3.2.1 Product Requirements

3.2.1.1 User Interface Requirements

The user interface for the game is a critical aspect that can make or break the player's experience. To ensure the best possible experience, the interface should be easy to use and navigate, providing players with an intuitive way to interact with the game. Furthermore, it should be compatible with different devices and operating systems, allowing players to enjoy the game regardless of the platform they are using. These factors can greatly enhance the overall enjoyment of the game, creating a positive user experience that keeps players engaged and coming back for more.

In addition to outlining the necessary functions, it is important to describe the characteristics of each interface supported by the game's sketches. This provides valuable insight into how the user interacts with the game and how the game responds to user input. By analyzing these characteristics, we can identify areas for improvement and make adjustments to create a more seamless and enjoyable user experience. The main interfaces are:

Main Menu

The Main Menu interface of World Around will be the first screen that the player sees upon launching the game. The screen will have a dark-colored background image that evokes the adventurous theme of the game. At the top of the screen, the game name, "Worlds Around," will be displayed in large and bold white letters using a clear and easily readable font, such as Arcade Classic. The game logo, which may feature elements such as the world map or an adventure-themed icon, will be placed next to the game name on the left-hand side of the screen.

Below the game name and logo, three buttons will be displayed in a row - Play, Settings, and Quit. These buttons will be rectangular in shape with rounded corners and have a slightly raised appearance. The text on each button will be written in a clear and legible font, such as Arcade Classic, and will be presented in a large size to make it easy for players to read. The Play button will have a green color scheme with white text, the Settings button will have a gray color scheme with white text, and the Quit button will have a red color

scheme with white text.

Finally, on the far right of the screen, a small square button with an info logo will be displayed, which players can click to access information about the game, such as the developer, game version and the game about. This button will have a light blue color scheme with white text and a transparent background.

After clicking the Play button on the Main Menu interface, the player will be taken to a new screen that displays the profiles available in the game. This screen will feature a white background with a simple and minimalist design that emphasizes the profiles.

The profiles screen will have three rectangular boxes aligned horizontally, each representing a different profile. The profile number (either one, two, or three) will be displayed prominently on the top of each box using a clear and legible font, such as Arcade Classic.

On the bottom of each profile box, there will be a Delete button with a red background, which players can click to delete the corresponding profile. When the Delete button is clicked, a warning message will appear, asking the player to confirm if they really want to delete the profile. The warning message will be displayed in a prominent red color to draw the player's attention and ensure that they do not accidentally delete the profile. There will be two buttons here, yes or no, to confirm it or not.

When a player selects a profile, they will be taken to the last point where they left the game previously or they will continue the game from the beginning if this profile has never been used.

Pause

The pause tab is essential that it is visually appealing, intuitive, and consistent with the main menu and should have a similar background image as the main menu and buttons with similar characteristics. The buttons should be clearly labeled with text and have a border to distinguish them from the background. The font style should be chosen to complement the overall design, and it should be easy to read. A sans-serif font like Arcade Classic is a good choice, as it is simple and legible. The font size should be large enough to be easily readable.

The resume, restart, settings, and quit buttons should be aligned in a logical order and be

visually distinguishable. When the player clicks the resume button, the game should immediately resume from where it was paused. Clicking the restart button should reset the game to where he started. The settings button should take the player to the settings tab, and clicking the quit button should bring the player back to the main menu, and the game should be saved where it was left in the corresponding profile.

The color scheme for the pause menu should be consistent with the game's overall aesthetic. The buttons should be a different color from the background to make them stand out, but the contrast shouldn't be too strong to avoid eye strain. A light-colored button with a dark-colored text can work well. The text on the buttons should be descriptive and easy to understand, and the buttons should be large enough to be easy to click.

Finally, when the player clicks on a button, there should be a clear visual indication that the action has been registered. This could be a brief animation, a sound effect, or a color change of the button.

Settings

The Settings tab will have a similar background image to the Main Menu and Pause tabs, providing a consistent visual theme throughout the game. In the middle of the tab, there will be a large text header that reads "Settings" in a clear, easy-to-read font, with a font size that is large enough to be easily visible.

