



Alliance with FPT Education

UNIVERSITY OF GREENWICH

COMP1682

Final Year Project

Student name	Nguyen Duy Quang
ID number	001353659
Lecturer/Tutor name	Thai Minh Tuan
Student submission date	April 29 th , 2024

Undergraduate Final Year Project
Developing A Social Network Application for IT Experts
NGUYEN DUY QUANG

Table of Contents

CHAPTER 1: INTRODUCTION	1
1. Overview	1
2. Purpose	2
3. Scope	2
4. Project Plan	2
CHAPTER 2: LITERATURE REVIEW.....	3
1. Cross-platform development vs native development	3
2. The impact of Social Media on Modern Society	4
3. Existing social networks aspects analysis	5
4. Programming language and Framework	6
4.1 Java	6
4.2 TypeScript	7
4.3 Dart	8
4.4 Spring Boot	9
4.5 Angular.....	11
4.6 Flutter	13
5. Database (Postgresql)	14
6. Other technologies.....	16
6.1 Bootstrap	16
6.2 Kafka	16
6.3 Keycloak.....	17
6.4 Digital Ocean.....	17
6.5 Cloudinary.....	17
6.6 Traefik	18
7. Conclusion	18
CHAPTER 3: DESIGIN	19
1. Database schema	19
2. Sitemap	31
3. Usecase diagram	32
3.1 User role	32
3.2 Admin role	33
4. Flow Chart	34

4.1	Register	34
4.2	Login.....	35
4.3	Reset password.....	35
4.4	Search friend.....	36
4.5	Update profile.....	37
4.6	Change password.....	37
4.7	Create new post.....	38
5.	Mockup UI design	39
5.1	Register	39
5.2	Login.....	39
5.3	Home.....	40
5.4	Chat message.....	40
5.5	Friend	41
5.6	Notification	41
5.7	Profile.....	42
5.8	About user	43
5.9	Friend list	44
CHAPTER 4:	METHODOLOGIES AND SOFTWARE DEVELOPMENT PROCESS	44
1.	Software development process	44
2.	Software development methodologies	45
3.	Conclusion	46
CHAPTER 5:	IMPLEMENTATION AND EXPLAINATION.....	47
1.	Backend.....	47
1.1	Folder structure	47
1.2	Setup environtment.....	48
2.	Frontend.....	51
2.1	Folder structure	51
2.2	Config proxy	52
2.3	Function and interface.....	53
3.	Deploy Digital Ocean.....	72
3.1	Create droplet.....	72
3.2	Config environment	73
CHAPTER 6:	TESTING AND EVALUATION	76
1.	Testing	76

1.1	Functions.....	76
1.2	Performance	87
2.	Evaluation.....	89
2.1	Self-evaluate	89
2.2	Survey from users	90
CHAPTER 7: CONCLUSIONS AND FUTUREWORK		97
1.	Conclusion.....	97
2.	Futurework.....	97
REFERENCES		98
APPENDIX		102

List of Figures

Figure 1.	Gantt chart (1)	2
Figure 2.	Gantt chart (2)	2
Figure 3.	Database schema.....	19
Figure 4.	Sitemap website.....	31
Figure 5.	Use case of user role	32
Figure 6.	Usecase of admin role.....	33
Figure 7.	Register function.....	34
Figure 8.	Login function	35
Figure 9.	Reset password function.....	35
Figure 10.	Search friend function.....	36
Figure 11.	Update profile function.....	37
Figure 12.	Change passwrod function.....	37
Figure 13.	Create new post function.....	38
Figure 14.	Register page	39
Figure 15.	Login page.....	39
Figure 16.	Home page.....	40
Figure 17.	Chat message page.....	40
Figure 18.	Add Friend page.....	41
Figure 19.	Notification page	41
Figure 20.	Profile page.....	42
Figure 21.	About user page.....	43
Figure 22.	Friend list page.....	44
Figure 23.	Software development process model.....	45
Figure 24.	Depicting the implementation in development life cycle	46
Figure 25.	Backend's folder structure	47
Figure 26.	Frontend's folder structure	51
Figure 27.	Register page	53
Figure 28.	Display error when sign up (1)	53
Figure 29.	Enter do not enough field when siging up	53
Figure 30.	Enter password and confirm password do not match	54
Figure 31.	Display message when signing up successfully.....	54
Figure 32.	Email verify account.....	54
Figure 33.	Login page.....	55
Figure 34.	User have to verify email before login	55
Figure 35.	Incorrect credentials	55
Figure 36.	Forgot password page	56
Figure 37.	User does not exist in system.....	56
Figure 38.	Check email to reset password	57
Figure 39.	Email to reset password.....	57
Figure 40.	Enter new password and confirm password	57
Figure 41.	Home page.....	58
Figure 42.	About user page.....	59

<i>Figure 43.</i>	<i>Update personal information.....</i>	59
<i>Figure 44.</i>	<i>Enter all information when edit profile</i>	60
<i>Figure 45.</i>	<i>Edit profile successfully</i>	60
<i>Figure 46.</i>	<i>Manage experience</i>	61
<i>Figure 47.</i>	<i>Add new work experience</i>	62
<i>Figure 48.</i>	<i>A work was added.....</i>	62
<i>Figure 49.</i>	<i>Add more work experience</i>	62
<i>Figure 50.</i>	<i>Add new education</i>	63
<i>Figure 51.</i>	<i>New education was added.....</i>	63
<i>Figure 52.</i>	<i>Manage skills</i>	64
<i>Figure 53.</i>	<i>Add new skill</i>	64
<i>Figure 54.</i>	<i>New skills was added</i>	65
<i>Figure 55.</i>	<i>Hover to change avatar</i>	65
<i>Figure 56.</i>	<i>Choose new avatar</i>	65
<i>Figure 57.</i>	<i>Choose new avatar and crop it</i>	66
<i>Figure 58.</i>	<i>New avatar was saved.....</i>	66
<i>Figure 59.</i>	<i>Hover to change cover</i>	67
<i>Figure 60.</i>	<i>Choose new cover</i>	67
<i>Figure 61.</i>	<i>Choose new cover and crop it</i>	68
<i>Figure 62.</i>	<i>New cover was saved.....</i>	68
<i>Figure 63.</i>	<i>View friend.....</i>	69
<i>Figure 64.</i>	<i>Send friend request.....</i>	69
<i>Figure 65.</i>	<i>Receive friend request.....</i>	69
<i>Figure 66.</i>	<i>List friends of user.....</i>	70
<i>Figure 67.</i>	<i>New conversation</i>	70
<i>Figure 68.</i>	<i>Message from current user.....</i>	71
<i>Figure 69.</i>	<i>Message from friend of user.....</i>	71
<i>Figure 70.</i>	<i>Notification</i>	72
<i>Figure 71.</i>	<i>Create droplet and assign domain</i>	72
<i>Figure 72.</i>	<i>Access to the server by using SSH</i>	73
<i>Figure 73.</i>	<i>.env file.....</i>	73
<i>Figure 74.</i>	<i>Docker compose file (traefik)</i>	73
<i>Figure 75.</i>	<i>Docker compose file (frontend and backend)</i>	74
<i>Figure 76.</i>	<i>Docker compose file (database and pgadmin)</i>	74
<i>Figure 77.</i>	<i>Docker compose file (keycloak and keycloak database)</i>	75
<i>Figure 78.</i>	<i>Docker compose file (kafka).....</i>	75
<i>Figure 79.</i>	<i>Docker logs.....</i>	76
<i>Figure 80.</i>	<i>Analysis all performance.....</i>	87
<i>Figure 81.</i>	<i>Testing performance</i>	87
<i>Figure 82.</i>	<i>Testing accessibility</i>	88
<i>Figure 83.</i>	<i>Testing best practice</i>	88
<i>Figure 84.</i>	<i>Testing SEO</i>	89
<i>Figure 85.</i>	<i>Question 1.....</i>	90
<i>Figure 86.</i>	<i>Question 2</i>	91

<i>Figure 87.</i>	<i>Question 3.....</i>	91
<i>Figure 88.</i>	<i>Question 4.....</i>	92
<i>Figure 89.</i>	<i>Question 5.....</i>	92
<i>Figure 90.</i>	<i>Question 6.....</i>	93
<i>Figure 91.</i>	<i>Question 7.....</i>	93
<i>Figure 92.</i>	<i>Question 8.....</i>	94
<i>Figure 93.</i>	<i>Question 9.....</i>	95
<i>Figure 94.</i>	<i>Question 10.....</i>	96

List of Tables

<i>Table 1.</i>	<i>Comparing the Fundamental Characteristics of Various Types of Social Networks Described.</i>	5
<i>Table 2.</i>	<i>Pros and Cons of Java.....</i>	6
<i>Table 3.</i>	<i>Pros and Cons of TypeScript</i>	7
<i>Table 4.</i>	<i>Pros and Cons of Dart.....</i>	8
<i>Table 5.</i>	<i>Pros and Cons of Spring Boot</i>	10
<i>Table 6.</i>	<i>Pros and Cons of Angular</i>	11
<i>Table 7.</i>	<i>Pros and Cons of Flutter</i>	13
<i>Table 8.</i>	<i>Pros and Cons of Postgresql</i>	15
<i>Table 9.</i>	<i>User data desctiption</i>	20
<i>Table 10.</i>	<i>User profile data desctiption.....</i>	20
<i>Table 11.</i>	<i>User hobbies data description</i>	21
<i>Table 12.</i>	<i>User experiences data desctiption</i>	21
<i>Table 13.</i>	<i>User skills data desctiption</i>	22
<i>Table 14.</i>	<i>User avatar data desctiption</i>	22
<i>Table 15.</i>	<i>User cover desctiption.....</i>	23
<i>Table 16.</i>	<i>Friend request data desctiption</i>	23
<i>Table 17.</i>	<i>Friendship data desctiption.....</i>	23
<i>Table 18.</i>	<i>Conversations data desctiption.....</i>	24
<i>Table 19.</i>	<i>Participants data desctiption.....</i>	24
<i>Table 20.</i>	<i>Message data description.....</i>	25
<i>Table 21.</i>	<i>Message attachment data description</i>	25
<i>Table 22.</i>	<i>Album data description.....</i>	26
<i>Table 23.</i>	<i>Album media data description.....</i>	26
<i>Table 24.</i>	<i>Posts data description.....</i>	27
<i>Table 25.</i>	<i>Post media data description</i>	27
<i>Table 26.</i>	<i>Post like data description.....</i>	28
<i>Table 27.</i>	<i>Post comments data description</i>	28
<i>Table 28.</i>	<i>Reply comments data description</i>	29
<i>Table 29.</i>	<i>Like comments data description</i>	29
<i>Table 30.</i>	<i>Notification data description</i>	30
<i>Table 31.</i>	<i>User activities data description</i>	30
<i>Table 32.</i>	<i>Test case</i>	76

CHAPTER 1: INTRODUCTION

1. Overview

Social network applications tailored to technology experts have gained prominence as the technology sector continues its rapid growth. This report explores the landscape of these specialized platforms, examining their motivations, challenges, and critical success factors. These networks are essential in connecting professionals, sharing knowledge, and staying updated in a rapidly evolving field (Smith, 2021).

The rise of social networks for technology experts is driven by the need to stay abreast of rapid technological changes. These networks offer a hub for knowledge exchange and best practices. Additionally, technology experts benefit from niche platforms that facilitate connections with peers who share similar skills and interests (Smith, 2021).

While the development of these networks is promising, it's not without challenges. Maintaining relevance in a fast-changing industry is critical. Data security and privacy are concerns due to the sensitive nature of the information shared. Attracting and retaining users requires a thoughtful strategy, balancing community building with professional networking (Brown, 2020).

Success in technology expert social networks relies on user engagement, achieved through features encouraging meaningful interactions and knowledge sharing. An effective content curation system, user-friendly interface, data security, and trust-building measures are vital. Fostering a sense of community where experts feel valued and heard is crucial for long-term success (Williams, 2019).

In conclusion, social network applications for technology experts are vital for professional development and networking in a dynamic industry. Overcoming challenges through user engagement, trust, and community building is key to success. As technology continues to evolve, these platforms remain essential for connecting, learning, and growing within the field (Johnson, 2018). The “Develop Social Network Applications for Technology Experts” initiative aspires to create a pivotal online domain wherein technology professionals can engage, collaborate, and remain apprised of advancements in their respective domains. The fulfillment of this vision is expected to make substantial contributions to technological progress and innovation. This report serves as a comprehensive guide to the project's goals, objectives, and the strategies devised for its execution.

2. Purpose

The project aims to create a user-friendly and engaging online multiple platform that fosters connections, discussions, and opinion sharing among its users.

3. Scope

Users have direct access to online posts within the system, where they can engage by liking or commenting. They can also connect with others, making friends and engaging in conversations.

4. Project Plan

As seen in Figures 1 and 2, I have created a Gantt chart below using the time estimates and tasks.

- **Start day:** October 28th, 2023
- **Finish day:** May 15th, 2024

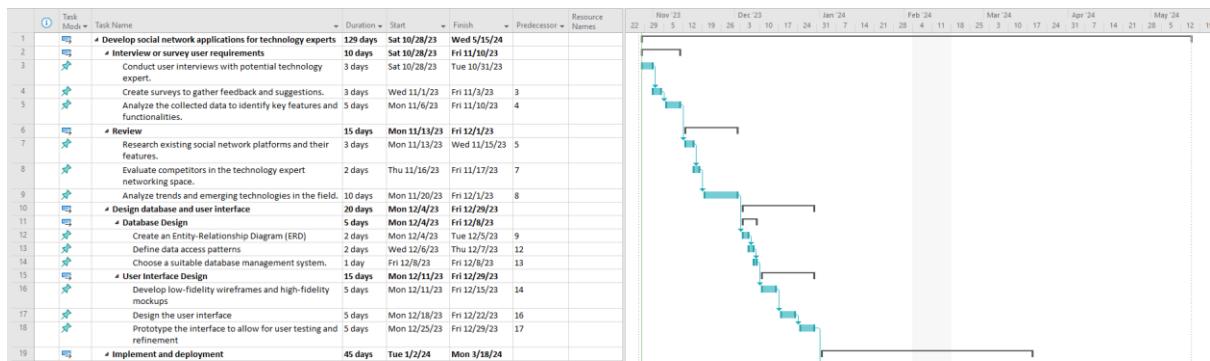


Figure 1. Gantt chart (1)

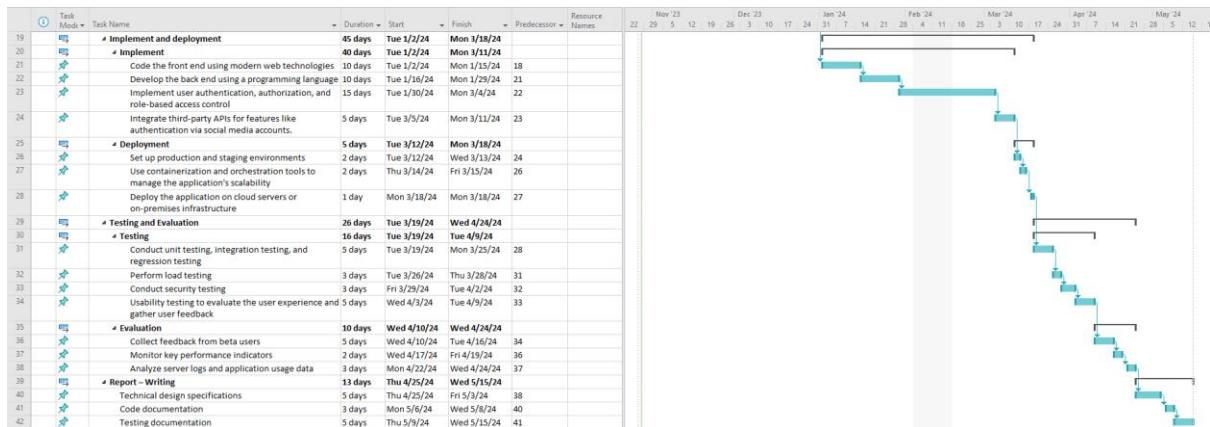


Figure 2. Gantt chart (2)

CHAPTER 2: LITERATURE REVIEW

1. Cross-platform development vs native development

In today's online social landscape, numerous social networks have emerged, each thriving in its own way. Users often engage with multiple platforms simultaneously, acting as bridges between these networks (Zhang and Philip, 2018). This interconnectedness, termed aligned social networks, allows researchers and practitioners to analyze users' behaviors across platforms and study information propagation.

Understanding users' activities across multiple networks offers a comprehensive view, enabling personalized services. Moreover, it enhances our grasp of information dissemination, benefiting various applications. However, linking users' accounts across platforms remains a challenge, crucial for effective cross-platform services such as friend recommendations and information propagation analysis.

Native mobile applications (NMAs) are software programs tailored for a particular mobile platform like iOS or Android. These apps are built using platform-specific languages and tools, maximizing performance by directly accessing device capabilities such as cameras and GPS. Examples include Instagram for Android and Facebook for iOS, which leverage native development to offer seamless user experiences (Josh, 2024).

Native mobile applications offer several advantages, including improved performance due to optimized resource utilization and platform-specific frameworks. They also leverage unique device features, enhancing functionality for users. With a focus on specific platforms, native apps deliver a smoother user experience with fewer bugs and better interface integration. Additionally, native development simplifies the implementation of offline functionality, making it ideal for apps requiring offline capabilities. Furthermore, native tools often provide stronger security measures through platform-specific updates, ensuring better protection against vulnerabilities. Overall, native mobile app development stands out for its reliability and ability to deliver superior performance, user experience, offline functionality, and security (Josh, 2024).

In summary, the rise of interconnected social networks offers researchers valuable insights into user behavior and information dissemination. However, linking user accounts across platforms remains a challenge for effective cross-platform

services. Meanwhile, native mobile applications excel in performance, user experience, offline functionality, and security, thanks to their platform-specific optimization. As we navigate this digital landscape, the integration of aligned social networks and the advancement of native apps promise to reshape our digital interactions significantly.

2. The impact of Social Media on Modern Society

A social network, alongside its corresponding website, empowers users to craft profiles and establish authorized accounts, thereby creating digital representations of themselves. Users can select other members of the site as contacts, establish connections with them, and engage in various social activities. These platforms encompass personal details, group affiliations, friends lists, contact information, business connections, personal preferences, and historical data. In today's era of pervasive connectivity and technological advancement, individuals are seamlessly integrated into social networks through their mobile smartphones, whether at work, home, or elsewhere. This constant connection ensures users are promptly informed of new posts, messages, and current news, fostering a constructive environment where individuals stay abreast of the latest developments in technology and beyond.

Social networking sites like Facebook, MySpace, YouTube, and Bebo surged in popularity during the early 21st century, prompting questions about their longevity and user demographics. While concerns about privacy, inappropriate content, and excessive online time for youth arose, systematic research on these platforms was hindered by their commercial nature and proprietary data. However, studies on platforms like MySpace provided insights into user behavior and demographics. Despite privacy challenges, social network data holds promise for understanding social behavior and providing valuable insights, though it necessitates addressing security concerns. Recent advances propose data sanitation techniques to balance data utility with privacy protection. As social networks increasingly shape social interactions and information dissemination, understanding their structure and applications becomes crucial, with platforms like Sina Weibo and Tencent Wechat playing significant roles (Rath, 2022).

3. Existing social networks aspects analysis

A social network comprises individuals, organizations, or other social entities interconnected through various relationships such as friendship, collaboration, or information exchange. Social network analysis delves into the examination of these relationship patterns among people, organizations, states, and similar entities, offering both visual and mathematical insights into human connections.

The Internet, particularly the Web, can be viewed as a vast social network. Within this framework, documents serve as nodes in the sociogram, while the links between them represent edges. Weblogs, a distinct subset of the Web, can also be seen as integral components of this social network (Jamali and Abolhassani, 2006).

Emerging concepts like the Semantic Web (SW) introduce the notion of structuring and interlinking data on the Web to make it usable for both humans and machines (Jamali and Abolhassani, 2006).

The Semantic Web and social network models complement each other, with Table 1 illustrating the fundamental properties of various types of social networks discussed herein and highlighting their differing formations (Jamali and Abolhassani, 2006).

Table 1. Comparing the Fundamental Characteristics of Various Types of Social Networks Described.

Social Network Type	Actors	Ties	Direction
Friendship Network	People in Society	Friendship relations between people	Undirected
Web's Social Network	Web Pages	Links Between Web Pages	Directed
Semantic Web Social Networks	Semantic Web Docs or Concepts in Them	Semantic Relations Between Documents or Concepts	Directed or Undirected

4. Programming language and Framework

4.1 Java

Java, a programming language and computing platform initially introduced by Sun Microsystems in 1995. It serves as a robust foundation for numerous services and applications, contributing substantially to the functioning of the modern world (java, 2024).

Renowned for its versatility, Java is a multi-platform, object-oriented, and network-centric language that also functions as a platform in its own right. Its attributes include speed, security, and reliability, making it a preferred choice for developing a wide array of software, ranging from mobile applications and enterprise solutions to big data applications and server-side technologies (amazon, 2024).

Table 2. Pros and Cons of Java

Pros	Cons
<ul style="list-style-type: none">Easy-to-learn language: Java stands out as an easy-to-learn language due to its simple syntax and high-level nature, catering to both novice and seasoned developers alike.Object-Oriented Programming (OOP): The object-oriented programming (OOP) paradigm in Java offers unparalleled advantages in terms of code organization, maintenance, and scalability, making it a preferred choice for developers.Enhanced Security: With its robust security features including encapsulation, inheritance, abstraction, and a built-in Security Manager, Java is highly regarded for developing secure applications.	<ul style="list-style-type: none">Performance Limitations: While Java offers ease of development, its performance may sometimes lag behind native languages like C++ due to factors such as the overhead of its virtual machine and memory management. Challenges such as garbage collection, suboptimal caching configurations, and potential thread deadlocks can impact performance if not carefully addressed.Licensing Costs: Businesses may find Java's licensing expenses, especially Oracle's commercial license terms, to be a potential financial burden.GUI Development Challenges: Java's GUI development process can pose challenges due to inconsistencies

<ul style="list-style-type: none"> Platform Independence: Java's platform independence is a cherished feature that allows developers to write code once and execute it on any platform with a Java Virtual Machine (JVM), making it an ideal choice for various projects such as web applications, mobile apps, cloud computing, and big data initiatives (Rafalski, 2023). 	<p>and a lack of native aesthetics, which may require additional effort and resources to overcome (Rafalski, 2023).</p>
---	---

4.2 TypeScript

TypeScript, an open-source programming language developed by Microsoft, enhances JavaScript by incorporating a robust type system. Widely utilized in conjunction with popular front-end frameworks like Angular, Vue, and React, TypeScript's versatility extends to backend development with Node.js, showcasing the adaptability of its type system (Zaitsev, 2023).

In its relationship with JavaScript, TypeScript offers all the features of the latter while introducing an additional layer: its sophisticated type system. Unlike JavaScript, TypeScript ensures type consistency, thereby enhancing code reliability (typescriptlang, 2024).

Table 3. Pros and Cons of TypeScript

Pros	Cons
<ul style="list-style-type: none"> Code Maintainability: TypeScript promotes a structured coding approach with explicit type annotations, enhancing code self-documentation. Enhanced Developer Productivity: TypeScript integrates seamlessly with IDEs and code editors, offering features like autocompletion, code navigation, and refactoring that 	<ul style="list-style-type: none"> Additional Code: TypeScript's adoption of static typing necessitates the inclusion of type annotations, potentially elongating development time, especially for smaller projects or during the initial development phases. Nuances of Use: While TypeScript offers significant benefits, it can pose challenges, particularly in complex

<p>significantly improve developer productivity.</p> <ul style="list-style-type: none"> Improved Collaboration: TypeScript facilitates better collaboration among developers, particularly in multi-member projects. Scalability: TypeScript proves advantageous for large-scale projects, where the ability to define and enforce types becomes crucial as the codebase expands. Reduced Number of Bugs: TypeScript's static type checking plays a vital role in identifying potential errors and bugs early in the development process (Zaitsev, 2023). 	<p>scenarios or when determining the correct types is ambiguous. In such instances, developers may opt to bypass type declarations to meet project deadlines or due to difficulties in finding suitable typing solutions.</p> <ul style="list-style-type: none"> Adding Optional Technology: Effective utilization of TypeScript within a project requires developers to possess a solid understanding of the language, ensuring optimal productivity and proficiency (Zaitsev, 2023).
---	--

4.3 Dart

Dart programming is a versatile language utilized in software application development. Renowned for its simplicity and clear syntax, Dart is easily grasped by developers of all levels. As an object-oriented language, Dart emphasizes the creation of reusable code components known as objects. Its applications span across web, mobile, and desktop platforms, making it a preferred choice for diverse software projects. Dart serves various purposes, including user interface construction, data handling, and server-side application development. (Badkar, 2023).

Table 4. Pros and Cons of Dart

Pros	Cons
<ul style="list-style-type: none"> Efficient Performance: Dart stands out for its swift execution and efficiency, making it an ideal choice for applications demanding rapid 	<ul style="list-style-type: none"> Limited Adoption: Despite its growing popularity, Dart still trails behind other programming languages like JavaScript in terms of adoption and community size.

<p>response times and seamless performance.</p> <ul style="list-style-type: none"> Ease of Learning: With its straightforward syntax and intuitive structure, Dart facilitates an easy learning curve for developers. Flutter Integration: Dart serves as the primary language for developing applications using the Flutter framework. Cross-Platform Development: Dart empowers developers to create applications that run seamlessly across multiple platforms, including mobile (iOS and Android), web, and desktop. Strong Typing and Static Analysis: Dart boasts a robust static type system and supports static analysis tools, facilitating early error detection and enhancing code reliability and maintainability throughout the development lifecycle (Badkar, 2023). 	<ul style="list-style-type: none"> Learning Curve for Existing Developers: Developers accustomed to working with other languages may encounter a learning curve when transitioning to Dart, necessitating time investment to familiarize themselves with its syntax, features, and best practices. Flutter Dependency: While Dart's tight integration with Flutter offers advantages for mobile app development, developers who prefer alternative frameworks or platforms may find themselves at a disadvantage (Badkar, 2023).
--	--

4.4 Spring Boot

Spring Boot, a highly acclaimed open-source framework, streamlines the development of Java applications for production environments with remarkable ease. Originating from Pivotal Software (now under VMware), Spring Boot simplifies Java application development by offering a comprehensive suite of conventions, templates, and an extensive library ecosystem. In this article, we delve into the benefits, drawbacks, and typical application scenarios associated with Spring Boot (Jayeshwarke, 2023).

Table 5. Pros and Cons of Spring Boot

Pros	Cons
<ul style="list-style-type: none"> • Rapid Development: Spring Boot empowers developers to focus on crafting business logic rather than being bogged down by boilerplate code and configuration. • Microservices Friendly: Spring Boot excels in constructing microservices-based architectures, facilitating the development of small, independent services that are easily deployable, scalable, and maintainable. • Spring Ecosystem Integration: Seamlessly integrating with various Spring projects such as Spring Data, Spring Security, and Spring Cloud, Spring Boot simplifies the creation of sophisticated applications requiring these components. • Auto-Configuration: Spring Boot streamlines application configuration by automatically configuring many aspects based on the libraries and dependencies included in your project. • Embedded Servers: With built-in support for embedded web servers like Tomcat, Jetty, and Undertow, Spring Boot eliminates the need for external server setup, enhancing development efficiency (Jayeshwarke, 2023). 	<ul style="list-style-type: none"> • Learning Curve: While Spring Boot simplifies many aspects of application development, it can still be complex for beginners. • Limited Flexibility: Spring Boot's auto-configuration and conventions may not align with your specific project requirements. • Resource Overhead: Spring Boot applications, particularly those using embedded servers, may consume more memory compared to traditional applications deployed on external servers (Jayeshwarke, 2023).

4.5 Angular

Angular, a robust development platform rooted in TypeScript, offers a component-based framework for crafting scalable web applications. It boasts an extensive array of seamlessly integrated libraries covering diverse functionalities such as routing, form management, and client-server communication. Additionally, Angular provides a comprehensive suite of developer tools facilitating the entire development lifecycle, from inception to deployment (angular, 2024).

