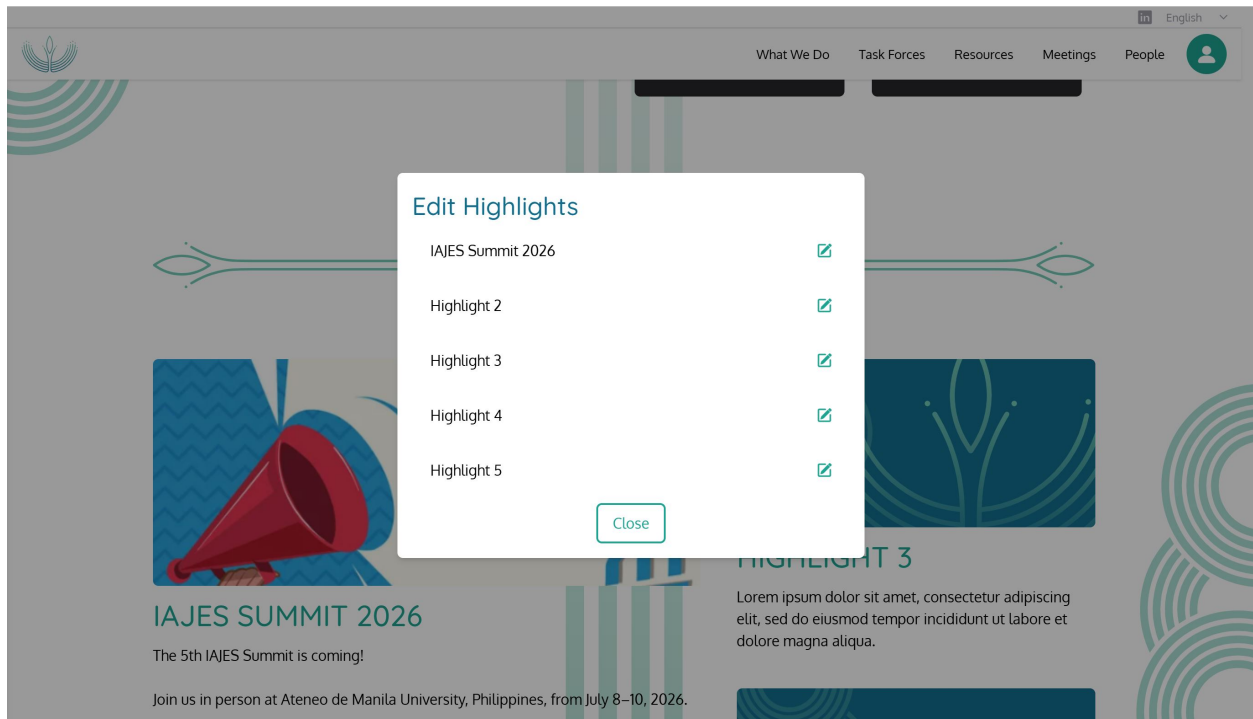


Figure 9.8: Landing Editing (1)

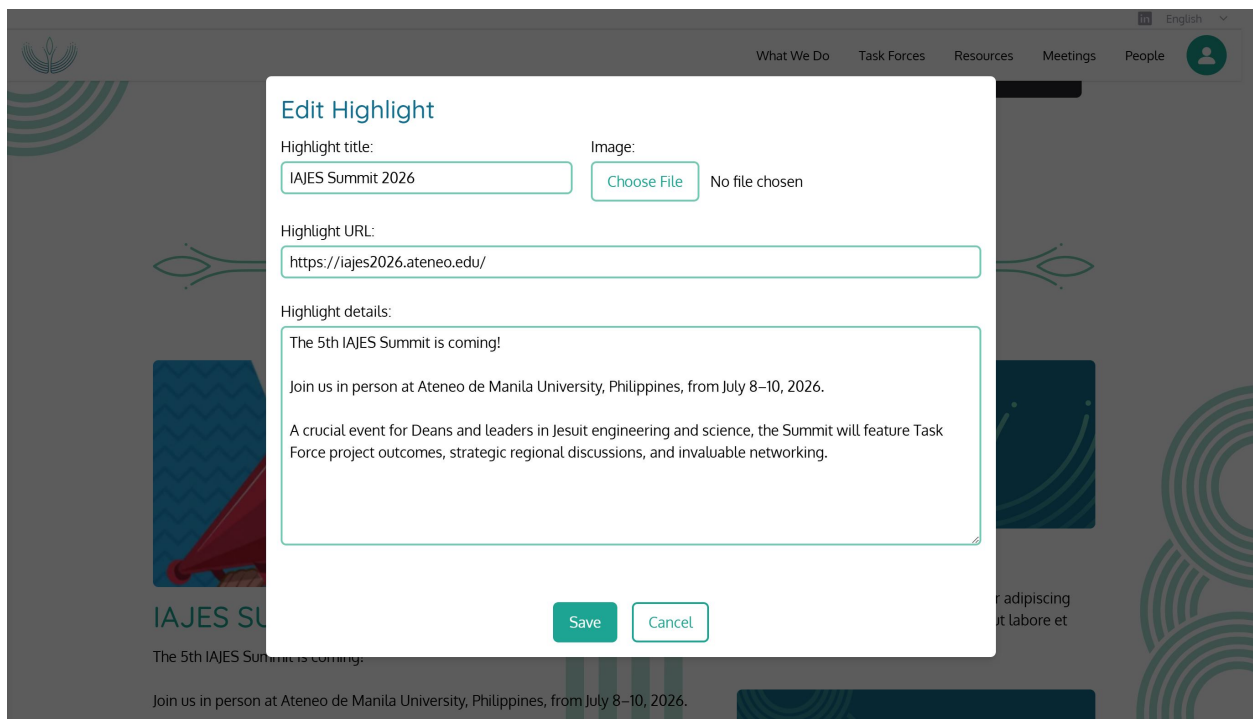


Figure 9.9: Landing Editing (2)

9.2.5 About IAJES

This page preserves the prototype's informational structure with clear sections and refined content organization for clarity and maintainability. (Figure 9.10)

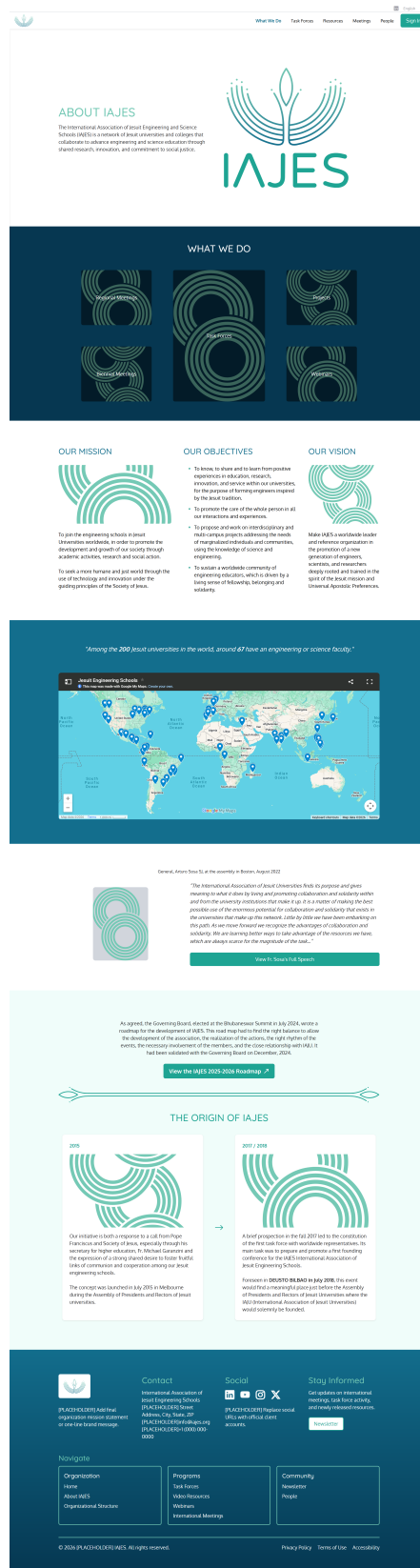
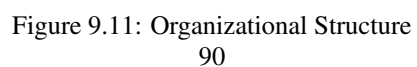


Figure 9.10: About IAJES

9.2.6 Organizational Structure

The organizational structure page reflects the prototype's visual clarity while integrating updated diagrams and responsive design elements for improved accessibility across devices. (Figure 9.11)



9.2.7 International Meetings (Biennial Meetings)

Originally designed as Biennial Meetings with nested detail views, this section was simplified into a clean archive of International Meetings for easier navigation and content management. (Figure 9.12)

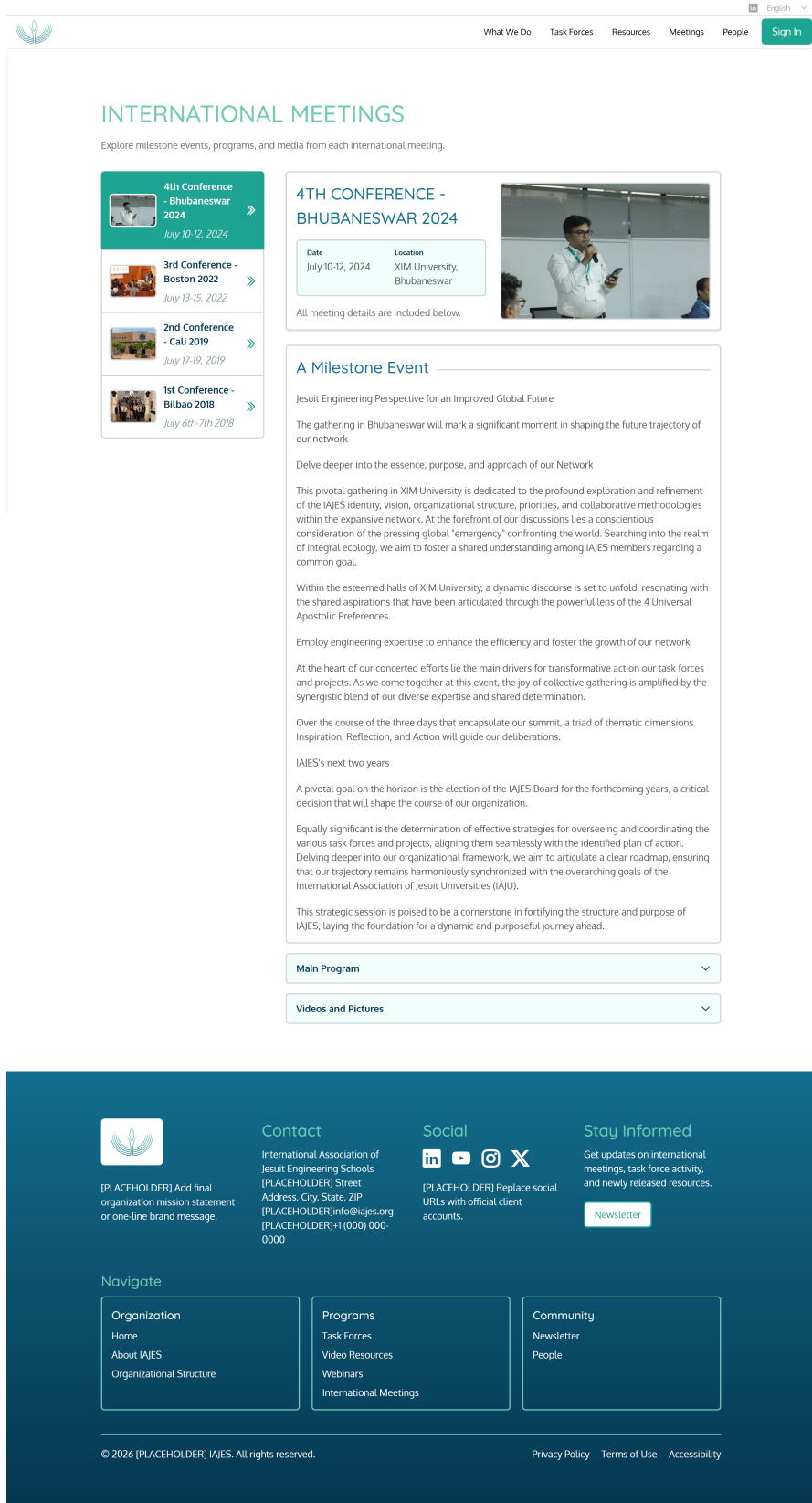


Figure 9.12: International Meetings

92

9.2.8 Regional Meetings

The structure of the regional meetings pages remain faithful to the prototype design, with modular galleries and editable meeting entries that allow administrators to easily add and update content. (Figures 9.13 through 9.18)

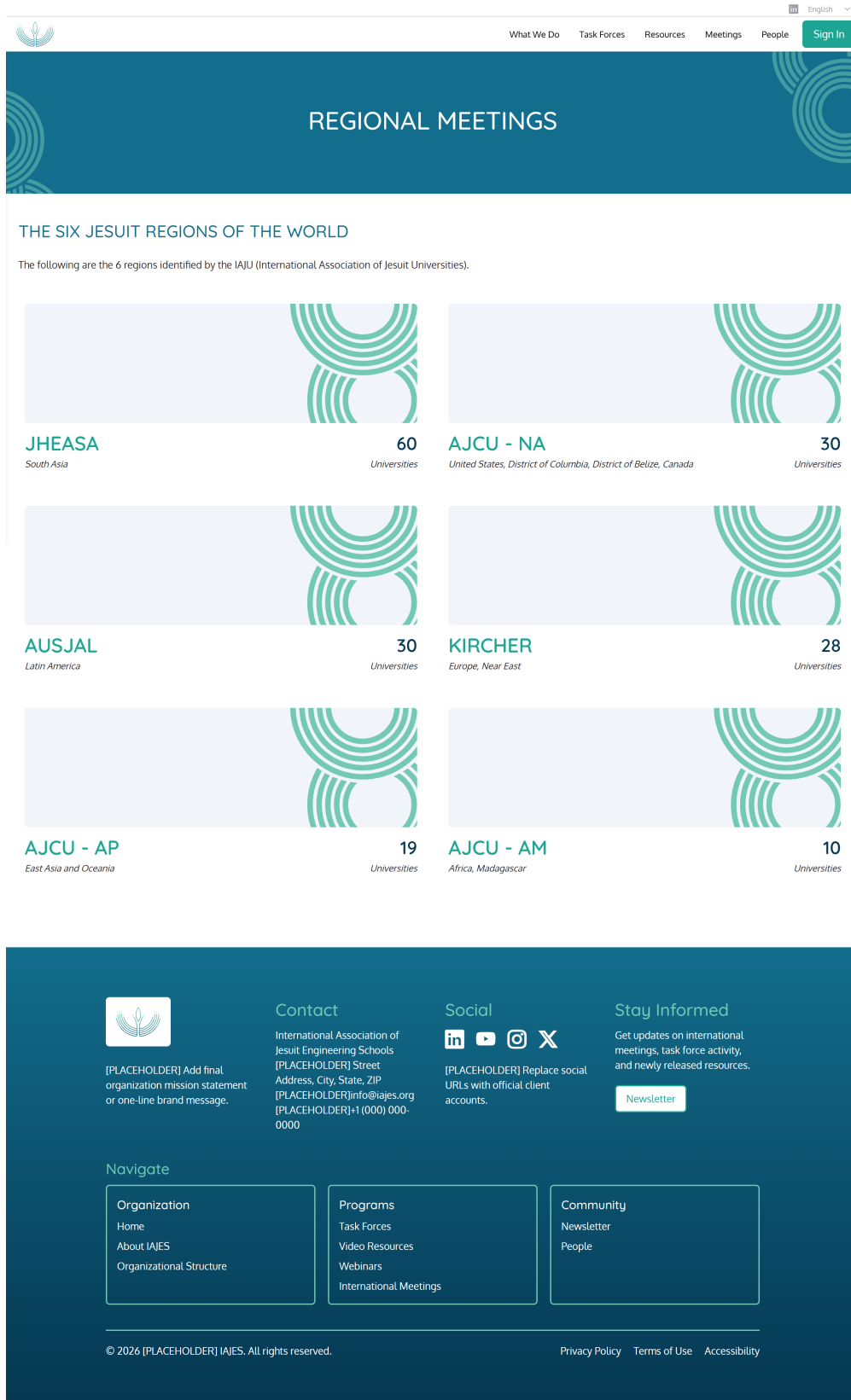


Figure 9.13: Regional Meetings (Region List)

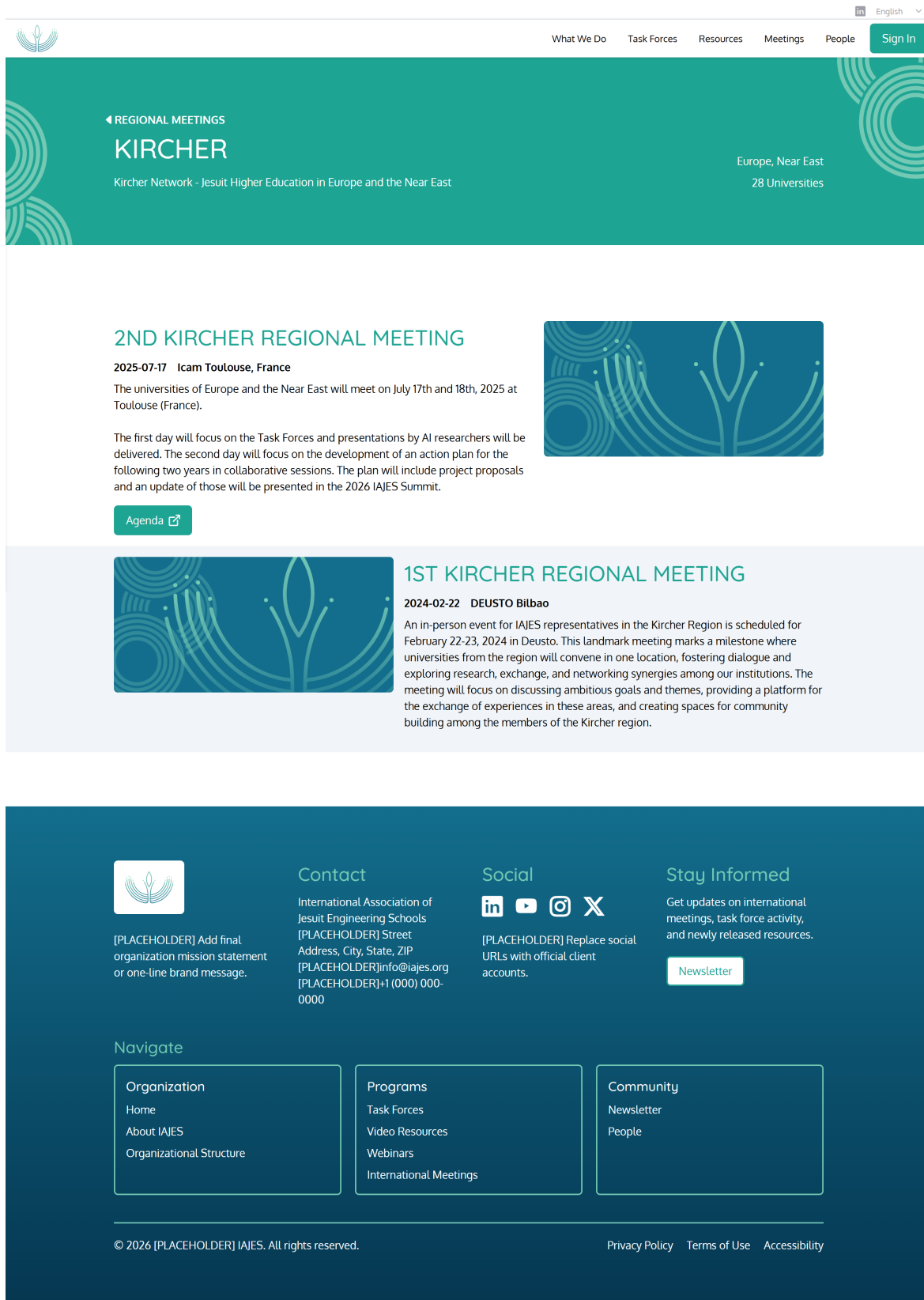


Figure 9.14: Regional Meetings (Meeting List)

