

Introsec

by

Michelle Fordos, Joshua Andre, Eric Turner

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

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

Michelle Fordos

04/07/2021

Michelle Fordos

Date

Joshua Andre

04/07/2021

Joshua Andre

Date

Eric Turner

04/07/2021

Eric Turner

Date

Tony Iacobelli

04/07/2021

Tony Iacobelli, Faculty Advisor

Date

University of Cincinnati

College of

Education, Criminal Justice, and Human Services

April 2021

Table of contents

Abstract.....	1
Introduction.....	2
Problem.....	2
Solution.....	3
Project Goals.....	3
Overview.....	4
Discussion.....	4
Project Concept.....	5
Design objectives.....	5
Technical Approach.....	8
Web Application.....	8
Servers and Databases.....	8
Virtualization.....	9
User Profile.....	9
Use Case Diagram.....	12
Testing.....	13
Testing Methodology.....	14
Scope of Testing.....	14
Objectives.....	15
Test Logs and Procedures.....	16
Test Results.....	17
Budget.....	18
Project Timeline.....	19
Problems and Analysis.....	22
Future Recommendations.....	23
Conclusion.....	23
Lessons Learned.....	24
Developed Skills.....	24
Plans for the Future.....	24
References.....	26

Table of Figures

Figure 1 10

Figure 2 11

Figure 3 12

Figure 4 13

Figure 5 16

Figure 6 18

Figure 7 19

Figure 8 20

Abstract

Introsec is an informational program that introduces middle and early high school students to the realm of cybersecurity. This web application consists of a series of interactive gamified labs that help students develop skills that are relevant to their personal cybersecurity and the broader industry. According to security firm ProtectWise, 67 percent of millennials have never taken a cybersecurity class (Delaney, 2018). Based on our research, there is a need for a quick and interactive course to introduce middle and high school level students to the basics of cybersecurity and gauge their interest in the field. Introsec provides a fast, fun and effective way to get hands-on experience without needing to commit to a long camp or series of courses. We have created a functioning web application that provides students with basic information about aspects of cybersecurity through multiple scenario-based games.

Introduction

This section will cover the problem Introsec aims to solve, as well as the core functionality of the solution. Outlined are the primary application features that allow Introsec to solve this problem. Finally, the overview briefly explains what content will be presented in the remainder of the report.

Problem

As the reach of the World Wide Web expands, more and more cybersecurity professionals are needed to keep it secure. The International Information System Security Certification Consortium (ISC)² estimates that by 2022 there will be 1.8 million unfilled cybersecurity jobs (ISC², 2020). However, many students who are interested in an IT or computer-related field may not even be aware of this opportunity. According to security firm ProtectWise, 67 percent of millennials have never taken a cybersecurity class (Delaney, 2018).

There are many organizations who provide free cybersecurity courses for teachers to bridge this gap. The advocacy group, Cyber.org, provides many such in-depth courses. However, these courses are long – often taking upwards of 20-30 hours and focus on very specific concepts (Cyber.org, 2020). Organizations such as GenCyber also host cybersecurity camps, but these are long and in-depth, in-person activities that require significant prior commitment (GenCyber, 2020). Based on our research, there was not a quick and interactive course to introduce middle and high school level students to the basics of cybersecurity and gauge their interest in the field. A short course with interactive demonstrations can help career coaches, advisors, and teachers convey basic cybersecurity information to students in a classroom setting.

Solution

Introsec is the solution for middle and high school students interested in cybersecurity. This web application provides a short commitment resource that introduces and provide hands-on cybersecurity information and demonstrations to complete this goal. Not only is Introsec readily available and accessible, it does not take up a significant amount of time to complete. Within this web application, users read information on cybersecurity, take part in demonstrations and activities, and have a short comprehensive quiz at the end to see how much of this information was retained. These demonstrations include a mock “man-in-the-middle” attack, a mock DoS attack, and encryption basics. Students get interactive experience with these concepts through scenario-based games using virtual machines to learn in a realistic and fun environment. For example, students learn about denial-of-service attacks by performing an attack on a virtual machine and witnessing the consequences. Lesson completion is tracked on a per-user basis and are followed up with a comprehensive quiz. This web application allows students and advisors to see if cybersecurity is the correct path for them.