At the top left corner, there will be a back button with an arrow pointing to the left and the text "Back" below it. The button should have a different color from the other buttons on the tab to make it stand out and should be easy to click. When you click it, it will take you to the main menu if you clicked the settings button from the main menu tab or the pause tab if you clicked the settings button from the pause tab.

Below the Settings header, there will be the control buttons for the game. There will be four texts on the left side, one below the other, which read "Enter Door", "Left", "Down", and "Right". On the right side of each of these four texts, there will be four buttons that are "W", "A", "S" and "D" respectively, corresponding to the four texts on their left.

At the right side of these four buttons, there will be three texts one below the other, that are "Jump", "Power1", and "Power2". On the right of each of these three texts, there will be three buttons that are "Space", "1", and "2" respectively, corresponding to the three texts on their left. The buttons should have a clear, easy-to-read font and a font size that

is large enough to be easily visible.

When the user clicks on these buttons, they can change the controls to their preferences and the new control scheme will be saved. Below these texts and buttons in the middle, there will be a reset button that the user can click if they want to reset the controls to the default.

At the end of the tab, there will be two small sound images, one for the background music and the other for the game sound. These images will be one below each other and will have two straight lines and a point for each line that will be on these lines. This will show how much the volume is for the background music and the game sound. If it is at the beginning of the line, it tells that there will be no sound or music for the game, and there will be a red cross on the image to show that this sound is muted.

The colors of the buttons and text should be chosen to provide good contrast against the background image and should be visually appealing. The buttons should be large enough to be easily clicked and the text should be easy to read.

3.2.1.2 Learnability

If you are working on a platformer game, here are some learnability requirements to consider:

- **User Interface Design:** The user interface of the game should be designed to be intuitive and easy to navigate. All the game mechanics, menus, and settings should be accessible through the interface.
- **Consistency:** Consistency is key when it comes to learnability. The game mechanics, controls, and visual feedback should be consistent throughout the game. The player should be able to predict the outcome of their actions based on their experience from previous levels.
- **Onboarding:** The game should have a clear and concise onboarding process that introduces the player to the game mechanics and controls. It is best to introduce new mechanics one at a time, rather than overwhelming the player with too much information at once.
- **Feedback:** The game should provide clear and immediate feedback to the player. This can be done through visual or audio cues. The player should know what action they need to take and understand the consequences of their actions.
- **Difficulty Curve:** The game should have a gradual difficulty curve that increases as the player progresses through the levels. It is important to balance the challenge with the player's skill level to keep them engaged and motivated.
- **Playtesting:** Playtesting is an essential part of the game development process. You should regularly test the game with real users to get feedback on the game mechanics,

controls, and difficulty level. This feedback can be used to make adjustments to the game to improve its learnability.

3.2.1.3 Accessibility

When designing a platformer game , it is essential to consider accessibility requirements to ensure that the game is playable by a wider audience. Here are some accessibility requirements to consider:

- **Input Options:** Provide different input options for players, such as keyboard, mouse and gamepad. Ensure that the game can be played with a single hand, as some players may have a disability that affects one hand.
- **Customizable Controls:** Allow players to customize the game controls to their liking. This feature is essential for players with disabilities that may not be able to use the default controls.
- **Text Size and Font:** Provide an option to adjust the text size and font of the game. This feature is important for players with visual impairments.
- **Colorblind Mode**: Provide a colorblind mode that adjusts the colors of the game to make it easier for colorblind players to distinguish between different objects.
- **Sound Options:** Provide different sound options, such as subtitles or closed captions for players who are deaf or hard of hearing.
- **Difficulty Options:** Provide different difficulty options, such as easy or hard mode, to accommodate players with varying skill levels.
- **Clear Visual Feedback:** Ensure that all the game mechanics, controls, and visual feedback are easy to understand and distinguish.
- **Avoid Time-Based Challenges:** Avoid time-based challenges that may be difficult for players with disabilities that affect their reaction times.

