**SPECIFICATION OF DOCUMENT**

**Product name** : rentmycam.io

**Product Description**: rentmycam.io is a camera rental application that allows users to browse and rent cameras for a psecific duration.Users can view list of available cameras,including their brand, model, and per-day rental amount. The application also provides the functionality to add a new camera to the list .Users can check their wallet balance, deposit money into their wallet, and rent a camera if they have sufficient funds.The application implements various Java concepts to optimize performance.

**Product Apperance:**

The application will have a user-friemdly interface with different screens for each feature. The screens will be designed to provide smooth user experience and intuitive navigation making it easy to use.

**User Interactions:**

1. Welcome Screen:

* Display the application name and ask for Login Credentials.
* Provide options for the user to access various features of the application.

1. Camera Listing:

* Display the list of available cameras along with their model, brand and perday rental amount.
* Allow the user to select a camera to rent.

1. Renting a Camera:

* Check if the User has sufficient funds in their Wallet.
* If the balance is sufficient, deduct the perday rental amount from wallet and allow user to rent the camera.
* If balance is not sufficient, display a message stating that balance is insufficient.

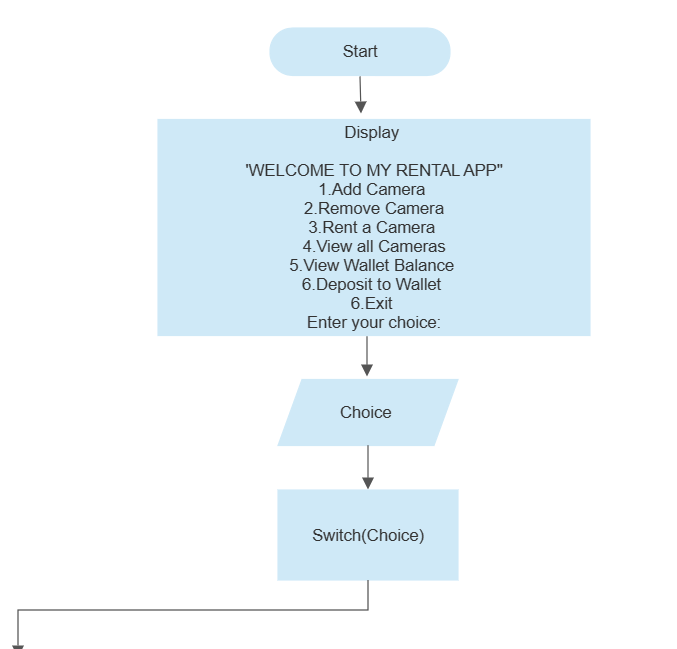
1. Wallet Manangement:

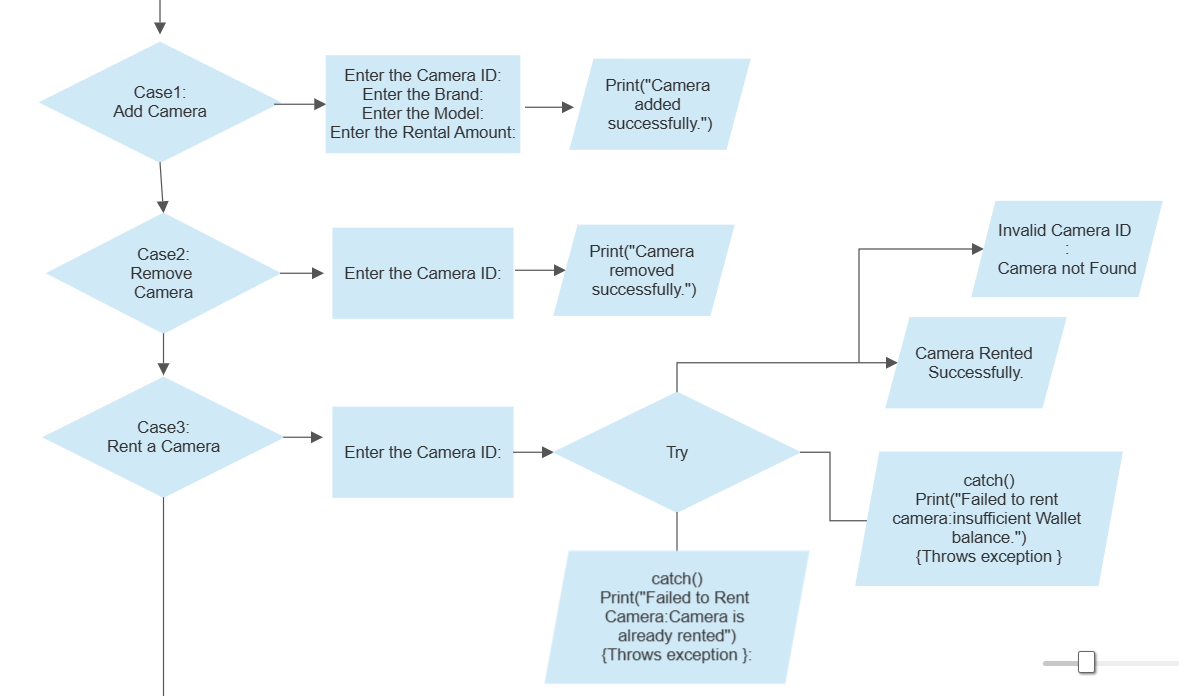
* Provide option to add or view the user’s wallet.
* Allow the user to deposit money in their wallet
* Display the updated wallet amount.