Project Goals

- Feature #1: Attack Simulation
 - Simulation/Demos of multiple attacks so that the user can see what these attacks look like in the real world
- Feature #2: Comprehensive Quiz
 - Quiz to measure how much information the user has retained throughout the course
- Feature #3: Login Portal
 - Used to authenticate users as well as link the progress they have made to a reusable token

- Feature #4: Save States Tied to Logins
 - Ensures that user progress is saved throughout the duration of the lessons
- Feature #5: Certificate of Completion
 - Certificate with the user's name and digital signature stating that he/she has completed the course
- Feature #6: Completion Tracking
 - Show lesson status while completing a lesson and subsequently show status for each lesson on home and lesson main page.

Overview

The totality of this paper will consist of what it takes to make Introsec function as a formidable learning tool. Our project timeline and design features will be included. The entirety of our project deliverables will be included. This includes our estimated budget, use cases, user profiles, and the project timeline. This report will disclose the information needed to fully understand how Introsec has been brought to completion.

Discussion

This section will cover the project concept, design objectives based on user needs, and a description of technical details such as the web application itself, servers and databases, and virtualization. User profiles will be illustrated and explained with figures as well as a use case diagram of Introsec, the budget and an in-depth timeline of project deliverables and due dates. Lastly, any problems that were experiences during the creation of Introsec are discussed with an analysis of these hurdles.

Project Concept

The inspiration for this project came from the lack of readily available resources for middle to high school level students who want to see if cybersecurity is an interest they wish to pursue within their academic career. Throughout the United States there are no easily accessible cyber courses for this demographic of students. 67% of millennial students have not taken any sort of cybersecurity course at all (Delaney, 2018). This is in a society where technology and the use of the Internet is at an all-time high. Additionally, there is the rise of cyber attacks and crime because more individuals have access to technology that can be used for this. As more individuals have access to technology, there is a rise in cyber attacks and crime. In 2018, “there were 80,000 cyber attacks per day,” which equates to over 30 million attacks per year (2019 Cyber Security Statistics Trends & Data, 2020). This short intro course can help these students not only learn a couple of tips and tricks to help keep them safe, but also gives those interested in cybersecurity an opportunity to get their feet wet and explore what the cybersecurity field has to offer.

Design objectives

To solve the stated problem by providing students with an introduction to cybersecurity, a solution would best be incorporated into a teacher’s curriculum or be used as a tool for a school advisor. Therefore, to fit the needs of these users, Introsec is easily accessible, captivating for a younger audience, extremely intuitive to use, informational enough to be adopted by school faculty, and measure the student’s progress and information retention. These requirements are what drive the project goals.

To be easily accessible, Introsec is a web application so users only need a browser and an Internet connection. Therefore, tracking user progress and an intuitive approach to learning can

be accomplished within the site. Students can log in with their email and pick up where they left off last in their learning. Google sign-in services are used for logging in, creating a unique identifier used to save a student's progress as they navigate through lessons. This way, lessons can be completed at any pace to conform to student or faculty needs.

Many of the goals listed in the project goal section are specific lessons students will learn from Introsec. These lessons were each chosen based on their importance in the field of cybersecurity, ability to be gamified, and how interesting each would be to the targeted age group. Additionally, topics needed to introduce students to cybersecurity without going too much into detail and therefore taking up less time (unlike other solutions). Ultimately, these goals provide a fun way for students to learn while being informational enough to be used in curriculum.

Another goal – to gamify each lesson – is accomplished by using consecutive lessons that unlock information in order to complete the next lesson, following a creative storyline. For maximum user engagement, students use a terminal within a VM (displayed within our site) to perform activities related to cybersecurity concepts, like an interactive lab. This choice was driven by considerations for making the learning environment fun for students, captivating their attention, and providing maximum content retention. Guidance, informational text, and hints are displayed next to the user's view of their VM within the Introsec site to guide them through each lesson.

Every user has their own VM created when they sign up for Introsec. Linux VMs contain adequate security measures for public use and students use simply named aliases for commands to complete tasks within a lesson, making terminal use straightforward and understandable for

non-technical users. These aliases also send HTTP requests to an Introsec endpoint to update and save a user's progress through each lesson.

