Avalon: A Networked Multiplayer iOS Game

By: Avalon Studio

Kyle Bashour, Joseph Connor, Elliot Greenlee, Tyler Stuessi

Table of Contents

1. Introduction          3
   1. Overview
2. Customer Value          4
   1. Customer Need
   2. Proposed Solution
   3. Measures of Success
3. Technology          5
   1. System
   2. Tools
4. Team          6
   1. Skills
   2. Roles
5. Project Management          7
   1. Schedule
   2. Constraints
   3. Resources
6. Reflection          8

Introduction

**Overview**

Avalon is a word-of-mouth game usually played with a deck of cards. Our proposal is rebuild it as a networked multiplayer game for iOS. We can also describe the project by saying...

We were motivated to pursue this project because: group of friends plays board games -> this game usually is played with cards -> the organization for the game is confusing this way, and also puts pressure on one person to have to manage the game. The game can also be played with multiple rule sets, which could easily be managed by an app. We were also motivated by...

Our team definitely has the experience to build this; Kyle Bashour interned at mobile devshop Metova over the summer, and has experience building apps on iOS (the first platform we'll target). Tyler Stuessi has experience with networking and Python through is job at OIT. Joseph Connor has tons of experience working with backends and networking through an internship at Cisco. Elliot Greenlee learned mysterious things he cannot talk about during his time at L3. We can also describe our backgrounds by saying...

We plan to build the backend in Python, with a mobile application written in Swift for our first client. The backend will manage game play, and will use WebSockets or TCP to communicate with clients for real time updates. We can also orient the reader by saying...

Customer Value

**Customer Need**

The primary customers are board game enthusiasts, anyone over the age of...13? Who likes to play games like mafia with their friends. Deception type of games.

Customers want easy/quick set up, a fun game to play, access to instructions, no obligation to manage organization, etc...

We have done this because the game is confusing with a deck of cards. The organization for the game is confusing this way, and also puts pressure on one person to have to manage the game. The game can also be played with multiple rule sets, which could easily be managed by an app. Also, people look on the app store to find things to do. People are bored. The other reasons why we made this are… We are creating the experience of… We are solving the problem of...

**Proposed Solution**

Customer will have easy access to download. Will not have to have a deck of cards, or worry about managing the game. Just, play, nothing to keep track of. The customer will also benefit from our proposed solution by…

Easier management, faster set up, faster to teach the rules for new players, easier to add new rules. More modern because it is an app. Better because players could play over skype across the world. It also provides the new capability of… It is much better, easier, or faster because...

We have proposed the idea to our friends who play games, the people we originally learned the game from, and others who have played the original card based game, and they all see the benefit of turning it into an app. We have also tested the idea on...

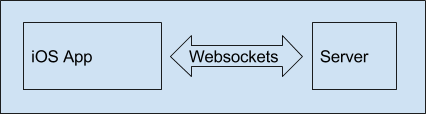
**Measures of Success**

‒ How will you know if customers got the benefits you want to provide? ‒ What are your customer-centric measures of success?

Technology

**System**

The main components of our system are a front end iOS app and a back end server. The front end iOS app is used for user interfacing, and logical gameplay organization. The back end server is used to manage networked game communication, and setting up the game. We can also describe the components by saying…



System architecture block diagram ^

A minimal system would allow the users to play the game. We can also describe this minimal system by saying…

Enhancements include the extra rule sets for the game, possibly a web client, possibly different color themes or content themes. Other possibile enhancements that the customers would value are...

playing it ourselves, putting it on the app store for testing, giving it to friends who want to play. We will also test our system by...

**Tools**

technology you use to build what you deliver

‒ What will you use to build the system?

Python, Swift, Websockets or TCP, Github, Travis CI

‒ Are there available tools you can leverage?

Team

**Skills**

‒ Has anyone on the team built something like this before?

‒ Are the tools known or new to the team?

Kyle Bashour- iOS experience, Swift. internship at mobile devshop Metova, previous iOS experience

Tyler Stuessi- Networking, Python. works for OIT.

Joseph Connor - Networking and backend. Internship at Cisco.

Elliot Greenlee- Backend management. Part time work at Alcoa Inc., and a co-op at L-3 Communications.

**Roles**

‒ What are the roles of the team members? ‒ Will the roles be fixed or rotating?

Project Management

**Schedule**

‒ Is completion of the system feasible?

‒ When and how often will you meet? Face to face?

Constant hipchat. We should pick a meeting time that is regular.

**Constraints**

‒ Are there any regulatory, or legal constraints?

Legal constraints regarding games: http://www.copyright.gov/fls/fl108.html

‒ Are there any ethical or social concerns?

We should give a rating for the game that is appropriate. Our game has deception, assassins, murder, …anything else? <https://developer.apple.com/app-store/review/guidelines/>

**Resources**

‒ Will you have access to the data you need?

Reflection

Subsections to be filled in later

‒ What Went Well?

‒ What Didn’t Go Well?

‒ Recommendations