



NEW WEBSITE & DATABASE INTEGRATION PROPOSAL

FOR: RAINBOW RECORDS



APRIL 9, 2022

STP SOLUTIONS INC
555 W HELM STREET

Derek Jamensky COMM 2216
Hareem Fatima
Sumeet Minhas

Table of Contents

Introduction	2
Project Details	4
Initial Deployment of Solution	4
Product Category Names	4
Search Bar Functions	5
Responsive Design	5
AWS Integration	6
Software and Development Frameworks	6
Upgraded Website Frameworks	6
Website Connection with AWS Database	10
Proposed Schedule	16
Phase 1: Analysis	17
Phase 2: Design	18
Phase 3: Develop	19
Phase 4: Testing	19
Phase 5: IT Support/ Maintenance	20
Conclusion	21
Recommendations	22
Glossary	23
References	24

Introduction

This proposal presents our plan to rebuild and launch a responsive eCommerce website for Rainbow Records. In addition, we will connect the website orders to the Amazon Web Services (AWS) database to automate shipping labels. This will streamline packaging and shipping to process orders faster and easier for customers.

The current website was designed in 2007 to handle occasional online sales of their small retail store. Due to the pandemic and increasing shift to e-commerce, online sales are now the primary source of revenue and make up over 50% of their business and are growing.

New Website Design

We will be building a new website to address the current issues Rainbow Records is dealing with as well as equipping the new website with scalability and functionality to simplify future upgrades and updates.

Responsive Design

The current website is unable to adjust to different browsers and screen sizes on mobile phones and iPads. We will be building a mobile-first website which means you create a website for a smaller screen. With a smaller screen, the design will be simple and efficient and adding scalability and browser-support such as Chrome, Firefox, and Safari will be simple. It will also be easier to add future features because the code will be uncluttered. With mobile-first development, the website will be able to handle traffic faster to ensure fast order processing and product updates. By implementing a responsive design, Rainbow Records will be able to reach a wider range of customers and increase their time spent on the website.

Search Functions and Categories/Filters

The current layout is cluttered with slow search functions and excessive categories. We will be implementing a search bar feature at the top of the website to streamline all search requests. It will use keyword matching to access the inventory database and return all the results that apply or are in similar sub-categories. There will be four categories: vinyl records, CDs, audio equipment, and memorabilia. Each category will have filters for price and date added. This will allow customers to filter their search by price and the latest products. By simplifying the search and adding filter functionality customers will be able to manoeuvre merchandise easily providing a positive experience and customer retention.

Amazon Web Services (AWS) Integration

The current database is hosted on Amazon Web Services (AWS). AWS is a standard hosting platform that is used by many retailers and is inexpensive. Current online sales are not being processed directly by the AWS database [1]. Orders are received as emails, printed, and typed into shipping labels causing clerical errors and delays. This process is time consuming and cannot handle the company's growth. This means the database needs to be connected to the website to automate shipping labels and packaging information.

To facilitate Rainbow Records' continued growth, we will take the existing database and use a querying language like SQL (Structured Query Language) to send requests back to the database. This will create a connection between the database and the new website to streamline operations.

Redesigning the website's responsiveness will provide a better user experience, increase time spent on your website, and enhance customer interaction. Connecting the database to the online sales will increase efficiency and reduce clerical errors by automating printing

shipping labels. It will also decrease packaging times by ensuring all parts of the database are updated.

Connecting the AWS database to the updated website will increase speed and flexibility of placing and shipping out orders. This will also reduce the margin of errors that were caused by accessing the orders from email and manually typing out the shipping labels. Most importantly, it will reduce the time it takes for Rainbow Records to ship out orders.

Project Details

In this section, we will further explain the details of how we are going to redesign the new website layout, search functions, and database connections for Rainbow Records' e-commerce website. We have conducted research and developed a plan with the advice of our developers and programmers on how to meet these goals.

