

Phase 1 Presentation



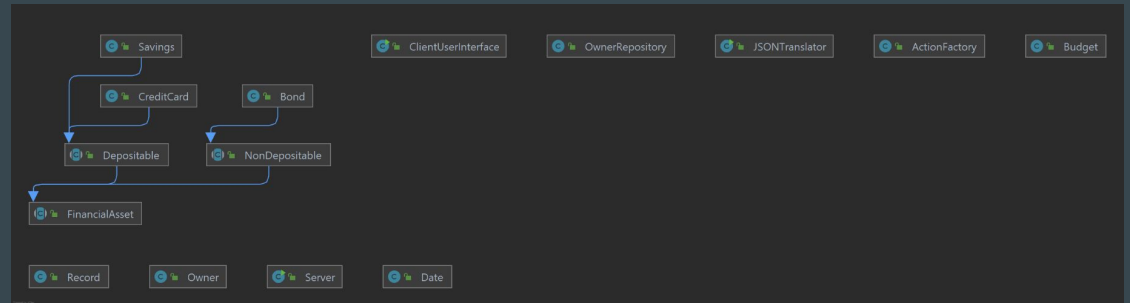
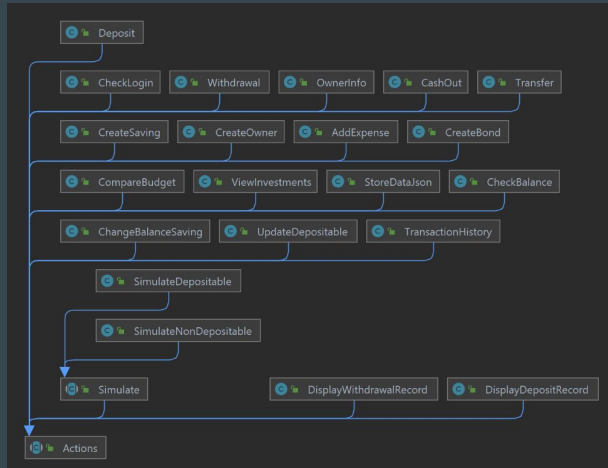
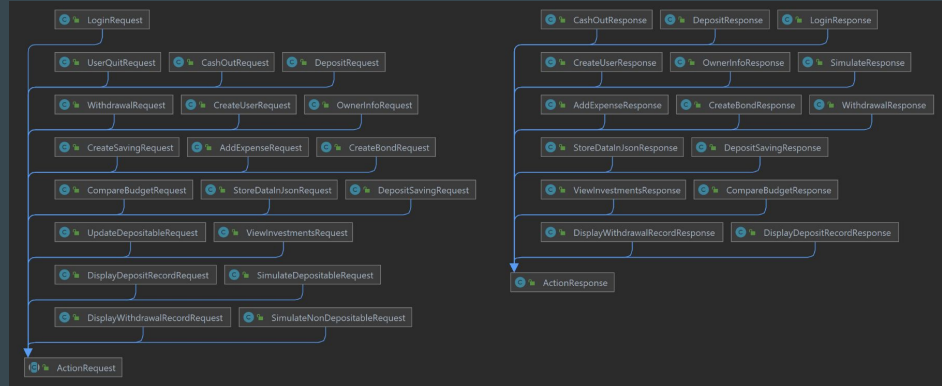
Group Members: Aliyan, Bernard, Ruiting, Tiffany, Miguel, Linh, Anita, Leya

Updated Specification

Domain: Financial Planning



Class Diagram



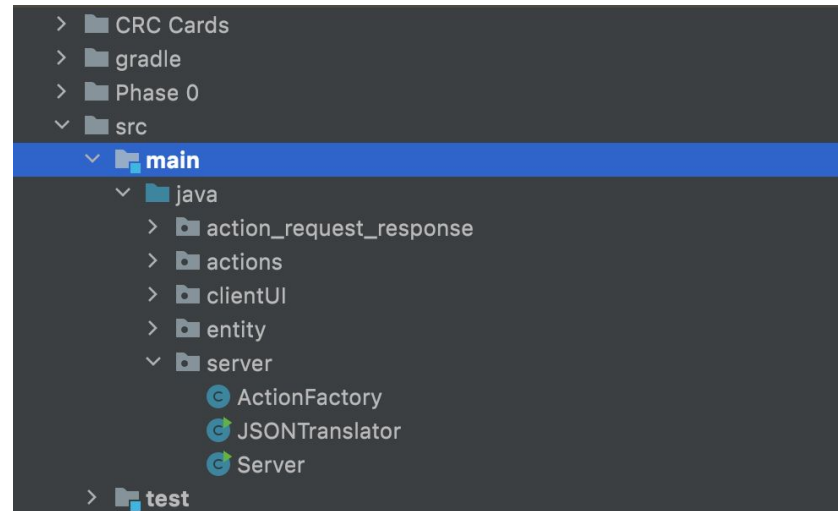
Clean Architecture and Scenario Walkthrough

