

Sprint 2 Report

Team: 13-SWEF25

Members: Alex Shaw, Eniola Akinola, Quinn Bankhead, Tomas Stoker

Course: CPTS 322: Software Engineering Principles I

Date: Thursday, October 30, 2025

An extension was verbally granted to Wednesday, November 5, 2025.

Repository: <https://github.com/alexander-shaw/seshy>

Kanban: <https://github.com/users/alexander-shaw/projects/3>

YouTube Video: <https://www.youtube.com/watch?v=4SMWwyvsiw>

Google Slides:

https://docs.google.com/presentation/d/1G5fcUSrCOH_X6aiAP8M2Y-CEaxbtnjC5dWtOpwnDuQq/edit?usp=sharing

Sprint Backlog

Our team worked collaboratively, with each member contributing in meaningful ways. Alex led iOS development—since he has the most experience with Swift and mobile architecture, and focused on implementing the app based on the higher-level data management framework that Eni and Tomas designed through system diagrams. Eni and Tomas provided the foundational data models and structural design that guided the project's overall organization. Quinn contributed to wireframing and UI/UX, helping shape the layout and visual flow of the app. Looking ahead to Sprint 3, Tomas, Quinn, and Eni plan to focus on developing the backend API using FastAPI (Python), Docker, GitHub Workflows, and either Google Cloud or Supabase to host and connect the system. Overall, the team's collective effort has been substantial and appreciated, and we are now positioned to integrate the backend with the app for a functional prototype.

Tomas: Individual contributions included working on UML diagrams, specifically the Class Diagram, Sequence Diagram, and the State Diagram. Additional communication with other team members on expectations of features and changes, both what they are and how they affect the project. Extensive discussions and testing of product as an outside perspective with additional live feature rating and usefulness. [20 hours altogether.]

Quinn: Worked on developing the Collaboration and sequence UML diagrams, as well as creating a traceability matrix that maps all our use cases and user stories to our functional requirements. Also helped determine how the app should look on the frontend. Participated in discussions about the app capabilities. [20 hours altogether.]

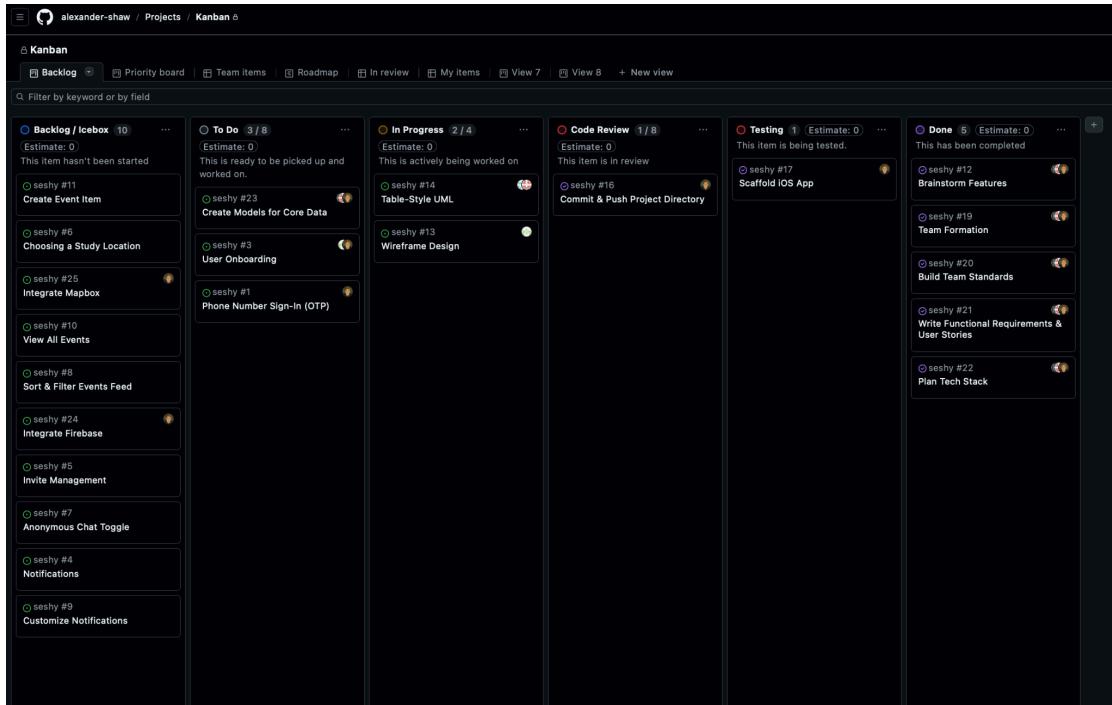
Eni: Worked on Use Case diagram and Sequence diagram using information from the traceable matrix and the User stories and scenarios. Contributed to the team discussion about some possible functions and organization of the app. [20 hours altogether.]

