**Use Case Diagram and Fully Dressed Use Cases for**

**DOST STARBOOKS: Whiz Challenge**

Project Documentation Submitted to the Faculty of the

School of Computing and Information Technologies

Asia Pacific College

In Partial Fulfillment of the Requirements for

Systems Analysis and Detailed Design

MSYADD1

Submitted by:

**Dumbrique, Kelly (Team Leader)**

**Gercan, Arcielle Marie (Member)**

**Quianzon, Shandrae Lois (Member)**

**Salipande, Janice Maxene (Member)**

G06 NEXUS – SF 231

Submitted to:

**Mr. Jose Eugenio L. Quesada**

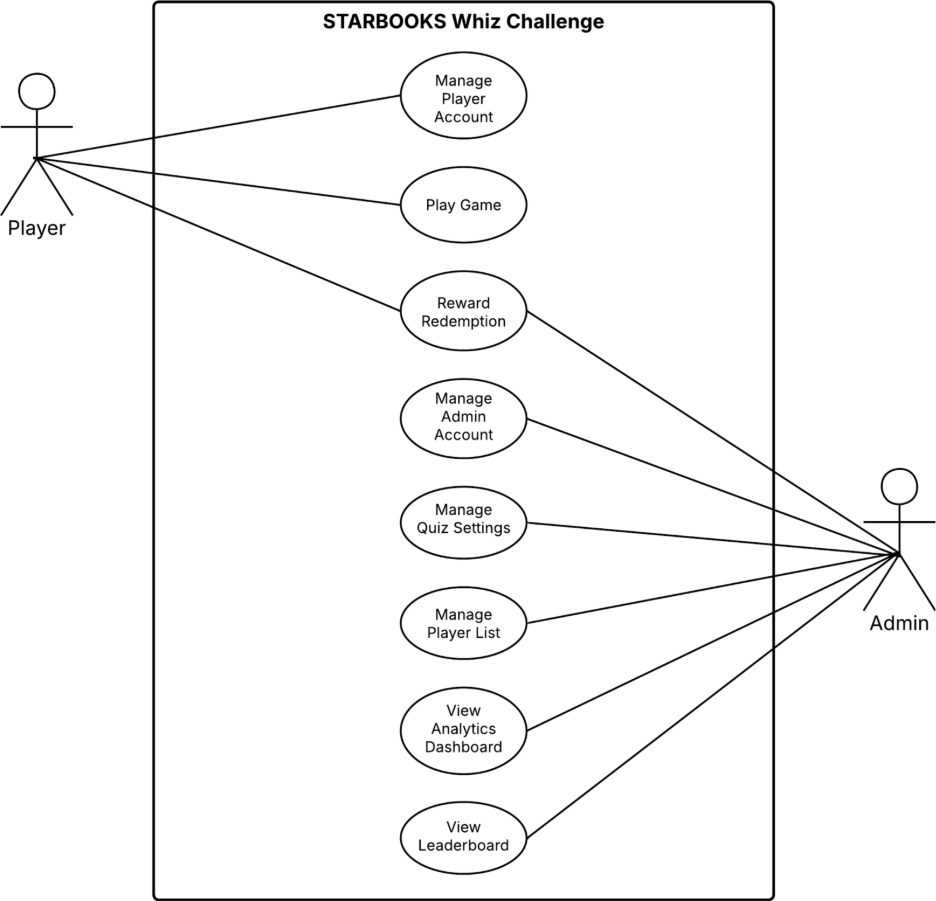
Course Instructor

August 2025

1st Term

A.Y. 2025 – 2026

**USE CASE DIAGRAM**



**FULLY DRESSED USE CASES**

|  |  |
| --- | --- |
| **Use Case ID** | UC-01 |
| **Use Case Name** | Manage Player Account |
| **Created By** | Shandrae Lois Quianzon |
| **Date Created** | September 11, 2025 |
| **Description** | This use case describes how a player manages their account, including registration, login, updating account information, changing password, tracking stats, and logging out. |
| **Priority** | High |
| **Primary Actor** | Player |
| **Secondary Actor** | System |
| **Preconditions** | 1. The player must have access to the system. 2. For login, the player must already be registered. |
| **Postconditions** | 1. Player account information is created, updated, or maintained. 2. The player is either logged in, logged out, or has updated account details. |
| **Main Flows** | **1. Register**  1.1. Player selects “Register here” from the login page.  1.2. System displays registration form.  1.3. Player enters required details such as username, password, name of school, age, category, sex, region, province, city, and chooses an avatar.  1.4. System validates the details.  1.5. System creates a new player account and confirms successful registration.  **2. Login** 2.1. System displays login form.  2.2. Player enters username and password. 2.3. System verifies credentials. 2.4. If valid, player is logged in and system displays the homepage.  **3. Edit Player Profile**  3.1. Player selects “Edit Profile”. 3.2. System displays editable fields. 3.3. Player updates information and submits. 3.4. System validates input and updates player profile. 3.5. System confirms successful update.  **4. Change Password**  4.1. Player selects “Change Password” from the Edit Profile page. 4.2. System prompts for current password and new password. 4.3. Player enters details. 4.4. System verifies current password and validates new password format. 4.5. System updates the password and confirms change.  **5. Track Stats**  5.1 Player selects “Stats” on the homepage.  5.2. System displays stats for every game mode to the 3layer.  **6. Logout**  6.1. Player clicks avatar and selects “Logout”.  6.2. System ends the session. 6.3. System redirects player to the login page. |
| **Alternative Flows** | **1.3. Invalid Registration Input** 1. Player enters incomplete or invalid details. 2. System displays error message and prompts correction.  **2.2a. Invalid Login** 1. Player enters incorrect username or password. 2. System rejects login attempt and displays error message.  **2.2b. Invalid Profile Update** 1. Player enters invalid or empty fields. 2. System displays validation error and requests correction.  **4.3a. Wrong Current Password**  1. Player enters incorrect current password. 2. System rejects request and asks to retry.  **4.3b. Weak New Password** 1. Player enters a password that does not meet security requirements. 2. System prompts the player to enter a stronger password.  **5.2. No Game Data**  1. System displays message: “No stats available.” |

|  |  |
| --- | --- |
| **Use Case ID** | UC-02 |
| **Use Case Name** | Play Game |
| **Created By** | Arcielle Marie Gercan |
| **Date Created** | September 11, 2025 |
| **Description** | This use case describes how a player selects and plays different game modes, including Whiz Challenge, Whiz Memory Match, Whiz Puzzle, and Whiz Battle. It also covers earning badges after completing conditions in Whiz Challenge. |
| **Priority** | High |
| **Primary Actor** | Player |
| **Secondary Actor** | System |
