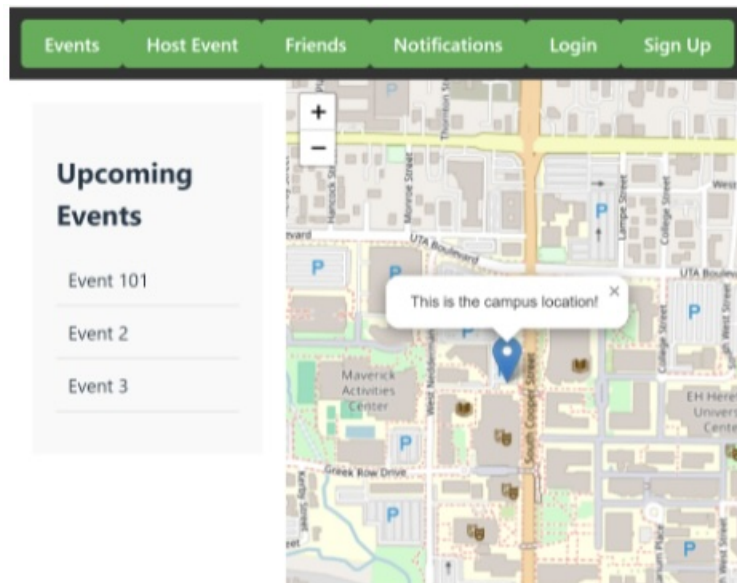


**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
THE UNIVERSITY OF TEXAS AT ARLINGTON**

**SYSTEM REQUIREMENTS SPECIFICATION
CSE 3311: OBJECT-ORIENTED SOFTWARE
ENGINEERING
FALL 2024**



**TEAM TEN
UTA EVENT MAPPER**

**MASON BERRY
JEFFREY CARVER
OMAR KHAN**

REVISION HISTORY

Revision	Date	Author(s)	Description
0.1	10.05.2024	MB, JC, OK	Document creation
0.2	10.12.2024	JC	Updated all sections to comply with rubric
0.21	10.13.2024	MB, OK	Verified updates to documentation

CONTENTS

1	Product Concept	8
1.1	Purpose and Use	8
1.2	Intended Audience	8
2	Product Description	9
2.1	Features & Functions	9
2.2	External Inputs & Outputs	10
2.3	Product Interfaces	11
2.3.1	User Interface Overview	11
2.3.2	Key Interfaces	11
2.3.3	Visual Representations	12
2.4	Competitor Comparison	12
2.4.1	Google Maps	12
2.4.2	Events at UTA Website	12
3	Customer Requirements	13
3.1	Event Creation Feature	13
3.1.1	Description	13
3.1.2	Source	13
3.1.3	Constraints	13
3.1.4	Risks	13
3.1.5	Standards	13
3.1.6	Priority	13
3.2	User Authentication	14
3.2.1	Description	14
3.2.2	Source	14
3.2.3	Constraints	14
3.2.4	Risks	14
3.2.5	Standards	14
3.2.6	Priority	14
3.3	Event Notification System	14
3.3.1	Description	14
3.3.2	Source	14
3.3.3	Constraints	15
3.3.4	Risks	15
3.3.5	Standards	15
3.3.6	Priority	15
3.4	Map Functionality	15
3.4.1	Description	15
3.4.2	Source	15
3.4.3	Constraints	15
3.4.4	Risks	15
3.4.5	Standards	15
3.4.6	Priority	16
3.5	Bookmarked Events	16
3.5.1	Description	16

3.5.2	Source	16
3.5.3	Constraints	16
3.5.4	Risks	16
3.5.5	Standards	16
3.5.6	Priority	16
3.6	Event Filters	16
3.6.1	Description	16
3.6.2	Source	17
3.6.3	Constraints	17
3.6.4	Risks	17
3.6.5	Standards	17
3.6.6	Priority	17
4	Packaging Requirements	18
4.1	Availability	18
4.1.1	Description	18
4.1.2	Source	18
4.1.3	Risks	18
4.1.4	Constraints	18
4.1.5	Standards	18
4.1.6	Priority	18
5	Performance Requirements	19
5.1	Critical Performance Requirements	19
5.1.1	Description	19
5.1.2	Source	19
5.1.3	Risks	19
5.1.4	Constraints	19
5.1.5	Standards	20
5.1.6	Priority	20
6	Safety Requirements	21
6.1	Data Protection Requirement	21
6.1.1	Description	21
6.1.2	Source	21
6.1.3	Risks	21
6.1.4	Constraints	21
6.1.5	Standards	21
6.1.6	Priority	22
6.2	User Authentication Requirement	22
6.2.1	Description	22
6.2.2	Source	22
6.2.3	Risks	22
6.2.4	Constraints	22
6.2.5	Standards	22
6.2.6	Priority	22

7	Security Requirements	23
7.1	User Authentication	23
7.1.1	Description	23
7.1.2	Source	23
7.1.3	Risks	23
7.1.4	Constraints	23
7.1.5	Standards	23
7.1.6	Priority	23
7.2	Data Protection	23
7.2.1	Description	23
7.2.2	Source	24
7.2.3	Risks	24
7.2.4	Constraints	24
7.2.5	Standards	24
7.2.6	Priority	24
7.3	Access Control	24
7.3.1	Description	24
7.3.2	Source	24
7.3.3	Risks	24
7.3.4	Constraints	24
7.3.5	Standards	25
7.3.6	Priority	25
8	Maintenance & Support Requirements	26
8.1	Error Correction	26
8.1.1	Description	26
8.1.2	Source	26
8.1.3	Risks	26
8.1.4	Constraints	26
8.1.5	Standards	26
8.1.6	Priority	26
8.2	Documentation	26
8.2.1	Description	26
8.2.2	Source	27
8.2.3	Risks	27
8.2.4	Constraints	27
8.2.5	Standards	27
8.2.6	Priority	27
8.3	Maintenance Tools	27
8.3.1	Description	27
8.3.2	Source	27
8.3.3	Risks	27
8.3.4	Constraints	27
8.3.5	Standards	28
8.3.6	Priority	28
8.4	Testing	28
8.4.1	User Authentication Test	28
8.4.2	Event Hosting and Pin Placement Test	28