Major Design Decisions

SOLID Principles

- Single responsibility Principle
- Open-Closed Principle
- Liskov substitution principle
- Interface Segregation principle
- Dependency inversion principle

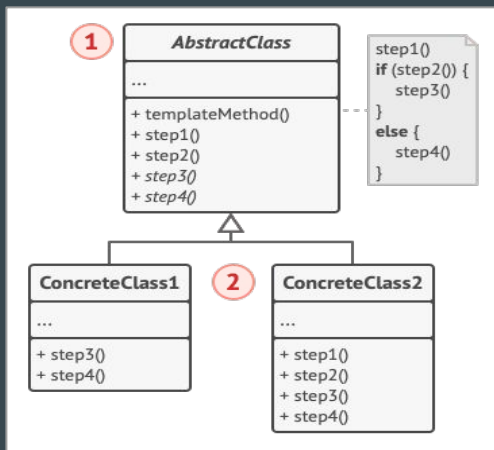
Packaging Strategies



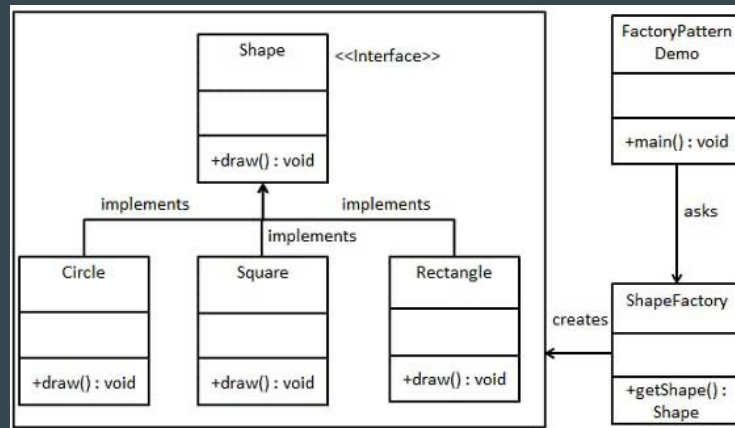
- Used the packaging by layer strategy
-

Design Pattern

- Template Method (Pull request #68):
 - Used in the Simulate, SimulateDepositable, and SimulateNonDepositable Class.



- Factory Design Pattern (Pull request #29):
 - Used in ActionFactory in Server package



Refactor

Github Features

- Utilized the Issues and Review/Conversation tab to identify any potential issues.

The screenshot shows a GitHub issue page for the repository 'CSC207-UofT / course-project-marble'. The issue title is 'Server doesn't appear to close a connection if incorrect object sent ClientUserInterface #21'. It is marked as 'Closed' and was opened by 'a-gohar' 24 days ago. The issue description states: 'The server will keep the connection open if an incorrect Object is sent from the client. Should not be doing this because the thread should destroy itself in the catch exception part.' There are two comments from 'a-gohar': one from 24 days ago and another from 15 days ago stating 'Fixed by adding try catch'. The issue was closed by 'a-gohar' 15 days ago.

The screenshot shows a GitHub Pull Request conversation. The top bar indicates 2 conversations, 1 commit, 2 checks, and 1 file changed. The conversation starts with a comment from 'MJAquarion2413' 5 days ago, stating: 'Will work but needs classes that will be serialized to implement serializable, will change that once I'm absolutely sure that serialization wont have any adverse effects. Testing area at bottom of code with example'. This is followed by a commit from 'Serialization Works, we need to add implements serialization to class...' with a green checkmark and the commit hash '1140b98'. Then, 'ruiting-chen' requested changes 5 days ago, suggesting renaming the class to 'ReadWrite' and renaming the methods 'Serialize' and 'Deserialize' to 'saveToFile' and 'readFromFile'. 'ruiting-chen' left a comment: 'I would suggest renaming this class to ReadWriter instead of Serialization, so that it is more clear what this class does. And rename the methods Serialize and Deserialize to saveToFile and readFromFile, so that it is expressed what the methods do more explicitly.' Finally, 'leya-595' reviewed the changes 4 days ago, suggesting to 'Add a bit more documentation for serialize and deserialize to explain the methods purpose.' The code snippet shown is from 'src/main/java/server/Serialization.java' and includes a method 'public void Serialize(Object obj) throws IOException'.

Demo of Program and Tests

What Worked Well and Future Plans

- Overall Future Plans:
 - Work on the UI
 - Implementing new Actions
 - Implementing new Financial Assets
-

Questions: What is a good method to test the server?