

# Unraveling History

## A walk through Chicago

**Group 24 – Ankur Agrawal, Ashwin Bansod, Sahib Bajaj, Sarthak Marathe, Uladzislau Tarsunou**

‘Unraveling History – A walk through Chicago’ is a **single player, strategy, virtual reality PC game** based on the historical events of the city of Chicago. The target audience is teenagers & gamers, aged fourteen and above, who are interested in learning about Chicago’s history. It is a mission based game inspired by different historical events in Chicago, spun across different centuries. The main objective of this game is to enhance the player’s awareness about the history of Chicago “by completing different missions based on real life events”. The game gives the user a feel of experiencing those historical events. Virtual Reality is supported through the Oculus Rift. The **Rift** is a virtual reality head-mounted display developed by Oculus VR.

### Game Description

Major and minor historical events are broken up into levels. At the start of each level, the user will be given a mission with a set of non-crucial sub-goals. The user is given the level’s objectives and time frame. For example, at one of the levels, the user finds himself in the 1871 at a small barn behind DeKoven Street, Chicago (where The Great Chicago Fire started). He faces the scenario of the **Great Chicago Fire** in this level. He is given the objective to **save a child who is trapped** at a particular location which will be shown to him throughout the level in the **map view**. This map view is displayed as a floating 2D GUI at the bottom left corner of game interface. The user has to overcome the **obstacles** in different parts of the city, such as Lake Area and Downtown to reach the location of the trapped child in the given time frame. If the player is unable to finish the level within the remaining time, he will **lose** a life and the level **restarts** from the last checkpoint. Upon losing all lives, the user must restart the entire level.

### Game Design

The game interface is very user friendly and minimalistic. On the main screen, users will get an option to create a profile. Although, it is not mandatory to create a profile, but if the profile is created, then the user can start the game from the last checkpoint. Alternatively, the user can go for ‘Quick Play’ mode, where the user is not required to create a profile, however the progress of the game will not be saved. There are 2 modes of game play.

- **Standard Mode** - In this mode, the user plays the game using the computer screen, keyboard and mouse. Game is played in first person with an overlaying 2D GUI.
- **Oculus Rift Mode** – In this mode, the user plays the game using a head mounted rift and controller. The view of the first person is controlled by Oculus Rift, which is synchronized with head movement of the user.

### Key Functional Requirements

- The game must provide the option to login from existing profiles and should retrieve the game state according to each profile.
- The game must allow the user to select the game mode i.e. Rift Mode or Normal mode.
- Each of the mission must involve interaction of the user with a historical building or other component, depicting history.
- There must be a map at the bottom left corner of the game to display the player’s location & destination.
- The game must be able to save the progress and continue the game from last saved checked point.

## Key Non-Functional Requirements

### Performance Requirements

- The game shall ensure high responsive time and consistent performance. It should get loaded in 3 seconds or less after the user open it.

### Reliability Requirements

- In case of momentary interruptions in the game, the content shall resume playing where it left off.

### Maintenance Requirements

- The game shall check for any patches or updates of the game from the web server.
- The game shall download updates from server (If any) and install it on the local machine.

### Supportability Requirements

- There will be no manual or live support provided for this application.
- The intuitive user interface of the game eliminates the need of manual user support for different functionalities such as switching from oculus rift mode to standard mode and vice-versa, etc.

## System Design

The software architectural pattern used for implementing the user interface in the game is Model-View-Controller based. The game comprises of 3-tiers

- **Presentation Tier** - This includes the graphical representation of the game.
- **Business Logic Tier** – It handles data validation, business rules and task-specific behavior.
- **Data Tier** – It is responsible for handling the user profile data and game states by interacting with the relevant APIs.

The reason behind following this architecture is better Separation of Concerns (SoC) and low coupling.

## Testing

The main objective of the game is to build a system which functions optimally and gives out the desired results. This can be accomplished through a series of test cases designed to test the functionality of our game. Several testing strategies should be carried out during the execution of the project such as Unit Testing, Integration testing, System testing, Acceptance testing & Usability testing. Unit testing should be performed for every module that is being developed. Integration testing must be performed soon after unit testing a module. Usability and Acceptance testing must be performed to validate that the product developed is in alignment with the requirements.

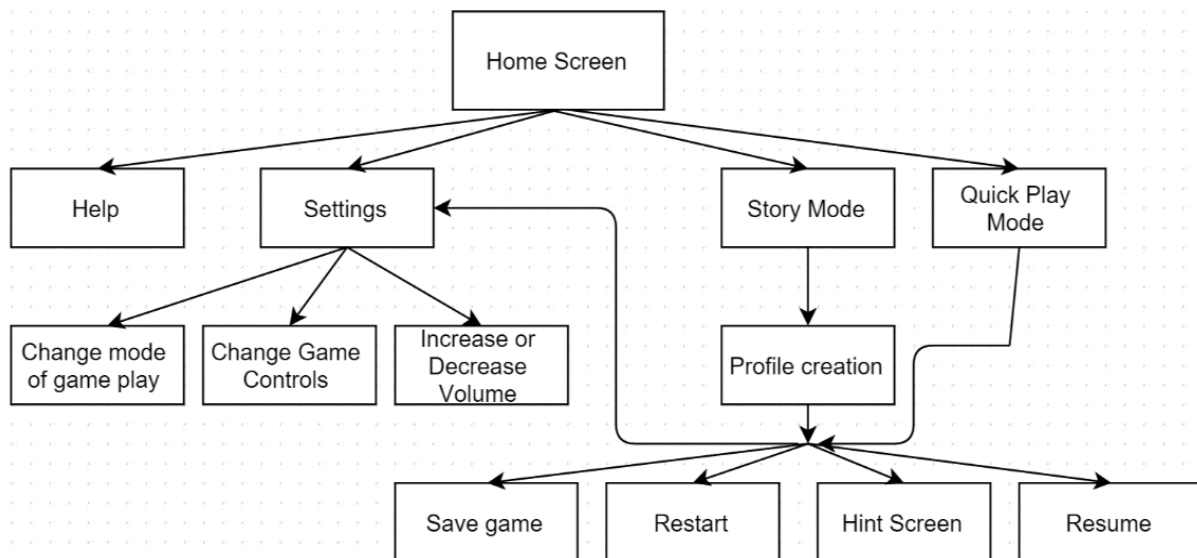


Figure 1: Navigation between different screens.