Lastly, to measure a student's success, comprehensive quizzes are to be taken after each lesson is completed. This goal was chosen so teachers and advisors can incorporate Introsec into graded curriculum and know if a student needs additional help with a concept. Once all lessons are completed and quizzes are passed, a user can generate a PDF certificate stating that he/she has finished the course. This goal of creating a certificate is important to prove student completion, especially if chosen to be graded.

As the reasoning and specific details for each goal has been described above, it should be noted that some goals for Introsec have changed during the project lifecycle. Before the availability and specifics of various technical considerations were known, the ability for students to demonstrate concepts they learn, like DoS attacks for example, was not clear. Initially, the goals for each lesson were instead centered around simulating concepts through a front-end game, possibly using a JavaScript library like Phaser to imitate actions that follow a lesson. In other words, faking a DoS attack instead of really doing one.

Subsequently, not knowing the technical backing for various solutions to gamifying lessons impacted the goals for the lessons chosen. The best user interaction for learning a gamified solution elicited different goals for what lessons best supported those technologies. After extensive research on various technical architectures, it became apparent that cybersecurity concepts could in fact be delivered through a virtual lab approach, a solution that best filled the project requirements listed earlier.

Technical Approach

In this section the technology used by the Introsec system will be covered. There are multiple technologies used by Introsec to become a fully functional application. There will be three key points touched upon: the web application, the servers and databases, and virtualization. These are all key essentials of Introsec.

Web Application

The business logic for the Introsec web application is written in Java 11 using the IntelliJ IDEA integrated development environment. It utilizes the Java Spring Boot framework to develop an easy-to-understand set of microservices that follow the MVC (Model View Controller) software design pattern. Our DTOs (Data Transfer Objects) and DAOs (Data Access Objects) were written with the help of the Project Lombok library that allows us to automate the creation of accessor (get) and mutator (set) methods. The website's front-end interface is built with HTML5, CSS, JavaScript, and Thymeleaf, a template engine that allows us to create and render HTML templates and connect them to the mapped methods in our controller. Maven is our dependency manager and Github is used for version control.

Servers and Databases

Our application is hosted on a cloud server through Microsoft Azure. The primary SQL database used by the Introsec application contains the following tables:

- Users – Contains user login information and profile data
- Progress – Contains progress data, and quiz results
- Quizzes – Contains quiz answers

There is also a secondary database hosted on a different server that is designed to be attacked by users during the SQL injection lesson. This database contains mostly random filler data, along with some VM credentials to be used later, and a few quiz answers to be found by those who dedicate extra time to the lesson. Both SQL databases used are also hosted through Azure using Tomcat 9.0.

Virtualization

User lessons are conducted on virtual machines hosted on the UC Sandbox platform. When new users log into the Introsec web application for the first time, a virtual machine running the Kali Linux operating system is created by the system administrator. Each machine is associated with its corresponding user via a unique identifier placed on the VM itself. Progression is tracked via an aliased command that executes a script on the virtual machine. This script then sends the unique identifier for that virtual machine with a PUT request to the web application to update the user's progress in the database.

User Profile

The primary users of Introsec are high school students, roughly aged 14-18. They will use the majority of the site's features to get the most from their learning experience. Some students may complete the lessons as part of a class or at the request of their academic advisor, so these users must be considered as well. Finally, the developers and administrators must be able to use the application effectively for testing.

Figure 1. User Profile 1

User profile description for developers and system administrators: These users need the tools necessary to maintain VMs and the web application. Familiarity with web development fundamentals and cybersecurity concepts are required to upkeep the app, add new features, and oversee VM use.

User Profile 1
Application: UC Sandbox, GitHub, SQL Server Management Studio, IntelliJ IDE, Kali Linux
Potential Users: Developers and System Admins.
Software and Interface Experience: The user should either have cybersecurity experience for the creation and maintenance of VMs used with our application or have web development experience for creating the web app. Additionally, users should be familiar with database management and be experienced in the construction of queries.
Experience with Similar Applications Users need to have experience with the fundamentals of web development such as: HTML, CSS, JavaScript, Java, and use of a database in the application. Users should also be familiar with VCS and managing servers. Additionally, system admin users need to have experience with cybersecurity concepts in order to securely maintain VMs and the integrity of the application.
Task Experience: Task experience with web development and cybersecurity aspects listed above for the creation and maintenance of the application and VMs.
Frequency of Use: Frequency of use is high during the creation of the application. After the site is finished, the users will upkeep the application in the form of maintaining the VMs, addressing any bugs within the site, and possibly updating content, layout, or adding new features with student and teacher feedback.
Key Interface Design Requirements that the Profile Suggests: To be designed so VMs are easily accessible, proper error notices display on the site during maintenance, the languages used to build the site are formatted and written effectively so updating or adding code is efficient. Includes adequate commenting/JavaDoc.