Table 6. Pros and Cons of Angular

Pros	Cons
<ul style="list-style-type: none">• Angular empowers Single Page Applications (SPAs), enabling users to seamlessly load new pages without server requests, enhancing dynamism and interactivity.• Angular boasts an extensive feature set: State management? Built-in. Reactivity? Built-in. Routing? Built-in. Dependency Injection (DI)? Built-in. Services? Built-in.• Angular facilitates cross-platform development: Progressive Web Apps (PWAs). Native mobile apps can be developed using strategies from Cordova, Ionic, or NativeScript. Moreover, Angular empowers desktop-installed apps across Mac, Windows, and Linux.• Angular intelligently optimizes templates into highly efficient code, tailored for today's JavaScript virtual machines, amalgamating the benefits of handcrafted code with	<ul style="list-style-type: none">• Angular, as a Single Page Application (SPA), traditionally faces challenges with SEO. However, leveraging Angular Universal allows for pre-rendering, significantly enhancing SEO capabilities.• While Angular does entail a steep learning curve, the investment pays off due to its robust capabilities and versatility.• Angular's complexity and verbosity make it less suited for smaller applications, as it's primarily designed to tackle complex systems and enterprise-level projects.• In developer surveys, Angular often receives mixed reviews. On platforms like The State of JavaScript, it garners significant negative opinions. Conversely, on the Stack Overflow developer survey, Angular ranks as the ninth most loved framework, trailing

<p>the efficiency of a comprehensive framework.</p> <ul style="list-style-type: none"> Angular implements automatic code-splitting, ensuring that users only download the necessary code to render requested views, enhancing performance and reducing load times. Angular adopts TypeScript, enhancing development with superior tooling, cleaner code, and scalability. Angular fosters productivity by furnishing all necessary tools for scaffolding apps and components. The Angular CLI simplifies the generation of production-ready apps and components with a single command. Angular's Model-View-Controller (MVC) architecture facilitates the separation of styles, logic, and markup, promoting maintainability and scalability. Angular features a versatile templating system supporting both one-way and two-way binding, pipes, custom structural directives, and more, providing flexibility and extensibility. Angular offers built-in support for Services and Dependency Injection (DI), enabling clear separation of 	<p>behind React and Vue. It also holds the eighth position as the most dreaded web framework but ranks third among the most desired frameworks (Siddharth, 2021).</p>
--	---

<p>business logic from components, resulting in cleaner and more modular code.</p> <ul style="list-style-type: none"> Angular incorporates a built-in animation API, facilitating the creation of high-performance, intricate animations and choreographies with minimal code (Siddharth, 2021). 	
---	--

4.6 Flutter

Flutter, Google's open-source platform, revolutionizes app development by enabling creation for mobile, desktop, and web platforms from a unified codebase. Unlike conventional solutions, Flutter transcends being merely a framework or library; it represents a comprehensive Software Development Kit (SDK).

An SDK encompasses a vast array of tools, including libraries, documentation, APIs, and occasionally frameworks, providing a complete suite for software development. Flutter epitomizes this concept by furnishing developers with all essential components needed to construct cross-platform applications seamlessly (altexsoft, 2022).

Table 7. Pros and Cons of Flutter

Pros	Cons
<ul style="list-style-type: none"> Single Codebase: Flutter enables the creation of apps for multiple platforms from a single codebase, significantly reducing development time and effort. Rapid Development: Flutter's hot reload feature accelerates the development process by instantly reflecting changes in the app. Performance: Flutter apps boast exceptional speed and performance 	<ul style="list-style-type: none"> Newness: Flutter, compared to other frameworks, is relatively new in the market. Consequently, it may have a smaller talent pool and fewer resources, particularly in comparison to more mature ecosystems. App Size: Due to including the Flutter engine and framework, initial download sizes of Flutter apps tend to be larger than native apps.

<p>as they are compiled into native machine code.</p> <ul style="list-style-type: none"> Rich UI Experience: Leveraging Flutter's vast array of UI widgets, developers can craft visually stunning and engaging user interfaces. Native Feature Access: Through Flutter's platform channels, developers can seamlessly access native features and APIs of underlying operating systems. Cross-Platform Development: Being cross-platform, Flutter allows developers to create apps for iOS, Android, web, and desktop platforms using a unified codebase (Bhanderi, 2023). 	<ul style="list-style-type: none"> Platform Specific Limitations: While Flutter provides access to native capabilities, certain platform-specific features or APIs may not be fully supported or may require additional workarounds. Limited Libraries and Packages: Despite the rapid expansion of the Flutter ecosystem, it may still offer fewer libraries and packages compared to more established frameworks. Dependency on Flutter Updates: Flutter's continuous development may lead to occasional breaking changes between versions (Bhanderi, 2023).
--	--

5. Database (Postgresql)

PostgreSQL stands as an open-source object-relational database management system, distributed under the PostgreSQL License. It excels in supporting both relational (SQL) and non-relational (JSON) data models, offering a wide array of advanced SQL functionalities such as foreign keys, subqueries, and triggers.

Highly extensible, PostgreSQL enables the definition of custom data types and the creation of bespoke functions. Its robust feature set encompasses key extensions like point-in-time recovery, fine-grained access controls, multi-version concurrency control (MVCC), and tablespaces. Additionally, being open source, it enjoys compatibility with major operating systems including Linux, Microsoft Windows, macOS, and Unix (cloud, 2024).

Table 8. Pros and Cons of Postgresql

Pros	Cons
<ul style="list-style-type: none"> PostgreSQL stands out as a reliable option for running dynamic websites and web applications, often integrated as part of the popular LAMP stack. Its robust write-ahead logging feature enhances fault tolerance, ensuring data integrity even in the face of unexpected failures. With its open source nature, PostgreSQL's freely available source code provides users with the flexibility to customize and adapt it to suit their specific business requirements. Support for geographic objects makes PostgreSQL an ideal choice for implementing location-based services and Geographic Information Systems (GIS), offering capabilities for managing geospatial data effectively. Learning PostgreSQL is straightforward, as it boasts an intuitive interface and user-friendly features, minimizing the need for extensive training. PostgreSQL's low maintenance requirements make it suitable for both embedded and enterprise applications, offering hassle-free 	<ul style="list-style-type: none"> PostgreSQL, being community-driven and lacking ownership by a single organization, has faced challenges in raising its profile despite its comprehensive feature set, which rivals other Database Management Systems (DBMS). Efforts to enhance speed often necessitate more extensive modifications in PostgreSQL compared to MySQL, primarily due to PostgreSQL's emphasis on maintaining compatibility. While many open-source applications support MySQL, there may be instances where support for PostgreSQL is lacking or less widespread. In performance comparisons, PostgreSQL tends to exhibit slower speeds than MySQL in certain metrics (Peterson, 2024).

administration for various use cases (Peterson, 2024).	
---	--

6. Other technologies

6.1 Bootstrap

Bootstrap, a freely available open-source front-end development framework, facilitates the creation of websites and web applications. Its primary focus is on enabling responsive development, particularly emphasizing mobile-first approaches.

With a comprehensive collection of syntax and components, Bootstrap streamlines the process of designing templates for websites. By leveraging a pre-defined grid system, developers can easily incorporate responsive design principles into their projects.

Utilizing a combination of HTML, CSS, and JavaScript, Bootstrap provides a robust foundation for building modern web experiences. By abstracting away the complexities of basic commands and functions, Bootstrap empowers developers to accelerate website development without sacrificing quality or responsiveness (Zola, 2022).

6.2 Kafka

Apache Kafka serves as a distributed data store tailored for efficiently ingesting and processing streaming data in real-time. Streaming data, originating from myriad sources, entails continuous generation and simultaneous transmission of data records.

Kafka caters to its users with three core functionalities:

- Facilitating the publication and subscription to streams of records.
- Efficiently storing streams of records in the sequential order of generation.
- Enabling real-time processing of streams of records.

Primarily utilized for constructing real-time streaming data pipelines, Kafka empowers the development of applications capable of seamlessly adapting to dynamic data streams. By seamlessly integrating messaging, storage, and stream processing capabilities, Kafka facilitates the storage and analysis of both historical and real-time data (amazon, 2024).

6.3 Keycloak

Keycloak, an open-source identity and access management tool developed by Red Hat, serves as a comprehensive solution for securing web applications and RESTful web services with single sign-on capabilities. Red Hat's blog defines Keycloak as a robust single sign-on solution designed to simplify the task of securing applications and services for developers.

With Keycloak, developers can effortlessly secure their applications and services while enjoying features such as user authentication and secure invocation of REST APIs and microservices with minimal coding required. Moreover, Keycloak offers the flexibility to integrate users from various sources and social networks, including Google, GitHub, Facebook, and Twitter, streamlining user management across platforms (Bayoglu, 2023).

6.4 Digital Ocean

DigitalOcean, an Infrastructure as a Service (IaaS) provider, has recently expanded its offerings to include an Application Platform as a Service (PaaS) option. Unlike traditional PaaS providers like Google App Engine or Heroku, DigitalOcean gives users direct access to the underlying virtual machines supporting their applications. This means users have full control over disk space, memory allocation, networking configurations, operating systems, and more, allowing them to tailor their environment to suit their application's specific requirements (papertrail, 2024).

While DigitalOcean offers a range of services, its primary focus is on application and website hosting. To deploy applications on DigitalOcean's IaaS environment, developers launch what the company refers to as "droplets" - private virtual machine instances. These droplets are similar to instances on Amazon Elastic Compute Cloud (EC2) or Azure, with the distinction that DigitalOcean does not support Microsoft Windows instances (Gillis, 2022).

6.5 Cloudinary

Cloudinary, a leading Software-as-a-Service (SaaS) solution, specializes in efficiently managing media assets stored in the cloud. Whether it's images or videos for web or mobile applications, Cloudinary provides a comprehensive suite of services. These include upload management, administration, optimized delivery, storage, and manipulation—all seamlessly executed on Cloudinary's scalable servers, dynamically accommodating sudden spikes in traffic and high loads.

At the heart of Cloudinary's offering is its robust Digital Media Management (DAM) platform. Its intuitive user interface caters to both creative designers and developers, facilitating smooth collaboration. Additionally, Cloudinary boasts a powerful API, enabling easy integration of user-generated and in-house media content into mobile apps or websites with minimal code requirements (javatpoint, 2021).

6.6 Traefik

Traefik, also known as Træfik, is a dynamic and adaptable reverse proxy and load balancer tailored for modern, distributed, and microservices architectures. Serving as a critical component of server infrastructure, Traefik effectively oversees and optimizes network traffic, ensuring streamlined routing and bolstering the performance and security of web services.

Designed for contemporary distributed setups, Traefik shines in environments where services experience frequent scaling fluctuations. With its dynamic configuration capabilities, support for multiple backends, automatic SSL/TLS, and rich customization features, Traefik streamlines web server management, boosts performance, fortifies security, and furnishes actionable insights for application monitoring and issue resolution.

In a competitive landscape alongside NGINX and HAProxy, Traefik's distinct advantages in dynamic and cloud-native settings set it apart. Installation and setup on platforms like CentOS and Ubuntu yield a potent solution for efficiently handling network traffic (Nek, 2024).

7. Conclusion

In my exploration of various programming languages and technologies, I've delved into their unique strengths and weaknesses. As I proceed, my aim is to carefully analyze the guidelines and documentation provided for each technology, ensuring I make informed decisions when applying them to my project. Recognizing that each technology offers distinct advantages and drawbacks, my approach involves skillfully combining them to achieve optimal results. By leveraging the strengths of each technology while mitigating their respective limitations, I endeavor to craft a solution that delivers the utmost efficiency and effectiveness for my project.

CHAPTER 3: DESIGN

1. Database schema

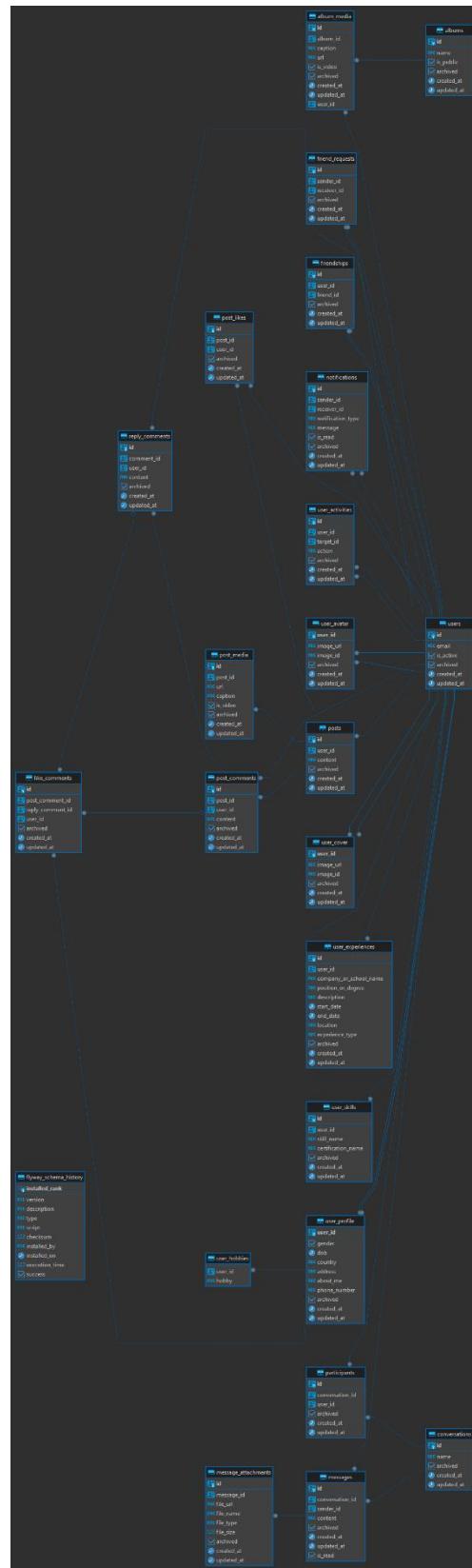


Figure 3. Database schema

Below is a description of the database. Since all tables have "archive", "create_at", "updated_at", I will not add a description to each table. Archive is used to hide data when users delete it. Created_at and Updated_at are used to store the date and time when a row was created or updated.

Table 9. User data desctiption

Because I use keycloak to perform authentication functions as well as permissions, full name, password, and role will all be saved on keycloak.

Stores essential information for user accounts.

Field name	Data type	Description
id	uuid	Unique identifier for each user, stored as a UUID for universal uniqueness.
email	varchar(255)	Email address of the user, stored as VARCHAR(255) for compatibility and identification.
is_active	bool	Indicates if the user account is active.

Table 10. User profile data desctiption

Contains extended profile information for each user.

Field name	Data type	Description
user_id	uuid	Unique identifier for each user, referencing the id column in the users table.
gender	bool	Indicates the gender of the user.
dob	date	Date of birth of the user.
country	varchar(255)	Country where the user resides.
address	varchar(255)	Physical address of the user.

about_me	text	Description or bio of the user
phone_number	varchar(255)	Contact phone number of the user

Table 11. User hobbies data description

Store hobbies of user

Field name	Data type	Description
user_id	uuid	Unique identifier for each user, referencing the id column in the users table.
hobby	varchar(255)	hobby or interest of the user

Table 12. User experiences data description

Logs the work and educational experiences of users.

Field name	Data type	Description
id	uuid	Unique identifier for each user experience entry
user_id	uuid	The ID of the user associated with this experience, referencing the id column in the users table
company_or_school	varchar(255)	Name of the company or school for this experience
position_or_degree	varchar(255)	Position held or degree obtained during this experience
description	text	Description or details about the experience
start_date	date	Start date of the experience

end_date	date	End date of the experience
location	varchar(255)	Location where the experience took place
experience_type	varchar(50)	Type of experience, such as work or education

Table 13. User skills data description

Records the skills and certifications that users have added to their profile.

Field name	Data type	Description
id	uuid	Unique identifier for each user skill entry
user_id	uuid	The ID of the user associated with this experience, referencing the id column in the users table
skill_name	varchar(255)	Name of the skill possessed by the user
certificate_name	varchar(255)	Name of any certification related to the skill

Table 14. User avatar data description

Store the avatar image of user

Field name	Data type	Description
user_id	uuid	Unique identifier for each user, referencing the id column in the users table.
image_url	text	URL to the user's avatar or profile picture
image_id	varchar(255)	Id to the user's cover photo

Table 15. User cover description

Store the cover image of user

Field name	Data type	Description
user_id	uuid	Unique identifier for each user, referencing the id column in the users table.
image_url	text	URL to the user's cover photo
image_id	varchar(255)	Id to the user's cover photo

Table 16. Friend request data description

Tracks friend requests sent between users.

Field name	Data type	Description
id	uuid	Unique identifier for each friend request
sender_id	uuid	The ID of the user who sent the friend request, referencing the id column in the users table
receiver_id	uuid	The ID of the user who receives the friend request, also referencing the id column in the users table

Table 17. Friendship data description

Represents friendships between users.

Field name	Data type	Description
id	uuid	Unique identifier for each friendship
user_id	uuid	The ID of the user who initiated the friendship,

		referencing the id column in the users table
friend_id	uuid	The ID of the friend who is being added, also referencing the id column in the users table

Table 18. Conversations data description

Manages conversation threads.

Field name	Data type	Description
id	uuid	Unique identifier for each conversation
name	varchar(255)	Name or title of the conversation

Table 19. Participants data description

Links users to the conversations they are part of.

Field name	Data type	Description
id	uuid	Unique identifier for each participant entry
conversation_id	uuid	The ID of the conversation in which the user is a participant, referencing the id column in the conversations table
user_id	uuid	The ID of the user who is a participant in the conversation, referencing the id column in the users table

Table 20. Message data description

Stores messages exchanged in conversations.

Field name	Data type	Description
id	uuid	Unique identifier for each message
conversation_id	uuid	The ID of the conversation to which the message belongs, referencing the id column in the conversations table
sender_id	uuid	The ID of the user who sent the message, referencing the id column in the users table.
content	text	Content or text of the message
is_read	bool	Indicates if the message is read.

Table 21. Message attachment data description

Manages files attached to messages.

Field name	Data type	Description
id	uuid	Unique identifier for each message attachment
message_id	uuid	The ID of the message to which the attachment belongs, referencing the id column in the messages table
file_url	text	URL of the attached file
file_name	varchar(255)	Name of the attached file

file_type	varchar(255)	Type of the attached file, such as image, video, audio, or document
file_size	bigint	Size of the attached file in bytes

Table 22. Album data description

Manages collections of media grouped by users into albums.

Field name	Data type	Description
id	uuid	Unique identifier for each album
name	varchar(255)	Name of the album
is_public	bool	Indicates whether the album is public or not

Table 23. Album media data description

Holds media items that belong to an album.

Field name	Data type	Description
id	uuid	Unique identifier for each album media entry
album_id	uuid	The ID of the album to which the media belongs
url	text	URL of the media file, which could be an image or video
caption	text	Caption or description for the media
is_video	bool	Indicates whether the media is a video or not

Table 24. Posts data description

Contains posts made by users.

Field name	Data type	Description
id	uuid	Unique identifier for each post
user_id	uuid	The ID of the user who created the post, referencing the id column in the users table.
content	text	Content or text of the post

Table 25. Post media data description

Manages media associated with posts.

Field name	Data type	Description
id	uuid	Unique identifier for each post media entry
post_id	uuid	The ID of the post to which the media belongs, referencing the id column in the posts table.
url	text	URL of the media file associated with the post
caption	text	Caption or description for the media
is_video	bool	Indicates whether the media is a video or not

Table 26. Post like data description

Records likes on posts by users.

Field name	Data type	Description
id	uuid	Unique identifier for each post like entry
post_id	uuid	The ID of the post that was liked, referencing the id column in the posts table.
user_id	uuid	The ID of the user who liked the post, referencing the id column in the users table

Table 27. Post comments data description

Stores comments made on posts.

Field name	Data type	Description
id	uuid	Unique identifier for each post comment
post_id	uuid	The ID of the post to which the comment belongs
user_id	uuid	The ID of the user who posted the comment, referencing the id column in the users table.
content	text	Content or text of the comment

Table 28. Reply comments data description

Holds replies to comments on posts.

Field name	Data type	Description
id	uuid	Unique identifier for each reply comment
comment_id	uuid	The ID of the parent comment to which the reply belongs, referencing the id column in the post_comments table.
user_id	uuid	The ID of the user who posted the reply comment, referencing the id column in the users table.
content	text	Content or text of the reply comment

Table 29. Like comments data description

Keeps track of likes on both post comments and reply comments.

Field name	Data type	Description
id	uuid	Unique identifier for each like on a comment
post_comment_id	uuid	The ID of the post comment being liked, referencing the id column in the post_comments table.
reply_comment_id	uuid	The ID of the reply comment being liked, referencing the id column in the reply_comments table.

user_id	uuid	The ID of the user who liked the comment, referencing the id column in the users table.
---------	------	---

Table 30. Notification data description

Manages notifications sent between users.

Field name	Data type	Description
id	uuid	Unique identifier for each notification
sender_id	uuid	The ID of the user who sent the notification
receiver_id	uuid	The ID of the user who receives the notification
notification_type	varchar(50)	Describes the type of notification. Possible types include friend request, message, post, comment, like, reply, etc.
message	text	Content or message of the notification
is_read	bool	Indicates whether the notification has been read by the receiver

Table 31. User activities data description

Tracks activities performed by users, such as follows and likes.

Field name	Data type	Description
id	uuid	Unique identifier for each user activity entry
user_id	uuid	The ID of the user performing the activity,

		referencing the id column in the users table
target_id	uuid	The ID of the target user or entity related to the activity, also referencing the id column in the users table
action	varchar(255)	Describes the action performed by the user. Possible actions include follow, like, comment, etc.

2. Sitemap

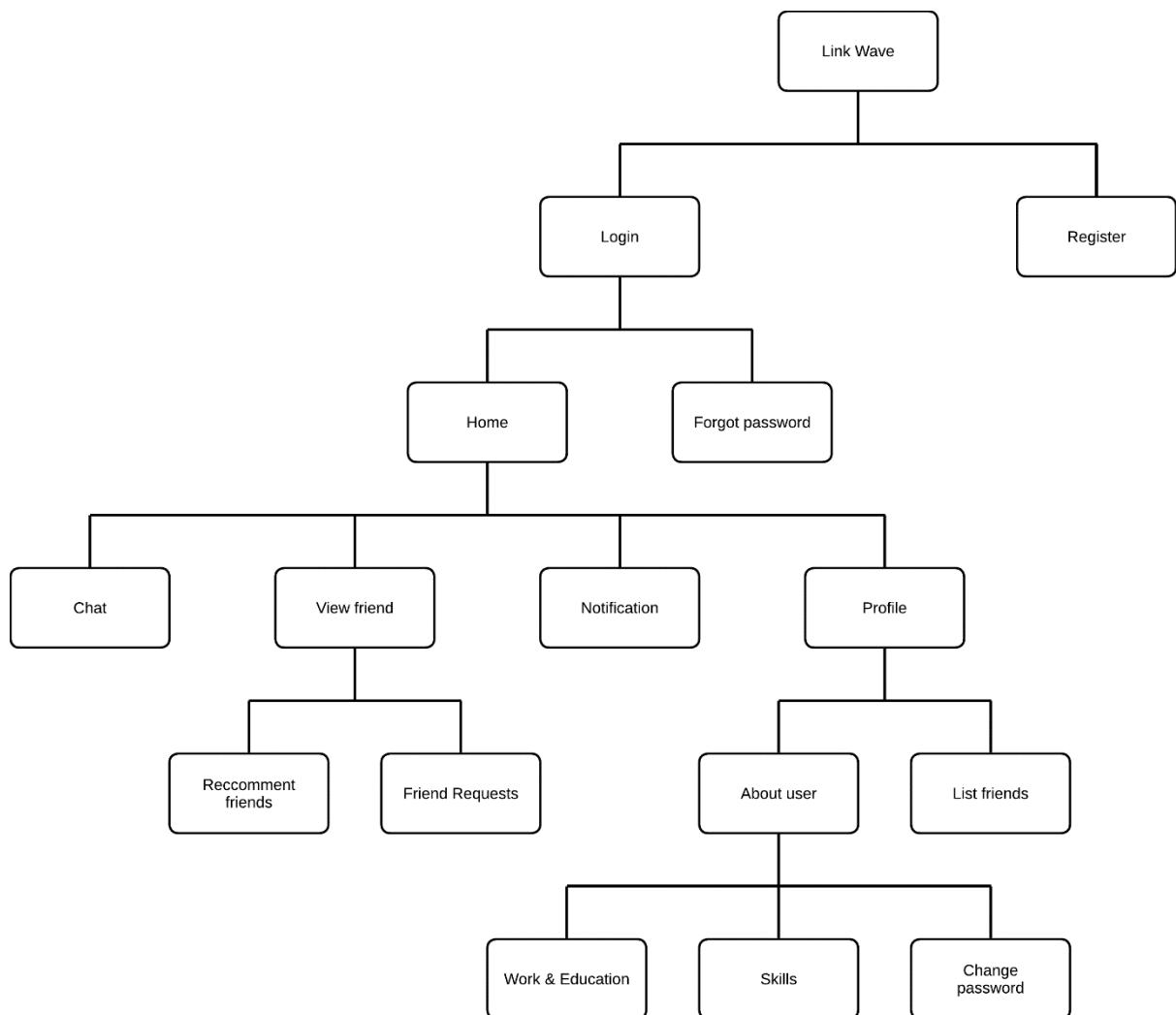


Figure 4. Sitemap website

3. Usecase diagram

3.1 User role

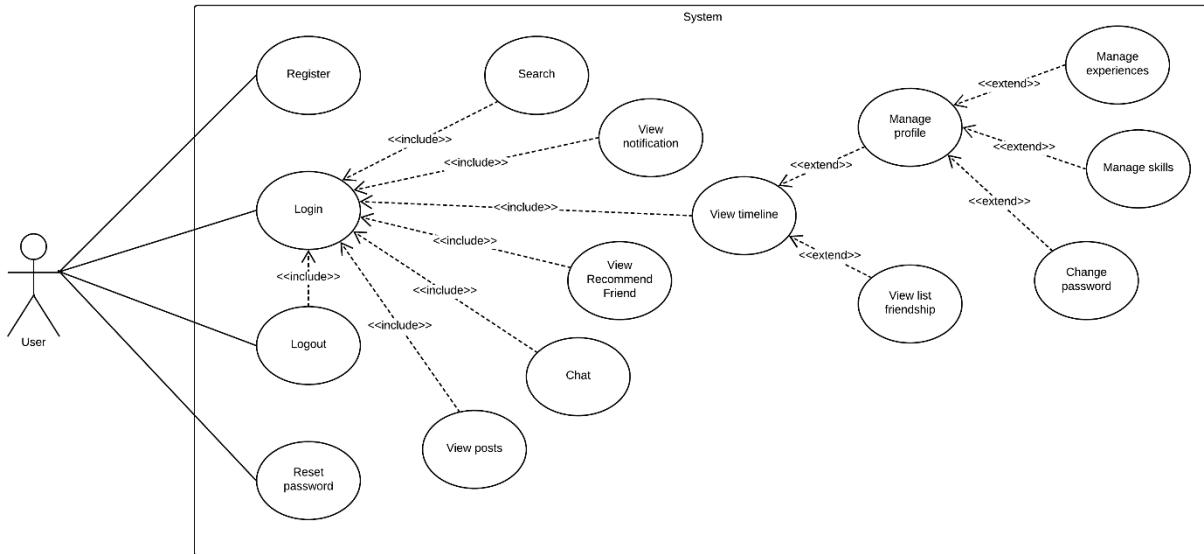


Figure 5. Use case of user role

Here's a description of the use case diagram in the image:

- Actor: There's a single actor identified as "User," which represents the person interacting with the system.
- System: The larger rectangle, named "System," encompasses various functionalities that the user can interact with.
- Use Cases:
 - Primary Functions:

Register: The user can create a new account.

Login: The user can log into the system. This function is central to the diagram, indicating that many other functions require the user to be logged in.

Logout: The user can log out of the system.

Reset Password: The user can reset their password if necessary.

- Post-Login Functions (these use cases are connected to the "Login" function, implying they are accessible after a user logs in):

View Posts: The user can view posts within the system.

Chat: The user can engage in chats.

View Recommended Friend: The system can suggest friends to the user.

View Notifications: The system displays notifications to the user.

View Timeline: The user can view their timeline of activities or posts.

View List Friendship: The user can view a list of their friends or connections.

Manage Profile: This seems to be a more complex use case that can extend into more specific actions, such as managing experiences and skills, which could refer to personal information or professional qualifications.

Change Password: The user can change their current password.

- Relationships:
 - **Include Relationship:** This is shown with a dashed arrow with the "include" stereotype, indicating that the target use case is included in the functionality of the source use case. For example, the "Login" use case includes "View Posts," "Chat," etc.
 - **Extend Relationship:** This is depicted with a dashed arrow with the "extend" stereotype, implying that the extended use case adds to the base use case under certain conditions. For example, "Manage Profile" extends into "Manage Experiences" and "Manage Skills."

