**UML System Design: Student Scheduling**

Jonah Hannett

The University of Arizona Global Campus

CST499: Capstone for Computer Software Technology (CSF2535A)

Joseph Rangitsch

9/08/2025

**UML System Design: Student Scheduling**

In this paper, we will look at the UML models created for the student class scheduling system. After showing the UML models, we will discuss what component, integration, system, and acceptance testing would entail for this system. Modeling and testing will make development of the system much easier and hopefully produce fewer errors.

**UML Diagrams**

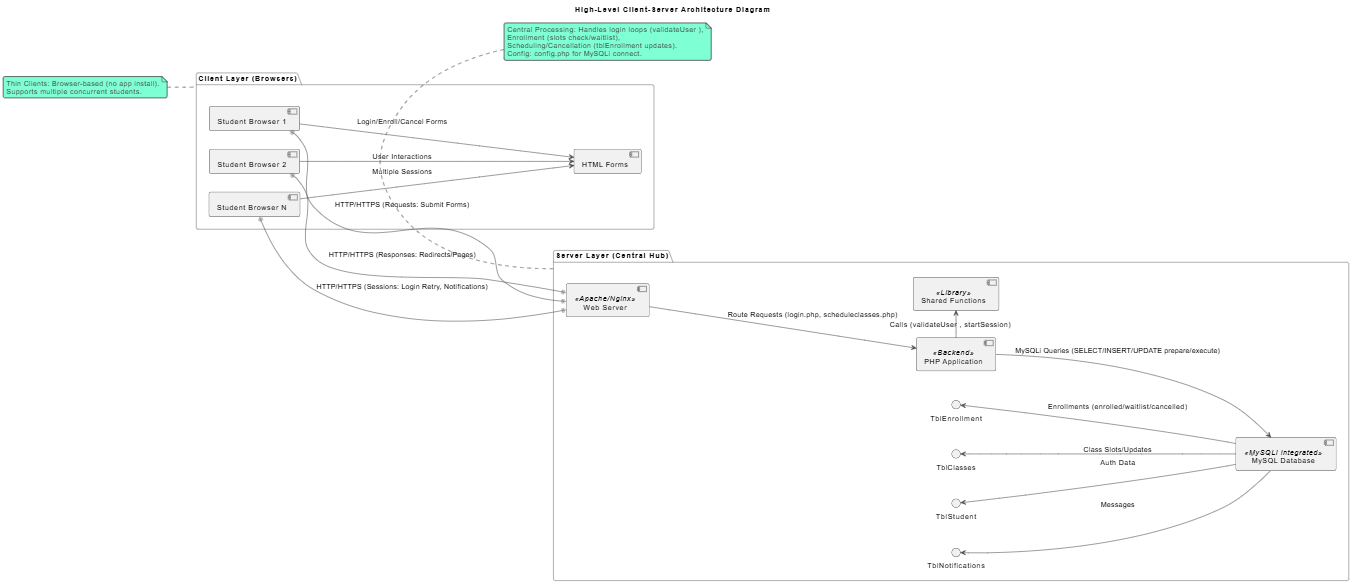


Figure . High-level system overview. This model shows the client-server architecture for the student scheduling system.

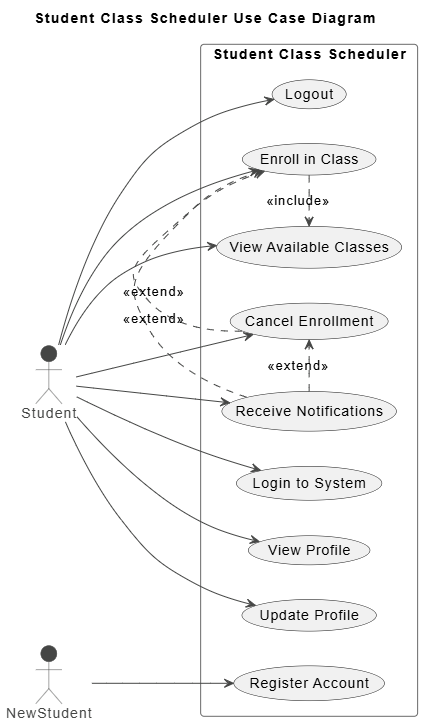


Figure . Use case diagram. This model includes the different use cases for this system broken up by existing and new students.