Figure 2. User Profile 2

User profile description for students: students need access to a browser in order to use Introsec and start a learning session. This requires experience with navigating through sites. Additionally, the use of UC Sandbox is required. Therefore, Introsec is designed in such a way that it is easy to navigate and understand.

User Profile 2
Application: Introsec web application.
Potential Users: Students.
Software and Interface Experience: web browsers, Windows/ any operating system
Experience with Similar Applications Chrome, Firefox, Windows
Task Experience: Navigate our website to complete courses, as well as UC Sandbox test accounts to access VMs.
Frequency of Use: For learning one time, to better scores as many times as you want.
Key Interface Design Requirements that the Profile Suggests: Easy to navigate and read buttons. Clear and concise instructions on command usage in each game.

Figure 3. User Profile 3

User profile description for teachers or advisors: these users require similar experience as students, in that experience with web browsers is needed. Additionally, they must be able to navigate the site and be familiar with UC Sandbox for instruction purposes.

User Profile 3
Application: Introsec web application.
Potential Users: Teachers, advisors, and school administrators.
Software and Interface Experience: web browsers, Windows/ any operating system
Experience with Similar Applications

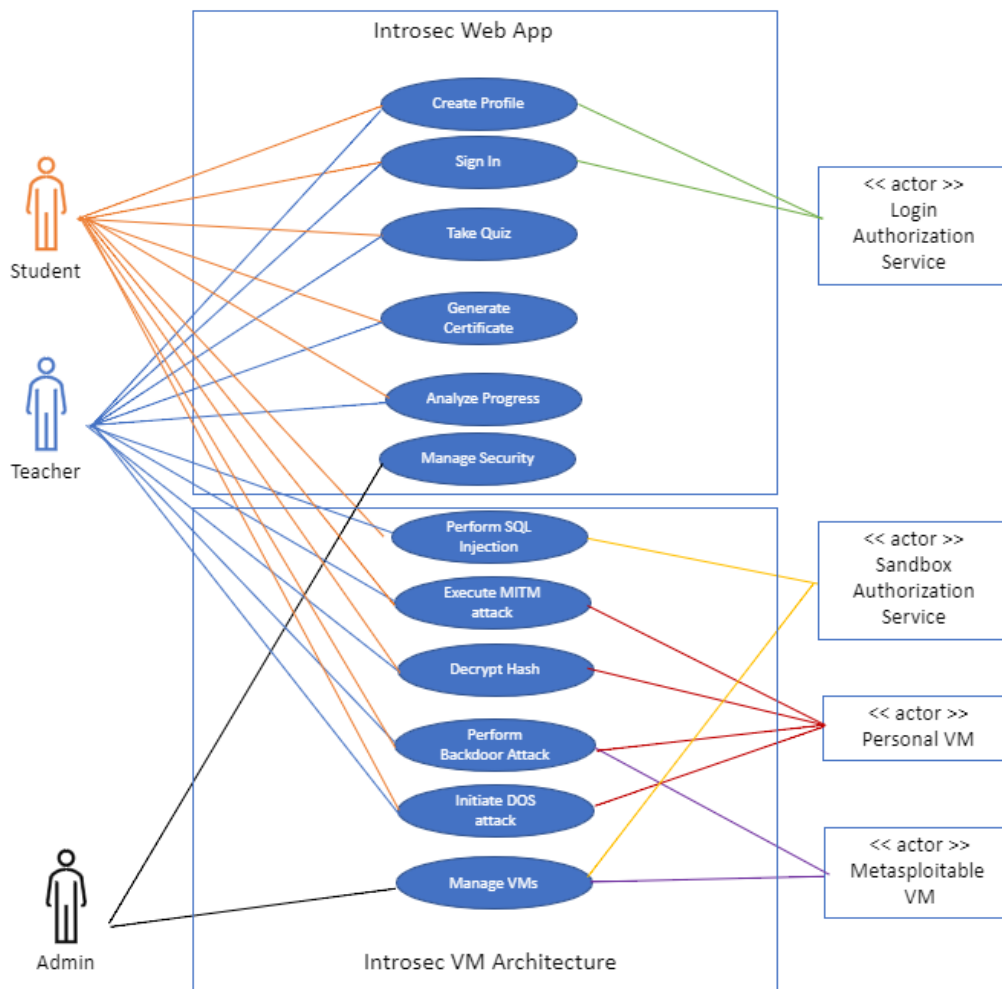
Chrome, Firefox, Windows
Task Experience: Navigate our website to complete courses, as well as UC Sandbox test accounts to access VMs.
Frequency of Use: One time use teachers, advisors, etc. would only go through one time to make sure the content is acceptable.
Key Interface Design Requirements that the Profile Suggests: Easy to navigate and read buttons. Clear and concise instructions on command usage in each game.