Alex: During this sprint, Alex led iOS development and implemented most of the app's core functionality and structure. Alex used AI-assisted tools like Cursor IDE and ChatGPT to

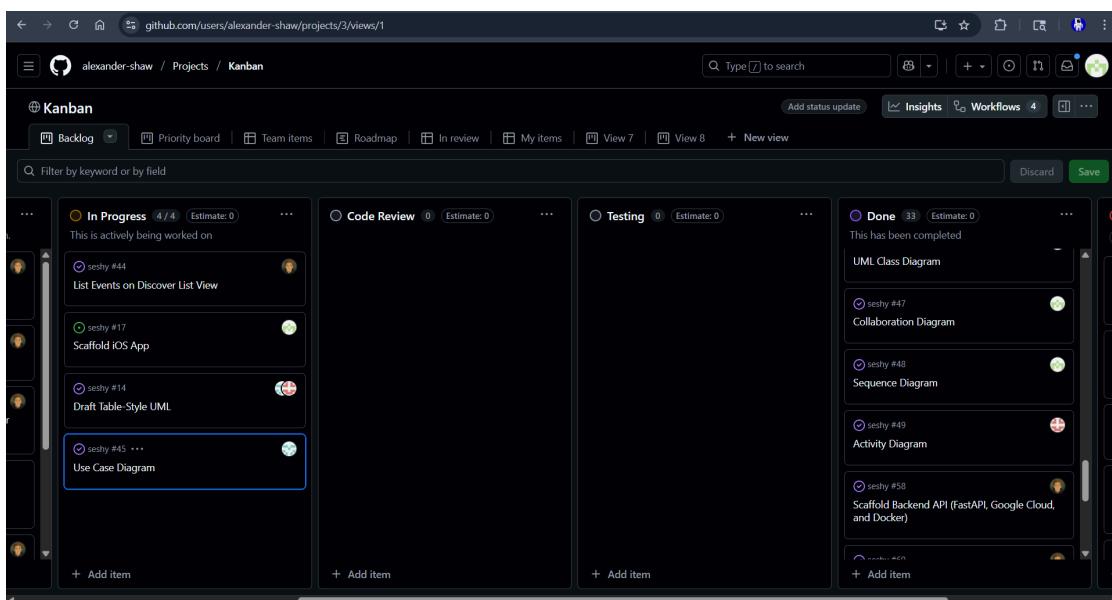
accelerate coding, debugging, and design refinement, which helped contribute to over 16,000 lines of Swift code. Alex scaffolded and implemented key flows and components including onboarding, event creation, map and list views, settings, and reusable UI elements while ensuring consistent MVVM architecture and high-cohesion-low-coupling design across the app. Looking ahead, Alex will help integrate the backend using FastAPI and Docker to create a fully connected prototype. [20 hours altogether.]

