

# TutorialHeaven

by

Brendan Powers, Alex Johnson, Travis Barton

Submitted to  
The Faculty of the School of Information Technology  
in Partial Fulfillment of the Requirements for  
the Degree of Bachelor of Science  
in Information Technology

©Copyright 2021 Brendan Powers, Alex Johnson, Travis Barton

The authors grant to the School of Information of Technology permission to reproduce and distribute copies of this document in whole or in part.

Brendan Powers

Brendan Powers

Alex Johnson

Alex Johnson

Travis Barton

Travis Barton

Yahya Gilany

Advisor Yahya Gilany

4/11/2021

Date

4/11/2021

Date

4/11/2021

Date

4/11/2021

Date

University of Cincinnati  
College of  
Education, Criminal Justice, and Human Services

April 2021

# Table of Contents

|                                            |    |
|--------------------------------------------|----|
| <b>Abstract</b>                            | 1  |
| <b>Introduction</b>                        | 2  |
| <b>Problem:</b>                            | 2  |
| <b>Solution:</b>                           | 3  |
| <b>Project Goals:</b>                      | 3  |
| <b>Overview:</b>                           | 4  |
| <b>Discussion</b>                          | 4  |
| <b>Project Concept:</b>                    | 4  |
| <b>Design Objectives:</b>                  | 5  |
| <b>Methodology and Technical Approach:</b> | 5  |
| <b>Web Application:</b>                    | 6  |
| <b>Database:</b>                           | 6  |
| <b>User Profile:</b>                       | 6  |
| <b>Table 1: Web Developer:</b>             | 6  |
| <b>Table 2: Learner:</b>                   | 7  |
| <b>Table 3: Teacher:</b>                   | 8  |
| <b>Use Case Diagram:</b>                   | 9  |
| <b>Technical Architecture:</b>             | 10 |
| <b>Testing:</b>                            | 11 |
| <b>Testing Methodology:</b>                | 11 |
| <b>Scope of Testing:</b>                   | 11 |
| <b>Objectives:</b>                         | 12 |
| <b>Test Procedures:</b>                    | 12 |
| <b>Testing Review:</b>                     | 13 |
| <b>Budget:</b>                             | 14 |
| <b>Project Timeline:</b>                   | 14 |
| <b>Problems Encountered:</b>               | 15 |
| <b>Recommendations for Improvement:</b>    | 16 |
| <b>Conclusion</b>                          | 17 |
| <b>Lessons Learned:</b>                    | 17 |
| <b>Skills Built and Acquired:</b>          | 17 |
| <b>References:</b>                         | 19 |
| <b>Appendixes:</b>                         | 20 |
| <b>Appendix A: Code</b>                    | 20 |

# List of Illustrations

## Tables

| <u>No.</u> |                    | <u>Page</u> |
|------------|--------------------|-------------|
| Table 1.   | Web Developer..... | 7           |
| Table 2.   | Learner.....       | 7           |
| Table 3.   | Teacher.....       | 8           |
| Table 4.   | Timeline.....      | 15          |

## Figures

| <u>No.</u> |                         | <u>Page</u> |
|------------|-------------------------|-------------|
| Figure 1.  | Use Case Diagram.....   | 9           |
| Figure 2.  | Website Deployment..... | 10          |
| Figure 3.  | Testing Results.....    | 13          |
| Figure 4.  | Budget.....             | 14          |

## **Abstract**

TutorialHeaven is an online learning solution to provide users a more personalized learning experience over the industry's leading online asynchronous platforms. With the combination of an asynchronous approach, a feedback loop that points the learner to their weakest subject material, the ability for course creators to make their course along with assessments in one platform, TutorialHeaven can greatly improve the experience both for the learner and instructor. The one-size-fits-all approach of asynchronous learning content neglects student's performance and ignores the effectiveness of conventional teaching practices, which leads to difficulties in knowledge retention and skill acquisition. These issues will be addressed by providing learners automatic feedback on their assessments, along with real communication channels for instructors and their course participants to engage with each other.