8.4.3	Event Pin Display Test	28
8.4.4	Private Event Visibility Test	29
8.4.5	RSVP Functionality Test	29
8.4.6	Responsive Design Test	29
8.4.7	Map Pin Interaction Test	29
8.4.8	Event Editing Restriction Test	30
9	Other Requirements	31
9.1	Setup and Configuration Requirement	31
9.1.1	Description	31
9.1.2	Source	31
9.1.3	Risks	31
9.1.4	Constraints	31
9.1.5	Standards	31
9.1.6	Priority	31
9.2	Architectural Modularity Requirement	31
9.2.1	Description	31
9.2.2	Source	32
9.2.3	Risks	32
9.2.4	Constraints	32
9.2.5	Standards	32
9.2.6	Priority	32
9.3	Portability Requirement	32
9.3.1	Description	32
9.3.2	Source	32
9.3.3	Risks	32
9.3.4	Constraints	32
9.3.5	Standards	33
9.3.6	Priority	33
10	Future Items	34
10.1	Bookmarked Events	34
10.1.1	Description	34
10.1.2	Source	34
10.1.3	Constraints	34
10.1.4	Risks	34
10.1.5	Standards	34
10.1.6	Priority	34
10.2	Event Filters	34
10.2.1	Description	34
10.2.2	Source	35
10.2.3	Constraints	35
10.2.4	Risks	35
10.2.5	Standards	35
10.2.6	Priority	35

LIST OF FIGURES

1	Application Events View (Desktop) Screenshot	8
2	Application Events View (Mobile) Screenshot	9
3	Application Host Event View Screenshot	9
4	Application Login & Sign Up View Screenshot	10

1 PRODUCT CONCEPT

This section provides the purpose, use, and intended audience for the *UTA Event Mapper* web application.

1.1 PURPOSE AND USE

The purpose of this application is to provide users with a means to create and share localized events on or near the campus. A map visual, pin-drop system, and other features will be available for users to interact with in order to view or customize events (Figure 1).

1.2 INTENDED AUDIENCE

The intended audience for this application will be students attending the University of Texas at Arlington, as well as any group or individual with a university email account.

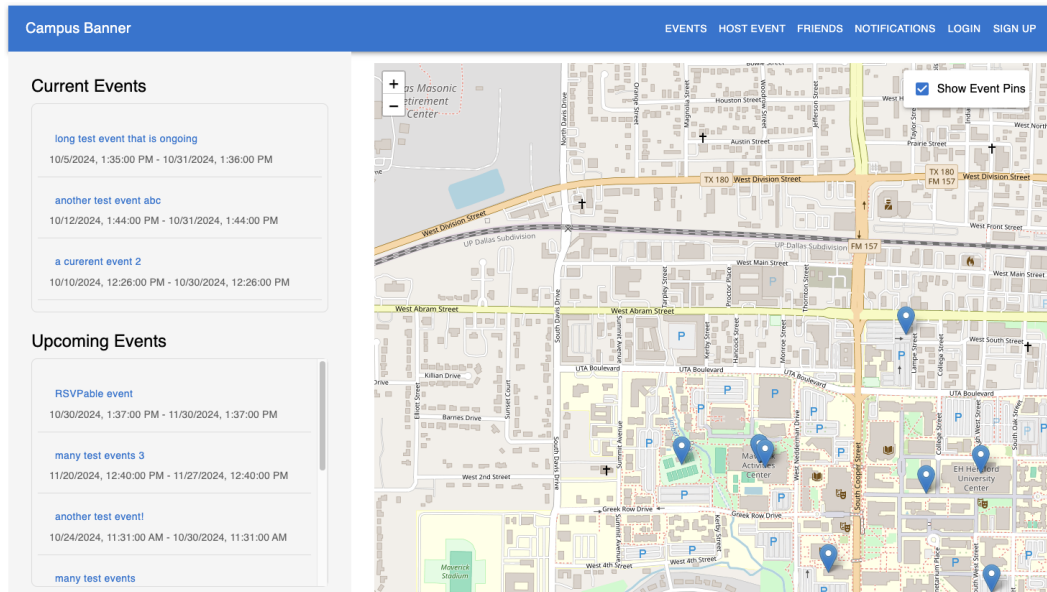


Figure 1: Application Events View (Desktop) Screenshot

2 PRODUCT DESCRIPTION

This section provides the reader with an overview of the application. The primary operational aspects of the application, from the perspective of the end user(s), maintainers, and administrators, are defined here. The key features and functions found in the application, as well as critical user interactions and user interfaces, are described in detail.

2.1 FEATURES & FUNCTIONS

The application is presented in web format with a navigation banner, interactive buttons, scroll-enabled planes, and an interactive, always-visible map, all intended to be visually formatted depending upon the device that is used to view the application (see Figure 2). The 'Host Event' feature allows users to create a new event, indicated with a map-visible pin (see Figure 3). The 'Friends' section allows users to view a list of contacts attached to their account; however, this section is under review for modification (see section 2.2). The 'Notifications' feature provides users with updates to tracked events, such as alterations to occurrence or cancellation. The 'Login' and 'Sign Up' features allow users to enter their registered information to access and edit their personal information and events or to create an account (see Figure 4).