| **Preconditions** | 1. The player must be logged in. 2. The player must have a valid account. |
| **Postconditions** | 1. The player successfully plays the chosen game mode. 2. Player stats are updated accordingly. 3. Badges may be earned based on performance. |
| **Main Flows** | **1. Play Whiz Challenge** 1.1. Player selects “Whiz Challenge” on the homepage. 1.2. System displays categories (Science or Math).  1.3. Players chooses a category.  1.4. System displays difficulty levels (Easy, Average, Difficult).  1.5. Player chooses a difficulty level. 1.6. System loads a set of questions based on chosen category and difficulty level. 1.7. Player answers questions within the given timer.  **2. Play Whiz Memory Match** 2.1. Player selects “Whiz Memory Match” on the homepage.  2.2. System displays difficulty levels (Easy, Average, Difficult).  2.3. Player chooses a difficulty level.  2.4. System loads the game board with cards based on the chosen difficulty level.  2.5. Player flips cards to find matches.  2.6. System checks matches, updates score and timer. 2.7. When game ends, system displays results and updates stats.  **3. Play Whiz Puzzle** 3.1. Player selects “Whiz Puzzle” on the homepage.  3.2. System displays available difficulty levels (Easy, Average, Difficult), and categories (Solar System, Scientists, The Human Body, Animals, Geometry, Random).  3.3. System displays scrambled puzzle pieces based on the chosen difficulty level and category.  3.4. Player arranges pieces to form the correct image. 3.5. System validates arrangement and updates score. 3.6. System displays results and updates stats.  **4. Play Whiz Battle** 4.1. Player selects “Whiz Battle” on the homepage. 4.2. System displays options (Create Battle or Join Battle).  **4.2.1 Create Battle**  4.2.1.1 Player chooses “Create Battle” option. 4.2.1.2. System displays categories (Science or Math), and difficulty levels (Easy, Average, Difficult). 4.2.1.3. Player chooses a category and difficulty level. 4.2.1.4. System creates a battle room showing the game code. 4.2.1.5. Opponent joins the battle room.  **4.2.2 Join Battle**  4.2.2.1. Player enters the game code and clicks “Join.”  4.3. Player starts the battle. 4.4. System loads a set of questions based on chosen category and difficulty settings. 4.5. Players answer questions in real-time. 4.6. System calculates stars and determines the winner. 4.7. System updates both players’ stats. |
| **Alternative Flows** | **1.7a. Time Expired (Whiz Challenge)**  1. Timer runs out before player answers all questions. 2. System automatically submits current answers and calculates score.  **1.7b. Earn Badge (Whiz Challenge only)** 1. Player answers all questions correctly.  2. System awards a badge for that difficulty level. 3. System updates badge collection in player profile.  **1.7c. No Badge (Whiz Challenge only)**  1. Player fails to answer all questions correctly.  2. System does not award a badge for that difficulty level.  3. System displays results and records stats.  **1.7d. Player Exits Mid-Game (Whiz Challenge)**  1. Player exits before completing all questions.  2. System automatically submits current answers.  3. System calculates score based only on answered questions and updates stats.  **2.5. Player Exits Mid-Game (Whiz Memory Match)**  1. Player exits before timer ends.  2. System ends the game and displays results.  3. System updates stats.  **3.4. Player Exits Mid-Game (Whiz Puzzle)**  1. Player exits before puzzle is completed.  2. System ends the game and displays results.  3. System updates stats.  **4.2.1.5. Opponent Fails to Join (Whiz Battle)** 1. Opponent does not join. 2. System prompts player to wait longer or cancel battle.  **4.2.2.1. Invalid Code (Whiz Battle)** 1. Player enters invalid or expired battle code. 2. System displays error message: “Invalid code. Please try again.”  **4.5a. Opponent Disconnects (Whiz Battle)** 1. Opponent leaves mid-game. 2. System ends match and declares remaining player as winner. 3. Stats are updated accordingly.  **4.5b. Player Exits Mid-Game (Whiz Battle)**  1. Player quits during an active battle.  2. System ends the session and declares the opponent as the winner.  3. Stats are updated for both players accordingly. |

