First Generation

CS4961/CS4962 Senior Design

Software Design Specification

Prepared by:
Faculty Advisor(s):
Kang, Elaine

Team Leader:
Crisolo, Joseph
Team Member(s):
Vo, Ashley
Shahbazian, Alisha
Arroyo, Daniel
Barragan, Maximiliano
Park, Junghwan

Section 1: Introduction

1.1 *Purpose*

This document will outline the software architecture and design for our game called First Generation (FG). This document will provide several views of the system's design in order to facilitate communication and understanding of the product scope. It intends to capture and convey the significant architectural and design decisions that have been indicated for FG.

1.2 *Scope*

This document provides the architecture and design of FG. It will show how the design will accomplish the functional and non-functional requirements detailed in the FG game design document.

1.3 *Intended Audience*

This document is written on a technical level to address the CSULA College of Arts and Letters and Computer Science department.

1.4 *References*

First Generation (FG) Functional Requirements and Design (FRD) Document.

1.5 *Definitions*

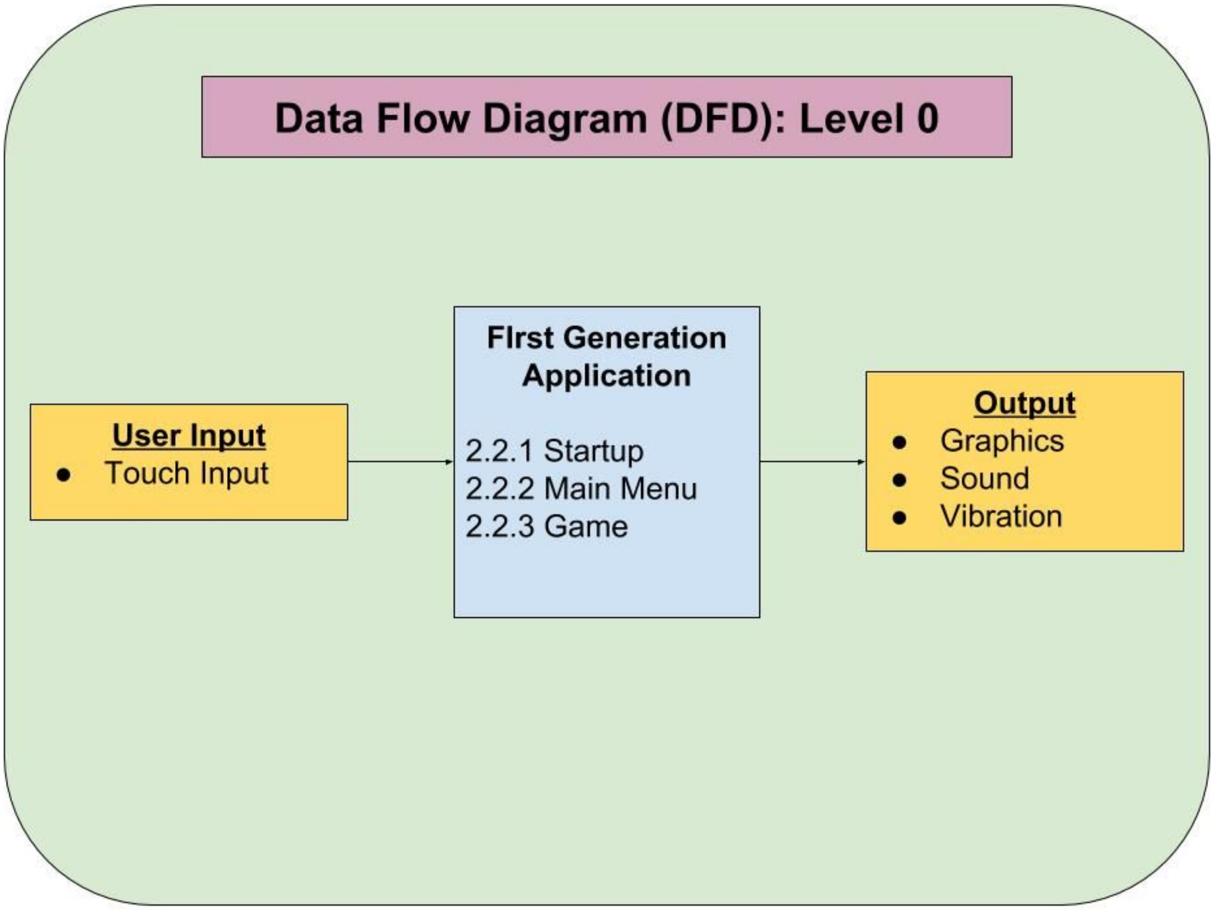
[SEE Section 8: Glossary]