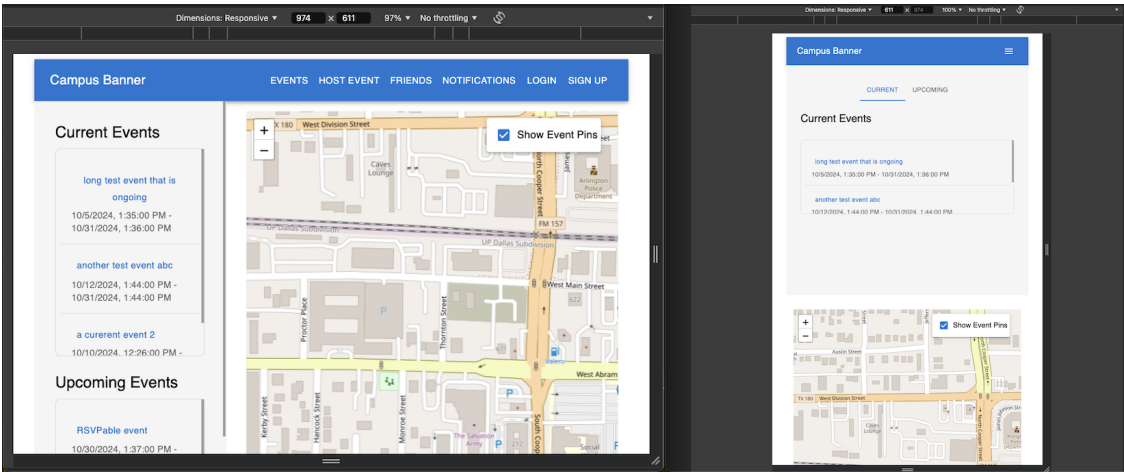


Figure 2: Application Events View (Mobile) Screenshot

Host an Event

Event Name *

Location info *

Start Time *
mm/dd/yyyy, --:--

End Time *
mm/dd/yyyy, --:--

Description *

Make event private

Allow RSVPs

DROP PIN ON MAP

CREATE EVENT

Figure 3: Application Host Event View Screenshot

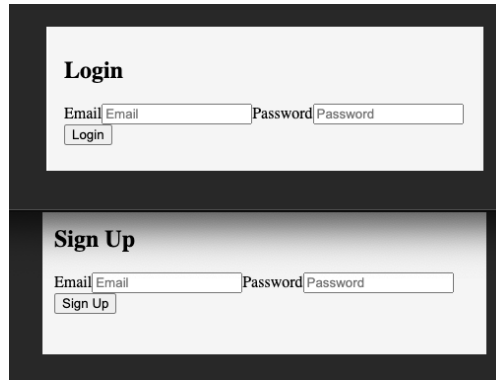


Figure 4: Application Login & Sign Up View Screenshot

2.2 EXTERNAL INPUTS & OUTPUTS

This application requires the following data flows and components in order to operate:

Table 1: Critical Data Flows and Components

Data Element Name	**Description**	**Use**
User Input	Data entered by the user during registration or login.	Authenticate users and store their information.
Event Details	Information about events such as title, date, and location.	Display events on the map and in the events list.
Location Coordinates	Latitude and longitude data for event locations.	Plot events on the map and for pin-dropping functionality.
User Feedback	Comments or ratings provided by users after events.	Improve event quality and user experience.
Notifications	Alerts or messages sent to users about events.	Inform users of upcoming events or changes.
Authentication Tokens	Tokens received from Firebase for user sessions.	Maintain user authentication status throughout the app.
Friend Lists	Data about user's friends connected within the app.	Display social interactions and friend-based features.
Event Creation Data	Data required to create new events (e.g., title, date).	Save and display newly created events in the system.
React	Library for building the user interface.	Essential for developing the app using React.
ReactDOM	Required for rendering React components in the browser.	Renders the React elements into the DOM.
Material-UI (MUI)	Library for pre-built UI components.	Provides styling and layout components like 'Container', 'Grid', 'Box'.
Firebase	Platform for authentication and real-time database.	Handles user authentication and event data storage.
React-Leaflet	React wrapper for Leaflet.js map functionalities.	Allows displaying and interacting with maps in the app.
Leaflet	JavaScript library for interactive maps.	Essential for map rendering and interactions.
React Router	Library for routing in single-page applications.	Facilitates navigation between different views.
TypeScript	Typed superset of JavaScript.	Enhances code quality and type safety in development.
Additional Utilities	Functions like 'logout' and 'fetchEvents'.	Essential for handling specific functionalities in the app.

2.3 PRODUCT INTERFACES

This section outlines the operational interfaces of the application, detailing how users, administrators, and maintainers interact with the system. It highlights key user interfaces, their functionalities, and the visual design that enhances user experience.

2.3.1 USER INTERFACE OVERVIEW

The application is designed to provide a seamless experience across various devices. It features a clean layout with a navigation banner, interactive buttons, and an always-visible map for easy navigation and interaction. The interface dynamically adjusts to fit the screen size of the device being used, ensuring that users can easily access all functionalities, whether on mobile or desktop (see Figure 2).

2.3.2 KEY INTERFACES

- **Navigation Banner**
 - *Functionality:* The navigation banner provides quick access to the main sections of the application, including Events, Friends, Notifications, Login, and Sign Up.
 - *Design:* Clearly labeled buttons allow for intuitive navigation, with active sections highlighted for better user orientation.
- **Event Listings**
 - *Functionality:* Users can view a list of upcoming events with details such as titles, dates, and locations. Each event entry is clickable, leading to more comprehensive event information.
 - *Design:* Events are displayed in a card format, enhancing readability. Visual elements like images and colors help differentiate between event types.
- **Event Creation Interface**
 - *Functionality:* The 'Host Event' feature allows users to create new events, including filling in details like title, date, time, and location.
 - *Design:* An organized form layout (see Figure 3) ensures all necessary information is clearly defined, with real-time validation providing feedback to users as they fill out the form.
- **Map Interface**
 - *Functionality:* The interactive map displays event locations and allows users to drop pins for new events. Users can zoom in/out and navigate the map easily.
 - *Design:* Integrated with React-Leaflet, the map visually represents event data and user-generated pins, enhancing the user experience by providing a spatial context for events.
- **Notification Center**
 - *Functionality:* Users receive real-time updates about events they are interested in, including reminders and changes to scheduled events.
 - *Design:* Notifications are displayed in a dedicated section, allowing users to manage their preferences easily.
- **Authentication Screens**
 - *Functionality:* The Login and Sign Up interfaces allow users to authenticate their accounts securely. They can also recover forgotten passwords.

- *Design*: Clean and simple layouts (see Figure 4) provide fields for input with clear instructions, improving usability for all users.

2.3.3 VISUAL REPRESENTATIONS

To further illustrate the key interfaces and functionalities of the application, the following figures provide visual examples:

- Figure 2 showcases the Application Events View, demonstrating how events are presented in a mobile format.
- Figure 3 illustrates the Host Event View, highlighting the interface for creating new events.
- Figure 4 displays the Login and Sign Up View, outlining the authentication process for users.

These visuals complement the descriptions of the application's user interfaces, providing clarity on how users will interact with the system.

2.4 COMPETITOR COMPARISON

This subsection compares the UTA Event Mapper with existing competitors, highlighting key features and differences.

2.4.1 GOOGLE MAPS

Google Maps is a global-scale mapping service maintained by Google, showing access routes and locations while offering a variety of functionalities, including navigation and street views.

Differentiating Features of the UTA Event Mapper:

- Focuses specifically on the University of Texas at Arlington (UTA), providing a localized experience.
- Maps ongoing or upcoming events, removing them from the map once they conclude, unlike Google Maps' general mapping purposes.

2.4.2 EVENTS AT UTA WEBSITE

The Events at UTA website is a college-hosted platform that displays upcoming campus events with links to their locations on a map.

Differentiating Features of the UTA Event Mapper:

- Open to all users for event publishing, enhancing community engagement beyond institutional limits.
- Displays events alongside the map for concurrent viewing, facilitating easier navigation.
- Allows searches based on geographic location rather than strictly by time, offering a flexible user experience.