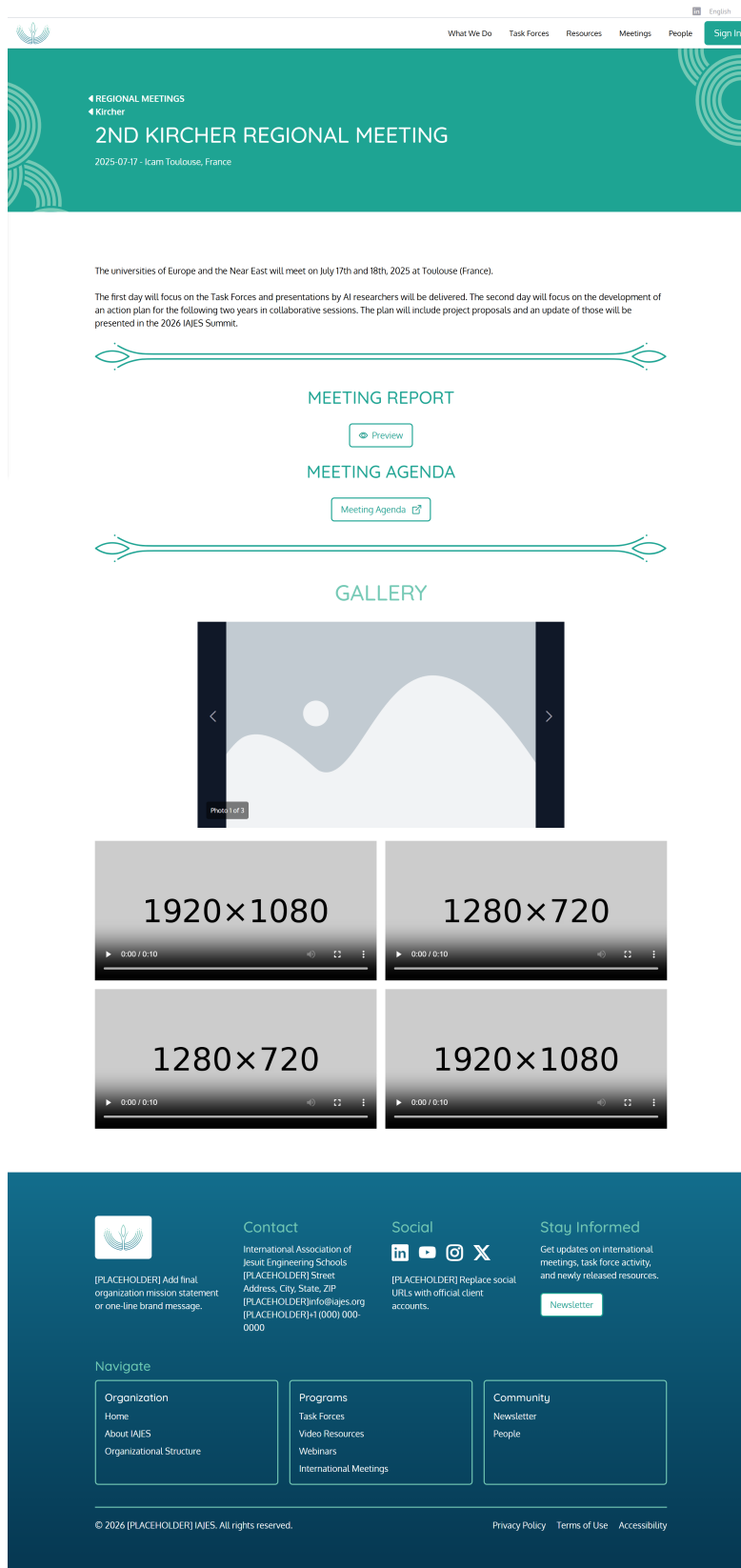


Figure 9.15: Regional Meetings Detail

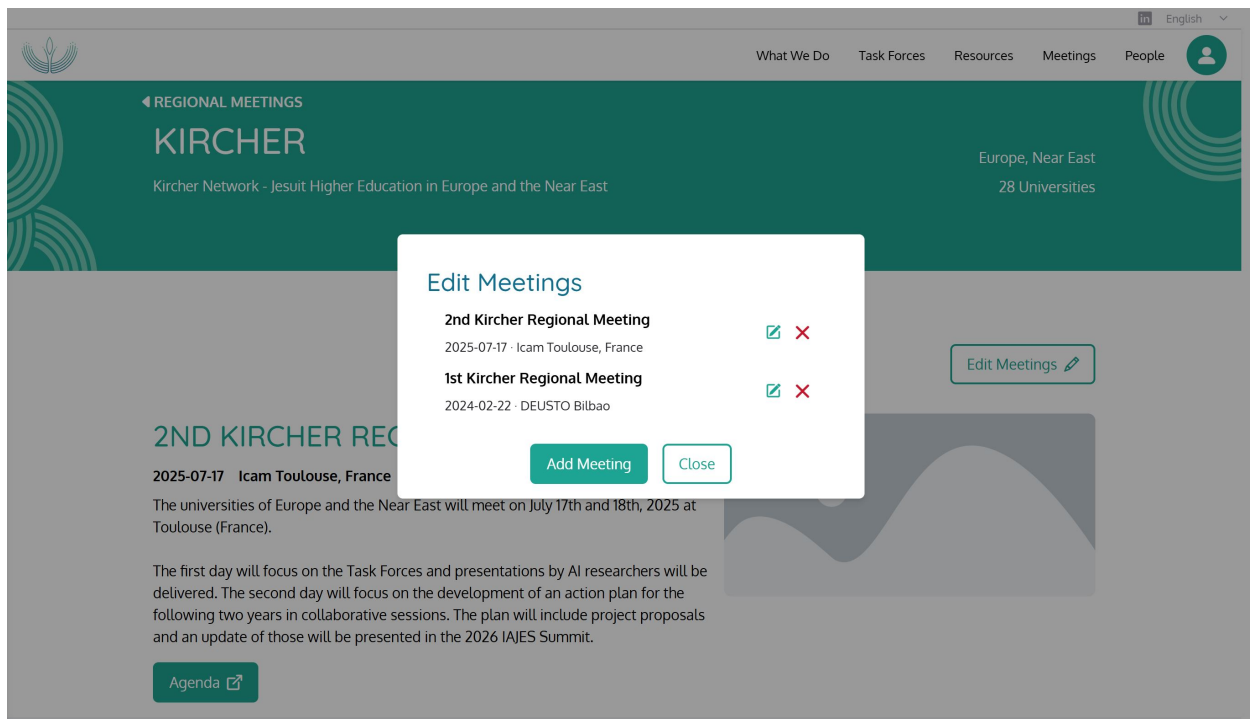


Figure 9.16: Regional Meetings Editing (1)

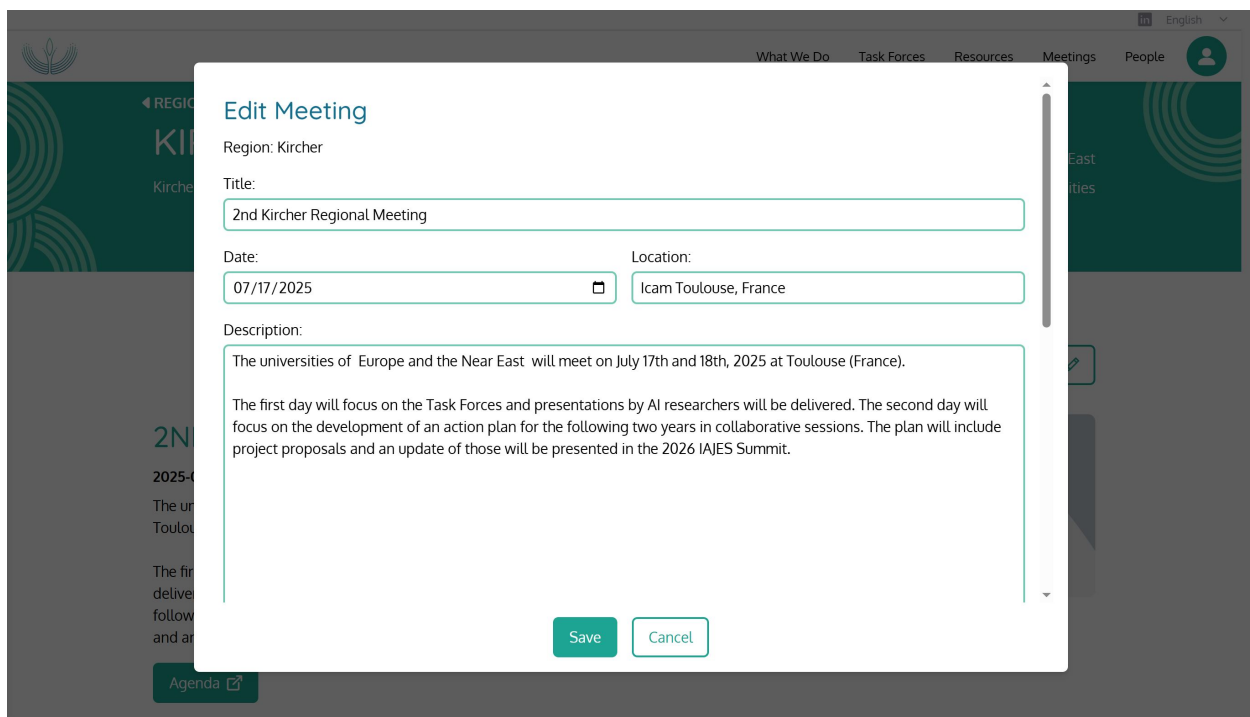


Figure 9.17: Regional Meetings Editing (2)

Agenda Link:

Meeting Report Link:

Agenda PDF: No file chosen

Meeting Report PDF:

Photos (upload new photos): No file chosen

Videos (upload new videos): No file chosen

1920×1080

1280×720

1280×720

1920×1080

Figure 9.18: Regional Meetings Editing (3)

9.2.9 Video Resources

The Video Resources page was introduced in the final product when we were notified that IAJES had added it to their original website. It acts as a flexible archive for recorded content, allowing administrators to upload and organize videos in a modular, easily editable format that was not fully envisioned in the original prototypes. (Figures 9.19 through 9.22)

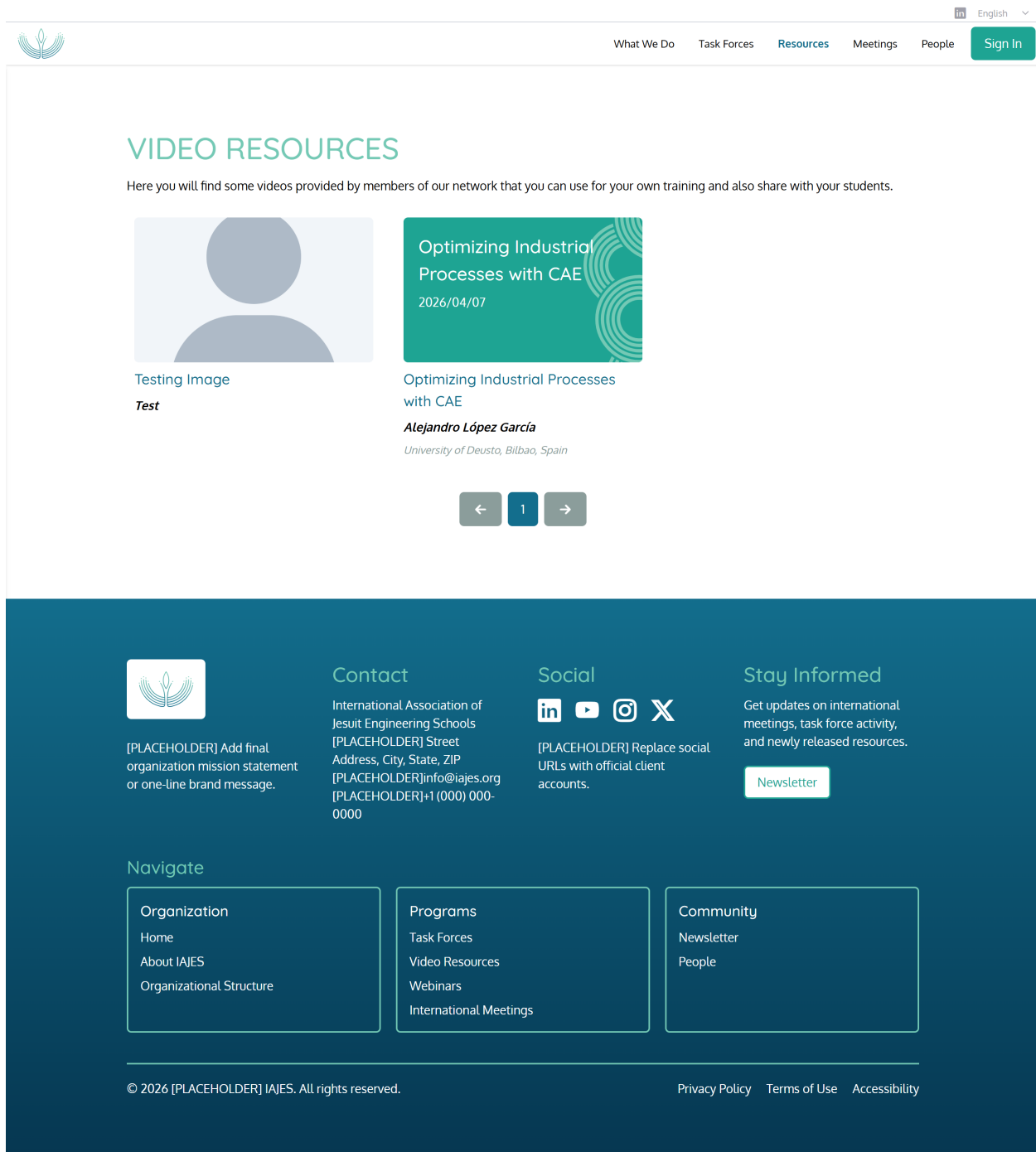


Figure 9.19: Video Resources



[What We Do](#)
[Task Forces](#)
[Resources](#)
[Meetings](#)
[People](#)
[Sign In](#)

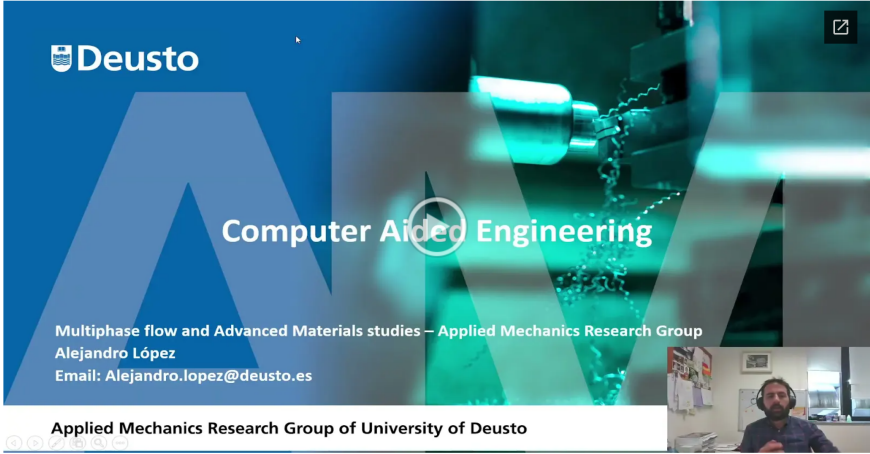


VIDEO RESOURCES

Optimizing Industrial Processes with CAE

By Alejandro López García 2026/04/07





Computer Aided Engineering

Multiphase flow and Advanced Materials studies – Applied Mechanics Research Group
Alejandro López
Email: Alejandro.lopez@deusto.es

Applied Mechanics Research Group of University of Deusto

How can we improve industrial processes with Computer Aided Engineering?

Alejandro López García
University of Deusto, Bilbao, Spain

Lecturer and Programme Leader at the University of Deusto - Research in Computational Mechanics, Fluid Mechanics, Granular matter and powder flow, additive manufacturing, Erosion processes



[PLACEHOLDER] Add final organization mission statement or one-line brand message.

Contact

International Association of Jesuit Engineering Schools
[PLACEHOLDER] Street
Address, City, State, ZIP
[PLACEHOLDER]info@iajes.org
[PLACEHOLDER]+1 (000) 000-0000

Social

[PLACEHOLDER] Replace social URLs with official client accounts.

Stay Informed

Get updates on international meetings, task force activity, and newly released resources.

Newsletter

Navigate

Organization

- Home
- About IAJES
- Organizational Structure

Programs

- Task Forces
- Video Resources
- Webinars
- International Meetings

Community

- Newsletter
- People

© 2026 [PLACEHOLDER] IAJES. All rights reserved.

[Privacy Policy](#)
[Terms of Use](#)
[Accessibility](#)

Figure 9.20: Video Resources Detail

100

Create new video resource

Video resource title:

Video resource thumbnail image: [Choose File](#) No file chosen

Video resource date:

Video resource link:
 This is the video link that will be embedded. Please only include the link and not the entire embed.

Video description:

Speaker Details

[Save](#) [Cancel](#)

Figure 9.21: Video Resources Editing (1)

Video description:

Speaker Details

Name: University:

Speaker image: [Choose File](#) No file chosen

Description:

[Save](#) [Cancel](#)

Figure 9.22: Video Resources Editing (2)

9.2.10 Webinars

The webinars section was split in two: one page maintains its original prototype design with editable webinars, very similar to video resources. Another page has been made into a single archive-style page of existing webinars, enabling streamlined storage and presentation of past webinar content while reducing structural complexity for administrators. (Figures 9.23 through 9.24)



Figure 9.23: Webinars


[What We Do](#)
[Task Forces](#)
[Resources](#)
[Meetings](#)
[People](#)
[Sign In](#)

WEBINARS ARCHIVE

Urban mobility in the Perspective of Integral Ecology & Digitalisation

05/23/2021

First "Research & Academic Cooperation" Task Force Webinar

05/19/2021

Embodying Integral Ecology in Learning Design

4/21/2021

Could TR Opened New Doors for Web-Cooperation

03/16/2021

Icam "Parcours Overt"

Engineering program

10/20/2020

Boston College Human Centered Engineering Program

07/08/2020

URBAN MOBILITY IN THE PERSPECTIVE OF INTEGRAL ECOLOGY & DIGITALISATION

05/23/2021

First webinar of the "Infrastructure" Task Force



Speakers

Nicolas Rincos Garcia
Javeriana University - Bogotá, Colombia
Professor of Industrial Engineering
Transport Observatory & Data Sciences

Marcus Moyers
Manchester Metropolitan University
Research fellow
Train infrastructure rehabilitation in Colombia

Karla Denis Castro Leite
Catholic University of Pernambuco - Brazil
Professor of Civil Engineering
The role of citizen participation in the implementation of public policies for sustainable mobility: the case of the bicycle office in Pernambuco, Brazil.

WEBINAR

Urban mobility in the Perspective of Integral Ecology & Digitalisation

05/23/2021



[PLACEHOLDER] Add final organization mission statement or one-line brand message.

Contact

International Association of
Brazil Engineering Schools
[PLACEHOLDER] Street
Address, City, State, ZIP
[PLACEHOLDER]@iajes.org
[PLACEHOLDER] (000) 000-0000

Social

[PLACEHOLDER] Register social
URLs with official client
accounts.

[Newsletter](#)

Stay Informed

Get updates on international
meetings, task force activity,
and newly released resources.

Navigate

Organization

Home

About IAJES

Organizational Structure

Programs

Task Forces

Video Resources

Webinars

International Meetings

Community

Newsletter

People

© 2016 [PLACEHOLDER] IAJES. All rights reserved.

[Privacy Policy](#) [Terms of Use](#) [Accessibility](#)

Figure 9.24: Webinars Archive

9.2.11 Task Forces

The task forces pages act as a central hub for IAJES' think-tank groups, and have expanded beyond the prototype to absorb projects and other organizational content into highly customizable, administrator-editable sections. (Figures 9.25 through 9.30)

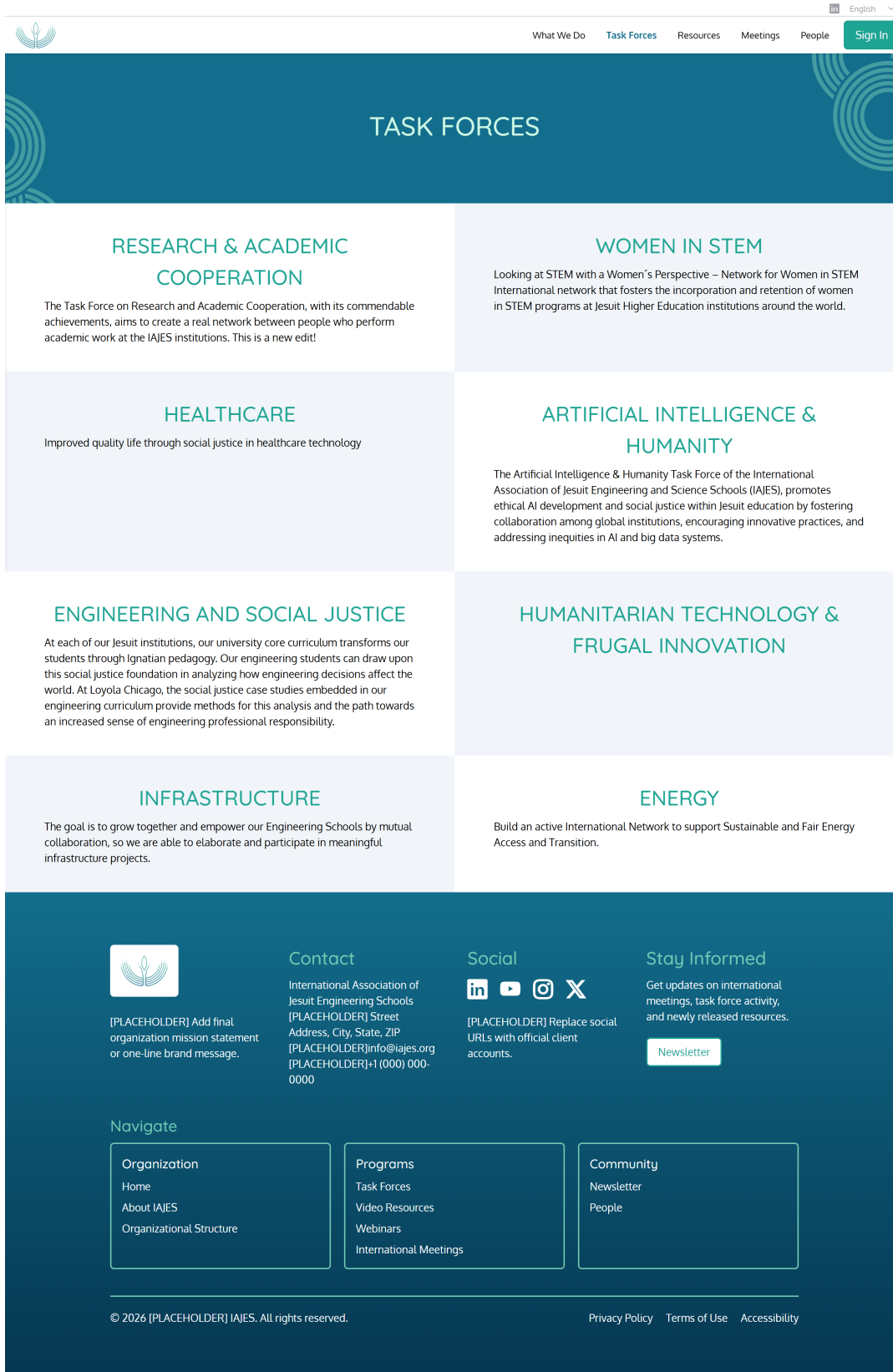


Figure 9.25: Task Force

106



Figure 9.26: Task Forces Detail

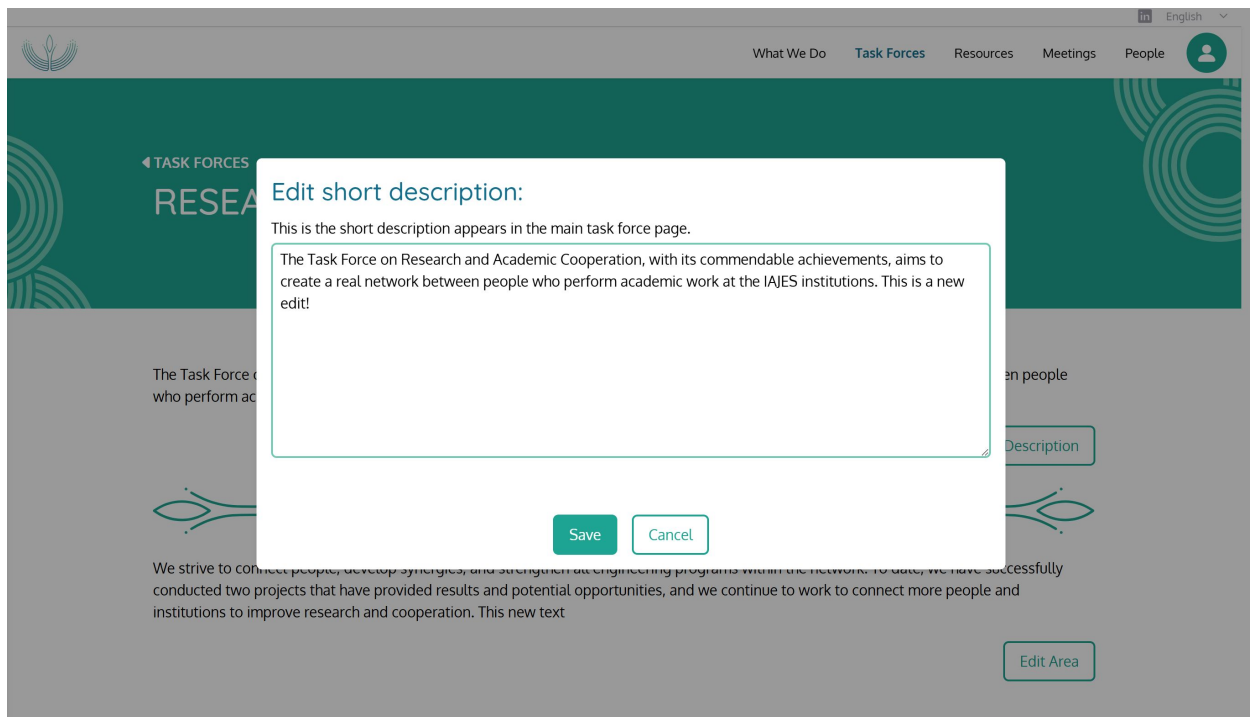


Figure 9.27: Task Forces Editing (1)

