

StellarAR

Aditya, Arsalan, Haoze, Abhiram

Group: fall2024-g20

Table of Contents

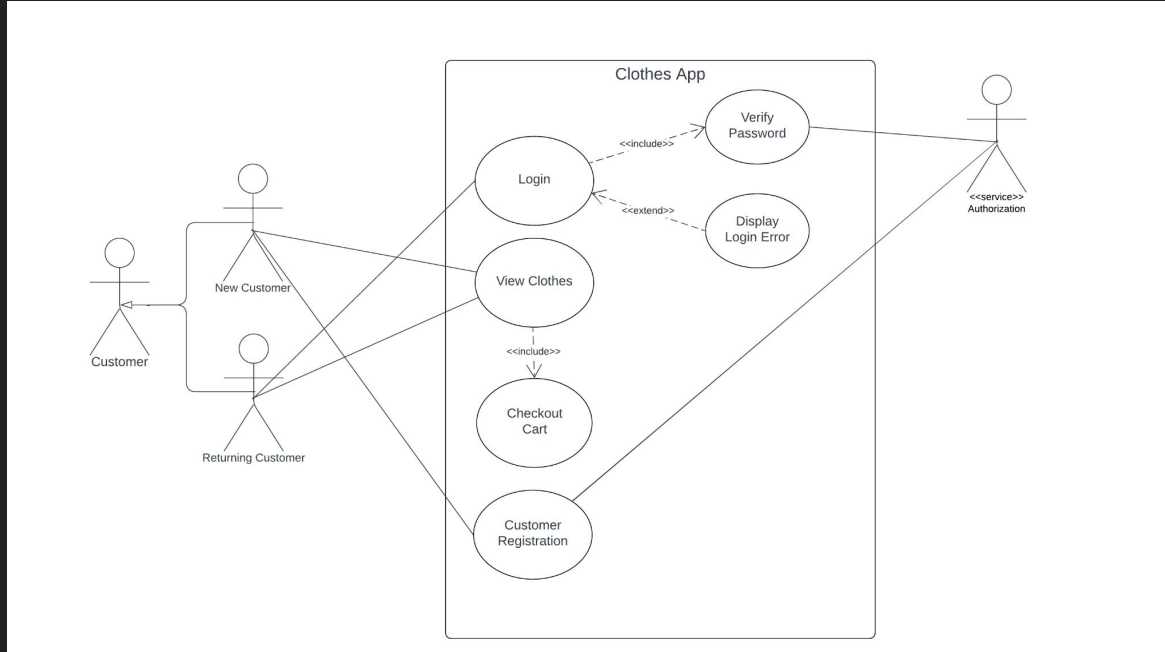
- Project Vision
- Requirements
- Analysis Model
- Architecture Model
- Design Model
- Remainder use case diagrams

Project Vision

Develop an online shopping application with an augmented reality (AR) try-on feature. Users can visually experience how clothing looks on them using the AR try-on feature, allowing them to make better decisions about purchasing the item.

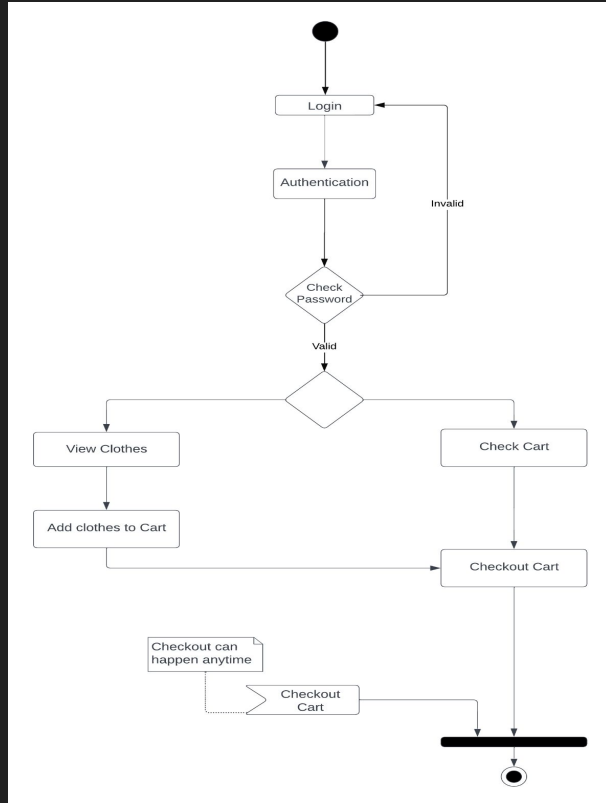
Additionally, the AR try-on feature saves time compared to physically going to stores to try on and compare different clothing items, significantly improving shopping efficiency.

