



Personal Investment

Introduction to Software Engineering

Prepared by

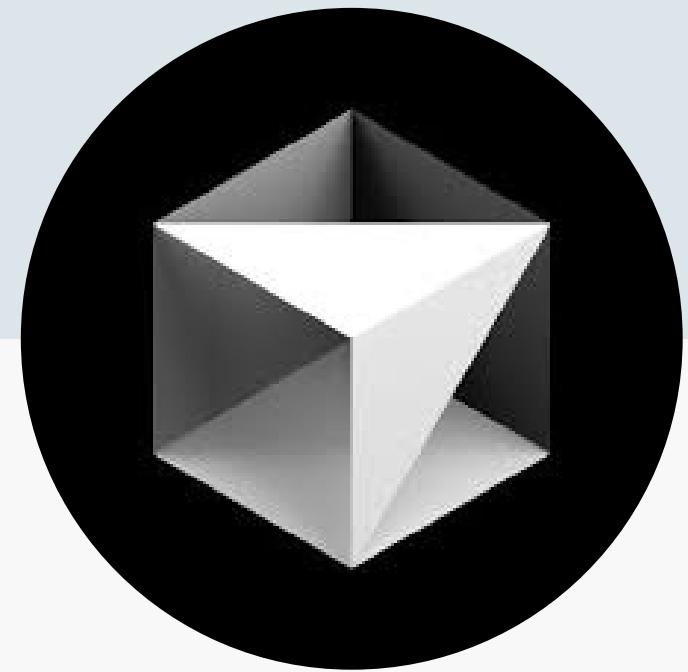
Naghm Wael Mohamed

Fatema El-Zhraa Ahmed

Aly El-Deen Yasser Ali



TechRadar Task



Cursor AI

By Ali



Claude AI

By Nagham

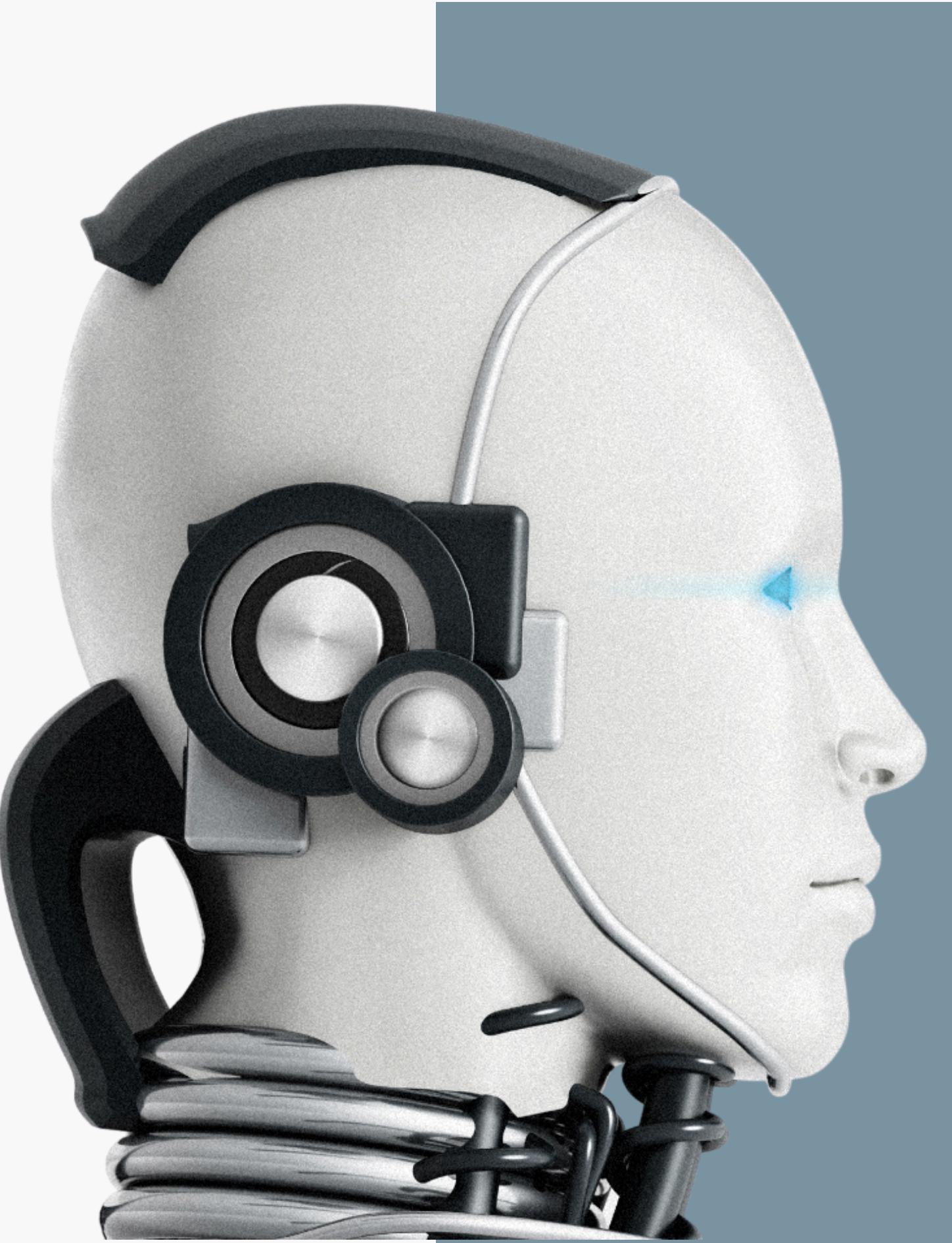


Cline

By Fatema

Claude AI

- Ethical & Private AI - Doesn't train on your data by default.
- Handles Any File - PDFs, Word, Excel, code, JSON, and more.
- Creates Anything - Code, diagrams, docs, HTML, translations.
- Adapts to You - Choose response styles (Concise, Formal, etc.).
- Multiple Models - Free Sonnet (smartest), fast Haiku, pro-grade Opus.
- For Everyone - Perfect for work, study, or creative projects.



Cursor AI

- AI-Powered IDE - VS Code-based with AI for code generation, refactoring, and queries.
- VS Code Familiarity - Easy transition for VS Code users.
- AI Chat & Codebase Interaction - Ask questions (Ctrl+L) and get code suggestions.
- Multi-File & Project Support - Generates project structures (Ctrl+I) for frameworks.
- Multi-Language Support - Works with Python, JS, Java, C++, etc., and flags errors.
- Limitations - Occasional mistakes, AI over-reliance risk, and paid plans.



