Final Report | Tavern

Semilore Abiodun-Adeniyi	Kevin Lam
Ahmed Ghazi	Eric Yang
Minh Cao	

Introduction

Tavern is a platform designed to streamline the tabletop role-playing game experience (aptly known as TTRPG). We were motivated to create *Tavern* because of a core issue every TTRPG player & community experiences: the lack of a specialized platform for finding, chatting, and engaging with other users. People currently use platforms such as Discord or Reddit to fill that niche, but those platforms were not built specifically for the TTRPG community, and their lack of TTRPG-specific functionality (such as matchmaking players with dungeon masters) is an inconvenience for the current-day community, including members of this team.

Thus, *Tavern* was born — a mobile social platform intended on connecting the TTRPG community as well as providing them with TTRPG-specific features such as player & campaign discovery, session scheduling, inventory management, and NFC-enabled transactional tools.

Approach

Our approach throughout the project was consistent with no major changes: create a social platform with all the features found on other social platforms in addition to TTRPG-specific ones. Our main focus in the scope of this project was the social aspect — account creation, home and account pages, matchmaking and scheduling, and a working chat system so that our users will have everything they need to track their TTRPG session and engage with the community as a whole.

We allocated around three weeks to each of our goals, with four goals total (account, matchmaking, scheduling, and chatting). We decided to use the first week to learn the tools we were using to create *Tavern*, and then quickly fell into a schedule of spending one week out of each two-and-a-half-week sprint designing the systems and then the remaining time implementing the system. We started on the designs for account creation before transitioning to implementing a prototype account system. Afterwards, our focus became creating intuitive home and account pages with personality to match the imagination of the TTRPG community. Then, after implementing a working prototype for those pages, we moved on to the matchmaking, scheduling, and chat system for our final few sprints. Throughout the project, we followed the scope tightly with only a few deviations in schedule to create necessary designs for other parts of *Tavern* (such as the community forum, navigation bar, and home page).

Results

Out of the four goals we set, we met two of them. Namely, we successfully implemented a login system + an account system, and we also have a working chat system. Although we were

not able to implement our other two goals — the scheduling and matchmaking system — we managed to create other important features, such as a working homepage with a carousel of upcoming events as well as working navigation bars to other parts of the app.

Customer Value

No changes.

Technology

Front End

We used the following for the front-end design:

- **Figma:** We used Figma to create the initial designs for each part of the software. Figma allows linking of different components that act as visual examples for the functionality of the actual software, and the collaborative tools within Figma makes it easy for us to design & communicate at the same time.

- **React Native:** We used React Native to implement the designs from Figma into the actual software. It also allowed us to develop a cross-platform for mobile devices, and we used Expo Go to simulate our designs.

- **Expo Go:** We used Expo Go to visualize the code onto the mobile application.

- **Paint 3D:** Eric drew rough sketches in Paint 3D! :D

Back End

We used the following for the back-end implementation:

- **Supabase:** We used Supabase to handle authentication for the login process. It also was used as the database to pull user information from.

- **Express:** We used the Express library for the chat system.

- **Heroku:** Used at the beginning to test python API's. We did not continue using Heroku throughout the project, and it was the only change in system architecture we made.

Testing

We did the following tests:

Cross-Platform: We asked friends to download Expo Go & had them scan a QR code to access our application. Our friends had different phones (namely Android and iOS iPhones) so it was easy to check whether or not our software worked cross-platform through Expo Go. For these tests, we measured success by whether the elements appeared correctly (none are missing or out-of-place) and that our mechanisms work properly.

Asset Sizes: We tested various screen widths on the devices of friends to ensure that our assets and design scaled accordingly to device sizes. We realized through these tests that the assets were set by pixel rather than a scalable variable and promptly changed it. For these tests, we measured success by how accurately the elements scaled across devices (so if we saw an insanely stretched image — it meant we messed up).

Mechanism Tests: Whenever we implemented new systems such as the carousel system on the home page and chat system, we tested again on previous devices to ensure both the design

elements and working components work as intended. For these tests, we measured success by whether they fulfilled the intended purpose (on mobile devices, they should slide with touch!)

Team

While a few members of the team occasionally helped outside their scope, the roles stayed static throughout the entire project scope. Three members of the team — namely Semilore, Kevin, and Ahmed — led the project while the other members contributed roughly equally.