# Introduction

## **Problem:**

2019 market reports on the online learning industry showed a revenue of 200 billion dollars, with projected revenues in 2026 being 375 billion dollars.<sup>1</sup> When online learning is becoming ever more prevalent, people highly value the ability to learn online. Factors such as cost-effectiveness as compared to in-person instruction, and a lower time investment, have made this industry so appealing. The current leaders in the market follow a model that doesn't resemble conventional teaching practices. They resemble more of a video hosting service for educational content.

Research has shown conclusively the meaningfulness of interaction between students and instructors for improving student outcomes and satisfaction.<sup>2</sup> Based on our research there is currently not a service that provides the instructor student relationship that facilitates effective learning. The lack of interaction can lead to low motivation for students on asynchronous platforms causing them to give up on the desired learning.

The use of low-stakes quizzing has shown better results for memory retention over prolonged periods.<sup>3</sup> The usage of periodic assessments has been a staple for educators, but elearning platforms fail to provide the necessary feedback to effectively direct the user in the right direction. These failures can compound off one another, making the students feel like their learning is stagnating.<sup>4</sup> When in fact the issue is with the little effort these platforms make to highlight the user's problem areas.

## **Solution:**

TutorialHeaven is an online learning platform that allows instructors to create content that aligns with conventional learning styles. With the ability to create courses, assessments, forums, and view statistics users will be able to have a more personalized learning experience. We also implement a results-based analysis to tailor the educational experience to the student's actual weaknesses and strengths.

Our goal is a data-driven, results-based, educational experience that is capable of adapting to a student's performance as well as their individual preferences and needs. Making our platform more engaging and personalized than our competitors

To accomplish this goal our team created a course creation suite that will be used to create content and assessments. The assessments being the main learning reinforcement point for the content, providing feedback after the assessment is taken. This feedback will then be used by the user to reinforce information that may not have understood completely. TutorialHeaven also provides statistics on the users and courses for the users to better understand their performance and their weak spots. Forums for each course provide communication avenues between students and teachers.

## **Project Goals:**

The goal for TutorialHeaven is to create an online learning service that provides a more personalized and engaging experience for students and to provide the framework for teachers to

create courses that provide that experience for the students. This includes developing a web application that has the ability to create assessments, and give feedback based on those assessments that can direct the student to information they need to review.

## **Overview:**

The rest of this report covers the process our group went through to complete the project. The report includes the following sections: design objectives, methodology, testing, budget, timeline, problems encountered, lessons learned, and future recommendations.

# **Discussion**

## **Project Concept:**

Tutorial Heaven spawned from our group's collective disappointment trying to learn programming from our classes and online. Our countless attempts on online learning platforms got us stuck in an experience we coined, "tutorial hell". A team member, Brendan started researching comprehension strategies and cognitive psychology to be implemented into TutorialHeaven.

The team formed because all three of us are in the Information Technology major and have been friends since freshman year. We had been planning on doing a group together for senior design, and when brainstorming ideas together it became apparent that we all shared an interest in the subject of online learning. We drew upon our personal experience initially, but

further research online brought forward entire communities who discuss “tutorial hell” and the shortcomings of learning platforms online.

## **Design Objectives:**

Our objective is to create a platform where people can come to share their knowledge and also learn. Creating a platform that provides an asynchronous model, assessment oriented feedback, and the tooling for creators to make their best content.

Our first goal was focused on an assessment creation tool for creators. That assessment would auto-grade the users and refer them back to content that needs reviewed. We built on this goal by trying to keep the tooling open to any type of content so the learning platform could host a plethora of content types.

Creating an avenue for communication between students and teachers is another one of our goals. To satisfy this objective we created a forum system for classes where teachers and students can communicate and discuss topics of the course.

Another goal of ours was to allow the users to view statistics about their courses they are in and are instructors of. We created statistics pages for users and instructors of courses that display information about their performance in that course.

## **Methodology and Technical Approach:**

In this section we will go into the technical aspects of our project. Going over the tooling we use for development, our user profiles, use case diagrams, budget, timeline, and problems that we have encountered along the way.