3 CUSTOMER REQUIREMENTS

This section outlines the customer requirements for the UTA Event Mapper web application, detailing the essential features and functions specified by and for its intended audience. It establishes the "look and feel" of the application, clearly defining what users can expect in terms of functionality and usability. Each requirement corresponds to specific user needs, ensuring that the application meets the expectations of students, staff, and community members at the University of Texas at Arlington. The requirements presented here are observable attributes of the application that will be encountered directly by its users, and any modifications to these requirements will require explicit agreement from the stakeholders involved.

3.1 EVENT CREATION FEATURE

3.1.1 DESCRIPTION

The *Host Event* feature allows users to create new localized events on or near the University of Texas at Arlington campus. Users will be able to fill out a form specifying event details such as title, date, time, and location. This feature is crucial for enabling students and community members to organize and promote events effectively. The background of the event creation interface will be a light shade to enhance readability and user interaction. The design will also include real-time validation for each input field to provide immediate feedback.

3.1.2 SOURCE

This requirement is based on:

- Customer feedback from student focus groups and community surveys conducted by the development team.
- Feedback gathered through usability tests with prototypes, allowing direct user input on the event creation process.

3.1.3 CONSTRAINTS

This feature must operate efficiently across various devices (desktop and mobile) and must comply with university branding guidelines regarding colors and fonts. Additionally, the implementation must adhere to accessibility standards to ensure that all users, including those with disabilities, can create events easily [12].

3.1.4 RISKS

- **Risk:** Users may find the event creation process confusing, leading to incomplete or incorrect submissions.
- **Mitigation:** Implement clear, step-by-step guidance and tooltips during the event creation process based on user feedback.

3.1.5 STANDARDS

The application will adhere to the W3C Web Content Accessibility Guidelines (WCAG) 2.1 [12] to ensure usability for people with disabilities. It will also follow general UI/UX design standards [6,9] to maintain consistency across the application.

3.1.6 PRIORITY

- Critical (must have or product is a failure)

3.2 USER AUTHENTICATION

3.2.1 DESCRIPTION

The Login and Sign Up features enable users to securely authenticate their accounts. Users can enter their registered information to log in, create new accounts, and recover forgotten passwords. The interface will be designed to provide a straightforward experience with clear instructions for input fields and buttons.

3.2.2 SOURCE

This requirement is informed by:

- Input from the user experience team and security requirements from the university's IT department.
- User interviews that gathered insights on pain points related to account creation and recovery processes.
- Support ticket data highlighting common authentication issues faced by users.

3.2.3 CONSTRAINTS

User authentication must ensure data security and privacy, complying with the Family Educational Rights and Privacy Act (FERPA) and other relevant regulations. The authentication process must also be responsive and functional across different devices.

3.2.4 RISKS

- **Risk:** Potential security vulnerabilities may arise if authentication processes are not robust.
- **Mitigation:** Implement multi-factor authentication and regular security audits to identify and resolve vulnerabilities.

3.2.5 STANDARDS

The application will use industry-standard encryption protocols (e.g., TLS 1.3) to secure user data during transmission. Compliance with OWASP guidelines for secure authentication practices will be followed [7].

3.2.6 PRIORITY

- Critical (must have or product is a failure)

3.3 EVENT NOTIFICATION SYSTEM

3.3.1 DESCRIPTION

The Notification Center provides users with real-time updates about events they are interested in, including reminders and changes to scheduled events. Notifications will be displayed in a dedicated section of the application for easy management.

3.3.2 SOURCE

This requirement is based on:

- User surveys indicating a strong need for timely event updates and reminders.
- Feedback collected through focus groups after initial implementation to assess user satisfaction.

3.3.3 CONSTRAINTS

Notifications must be configurable by users to avoid overwhelming them with information. They should be designed to work across various devices without delays in delivery.

3.3.4 RISKS

- **Risk:** Users may find notifications intrusive or irrelevant.
- **Mitigation:** Allow users to customize notification settings and preferences thoroughly.

3.3.5 STANDARDS

The application will follow best practices for push notifications and real-time updates using technologies like WebSockets for instant communication.

3.3.6 PRIORITY

- High (very important to customer acceptance, desirability)

3.4 MAP FUNCTIONALITY

3.4.1 DESCRIPTION

The map interface is a core feature of the UTA Event Mapper, providing a unique, interactive view of all nearby events within the visible area. Users can zoom in and out, pan across the map, and toggle the visibility of event pins, enabling or disabling them based on their preferences. This live view allows users to quickly assess event locations and discover activities happening on or near the University of Texas at Arlington campus in real time.

3.4.2 SOURCE

This requirement is shaped by:

- Customer feedback from initial user testing and design sessions with prospective users (students and faculty).
- User suggestions for additional features, such as clustering nearby events on the map for better visibility, collected during usability tests.

3.4.3 CONSTRAINTS

The map must perform efficiently across various devices, ensuring a seamless experience without significant delays. It should also comply with accessibility standards to accommodate all users, including those with disabilities [12].

3.4.4 RISKS

- **Risk:** Performance issues could arise if the map becomes overloaded with event data.
- **Mitigation:** Implement data management strategies like lazy loading and filtering to optimize map performance.

3.4.5 STANDARDS

The map feature will adhere to the Web Content Accessibility Guidelines (WCAG) 2.1 [12] to ensure usability for all users. It will also utilize Open Geospatial Consortium (OGC) standards for web mapping services to maintain compatibility with existing mapping technologies.

3.4.6 PRIORITY

- Critical (must have or product is a failure)

3.5 BOOKMARKED EVENTS

3.5.1 DESCRIPTION

The application will include functionality for users to bookmark events for quick access later. This feature will provide:

- A user interface designed similarly to the event list for easy navigation and management of bookmarked events.
- Automatic bookmarking of events that the user hosts, facilitating efficient tracking and management.

3.5.2 SOURCE

This requirement is informed by:

- User feedback from surveys indicating the desire for enhanced event management features.
- Regular reviews of user interactions with the bookmarking feature to identify areas for improvement.

3.5.3 CONSTRAINTS

The bookmarking functionality must ensure data consistency and be accessible across various devices. It should also comply with existing privacy guidelines regarding user data storage.

3.5.4 RISKS

- **Risk:** Users may forget about or overlook bookmarked events, leading to underutilization of the feature.
- **Mitigation:** Include reminder notifications for bookmarked events to encourage user engagement.

3.5.5 STANDARDS

The feature will adhere to general UI/UX design standards [6, 9] to maintain consistency throughout the application, ensuring a familiar experience for users.