Requirements - Use case diagram



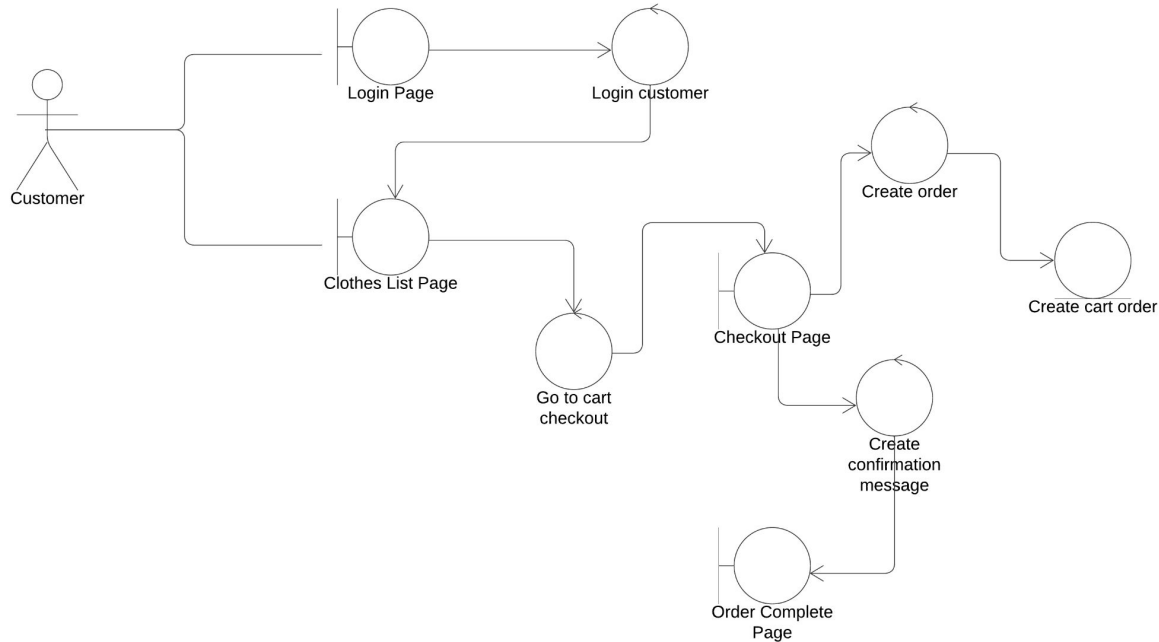
- Use-case: Buying Clothes
- Support 2 types of users
- Both can view clothes
- For prototype we do not consider any payment processor