Initial Deployment of Solution

Product Category Names

The current website is lacking a clear layout and there are an excessive number of categories that clutter the pages and add no additional features. To streamline the efficiency of the website we will create a drop-down menu with an option to view all products and four sub-categories: vinyl records, CDs, audio equipment, and memorabilia in our Ruby on Rails code. Within each category, we will include filter features that will allow customers to filter products by price and date added. Product details will be automatically updated when inventory is updated. By reducing the number of categories but adding additional filter

features, customers will be able to easily find their desired product and modify their search. This will positively impact the customer experience while visiting Rainbow Records' website.

Search Bar Functions

The search functions of the current website are slow and unreliable. To increase the efficiency of this function, we will include one search bar at the top of the website which will allow customers to explore the inventory quickly and easily. The search engine will match keywords with products in the database and display the details. The search bar function will collect data to help track user activity. This will be a helpful resource in making future business decisions by keeping track of what products and categories are most sought after. By simplifying the search feature, we will make the customer experience simple and efficient.

Responsive Design

The current website is unable to adjust to browsers and shift screen sizes between mobile phones and iPads. As devices continue to evolve and become more integrated in our daily lives it is important to build a website that can adjust to these changes.

We will develop the website through a mobile-first design process using a Bootstrap framework which will consider future changes that may need to be made. A mobile-first design means that the website will appeal to more users because it will be built to adjust to changing screen dimensions. It will also contain fewer bugs because the code is intended for a smaller screen which makes it simple and concise. By proxy, this will create faster load times and increase search engine optimization.

We will be designing three different wireframe websites for Rainbow Records: one black and white layout, and two others with distinct colour design concepts. These will then be

subjected to two rounds of revision by Rainbow Records. After completing these revisions, we will begin to launch the new website.

AWS Integration

The current database is hosted by AWS, but it is not connected to the website. The current website is not correctly designed to process the online orders directly by the AWS database. So, the orders are received as emails, printed and typed into shipping labels which causes clerical errors and delays. This process is time consuming and cannot handle the company's growth. This means the database needs to be connected to the website to automate shipping labels and packaging information. By using SQL Server Management Studio (SSMS) and Relational Database Services (RDS), we will make the connection between the new website and its current database. This connection will improve order efficiency and the order processing time. This will save time that was taken to print out orders and the processing of manual shipping labels.

Software and Development Frameworks

Upgraded Website Frameworks

We will be building a new website for Rainbow Records with updated frontend and backend functionality. The frameworks we have chosen are specific to eCommerce websites to ensure security and scalability are at the forefront of all designs. We have listed the technologies, their benefits, and testing strategy below.

Website Wireframes with Figma

To plan the layout of the new website we will be using Figma. Figma is an online platform that allows users to create custom, interactive wireframes. Wireframes are diagrams with

simple lines and shapes that represent the skeleton of a website. They show what will be displayed on the interface and where the elements will be placed. They are also interactive, so you will be able to navigate the website and decide on the format. Wireframes encourage feedback and collaboration and ensure that our team knows Rainbow Records' preferences before moving forward with the actual buildout of the website.

Layout and Design with Bootstrap

To improve the functionality and speed of the website we will be re-designing the layout using Bootstrap, a framework used to create modern, fast websites. Bootstrap is built on HTML (hypertext markup language), CSS (cascading style sheets), and JavaScript. HTML is a standard coding language used to build web pages. CSS is a style sheet language used to describe how a web page will look on a screen. JavaScript is a programming language that controls web page behaviours and functions. These three languages are what Bootstrap uses which allows us to create custom designs that will ensure your website is unique to Rainbow Records. Bootstrap facilitates a mobile-first development process which means the code is intended for a mobile phone [2]. This means the code structure is simple and easier to build upon because it is meant for a smaller screen [2]. Websites developed browser-first are more prone to bugs because scaling a website down is more complicated than building out [2]. Ensuring the functionality is straight-forward and simply means the load times will be faster and this will improve the search function optimization. Customers will be able to access products faster and easily, increasing satisfaction.