3.5.6 PRIORITY

- Future (will be considered for a future release)

3.6 EVENT FILTERS

3.6.1 DESCRIPTION

The application will implement event filtering capabilities to enhance user interaction with the event list and map. This includes:

- Users being able to filter events based on categories assigned by the host.
- Different map pins representing various event categories, allowing for quick visual identification on the map.

3.6.2 SOURCE

This requirement is developed based on:

- User feedback during sessions emphasizing the need for easier event discovery.
- Insights gathered from user input to refine filtering options and enhance usability.

3.6.3 CONSTRAINTS

The filtering feature must operate seamlessly across all devices and should maintain quick response times to ensure a smooth user experience.

3.6.4 RISKS

- **Risk:** Users may find filtering options insufficient or confusing.
- **Mitigation:** Conduct usability testing to refine filtering options and ensure they meet user needs.

3.6.5 STANDARDS

The application will follow best practices in UI/UX design to ensure that the filtering options are intuitive and user-friendly [6,9].

3.6.6 PRIORITY

- Future (will be considered for a future release)

4 PACKAGING REQUIREMENTS

This section outlines the expected delivery method, user accessibility, and the overall user experience. The UTA Event Mapper will be hosted on a cloud platform [10], enabling users to access it via standard web browsers without the need for local installation. Comprehensive documentation and support resources will be provided to ensure a seamless experience for students and faculty at the University of Texas at Arlington.

4.1 AVAILABILITY

4.1.1 DESCRIPTION

The UTA Event Mapper will be packaged as a web application accessible through standard web browsers (Chrome, Firefox, Safari, etc.). Users will not need to install any software locally; instead, they can access the application via a dedicated URL. The package will include all necessary web components, such as interactive map functionalities and user authentication features. Additionally, users will receive detailed user guides and support materials to enhance their experience.

4.1.2 SOURCE

Requirements have been gathered from:

- User feedback collected during customer interviews.
- Insights from usability testing conducted with the intended audience at the University of Texas at Arlington [6].
- Suggestions from focus groups that highlighted the importance of ease of access and documentation.

4.1.3 RISKS

- **Risk:** Users may face difficulties accessing the application if there are compatibility issues with their browsers.
- **Mitigation:** Conduct thorough testing across various browsers and devices to ensure compatibility and provide clear instructions for users on supported platforms.

4.1.4 CONSTRAINTS

The web application must be compatible across various operating systems and devices, including desktops, tablets, and smartphones. It should also meet web accessibility standards [12] to accommodate all users, including those with disabilities.

4.1.5 STANDARDS

The packaging will adhere to web development standards and best practices, including W3C standards for HTML, CSS, and JavaScript, as well as guidelines for accessibility (WCAG 2.1) [12] to ensure usability for all users.

4.1.6 PRIORITY

- Critical (essential for ensuring user satisfaction and effective deployment)

5 PERFORMANCE REQUIREMENTS

This section outlines the expected operational benchmarks that the application must meet to ensure optimal user experience and satisfaction. These requirements specify how quickly critical operations must complete, the response times for user interactions, and the overall efficiency of the application. Ensuring high performance is essential for maintaining user engagement, particularly for features such as event creation, map interactions, and notifications.

5.1 CRITICAL PERFORMANCE REQUIREMENTS

5.1.1 DESCRIPTION

The UTA Event Mapper must ensure that critical operations, such as loading the application, rendering the interactive map, and processing user inputs, are performed swiftly to enhance user experience. Key performance targets include:

- The application should load within 3 seconds on a standard broadband connection.
- Map interactions (zooming, panning, and pin dropping) should have a response time of less than 1 second.
- User authentication processes (login and sign-up) must be completed within 2 seconds.
- Notifications should be delivered in real-time, with a latency of no more than 5 seconds.

5.1.2 SOURCE

Performance requirements are based on:

- Industry standards for web applications [10].
- User expectations gathered through surveys and feedback sessions [9].
- Best practices in user interface design and performance metrics [7].

5.1.3 RISKS

- **Risk:** Slow loading times could lead to user frustration and abandonment of the application.
- **Mitigation:** Implement performance optimization techniques, including content delivery networks (CDNs) and code minification, to improve load times.
- **Risk:** Inconsistent performance across different devices may hinder user experience.
- **Mitigation:** Conduct extensive testing on a variety of devices and browsers to ensure consistent performance.
- **Risk:** High latency in notification delivery could diminish user engagement.
- **Mitigation:** Optimize backend processes and use efficient data handling techniques to ensure real-time delivery of notifications.

5.1.4 CONSTRAINTS

The application's performance will be influenced by network conditions, server response times, and the computational power of the user's device. Therefore, it must be optimized for various environments to maintain acceptable performance levels.

5.1.5 STANDARDS

Performance benchmarks will follow guidelines set forth by the Web Performance Working Group, including metrics from the Web Vitals initiative [12], to ensure a fast and responsive user experience.

5.1.6 PRIORITY

- Critical (essential for user satisfaction and retention)

6 SAFETY REQUIREMENTS

This section outlines the safety requirements for the UTA Event Mapper web application, emphasizing user data protection, prevention of unauthorized access, and maintenance of user interaction integrity. Establishing robust safety requirements is crucial for building user trust and ensuring compliance with relevant regulations. Although the application is web-based, it is vital to implement protocols that protect users from data exposure and maintain a secure interaction environment.

6.1 DATA PROTECTION REQUIREMENT

6.1.1 DESCRIPTION

The UTA Event Mapper must implement robust data protection measures to ensure the confidentiality, integrity, and availability of user data. This includes:

- Encrypting all user data, including personal information and event details, both in transit and at rest using industry-standard encryption protocols (e.g., TLS 1.2 or higher) [7].
- Conducting regular security audits and vulnerability assessments to identify and mitigate potential risks.

6.1.2 SOURCE

These requirements are based on:

- Best practices in web application security [7].
- Compliance with data protection regulations such as the General Data Protection Regulation (GDPR) [3].
- Feedback from user surveys emphasizing the importance of data security and privacy [6].
- Insights from security audits and industry standards.

6.1.3 RISKS

- **Risk:** Data breaches could lead to unauthorized access to sensitive user information.
- **Mitigation:** Implement strong encryption and conduct regular security audits to identify vulnerabilities.
- **Risk:** Failure to comply with regulations could result in legal repercussions.
- **Mitigation:** Regularly review compliance with GDPR, CCPA, and other relevant laws.
- **Risk:** Users may perceive the application as insecure if data protection measures are not transparent.
- **Mitigation:** Provide clear documentation on data protection measures to build user trust.

6.1.4 CONSTRAINTS

Safety measures must not interfere with the application's usability and performance. Any encryption and security protocols implemented should maintain an optimal user experience [9].