Requirements - Activity Diagram



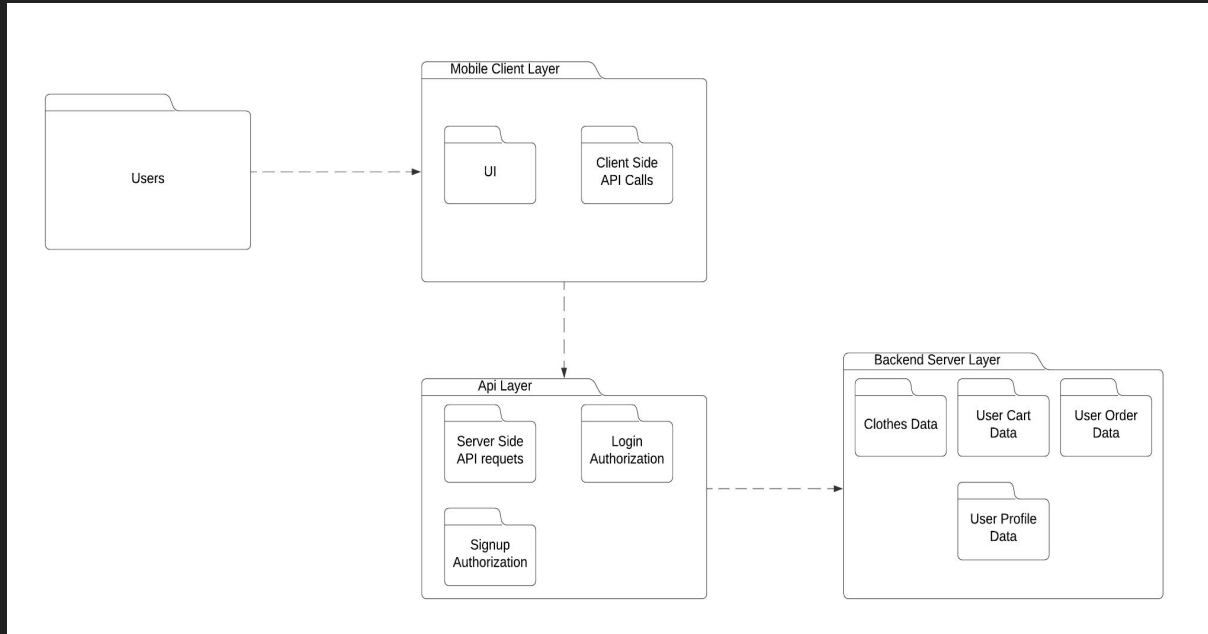
- Since not all users are logged in checkout can happen without login
- Logged in users can either checkout their existing cart
 - Or shop for more clothes

Analysis Model - Robustness Diagram



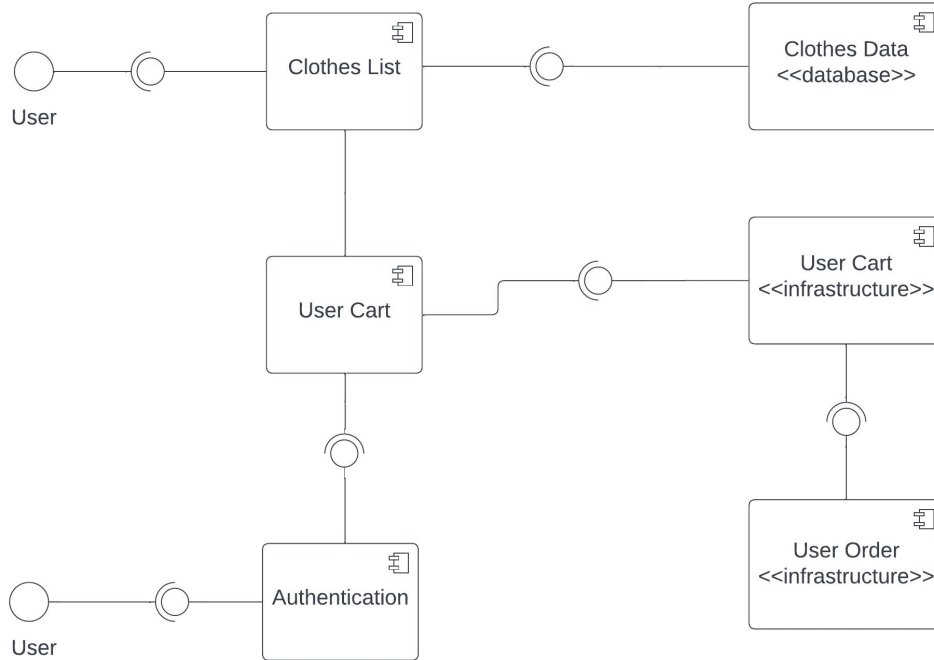
- User can either login or not
- Both types of users can view clothes
- Both can get to checkout page
- Controllers handle errors
 - E.g login controller handles wrong password or unregistered user trying to signin