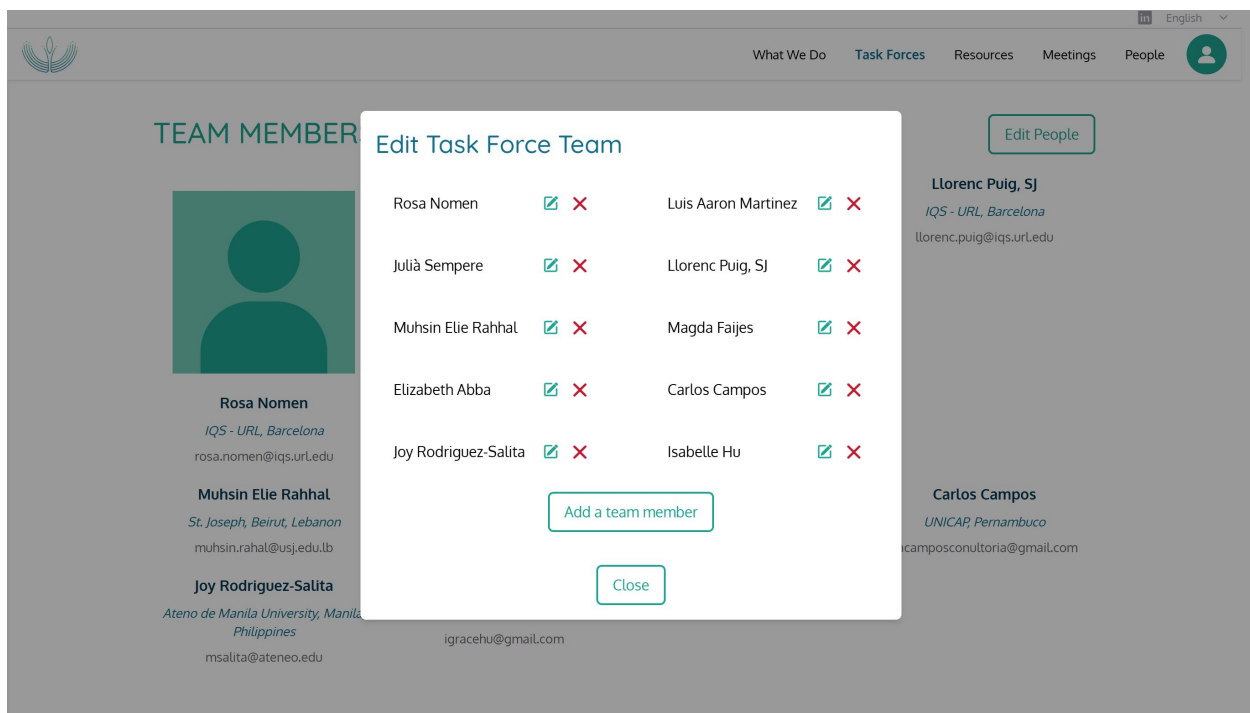


Figure 9.28: Task Forces Editing (2)

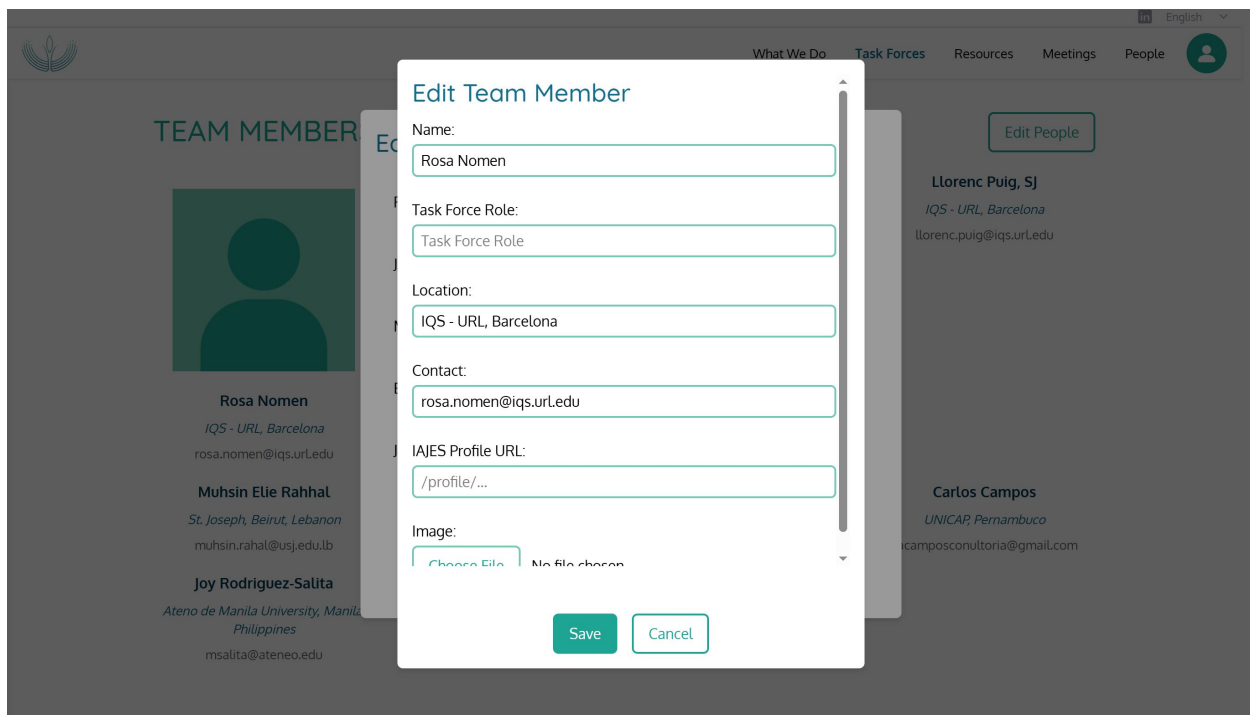


Figure 9.29: Task Forces Editing (3)

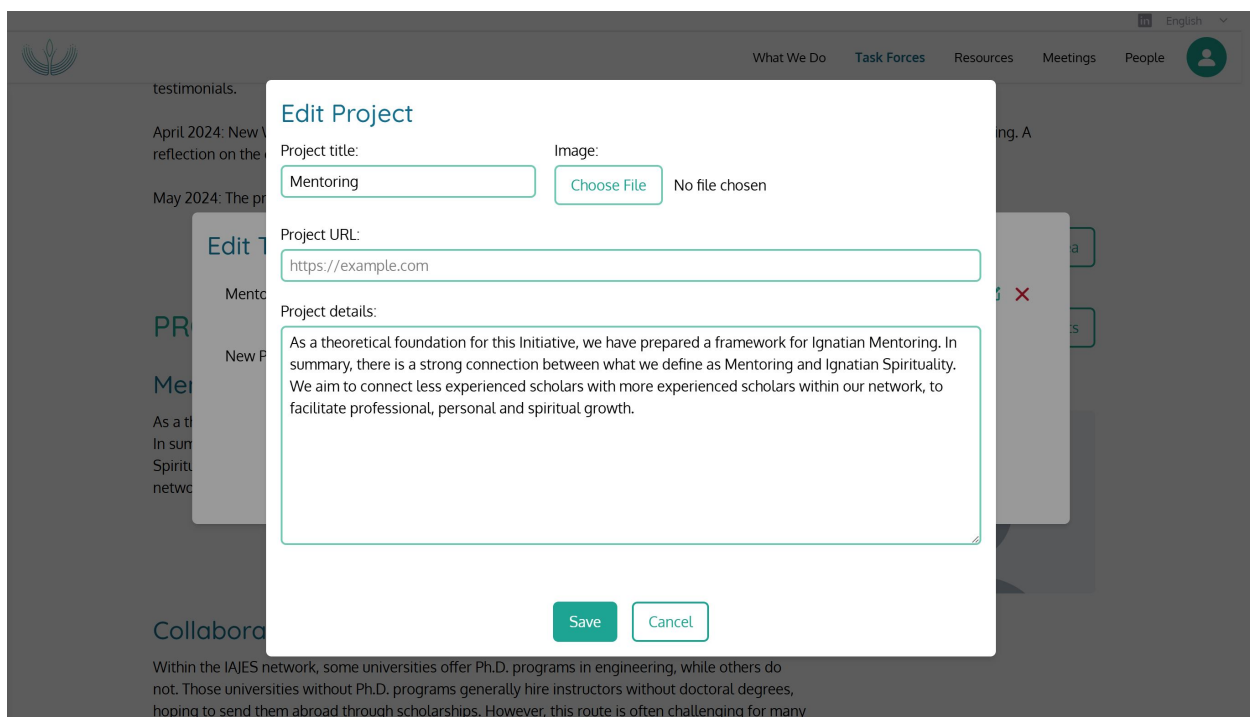


Figure 9.30: Task Forces Editing (4)

9.2.12 Newsletter and Newsletter Editing

This section implements both the public-facing newsletter viewer and archive, and the prototype vision of an admin-only editing interface. In the future, it will be connected to IAJES' mailing system and database to support real newsletter drafting and distribution workflows. (Figures 9.31 through 9.33)

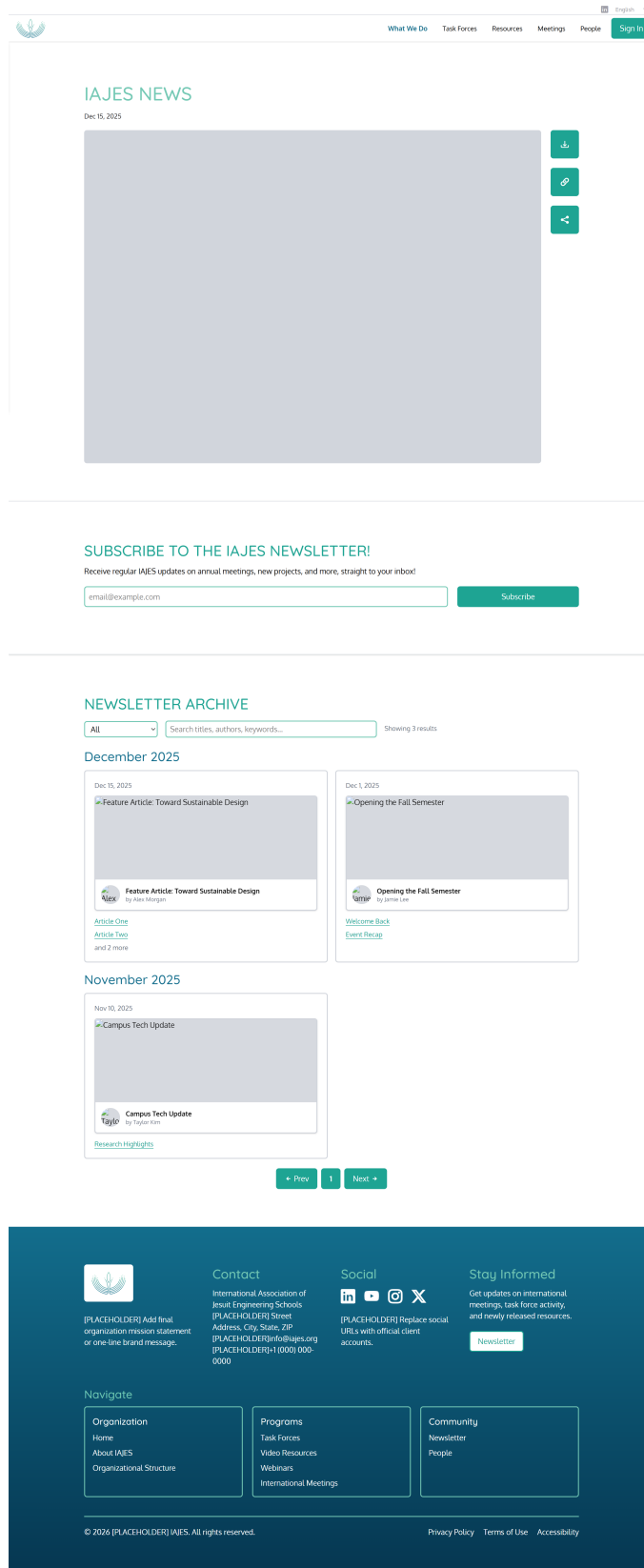


Figure 9.31: Newsletter



[What We Do](#)
[Task Forces](#)
[Resources](#)
[Meetings](#)
[People](#)
[Sign In](#)

NEWSLETTER DRAFTS

December Newsletter

Dec 20, 2025

Delete Draft

Add Article

Publish Newsletter

Drafted Feature · Featured

Last edited: Dec 19, 2025

Edit Delete

Draft Other 1

Last edited: Dec 18, 2025

Edit Delete

Draft Other 2

Last edited: Dec 17, 2025

Edit Delete

November Newsletter

Nov 25, 2025

Delete Draft

Add Article

Publish Newsletter

Another Draft Feature · Featured

Last edited: Nov 24, 2025

Edit Delete

Short piece

Last edited: Nov 23, 2025

Edit Delete

Create Newsletter +

Start from Upload ↗

← Prev

1

Next →



[PLACEHOLDER] Add final organization mission statement or one-line brand message.

Contact

International Association of Jesuit Engineering Schools
 [PLACEHOLDER] Street
 Address, City, State, ZIP
 [PLACEHOLDER]info@iajes.org
 [PLACEHOLDER]+1 (000) 000-0000

Social






[PLACEHOLDER] Replace social URLs with official client accounts.

Stay Informed

Get updates on international meetings, task force activity, and newly released resources.

Newsletter

Navigate

Organization
[Home](#)
[About IAJES](#)
[Organizational Structure](#)

Programs
[Task Forces](#)
[Video Resources](#)
[Webinars](#)
[International Meetings](#)

Community
[Newsletter](#)
[People](#)

© 2026 [PLACEHOLDER] IAJES. All rights reserved.

[Privacy Policy](#)
[Terms of Use](#)
[Accessibility](#)

Figure 9.32: Newsletter Draft Selection



What We DoTask ForcesResourcesMeetingsPeopleSign In

English

EDIT ARTICLE

SaveCancel

Title

Sample Article Title

Authors

 Alex Morgan Remove

Add author (mock) Add

Uploaded media

 Remove

Choose FileNo file chosen

Article content

Block 1Up Down Remove

Paragraph one.

Block 2Up Down Remove

Paragraph two.

Add block +

Figure 9.33: Newsletter Draft Article Editing

Chapter 10

Conclusion

In this project, we designed and developed a dynamic website for the International Association of Jesuit Engineering and Science Schools (IAJES) to replace its previous static, no-code platform with a scalable system that better supports communication, content management, and member engagement. The work involved translating organizational needs into concrete project requirements, implementing secure authentication and role permissions, creating tools for announcements, newsletters, task force pages, and resource galleries, and applying human-centered design principles through iterative prototyping and client feedback. Throughout development, we were guided by ethical considerations regarding data collection, privacy, and appropriate role access, ensuring that only essential member information was stored and that permissions aligned strictly with organizational responsibilities.

From this work, we learned that successful technical solutions for organizations like IAJES depend on understanding user roles, the workflows of administrators, and ethical responsibilities when serving a large international audience. This realization shaped our iterative design process, in which empathy for IAJES members and administrators drove our decisions and priorities. In addition to these design insights, we gained significant technical experience in using JavaScript to connect front-end components with a Supabase backend for real-time interactivity and database management, as well as adopting best practices for collaborative development through GitHub version control.

Our solution offers advantages in scalability, maintainability, and usability for non-technical administrators, while giving members meaningful ways to interact, discover one another, and share resources within the organization. The system is lightweight, accessible, and designed to be easily handed off to IAJES for long-term ownership.

However, the current system is still limited by its demonstration hosting configuration and the need for full deployment within IAJES infrastructure. Several features still require further development, including a completed newsletter generation interface and a centralized administrative dashboard that allows role management without directly editing the database. Additionally, the long-term success of the platform depends on continued adoption, content updates, and administrative engagement from IAJES itself. Later this summer, we will present the website to IAJES regional

leaders and members with the goal of having it approved as a full replacement for their existing site.

Future work includes full Docker-based integration with IAJES servers, refinement of administrative tools based on real user interaction, expansion of accessibility features, and continued evolution of the platform to match the needs of a growing global network of Jesuit engineering institutions. We plan to complete both server integration and all in-progress features before the IAJES summer conference so the organization can experience the platform at a more complete stage. Overall, our project demonstrates how thoughtful design, ethical responsibility, and technical implementation can come together to create a sustainable digital hub for international academic collaboration.

Chapter 11

Acknowledgments

We would like to express our gratitude to our advisor Dr. Silvia Figueira, whose guidance was helpful at every stage of this project. She provided positive encouragement and important feedback on our project and presentation. Her knowledge of human-centered design, software engineering, and general best project management practices was invaluable. She is gracious at all times, and offered us resources when we needed them most.

Our sincere thanks also go to Mr. Allan Báez Morales, from the SCU Frugal Innovation Hub. Our project would not have been possible without him, as we connected with IAJES through the Frugal Innovation Hub. He was instrumental in coordinating us with Carlos Rivas and Raúl Arturo Huerta, especially helping us with organization and communication when we first started this project in the summer. He also gave us critical feedback on our project. As always, he is a joy to work with, and we are deeply appreciative of his investment in our project every step of the way. Additionally, thanks to him and the Frugal Innovation Hub for funding temporary hosting of our project, as this truly contributed to the success of our demonstration.

We also want to deeply thank Carlos Rivas, President of IAJES, and Raúl Arturo Huerta, Vice President of IAJES. As the clients of our project, we want to express our gratitude for allowing us to take on this responsibility. This project has been a great experience, and our team is so proud to make a positive impact on the IAJES organization. Thank you for your time, critical feedback, invaluable suggestions, patience, and positive attitudes over the past year.

Lastly, we acknowledge the use of free or affordable third-party software such as GitHub, Supabase, and Firebase, that made development of this website possible.