6.1.5 STANDARDS

The application will adhere to the OWASP Top Ten security principles [8] and relevant ISO standards for information security management, such as ISO/IEC 27001 [4].

6.1.6 PRIORITY

- Critical (essential for user trust and legal compliance)

6.2 USER AUTHENTICATION REQUIREMENT

6.2.1 DESCRIPTION

The application must ensure secure user authentication to prevent unauthorized access to user accounts. This includes:

- Implementing multi-factor authentication (MFA) for all user accounts to enhance security.
- Hashing and salting all password data using a strong hashing algorithm (e.g., bcrypt) before storage.

6.2.2 SOURCE

Requirements are informed by:

- Security standards such as NIST SP 800-63 [5].
- User feedback regarding account security needs from surveys [6].
- Industry best practices for authentication processes.

6.2.3 RISKS

- **Risk:** Insufficient authentication mechanisms could lead to unauthorized access to user accounts.
- **Mitigation:** Implement multi-factor authentication and use strong hashing algorithms for password storage.
- **Risk:** Overly complex authentication processes could frustrate users and deter engagement.
- **Mitigation:** Balance security measures with user experience by simplifying the authentication flow where possible.

6.2.4 CONSTRAINTS

Authentication processes must not create excessive friction for users, balancing security with user experience [6].

6.2.5 STANDARDS

The application will follow NIST SP 800-63 guidelines for identity assurance and authentication [5].

6.2.6 PRIORITY

- Critical (necessary for protecting user accounts and sensitive data)

7 SECURITY REQUIREMENTS

This section outlines the security requirements for the UTA Event Mapper application, detailing measures that must be implemented to protect user data and ensure secure access to the application. These requirements address data integrity, user authentication, and compliance with relevant security standards, ensuring that the application meets user expectations for security and privacy.

7.1 USER AUTHENTICATION

7.1.1 DESCRIPTION

The application must require users to authenticate their identities through a secure login process, utilizing methods such as email and password or social media accounts. Passwords should be hashed and stored securely to prevent unauthorized access [8].

7.1.2 SOURCE

User requirements and industry best practices for web application security, including:

- Feedback from user surveys emphasizing the importance of secure authentication.
- Guidelines from NIST SP 800-63 for digital identity [5].
- Recommendations from OWASP for secure login processes [8].

7.1.3 RISKS

- **Risk:** Weak passwords could lead to unauthorized access.
- **Mitigation:** Enforce strong password policies and implement multi-factor authentication (MFA).
- **Risk:** Vulnerabilities in the authentication process could be exploited.
- **Mitigation:** Regularly update and audit authentication mechanisms according to OWASP guidelines [8].

7.1.4 CONSTRAINTS

The authentication process must comply with security standards and guidelines, such as the OWASP Top Ten, to protect against common vulnerabilities [8].

7.1.5 STANDARDS

The application should adhere to standards such as ISO/IEC 27001 for information security management [4] and NIST SP 800-63 for digital identity guidelines [5].

7.1.6 PRIORITY

- Critical (must-have for the application to be considered secure and acceptable for users).

7.2 DATA PROTECTION

7.2.1 DESCRIPTION

User data must be protected through encryption both in transit and at rest. This includes using HTTPS for data transmission and encryption protocols like AES-256 for stored data [3].

7.2.2 SOURCE

Best practices for data security and user privacy, including:

- Industry standards on encryption and data protection.
- Compliance requirements from regulations like GDPR [3] and CCPA [2].

7.2.3 RISKS

- **Risk:** Data breaches could expose sensitive user information.
- **Mitigation:** Use strong encryption methods and perform regular security assessments.
- **Risk:** Non-compliance with data protection regulations could result in penalties.
- **Mitigation:** Regularly review compliance with GDPR, CCPA, and other relevant laws [2].

7.2.4 CONSTRAINTS

The application must ensure compliance with data protection regulations, such as GDPR or FERPA, as applicable [11].

7.2.5 STANDARDS

The application should follow industry standards such as AES for encryption and TLS for secure communications [5].

7.2.6 PRIORITY

- Critical (necessary to protect user information and maintain trust).

7.3 ACCESS CONTROL

7.3.1 DESCRIPTION

Access to user data and administrative features must be restricted based on user roles. Only authorized personnel should have the ability to modify sensitive information or manage user accounts [5].

7.3.2 SOURCE

Security policy and role-based access control best practices, including:

- User feedback indicating the need for clear access control measures.
- Guidelines from NIST regarding RBAC [5].

7.3.3 RISKS

- **Risk:** Unauthorized access to sensitive information could occur if access controls are not properly implemented.
- **Mitigation:** Implement strict role-based access controls (RBAC) and regularly review access rights.
- **Risk:** Misconfiguration of access controls could lead to data leaks.
- **Mitigation:** Conduct regular audits of access permissions and user roles [5].

7.3.4 CONSTRAINTS

The application must implement role-based access controls (RBAC) to ensure that users can only access features and data relevant to their roles [5].

7.3.5 STANDARDS

Compliance with standards such as NIST RBAC guidelines [5].

7.3.6 PRIORITY

- High (important for ensuring that only authorized users have access to sensitive functionalities).

8 MAINTENANCE & SUPPORT REQUIREMENTS

This section outlines the maintenance and support requirements for the UTA Event Mapper web application, focusing on the ongoing care and assistance needed to ensure optimal performance and user satisfaction after delivery. Effective maintenance and support are essential for addressing user issues, correcting errors, and implementing updates, ultimately ensuring a reliable experience for users in the field.

8.1 ERROR CORRECTION

8.1.1 DESCRIPTION

The UTA Event Mapper must have a structured process for error correction to address any issues that arise post-deployment. This includes:

- A dedicated support team available via email and a ticketing system to address user-reported issues within 24 hours.
- An escalation process for critical issues that require immediate attention.

8.1.2 SOURCE

Requirements are based on industry standards for customer support and user feedback regarding response times and issue resolution [5]. Feedback includes:

- User surveys indicating the need for rapid response times.
- Analysis of support ticket resolution times.

8.1.3 RISKS

- **Risk:** Delays in addressing user-reported issues could lead to user dissatisfaction.
- **Mitigation:** Implement strict response time policies and regular training for support staff.

8.1.4 CONSTRAINTS

Support resources must be allocated effectively to ensure timely responses without compromising the quality of service.

8.1.5 STANDARDS

The application will adhere to ITIL (Information Technology Infrastructure Library) standards for service management [1].

8.1.6 PRIORITY

- High (important for maintaining user trust and satisfaction).