Architecture Model – Package Diagram



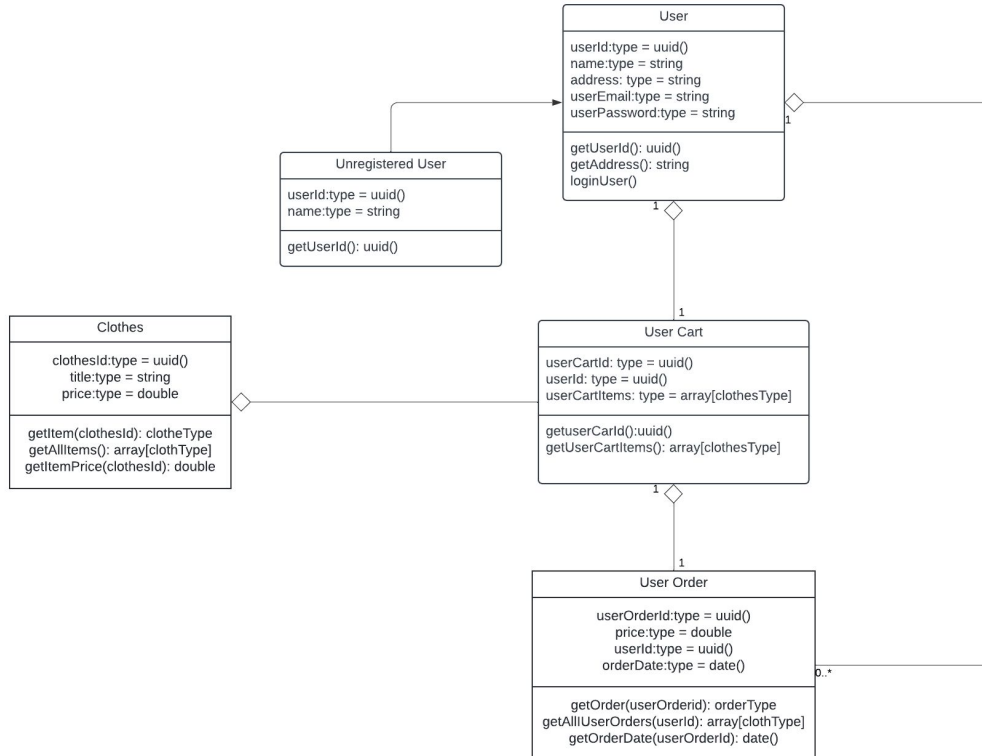
- Phone app is divided into two main components
 - The API layer acts as bridge between both front and back end
- Data is held in the back-end
- Front-end deals with user interface and calls to query API

Architecture Model – Component Diagram



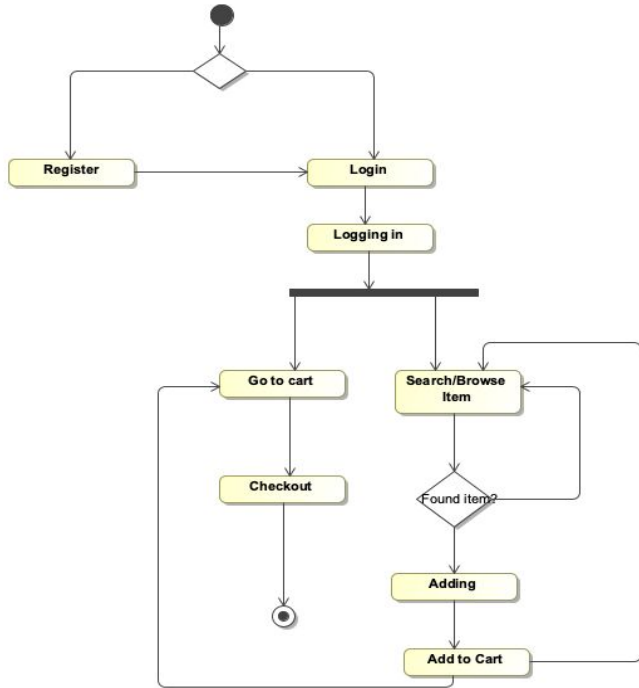
- Authentication only for users that log in
- Our back-end infrastructure provides the services for different actions possible
- Clothes List and User Cart are UI interfaces for the users to interact with

