Coursework 3:

Hrishikesh Srinivasan

S1913948

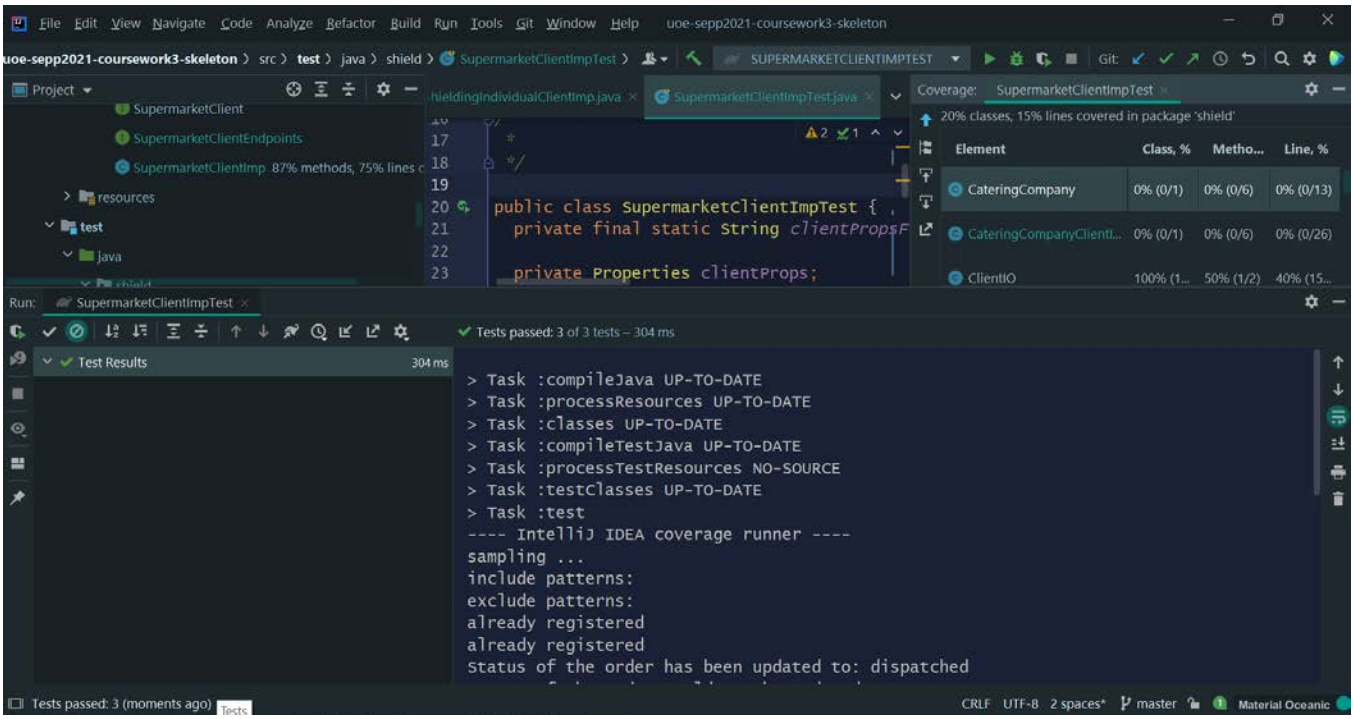
&

Jatin Soni

S1851999

4.5 Experience with the above tasks and with approaches, tools, agile processes

We organised our work in terms of dividing up the implementation of classes and tests amongst ourselves, meaning one individual would implement some of the classes along with their tests and the other would also do the same for the remaining classes and tests. We adapted a test first approach as tests were as important as the implementation itself. We made use of tools such as version control to ensure that each induvial were clear on what parts they were working on and also the IntelliJ test coverage tool since we did our coding on IntelliJ. We made use of Kanban as it mainly focuses on defining and visualizing a workflow, managing items in a workflow, and improving a workflow. The reason as to why we chose to use Kanban rather than other practises like Scrum and XP (Xtreme Programming) is because as it strategizes in optimizing value by optimizing flow meaning striving to find the right balance of effectiveness, efficiency, and predictability in how work gets done. We chose Kanban over XP since XP requires the development team to continuously provide feedback to developers, deliver valuable software and works on-site with the customer which was not really the focus for this coursework. Also, we chose version control due to the coursework itself being on GitHub and as it allowed us to manage and monitor the work being done through the use of commits. Using Kanban strategy worked well for us in the sense that we were able to make use of a Kanban Board which meant it was easy for us to keep track of the work that we had completed, work that was still in progress and the work that was still left to be done. An advantage of using Kanban was that since each of us knew as to what stuff to work it prevented team overburden meaning that not one individual had been burdened with too much work to do. Another benefit was that there was a continuous flow of development as tasks and issues were removed from the board once they were completed and others being added to the board as they were identified. I think the thing we would differently next time, developing a system is to try to make use of more tools like static analysis and bug testing tools like Jira.



This represents the usage of the testing with coverage tool which was run on the Supermarket Client Implementation Test file.

4.6 How the Client-Server Architecture would impact design

The Client-Server Architecture would impact our design from coursework 2 since there would be some changes that we would have to make in our class model in terms of adding in new classes to accommodate for the client classes that were given. This updated design would be a better choice due to the fact that it would be easier to scale in the sense that it would make it easier to increase the capacity of clients and servers individually. Another reason is because it would allow for easy maintenance to be carried out as responsibilities are dispersed amongst individual computers integrated across a network meaning, it would become easier to replace, repair, upgrade and relocate a server while the client would remain unaffected. Another reason is because there would be better security as servers would have better control access and resources to ensure that only authorized clients can access or manipulate data and server updates are administered effectively.

