

Coursemeleon

Group 11 | Product Backlog

Bilal Areeb Baig, Zachary Bryant, Mitchell Demerly, Rujula Kapoor, Lauren Lum

Problem Statement

Course websites tend to be hosted on the platform an individual instructor is comfortable with. Users are expected to mentally save a new link to check back to, while site functionality and standardization are not maintained. We aim to fix this by creating a standard plug-n-play platform for instructors to host and maintain their course pages.

Background Information

The Problem

A student's learning experience is greatly improved by the accessibility of digital course resources. It follows that course websites would be hosted on a single standard, easily used, and maintained platform. Presently there is no modern solution for course sites at Purdue (and, we suspect, other schools as well). Professors are limited by their budgets, time, and knowledge to using out-of-date or poorly designed technologies, which lack expected functionality and accessibility.

Target Users

The app is being created primarily for educational instructors. The students using the resulting website(s) are a secondary group, as the intended outcome is for professors to more effectively create course websites for their students to use.

Limitations of Current Platforms

Several platforms exist that allow professors to create course websites. Learning management systems such as Blackboard and Canvas provide for this, but are somewhat limited in their range of customizability. Custom websites can be created on Google Docs, or website builders such as Squarespace, but these tools still require technical knowledge and overhead for setting up pages. Villanova University has a tool similar to ours, as it uses one standard template to serve as a starting point for all professor websites.

Coursemeleon overcomes these problems by creating an educationally-focused tool to for designing course websites. Instead of the stringent format enforced by learning management systems, Coursemeleon will offer modular components based on a student-focused template. No technical knowledge will be required to create the website,

as a user interface will simply ask for input regarding key course information, with additional options to import data from other sources. The time and cost overhead of creating a new course website is significantly reduced by the template approach.

Functional Requirements

Student-Focused Initiatives

1. As a student, I would like to view course information on a course site so I can stay up to date about the course.
2. As a student, I would like to view past course information so that I can gain an understanding of the course background.
3. As a student, I would like to see course information in an organized way so that I can effectively make use of the course website in a timely manner.
4. As a student, I would like to find view professor announcements on a single page so that I can know when an important announcement is made regarding the course.
5. As a student, I would like to have access to a class calendar so that I can see information about upcoming assignments and exams.
6. As a student, I would like to be able to export a course calendar onto a calendar service I use so that I can combine the course information with other activities.
7. As a student, I would like to have access to course resources so that I can have all of the necessary files and information for the course.
8. As a student, I would like to view information of instructors so that I can contact them if I need assistance.
9. As a student, I would like to see course statistics so that I can determine my standing in the course and approximate my final grade.
10. (If time allows) As a student, I would like to be able to communicate with others in the course so that I can speak with peers about course issues or assignments.
11. As a student, I would like to see a list of course information I recently viewed so that I can navigate the website more effectively.
12. As a student, I would like to be notified of recently updated course information so that I can be aware of any changes.
13. As a student, I would like to download course resources so that I can access resources offline.
14. As a student, I would like to see all relevant course links so that I do not have to navigate many websites.

Professor-Focused Initiatives

15. As a professor, I would like to be able to make quick changes to the course site so that students will see the most updated information.
16. As a professor, I would like to create an organized course site without worrying about formatting so that I can spend more time on course content.

17. As a professor, I would like to use a simple, step-by-step process to insert information onto my course site so that I can make a course site efficiently.
18. As a professor, I would like to display information about the course instructors so that students are available to contact them.
19. As a professor, I would like to import a course calendar for my students to see so that students will always be aware of assignments or important dates in my class.
20. As a professor, I would like to be able to share resources for my students to use so that they will have course information readily available.
21. As a professor, I would like to upload course statistics for my students so that sharing information is fast and intuitive.
22. As a professor, I would like to graphically display course statistics for my students so that they have a better grasp of my course content..
23. As a professor, I would like to upload course resources for users to download so that they can access the materials offline.
24. As a professor, I would like a way to announce information to my students so they are up-to-date with my course.
25. As a professor, I would like to decide what kind of information is included in my course site so that I can control what my students have access to.
26. As a professor, I would like to decide who can modify my course site so that course integrity is maintained.
27. As a professor, I would like to customize the layout of my course site as I see fit so that I can decide how information is displayed.
28. As a professor, I would like to add custom pages to my course site so that I have more flexibility with information and design.
29. As a professor, I would like to add custom functionality to my course site so that my course is not restricted by the template.
30. As a professor, I would like to answer questions and communicate with my students with a course chat so that my students are better informed about changes in the course.
31. As a professor, I would like to be able to reuse the website for the next year so that I can save time and not have to redo my work.
32. As a professor, I would like to restrict my website to only my students so that other people are not taking advantage of course materials.
33. As a professor, I would like to theme my website so that it matches my school brand guidelines.
34. As a professor, I would like to hide or publish my course page as I see fit so that I can finish customization at a later date.

Non-Functional Requirements

Usability

The interface should be easy to navigate and simple enough to enable quick start-up time for course pages. It should be flexible enough to seamlessly integrate additional functionality. The result generated from the Coursemeleon should have standardized navigation so that students are not confused about where relevant information is. We also want to maintain accessibility for various devices and user abilities.

Scalability

Coursemeleon should be able to store all customization of regular templates. Our biggest challenge will be handling the data uploaded such as PDFs, pictures, and other data. Most of the workload will be on the frontend (i.e. rendering is done by the user) using Vue.js. We will store the data in a NoSQL database hosted by Firebase to take advantage of the loosely enforced structure of data. Our backend will be written using Express.js, and will enable our RESTful API. The frontend will request data from this API. The server should perform its necessary duties and respond with data in under 1 second, and should be able to handle 10,000 simultaneous requests. Server uptime should be 24 hours, 7 days a week.

Security

The login system will be based on sending a random string of words as a password to the email. This reduces our risk of storing and encrypting passwords and relies on the security of the user's email service. Integrity of course pages will be maintained through this login by via a permissions system. Unless otherwise whitelisted, students will be able to access any course page.

Our backend will be secured using Helmet for Express, which blocks common vulnerabilities such as cross-site scripting. Our backend will also take care of sanitization to prevent SQL injection. All modify requests are subject to authentication by the server, and courses with whitelisting enabled will check GET requests as well.