|  |  |
| --- | --- |
| **Use Case ID** | UC-03 |
| **Use Case Name** | Reward Redemption |
| **Created By** | Arcielle Marie Gercan |
| **Date Created** | September 11, 2025 |
| **Description** | This use case describes how a player redeems a reward after meeting badge requirements, and how an admin verifies and grants the reward. |
| **Priority** | High |
| **Primary Actor** | Player |
| **Secondary Actor** | Admin, System |
| **Preconditions** | 1. Player must be logged in. 2. Player must have earned all required badges in a difficulty level (3 badges in a difficulty level = eligible reward). 3. Admin must be logged in to process the reward claim. |
| **Postconditions** | 1. Player reward claim is recorded in the system. 2. Admin successfully grants or rejects the request. 3. Player’s profile is updated to reflect claimed rewards. |
| **Main Flows** | **1. Player Requests Reward** 1.1. Player selects “Your Badges.” 1.2. System displays badge collection and shows eligibility to claim a reward per difficulty level. 1.3. Player selects “Claim” on the chosen difficulty level. 1.4. System validates eligibility (e.g., 3 badges earned in one difficulty). 1.5. System records the request for admin processing.  **2. Admin Grants Reward** 2.1. Admin navigates to the “List of Players” in the admin panel. 2.2. Admin selects the player profile and clicks “View” in the badges column. 2.3. System displays the player’s badge collection with the “Reward” button unlocked (if eligible). 2.4. Admin clicks “Reward.” 2.5. System updates the badge collection of the chosen difficulty level, resetting it to no badges. 2.6. System logs the reward claim transaction. |
| **Alternative Flows** | **1.1a. No Badges Yet:**  1. Player opens “Your Badges” without having played any games.  2. System displays empty badge slots.  **1.2a. Claim Button Locked:**  1. Player sees “Locked” buttons instead of “Claim”.  2. Player cannot request to claim a reward without having met the requirement.  **2.3a. Reward Button Locked:**  1. Admin views a player’s badge collection but the “Reward” button is locked.  2. System indicates that the player has not met eligibility.  3. Admin cannot proceed until requirements are met.  **2.3b. Admin Cancels Reward Action:**  1. Admin views eligible reward but chooses not to proceed.  2. System retains the current badge collection without resetting. |