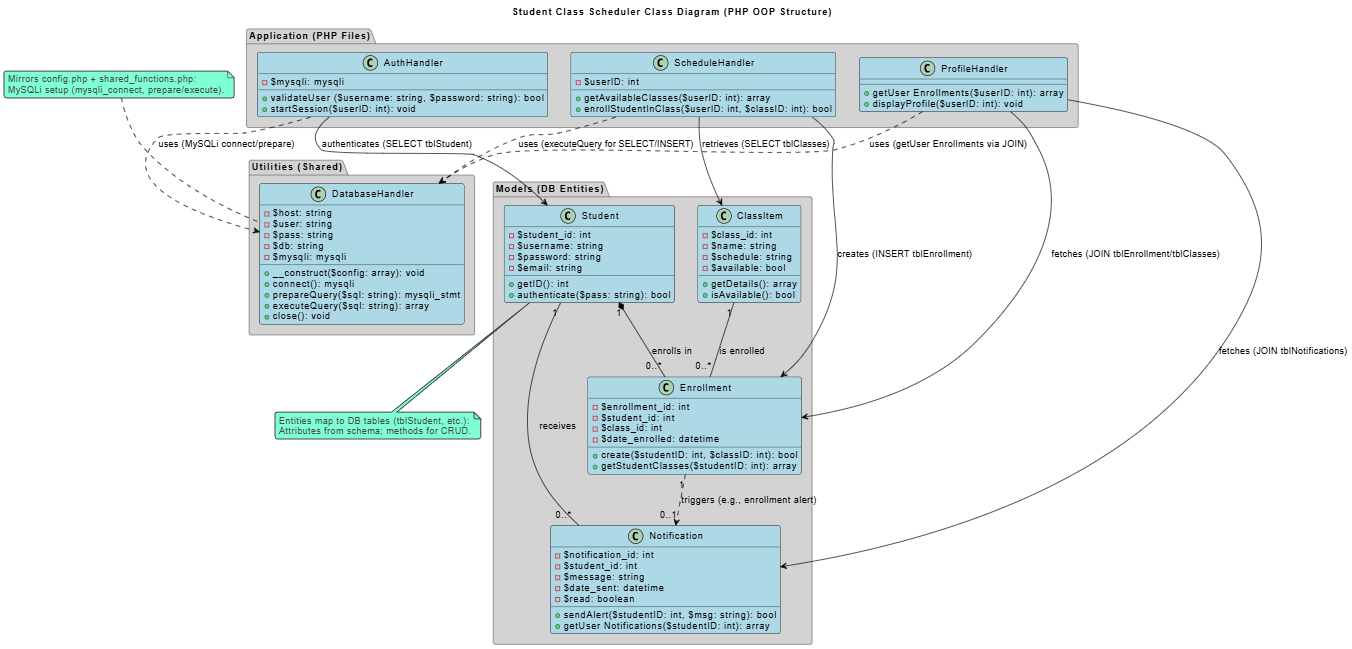


Figure . Class diagram. This model covers the different classes which were built for the scheduling software.