Bibliography

- [1] ACM, “Software Engineering Code of Ethics and Professional Practice,” *ACM*, 2024. [Online]. Available: <https://www.acm.org/code-of-ethics/software-engineering-code>
- [2] California State Legislature, “California Consumer Privacy Act of 2018,” Cal. Civ. Code § 1798.100 *et seq.*, 2018. [Online]. Available: <https://oag.ca.gov/privacy/ccpa>
- [3] European Parliament and Council of the European Union, “Regulation (EU) 2016/679 on the protection of natural persons with regard to the processing of personal data (General Data Protection Regulation),” *Official Journal of the European Union*, L 119, pp. 1–88, Apr. 2016. [Online]. Available: <https://eur-lex.europa.eu/eli/reg/2016/679/oj>
- [4] L. Floridi and J. Cowls, “A Unified Framework of Five Principles for AI in Society,” *Harvard Data Science Review*, vol. 1, no. 1, Jul. 2019, doi: 10.1162/99608f92.8cd550d1.
- [5] Google, “Google Sites: Build & Host Business Websites,” *Google Workspace*. [Online]. Available: <https://workspace.google.com/products/sites/>
- [6] OWASP Foundation, “OWASP Top 10: The Ten Most Critical Web Application Security Risks,” 2021. [Online]. Available: <https://owasp.org/Top10/>
- [7] World Wide Web Consortium (W3C), “Web Content Accessibility Guidelines (WCAG) 2.1,” W3C Recommendation, Jun. 5, 2018. [Online]. Available: <https://www.w3.org/TR/WCAG21/>
- [8] M. Bilucaglia *et al.*, “Looking through blue glasses: bioelectrical measures to assess the awakening after a calm situation,” in *Proc. 41st Annu. Int. Conf. IEEE Engineering in Medicine and Biology Society (EMBC)*, Berlin, Germany, 2019, pp. 526–529, doi: 10.1109/EMBC.2019.8856486.
- [9] Object Management Group, “Unified Modeling Language Specification, Version 2.5.1,” OMG, Dec. 2017. [Online]. Available: <https://www.omg.org/spec/UML/>

- [10] Meta Open Source, “React—The library for web and native user interfaces,” *react.dev*, 2024. [Online]. Available: <https://react.dev/>
- [11] Remix Software, “React Router—Declarative routing for React applications,” *reactrouter.com*. [Online]. Available: <https://reactrouter.com/>
- [12] Tailwind Labs, “Tailwind CSS Documentation,” *tailwindcss.com*. [Online]. Available: <https://tailwindcss.com/docs>
- [13] Supabase, “Supabase Documentation: The Open Source Firebase Alternative,” *supabase.com*. [Online]. Available: <https://supabase.com/docs>
- [14] Google, “Firebase App Hosting,” *firebase.google.com*. [Online]. Available: <https://firebase.google.com/docs/app-hosting>
- [15] The PostgreSQL Global Development Group, “PostgreSQL Documentation,” *postgresql.org*. [Online]. Available: <https://www.postgresql.org/docs/>
- [16] Figma Inc., “Figma: The Collaborative Interface Design Tool,” *figma.com*. [Online]. Available: <https://www.figma.com/>
- [17] J. Nielsen, “10 Usability Heuristics for User Interface Design,” Nielsen Norman Group, Nov. 15, 2020. [Online]. Available: <https://www.nngroup.com/articles/ten-usability-heuristics/>
- [18] D. A. Norman, *The Design of Everyday Things*, revised and expanded ed. New York, NY, USA: Basic Books, 2013.
- [19] International Organization for Standardization, *Ergonomics of human-system interaction—Part 210: Human-centred design for interactive systems*, ISO 9241-210:2019, 2019.
- [20] K. Beck *et al.*, “Manifesto for Agile Software Development,” 2001. [Online]. Available: <https://agilemanifesto.org/>
- [21] K. Schwaber and J. Sutherland, “The Scrum Guide: The Definitive Guide to Scrum,” Nov. 2020. [Online]. Available: <https://scrumguides.org/scrum-guide.html>
- [22] D. F. Ferraiolo, R. Sandhu, S. Gavrila, D. R. Kuhn, and R. Chandramouli, “Proposed NIST standard for role-based access control,” *ACM Trans. Inf. Syst. Secur.*, vol. 4, no. 3, pp. 224–274, Aug. 2001, doi: 10.1145/501978.501980.

- [23] D. Hardt, Ed., “The OAuth 2.0 Authorization Framework,” Internet Engineering Task Force, RFC 6749, Oct. 2012. [Online]. Available: <https://www.rfc-editor.org/rfc/rfc6749>
- [24] E. Marcotte, “Responsive Web Design,” *A List Apart*, May 25, 2010. [Online]. Available: <https://alistapart.com/article/responsive-web-design/>
- [25] J. J. Garrett, *The Elements of User Experience: User-Centered Design for the Web and Beyond*, 2nd ed. Berkeley, CA, USA: New Riders, 2011.

Appendix A

Code Files

A.1 root.jsx

```
1 import {
2   isRouteErrorResponse,
3   Links,
4   Meta,
5   Outlet,
6   Scripts,
7   ScrollRestoration,
8 } from "react-router";
9
10 import "./app.css";
11
12 export const links = () => [
13   { rel: "preconnect", href: "https://fonts.googleapis.com" },
14   {
15     rel: "preconnect",
16     href: "https://fonts.gstatic.com",
17     crossOrigin: "anonymous",
18   },
19   {
20     rel: "stylesheet",
21     href: "https://fonts.googleapis.com/css2?family=Inter:ital,opsz,wght@0,14..32,100..900;1,14..32,100..900&display=swap",
22   },
23   {
24     rel: "stylesheet",
25     href: "https://fonts.googleapis.com/css2?family=Oxygen:wght@300;400;700&display=swap",
26   },
27   {
28     rel: "stylesheet",
29     href: "https://fonts.googleapis.com/css2?family=Oxygen:wght@300;400;700&family=Quicksand:wght@300..700&display=swap",
30   },
31   {
32     rel: "stylesheet",
33     href: "https://cdn.jsdelivr.net/npm/bootstrap-icons@1.13.1/font/bootstrap-icons.min.css"
34   },
35 ];
36
37 export function Layout({ children }) {
38   return (
39     <html lang="en" className="scroll-smooth">
40       <head>
41         <meta charSet="utf-8" />
```

```

42     <meta name="viewport" content="width=device-width, initial-scale=1" />
43     <Meta />
44     <Links />
45   </head>
46   <body>
47     {children}
48     <ScrollRestoration />
49     <Scripts />
50   </body>
51 </html>
52 );
53 }
54
55 export default function App() {
56   return <Outlet />;
57 }
58
59 export function ErrorBoundary({ error }) {
60   let message = "Oops!";
61   let details = "An unexpected error occurred.";
62   let stack;
63
64   if (isRouteErrorResponse(error)) {
65     message = error.status === 404 ? "404" : "Error";
66     details =
67       error.status === 404
68         ? "The requested page could not be found."
69         : error.statusText || details;
70   } else if (import.meta.env.DEV && error && error instanceof Error) {
71     details = error.message;
72     stack = error.stack;
73   }
74
75   return (
76     <main className="pt-16 p-4 container mx-auto">
77       <h1>{message}</h1>
78       <p>{details}</p>
79       {stack && (
80         <pre className="w-full p-4 overflow-x-auto">
81           <code>{stack}</code>
82         </pre>
83       )}
84     </main>
85   );
86 }

```

A.2 styleguide.jsx

```

1 import { useState } from 'react';
2 import { redirect } from "react-router";
3 import { Menu } from "../components/menu";
4 import { Footer } from "../components/footer";
5 import { Popup, PopupForm } from "../components/popup";
6 import { Pagination } from "../components/pagination";
7 import { Break, H1Middle, H1Left, H2Middle, H2Left, Banner } from "../components/graphics";
8 import { updateRequired } from "../helpers/form";
9
10 export function meta() {
11   return [
12     { title: "Style Guide" },
13     { name: "", content: "" },
14   ];

```



```

15 }
16
17 export function action({ url }) {
18     return redirect(`${url}`);
19 }
20
21 export default function StyleGuide() {
22
23     // PAGINATION -----
24     const [pg1, setPg1] = useState(0);
25     const [pg2, setPg2] = useState(0);
26
27     const [formMode, setFormMode] = useState("");
28
29     // POPUP -----
30     // The Popup component takes three parameters:
31     // id - Unique ID for the popup. Ensures that popups from different components do not overlap
32     // There should only be one Popup component used in a component.
33     // className
34     // show - Determines if the popup is visible or not
35     // setShow - set function for show
36     // buttons - Optional. An array of objects with properties 'text' and 'onclick'
37     // text - Required. A string with the button text
38     // onclick - Required. A function that clicking the button will execute
39     // stayOnBlur - Optional. Defaults to false. Determines if clicking outside the popup will
40     // close the popup or not
41
42     const [showPopup1, setShowPopup1] = useState(false);
43     const [showPopup2, setShowPopup2] = useState(false);
44     const [showPopupForm, setShowPopupForm] = useState(false);
45     const [popupFormResults, setPopupFormResults] = useState("");
46
47     const buttons = [{text: "Action", onclick: () => console.log("Action")}]
48
49     // REQUIRED FIELDS Quickguide to functions
50     // handleShowPopupForm - Resets required inputs
51     // validate PopupForm
52     // - Checks if required inputs are filled in + other validations
53     // - Extracts formData
54     // checkEmpty - Puts an input back to normal once it is filled in
55
56     // formRequired: An object containing the current state of all required fields, true if they
57     // are current required, false if not
58     const [formRequired, setFormRequired] = useState({popupRequired: false})
59
60     function handleShowPopupForm() {
61         // This resets formRequired so that there are no error messages when the form is reloaded
62         setFormRequired({popupRequired: false})
63         setShowPopupForm(true);
64     }
65
66     function validatePopupForm(formData) {
67         let isValidated = true;
68         const isRequired = {
69             popupRequired: formData.get('popupRequired') === (null || "")
70         }
71         for (let value of Object.values(isRequired)) {
72             if (value) {
73                 isValidated = false;
74                 break;
75             }
76         }
77         if (!isValidated) {
78             setFormRequired(isRequired);
79             return false;
80         }
81     }

```

```

78     }
79
80     // Get formData
81     console.log(formData);
82     setPopupFormResults("Text Input: " + formData.get("popupText") + "\n" + "Required Text
Input: " + formData.get("popupRequired") + "\n")
83     // setShowPopupForm(false);
84     return true;
85 }
86
87 // checkEmpty uses updatedRequired from helpers/form.js
88 // checkEmpty is called from the input element itself. Go to the input with id popupRequired
to see how this works.
89 // PARAMETERS
90 //   value - User inputted value = e.target.value
91 //   inputName - The name of the input in formRequired, NOT the input element name
92 //   I suggest you make the formRequired variable names and the input elements
the same to avoid confusion
93 function checkEmpty(value, inputName) {
94     const updatedFormRequired = updateRequired(value, inputName, formRequired);
95     if (updatedFormRequired !== formRequired) {
96         setFormRequired(updatedFormRequired);
97     }
98 }
99
100
101 return (
102     <>
103     <Popup id="style1" show={showPopup1} setShow={setShowPopup1} >
104         <p>This is the default popup</p>
105     </Popup>
106     <Popup id="style2" show={showPopup2} setShow={setShowPopup2} buttons={buttons}
stayOnBlur>
107         <p>This is a popup that has a custom action button and will not exit when clicked
off of.</p>
108     </Popup>
109     <PopupForm id="styleform" className="w-200 relative" show={showPopupForm} setShow={
setShowPopupForm} validate={validatePopupForm}>
110         <p>This is a popup form. Popup forms only have two buttons, 'Save' and 'Cancel'.
Popup forms also do not close upon blur.</p>
111         <p>PopupForms include a form, any elements within the PopupForm goes into the
form. You cannot nest PopupForms.</p>
112         <p>PopupForm requires a validate function that takes a parameter 'formData'. When
'Save' is clicked, the form is submitted and the contents of the form can be accessed in the
validate function through 'formData'. Please remember that only input fields with 'name'
attributes will be recognized.</p>
113         <p>The parameter 'formData' is a <a href="https://developer.mozilla.org/en-US/
docs/Web/API/FormData">FormData</a> object. You can use formData to get the submitted form
values with formData.get('input-name')</p>
114         <p><strong>If there are any required fields to fill in, please refer to source
code for implementing that part.</strong></p>
115         <br />
116         <p>This sample popup form takes two text fields and prints them on the console
upon saving.</p>
117         <br />
118         <p>{popupFormResults}</p>
119         <br />
120         <label for="popupText">Text Input:</label><br />
121         <input id="popupText" name="popupText" type="text" className={"input input-text w
-full"} placeholder="Enter text here..." defaultValue="Default Value" />
122         <br /><br />
123         <label for="popupRequired">Required Text Input:</label><br />
124         <input id="popupRequired" name="popupRequired" type="text"
125             className={"input input-text w-full " + (formRequired?.popupRequired && "
input-required")}>

```

```

126         placeholder="Enter text here..."
127         onChange={(e) => checkEmpty(e.target.value, "popupRequired")} />
128         <div className="input-error">This field is required.</div>
129         <br/>
130     </PopupForm>
131
132     <Menu />
133     <div className="lg:px-40 px-10 py-20 duration-200">
134         <h1>Style Guide</h1>
135         <p>Note that we are using TailwindCSS, so unless needed, sizing and layout will
be done with TailwindCSS variables and presets.</p>
136         <div className="mt-10">
137             <h2>Fonts</h2>
138             <h1>Heading 1</h1>
139             <h2>Heading 2</h2>
140             <h3>Heading 3</h3>
141             <h4>Heading 4</h4>
142             <h5>Heading 5</h5>
143             <h6>Heading 6</h6>
144             <p> Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nunc in nisi
venenatis, faucibus lorem eu, lobortis magna. Donec at ante vel arcu mattis sagittis.
Curabitur efficitur ex aliquam sapien dictum fringilla. Vestibulum sagittis sit amet quam in
congue. Cras consequat, nibh auctor dapibus ultrices, arcu ex lobortis mi, eu pharetra sem
metus eget mi. Sed eget bibendum ipsum. Vivamus bibendum nulla hendrerit ligula facilisis,
non euismod mi cursus. Proin nec risus vel elit ullamcorper venenatis. Donec lacinia mi eu
nunc vehicula, quis ullamcorper ipsum suscipit. </p>
145             <p>Example for <a href="">inline links</a></p>
146             <ul>
147                 <li>Unordered list item 1</li>
148                 <li>Unordered list item 2</li>
149                 <li>Unordered list item 3</li>
150             </ul>
151             <ol>
152                 <li>Ordered list item 1</li>
153                 <li>Ordered list item 2</li>
154                 <li>Ordered list item 3</li>
155             </ol>
156         </div>
157
158         <div className="mt-10">
159             <h2>Colors</h2>
160             <p>Custom colors have been defined in the theme, so they can be used like
TailwindCSS colors. For example, having a primary light background can be done using the
class <span className="text-primary-dark">bg-primary-light</span>.</p>
161             <div className="grid lg:grid-cols-4 grid-cols-2 gap-5 my-4">
162                 <div className="px-5 py-7 rounded-md text-center bg-primary-light">
163                     <p>
164                         #74C9B5
165                         <br />
166                         color-primary-light
167                     </p>
168                 </div>
169                 <div className="px-5 py-7 rounded-md text-center bg-primary-dark text-
white">
170                     <p>
171                         #1EA493
172                         <br />
173                         color-primary-dark
174                     </p>
175                 </div>
176                 <div className="px-5 py-7 rounded-md text-center bg-secondary-light text-
white">
177                     <p>
178                         #136E8D
179                         <br />

```

```

180         color-secondary-light
181     </p>
182 </div>
183 <div className="px-5 py-7 rounded-md text-center bg-secondary-dark text-
white">
184     <p>
185         #063751
186         <br />
187         color-secondary-dark
188     </p>
189 </div>
190 <div className="px-5 py-7 rounded-md text-center bg-white border-2 border
-gray-light">
191     <p>
192         #FFFFFF
193         <br />
194         White
195         <br />
196         color-white
197     </p>
198 </div>
199 <div className="px-5 py-7 rounded-md text-center bg-gray-light">
200     <p>
201         #D1D5DC
202         <br />
203         Light Gray (Gray 300)
204         <br />
205         color-gray-light
206     </p>
207 </div>
208 <div className="px-5 py-7 rounded-md text-center bg-gray-dark text-white"
>
209     <p>
210         #27272A
211         <br />
212         Dark Gray (Zinc 800)
213         <br />
214         color-gray-dark
215     </p>
216 </div>
217 <div className="px-5 py-7 rounded-md text-center bg-black text-white">
218     <p>
219         #000000
220         <br />
221         Black
222         <br />
223         color-black
224     </p>
225 </div>
226 <div className="px-5 py-7 rounded-md text-center bg-primary-extralight">
227     <p>
228         #ECFDF5
229         <br />
230         Primary Extra Light
231         <br />
232         color-primary-extralight
233         <br />
234         <span className="text-xs">(Applied automatically to checkbox and
radio buttons on hover)</span>
235     </p>
236 </div>
237 <div className="px-5 py-7 rounded-md text-center bg-disabled-light text-
white">
238     <p>
239         #8B9D9B

```

```

240         <br />
241         Disabled Light
242         <br />
243         color-disabled-light
244         <br />
245         <span className="text-xs">(Applied automatically to disabled
buttons)</span>
246     </p>
247 </div>
248 <div className="px-5 py-7 rounded-md text-center bg-disabled-dark text-
white">
249     <p>
250         #404547
251         <br />
252         Disabled Dark
253         <br />
254         color-disabled-dark
255         <br />
256         <span className="text-xs">(Meant for disabled buttons in a future
dark mode)</span>
257     </p>
258 </div>
259 <div className="px-5 py-7 rounded-md text-center bg-selection-highlight
text-white">
260     <p>
261         #4388A1
262         <br />
263         Selection Highlight
264         <br />
265         color-selection-highlight
266         <br />
267         <span className="text-xs">(Applied automatically to text
selection)</span>
268     </p>
269 </div>
270 <div className="px-5 py-7 rounded-md text-center bg-error text-white">
271     <p>
272         #c2162d
273         <br />
274         Error
275         <br />
276         color-error
277         <br />
278     </p>
279 </div>
280 <div className="px-5 py-7 rounded-md text-center bg-error-dark text-white
">
281     <p>
282         #a01225
283         <br />
284         Error Dark
285         <br />
286         color-error-dark
287         <br />
288         <span className="text-xs">(Applied to delete buttons on hover)</
span>
289     </p>
290 </div>
291 <div className="px-5 py-7 rounded-md text-center bg-error-extradark text-
white">
292     <p>
293         #7a0e1d
294         <br />
295         Error Extra Dark
296         <br />

```

```

297         color-error-extradark
298         <br />
299         <span className="text-xs">(Applied to delete buttons on click)</
span>
300     </p>
301 </div>
302
303 </div>
304 </div>
305
306 <div className="mt-10">
307     <h2>Buttons</h2>
308     <div>
309         <button className="button mr-5 mb-5">.button</button>
310         <button className="button button-light mr-5 mb-5">.button .button-light</
button>
311         <br />
312         <button className="button button-delete mr-5 mb-5">.button .button-delete
</button>
313         <button className="button button-light button-delete mr-5 mb-5">.button .
button-light .button-delete</button>
314         <br />
315         <button className="button button-disabled mr-5 mb-5">.button .button-
disabled</button>
316         <button className="button button-light button-disabled mr-5 mb-5">.button
.button-light .button-disabled</button>
317         <br />
318         <button className="button mr-5 mb-5" disabled>.button disabled</button>
319         <button className="button button-light mr-5 mb-5" disabled>.button .
button-light disabled</button>
320     </div>
321     <p>Using either <span className="text-primary-dark">.button-disabled</span>
or the <span className="text-primary-dark">disabled</span> attribute will have the same
result.</p>
322 </div>
323
324 <div className="mt-10">
325     <h2>Spacing</h2>
326     <div className="grid lg:grid-cols-3 sm:grid-cols-2 grid-cols-1 gap-5">
327         <div>
328             <h3>Margin & Gap</h3>
329             <p>Margin between buttons: <span className="text-primary-dark">.m-5</
span></p>
330             <p>Grid gap: <span className="text-primary-dark">.gap-5</span> or <
span className="text-primary-dark">.gap-10</span></p>
331         </div>
332         <div>
333             <h3>Padding</h3>
334             <p>Page Content: <span className="text-primary-dark">lg:px-40 px-10 .
py-20 .duration-200</span></p>
335             <p>Cards and Buttons: <span className="text-primary-dark">.p-4</span
></p>
336         </div>
337         <div>
338             <h3>Border</h3>
339             <p>Border Radius: <span className="text-primary-dark">.rounded-md</
span></p>
340             <p>Border Width: <span className="text-primary-dark">.border-2</span>
(2px)</p>
341         </div>
342     </div>
343 </div>
344
345 <div className="mt-10">
346     <h2>Forms</h2>

```

```

347         <p>Checkboxes and radios require more structural attention compared to
textboxes, so please refer to the source code to implement them properly.</p>
348         <div className="flex space-x-5">
349             <button className={"button " + (formMode == "" && "button-light")}
onClick={() => setFormMode("")}>Default</button>
350             <button className={"button " + (formMode == "input-disabled" && "button-
light")} onClick={() => setFormMode("input-disabled")}>Disabled</button>
351             <button className={"button " + (formMode == "input-required" && "button-
light")} onClick={() => setFormMode("input-required")}>Required</button>
352         </div>
353         <div className="grid lg:grid-cols-2 grid-cols-1 gap-x-10 gap-y-15 my-5">
354             <div className="w-full">
355                 <label htmlFor="form-text">Text Input:</label><br />
356                 <input id="form-text" type="text" className={"input input-text w-full
" + formMode} placeholder="Enter text here..." />
357                 <div className="input-error">This field is required.</div>
358                 <br /><br />
359                 <label htmlFor="form-textarea">Textarea Input:</label><br />
360                 <textarea id="form-textarea" className={"input input-text w-full " +
formMode} placeholder="Enter text here..." />
361                 <div className="input-error">Error messages must come after the input
.</div>
362                 <br /><br />
363                 <label htmlFor="form-select">Select:</label><br />
364                 <select id="form-select" className={"input input-text w-full " +
formMode}>
365                     <option value="1">1</option>
366                     <option value="2">2</option>
367                     <option value="3">3</option>
368                     <option value="4">4</option>
369                 </select>
370                 <div className="input-error">They are set to position: absolute to
avoid changing page layout.</div>
371             </div>
372             <div>
373                 <p>Checkbox</p>
374                 <label htmlFor="checkbox1" className="checkbox">
375                     <input id="checkbox1" type="checkbox" className={formMode} /><p>
Checkbox 1</p>
376                 <div className="input-error">This field is required.</div>
377                 </label>
378                 <label htmlFor="checkbox2" className="checkbox">
379                     <input id="checkbox2" type="checkbox" className={formMode} /><p>
Checkbox 2</p>
380                 </label>
381                 <label htmlFor="checkbox3" className="checkbox">
382                     <input id="checkbox3" type="checkbox" className={formMode} /><p>
Checkbox 3</p>
383                 </label>
384                 <br /><br />
385                 <p>Radio</p>
386                 <label htmlFor="radio1" className="radio">
387                     <input id="radio1" type="radio" name="radio" className={formMode}
/><p>Radio 1</p>
388                 <div className="input-error">This field is required.</div>
389                 </label>
390                 <label htmlFor="radio2" className="radio">
391                     <input id="radio2" type="radio" name="radio" className={formMode}
/><p>Radio 2</p>
392                 </label>
393                 <label htmlFor="radio3" className="radio">
394                     <input id="radio3" type="radio" name="radio" className={formMode}
/><p>Radio 3</p>
395                 </label>
396             </div>