## **Web Application:**

The web application is built using vue.js and node.js. Vue.js being a fully-fledged frontend framework gives us the tools we need to build a reactive and efficient frontend. Node.js is the goto backend for JavaScript, allowing us to use one programming language across our entire development stack. With these tools, we can create a website that meets our goals and objectives. -

## **Database:**

We needed a place to store our user information so we could use it in the future, this required a database. We decided on using MongoDB, Mongo is a NoSQL document oriented database. Being so closely tied to JavaScript and our familiarity with it we concluded it was our best choice. The database is used to hold the data for Users and Courses. Through our Node.js backend, we interact with the database creating and changing data as requested.

## **User Profile:**

User profiles is a collection of data associated with a specific user. Our application had three user profiles: Web Developer, Learner, and Teacher. Each one of these profiles define the roles that the user plays in the application and the experience and use of those roles. A specific user can be in more than one profile.

### **Table 1: Web Developer:**

The first user profile is the Web Developer. This user will be maintaining and updating the website on a regular basis. Interacting with applications such as AWS and Github while also

programming in JavaScript using Node and Vue.js. To fit this profile the user should have experience being a full stack developer and history using Github and AWS.

| User Profile: Web Developer                                                                                                                                                                                                                   |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Application:</b><br>Amazon web services(AWS), Github, MongoDB, Node.js, Express.js, Vue.js, Linux command line.                                                                                                                            |
| <b>Software Experience:</b><br>The user should have experience doing full stack development software especially with the MEVN stack (Mongo,Express,Vue,Node). The user should also have experience with Github and AWS to deploy the website. |
| <b>Task Experience:</b><br>Using the previously mentioned software in the creation of website updates and maintenance                                                                                                                         |
| <b>Frequency Use:</b><br>The user will be interacting with the website almost daily. Monitoring the website and user activity and pushing and developing new updates.                                                                         |
| <b>Key Interface Design Requirements that the Profile Suggests:</b><br>The user will need to be familiar with the CI/CD workflows in order to provide updates and maintain the website.                                                       |

## Table 2: Learner:

The second user profile is the learner this user is the person using our application to learn new skills. This user should have experience using the internet and websites, they will be interacting with forms, taking assessments and logging in and updating their profile information.

| User Profile: Learner                                                                                                                                        |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Application:</b><br>The TutorialHeaven website                                                                                                            |
| <b>Software Experience:</b><br>The user should be familiar with how websites and the internet works.                                                         |
| <b>Task Experience:</b><br>Basic website interaction, entering into forms to complete their assessments. Knowing how to login and update their user profile. |
| <b>Frequency Use:</b><br>The user would only interact with the application as instructed, there will be a cap on                                             |

**Key Interface Design Requirements that the Profile Suggests:**

The website needs to be easily accessible and easy to navigate. The user will also need a way to view a list of available courses and the courses they are subscribed to.

**Table 3: Teacher:**

The third user profile is the teacher, this user is the person creating the content on the application. They should have experience with basic web interaction and also know about video, image, and file formats.

| <b>User Profile: Teacher</b>                                        |                                                                                                                                                       |
|---------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Application:</b>                                                 | The TutorialHeaven website                                                                                                                            |
| <b>Software Experience:</b>                                         | The user should be familiar with how websites and the internet works.                                                                                 |
| <b>Task Experience:</b>                                             | Basic website interaction. Experience with video, image, and file formats.                                                                            |
| <b>Frequency Use:</b>                                               | Whenever the user feels the need to access their course content, Provide feedback, or answer current learner questions.                               |
| <b>Key Interface Design Requirements that the Profile Suggests:</b> | The website needs to be easily accessible and easy to navigate. The user will also need a way to create content on the website that is easy to learn. |