Kanban Board Screenshots

Start



Middle



End

This screenshot shows a GitHub Kanban board with the following columns:

- Backlog / Icebox**: 17 items. Sub-tasks include: Implement Permission Requests (Onboarding), Finish User Onboarding, Implement Core Motion Engine, Add & Implement Recurrence Rules Attribute to UserEvents, and Implement Edit Profile View & Logic.
- To Do**: 3 / 8 items. Sub-tasks include: Implement Map Preview Inside Event Full Sheet View, Implement Map Full Sheet View, and Implement Swiping Left/Right on Media Carousel View.
- In Progress**: 4 / 4 items. Sub-tasks include: Display Joined/Watching Events in Calendar View, Implement Activity Item in Notifications View, Implement Place Selection View, and Implement New Place View.
- Code Review**: 0 items.
- Testing**: 0 items.

This screenshot shows a GitHub Kanban board with the following columns:

- Code Review**: 0 items.
- Testing**: 0 items.
- Done**: 37 items. Sub-tasks include: Implement Settings View & Logic, Implement Bottom Bar Toggle Button for Discover Map & List, Display Map on Discover Map View, UML Class Diagram, Use Case Diagram, Persist Local Media After Logging Out & Signing Back In, Better Organize List of Colors in New Event View, Polish Settings View, Format Date/Time Better on Event Preview & Event Full Sheet View, and others.
- Build Improvements**: 6 items. Sub-tasks include: seshy #83, seshy #43, seshy #28, seshy #46, seshy #45, seshy #84, seshy #85, seshy #86, seshy #87, and seshy #88.

What's New

- Welcome View
- User Authentication
 - Phone Number
 - Verification Code
- User Onboarding
 - Name
 - Date of Birth
 - Gender
 - Profile Pictures
 - Tag Selection
- Events Listed (Preview + Full Sheet View)
- Events Displayed as Hexagons on 3D Map
- New Event View
 - Pictures
 - Name
 - Color
 - All Day
 - Start Date & Time
 - End Date & Time
 - Location (Random)
- Settings View
 - App Appearance + Map Style
 - Preferred Units
 - Log Out

Work Summary

During this sprint, Alex led development efforts and leveraged AI tools to accelerate implementation and debugging. Our team followed an agile approach—rapidly prototyping, testing, and refining features based on usability and design feedback. For example, we initially implemented a toggle between list and map views within the Discover tab, but after testing, we determined the map view did not effectively present events. We pivoted towards single-event previews emphasizing visual media to improve clarity and engagement. Through this process, we learned the importance of iterative design and focusing early on user experience over feature quantity.

Additional development efforts focused on evaluating which user data and features were truly necessary for our MVP. The team discussed whether to include real-time location tracking for event attendees but ultimately decided against it due to limited functional benefit, potential privacy concerns, and excessive implementation effort relative to user value. We also reassessed the personal information collected during onboarding—specifically birthday, age, and gender. Since in-app messaging was removed, these data points were no longer required for verification or legal purposes. Birthday collection remains in the current build and could be removed in a future update; gender was deemed optional and unnecessary for our core functionality.

Unfinished Work

From the start, we knew this product would be complex; other products have not succeeded in this market, and our small team is balancing multiple responsibilities. Unfinished work from this sprint includes: implementing a map view within the event full-sheet view (a pivot from our previous strategy), adding a calendar for watched and joined events, expanding event creation attributes, enabling notifications, and building profile and settings pages. The most significant remaining task is integrating with a cloud backend, which involves complex read/write operations and computationally intensive synchronization, most of which performs asynchronously in the background.

Workflow Adjustments

While we have effectively used GitHub issues to track progress, user stories have been less helpful, often slowing development with unnecessary and wasteful verbosity. The Columns for Code Review and Testing in the Kanban board were temporarily removed because they slowed progress without adding value. Testing is done directly on device or in the simulator after each implementation, ensuring functionality without redundant workflow steps. A column for Improvements would be more useful, but to meet assignment requirements, we will include these standard columns.

