CS310 Team-23 Project

SUEvent

Final Report

Fall 2025

Sabanci University

Our Team Members:

Arda Ösün - 32319 - Quality Assurance Lead
Mehmet Ali Bağıbala - 33907 - Integration & Repository Lead
Ulaş Deniz - 34034 - Documentation & Submission Lead
Yağız Tüten - 33897 - Project Coordinator

App Overview

What is SUEvent?

SUEvent is a CS310 project designed to help Sabancı University students connect with others who share similar interests. Whether you'd like to watch a movie, play a sport, or engage in other fun activities, SUEvent makes it easy to find someone to join you.

Our app is a social event management app that helps its users find events they can attend or announce events that they host. The main objective we have had was to create a platform for sabanci university members to socialise with each other in an easier way.

Users of the app need to first register to the app with an email address that ends with "@sabanciuniv.edu". After our users register or log into an existing account they can access the app.

On the main page users are presented with upcoming events that were posted by other users, here they can view the details of a specific event they find interesting by clicking on one of the "view details" button of the said card for the event, this will route them to a more detailed event page where users can comment and/or view other comments. They can also bookmark events or click the share button to get a prepared text to share with their friends etc..

Every page has our main navigation bar, this bar has 5 main partitions. The first one is the home page button, with this users can return to the main page introduced above.

The second button leads to the event searching page, on this page users can search events by their category or name. The same event cards that display information as explained above are also displayed here.

The third button leads the user to the event creation page, where users can set details of their event such as name, date, time, description, images, category.

The fourth button on the navigation bar leads the user to a calendar page where users are displayed a calendar with events on it. They can further click on one of the days of the calendar to view the events of that day in a separate page. This page works similarly to the main page where we had event cards again.

The last button navigates the user to their own profile page, in the profile page a user can change their information like profile avatar, description, etc. Additionally right below their profile users can view their own hosted events and bookmarked events, a user can delete their own event at any time from this part of the application.

Of course requiring pages have buttons to go back to the previous page.

So our main features include:

- Account creation
- Changing password
- Profile customization
    - Profile image customization
    - Profile description customisation
    - Editing username
- Posting events
    - Event customisation
    - Adding images to events
- Deleting events
- Bookmarking events
- Ability to see bookmarked events
- Viewing all events
- Commenting on events and viewing comments
- Sharing events
- Dynamic event calendar page
- Viewing events in different ways such as category/date etc.

Firebase Usage

Firebase is used for authentication and data storage. Email/password authentication is handled through Firebase Auth, with session state tracked via auth state changes. Firestore stores user profiles (users collection), event data (events collection), and event comments (events/{eventId}/comments subcollection). User documents include profile fields (username, email, bio, profile_picture) and saved event IDs. Event documents include metadata and image data (stored as data

URLs). Realtime streams are used to listen for events and comments so UI updates instantly.
State management is implemented with Provider and ChangeNotifier. AppAuthProvider manages authentication state and user profile data. EventProvider manages event lists and comment streams. Screens use Consumer/Consumer2 to reactively update the UI based on provider state.

Individual Contributions

Yağız Tüten: I led the early planning by helping define and distribute the first set of tasks, and I guided the wireframe/UI direction. I created the login page wireframes. I implemented the profile page and supported teammates during the first UI build-out on other screens. I also led the Firebase setup and project configuration, making structural updates in the GitHub repo. On the feature side, I handled media and UI consistency: profile image selection, storage, and display (including a default empty-photo asset), plus event image selection, character limits, database storage, and consistent rendering across the home list, calendar day list, and event detail view. I also created and integrated mock/placeholder photos (category-based placeholders and default user image) to ensure events and profiles always render cleanly.I fixed the calendar/day events page by sorting events by time and aligning its card image behavior with the rest of the app. I improved the comments section by showing user photos and adding a profile popup (photo, username, email, bio) on avatar tap. I also linked saved events to the profile's "Saved Events" tab, added title/location character limits during event creation, and worked on the change-password flow.

Mehmet Ali Bağıbala: I started with the creation and management of the project repository. Also worked on the ui design and wireframes of multiple items like the daily event page, the product cards that are used almost everywhere. I have worked on the navbar too. Also did some overall debugging throughout the app(please see the github for details). We have fully completed the firebase implementation that Yağız started, as Ulaş and I together. During the implementation we cleared the mock data and made sure things like login/register and event data were correctly working with firebase. We have also set up the firebase authentication and database. We have worked on the color palette/theme of the app which was later finalised by Ulaş into an eye friendly theme. Kept going on with smaller fixes and updates like font color, internal routings etc. I have also finalised the category system and made the app logo. Lastly I have made sure that some navigation buttons were added(or deleted/moved if they were in an unwanted place) to make sure we could navigate with ease. I have prepared the demo video for the final submission.

Ulaş Deniz: I focused on both the functional development and the visual refinement of the application. I laid the groundwork for the user experience by planning the general app navigation between the regularly used pages and implementing key UI elements, especially the reusable Event Cards, which are used on most of the pages. I started the development of our app by creating the primary interface pages, including the Home, Search, and Create Event pages, and finalized the general UI logic of the mentioned pages. I also finalized the color theme, which my friends had started working on, to ensure a visually appealing user interface. On the technical side, I handled the Firebase integration for the majority of these pages with my friend Mehmet Ali, ensuring dynamic data handling and updating the UI according to this dynamic data. I also conducted extensive code maintenance by fixing visual bugs, removing mock data, and standardizing the code syntax. Throughout the lifecycle of the project, I worked on the organization and submission of all project documentation, with the exception of the final submission document, which was completed by all the group members.

Arda Ösün:
Most of my time was spent on the development of the event page which went under many changes. I designed the view event page wireframe, its functionality, the comments section and its firebase implementation. Other than that page, I have had contributions to other pages such as the profile page, which I updated to have a change password function, and have done testing and bug fixing, such as our app was not opening the keyboard when trying to write into a text field in the startup page when opened from a high resolution device. In a similar fashion, I have conducted the unit tests and widget tests, and lastly in phase 3, I have recorded the demo video.

Lessons Learned

Throughout the project we realized how important early alignment is: clear task division, shared design decisions, and consistent UI patterns prevented rework later on. We learned to communicate small implementation changes quickly, because even minor UI or Firebase adjustments affected multiple screens. Using a shared state management approach (Provider) and shared assets taught us the value of keeping logic centralized so the app behaves consistently across pages. We also improved our teamwork skills by reviewing each other’s work, asking for feedback before large changes, and maintaining a steady rhythm of small, testable improvements.

Challenges Faced

One challenge was handling image upload without Firebase Storage. We resolved this by storing images as data URLs in Firestore and building consistent fallback logic with mock/placeholder assets, so the UI still looked complete. Another challenge was keeping event cards and detailed views visually consistent across multiple pages; we solved this by unifying image display rules and adding category-based placeholders. We also faced integration issues when multiple UI screens were implemented in parallel, which we addressed by aligning on shared patterns (Provider for state, common assets, and standardized input limits). Finally, plugin and dependency mismatches required cleanup; running dependency syncs and normalizing configuration files fixed those build issues.

DEMO Video Drive Folder Link:

CS310 Team-23 Final Demo

GitHub Repository Link:

https://github.com/DeliMaliBagibala/SUEvent