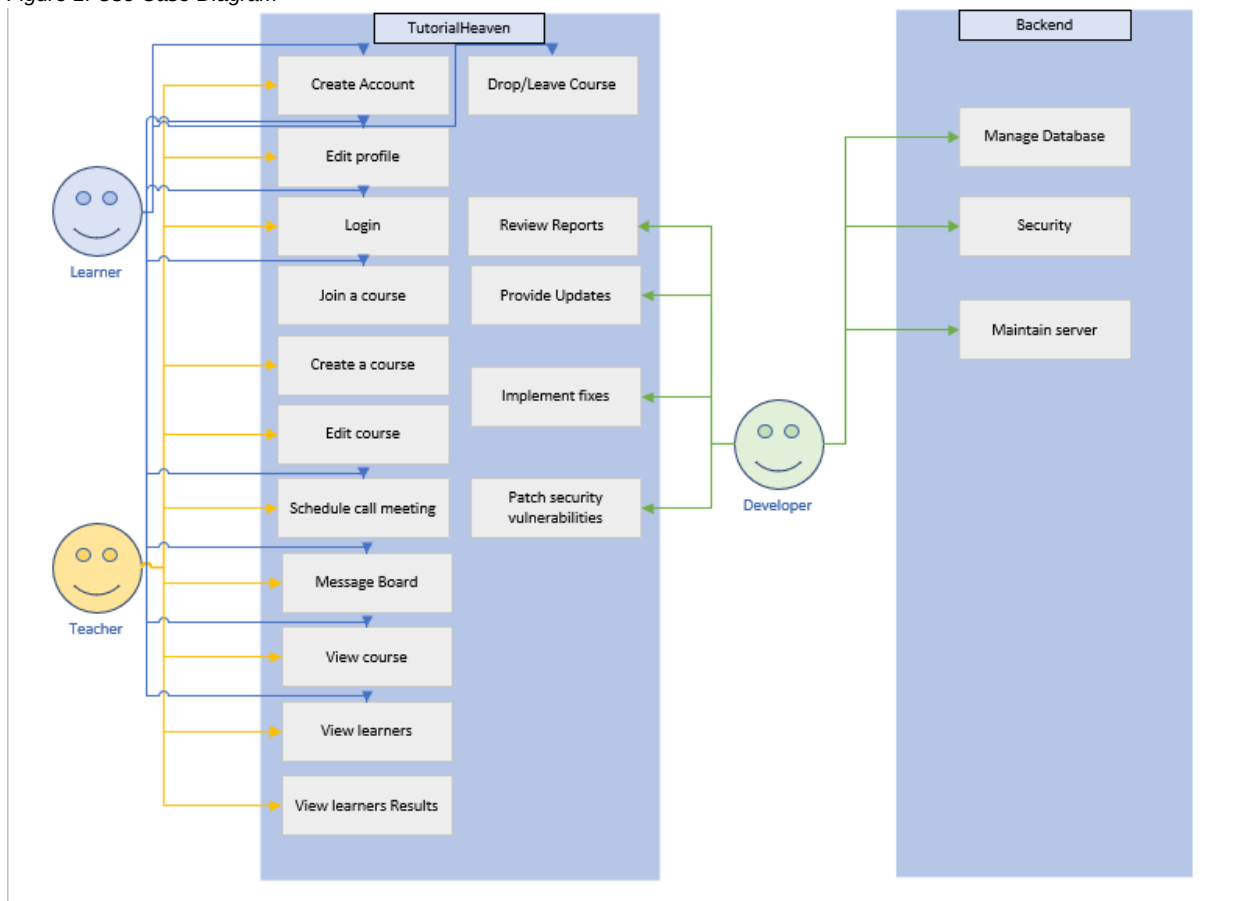
## Use Case Diagram:

Our use case diagram shows each part of our application that our users interact with. The user type will determine which components they will interact with.

**Figure 1:**

This figure shows our use case diagram. The color coordinated arrows show which component that specific user will interact with. Our components are separated between TutorialHeaven the forward facing web application and the backend.