Testing Responsive Design of Bootstrap

To ensure the responsive design of our website we will conduct various tests. We will use BrowserStack, an online testing tool to carry out our tests. We have followed a recommended checklist of all the types of responsive design tests necessary to produce a full functioning website [3]. We will use this tool to:

- Test on multiple devices
- Test for navigation
- Test fonts on multiple devices
- Test on small screen devices
- Test the speed of the website
- Test the element alignment
- Test for content placement
- Test the photo resolution across screens
- Test cross browser responsibility

Backend eCommerce with Ruby On Rails

To re-build the functionality of the website we will be shifting the backend framework to Ruby on Rails (RoR). Ruby on Rails builds fast web-applications and is a primary framework used in e-commerce [4]. It is flexible, allowing you to update or modify changes in the future if there need to be more functions or categories added. You will not need to dismantle anything, RoR's flexibility allows you to add new plugins to expand functionality. RoR is built for a fast MVP (minimum viable product) because it has many built-in configurations for ecommerce [4]. This means features common in most ecommerce websites are easily buildable because they come in RoR's framework [4]. RoR is easy and straightforward to use because it comes with standardised policies decreasing the likelihood of facing challenges during development. These standard built-in policies ensure code is clean and efficient. Security is a top priority, especially on an ecommerce website. RoR comes with many security-focused plugins and extensions to handle customer data, such as personal information and payments [5]. We will have all the security tools available to handle vulnerabilities to protect your website. RoR comes with multiple automation features within its framework that ease the process of deploying the platform. This means it resolves bugs within the language to simplify the debugging process to make sure the final product is

seamless [5]. Ruby on Rails is a powerful, fast, and easy framework to ensure your website is functional and secure.

Test Functionality of Ruby on Rails

To test the functionality of the backend framework of our website we will be creating multiple test scripts that will focus on specific features. By conducting these tests, we will check that the new website is secure and functional ensuring customer satisfaction, retention, and product promotion. We will create scripts to test:

Online Purchases

- Security:
 - SQL injection attacks for credit card credentials
 - Two-way authentication through dynamic passwords on mobile/desktop for banking applications
 - After 3 or 5 unsuccessful CVV attempts, lock user access
- Session:
 - If a user clicks the back button or refresh button, ensure they remain within same session
 - Maximum session time-out after period of inactivity
- Browser
 - Browser cookies to save data for users
 - 'Remember Form Data' settings of the browser should not save credit card information

Search Bar Scenarios

- Results should display relevant items to search keyword

- % Sign in search keyword should not redirect to 404 ERROR
- Application should not crash if user inserted % in search field
- Suggest matches while user is typing in text box
- When user clicks on link from result, result should be maintained
- Keyword/filters should clear on clicking reset button
- History in search field should be relevant to current user
- 'Enter' button should work on all devices

Category Fields

- Items should update when inventory is added
- Items should update after every sale
- Item suggestions should display on page
- Total number of items should display for each category

Website Connection with AWS Database

We will connect the database to the new website and improve its functionality and compatibility with the website framework. The website is to be hosted by AWS. AWS is a cloud computing platform hosted by Amazon. We will use AWS services for the database connection [6]. We will further perform different types of testing on different parts of the program. This will test for load balancing, compatibility, security, and secure connection of the database. These tests will be performed by our software developers where they will isolate some parts of the code and test them concurrently. These tests are automated and verify their functionality by stating if the tests failed or passed. The tests that will be performed are load testing, database integration testing, security testing and vulnerability testing.

Amazon Relational Database Service (RDS)

RDS is a SQL database service which is provided by Amazon Web Services. This service supports different kinds of database engines to organise and store data [7]. RDS also supports database management tasks, like backups, data migration, recovery, and patching. The RDS service that we will be using for this project is MySQL. It is a database management system which is commonly used to add, access, and process data that is stored in any computer database. This will allow us to interact directly with the database to access customer information and their orders.

Authorised Logins

To access the database from the warehouse, we will create all the login information necessary, like a username which is the same as the database name and a password using Gear Host. Gear host is a platform as a service (PaaS) which is also one of the RDS services [7]. It is a cloud platform which helps in scalability of businesses. Logins will make it easier for you to access the database and be able to make any changes to the database from the warehouse. The database can also be accessed remotely with the authorised login information and on any other computer.

