Conservatory Catch Project Final Two-Page Summary

Group 09: Jose Cuellar-Aguirre, Marcos Echevarria, Christian Garcia, and Nicholas Tryba

For our project, we are designing an Augmented Reality (AR) mobile game that will promote interest in the Garfield Park Conservatory (GPC). More specifically, we will be working alongside the Garfield Park Conservatory Alliance (GPCA), which is the nonprofit that works with the city to provide programming and resources for the conservatory and its visitors. The game itself will function as an AR game where players will be able to walk around the conservatory alongside their in-game character through the use of GPS. As players walk around, random plant species will have the chance of appearing on screen, where players can play a minigame to add the species to their collection. Collections will be where most of the learning takes place, as players can view information and trivia on each of the species they have collected.

The purpose of our project is to promote interest in the conservatory, and to offer visitors an exciting new means of educational programming. We believe that the increased interest in the GPC generated by the app will lead to increased donations for the GPCA, and increased visitors for the conservatory. The GPCA is one of the primary clients for this project. They provide educational programming, events, and create resources for visitors.

The requirements and acceptance tests are split up into sections, pertaining to various problems and concerns we expect to find down the line. These sections are: Functional, Data, Performance, Dependability, Maintainability and Supportability, Security, Usability and Humanity, Look and Feel, Operational and Environmental, Cultural and Political, and finally, Legal.

Functional requirements specifically deal with the functionality the system must have, and for our project, we have three requirements. The three deal with smooth scene transitioning, player reward tracking, and the user's location accurately being updated. The first one focuses on making sure there's some sort of transition between scenes as to not be jarring to the users, which the acceptance test addresses by making sure there's no less than 0.5 seconds of transitioning time between the scenes, as well as making sure there's an animation of sorts that transitions from the previous scene to the new scene. The second one deals with players being awarded their goods as they complete minigames, something that is essential. The acceptance test checks to make sure the users get their rewards, even if events were to happen, such as the game being closed, connection being lost, or even the servers going down and crashing. Last, but not least, we have the user's position relative to their GPS position, which is needed to function if we are to ever tell if they're actually in the Garfield Park Conservatory or not, and to be able to engage with the AR function that our product boasts. The acceptance test checks to see if the user's position relative to their GPS position is within a certain threshold, akin to "close enough" so that they're at or at least near the spot that the game thinks they are.

Data requirements deals with things that has anything to do with data, and we have two requirements revolving around data. One deals with data in the database, specifically plant names, and another deals with usernames.

Dependability deals with reliability, availability, robustness, and safety-critical concerns. For reliability and availability, we want users to be able to see their collection of plants even when they're not connected to the servers, which the acceptance test checks for. For robustness, we don't want the server to be down for too long, so we have our acceptance test check a function that would notify us whenever they are down so we can get them back up and running ASAP. As for safety-critical concerns, we are concerned with the possibility of epileptic seizures, the acceptance test checking to see if there's any animations or transitions that are 3 seconds or longer, and if so, checking to see if any of them do not meet the safety standards of not possibly causing a seizure. Maintainability and Supportability points towards making the project able to be maintained and supported throughout its lifetime. We have requirements that deal with server maintenance, broken game maintenance, ability to report bugs, being able to be played on mobile devices, being able to receive patches and updates, and even the idea of this product being applied to other plants/animals. Usability and Humanity provides many accessibility options to those who need them or to those who like them: things such as color blindness mode, subtitles and closed captions, changing font size, brightness, contrast, language, volume for specific in-game audio, voice-enabled commands, and many more things.

The design goals of the project include seamless integration of new minigame types, fast and accurate storage of newly collected rewards into a Player's collection, and a reliable representation of players on Overworld map. More details on the proposed design system are included in the final report.

For the final system design, we have decided to that it would be best to implement a Client-Server architecture. The reason for this is that data will need to be exchanged and updated between users and the conservatory. Players also need up-to-date data on other users, further solidifying our choice.

Most of the issues with our project stem from latency and security concerns. These are further detailed in the final report.

We estimate the application to cost around $200,000. This accounts for the support and development of the game. We estimate a timetable of around 2-3 years, assuming anywhere from 10 to 15 people are working on the game.

As a whole, the development project went smoothly. The main concerns were with time management, which is something we would prioritize where we to do this project over. More details on our retrospective can be found in our final report.