```

```

397         </label>
398         <br /><br />
399
400         <p>Radio Buttons (need to be contained within an element)</p>
401         <div>
402             <label htmlFor="radiobutton1" className="radio-button">
403                 <input id="radiobutton1" type="radio" name="radiobutton"
404                 className={formMode} /><p>Radio 1</p>
405                 <div className="input-error">This field is required.</div>
406             </label>
407             <label htmlFor="radiobutton2" className="radio-button">
408                 <input id="radiobutton2" type="radio" name="radiobutton"
409                 className={formMode} /><p>Radio 2</p>
410                 </label>
411                 <label htmlFor="radiobutton3" className="radio-button">
412                     <input id="radiobutton3" type="radio" name="radiobutton"
413                     className={formMode} /><p>Radio 3</p>
414                     </label>
415                 </div>
416             <br /><br />
417
418             <p>File</p>
419             <label>
420                 <input id="upload" type="file" className={formMode} />
421                 <div className="input-error">This field is required.</div>
422             </label>
423         </div>
424
425         <p>Disabled inputs use the <span className="text-primary-dark">disabled</span>
426         > attribute or <span className="text-primary-dark">.input-disabled</span>. Inputs still need
427         the <span className="text-primary-dark">disabled</span> attribute to be functionally disabled
428         .</p>
429
430         <p>Required inputs use <span className="text-primary-dark">.input-required</span>,
431         though the class should only be added if the error is active.</p>
432     </div>
433
434     <div className="mt-10">
435         <h2>Pagination</h2>
436         <Pagination currentPage={pg1} setCurrentPage={setPg1} totalItems={10}
437         itemsPerPage={1} pageRange={5} />
438         <Pagination currentPage={pg2} setCurrentPage={setPg2} totalItems={20}
439         itemsPerPage={1} pageRange={9} />
440     </div>
441
442     <div className="mt-10">
443         <h2>Popups</h2>
444         <p>Please refer to the Popup component for popup usage</p>
445         <br />
446         <button className="button mr-5 mb-5" onClick={() => { setShowPopup1(true) }}>
447         Popup Example 1</button>
448         <button className="button mr-5 mb-5" onClick={() => { setShowPopup2(true) }}>
449         Popup Example 2</button>
450         <button className="button mr-5 mb-5" onClick={handleShowPopupForm}>Popup Form
451         </button>
452     </div>
453
454     <div className="mt-10">
455         <h2>Icons</h2>
456         <p>Icons from <a href="https://icons.getbootstrap.com/">Bootstrap Icons</a>.
457         Use icon font mode for including them in HTML.</p>
458     </div>
459
460     <div className="mt-10">

```



```

449         <h2>Breaks</h2>
450         <p>Breaks and Heading Breaks are in the graphics.jsx component. Left headings
can stretch across the screen or stretch 60% of the way.</p>
451         <Break />
452         <H1Middle>Heading 1 Middle</H1Middle>
453         <H1Left>Heading 1 Left</H1Left>
454         <H1Left stretch>Heading 1 Left</H1Left>
455         <H2Middle>Heading 2 Middle</H2Middle>
456         <H2Left>Heading 2 Left</H2Left>
457         <H2Left stretch>Heading 2 Left</H2Left>
458     </div>
459
460 </div>
461 <div className="my-10">
462     <div className="lg:px-40 px-10 duration-200">
463         <h2>Banners</h2>
464         <p>Banner content is customizable. Default banner type is 'green', 'blue' and
'dark' need to be specified using the 'type' prop.</p>
465     </div>
466     <br />
467     <Banner>
468         <h1>Banner</h1>
469     </Banner>
470     <br />
471     <Banner type="blue">
472         <h1>Banner</h1>
473         <p>But blue this time.</p>
474     </Banner>
475     <br />
476     <Banner type="dark">
477         <h1>Banner</h1>
478         <p>But dark this time.</p>
479     </Banner>
480 </div>
481 <Footer />
482 </>
483 );
484 }

```

A.3 app.css

```

1 @import "tailwindcss";
2
3 /*
4
5 This stylesheet is for global styles only.
6 Please style individual elements in separate dedicated stylesheets
7
8 */
9
10 @theme {
11     --font-sans: "Oxygen", ui-sans-serif, system-ui, sans-serif,
12         "Apple Color Emoji", "Segoe UI Emoji", "Segoe UI Symbol", "Noto Color Emoji";
13
14     /* Old colors for compatibility */
15     --color-seagreen: var(--color-primary-light);
16     --color-turquoise: var(--color-primary-dark);
17     --color-betta: var(--color-secondary-light);
18     --color-tarawera: var(--color-secondary-dark);
19     --color-potassium: var(--color-disabled-light);
20     /*--color-gray-300 replaced with --color-gray-light*/
21     /*--color-zinc-800 replaced with --color-gray-dark*/

```

```

22  /*--color-emerald-100 replaced with color-primary-extralight*/
23
24  /* New colors */
25  --color-primary-light: #74C9B5;
26  --color-primary-dark: #1EA493;
27  --color-secondary-light: #136E8D;
28  --color-secondary-dark: #063751;
29  --color-gray-light: var(--color-gray-300);
30  --color-gray-dark: var(--color-zinc-800);
31  --color-primary-extralight: var(--color-emerald-100);
32  --color-disabled-light: #8b9d9b;
33  --color-disabled-dark: #404547;
34  --color-selection-highlight: #4388A1;
35  --color-error: #c2162d;
36  --color-error-dark: #7A0e1d;
37  --color-error-extradark: #36070c;
38  --color-slate-100: #f1f5f9;
39
40 }
41
42 html,
43 body {
44   @apply bg-white;
45   --position-top: 0;
46 }
47
48 .noscroll {
49   position: fixed;
50   top: var(--position-top, 0);
51   inline-size: 100%;
52   overflow-y: scroll;
53 }
54
55
56 /* TEXT *****/
57
58 ::selection {
59   /* color: #c2162d; */
60   background-color: var(--color-selection-highlight);
61   color: var(--color-white);
62 }
63
64 h1,
65 h2,
66 h3,
67 h4,
68 h5,
69 h6 {
70   font-family: "Quicksand";
71   font-weight: 500;
72   margin-bottom: calc(var(--spacing) * 2);
73   transition: 0.2s;
74 }
75
76 h1 {
77   font-size: 2.5rem;
78   color: var(--color-primary-light);
79   text-transform: uppercase;
80 }
81
82 h2 {
83   font-size: 2rem;
84   color: var(--color-primary-dark);
85   text-transform: uppercase;
86 }

```

```

87
88 h3 {
89   font-size: 1.7rem;
90   color: var(--color-secondary-light);
91   text-transform: uppercase;
92 }
93
94 h4 {
95   font-size: 1.7rem;
96   color: var(--color-secondary-light);
97 }
98
99 h5 {
100   font-size: 1.5rem;
101   color: var(--color-secondary-light);
102 }
103
104 h6 {
105   font-size: 1.2rem;
106   color: var(--color-secondary-light);
107 }
108
109 a h1:hover,
110 a h2:hover {
111   color: var(--color-secondary-light);
112 }
113
114 a h3:hover,
115 a h4:hover,
116 a h5:hover,
117 a h6:hover {
118   color: var(--color-primary-light);
119 }
120
121 p {
122   margin-bottom: calc(var(--spacing) * 2);
123   white-space: pre-line;
124 }
125
126 p:last-of-type {
127   margin-bottom: 0;
128 }
129
130 p>a,
131 .link {
132   color: var(--color-primary-dark);
133   transition: 0.2s;
134 }
135
136 p>a:hover,
137 .link:hover {
138   color: var(--color-primary-light);
139   cursor: pointer;
140 }
141
142 p a,
143 li a {
144   /* color: var(--color-primary-dark); */
145   border-bottom: 1px solid;
146   border-color: var(--color-primary-dark);
147 }
148
149 p a:hover,
150 li a:hover {
151   border-color: var(--color-primary-light);

```

```

152 }
153
154 ul {
155   list-style-type: disc;
156   margin-left: 1rem;
157   margin-bottom: calc(var(--spacing) * 2);
158 }
159
160 ol {
161   list-style-type: decimal;
162   margin-left: 1rem;
163   margin-bottom: calc(var(--spacing) * 2);
164 }
165
166 /* MENU *****/
167
168 #menu .link,
169 #side-menu .link {
170   z-index: 1;
171   color: var(--color-black);
172   font-size: var(--text-sm);
173 }
174
175 #menu .active {
176   color: var(--color-secondary-light);
177   font-weight: bold;
178 }
179
180 #menu .link:hover,
181 #side-menu .link:hover {
182   color: var(--color-primary-light);
183   cursor: pointer;
184 }
185
186 #menu .menu-dropdown-button {
187   z-index: 1;
188   /* padding: calc(var(--spacing) * 4); */
189   position: relative;
190 }
191
192 #menu .menu-dropdown {
193   z-index: 0;
194   position: absolute;
195   display: flex;
196   flex-direction: column;
197   background-color: var(--color-white);
198   border: 2px solid var(--color-primary-light);
199   border-radius: var(--radius-md);
200   box-shadow: var(--shadow-sm);
201
202   opacity: 0;
203   transition: margin-top 0.5s ease-in, opacity 0.2s 0s;
204   /* When disappear */
205 }
206
207 #menu .menu-dropdown-container.active-popup .menu-dropdown-button {
208   color: var(--color-primary-light);
209 }
210
211 #menu .menu-dropdown-container.active-popup .menu-dropdown {
212   opacity: 1;
213   margin-top: -4px;
214   transition: margin-top 0.5s ease-out, opacity 0.2s 0.3s;
215   /* When appear */
216 }

```

```

217
218
219 #side-menu .side-menu-dropdown-button {
220     transition: 0.2s;
221 }
222
223 #side-menu .side-menu-dropdown {
224     max-height: 0px;
225     opacity: 0;
226     overflow-y: hidden;
227     transition: 0.8s;
228 }
229
230 #side-menu .side-menu-dropdown-container.active-popup .side-menu-dropdown-button {
231     color: var(--color-primary-light);
232 }
233
234 #side-menu .side-menu-dropdown-container.active-popup .side-menu-dropdown {
235     max-height: 200px;
236     opacity: 1;
237 }
238
239 /* GOOGLE TRANSLATE WIDGET *****/
240
241 /* Hide Google banner */
242 .goog-te-banner-frame.skiptranslate {
243     display: none !important;
244 }
245
246 body>.skiptranslate {
247     display: none !important;
248 }
249
250 .skiptranslate {
251     display: none !important;
252 }
253
254 /* Google adds inline style "top: 40px" to body when banner appears */
255 body {
256     top: 0px !important;
257 }
258
259 /* Hide Google default UI */
260 .goog-te-gadget {
261     display: none !important;
262 }
263
264 .goog-te-gadget img {
265     display: none !important;
266 }
267
268 /* Hide the small "translated" tooltip highlight (doesn't work) */
269 body>.goog-text-highlight {
270     background: none !important;
271     box-shadow: none !important;
272 }
273
274
275 /* BUTTONS *****/
276
277 .button {
278     background-color: var(--color-primary-dark);
279     padding: calc(var(--spacing) * 2) calc(var(--spacing) * 4);
280     border: 2px solid var(--color-primary-dark);
281     border-radius: var(--radius-md);

```

```

282 color: var(--color-white);
283 text-align: center;
284 transition: 0.2s;
285 }
286
287 .button:hover {
288 color: white;
289 background-color: var(--color-secondary-light);
290 border-color: var(--color-secondary-light);
291 cursor: pointer;
292 }
293
294 .button:hover:active {
295 background-color: var(--color-secondary-dark);
296 border-color: var(--color-secondary-dark);
297 }
298
299 .button.button-light {
300 background-color: var(--color-white);
301 color: var(--color-primary-dark);
302 }
303
304 .button.button-light:hover {
305 background-color: var(--color-white);
306 color: var(--color-secondary-light);
307 cursor: pointer;
308 }
309
310 .button.button-light:hover:active {
311 color: var(--color-secondary-dark);
312 }
313
314 .button.button-disabled,
315 button:disabled.button {
316 background-color: var(--color-disabled-light);
317 border: 2px solid var(--color-disabled-light);
318 cursor: default;
319 }
320
321 .button.button-light.button-disabled,
322 button:disabled.button.button-light {
323 background-color: var(--color-white);
324 border: 2px solid var(--color-disabled-light);
325 color: var(--color-disabled-light);
326 cursor: default;
327 }
328
329 .button.button-pg {
330 width: calc(var(--spacing) * 10);
331 padding: 0;
332 display: flex;
333 justify-content: center;
334 align-items: center;
335 }
336
337 .button.button-pg.button-light {
338 transition: 0s;
339 }
340
341 .button.button-pg-arrow {
342 width: calc(var(--spacing) * 14);
343 padding: 0;
344 display: flex;
345 justify-content: center;
346 align-items: center;

```

```

347   font-size: 1.7rem;
348 }
349
350 .button.button-delete {
351   background-color: var(--color-error);
352   border-color: var(--color-error);
353   color: var(--color-white);
354 }
355
356 .button.button-delete:hover {
357   background-color: var(--color-error-dark);
358   border-color: var(--color-error-dark);
359 }
360
361 .button.button-delete:hover:active {
362   background-color: var(--color-error-extradark);
363   border-color: var(--color-error-extradark);
364 }
365
366 .button.button-light.button-delete {
367   background-color: var(--color-white);
368   color: var(--color-error);
369   border-color: var(--color-error);
370 }
371
372 .button.button-light.button-delete:hover {
373   background-color: var(--color-white);
374   color: var(--color-error-dark);
375   border-color: var(--color-error-dark);
376 }
377
378 .button.button-light.button-delete:hover:active {
379   color: var(--color-error-extradark);
380   border-color: var(--color-error-extradark);
381 }
382
383 .button-icon {
384   color: var(--color-primary-dark);
385   transition: 0.2s;
386 }
387
388 .button-icon:hover {
389   color: var(--color-secondary-dark);
390   cursor: pointer;
391 }
392
393 .button.button-red {
394   color: var(--color-error);
395   border-color: var(--color-error);
396   transition: 0.2s;
397 }
398
399 .button.button-red:hover {
400   color: #71000f;
401   border-color: #71000f;
402   cursor: pointer;
403 }
404
405
406 /* INTERNATIONAL MEETINGS *****/
407
408 .meeting-dropdown-content {
409   display: grid;
410   grid-template-rows: 0fr;
411   opacity: 0;

```

```

412     transition: grid-template-rows 0.25s ease, opacity 0.25s ease;
413 }
414
415 .meeting-dropdown-content > div {
416     overflow: hidden;
417 }
418
419 .meeting-dropdown[open] .meeting-dropdown-content {
420     grid-template-rows: 1fr;
421     opacity: 1;
422 }
423
424 .meeting-dropdown[open] .meeting-dropdown-content > div {
425     animation: meeting-dropdown-slide 0.25s ease;
426 }
427
428 @keyframes meeting-dropdown-slide {
429     from {
430         opacity: 0;
431         transform: translateY(-8px);
432     }
433
434     to {
435         opacity: 1;
436         transform: translateY(0);
437     }
438 }
439
440 @keyframes meeting-gallery-slide-next {
441     from {
442         opacity: 0.45;
443         transform: translateX(26px);
444     }
445
446     to {
447         opacity: 1;
448         transform: translateX(0);
449     }
450 }
451
452 @keyframes meeting-gallery-slide-prev {
453     from {
454         opacity: 0.45;
455         transform: translateX(-26px);
456     }
457
458     to {
459         opacity: 1;
460         transform: translateX(0);
461     }
462 }
463
464 .meeting-gallery-image.is-next {
465     animation: meeting-gallery-slide-next 0.35s ease;
466 }
467
468 .meeting-gallery-image.is-prev {
469     animation: meeting-gallery-slide-prev 0.35s ease;
470 }
471
472
473 /* FORMS *****/
474
475 .input-text {
476     background-color: var(--color-white);

```



```

477 margin-top: var(--spacing);
478 padding: var(--spacing) calc(var(--spacing) * 2);
479 border: 2px solid var(--color-primary-light);
480 border-radius: var(--radius-md);
481 transition: 0.2s;
482 }
483
484 .input-text:focus {
485   border-color: var(--color-secondary-light);
486   outline: none;
487 }
488
489 input:disabled.input-text,
490 textarea:disabled.input-text,
491 select:disabled.input-text,
492 .input-text.input-disabled {
493   border-color: var(--color-disabled-light);
494   color: var(--color-disabled-light);
495   cursor: default;
496 }
497
498 input[type="file"]::file-selector-button {
499   background-color: var(--color-white);
500   margin-top: var(--spacing);
501   margin-right: calc(var(--spacing) * 3);
502   padding: calc(var(--spacing) * 2) calc(var(--spacing) * 4);
503   border: 2px solid var(--color-primary-light);
504   border-radius: var(--radius-md);
505   color: var(--color-primary-dark);
506   text-align: center;
507   transition: 0.2s;
508 }
509
510 input[type="file"]::file-selector-button:hover {
511   border-color: var(--color-primary-dark);
512   cursor: pointer;
513 }
514
515 input[type="file"]::file-selector-button:hover:active {
516   border-color: var(--color-secondary-dark);
517   color: var(--color-secondary-dark);
518 }
519
520 input[type="file"]:disabled::file-selector-button,
521 input[type="file"].input-disabled::file-selector-button {
522   border-color: var(--color-disabled-light);
523   color: var(--color-disabled-light);
524 }
525
526 input[type="file"]:disabled::file-selector-button:hover,
527 input[type="file"].input-disabled::file-selector-button:hover {
528   cursor: default;
529 }
530
531 .checkbox {
532   display: inline-block;
533   margin-right: calc(var(--spacing) * 4);
534 }
535
536 .checkbox:hover {
537   cursor: pointer;
538 }
539
540 .checkbox>input {
541   display: none;

```

```

542 }
543
544 .checkbox>p {
545   width: fit-content;
546   display: flex;
547   align-items: center;
548 }
549
550 .checkbox>p::before {
551   content: "\F26E";
552   font-family: "Bootstrap-icons";
553   width: 1.25rem;
554   height: 1.25rem;
555   margin-right: 5px;
556   border: 2px solid var(--color-primary-light);
557   border-radius: var(--radius-md);
558
559   color: var(--color-white);
560   background-color: var(--color-white);
561
562   display: flex;
563   align-items: center;
564   -webkit-text-stroke: 1px;
565
566   transition: 0.2s;
567 }
568
569 .checkbox:hover>p::before {
570   background-color: var(--color-primary-extralight);
571 }
572
573 .checkbox>input:checked+p::before {
574   background-color: var(--color-primary-dark);
575   border-color: var(--color-primary-dark);
576 }
577
578 .checkbox>input[type="checkbox"]:disabled+p::before,
579 .checkbox>input[type="checkbox"].input-disabled+p::before {
580   border-color: var(--color-disabled-light);
581 }
582
583 .checkbox:hover>input[type="checkbox"]:disabled+p::before,
584 .checkbox:hover>input[type="checkbox"].input-disabled+p::before {
585   background-color: var(--color-white);
586 }
587
588 .checkbox>input[type="checkbox"]:checked:disabled+p::before,
589 .checkbox>input[type="checkbox"]:checked.input-disabled+p::before {
590   background-color: var(--color-disabled-light);
591 }
592
593 .checkbox:hover>input[type="checkbox"]:disabled+p,
594 .checkbox:hover>input[type="checkbox"].input-disabled+p {
595   cursor: default;
596 }
597
598
599 .radio {
600   display: inline-block;
601   margin-right: calc(var(--spacing) * 4);
602 }
603
604 .radio:hover {
605   cursor: pointer;
606 }

```

```

607
608 .radio>input {
609     display: none;
610 }
611
612 .radio>p {
613     width: fit-content;
614     display: flex;
615     align-items: center;
616 }
617
618 .radio>p::before {
619     content: "";
620     display: block;
621     width: 1.25rem;
622     height: 1.25rem;
623     margin-right: 5px;
624     border: 2px solid var(--color-primary-light);
625     border-radius: calc(infinity * 1px);
626     background-color: var(--color-white);
627     transition: 0.2s;
628 }
629
630 .radio:hover>p:before {
631     background-color: var(--color-primary-extralight);
632 }
633
634 .radio>input:checked+p:before {
635     border: 6px solid var(--color-primary-dark);
636 }
637
638 .radio:hover>input:checked+p:before {
639     background-color: var(--color-white);
640 }
641
642 .radio>input[type="radio"]:disabled+p::before,
643 .radio>input[type="radio"].input-disabled+p::before {
644     border-color: var(--color-disabled-light);
645 }
646
647 .radio:hover>input[type="radio"]:disabled+p,
648 .radio:hover>input[type="radio"].input-disabled+p::before {
649     background-color: var(--color-white);
650 }
651
652 .radio:hover>input[type="radio"]:disabled+p,
653 .radio:hover>input[type="radio"].input-disabled+p {
654     cursor: default;
655 }
656
657
658 .radio-button {
659     display: inline-block;
660     margin-right: var(--spacing);
661     margin-bottom: var(--spacing);
662 }
663
664 .radio-button:hover {
665     cursor: pointer;
666 }
667
668 .radio-button>input {
669     display: none;
670 }
671

```

```

672 .radio-button>p {
673   background-color: var(--color-white);
674   color: var(--color-primary-dark);
675   padding: calc(var(--spacing) * 2) calc(var(--spacing) * 4);
676   border: 2px solid var(--color-primary-dark);
677   border-radius: var(--radius-md);
678   text-align: center;
679   transition: 0.2s;
680 }
681
682 .radio-button:hover>p {
683   border-color: var(--color-secondary-light);
684   color: var(--color-secondary-light);
685 }
686
687 .radio-button>input:checked+p {
688   background-color: var(--color-primary-dark);
689   color: var(--color-white);
690 }
691
692 .radio-button:hover>input:checked+p {
693   border-color: var(--color-primary-dark);
694 }
695
696 .radio-button>input[type="radio"]:disabled+p,
697 .radio-button>input[type="radio"].input-disabled+p {
698   cursor: default;
699   border-color: var(--color-disabled-light);
700   color: var(--color-disabled-light);
701 }
702
703 .radio-button>input[type="radio"]:disabled:checked+p,
704 .radio-button>input[type="radio"].input-disabled:checked+p {
705   background-color: var(--color-disabled-light);
706   color: var(--color-white);
707 }
708
709 .input-text.input-required {
710   border-color: var(--color-error);
711 }
712
713 .checkbox>input[type="checkbox"].input-required+p::before {
714   border-color: var(--color-error);
715 }
716
717 .checkbox:hover>input[type="checkbox"].input-required+p::before {
718   background-color: var(--color-red-100);
719 }
720
721 .checkbox>input[type="checkbox"]:checked.input-required+p::before {
722   background-color: var(--color-error);
723 }
724
725 .radio>input[type="radio"].input-required+p::before {
726   border-color: var(--color-error);
727 }
728
729 .radio:hover>input[type="radio"].input-required+p::before {
730   background-color: var(--color-red-100);
731 }
732
733 .radio-button>input[type="radio"].input-required+p {
734   border-color: var(--color-error);
735   color: var(--color-error);
736 }