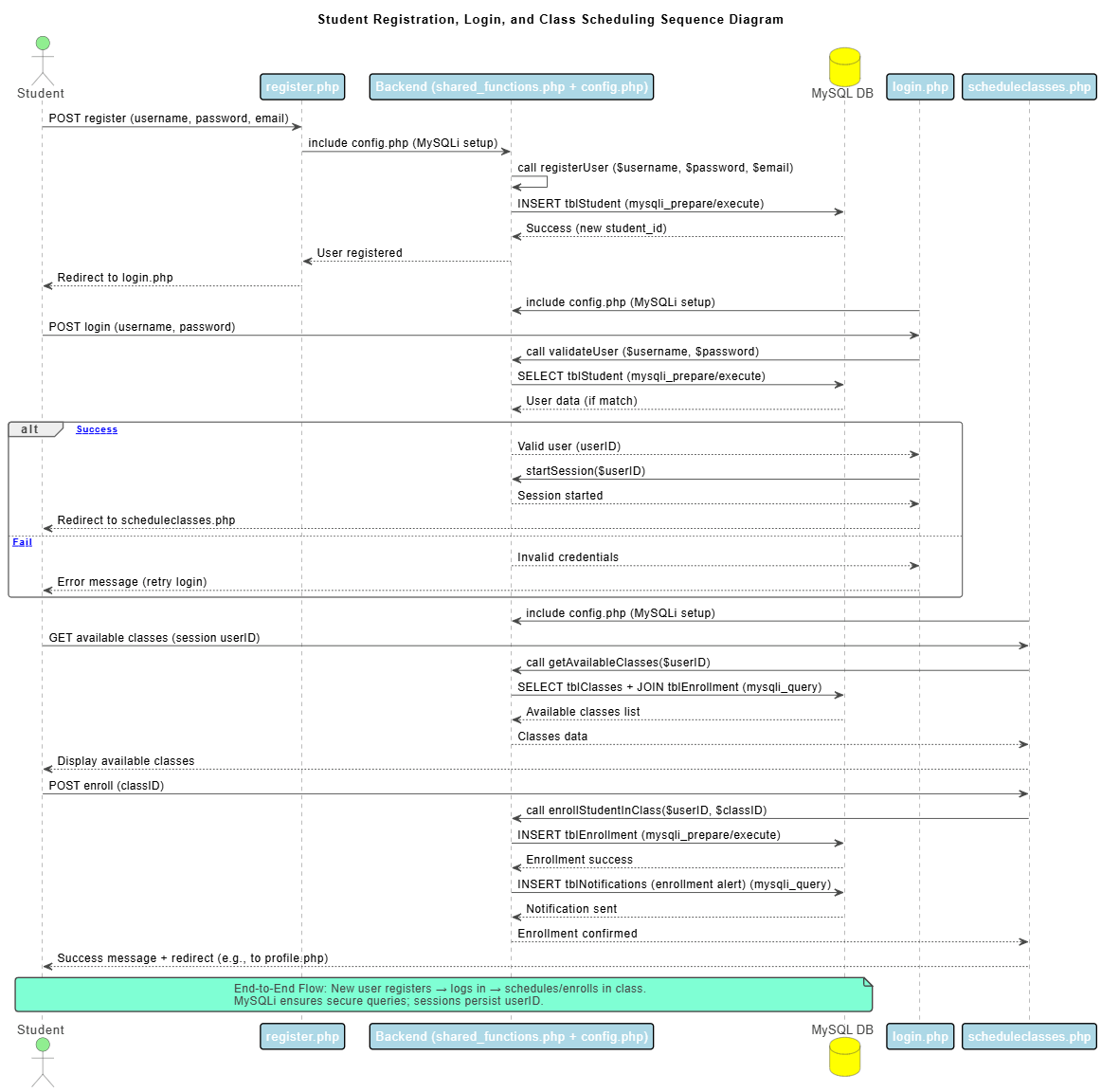


Figure . Sequence diagram. The model shown provides the sequence of events that take place between systems during registration, login, and scheduling a class.

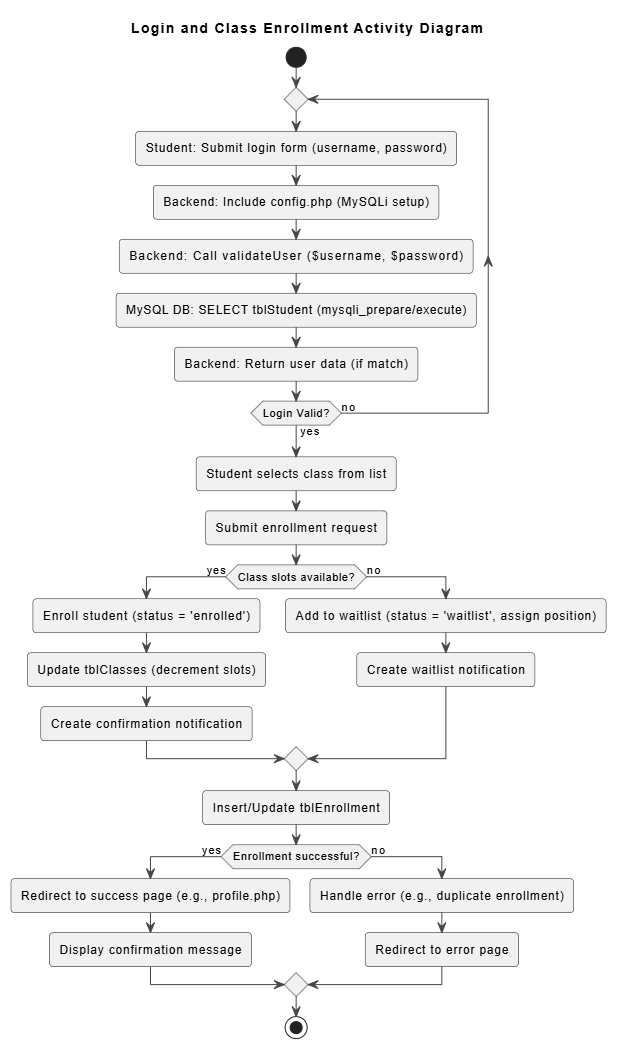


Figure . Activity diagram. Unified model showing the flow a user would take when logging into the website and then scheduling a class.