Completed Issues/User Stories

Here are links to some of the issues that we completed in this sprint:

- [#28: Display Map on Discover Map View](#)
- [#35: Implement Onboarding Steps for Display Name / Date of Birth / Gender](#)
- [#40: Display Events as Hexagons on Discover Map View; Circles Are Ugly & Not Scalable!](#)
- [#44: List Events on Discover List View](#)
- [#58: Scaffold Backend API \(FastAPI, Google Cloud, and Docker\)](#)
- [#69: Create Simple Event containing: Name, Color, Start & End, Location \(randomized\), and Status](#)

Here are links and brief explanations to some of the issues that we did not complete in this sprint:

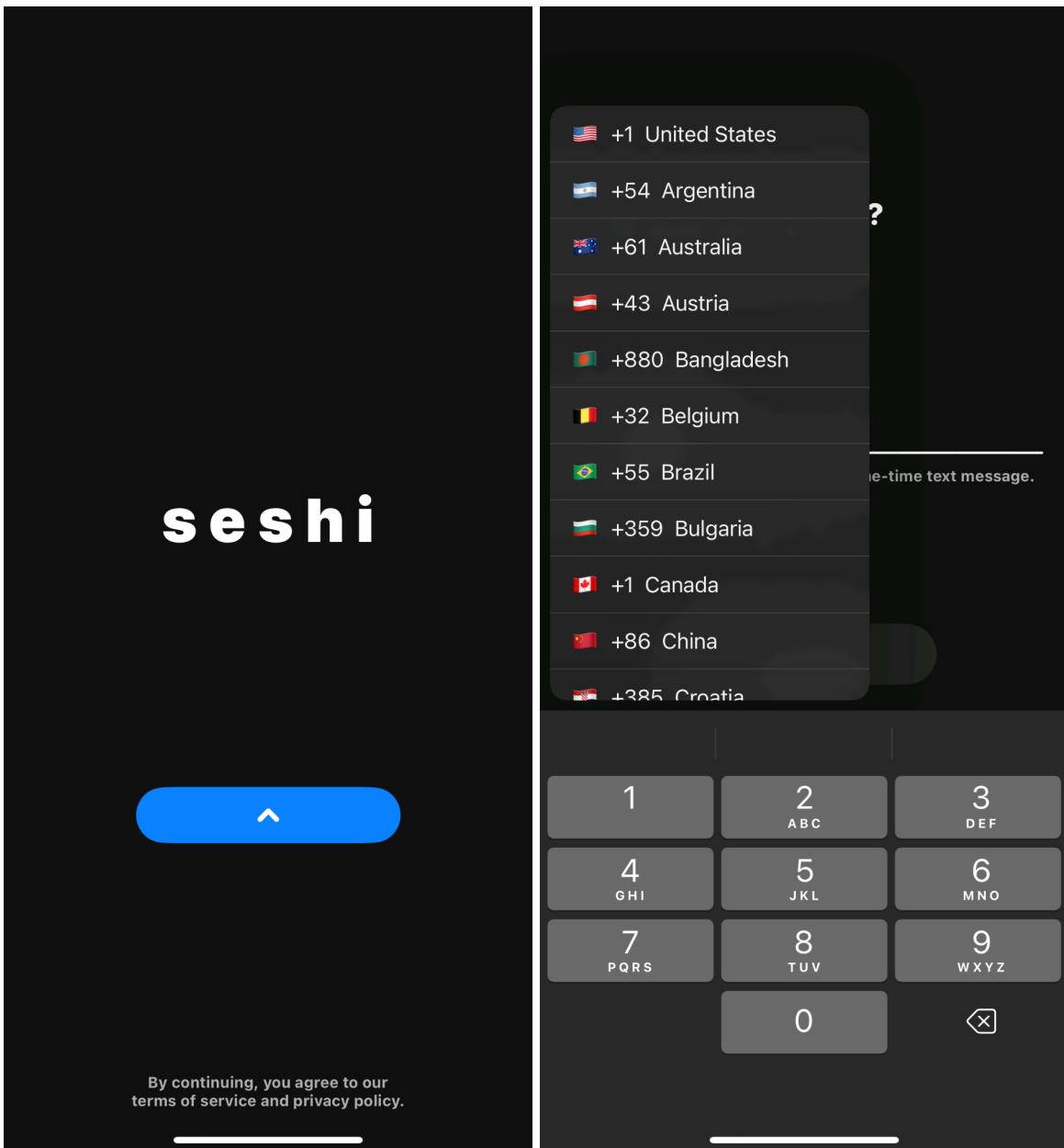
- [#72: Implement Place Selection View](#): Removed after functionality issues to prioritize other core features.
- [#73: Implement New Place View](#): Began development but was cut due to poor design and usability.
- [#75: Display Joined/Watching Events in Calendar View](#): Requires brainstorming wireframes and finalizing UI/UX design.
- [#76: Implement Map Preview Inside Event Full Sheet View](#): New direction pivots from earlier approach; next to be developed.
- [#77: Implement Map Full Sheet View](#): Completed but no longer the main interface; will later connect to map preview.
- [#78: Implement Activity Item in Notifications View](#): Awaiting UI/UX/UE wireframes and design planning before development.