Database Security Testing

Securing the database server will prevent any malicious attack on the server and personal user data. This process is also known as hardening the database, which consists of network, and physical security with secure system configuration [8]. This process includes:

- Locking down untrusted accounts
- Regulating patch servers
- Disable any unauthorised network access
- Encrypting all user data, database information

- Creating backups of the database system

The next step is to perform SQL injection testing [9]. This will check for any possibilities if the database is prone to any unauthorised and invalid user-controlled SQL queries. Our software developers will write SQL scripts that will look for any SQL injection vulnerabilities in the database by searching for unauthorised users who are trying to access and manipulate the database.

This will prevent any sensitive data from being exposed to the unauthorised user. Furthermore, it will secure the database from any modification to the data. For example, preventing SQL injection attacks from inserting, deleting, and updating the existing database entries [9]. The script will be run continuously on the database, so it can look for the attacks and block them as they arise.

MySQL Database Connection with SSMS

The current SQL database provides full functionality; however, it is not connected to the website. We will make use of SQL (Structured Query Language) to communicate with the SQL database. In addition to that, we will use SQL Server Management Studio (SSMS) [10]. SSMS is a database management tool for SQL infrastructures.

Some of the advantages of using SSMS are [10]:

- Cost free
- Installation process is streamlined
- Provides better security
- Enhanced performance
- Easier to manage data and log files

The first step is to install SSMS on the warehouse computer. We must collect the database information from Gear Host to connect the database to the website [11]. We will then enter all the required information. For instance, the server type, server name, server authentication, database name and password for the very first time [11]. By saving all the default settings, it will be easier for you to log in the second time with just the database name and the database password. One main benefit of not saving the login information to the warehouse system is that it restricts any unauthorised access to the database and reduces the risk of losing website and customer data. It will also prevent errors regarding customer orders and customer information to make sure that the connection works well with the website.

Website connection with the SQL Database

To connect the website with the SQL database, we will create a server-side framework or a script to perform the HTTP request or the Database interaction. Server side is also known as the back end, and this is where all the code and programming are found [12]. A server-side script is a script that is run whenever there is a client or user request. It is run before the web page is loaded. This is important for storing user login details and user information. This script will run on the server and will hold the connection details to connect the database to the upgraded website. It will also perform queries that we will need for formatting the results. We will use PHP to write the scripts that will capture all the user information and order details. PHP is an open-source scripting language which is commonly used for back-end development [13]. It is easier to write, provides easy integration and compatibility, enhances performance, and is cost efficient [13]. We will create another script that will extract the user shipping information associated with the order number for automated shipping labels. The shipping labels will be automatically printed whenever there is an order placed on the website.

Database Integration Testing

We will also perform Database integration testing on the database connection. Integration testing tests the connection between the database and the website, whereas unit testing tests are only performed on the database [14]. This ensures that two systems are compatible with one another. Due to the complexity of the integration test, we will need additional time to perform the tests. We will need to configure a real-world environment using Docker. Docker is a PaaS (Platform as a service) and is a set of products that delivers software in packages called containers using virtualization [15]. It will be used to encapsulate parts of the database with some relevant test data for those specific parts [15].

The very first step will be the installation of the Docker and the Docker Compose on the warehouse computer. The following steps show the flow of operations throughout the testing phase [15]:

- Creating a Docker container
- Starting the container
- Creating a database
- Migrating a Schema (logical constraints)
- Running different tests for compatibility
- Destroying the containers after the tests are successful

All the above operations will test the following functions:

- gathering information on customer placing an order
- Gathering information on products being ordered
- Checks if the product is available otherwise displays “Out of Stock”
- Checks if there is an existing customer account otherwise creates a new one
- Create a random order to test for functionality and order placement

These operations and tests will be run against the database in the encapsulated test environment. Database integration testing will be performed from the warehouse after deployment of the new website and the new database connection.