Below is a table of the roles and responsibilities assigned to the team:

Semilore Abiodun-Adeniyi **Role:** Project Lead, Flex **Contributions:** - Created the initial codebase. - Programmed the home page, login page, and profile page. - Programmed various systems on main pages including homepage carousels, the login system, and the multi-chat system. - Designs for the chat page, profile page, and home page. - Fine-tuned various designs.	**Kevin Lam** **Role:** Backend Developer and Flex **Contributions:** - Streamlined asset sizes to stay consistent across different devices. - Fixed homepage carousel. - Creation of a chat system
Ahmed Ghazi **Role:** Frontend & Design **Contributions:** - Designs for initial onboarding pages, login page, and account creation pages. - Creation of a chat system - Fine-tuned designs	**Eric Yang** **Role:** Frontend & Design **Contributions:** - Polished the appearance of the home page & coded the profile picture. - Designs for initial onboarding pages, login page, account creation pages, forum page + forum post page.

	- Developed art assets to be used throughout the project. - Wrote the final paper.
Minh Cao **Role:** Database and Backend Developer **Contributions:** - Assisted with Testing & debugging - Testing Chat system across various ports and devices - Wrote final paper	

Project Management

We had four main goals for *Tavern* in the scope of this class:

- **Account** | ability to log in + a dedicated account page

- **Matchmaking** | ability to match with other users based on certain criteria

- **Scheduling** | ability to schedule game sessions within *Tavern*

- **Messaging** | ability to chat with other users in real-time

We drafted a rough outline of the time it would take to complete each task (around three weeks per task), and included both design and implementation within that outline.

As for our results, we were able to create a functional login system and functional account pages. However, we underestimated the time it would take to create those initial onboarding systems, and we realized too late that we did not have enough time to learn & complete the three remaining overarching goals. Our schedule was severely delayed because most of us were inexperienced with React Native and programming these specific systems, so we

had to take extra time each week to learn them as our initial learning week was not enough. That caused work from one sprint to bleed over to the next, and we realized near the halfway point that we would not finish all of our goals at the current pace. We decided that it would be best to focus on the chat system, as even with a basic chat system, we would have a prototype that fits the customers' needs of having a dedicated social platform for the TTRPG community.

Near the end of the semester, we were able to complete the chat system! The chat system took longer than expected, and we were two weeks behind schedule in accordance with the rough sprints created at the start of the class (due to knowledge gaps and the delay mentioned above). Rather than spending the remaining few days on a new system, we decided to polish up the designs and implementation in accordance to our final sprint.

Reflection

What went well?

Because of our common background as TTRPG players, the easiest part was planning the designs and functionalities that we wanted. We knew what was missing from the community as users within the community, and our design process went very smoothly from the color scheme to the main flow of the app to the individual pages. We had a lot of fun coming up with interesting ideas that balance the theme of our design (a cozy tavern — with wood & fantasy motifs) with modern-day designs seen in other social apps like Instagram and Reddit, and we're more-than-happy with the unique look and feel of all the final designs.

In terms of development, we're also really happy with how our designs almost translate 1-to-1 into the final app prototypes (with the exception of any incomplete elements)! For the most part, we didn't need to adjust any of the designs in any major way, and it scaled very well from Figma (on PC/laptops) to Expo Go on mobile devices!

What did not go well?

Team management was one of the core issues during the development of *Tavern*. We feel that we could have done better at communicating with each other about our expected schedule and our current progress at our tasks. Additionally, we didn't use anything apart from GitHub for organizational purposes (which isn't exactly the best for noting down important things like design expectations. One of us designed something but was unaware of newly discussed constraints so we had to change the design), so there were oftentimes miscommunication regarding certain tasks and certain pull requests. Most of our meetings were individually with one or two other people, and there were rarely any meetings where everyone was involved.

Additionally, the initial development was slow as we were trying to figure out the software we were using to create *Tavern*. We did not account for enough days for learning and familiarizing ourselves with the software, and that resulted in us falling short on half of our goals.

Finally, on presentation day, Expo Go unfortunately released a new update and it broke most of our code. The web version still worked, but it was definitely worrying those few minutes where we did not know how to fix all the bugs that had appeared because of the Expo Go updates.

Do you consider the final project a success?

Yes! We consider the final project a great success! Not only have we designed more than half the pages for the application, we have also implemented a few of them. Many of us also had personal goals during the project (such as gaining experience with app design and development, working towards legitimizing the project as a whole, and creating something resume-worthy), and we all feel that those personal goals have been completed! Apart from personal goals, all of us gained valuable experience working on *Tavern*; in the short term, it will help us know what to do and what not to do for the upcoming senior design project, and in the long term, the experience is invaluable practice for any team-related work in our future careers.

We also said in our project proposal that we'll measure our success on whether or not we meet our users' needs for a centralized method to meet other players and plan game sessions, and with the forum designed and the chat system implemented, we believe we've met our success!