Use Case Diagram

Figure 4. Use Case Diagram

In this document we show what components are interacting as well as how they are doing so.

The first main user group we have consists of administrators and developers who create the application and maintain the quality and availability of Introsec. Our student user group comprises the majority of interactions within our application. They navigate around the site, completing lessons, creating accounts, and downloading the certificate of completion once they have successfully completed the course. Our final interacting party are teachers. These users vet content for students to make sure that it is appropriate.



Testing

In this section our testing method will be outlined and explained, starting with the testing methodology. This will explain what method we use to best test our product. After that, the scope will be explained. This will show everything that is going to be covered in testing. There will be a list of things that are covered in the scope. Once the scope is established, the objectives will be explained. The objective section is going to explain what exactly we want to accomplish in our testing. This will lay out each individual thing that is going to be tested and what outcome

needs to be achieved. Next will be the test logs and procedures. This will include how the testing will be done as well as the pass-fail conditions for each test case.

Testing Methodology

Our testing methodology is to have users from each user profile navigate through site with the goal of accomplishing certain objectives without guidance. Each user will test different criteria on our site, such as the user experience (intuitive navigation, ease of use, accessibility), and ensure that the lesson content and use of tools is comfortable.

Our approach to testing also aims to include at least two users for the teacher/instructor user profile, two student profiles, and one admin/developer user profile. We chose this approach so we could include have multiple users' tests for each user profile to increase the likelihood of finding any bugs or issues and make sure results are reproducible among user profiles.

Scope of Testing

- Login Authorization Service - Simple testing of the Google sign-in so that users can log onto the site and view their progress.
 - Create Profile (student, teacher)
 - Sign In (student, teacher)
- Take Quiz (student, teacher) - Making sure that quizzes display correctly and are properly scored in the database.
- Generate Certificate (student, teacher) - Testing the PDF generator to ensure certificates are readable and have the correct information.
- Analyze Progress (student, teacher) - This includes taking quizzes and progressing through lessons. The database values and front-end progress bars are watched to make sure that the correct values are being passed between them.

- **Manage Security (admin)** - This test case is a quick run-through of security settings so only VMs that are meant to be exploited can be accessed by users.
- **Sandbox Authorization Service** - This test case is to ensure that the admin can create new VMs for users and fix problems in Sandbox. The user also needs to be redirected from the Introsec site to the Sandbox login page.
 - Log Into VM (student, teacher)
 - Manage VMs (admin)
- **Personal VM** – These features are the core of Introsec’s lessons. Ensuring that each of these test cases work is essential for the functionality of lessons 1-6.
 - Perform SQL Injection (student, teacher)
 - Execute MITM Attack (student, teacher)
 - Decrypt Hash (student, teacher)
 - Initiate DOS Attack (student, teacher)
- **Metasploitable VM** – Making sure that the user can connect to the Metasploitable server via a backdoor is necessary for the lesson.
 - Perform Backdoor Attack (student, teacher)

Objectives

- **Test case 1:** Logging in (user account for site)
- **Test case 2:** UI navigation
- **Test case 3:** Lesson content navigation
- **Test case 4:** Sandbox sign in
- **Test case 5:** Access to VMs (can user clearly get to VMs and open remote console)
- **Test case 6:** Navigating Kali Linux with supplemental instruction
- **Test case 7:** Executing a lab alias post instruction
- **Test case 8:** Progress bar and lesson content updating
- **Test case 9:** Upon completion take quiz icon appears and is selectable