4.7 Quality attributes

Security OR privacy:

In the modern world of technology, systems are more prone to security and privacy attacks now than ever. Our system could also be vulnerable to security.

Man-in-the-middle attacks occur when attackers insert themselves into a two-party transaction and then try to filter and steal data. Man-in-the-middle is a great threat to our system since there is the interaction between the users and the system. When an attacker does this to our system, many users' data could be stolen and used without their permission. In order to prevent this, we can warn the users that they use the system only in a secure Wi-Fi network and ask them to try avoiding the usage of public Wi-Fi whenever possible.

Password attacks are also a huge threat to the users of our system because attackers can use various methods such as brute force and phishing to obtain the passwords of users and steal data. This is always a big risk to the system. We can prevent this by making the password and entry credentials stricter and we can also make the users aware of having difficult passwords and not sharing their passwords with anyone.

Reliability OR availability:

Software Reliability is the probability of failure-free software operation for a specified period of time in a specified environment. Some of the things that come under reliability are POFOD(Probability of failure on demand) and ROCOF(Rate of occurrence of failure). MTTF(Mean time to failure) and MTBF(Mean time between failures) are popular methods used to test the reliability of a system. For our system, we think it is best to use MTBF because MTBF is when the system can recover from failures. Every system is prone to failure and our system is no different. We should expect failures at times and for that, we can use MTBF to assess and test the reliability of our system. If the MTBF value is high, it will mean that our system is more reliable than when if the MTBF is low.

Usability:

The 2 principles of user interface design that we think would be very important for us to consider for this system would be Consistency and User diversity. In simple words, consistency is the term used to describe the similarity of operations in the system over different devices and applications. Consistency is very important for this system because we have a wide range of people using the system and they will be using it from many different devices. In order to address then in the user interface, we should make sure that the system design for all platforms remains consistent and the usability remains simple. The features and functionalities will be consistent throughout all platforms. User diversity is important for our system as a wide range of people will be using our system. This includes people with various disabilities. In order to address this in our user interface, we need to make sure that we keep the functionalities of the system simple. It shouldn't be unnecessarily complicated and we should try to make the user experience fairly straightforward across all platforms.

4.8 SE Self-Assessment

1. We think that we understood the task and used OOP in Java to a good extent. There were already implemented methods that we used and referred to in order to construct the required methods and classes.

