Double Tap Game Launcher

# Functional Requirements

## Stakeholders

A stakeholder is anyone who has interest in this system (users, managers, sponsors, etc.).

1. Registered Customer
2. Unregistered Customer
3. System Admin
4. Double Tap Employee

## Actors and Goals

An actor is anyone who directly interacts with the system

1. Initiating
   1. Registered Customer
   2. Unregistered Customer
   3. System Admin
   4. Double Tap Employee
   5. Timer
2. Participating
   1. Host PC
   2. Central Server
   3. Mobile Display
   4. Event Log
   5. Host PC Database

## Use Cases

### UC-1 View Game Library

A registered or non-registered customer wants to view what games are available to play on the host PC. A user will request to view all games. The system will then gather all the game data from the Host PC Database and return the data to the mobile display for the user to view.

### UC-2 Categorize Games

A customer wants to sort the games to display based on category. They will select all the category types desired in the UI and the system will show games that fit only those categories. If the customer is a registered user, they may be able to sort by favorited games

### UC-3 View Game Details

A customer has found a game they want to play and want to know more about it. They select the game on the mobile display which will pull up a new view showing all the games info. The system will gather all the data about the game and display.

Available User Actions will include:

1. Launch Game
2. Add To Queue
3. Favorite
4. View Gameplay Video
5. Return to Library

### UC-4 Launch Game

* If the mobile device is connected to a host PC:
  + Send signal to host to Launch Game
* If mobile device is disconnected:
  + Call Central Server to Check Host PC Database
  + Display available host PC connections
  + Connect to host PC
  + Launch Game

### UC-5 Add to Queue

If the customer wants to add a game to the queue, the mobile display device will store the launch command locally. The customer will be able to view the queue later and decide when the next game in queue or remove games from queue.

### UC-6 Favorite Game

FEATURE ONLY AVAILABLE FOR REGISTERED USERS. A registered customer wants to save a game so they can find it easily the next time they come in or if they visit another location. Favorited games will be able to be sorted in category search. Favorited games will be stored in registered customers database.

### UC-7 View Gameplay Video

A customer has viewed all the data about the game they have selected, but now they want to watch some gameplay of the game. A video will be displayed showing gameplay footage of the game.

### UC-8 Return to Library

A customer would like to return to the library to view all the games again. Once the button is selected, the application will return to View Game Library screen and reload all games to be viewed.

### UC-9 Register

An unregistered customer wants to register a new account. The system will show a registration form, which the user will fill out and submit. The system will the validate the submitted info (email and password) and store it in the database, assigning the user a unique customer id.

### UC-10 Authenticate User

A registered customer wants to log in to the system. The system will present a log in form (email, password) which the customer will fill out and submit. The system will search for and find the customer in the database. The system will then start a session for the user, which will last until the user logs out or closes his/her lane session.

## Use Case Diagram

## Implementation User Stories

1. Gather Game Data
   1. As a developer I want to be able to access the game data on a host pc to display to the user later. To achieve this, we want to have a script that goes through the game library and writes all the game data into a JSON file.
   2. JSON file needs to hold name, description, categories, gameplay photo file paths, game location, gameplay video location.
2. Store Host PC data in central server database.
   1. As a developer I want to be able to access the host PC data from a mobile display device. To achieve this, whenever we connect a new host PC to the central server, we want to save the computers data (IP, port, name) to a database for access later.
3. Launch Game Function
   1. As a user I want to be able to launch a game when given the name of the game. Using the name, the program will access a JSON file to find the game executable location and launch the game.
4. Setup Server side of wireless connection
5. Transmit game data to mobile device
   1. As a developer I want to be able to display all the game data on the mobile device including game image.
   2. This will be done using packets and server-client programming
6. Handle Game Launch Command from mobile device
7. Develop Local Queue