

Lemon Social Network

Bianca Antoci

Supervisor: Renuga Jayakumar

*Project submitted as part of the requirements
for the award of BSc Cloud Computing*

18.03.2022

Declaration of Originality

I, Bianca Antoci, declare that I am the sole author of this Project; that all references cited have been consulted; that I have conducted all work of which this is a record, and that the finished work lies within the prescribed word limits.

This work has not previously been accepted as part of any other degree submission.

Signed: Bianca Antoci.....

Date: 18.03.2022.....

FORM OF CONSENT

I, Bianca Antoci hereby consent that my Project, submitted in candidature for the B.Sc.(Hons) Cloud Computing degree, if successful, may be made available for inter-library loan or photocopying (subject to the law of copyright), and that the title and abstract may be made available to outside organisations.

Signed: Bianca Antoci.....

Date: 18.03.2022.....

Copyright Acknowledgement

I acknowledge that the copyright of this project report, and any product developed as part of the project, belong to the University of Wales Trinity Saint David, Swansea.

ACKNOWLEDGEMENTS

First and foremost, I'd like to express my gratitude to Renuga Jayakumar, my supervisor, for her support and guidance during this process. Her advice got me through some tough times throughout my study and led me in the right direction..

Second, I'd like to express my gratitude to my brother Adrian for his willingness, support, and guidance during these academic years, both in my professional and personal life.

I am also appreciative of my family's support throughout this time and throughout my life. Everything I have today would have been a dream if it hadn't been for them.

Abstract

Social networks have evolved since their inception in the mid 2000's and different types have emerged as a result. Therefore, this study represents the research on the implementation of a new type of social network focused on healthy lifestyle. The researcher sought to explore what are the main building blocks when developing a software platform, the technological changes and the impact of the platform on its users.

In order to better understand how a software platform like a social network is implemented and hosted, multiple types of Cloud Providers and software development frameworks were carefully analysed. Starting from frontend frameworks used to build the user experience and interfaces, to backend solutions capable of automatically scaling to millions of users.

Lemon Social Network, the proof of concept platform presented in this study, aims at connecting people locally by sharing diets, recipients and products related to a healthy lifestyle.

The findings of this research show that users have a preference towards certain software platforms that better represent their personal life, community and culture. Therefore a new platform like Lemon has a big potential to change users' habits as there is a rapidly increasing problem with obesity all around the world. Lemon is hosted in Google Cloud, one of the biggest host providers and developed using Angular, one of the most popular coding frameworks in 2021. The UX was a particular focus, and the latest trends like Material and Flat Design languages were analysed and used.

Table of content

ACKNOWLEDGEMENTS	2
Abstract	3
CHAPTER 1 – INTRODUCTION	6
1.1 Introduction	6
1.2 Aim	6
1.3 Objectives	6
1.4 Structure of the project	7
2.1 Introduction	8
2.2 Background	8
2.3 Social Network and Food Behaviour	9
2.4 Social Network Analysis	10
2.5 Analysis of Tools required	11
2.5.1 Server-side vs Client-side rendered solutions	12
2.5.2 Frontend frameworks	13
2.5.3 JQuery	14
2.5.4 AngularJS	15
2.5.5 React	15
2.5.7 Firebase	18
2.6 Software Development Life Cycle	19
2.6.2 UX Research	20
2.6.3 Coding phase	21
2.6.4 Testing	21
2.6.5 Releasing in production	22
CHAPTER 3 – ANALYSIS	23
3.1 UML Diagrams- use case	26
3.2 System Requirements	27
3.3 Risk analysis	30
CHAPTER 4 - DESIGN	31
4.1 Introduction	31
4.2 Design	32
Chapter 5 - IMPLEMENTATION	35

Figure 5.10 - Backend implementation	44
CHAPTER 6 - TESTING	45
CHAPTER 7 - EVALUATION	47
7.1 Personal Evaluation	47
7.2 System Evaluation	48
CHAPTER 8 - FURTHER ENHANCEMENTS	49
CHAPTER 9 – CONCLUSION	51
CHAPTER 10 – REFERENCES	52
APPENDICES	56

CHAPTER 1 – INTRODUCTION

1.1 Introduction

Over the last decade, technology has advanced to create better, faster, and more effective platforms for people to interact and share information with others. The growing popularity of these networks, as well as the vast amount of material available in the form of text, photos, and videos, opens up new possibilities for data analytics in the context of social networks.

