# TechTime’s What to Play

|  |
| --- |
| Table of Contents |
| Section 1: Description  1a: Intended Users and Purpose  1b: Description of Documentation |
| Section 2: Java file descriptions  2a: Controller.java  2b: Main.java  2c: adminController.java  2d: User.java  2e: Game.java |
| Section 3: System Features and Requirements  3a: System Features  3b: Interface Requirements  3c: Admin-side Requirements  3d: Nonfunctional Requirements |

1. Description
   1. This application was created for users to search or browse an extensive catalog of games in order to find something new to play. It allows the user to select which games they like to add to their favorites list after logging in, as well as search for specific types of games or a random one determined by the application.
   2. This Documentation will give brief descriptions of the various associated .java files and what they do. All included .fxml files are what the .java files use to build the different screens that the user sees, and all code within them is for aesthetic purposes. All Features are listed nearer the end of this documentation.
2. Java file descriptions (What each part of the working code does):
   1. Controller.java:
      1. Loads the various screens the application uses, as well as builds account information based on inputs given from the user
      2. Handles Wheel of Random Game feature by spinning wheel and communicating with Random.fxml to find a random game from the database to return to the user, which they can then claim
      3. Manages click events to open a game’s page or favorite a game
      4. Handles user Updating Account Information (changing password, email, etc.) by altering the Users data within the userDatabase.txt file
      5. Runs the changes the User makes to their favorites list by adding or removing them
   2. Main.java:
      1. Launches the application using the arguments
      2. Sets the current User empty for login purposes
      3. Determines the location on screen that the application will occupy
   3. adminController.java:
      1. Used by Admins to build a Game object that is then stored in the gameDataBase.csv file
   4. User.java:
      1. Creates the User objects stored into the userDatabase.txt file, using data given by the user when creating an account
      2. Takes the user’s first and last name, their unique username and password, and age to build the account information
      3. Associates that account with a Hashmap containing that specific user’s favorites list
   5. Game.java:
      1. Holds the constructor used to build the Game objects added by adminController.java
3. System Features and Requirements
   1. System Features
      1. User can create an account
         1. Using unique username to add to userDatabase
         2. Saves username to a specific list in the Favorites map
      2. User can log in to the application
         1. Allows user to add chosen games to their personal Favorites list
      3. User can use the Random Game Wheel
         1. Gives User a random game not on their Favorites list
         2. Lets User find new games they may not have looked at otherwise once per day
      4. User can Search for games
         1. Find specific games they know they will enjoy
         2. Research into various games they may not be as knowledgeable on
      5. User can submit a suggestion for a game to be added to the database by an Admin
      6. User can change/update personal information
   2. Interface Requirements
      1. Easy to follow
         1. Screen elements are relevant
         2. No unnecessary buttons on screen
      2. Uniform design throughout
         1. Aesthetically pleasing
         2. Makes the user comfortable and want to return
   3. Admin-side Requirements
      1. Allow additions to database
      2. Edit database content
   4. Nonfunctional Requirements
      1. Keeps User information private
      2. Games are easy to find/search for, relevant tags applied
      3. Keeps usernames unique between users