- **Test case 10:** Directs to correct quiz and UI functions properly
- **Test case 11:** Quiz form functions properly
- **Test case 12:** Games page updating based upon what quizzes are available
- **Test case 13:** Quiz results returning properly and retaining highest value after multiple attempts
- **Test case 14:** Games page navigation and updating based upon completion
- **Test case 15:** Upon completion generate certificate
- **Test case 16:** Log out functions correctly

Test Logs and Procedures

Figure 5. Test Case Conditions

This table shows the pass and fail conditions for each of the above test cases.

Test case	Pass condition	Fail condition
Test case 1	Log in completes all info stored to database	Log in completes no information stored, log in fails
Test case 2	User can navigate to all pages and buttons lead them to correct page	User selects one button and it takes them to an incorrect page
Test case 3	Lesson can be selected, and content can be accessed	Lesson page does not load, Lesson page is accessible and content does not load
Test case 4	Sandbox button navigates to sandbox and user can log in	Sandbox button does not load Sandbox, Sandbox loads user cannot log in
Test case 5	Access VM user can open remote console	User cannot access remote console
Test case 6	User can navigate Kali with written instruction	Instruction are not clear enough to navigate kali

Test case 7	Alias command completes the lab task as well as sends completion key to database	Alias command cannot run, Alias command runs but does not update database
Test case 8	Progress bars update after checkpoint is reached	Progress bars do not update
Test case 9	Take Quiz icon appears once progress bar is full and redirects to proper quiz	Take Quiz button does not appear or appears too early, does not redirect to correct quiz
Test case 10	After completion of lesson, intuitive direction to correct quiz and UI functions properly	After completion of lesson, user fails to direct to correct quiz and/or UI functions improperly
Test case 11	Quiz form functions properly, multiple choice captures input, validates, and returns grade.	Quiz form functionality fails
Test case 12	Games page updated based upon what quizzes are available to take, which are completed (with score), and which ones are inaccessible.	Quiz page inaccurately represents quiz status, quiz grades, or is not intuitively understood by user
Test case 13	Quiz results returning properly and retaining highest value after multiple attempts	Quiz results return incorrect or incomplete results for quiz and does not properly handle multiple attempts at a quiz
Test case 14	Games link to correct game and update to unlock next game when one is complete	Games link does not go to right game, games do not unlock upon completion
Test case 15	Certificate generates including users name	Certificate generates but name is blank, certificate does not generate, certificate generates but half of the name is missing, certificate does not generate
Test case 16	Log out button works user can log out	Log out button does not appear, log out button does not work

Test Results

Figure 6. User Test Results

This table contains the results from user testing, as well as the feedback provided by the test user and the Introsec developers on ways the application can be improved.

Test case	Result	User Feedback
Test case 1	PASSED	N/A
Test case 2	PASSED	Slight issues with navbar and button text when window sized is changed. All other UI elements still display perfectly, however.
Test case 3	PASSED	N/A
Test case 4	PASSED	N/A
Test case 5	PASSED	More detailed written instructions to switch between machines and open remote console would be helpful.
Test case 6	PASSED	Instructions should contain more info on Windows machine. Navigation of Kali is understandable.
Test case 7	PASSED	Aliases work properly. Wireshark and progress update issues when scaling to new users. Works on testbed.
Test case 8	PASSED	N/A
Test case 9	PASSED	N/A
Test case 10	PASSED	N/A
Test case 11	PASSED	N/A
Test case 12	PASSED	N/A
Test case 13	PASSED	N/A
Test case 14	PASSED	Progress update issues when scaling to new users. Works on testbed.
Test case 15	PASSED	N/A
Test case 16	PASSED	N/A

Budget

Figure 7. Budget

This is the estimated budget for Introsec. This shows all software and hardware that is being used as well as the cost of these items. The main cost stemmed from hours invested, because we found a decent amount of freeware and student packs to help us achieve our goals. The final budget did not deviate from our original estimation. The price of any software or tools we used throughout the project stayed the same and our original hours of labor did not change from the final amount.

