Software Engineering Project Report

GEODestiny

Prepared by
Glenn Turner, Eilbron Davood, Opin Patel, David Cho
for use in CS 440
at the
University of Illinois Chicago

December 2016

Table of Contents

I. Project Description

1. Project Overview

The game GEODestiny will be one of a kind full of adventures and challenges where the player will become intergraded with its rich graphics environment and realistic animation. The game will allow player to create the unique character that matches their personality and the area they prefer to start the game in. Then the player will be presented with the available tools and unique looking weapons that they have access to. Then the game will start with the player character in its particular environment related to that area. There he/she will be presented to an *NPC* called "The Guru" who has enough knowledge of that area and will help the player character guide through the challenges, win battles against monsters and mighty bosses and help accomplish the goal. The game will also allow remote multiplayer for *PVE*, in which the players create a team and fight as a team to overcome *quests*. The earned rewards will be divided among the members of the team. As the player gains *experience* through the game, quests will become difficult and the player character will become gradually stronger.

2. The Purpose of the Project

a) The User Business or Background of the Project Effort

Right now, the gaming market is booming and especially the Role-Playing Games are becoming attractions with its improved graphics and realistic animations. The performances of gaming platform have improved in a past few years and the game can be distributed among many platforms. This game will be a perfect suit to the virtual reality headset where the user will be completely encapsulated into a virtual world. This game will hit the market hard when the technology improves and people start hitting their money on the virtual headset.

GEODestiny will have good chances of becoming the favorite game for players. Where they will have an opportunity to explore the world meet their friends online and work as a team or alone to fight bosses. The challenges will keep them as a team and encourage them to work as a team. It will also motivate their friend to buy this game and join them as a team. For most part this will be a friendly game that anyone can play and increase a team spirit.

The game doesn't require much of a work to start playing if they have compatible console and good internet speed to take the full advantage of the game. The player will need to sit in front of the T.V. or monitor depending on the setup and use remote to interact with the game.

b) Goals of the Project

Deliver an exceptional RPG game that brings joy and team hood to player and have fun.

c) Measurement

GEODestiny will be fun in many ways. And we will measure the satisfaction of player in many ways. When they have finished certain stages, we will ask user to take a moment and asked for their brief response to whether they are enjoying or the game needs some improvement.

The system will keep track of minutes played. Depending on how long the user has played will determine the reward. Reward points are handed out daily for playing a certain amount of time. The more time you play, the more reward points the player receives. We will also reward the player with reward points every time their friends accept the request to be their character *budd,* in which they can showcase each other's in-game achievements. If they are enjoying the game, then they would be more likely to tell their friends and invite them to be their buddies. This doesn't bind them to fight as a team. Thus, more copies of this game will be sold.

In addition, we will have an online website where the users can take the review, rate the game and share sweet or bitter experience and in the next release or update we can fix that if they do not like that part of the game. We can also have a social media account where fans share their stories creating buzz among teens.

Also after the customer service help, we will invite the user to take a brief survey about their experience. From that we can try to learn and improve the next release for greater satisfaction.

3. The Scope of the Work

The game will be played mostly at home in front of a T.V. using the console like XBOX and Play Stations. The user may not be able participate in multiplayer activity if the supporting internet connection is poor. The game may take the entire screen of the T.V. to provide the best visual quality. The game may have a sound and user can control whether to use headphone or not which may disturb other members in the family. The user may need to sit close to T.V. if the remote is wired.

a) The Current Situation

The user may or may not have the compatible console. A supported console will be required to play the game. Users may also need compatible internet speeds (10+ Mbps) for the multiplayer gaming experience to reach optimal performance. If the user never used any console before, he or she may need to learn to how to set up the console and use learn to use the required hardware such as the remote. With the graphics requiring state of the art resolution and display, the user mush meet the minimal specifications for accepted screen resolution.

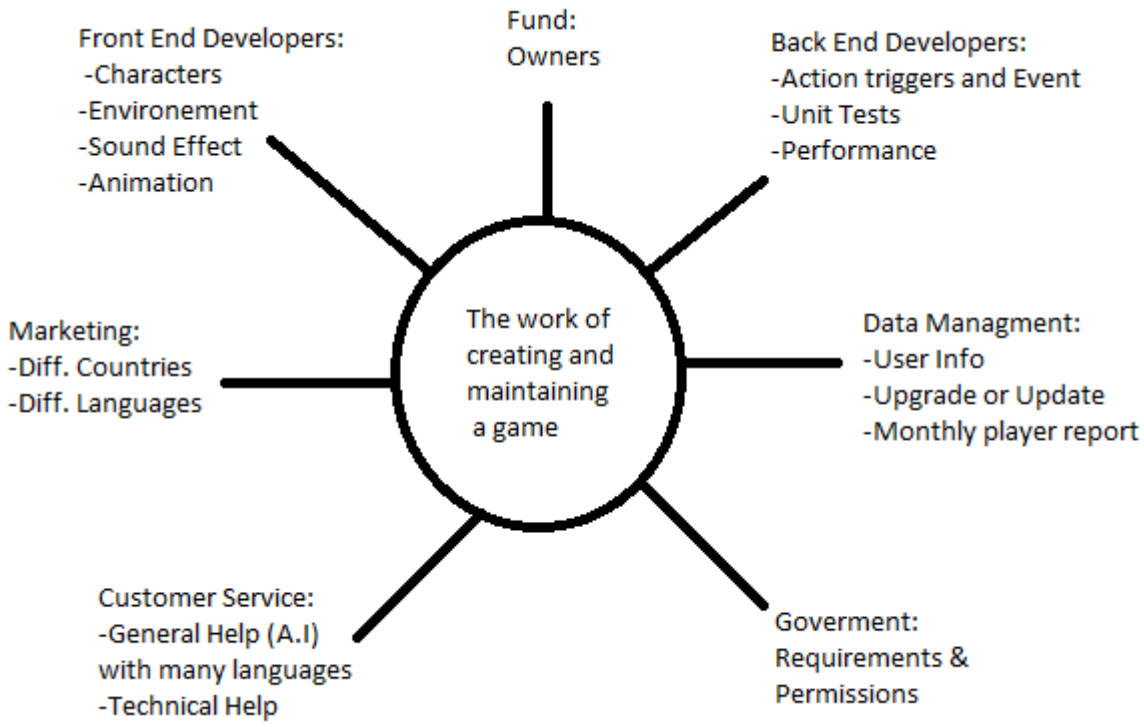
As a developer, we will try to make the game available on as many consoles as possible so the user doesn't have to worry about upgrading the devices. Currently, gaming has become very common among children and teens. The average internet speed of U.S. home is about 10mbps and most homes have newer compatible T.V. that can connect with the console. User may not require much of an upgrade if they have played other games on their console.

b) The Context of the Work

During the development process, the game project will be divided into many parts which can be further divided into other sub parts. The main division of the game will be the front-end work, the back-end work, database management, marketing, customer service and funds. We can partition front end work into people who create the characters vs. who create the environment for the game vs. the sound effect vs. the animation. The back-end developers can be divided into people who creates an action triggers an event vs. the Unit Testers vs. performance measurer of the game. Sales can be in different countries in different markets. We can also have game available in multiple languages to compete in different markets around the world. The customer service can be divided into a general setup of the game vs. Technical problems like data loss, connectivity with internet issue. We need a database system that stores the user informational and possibly game update after the game has been distributed. We may need to fulfill any government requirement and permissions before it makes it to the market.

The motivation behind it is the successful end product that will put smile on the users face.

Context Diagram:



c) Work Partitioning

Business Event List

Event Name	*Input and Output*	*Summary*
1. Designers Creates Characters	Characters (Out)	Player characters, unique tools and weapons, bosses and monsters
2. Designers creates Environment	Environments (Out)	Different environment for different areas around the world
3. Sound to Game	Sound(Out)	Theme Song, background Sound, Action Sound
4. Animation	Character movement (Out)	Character movements, different views from different angle
5. Back end programmer connects Actions with Events	User Input (in) Event output (Out)	Controller input triggers an event inside a game

6. Unit Testing and security	Input the unit tests Advise to programmers	Programmers and security experts test the code for any flaws and test on different consoles
7. Performance Measure	Code Performance	Improve the speed of the program, increase the compatibility across all platform
8. Marketing	Find market for the game, publication	Find the suitable market to publish the game, advertise it and sell around the world
9. Government	Copyright and permissions	Comply with government policies and codes
10 Customer Service	Help and Support(Out)	Provides help and support to customers in different languages
11. Handling Data	Handling Upgrade and User Info	Stores player info and any updates to game

d) Competing Products

Use Ice Scrum to remotely collaborate on the project. It is helpful tool to see the process of the project, due dates, work being done so far. Members decide to work on a certain part of the project. It is easy to see the member's progress, easy to add new ideas to the project. Create new stories and assign the specific time limit to it.

Unity is a powerful game development engine that can be used to create the specific environment and speed up the development process. It can be used to create a 3D environment.