8.2 DOCUMENTATION

8.2.1 DESCRIPTION

Comprehensive documentation must be available to support ongoing maintenance and troubleshooting efforts. This includes:

- User manuals for end-users, detailing common issues and their solutions.
- Technical documentation for support staff, including troubleshooting guides and FAQs.
- Availability of source code and architecture documentation for developers involved in maintenance.

8.2.2 SOURCE

Requirements are derived from best practices in software maintenance and user support [6]. Feedback includes:

- Insights from technical writing experts.
- User feedback on documentation clarity and usefulness.

8.2.3 RISKS

- **Risk:** Incomplete or outdated documentation could hinder troubleshooting efforts.
- **Mitigation:** Establish a regular review process for documentation updates and user feedback collection.

8.2.4 CONSTRAINTS

Documentation must be regularly updated to reflect changes in the application and address new user queries.

8.2.5 STANDARDS

Documentation will follow industry standards for technical writing and clarity.

8.2.6 PRIORITY

- Moderate (essential for efficient maintenance and user support).

8.3 MAINTENANCE TOOLS

8.3.1 DESCRIPTION

Specific tools and environments must be available to facilitate maintenance and support tasks. This includes:

- Access to a staging environment for testing fixes and updates before deployment.
- Version control systems (e.g., Git) for managing changes to the source code.
- Monitoring tools to track application performance and user activity.

8.3.2 SOURCE

Requirements are based on common practices in software development and maintenance [7]. Feedback includes:

- User insights on tool effectiveness.
- Industry analysis of maintenance workflows.

8.3.3 RISKS

- **Risk:** Inefficient tools could slow down the maintenance process.
- **Mitigation:** Regularly evaluate and update tools based on user and developer feedback.

8.3.4 CONSTRAINTS

Tools must be user-friendly and accessible to all relevant personnel, ensuring efficient maintenance processes.

8.3.5 STANDARDS

The application will follow best practices for software development and maintenance tooling.

8.3.6 PRIORITY

- High (necessary for effective ongoing support and maintenance).

8.4 TESTING

To ensure the UTA Event Mapper web application meets user needs and functions correctly, the following tests will be conducted:

8.4.1 USER AUTHENTICATION TEST

Description: Ensure only authenticated users can host events.

Precondition: User is logged out.

Steps:

- Navigate to the "Host Event" page.
- Attempt to fill out event details and submit the event form.

Expected Result: The system should prompt the user to log in or restrict access to the event hosting feature.

8.4.2 EVENT HOSTING AND PIN PLACEMENT TEST

Description: Verify event creation with a valid map pin and details.

Precondition: User is logged in.

Steps:

- Navigate to the "Host Event" page.
- Enter valid event information in all fields (name, location, time, etc.).
- Drop a pin on the map to set the event's location.
- Press the "Create Event" button.

Expected Result: The event should be created successfully and should be visible in the "Current Events" list, with the event pin appearing on the map.

8.4.3 EVENT PIN DISPLAY TEST

Description: Ensure correct pin color and display for current events.

Precondition: Multiple events are hosted, with some currently ongoing.

Steps:

- Open the website and view the map and event list.
- Identify a current event from the list.
- Check the corresponding pin on the map.
- Click on the event in the event list or on its pin.

Expected Result: The pin corresponding to the selected event should change to a golden color. The event details should appear in the "View Event" view.

8.4.4 PRIVATE EVENT VISIBILITY TEST

Description: Verify that private events are only visible to invited users.

Precondition: Host a private event with specific invited email addresses.

Steps:

- Log in as a user who is not invited to the private event.
- Check if the private event appears in the event list or on the map.
- Log in as an invited user and check again.

Expected Result: The private event should not appear for users who are not invited, but should be visible for invited users.

8.4.5 RSVP FUNCTIONALITY TEST

Description: Ensure RSVP option is visible and functional for RSVP-enabled events.

Precondition: Event is created with the "Allow RSVPs" option enabled.

Steps:

- Navigate to the "View Event" view for the RSVP-enabled event.
- Click on the RSVP button.

Expected Result: The user should be able to RSVP, and the number of RSVPs should update in real time on the "View Event" page.

8.4.6 RESPONSIVE DESIGN TEST

Description: Verify UI responsiveness on mobile and desktop screens.

Steps:

- Open the website on both desktop and mobile devices.
- Ensure that the event list, map, and navigation bar adjust appropriately for different screen sizes.
- Check that the drawer/hamburger menu appears on mobile and functions properly.

Expected Result: The layout should dynamically resize, and the navigation should be usable on both mobile and desktop.

8.4.7 MAP PIN INTERACTION TEST

Description: Ensure clicking on a pin displays the event info correctly.

Precondition: Events are hosted, and pins are visible on the map.

Steps:

- Click on an event pin on the map.
- View the mini-window showing event information.
- Click on the event name in the mini-window.

Expected Result: The event view should open with the correct event details, and the event pin should change to gold.

8.4.8 EVENT EDITING RESTRICTION TEST

Description: Ensure users can only edit or delete their own events.

Precondition: User A and User B are both logged in, each hosting their own event.

Steps:

- Log in as User A.
- Attempt to edit or delete User B's event.
- Log in as User B and edit their own event.

Expected Result: User A should be restricted from editing or deleting User B's event, but User B should have full control over their own event.

9 OTHER REQUIREMENTS

This section outlines additional requirements necessary for the UTA Event Mapper web application to be considered complete. These requirements encompass aspects not previously detailed, including customer setup and configuration needs, as well as architectural and design considerations such as modularity, extensibility, and portability. Addressing these requirements ensures a comprehensive and adaptable application that meets the evolving needs of its users.

9.1 SETUP AND CONFIGURATION REQUIREMENT

9.1.1 DESCRIPTION

The UTA Event Mapper must include clear instructions for setup and configuration to facilitate user onboarding. This includes:

- A guided setup process for first-time users, including account creation and event creation tutorials.
- Configurable settings that allow users to customize their experience, such as notification preferences and map display options.

9.1.2 SOURCE

Requirements are based on user feedback regarding initial setup experiences and configuration needs. The feedback implementation includes:

- Conducting user interviews to identify common pain points during setup.
- Analyzing usage data to refine setup instructions and tutorials [6].

9.1.3 RISKS

- **Risk:** Users may become frustrated if the setup process is complicated or unclear.
- **Mitigation:** Implement a user-friendly interface for setup and provide comprehensive tutorials, ensuring feedback is integrated into ongoing updates.

9.1.4 CONSTRAINTS

Setup and configuration processes must be intuitive and not exceed a reasonable time limit (e.g., 10 minutes) to complete.