3.2.1.4 Efficiency

Efficiency requirements for a platformer game are related to the performance of the game, its load times, and its ability to run smoothly on different hardware configurations. Here are some efficiency requirements to consider:

- **Optimization:** Optimize the game code and assets to reduce load times and improve performance. Use efficient algorithms, data structures, and rendering techniques to minimize the CPU and GPU usage.
- **Level Design:** Design the levels with efficiency in mind, avoiding large areas or overly complex structures that may affect the game's performance. Use level streaming or loading techniques to keep the memory usage under control.
- **Asset Management:** Manage the game assets efficiently, using compressed formats and reducing the number of textures or models to improve load times and reduce memory usage.
- **Platform Support:** Ensure that the game can run smoothly on different hardware configurations, including low-end and high-end systems. Optimize the game for different platforms, such as PC, console, or mobile, to ensure good performance on

each platform.

- **Memory Management:** Implement efficient memory management techniques, such as object pooling or garbage collection, to reduce memory fragmentation and improve performance.
- **Testing:** Test the game on different hardware configurations and use profiling tools to identify performance bottlenecks. Address any performance issues identified during testing to improve the game's efficiency.

3.2.1.5 Memorability

Memorability requirements for a platformer game are related to the ability of the player to remember how to play the game after a break or when returning to the game after some time. Here are some memorability requirements to consider:

- **Tutorial and Onboarding:** Include a tutorial and onboarding process that is easy to understand and remember. This will help the player remember the game mechanics and controls when returning to the game after a break.
- **Consistency:** Ensure that the game mechanics, controls, and visual feedback are consistent throughout the game. This will make it easier for the player to remember how to play the game when returning to it.
- **Clear Objective:** Provide a clear objective or goal for the player to remember. This will help the player understand what they need to do when returning to the game after a break.
- **Save Progress:** Include a save progress feature that allows the player to pick up where they left off. This will help the player remember where they were in the game and what they need to do next.
- **Reminders:** Include reminders or hints that help the player remember how to play the game. This could be in the form of pop-ups or visual cues that highlight important game mechanics or controls.
- **Feedback:** Provide clear and immediate feedback to the player. This will help the player remember the consequences of their actions and what they need to do next.

3.2.1.6 Errors

- The error rate is lower than the current error rate.
- If an error occurs it can be edited and corrected immediately.

3.2.1.7 Satisfaction

Satisfaction requirements for a platformer game are related to the player's emotional response to the game and their overall satisfaction with the game experience. Here are some satisfaction requirements to consider:

- **Engaging Gameplay:** Provide engaging gameplay that is challenging but not frustrating. This will keep the player motivated and interested in playing the game.
- **Interesting Storyline:** Include an interesting storyline or narrative that captures the player's attention and provides motivation to progress through the game.
- **Reward System:** Implement a reward system that incentivizes the player to complete objectives and progress through the game. This could include unlocking new levels, power-ups, or cosmetic items.
- **Replayability:** Provide replayability features, such as different difficulty modes or

randomized levels, to encourage the player to play the game again and again.

- **Visual Appeal:** Create visually appealing graphics and animations that enhance the player's immersion in the game world.
- **Audio Design:** Create an immersive audio design that includes sound effects and music that enhances the player's emotional response to the game.
- **Accessibility:** Ensure that the game is accessible to a wide range of players, including those with disabilities, to provide a more inclusive and satisfying gaming experience.

3.2.1.8 Capacity

Capacity requirements for a platformer game are related to the storage and processing resources required to develop and run the game. Here are some capacity requirements to consider:

- **Storage Capacity:** Estimate the amount of storage capacity required for game assets such as levels, characters, textures, sound effects, and music. Ensure that there is sufficient storage capacity available for the development and distribution of the game.
- **Processing Power**: Estimate the amount of processing power required to render the game graphics and animations, calculate physics, and handle game logic. Ensure that there is sufficient processing power available for the game to run smoothly and efficiently.
- **Cloud Infrastructure:** Consider using cloud infrastructure services such as Amazon Web Services or Microsoft Azure to scale up the capacity of the game. This can help reduce costs and improve scalability.
- **Hardware Compatibility:** Ensure that the game is compatible with a wide range of hardware configurations, including low-end and high-end systems. Optimize the game for different platforms, such as PC, console to ensure good performance on each platform.