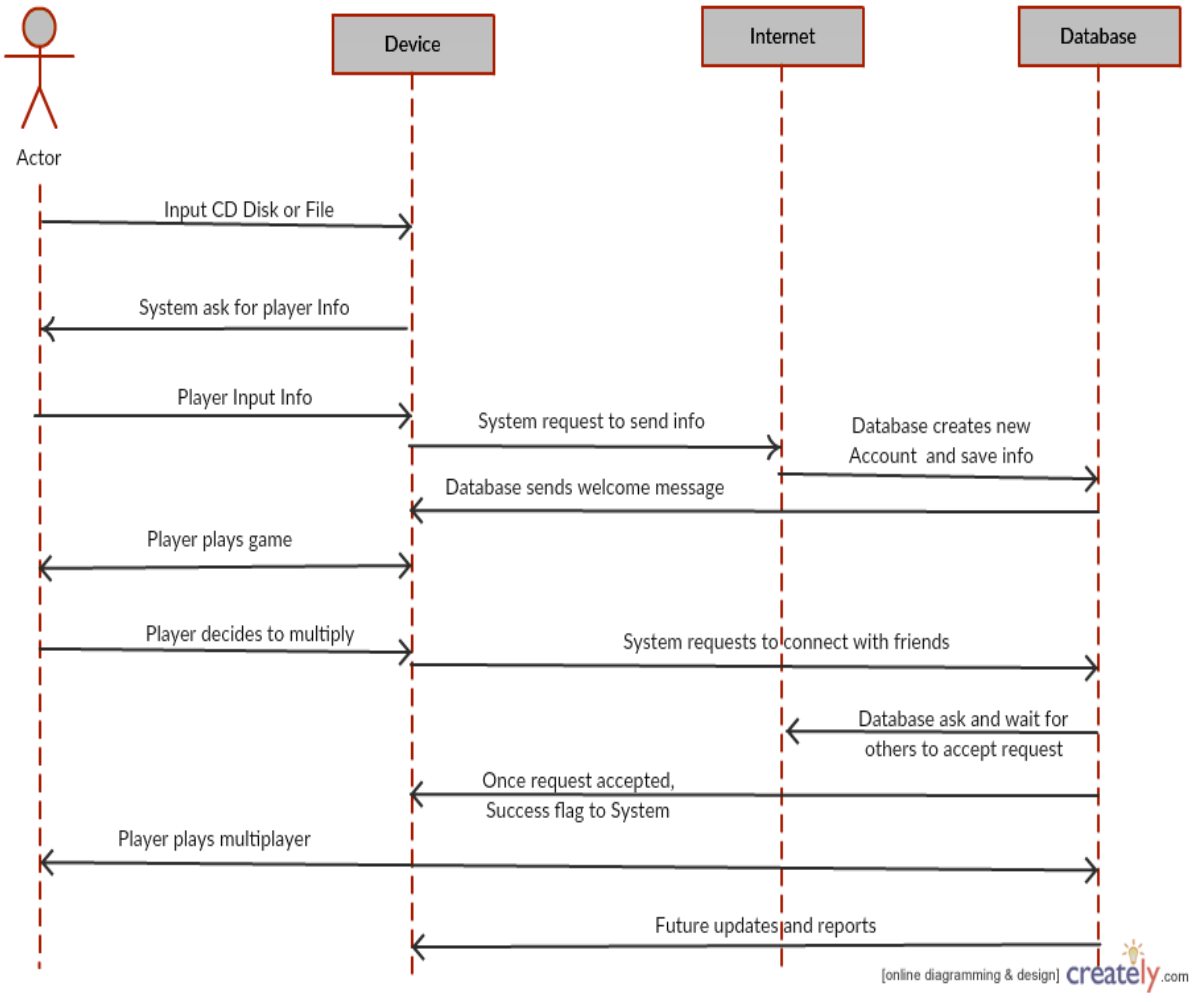
We can use artificial intelligence voice to help the customer with basic questions they may have in different languages to reduce the cost of customer service. We can also outsource the technical support calls to reduce the expense cost. Also, we can have a product website where users interact with each other and help the community.

As far as the data management is concerned we can use the services like Amazon cloud to store player data and updates. This way we don't have to deal with physical servers, maintenance cost or scalability.

4. The Scope of the Product

The product will be in a form of a CD disk if purchased from the store or a downloaded file if purchased from the online store. The CD will be configured to work with compatible consoles and devices. Once it's on and running the system will ask the player to create an account and add information. Then, the system will send data to the remote server via internet connection and save user information. If player decides to play online, he will request system to connect with friends. System will wait for friends to accept the request. Once accepted player will be notified and they can multiplayer. After that, most of the interaction will happen between the user and the system.

a) Scenario Diagram



b) Product Scenario List

Starting the game

Creating an account

Local if no internet connection

Database will save player info if internet connection

Playing game

Choosing an environment

Choosing a player character and familiarizing with weapons

Starting the game

Multiplying

Player request to connect with friends and play

Future updates and reports

Share the report

c) Individual Product Scenarios

Starting the game: A customer will buy a physical CD from the store or download it from the online store. Before he starts the game, he should have a console connected with the internet. He should also have a proper remote connected with the console to interact with the game. Then he may insert the cd into a cd try and let the system boot up. Once the system boots up, it will show a small clip of a game trailer. which will show all the future that the game provides.

Creating an account: After the video clip, a new screen will pop up with key board in it. The player will be asked to provide his name, valid email address and unique name, if connected with the internet, to distinguish players in the database. If the player is not connected with the internet system will store info locally. There will be plus sign symbol incase if the player needs to create multiple accounts which will store individual process on hard drive.

Playing the game: Once the account has been created, the system will bring new screen where the player will choose his initial environment. He will be presented with the different environments and the futures that they come with it. There will be about 20 different areas around the world.

Once the player makes choice of an environment, he will be presented with the available player characters in that environment. He can change the style of a character which includes clothes, shoes, hair style and other ornaments.

Once he is satisfied, he will be presented with the unique looking weapons he has and the power that comes with it. The system will reward 50 points for having an awesome look which the player can use to buy stuff and upgrade. After that, the screen will load and display a message “Creating an environment”.

Once the environment has loaded, he will be ready to play the game. If he needs any help anytime while playing he would be able to quickly access the help by pressing the button on the screen. The game will pause and help screen will slide from right side of the display screen and show help related to that stage. When he is satisfied with the help he will play button to resume the game.

Multiplying: In the game, if the player decides to fight a match with his friends rather than alone, the system will check if the user is connected with the internet. If he is then the system will ask the player to enter the unique name of his friends. Once entered, the system will send request to that friend to accept the invitation to fight the game. Upon acceptance, player will be asked to enter the team name, so other friends can easily find a team and join it. When they battle as a team, all the member of the team will appear on the screen.

Future updates and report: Whenever there is update from the developers, the system will let the player know if there have been any changes from the last play. If it requires any authentication the pop up window will ask for the user input. At the end of each month, the system will request the user to view his progress. They would also be able to share the report with the team or particular member of a team.

5. Stakeholders

a) The Client

There are multiple key stakeholders for GEODestiny. Since the game is developed in-house, our key stakeholders will be departments working in our company. Listed below are groups that are most important.

Owner

The owner is the main resource for capital. He funds all projects and makes key decisions about which projects are allowed to go through.

Content Creators

The content creators are extremely the ones creating the game content and characters. The success of the game relies heavily on this department.

Designers

The designers implement the actual game. They coordinate together with the content creators to provide maximum efficiency and minimize errors. Game aesthetics is a key component to our vision so work heavily on visual effect, game flow, and smoothness are important.

b) The Customer

Gamers

The bulk of our key stakeholders will be the gamers themselves. Without them, GEODestiny will not be able to grow. The number of users will directly relate to newer game content, technology, and innovations for the game.

This game was created for people who like to play single or multiplayer *RPG* games. It is also aimed more towards *casual gamers*, but also benefits the *hardcore gamers*. The target audience are young teens to adults (13-30+) and for any gender.

c) Hands-On Users of the Product

Parents

Role – Their responsibility is to purchase the game for their kids, but can also play the game themselves.

Subject Experience – For the most part parents know the popularity and genre of the game. Since most parents are concerned about the type of game, they tend to have a good knowledge of the story line of the game.

Technological Experience – In terms of game related jargons, quests and quests, parents tend to have very little knowledge of the game.

Bloggers

Role – Their responsibility is to provide information to audiences that might be interested about this game.

Subject Experience – Bloggers tend to be more casual gamers. They tend to have good knowledge about the game, and can explain casually of the story line.

Technological Experience – As stated above blogger's are casual gamers. They tend to know the storyline, in-game features, and jargons used in the game; however, they do not know all the ins-out of the game.

Game Enthusiasts

Role – Their responsibility is to know in-depth about the game. Provide guides, short cuts, and any hidden secrets to the users playing this game.

Subject Experience – Game enthusiasts tend to be more hardcore gamers. They tend to have really good knowledge about the game, and can explain in-depth of the story line.

Technological Experience – Game enthusiasts are experts in the game. They know every little detail from storyline, in-game features, and game mechanics.

d) Hands-On Users of the Product

Key Users

The key users for this game are the gamers themselves. It does not matter if they are casual or hardcore. If they have a complaint of suggestion about the game, their voices will be heard. In-game features, character mechanics will reflect on the community's suggestion/complaint. Also these users are users that have played for at least 15 total hours since the game's release. They are categorized as active. Active players will have in-game perks such as daily rewards.

Secondary Users

Secondary users are the users that have accounts, but played between 5-10 hours of the game since the release. They are categorized as semi-active. They will not have access to daily rewards, but their suggestions and complaints will be heard.