Section 2: System Overview

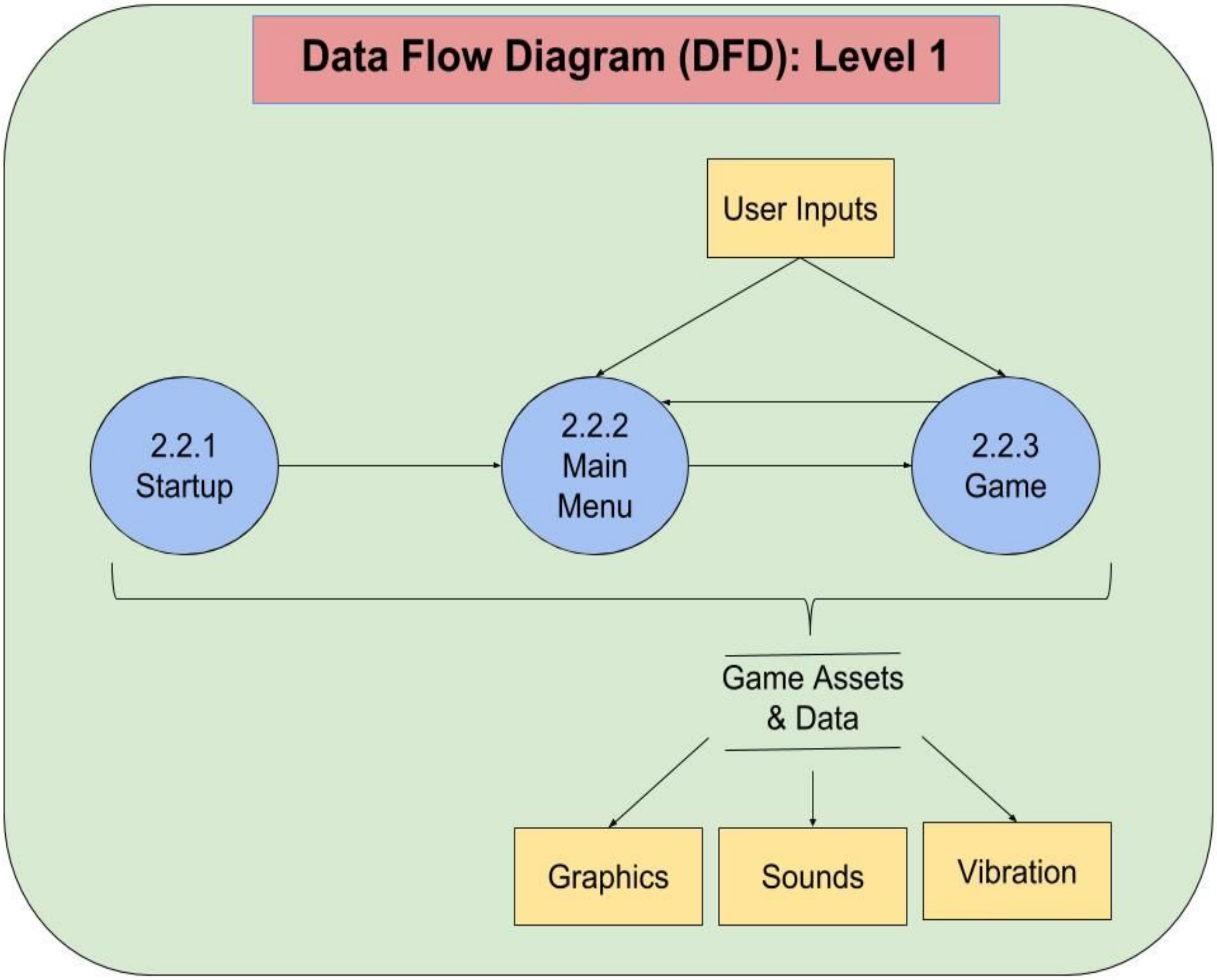
First Generation (FG) is built for the College of Arts and Letters to help simulate a college experience for high school students. FG is built for mobile devices.

The scope of the project is to make a mobile application for mobile platforms where the user can experience a year at college. Users of FG will be able to choose their own adventure. This adventure will have many options for the user to choose from. These options will change the story depending on what the user chooses. The player also have quest that can be completed to get more credits.

