\*Functional and Non-Functional Requirements\*

#### \*Functional Requirements (FR)\*

1. \*FR-1.1\*: The game shall display a tutorial when launched for the first time.

2. \*FR-1.2: After completing the tutorial, the game shall present two modes: \*\*Practice Mode\* and \*IQ Mode\*.

3. \*FR-1.3: In IQ Mode, the system shall require the player to log in using their \*\*username\* and \*age\* before starting the game.

\*FR-5\*: The game shall verify the player's age and allow players between the ages of 3 and 90 to proceed.

4. \*FR-1.4: The system shall allow the player to log in by clicking a \*\*Login\* button after entering their details.

5. \*FR-1.5: After login, the game shall display three stages in both modes: \*\*Easy, \*\*Intermediate, and \*\*Hard\*.

\*FR-1.7: In \*\*IQ Mode, the game shall include a \*\*countdown timer\* and a \*health slider\* during gameplay.

10. \*FR-10\*: The game shall allow players to track their IQ score based on their gameplay in IQ Mode.

6. \*FR-1.6: In each stage for the practice mode, the game shall have the following buttons: \*\*Reset, \*\*Next, \*\*Hint, \*\*Back, and \*\*Quit\*.

7.

8. \*FR-1.8: In \*\*Practice Mode\*, the game shall not display a countdown timer or health slider.

9. \*FR-1.9: The system shall allow the player to reset the current level using the \*\*Reset\* button.

10. \*FR-1.10: The system shall allow the player to quit the game or move to the next level using the \*\*Quit\* and \*Next\* buttons.

11. \*FR-1.11: The system shall provide hints during gameplay using the \*\*Hint\* button.

12. \*FR-1.12: The system shall return the player to the previous screen using the \*\*Back\* button.

#### \*Non-Functional Requirements (NFR)\*

1. \*NFR-1: The system shall provide \*\*quick response times\* for all user actions (login, button clicks, etc.),

2. \*NFR-2: The game shall be running on android mobiles

3. \*NFR-3: The tutorial and login process shall be designed to provide a \*\*user-friendly experience\*.

5. \*NFR-5: The game shall support a \*\*minimum frame rate\* of 30 FPS during gameplay to ensure smooth animation and button interactions.

6. \*NFR-6: The game’s user interface shall be consistent across different game modes.

8. \*NFR-8: The system shall allow \*\*real-time updates\* of the timer and health slider in IQ mode with no noticeable lag.

---

### \*Fully Dressed Use Cases\*

---

### \*Use Case 1: Launch Tutorial\*

- \*ID\*: UC-1

- \*Primary Actor\*: Player

- \*Goal\*: Display the tutorial on game launch for first-time players.

- \*Preconditions\*: The game has been opened for the first time by the player.

- \*Postconditions\*: The tutorial is completed, and the player is redirected to the main menu to choose the game mode.

- \*Trigger\*: The player opens the game for the first time.

#### \*Main Success Scenario\*:

1. The player launches the game.

2. The system checks if the player has completed the tutorial before.

3. the system displays the tutorial.

4. The player follows the steps of the tutorial.

5. After completion, the system redirects the player to the main menu (with the options for Practice Mode and IQ Mode).

#### \*Extensions\*:

- \*E1\*: If the player complete the tutorial before, they are taken directly to the main menu.

---

### \*Use Case 2: Login for IQ Mode\*

- \*ID\*: UC-2

- \*Primary Actor\*: Player

- \*Goal\*: Log in to start IQ Mode.

- \*Preconditions\*: The player is at the main menu and selects IQ Mode.

- \*Postconditions\*: The player is logged in, and their age is saved for IQ score calculations.

- \*Trigger\*: The player selects IQ Mode and is prompted to log in.

#### \*Main Success Scenario\*:

1. The player selects IQ Mode from the main menu.

2. The system displays a login form asking for the player's username and age.

3. The player enters their username and age.

4. The player clicks the "Login" button.

5. The system verifies the age.

6. Upon successful verification, the player is directed to the stage selection screen (Easy, Intermediate, Hard).

#### \*Extensions\*:

- \*E1\*: If the player enters invalid data (e.g., invalid age), the system displays an error message.

---

### \*Use Case 3: Select Stage (Practice or IQ Mode)\*

- \*ID\*: UC-3

- \*Primary Actor\*: Player

- \*Goal\*: Select a stage (Easy, Intermediate, or Hard) to start gameplay.

- \*Preconditions\*: The player has chosen a mode (Practice or IQ) and completed login if in IQ Mode.

- \*Postconditions\*: The player is taken to the selected stage for gameplay.

- \*Trigger\*: The player selects a stage from the stage selection screen.

#### \*Main Success Scenario\*:

1. The player selects either \*Practice Mode\* or \*IQ Mode\*.

2. The system displays three stage options: Easy, Intermediate, and Hard.

3. The player selects a stage.

4. The system loads the corresponding stage and prepares the game interface (with buttons and, in IQ Mode, the timer and slider).

#### \*Extensions\*:

- \*E1\*: If the player tries to start a stage without completing the login process in IQ Mode, the system redirects them back to the login screen.

---

### \*Use Case 4: Gameplay with Buttons (IQ Mode)\*

- \*ID\*: UC-4

- \*Primary Actor\*: Player

- \*Goal\*: Allow the player to interact with various buttons during gameplay in IQ Mode.

- \*Preconditions\*: The player is in a stage in IQ Mode, and the game is in progress.

- \*Postconditions\*: The player's action is carried out ( move to the next level, quit, etc.).

- \*Trigger\*: The player clicks one of the buttons ( Next , Quit).

#### \*Main Success Scenario\*:

1. The player starts a level in IQ Mode.

2. The player can click the following buttons:

- \*

- \*Next\*: Moves to the next level after completing the current one.

-

- \*

- \*Quit\*: Quits the game and returns to the main menu.

3. The system responds to the player’s action and updates the game accordingly.

#### \*Extensions\*:

- \*E1\*: If the player tries to quit, the system asks for confirmation before exiting the game.

---

### \*Use Case 5: Countdown Timer and Health Slider (IQ Mode)\*

- \*ID\*: UC-5

- \*Primary Actor\*: Player

- \*Goal\*: Implement a countdown timer and health slider in IQ Mode during gameplay.

- \*Preconditions\*: The player is in IQ Mode and has started a level.

- \*Postconditions\*: The game stops when the timer runs out, and the player sees a game-over screen if not completed within time.

- \*Trigger\*: The player starts a level in IQ Mode.

#### \*Main Success Scenario\*:

1. The player starts a level in IQ Mode.

2. The system starts the countdown timer.

3. The health slider decreases in sync with the timer.

4. The player completes the level within the time limit, and the timer is reset for the next level.

5. If the player completes all levels, the system calculates the final IQ score.

#### \*Extensions\*:

- \*E1\*: If the timer reaches zero, the system displays a IQ score on screen

#### Use Case 6: \*Start Game in Practice Mode\*

- \*Primary Actor\*: Player

- \*Preconditions\*: The player selects Practice Mode and a stage.

- \*Main Success Scenario\*:

1. The system starts the game without a timer or health slider.

2. The player interacts with buttons (Reset, Next, Hint, Back, Quit) as necessary.

3. The player completes the level.

- \*Postconditions\*: Player progresses to the next level in Practice Mode.

- \*Extensions\*:

- 2a. If the player presses the reset button, the game resets the level.tops gameplay.