Unimportant Users

Unimportant users are users that have played for 1 hour or less total hours of the game since release date. They are categorized as inactive. They do not receive in game rewards. The only value we have for them is for data analysis.

e) User Participation

User participation will be essential during pre, beta, and after release of the game. During the pre-stage, only select handful will be allowed to test the game. During the beta phase,

30% of the general public will have access to the game. They will be given game-keys so they can access and download the game. At this stage, user's feedback will be a crucial because this determines whether we keep certain features or remove them completely. Finally, the after-release stage anyone who bought the game can play the game.

f) Maintenance

There will be several types of maintenance to the game. There will be in-game maintenance which happens bi-weekly. There are quest updates that happens monthly. Finally, there will be hot fixes which can happen any point in time, but users will not be affected by this.

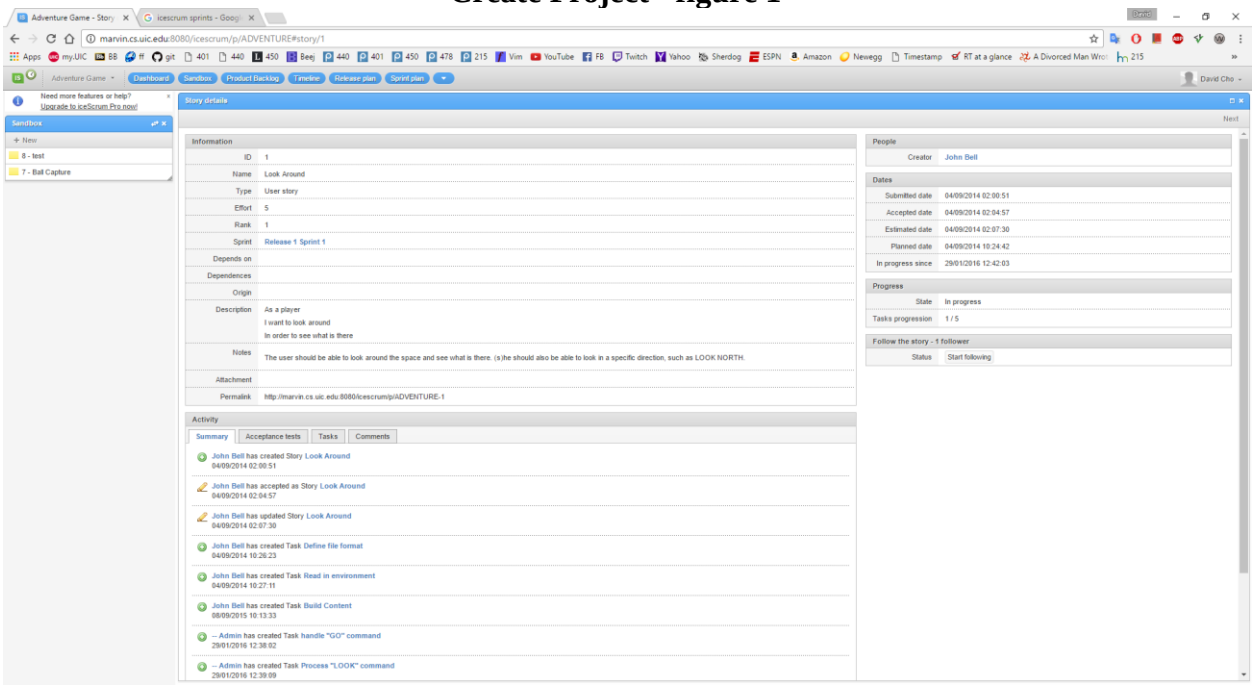
6. Mandated Constraints

Constraints involve using iceScrum to collaborate on projects. All features in ice Scrum will be used so employees will need to be very familiar with this application. Ice Scrum will be used on a daily basis.

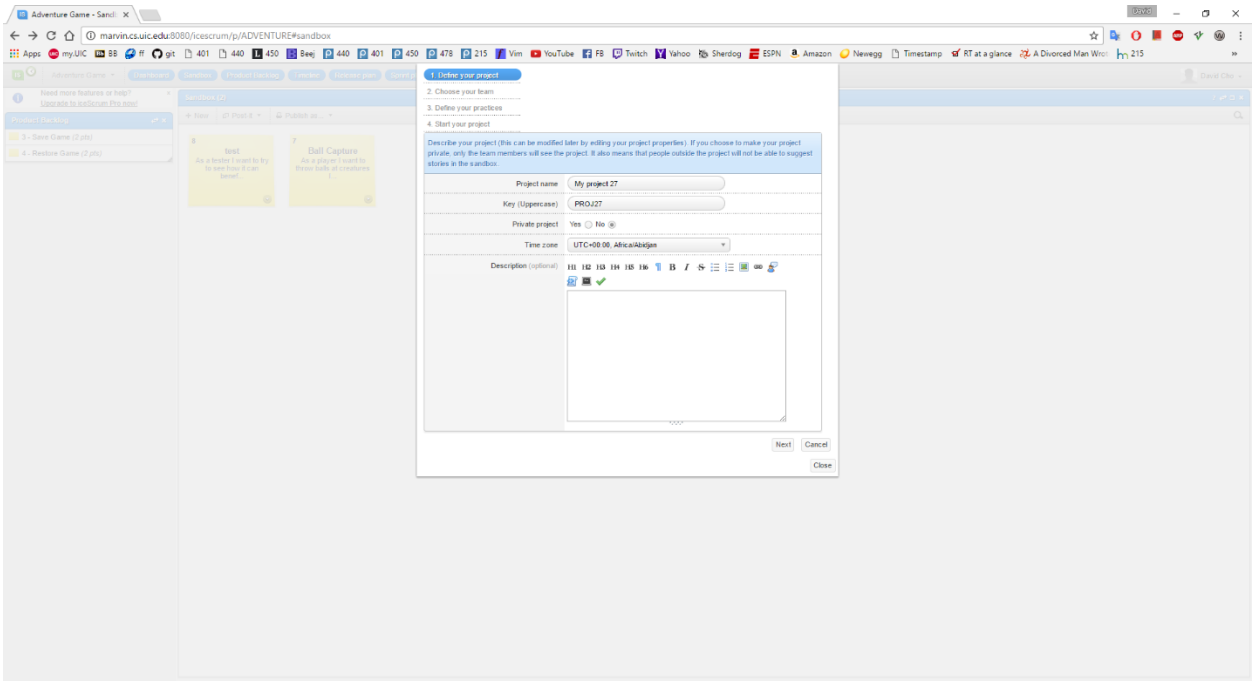
a) Partner or Collaborative Applications

Since Ice Scrum is purely a collaborative workspace, every department will be using this application. We will be collaborating with various departments to work on several projects. Below are the key features for using iceScrum.

