İstanbul
Bilgi Üniversitesi

AlgoLingo

by

Ahmed Ibrahim Attya Hamouda, 120200006
Khaled El Arabi El Azzouzi, 120200125
Uğur Türkel, 119200070

Supervised by

Murat Orhun

Submitted to the
Faculty of Engineering and Natural Sciences
in partial fulfillment of the requirements for the

Bachelor of Engineering

in the
Department of Computer Engineering

January,2024

Abstract

AlgoLingo represents a revolutionary approach in the realm of e-learning, specifically in the context of computer science and algorithms. This report introduces AlgoLingo as a groundbreaking web platform that bridges the gap between complex algorithmic concepts and interactive learning. Targeting one of the most challenging areas in computer science, AlgoLingo simplifies the understanding of algorithms, traditionally seen as a daunting subject, through an immersive gaming experience. The platform employs visual representations and interactive gaming mechanics to transform the learning of algorithms into an engaging and enjoyable activity, departing significantly from conventional, theory-heavy educational methods. AlgoLingo's unique approach combines problem-solving within a game-like environment with practical coding exercises, making it an innovative tool for learners ranging from beginners to advanced developers. This report details the design, development, and theoretical underpinnings of AlgoLingo, highlighting its potential to redefine algorithmic learning and education.

Table of Contents

List of Figures

1 Introduction

AlgoLingo is a new revolutionary website bridging the gap between algorithms and interactive Learning. Algorithms have always been one of the most critical topics in the computer science world, and they serve as the foundational building blocks for everyone from juniors to highly experienced developers. For some people, algorithms are difficult to visualize or comprehend to their full potential. AlgoLingo is a whole new experience that will aid users in learning by making them solve algorithmic problems with games, visual representations, and combine codes with these features to present the finest algorithmic learning system in the market. Imagine stepping into a virtual world where you solve algorithmic puzzles as part of an exciting game. AlgoLingo turns these visions into reality with a roadmap that takes you through all the steps needed to master algorithms. As technology continues to make progress, AlgoLingo surfaces as a groundbreaker in the world of algorithmic learning. It's more than just a platform; it's a revolutionary way to explore algorithms. It introduces a new perspective where algorithms are not just conceptual concepts but a thrilling journey where you play, learn, and improve yourself distinctively from other sources. Instead of learning algorithms by reading dull theories like doing a chore, AlgoLingo makes learning an entertaining and exciting activity. AlgoLingo provides you with various algorithmic games like dragging and dropping an algorithmic sort function into its right place meanwhile teaching code. Therefore, leading to a fun experience for the user.

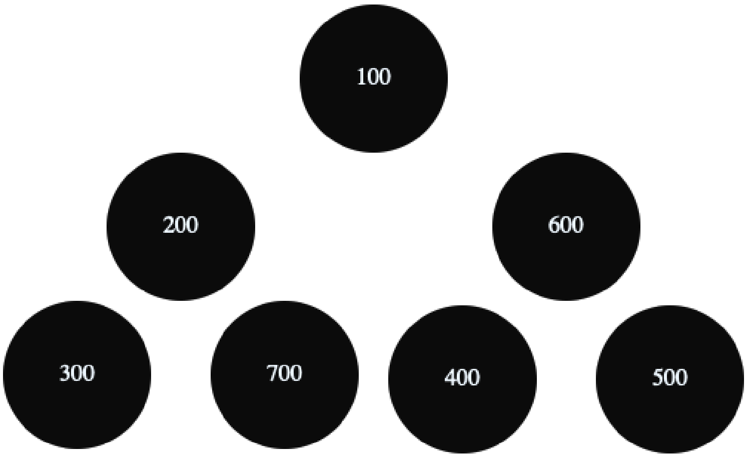

Figure 1: AlgoLingo in action.

2 Literature Review

2.1 History of E-Learning

The history of online classes and e-learning can be traced back to the 1960s when Donald Bitzer, an assistant at the University of Illinois, played a pivotal role in its development. During this era, Bitzer created the first e-learning system known as PLATO. This groundbreaking system laid the foundation for the future of online education, as it was designed for computer-based learning and demonstrated significant networking capabilities.The evolution of e-education gained substantial momentum in the 1980s and 90s. The introduction of the Macintosh computer in the 1980s marked a significant milestone, making personal computers more accessible to a wider audience. This accessibility allowed individuals to learn and acquire new skills in the comfort of their homes, furthering the reach of e-learning.