```

```

737
738 .radio-button:hover>input[type="radio"].input-required+p {
739   border-color: #71000f;
740   color: #71000f;
741 }
742
743 .radio-button>input[type="radio"].input-required:checked+p {
744   background-color: var(--color-error);
745   color: var(--color-white);
746 }
747
748 .radio-button:hover>input[type="radio"].input-required:checked+p {
749   border-color: var(--color-error);
750 }
751
752 input[type="file"].input-required::file-selector-button {
753   border-color: var(--color-error);
754   color: var(--color-error);
755 }
756
757 input[type="file"].input-required::file-selector-button:hover {
758   border-color: var(--color-red-500);
759   color: var(--color-red-500);
760 }
761
762 .input-error {
763   display: none;
764   position: absolute;
765   font-size: 0.9rem;
766   text-align: right;
767   color: var(--color-error);
768 }
769
770 .input-text.input-required+.input-error {
771   display: block;
772 }
773
774 input[type="checkbox"].input-required~.input-error,
775 input[type="radio"].input-required~.input-error,
776 input[type="file"].input-required~.input-error {
777   display: block;
778 }
779
780 /* ICONS *****/
781
782 .bi {
783   -webkit-text-stroke: 0.5px;
784   /*Originally 1px*/
785 }
786
787
788
789 .banner>.banner-content>h1,
790 .banner>.banner-content>h2,
791 .banner>.banner-content>h3,
792 .banner>.banner-content>h4,
793 .banner>.banner-content>h5,
794 .banner>.banner-content>h6,
795 .banner>.banner-content>p {
796   color: white;
797 }
798
799 .banner.bg-secondary-dark>.banner-content>h1,
800 .banner.bg-secondary-dark>.banner-content>h2,
801 .banner.bg-secondary-dark>.banner-content>h3,

```

```

802 .banner.bg-secondary-dark>.banner-content>h4,
803 .banner.bg-secondary-dark>.banner-content>h5,
804 .banner.bg-secondary-dark>.banner-content>h6,
805 .banner.bg-secondary-dark>.banner-content>p {
806   color: var(--color-primary-light);
807 }
808
809 .banner-breadcrumb {
810   margin-left: -15px;
811   transition: 0.2s;
812   color: white;
813 }
814
815 .banner-breadcrumb>strong {
816   border-bottom: 2px solid #ffffff00;
817   transition: 0.2s;
818 }
819
820 .banner-breadcrumb:hover {
821   margin-left: -20px;
822   color: var(--color-primary-light);
823 }
824
825 .banner-breadcrumb:hover>strong {
826   margin-left: 5px;
827   color: var(--color-primary-light);
828   border-color: var(--color-primary-light);
829 }
830
831
832 #task-forces>div:nth-child(4n-2),
833 #task-forces>div:nth-child(4n-1) {
834   background-color: var(--color-slate-100);
835 }
836
837 @media (width < 64rem) {
838   #task-forces>div:nth-child(n) {
839     background-color: var(--color-white);
840   }
841
842   #task-forces>div:nth-child(even) {
843     background-color: var(--color-slate-100);
844   }
845
846 }

```

A.4 search.jsx

```

1 import { useMemo, useState, useEffect } from "react";
2 import { useNavigate } from "react-router";
3 import { supabase } from "../supabase";
4 import { Menu } from "../components/menu";
5 import { Pagination } from "../components/pagination";
6 import { Footer } from "../components/footer";
7
8 export function meta() {
9   return [
10     { title: "People" },
11     { name: "", content: "" },
12   ];
13 }
14

```

```

15 async function getPeople() {
16   const { data, error } = await supabase
17     .from('users')
18     .select('id, fname, lname, email, image_url, job_position, languages, country, institution,
19       major, research_interests, task_force_role, task_force')
20   if (data) {
21     data.sort((a, b) => { return `${a.fname} ${a.lname}` > `${b.fname} ${b.lname}` ? 1 : -1 });
22   }
23   return data || error;
24 }
25
26 const PAGE_SIZE = 8;
27
28 const formatFilterLabel = (value) => {
29   if (!value) return value;
30   return value
31     .split(" ")
32     .map((word) => {
33       if (!word) return word;
34       if (word === word.toUpperCase()) return word;
35       return word
36         .split("-")
37         .map((part) => {
38           if (!part) return part;
39           if (part === part.toUpperCase()) return part;
40           return part.charAt(0).toUpperCase() + part.slice(1).toLowerCase();
41         })
42         .join("-");
43     })
44     .join(" ");
45 };
46
47 export async function loader({}) {
48   return getPeople();
49 }
50
51 export default function SearchRoute({ loaderData }) {
52   const [query, setQuery] = useState("");
53   const [page, setPage] = useState(0);
54   const [showFilters, setShowFilters] = useState(false);
55   const [selectedTaskForces, setSelectedTaskForces] = useState([]);
56   const [selectedCountries, setSelectedCountries] = useState([]);
57   const [selectedMajors, setSelectedMajors] = useState([]);
58
59   const taskForceOptionsSet = new Set(loaderData.map((person) => { return (person.task_force !== "" ? person.task_force : null) })));
60   taskForceOptionsSet.delete(null);
61
62   const countryOptionsSet = new Set(loaderData.map((person) => { return (person.country !== "") ? person.country : null })));
63   countryOptionsSet.delete(null);
64
65   const majorOptionsSet = new Set(loaderData.map((person) => { return (person.major !== "") ? person.major : null })));
66   majorOptionsSet.delete(null);
67
68   const taskForceOptions = useMemo(
69     () => Array.from(taskForceOptionsSet).sort(),
70     []
71   );
72   const countryOptions = useMemo(
73     () => Array.from(countryOptionsSet).sort(),
74     []
75   );

```

```

76  const majorOptions = useMemo(
77    () => Array.from(majorOptionsSet).sort(),
78    []
79  );
80
81
82  const results = useMemo(() => {
83    const q = query.trim().toLowerCase();
84    return loaderData.filter((person) => {
85
86      const matchesQuery =
87        !q ||
88        person?.fname?.toLowerCase().includes(q) ||
89        person?.institution?.toLowerCase().includes(q) ||
90        person?.major?.toLowerCase().includes(q) ||
91        person?.languages?.toLowerCase().includes(q) ||
92        person?.research_interests?.toLowerCase().includes(q);
93
94      const matchesTaskForce =
95        selectedTaskForces.length === 0 || selectedTaskForces.includes(person.task_force);
96      const matchesCountry =
97        selectedCountries.length === 0 || selectedCountries.includes(person.country);
98      const matchesMajor =
99        selectedMajors.length === 0 || selectedMajors.includes(person.major);
100
101      return matchesQuery && matchesTaskForce && matchesCountry && matchesMajor;
102    });
103  }, [query, selectedTaskForces, selectedCountries, selectedMajors]);
104
105  // Reset to page 1 whenever query or filters change
106  useEffect(() => {
107    setPage(0);
108  }, [query, selectedTaskForces, selectedCountries]);
109
110  const totalPages = Math.ceil(results.length / PAGE_SIZE);
111  const startIdx = page * PAGE_SIZE;
112  const pageResults = results.slice(startIdx, startIdx + PAGE_SIZE);
113
114  // If results shrink and current page becomes invalid
115  useEffect(() => {
116    const lastPageIndex = Math.max(0, totalPages - 1);
117    if (page > lastPageIndex) setPage(0);
118  }, [page, totalPages]);
119
120  const hasQuery = query.trim().length > 0;
121  const hasActiveFilters = selectedTaskForces.length > 0 || selectedCountries.length > 0;
122  const showHeader = hasQuery || hasActiveFilters;
123  const isEmptyState = !showHeader;
124
125  const toggleSelection = (setState, value) => {
126    setState((prev) => (prev.includes(value) ? prev.filter((item) => item !== value) : [...prev, value]));
127  };
128
129  const handleQueryChange = (event) => {
130    const value = event.currentTarget.value;
131    if (value !== query) {
132      setQuery(value);
133    }
134  };
135
136  return (
137    <div className="min-h-screen bg-white">
138      <Menu />
139

```



```

140 <div className="mx-auto max-w-[1100px] px-6 pb-20 pt-10">
141   <div className="text-center">
142     <h2 className="mb-2 normal-case text-primary-dark">
143       Search People
144     </h2>
145     <p
146       className={`text-sm text-gray-dark/70 transition-opacity duration-200 ${
147         showHeader ? "opacity-100" : "opacity-0"
148       }`}
149     >
150       <div>
151         {results.length} result{results.length === 1 ? "" : "s"} found
152       </div>
153     </p>
154   </div>
155
156   <div className="mt-8 flex flex-wrap items-center justify-between gap-4">
157     <div className="flex w-full max-w-[450px] items-center gap-2 rounded-md border-2 border
158     -primary-light bg-white px-4 py-2 focus-within:bg-teal-50">
159       <i className="bi bi-search text-gray-dark/60" aria-hidden="true" />
160       <input
161         id="search-input"
162         value={query}
163         onChange={handleQueryChange}
164         onInput={handleQueryChange}
165         placeholder="Search people, institutions, or research interests"
166         className="w-full bg-transparent text-sm text-gray-dark outline-none"
167       />
168       <button className="size-5 duration-200 relative hover:cursor-pointer hover:text-
169       primary-dark text-gray-dark/60"
170         onClick={() => {document.getElementById("search-input").value = ""; setQuery(
171         "");}}>
172         <i className="bi bi-x text-[1.5rem] absolute -top-2 -left-1" />
173       </button>
174     </div>
175
176     <div className="flex flex-wrap gap-3">
177       <button
178         type="button"
179         className="button button-light flex items-center gap-2 text-sm font-semibold"
180         onClick={() => setShowFilters((prev) => !prev)}
181         aria-expanded={showFilters}
182         aria-controls="search-filters"
183       >
184         <i className="bi bi-funnel" aria-hidden="true" />
185         Filter
186       </button>
187     </div>
188
189     <div
190       id="search-filters"
191       aria-hidden={!showFilters}
192       className={`origin-top overflow-hidden transition-all duration-300 ease-out ${
193         showFilters
194           ? "mt-4 max-h-[800px] translate-y-0 opacity-100 pointer-events-auto"
195           : "mt-0 max-h-0 -translate-y-2 opacity-0 pointer-events-none"
196       }`}
197     >
198       <div className="rounded-md border-2 border-gray-light bg-white p-4">
199         <fieldset disabled={!showFilters} className="grid gap-6 md:grid-cols-2">
200           <FilterGroup
201             title="Task Force"
202             options={taskForceOptions}
203             selected={selectedTaskForces}

```

```

202         onToggle={(value) => toggleSelection(setSelectedTaskForces, value)}
203     />
204     <FilterGroup
205         title="Country"
206         options={countryOptions}
207         selected={selectedCountries}
208         onToggle={(value) => toggleSelection(setSelectedCountries, value)}
209     />
210     <FilterGroup
211         title="Major"
212         options={majorOptions}
213         selected={selectedMajors}
214         onToggle={(value) => toggleSelection(setSelectedMajors, value)}
215     />
216 </fieldset>
217 </div>
218 </div>
219
220 <div className="mt-8 flex flex-col gap-6">
221     { results.length === 0 ? (
222         <div className="rounded-md border-2 border-gray-light bg-white px-6 py-8 text-sm text
-gray-dark/70">
223         No results found. Try a different name, institution, or research interest.
224         </div>
225     ) : (
226         <>
227             {pageResults.map((person) => (
228                 <PersonResultCard key={person.id} person={person} />
229             ))}
230
231             {totalPages > 1 ? (
232                 <Pagination
233                     currentPage={page}
234                     setCurrentPage={setPage}
235                     totalItems={results.length}
236                     itemsPerPage={PAGE_SIZE}
237                 />
238             ) : null}
239         </>
240     )}
241 </div>
242 </div>
243 <Footer />
244 </div>
245 );
246 }
247
248 function FilterGroup({ title, options, selected, onToggle }) {
249     if (options.length > 0) {
250         return (
251             <div>
252                 <div className="mb-3 font-semibold text-secondary-dark">{title}</div>
253                 <div className="flex flex-col gap-2">
254                     {options.map((option) => (
255                         <label key={option} className="checkbox">
256                             <input
257                                 type="checkbox"
258                                 checked={selected.includes(option)}
259                                 onChange={() => onToggle(option)}
260                             />
261                             <p className="text-gray-dark/80">{formatFilterLabel(option)}</p>
262                         </label>
263                     ))}
264                 </div>
265             </div>

```

```

266   });
267 }
268 }
269
270 function PersonResultCard({ person }) {
271   return (
272     <a
273       href={"/profile/${person.id}`}
274       role="link"
275       tabIndex={0}
276       aria-label={"View profile for ${person.fname}`}
277       onKeyDown={(event) => {
278         if (event.key === "Enter" || event.key === " ") {
279           event.preventDefault();
280           handleNavigate();
281         }
282       }}
283       className="grid cursor-pointer gap-6 rounded-md border-2 border-gray-light bg-teal-50 p-6
284       transition hover:shadow-md md:grid-cols-3"
285     >
286       <div className="flex items-center gap-4">
287         <div
288           className="flex shrink-0 h-20 w-20 items-center justify-center overflow-hidden rounded-
289           full bg-gray-light border-primary-dark border-3 border-box"
290           aria-hidden="true"
291         >
292           {person.image_url ? (
293             <img
294               src={person.image_url}
295               alt={` ${person.fname} ${person.lname}`}
296               className="h-full w-full object-cover"
297             />
298           ) : (
299             <i className="bi bi-person-fill text-[40px] text-secondary-dark/60" aria-hidden="true
300             " />
301           )}
302         </div>
303         <div>
304           <div className="text-xl font-semibold text-secondary-dark">{person.fname} {person.lname}
305           </div>
306           <div className="my-1 italic text-secondary-light">
307             {person.job_position}{ person?.job_position && person?.institution && <span>, </span>
308             >}{person.institution}
309           </div>
310           <div className="text-sm text-gray-dark/70">{person.email}</div>
311         </div>
312       </div>
313
314       <div className="grid text-sm text-gray-dark/80 grid-cols-2 gap-2 italic">
315         { person?.country ?
316           <div className="">
317             <span className="mr-1">Country</span> <span className="font-semibold text-secondary-
318             dark">{person.country}</span>
319           </div>
320           : <div></div>
321         }
322         { person?.languages ?
323           <div className="">
324             <span className="mr-1">Languages</span> <span className="font-semibold text-secondary
325             -dark">{person.languages}</span>
326           </div>
327           : <div></div>
328         }
329         { person?.major ?
330           <div className="">

```

```

324     <span className="mr-1">Major</span> <span className="font-semibold text-secondary-
dark">{person.major}</span>
325   </div>
326   : <div></div>
327 }
328 { person?.research_interests ?
329   <div className="">
330     <span className="mr-1">Research Interests</span> <span className="font-semibold text-
secondary-dark">{person.research_interests}</span>
331   </div>
332   : <div></div>
333 }
334 </div>
335 <div>
336   { person.task_force &&
337     <div className="w-full rounded-md border-2 border-primary-light bg-white px-4 py-3 text-
xs">
338       <div className="font-semibold text-secondary-dark">{person.task_force}</div>
339       <div className="mt-1 text-gray-dark/70">{person.task_force_role}</div>
340     </div>
341   }
342 </div>
343
344 </a>
345 );
346 }

```

A.5 video-resources.jsx

```

1 import { useState, useEffect } from "react";
2 import { useNavigate } from "react-router";
3 import { supabase } from "../supabase";
4 import { Menu } from "../components/menu";
5 import { Footer } from "../components/footer";
6 import { Popup, PopupForm } from "../components/popup";
7 import { Pagination } from "../components/pagination";
8 import { updateRequired } from "../helpers/form";
9 import "../styles/video-resources.css";
10
11 export function meta() {
12   return [
13     { title: "Video Resources" },
14     { name: "", content: "" },
15   ];
16 }
17
18 async function getVideoResources() {
19   const { data, error } = await supabase
20     .from('video resources')
21     .select('id, title, date, speaker, speaker_university, video_thumbnail')
22   return data || error;
23 }
24
25 export async function loader({ params }) {
26   const videoResources = await getVideoResources();
27   return { videoResources: videoResources };
28 }
29
30 async function createVideoResource(formData) {
31   let thumbnailUrl = null;
32   let speakerImgUrl = null;
33 }

```

```

34 const thumbnailFile = formData.get("vid-resource-thumbnail");
35 if (thumbnailFile && thumbnailFile.name && thumbnailFile.size > 0) {
36   const path = `${Date.now()}-${thumbnailFile.name.replace(/^[a-zA-Z0-9.-]/g, "_")}`;
37   const { error: uploadError } = await supabase.storage
38     .from('video-resources')
39     .upload(path, thumbnailFile);
40
41   if (uploadError) {
42     console.error("Error uploading thumbnail:", uploadError);
43     return uploadError;
44   }
45
46   const { data, error: urlError } = supabase.storage
47     .from('video-resources')
48     .getPublicUrl(path);
49
50   if (urlError) {
51     console.error("Error getting thumbnail URL:", urlError);
52     return urlError;
53   }
54
55   thumbnailUrl = data.publicUrl;
56 }
57
58 const speakerImgFile = formData.get("vid-resource-speaker-img");
59 if (speakerImgFile && speakerImgFile.name && speakerImgFile.size > 0) {
60   const path = `${Date.now()}-${speakerImgFile.name.replace(/^[a-zA-Z0-9.-]/g, "_")}`;
61   const { error: uploadError } = await supabase.storage
62     .from('video-resources')
63     .upload(path, speakerImgFile);
64
65   if (uploadError) {
66     console.error("Error uploading speaker image:", uploadError);
67     return uploadError;
68   }
69
70   const { data, error: urlError } = supabase.storage
71     .from('video-resources')
72     .getPublicUrl(path);
73
74   if (urlError) {
75     console.error("Error getting speaker image URL:", urlError);
76     return urlError;
77   }
78
79   speakerImgUrl = data.publicUrl;
80 }
81
82 const { error } = await supabase
83   .from('video resources')
84   .insert({
85     title: formData.get("vid-resource-title"),
86     video_url: formData.get("vid-resource-link"),
87     date: formData.get("vid-resource-date"),
88     video_thumbnail: thumbnailUrl,
89     video_description: formData.get("vid-resource-desc"),
90     speaker: formData.get("vid-resource-speaker-name"),
91     speaker_university: formData.get("vid-resource-speaker-uni"),
92     speaker_image: speakerImgUrl,
93     speaker_details: formData.get("vid-resource-speaker-desc"),
94   })
95
96 if (error) {
97   console.error("Database insert error:", error);
98 }

```

```

99
100     return error;
101 }
102
103 function ResourceCard({ resourceInfo }) {
104     resourceInfo.video_thumbnail = (resourceInfo.video_thumbnail == "{}") ? null : resourceInfo.
        video_thumbnail;
105     return (
106         <div className="resource-card">
107             <a href={"video-resource/" + resourceInfo.id} className="block w-full p-2 border-2
                border-transparent hover:border-primary-light rounded-md">
108                 <div className="w-full lg:h-[14vw] sm:h-[28vw] h-[52vw] rounded-md mb-2 overflow-
                    hidden bg-primary-dark flex items-center">
109                     {resourceInfo.video_thumbnail ?
110                         <img className="min-w-full grow-0 shrink-0" src={resourceInfo?.
                            video_thumbnail} />
111                         :
112                         <div className="relative w-full h-full p-5">
113                             
114                             <h5 className="relative z-1" style={{ color: "var(--color-white)"
                                    }}>{resourceInfo.title}</h5>
115                             <p style={{ color: "var(--color-white)" }}>{resourceInfo.date.replace
                                    (/-/g, '\')}</p>
116                             </div>
117                         }
118                     </div>
119                     <h6>{resourceInfo.title}</h6>
120                     <p className="font-semibold text-black"><i>{resourceInfo.speaker}</i></p>
121                     <p className="text-sm text-disabled-light"><i>{resourceInfo.speaker_university}</
                        i></p>
122                 </a>
123             </div>
124         )
125 }
126
127 export default function VideoResources({ loaderData }) {
128     const navigate = useNavigate();
129     const [isAdmin, setIsAdmin] = useState(false);
130     const [showCreatePopup, setShowCreatePopup] = useState(false);
131     const [currentPage, setCurrentPage] = useState(0);
132
133     useEffect(() => {
134         const getIsAdmin = async (userId) => {
135             try {
136                 const { data, error } = await supabase
137                     .from('users')
138                     .select('role')
139                     .eq("id", userId);
140                 if (data[0]) {
141                     setIsAdmin(data[0].role == "admin");
142                 }
143                 else { console.log("error"); }
144             } catch (error) {
145                 console.log("error");
146             }
147         }
148     })
149
150     // Listen for auth state changes
151     const { data: { subscription } } = supabase.auth.onAuthStateChange((_event, session) => {
152         if (session?.user.id) {
153             getIsAdmin(session?.user.id);
154         }
155     });