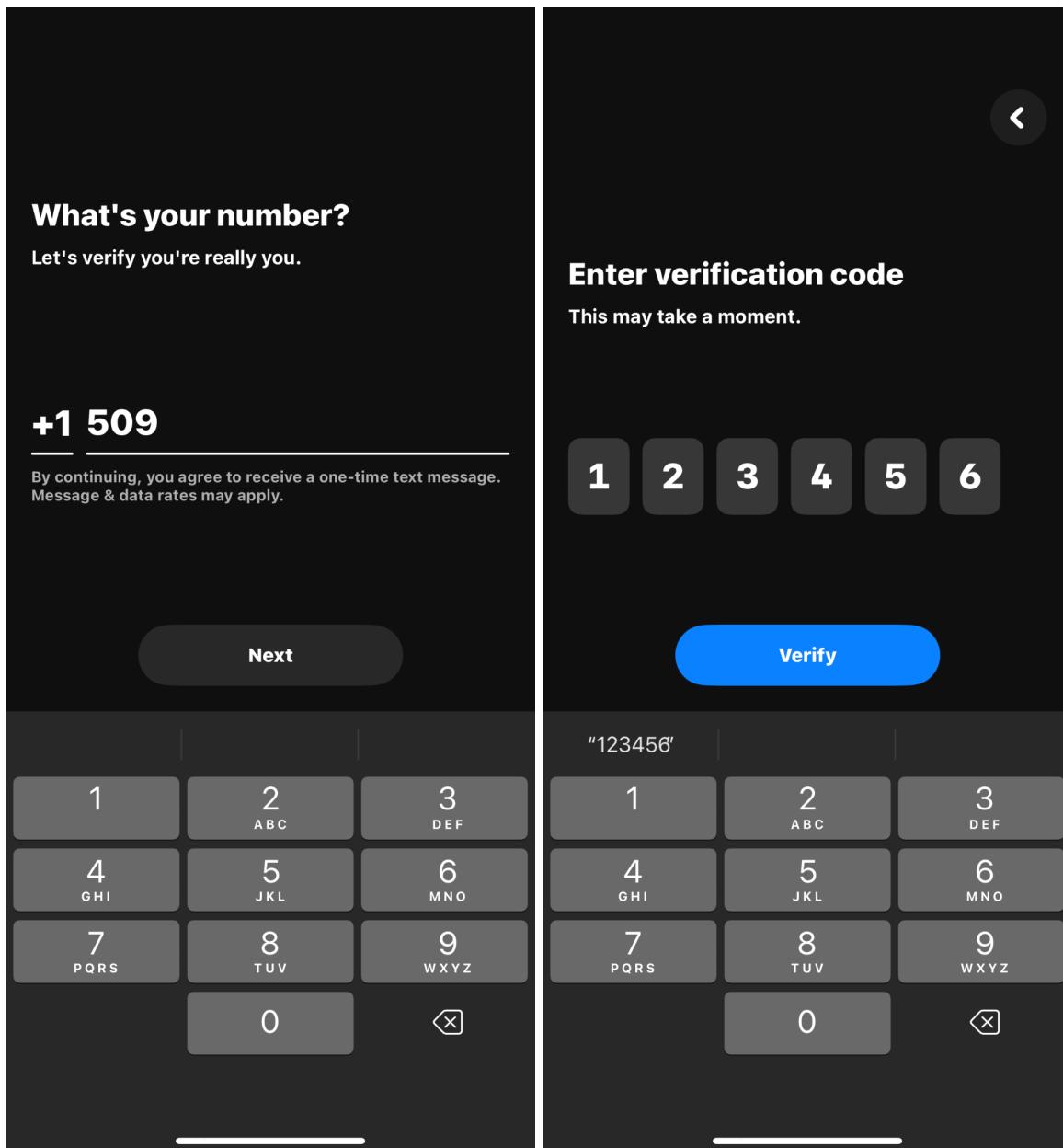
Code Files for Review

The following code files were actively developed during this sprint:

- [Functions.swift](#): A shared utilities file providing reusable helper functions for H3 geospatial logic, serving as a Swift wrapper around underlying C functions that perform hexagon indexing, coordinate conversion, and related operations used across the app.
- [UserEvent.swift](#): The data model defining a user-facing event (id, title, time, location, media, status) and its Codable/Identifiable conformance.
- [UserEventViewModel](#): An observable view model that manages UserEvent state, business logic, and async operations like loading, saving, and filtering.
- [DiscoverListView](#): A SwiftUI list view that surfaces discoverable events with sorting/filtering and navigation to each event's detail/full sheet.
- [NewEventView](#): A SwiftUI form for creating or editing events, handling input validation and submission to persistence/backends.
- [PrimaryButton.swift](#): A reusable, theme-styled button component with sizing, disabled/loading states, and consistent app-wide interactions.

Screenshots





What's your name?

Be yourself. Being real starts with being you.

Name

Next

Alex

q w e r t y u i o p
a s d f g h j k l
↶ z x c v b n m ↶
123 ☺ space done
🌐 🔍

Your b-day?

You cannot change this later.

0 1 0 1
2 0 0 6

YOUR AGE IS PUBLIC

Next

1 2 3
4 5 6
7 8 9
0 ↶

What's your gender?

You can always update this later.