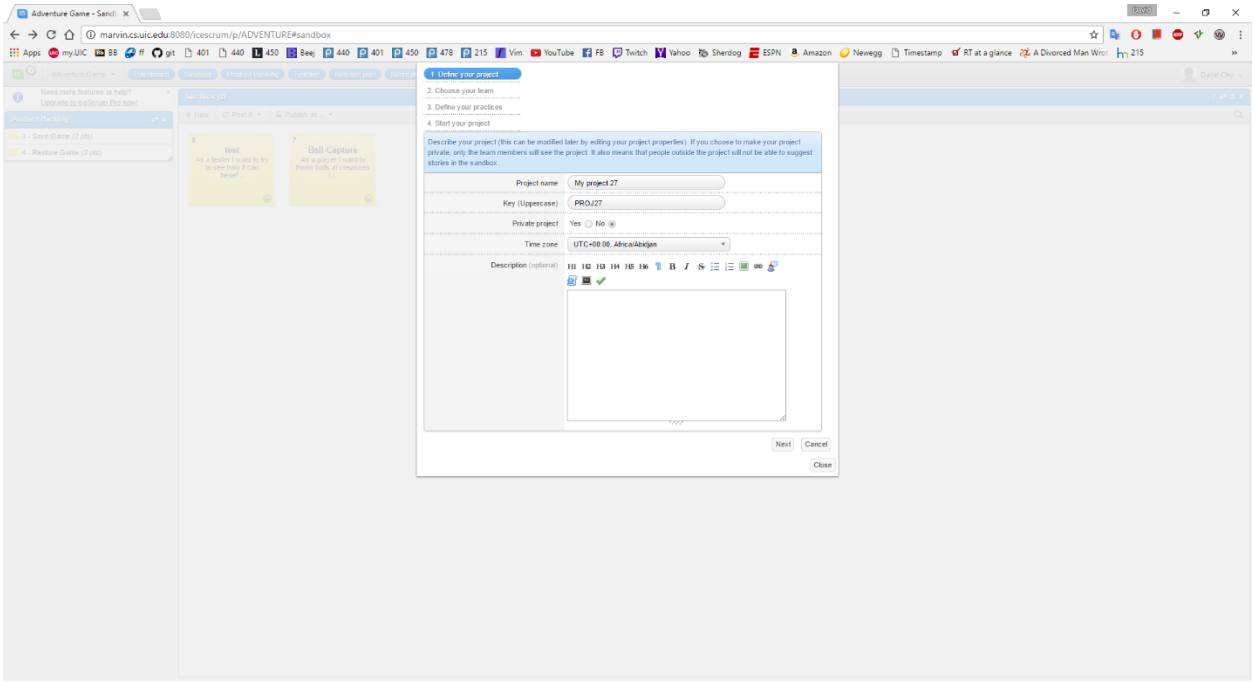
Create Project - figure 1



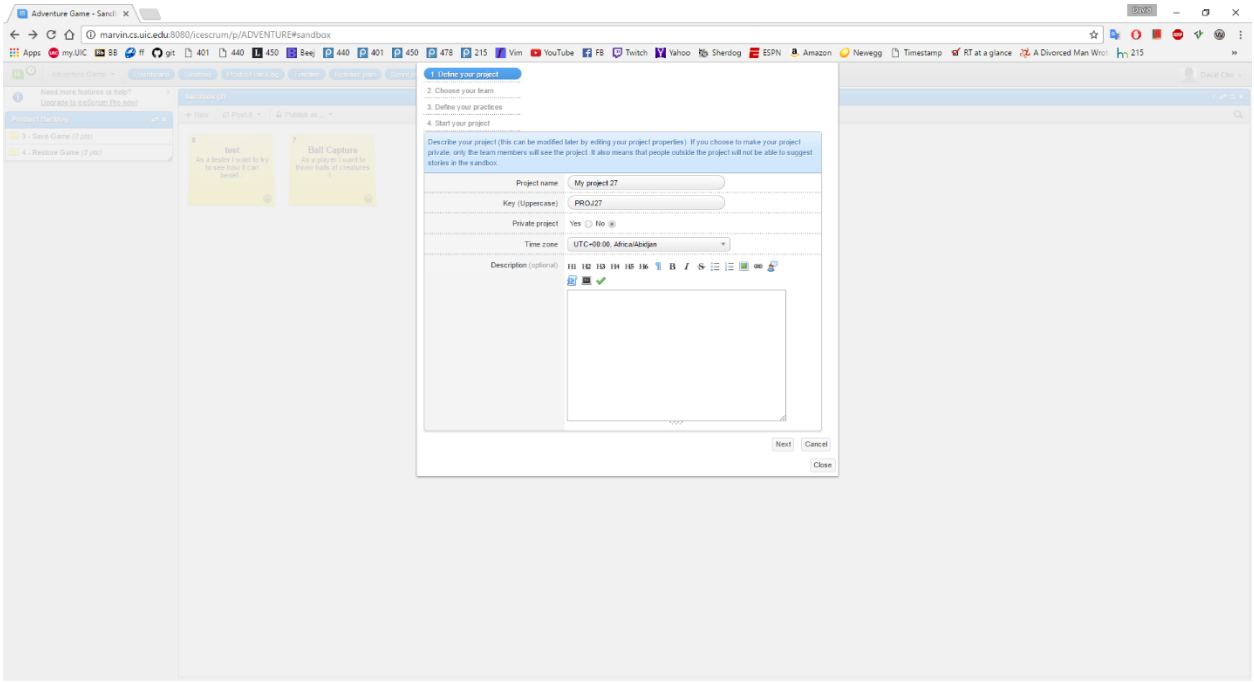
Select Team - figure 2



Start Project - figure 3



(4) View backlog and work on with team - figure 4



When logging into Ice Scrum, the user has the option to create project (figure 1). When creating project, the user will set the project name and add in a comment. The next step is to select the team (figure 2). In the figure there is only one user, however, once everything is setup, there will be several the user can choose from. Everything the project has been created, the user will be back to the home screen. From there, the user can select backlog and all projects that the user is working on will display (figure 3). Finally, when selecting (figure 4) a project from the backlog, the user can view the details, and every update, edit any user has made. Users will be doing these four steps on a daily basis.

b) Anticipated Workplace Environment

Anticipated workplace environment for Ice Scrum will be at the workplace and remote locations such as home. Ice Scrum can be accessed from anywhere as long as there is a network connection.

c) Schedule Constraints

All deadlines will be notified in Ice Scrum. If a project is not met by the deadline, there will be an extension period. Generally speaking, most project extension will be one week.

d) Budget Constraints

All projects will be funded by the owner. He will determine whether a project gets funded or not.

7. Naming Conventions and Definitions

a) Definitions of Key Terms

All Terms, Including Acronyms and Abbreviations, used in the Project must be defined at some point. List the most important ones here, and refer the reader to the glossary on page **Error! Bookmark not defined.** for a complete list. (Note: that page number is a cross-reference, and will automatically be updated whenever the glossary moves)

NPC – Referred to as non-player character. The NPC is The Guru who is a character that is not controlled by the player. The NPC is controlled by the computer and is programmed to give knowledge to the player.

PVE – Player versus environment. This term is used to fighting versus controlled enemies.

Quests – A task that a user or group of characters may complete in order to gain a reward and experience.

Experience – Is a tool to determine a player's character rank. When a user completes a quest the user gains experience. Experience leads to stronger characters and better equipment.

Role Playing Games (RPG) - A game in which players assume the roles of characters in a virtual world.

Megabits per second (Mbps) – A term that often measures of data transfer through the network. It is used to measure internet speed.

Budd - connect with other friends and share their achievements but they are not bound to fight as a team

Casual Gamer – a user who plays a game as a means of socializing or connecting with friends. Casual gamers main objective is to have fun rather than increasing their skill in the game. Casual gamers play anywhere between 1-5 hours a week.

Hardcore gamer – A user who plays games as a primary hobby. Hardcore gamers main objective is to increase their skill in the game. They typically play 10+ hours a week.

8. Relevant Facts and Assumptions

a) Facts

Creativity is the key factor of this games. In RPGs players have the freedom to choose their own character and move the story forward the way they want it which makes the game more interesting for them.

b) Assumptions

Project scope may be modified in the future

Teams stay together over time

The first prototype will be released in march 2017

Project should be completed in 2 years with minimum cost.

II. Requirements

9. Product Use Cases

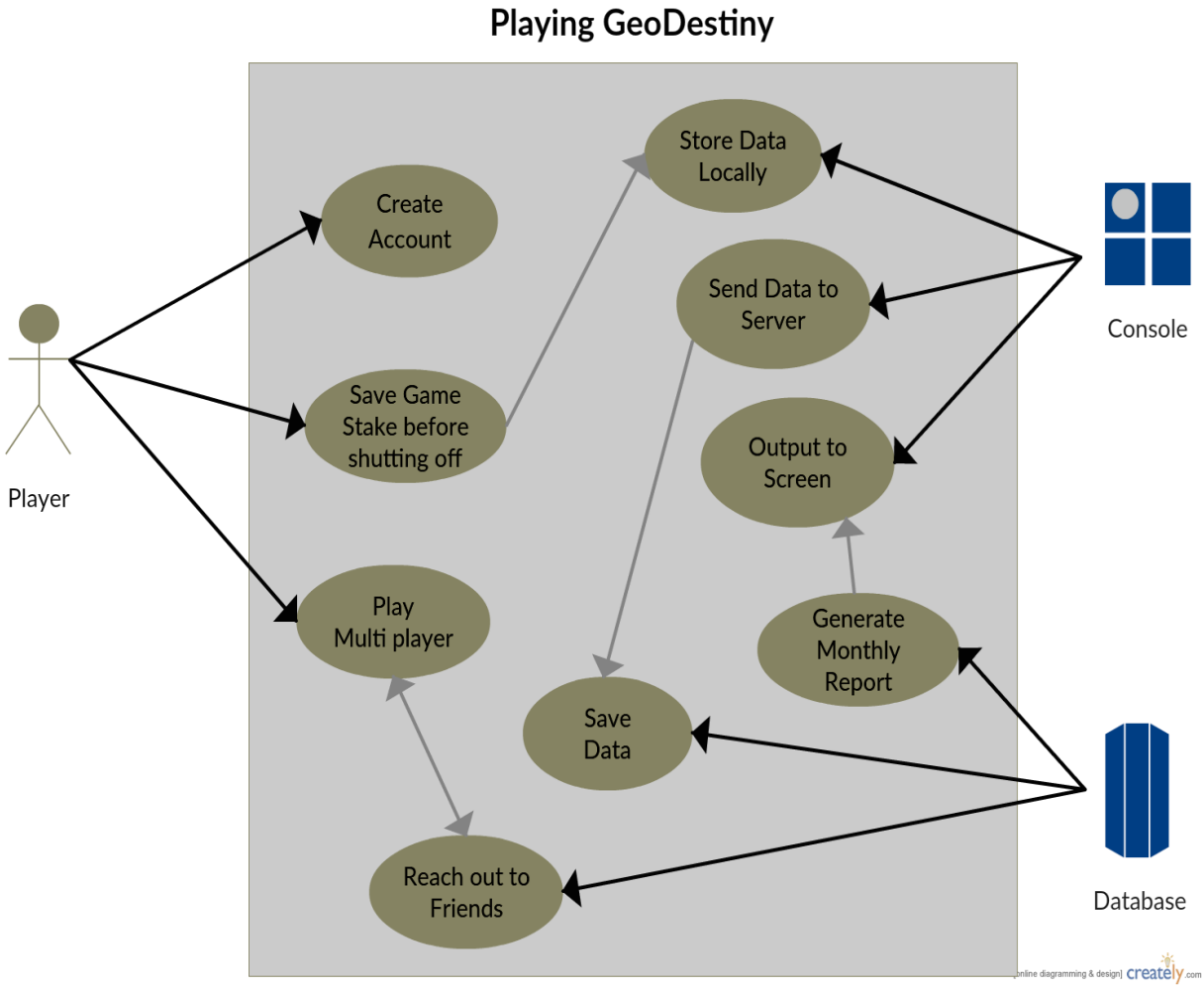
Use cases are important when it comes to understanding the system. It clears who is involved in what action. Here we will try to ease and show all the entities that are involved when the player is playing GEODestiny.

a) Use Case Diagrams

Use Case diagrams serve two purposes: As a form of graphical table of contents listing the individual use-cases, and to define the boundary of what is included as part of the proposed system and what is not included.