A social network is a platform that brings individuals together who share the same interests and allows them to share text, photos and videos.

Social networks have become an important part of the modern lifestyle. It allows people to connect, share and interact in a virtual environment [1]. They provide a sense of belonging, stability, and a sense of community in which people may share their worries and wants while also supporting others.

The world is now dominated by social networks, they may be a great tool for encouraging people to eat healthier and move more, especially young people. A social network may significantly impact health-related attitudes and behaviours in all modern civilizations.

It's difficult to navigate online these days without seeing photos and videos of food, recipes, or messages from family and friends reporting what they've eaten or are about to eat. The vast amount of information available on social networks may cause consumers to lose trust or become confused. As a result, people seek food guidance from online communities and consumer organisations [2].

1.2 Aim

The aim of this project is to create a new social network to help individuals to start a healthy lifestyle.

1.3 Objectives

A few objectives will be followed in order to achieve the project's goal.
The objectives are:

1. Research on competitors in the social network health lifestyle market;
2. To analyse necessary technical frameworks;

3. Research target audience;
4. Uplevel the technical skills required;
5. Research technologies for frontend and backend implementation;
6. Implement a UX design;
7. Conduct end to end testing of the platform.

1.4 Structure of the project

This report is made by nine-chapter in total:

1. **Chapter 1 - Introduction:** This chapter gives a rundown of the report. It diagrams the project's objectives and underlines the goals that must be met in order for the extent to succeed.
2. **Chapter 2 – Research and review of literature:** This chapter focuses on social network and food behaviour studies and literature. Background knowledge is necessary for comprehending and developing this project.
3. **Chapter 3 – Analysis:** This chapter includes the functionality of Lemon Social Network.
4. **Chapter 4 – Design:** This chapter explains how the UX design was developed.
5. **Chapter 5 - Implementation:** This chapter describes the process of coding and releasing a software product.
6. **Chapter 6 – Evaluation:** This chapter assesses the extent and gives a conclusion from the information collected.
7. **Chapter 7 – Further Enhancement:** This chapter incorporates an individual reflection of the creator based on the encounter of this venture and what has been learnt.
8. **Chapter 8 – Conclusions:** This chapter abridges what was accomplished within the venture and in the event that any encouraging inquiry is required.
9. **Chapter 9 – References:** Records all references used.

CHAPTER 2 - RESEARCH & REVIEW OF LITERATURES

2.1 Introduction

Healthy food consumption is an essential component of long-term development. Consumers are faced with a multitude of options when it comes to living a sustainable food lifestyle. Food is a necessary part of everyone's daily routine. As a result, from a planning and research aspect, it's critical to look at what people eat and how they feel about certain foods. Unfortunate eating is an issue in numerous countries, and in case national governments need to form activity plans to empower sound eating, they'll require a total get a handle on buyer attitudes, encounters, and practices when it comes to nutritious meals. Social networks analysis is one strategy for getting such data. Numerous people's everyday lives are presently ruled by social systems. Clients are investing more time on these stages and taking off dynamic and inactive advanced impressions as a result of their intuitiveness with other clients. These data offer a lot of study potential in a lot of areas, especially since knowing people's social media communication is important in order to understand their views, experiences, actions, and beliefs. [3]

2.2 Background

Rural adults have high rates of obesity and poor adherence to national dietary requirements, so understanding their food choices is important. Obesity and overweight have evolved into a global issue that is rapidly expanding. Food settings and dietary patterns are integrally linked to obesity, and food is reflected in our way of life, culture, and well-being. As a result of these factors, there is a growing interest in public health. Healthy eating is a key factor of good health, but poor eating can put your health in danger. Understanding the factors that influence consumer food choices and eating habits is critical to addressing this problem. The social environment, which includes friends and the shopping environment, has a significant impact on their normal meal choosing process. In addition, two major research found that social media had a significant influence on customer buying behaviour [4].