Design Model – Class Diagram



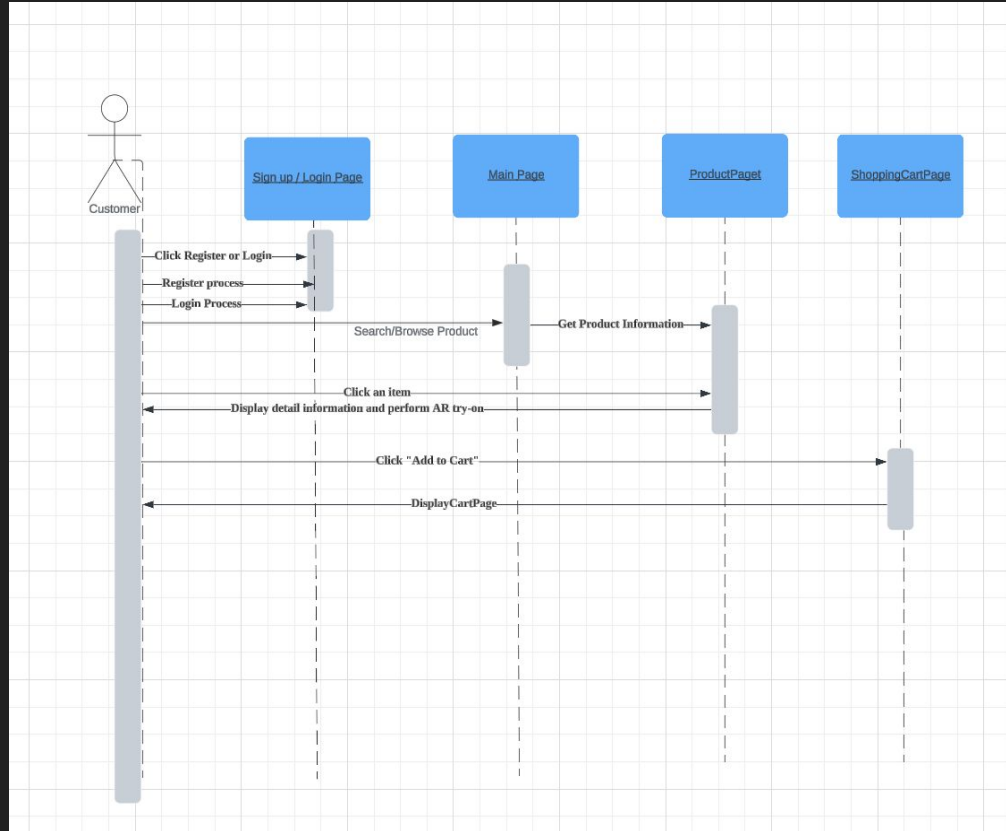
- User parent class
 - Unregistered Users are child class to User class
- A user can have many User Orders
 - But only one User Cart class association at a time
- A User Cart class can only have one User Order class association
- Uuid will provide unique id's for all class objects

Design Model – State Diagram



- For buying an item the User cart will be between two states empty or not-empty
- Browse for an item
 - If found add to cart
 - User Cart state has now changed
- When checkout happens
 - User Cart state has changed again