3.2 Admin role

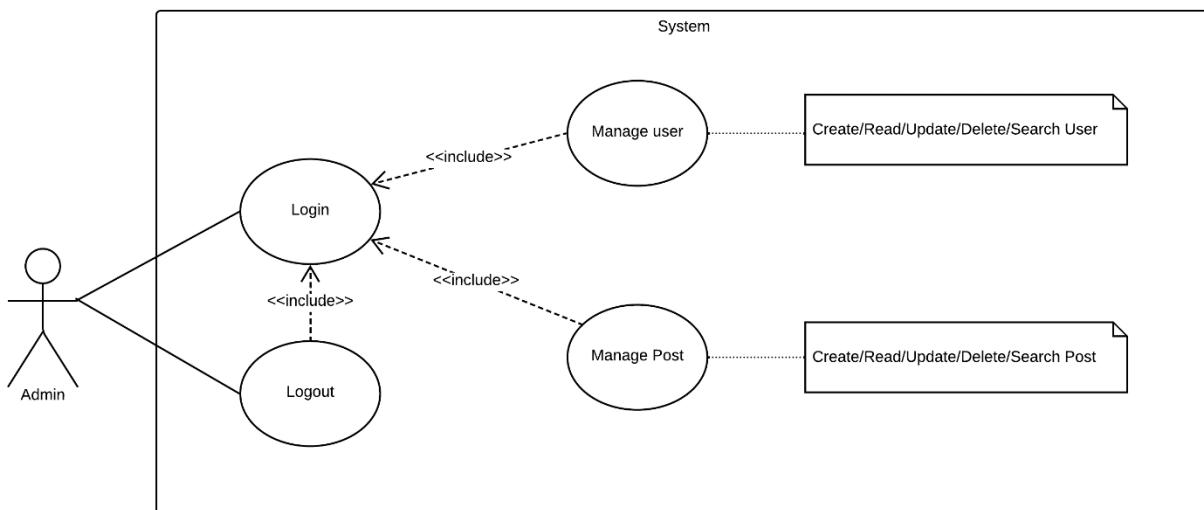


Figure 6. Usecase of admin role

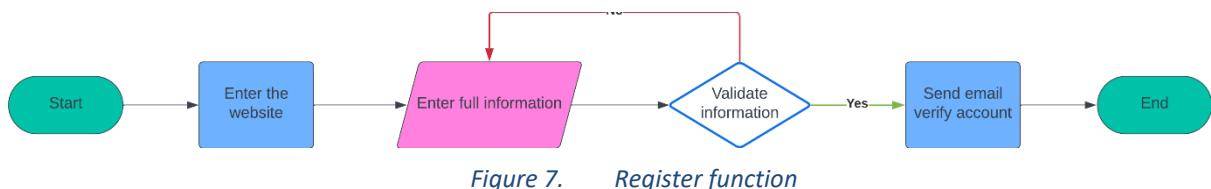
Here's a breakdown of the use case diagram:

- Actor: As with the previous diagram, there is one actor identified as "Admin".
- System: This is indicated by the boundary that encapsulates the use cases.
- Use Cases:
 - Login: The admin can log into the system, which is a prerequisite for accessing other functionalities.
 - Logout: The admin can log out of the system.
- Secondary Use Cases (possibly only accessible after logging in):

- Manage User: This use case is connected to the "Login" use case with an "include" relationship, suggesting that managing users is a function that is part of the login process. This use case further expands into CRUD (Create, Read, Update, Delete) and Search functionalities specific to users.
- Manage Post: Similar to "Manage User", this use case is connected to the "Login" and involves CRUD and Search functionalities, but it pertains to posts rather than user accounts.

4. Flow Chart

4.1 Register



Here's a step-by-step breakdown:

- Start: This is the initial step where the user begins the registration process.
- Enter the website: The user accesses the website where they wish to register.
- Enter full information: The user is prompted to fill in all required information, such as name, email, password, etc.
- Validate information: This is a decision point where the system checks if the information provided by the user is valid. If not, the process loops back to the "Enter full information" step (indicated by the red arrow) for the user to correct or complete their information.
- Send email to verify account: If the information is validated (as indicated by the "Yes" path from the decision diamond), the system proceeds to send a verification email to the user's provided email address.
- End: This is the final step of the process after the email has been sent, concluding the registration function.

4.2 Login

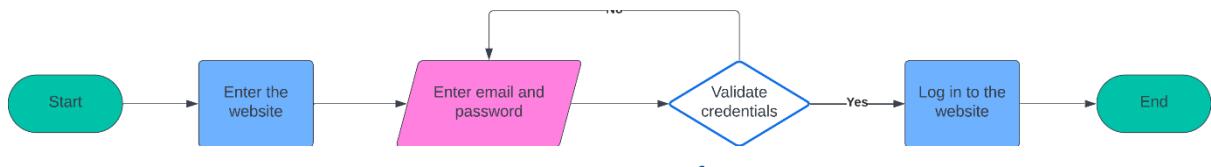


Figure 8. Login function

Here's what each step represents:

- Start: The initiation point where a user begins the login process.
- Enter the website: The user navigates to the website's login page.
- Enter email and password: The user is prompted to input their email address and password into the designated fields.
- Validate credentials: This step is a decision node where the system checks the entered credentials against its records to see if they match an existing account. If the credentials are incorrect, the process loops back to the "Enter email and password" step (indicated by the white arrow) for the user to try again.
- Log in to the website: If the credentials are validated (following the "Yes" path), the system allows the user to access their account and proceeds to the website's user-specific area.
- End: This marks the conclusion of the login process after successful authentication.

4.3 Reset password

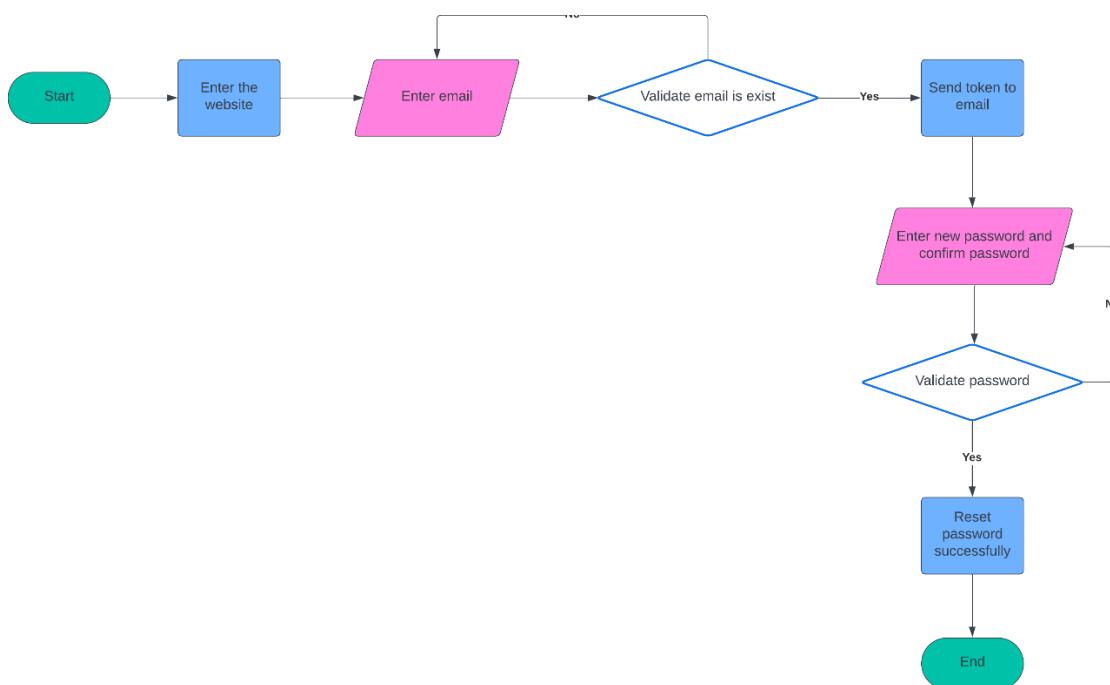


Figure 9. Reset password function

Here's the detailed description:

- Start: The initial point where a user decides to reset their password.
- Enter the website: The user navigates to the given website and finds the option to reset the password.
- Enter email: The user is prompted to enter their email address into the reset password feature.
- Validate email is exist: This is a decision step where the system checks if the provided email address exists in the database. If the email does not exist, the flow ends, indicating that the process cannot continue without a valid email account on record.
- Send token to email: If the email exists, the system sends a token or a link to the user's email address to proceed with password resetting.
- Enter new password and confirm password: The user receives the token or link, uses it to access the reset password area, and enters a new password along with a confirmation of that password.
- Validate password: This decision step verifies if the new password meets the system's criteria and if the confirmation matches the new password. If the validation fails, the flow loops back to the "Enter new password and confirm password" step.
- Reset password successfully: Upon successful validation, the password is reset in the system.
- End: The process concludes, and the user can now log in with the new password.

4.4 Search friend



Figure 10. Search friend function

Here's a step-by-step explanation of the process:

- Start: The user initiates the friend search process.
- Login to the website: The user logs into their account on the website.
- Enter friend name: The user is prompted to enter the name of the friend they are looking for into a search field.

- Show the result: After entering the name and submitting the search request, the website processes the search and displays the results to the user.
- End: The process concludes after the search results have been presented to the user.

4.5 Update profile

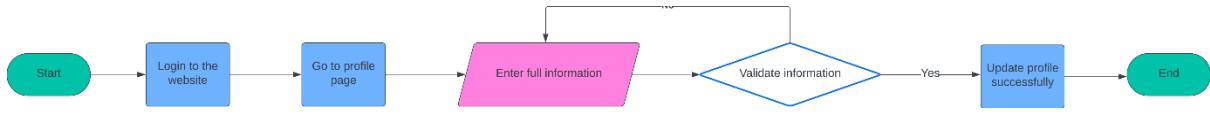


Figure 11. Update profile function

Here's the process outlined in the chart:

- Start: This is where the user initiates the update profile process.
- Login to the website: The user must log in to their account on the website to make changes to their profile.
- Go to profile page: After logging in, the user navigates to their profile page where they can make updates.
- Enter full information: The user enters or updates their personal information, such as name, contact details, bio, etc.
- Validate information: This step represents a decision point where the system checks the new information entered by the user for completeness and accuracy. If the information is not valid (the "No" path), the process redirects back to the "Enter full information" step for the user to make corrections.
- Update profile successfully: If the information is validated ("Yes" path), the profile is updated with the new details.
- End: The process concludes, and the user's profile has been successfully updated.

4.6 Change password



Figure 12. Change password function

Here's a description of each step:

- Start: The point at which the user decides to change their password.
- Login to the website: The user must first log into the website with their current credentials to access their account settings.

- Go to profile page: After successfully logging in, the user navigates to their profile page or directly to the password change option within their account settings.
- Enter old password, new password, confirm password: On the password change screen, the user must enter their current (old) password and then enter the new password twice, once to set it and again to confirm it, ensuring they've typed it correctly.
- Validate password: The system checks the entered passwords to confirm that the old password is correct, the new password meets any specified security requirements, and that the new password has been entered identically in both fields. If there's an issue (following the "No" path), the process loops back to allow the user to re-enter their passwords.
- Change password successfully: If all entered passwords are validated (following the "Yes" path), the system updates the user's password with the new one.
- End: The process concludes, indicating the user has successfully changed their password.

4.7 Create new post

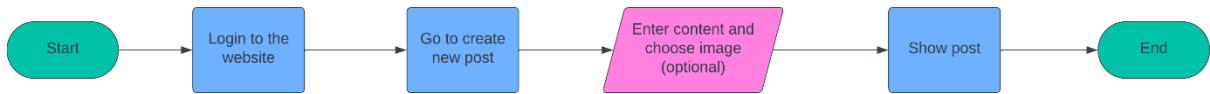


Figure 13. Create new post function

Here's the flow explained:

- Start: The user initiates the process of creating a new post.
- Login to the website: The user must first log into their account on the website to access the posting feature.
- Go to create new post: Once logged in, the user navigates to the section where they can create a new post. This is typically found in the user's dashboard or in a clearly labeled section of the site.
- Enter content and choose image (optional): In the post creation area, the user enters the text for their new post. They also have the option to add an image to accompany the text. The "optional" label indicates that adding an image is not required to proceed.
- Show post: After the content (and possibly an image) is added, the post is published or previewed, showing how it will appear on the website.

- End: The process concludes, and the new post is now created and possibly visible to other users, depending on the website's functionality.

5. Mockup UI design

5.1 Register

LinkWave

Sign up

Email Email
Email
Password
Confirm Password

I agree all statement in Term of service

[Sign up](#)

Text

Figure 14. Register page

5.2 Login

LinkWave

Login

Email
Password

Text Text

[Login](#)