2.2 Learning Algorithms in Coding With Games

Most online coding learning websites teach direct coding and boring theory. In terms of algorithmic and code learning, there is a variety of instructional applications and learning environments, that include web-based classes, simple games, and face-to-face classes that teach theory and coding directly. There are also applications like ”DuoLingo” and ”Scratch” that facilitate game-like teaching for its users online for a better and improved learning system. Platforms like Scratch aim to teach the user algorithmic learning and learning techniques through game-like drag-and-drop code pieces, AlgoLingo on the other hand will show the user how algorithms work presently, teaching the basics and advanced techniques through games and subsequent coding. ”Scratch” is a block-based programming language designed to be easy to use, it first appeared in 2003 and was developed by MIT Media Lab. One of the first and most effective learning games for coding, ”Scratch” was widely used for beginners to learn the basics of coding. For Algorithmic learning in coding, AlgoLingo will teach users from basic to advanced and make use of various game styles for better visualization and easier comprehension.

Some Examples:
Duolingo: Duolingo extends its gamified approach to coding education, providing an engaging platform for learners. 1

Khan Academy: Khan Academy's coding courses offer a theoretical, and comprehensive learning experience, covering fundamental to advanced coding concepts. 2

Scratch: Scratch uses a block-based programming language to teach coding through creative games. 3

Move Code Lines: Involves rearranging or manipulating lines of code, offering a hands-on experience in understanding code structure and logic. 4

CodeMonkey: CodeMonkey is an educational game platform designed to teach coding through fun and challenging games. 5

CodeWars: Providing a platform for users to improve their coding skills by solving real-world problems and competing with others. 6

CodinGame: CodinGame turns coding into a multiplayer online game, offering a unique, and interactive challenges. 7

CodeCombat: CodeCombat provides an immersive coding experience through a real programming language, making learning to code feel like an adventure with its game-like structure. 8

Lightbot: Lightbot, a puzzle game, introduces programming logic in a playful manner, making it accessible for beginners and younger learners to have coding fundamentals. 9

Screeps: Screeps is a distinctive MMO strategy game where players code their units' AI, creating a hands-on and competitive environment for learning coding and algorithms in a real-time strategy setting. 10

2.3 Benefits of educational coding games

E-learning websites offer many benefits, including the opportunity for repetitive practice to perfect skills, visualization, which helps people understand better, and an interactive approach, facilitating a deeper understanding of the subject and a fun way of learning. In addition to that these are some other benefits:

- Users will be able to visualize their code during every step.
- The drag-and-drop system allows users to manipulate their code however they desire.
- AlgoLingo offers flexibility in terms of when and where users can access the content.

- AlgoLingo can provide immediate feedback on algorithm implementations, helping users identify and correct errors in real time.

- AlgoLingo can provide a more engaging and interactive learning experience making it easier for users to understand and apply algorithms.

3 Methodology

Developing modern software is a complex process that requires a certain level of understanding. The specific technologies and tools used in the development of a web application can vary depending on the complexity of the product, the skills of the developer, and the resources available. All the technologies and tools that are used in this study are implemented to develop an e-learning website in order to make a successful business with the best results. The theoretical portion of this thesis serves as the basis for developing an e-learning website. The information gathered from literature and other sources is used to guide the creation of the e-learning site.

3.1 Theory of modern website architecture

Modern website architecture theory is the peak of principles and practices that guide the designing and developing of websites in the modern era. In developing a website, the first necessity is to decide on whether to go with a static or dynamic approach, this will necessarily cast a considerable effect over the tools and technologies used in development. From these, analysis of the development specific tools and services needed is created with regard to the website's aims. Knowledge of the contrasting features and capabilities of static and dynamic website architectures may help in informing this decision as well as ensuring that the right approach is taken for the project at hand.

A website with fixed content is called a static website. A static website's content is kept in HTML files, which are then shown to the user exactly as they are stored. This indicates that until the website owner or developer manually updates the HTML files, the content on a static website remains unchanged. Informational websites that don't need to be updated or updated frequently are frequently made using static websites. Figure 2 displays a diagram of a static webpage 11.