|  |  |
| --- | --- |
| **Use Case ID** | UC-04 |
| **Use Case Name** | Manage Admin Account |
| **Created By** | Kelly Dumbrique |
| **Date Created** | September 11, 2025 |
| **Description** | This use case describes how an admin logs in, logs out, and manages the list of administrators by adding new admins, editing admin profiles, or deleting admin accounts. |
| **Priority** | High |
| **Primary Actor** | Admin |
| **Secondary Actor** | System |
| **Preconditions** | 1. Admin must have access to the system. 2. For login, the admin must already be added to the admin list. |
| **Postconditions** | 1. Admin is logged in or logged out. 2. Admin list is updated (new admin added, existing admin edited, or deleted). |
| **Main Flows** | **1. Admin Login** 1.1. System displays the login form. 1.2. Admin enters username and password. 1.3. System verifies credentials. 1.4. If valid, admin is logged in and system displays the admin dashboard.  **2. Admin Logout** 2.1. Admin clicks the avatar and selects “Logout.”  2.2. System ends the session. 2.3. System redirects admin to the login page.  **3. Add New Admin** 3.1. Admin navigates to List of Admins. 3.2. Admin clicks “Add New Admin.” 3.3. System displays a form for admin details such as username, password, sex, and image.  3.4. Admin fills out details and submits. 3.5. System validates input and creates the new admin account. 3.6. System confirms successful addition.  **4. Edit Admin Profile** 4.1. Admin navigates to List of Admins. 4.2. Admin selects an existing admin and clicks Edit icon. 4.3. System displays editable fields. 4.4. Admin updates information and submits. 4.5. System validates and updates the profile. 4.6. System confirms successful update.  **5. Delete Admin** 5.1. Admin navigates to List of Admins. 5.2. Admin selects an admin and clicks Delete icon. 5.3. System prompts confirmation. 5.4. Admin confirms deletion. 5.5. System removes admin account and updates the list. |
| **Alternative Flows** | **1.2. Invalid Login**  1. Admin enters incorrect username or password.  2. System rejects login attempt and displays error message: “Invalid credentials. Please try again.”  **3.4a. Invalid Input (Add New Admin)**  1. Admin leaves required fields blank or enters invalid details.  2. System displays error message and requests correction.  **3.4b. Duplicate Username (Add New Admin)**  1. Admin enters a username that already exists.  2. System rejects submission and prompts to enter a unique username.  **3.4c. Invalid Profile Update (Edit Admin Profile)**  1. Admin enters invalid or empty fields during update.  2. System displays validation error and requests correction.  **5.2a. Cancel Delete (Delete Admin)**  1. Admin clicks Delete icon but cancels at the confirmation prompt.  2. System keeps the admin account unchanged.  **5.2b. Attempt to Delete Self (Delete Admin)**  1. Admin tries to delete their own account while logged in.  2. System rejects action with message: “You cannot delete your own account while logged in.” |