A use case diagram identifies the boundaries between the users (actors) and the product. You arrive at the product boundary by inspecting each business use case and determining, in conjunction with the appropriate stakeholders, which part of the business use case should be automated (or satisfied by some sort of product) and what part should be done by the user. This task must consider the abilities of the actors (section 3), the constraints (section 4), the goals of the project (section 1), and your knowledge of both the work and the technology that can make the best contribution to the work.

On the next page we have shown the use case diagram with all of the outside entities envolved in successfully playing game



b) Product Use Case List

Since out system deal with the limited Use Cases we do not need the Product Use Case List.

c) Individual Product Use Cases

Use cases are like scenarios; in that both tell the story of how the system interacts with the user(s) in response to some business event or while conducting some business task. The difference is that use-cases are much more formal, with certain pre-determined sections for each use-case, and that use-cases indicate clearly what action the system takes in response to what actions taken by the user.

Use case ID: 1 Name: Create Account Pre-conditions: Player must have remote to interact with the system Post-conditions: Player account will be created Initiated by: Player Triggering Event: System creates a new account Additional Actors: Console
Sequence of Events: 1. First time using the product, Player inserts the media 2. System welcome the player and display button to create account 3. Player clicks on create account button 4. System Displays form to fill out 5. Player fill out form and click submit button 6. System will save player info and save it to local hard-drive
Alternatives: Creating an account is normal and expected Exception: NA

Use case ID: 2 Name: Play Multi player Pre-conditions: Player must be connected to an internet Post-conditions: Player will be able to play with his friends online Initiated by: Player Triggering Event: Player will play with his friends Additional Actors: Database
Sequence of Events: 1. Player decides to play multiplayer and click button 'Join the Team' 2. System buffer for less than 5 seconds and display the teams 3. Player selects the team to join 4. System load other players and the mission 5. Now player will be playing online with his friends.
Alternatives: Player can play offline without the need of an internet Exceptions: Slow internet might lag the game

Use case ID: 3 Name: Storing data locally Pre-conditions: Console must have memory drive with 1 Giga Byte of space Post-conditions: Game data will be saved Initiated by: Console Triggering Event: Saves game stake Additional Actors: Player
Sequence of Events: 1. Player decides to quit the game 2. System will ask the user to press ‘A’ button to save game or ‘X’ to quit 3. Player clicks either button 4. System will take input and update the stake 6. System will display message ‘Saved Changes’ and quit
Alternatives: NA Exceptions: drive might not be properly installed or might not have enough space then system will display appropriate message to player

Use case ID: 4 Name: Sending data to server Pre-conditions: Memory card with data must be installed on the console and have internet connection Post-conditions: Remote database will be updated Initiated by: Console Triggering Event: Update data base Additional Actors: Database
Sequence of Events: 1. Console will check for any new update locally and reach out to remote database 2. Console will send data to remote server 4. Server will update the player data and info
Alternatives: NA Exceptions: Player might not have an internet and he will never be able to save data to remote server

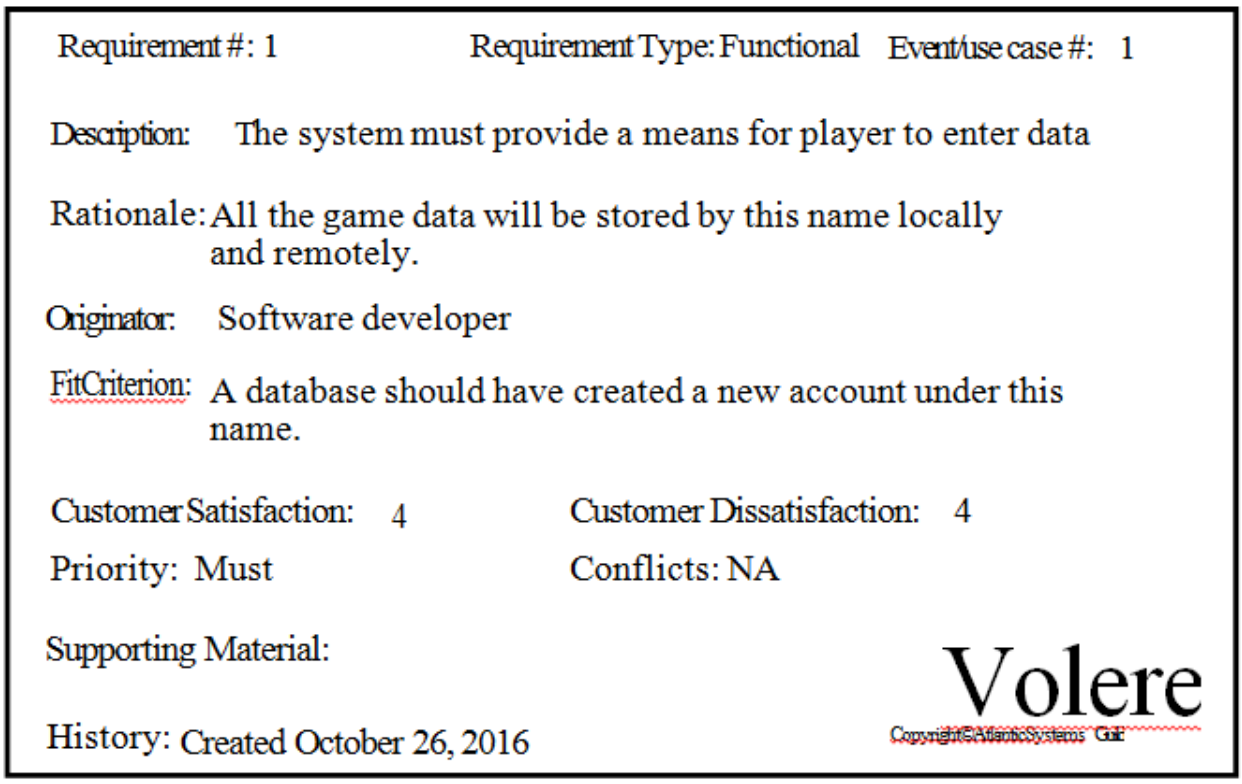
Use case ID: 5 Name: Monthly report generator Pre-conditions: Player is an active player means he plays more than 10 days a month and connected to an internet Post-conditions: Player will be able to see the monthly report Initiated by: Database Triggering Event: Display the report Additional Actors: Console, Player
Sequence of Events: 1. Server generate the player report 2. Server send it out to the console 3. Console saves monthly report 4. when the player is on the game, System pops up 'Monthly Report' icon 5. Player clicks on it to view 6. System displays a report
Alternatives: NA Exceptions: NA

10. Functional Requirements

Content

A specification for each functional requirement. As with all types of requirements, use the requirements shell. A full explanation is included in this template's introductory material.

Motivation



Requirement #: 2 Requirement Type: Functional Event/use case #:2

Description: The system must try to communicate with the remote server

Rationale: It will enable multiplayer option and store data remotely in case of backup

Originator: Network developer

FitCriterion: Must enable multiple consoles to communicate with each other

Customer Satisfaction: 5 Customer Dissatisfaction: 5

Priority: Must Conflicts: Poor connectivity or no internet

Supporting Material: Internet

History: Created October 26, 2016

Requirement #: 3 Requirement Type: Functional Event/use case #:3

Description: The system must store data locally

Rationale: It will save player progress locally and take any update from the remote server

Originator: Network developer

FitCriterion: Local hard drive must update its files to record any change

Customer Satisfaction: 5 Customer Dissatisfaction: 5

Priority: Must Conflicts: Not enough space on drive

Supporting Material: NA

History: Created October 26, 2016

Requirement #: 4 Requirement Type: Functional Event/use case #: 10

Description: The system must provide a means for user to play/pause

Rationale: It will provide more flexibility to player when taking short breaks.

Originator: Software developer

FitCriterion: Game stops and graphics stays still until the player hit the play button. Any update to score will also stop

Customer Satisfaction: 3 Customer Dissatisfaction: 4

Priority: Must Conflicts:

Supporting Material: NA

History: Created October 26, 2016

Requirement #: 5 Requirement Type: Functional Event/use case #: 10

Description: The system should provide a small tutorial when played first time

Rationale: It will more familiarize the player with the product use.

Originator: Game Designer

FitCriterion: The step by step tutorial should cover the remote functionality and available options to the player

Customer Satisfaction: 3 Customer Dissatisfaction: 2

Priority: should Conflicts:

Supporting Material: NA

History: Created October 26, 2016

Requirement #: 6 Requirement Type: Functional Event/use case #: 5

Description: The system must generate monthly progress and report to the player

Rationale: It wills unable player to look at the achievements, stages finished, progress made compared to other moths.

Originator: Software Engineer

FitCriterion: Keep track of the data from the last month and notify any changes at this month

Customer Satisfaction: 4 Customer Dissatisfaction: 2

Priority: Must Conflicts: NA

Supporting Material: Database

History: Created October 26, 2016

Volere

11. Data Requirements

The game will use single database where the player information will be stored and maintained. It will keep track of all the cities the player has visited, achievements, scores, weapons available to the player.