2. We gave a lot of attention to this section because it is one of the important outcomes of this course. We spent a lot of time working on the design of the system. I think that we did a decent job implementing it and maybe we could have worked more on it.

3. We added comments wherever possible and where we felt that explanation was needed. Overall, we think that the quality in terms of readability and maintenance is decent and we put a good effort into this section. Maybe, we could've added more comments and documentation and we feel this would improve our code overall.

4. We made good use of JUnit 5 to construct a system-level test and unit tests for our methods and classes. We also worked well with JUnit 5 for constructing system-level tests. Overall, we are quite satisfied with how we used JUnit 5 for constructing tests.

5. We think that our choice of system-level tests was good as we were able to cover a good mixture of classes of inputs. The main strength of our solution is that we included quite a majority of tests for all our classes in the system-level tests to cover a variety of classes of inputs along with including tests for unusual or incorrect inputs and some edge cases as well. The main difficulty we had was that at first, we were not really sure on how to go about implementing the system level tests. We resolved this by reviewing back to the lectures again to get a better understanding and also checking online to gain a better understanding on how to go about doing this.

6. We feel that we had covered all the 8 use case scenarios of the code of the system-level tests. The main strength of our solution was that in our system tests we had included all the 8 use case scenarios and we had also made use of the test coverage tool in IntelliJ to test for each of the 8 cases. The main difficulty we faced was that we were struggling a bit to get the coverage tests to run due to problems in our IDE. To resolve this, we had to check up online on the issue that we were facing to find out on how to resolve it.

7. JUnit 5 was really useful for unit tests as well. We felt that more unit tests could have been written for the methods in the shielding individual class since it would have given us more guarantee of the right code. The tests helped us figure out many errors in the code and helped us rectify them.

8. We think that the unit tests that we implemented were sufficient in the sense that they covered a good mix of inputs for each of the classes. The main strength of our solution is that we implemented tests to cover for all the methods that we implemented in our classes. We also made sure to include tests to check for any possible edge cases and incorrect inputs for our classes.

9. We feel that we had covered all requirements that are the responsibility of each class and each code of each of the classes. The main strength of our solution was that we believe that we have covered all the requirements when implementing unit tests for our classes. We had also made use of the test coverage tool in IntelliJ to test for each method within our test classes.

10. We think that we made a good effort in terms of using tools to aid us in completing this coursework along with the agile practises. We also feel that we had a good understanding of in terms of how to go about using the tools. The only major difficulty that we faced was that at first we were a bit uncertain as to how to make correct use of some of the tools that we learnt about. To resolve this, we had to go back and re-watch the lectures where this was being discussed about.

11. We believe that the solution we gave for this part within our coursework was fair and honest. The main strength of our solution is that we feel that we had provided enough detail into the points we were making to justify what we were trying to get across. The only improvement that we would make is to try to make use of more tools that were given to us since we did not make use of that many tools.

12. We feel that the response we gave for this section was sufficient and satisfactory. We believe that the main strength of our solution for this part was that we went into enough detail when explaining these points to satisfy the reader. A possible improvement that we could make is to try to talk about more points to further justify our argument.

13. We used a lot of resources from lectures and online websites to understand what types of security attacks could possibly take place in our system. We understood certain security attacks that our system was most

vulnerable to and addressed and explained them. We explored the possible solutions to these attacks and explained them in our essay. Overall, we are quite happy with how we understood and presented the security section of part 4.7 of the essay.

14. The lectures on reliability and availability were really useful for this section. We also used some online resources to understand this section further. We explained the required terms and felt that the question was answered properly and included most of the points. Overall, we were very pleased with our effort in this section.

15. We felt that we did a pretty good and satisfactory job with this section overall. In order to get a deeper understanding of this section, we used the lectures and some websites as resources and answered the questions. The terms were explained well and we put in a lot of effort to explain the application part as well.

16. We analysed our effort in each section and realized our strengths and weaknesses. For the self-assessment section, we realized which areas to reflect upon and we explained how we went about doing every task. We tried our best to write about weaknesses wherever possible and overall, we are very happy with our effort in the self-assessment section.