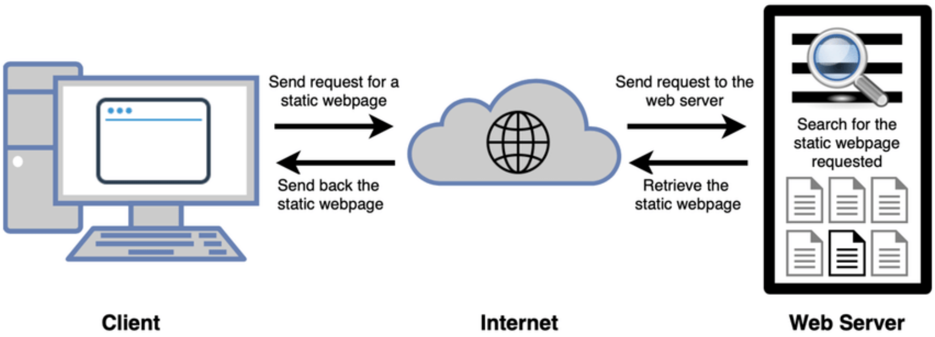
Send request for a static webpage
Send back the static webpage
Send request to the web server
Retrieve the static webpage
Search for the static webpage requested
Client
Internet
Web Server

Figure 2: Static website

However, a dynamic website is one that updates its content automatically in response to user input or other changes. A server-side language like PHP, ASP, or Python powers dynamic websites, enabling them to dynamically retrieve and display data in real-time from a database or other source. This implies that a dynamic website's content may alter regularly based on user activity or other variables. Websites that need to be updated often, like social networking platforms or online commerce, are frequently utilized for dynamic websites. Client-side and server-side scripting are used on dynamic websites to create new content for each visitor each time they visit.

Dynamic websites can minimize server load times and adjust to each user's view of the page by utilizing both forms of scripting. By using scriptings in network penetration testing, vulnerabilities are generally reduced, which can enhance user experience and increase website security. On the other hand, manually updating the HTML files of static websites is necessary to make changes to the content of the website. Figure 3 shows the diagram of a dynamic website.

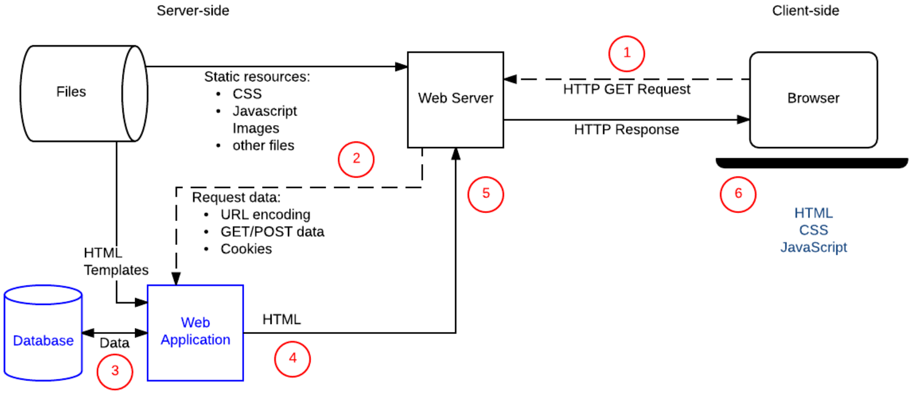

Figure 3: Dynamic website

As there will be interactions between the user and the website, the AlgoLingo project at this point needs an interactive format website. As a result, the decision was made to use Django for the back end and HTML, CSS, and JavaScript for the front end to create a dynamic website.

Web frameworks are widely used by developers these days to facilitate and expedite the process of creating web applications. A web framework is a set of resources and tools that programmers can utilize to create web-based applications. It offers pre-built components and a set of conventions that make it simpler to create web-based applications by giving the process a framework and set of guidelines to follow. By giving developers a base to work upon rather than having to start from scratch, this can save them time and effort.

Web frameworks are frequently used by developers in the following domains: Back-end: Server-side software that delivers websites, applications, and APIs for data exchange. Front-end: The program that operates on clients and in web browsers to give users access to content and interactive features. Web frameworks can be created using a variety of programming languages and approaches. A particular service's collection of several programming languages, applications, and frameworks is referred to as its "stack". An illustration of a popular framework for developing web applications may be found in Figure 4