12. Performance Requirements

a) Speed and Latency Requirements

- The game should not load more than 5 seconds
- System should not take more than 8 seconds to connect to server to multiply
- Multiplayer response should not be lag with the average internet speed of 10 mbps

b) Precision or Accuracy Requirements

- The score should be precisely calculated and updated

c) Capacity requirements

- The game should allow up to 50 people to join as a group to work on a single mission.
- Database must store the information for at least 5 years for non-active player after the account creation
- Up to 10 accounts can be created on 1 C.D. product

13. Dependability Requirements

a) Reliability Requirements

- The server should be up 99% of the time
- Player data should be backed in server at certain point in the game
- Loading the particular stage almost never fails
- Scores should be up-to-date at all time
- Appropriate message should be displayed in case of failure
- In the case of unexpected interrupt, it should save all the data before reloading the system again

b) Availability Requirements

Server should be up and running everyday unless it is being maintained

c) Robustness or Fault-Tolerant Requirements

In the case of internet connectivity loss to one player, system should not stop group from moving forward

d) Safety-Critical Requirements

The game should not have violent content and must comply with the government regulations.

14. Maintainability and Supportability Requirements

a) Maintenance Requirements

Database should be maintained semi-annually

b) Supportability Requirements

- Technical support should be provided via customer representative, chat box and email
- Non-technical support via artificial intelligence generated phone call
- Online community support using discussion forum

c) Adaptability Requirements

The product must work with the current compatible consoles and upcoming products.

d) Scalability or Extensibility Requirements

Initially, the database should be capable of handling 100,000 players and it should add additional 100,000 entries as the limit hits.

e) Longevity Requirements

- The product c.d. should last for at least 5 years on normal use of 20 hours per week
- The data on the server will stay for at least 5 years after the player becomes non-active

15. Security Requirements

a) Access Requirements

- Access to GEODestiny is free with your copy of GEODestiny
- The players must have PlayStation or Microsoft account to download updates.

b) Integrity Requirements

- The product shall protect the player personal information
- Any unauthorized modification of data must yield an auditable security-related event.
- The product shall prevent incorrect data from being introduced**.**

c) Privacy Requirements

- The product shall describe the type of personal information collected and how they are collected.
- The product shall protect private information in accordance with the relevant privacy laws and the organization's information policy.
- The product shall describe how the personal information will be used.
- The product shall provide a list of all operators collecting personal information.

d) Audit Requirements

The product shall perform verification and validation checks in real-time

e) Immunity Requirements

- The product shall update its list of published definitions of known harmful programs daily.
- The product shall notify the security team if it detects a harmful program during a scan.
- The product shall scan all downloaded data against unauthorized or undesirable software program.
- The published definitions of known computer viruses, worms, and Trojan horses.

16. Usability and Humanity Requirements

a) Ease of Use Requirements

- The product shall be easy for 13-year-old children and up to use.
- The product shall be easy to learn for both novices and experienced users.
- Once the player has signed-in for the first time, he will no longer need to sign in future games.
- The user shall feel satisfied with the product.

b) Personalization and internationalization Requirements

The product shall retain the buyer's buying preferences.

The product shall support the following native languages:

- English
- Chinese
- Spanish
- German
- French
- Italian
- Polish

c) Learning Requirements

An average user should be able to learn how to play this game with little to no training.

d) Understandability and Politeness requirements

Interface elements (e.g. menus) and game environment should be easy to understand

e) Accessibility Requirements

There are no accessibility requirements for this product.

f) Training Requirements

The product does not need a special training. The "Guru" (character) is the main learning tool for this game.

17. Look and Feel Requirements

a) Appearance Requirements

- The product shall be attractive to a teenage audience.
- The characters shall be customizable.
- The product shall comply with corporate branding standards.

b) Style Requirements

The product shall have an updated and modern user interface that is appealing to the average user.

18. Operational and Environmental Requirements

a) Expected Physical Environment

There are no expected physical environment requirements now.

b) Requirements for Interfacing with Adjacent Systems

The product shall interface with the applications that run on the remote weather stations.

c) Productization Requirements

The product shall be able to be installed by an untrained user.

The product shall be available on DVD to purchase at retail stores or as digital version to download.

d) Release Requirements

- Each release shall not cause previous features to fail.
- The download of a new release shall be done through the gaming consoles in the same way as downloading the game.
- Official releases shall happen approx. every 6 months.

19. Cultural and Political Requirements

a) Cultural Requirements

The product shall not be offensive to religious or ethnic groups.

b) Political Requirements

There are no specific political requirements

20. Legal Requirements

a) Compliance Requirements

Personal information shall be implemented to comply with the Data Protection Act.

III Design

21 System Design

Design goals

- **Manageability**: The software should efficiently and easily be monitored and maintained to keep the system performing, secure, and running smoothly.

- **Performance**: The time required to add/update/retrieve data to the database should be minimum

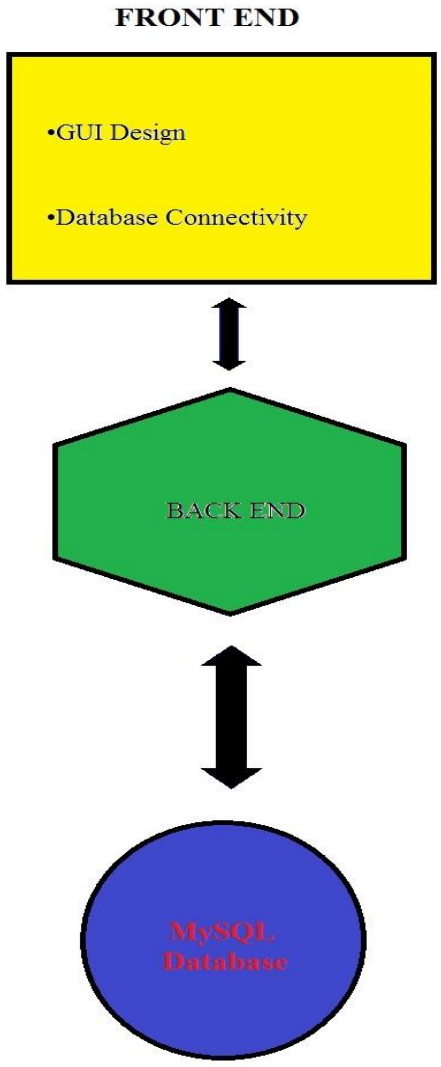
- **Reliability**: Software reliability is an important factor that affects system reliability. In order to keep our data safe, a backup of the data is generated and stored each day.

22 Current Software Architecture

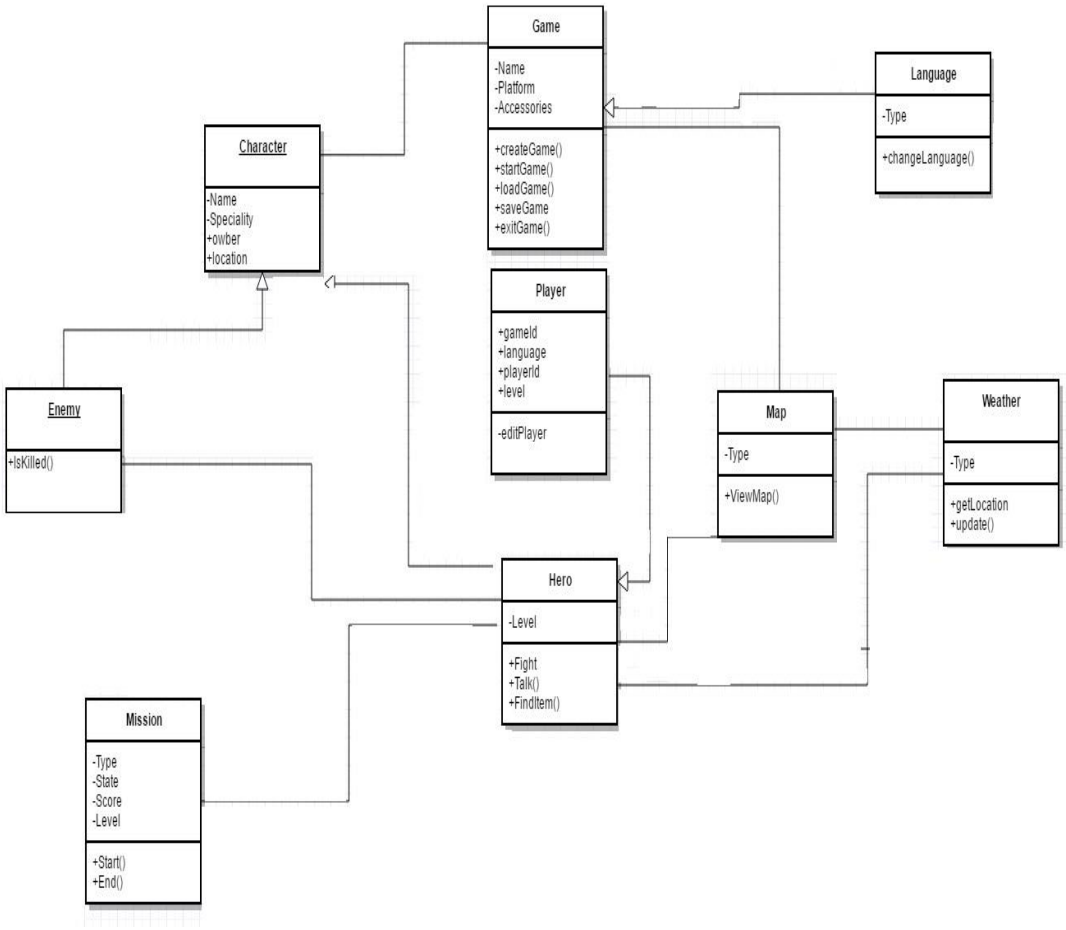
Currently there is no software architecture for this product.