Cline

- Open-source VSCode AI assistant - Integrates directly into the IDE for seamless coding support
- Powerful task handling - Excels at complex development tasks using Claude 3.5 Sonnet
- Privacy-focused design - Stores all data locally rather than in the cloud
- Smart workflow features - Plan & Act mode helps manage large codebases efficiently
- Resource considerations - High token usage and API rate limits may impact performance
- Community-powered - Open-source model enables transparency and collaborative improvements



Software Design Specification (SDS)

I

Architecture
Diagram

II

Class
Diagram

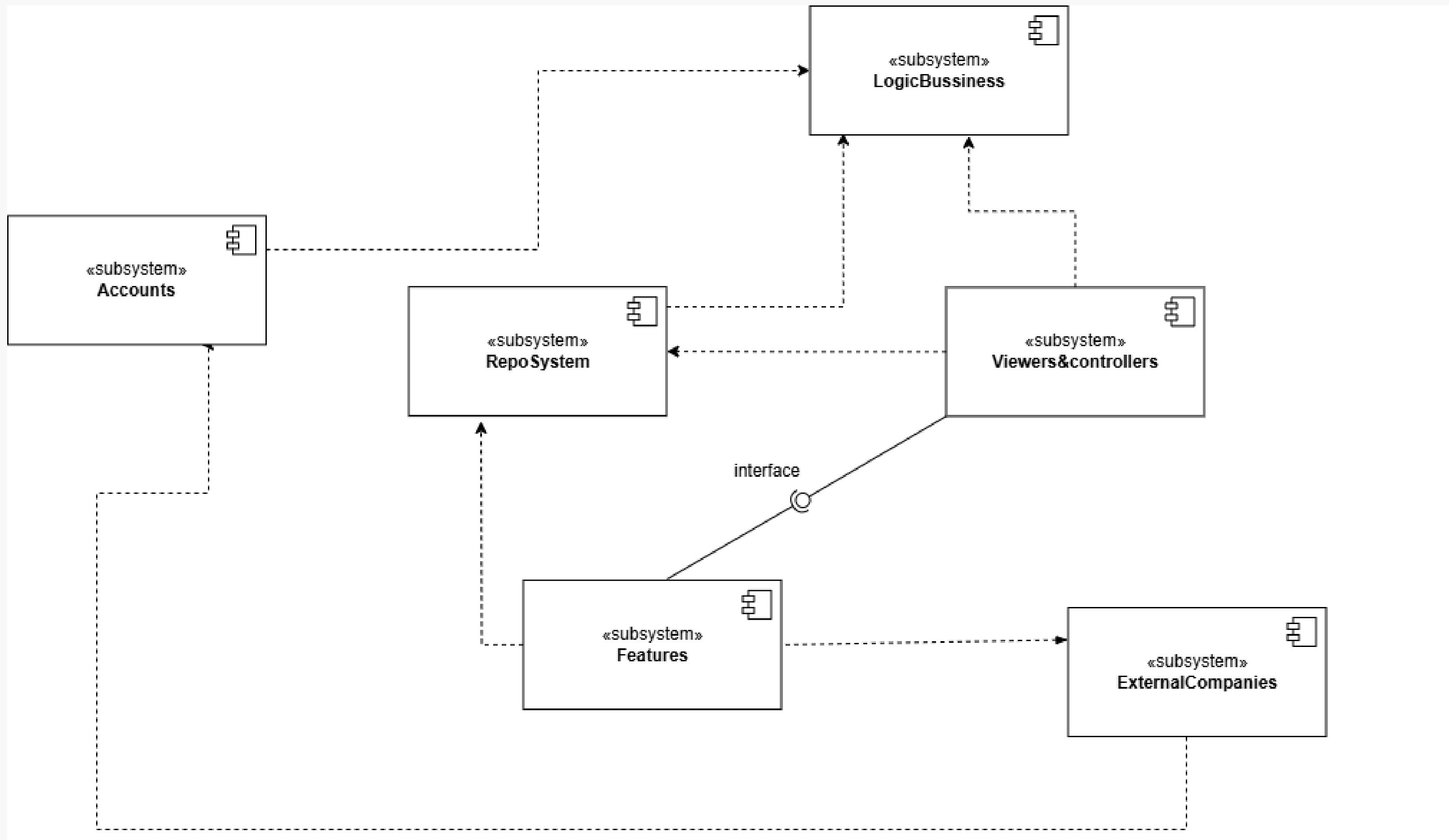
III

Sequence
Diagram

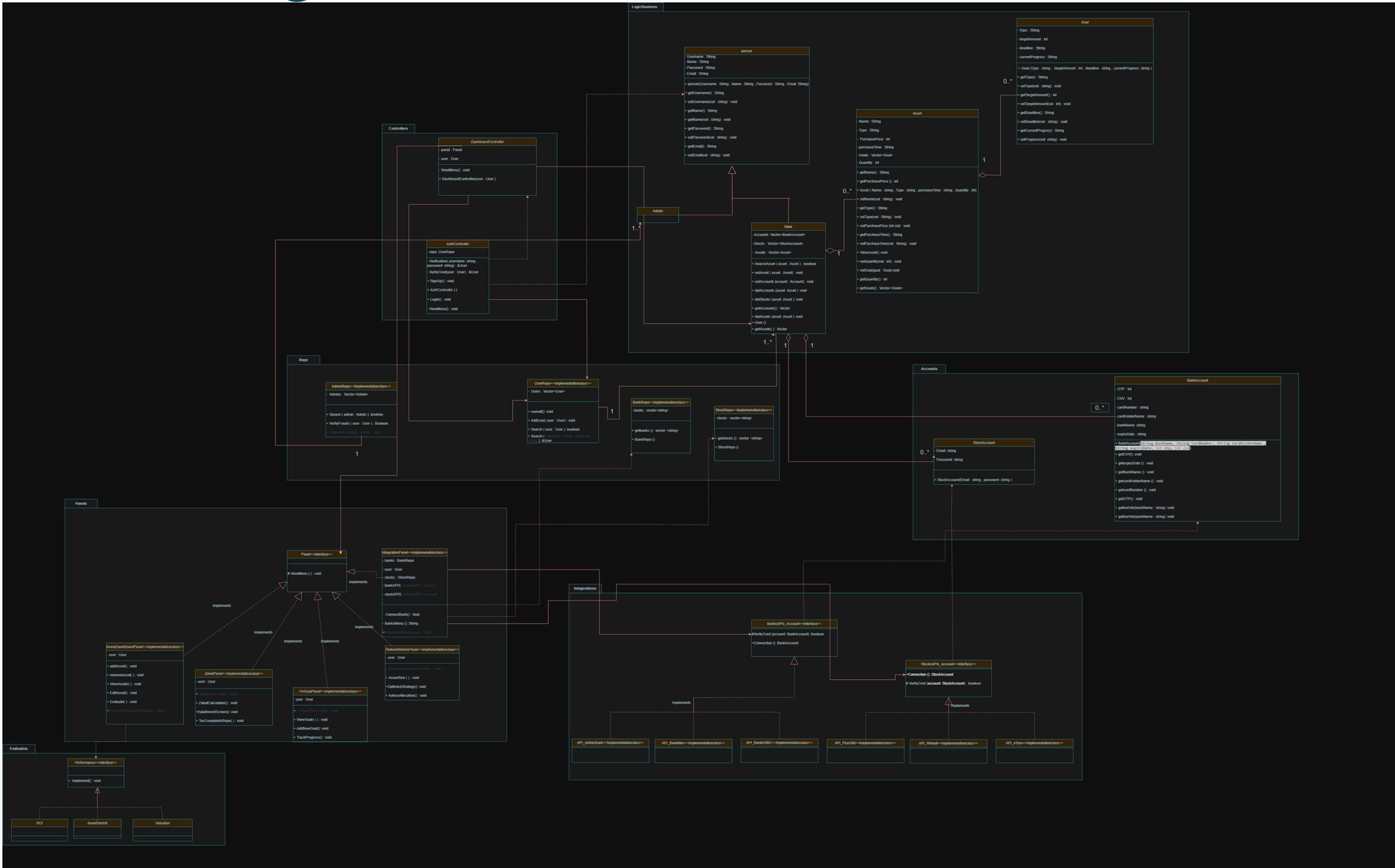