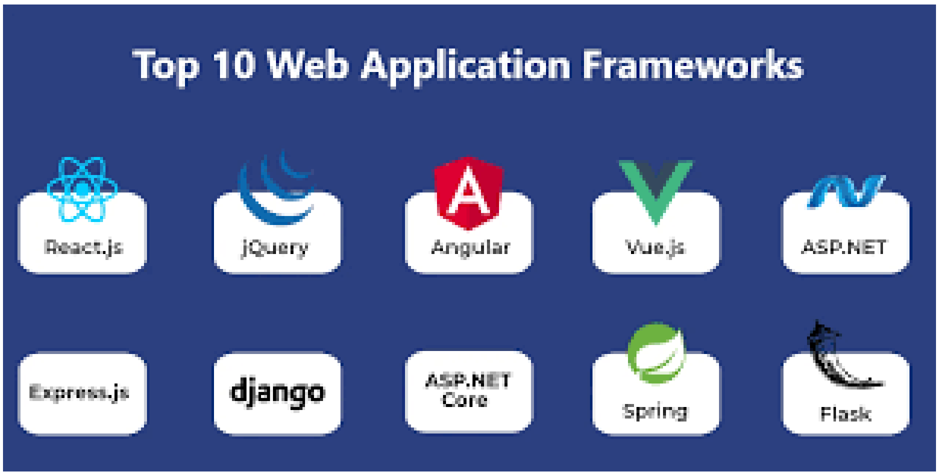

Figure 4: Web Application Frameworks

3.2 Tools and technologies

It is crucial to comprehend the project's requirements when selecting the stack for the job in order to choose the right tools. Comprehending the requirements of the project enhances its performance and success rate, reduces errors, and aids in project maintenance. The technologies and tools utilized in this project are briefly outlined in this section.

3.2.1 Back-end

The portion of a website or online application that is not visible to the user and is in charge of processing, managing, and storing data is known as the back-end. Usually, it is made up of a server, a database, and an application or program that runs on the server and is in charge of overseeing the operation and data of the website. The front-end, or user-interactive portion of the website, usually uses an application programming interface (API) to access the back-end. In response to requests from the front end, the back end sends back the required information or carries out the desired actions.

Django is a popular Python web framework known for its high-level abstractions, promoting rapid development by reducing boilerplate code. Its modular structure ensures scalability, and built-in security features protect against common vulnerabilities. With a large community and comprehensive documentation, Django simplifies database interactions through an Object-Relational Mapping (ORM) system. Following a "batteries-included" philosophy, it offers built-in features like authentication and URL routing, making it a versatile and efficient choice for backend development.

Furthermore, a relational database is required to manage the website's functionalities. Although there are numerous solutions available, it was determined after some investigation that MySQL is the most appropriate for these kinds of projects. Based on the Structured Query Language (SQL), MySQL is a well-known open-source relational database management system (RDBMS). It is frequently used because of its great performance, dependability, and simplicity of use to store data for web-based applications. Data in a MySQL database is arranged into tables for easy access, management, and updating with SQL commands.

3.2.2 Front-end

The front-end of a website or web application is the part of the site that is supposed to be visible to the user, and it is responsible for the look and feel as well as the user experience of the site. As the first interaction between the web operator and the user gets Front-end application will be the front-end applications, thus it is of utmost importance to design and develop the responsive, interactive, user friendly and appealing layout that will in one way or another influence the overall impression of the site and will increase the willingness to work with. it. sometimes it even improves the web performance by optimization of content loading and rendering. The three foundations on which front-end development firmly stands are: HTML for markup, CSS for styling and JavaScript for logic as well as interactions.

HTML is an acronym for HyperText Markup Language. It's a standard markup language used in web page development. It makes it possible to present grouped sections, paragraphs and information through HTML elements (the building blocks of the page) such as tags and attributes. HTML itself is built up from a collection of elements identified by tags that help to structure the contents of a web page. Thus, an <h1>element identifies a heading, the ¡p¿element is used to describe a paragraph while the <div>element is used in grouping other elements so as to form a section. HTML forms the backbone of the web acting like a building block. and most of the contents. It's a simple, yet effective, language that is easy to learn and used by web developers in creating everything from simple static websites to complex, dynamic web-based applications.(12)

CSS stands for Cascading Style Sheets and is a style sheet language used to describe the appearance and formatting of a document written in HTML. CSS controls how contents are presented on a web page, and it's therefore typically used to define styles for a group of HTML elements. For example, you can use CSS to specify the font, colour, size and spacing of the text on a web page, or to specify the size, colour and background of a button.(13)

Finally, to combine HTML and CSS together in some sort of way to add interactivity and dynamic behaviour to websites, JavaScript is the best option for that. It's a client side language meaning that it gets executed by the web server, but rather on the user's web browser. JavaScript is often used to create interactive elements on a web page, such as forms that can either be validated without the need to submit the page to the server or even be enhanced with a little animation and effects on the other pages.