|  |  |
| --- | --- |
| **Use Case ID** | UC-05 |
| **Use Case Name** | Manage Quiz Settings |
| **Created By** | Kelly Dumbrique |
| **Date Created** | September 11, 2025 |
| **Description** | This use case describes how an admin logs in, logs out, and manages the list of administrators by adding new admins, editing admin profiles, or deleting admin accounts. |
| **Priority** | Medium |
| **Primary Actor** | Admin |
| **Secondary Actor** | System |
| **Preconditions** | 1. Admin must be logged into the system. 2. Question bank and difficulty settings must exist in the database. |
| **Postconditions** | 1. Question list is updated (new question added, existing question edited, or deleted). 2. Difficulty-level settings are updated (number of questions, stars, or timer adjusted). |
| **Main Flows** | **1. Update Question List** 1.1. Admin navigates to Questions. 1.2. System displays the current list of questions with search and filter options. 1.3. Admin selects one of the available actions: Add, Edit, or Delete.  **1.3.1. Add New Question** 1.3.1.1. Admin clicks “Add New Question.” 1.3.1.2. System displays a form for question details such as question text, options, correct answer, category, year level, difficulty, and image if there’s any. 1.3.1.3. Admin fills out the form and submits. 1.3.1.4. System validates input. 1.3.1.5. If valid, system saves the new question in the database. 1.3.1.6. System confirms successful addition and updates the question list.  **1.3.2. Edit Question Info** 1.3.2.1. Admin clicks Edit icon. 1.3.2.2. System displays editable fields. 1.3.2.3. Admin updates information and submits. 1.3.2.4. System validates and updates the question. 1.3.2.5. System confirms successful update.  **1.3.3. Delete Question** 1.3.3.1. Admin clicks Delete icon.  1.3.3.2. System prompts confirmation. 1.3.3.3. Admin confirms deletion. 1.3.3.4. System removes the question and updates the list.  **2. Update Difficulty Level Settings** 2.1. Admin navigates to Difficulty. 2.2. System displays current parameters for Easy, Average, and Difficult.  **2.3.1. Set Number of Questions** 2.3.1.1. Admin specifies the number of questions per difficulty. 2.3.1.2. System validates the number. 2.3.1.3. System updates the setting and confirms.  **2.3.2. Set Stars** 2.3.2.1. Admin specifies star values for correct answers per difficulty.  2.3.2.2. System updates and confirms the change.  **2.3.3. Set Timer** 2.3.3.1. Admin specifies the time limit per difficulty level. 2.3.3.2. System updates and confirms the change. |
| **Alternative Flows** | **1.3.1.4. Invalid Input (Add New Question)**  1. Admin leaves required fields blank or enters invalid data.  2. System displays error message and requests correction.  **1.3.2.4. Invalid Update (Edit Question Info)**  1. Admin provides invalid or incomplete information.  2. System displays validation error and requests correction.  **1.3.3.2. Cancel Delete (Delete Question)**  1. Admin clicks “Delete” but cancels at the confirmation prompt.  2. System keeps the question unchanged.  **2.3.1.2a. Insufficient Questions (Set Number of Questions)**  1. Admin specifies more questions than available in the database for that difficulty.  2. System displays warning: “Not enough questions available. Please reduce the number.”  **2.3.1.2b. Invalid Number of Questions (Set Number of Questions)**  1. Admin enters non-numeric or negative value.  2. System rejects input and prompts admin to enter a valid number.  **2.3.2.1. Invalid Star Value (Set Stars)**  1. Admin enters invalid star values (e.g., negative or zero).  2. System rejects and prompts admin to enter a valid value.  **2.3.3.1. Invalid Timer Value (Set Timer)**  1. Admin enters invalid timer values (e.g., negative, zero, or excessively high).  2. System rejects and prompts admin to input a valid time. |

|  |  |
| --- | --- |
| **Use Case ID** | UC-06 |
| **Use Case Name** | Manage Player List |
| **Created By** | Shandrae Lois Quianzon |
| **Date Created** | September 11, 2025 |
| **Description** | This use case describes how an admin manages the player list by adding new players, editing player profiles, or resetting player passwords when needed. |
| **Priority** | Medium |
| **Primary Actor** | Admin |
| **Secondary Actor** | System |
| **Preconditions** | 1. Admin must be logged into the system. 2. Player list must exist in the database. |
| **Postconditions** | 1. Player account is created, updated, or maintained. 2. Password may be reset if required. |
| **Main Flows** | **1. Add New Player** 1.1. Admin navigates to Player List. 1.2. System displays the current list of players with search and filter options. 1.3. Admin clicks “Add New Player.” 1.4. System displays a form for player details such as username, password, name of school, age, category, sex, region, province, city, and avatar. 1.5. Admin fills out the form and submits.  1.6. System validates input. 1.7. If valid, system saves the new player in the database. 1.8. System confirms successful addition and updates the player list.  **2. Edit Player Profile** 2.1. Admin navigates to Player List. 2.2. Admin selects an existing player and clicks the Edit icon.  2.3. System displays editable fields. 2.4. Admin updates information and submits. 2.5. System validates and updates the player profile. 2.6. System confirms successful update.  **3. Change Player Password** 3.1. Admin navigates to Player List. 3.2. Admin selects an existing player and clicks “Change Password.” 3.3. System prompts for new password. 3.4. Admin enters the new password and submits. 3.5. System validates the new password format. 3.6. System updates the password and confirms change.  **4. Delete Player** 4.1. Admin navigates to Player List. 4.2. Admin selects a player and clicks Delete icon. 4.3. System prompts confirmation. 4.4. Admin confirms deletion. 4.5. System removes player account and updates the list. |
| **Alternative Flows** | **1.6a. Invalid Input (Add New Player)**  1. Admin leaves required fields blank or enters invalid data.  2. System displays error message and requests correction.  **1.6b. Duplicate Username (Add New Player)**  1. Admin enters a username that already exists.  2. System rejects submission and prompts admin to enter a unique username.  **2.5. Invalid Update (Edit Player Profile)**  1. Admin enters invalid or incomplete data.  2. System displays validation error and requests correction.  **3.5. Weak Password (Change Player Password)**  1. Admin enters a password that does not meet security requirements.  2. System rejects and prompts admin to enter a stronger password.  **4.2. Cancel Delete (Delete Player)**  1. Admin clicks Delete icon but cancels at the confirmation prompt.  2. System keeps the player account unchanged. |