IV

State
Diagram

Architecture Diagram

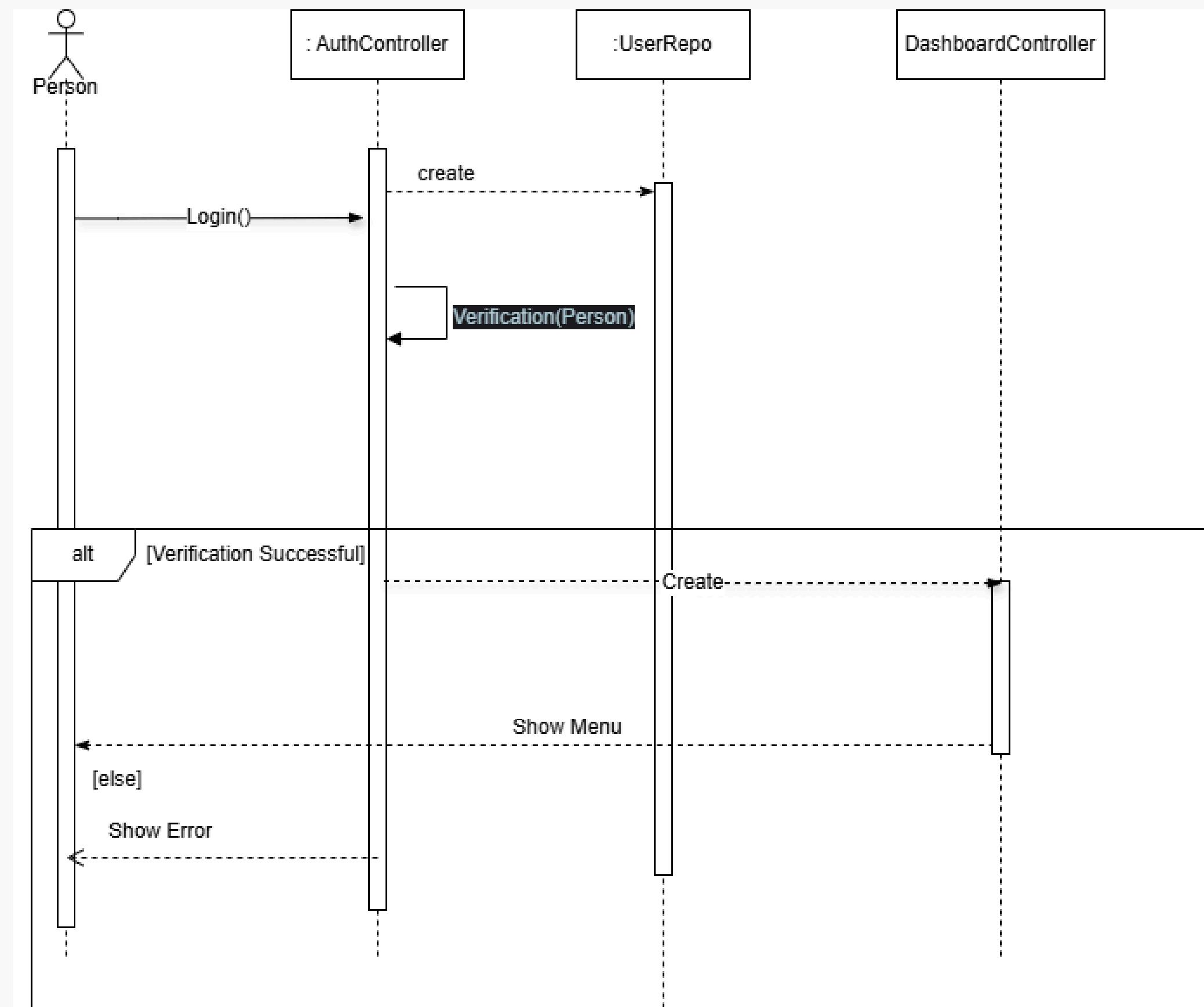


Class Diagram

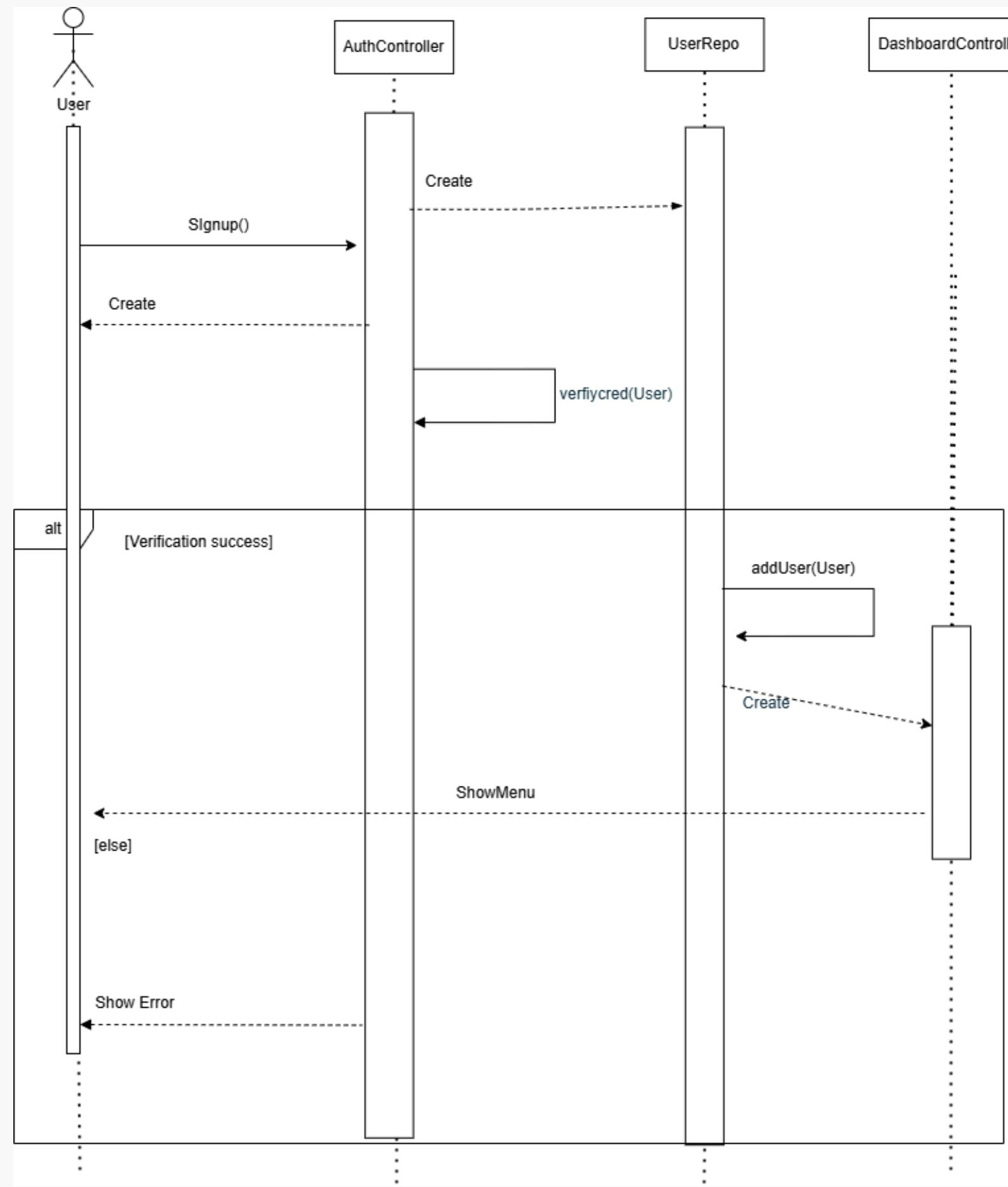


You can reach through that link :
<https://app.diagrams.net/#G1ur9hwcoiEpEGw7NfDLpcxsdz06gHVJE%7B%22pageId%22%3A%22C5RBs43oDa-KdzZeNtuy%22%7D>

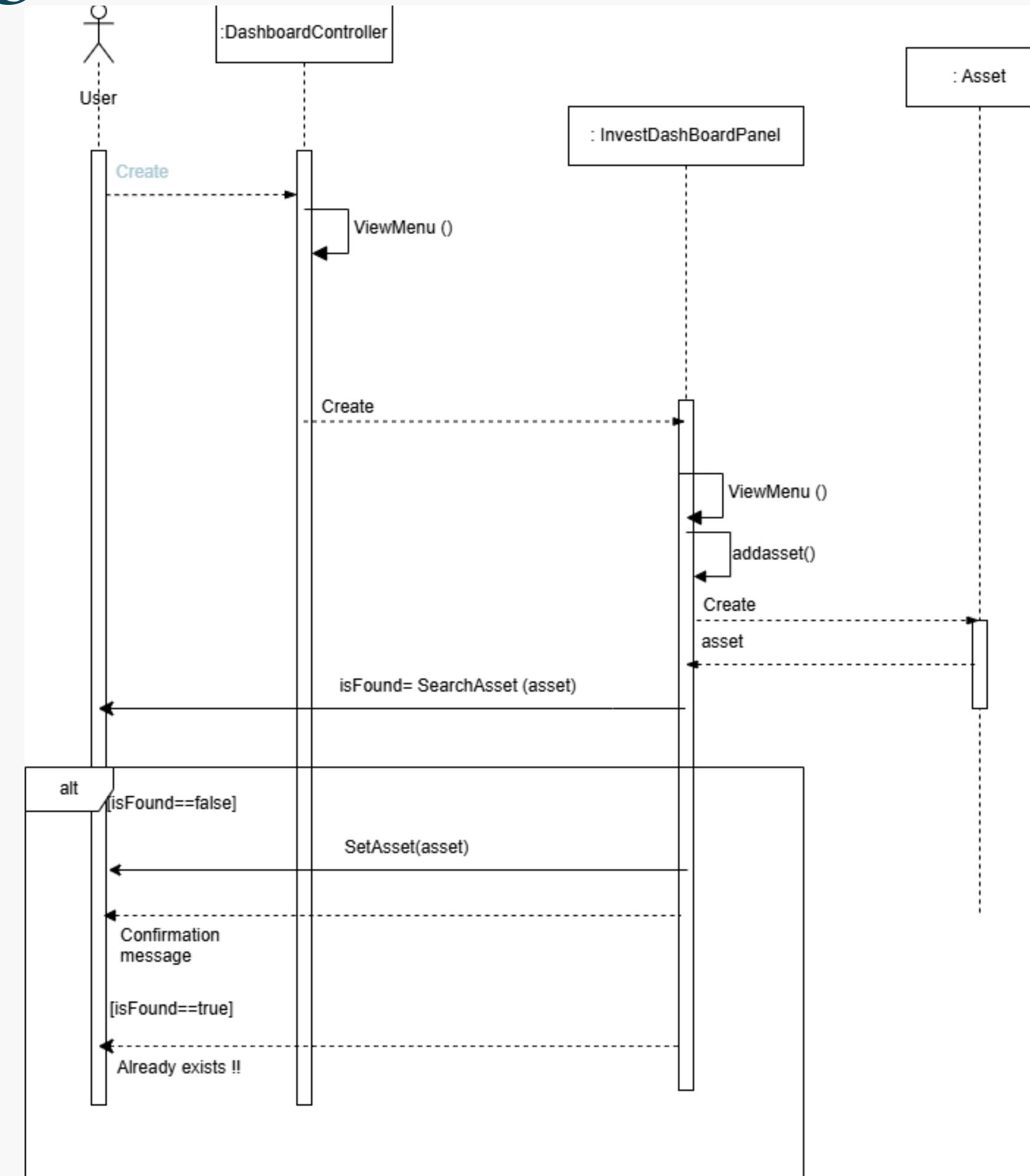
Sequence Diagram(Login)