```

```

156
157 // Check current session on mount
158 supabase.auth.getSession().then(({ data: { session } }) => {
159   if (session?.user.id) {
160     getIsAdmin(session?.user.id);
161   }
162 });
163
164
165   return () => subscription.unsubscribe();
166 }, []);
167
168 const videoResourcesData = loaderData.videoResources.sort(function (a, b) { return new Date(b
169   .date) - new Date(a.date) });
170
171 const videoResources = [];
172 for (let i = (currentPage * 6); i < currentPage + 6; i++) {
173   if (i < videoResourcesData.length) {
174     videoResources.push(<ResourceCard key={videoResourcesData[i].id} resourceInfo={
175       videoResourcesData[i]} />)
176   } else {
177     break;
178   }
179 }
180
181 // EVERYTHING BELOW IS POPUP EDIT THINGS
182 -----
183 const [showResolvePopup, setShowResolvePopup] = useState(false);
184
185 const [formRequired, setFormRequired] = useState({
186   vidResourceTitle: "",
187   vidResourceLink: "",
188   vidResourceSpeakerName: "",
189 });
190 const [hasError, setHasError] = useState(false);
191
192 async function validate(formData) {
193   let isValidated = true;
194   const isRequired = {
195     vidResourceTitle: formData.get('vid-resource-title') === (null || ""),
196     vidResourceLink: formData.get('vid-resource-link') === (null || ""),
197     vidResourceSpeakerName: formData.get('vid-resource-speaker-name') === (null || "")
198   }
199   for (let value of Object.values(isRequired)) {
200     if (value) {
201       isValidated = false;
202       break;
203     }
204   }
205   if (!isValidated) {
206     setFormRequired(isRequired);
207     return false;
208   }
209
210   const createError = await createVideoResource(formData);
211   if (createError === null) {
212     setShowResolvePopup(true);
213   } else {
214     console.error("Failed to create video resource:", createError);
215     setHasError(true);
216   }
217 }
218
219 // This resets formRequired so that there are no error messages when the form is reloaded

```

```

218 function handleShowCreatePopupForm() {
219     setFormRequired({
220         vidResourceTitle: "",
221         vidResourceLink: "",
222         vidResourceSpeakerName: ""
223     })
224     setShowCreatePopup(true);
225 }
226
227 // This resolves required fields once fields are no longer empty
228 function checkEmpty(value, inputName) {
229     const updatedFormRequired = updateRequired(value, inputName, formRequired);
230     if (updatedFormRequired !== formRequired) {
231         setFormRequired(updatedFormRequired);
232     }
233 }
234
235 const today = new Date();
236
237 let day = today.getDate();
238 let month = today.getMonth() + 1;
239 let year = today.getFullYear();
240
241 if (month < 10) month = "0" + month;
242 if (day < 10) day = "0" + day;
243
244 const todayString = `${year}-${month}-${day}`;
245
246 function closeResolvePopup() {
247     setShowResolvePopup(false);
248     setShowCreatePopup(false);
249     navigate(0);
250 }
251
252 return (
253     <>
254         {isAdmin &&
255             <div className="z-1000 absolute top-0 left-0">
256                 <PopupForm id="video-resource" show={showCreatePopup} setShow={
257                     setShowCreatePopup} validate={validate} hasError={hasError} encType="multipart/form-data">
258                     <h4>Create new video resource</h4>
259                     <div className="grid md:grid-cols-2 grid-cols-1 gap-5 mb-5 relative">
260                         <div>
261                             <label htmlFor="vid-resource-title">Video resource title:</label>
262                             <input id="vid-resource-title" name="vid-resource-title" type="
263                                 text"
264                                 className={"input input-text w-full " + (formRequired?.
265                                     vidResourceTitle && "input-required")}
266                                 placeholder="Video title"
267                                 onChange={(e) => checkEmpty(e.target.value, "vidResourceTitle
268                                     ")} />
269                             <div className="input-error">This field is required.</div>
270                             <br /><br />
271                             <label htmlFor="vid-resource-date">Video resource date:</label><
272                                 br />
273                                 <input id="vid-resource-date" name="vid-resource-date" type="date
274                                     " className="input input-text w-full" defaultValue={todayString} />
275                                 </div>
276                                 <label>
277                                     Video resource thumbnail image:
278                                     <input id="vid-resource-thumbnail" name="vid-resource-thumbnail"
279                                     type="file" accept=".jpg,.jpeg,.png" className="ml-3" />
280                                     <div className="input-error">This field is required.</div>
281                                 </label>

```



```

275         </div>
276
277         <div className="relative">
278             <label htmlFor="vid-resource-link">Video resource link:</label><br />
279             <p className="text-sm text-disabled-dark mt-1">This is the video link
that will be embedded. Please only include the link and not the entire embed.</p>
280             <input id="vid-resource-link" name="vid-resource-link" type="text"
281                 className={"input input-text w-full " + (formRequired?.
vidResourceLink && "input-required")}>
282                 placeholder="e.g. https://www.youtube.com/embed/VIDEO_ID or https
://drive.google.com/file/d/...."
283                 onChange={(e) => checkEmpty(e.target.value, "vidResourceLink")}
/>
284             <div className="input-error">This field is required.</div>
285             <br /><br />
286             <label htmlFor="vid-resource-desc">Video description:</label><br />
287             <textarea id="vid-resource-desc" name="vid-resource-desc" className="
input input-text w-full h-30" placeholder="Enter your video description..."></textarea>
288             <br /> <br />
289         </div>
290
291         <fieldset className="border-t-2 border-gray-light pt-4 relative">
292             <div className="text-sm font-semibold text-secondary-dark">Speaker
Details</div>
293
294             <div className="mt-3 grid md:grid-cols-2 grid-cols-1 gap-5">
295                 <div>
296                     <label htmlFor="vid-resource-speaker-name">Name:</label><br
/>
297                     <input id="vid-resource-speaker-name" name="vid-resource-
speaker-name" type="text"
298                         className={"input input-text w-full " + (formRequired?.
vidResourceSpeakerName && "input-required")}>
299                         placeholder="Name"
300                         onChange={(e) => checkEmpty(e.target.value, "
vidResourceSpeakerName")} />
301                     <div className="input-error">This field is required.</div>
302                     <div>
303                         <label htmlFor="vid-resource-speaker-uni">University:</label>
304                         <input id="vid-resource-speaker-uni" name="vid-resource-
speaker-uni" type="text" className="input input-text w-full" placeholder="University" />
305                     </div>
306                 </div>
307                 <br />
308                 <label>
309                     Speaker image:
310                     <input id="vid-resource-speaker-img" name="vid-resource-speaker-
img" type="file" accept=".jpg,.jpeg,.png" className="ml-3" />
311                     <div className="input-error">This field is required.</div>
312                 </label>
313                 <br /><br />
314                 <label htmlFor="vid-resource-speaker-desc">Description:</label><br />
315                 <textarea id="vid-resource-speaker-desc" name="vid-resource-speaker-
desc" className="input input-text w-full h-20" placeholder="Enter your speaker description..."
></textarea>
316             </fieldset>
317         </PopupForm>
318         <Popup id="resolve" className="text-center" show={showResolvePopup} setShow={
setShowResolvePopup} closePopup={closeResolvePopup} nested stayOnBlur>
319             <br /><p className="m-2">New video resource created!</p>
320         </Popup>
321     </div>
322 }
323 <Menu currentEndUrl="/video-resources" />

```

```

324         <div className="py-20 px-10 lg:px-40 duration-200">
325             <div className="flex justify-between md:items-center md:flex-row flex-col md:mb-0
mb-5">
326                 <h1>Video Resources</h1>
327                 {isAdmin && <button className="button" onClick={handleShowCreatePopupForm}><i
className="bi bi-plus-lg mr-3"></i>Create new video resource</button>}
328             </div>
329             <p>
330                 Here you will find some videos provided by members of our network that you
can use for your own training and also share with your students.
331             </p>
332             <div className="my-5 grid lg:grid-cols-3 sm:grid-cols-2 grid-cols-1 gap-x-5 gap-y
-7">
333                 {videoResources}
334             </div>
335             <Pagination currentPage={currentPage} setCurrentPage={setCurrentPage} totalItems
={videoResourcesData.length} itemsPerPage={6} pageRange={5} />
336         </div>
337         <Footer />
338     </>
339 );
340 }

```

A.6 video-resource.jsx

```

1 import { useState, useEffect } from "react";
2 import { useNavigate } from "react-router";
3 import { supabase } from "../supabase";
4 import { Menu } from "../components/menu";
5 import { Banner } from "../components/graphics";
6 import { Footer } from "../components/footer";
7 import { Popup, PopupForm } from "../components/popup";
8 import { updateRequired } from "../helpers/form";
9 import "../styles/video-resources.css";
10
11 export function meta() {
12     return [
13         { title: "Video Resource" },
14         // { name: "description", content: "Welcome to React Router!" },
15     ];
16 }
17
18 async function getVideoResource(resourceId) {
19     const { data, error } = await supabase
20         .from('video resources')
21         .select()
22         .eq('id', resourceId)
23
24     if (error) {
25         console.error("Error fetching video resource:", error);
26         return null;
27     }
28
29     return data ? data[0] : null;
30 }
31
32 function extractVideoResourcePath(url) {
33     if (!url) return null;
34     const parts = url.split('/video-resources/');
35     return parts.length > 1 ? parts[parts.length - 1] : null;
36 }
37

```

```

38 async function removeVideoResourceFiles(urls) {
39   const paths = urls.map(extractVideoResourcePath).filter(Boolean);
40   if (paths.length === 0) return;
41   const { error } = await supabase.storage.from('video-resources').remove(paths);
42   if (error) {
43     console.error("Error deleting files from storage:", error);
44   }
45 }
46
47 async function deleteVideoResource(resourceId, thumbnailUrl, speakerImgUrl) {
48   await removeVideoResourceFiles([thumbnailUrl, speakerImgUrl]);
49
50   const response = await supabase
51     .from('video resources')
52     .delete()
53     .eq('id', resourceId)
54
55   return response;
56 }
57
58 async function updateVideoResource(resourceId, formData, existingData) {
59   let thumbnailUrl = existingData.video_thumbnail;
60   let speakerImgUrl = existingData.speaker_image;
61   const filesToRemoveOnSuccess = [];
62
63   const thumbnailFile = formData.get("vid-resource-thumbnail");
64   if (thumbnailFile && thumbnailFile.name && thumbnailFile.size > 0) {
65     const path = `${Date.now()}-${thumbnailFile.name.replace(/^[a-zA-Z0-9.-]/g, "_")}`;
66     const { error: uploadError } = await supabase.storage
67       .from('video-resources')
68       .upload(path, thumbnailFile);
69
70     if (uploadError) {
71       console.error("Error uploading thumbnail:", uploadError);
72       return uploadError;
73     }
74
75     const { data, error: urlError } = supabase.storage
76       .from('video-resources')
77       .getPublicUrl(path);
78
79     if (urlError) {
80       console.error("Error getting thumbnail URL:", urlError);
81       return urlError;
82     }
83
84     if (existingData.video_thumbnail) {
85       filesToRemoveOnSuccess.push(existingData.video_thumbnail);
86     }
87     thumbnailUrl = data.publicUrl;
88   }
89
90   const speakerImgFile = formData.get("vid-resource-speaker-img");
91   if (speakerImgFile && speakerImgFile.name && speakerImgFile.size > 0) {
92     const path = `${Date.now()}-${speakerImgFile.name.replace(/^[a-zA-Z0-9.-]/g, "_")}`;
93     const { error: uploadError } = await supabase.storage
94       .from('video-resources')
95       .upload(path, speakerImgFile);
96
97     if (uploadError) {
98       console.error("Error uploading speaker image:", uploadError);
99       return uploadError;
100   }
101
102   const { data, error: urlError } = supabase.storage

```

```

103         .from('video-resources')
104         .getPublicUrl(path);
105
106     if (urlError) {
107         console.error("Error getting speaker image URL:", urlError);
108         return urlError;
109     }
110
111     if (existingData.speaker_image) {
112         filesToRemoveOnSuccess.push(existingData.speaker_image);
113     }
114     speakerImgUrl = data.publicUrl;
115 }
116
117 const { data, error } = await supabase
118     .from('video resources')
119     .update({
120         title: formData.get("vid-resource-title"),
121         video_url: formData.get("vid-resource-link"),
122         date: formData.get("vid-resource-date"),
123         video_thumbnail: thumbnailUrl,
124         video_description: formData.get("vid-resource-desc"),
125         speaker: formData.get("vid-resource-speaker-name"),
126         speaker_university: formData.get("vid-resource-speaker-uni"),
127         speaker_image: speakerImgUrl,
128         speaker_details: formData.get("vid-resource-speaker-desc"),
129     })
130     .eq('id', resourceId)
131     .select();
132
133 if (error) {
134     console.error("Database update error:", error);
135     return error;
136 }
137
138 if (!data || data.length === 0) {
139     const rlsError = new Error("No rows updated. You may be missing an UPDATE policy in Supabase.");
140     console.error(rlsError.message);
141     return rlsError;
142 }
143
144 await removeVideoResourceFiles(filesToRemoveOnSuccess);
145
146 return null;
147 }
148
149 export async function loader({ params }) {
150     const vid = await getVideoResource(params.vidId);
151
152     if (!vid) {
153         throw new Response("Video resource not found", { status: 404 });
154     }
155
156     // Ensure expected arrays/fields exist to avoid runtime errors when mapping
157     vid.title = vid.title || "";
158     vid.date = (vid.date || "").replace(/-/g, '\\') || "";
159     vid.speaker = vid.speaker || "";
160     vid.speaker_university = vid.speaker_university || "";
161     vid.speaker_details = vid.speaker_details || "";
162     vid.speaker_image = (vid.speaker_image == "{}") ? null : vid.speaker_image;
163     vid.video_url = vid.video_url || null;
164     vid.video_thumbnail = (vid.video_thumbnail == "{}") ? null : vid.video_thumbnail;
165     vid.video_description = vid.video_description || "";
166     return vid;

```

```

167 }
168
169 export default function VideoResource({ loaderData }) {
170     const navigate = useNavigate();
171     const [showDeletePopup, setShowDeletePopup] = useState(false);
172     const [isAdmin, setIsAdmin] = useState(false);
173
174     // EDIT LOGIC ADDITION
175     const [showEditPopup, setShowEditPopup] = useState(false);
176     const [showResolvePopup, setShowResolvePopup] = useState(false);
177     const [formRequired, setFormRequired] = useState({
178         vidResourceTitle: false,
179         vidResourceLink: false,
180         vidResourceSpeakerName: false
181     });
182     const [hasError, setHasError] = useState(false);
183
184     async function validate(formData) {
185         let isValidated = true;
186         const isRequired = {
187             vidResourceTitle: formData.get('vid-resource-title') === (null || ""),
188             vidResourceLink: formData.get('vid-resource-link') === (null || ""),
189             vidResourceSpeakerName: formData.get('vid-resource-speaker-name') === (null || "")
190         }
191         for (let value of Object.values(isRequired)) {
192             if (value) {
193                 isValidated = false;
194                 break;
195             }
196         }
197         if (!isValidated) {
198             setFormRequired(isRequired);
199             return false;
200         }
201
202         const updateError = await updateVideoResource(loaderData.id, formData, loaderData);
203         if (updateError === null) {
204             setShowResolvePopup(true);
205         } else {
206             console.error("Failed to update video resource:", updateError);
207             setHasError(true);
208         }
209     }
210
211     function handleShowEditPopupForm() {
212         setFormRequired({
213             vidResourceTitle: false,
214             vidResourceLink: false,
215             vidResourceSpeakerName: false
216         });
217         setHasError(false);
218         setShowEditPopup(true);
219     }
220
221     function checkEmpty(value, inputName) {
222         const updatedFormRequired = updateRequired(value, inputName, formRequired);
223         if (updatedFormRequired !== formRequired) {
224             setFormRequired(updatedFormRequired);
225         }
226     }
227
228     function closeResolvePopup() {
229         setShowResolvePopup(false);
230         setShowEditPopup(false);
231         navigate(0);

```

```

232 }
233
234 useEffect(() => {
235     const getIsAdmin = async (userId) => {
236         try {
237             const { data, error } = await supabase
238                 .from('users')
239                 .select('role')
240                 .eq("id", userId);
241             if (data[0]) {
242                 setIsAdmin(data[0].role == "admin");
243             }
244             else { console.log("error"); }
245         } catch (error) {
246             console.log("error");
247         }
248     }
249 }
250
251 // Listen for auth state changes
252 const { data: { subscription } } = supabase.auth.onAuthStateChange((_event, session) => {
253     if (session?.user.id) {
254         setIsAdmin(session?.user.id);
255     }
256 });
257
258 // Check current session on mount
259 supabase.auth.getSession().then(({ data: { session } }) => {
260     if (session?.user.id) {
261         setIsAdmin(session?.user.id);
262     }
263 });
264
265 return () => subscription.unsubscribe();
266 }, []);
267
268 async function handleDelete() {
269     try {
270         await deleteVideoResource(loaderData.id, loaderData.video_thumbnail, loaderData.
271 speaker_image);
272         navigate({ pathname: "/video-resources" });
273     }
274     catch (error) {
275         console.error("Error during deletion:", error);
276     }
277 }
278
279 return (
280     <>
281     <Popup id="delete-vidr" show={showDeletePopup} setShow={setShowDeletePopup}
282         buttons={[{ text: "Delete", onclick: handleDelete }]}>
283         <div className="text-center mt-6">Delete this video resource?</div>
284     </Popup>
285
286     {isAdmin &&
287         <div className="z-1000 absolute top-0 left-0">
288             <PopupForm id="edit-video-resource" show={showEditPopup} setShow={
289 setShowEditPopup} validate={validate} hasError={hasError} encType="multipart/form-data">
290                 <h4>Edit video resource</h4>
291                 <div className="grid md:grid-cols-2 grid-cols-1 gap-5 mb-5 relative">
292                     <div>
293                         <label htmlFor="vid-resource-title">Video resource title:</label>

```

```

text"
294         className={"input input-text w-full " + (formRequired?.
vidResourceTitle && "input-required")}]
295         placeholder="Video title"
296         defaultValue={loaderData.title}
297         onChange={(e) => checkEmpty(e.target.value, "vidResourceTitle
")} />
298         <div className="input-error">This field is required.</div>
299         <br /><br />
300         <label htmlFor="vid-resource-date">Video resource date:</label><
br />
301         {/* Expected format for <input type="date"> is YYYY-MM-DD */}
302         <input id="vid-resource-date" name="vid-resource-date" type="date
" className="input input-text w-full" defaultValue={loaderData.date.replace(/\//g, '-')}/>
303         </div>
304         <label>
305         Video resource thumbnail image:
306         <p className="text-sm text-disabled-dark">Leave empty to keep
existing thumbnail.</p>
307         <input id="vid-resource-thumbnail" name="vid-resource-thumbnail"
type="file" accept=".jpg,.jpeg,.png" className="ml-3" />
308         </label>
309         </div>
310
311         <div className="relative">
312         <label htmlFor="vid-resource-link">Video resource link:</label><br />
313         <p className="text-sm text-disabled-dark mt-1">This is the video link
that will be embedded. Please only include the link and not the entire embed.</p>
314         <input id="vid-resource-link" name="vid-resource-link" type="text"
315         className={"input input-text w-full " + (formRequired?.
vidResourceLink && "input-required")}]
316         placeholder="e.g. https://www.youtube.com/embed/VIDEO_ID or https
://drive.google.com/file/d/...."
317         defaultValue={loaderData.video_url}
318         onChange={(e) => checkEmpty(e.target.value, "vidResourceLink")}
/>
319         <div className="input-error">This field is required.</div>
320         <br /><br />
321         <label htmlFor="vid-resource-desc">Video description:</label><br />
322         <textarea id="vid-resource-desc" name="vid-resource-desc" className="
input input-text w-full h-30" placeholder="Enter your video description..." defaultValue={
loaderData.video_description}></textarea>
323         <br /> <br />
324         </div>
325
326         <fieldset className="border-t-2 border-gray-light pt-4 relative">
327         <div className="text-sm font-semibold text-secondary-dark">Speaker
Details</div>
328         <div className="mt-3 grid md:grid-cols-2 grid-cols-1 gap-5">
329         <div>
330         <label htmlFor="vid-resource-speaker-name">Name:</label><br
/>
331         <input id="vid-resource-speaker-name" name="vid-resource-
speaker-name" type="text"
332         className={"input input-text w-full " + (formRequired?.
vidResourceSpeakerName && "input-required")}]
333         placeholder="Name"
334         defaultValue={loaderData.speaker}
335         onChange={(e) => checkEmpty(e.target.value, "
vidResourceSpeakerName")}/>
336         <div className="input-error">This field is required.</div>
337         </div>
338         <div>
339         <label htmlFor="vid-resource-speaker-uni">University:</label
><br />

```

```

340         <input id="vid-resource-speaker-uni" name="vid-resource-
speaker-uni" type="text" className="input input-text w-full" placeholder="University"
defaultValue={loaderData.speaker_university} />
341     </div>
342 </div>
343 <br />
344 <label>
345     Speaker image:
346     <p className="text-sm text-disabled-dark">Leave empty to keep
existing speaker image.</p>
347     <input id="vid-resource-speaker-img" name="vid-resource-speaker-
img" type="file" accept=".jpg,.jpeg,.png" className="ml-3" />
348 </label>
349 <br /><br />
350 <label htmlFor="vid-resource-speaker-desc">Description:</label><br />
351 <textarea id="vid-resource-speaker-desc" name="vid-resource-speaker-
desc" className="input input-text w-full h-20" placeholder="Enter your speaker description..."
defaultValue={loaderData.speaker_details}></textarea>
352 </fieldset>
353 </PopupForm>
354 <Popup id="resolve" className="text-center" show={showResolvePopup} setShow={
setShowResolvePopup} closePopup={closeResolvePopup} nested stayOnBlur>
355     <br /><p className="m-2">Video resource updated successfully!</p>
356 </Popup>
357 </div>
358 }
359
360 <Menu />
361 <Banner type="blue">
362     <div className="relative z-1">
363         <a href="/video-resources" className="banner-breadcrumb">
364             <i className="bi bi-caret-left-fill"></i>
365             <strong>VIDEO RESOURCES</strong>
366         </a>
367         <h1 style={{ color: "white", textTransform: "none !important" }}>{loaderData.
title}</h1>
368         <p>
369             <span className="text-lg">By {loaderData.speaker}</span> <span className=
"ml-5 opacity-70"><i>{loaderData.date}</i></span>
370         </p>
371     </div>
372 </Banner>
373
374 <div className="py-20 px-10 lg:px-40 duration-200">
375     {isAdmin ? (
376         <div className="text-right mb-4 flex justify-end gap-2">
377             <button className="button button-light" onClick={handleShowEditPopupForm
}>Edit Video Resource</button>
378             <button className="button button-light button-red" onClick={() =>
setShowDeletePopup(true)}>Delete Video Resource</button>
379         </div>
380     ) : <></>
381     <div className="mb-5 w-full lg:h-[40vw] h-[50vw]">
382         <iframe src={loaderData.video_url} width="100%" height="100%"></iframe>
383     </div>
384     <p>{loaderData.video_description}</p>
385
386     <div className="relative mt-5 rounded-md border-2 border-gray-light p-5 flex flex
-col md:flex-row place-items-center">
387         {loaderData.speaker_image} <img className="mx-auto w-50 shrink-0 grow-0"
src={loaderData.speaker_image} alt="" />
388         <div className="w-full md:w-70 shrink-0 grow-0 m-3">
389             <p className="font-semibold mr-2"><i>{loaderData.speaker}</i></p>
390             <p className="text-disabled-light">{loaderData.speaker_university}</p>
391         </div>