Text

Figure 15. Login page

5.3 Home

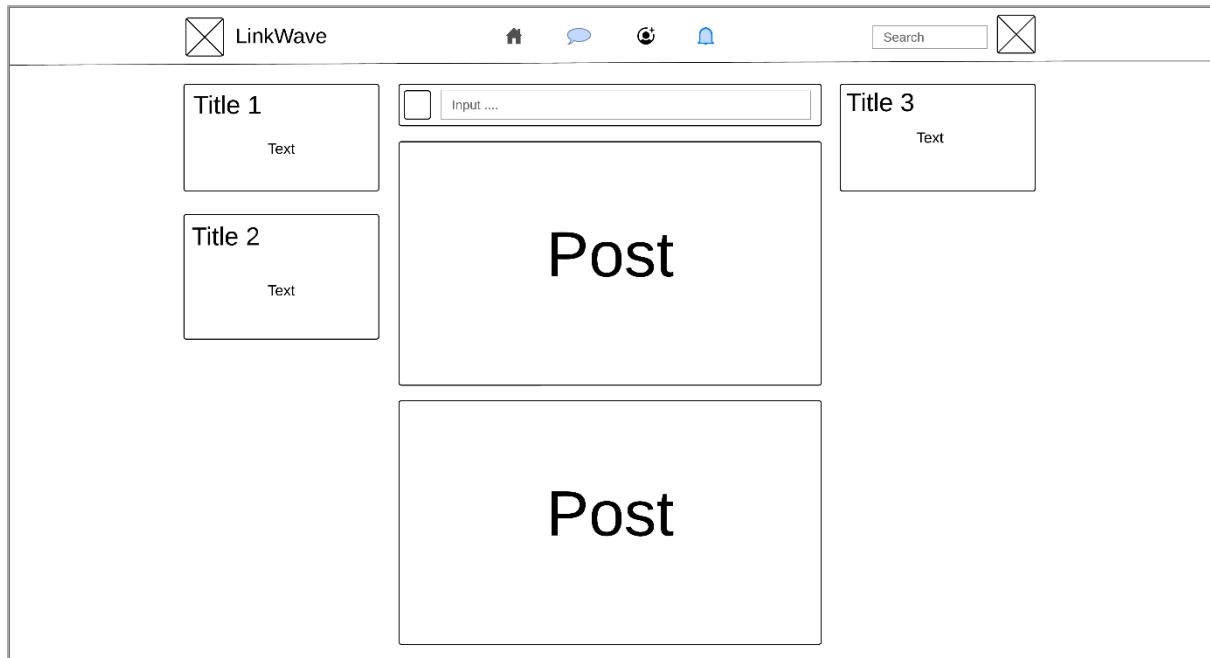


Figure 16. Home page

5.4 Chat message

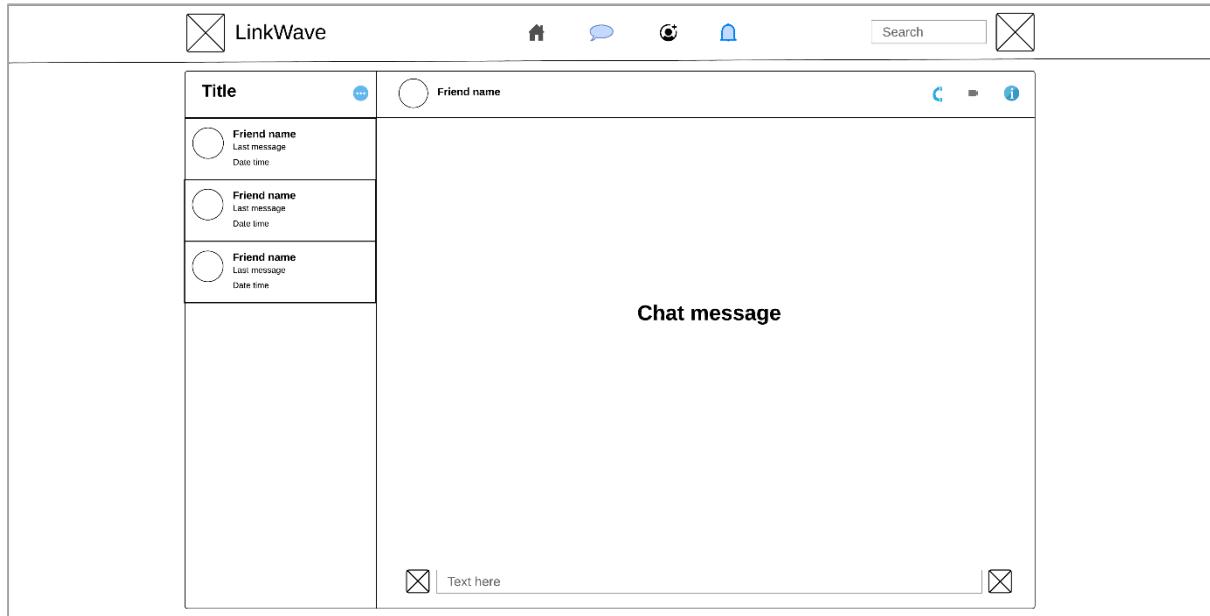


Figure 17. Chat message page

5.5 Friend

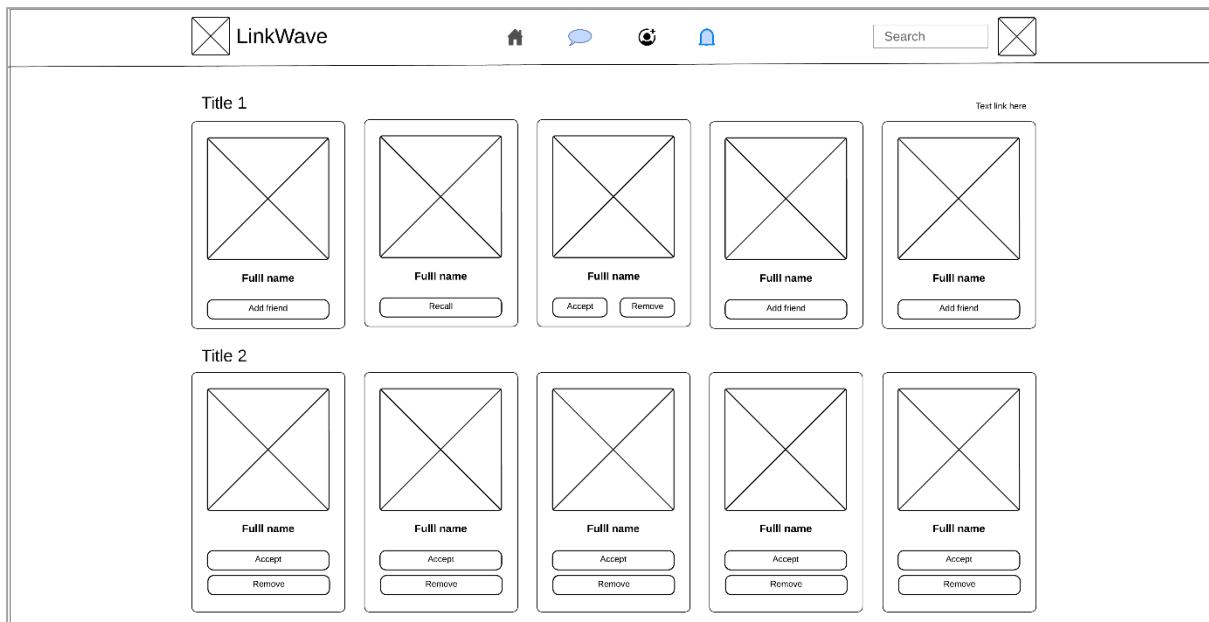


Figure 18. Add Friend page

5.6 Notification

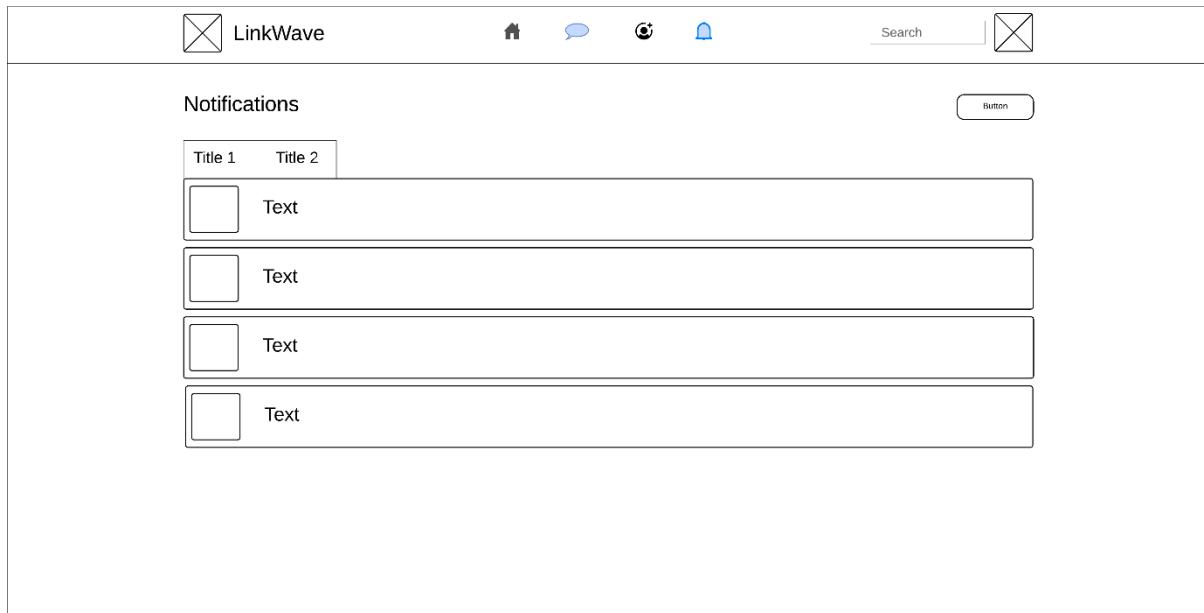


Figure 19. Notification page

5.7 Profile

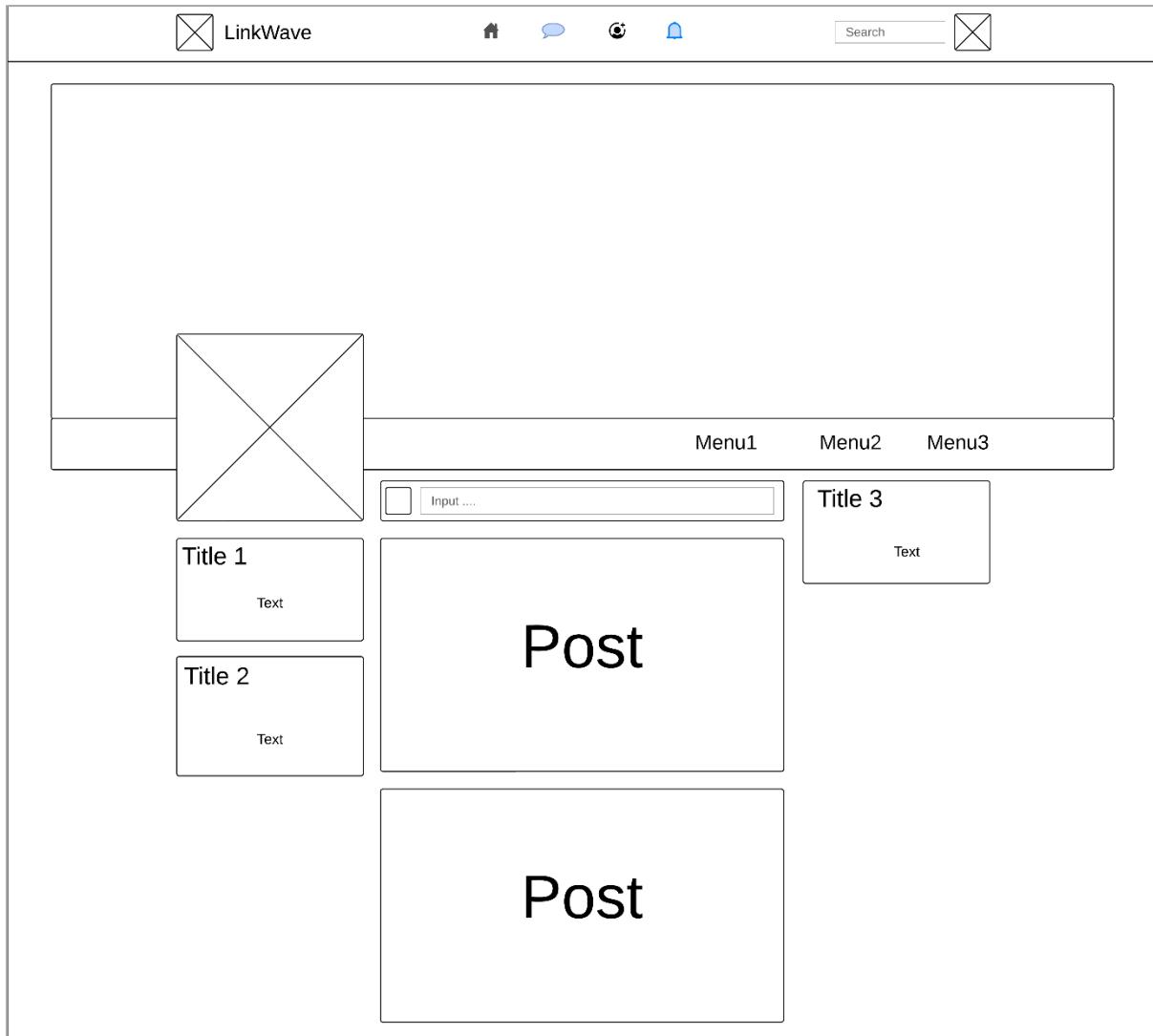


Figure 20. Profile page

5.8 About user

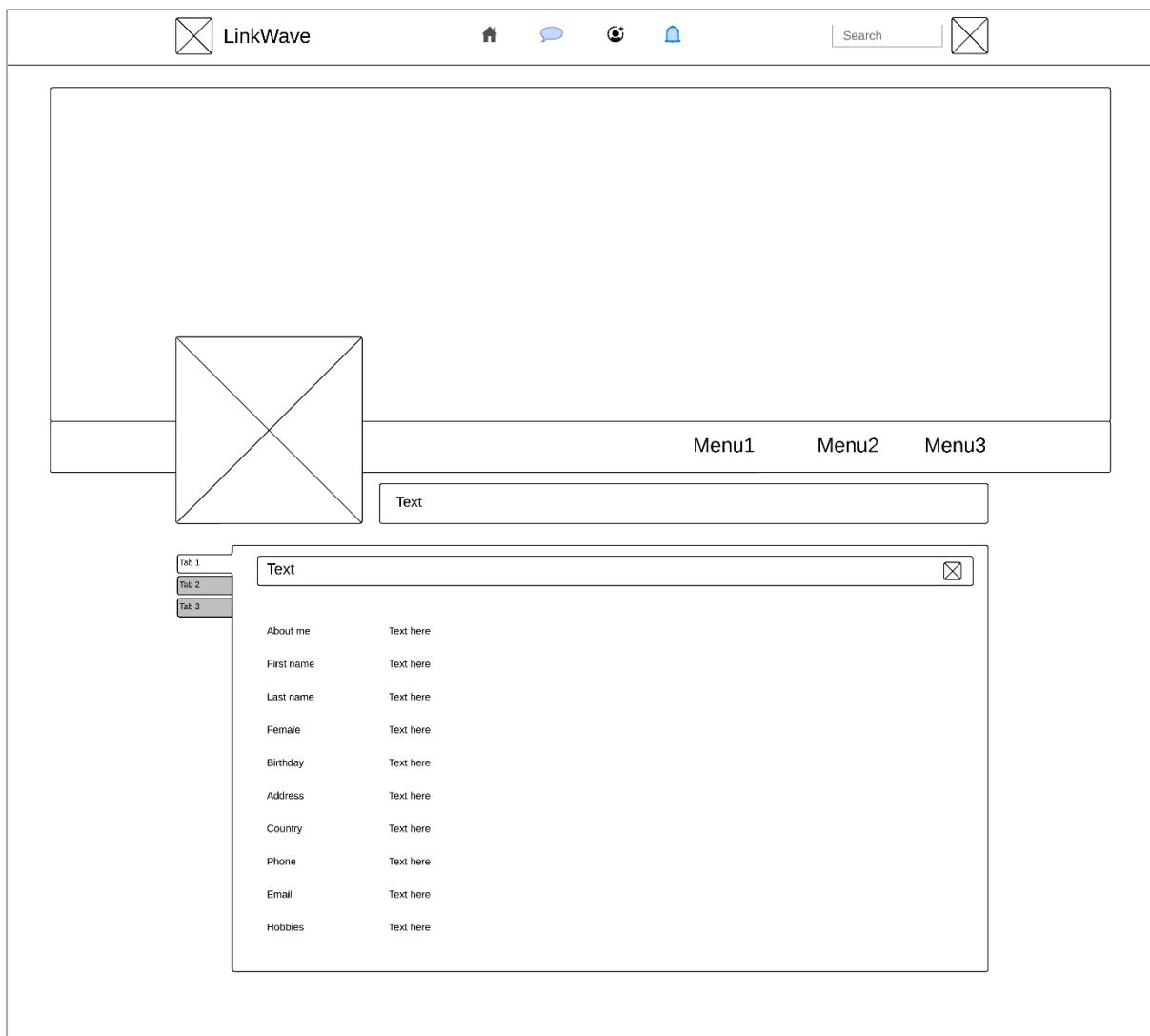


Figure 21. About user page

5.9 Friend list

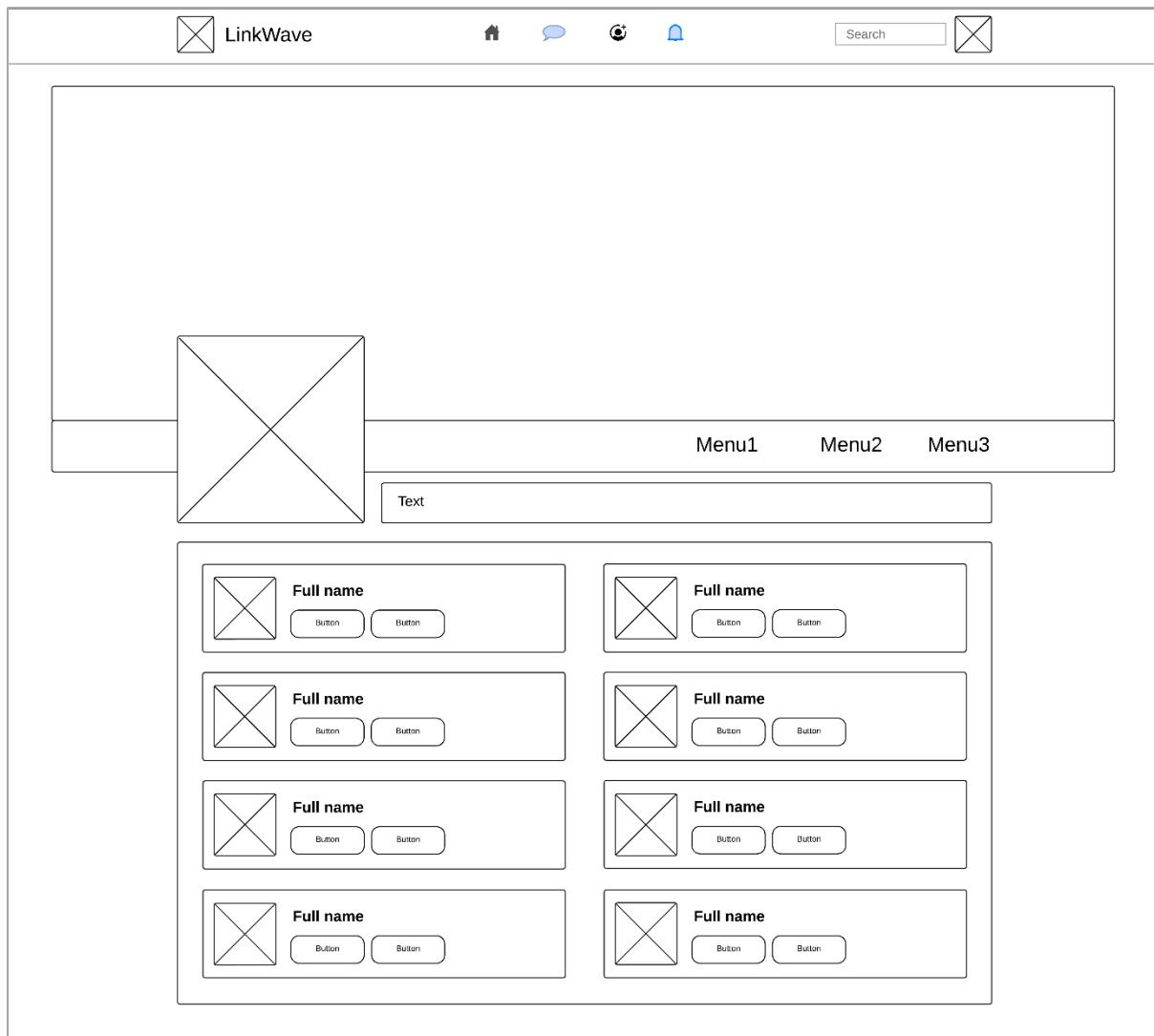


Figure 22. Friend list page

CHAPTER 4: METHODOLOGIES AND SOFTWARE DEVELOPMENT PROCESS

1. Software development process

In this project, I've implemented the Scrum framework, designed to facilitate seamless adaptation to evolving conditions and user needs. With its emphasis on flexibility, Scrum enables frequent re-prioritization and short release cycles, fostering continuous learning and improvement within the team. (DRUMOND, 2024)

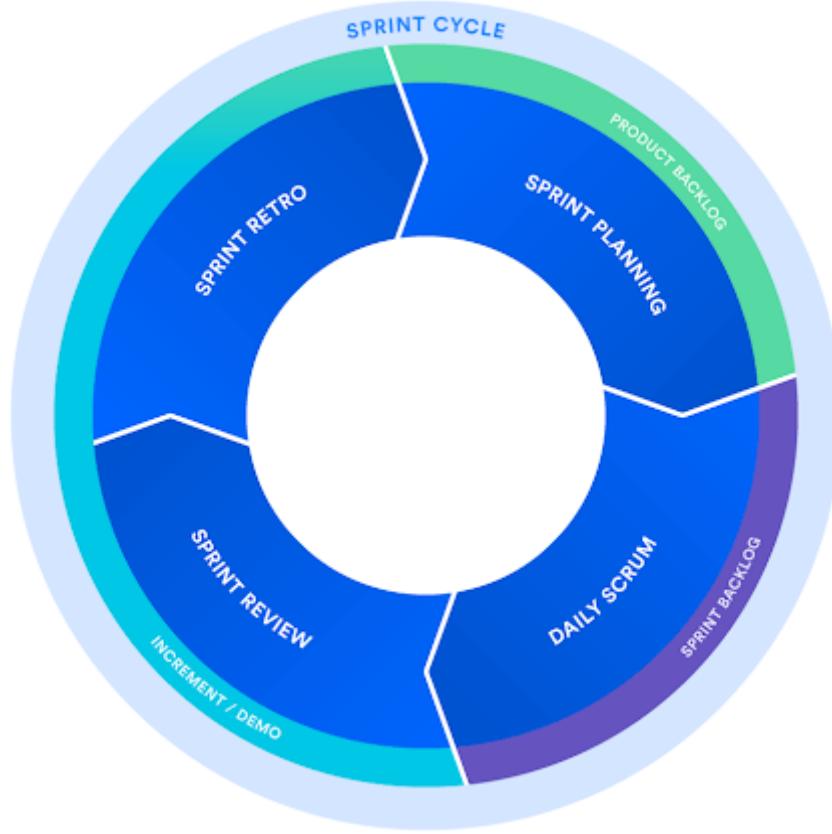


Figure 23. Software development process model

2. Software development methodologies

In today's highly competitive IT industry, the focus has shifted towards developing high-quality products. Speedy delivery is no longer the sole priority; customer satisfaction is equally important. This shift has led to the adoption of rigorous testing techniques and tools, creating complex challenges in project management and team coordination at all levels.

Companies are now prioritizing the delivery of quality products that meet customer expectations. Each software development life cycle model has its significance and limitations. Choosing a specific model allows for tailored processes and structures to suit various types of products and projects. For my project, I have opted for the Agile methodology as it aligns well with these goals and requirements (Singh, et al., 2019).

As per the definition from "AgileAlliance.org," "Agile" serves as a broad term encompassing methodologies and practices aimed at improving software development through collaborative efforts. It is rooted in the Agile Software Development Manifesto and its 12 principles.

These principles are inspired by the insights found in the Agile Manifesto by Cunningham and include the following key ideas:

- The highest priority is to satisfy the customer through continuous and early software development.
- It's crucial to maximize the amount of work not done, emphasizing efficiency.
- Development teams should regularly reflect on their processes during fixed intervals to enhance productivity and make necessary adjustments.
- Creating a conducive environment for motivated individuals and entrusting them with the responsibility of delivering results is vital.
- Agile processes support accommodating changes in requirements even during the later stages of development to ensure the software's benefits for the customer (Singh, et al., 2019).

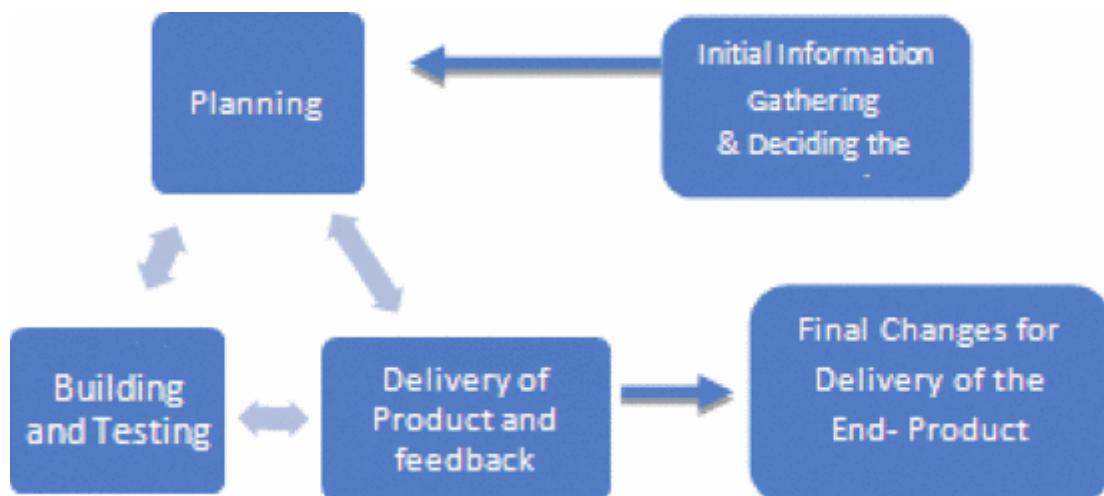


Figure 24. Depicting the implementation in development life cycle

3. Conclusion

I've decided to utilize the Scrum methodology for this project, finding it to be the most suitable approach among the various available models. While it may not be universally acclaimed as the best, its adaptability to changing requirements and iterative nature align perfectly with the project's needs. Embracing Scrum will require a shift in my workflow, as it emphasizes regular sprints, constant collaboration, and frequent review cycles. However, I am committed to implementing Scrum to ensure

the development of the highest quality product possible, leveraging its framework to enhance communication, transparency, and ultimately, project success.

CHAPTER 5: IMPLEMENTATION AND EXPLANATION

1. Backend

1.1 Folder structure

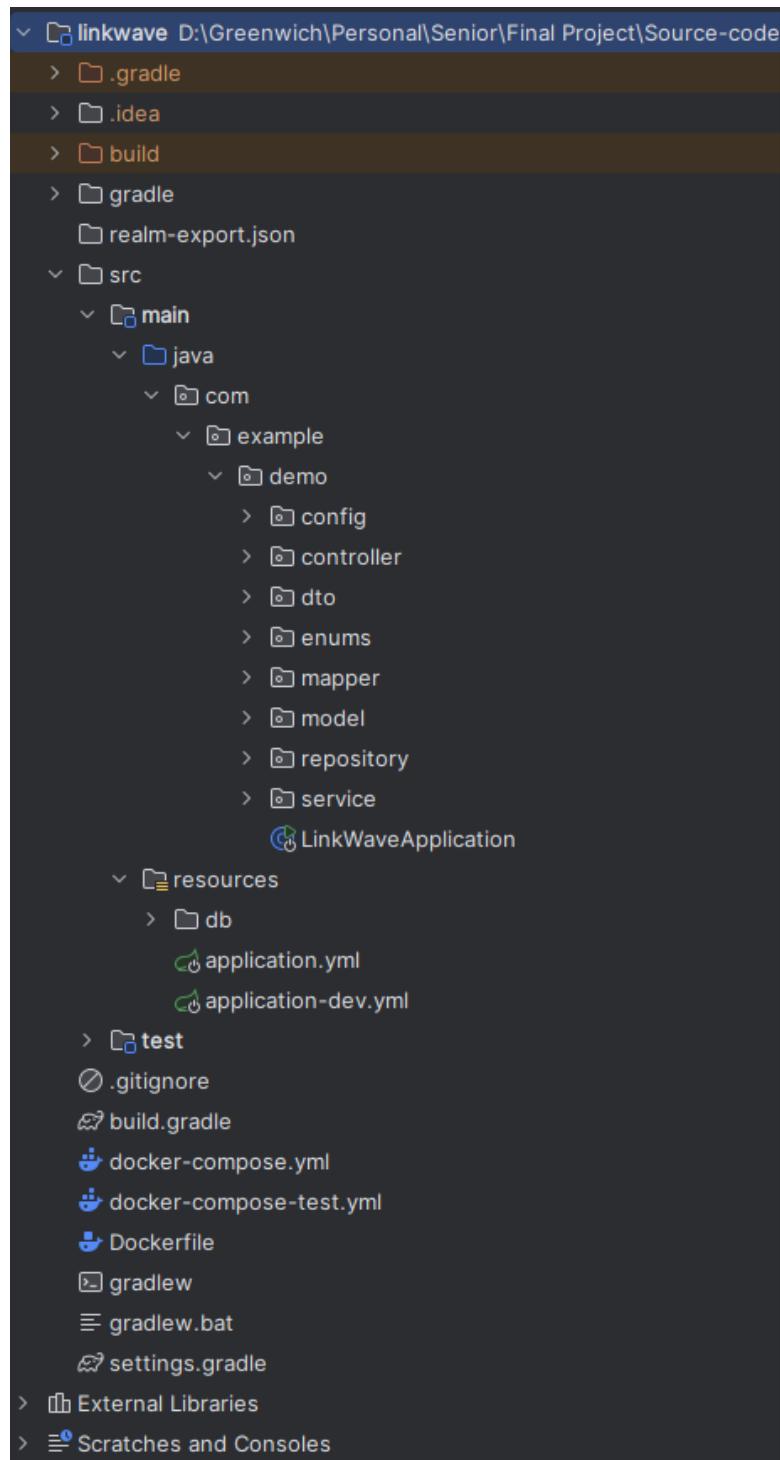


Figure 25. Backend's folder structure

1.2 Setup environment

1.2.1 Docker compose

```
version: '3'

services:
  linkwave-db:
    image: postgres:15
    container_name: linkwave-db
    restart: unless-stopped
    environment:
      POSTGRES_USER: linkwave_user
      POSTGRES_PASSWORD: ndquang@2002
      POSTGRES_DB: linkwave_db
    ports:
      - "5005:5432"
    volumes:
      - postgres_data:/var/lib/postgresql/data
    networks:
      - linkwave-network

  linkwave-keycloak:
    image: bitnami/keycloak:22.0.5
    depends_on:
      - keycloak-db
    container_name: linkwave-keycloak
    environment:
      PROXY_ADDRESS_FORWARDING: "true"
      DB_VENDOR: postgres
      DB_ADDR: keycloak-db
      DB_DATABASE: keycloak_db
      DB_USER: linkwave_user
      DB_PASSWORD: ndquang@2002
      KEYCLOAK_ADMIN: admin
      KEYCLOAK_ADMIN_PASSWORD: admin
    volumes:
      - ./realm-export.json:/opt/bitnami/keycloak/data/import/realm-
export.json
    ports:
      - "8081:8080"
    restart: unless-stopped
    networks:
      - linkwave-network

  keycloak-db:
    image: postgres:15
    container_name: keycloak-db
    restart: unless-stopped
    ports:
      - "5008:5432"
    environment:
      POSTGRES_DB: keycloak_db
```

```

    POSTGRES_USER: linkwave_user
    POSTGRES_PASSWORD: ndquang@2002
  volumes:
    - keycloak-db-data:/var/lib/postgresql/data
  networks:
    - linkwave-network

kafdrop:
  image: obsidiandynamics/kafdrop
  container_name: kafdrop
  restart: unless-stopped
  ports:
    - "9000:9000"
  environment:
    KAFKA_BROKERCONNECT: "kafka:29092"
  depends_on:
    - "kafka"
  networks:
    - linkwave-network

kafka:
  image: obsidiandynamics/kafka
  container_name: kafka
  restart: unless-stopped
  ports:
    - "2181:2181"
    - "9092:9092"
  environment:
    KAFKA_LISTENERS: "INTERNAL://:29092,EXTERNAL://:9092"
    KAFKA_ADVERTISED_LISTENERS:
"INTERNAL://kafka:29092,EXTERNAL://localhost:9092"
    KAFKA_LISTENER_SECURITY_PROTOCOL_MAP:
"INTERNAL:PLAINTEXT,EXTERNAL:PLAINTEXT"
    KAFKA_INTER_BROKER_LISTENER_NAME: "INTERNAL"
    KAFKA_ZOOKEEPER_SESSION_TIMEOUT: "6000"
    KAFKA_RESTART_ATTEMPTS: "10"
    KAFKA_RESTART_DELAY: "5"
    ZOOKEEPER_AUTOPURGE_PURGE_INTERVAL: "0"
    KAFKA_CREATE_TOPICS: "linkwave:1:1"
  networks:
    - linkwave-network

volumes:
  postgres_data:
    driver: local
  keycloak-db-data:
    driver: local

networks:
  linkwave-network:
    driver: bridge

```

1.2.2 Application (dev)

```
spring:
  datasource:
    url: jdbc:postgresql://localhost:5005/linkwave_db
    username: linkwave_user
    password: ndquang@2002
  flyway:
    locations:
      - classpath:db/migration/schema
      - classpath:db/migration/data
    clean-on-validation-error: true
  jpa:
    generate-ddl: off
    hibernate.ddl-auto: none
    hibernate.naming.physical-strategy:
      org.hibernate.boot.model.naming.PhysicalNamingStrategyStandardImpl
    database-platform: org.hibernate.dialect.PostgreSQLDialect
    show-sql: true
    properties:
      hibernate:
        jdbc.batch_size: 20
        order_inserts: true
        order_updates: true
        format_sql: true
        show_sql: true
  security:
    oauth2:
      resourceserver:
        jwt:
          issuer-uri: http://localhost:8081/realmss/linkwave
          jwk-set-uri:
            http://localhost:8081/realmss/linkwave/protocol/openid-connect/certs
      data.rest.base-path: /api
    servlet:
      multipart:
        file-size-threshold: 2KB
        max-file-size: 200MB
        max-request-size: 215MB
    kafka:
      consumer:
        bootstrap-servers: localhost:9092
        group-id: chat-group
        auto-offset-reset: earliest
      producer:
        bootstrap-servers: localhost:9092
  server:
    port: 8080
  jwt:
    auth:
      converter:
```

```

principle-attribute: preferred_username

keycloak:
  url: http://localhost:8081
  realm: master
  master-client-id: admin-cli #for master realm
  client-id: document-client
  username: admin
  password: admin
  roles: SUPER_ADMIN, ADMIN, USER
  defaultRole: USER
  token-life-span: 72000

cloudinary:
  cloud-name: dtywjboyt
  api-key: 435254598946868
  api-secret: LxCHcYYni46KYRnqkiemtLxL_q8

```

2. Frontend

2.1 Folder structure

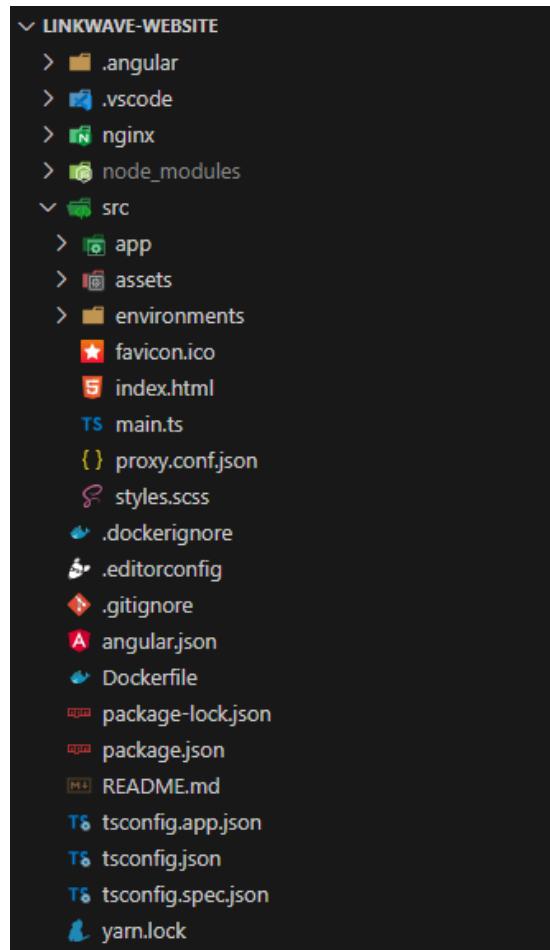


Figure 26. Frontend's folder structure

2.2 Config proxy

```
{  
  "/api/*": {  
    "target": "http://localhost:8080",  
    "secure": false  
  }  
}
```

In Angular, when I develop a frontend application, I need to communicate with backend APIs or services. However, during development, the backend might not be hosted on the same domain or port as my frontend application. This can lead to issues with CORS (Cross-Origin Resource Sharing) policies, which restrict requests from one origin to another.

To address this, Angular provides a way to configure a proxy. The proxy configuration allows me to route certain HTTP requests from your Angular app to a different backend server during development.

Here's a breakdown of the configuration:

"`/api/*`": This is a path pattern indicating which requests should be proxied. In this case, any request that starts with `/api/` will be proxied.

"`target`": "`http://localhost:8080/api/*` will be forwarded to `http://localhost:8080`.

"`secure`": `false`: This indicates whether the proxy should use HTTPS (true) or HTTP (false) when forwarding requests to the target server. In this case, it's set to false, meaning it will use HTTP.

So, in summary, this configuration ensures that any HTTP requests made to paths starting with `/api/` in app during development will be forwarded to `http://localhost:8080`, allowing me to interact with my backend API without running into CORS issues.

2.3 Function and interface

2.3.1 Register

The screenshot shows the 'Sign up' page for Link Wave. At the top right are 'Login' and 'Sign Up' buttons. Below them is a decorative illustration of a person sitting at a desk with a laptop, a potted plant, and a cup of coffee. A small link 'I am already a member' is located next to the illustration. The main form area contains fields for 'First Name' and 'Last Name', both with red validation circles. Below these are fields for 'Email address' and 'Password' (with a visibility toggle), and 'Confirm Password'. A checkbox for 'I agree all statements in Terms of service' is present. At the bottom is a red 'Sign up' button.

Figure 27. Register page

When the user clicks on the registration button, the system will redirect to the registration page. Here users can create their account after filling in all the fields.

If the user leaves any field blank or receives an incorrect password and password confirmation, the system displays an error message.

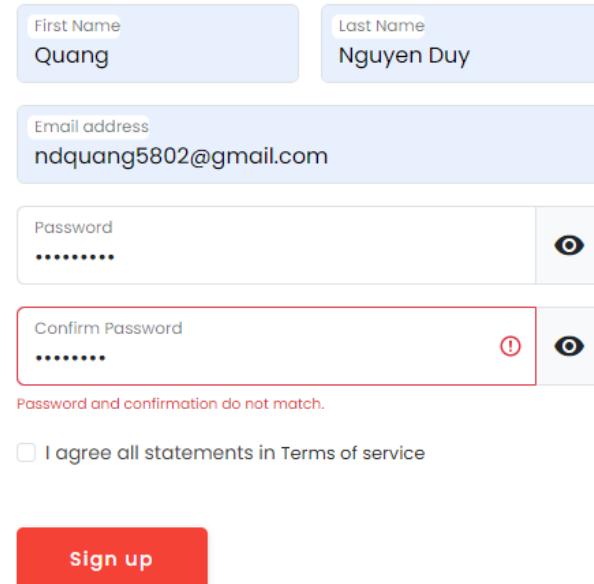


Figure 28. Display error when sign up (1)

This screenshot shows the 'Sign up' page with validation errors for empty fields. The 'First Name' and 'Last Name' fields have red validation circles with exclamation marks. Below each field is a red error message: 'Please enter first name' and 'Please enter last name'. The 'Email address' field also has a red validation circle with an exclamation mark and the error message 'Please enter email address'. The 'Password' and 'Confirm Password' fields have red validation circles with exclamation marks and the error message 'Please enter password'. The 'Confirm Password' field also has a red validation circle with an exclamation mark and the error message 'Please enter confirm password'. At the bottom left is a checkbox for 'I agree all statements in Terms of service'. A red 'Sign up' button is at the bottom center.

Figure 29. Enter do not enough field when siging up

Sign up



The screenshot shows a sign-up form with four input fields: First Name (Quang), Last Name (Nguyen Duy), Email address (ndquang5802@gmail.com), and Password (*****). Below the password field is a 'Confirm Password' field containing the same value (*****). A red border surrounds the 'Confirm Password' field, and a red exclamation mark icon is positioned next to it. Below the fields, a message states 'Password and confirmation do not match.' There is also a checkbox labeled 'I agree all statements in Terms of service' and a red 'sign up' button.

Figure 30. Enter password and confirm password do not match

After filling in all information, the system begins processing and sends a confirmation email to the user.

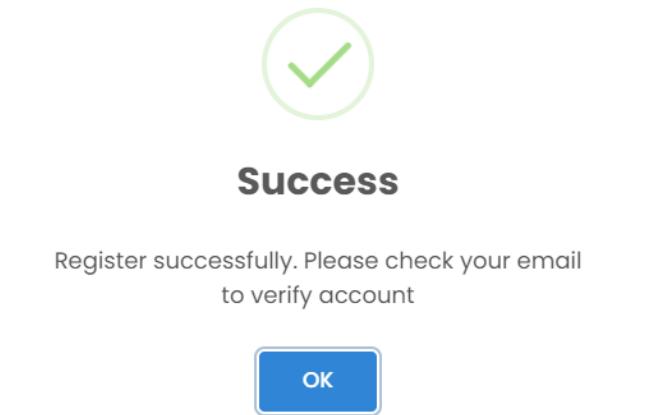


Figure 31. Display message when signing up successfully

Update Your Account

From: <no-reply@linkwave.com>
To: <ndquang5802@gmail.com>

2024-04-29 08:25, 2.8 KB

Show Headers

[HTML](#) [HTML Source](#) [Text](#) [Raw](#) [Spam Analysis](#) [HTML Check](#) [Tech Info](#)



Your administrator has just requested that you update your Linkwave account by performing the following action(s): Verify Email. Click on the link below to start this process.

[Link to account update](#)

This link will expire within 12 hours.

If you are unaware that your administrator has requested this, just ignore this message and nothing will be changed.

Figure 32. Email verify account

2.3.2 Login

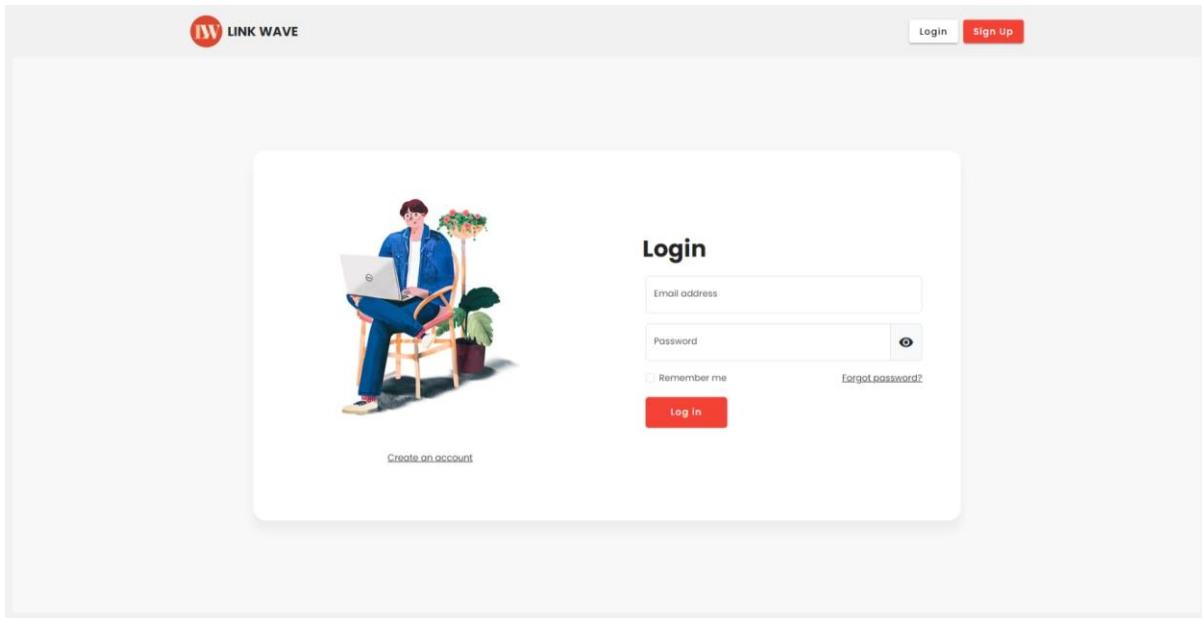


Figure 33. Login page

When the user successfully registers, the system will redirect to the login page. Here users need to enter email and password to log in.