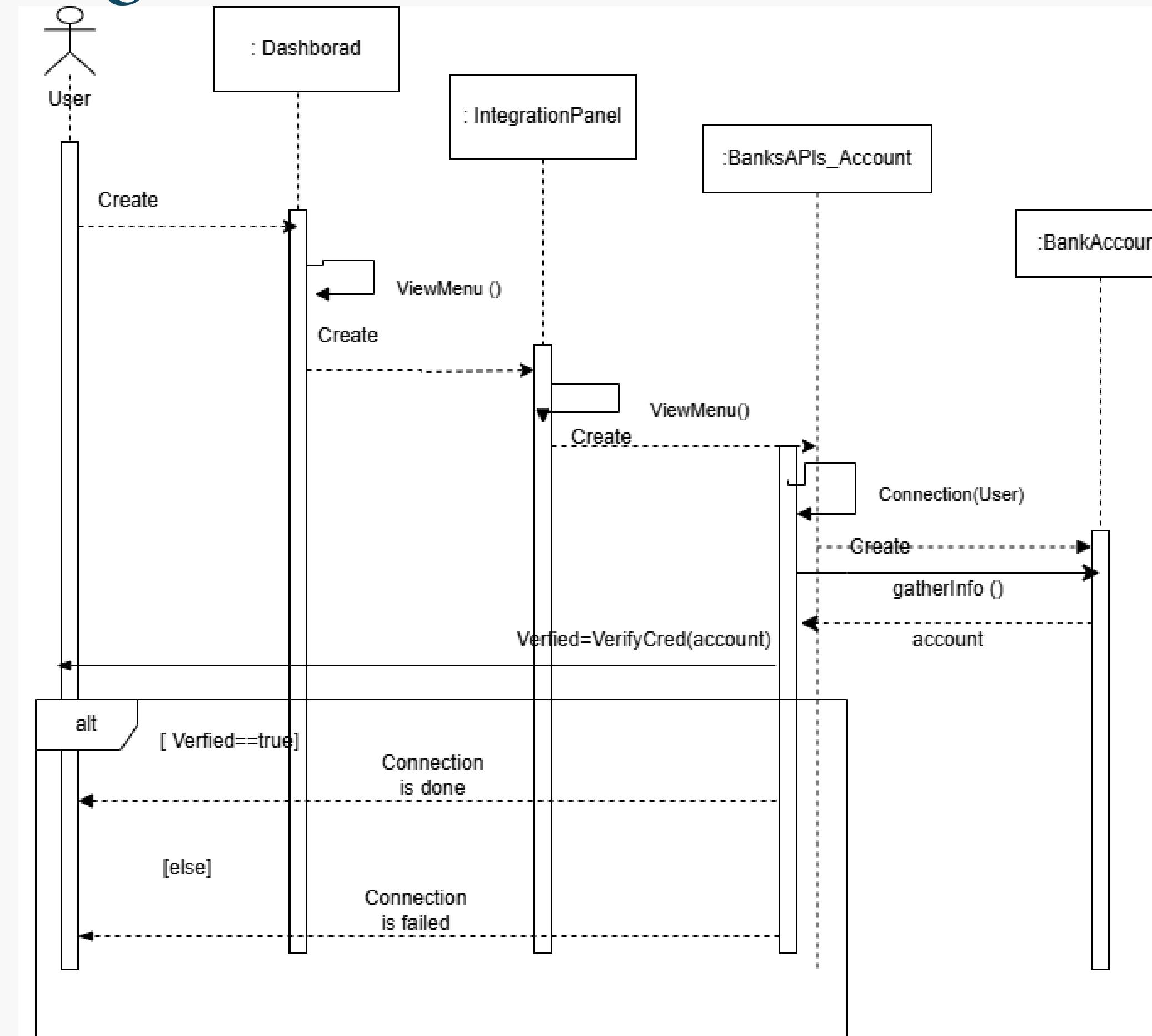
Sequence Diagram(Sign up)



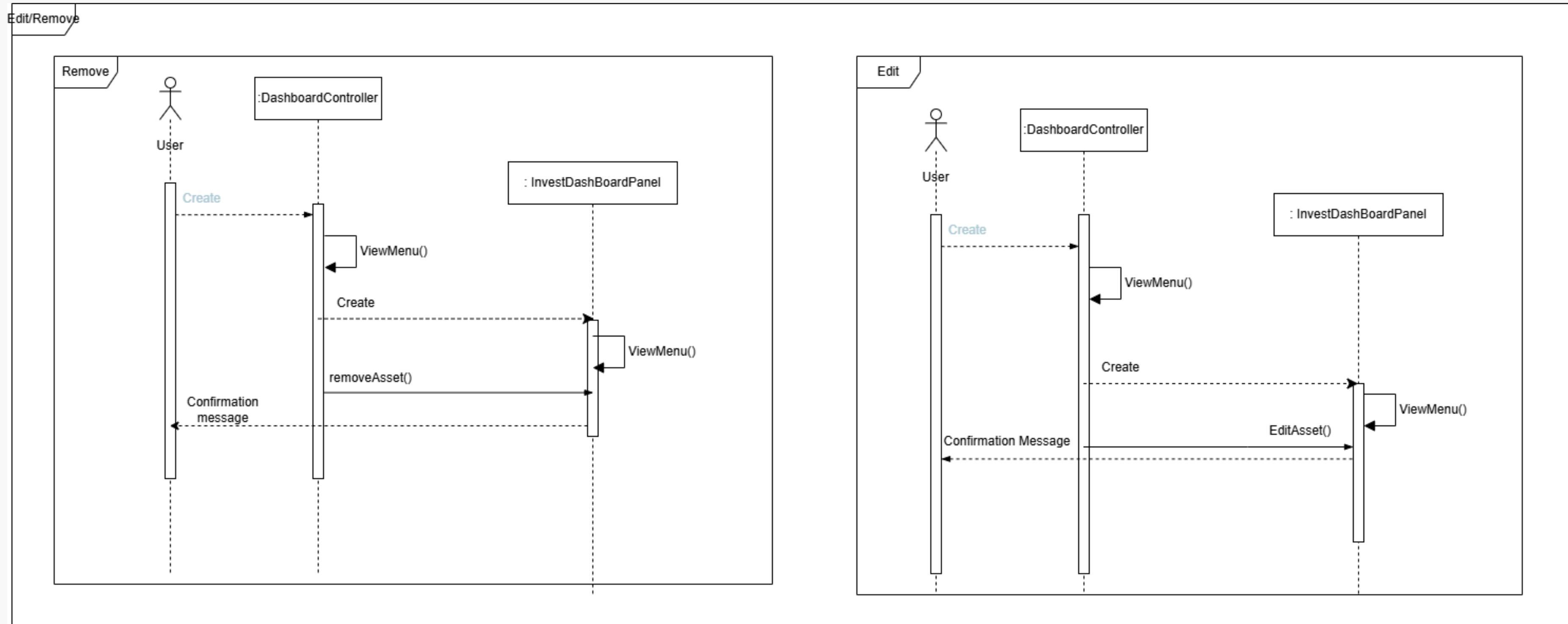
Sequence Diagram (Add Asset)



