

LearnIt Project Report

Brandon Kelley, Hunter Schattie(Group Leader), Serghei Berezovschi

Capstone Project [Spring, 2022]

The University of West Florida

Date

CIS4595 & Capstone Project

Dr. Bernd Owsnicki-Klewe

Abstract

The goal of this document is to describe the process and the development of the project that was done throughout the project's timeline. In addition, the document would compare the initial to the final requirements, the software used for the project and the reason behind the use of the tool(s).

Table of Contents

Table of Contents

Executive Summary	2
Table of Contents	3
1 Executive Summary	4
2 Final Requirements and comparison with initial requirements	5
3 Final timeline and comparison with the initial timeline	6
4 Project results compared with expectations	7
5 Software evaluation	8
6 Word to be done	9

1 Executive Summary

LearnIt is an application where any user can create and share information through what would be called “courses.” These courses can be made by any registered user with an account on the platform and range from software, business, science, engineering, or any other topic the user decides. The goal of this project is to allow anyone with the desire to learn to have access to any information about that person for their particular needs.

2 Final Requirements and Comparison with Initial Requirements

Initial Requirements:

- Login
- Register
- Create a course
- Join a course
- Rate a course
- Forum discussion board

Final Requirements:

- Login
- Register
- Create a course
- Join a course
- Rate a course
- Forum discussion board

As the project progressed, the initial and the final requirements for the project have stayed the same. There were no additional features that needed to be implemented.

3 Final timeline and comparison with the Initial Timeline

Initial timeline:

- Week 5. Presentation 1
- Week 5-7. UI Prototype. Login/Account feature
- Week 8-10. Course Creation/Course Update. Group Report 1, Presentation 2
- Week 11-13. Course Forums. Group Report 2-4
- Week 14: Course Rating. Final Deliverables
- Week 16: Final Presentations.

Final timeline:

- Week 5. Presentation 1
- Week 5-8. UI Prototype. Login/Account feature
- Week 9-11. Course Creation/Course Update. Group Report 1, Presentation 2
- Week 12-14. Course Forums. Group Report 2-4
- Week 15: Course Rating. Final Deliverables
- Week 16: Final Presentations.

For the most part, our assumed timeline created at the beginning of the project was followed. While the group was working on our first features we had to extend the expected time it would take by one week. This pushed all other deadlines back one week and gave us only one week to work on the course rating feature and fix any last minute issues.

4 Project results compared with Expectations

Expectations:

- Registration
- Login/Logout
- Create a course
- Join a course
- Forum
- Rating system

Final Results:

- Registration
- Login/Logout
- Create a course
- Join a course
- Forum

With the exception of the rating system all the project's features have been implemented.

5 Software Evaluations

MongoDB Atlas

We utilized MongoDB atlas for multiple reasons. For one, it is free and very easy to setup compared to other databases. There are also many security features built-in to MongoDB Atlas the alleviate having to worry about implementing them ourselves. Finally, MongoDB Atlas is cloud based, so none of us had to host the database or find an external provider.

Express / React

We chose to use Express/React stack due mainly to the fact that the development team had experience working with them. Our goal was to eliminate the learning curve as much as possible in order to speed up the development process. At the same time, React/Express are widely utilized in the software industry and becoming proficient in using them is an important step in building the skills necessary to succeed in the industry.

Testing Framework

We chose to use the Jest testing framework since it was one of the mainstream and widely used testing frameworks used within a Javascript tech stack. Jest allowed the use of integration testing and end-to-end testing. The integration testing that was used was to test the connection between the backend and the MongoDB collection that we have established. For end-to-end testing, on top of using Jest, puppeteer was used to help create user test runners that would open a browser that the test chooses and goes through the use case test that is established. Since there were no additional browsers to test the default is chrome so as a result the application is only tested through a chrome browser.

6 Work to be done

Work that needs to be done:

- Course rating system
- Finish up course forums
- Secure socket between frontend and backend if ever gone into production
- More Elaborate End-to-End test cases
- Deployment