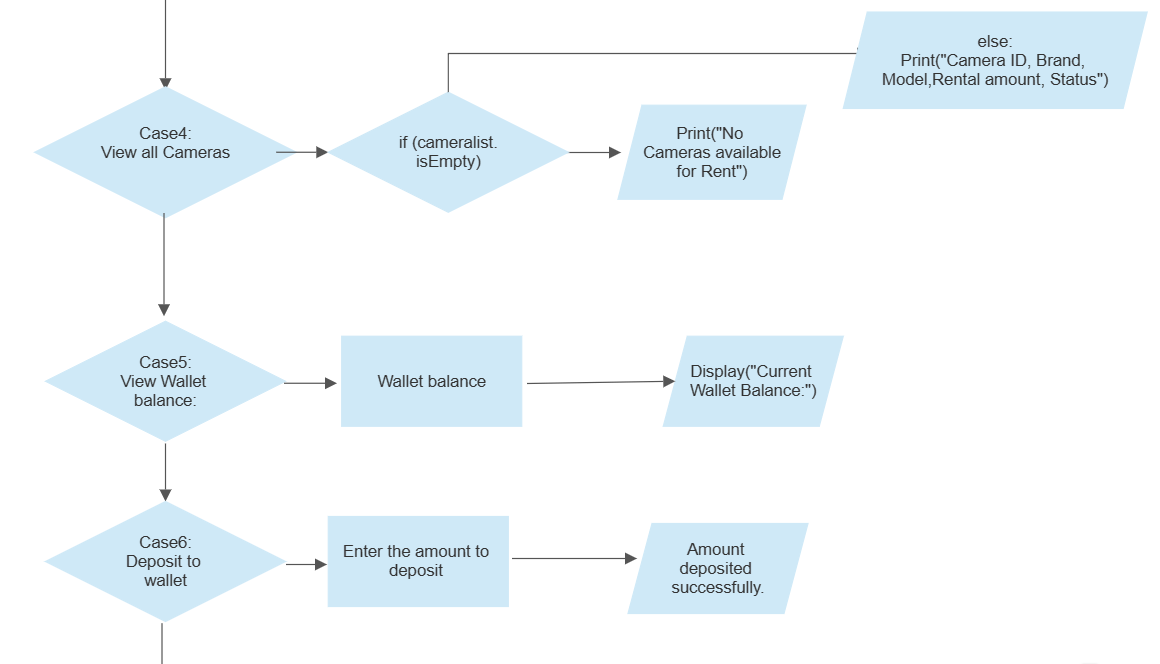
1. Application Navigation:

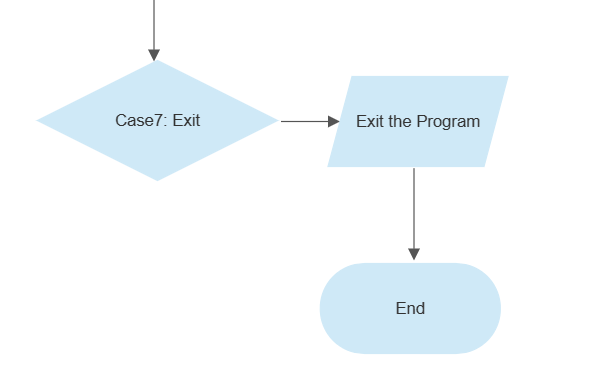
* Allow the user to navigate from current execution context to main context.
* Provide an option to close application .

**FLOWCHART**

****

****

****

****

**SPRINT**

Sprint1:

1. Task:Setup and Basic Functionality

* Set up the project structure and version control(Git and Github)
* Create the welcome screen with the application name and User details.
* Implement the data structure for storing camera information.
* Implement the operation to list cameras along their brand, model, and per-dar rental amount.

1. Task: Renting functionality

* Implement the operation to select froma listing to rent.
* Implement the operation to check and deduct the rental amountfrom User’s wallet balance.
* Implement exception handling for scenarios such as insufficient wallet balance.

Sprint 2:

1. Task: User wallet management

* Implement the operation to add or view the user’s wallet balance.
* Implement the operation to deposit an amount to increase the wallet balance.
* Implement error handling for invalid input during wallet management.

1. Task:Application flow and navigation

* Document the application flow and user interactions.
* Implement navigation from the current execution context to main context.
* Implement the operation to exit the program.

Sprint 3:

1. Task Testing ,Documentation and Finalizatioin

* Perform thorough testing of the application to identify and fix any bugs or issues.
* Review and update the documentation, including the specification document.
* Ensure the code follows coding conventions and best practices.
* Perform a final round of testing and bug fixes.
* Prepare the application for deployment and release.