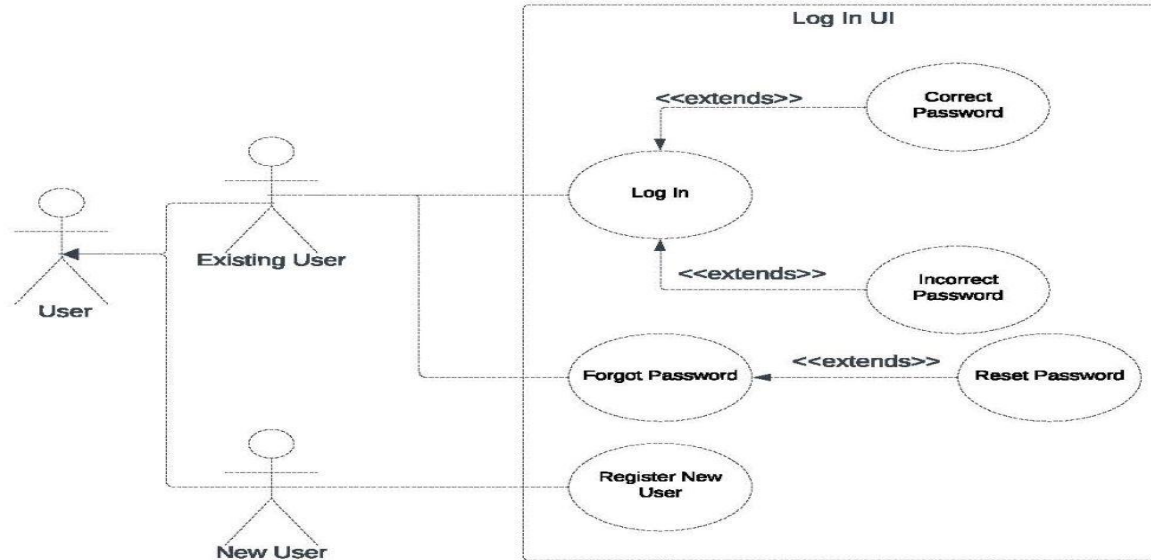
Design Model – Sequence Diagram



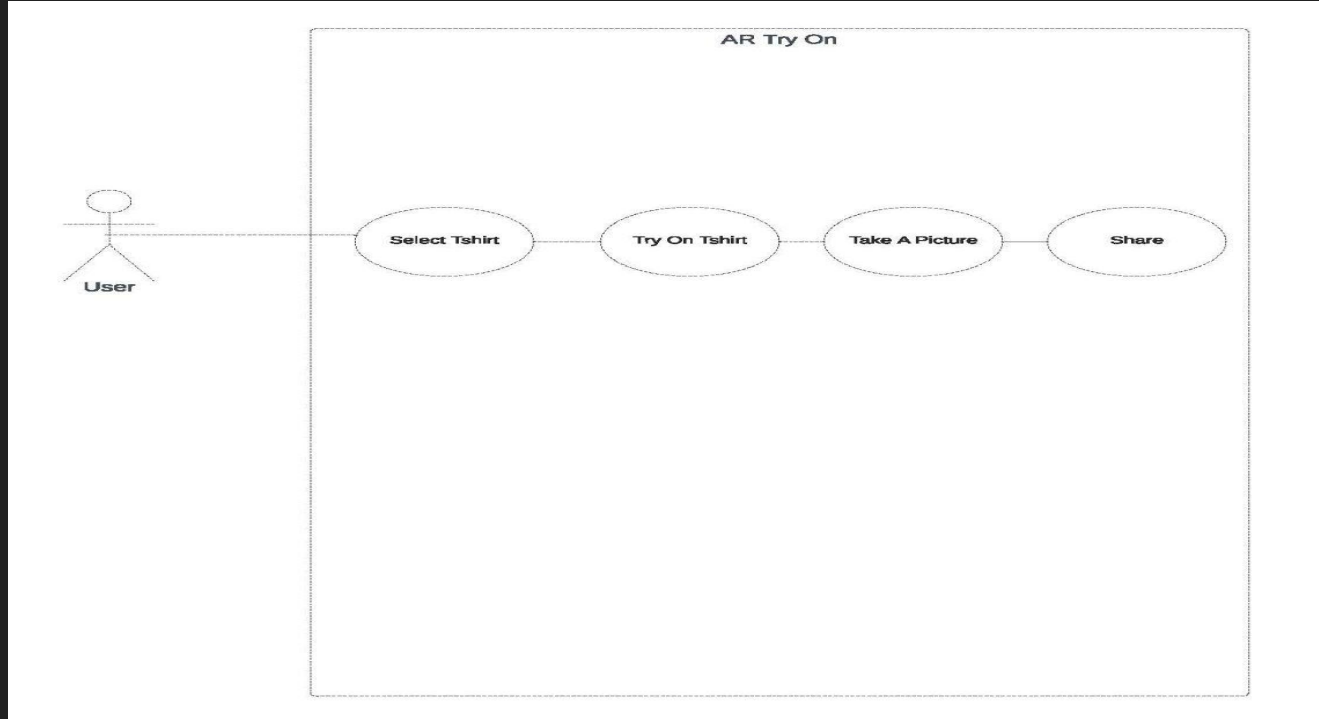
- All objects are ordered by a subordinate sequence
- Main Page can still be directly accessed even if not logging in
- Focus of control lines also depict how long the object is performing its action for our use case