JavaScript is a powerful and flexible language, which is frequently used by many web developers. This language forms an important part of modern interactive site creating, in addition to such technologies as HTML and CSS to create engaging and user-friendly web experiences. Moreover, JavaScript libraries are groups of pre-written JavaScript code which any developer uses as a way to implement a specific functionality on his/her website. Such libraries make it easier for the user to create and add desired features. In general, one may say that to implement complex features or functionality without having to code everything from scratch. They usually have a simple, easy-to-understand interface that grants developers the ability to do and cover a lot in just a few lines of code.

3.3 Development Approach

3.3.1 Waterfall Method

The waterfall methodology shown in figure 5, also dubbed the Waterfall model, is a sequential design process that flows like a waterfall through all stages of a project (an analysis, design, development, and testing, for instance), with each stage fully completing, before the next one begins.

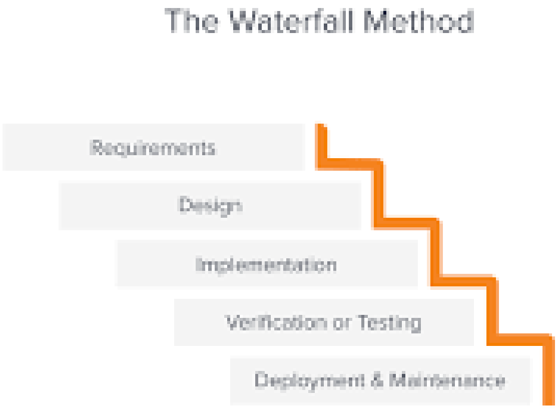

Figure 5: Waterfall Method

The Waterfall methodology has been said to follow the adage to "measure twice, cut once." Success of the Waterfall method will depend on the amount and quality of front end work that was done documenting everything in advance, including the user interface, their stories and all variations and outcomes of the features.

Most of the background research working has been done, therefore, the estimates regarding the period required for each requirement are highlighted more exactly which can provide a more predictable delivery date. With a Waterfall project, if along the way parameters change it alerts to change, it's harder to change course than with Agile methodology. The choice of the waterfall methodology comes perfect in this study because its suitable for small start up projects where the client feedback is not necessary as well they will be able to trap design errors during the analysis and write an error free code.

4 Design

4.1 Web Application Workflow

Starting with the landing page, users will be able to sign up or log in to AlgoLingo. After that users will see a list of courses to practice on. Then they will see the levels they have locked or unlocked based on their progress in the selected course.
in more detail, customers can:

- view account information
- edit account information
- browse courses
- bookmark levels and/or courses
- view history

