

B. Technical Specifications and Requirements

Updated Project Description:

The current project as it is includes one big incremental game and within that game there are three mini-games. These mini-games include Sudoku, a space shooter game, and an RPG. The development and fixing of these games is the group's first task. The team has been assigned individual tasks to allow the team to better develop and fix the game as the project moves along. The team now has a better understanding of Unity and MongoDB. The team was given a supplementary task to develop parser code for the database so that we may pull user information from it and analyze it visually.

Each member of the team has a specific task. These tasks are assigned to each member according to their skillset. Kevin Tran's task includes making a list of incremental features on how to add or incorporate game features from the game Anti-Idle. His task also includes improving the incremental tutorial and designing a help button where users can either read the tutorial again or access information about what each feature does. Stefon Martin's task includes writing an outline of ideas for a complete daredevil game, with the goal of users experiencing high speed and vertigo during gameplay. This outline will include a timeline for feature development and game design. Brendan Lauck's task will include coming up with a game design for additional puzzle games, including Kakuro and another small minigame of his own design. Nigel Haney's tasks include listing out any bugs within the current game and fixing 2-3 bugs per week. If there aren't any currently known bugs, he will assist others where and when needed. Nigel will also be responsible for listing a batch of important changes to work on over the next month. Lastly, Ash Mahein's tasks include balancing game features. This means testing incremental progression, active/passive coin progress within and outside of the mini games, upgrade cost prices, progress bar speed, level-up speed, offline/online progress, and experience gained progress.

The team has also been given the task to develop two new mini-games by the end of the project. One of these mini-games will be a complete and feature rich Daredevil game. This game will appeal to thrill-seeking students by incorporating free-falling and allowing the user to dodge objects and platforms. The second game will be a puzzle game similar to Sudoku. This game will appeal to users that enjoy critical thinking and problem solving. Many technical requirements for Kukuro have already been met because its framework is similar to that of Sudoku, which is already implemented. At least three deliverables will be given during weekly team meetings to ensure quality and meet stakeholder requirements.

The final task is to create a web-interface for our game. Currently, the game is only compatible through the software, requiring the user to download and launch the game on their local machine. The issue with this is that software releases (each time a major bug has been fixed) require all users to delete their previous software and download the updated version of the software, negatively affecting user experience and requiring hard disk space. The web-interface will include basic gameplay features due to our team's lack of knowledge and experience in web development. Our team has decided to leave gameplay in 2D instead of 3D, also due to our team's lack of knowledge and experience developing 3D games in Unity.

Complete List of Client and Stakeholder Requirements and Needs:

Our current stakeholders are the students and professor of CptS 111, mentor Chris Cain, and all members of our development team.

At this point, we are prioritizing most of our development around the requirements and needs of the students and professor of CptS 111. The general requirement of this stakeholder group is that we provide a user experience that results in increased student motivation for the course material. There are many specifics to this requirement such as: patching all bugs in the current version of the software, enhancing the user interface to make maneuvering more intuitive and fluid, and giving the students additional minigames to attract a more invested and broader audience.

Our mentor Chris Cain also expects this software to be something that is directly tied into a class setting. As of now our gamification project is only available to students in college. The way to play the game relies on students being in class and completing assignments. If a student neglects going to class or turning in their assignments, they will be punished by not being able to play the game and will not be able to progress further. Instead of playing the game to be distracted from school, students will play the game for a limited time as a reward for doing well in class and doing well in class will serve as a reward in-game. This balance between time spent playing the game and time spent in class must be finely tuned, as Chris wants to deploy this game in a wide variety of classroom settings and student age groups in the future. Chris also wants the software to be deployable on the web, or at least have functionality to be updated without downloading a whole new executable so it will be easier for many users to play the game on any machine they would like. With the increase in users we will need to make sure that our software is scalable for the future.

The current Senior Design development team is the last stakeholder on the list. Our requirements are a little vaguer than the previous two, as they mostly relate to getting experience. We are looking to learn more about the game development process and everything that goes into it. Another requirement is to gain a specialized understanding in the tools that we are using to create the software, namely Unity (for the development of the game) and MongoDB (for management of our backend database) and how these two interact. We are also looking to learn about working in a team-based environment and using version control (Github) to give us good experience for our future careers in the industry. The final requirement that our team has for this project is to satisfy the requirements for our major and graduate next semester.

Mapping of Requirements to Technical Specifications:

The above requirements can translate into the technical specifications that must be met in order for the software to be functional and accepted by our stakeholders.