Remainder Diagrams

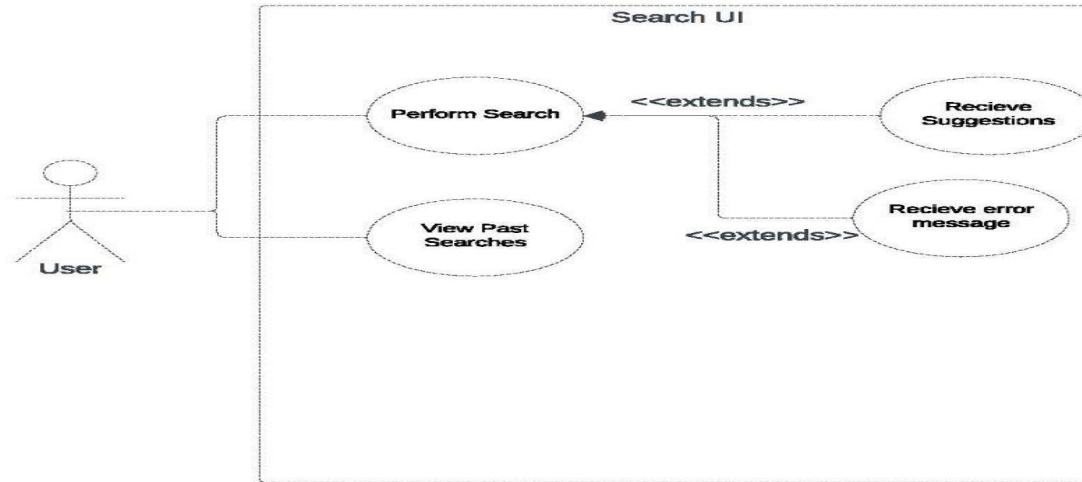
Login Use Case Diagram



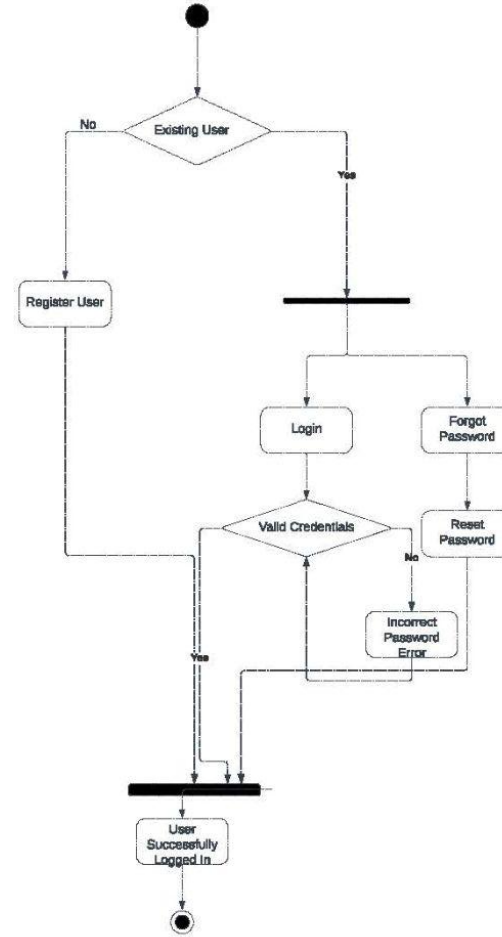
Ar Try On Feature Use Case



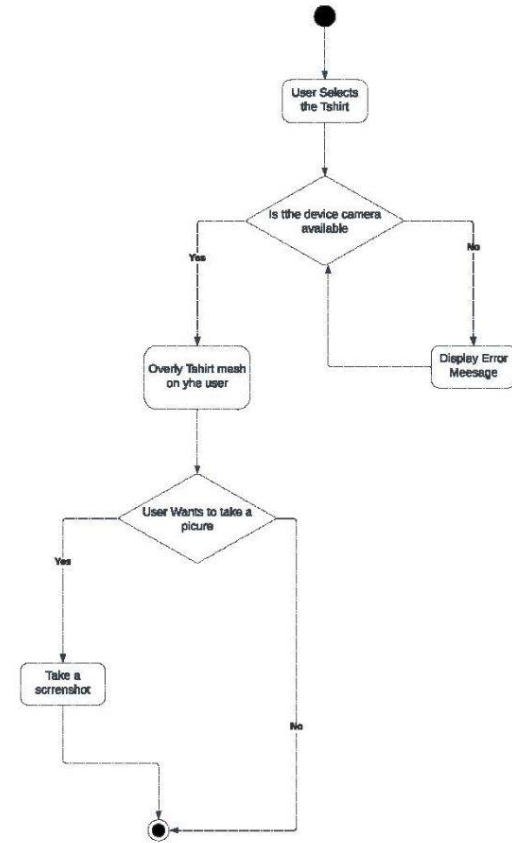