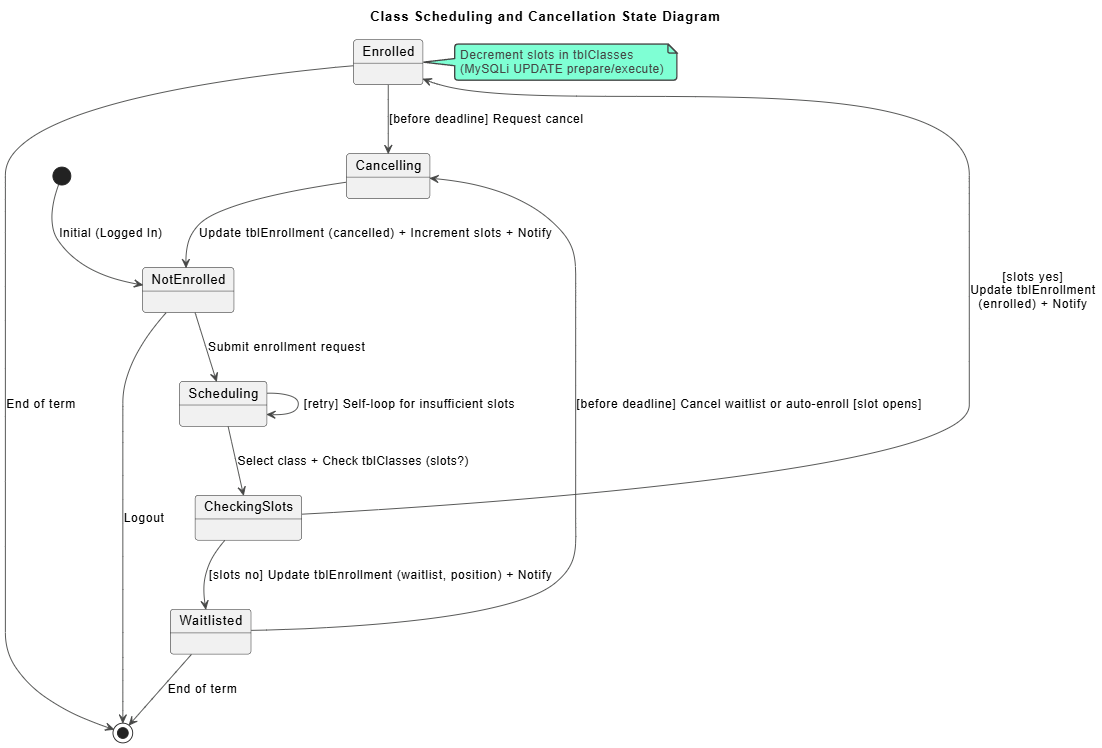


Figure . State diagram. This model shows the different states of the system that will be available for students scheduling or cancelling a class.

**Testing the System**

In this section we will discuss shortly component testing, integration testing, system testing, and acceptance testing. First, component testing is also sometimes referred to as unit testing, which adds some clarity since this type of testing covers individual software units and looks to verify proper output from each component and is used for early bug detection (BrowserStack, 2025). Integration testing comes after component testing, and its main purpose is to confirm that units that interface do so with the proper sequence and without error (GeeksforGeeks, 2025). After integrated components have been confirmed working, it is time for system testing which no longer focuses on individual items but the overall functionality using a “pass/fail” perspective (Powell & Smalley, 2025). Finally, acceptance testing gathers the customers and asks them to perform testing to confirm the system is working as they intended based on requirements (Agile Alliance, 2023). Once testing is complete, the system will be handed off to the client and will be maintained by whoever has agreed to apply increments to the software.

**References**

Agile Alliance. (2023, August 25). *What is acceptance testing?*. Agile Alliance | Promoting a more effective, humane, and sustainable way of working. https://agilealliance.org/glossary/acceptance-testing/

BrowserStack. (2025, May 29). *What is component testing? (with examples)*. <https://www.browserstack.com/guide/what-is-component-testing>

GeeksforGeeks. (2025b, July 20). *Integration testing - software engineering*. https://www.geeksforgeeks.org/software-testing/software-engineering-integration-testing/

Learn with Karl. (2024, January 24). *Create Your First UML Activity Diagram: StarUML Made Simple!*. YouTube. https://www.youtube.com/watch?v=IOyLgYxA28E

Lucid Software. (2018, August 27). *How to Make a UML Sequence Diagram*. YouTube. https://www.youtube.com/watch?v=pCK6prSq8aw

Lucid Software. (2023, August 10). *UML class diagrams*. YouTube. https://www.youtube.com/watch?v=6XrL5jXmTwM

Lucid Software. (2023b, September 20). *UML use case diagrams*. YouTube. https://www.youtube.com/watch?v=4emxjxonNRI

Powell, P., & Smalley, I. (2025, April 11). *What is system testing?*. IBM. https://www.ibm.com/think/topics/system-testing

TechVedas .learn. (2018, January 19). *State diagram with example*. YouTube. https://www.youtube.com/watch?v=L9UCsQxuWmw