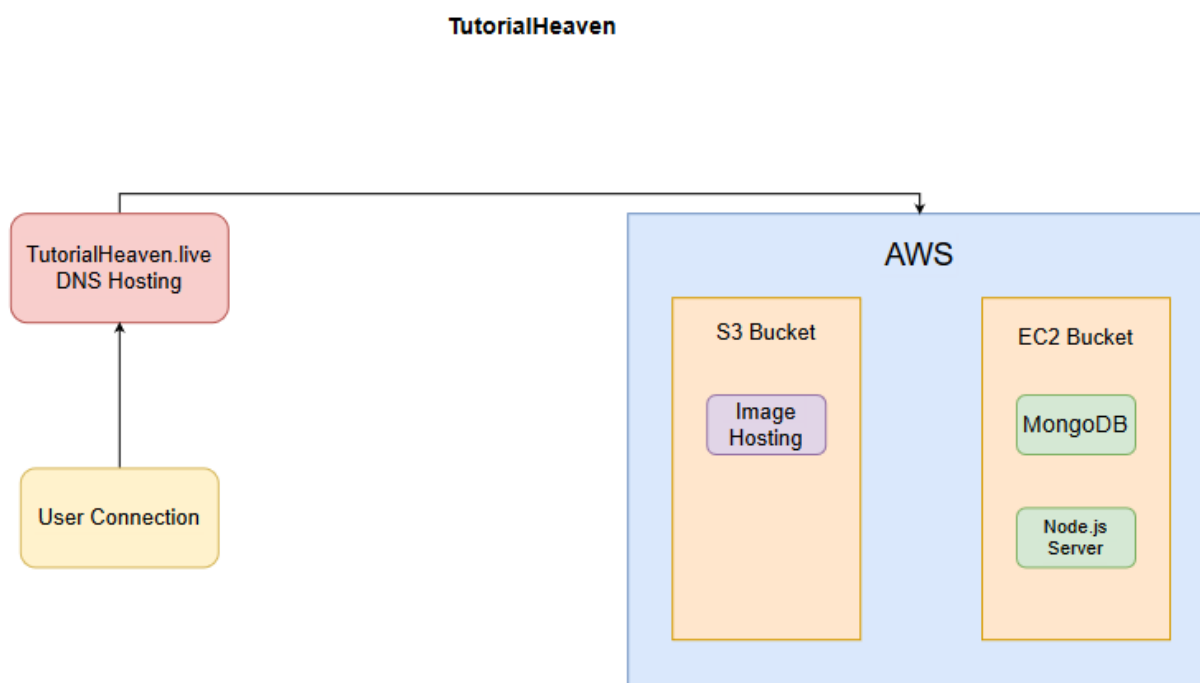
Figure 1: Use Case Diagram



## Technical Architecture:

Our project's technical architecture exists primarily on Amazon Web Services (AWS) instances. Below is a diagram of our architecture.

Figure 2: Website Deployment



Our DNS provider is Names.com using TutorialHeaven.live as our domain name. In AWS we have an S3 and EC2 bucket. Our S3 bucket is used to host our images for the user's profile, forum images, and course images. Our website is running on Node in the EC2 bucket along with our mongo database.

Our Node server is running the Express web server framework serving up our Vue web application. The node server is also handling api requests and database interactions using mongoose.

## **Testing:**

In the testing section, the team shows feedback from users that helped locate issues or bottlenecks to the user experience. The focus of our final testing on TutorialHeaven is User Acceptance Testing (UAT). UAT is one of the final steps in testing and aided the team in finding errors in the site that might have gone unnoticed during development. Throughout this report the method of testing used along with changes made will be broken down to explain how UAT affects TutorialHeaven on its way to production.

## **Testing Methodology:**

The team's main testing method before getting to UAT has been integration testing. Integration testing has worked to make sure that our core functions operate as expected. We chose to use integration testing because it was easy to test as changes were made instead of testing after one big coding merge.

UAT is going to be used to get feedback on the user experience along with finding bugs that the team might have overlooked. The team's approach to UAT is going to be giving users some use case scenarios and letting them work on them giving us feedback at the end. This hands-off approach should benefit us the most in terms of testing the user experience as if the user is a new person visiting the website for the first time.

## **Scope of Testing:**

These are the use cases users will be asked to complete and provide feedback on.

