**UML Design Modeling**

Tamara Whitfield

The University of Arizona Global Campus

Gen499: General Education Capstone

Prof. Joseph Rangitsch

August 5, 2024

**UML Design Modeling**

Software testing levels are the stages in the software development life cycle (SDLC) where testing is conducted. There are four levels as each area concentrates of a specific goal and is performed by a team of software testers or developers (Signh, 2022). Each stage runs a thorough process of the software in each level to check for defects in the early processes of development. This approach will enhance the overall quality of software by minimizing risks and satisfaction from clients.

**Component Testing**

Component testing is the first level of testing. It is the most fundamental testing level executed by software developers before it is handed off to the testing team. The main goal is to isolate each component and perform tests to display if each of the components are meeting requirements and having the expected outcome (Singh, 2022). Testing the usability of each component is an important aspect as it helps with detecting issues before processing. With usability testing, the behavioral evaluation is completed for the components. To perform this testing, the components must be in an independent and controllable state first. Then, each component must be user comprehensible (GeeksforGeeks, 2023). An example of component testing are the individual modules of a mobile testing application. If we look at the login functionality, there are different components that must be tested individually to ensure they function as intended. The login functionality grants the user access to enter their username and password into the application. Next, is focusing on a successful login that results in showing a successful transition to the website. Another functionality is an error as the user entered their username or password incorrectly. It should show an error stating that their information was entered incorrectly and to try again.

**Integration Testing**

Integration testing is the second level of testing. The test components are tested, and defects have been fixed. The individual components are integrated and tested in a group. The goal is verifying the combined components to ensure they work coherently as well as detecting defects in the interface and the interactions within the compose components (Singh, 2022). The top-down or bottom-up approaches are typically used when integrating the modules in the entire system. Gray-box testing is used in integration testing as it is the combination of white-box and black-box testing. An example is buying a plane ticket from the airline’s website. The users can see the flight details and payment information displayed when they are in the process of buying a ticket. However, the flight details and payment information are on two different systems. With integration testing, the airline’s website and payment processing system can be seen together on the webpage (Vijay, 2023).

**System Testing**

System testing is the third level of testing. It helps the team to detect bugs and defects while making sure the software meets every requirement. A specialized testing team is typically involved in this level. The goal is to ensure the product is running as smoothly as possible in the operating environment (Singh, 2022). The software is now seen from the customer and end-user’s perspective as it validates if the requirements are implemented in its entirety. Functions of the system result from the interaction of every system component, however, the only time they are present in the entire system can be observed and tested in there. An example of system testing is a tester testing a pet insurance website. In this process, end-to-end testing takes place as the tester goes through the process of buying an insurance policy, LPM, tag, adding additional pets, updating payment information on the user’s account, updating addresses, and receiving an order confirmation email and policy document (Vijay, 2023).