```



```

392         <p>{loaderData.speaker_details}</p>
393     </div>
394
395     </div>
396
397     <Footer />
398 </>
399 )
400 }

```

A.7 menu.jsx

```

1  import { useState, useEffect } from "react";
2  import { NavLink, useNavigate } from "react-router";
3  import { supabase } from "../supabase";
4  import { Popup } from "../components/popup";
5
6  const LANGUAGES = [
7    { code: "en", label: "English" },
8    { code: "es", label: "Español" },
9    { code: "fr", label: "Français" },
10   { code: "de", label: "Deutsch" },
11   { code: "it", label: "Italiano" },
12   { code: "pt", label: "Português" },
13   { code: "ja", label: "日本語" },
14   { code: "zh-CN", label: "简体中文" },
15   { code: "ar", label: "العربية" },
16   { code: "hi", label: "हिन्दी" },
17 ];
18
19 let googleLoaded = false;
20
21 function loadGoogleTranslate() {
22   if (googleLoaded) return;
23   googleLoaded = true;
24
25   const addScript = () => {
26     const script = document.createElement("script");
27     script.src =
28       "https://translate.google.com/translate_a/element.js?cb=googleTranslateElementInit";
29     script.async = true;
30     document.body.appendChild(script);
31   };
32
33   window.googleTranslateElementInit = () => {
34     new window.google.translate.TranslateElement(
35       {
36         pageLanguage: "en",
37         autoDisplay: false,
38       },
39       "google_translate_element"
40     );
41   };
42
43   addScript();
44 }
45
46 export function Menu({ currentEndUrl }) {
47   const navigate = useNavigate();
48   const [currentUser, setCurrentUser] = useState(null);
49   const loggedIn = !!currentUser;
50   const [sideMenuActive, setSideMenuActive] = useState(false);

```

```

51  const [userInfo, setUserInfo] = useState(null);
52  const [signOutPopup, setSignOutPopup] = useState(false);
53  const [language, setLanguage] = useState("en");
54
55  let isGroupActive = {
56    whatWeDo: false,
57    resources: false,
58    meetings: false
59  }
60
61  switch (currentEndUrl) {
62    case "/about":
63    case "/organizational-structure":
64    case "/newsletter":
65      isGroupActive.whatWeDo = true;
66      break;
67    case "/video-resources":
68    case "/webinars":
69      isGroupActive.resources = true;
70      break;
71    case "/regional-meetings":
72    case "/international-meetings":
73      isGroupActive.meetings = true;
74      break;
75  }
76
77  useEffect(() => {
78    // Create hidden container AFTER hydration
79    const el = document.createElement("div");
80    el.id = "google_translate_element";
81    el.style.display = "none";
82    document.body.appendChild(el);
83
84    loadGoogleTranslate();
85  }, []);
86
87  useEffect(() => {
88    const savedLang = localStorage.getItem("lang");
89
90    if (savedLang && savedLang !== "en") {
91      document.cookie = 'googtrans=/en/${savedLang}; path=/';
92    }
93
94    // Set dropdown UI state only
95    if (savedLang) {
96      setLanguage(savedLang);
97    }
98  }, []);
99
100  useEffect(() => {
101
102    const user = async (userId) => {
103      try {
104        const { data, error } = await supabase
105          .from('users')
106          .select('fname, lname, email, image_url')
107          .eq("id", userId);
108        if (data[0]) {
109          setUserInfo(data[0]);
110        }
111        else { console.log("error"); }
112      } catch (error) {
113        console.log("error");
114      }
115    }

```

```

116 }
117
118 // Listen for auth state changes
119 const { data: { subscription } } = supabase.auth.onAuthStateChange((_event, session) => {
120   setCurrentUser(session?.user ?? null);
121   if (session?.user.id) {
122     user(session.user.id);
123   }
124 });
125
126 // Check current session on mount
127 supabase.auth.getSession().then(({ data: { session } }) => {
128   setCurrentUser(session?.user ?? null);
129   user(session.user.id);
130 });
131
132 return () => subscription.unsubscribe();
133 }, []);
134
135
136 const handleSignOut = async () => {
137   await supabase.auth.signOut();
138   setCurrentUser(null);
139   setSignOutPopup(true);
140 };
141
142 const changeLanguage = (e) => {
143   const lang = e.target.value;
144
145   // set the cookie that Google itself uses
146   document.cookie = 'googtrans=/en/${lang}; path=/';
147   document.cookie = 'googtrans=/en/${lang}; domain=${window.location.hostname}; path=/';
148
149   // Store for your dropdown UI only
150   localStorage.setItem("lang", lang);
151
152   // Reload so Google re-runs cleanly
153   window.location.reload();
154 };
155
156 useEffect(() => {
157
158   const menuDropdownEls = document.getElementsByClassName("menu-dropdown-container");
159
160   for (let menuDropdown of menuDropdownEls) {
161     menuDropdown.addEventListener("mouseout", function () { menuDropdown.classList.remove("active-popup"); });
162     menuDropdown.addEventListener("mouseover", function () { menuDropdown.classList.add("active-popup"); });
163   }
164
165   const sideMenuDropdownEls = document.getElementsByClassName("side-menu-dropdown-container");
166   for (let sideMenuDropdown of sideMenuDropdownEls) {
167     sideMenuDropdown.addEventListener("mouseout", function () { sideMenuDropdown.classList.remove("active-popup"); });
168     sideMenuDropdown.addEventListener("mouseover", function () { sideMenuDropdown.classList.add("active-popup"); });
169   }
170
171   if (sideMenuActive) {
172     const sideMenuContEl = document.getElementById("side-menu-container");
173     sideMenuContEl.children[1].classList.add("left-0");
174     sideMenuContEl.children[0].classList.remove("hidden");
175   } else {
176     const sideMenuContEl = document.getElementById("side-menu-container");

```

```

177     sideMenuContEl.children[1].classList.remove("left-0");
178     sideMenuContEl.children[0].classList.add("hidden");
179 }
180
181 }, [sideMenuActive, currentUser])
182
183
184 return (
185   <>
186     <Popup show={signOutPopup} setShow={setSignOutPopup} closePopup={() => { setSignOutPopup(
187       false); navigate("/"); }} stayOnBlur={true}>
188       <p className="text-center mt-6">You have been signed out.</p>
189     </Popup>
190     <div id="menu" className="sticky top-0 bg-white z-100">
191       <div className="border-b border-b-gray-light h-6 relative z-999 flex flex-row justify-end
192         ">
193         { /* Secondary Header */ }
194         { /* LinkedIn Social Media Link */ }
195         <a
196           href="https://www.linkedin.com/company/iajes"
197           target="_blank"
198           rel="noopener noreferrer"
199           className="mr-3 flex items-center text-zinc-400 hover:text-[#0A66C2] transition-
200             colors"
201           aria-label="LinkedIn"
202         >
203           <i className="bi bi-linkedin text-[0.9rem]"></i>
204         </a>
205         { /* Google Translate Dropdown */ }
206         <select
207           value={language}
208           onChange={changeLanguage}
209           className="link mr-3 text-[0.75rem]! z-100 notranslate text-zinc-400!"
210         >
211           {LANGUAGES.map((l) => (
212             <option key={l.code} value={l.code}>
213               {l.label}
214             </option>
215           ))}
216         </select>
217       </div>
218
219       <div className="relative flex items-center content-center md:justify-start justify-
220         between shadow-sm z-1 duration-200">
221         <button onClick={() => setSideMenuActive(!sideMenuActive)} className="relative md:
222           hidden block px-4 text-primary-dark hover:cursor-pointer hover:text-secondary-light duration
223           -200 bg-white z-1">
224           <i className="bi bi-list text-[1.7rem]"></i>
225         </button>
226         <NavLink to="/" end className="relative duration-200 hover:opacity-70 md:px-4 bg-white
227           z-1 my-1 md:mr-auto">
228           
229           { /*  */ }
231         </NavLink>
232
233
234         <div className="relative flex items-center justify-end w-130 hidden md:flex">
235           <div className="menu-dropdown-container">
236             <div className={ "menu-dropdown-button link p-4" + (isGroupActive.whatWeDo ? "
237               active" : "")}>
238               What We Do
239             </div>
240             <div className="menu-dropdown -mt-[240px] py-1">
241               <NavLink to="/about" end className="link py-3 px-4 pr-10">

```

```

233         About IAJES
234     </NavLink>
235     <NavLink to="/organizational-structure" end className="link py-3 px-4 pr-10">
236         Organizational Structure
237     </NavLink>
238     <NavLink to="/newsletter" className="link py-3 px-4 pr-10">
239         Newsletter
240     </NavLink>
241 </div>
242 </div>
243
244     <NavLink to="/task-forces" className="link p-4">
245         Task Forces
246     </NavLink>
247
248     <div className="menu-dropdown-container">
249         <div className={"menu-dropdown-button link p-4" + (isGroupActive.resources ? "
active" : "")}>
250             Resources
251         </div>
252         <div className="menu-dropdown -mt-[220px] py-1">
253             <NavLink to="/video-resources" end className="link py-3 px-4 pr-10">
254                 Video Resources
255             </NavLink>
256             <NavLink to="/webinars" end className="link py-3 px-4 pr-10">
257                 Webinars
258             </NavLink>
259         </div>
260     </div>
261
262     <div className="menu-dropdown-container">
263         <div className={"menu-dropdown-button link p-4" + (isGroupActive.meetings ? "
active" : "")}>
264             Meetings
265         </div>
266         <div className="menu-dropdown w-45 -mt-[220px] py-1">
267             <NavLink to="/regional-meetings" end className="link py-3 px-4">
268                 Regional Meetings
269             </NavLink>
270             <NavLink to="/international-meetings" end className="link py-3 px-4">
271                 International Meetings
272             </NavLink>
273         </div>
274     </div>
275
276     <NavLink to="/search" end className="link p-4">
277         People
278     </NavLink>
279
280 </div>
281
282 {!loggedIn ?
283     <NavLink to="/signin" end className="block button m-1.5">
284         Sign In
285     </NavLink>
286     :
287     <div className="menu-dropdown-container">
288         <div className="menu-dropdown-button rounded-full border-2 border-primary-dark
overflow-hidden bg-primary-dark size-11 my-1.5 ml-2 mr-4 flex items-center justify-center
text-white hover:text-primary-light hover:border-secondary-light hover:bg-secondary-light
duration-200 cursor-pointer">
289             {(userInfo?.image_url != null && userInfo?.image_url != "") ?
290                 <img className="hover:opacity-90 duration-200 min-w-full min-h-full" src={
userInfo.image_url} />
291                 :

```

```

292         <i className="bi bi-person-fill text-[1.5rem]"></i>
293     </div>
294     <div className="menu-dropdown right-3 py-1 -mt-[220px]">
295         <div className="text-sm py-3 px-3 mx-2 mb-2 border-b-2 border-primary-light flex
items-center">
296
297             <div className="mr-4">
298                 <div className="rounded-full bg-primary-dark size-15 border-2 border-primary-
dark overflow-hidden flex items-center justify-center">
299                     {(userInfo?.image_url != null && userInfo?.image_url != "") ?
300                         <img className="min-w-full min-h-full" src={userInfo.image_url} />
301                         :
302                         <i className="bi bi-person-fill text-[2rem] text-white"></i>}
303                     </div>
304                 </div>
305                 <div className="pr-6">
306                     {userInfo?.fname} {userInfo?.lname}
307                     <br />
308                     <span className="text-disabled-light italic">{userInfo?.email}</span>
309                 </div>
310
311             </div>
312             <NavLink to={"/profile/" + currentUser?.id} end className="link py-3 px-4 pr-10">
313                 Profile
314             </NavLink>
315             <div className="px-4 pt-2 pb-3">
316                 <button onClick={handleSignOut} className="block button w-full">
317                     Sign Out
318                 </button>
319             </div>
320         </div>
321     </div>
322 }
323
324 </div>
325
326 <div id="side-menu-container" className="absolute w-full z-0">
327     <div id="side-menu-bg" className="absolute w-full h-svh top-0 left-0 z-0 bg-white
opacity-10 hidden" onClick={() => setSideMenuActive(false)}></div>
328     <div id="side-menu" className="absolute bg-white flex flex-col w-70 h-dvh duration-400
ease-in-out shadow-md z-1 -left-70" >
329
330         <div className="side-menu-dropdown-container mt-7 pb-3 px-7">
331             <div className={"side-menu-dropdown-button link py-3" + (isGroupActive.whatWeDo ? "
active" : "")}>
332                 What We Do
333             </div>
334             <div className="side-menu-dropdown flex flex-col border-l-2 border-primary-light">
335                 <NavLink to="/about" end className="link py-2 px-4">
336                     About IAJES
337                 </NavLink>
338                 <NavLink to="/organizational-structure" end className="link py-2 px-4">
339                     Organizational Structure
340                 </NavLink>
341                 <NavLink to="/newsletter" end className="link py-2 px-4">
342                     Newsletter
343                 </NavLink>
344             </div>
345         </div>
346
347         <NavLink to="/task-forces" end className="link py-3 pb-6 px-7">
348             Task Forces
349         </NavLink>
350
351         <div className="side-menu-dropdown-container py-3 px-7">

```

```

352     <div className={"side-menu-dropdown-button link pb-3" + (isGroupActive.resources ?
    " active" : "")}>
353         Resources
354     </div>
355     <div className="side-menu-dropdown flex flex-col border-l-2 border-primary-light">
356         <NavLink to="/video-resources" end className="link py-2 px-4">
357             Video Resources
358         </NavLink>
359         <NavLink to="/webinars" end className="link py-2 px-4">
360             Webinars
361         </NavLink>
362     </div>
363 </div>
364
365     <div className="side-menu-dropdown-container py-3 px-7">
366         <div className={"side-menu-dropdown-button link pb-3" + (isGroupActive.meetings ? "
    active" : "")}>
367             Meetings
368         </div>
369         <div className="side-menu-dropdown flex flex-col border-l-2 border-primary-light">
370             <NavLink to="/regional-meetings" end className="link py-2 px-4">
371                 Regional Meetings
372             </NavLink>
373             <NavLink to="/international-meetings" end className="link py-2 px-4">
374                 International Meetings
375             </NavLink>
376         </div>
377     </div>
378
379     <NavLink to="/search" end className="link py-3 px-7">
380         People
381     </NavLink>
382 </div>
383 </div>
384
385 </div>
386 </>
387 );
388 }

```

A.8 popup.jsx

```

1  import { useEffect } from 'react';
2
3  // POPUP -----
4  // PARAMETERS
5  // id - Unique ID for the popup. Ensures that popups from different components do not overlap
6  //       There should only be one Popup component used in a component.
7  //
8  // show - Determines if the popup is visible or not
9  //
10 // setShow - set function for show
11 //
12 // buttons - Optional. An array of objects with properties 'text' and 'onclick'
13 //       text - Required. A string with the button text
14 //       onclick - Required. A function that clicking the button will execute
15 //
16 // closePopup - Optional. A function that runs when the close button is clicked.
17 //       Note: This does not apply to closeOnBlur. If the user clicks outside of the popup, it will
18 //       just close normally
19 //

```

```

19 // stayOnBlur - Optional. Defaults to false. Determines if clicking outside the popup will close
   // the popup or not
20
21 export function Popup({ id, className, show, setShow, buttons, closePopup = null, stayOnBlur =
   false, nested = false, children }) {
22
23   const popupId = "popup-" + id;
24   useEffect(() => {
25     if (show) {
26       const popupContEl = document.getElementById(popupId);
27       const popupEl = popupContEl.children[0].children[0];
28       popupContEl.classList.remove("invisible");
29       popupContEl.classList.remove("opacity-0");
30       popupEl.classList.remove("mt-10");
31
32       // Compensate for when scrollbar is visible lmao
33       if (!nested) {
34         if (document.body.clientHeight > window.innerHeight) {
35           document.body.style.overflow = "hidden";
36           document.body.style.paddingRight = "15px";
37         }
38       }
39     } else {
40       const popupContEl = document.getElementById(popupId);
41       const popupEl = popupContEl.children[0].children[0];
42       popupContEl.classList.add("opacity-0");
43       popupContEl.classList.add("invisible");
44       popupEl.classList.add("mt-10");
45
46       // Return scrollbar after popup disappears
47       if (!nested) {
48         setTimeout(() => { document.body.style.overflow = "auto"; document.body.style.
paddingRight = "0"; }, 200);
49       }
50     }
51   }, [show])
52
53   function defaultClosePopup() {
54     setShow(false);
55     show = false;
56   }
57
58   if (closePopup == null) {
59     closePopup = defaultClosePopup;
60   }
61
62
63   const buttonsEl = [];
64   let i = 0;
65   if (buttons) {
66     buttons.map((button) => {
67       let btnId = "popup-" + id + "-button-" + i;
68       buttonsEl.push(<button key={btnId} id={btnId} className="button mx-2" onClick={button
.onClick}>{button.text}</button>);
69       i++;
70     })
71   }
72
73   return (
74     <div id={popupId} className="fixed top-0 left-0 size-full flex items-center justify-
center duration-200 z-999 invisible opacity-0">
75       <div className="z-1">
76         <div className="mt-10 min-w-lg max-w-[90vw] min-h-50 max-h-[85vh] p-4 bg-white
rounded-md shadow-md duration-200 flex flex-col justify-between " + className>
77         <div className="overflow-y-auto overflow-x-hidden w-full relative">

```



```

78         {children}
79       </div>
80       <div className="bottom-0 mt-4 flex justify-center shrink-0 grow-0">
81         {buttonsEl}
82         <button type="button" className="button button-light mx-2" onClick={
closePopup}>Close</button>
83       </div>
84     </div>
85   </div>
86
87   <div className="absolute size-full bg-black opacity-40 z-0" onClick={() => { if (!
stayOnBlur) { defaultClosePopup() } }} ></div>
88 </div>
89 )
90 }
91
92 // POPUPFORM -----
93 // PARAMETERS
94 // id - Unique ID for the popup. Ensures that popups from different components do not overlap
95 //   There should only be one Popup component used in a component.
96 //
97 // show - Determines if the popup is visible or not
98 //
99 // setShow - set function for show
100 //
101 // validate - Required. A function with paramters formData that runs when the form is submitted
102 //
103 // hasError - Optional. Default is false
104
105 export function PopupForm({ id, className, show, setShow, validate, hasError, nested = false,
children, encType }) {
106   // const [state, formAction] = useActionState(saveForm, {});
107
108   const popupId = "popup-" + id;
109   useEffect(() => {
110     if (show) {
111       const popupContEl = document.getElementById(popupId);
112       const popupEl = popupContEl.children[0].children[0];
113       popupContEl.classList.remove("invisible");
114       popupContEl.classList.remove("opacity-0");
115       popupEl.classList.remove("mt-10");
116
117       const formEl = document.getElementById(popupId + "-form");
118       formEl.reset();
119       formEl.children[0].scrollTo(0, 0);
120
121       // Compensate for when scrollbar is visible lmao
122       if (!nested) {
123         if (document.body.clientHeight > window.innerHeight) {
124           document.body.style.overflow = "hidden";
125           document.body.style.paddingRight = "15px";
126         }
127       }
128     } else {
129       const popupContEl = document.getElementById(popupId);
130       const popupEl = popupContEl.children[0].children[0];
131       popupContEl.classList.add("opacity-0");
132       popupContEl.classList.add("invisible");
133       popupEl.classList.add("mt-10");
134
135       // Return scrollbar after popup disappears
136       if (!nested) {
137         setTimeout(() => { document.body.style.overflow = "auto"; document.body.style.
paddingRight = "0"; }, 200);
138       }
139     }
140   });
141 }

```

```

139     }
140   }
141   }, [show])
142
143   const handleSubmit = (event) => {
144     event.preventDefault();
145     const formData = new FormData(event.target);
146     validate(formData)
147   }
148
149   function closePopup() {
150     setShow(false);
151     show = false;
152   }
153
154   return (
155     <div id={popupId} className="fixed top-0 left-0 size-full flex items-center justify-
156     center duration-200 z-999 invisible opacity-0">
157       <div className="z-1">
158         <form id={popupId + "-form"} encType={encType} onSubmit={handleSubmit} className
159         ={"mt-10 min-w-lg max-w-[90vw] min-h-50 max-h-[85vh] p-4 bg-white rounded-md shadow-md
160         duration-200 flex flex-col justify-between " + className}>
161           <div className="overflow-y-auto overflow-x-hidden w-full relative">
162             {children}
163           </div>
164           <div className="h-25 flex flex-col justify-end">
165             {hasError &&
166               <p className="text-error text-center my-2">An error occurred. Please
167               try again later.</p>
168             }
169             <div className="bottom-0 mt-4 flex justify-center shrink-0 grow-0">
170               <input id="popup-form-submit" type="submit" className="button mx-2"
171               value="Save" />
172               <button key="popup-close-default" type="button" className="button
173               button-light mx-2" onClick={closePopup}>Cancel</button>
174             </div>
175           </form>
176         </div>
177       </div>
178       <div className="absolute size-full bg-black opacity-40 z-0"></div>
179     </div>
180   )
181 }

```