figure 6 shows the website's flowchart

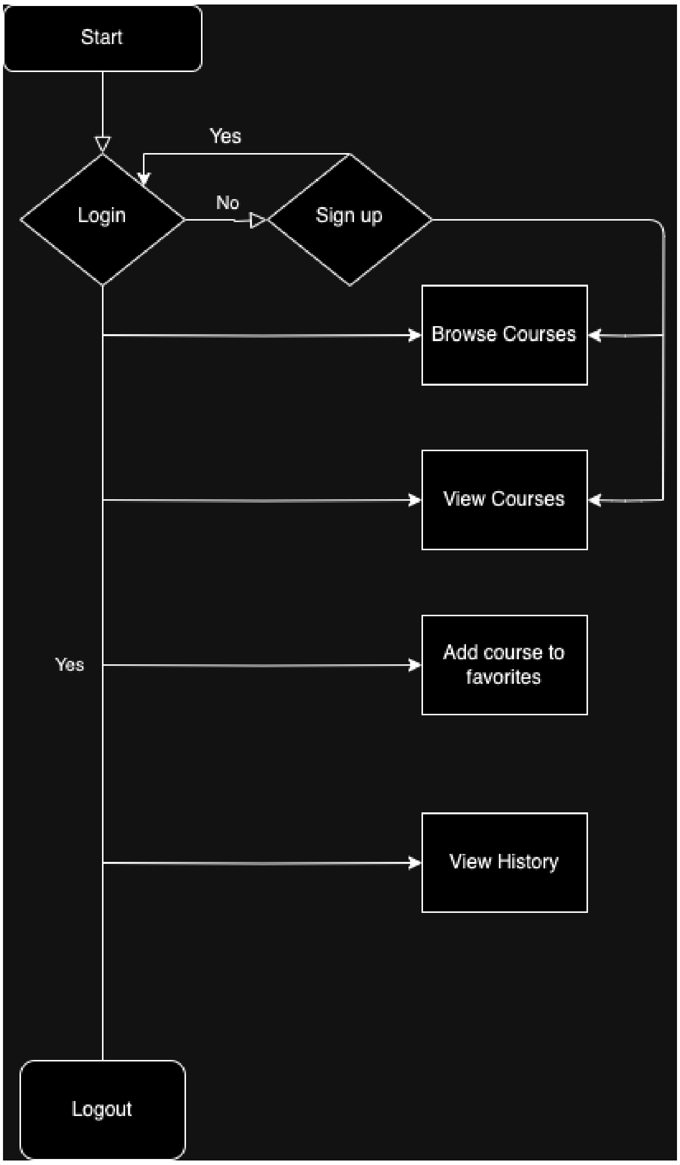

Figure 6: Flowchart

Here it shows the user's path when opening the web application. It starts by asking the user to log in, if the user does not have an account it takes them to the sign up page. if the user decides to not make an account they are still able to browse and view the courses available. If the user signs up for AlgoLingo, it takes them back to the log in page. After logging in the user will be able to browse, view, add courses to favorites and view their history.

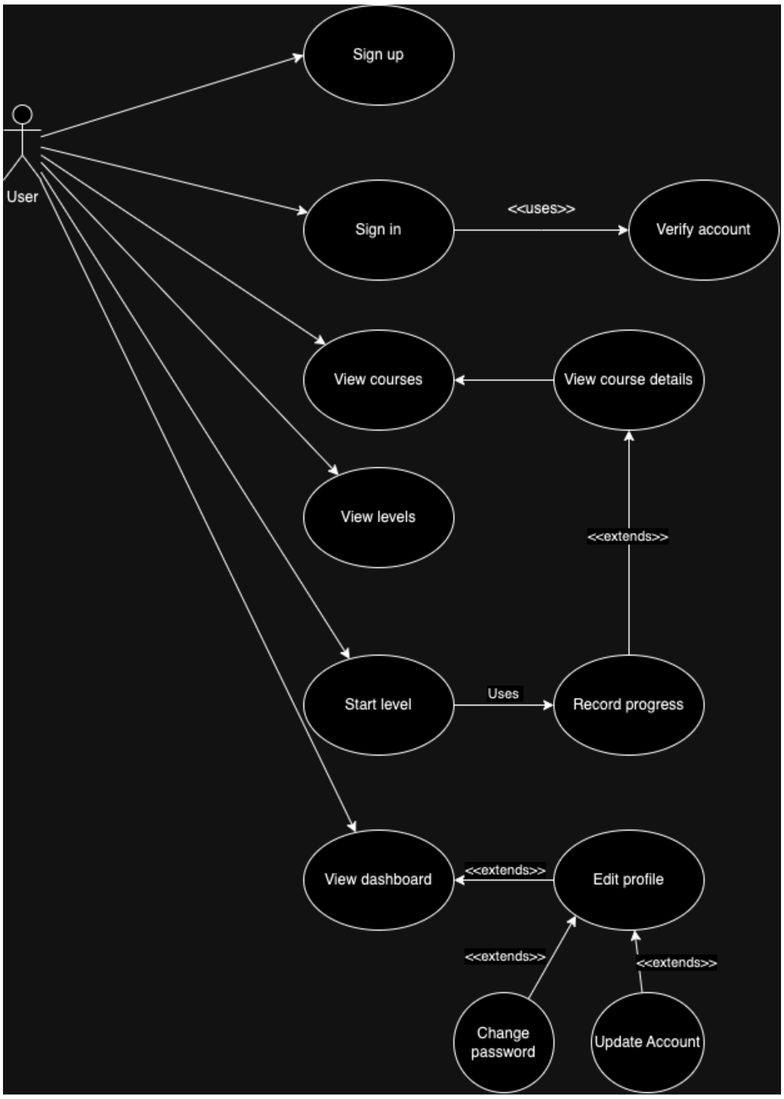

Figure 7: Use case diagram

This use-case diagram shows what the user will be able to do when opening the web application. users will be able to sign up, log in, view courses, view levels, start levels and view their dashboard. When viewing a course, users will be able to see the course details. After starting a level, the user's progress will be recorded, and when viewing the dashboard, users will be able to edit their profiles whether it is to change their password or their usernames.