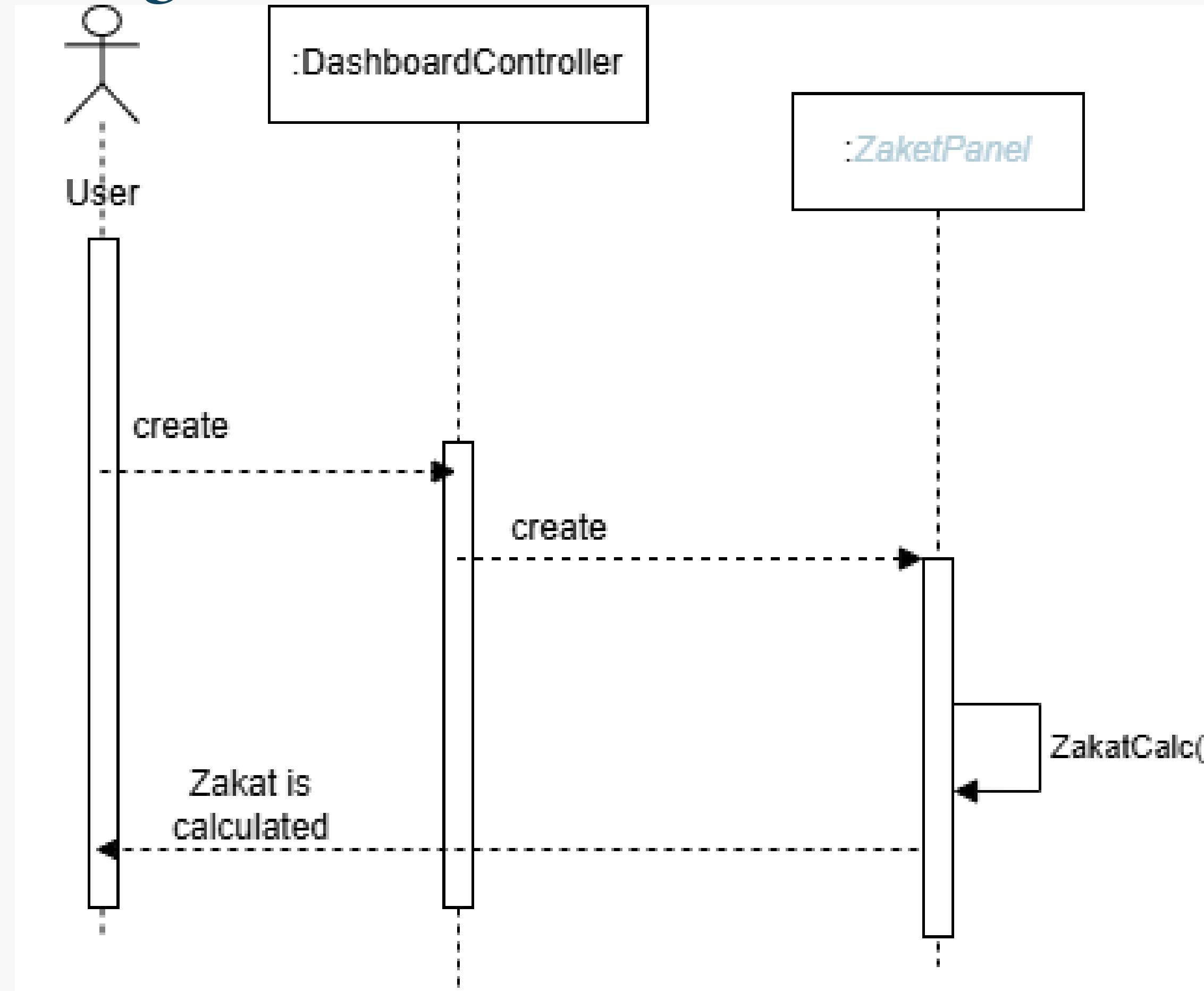
Sequence Diagram(Connect Bank Account)



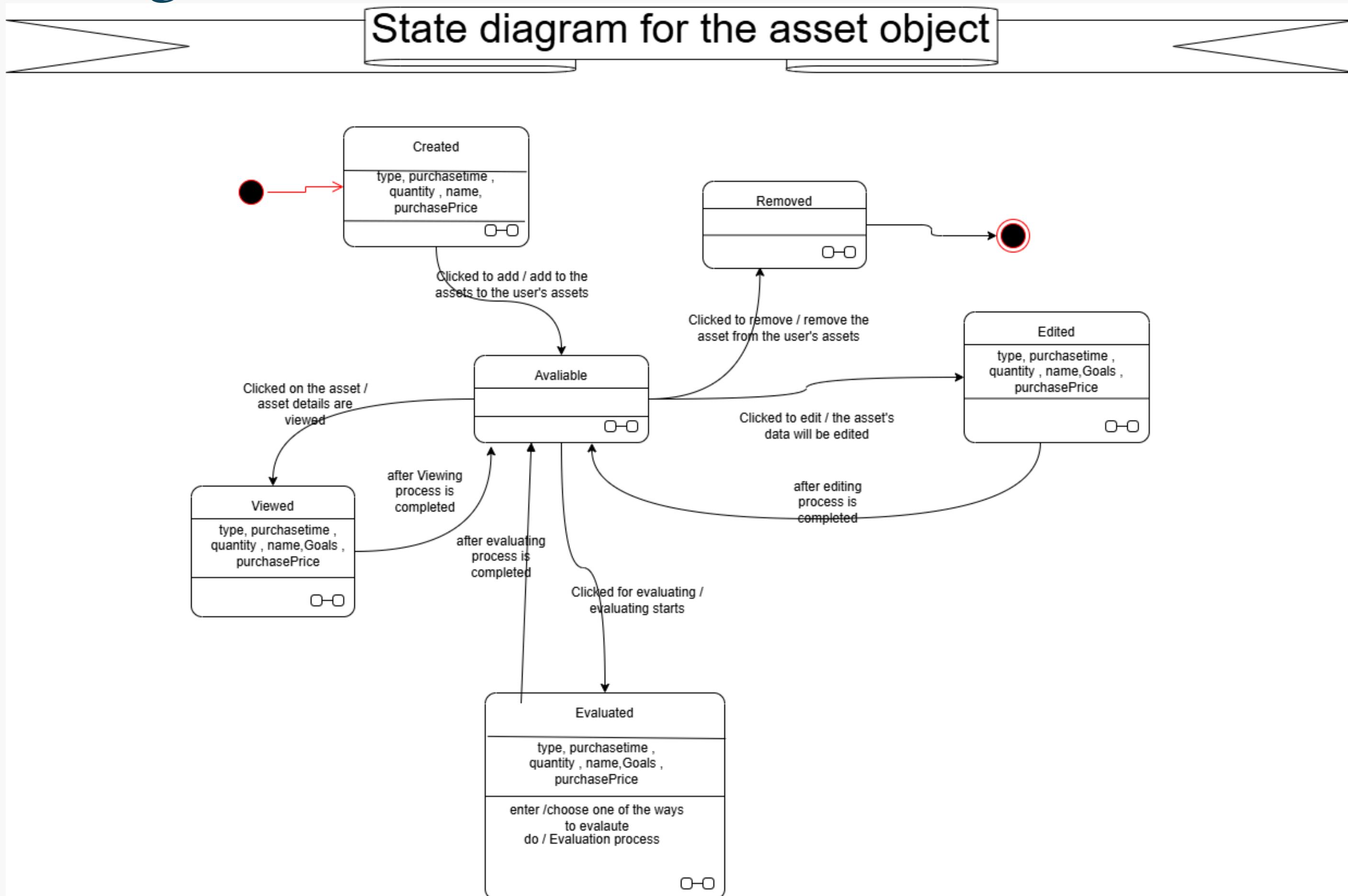
Sequence Diagram(Edit/Remove Asset)