Social networks have evolved into spaces where individuals may produce and share information on their own. By 2025, it is estimated that half of the world's population will utilise social networks. Given the increased use of social media among young adults, health researchers are searching for methods to use this involvement to provide treatments and health promotion programmes. Many research on the use of social networks and health concerns are centred on the worldwide problem of obesity and health problems. because social media has the ability to share information about healthy practices. Based on the aforementioned examination of food-related communication on social networks, we may be able to obtain a better understanding of consumer behaviours and attitudes about healthy eating [5].

2.3 Social Network and Food Behaviour

Food production and distribution make for around a third of all household environmental impact worldwide. Consumers play a critical part in the transition to more environmentally friendly foods and diets. Our social environment is one component that influences what and why we eat. The use of social networks has been found to alter how people look for and choose products and services. Social media, particularly social networks, are rapidly becoming important sources of information and communication, even in the realm of food. Social media is also used to expand food options, gather recipes, provide a platform for adults to promote food they've made or eaten, and, in some circumstances, to confuse adults about which foods to eat [6]. It's difficult to navigate through social media these days without seeing photos and videos of food, recipes, or messages from family and friends reporting what they've eaten or are about to eat. The vast amount of information available on social media may cause consumers to lose trust or get confused. As a result, people seek food guidance from online communities and consumer organisations. It's fascinating to learn how one's eating habits are affected by online food exposure. Some research implies that social media-based perceptions of real consumption may influence the consumption of low- and high-energy foods and beverages.

Social networks and social support are two social instruments that have been connected to the appropriation of sound practices within the common population. Social support is defined as "any process by which social relationships could promote health and well-being," while social networks are defined as "the web of social contacts that surround an individual and the features of those links." By offering social support, facilitating access to resources, and establishing social norms, social networks can impact health habits. [7].

Social support, like social networks, can influence how people receive healthy lifestyle information and how they perceive normative health behaviours. Social eating norms, eating habits of social network members, and perceived general social support from family and friends have all been proven to influence an individual's meal choices. As a result, social networks may have an impact on what people purchase and eat, and they can be utilised as a policy tool to enhance food literacy and promote healthy eating and a better lifestyle in general.

2.4 Social Network Analysis

We live in the era of social networks, which link and coordinate social players across the world, such as individuals and companies. The first social network was Six Degrees developed in 1997 by Andrew Weinreich and was based on the Web of Contacts model of social networking.

A social network could be a site that brings together individuals who share the same interface and permits them to trade data, photos, and recordings where users spend a part of time, clearing out dynamic and detached advanced impressions as a result of their interactions [8]. They are web-based systems that allow users to create public profiles, generate a list of persons with whom they may exchange connections, and observe and cross relationships inside the system.

Frequently people confuse social networks with social media but there is a distinction between them. In the broad sense, social media is mainly used to broadcast information, for example, an online news site or a blog [9]. On the other hand, a social network allows users to interact with them as well as broadcast information. Often Social networks and social media are interrelated concepts. The Twitter platform, for example, is a social medium based on users' social networks. Users of the Twitter network create communications known as "tweets." In order for Twitter to function as a social medium where millions of users can share content, it must first establish a network of users based on reciprocal relationships, which is the essence of a social network [10] .

Users feel more comfortable communicating in their trusted social networks, and because they are already familiar with the platforms, it is easier to integrate into their daily routine. Because people are more comfortable communicating in their trusted networks and because they are already using the sites, social network platforms like Facebook, Twitter, and Instagram perform better than digital media platforms designed particularly for the research project (i.e., website, app) [11].