2.1 Context Diagram (level 0 Data Flow Diagram)



2.2 Context Diagram (level 1 Data Flow Diagram)



Section 3: Design Considerations

3.1 Goals

- The First Generation mobile game will primarily target users in high school where they often choose whether to continue or discontinue higher learning education and impact their decision making.
- The First Generation mobile game aims to allow users to experience aspects and challenges often experienced in College. This includes:
  - Managing time to complete tasks, commonly known as due dates in College.
  - Exploring and venturing into college with little to no guidance as usually experienced by First Generation students whose family have little to no education.
- The First Generation mobile game will allow users to replay the game to gain a better score illustrating the real difference of First Generation students versus other generation students coming from higher educated backgrounds. The inexperienced player will struggle to complete the tasks while players that play the game multiple times will

represent an incremental generation and will complete tasks much quicker due to their own knowledge that was acquired through playing one generation of the game.

- The First Generation mobile game will portray responsibility to users and relate their score to their actions taken and tasks completed.

3.2 Development Methods

The development method employed by this product is following the Iterative and Incremental Model. In order to collaborate with the different teams working on this project. The Iterative and Incremental model was the best solution for this project to provide feedback and implement changes. As the project requirements were often changing, the iterative and Incremental model allowed us to to build and test between iterations to flow with the direction of the game. The game engine, Unity3D, used to build the mobile game follows a component structure and our modules are implemented as components as well which allows us follow the Iterative and Incremental model to add features and requirements through different iterations and receive feedback as soon as the individual component is added without having to wait for other parts that might be unrelated to that specific component.

Section 4: Architectural Strategies

4.1 Use of a particular type of product (programming language, database, library, etc.)

The Unity3D software game engine was used to implement and program the video game. The programming language used to program the game is C#. The Maya animation and modeling software was used to create 3D models as assets for the game. Maya was also used to create various different animation assets for the game.

4.2 Reuse of existing software components to implement various parts/features of the system

The Unity3D software provides pre-made assets such as models and scripts that may be used in various aspects of game development. Some models include basic geometric shapes such as cubes, spheres, and cylinders, but there are also basic trees, people models, and other objects. Some pre-made scripts include different camera control options and player movement options. There are many more reusable components that the Unity software is able to provide for game developers.

4.3 Future plans for extending or enhancing the software

Future plans for FIRST GENERATION GAME are including but not limited to:

- Implement better environments by adding more 3d structures and the objects into the scene.
- Adding a multiplayer aspect to the game in order to to interact with other players online.
- Online activities like accessing into the twitter and the facebook to share.
- Adding the GPS system in our game to find the current location of the player and use it as part of the game objectives.
- High score information with a list of the player's highest scores.

4.4 User interface paradigms (or system input and output models)

The user interface includes the main menu UI. The main menu UI includes the "New Game" button which leads to an avatar selection screen. After selecting the player avatar the user will enter a name, and after entering a valid name the player can start the game. The main menu UI also includes the continue and options buttons. The continue button starts the game from where the player played the game last. The option button allows the user to select graphics qualities between low medium, and high. Options also allow the the user to adjust or toggle the music and sound effects. Options will allow the user to toggle in-game vibration functions. The main menu UI will provided buttons to go back to previous menu screens.

The main game takes touch input data from the user's touch screen and uses that to move the player avatar in the map. The point on the floor that is touched on the screen is the point that the player avatar moves. The main game takes touch input to active various buttons. One such button is the talk buttons above the non-playable characters (NPCs) in order to initiate conversations.

The in-game heads-up display (HUD) provides different functions for the user while playing the game. The user can select the minimap button on the HUD to display a wider view of the game map on their screen. The HUD also includes a cell phone button which allows the player to display the cell phone and its different functionalities. Some of the different functions and buttons that the cell phones includes, but is not limited to, accessing quest information, accessing game settings, and exiting the game.

4.5 Error detection and recovery

Data throughout the gameplay is going to be saved so any errors that may intrude gameplay will not affect saved data.

4.6 External databases

No external databases will be utilized.

4.7 Distributed data or control over a network

After downloading this application from the internet, this application will not need to be connected to a network as part of its primary game functionality.

4.8 Generalized approaches to control

All game data will only need to be downloaded once onto the mobile device.

4.9 Concurrency and synchronization

The application will only be synchronized with it's own internal game data and information.

4.10 Communication mechanisms

The game application will not need to communicate with any other network, device, application, or software.

Section 5: System Architecture

5.1 component-1 (subsystem-1) Mobile Device (AOS/IOS)

5.1.1 – The mobile device will be where the application will be installed. The mobile device will provide the user touch screen interaction with the software.

Section 6: Detailed System Design

6.1 Module-1 Startup (SM)

6.1.1 - The Startup Module (SM) is responsible for initializing all necessary game data when the software application is opened. The SM will check if there is previous saved game data; the saved game data will be loaded into the application if it exists.