5 Risk Analysis

SQL Injection Attacks: How it Works: This type of attack occurs when an attacker injects malicious SQL code into a website's input fields or URL parameters. The goal is to manipulate the website's database, allowing the attacker to access, modify, delete, or steal data. Impact on an E-Learning Website: SQL injection can compromise user data including personal information, login credentials, and payment information if applicable. It can also corrupt or erase course content, student records, and other critical data stored in the website's database.(14)

Cross-Site Scripting (XSS) Attacks: How it Works: XSS attacks involve injecting malicious scripts into web pages viewed by other users. These scripts run in the context of the user's browser, allowing the attacker to steal cookies, session tokens, or other sensitive information directly from the user's browser. Impact on an E-Learning Website: XSS can be used to hijack user sessions, deface the website, redirect users to malicious sites, or manipulate on-page content. This can lead to a loss of trust and credibility among users, potentially damaging the reputation of the e-learning platform.(15)

Distributed Denial of Service (DDoS) Attacks: How it Works: In a DDoS attack, a multitude of compromised systems attack a single target, such as a server, causing a denial of service for users of the targeted system. The flood of incoming messages to the target system essentially forces it to shut down, denying service to the system's legitimate users. Impact on an E-Learning Website: A DDoS attack can make the e-learning website inaccessible to learners and instructors. Prolonged downtime can disrupt learning processes, cause frustration among users, and lead to a loss of users if the issue is not resolved quickly.(16)

The content included in AlgoLingo will be updated regularly to keep in accordance to the questions asked by interviewers when applying to a job.

6 Conclusion

The main objective of this thesis was an implementation of an e-learning web application designed for learning algorithms. The entire process of the development was explained, starting from the literature review 2, showing the history of e-learning and the benefits of it, continuing with methodology 3, describing and explaining the modern website architecture, the tools and technologies used and the development approach. Following the methodology, the next section was design 4, focused on the design layout of the web application, briefly going through technologies which were used and briefly describing the workflow of the web application. After the design risks regarding our projects were spoken about to raise awareness for future updates and projects. To conclude, this thesis has shown the process of developing an e-learning web application. the field of developing web applications is very huge and therefore is still a lot of work to do in the future.

References

[1] Xiangying Jiang, Joseph Rollinson, Luke Plonsky, Erin Gustafson and Bozena Pajak (2020) DuoLingo final report. http://static.duolingo.com/s3/DuolingoReport_Final.pdf

[2] KhanAcademy: Scaling and simplifying. https://cloud.google.com/customers/khan-academy

[3] Scratch, MIT Media Lab (2003) http://www.scratchjr.org/pdfs/scratchjr-interface-guide.pdf

[4] Haoose(2020).https://steamcommunity.com/sharedfiles/filedetails/?id=2213531683

[5] CodeMonkey: Coding for kids, Jonathan Schor (2014). https://www.codemonkey.com/

[6] CodeWars: Achieve mastery through challenge. https://www.codewars.com/

[7] CodinGame: Learn the programming fundamentals by playing a game. https://files.codingame.com/codingame/learning/CodinGame_for_Teachers_Onboarding_Lesson_2.pdf

[8] CodeCombat: Learn to code through the power of play, Nick Winter(2013).https://files.codecombat.com/presspack/AboutCodeCombat.pdf

[9] LightBot (2017). https://lightbot.com/LightbotSolns.pdf

[10] Screeps. https://bencbartlett.wordpress.com/2019/01/02/screeps-6-verifiably-refreshed/

[11] https://developer.mozilla.org/en-US/docs/Learn/Server-side/First_steps/Introduction

[12] https://www.hostinger.com/tutorials/what-is-html

[13] https://www.tutorialspoint.com/css/what_is_css.htm

[14] https://www.linode.com/docs/guides/sql-injection-attack/#:~:text=,SQL

[15] https://cheatsheetseries.owasp.org/cheatsheets/Cross_Site_Scripting_Prevention_Cheat_Sheet.html#:~:text=,Framework%20Security

[16] https://www.apriorit.com/dev-blog/559-ddos-protection-techniques#:~:text=,Apriorit%E2%80%A0www.apriorit.com%E3%80%91