CEN 4010 Principles of Software Engineering,

Spring and 2023

Milestone 1

Project Proposal and High-level description

Team name: T.E.A.M

Project name: ModuLearn

Group 5

Names and Emails

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Ethan Curtis - [ecurtis2019@fau.edu](mailto:ecurtis2019@fau.edu)

Kevin Horta - [hortak2022@fau.edu](mailto:hortak2022@fau.edu)

Bryan Cooke - [bcooke2020@fau.edu](mailto:bcooke2020@fau.edu)

Marco Parucho - [mparucho2021@fau.edu](mailto:mparucho2021@fau.edu)

Oliver Pennanen - [opennanen2018@fau.edu](mailto:opennanen2018@fau.edu)

History Revision dates

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Created: 3/2/2023

Executive Summary

**Product Name: ModuLearn**

ModuLearn is an e-learning website that allows for a unique, modular approach. The new modular design allows for a higher level of customizability and personalization. Each module can be created by institutions, teachers, and learning providers, and allow for students to understand different concepts without all the unnecessary baggage of other online learning platforms. The platform allows for students and teachers to customize their learning and teaching experience in an easy to use system.

With the new modular system we developed, students can learn at their own pace. Without the pressure of deadlines or assignments, the student can focus on learning what they are interested in. Students can explore, search, and select various different topics and select which they want to learn about, very easily.

ModuLearn is perfect for busy students and curious individuals who don’t have time for bulky, unnecessary courses, but still want to learn more about various subjects. No matter what age, or grade you are in, you can get a customizable learning experience without having to sit through boring lectures.

In conclusion, ModuLearn is an alternative and unique educational website that offers a modular, flexible, and different experience to online learning. The modularity features, and customization allows for a very pleasant and productive experience. If you are curious and interested in various subjects, but don’t have the time to commit, ModuLearn is a great platform for you.

**Competitive analysis**

| **Feature** | **ModuLearn** | **Khan Academy** | **Canvas** |
| --- | --- | --- | --- |
| Modular Topic Design | ✓ | ✓ | ✓ |
| Customizability | ✓ |  | ✓ |
| Variety of Topics | ✓ | ✓ | ✓ |
| Topic Experts | ✓ | ✓ | Varies |
| Flexibility | ✓ |  |  |
| User Cost | Free | Free | Higher |

**Data Definition**

Student - Object that will have as its attributes student id, array of classroom ids, gradebook, array of documents

Classroom - an object that will have as its attributes modules, students, teacher, TAs, syllabus, gradebook, calendar, assignments

Module - an object that will have as its attributes text, pointers to documents, pointers to assignments.

Gradebook - an object that will have as its attributes an array of dictionaries that list a grade, classroom id, and student id.

**Overview, scenarios, & use cases**

The majority of online learning environments give students adequate areas to interactively learn different ideas and concepts. However, it can be very confusing, and filled with unnecessary bloat and ads. Most students already have a basic understanding of many subjects, and do not necessarily need to review the material. With existing platforms, you may be forced to relearn all this material, wasting valuable time. ModuLearn fixes this.

Let’s say you are a busy student, or simply a curious person. If you have a homework assignment, test, or quiz coming up, it may take a lot of time to find accurate, easy to digest, educational content. With ModuLearn, you can quickly find useful information that will help you ace your assignments. Easy to view, search, read, and consume modules let you personalize *your* education.

Another scenario/use case is from the perspective of a teacher. When managing a class, not every student will understand the lesson when taught. A centralized and customizable location to include learning material, besides a textbook, makes it easier for the student to learn what is taught. Students can track their progress, challenge themselves, and help further their educational goals and achievements.

**High-level functional requirements**

There are two “modes” that a user can be in:

* Student
* Teacher

Students should be able to view courses and topics set by instructors and view their gradebook.

The courses list the topics in the order set by the instructor; strictly linear. Some topics are required for students to complete; they cannot progress to later topics until clearing required earlier ones. However, topics should also be optional.

The gradebook should be a summary of assignment grades in classes that the student is currently enrolled in.

Teachers will be able to create courses and topics for groups of students and have their own gradebook for viewing student’s grades.

A course contains topics, and topics include lectures, readings, assignments, quizzes, etc.

The gradebook will be in the form of a set of tables (for each course) showing various assignments, quizzes, etc, against students.

UI requirements

* Home page should be populated with enrolled class(es) with a clickable link to assignments due the current day. The link should be titled the same as the assignment, and be either **bold** or colored red.
* Every student should get a single popup that lists all assignments due the current day. This is a default option (can be disabled).
* Students’ homepage has a list of manually bookmarked modules, assignments, discussions etc.

**Non-functional requirements**

* LOGIN && USER AUTHENTICATION
  + When a teacher creates / edits / deletes a topic / course, a message must be sent to the server containing the data content
    - This message must be secure and encrypted (requires auth)
    - This message will be decrypted on the server, then from which its content will be uploaded to the database
  + When a student logs in, a request to the database must be made
    - This request must be secure (requires auth)
    - The returned message must be encrypted for privacy
* Database
  + The database must be organized such that the content of topics can be stored (i.e. HTML presets must be made, customized features then saved)
  + The database must store lots of text
  + Database should not contain video (too expensive)
  + Must store preferences for users

**Team**

Ethan Curtis - Product owner, back end

Kevin Horta - Scrum master, front end

Bryan Cooke - Front end

Marco Parucho - Back end

Oliver Pennanen - Github master, back end

**Checklist**

a) Team decided on basic means of communications: DONE

b) Team found a time slot to meet outside of the class: DONE

c) Front and back end team leads chosen: ON TRACK

d) Github master chosen: DONE

e) Team ready and able to use the chosen back and front-end frameworks : ON TRACK

f) Skills of each team member defined and known to all: DONE

g) Team lead ensured that all team members read the final M1 and agree/understand it before submission : DONE