Introsec Budget				
NO.	Item	Unit/Hours	Unit Price	Total
Software				
1	IntelliJ Idea	1	\$500.00	\$500.00
2	UC Sandbox	1	\$0.00	\$0.00
3	Microsoft Azure	1	\$10.00 (Monthly)	\$10.00
	Subtotal			\$510.00
Labor				
4	Design	13 hours	\$780.00	\$780.00
5	Set Up	75 hours	\$1500.00	\$1500.00
6	Development	411 hours	\$8220.00	\$8220.00
	Subtotal			\$10,500.00
	Total	499 hours		\$11,010.00

Project Timeline

Figure 8. Timeline

This figure shows the design timeline for Introsec. It spans from our very first meeting to the 2021 IT expo. Each section has a breakdown of when each task was completed with dates, as well as the overall number of days that specific section took to complete.

Task #	Task Name	Duration (days)	Start Date	End Date
1	Project Management and Deliverables	232	8/24/20	4/13/20
1.1	Team Building	7	8/24/20	8/31/20
1.2	Ideas and Brainstorming	7	8/24/20	8/31/20
1.3	Assignment 0: Team Members and Project Name	1	8/24/20	8/24/20
1.4	Assignment 1: Team Contract	7	8/24/20	8/31/20
1.5	Assignment 2: Project Abstract for Tech Expo	39	9/4/20	10/12/20
1.6	Assignment 3: Team Contract Resubmission	7	10/5/20	10/12/20
1.7	3-Minute Elevator Speech	14	10/5/20	10/19/20
1.8	Assignment 4: User Profile	7	10/12/20	10/19/20
1.9	Assignment 5: Use Case Diagram	36	9/14/20	10/19/20
1.10	Assignment 6: Draft Report	35	10/5/20	11/9/20
1.11	Assignment 7: Final Fall semester Report	56	10/5/20	11/30/20
1.12	Oral Presentations	28	11/2/20	11/30/20
1.13	Completed Alpha version of project	90	9/1/20	11/30/20
1.14	IT Expo Presentation	30	3/13/20	4/13/20
2	Research	7	9/1/20	9/8/20
2.1	Establish all Requirements	7	9/1/20	9/8/20
2.2	Research implementation of Cyber features	7	9/1/20	9/8/20

2.3	Research database setup	2	9/2/20	9/3/20
3	Prototyping	30	9/9/20	10/26/20
3.1	Prototype Drafting	5	9/1/20	9/5/20
3.2	Prototype Design	5	9/1/20	9/5/20
3.3	Create Prototype	7	10/1/20	10/8/20
3.4	Prototype UI testing	4	10/12/20	10/16/20
3.5	User feedback analysis	3	10/16/20	10/19/20
3.6	User feedback updates	2	10/19/20	10/21/20
3.7	prototype UI testing 2	3	10/21/20	10/24/20
3.8	final updates	2	10/24/20	10/26/20
4	Design	13	9/9/20	9/16/20
4.1	Design Database	1	9/15/20	9/16/20
4.2	Design Server	2	9/12/20	9/14/20
4.3	Design layout/UX	7	9/9/20	9/16/20
4.4	Determine which server/database/various technologies etc.	3	9/9/20	9/12/20
5	Set-Up	31	8/30/20	11/17/20
5.1	Setup GitHub	1	8/30/20	8/30/20
5.2	Setup databases	14	10/20/20	11/3/20
5.3	Setup Server	7	10/20/20	11/3/20
5.4	Setup supporting tools for Cyber features	23	10/25/20	11/17/20
6	Development	137	10/12/20	1/16/21
6.1	Create static layout	33	10/12/20	11/14/20
6.2	Create each educational feature	76	11/2/20	1/16/21
6.3	Implement Login	7	11/1/20	11/8/20
6.4	Create certificate generator	7	1/2/21	1/9/21
6.5	User progress tracking	14	12/1/20	12/14/20
6.6	Gamification	76	11/2/20	1/16/21

7	Testing	37	1/1/21	3/24/21
7.1	Database Security testing	5	1/17/21	1/22/21
7.2	Web Security Testing	5	1/22/21	1/27/21
7.3	Login Security Testing	5	1/27/21	2/2/21
7.4	Hash Security Testing	5	2/2/21	2/7/21
7.5	Demo Testing	1	2/7/21	2/8/21
7.6	UI Functionality	2	1/8/21	2/10/21
7.7	Unit Testing	2	2/10/21	2/12/21
7.8	Periodic Browser accommodation testing	2	2/12/21	2/14/21
7.9	Game Testing	10	2/14/21	2/24/21
8	Presentation	30	3/13/21	4/12/21
8.1	Prepare presentation	20	3/13/21	4/2/21
8.2	Practice Presentation	10	4/2/21	4/12/21
9	IT EXPO	1	04/13/21	4/13/21
9.1	Present	1	04/13/21	4/13/21