For the students and professor of CptS 111, their main requirement is to have a quality product at the end of our time working on this project. The technical specifications that must be met for this are as follows: First we would need to make sure the game is constantly getting feedback from the users. In order to do this, we need to keep analyzing the login trends in the database and use the already implemented tools to do so on a regular basis to check to see how our patches of the game affect user behavior within it. At some point we would implement a feedback feature within the game (this would be coded in Unity) or even more directly send an email to the users (which could be written by a script to automatically do using the database) to get a clearer idea on what they think could be improved within the game. Secondly, even without user feedback we need to constantly be working on new content to add to the game (the two new games being developed using Unity) and expand upon the existing games on a regular basis. This will help retain users and give them a fresh experience occasionally when they log in. The CptS 111 professor's requirement of "user experience that results in an increase in student motivation for the course material" would be met by the technical requirement of finely tuning balance and analysis of the time spent by the player. To do this we will need to write programs or use existing ones to analyze the database and get the information on how the player is spending time within the game. Also, math will need to be performed in order to figure

out the optimal way that progress is made in the game to ensure the student is not spending too much or too little time playing it. This math will be directly implemented into the existing code in Unity. We will also use this analysis to correlate student progress in the game vs. in class to blur the line between the two and make sure both the class and the game go hand in hand.

The requirements of our mentor Chris Cain also tie into this analysis previously mentioned. To make sure that the progress students make in the game tie in with Chris' goals for the project, our team must have an intimate understanding of both how to analyze the trends of users using our backend MongoDB database and how to alter existing code to account for these trends in Unity. We will be in close contact with our mentor throughout the entire project to make sure we do not stray from his vision for the project. To meet Chris' requirements for scalability for the future, we must make sure that our backend will be ready to handle any number of users. To do this, additional testing information and understanding of the database that uses the MongoDB system will need to be acquired by our team. The final requirement of a web-deployment will need to be researched thoroughly by our team, (as we have little experience in web development) but will eventually be implemented via Unity as it supports tools to export our game as a web application. This will also include work navigating the firewalls of WSU and EECS so users can connect seamlessly to the game stored on the EECS servers. This web-based version (or at least standalone launcher to update the game) will need to tie in with both the database and the code of the Unity project.

To meet the requirements of our final stakeholder group, our team, we must all be diligent about learning how to use the tools in our project. We must all spend time learning how MongoDB and Unity interact in order to advance not only the project for our other stakeholders, but so we can advance our own skills as computer scientists. We also must continually meet and be in communication with each other using programs such as Slack and Messenger to make sure that we are all on the same page in the project and everyone is doing their part in it. For this project from successful from our point of view (passing the class), we must be in constant communication with our instructor, Behrooz Shirazi, and our mentor, Chris Cain, as they will be the ones who grade our work. We will be in communication with them at our weekly meetings and by using our Slack channel or our Github Repository.

Preliminary Implementation Plan:

The current plan for the projects development using software is to use Unity as the primary way to develop the game. We will stick to developing the two new mini games and as well as the fixing the current mini games using the Unity 2D game development. Another important software the team is currently using is MongoDB. This is in order to get user data stored into a database safely. Using MongoDB allows for easier access of the data when the team needs to look into user activity. Lastly, the only other piece of technology that we would need to include in the project would be a website so we may host a web interface for the game later on in the project development. The team is currently more focused on getting the current game fixed and the two new mini games up and running before we start to make a web interface for the game. Unity allows the deployment of games on web interface or a mobile device, our team would have to look a little more in depth to see how exactly we may deploy the game across many platforms.

The only hardware being used in this project is the server, which is already set up and ready to use. The server is currently storing user information and is housed within the EECS department. Our team has been able to access this server for data analysis and will continue to do so as the project progresses. Our team will not need any other hardware other than our personal computers in order to work and hopefully finish this project. The data being accessed and analyzed through the server is a very crucial part for our team because it is how we will track our stakeholder's investment in our project. The more content and newer games the the

team releases will translate into more logins which we will be able to monitor through the database data component.

All the software, hardware, and data components are crucial pieces for our team's success and completion of this project. Each piece is closely related and has to work well in order for there to be an increase in stakeholder interest and team morale. If the software doesn't work well, then gameplay will lack leaving our stakeholder wanting to invest elsewhere. If our database or server goes down and the team can't analyze data then the team will be left in the dark about user experience. Our team needs to manage all of the above item very closely in order to keep our stakeholders interested and be successful in completing this project.