- Create a course with one content module and one assessment module

- Join a course via the course catalog
- View the joined courses content
- Take the joined courses assessment
- Make a post on the course forum
- Like a post on the course forum
- Look at the users stats page for detailed information

We chose to include these use cases because they best represent the core features of our website that we want to be fleshed out by the time of the IT expo. This will allow us to improve on them and/or fix bugs if the test results require this.

### **Objectives:**

- All major site functions tested
- Any user feedback will be addressed for fixes/changes
- All bugs encountered need to be fixed before the IT expo
- Both instructor and student user profiles have their use cases tested
- If time allows it, making user-requested additions

### **Test Procedures:**

The team started UAT testing by getting candidates to test TurtorialHeaven. Once there was a handful of testers the team scheduled time with them to have a call where we could observe them completing the objective list above. The team did not advise the testers other than relaying the objectives to them. Notes were taken from the observations and final feedback from the testers after completion. In total we had six testers complete the objectives. The team used an

excel sheet to keep notes of the time taken to complete the task, any bugs encountered, and extra notes. The excel sheet used is shown in the screenshot below.

Figure 3: Testing Results

| Test 1 | Feature         | Time Taken | Bugs                                          | Notes                                                      |
|--------|-----------------|------------|-----------------------------------------------|------------------------------------------------------------|
| Test 1 | Course Creation | 2 min.     | N/A                                           |                                                            |
|        | Join Course     | < 1 min.   | N/A                                           |                                                            |
|        | Course Content  | 10 min.    | N/A                                           |                                                            |
|        | Fourms          | 2 min.     | Like button not functioning                   |                                                            |
|        | Take Quizzes    | 5 min.     | Boards for incorrect were broke on true/false | Took Quizzes in test course seemed normal.                 |
|        | Stats Page      | < 1 min.   | N/A                                           |                                                            |
| Test 2 | Feature         | Time Taken | Bugs                                          | Notes                                                      |
| Test 2 | Course Creation | 1 min.     | N/A                                           |                                                            |
|        | Join Course     | 1 min.     | N/A                                           |                                                            |
|        | Course Content  | 15 min.    | N/A                                           | Took a bit to add an image but overall seemed smooth       |
|        | Fourms          | 2 min.     | N/A                                           |                                                            |
|        | Take Quizzes    | 7 min.     | Open answer question graded wrong             |                                                            |
|        | Stats Page      | 1 min.     | Stats page didn't update                      |                                                            |
| Test 3 | Feature         | Time Taken | Bugs                                          | Notes                                                      |
| Test 3 | Course Creation | 2 min.     | N/A                                           |                                                            |
|        | Join Course     | < 1 min.   | N/A                                           |                                                            |
|        | Course Content  | 12 min.    | N/A                                           |                                                            |
|        | Fourms          | 1 min.     | N/A                                           |                                                            |
|        | Take Quizzes    | 6 min.     | N/A                                           |                                                            |
|        | Stats Page      | 1 min.     | N/A                                           | Good test no noticeable bugs/issues occurred while testing |
| Test 4 | Feature         | Time Taken | Bugs                                          | Notes                                                      |
| Test 4 | Course Creation | 3 min.     | N/A                                           | Missed Images so had to redo                               |
|        | Join Course     | 2 min.     | N/A                                           |                                                            |
|        | Course Content  | 14 min.    | Adding image failed                           |                                                            |
|        | Fourms          | 1 min.     | N/A                                           |                                                            |
|        | Take Quizzes    | 7 min.     | N/A                                           | Took Quizzes in test course worked as expected.            |
|        | Stats Page      | < 1 min.   | N/A                                           |                                                            |
| Test 5 | Feature         | Time Taken | Bugs                                          | Notes                                                      |
| Test 5 | Course Creation | 2 min.     | N/A                                           |                                                            |
|        | Join Course     | 1 min.     | N/A                                           |                                                            |
|        | Course Content  | 12 min.    | N/A                                           |                                                            |
|        | Fourms          | 2 min.     | N/A                                           |                                                            |
|        | Take Quizzes    | 5 min.     | N/A                                           | Took Quizzes in test course worked as expected.            |
|        | Stats Page      | < 1 min.   | N/A                                           |                                                            |
| Test 6 | Feature         | Time Taken | Bugs                                          | Notes                                                      |
| Test 6 | Course Creation | 1 min.     | N/A                                           |                                                            |
|        | Join Course     | < 1 min.   | N/A                                           |                                                            |
|        | Course Content  | 13 min.    | N/A                                           |                                                            |
|        | Fourms          | 2 min.     | N/A                                           |                                                            |
|        | Take Quizzes    | 5 min.     | N/A                                           | Took Quizzes in test course worked as expected.            |
|        | Stats Page      | 1 min.     | N/A                                           |                                                            |