In case the user has not confirmed the email, the system will display an error message.

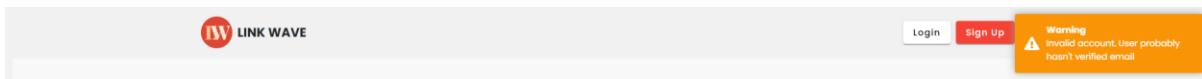


Figure 34. User have to verify email before login

Besides, if the user enters the wrong email or password, the system will also display an error message.

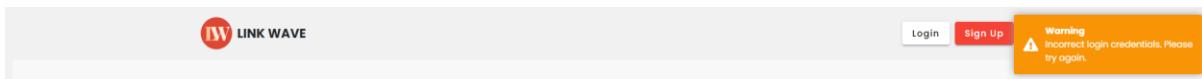


Figure 35. Incorrect credentials

2.3.3 Forgot password

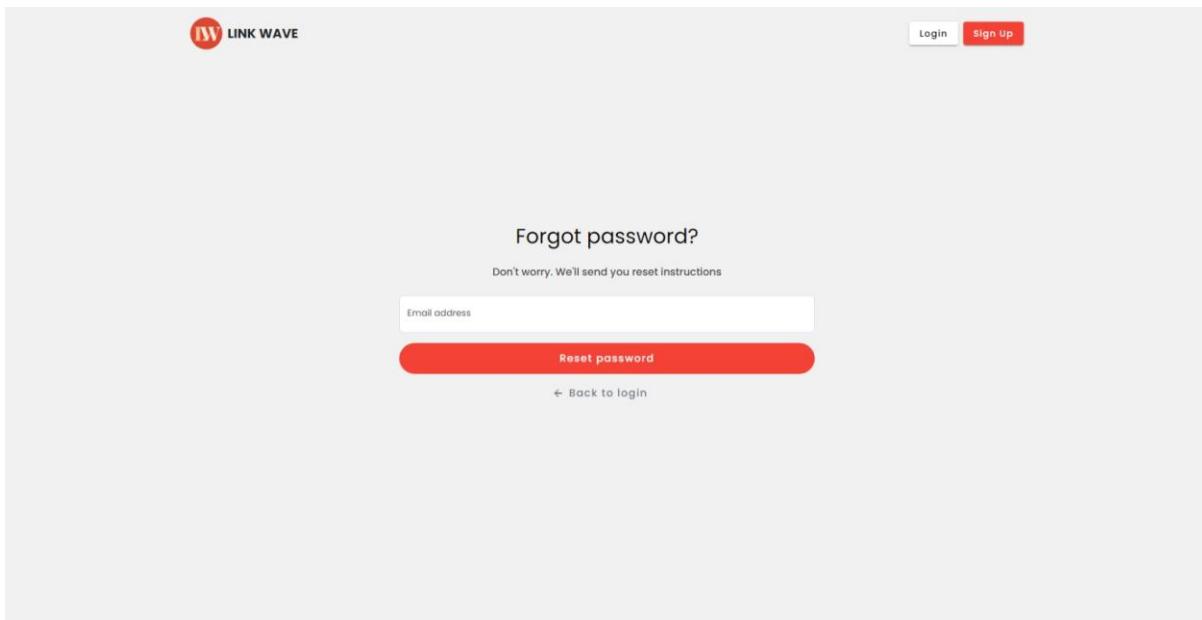


Figure 36. Forgot password page

During use, if the user forgets their password, they can reset it by going to the login page and clicking "forgot password". Users just need to enter their email and the system will send an email to change their password.

If the email does not exist, the system will display an error message.

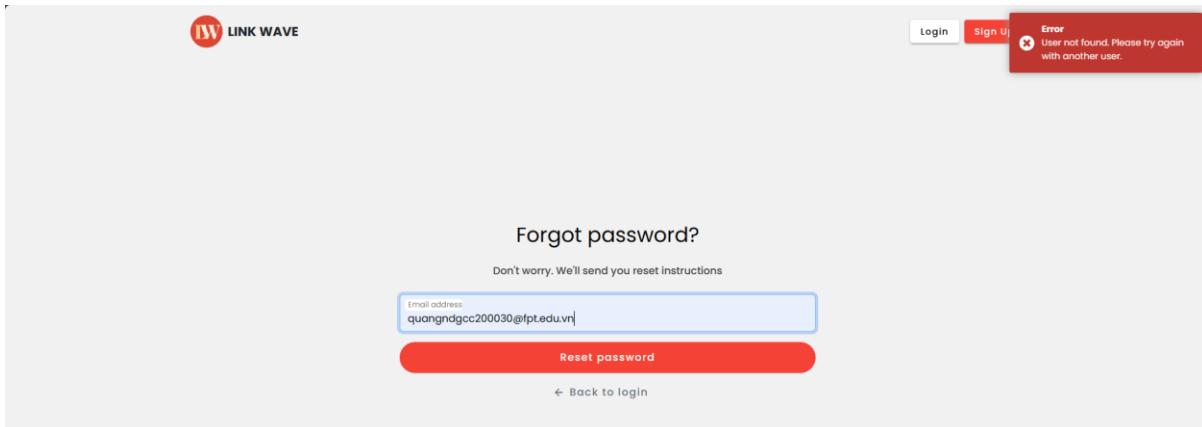


Figure 37. User does not exist in system

After receiving the email and the system checks whether the account exists, it will redirect to the email check page. Here, if the user does not receive the email after 1 minute, they can click resend the email to send it again.

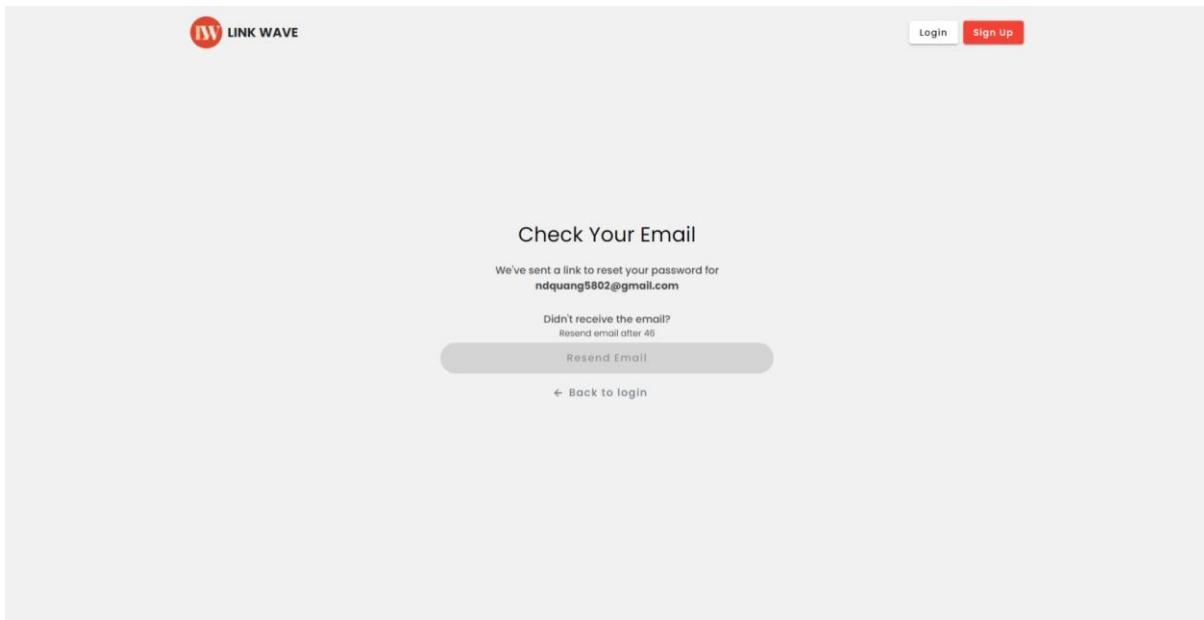


Figure 38. Check email to reset password

An email message titled 'Update Your Account'. The header shows 'From: <no-reply@linkwave.com>', 'To: <ndquang5802@gmail.com>', and a timestamp '2024-04-29 08:39, 2.8 KB'. Below the header are tabs for 'HTML' (which is selected), 'HTML Source', 'Text', 'Raw', 'Spam Analysis', 'HTML Check' (with a red error icon), and 'Tech Info'. The main body of the email contains instructions for account update, a link to 'Link to account update', a note about the link expiring in 12 hours, and a disclaimer about ignoring the message if unaware.

Figure 39. Email to reset password

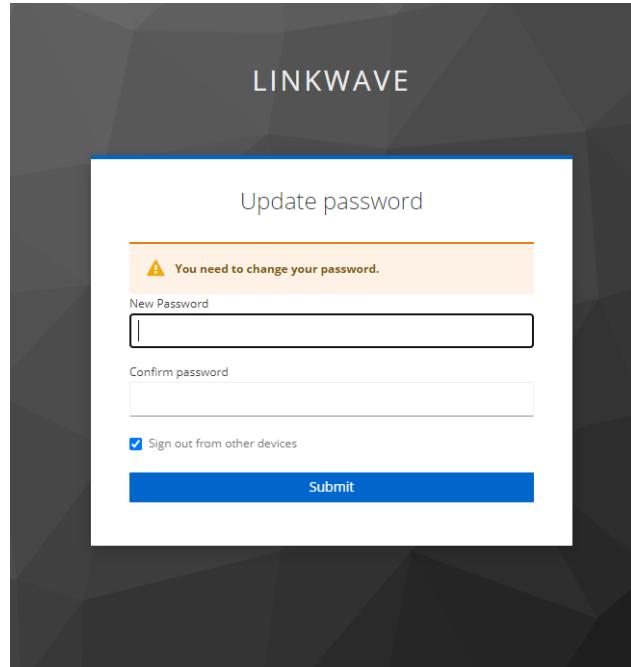


Figure 40. Enter new password and confirm password

2.3.4 Home

After successfully logging in, the system will redirect to the home page. Here users can surf to see other users' posts.

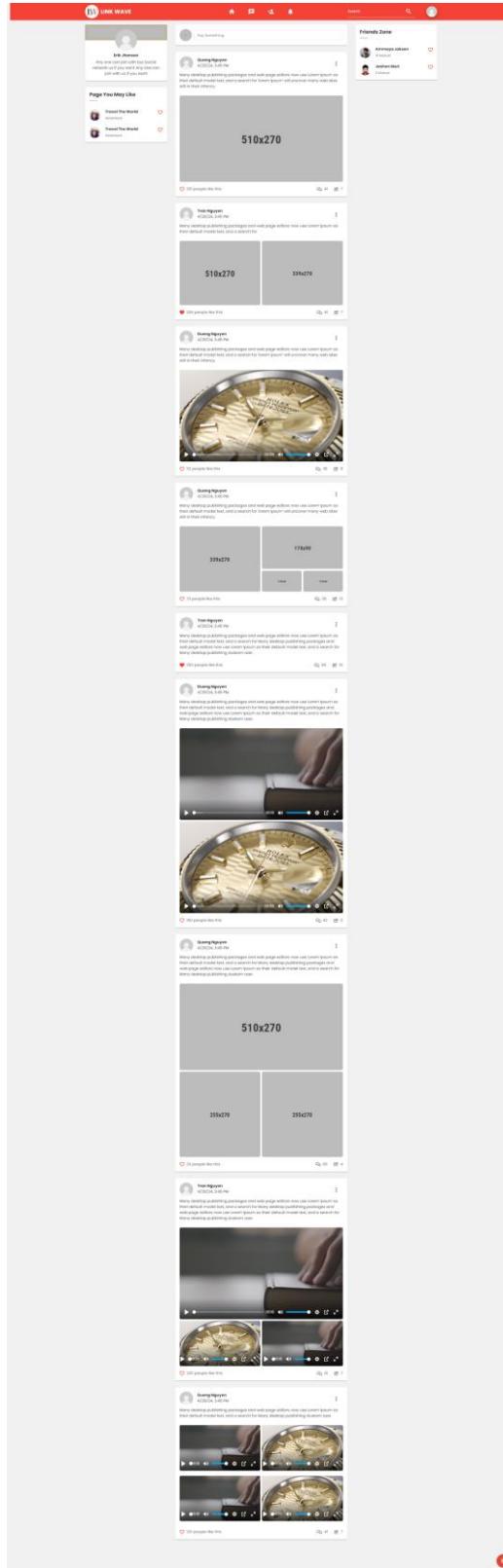


Figure 41. Home page

2.3.5 Manage profile

If the user wants to update personal information, they will go to "Profile" and select "About". Because the account is newly created, it only has information when the user registered.

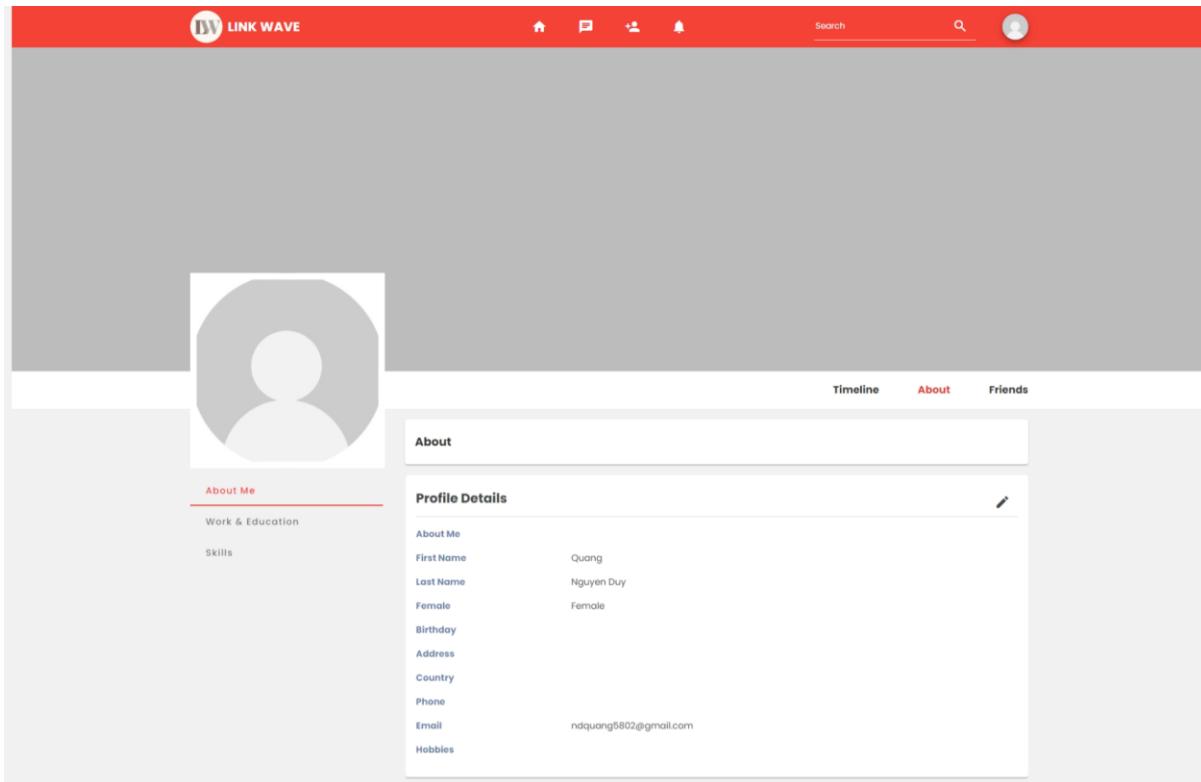


Figure 42. About user page

If you want to update information, users need to click on the pen icon

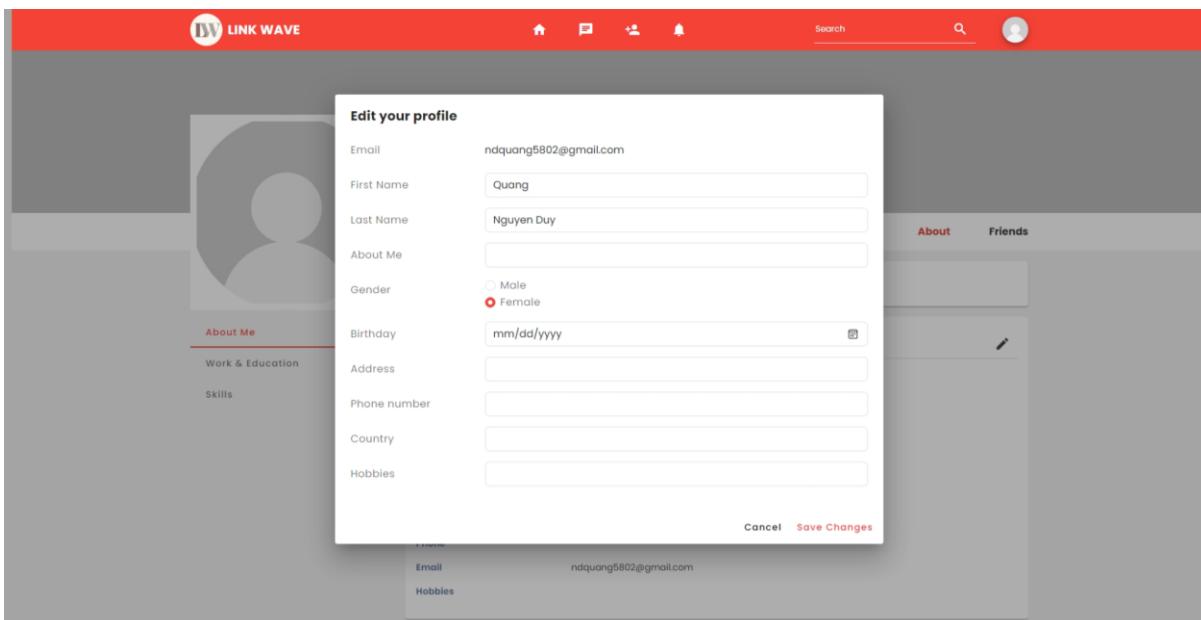


Figure 43. Update personal information

Edit your profile

Email	ndquang5802@gmail.com
First Name	Quang
Last Name	Nguyen Duy
About Me	Armed with early exposure to Information Technology and a passion for programming, I am eager to seek opportunities with a company that can provide practical experience.
Gender	<input type="radio"/> Male <input checked="" type="radio"/> Female
Birthday	08/05/2002 [Calendar icon]
Address	Ninh Kieu, Can Tho city
Phone number	0916843367
Country	Vietnam
Hobbies	Coding Travel
Cancel Save Changes	

Figure 44. Enter all information when edit profile

The screenshot shows the BV Link Wave profile editing interface. At the top, there's a navigation bar with the BV Link Wave logo, search, and friend-related icons. A green success message box on the right says "Success: Update profile successfully". Below the navigation, there's a large placeholder for a profile picture. On the left, a sidebar has tabs for "About Me" (which is active), "Work & Education", and "Skills". The main content area has tabs for "Timeline", "About" (which is active), and "Friends". Under the "About" tab, there's a section titled "Profile Details" with a table of information:

Profile Details	
About Me	Armed with early exposure to Information Technology and a passion for programming, I am eager to seek opportunities with a company that can provide practical experience.
First Name	Quang
Last Name	Nguyen Duy
Female	Female
Birthday	August 5, 2002
Address	Ninh Kieu, Can Tho city
Country	Vietnam
Phone	0916843367
Email	ndquang5802@gmail.com
Hobbies	Coding Travel

Figure 45. Edit profile successfully

2.3.6 Manage experience

In the experience management section, I divide it into two parts: education and work.

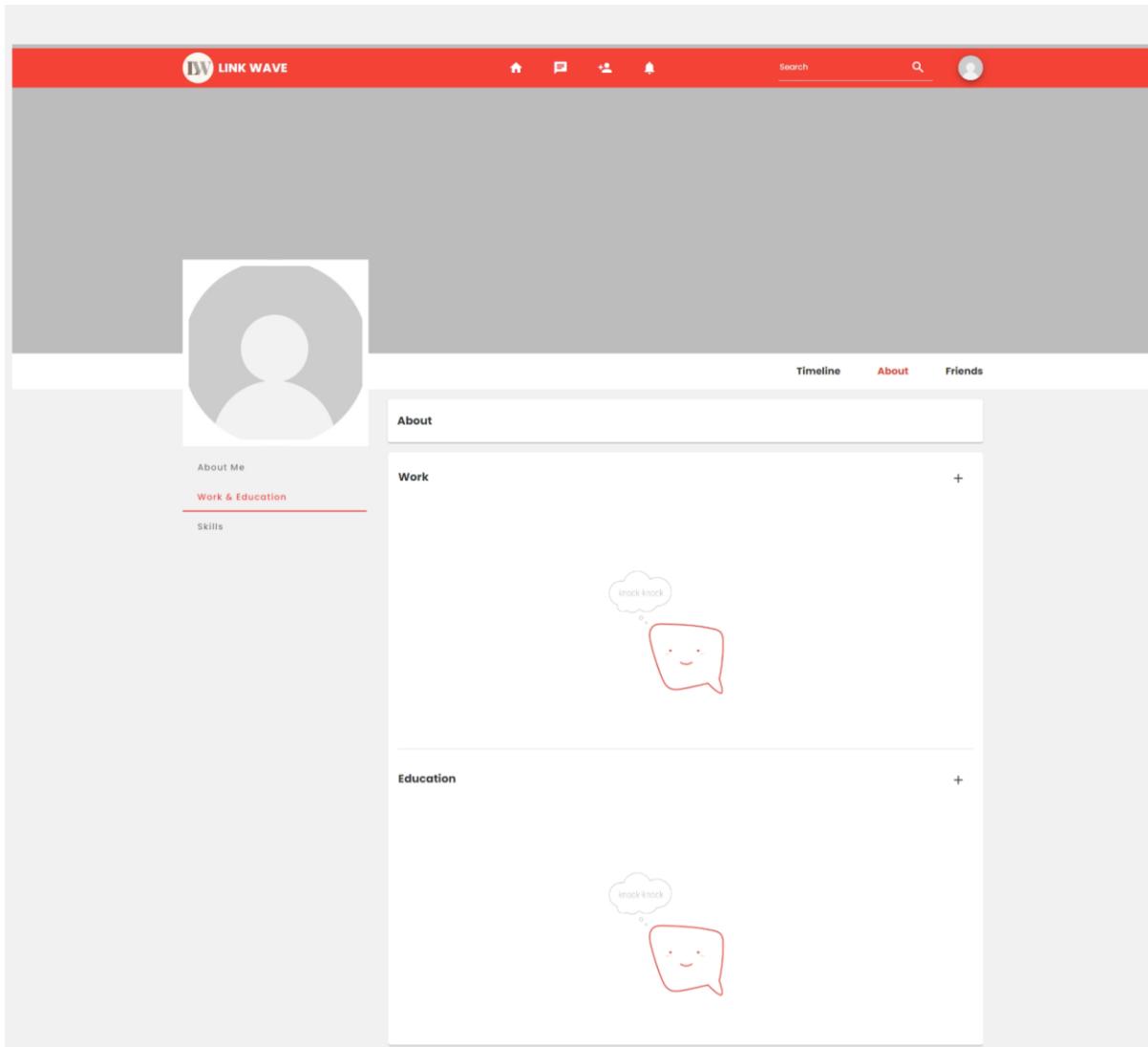


Figure 46. Manage experience

In the experience management section, I divide it into two parts: education and work.

First about work. To be able to add, users will click on the plus icon.

Your Experience

Company Name (School)

Position (Degree)

Description

Start Date mm/dd/yyyy

End Date mm/dd/yyyy

Location

[Cancel](#) [Save Changes](#)

Figure 47. Add new work experience

Work

07/2023 - 10/2023

Agilino
Software Developer
Ninh Kieu, Can Tho city

Develop and maintain web applications. Build quality source code, maximizing scalability. Develop and deploy new features. Writing testing and bug fixes to ensure source code quality.

Figure 48. A work was added

If the user is still working at that company, just leave the end date blank

Work

11/2023 - Present

Teknix Corporation
Back-end developer
Ninh Kieu, Can Tho city

Develop and maintain web applications. Build quality source code, maximizing scalability. Develop and deploy new features.

07/2023 - 10/2023

Agilino
Software Developer
Ninh Kieu, Can Tho city

Develop and maintain web applications. Build quality source code, maximizing scalability. Develop and deploy new features. Writing testing and bug fixes to ensure source code quality.

Figure 49. Add more work experience

Next is about education. To be able to add, users will click on the plus icon.

Your Experience

Company Name (School)	University of Greenwich
Position (Degree)	Bachelor of Science
Description	A total of three years of intensive study in programming, application development, project management, and information technology principles. In particular, we have focused on improving skills in design analysis, database knowledge, and logical thinking.
Start Date	08/20/2020
End Date	mm/dd/yyyy
Location	Ninh Kieu, Can Tho city

Cancel **Save Changes**

Figure 50. Add new education

Education

08/2020 – Present	University of Greenwich	Bachelor of Science	Ninh Kieu, Can Tho city	
A total of three years of intensive study in programming, application development, project management, and information technology principles. In particular, we have focused on improving skills in design analysis, database knowledge, and logical thinking.				

Figure 51. New education was added

2.3.7 Manage skills

Next is the user skill management section. Here they can add the skills they currently have.

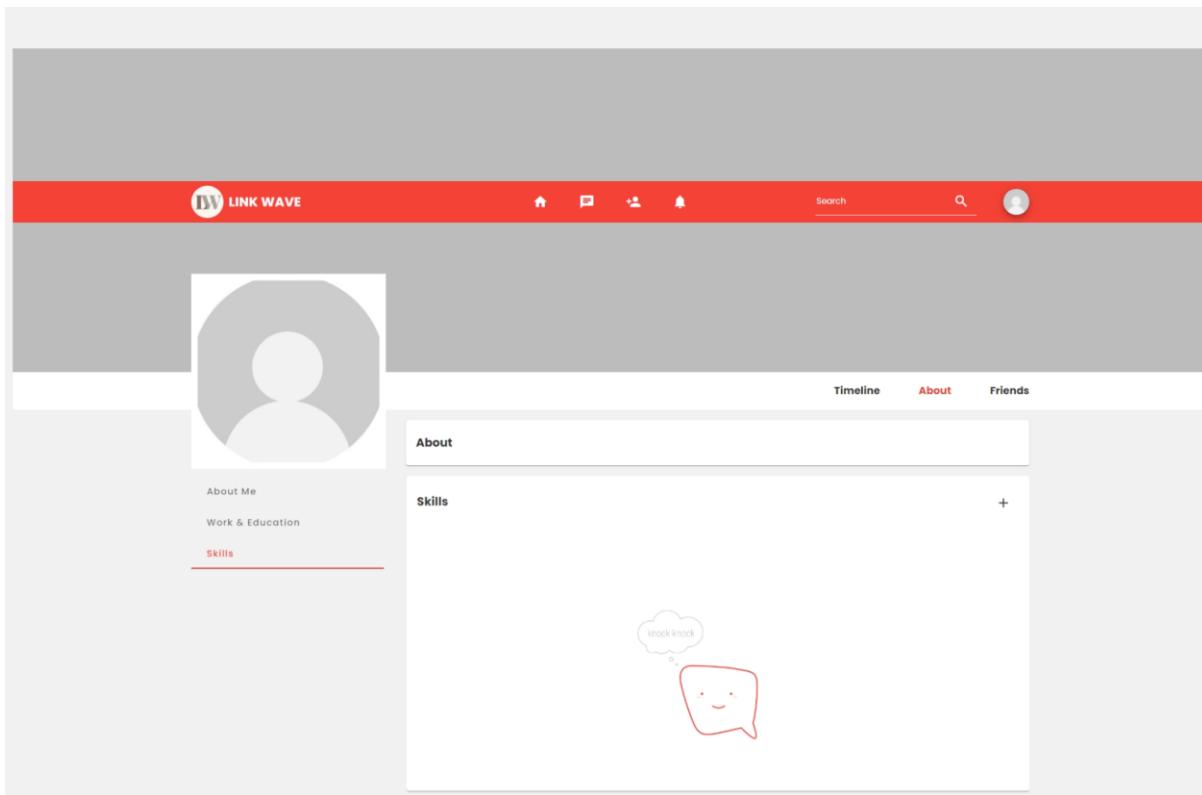


Figure 52. Manage skills

To be able to add, users will click on the plus icon.

