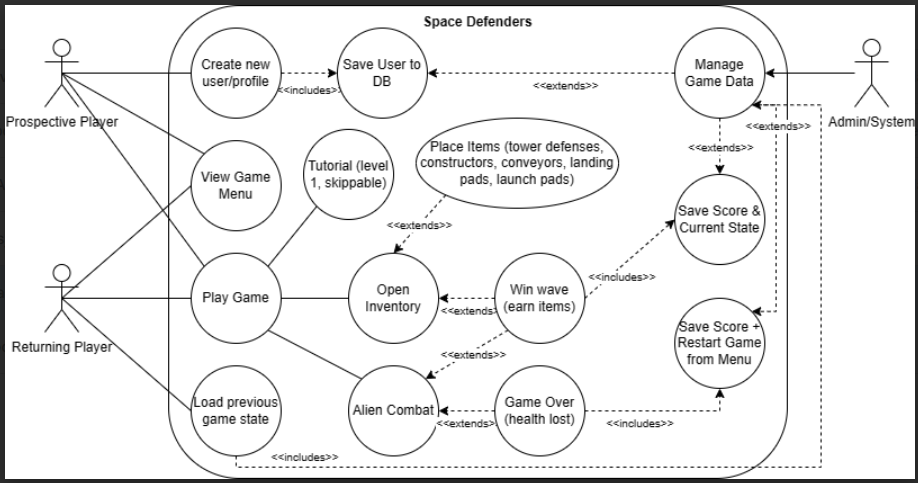
# Use Case Document

——————————————————————————————————————————



*Fig. 1. The use case diagram, provided for reference*

**The Actors**

* Prospective Player: a player who has never played the game before or would like to start from an unsaved data slot
* Returning Player: a player who would like to play the game using a previously saved game data slot
* Admin/System: our coded system manages the players, their scores, and their game data

**The Use Cases**

* Viewing the game menu
  + **Preconditions**: (first thing) a game has not been initiated
  + **Actors**: Both players
  + **Flows**: Player sees menu from starting up game
* Creating a new user
  + **Preconditions**: this new user must not previously exist (unique username in DB)
  + **Actors**: Prospective player
  + **Flows**: Player chooses this option from the menu; can return to main menu options
* Loading a previous game
  + **Preconditions**: a saved game file must exist
  + **Actors**: Returning player, Admin/System
  + **Flows**: Player chooses this option from the menu; can return to main menu options
* Playing the game
  + **Preconditions**: new user created or a game has been loaded from level 1
  + **Actors**: Both players
  + **Flows**: Player chooses this option from the menu, plays non-scored version to teach basic functionality
* Tutorial
  + **Preconditions**: a user has been created or a game has been loaded
  + **Actors**: Both players
  + **Flows**: Player chooses this option from the menu
* Open Inventory
  + **Preconditions**: a game must be started
  + **Actors**: Both players
  + **Flows**: Player started a game, can open inventory
* Alien Combat
  + **Preconditions**: a game must be started
  + **Actors**: Prospective player, Returning player
  + **Flows**: Player started a game, after making adjustments to defenses and conveyors, combat begins
* Manage Game Data
  + **Preconditions**: a new player has been created, a game has been loaded, **or** a wave of combat has ended
  + **Actors**: Admin/System
  + **Flows**: After a menu options to load or create are chosen, DB is accessed or new information is added; Once combat ends, current state & score are saved if they won **or** current score saved and game is sent back to menu

**Related Use Cases** (the extends and includes)

* Saving user to DB
  + **Preconditions**: Creating a new user
  + **Actors**: Prospective player, Admin/System
  + Flows: Information is sent to be updated in database, begins game
* Placing Items
  + **Preconditions**: opening your inventory
  + **Actors**: Both players
  + **Flows**: Items placed, allows user to place new items or start wave
* Winning combat wave
  + **Preconditions**: combat initiated
  + **Actors**: Both players
  + **Flows**: proceeds to saving score and current game state
* Game over
  + **Preconditions**: combat initiated
  + **Actors**: Both players
  + **Flows**: proceeds to saving score + restarting
* Saving score & game state
  + **Preconditions**: combat wave won
  + **Actors**: Both players, Admin/System
  + **Flows**: proceed to next level with details of next level (story) and option to place items and open inventory
* Saving score + restarting game
  + **Preconditions**: Game lost
  + **Actors**: Both players, Admin/System
  + **Flows**: return to main menu