## Testing Review:

During our integration testing, we learned that testing along the way saved us from big headaches down the line. We made changes to old code throughout the project to update different components which sometimes resulted in errors. If the code was not known to be



functioning before the changes it could have been hours of work towards finding the issues in the code. Most of these changes were always to our core functionality of the assessments and courses which we have been updating along the way to complete our project scope.

## Budget:

This budget shows a breakdown of the cost for the development of TutorialHeaven. Some totals vary like AWS as it increases with traffic to the website.

| TutorialHeaven: Budget |                |      |                |         |
|------------------------|----------------|------|----------------|---------|
| NO.                    | ITEM           | UNIT | PRICE          | TOTAL   |
| Software               |                |      |                |         |
| 1                      | AWS Hosting    | 1    | \$600 (Yearly) | \$600   |
| 2                      | DNS Name       | 1    | \$15 (Yearly)  | \$15    |
| 3                      | VS Code        | 1    | \$0            |         |
| 4                      | GitKraken      | 1    | \$0            |         |
| 5                      | GitHub         | 1    | \$0            |         |
|                        | Subtotal       |      |                | \$615   |
| Labor                  |                |      |                |         |
| 6                      | Website Build  | 1    | \$5,000        | \$5,000 |
| 7                      | Database Build | 1    | \$500          | \$500   |
|                        | Subtotal       |      |                | \$5,500 |
| Total                  |                |      |                | \$6,115 |

Figure 3: Budget

## Project Timeline:

The timeline shows a breakdown of the objectives and the dates they were completed on.

| Task # | Task Name                       | Duration | Start Date | End Date  |
|--------|---------------------------------|----------|------------|-----------|
| 1      | Research Tools                  | 7 days   | 8/28/2020  | 9/4/2020  |
| 2      | Research Learning Methodologies | 14 days  | 9/4/2020   | 9/18/2020 |
| 3      | Requirement Collection          | 3 days   | 9/18/2020  | 9/21/2020 |
| 4      | Create DB schemas               | 5 day    | 9/21/2020  | 9/26/2020 |
| 5      | Mockup wireframes               | 7 days   | 9/21/2020  | 9/28/2020 |
| 6      | Configure AWS servers           | 5 days   | 9/28/2020  | 10/3/2020 |
| 7      | MVP Development                 | 60 days  | 10/3/2020  | 12/3/2020 |
| 8      | First Semester Presentation     | 1 days   | 12/3/2020  | 12/4/2020 |
| 9      | Winter Break                    | 29 days  | 12/8/2020  | 1/11/2021 |
| 10     | Continued MVP Development       | 78 day   | 1/11/2021  | 3/29/2021 |
| 11     | Create Demo Content             | 30 days  | 1/11/2021  | 2/1/2021  |
| 12     | Bug Hunting                     | 30 days  | 1/22/2021  | 2/22/2021 |
| 13     | Testing                         | 7 days   | 1/25/2021  | 1/31/2021 |
| 14     | Code Cleanup                    | 5 days   | 2/2/2021   | 2/6/2021  |
| 15     | Create Expo Presentation        | 7 days   | 3/15/2021  | 3/21/2021 |
| 16     | Class Presentation              | 1 day    | 3/29/2021  | 3/29/2021 |
| 17     | Final Report                    | 7 day    | 4/5/2021   | 4/12/2021 |
| 18     | IT Expo                         | 1 day    | 4/13/2021  | 4/13/2021 |

Table 4: Timeline

## Problems Encountered:

While developing our project we had to start looking into the cognitive science behind memory retention and learning. We looked to books and textbooks written by cognitive scientists and professors called *Make it Stick*, *Strategies That Work*, and *Deep Work* to supplement our

implementation <sup>5,6</sup>. Additionally, we interviewed a professor and people who taught themselves skills or tools in our attempts for initial feedback and brainstorming.