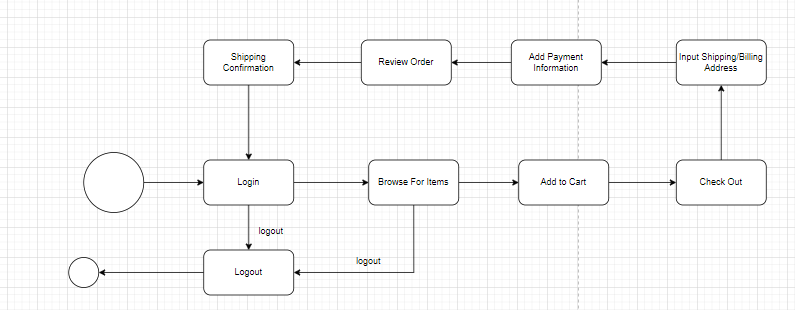
**Acceptance Testing**

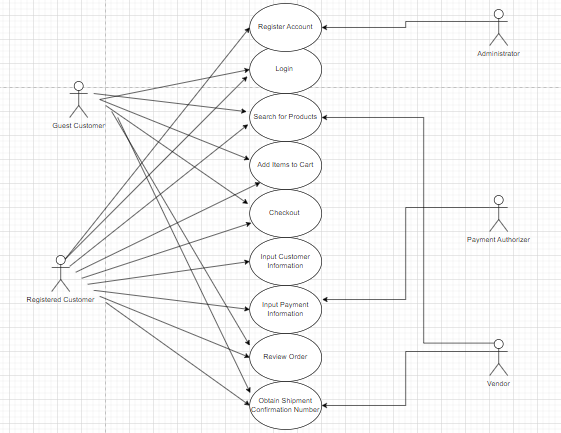
By now, all the test levels have represented testing activities under the producer’s responsibility. Now, they are executed before it can be shown to the end-user and customers. Before the software can be used in the real world, it must undergo the last test, acceptance testing. This type of testing determines if the software should be released to the public. The main goal is to ensure it complies with the stakeholder’s requirements and if it can be deployed (Singh, 2022). Customers can be involved in this testing process as they test the protype to ensure everything works from their end. The end-user or customer will go through real-life scenarios and can only accept it if all the features and functionalities work as expected. If they accept it, then the software goes into production (Vijay, 2023).

In conclusion, each software testing level is vital to strengthening the software’s reliability and quality. They develop a vigorous framework that safeguards the product against defects and issues as well as create reliance from clients and stakeholders. Eventually, they supply high-quality software products that meet the criteria and ensure long-term success.

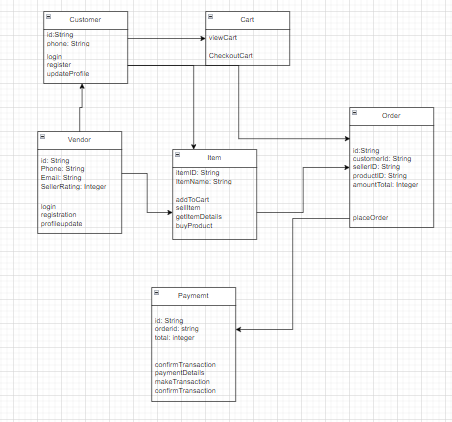
**UML Models**

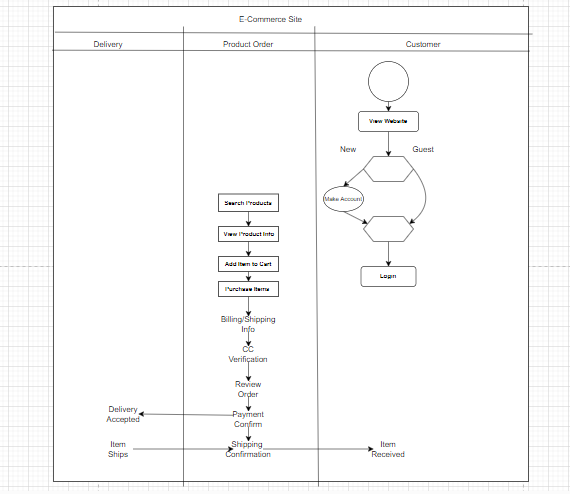
Unified Modeling Language (UML) is a fundamental approach to see the visualized design of the product. The development team, stakeholders, clients, and designers can better comprehend the architecture in the system through diagrams such as class, sequence, activity, state, and use case diagrams.

**Class Diagram**

**Sequence Diagram**

**Activity Diagram**

**State Diagram**

****

**Use Case Diagram**

***A diagram of a diagram

Description automatically generated***

**References:**

GeeksForGeeks. (2023). *Component Software Testing*

<https://www.geeksforgeeks.org/component-software-testing/>

Singh A. (2022). *Understanding the Different Levels of Testing in Software Development.* <https://www.shiksha.com/online-courses/articles/understanding-the-different-levels-of-testing-in-software-development/>

Vijay (2023). *Types Of Software Testing: Different Testing Types with Details.* <https://www.softwaretestinghelp.com/types-of-software-testing/#4_Acceptance_Testing>