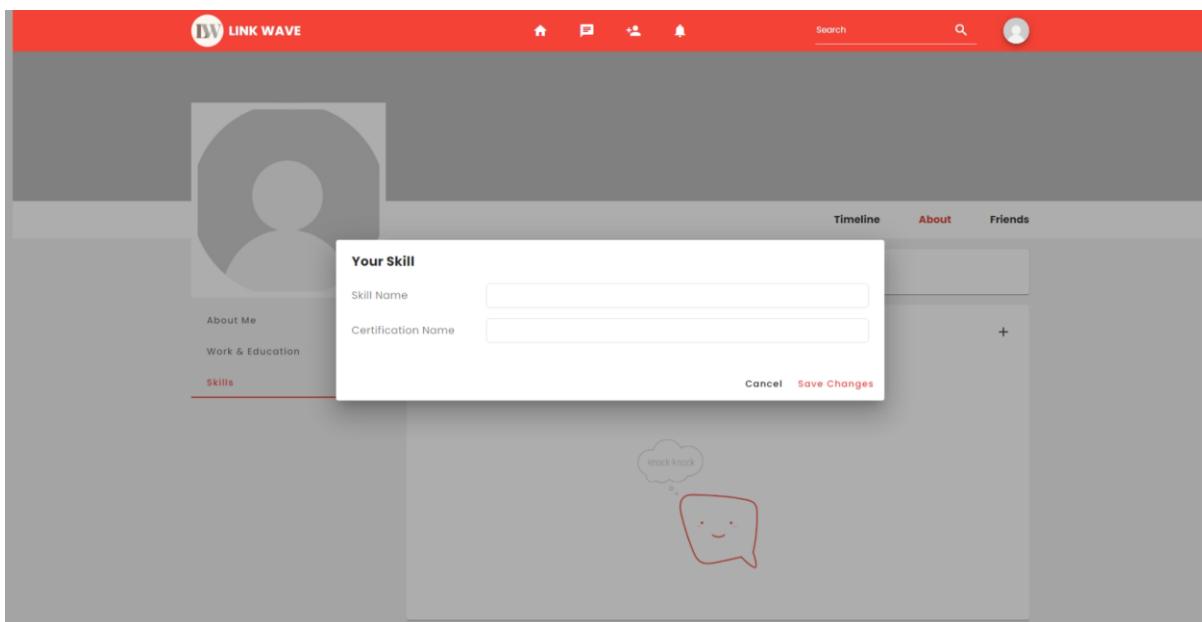


Figure 53. Add new skill

The screenshot shows a section titled "Skills" with a "+" button in the top right corner. Below it, there are two entries: "Spring Boot" with the subtext "Master Spring Boot 3 and Spring Framework 6 with Java" and "Angular" with the subtext "Angular for Beginners Course (Angular 17)". Each entry has a pencil icon to its right.

Figure 54. New skills was added

2.3.8 Change avatar

When needing to change the avatar image, the user points to the avatar image and clicks "Change Picture".

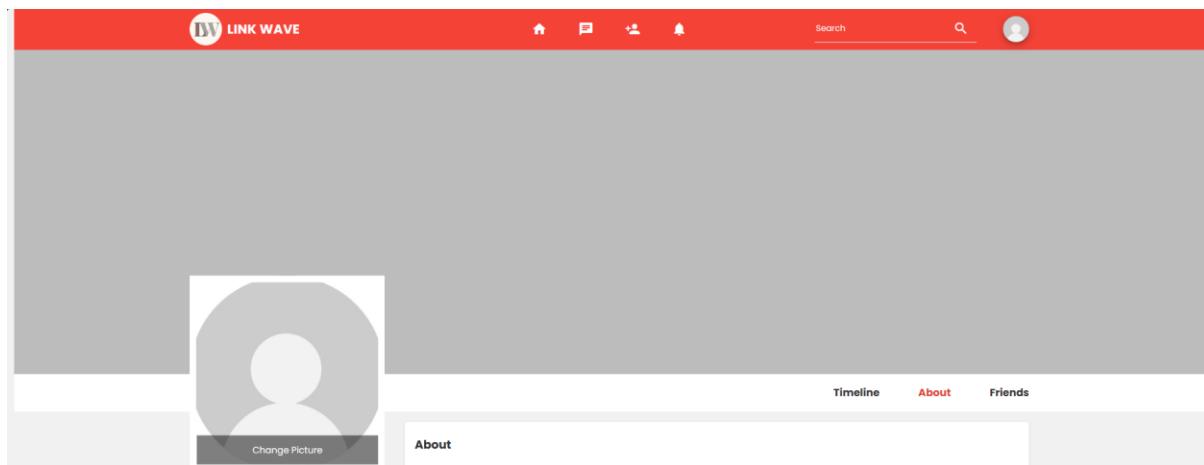


Figure 55. Hover to change avatar

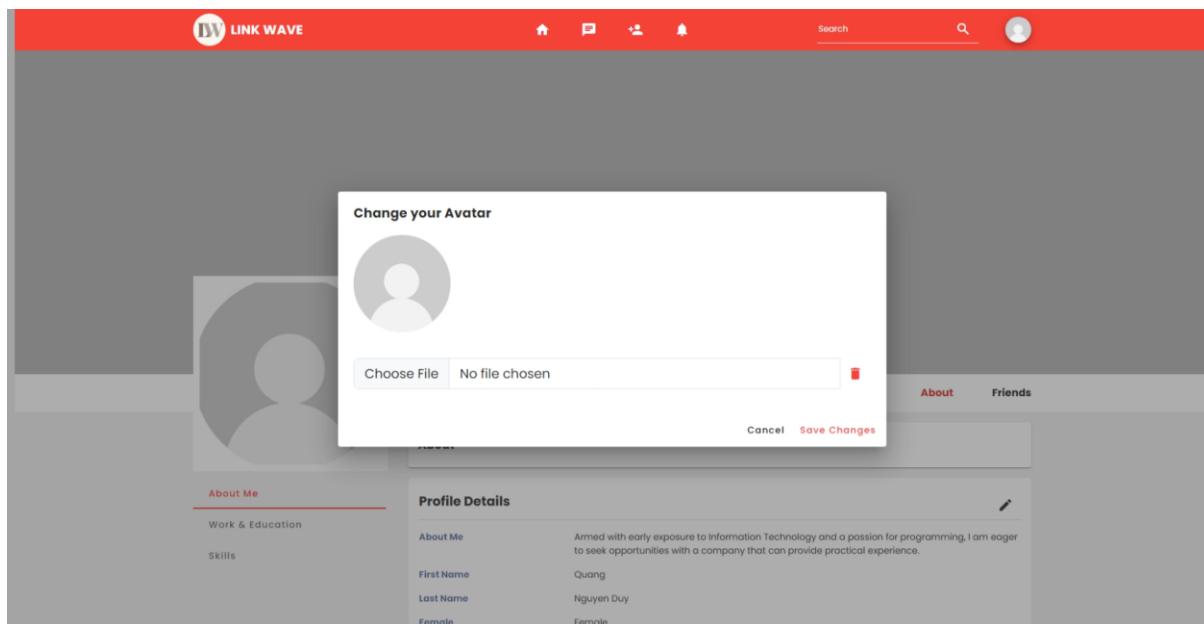


Figure 56. Choose new avatar

Next, the user chooses their avatar and the system will ask to crop the image to fit the frame.

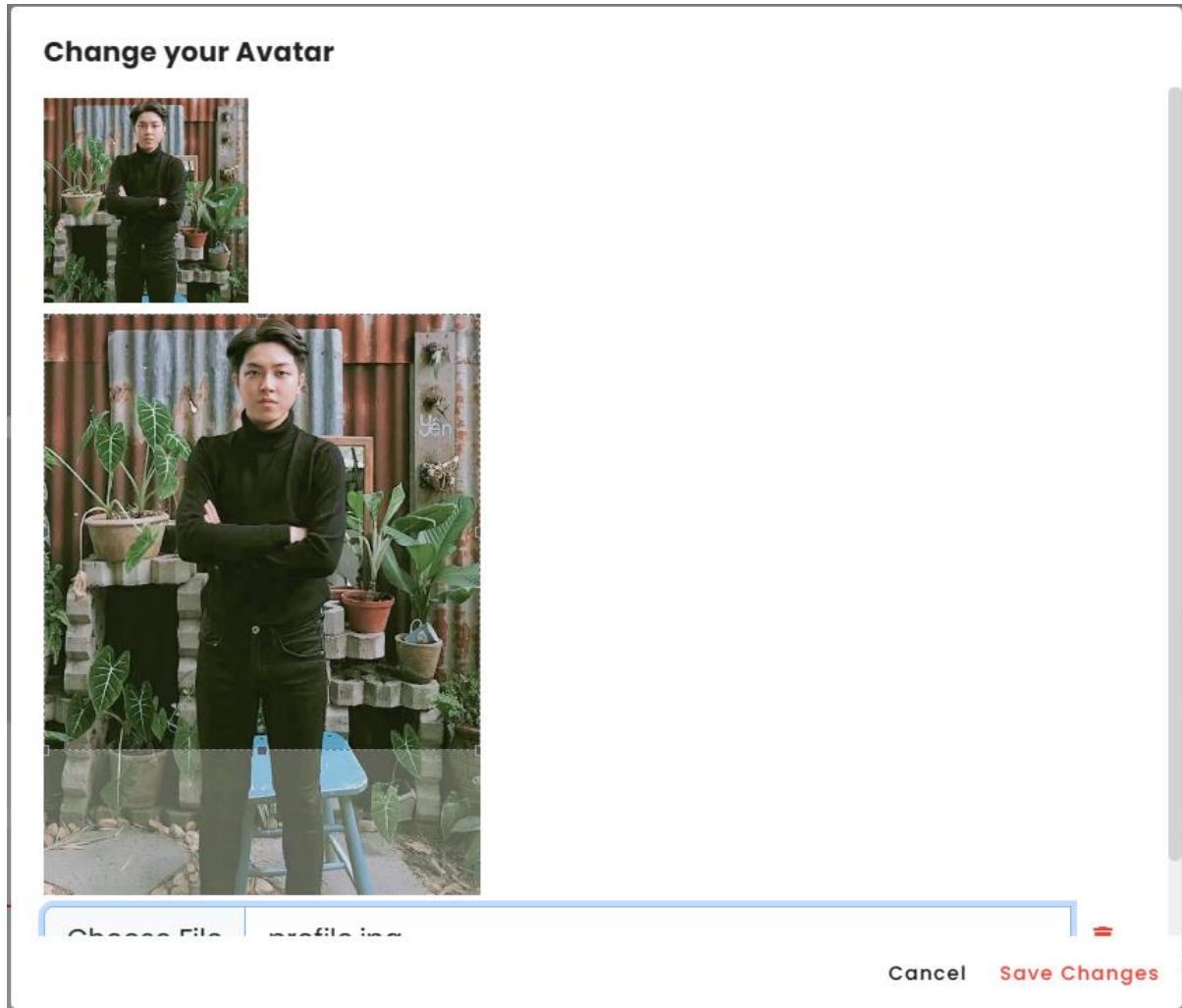


Figure 57. Choose new avatar and crop it

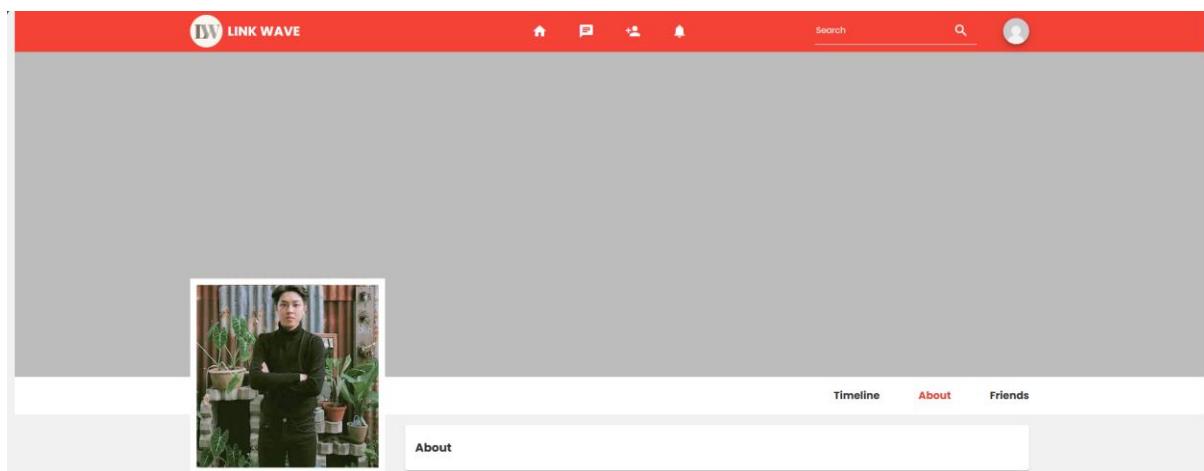


Figure 58. New avatar was saved

2.3.9 Change cover

When needing to change the cover image, the user points to the cover image and clicks "Change Picture".

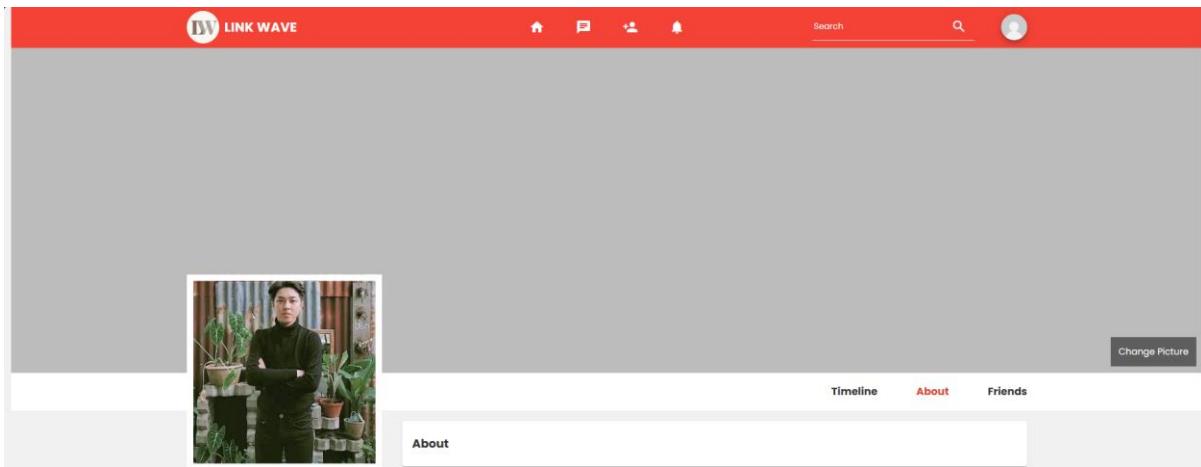


Figure 59. Hover to change cover



Figure 60. Choose new cover

Next, the user chooses their cover and the system will also ask to crop the image to fit the frame.

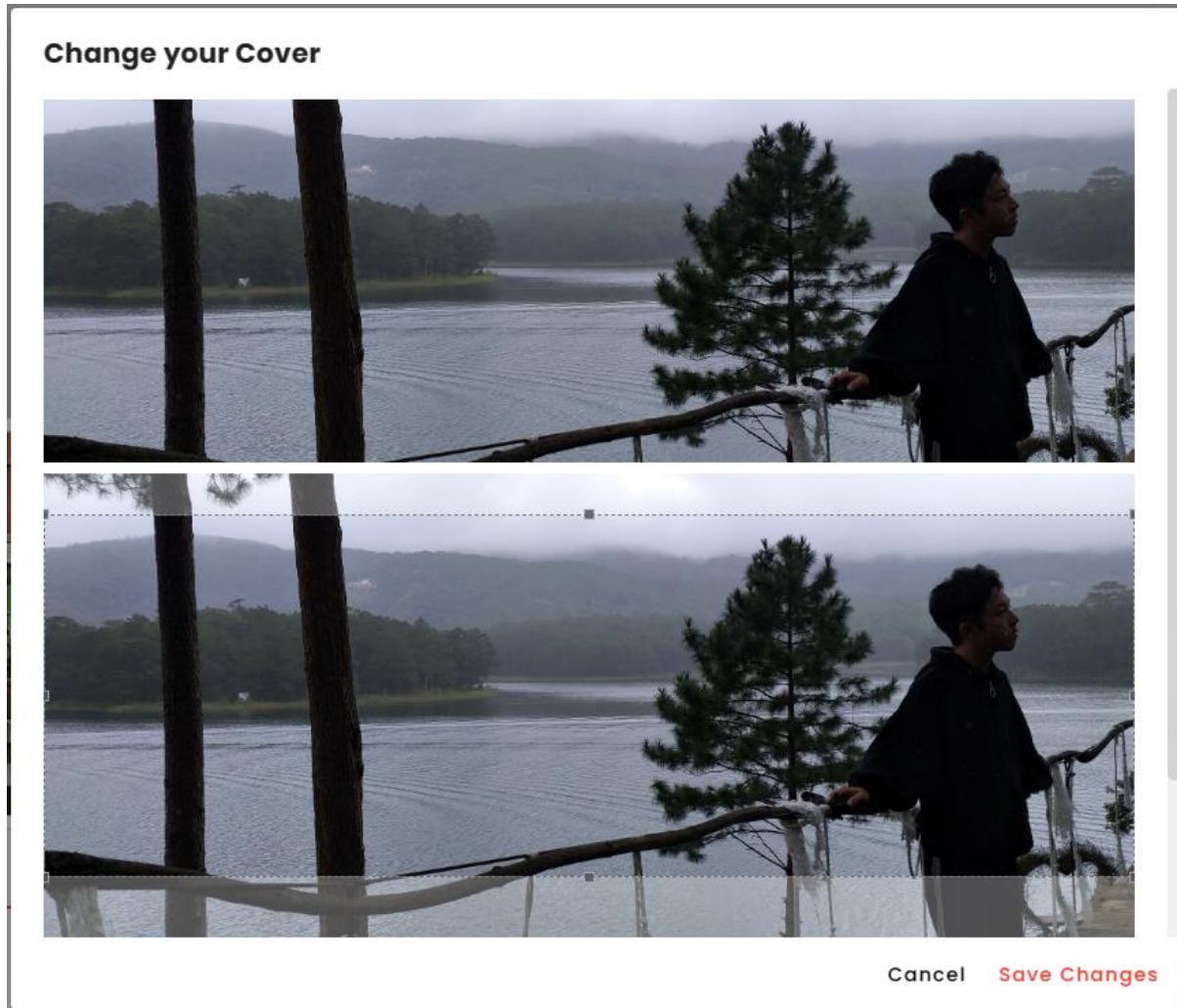


Figure 61. Choose new cover and crop it

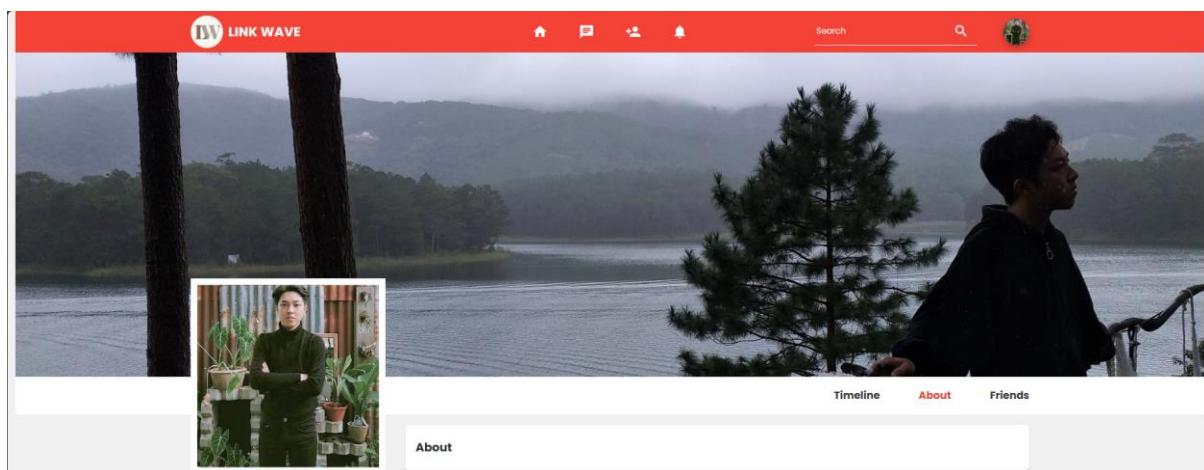


Figure 62. New cover was saved

2.3.10 Friend request

When users want to make friends, they click on the add friend icon on the navigation bar. This page displays friend suggestions and requests from other users.

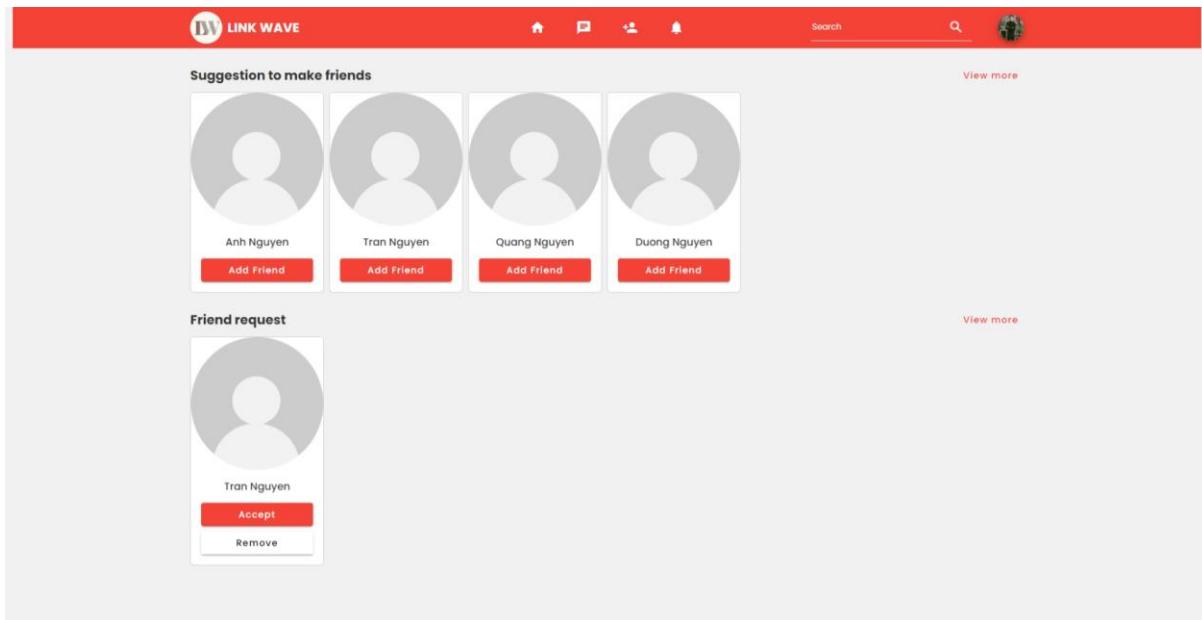


Figure 63. View friend

When the user clicks on "Add friend", the invitation will be sent to the recipient and the button will switch to "Recall" so the user can revoke the friend request.

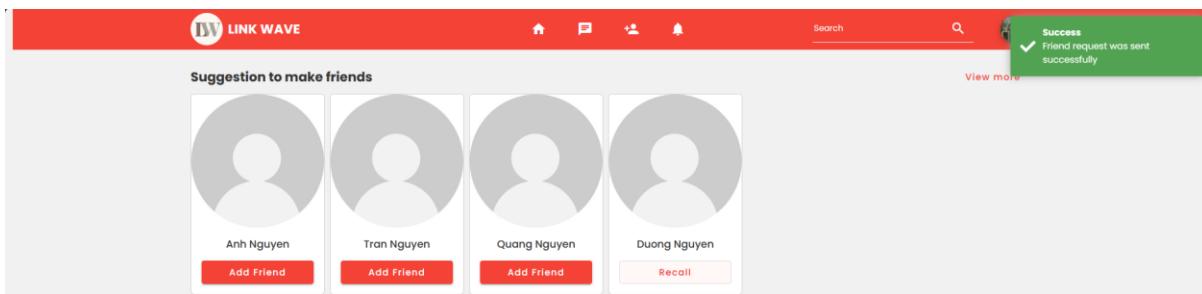


Figure 64. Send friend request

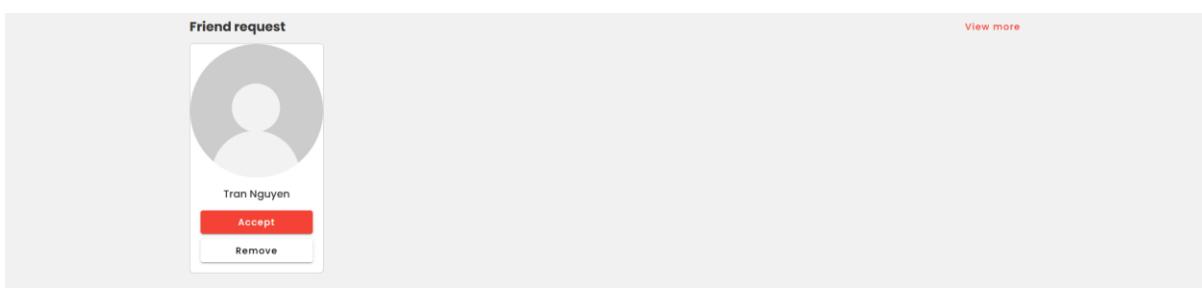


Figure 65. Receive friend request

After accepting the friend request, users can go to their friends list.