Man

Woman

Nonbinary

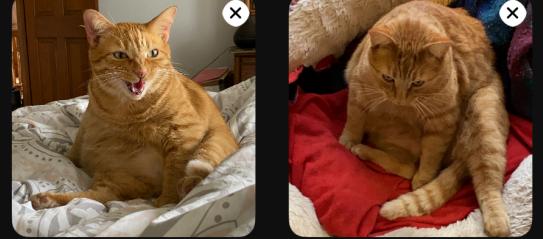
More

HIDDEN

Next

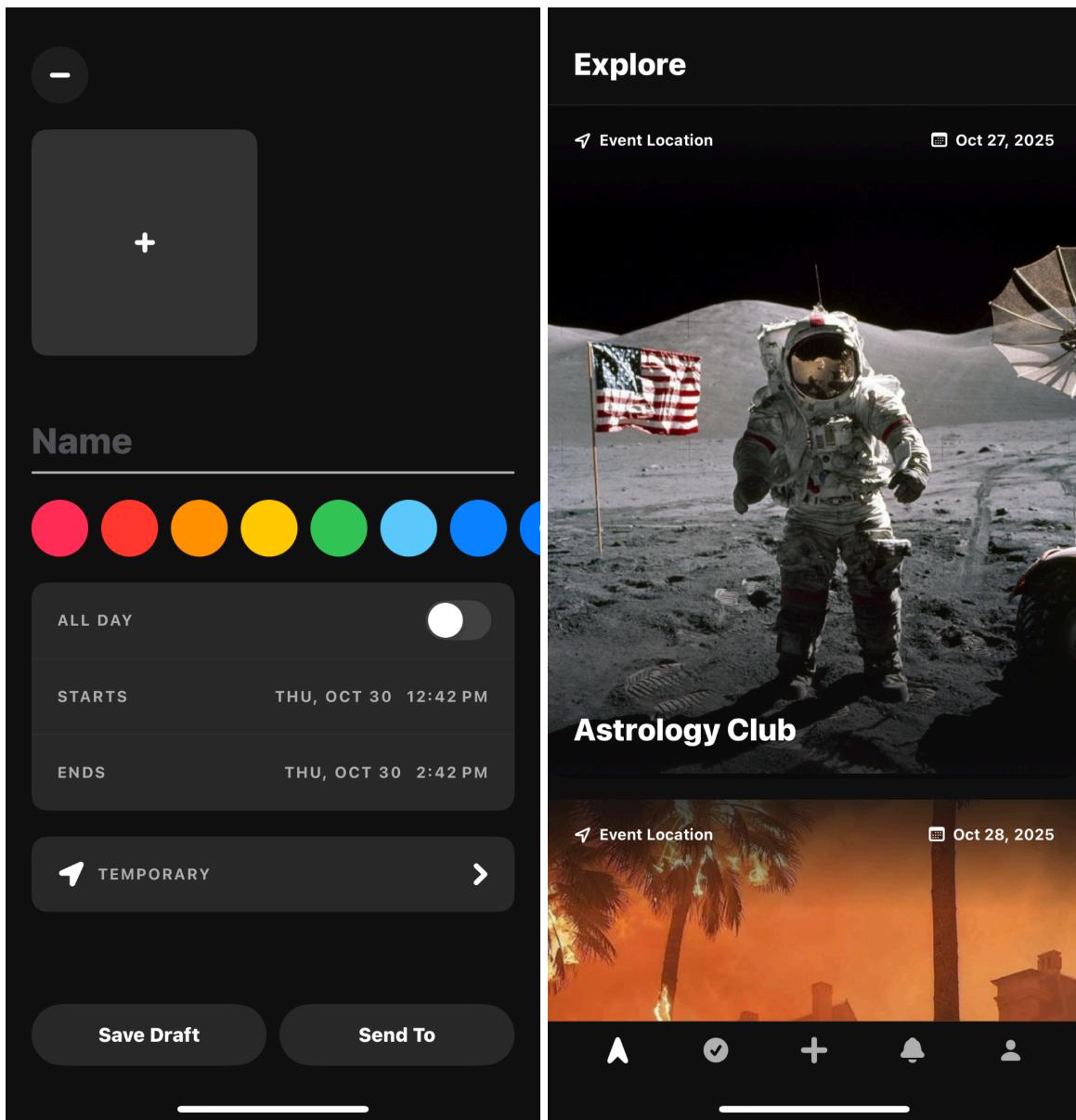
Pick your photos & videos

At least one required—tap to delete, drag to recorder.



+

Next



Retrospective Summary

What Went Well

- Significant development progress was made during this sprint, with multiple new features implemented and existing components refined.
- Team members who contributed to coding worked collaboratively and reached consensus on key implementation decisions, ensuring consistency across the codebase.
- The overall business strategy remains strong; our concept is sound, and the main challenge now lies in executing a product compelling enough for users to adopt and engage with actively.

What We Want to Improve

- Implementation discussions should be productive, contribute to team alignment, and should be valued as part of the development process.
- We aim to encourage more open communication, ensuring that all team members feel comfortable speaking up about how they can assist and contribute to the project.
- We also intend to minimize perfectionism in UI/UX design, focusing first on functionality and broader feature completion, with visual polish reserved for later refinement.

Changes for Next Sprint

- In the next sprint, we plan to expand coding responsibilities to other team members, whether through backend cloud development, testing builds on devices and submitting issues, or building a companion web or Android app.
- We will work to clarify requirements earlier in the sprint cycle to ensure that all deliverables are well-understood and deadlines are met ahead of schedule.
- Finally, we will promote greater flexibility and adaptability across the team to better respond to evolving priorities and challenges during development.