UML Design Modeling

Jeremy Robinson

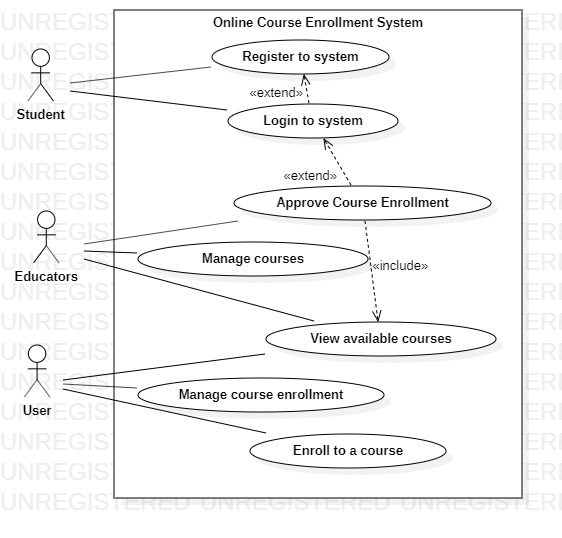
CST499: Capstone For Computer Software

Professor Elchouemi

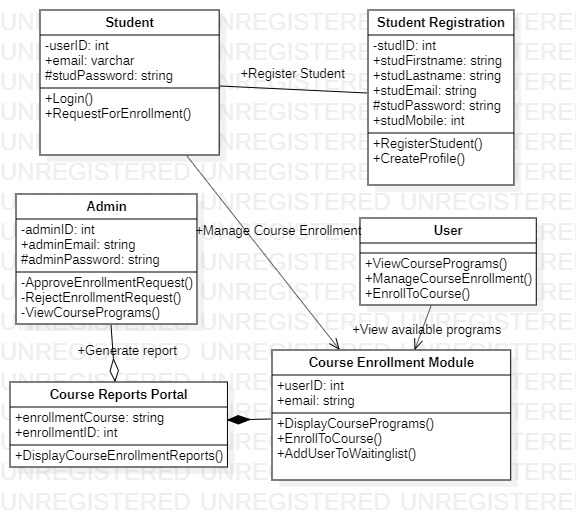
December 20, 2022

**UML Diagrams**

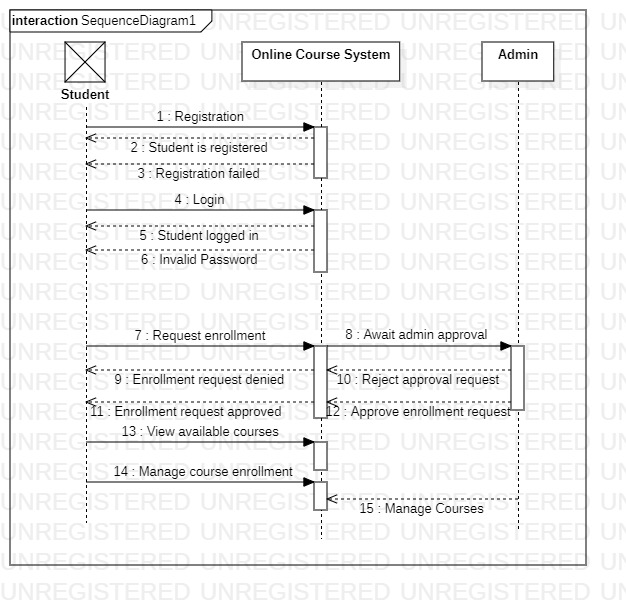
1. **Use Case Diagram**

****

1. **Class Diagram**

****

1. **Sequence Diagram**

****

**Component Testing**

Component testing, module testing or program testing is done after unit testing. In component testing, then testing is done on each component separately without integration into other components. Software is made of different components; thus, component testing performs level testing of each component individually. It is performed immediately after developers perform unit testing, and the build is then released to the testing team. Component testing can be categorized into two: CTIS and CTIL. CTIS (Component Testing in Small) can be done with or without separating the other components in the software or application under test (Murad, Badarneh, Qusef, & Almasalha, 2018). On the other hand, Component Testing at Large is done without isolating other components in the application under test or software. The main goal of component testing is to validate the output/input performance of the test object and ensure that the test objects are functioning according to the specifications required.

**Integration Testing**

Integration testing (I & T), also called Integration and Testing, is a phase of software testing where a combination of individual software modules takes place and are tested as a group. It is mainly conducted to elevate the component or system compliance with specific functional requirements. Integration testing is essential in IT and software development, especially when the requirements are tight and with dynamic requirements (Murad, Badarneh, Qusef, & Almasalha, 2018). It also helps eliminate common errors and issues that might have been missed during unit testing.

**System Testing**

The method through which a quality assurance (QA) crew assesses how the different aspects of software relate to one another in the entire integrated product or software is known as system testing.  System testing makes sure an application works as intended. This procedure, which functions as black box testing, focuses on an application's performance (Talley & Pathak, 2020). For instance, system testing may examine whether each type of user input results in the desired output throughout the software. A testing team uses system testing to ascertain if a test case conforms to each of the key important specifications and user experiences for a system. These particular test cases enable the developer to identify key flaws that impair an application's fundamental functionality before launch and establish the overall test coverage for an application. Each problem can be recorded and summarized per necessity by a QA team. System testing is performed, and the findings are reviewed by QA teams using various open-source and commercial technologies. These tools can create, manage, and automate tests or test cases. They may also provide aspects besides system testing, like the capability for requirement analysis. Open-source tools include Robotium and SmartBear's SoapUI, while commercial system testing tools include froglogic's Squish and Inflectra's SpiraTest (Talley & Pathak, 2020).

**Acceptance Testing**

After System Testing and before the system is made accessible for practical use, Acceptance Testing is the final stage of software testing. Software testing, called "acceptance testing," involves evaluating a system's usefulness (Taley, & Pathak, 2020). This test's main objective is to determine whether the system complies with customer specifications and whether it is suitable for deployment. User acceptability testing, business acceptance testing, contract acceptance testing, and regulations acceptance testing are the four different forms of acceptance testing. User acceptance testing is conducted to ascertain whether the product is operating appropriately for the user. Most of the requirements chosen for testing are those that customers frequently use.

Additionally, Business Acceptance Testing is done to verify whether or not the product satisfies the business's objectives. Due to rapidly changing market conditions and new technology, BAT's primary focus is on maximizing corporate earnings, which might be difficult and needs updating the current implementation, which adds to the budget (Taley, & Pathak, 2020). Contrarily, a contract requiring contract acceptance testing stipulates that when the product is online, within a specific timeframe, the acceptance test must be carried out, and pass all acceptance use cases. Regulations Acceptance Testing is performed to ascertain whether the product complies with the laws and rules established by the nation's government in which it is being made available. Although it might not have been deliberate, the firm will suffer.

References

Murad, G., Badarneh, A., Qusef, A., & Almasalha, F. (2018, July). Software testing techniques in it. In *2018 8th International conference on computer science and information technology (CSIT)* (pp. 17-21). IEEE.

Taley, D. S., & Pathak, B. (2020). Comprehensive Study of Software Testing Techniques and Strategies: A Review [J]. *International Journal of Engineering and Technical Research*, *9*(8).