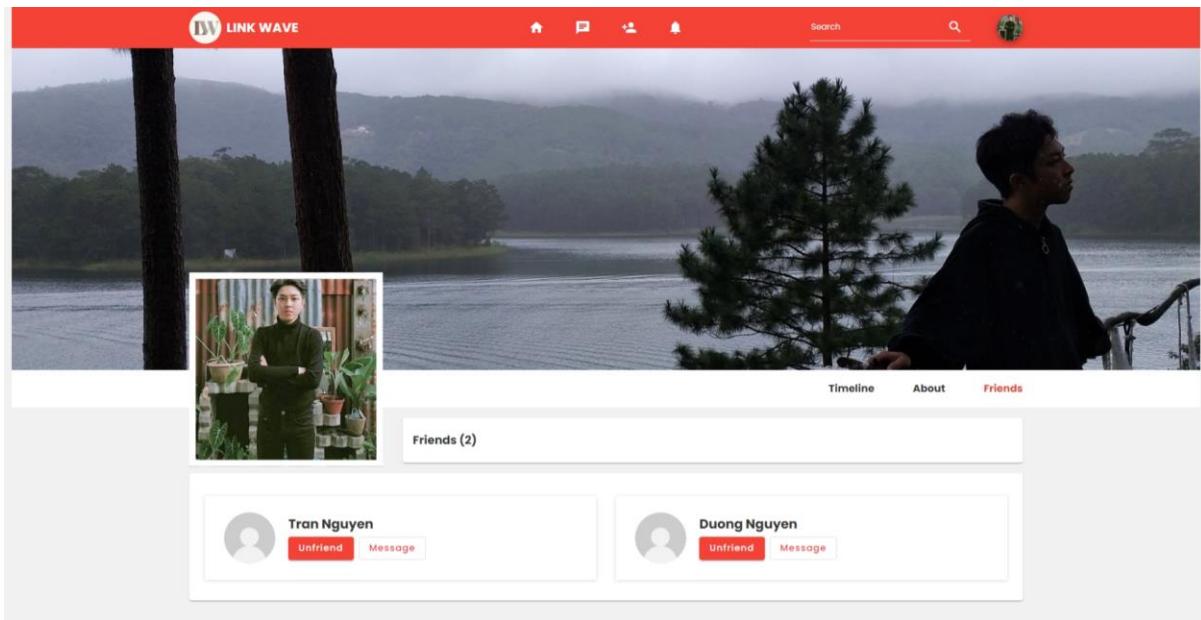


Figure 66. List friends of user

2.3.11 Chat message

To start a conversation, users can click the "Message" button in the friends list. At this point, the system will redirect to the chat page so they can chat with each other

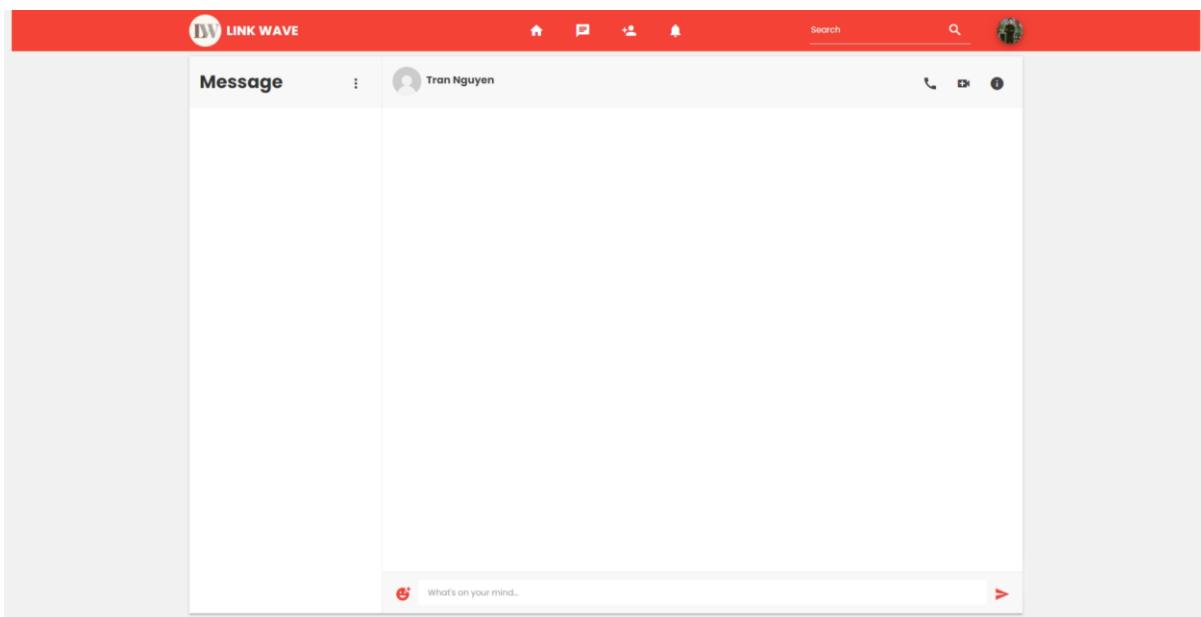


Figure 67. New conversation

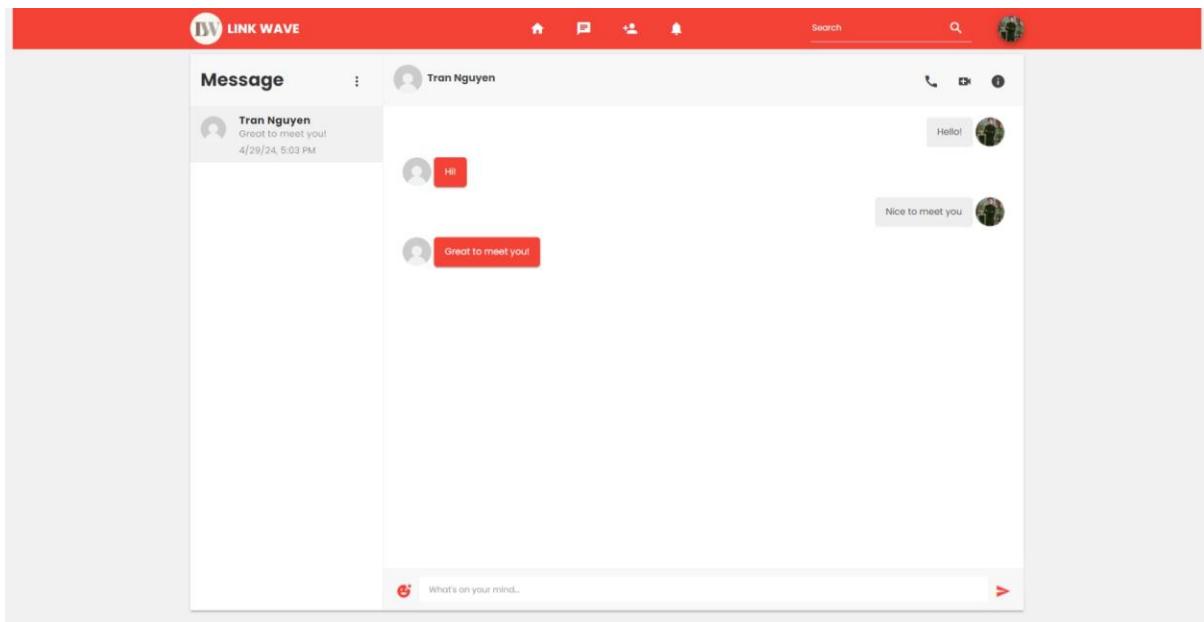


Figure 68. Message from current user

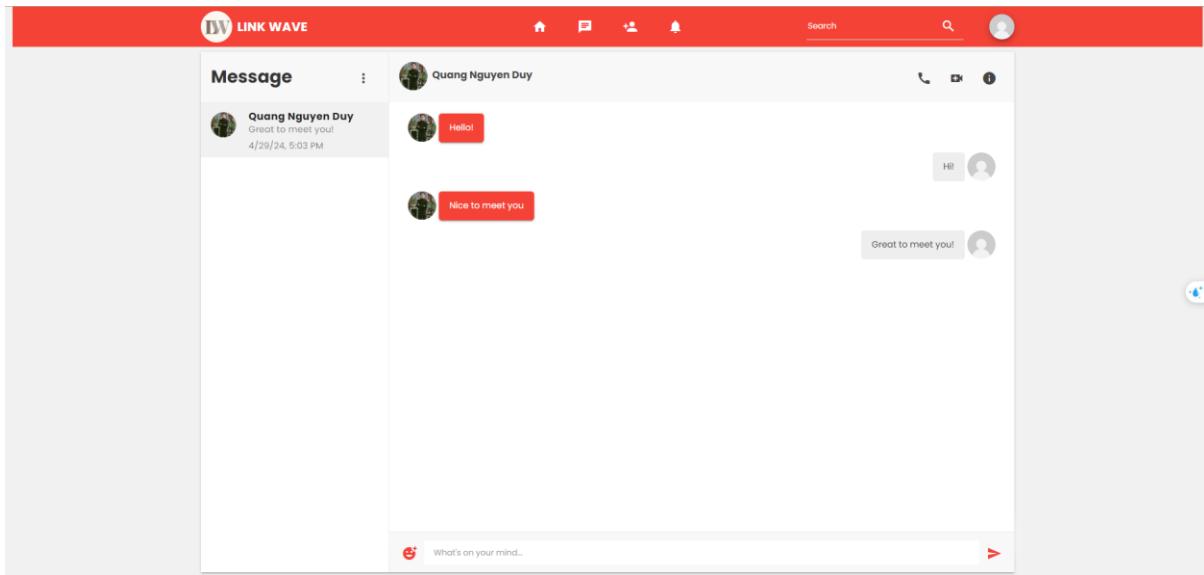


Figure 69. Message from friend of user

2.3.12 Notification

If users want to review notifications, they can click the bell icon on the navigation bar. This page will display all user notifications as well as unread notifications.

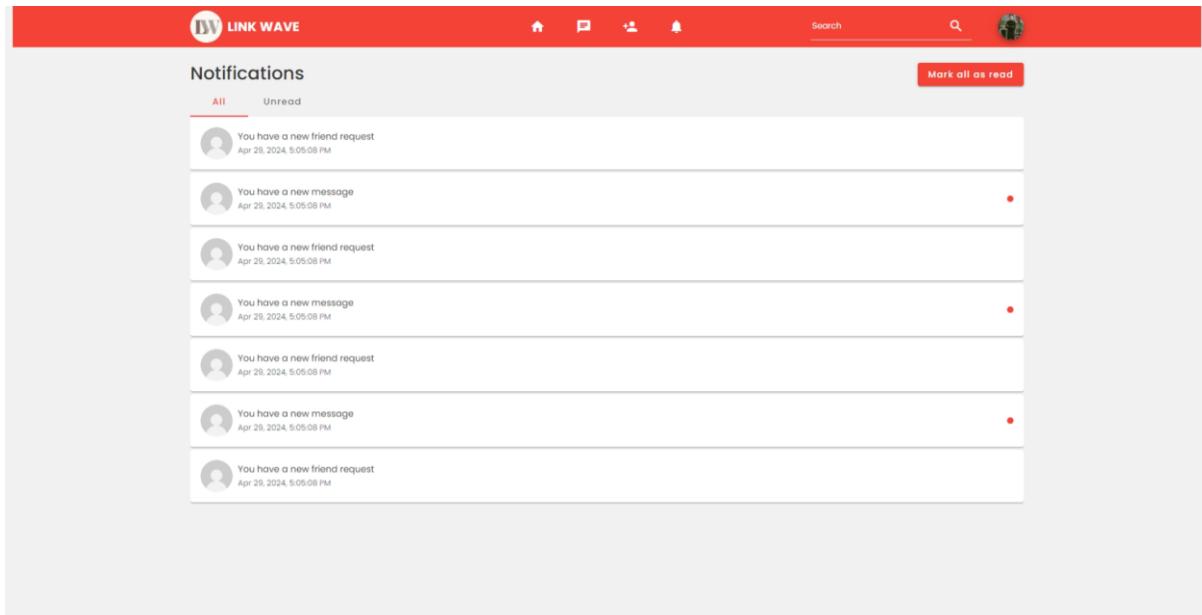


Figure 70. Notification

3. Deploy Digital Ocean

3.1 Create droplet



Figure 71. Create droplet and assign domain

3.2 Config environment

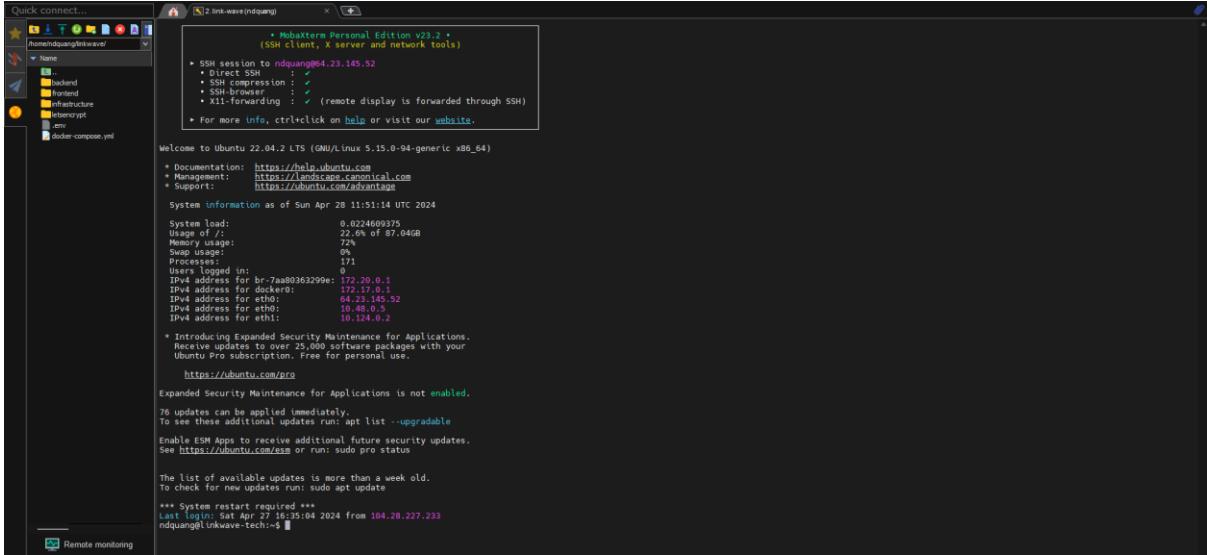


Figure 72. Access to the server by using SSH



Figure 73. .env file

```
GNU nano 6.2
version: '3.8'

services:
  traefik:
    image: traefik:v2.5
    container_name: traefik
    restart: unless-stopped
    logging:
      # Configure the logs retention according to your needs
      options:
        max-size: "10m"
        max-file: "3"
    env_file:
      - ./.env
    command:
      - "--api.insecure=true"
      # Get Docker as the provider
      - "--providers.docker=true"
      # Avoid that all containers are exposed
      - "--providers.docker.exposedbydefault=false"
      # Settle the ports for the entry points
      - "--entrypoints.web.address=:80"
      - "--entrypoints.web.http.redirects.entrypoint.to=websecure"
      - "--entrypoints.web.http.redirects.entrypoint.scheme=https"
      - "--entrypoints.websecure.http.tls=true"
      - "--entrypoints.websecure.http.tls.certResolver=letsencrypt"
      - "--certificatesresolvers.letsencrypt.acme.tlschallenge=true"
      - "--entrypoints.websecure.http.tls.domains[0].main=${ROOT_DOMAIN}"
      - "--entrypoints.websecure.http.tls.domains[0].sans=*.${ROOT_DOMAIN}"
      - "--entrypoints.websecure.address=:443"
      - "--certificatesresolvers.letsencrypt.acme.email=${USER_MAIL}"
      - "--certificatesresolvers.letsencrypt.acme.storage=/letsencrypt/acme.json"
    ports:
      - "80:80"
      - "443:443"
      - "8080:8080"
    networks:
      - linkwave-network
    volumes:
      # Store certificates in ./letsencrypt/acme.json
      - ./letsencrypt:/letsencrypt
      # Connect to Docker socket
      - /var/run/docker.sock:/var/run/docker.sock:ro
```

Figure 74. Docker compose file (traefik)

```

frontend:
  build:
    context: ./frontend
    dockerfile: Dockerfile
    container_name: frontend
  networks:
    - linkwave-network
  labels:
    traefik.enable: true
    traefik.http.routers.frontend.rule: Host(`${ROOT_DOMAIN}`)
    traefik.http.routers.frontend.entrypoints: websecure
    traefik.http.routers.frontend.tls.certresolver: letsencrypt
    traefik.http.routers.frontend.service: frontend
    traefik.http.services.frontend.loadbalancer.server.port: 80

backend:
  build:
    context: ./backend
    dockerfile: Dockerfile
    container_name: backend
    restart: unless-stopped
  environment:
    SPRING_DATASOURCE_URL: jdbc:postgresql://linkwave-db:5432/linkwave_db
    SPRING_DATASOURCE_USERNAME: linkwave_user
    SPRING_DATASOURCE_PASSWORD: ndquang@2002
    KEYCLOAK_URL: https://keycloak.linkwave.tech
    KEYCLOAK_USERNAME: admin
    KEYCLOAK_PASSWORD: admin
    KEYCLOAK_TOKEN_LIFE_SPAN: 72000
    CLOUDINARY_CLOUD_NAME: dtywjb0yt
    CLOUDINARY_API_KEY: 435254598946868
    CLOUDINARY_API_SECRET: LxChcYYni46KYRnqkiemtLxL_q8
    KAFKA_BOOTSTRAP_SERVERS: 64.23.145.52:9092
    KAFKA_GROUP_ID: chat-group
  networks:
    - linkwave-network
  depends_on:
    - linkwave-db
  labels:
    traefik.enable: true
    traefik.http.routers.backend.rule: Host(`api.${ROOT_DOMAIN}`)
    traefik.http.routers.backend.entrypoints: websecure
    traefik.http.routers.backend.tls.certresolver: letsencrypt
    traefik.http.routers.backend.service: backend
    traefik.http.services.backend.loadbalancer.server.port: 8080

```

Figure 75. Docker compose file (frontend and backend)

```

linkwave-db:
  image: postgres:15
  container_name: linkwave-db
  restart: unless-stopped
  environment:
    POSTGRES_USER: linkwave_user
    POSTGRES_PASSWORD: ndquang@2002
    POSTGRES_DB: linkwave_db
  ports:
    - "5005:5432"
  volumes:
    - postgres_linkwave_data:/var/lib/postgresql-linkwave/data
  networks:
    - linkwave-network

pgadmin4:
  image: dpage/pgadmin4:latest
  container_name: pgadmin4
  restart: unless-stopped
  environment:
    PGADMIN_DEFAULT_EMAIL: quangndgcc200030@fpt.edu.vn
    PGADMIN_DEFAULT_PASSWORD: ndquang@2002
  depends_on:
    - linkwave-db
  networks:
    - linkwave-network
  labels:
    traefik.enable: true
    traefik.http.routers.pgadmin4.rule: Host(`pgadmin.${ROOT_DOMAIN}`)
    traefik.http.routers.pgadmin4.entrypoints: websecure
    traefik.http.routers.pgadmin4.tls.certresolver: letsencrypt
    traefik.http.routers.pgadmin4.service: pgadmin4
    traefik.http.services.pgadmin4.loadbalancer.server.port: 80

```

Figure 76. Docker compose file (database and pgadmin)

```

keycloak:
  image: bitnami/keycloak:22.0.5
  depends_on:
    - keycloak-db
  container_name: keycloak
  restart: unless-stopped
  environment:
    PROXY_ADDRESS_FORWARDING: true
    DB_VENDOR: postgres
    DB_ADDR: keycloak_db
    DB_DATABASE: keycloak_db
    DB_USER: linkwave_user
    DB_PASSWORD: duyquang2002.
    KEYCLOAK_ADMIN: admin
    KEYCLOAK_ADMIN_PASSWORD: admin
    KC_PROXY: edge
    KEYCLOAK_EXTRA_ARGS: --import-realm
  volumes:
    - ./infrastructure/keycloak/realm:/opt/bitnami/keycloak/data/import
  networks:
    - linkwave-network
  labels:
    traefik.enable: true
    traefik.http.routers.keycloak.rule: Host(`keycloak.${ROOT_DOMAIN}`)
    traefik.http.routers.keycloak.entrypoints: websecure
    traefik.http.routers.keycloak.tls.certresolver: letsencrypt
    traefik.http.routers.keycloak.service: keycloak
    traefik.http.services.keycloak.loadbalancer.server.port: 8080

keycloak-db:
  image: postgres:15
  container_name: keycloak-db
  restart: unless-stopped
  ports:
    - "5008:5432"
  volumes:
    - postgres_keycloak_data:/var/lib/postgresql-keycloak/data
  environment:
    POSTGRES_DB: keycloak_db
    POSTGRES_USER: linkwave_user
    POSTGRES_PASSWORD: duyquang2002.
  networks:
    - linkwave-network

```

Figure 77. Docker compose file (keycloak and keycloak database)

```

kafdrop:
  image: obsidiandynamics/kafdrop
  container_name: kafdrop
  restart: unless-stopped
  ports:
    - "9000:9000"
  environment:
    KAFKA_BROKERCONNECT: "kafka:29092"
  depends_on:
    - "kafka"
  networks:
    - linkwave-network

kafka:
  image: obsidiandynamics/kafka
  container_name: kafka
  restart: unless-stopped
  ports:
    - "2181:2181"
    - "9092:9092"
  environment:
    KAFKA_LISTENERS: "INTERNAL://:29092,EXTERNAL://:9092"
    KAFKA_ADVERTISED_LISTENERS: "INTERNAL://kafka:29092,EXTERNAL://64.23.145.52:9092"
    KAFKA_LISTENER_SECURITY_PROTOCOL_MAP: "INTERNAL:PLAINTEXT,EXTERNAL:PLAINTEXT"
    KAFKA_INTER_BROKER_LISTENER_NAME: "INTERNAL"
    KAFKA_ZOOKEEPER_SESSION_TIMEOUT: "6000"
    KAFKA_RESTART_ATTEMPTS: "10"
    KAFKA_RESTART_DELAY: "5"
    ZOOKEEPER_AUTOPURGE_PURGE_INTERVAL: "0"
    KAFKA_CREATE_TOPICS: "linkwave:1:1"
  networks:
    - linkwave-network

volumes:
  postgres_linkwave_data:
  postgres_keycloak_data:

networks:
  linkwave-network:
    driver: bridge

```

Figure 78. Docker compose file (kafka)

```

ndquang@linkwave-tech:~/linkwave$ docker ps -a
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS
 NAMES
03423f765bed obsidian-dynamics/kafdrop "/kafdrop.sh" 19 hours ago Up 19 hours 0.0.0.0:9000->9000/tcp, :::9000->9000/tcp
7b090320e04c obsidian-dynamics/kafka "/bin/sh -c /opt/kaf..." 19 hours ago Up 19 hours 0.0.0.0:2101->2101/tcp, :::2101->2101/tcp, 0.0.0.0:9092->9092/tcp, :::9092->9092/tcp
e6f3a281f76c linkwave-backend "java -jar /app/link..." 19 hours ago Up 19 hours 8080/tcp
01b6add0d04c postgres:15 "docker-entrypoint.s..." 19 hours ago Up 19 hours 0.0.0.0:5005->5432/tcp, :::5005->5432/tcp
cff5f196c4fe linkwave-frontend "docker-entrypoint...." 19 hours ago Up 19 hours 80/tcp
6f4e7115b94e pgadmin4:latest "/entrypoint.sh" 2 months ago Up 2 months 80/tcp, 443/tcp
947a5afc7cc6 bitnami/keycloak:22.0.5 "/opt/bitnami/script..." 2 months ago Up 2 months
90d771dd1080 traefik:v2.5 "/entrypoint.sh --ap..." 2 months ago Up 2 months 0.0.0.0:80->80/tcp, :::80->80/tcp, 0.0.0.0:443->443/tcp, :::443->443/tcp, 0.0.0.0:8080->8080/tcp, :::8080->808
0/tcp
6989f0e630df0 postgres:15 "docker-entrypoint.s..." 2 months ago Up 2 months 0.0.0.0:5008->5432/tcp, :::5008->5432/tcp
keycloak-db

```

Figure 79. Docker logs

CHAPTER 6: TESTING AND EVALUATION

1. Testing

1.1 Functions

Table 32. Test case

No	Function	Action	Data Inputs	Expected Output	Actual Output	Result
1	Login	Valid credentials	Email address: ndquang5802@gmail.com, Password: 12345678	Successful login	Successful login	Pass
2		Invalid username	Email address: ndquang5, Password: 12345678	Display error message: “Please enter a valid email address”	Display error message: “Please enter a valid email address”	Pass
3		Unconfirmed email	Email address: ndquang@gmail.com, Password: 12345678	Show message system: “Invalid account. User probably hasn't	Show message system: “Please enter a valid email address”	

				verified email"		
4	Register	Valid registration	First Name: Nguyen, Last Name: Duy Quang Email address: quangndgc c200030@f pt.edu.vn, Password: 12345678, Confirm Password: 12345678	Successful registration	Successful registration	Pass
5		Weak password	First Name: Nguyen, Last Name: Quang Email address: ndquang58 02@gmail. com, Password: pass123, Confirm Password: pass123	Error message: "Please enter at least 8 characters."	Error message: "Please enter at least 8 characters."	Pass
6	Reset password	Valid email	Email: quangnguy	Email sent with password	Email sent with password	Pass

			en@gmail.com	reset instructions	reset instructions	
7	Chat	Send message	Type message: "Hello, what your name?"	Message sent successfully	Message sent successfully	Pass
8		Receive message		New message notification appears	New message notification appears	Pass
9	Friend request	Send request	Send request to Que Tran	Show system notifications "Friend request was sent successfully"	Show system notifications "Friend request was sent successfully"	Pass
10		Recall request	Recall sent request to Que Tran	Show system notifications "Friend request was deleted successfully"	Show system notifications "Friend request was deleted successfully"	Pass
11		Accept request	Accept friend request from Hung Dung	Show system notifications "Friend request was accepted successfully"	Show system notifications "Friend request was accepted successfully"	Pass
12		Remove friend	Remove Hung Dung from friend list	Show system notifications "Unfriended successfully"	Show system notifications "Unfriended successfully"	Pass

13	Manage profile	Update profile	Email: ndquang58 02@gmail. com First Name: Quang Last Name: Nguyen Gender: Male Birthday: 08/05/200 2 Address: Can Tho city Phone number: 091684336 7 Country: Vietnam Hobbies: Coding	Profile updated successfully	Profile updated successfully	Pass
14		Upload avatar picture	Choose file: profile.jpg	Display success message “Update avatar successfully”	Display success message “Update avatar successfully”	Pass
15	Manage Work &	Add new work	Company Name:	Display success message	Display success message	Pass

	Educatio n		Harvard University Position: Bachelor of Arts in Economics Description : Completed coursewor k in microecon omics, macroecon omics, and econometr ics. Start Date: 09/01/201 8 End Date: 05/15/202 2 Location: Cambridge, Massachus etts	"Add work experience successfully"	"Add work experience successfully"	
16		Update experience	Company Name: Harvard University Position: Bachelor of	Display success message "Update work experience successfully"	Display success message "Update work"	Pass

			<p>Arts in Economics</p> <p>Description :</p> <p>Completed course work in information technology, including database management, software development, and network administration.</p> <p>Start Date: 09/01/2018</p> <p>End Date: 05/15/2022</p> <p>Location: Cambridge, Massachusetts</p>		experience successfully”	
17		Delete experience	Clicking delete job experience	Display success message	Display success message	Pass

				“Delete work experience successfully”	“Delete work experience successfully”	
18	Add new education	Company Name: University of Greenwich Position: Bachelor of Science in Computer Science Description : Studied computer programming, algorithms, and software engineering principles. Start Date: 09/20/2020 End Date: 06/15/2024 Location: Greenwich, London,	Display success message “Add education experience successfully”	Display success message “Add education experience successfully”	Display success message “Add education experience successfully”	

			United Kingdom			
19	Do not fill in as updating education	Company Name: " " Position: Bachelor of Science in Computer Science Description : Studied computer programming, algorithms, and software engineering principles. Start Date: 09/20/2020 End Date: 06/15/2024 Location: Greenwich, London, United Kingdom	Display error message "Please fill in all fields"	Display error message "Please fill in all fields"		Pass

20		Post text content	Type text post content	Post created successfully	Post created successfully	Pass
21	Create new post	Post image content	Choose image: image.jpg	Image post created successfully	Image post created successfully	Pass
22		Post video content	Choose video: video.mp4	Video post created successfully	Video post created successfully	Pass
23		Valid password change	Current password: 12345678, New password: quang0508 , Confirm Password: quang0508	Password changed successfully	Password changed successfully	Pass
24	Change password	Invalid current password	Current password: invalidpass, New password: newpass123, Confirm Password: newpass123	Error message: "Incorrect current password"	Error message: "Incorrect current password"	Pass
25		Weak new password	Current password: pass123, New password:	Error message: "New password must be at	Error message: "New password must be at	Pass

			1234, Confirm Password: 1234	least 6 characters"	least 6 characters"	
26		Password mismatch	Current password: pass123, New password: newpass12 3, Confirm Password: newpass45 6	Error message: "Passwords do not match"	Error message: "Passwords do not match"	Pass
27	Manage notifications	Mark all as read	Clicking the "Mark all as read" button	All notifications marked as read	All notifications marked as read	Pass
28		View all notifications	Clicking the notification icon.	All notifications displayed	All notifications displayed	Pass
29		View all unread notifications	Clicking the "Unread" button	All unread notifications displayed	All unread notifications displayed	Pass
30		Remove a notification	Clicking the "Remove this notification" button	Display success message "Remove notification successfully"	Display success message "Remove notification successfully"	Pass
31		Mark notification as read	Clicking the "Mark as	The notification will change to	The notification will change	Pass

			read" button	be read status	to be read status	
32		Search for user	Search: "John"	User account matching query displayed	User account matching query displayed	Pass
33	Search	Search for post	Type search query: "technology"	Posts containing "technology" displayed	Posts containing "technology" displayed	Pass
34		Search for skill	Type search query: "programming"	Skills related to "programming" displayed	Skills related to "programming" displayed	Pass
35	Log out	Log out	Click on logout button	User logged out successfully	User logged out successfully	Pass

1.2 Performance

Below is a performance test image of the website through Lighthouse

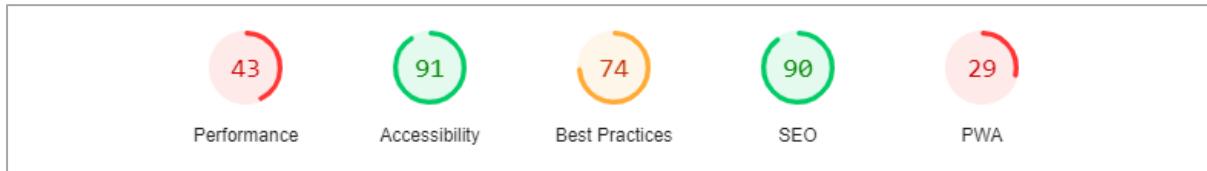


Figure 80. Analysis all performance

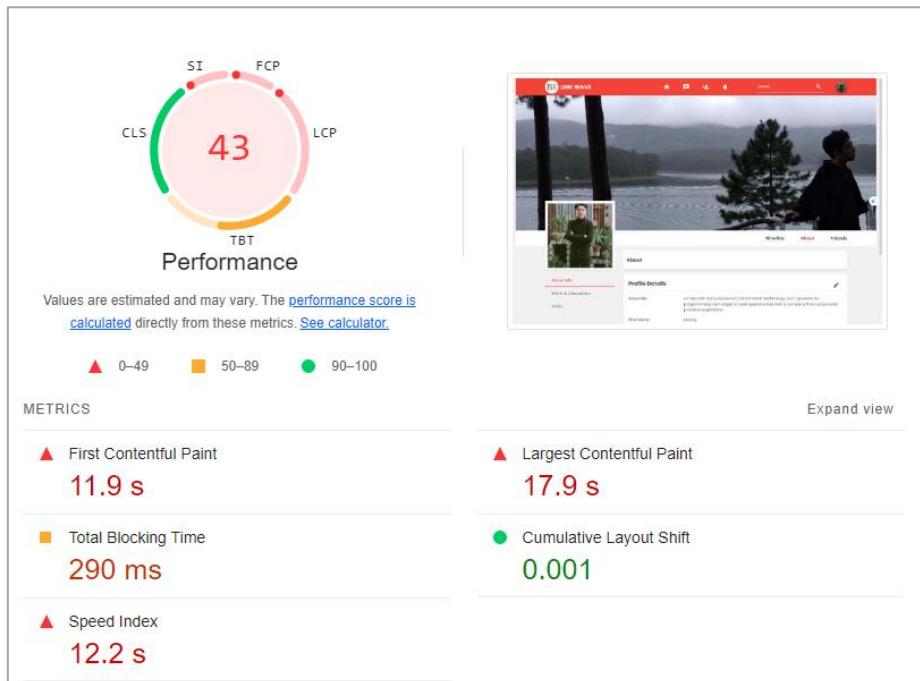


Figure 81. Testing performance

- Optimize images: Use modern, compressed formats like WebP.
- Minimize render-blocking resources: Inline critical CSS and defer non-critical JavaScript/CSS.
- Remove unused JavaScript/CSS: Use tools to identify and eliminate unnecessary code.
- Improve server response times: Optimize server or use a Content Delivery Network (CDN).