Load Testing

Load testing checks for performance issues before the deployment of the database for the users [16]. It helps with testing the scalability and performance by simulating real-life user load for the database. This checks for database performance if there are multiple users running transactions at the same time. The load test will further check for [16]:

- Response time for database to gather product information
- Response time for executing multiple transactions at the same time and its impact on its performance
- Errors that will arise from higher user load

SQL tests include load, performance, and stress tests for the testing of the database [16]. It is a tool that plays an important role in the deployment cycle. We will write three scripts that will cover all aspects of the SQL test. For example, load and performance testing script and stress testing script performance. These scripts will be written in SQL. These will help with error checking and minimise bugs along the way.

The load testing script will result in checks for higher performance for higher user load, and reduced wait times for data to streamline operations.

The stress testing is designed to make the database fail at one point which is also known as the database breakpoint [16]. The script will then test the database for its breakpoints and generate warnings when the system fails [16]. This will help us fix the errors and increase

database integrity due to higher stress load. All these tests will happen before the launch of the website.

Benefits and Performance of the connection

Once the re-designed website is connected to the database, it will be easier for you to keep track of the customer orders. With the automated shipping labels, it will save you time and resources that were used to manually type and print those shipping labels. Since it will save you time, you can now focus on the content of your website and provide better customer service. Overall, this connection will be cost and time efficient and will effectively streamline the company's operations without any margin of errors.

Backing up current data

To prevent any data loss SSMS also comes with a built-in option to back up the database and its logs. We can choose the option for the full database backup, and we can further choose differential database backup. Differential database backup backs up only the data that has been updated since the last full backup [17]. It is important that the database is fully backed up before selecting differential backup. Since databases and information increase in size, over time they will require more time for full backups [17]. We recommend fully backing up the system every week and follow with differential database backups during the week to save any changes made to the database.

Proposed Schedule

The project schedule, Figure 1 as seen in the Glossary, outlines the tasks and stages of how we will be assessing, redesigning, and deploying the new website. We are following a System Development Life Cycle (SDLC) guide composed of 5 phases. There are two teams

tasked with this project: a website design team and a database programming team. Each team will be participating in each phase but within their specific operations. This project will span a total of 8 weeks with each phase being dependent on one another.

Phase 1: Analysis

Duration: approximately 7 days

Database Programming: Understand Warehouse Structure & Workflow

1.1.1: 7 days

In phase 1.1.1, our database developers will be observing the daily warehouse system and structure. They will take this time to understand the current workflow and day to day processes. They will write a summary report outlining the daily structure and identify where tasks can be streamlined and condensed within the database system.

Website Design: Learn Website Code Base & Understand Company Culture

1.1.2: 7 days

In phase 1.1.2, our designers will be interviewing company managers and customers to better understand the wants and needs of the website. This will give them an opportunity to further understand the company culture to conceptualise a visual aesthetic. They will be taking a closer look at how the current code base is structured and connected to the inventory and sales. Based on these findings, they will create a written report summarising their intended steps to move forward.

Phase 2: Design

Duration: approximately 14 days

Database Programming: Design AWS Integration Layout

1.2.1: 14 days

The database developers will focus on creating shell scripts for three separate functions: login credentials, connecting the AWS to MySQL, and backup data logs. A shell script is a text file that contains skeleton code of how the website will be implemented. This is a planning script which allows developers to trial and error logic in their code to ensure they are utilising efficient methodology and work through any issues.

Website Design: Design Website Wireframe

1.2.2: 4 days

We will be using Figma in phase 1.2.2 to develop 3 distinct wireframes. Our designers will create interactive wireframes with varying functionality to give Rainbow Records options for the layout. They will be presenting these wireframes after 4 days to begin the collaborative process.

Website Design: Feedback and Reiteration

1.2.3: 10 days

Phase 1.2.3 will be a collaborative process where our designers will be taking feedback from the Rainbow Records management to implement the changes, they would like to create a clear design layout. This will be an iterative process which will be conducted through meetings and online video calls. Our team will take notes to adjust the design to meet Rainbow Records' needs while ensuring the website is efficient and effective.