|  |  |
| --- | --- |
| **Use Case ID** | UC-07 |
| **Use Case Name** | View Analytics Dashboard |
| **Created By** | Janice Maxene Salipande |
| **Date Created** | September 11, 2025 |
| **Description** | This use case describes how the admin views and analyzes player and game-related data through the Analytics Dashboard. The dashboard provides insights on trends, player demographics, game performance, and reward claims. |
| **Priority** | High |
| **Primary Actor** | Admin |
| **Secondary Actor** | System |
| **Preconditions** | 1. Admin must be logged into the system. 2. Analytics data must be available in the database. |
| **Postconditions** | 1. Admin views and interprets analytics reports. 2. Data insights are displayed but no changes are made to the database. |
| **Main Flows** | * 1. Admin navigates to the Analytics Dashboard.   2. System displays dashboard sections, which may include:   * Total Registered Players * Most Played Game Mode * Average Session Duration * Male vs. Female Registered Players (pie chart) * Age Distribution of Players (bar chart) * Registered Players by Region (bar chart) * Male vs. Female Players Per Game Mode (bar chart) * Reward Distribution by Gender and Level (bar chart) * Most Played Game Mode by Age (bar chart)   3. Admin interacts with sort/filter options.  4. System updates and redisplays graphs based on selected filters.  5. Admin views insights and trends for decision-making.  6. Admin clicks Export icon.  7. System generates an Excel file with the selected analytics data and prompts the admin to download it. |
| **Alternative Flows** | **2A. No Data Available**  1. If there is no available data (e.g., new system with no players yet),  2. System displays an empty state message such as “No analytics data available yet.”  **4A. Invalid Filter Selection**  1. Admin applies a filter combination that returns no results (e.g., selecting a date range before launch).  2. System displays “No results found for the selected criteria.”  **4B. System Error During Data Load**  1. If the dashboard fails to retrieve analytics data due to a system error,  2. System displays an error message and prompts the admin to retry.  **6A. Export Fails**  1. If the system encounters an error during Excel export (e.g., file generation issue),  2. System displays “Export failed. Please try again.” |

|  |  |
| --- | --- |
| **Use Case ID** | UC-08 |
| **Use Case Name** | View Leaderboard |
| **Created By** | Janice Maxene Salipande |
| **Date Created** | September 11, 2025 |
| **Description** | This use case describes how an admin views the leaderboard for Whiz Challenge and Whiz Battle. |
| **Priority** | High |
| **Primary Actor** | Admin |
| **Secondary Actor** | System |
| **Preconditions** | 1. Admin must be logged in. 2. At least one player must have played a game to generate leaderboard data. |
| **Postconditions** | 1. Leaderboard data is displayed based on selected filters. 2. Admin may export the leaderboard data to Excel. |
| **Main Flows** | 1. Admin navigates to the Leaderboard page.  2. System displays two leaderboard sections:  3. Whiz Challenge Leaderboard (Top 10)   * Username * Total Rewards for All Levels * Rewards for Easy, Average, Difficult * Last Reward Claim date/time   4. Whiz Battle Leaderboard (Top 10)   * Username * Total Stars * Last Battle date/time   5. Admin applies sort and filter options (e.g., by difficulty, date range, gender).  6. System updates and redisplays the leaderboard based on selected filters.  7. Admin reviews leaderboard standings.  8. Admin clicks Export to Excel option.  9. System generates an Excel file containing leaderboard data and prompts the admin to download it. |
| **Alternative Flows** | **2A. No Leaderboard Data**  1. If no players have participated yet in either game mode, system displays “No leaderboard data available yet.”  **4A. Invalid Filter Selection**  1. Admin applies filter criteria that return no matching players.  2. System displays “No results found for the selected filters.”  **4B. System Error While Loading Leaderboard**  1. If the system fails to fetch leaderboard data due to a technical issue,  2. System displays an error message and prompts the admin to retry.  **6A. Export Fails**  1. If the system encounters an error during Excel export,  2. System displays “Export failed. Please try again.” |