Another Problem we faced was in addressing the user experience through designing screens and tools that were intuitive enough for people with little experience to use, while also providing advanced users the ability to get the most out of their course design. We have been drawing inspiration from platforms like GitHub and Canvas, and from our testers to implement the best user experience for both advanced and novice users.

Throughout development we ran into problems that we needed to solve to progress. These problems ranged from how to implement a new library into our website to how do we host this website. Solving these problems required us to use online resources, personal connections, and each other to find a solution.

## **Recommendations for Improvement:**

If we were to do this project over again, we would be able to implement more features in the same time. Starting off the project some of us had to learn the technology we were using; this took up some time from development. Doing it again we would have this knowledge already and be able to start development off straight away.

Features we would like to add if we had more time are initial assessments so users can skip content they already know, more question types for assessments, more in-depth feedback, ability to add project assignments.

The future of this project is unknown, most group members are working on other things that require our attention and are spread thin. As it stands the project will stay in indefinite hiatus until we decide to return to it.

## Conclusion

This next section will go over the lessons we have learned throughout our time on this project and the skills we gained from working on this project.

### **Lessons Learned:**

We have made great leaps in our knowledge of web development and specifically technologies like Vue, MongoDB, AWS, and JavaScript. Having gained this knowledge over the lifespan of the project we now know more than we did in the beginning. This means that we can look back in retrospect and find numerous things that we could do differently than we did.

We have learned a lot about reactivity in the Vue framework and how data not initialized on create will have to be set a certain way in order for it to react to changes. This was a lesson learned that we will have knowledge of in the future. Another Lesson is when we named data differently in the frontend then in the backend that caused us some problems. The team learned that creating a simple naming scheme to start with would have saved us the pain of checking over code for any mismatched names to fix components.

### **Skills Built and Acquired:**

While working on this project the team all primarily being in the cybersecurity track have learned a great deal on the side of software development. Building knowledge on Javascript and Vue to produce our website. On the database side of things, we picked up more skills on AWS

and MongoDB. While programming we also researched a good amount of open source software such as QuillJS, Vue Draggable, and Express.

Past project management content was reinforced helping to cement past course knowledge. The work on the different sections of the contract and user diagrams refreshed a lot of the knowledge that was starting to fade from the past project management course.

## References:

- [1] Wadhvani, P., & Gankar, S. (2020, May). E-Learning Market Trends 2020-2026: Global Research Report. Retrieved December 07, 2020, from <https://www.gminsights.com/industry-analysis/elearning-market-size>
- [2] Protopsaltis, S., & Baum, S. (2019). Does online education live up to its promise? A look at the evidence and implications for federal policy. Center for Educational Policy Evaluation.
- [3] McDaniel, M. A., Agarwal, P. K., Huelser, B. J., McDermott, K. B., & Roediger III, H. L. (2011). Test-enhanced learning in a middle school science classroom: The effects of quiz frequency and placement. *Journal of Educational Psychology*, 103(2), 399.
- [4] Martínez-Argüelles, M., Castán, J., & Juan, A. (2009). How do students measure service quality in e-learning? A case study regarding an internet-based university. In *Proceedings of the European Conference on E-Learning* (pp. 366-373).
- [5] Brown, Peter C. *Make it Stick: The Science of Successful Learning*. Belknap Harvard, 2018. Print.
- [6] Newport, Cal. *Deep Work*. London: Piatkus, 2016. Print.

## **Appendixes:**

### **Appendix A: Code**

Our GitHub repo: <https://github.com/UndecidedTech/TutorialHeaven>