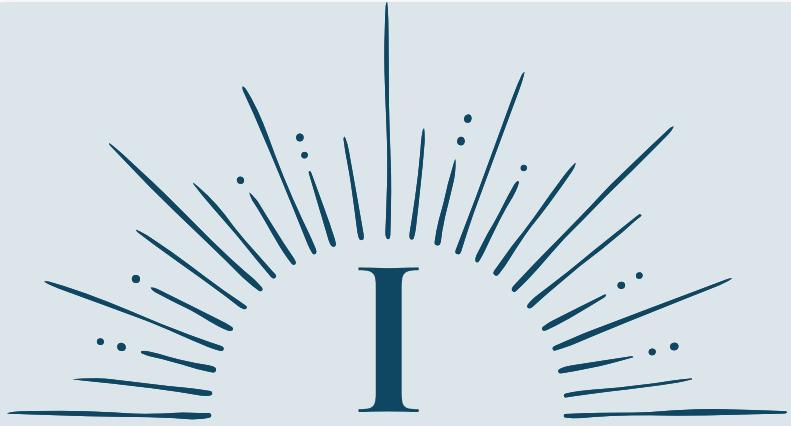
Sequence Diagram(Zakat Calculation)



State Diagram



Design Patterns



**Strategy
Pattern**



**Repository
Pattern**



**MVC
Pattern**

Strategy Pattern

Purpose:

- Encapsulate interchangeable behaviors for APIs.

Benefits:

- Clean, modular, and easily extensible for new APIs.

Implementation:

- Performance interface enforces a common rule (`implement()`).
- `BanksAPIs_Account` and `StocksAPIs_Account` standardizes connection and credential verification across APIs.

Repository Pattern

Purpose:

- Clean, modular, and easily extensible for new APIs. Encapsulate interchangeable behaviors for APIs.

Benefits:

- Single point of change for data operations.
- Single point of change for data operations.

Implementation:

- UserRepo and AdminRepo manage database operations (e.g., AddUser, VerifyFrauds).
- Encapsulates queries and in-memory collections (Vector<User>, Vector<Admin>).

Future Use:

- BankRepo and StockRepo for managing company data, enabling easy API integration later.

MVC Pattern

Purpose:

- Separate data, UI, and logic for maintainability.

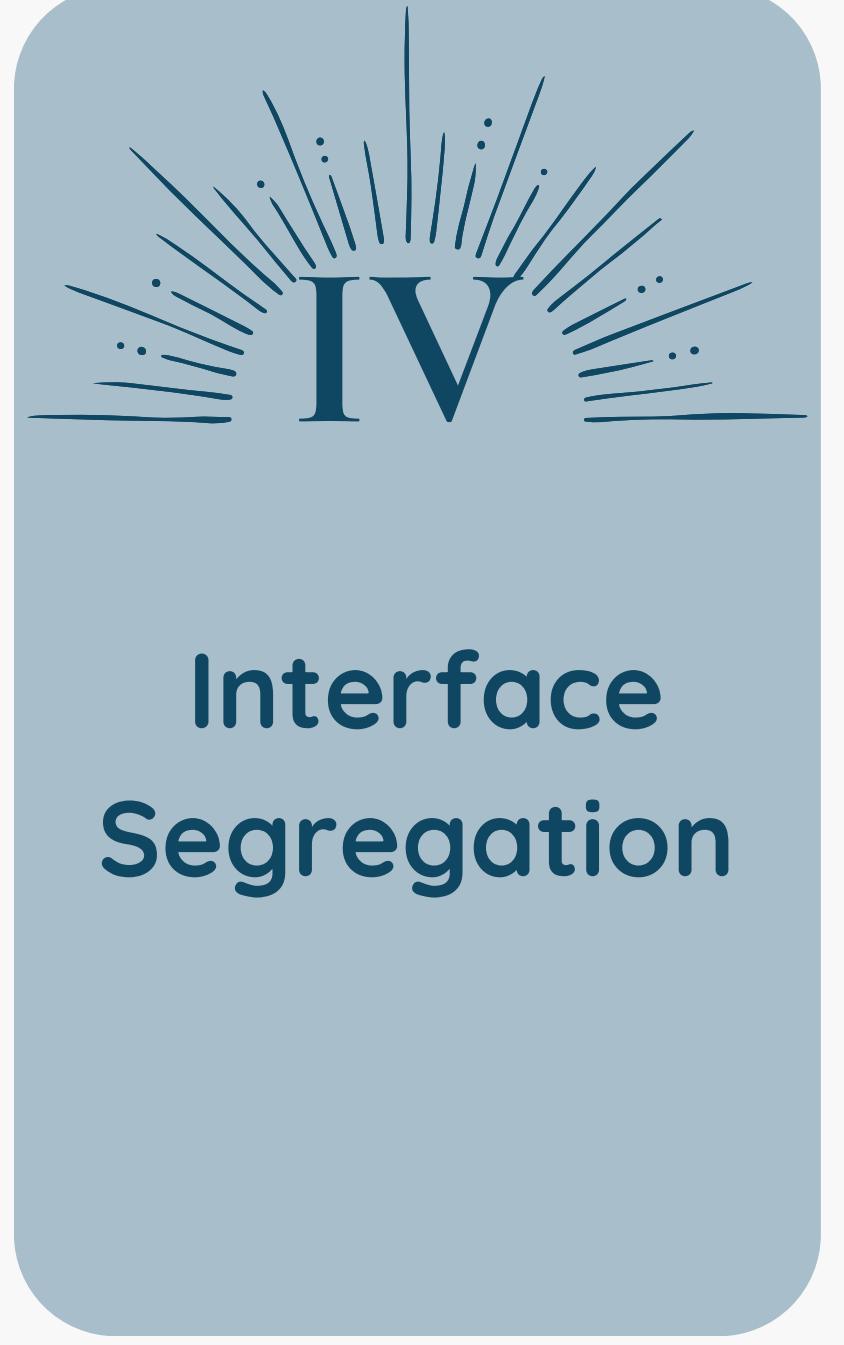
Benefits:

- Easy UI updates without modifying models.
- Easy UI updates without modifying models.
- Clear user-action flow (e.g., login → dashboard).

Components:

- **Model:** Handles data and business rules (e.g., Person, Asset, repositories).
- **View:** Displays UI (Panel implementations like ZakeiPanel, RiskandAdvicePanel).
- **Controller:** Manages flow (AuthController for login, DashboardController for routing).