Phase 3: Develop

Duration: approximately 21 days

Database Programming: Connect Database to Website

1.3.1: 21 days

In phase 1.2.1, our team will download the necessary software and create login credentials for all users. They will implement the scripts to connect the website and database. They will then automate the shipping labels and add the database scripts to the warehouse computers.

Website Design: Code Website

1.3.2: 21 days

Phase 1.3.2 will be the complete website design and code implementation. They will use the approved wireframes to build out the desired website for Rainbow Records.

Phase 4: Testing

Duration: approximately 14 days

Database Programming: Testing Database Connections

1.4.1: 14 days

In this phase, our team will be testing separate functions of the database for 14 days. They will create separate test scripts that will be run to ensure the database is secure and responsive. If there are errors, they will implement the necessary changes.

Website Design: Testing Website Search Functions and Buttons

1.4.2: 14 days

This phase will be checking the responsive functionality of the website. Our team will be using test scripts to debug any internal function issues as well as using fake customer orders to check for accuracy and efficiency in merchandise updates.

Phase 5: IT Support/ Maintenance

Duration: approximately 3 months

We will provide IT support and maintenance for 3 months as requested by Rainbow Records. Within this 3-month period, our customer support team will be available via phone and email to troubleshoot any issues, ensuring optimal performance of the new changes.

Training

Within this period, we will provide 3 onsite training sessions for staff on how to operate and maintain the website to increase knowledge of all functions of the website. This will streamline future updates and maintenance efforts.

Reference Documents

We will also create a written and online PDF user manual that will be the point of reference when troubleshooting future problems. There will also be a training guide within the manual as requested by Rainbow Records to outline the essential steps for training future staff.

Conclusion

Our company aims to upgrade the existing website and securely connect it to the AWS website. Upgrading the website will enhance customer interaction with the interface and it will increase its functionality even with a higher customer load on the website. With the new implementations, the website will be less clustered, and customers will be able to manoeuvre merchandise easily providing a positive experience.

Additionally, the connection between the new website and the database will further streamline operations. The connection will be secured by SQL and will prevent any unauthorised logins to the database. The scripts will help in automatic shipping labels instead of printing them manually. This will result in increased time for Rainbow records to upgrade their e-commerce platform to increase customer traffic and sales.

We will also provide ongoing IT support and maintenance for up to three months for any further updates to the website and the database. We will also provide training within the time of IT support; this will train staff on how to operate and maintain the database and the website to streamline the company's operations.

The upgrade of the website and the connection to the AWS database is estimated to take around four months to complete plus three months for ongoing support as requested.

Please contact us at your earliest convenience to schedule a consultation meeting to discuss the further steps in analysing the details about your request and make any further changes to the scheduling to your needs. Alongside Amazon Web Services, we aim to provide Rainbow Records with a reliable and secure software system.

Recommendations

We recommend that Rainbow Records back up their user information and any other important data before or analysis and changes to the legacy system. Since the website is already hosted by AWS, its backup service will automate the data backups. The backups will be stored on cloud so we can easily access it for the warehouse. This will prevent any data loss or any modifications to the existing data without any backups. This will also help us to easily start manipulating the existing website and the database to improve its functionality and streamline the company's operations.

Please take these recommendations into consideration so we can start the initial phase of the software development.

