IDEA 380A3: Augmented Reality

Prototyping Erekani: A Location-Based Note Discovery and Creation Mobile AR Application

Jocelyn Villegas

1 Introduction

The following is a report on my interactive AR prototype design idea and how I approached the overall design for a mobile mixed-reality application concept that utilizes real life GPS locations for users to interact with one another across time and space. The prototype was developed with Unity and overall technical aspects were designed based on proposal feedback.

1.1 About

The concept of this AR protoype is a mobile application that connects users through strings of text or notes. The final product goal is that users can point their mobile device at real world locations, like a park bench, a café, or an alley and can "pin" short text messages in these locations that other future users can come across and view when visiting the same location.

This app's goal is to connect human experiences in a unique way utilizing spatial persistence, location-based services, and augmented reality. Future development would use real life GPS locations with the goal of a final product that can pinpoint finer-tuned user positions with location-based services for users to interact with one another across time and space.

2 Design Idea

Overall design began with developing user personas to decide the specific actions user's would need to take to meet their goals. Wireframes were then drafted from these needed actions. Figure 1 shows a finalized mobile application design made in figma based off of the wireframe drafts that demonstrates what the user flow could like. This finalized design also made the transition to the prototype much more efficient as I knew what buttons, configurations, scenes, and their corresponding scripts would be needed. For example, the "write a note" scene

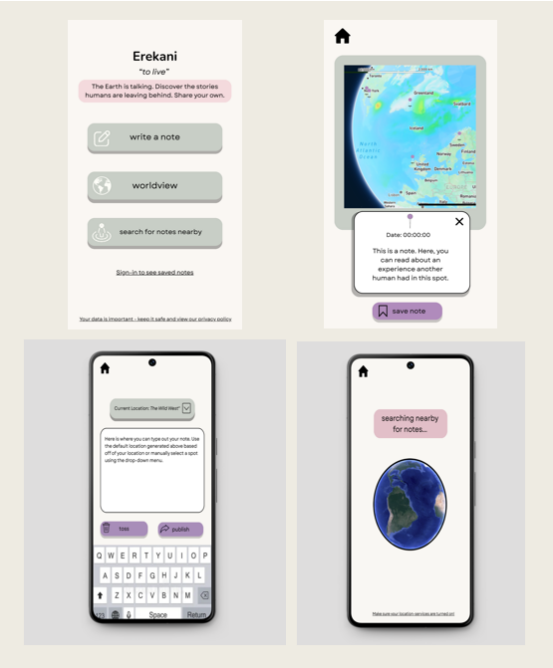

Figure 1: Mobile App Screen Designs

would need access to the note database, while the home page would not.

2.1 Technologies Used

The Unity engine along with its AR foundation resources such as the XR Interaction Toolkit was used for the "XR worldview" aspect of the application. For efficient data persistence, Unity's JSON utility class and systemIO/collections classes were utilized. Additional C# scripts were coded to facilitate the processes described below. Additonally, the source code is configured in GitHub using GitLFS to manage large unity files for version control.

3 Design Development

Prototype design consisted of scaling down and developing a simulation for how the above design

Figure 2: Pseudo Note Database: Local JSON File

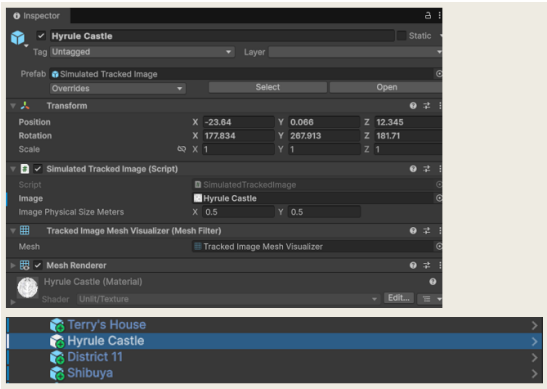

Figure 3: Pseudo Location Tags: Simulated Tracked Images

idea could be first approached on a local system. This was based on the initial proposal feed back that a prototype could focus on single object detection using markers and developing a design to "mock up" the viewing part of the app by saving the messages locally rather than trying to get them shared across devices.

3.1 Simulating a Note Database

A prototype design with the above considerations was achieved by focusing on "data persistence" rather than "spatial persistence". The app prototype uses the local machine it is running on to store a JSON file in its default program data folder that serves as a pseudo notes database that is accessed, retrieved and managed by a note manager script attached to a scene's event handler game object seen in figure **??**. Each visual note itself is a prefab that contains a text mesh pro child object with default text that can be modified through a method created in an attached script.

3.2 Simulating Location

For this implementation, the instantiation of a simulated tracked image object and its corresponding marker texture serve as a static pseudo GPS data or location database, in order to check for the location key used to access notes in the JSON file as seen in figure 3. In other words, simulated tracked image objects in the XR environment simulate geographic locations as they contain the metadata used to retrieve the corresponding note texts.

4 Design Conclusion

This resulted in a prototype where we can test many key components of the overall concept, such as user input stored to a database, observation of the visual of a note in an XR environment, note access and retrieval, and how the prefab changes based on some external property. The prototype unity files created are public and can be found on my GitHub [1].

References

[1] Jocelyn Villegas. Erekani-xr-prototype. `https://github.com/jocelynvillegas/Erekani-XR-Prototype`, 2025.