SOLID Principles



Single Responsibility

Goal:

- Each class/interface has one clear responsibility.

Benefits:

- Improved modularity, maintainability, and scalability.

Examples:

- AuthController: Handles only authentication (login/signup).
- Panel interface: Child classes (e.g., ZakeiPanel, FinGoalPanel) implement specific UI logic independently.
- User, Goal, Asset classes: Manage their own data without overlapping logic.

Liskov Substitution

Goal:

- Subclasses extend base classes without altering behavior.

Benefits:

- Safe extensibility; new features won't break existing code.

Examples:

- Panel interface: All child panels (e.g., RiskAndAdvicePanel) implement ViewMenu() interchangeably.
- BankAPIs_Account: Implemented by bank APIs (e.g., API_AIAhlyBank), ensuring consistent connection/verification.
- StockAPIs_Account: Implemented by bank APIs (e.g., API_500plus), ensuring consistent connection/verification.

Open/Closed

Goal:

- Extend functionality without modifying existing code.

Benefits:

- Stable core system with flexible extensions.

Examples:

- User aggregates Assets/BankAccounts: New asset types can be added without changing User.
- Composition: Asset-Goal association allows new goal logic without altering Asset.
- Inheritance: Person superclass supports future roles (e.g., Admin, User).

Interface Segregation

Goal:

- Avoid "fat interfaces"; split into role-specific contracts.

Benefits:

- No forced overrides (e.g., DashboardController depends only on relevant panels).
- Cleaner contracts (e.g., Evaluate() in FilesPanel, OptimizeStrategy() in RiskAndAdvicePanel).

Examples:

- Panel interface divided into smaller roles (e.g., FilesPanel, FinGoalPanel).
- RiskAndAdvicePanel implements only risk-related methods, avoiding unused dependencies.

.....

*Methodology,
Technologies
& Code Organising*

.....

Methodology

Designed the System (SDS):

- Created and validated all diagrams
- Ensured alignment with requirements and user stories.

Task Planning & Division:

- Held online meetings to plan implementation.
- Assigned tasks to team members.

Implementation:

- Structured the project into packages.
- Developed individual code components.
- Used GitHub for version control and collaboration.

Iterative Refinements:

- Fixed code issues during development.
- Updated diagrams/modifications as needed.

Testing & Finalization:

- Created data files for testing and resolved bugs

Technologies & Tools

- **GitHub:** version control and sharing work
- **Java:** Programming language
- **draw.io:** Making system diagrams
- **Miro:** Making system navigation diagram
- **IntelliJ Idea:** IDE for writing code
- **Javadoc:** For generating documentation

Code Organising

The code is organised in a main function and 7 Packages

- **Controllers:** Contains AuthController.java and DashboardController.java
- **ExternalAccs:** Contains BankAccount.java and StockAccount.java
- **Funcs:** Contains HelperFunc.java
- **Integrations:** Contains API'S simmulation
- **LogicBusiness:** Contains Asset.java , Goal.java , Admin.java and User.java
- **Panels:** Contains IntegerationPanel.java , InvestDashBoardPanel.java , Panel.java and ZaketPanel.java
- **Repo:** Contains UserRepo.java , StockRepo.java , BankRepo.java , Users and Banks

.....

*Implementation
& Documentation*

.....

Implementation & Documentation

fatema123elfiky / Tharwa-Project

Type / to search

Issues 6 Pull requests Actions Projects Security Insights

Tharwa-Project Private Watch 1 Fork 1 Starred 1

main 2 Branches 0 Tags Go to file Add file Code

Aly-El-Badry Merge pull request #33 from Aly-El-Badry/main 89ec0b0 · 15 minutes ago 168 Commits

.idea	professional modification	29 minutes ago
CS251-2025-S14-Mostafa-A2FINAL-Personal I...	submission file	7 hours ago
Java Side-Project - LCNC Project	professional modification	29 minutes ago
TechRadar	Added Final TechRadar Report	18 hours ago
Tharwa-Implementation	professional modification	29 minutes ago
Tharwa-Pre-project	professional modification	29 minutes ago
Tharwa-SDS	professional modification	29 minutes ago
Tharwa-SRS	professional modification	29 minutes ago
README.md	professional modification	30 minutes ago

About

That Project implements idea of application "Tharwa" The idea about investing . That project the first trial for us to implement software from scratch and applying the cycle of the software engineering. Here you can see more details about the project in the README. What about having a quick look :)

Readme Activity 1 star 1 watching 1 fork

Releases



GitHub Repo URL:

<https://github.com/fatema123elfiky/Tharwa-Project>

Implementation & Documentation

github.com/fatema123elfiky/Tharwa-Project/commits/main/

1 Gmail YouTube Adobe Acrobat

NaghmProgrammer authored 2 days ago

Commits on May 9, 2025

- Helper is added + coding**
fatema123elfiky committed 2 days ago
- Delete Implementation phase/src/BankInteger direct**
fatema123elfiky authored 3 days ago
- Adding class**
fatema123elfiky committed 3 days ago
- Delete Implementation phase/src/LogicBusiness dire**
fatema123elfiky authored 3 days ago
- Delete Implementation phase/src/BankInteger/Bank/**
fatema123elfiky authored 3 days ago
- returned var names to original to match class diagram**
NaghmProgrammer authored 3 days ago
- linked User.java with ExternalAccs package**
NaghmProgrammer authored 3 days ago
- Refactoring Packages**
fatema123elfiky committed 3 days ago
- Update User.java**
NaghmProgrammer authored 3 days ago

github.com/fatema123elfiky/Tharwa-Project/commits/main/

YouTube Adobe Acrobat

Commits on May 12, 2025

- Added combined TechRadar tasks**
NaghmProgrammer authored 10 hours ago

Commits on May 11, 2025

- added remove/edit asset**
NaghmProgrammer authored yesterday

Commits on May 10, 2025

- Add files via upload**
Aly-El-Badry authored 2 days ago
- Fourth iteration**
fatema123elfiky committed 2 days ago
- Update AuthController.java**
NaghmProgrammer authored 2 days ago
- Added Login , Verification**
NaghmProgrammer authored 2 days ago

Merge pull request #15 from fatema123elfiky/Zhraa

Merge pull request #14 from fatema123elfiky/main

github.com/fatema123elfiky/Tharwa-Project/commits/main/

YouTube Adobe Acrobat

Commits on May 3, 2025

- final of the finalist final**
Ali-El-Badry committed last week
- Merge pull request #32 from fatema123elfiky/main**
- Merge pull request #31 from fatema123elfiky/Zhraa**
- Merge pull request #1 from fatema123elfiky/Zhraa**
- Update README.md**
fatema123elfiky authored last week
- Merge pull request #30 from fatema123elfiky/main**
- small edit**
fatema123elfiky committed last week
- Merge pull request #29 from Ali-EL-Badry/new**



Thank you