23 Proposed Software Architecture

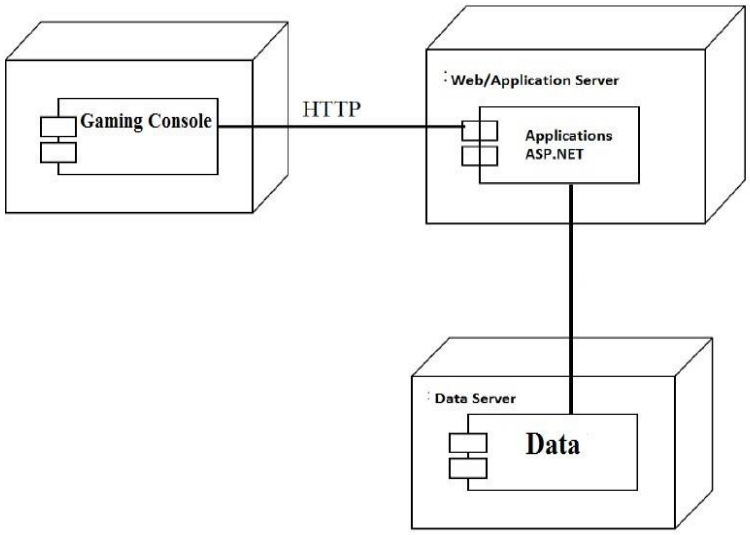
a) Overview


FRONT END
•GUI Design
•Database Connectivity
BACK END
MySQL
Database

b) Class Diagrams



c) Hardware / software mapping



d) Global software control

- Database connection must not be open all the time, instead the connection to the database should be made only when any functions requests data or wishes to update data.

e) Boundary conditions

- Depending on the type of database we use, the fields might have a maximum width value. We need to make sure there is enough space in each of the fields.

26 Object Design

a) OBJECT DESIGN TRADEOFFS:

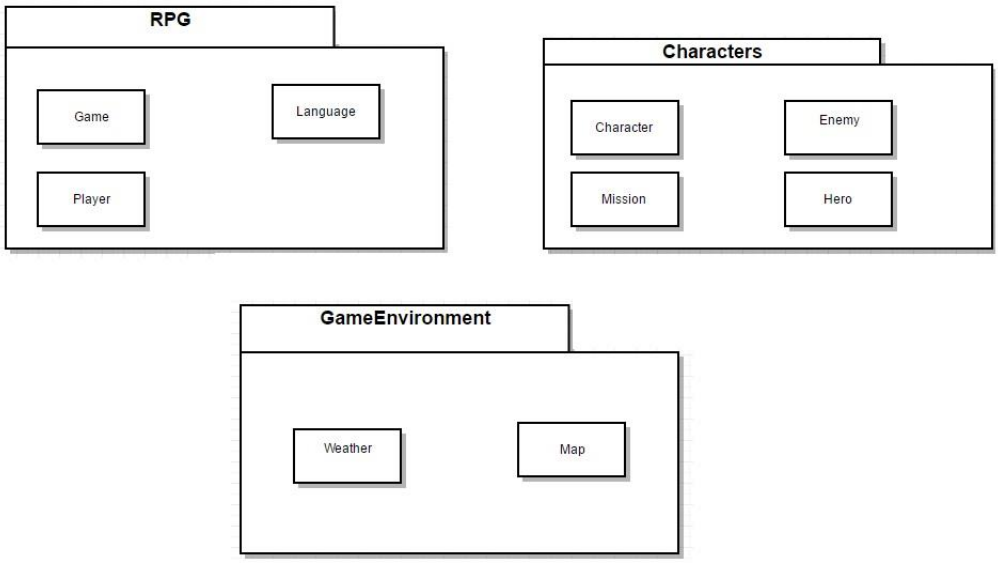
- **Space vs. Speed:** This product requires a lot of space for the storage of data. Therefore to make the system read and write data at a faster speed we need more memory space.

- **Buy vs. Build:** *"If you can get 80% of the functionality you need from a packaged solution, then buy it. Otherwise, build it."*

b) Interface Documentation guidelines

- The names of the class should start with an uppercase letter and if the class name consists of more than one word then *CamelCasing* should be used.
- The class methods begin with a lowercase and if the method name consists of more than one word, *camelCasing* should be used.
- The Package names should start with and uppercase letter. Also the package names must end with the word 'Package'.
- Methods are named with verb phrases; field and parameters are named with noun phrases.
- Constants are represented by all uppercase letters and constant names with multiple words should be separated with underscores.
- Errors status is returned in the form of an integer.

c) Packages


RPG
Game
Language
Player
Characters
Character
Enemy
Mission
Hero
GameEnvironment
Weather
Map

IV. Test Plans

a) Overview

This test plan describes the testing approach and overall framework of GEODestiny. The document is to be used as a guide for the testing activity.

The intended audience for this document: The Project Managers, Developers, and Test engineers.

b) Scope

The scope of testing is to test the operating characteristics of GEODestiny on all Android and iOS mobile platforms. The tests are organized by requirement which includes functional, data, performance, dependability, maintainability, and support requirements.

The functionality of this test plan spans through the entire system, making data, saved instances available whether connected to a network or offline.

The test plan does not address the following:
Testing requirements specific to the mobile device's hardware

Assessing the maturity level of the game compared to other equivalent games

c) Objectives

The objective of the test is to verify FULL functionality of GEODestiny works per the requirements.

The test will execute and verify the test scripts and identify defects. At the end of each testing, a Test Closure Report will be generated and reviewed by the project manager.

The end goal objective for this test plan is to have a near-fully running game with minimal defects.

d) Test Assumptions

Production like data be made available in the system prior to testing. This means that all character attributes, worlds, quests, etc. have already been generated.

In each testing phase, testers will report to the Lead testers. The lead tester will decide whether to move onto next phase or continue testing current phase.

e) Test Cases

e.1 Functional Testing

Purpose – The purpose of this test is to make sure backend operations are fully functional. This includes good connection latency, data being routed to their respective data tables, and making sure hardware defects are neutralized.

Scope – Testers will check the server status and make sure there are no hardware defects. In addition, run test scripts for connection latency and data entry.

Schedule – Daily maintenance will occur to check for faults.

e.2 Data Testing

Purpose – The purpose of data requirements is to make sure the necessary data is being routed to its respective data table. Testing will be carried out by creating test scenarios and test cases.

Scope - Testers will confirm that the test scenarios and test cases measures all inputs. Will keep track of the data and input.

Schedule - Data validation will occur during system maintenance. System maintenance will occur daily.

e.3 Performance Testing

Purpose – The purpose of performance requirements is to make sure the gameplay runs smoothly when millions of users are playing the game.

Scope – Have a simulation where hundreds of thousands of users are using the application simultaneously. Generate data reports where the suspension and resumption occurs during checkpoint.

Schedule – Happen at the end of the testing period.

e.4 Dependability, Maintainability, and Supportability Testing

Purpose – To test for error checking.

Scope – Confirm client and server have strong network connectivity. Display error messages and restore current game if the user has disconnected.

Schedule – Schedule will occur once at the beginning, middle, and end of each testing period.

f) Pass/Fail Criteria

Pass Criteria

- Test cases approved to start of testing period
- Development completed, unit tested with pass status
- Result shared Testing team to avoid duplicate entries
- Successful completion of unit testing for the application

- Dedicated resources are allocated to necessary hardware components
- Test environment fully functional

Fail Criteria

- If less than 95% of all test cases have passed.
- Game up-time should be at least 90%.

g) Roles and Responsibilities

Projector Manager

- Responsible for managing the whole testing process
- Approval of test documents
- Provide clear direction to the test team.

Lead Tester

- Requirement gathering
- Planning and forecasting for testing
- Reporting the tested project

Test Team

- Handles all the testing
- Reports to the Lead Tester

Test Engineer

- Creating the test cases for the test plan
- Documenting the results and fixing defects
- Executing the test cases

h) Project Tasks

Below are the test related tasks for testing during test phase.

Test Plan

- Identify Requirements for Test
- Asses Risk
- Develop Test Strategy

Design Test

- Identify and Describe Test Cases
- Identify and Structure Test Scripts
- Review and Access Test Cases/Scripts

Implement Test

- Setup Test Environment
- Program Test Scripts and Test Cases

Test Closure Report

- Evaluate Test Case
- Analyze Defects

- Create Test Evaluation Reports

Test Deliverables

Test No.	Deliverable Name	Contributor	Sign-off
1	Test Plan	Lead Tester	Project Manager
2	Design Test	Test Team	Lead Tester
3	Implement Test	Test Team	Test Engineer
4	Test Closure Report	Test Team	Projector Manager

V Project Issues

34 Open Issues

Our integration from console platform to mobile platforms is not yet complete. Our issue stems from our hiring process. The skillset we were looking for were developers from a console platform, instead of mobile platform.