Problems and Analysis

A theme to the problems encountered was not being familiar with various technologies needed to create Introsec. Important stages of the project were often delayed because of this unfamiliarity. When deciding how to gamify the cybersecurity lessons, it was unknown if an interactive lab approach could be possible for users to explore cybersecurity concepts. Would VMs be required? How many databases are needed? What cloud services are available for student developers that are free? What networking requirements need to be met to support our lessons? What development stack best fits the project requirements? There are many answers to questions like these, which had to be found through extensive research and trial and error.

About 50% of the project timeline was used researching various technologies to decide possible directions of the project. This was not anticipated. Implementing the chosen technologies and architecture also took up more time than expected, because it was the first time using these for all group members.

Completing tasks like setting up a server or configuring a cloud hosted database were challenging endeavors that were most successfully accomplished as a team instead of individually.

Future Recommendations

If we had to do the project over again, there would be some slight changes to the layout of the instructional page for each lesson. Ideally, we wanted Introsec to display the window of a user's virtual machine within the instructional pages of our website. However, using CECH Sandbox did not allow for this functionality. We did not discover that this desired layout would not be an option until we were a considerable way into our project. If we had more time, we would use a different way to host virtual machines so that the user would not have to have two tabs open for both the website and their virtual machine, and could instead see everything in one place. This future change has been echoed as feedback from others as well.

Conclusion

This section covers our lessons learned, developed skills, and future plans to be discussed. There has been a learning curve for us because most of our group has not had real development experience in this format. Our skills were increased because of the knowledge we

had to learn. These skills are primarily increasing familiarity with enterprise app development, and cloud networking-based systems – more specifically, Azure.

Lessons Learned

There were multiple lessons learned throughout this process. This is the first time that our group has developed something to this scale before, so it was an interesting process. The main lesson we learned was perseverance passed our obstacles. The main obstacle being our lack of experience in the cloud computing environment. We proceeded to use Azure as our web hosting platform which is now working well for us but was a bit of a struggle to get set up. However now that we have everything set up, we are able to maintain and monitor our resources with ease.

Developed Skills

Overall, we've learned quite a lot about the production of enterprise-level web applications and the integration of servers and databases that come along with it. For those of us with app development experience, it was interesting to work with virtual machines and manage their integration into the broader project, as our previous experience with these was minimal. The project also taught us a lot about cybersecurity concepts and the software used by cybersecurity professionals. For those of us whose experience revolves around cybersecurity, it was an opportunity to practice our skills and learn about application design and database interaction.

Plans for the Future

The current plans for the future of Introsec are to hand it over to the Ohio Cyber Range so that they can use it as a teaching tool. In all phases of Introsec's development, the Cyber Range has been interested in the concept and what the project has to offer. We would be not have any

problem handing it over to them and allowing it to do with it what it was designed to do:
introduce young adults to cybersecurity.

References

Delaney, M. (2018, July 02). High Schools Prep Students to Fill Cybersecurity Skills Shortage.

Retrieved April 12, 2021, from <https://edtechmagazine.com/k12/article/2018/07/high-schools-prep-students-fill-cybersecurity-skills-shortage>

Find Curricula. (n.d.). Retrieved April 12, 2021, from [https://cyber.org/find-curricula?f\[0\]=grade_levels:58](https://cyber.org/find-curricula?f[0]=grade_levels:58)

GenCyber. (n.d.). Retrieved April 12, 2021, from <https://www.gen-cyber.com/>

The International Information System Security Certification Consortium (ISC)². (2020). (ISC)² Cybersecurity Workforce Study, 2020. Retrieved April 12, 2021, from <https://www.isc2.org/Research/Workforce-Study>

2019 Cyber Security Statistics Trends & Data. (2020, August 26). Retrieved April 12, 2020, from <https://purplesec.us/resources/cyber-security-statistics/>