References

- [1] A. S. Gillis, "What is AWS (Amazon Web Services) and how does it work?," SearchAWS, 07-Apr-2020. [Online]. Available: <https://www.techtarget.com/searchaws/definition/Amazon-Web-Services>. [Accessed: 09-Apr-2022].
- [2] M. Stewart, "What are the advantages and disadvantages of Mobile-first design?," *The Creative Momentum . Atlanta Top Web Design Firm*, 10-Apr-2022. [Online]. Available: <https://www.thecreativemomentum.com/blog/what-are-the-advantages-and-disadvantages-of-mobile-first-design>. [Accessed: 09-Apr-2022].
- [3] H. Rajora, "Responsive Design Testing Checklist: All you need to know," *LambdaTest*, 07-Apr-2022. [Online]. Available: <https://www.lambdatest.com/blog/responsive-design-testing-checklist/>. [Accessed: 09-Apr-2022].
- [4] J. Moore, "Why ruby on rails is great for Ecommerce development," *BairesDev*, 21-Jan-2022. [Online]. Available: <https://www.bairesdev.com/blog/ruby-on-rails-ecommerce-development/>. [Accessed: 09-Apr-2022].
- [5] D. Karczewski, "8 reasons to choose Ruby on Rails for your next e-commerce project," *Agile Software Development Agency in Europe*, 27-Nov-2018. [Online]. Available: <https://www.ideamotive.co/blog/ruby-rails-ecommerce-product>. [Accessed: 09-Apr-2022].
- [6] A. S. Gillis, "What is AWS (Amazon Web Services) and how does it work?," SearchAWS, 07-Apr-2020. [Online]. Available: <https://www.techtarget.com/searchaws/definition/Amazon-Web-Services>. [Accessed: 09-Apr-2022].
- [7] "RDS: Understanding animal research in medicine," Amazon, 2007. [Online]. Available: <https://aws.amazon.com/rds/mysql/>. [Accessed: 09-Apr-2022].
- [8] "What is database security: Threats & best practices: Imperva," Learning Center, 08-Nov-2021. [Online]. Available: <https://www.imperva.com/learn/data-security/database-security/>. [Accessed: 09-Apr-2022].
- [9] "What is SQL injection: SQLI attack Example & Prevention Methods: Imperva," Learning Center, 11-Mar-2021. [Online]. Available: <https://www.imperva.com/learn/application-security/sql-injection-sqli/>. [Accessed: 09-Apr-2022].
- [10] "SQL Server Management Studio (SSMS) - SQL server management studio (SSMS)," SQL Server Management Studio (SSMS) - SQL Server Management Studio (SSMS) | Microsoft Docs.

- [Online]. Available: <https://docs.microsoft.com/en-us/sql/ssms/sql-server-management-studio-ssms?view=sql-server-ver15>. [Accessed: 09-Apr-2022].
- [11] "Connect to a MSSQL database," GearHost. [Online]. Available: <https://www.gearhost.com/documentation/connect-to-a-mssql-database>. [Accessed: 09-Apr-2022].
- [12] "What do client side and server side mean? - cloudflare." [Online]. Available: <https://www.cloudflare.com/en-ca/learning/serverless/glossary/client-side-vs-server-side/>. [Accessed: 10-Apr-2022].
- [13] "What is php?," php. [Online]. Available: <https://www.php.net/manual/en/intro-what-is.php>. [Accessed: 09-Apr-2022].
- [14] "Integration testing with Prisma," Prisma. [Online]. Available: <https://www.prisma.io/docs/guides/testing/integration-testing#:~:text=Integration%20tests%20focus%20on%20testing,scenarios%20intended%20to%20be%20tested>. [Accessed: 09-Apr-2022].
- [15] By: IBM Cloud Education, "What is Docker?," IBM. [Online]. Available: <https://www.ibm.com/en/cloud/learn/docker>. [Accessed: 09-Apr-2022].
- [16] Esat Erkec Esat Erkec is a SQL Server professional who began his career 8+ years ago as a Software Developer. He is a SQL Server Microsoft Certified Solutions Expert. Most of his career has been focused on SQL Server Database Administration and Development, "Performing a load test on SQL Server using Apache JMeter," SQL Shack - articles about database auditing, server performance, data recovery, and more, 07-Oct-2020. [Online]. Available: <https://www.sqlshack.com/performing-a-load-test-on-sql-server-using-apache-jmeter/>. [Accessed: 09-Apr-2022].
- [17] LitKnd, "Differential Backup - SQL Server," Differential backup - SQL Server | Microsoft Docs. [Online]. Available: <https://docs.microsoft.com/en-us/sql/relational-databases/backup-restore/create-a-differential-database-backup-sql-server?view=sql-server-ver15>. [Accessed: 09-Apr-2022].