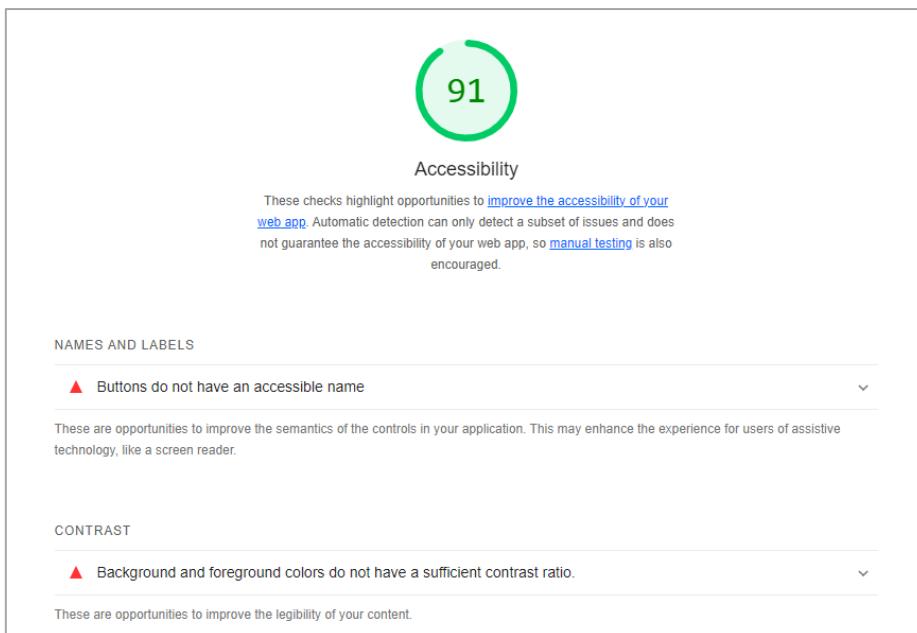


Figure 82. Testing accessibility

While the score is good, ensure all interactive elements are keyboard navigable and all images have alt text for screen readers. You can also manually check the site with assistive technology or consult WCAG guidelines for any further improvements.

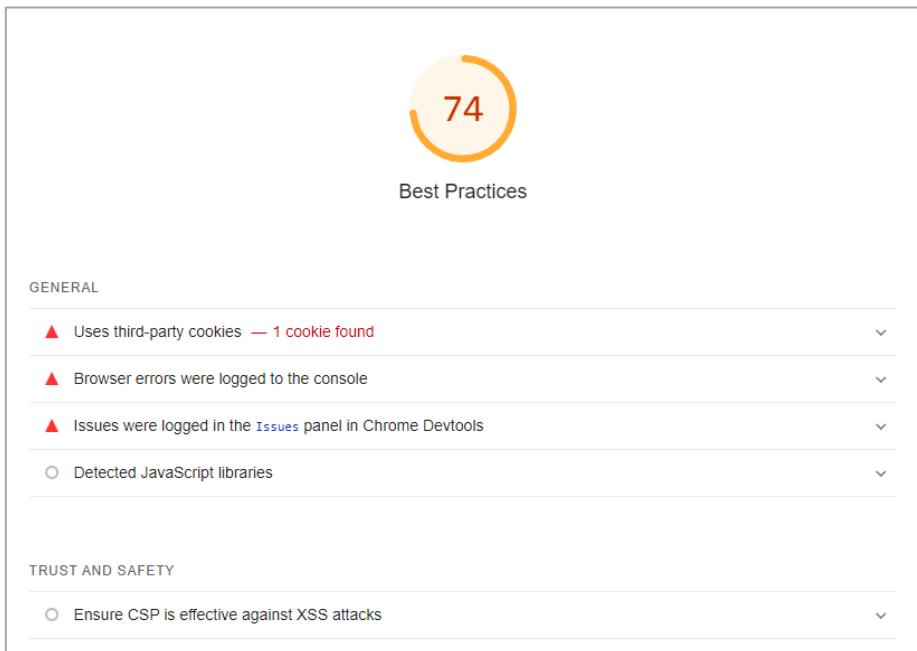


Figure 83. Testing best practice

- Enforce HTTPS to ensure data is encrypted in transit.
- Remove any deprecated APIs and replace them with modern alternatives.
- Ensure all external links use rel="noopener" to prevent security vulnerabilities.

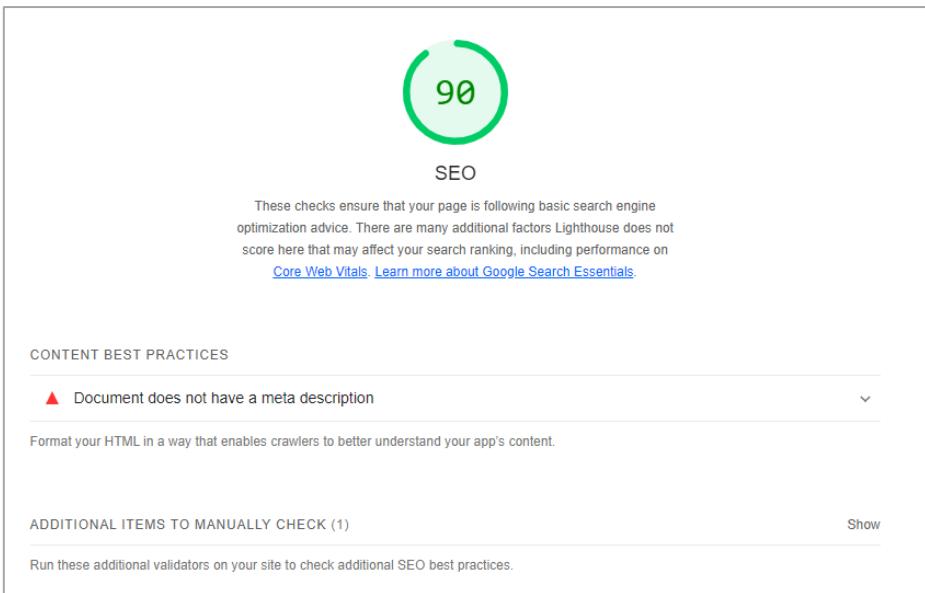


Figure 84. Testing SEO

Since this score is already high, look at the detailed recommendations from Lighthouse and address any small issues, such as meta tags or link texts that could be optimized further.

2. Evaluation

2.1 Self-evaluate

Based on my self-assessment, I find the user interface intuitive and user-friendly, with visually distinct icons and colors facilitating easy function input. The website adheres to security standards with HTTPS validation and employs encryption for user passwords. However, I recognize that there are vulnerabilities that need to be addressed before the site can be safely deployed for commercial use. In particular, certain areas are susceptible to exploitation by hackers.

Regarding performance, while the website is functional, its loading speed is subpar compared to contemporary standards. This impacts user experience and may deter potential users from engaging with the site efficiently.

To improve the website's security, I plan to implement additional layers of protection such as robust firewall systems, regular security audits, and intrusion detection mechanisms. Additionally, I will conduct thorough testing to identify and patch any existing vulnerabilities.

To enhance performance, I intend to optimize the website's code, compress images, and leverage caching techniques to reduce loading times. Furthermore, I will

explore options for scalable hosting solutions to accommodate increased traffic without compromising speed.

By addressing these concerns, I aim to elevate the website to a level where it meets both security best practices and performance expectations for commercial deployment.

2.2 Survey from users

The survey administered through Google Form involved 50 participants, encompassing 8 quantitative and 2 qualitative inquiries. Each question underwent a thorough evaluation to ascertain its depth and relevance. This report will provide a detailed analysis of each question and highlight the key takeaways from the survey. The survey results will provide valuable insights into the strengths and weaknesses of this social network project while allowing the developer to make informed decisions and prioritize improvements.

Question 1: Please rate your satisfaction with the LINK WAVE app experience on a scale of 1 to 5.

50 responses

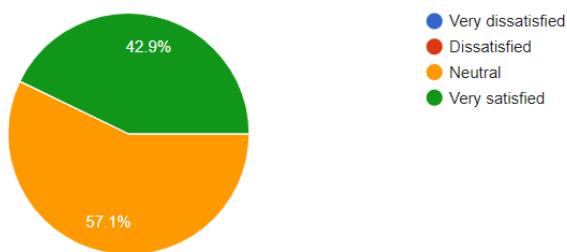


Figure 85. Question 1

The first question seeks to gauge participants' satisfaction levels with the LINK WAVE application, ranging from 1 (very dissatisfied) to 5 (very satisfied). Among the 50 responses, it's noteworthy that 42.9% expressed a neutral stance, while 57.1% indicated high satisfaction. This outcome suggests a predominantly positive sentiment towards the website throughout the user experience journey. However, particular attention should still be given to the neutral feedback from customers to enhance service quality and maximize user satisfaction.

Question 2: How many minutes do you use LINK WAVE daily?

50 responses

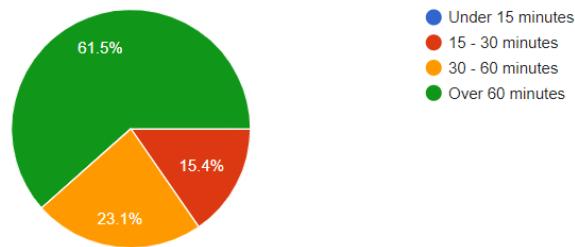


Figure 86. Question 2

The second question investigates participants' daily usage duration of the LINK WAVE app, segmented into four categories. Out of the 7 responses, the data indicates that 61.5% of users spend over 60 minutes daily, with smaller percentages allocating 15-30 minutes (15.4%) and 30-60 minutes (23.1%). The distribution of usage time suggests a considerable number of users invest significant time in the app, indicating its appeal and functionality. While a significant portion spends extensive time on the website, indicating its appeal and utility, the percentages allocated to 15-30 minutes and 30-60 minutes suggest areas for improvement. Implementing features such as user behavior tracking and personalized content recommendations could enhance engagement and attract more users in the future. This targeted approach ensures the platform remains relevant and adaptable to users' evolving needs and preferences.

Question 3: On a scale of 1 to 5, how reliable do you find the information and content shared on the LINK WAVE app?

50 responses

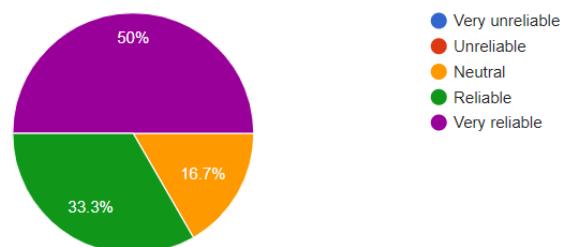


Figure 87. Question 3

The survey findings indicate a significant portion, 50%, of users perceive the content on the LINK WAVE app as "Very reliable," reflecting a considerable level of trust. Moreover, 33.3% of respondents find it "Reliable," reinforcing the perceived value of the platform's content. However, the presence of 16.7% neutrality

underscores the need to enhance content credibility. Implementing solutions such as integrating AI-driven content verification tools like ChatGPT could establish more robust content validation mechanisms. Leveraging such technology, the LINK WAVE website could enhance its ability to detect and filter out inaccurate or misleading information, thereby improving overall information reliability and credibility.

Question 4: On a scale of 1 to 5, please rate your satisfaction with the loading speed and overall performance of this website.

50 responses

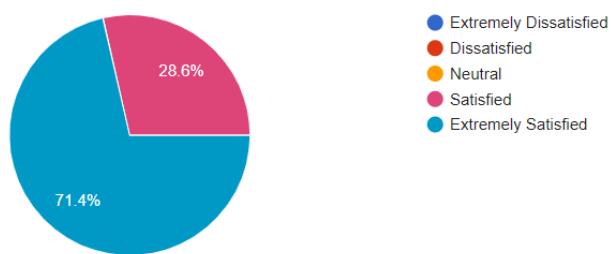


Figure 88. Question 4

The survey aimed to assess users' satisfaction with the loading speed and overall performance of the website on a scale from 1 to 5. Results show a significant 71.4% of respondents reported being "Extremely Satisfied," indicating a high level of contentment with the website's performance. Additionally, 28.6% expressed being "Satisfied." These findings suggest that the majority of users are pleased with the loading speed and overall performance. However, it's essential to continue monitoring and optimizing website performance to maintain user satisfaction.

Question 5: On a scale of 1 to 5, rate the user-friendliness of the LINK WAVE app interface.

50 responses

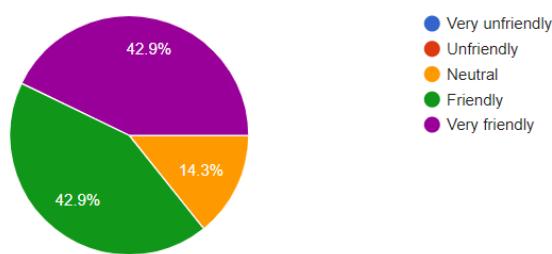


Figure 89. Question 5

The pie chart presents the results from a survey in which 50 respondents were asked to rate the user-friendliness of the LINK WAVE app interface on a scale from 1 to 5. The findings illustrate a positive reception, with a combined total of 85.8% of

participants perceiving the app as user-friendly; this includes 42.9% who rated it as "Very friendly" and another 42.9% who considered it "Friendly". A modest 14.3% of users reported a neutral stance, neither satisfied nor dissatisfied with the interface. Significantly, there were no negative ratings; none of the users found the app "Unfriendly" or "Very unfriendly," suggesting a generally favorable view of the app's user interface among the survey participants.

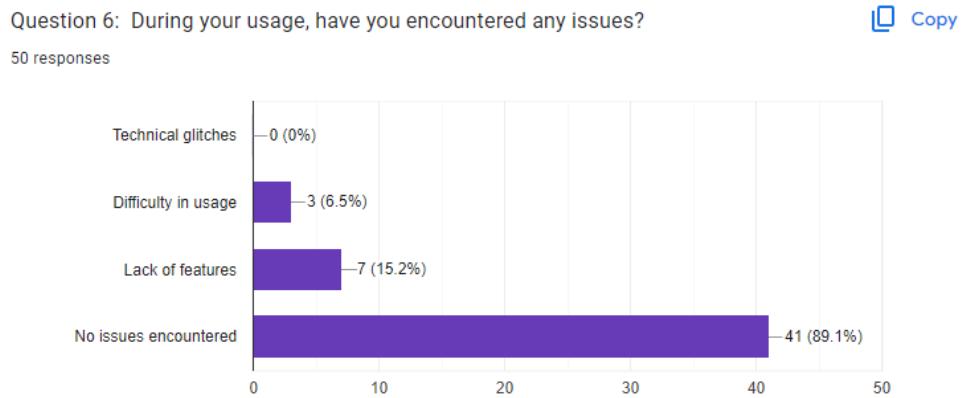


Figure 90. Question 6

The bar graph displays the feedback from 50 users concerning issues encountered while using a product or service. An overwhelming majority, 89.1% (41 respondents), reported no issues during their usage. A smaller fraction of users experienced problems, with 15.2% (7 respondents) indicating a lack of features and 6.5% (3 respondents) reporting difficulty in usage. Notably, there were no reported technical glitches among the respondents, highlighting a robust technical performance of the product or service in question. The data suggests that the user experience is largely positive, with only a few users encountering specific usability concerns or desiring additional features.

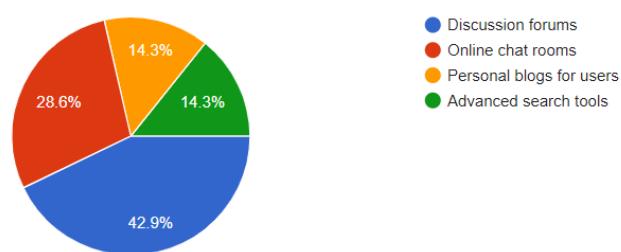
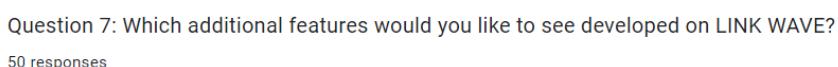


Figure 91. Question 7

The pie chart represents the preferences of 50 respondents regarding additional features they would like to see developed on LINK WAVE. The most requested feature is "Advanced search tools," accounting for 42.9% of the responses. "Discussion forums" are the next most popular, with 28.6% of users expressing interest in this feature. Both "Online chat rooms" and "Personal blogs for users" garnered an equal level of interest, each with 14.3% of respondents wanting these features. The data suggests that while there's a clear demand for enhanced search capabilities, there's also significant interest in more community-driven features such as forums and chat rooms, as well as personal expression through blogs.

Question 8: Would you recommend the LINK WAVE app to friends/colleagues?
50 responses

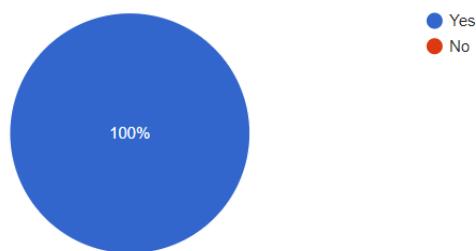


Figure 92. Question 8

The pie chart indicates a unanimous response from 50 participants to the question of whether they would recommend the LINK WAVE app to friends or colleagues. All respondents, which equates to 100%, answered "Yes," suggesting that the app has a highly favorable reputation among its users and they would endorse its use to others. This level of advocacy is indicative of strong user satisfaction and is a positive sign for the app's potential growth and popularity through word-of-mouth.

Question 9: What aspects of LINK WAVE do you find most appealing or valuable? Why?

50 responses

The aspect of LINK WAVE I find most appealing is its intuitive interface, making navigation effortless. Additionally, the diversity of content available keeps me engaged.

What I value most about LINK WAVE is its emphasis on fostering meaningful connections. The platform's focus on genuine interactions rather than superficial engagements sets it apart.

I appreciate LINK WAVE's commitment to user privacy and security, which instills confidence in sharing personal content. Its robust privacy settings ensure control over what information is shared.

The personalized content recommendations on LINK WAVE enhance my browsing experience, providing relevant updates tailored to my interests. Moreover, the platform's seamless integration of multimedia content adds value.

LINK WAVE's feature-rich environment, including discussion forums and personal blogs, offers diverse avenues for expression and interaction. Its inclusive community fosters open dialogue and collaboration.

Figure 93. Question 9

The user feedback on LINK WAVE underscores a variety of features that resonate positively with its audience. Users find the platform's intuitive interface particularly appealing, citing effortless navigation and a rich diversity of content that maintains engagement. There's a strong appreciation for the emphasis on fostering genuine, meaningful connections over superficial interactions, setting LINK WAVE apart as a community-centric platform. Equally important to users is the commitment to privacy and security, instilling confidence to share personal content securely, thanks to comprehensive privacy settings. The tailored content recommendations are also highlighted, enhancing the browsing experience with relevant and personalized updates, while the seamless integration of multimedia content enriches the overall value. Additionally, the platform's inclusive environment, complete with discussion forums and personal blogs, is celebrated for offering varied avenues for user expression and interaction, fostering an open and collaborative community spirit.

Question 10: How do you perceive the overall performance of LINK WAVE compared to other social networking platforms you use?

50 responses

As for its performance, LINK WAVE surpasses other platforms I use due to its reliability and seamless user experience."

Compared to other social networks, LINK WAVE stands out for its authenticity and community-driven approach.

In terms of performance, LINK WAVE's dedication to user safety makes it a preferred choice compared to other platforms.

Compared to other social networks, LINK WAVE excels in delivering curated content that resonates with users.

When compared to other platforms, LINK WAVE's versatility and inclusivity make it a standout choice for connecting with others.

Figure 94. Question 10

Responses to the performance of LINK WAVE compared to other social networking platforms reveal a consensus that positions it favorably. Users perceive LINK WAVE to surpass other platforms due to its reliability and seamless experience. It is distinguished for its authenticity and a community-driven approach that seems to resonate well with the user base. LINK WAVE's commitment to user safety is a significant factor in its preferred status among users, emphasizing a secure online environment. Additionally, the platform is praised for its ability to deliver curated content that aligns with user preferences, suggesting a personalized and engaging experience. Finally, its versatility and inclusivity are highlighted as key attributes, making LINK WAVE a comprehensive tool for social connection and interaction, further solidifying its place as a standout choice among social networks.

CHAPTER 7: CONCLUSIONS AND FUTUREWORK

1. Conclusion

Throughout this final year project, I delved into a plethora of new technologies, significantly broadening my skill set. Notably, I gained proficiency in implementing authentication and authorization functionalities using Keycloak, thereby bolstering security measures. Moreover, I acquired the know-how to deploy websites efficiently through Docker and other cutting-edge technologies. These newfound skills not only enhance my current capabilities but also lay a robust foundation for future endeavors.

Undoubtedly, the journey wasn't devoid of challenges. Yet, each obstacle served as a catalyst for growth, pushing me to delve deeper and expand my expertise. Through relentless self-research and problem-solving, I overcame hurdles and propelled the project forward.

2. Futurework

Moving forward, my focus will be on extending the usability of my system to mobile devices. Leveraging Flutter, I aim to craft features akin to those found in web applications, ensuring seamless access and functionality across various platforms. Additionally, optimizing algorithms will be paramount in enhancing user experience, prioritizing efficiency and responsiveness. By integrating these advancements, I strive to elevate the system's performance and accessibility, catering to the evolving needs of users and ensuring a superior experience across all devices.

REFERENCES

- Jamali, M. and Abolhassani, H. (2006) ‘Different aspects of social network analysis’, in. *2006 IEEE/WIC/ACM International Conference on Web Intelligence (WI 2006 Main Conference Proceedings)(WI’06)*, IEEE, pp. 66–72.
- Rath, M. (2022) ‘Application and Impact of Social Network in Modern Society’, in *Research Anthology on Usage, Identity, and Impact of Social Media on Society and Culture*. IGI Global, pp. 637–650.
- Zhang, J. and Philip, S.Y. (2018) ‘Cross-Platform Social Network Analysis.’
- altexsoft, 2022. *The Good and the Bad of Flutter App Development*. [Online] Available at: <https://www.altexsoft.com/blog/pros-and-cons-of-flutter-app-development/> [Accessed 20 April 2024].
- amazon, 2024. *What is Apache Kafka?*. [Online] Available at: <https://aws.amazon.com/what-is/apache-kafka/> [Accessed 20 April 2024].
- amazon, 2024. *What is Java?*. [Online] Available at: <https://aws.amazon.com/what-is/java/> [Accessed 19 April 2024].
- angular, 2024. *What is Angular?*. [Online] Available at: <https://angular.io/guide/what-is-angular> [Accessed 20 April 2024].
- Badkar, A., 2023. *What is Dart Programming? Everything You Need to Get Started!*. [Online] Available at: <https://www.simplilearn.com/what-is-dart-programming-article> [Accessed 19 April 2024].
- Bayoglu, M. B., 2023. *What Is Keycloak? What Is Behind Of The Keycloak? How Does Keycloak Work?*. [Online] Available at: <https://bayoglubatuhan.medium.com/what-is-keycloak-what-is-behind-of-the-keycloak-how-does-keycloak-work-d1be308d3227> [Accessed 20 April 2024].

- Bhanderi, P., 2023. *Flutter's Benefits and Drawbacks*. [Online] Available at: <https://medium.com/@parthbhanderi01/flutter-benefits-and-drawbacks-b268c1fe0f7f> [Accessed 20 April 2024].
- Brown, L. C., 2020. The Evolution of Online Communities for Technology Enthusiasts.. *International Journal of Computer Science and Technology*, 9(2), pp. 78-91.
- cloud, G., 2024. *PostgreSQL vs SQL Server: What are the key differences?*. [Online] Available at: <https://cloud.google.com/learn/postgresql-vs-sql> [Accessed 20 April 2024].
- DRUMOND, C., 2024. *What is scrum and how to get started*. [Online] Available at: <https://www.atlassian.com/agile/scrum> [Accessed 18 April 2024].
- Gillis, A. S., 2022. *DigitalOcean*. [Online] Available at: <https://www.techtarget.com/searchcloudcomputing/definition/DigitalOcean> [Accessed 20 April 2024].
- java, 2024. *What is Java technology and why do I need it?*. [Online] Available at: https://www.java.com/en/download/help/whatis_java.html [Accessed 19 April 2024].
- javatpoint, 2021. *Cloudinary*. [Online] Available at: <https://www.javatpoint.com/cloudinary> [Accessed 28 April 2024].
- Jayeshwarke, 2023. *Getting Started with Spring Boot: Advantages, Disadvantages, and Use Cases*. [Online] Available at: <https://medium.com/@jayeshwarke011/getting-started-with-spring-boot-advantages-disadvantages-and-use-cases-497b0f04fb86#:~:text=In%20conclusion%2C%20Spring%20Boot%20simplifies,and%20customization%20can%20be%20challenging> [Accessed 20 April 2024].
- Johnson, M. S., 2018. Building Successful Online Communities for Technology Experts.. *Technology and Innovation*, 15(2), pp. 73-88.

- Josh, 2024. *Native vs Cross-Platform: Navigating Social Media App Development*. [Online]
- Available at: <https://www.developers.dev/tech-talk/social-media-app-development-native-vs-cross-platform.html>
- [Accessed 18 April 2024].
- Nek, D., 2024. *What is Traefik? How it Works, Features, Functions, and Benefits Explained*. [Online]
- Available at: <https://webhostinggeeks.com/blog/traefik-features-functions-benefits/>
- [Accessed 28 April 2024].
- papertrail, 2024. *What Is DigitalOcean?*. [Online]
- Available at: <https://www.papertrail.com/solution/guides/digitalocean/>
- [Accessed 20 April 2024].
- Peterson, R., 2024. *What is PostgreSQL? Introduction, Advantages & Disadvantages*. [Online]
- Available at: <https://www.guru99.com/introduction-postgresql.html>
- [Accessed 20 April 2024].
- Rafalski, K., 2023. *Pros and Cons of Java Development in 2023*. [Online]
- Available at: <https://www.netguru.com/blog/java-pros-and-cons#:~:text=Java%20is%20a%20versatile%20and,costs%2C%20and%20GUI%20development%20difficulties.>
- [Accessed 19 April 2023].
- Siddharth, 2021. *Angular: Pros and Cons*. [Online]
- Available at: <https://dev.to/siddharthshyniben/angular-pros-and-cons-m9l>
- [Accessed 20 April 2024].
- Singh, R., Kumar, D. & Sagar, B. B., 2019. *Analytical Study of Agile Methodology in Information Technology Sector*. Mathura, India, IEEE.
- Smith, J. A., 2021. Social Networking and Professional Development: The Role of Technology Experts.. *Journal of Technology Advancements*, 12(3), pp. 45-58.
- typescriptlang, 2024. *TypeScript for JavaScript Programmers*. [Online]
- Available at: <https://www.typescriptlang.org/docs/handbook/typescript-in-5->

minutes.html

[Accessed 19 April 2024].

Williams, R. D., 2019. Networking in the Digital Age: Strategies for Technology Professionals.. *Journal of Information Technology Networking*, 7(4), pp. 121-136.

Zaitsev, P., 2023. *What is TypeScript? Benefits of TypeScript and Disadvantages*. [Online]

Available at: <https://itjet.io/blog/what-is-typescript>

[Accessed 19 April 2024].

Zola, A., 2022. *Bootstrap*. [Online]

Available at: <https://www.techtarget.com/whatis/definition/bootstrap>

[Accessed 20 April 2024].

APPENDIX

- **Source code:**

Front end	https://github.com/MyPersonalApplication/linkwave-website
Back end	https://github.com/MyPersonalApplication/linkwave

- **Link demo:** <https://linkwave.tech/>

- **Account test:**

Roles	Email	Password
Admin	admin@linkwave.com	Admin@2024
User	ndquang5802@gmail.com	12345678