Our considerations in the near future is to create a mobile department, whose sole focus is to develop GEODestiny on the mobile end.

35 Off-the-Shelf Solutions

35a Ready-Made Products

Currently there are no ready made products that we can inherit to make GEODestiny a mobile game, however, we are in the process of hiring a director that has sound knowledge and clear direction into our game.

Already existing code will not be usable since mobile gaming and console gaming use different platforms. We will be implementing a new architecture and database from scratch.

35b Reusable Components

Reusable components for the mobile side will be its code, functions, and nonfunctional requirements. Classes include:

- Character class
- World class
- Weapon class
- Armor class
- Helmet class
- Ability class

Reusable components for the mobile side will be its code, functions, and nonfunctional requirements. Classes include: We will be using 3rd party software where it will produce code from console base to mobile base. After the code has been made, our developers will be modified so it can be properly used in production.

35c Products That Can Be Copied

There are similar games out there in the mobile world. We have already examined our competitor's hardware for their games and we will be implementing nearly identical hardware. This can cut our analysis effort by 63%.

36 New Problems

36a Effects on the Current Environment

Content

Any changes to the software tools and training will affect the work of our developers. Our developers use very specific tools that need a lot of training. The training was done before hand so that the developers had ample time to learn. If new tools are implemented, then time and cost will be affected. New tools will be considered, but will not be used until after the release of our product.

We will NOT be implementing new tools for our developers in this current state.

36b Effects on the Installed Systems

Our current system can run optimally up to 1,000,000 users actively playing, however, our forecast shows that there will be a little over 1,000,000 users playing. The only bottleneck of having more than this many users online is our hardware.

We will be doubling our hardware so that more than 1,000,000 users can actively play at one time. Our goal is to be using this hardware for the next 5-10 years.

36c Potential User Problems

Currently, users are having issues with the in-store purchases. Sometimes when a user cancels a transaction, the user is charged even though he/she cancelled it. There has been some issues with the programs and we are implementing new measures to avoid this step. One idea is to use the in-game currency to purchase items instead of using their bank account.

Another issue was that when characters were battling, there were animation issues. For instance the character attacked twice instead of just once. This is a current bug that needs to be fixed as soon as possible.

36d Limitations in the Anticipated Implementation Environment That May Inhibit the New Product

The planned new hardware is not powerful enough to cope with current projection growth pattern.

The amount of work it produces is not enough to sustain over 1,000,000 users at a time.

36e Follow-Up Problems

There are a few concerning problems that we may not be able to cope with in the current time:

- Staffs commitment to the game
- Will staff turnover be low enough to allow continuity
- Will the management make decisions that reduce the development team's motivation

We are not concerned about the game itself, but rather the environment our staff is in. Concerns that are of great importance is how can we maintain our current staff and motivate them to create and maintain a great game?

37 Tasks

37a Project Planning

There will be four phases in the project planning.

1. Order Equipment
2. Implement planned project
3. Test/Inspect project
4. Create report

Below is a table of our project planning. This table will be continuously monitored and updated throughout the life of the project.

Planning	Importance Level Low/Medium/High	Description
Project Size		
Person Hours	**High**	Assigned Project Manager, engaged consultant, comprehensive project management approach and communications plan
Estimated Project Schedule	**High**	Create comprehensive project timeline using iceScrum
Team Size at Peak	**High**	Comprehensive communications plan, frequent meetings, tight project management oversight
Number of Interfaces to Existing Systems Affected	**High**	Develop interface control document immediately
Project Definition		

Planning	Importance Level Low/Medium/High	Description
Narrow Knowledge Level of Users	**Medium**	Assigned Project Manager(s) to assess global implications
Available documentation clouds establishment of baseline	**Medium**	Balance of information to be gathered by consultant
Project Scope Creep	**Low**	Scope intially defined in project plan, reviewed monthly by three groups (Project Manager and Steering Committee) to prevent undetected scope creep
Consultant Project Deliverables unclear	**Low**	Included in project plan, subject to amendment
Vendor Project Deliverables	**Medium**	Included in project plan, subject to amendment
Cost Estimates Unrealistic	**Low**	Included in project plan, subject to amendment as new details regarding project scope are revealed
Timeline Estimates Unrealistic	**Medium**	Timeline reviewed monthly by three groups (Project Manager and Steering Committee) to prevent undetected timeline departures

Planning	Importance Level Low/Medium/High	Description
Number of Team Members Unknowledgeable of Business	**Low**	Project Manager and consultant to identify knowledge gaps and provide training, as necessary
Project Staffing		
Project Team Availability	**Medium**	Continuous review of project momentum by all levels. Consultant to identify any impacts caused by unavailability. If necessary, increase committmment by participants to full time status
Physical Location of Team prevents effective management	**Medium**	Use of Intranet project website, comprehensive Communications Plan
Project Team's Shared Work Experience creates poor working relationship	**Medium**	Comprehensive Communications Plan
Weak User Participation on Project Team	**Low**	User Group Participants coordinated by full time employee
Project Management		
Procurement Methodology Used foreign to team	**Low**	N/A
Change Management Procedures undefined	**Low**	N/A
Quality Management Procedures unclear	**Low**	N/A

Planning	Importance Level Low/Medium/High	Description
User Training		
Number of times users will be trained	**High**	Users will be trained extensivly for each training. All trainings will be used when actually working
Team's Lack of Knowledge of Package	**Medium**	Comprehensive techniques and methods for users to get training in packages and libraries

37b Planning of the Development Phases

There are four phases to the planning of the development stage:

1. Order Equipment
2. Implement planned project
3. Test/Inspect project
4. Create report

Below will list the required date of when it will be implemented but also what components are included.

Phase 1 – Order Equipment

Required operation date: 12/5/2016

Components includes:

- 3xServer
- 3x1024GB RAM for the servers
- 300x1TB Storage space
- TitanX GPUs
- DirectX
- Flash development Tool
- XNA tools

Phase 2 – Implement planned project

Required operation date: 1/25/2017

Components includes:

- Java JDK for programming
- Implement classes
- Create documentations for each class created
- Store data in database

Phase 3 – Test/Inspect project

Required operation date: 3/25/2017 – 5/25/2017

Components includes:

- Scripts for test cases
- Analysis of each test case

Phase 4 – Test/Inspect project

Required operation date: 6/25/2017 – 7/25/2017

Components includes:

- A comprehensive report of the overall project.
- Forecasts
- Product analysis
- Risk assessment

38 Migration to the New Product

38a Requirements for Migration to the New Product

As of right now, there is no need to migrate our product to a new system or architecture. We will update this section when there are future migrations.

38b Data That Has to Be Modified or Translated for the New System

Currently not available.

39 Risks

Listed below are potential project risks.

- What is the degree of confidence in estimated size estimate?
- Number of projected changes to the requirements for the product, before and after delivery?
- Costs associated with new projects
- Government constraints
- How efficient is our maintenance period? Can we have the 99% uptime?
- Are tools and software made available to design our game?
- Are all the software tools integrated with one another?
- Are the best people available?
- Do the people have the right skills?
- Can colleagues get along with each other?
- Will turnover among staff be low enough to allow continuity?
- Meeting the overall communities expectation of the game
- Costs associated with late delivery
- Cancelled projects

40 Costs

The table below is a high level cost analysis of what purchases were made.

Category	**Details**	**Cost**
Lease Building	12,000 sq. feet building	$100,000
Hire developers	Salary, including benefits, recruitment costs, training	$1,500,000
Leasing equipment	Includes server, RAM, HD, software, database, workstations	$5,000,000
Additional miscellaneous costs	Food, conferences, training, etc.	$1,000,000

41 Waiting Room

The next release for development are listed below.

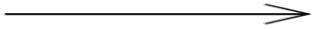
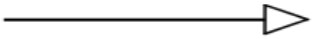
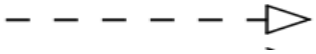
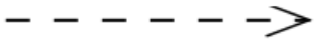
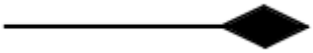
Functional Requirements:

- Allow users to see statistics and their characters without having to login to the game. Users can access a web portal see their characters and additional statistics.
- Another functional requirement that will be used will be a two level authentication where a user will need to have a auto key generator that will be linked to their phone. Whenever the user logins, he/she will need to enter the key along with their password.
- Improve transaction corrections, adjustments and cancellation algorithm.

Non-Functional Requirements:

- Increase scalability by purchasing new equipment.
- Create procedure to let users know about our maintenance period. We would like to show consistency and visibility to our clients.
- Create a detailed guideline about recycling. Our objective is to be environmentally friendly as possible.

VI Glossary

Class diagram	In software engineering, a class is a type of static structure diagram that describes the structure of a system by showing the system's classes
Use Case Diagram	Use case diagrams are used to describe a set of actions that some system or systems should or can perform in collaboration with one or more external users of the system.
RPG	Roll Playing Game
	Association
	Inheritance
	Realization / Implementation
	Dependency
	Aggregation
	Composition

VII References / Bibliography

[1] Robertson and Robertson, Mastering the Requirement Process.

[2] M. Fowler, UML Distilled, Third Edition, Boston: Pearson Education, 2004.

[3] J.Bell, “Software Engineering Project Report, A sample Document for Generating Consistent Professional Reports”, University of Illinois Chicago, Fall 2016.