

App Name: Spartan Sync
Andrew Dover, Aryan Gaur, Andrew Porasi
Repo: <https://github.com/adover06/CMPE131-LMS>

Problem Statement:

Current LMS tools like Canvas can be very overwhelming and confusing for students and instructors to use, leading to things such as missed assignments, announcements, and a bad user experience. A system that is simple to use, prioritizes tasks the same as a student would, and creates assignments intuitively for instructors is desperately needed.

Proposed Solution:

With Spartan Sync, there's an easy-to-use, more effective way of navigating coursework for the student, and an effective way of managing it for instructors. Instead of various different pages with cluttered home screens and ambiguous pages, SpartanSync is a lightweight, simple application that guides you to everything you need intuitively. Grades are displayed clearly, and announcements are separately shown and prioritized, all while retaining all necessary information. Removing the clutter minimizes the navigation friction.

In Milestones 2 and 3, we'll implement Flask-Login, SQLite, and WTForms. This will mean we'll have a functional login system, student dashboard, course page, assignment page, along with an assignment creation page for the instructors to publish assignments. Essentially, everything that's a UI prototype right now will be feature complete.

Key Users:

Students

- Needs to be able to access assignments, due dates, and grades
- Consistent and simple navigation across all courses

Instructors

- Clear organization of student submissions
- One-page, fast, easy assignment creation page

TAs

- Access to grading and reviewing submissions
- Limited access to modifying anything else regarding student grades and course materials

MVP

1. Log in form with WTForms
2. Student Dashboard
3. Instructor Dashboard

4. Assignment List
5. Assignment Details
6. Assignment Creation Form
7. Course Overview Cards
8. Consistent Navigation Layout
9. Basic User Roles (TA, Instructor, Student)
10. Announcement Sidebar
11. Announcement Details
12. Progress Indicators (Submitted, Not Submitted)
13. Rubric-Based Grading

Stretch Goals

1. Each instructor can define an estimated time taken for an assignment. Then, on a “Student-type” questionnaire on registration, a student can indicate if they are more likely to prefer starting on the work that takes the longest, or the shortest first. This way, the assignment to-do list will be prioritized by due date first, and preferred order by the duration of the assignment next.
2. A simple chat or email-style communication between students and instructors.
3. An AI time manager that summarizes and outputs a study plan based on the assignments due and the student questionnaire answered previously.

Success Criteria

1. All routes described, load and match the prototype designs without server errors. Each page will look like the UI screenshots given in M1.
2. Users can navigate through a complete workflow. This means students can log in, go to the dashboard, go to the course, assignment list, and assignment details. This means that instructors can log in, go to the dashboard, go to the course, and create an assignment form.
3. Navigation structure is consistent, complete, and intuitive. All the links work correctly and lead to the intended pages. No extra clutter is shown, and all course pages are extremely similar in format and use, with little to no navigational inconsistencies.