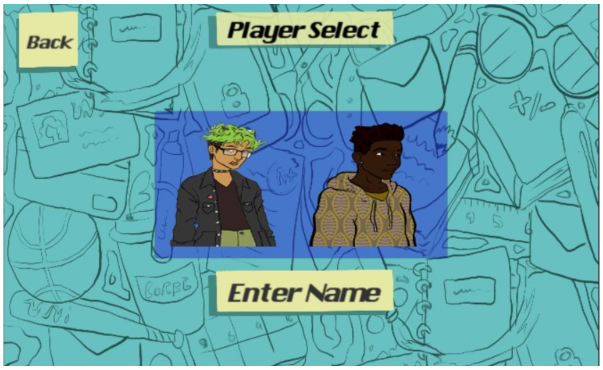
6.2 Module-2 Main Menu (MMM)

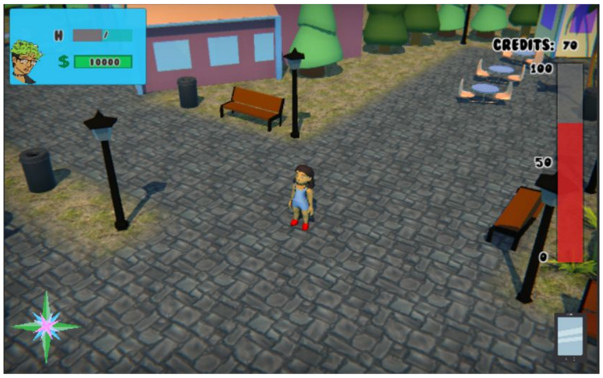
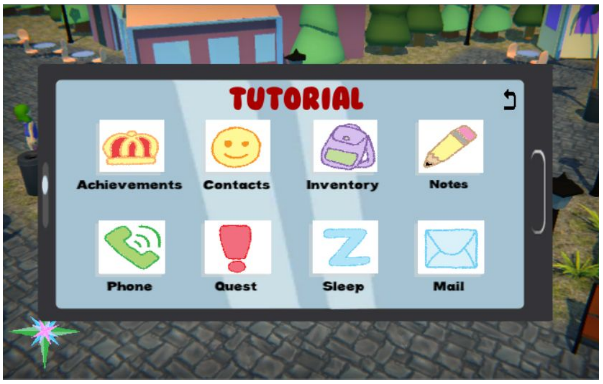
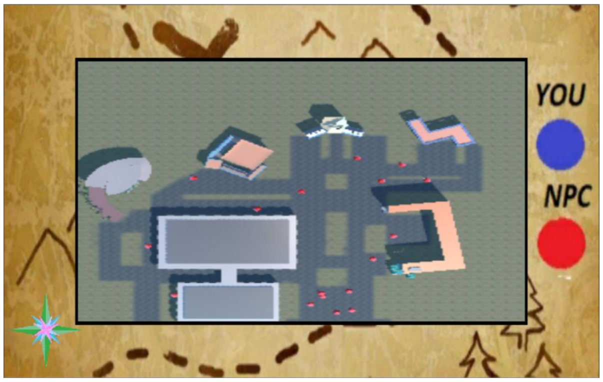
6.2.1 - The Main Menu Module (MMM) is responsible for loading and displaying the initial menu screens when the software application is opened. The (MMM) allows the user to select a player avatar and enter a name for the player avatar to load into the game. Additionally, the (MMM) is responsible for allowing the user to alter several different game settings.

6.3 Module-3 Game (GM)

6.3.1 - The Game Module (GM) is responsible for running the main game in an open world map including 3D models and a 2D HUD. The (GM) will be responsible for displaying, tracking, and modifying time, player, college, NPC, and quest statistics. The (GM) is also responsible for controlling dialogue interactions between the player and the NPCs. Additionally, the (GM) is responsible for allowing the user to alter several different game settings.

Section 7: Graphical User Interface Design

	The GUI was designed to be simple and bold.

	There was a desire to not clutter the screen during the main gameplay.
 TUTORIAL Achievements Contacts Inventory Notes Phone Quest Sleep Mail	The cell phone has big, simple buttons that are clearly marked for their respective functionalities.
	The compass button activates the Mini Map, which allows for an overhead view of the map. The player is represented by a blue ball and the NPCs are represented by red.

Section 8: Glossary

Abbreviation	**Definition**
2D	Two Dimensional

3D	Three Dimensional
AOS	Android Operating System
FG	First Generation
GUI	Graphical User Interface
HUD	Heads-Up Display
IAP	In-Application Purchase
IOS	iPhone Operating System
MT	Micro Transactions
NPC	Non-Playable Character
UI	User Interface