Search Use Case



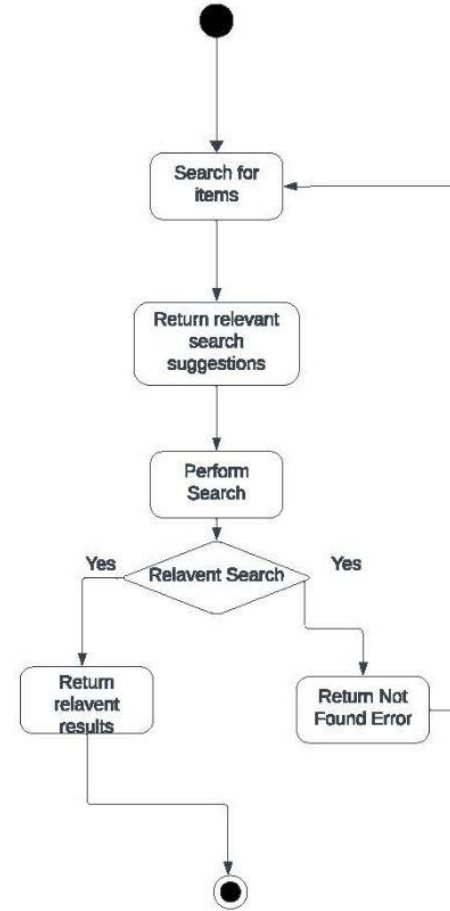
Login Activity Diagram



AR Tryon Activity Diagram



Search Activity Diagram



AR Tryon Class Diagram