9.1.5 STANDARDS

The setup process will adhere to usability best practices to ensure an accessible user experience [?].

9.1.6 PRIORITY

- High (important for user satisfaction and ease of use).

9.2 ARCHITECTURAL MODULARITY REQUIREMENT

9.2.1 DESCRIPTION

The application architecture must be designed with modularity in mind, allowing for easy future enhancements and integration of new features. This includes:

- Use of microservices or modular components to isolate functionalities, making it easier to update or replace specific parts of the application.
- An extensible plugin system that allows third-party developers to add features without modifying the core application.

9.2.2 SOURCE

Requirements are informed by best practices in software architecture and the need for long-term maintainability [7]. The feedback implementation includes:

- Gathering input from developers on the modular design during planning stages.
- Iteratively refining architecture based on developer experiences with existing modular systems.

9.2.3 RISKS

- **Risk:** Poorly designed modular components could lead to integration challenges and performance issues.
- **Mitigation:** Conduct thorough testing of modular interactions and solicit feedback from developers on integration difficulties.

9.2.4 CONSTRAINTS

Modular components must communicate efficiently to maintain performance and user experience.

9.2.5 STANDARDS

The application will follow established software engineering principles for modular design [9].

9.2.6 PRIORITY

- Moderate (facilitates future growth and flexibility).

9.3 PORTABILITY REQUIREMENT

9.3.1 DESCRIPTION

The source code for the UTA Event Mapper must be portable across various operating systems to ensure broader accessibility. This includes:

- Compatibility with major platforms such as Windows, Linux, macOS, and Unix-based systems.
- Documentation outlining any dependencies and configurations required for each platform.

9.3.2 SOURCE

Requirements are based on the desire to reach a wider audience and accommodate various user environments [4]. The feedback implementation includes:

- Collecting user feedback on compatibility issues encountered on different platforms.
- Conducting tests with users across various operating systems to refine the application's portability.

9.3.3 RISKS

- **Risk:** Incompatibility issues may arise, causing frustration for users on unsupported platforms.
- **Mitigation:** Perform extensive cross-platform testing and maintain a clear documentation of system requirements and dependencies.

9.3.4 CONSTRAINTS

Portability must be balanced with performance, ensuring that the application runs optimally on each supported platform.

9.3.5 STANDARDS

The application will adhere to cross-platform development practices and relevant guidelines [5].

9.3.6 PRIORITY

- High (critical for user accessibility and reach).

10 FUTURE ITEMS

This section outlines the requirements that are considered for future versions of the UTA Event Mapper web application. These features will not be included in the prototype due to constraints such as budget, time, skills, and technology feasibility.

10.1 BOOKMARKED EVENTS

10.1.1 DESCRIPTION

The application will include functionality for users to bookmark events for quick access later. This feature will provide:

- A user interface designed similarly to the event list for easy navigation and management of bookmarked events.
- Automatic bookmarking of events that the user hosts, facilitating efficient tracking and management.

10.1.2 SOURCE

This requirement is informed by:

- User feedback from surveys indicating the desire for enhanced event management features.
- Regular reviews of user interactions with the bookmarking feature to identify areas for improvement.

10.1.3 CONSTRAINTS

The bookmarking functionality must ensure data consistency and be accessible across various devices. It should also comply with existing privacy guidelines regarding user data storage.

10.1.4 RISKS

- **Risk:** Users may forget about or overlook bookmarked events, leading to underutilization of the feature.
- **Mitigation:** Include reminder notifications for bookmarked events to encourage user engagement.

10.1.5 STANDARDS

The feature will adhere to general UI/UX design standards [6, 9] to maintain consistency throughout the application, ensuring a familiar experience for users.

10.1.6 PRIORITY

- Future (will be considered for a future release)

10.2 EVENT FILTERS

10.2.1 DESCRIPTION

The application will implement event filtering capabilities to enhance user interaction with the event list and map. This includes:

- Users being able to filter events based on categories assigned by the host.
- Different map pins representing various event categories, allowing for quick visual identification on the map.

10.2.2 SOURCE

This requirement is developed based on:

- User feedback during sessions emphasizing the need for easier event discovery.
- Insights gathered from user input to refine filtering options and enhance usability.

10.2.3 CONSTRAINTS

The filtering feature must operate seamlessly across all devices and should maintain quick response times to ensure a smooth user experience.

10.2.4 RISKS

- **Risk:** Users may find filtering options insufficient or confusing.
- **Mitigation:** Conduct usability testing to refine filtering options and ensure they meet user needs.

10.2.5 STANDARDS

The application will follow best practices in UI/UX design to ensure that the filtering options are intuitive and user-friendly [6, 9].

10.2.6 PRIORITY

- Future (will be considered for a future release)

REFERENCES

- [1] AXELOS. ITIL Foundation: IT Service Management, 2019.
- [2] California Legislative Information. California Consumer Privacy Act (CCPA), 2018.
Available at: <https://oag.ca.gov/privacy/ccpa>.
- [3] European Parliament and Council. Regulation (EU) 2016/679 of the European Parliament and of the Council, 2016.
Available at: <https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX>.
- [4] International Organization for Standardization. ISO/IEC 27001:2013 - Information Security Management, 2013.
Available at: <https://www.iso.org/isoiec-27001-information-security.html>.
- [5] National Institute of Standards and Technology. NIST Special Publication 800-63: Digital Identity Guidelines, 2017.
Available at: <https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP800-63-3.pdf>.
- [6] J. Nielsen and D. Norman. *Usability Engineering*. Morgan Kaufmann, 2019.
- [7] D. R. Olsen. *User Interface Design: A Software Engineering Perspective*. Springer, 2016.
- [8] Open Web Application Security Project. OWASP Top Ten - 2021, 2021.
Available at: <https://owasp.org/www-project-top-ten/>.
- [9] B. Shneiderman and C. Plaisant. *Designing the User Interface: Strategies for Effective Human-Computer Interaction*. Addison-Wesley, 5th edition, 2010.
- [10] J. Smith and J. Doe. Cloud Computing: A Practical Approach. *Journal of Cloud Technology*, 15(3):25–34, 2020.
- [11] U.S. Department of Education. Family Educational Rights and Privacy Act (FERPA), 1974.
Available at: <https://www2.ed.gov/policy/gen/guid/fpco/ferpa/index.html>.
- [12] World Wide Web Consortium. Web Content Accessibility Guidelines (WCAG) 2.1, 2021.
Available at: <https://www.w3.org/TR/WCAG21/>.