There are several groups, pages, and profiles on the Internet that discuss diets, recipes, and health. Facebook is one of the most popular social networks with a large number of groups. The people in this group are likely to face similar challenges, successes, and

questions. Instagram, on the other hand, is flooded with accounts offering to help with weight loss, eat healthier, and improve health .

Table 2.1 - Differences in main features and functions among key Social Networks platforms

Social networking sites	Main feature/function
 Twitter	Post short messages, containing links to articles, videos and photos; Use of # (hashtags).
 Facebook	Post long-form content like video, article, photo; Create groups regarding different topics; Add 'friends', comment, like and share content.
 Instagram	Post photo and short video; Like and comment posts; Share others content; Use of # (hashtags) searchable by topics.
 TikTok	Post a short video; Like and comment on the videos.

2.5 Analysis of Tools required

The nature of the platform, a social network, requires a detailed investigation of multiple programming languages, frontend frameworks and backend solutions. The research was divided based on the platform architecture:

- Server-side vs Client-side rendered solutions;
- Frontend frameworks (the part responsible for all the UX and UI elements);
- Backend solutions (the part responsible for hosting the data and functional platform logic).

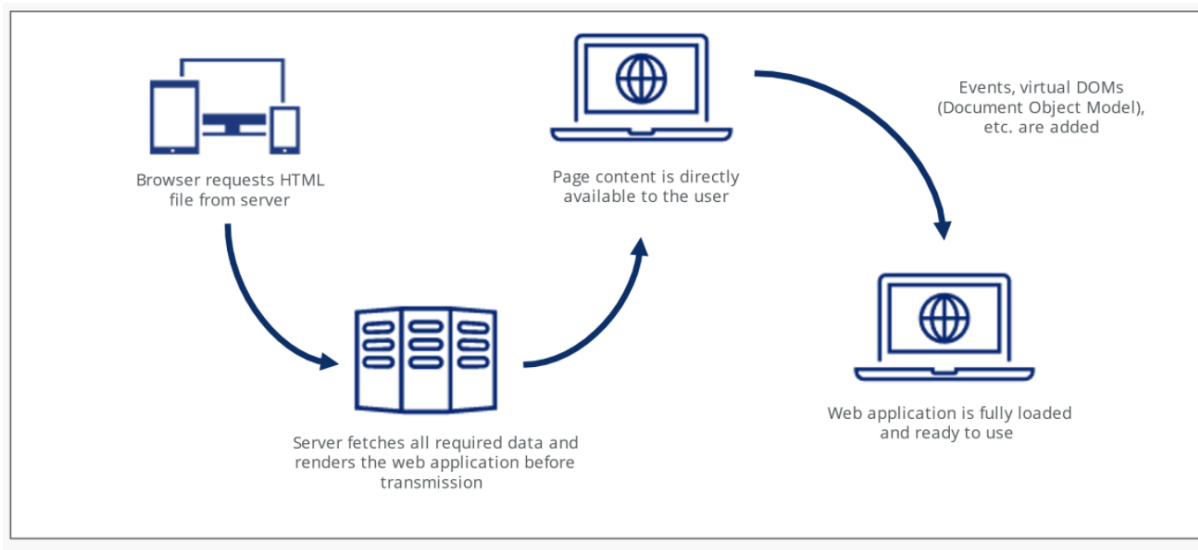
2.5.1 Server-side vs Client-side rendered solutions

One of the fundamental questions that needed to be answered before proceeding in choosing a frontend framework is if the web application should be a 'client-side' or a 'server side' rendered.

A server-side rendered solution requires the server to generate the HTML in real-time as requested by the client. This is a very compute-intensive process as it requires both server-side and client-side logic to be computed at the same time by the server. For example, on a page where we display a list of products, the server will have to query the list of products, compile an HTML page and return that to the client, all in one go [12].

Server-side rendering is considered now obsolete and most of the big tech companies use a different approach called 'client-side rendering'. The transition began to happen around 2010.

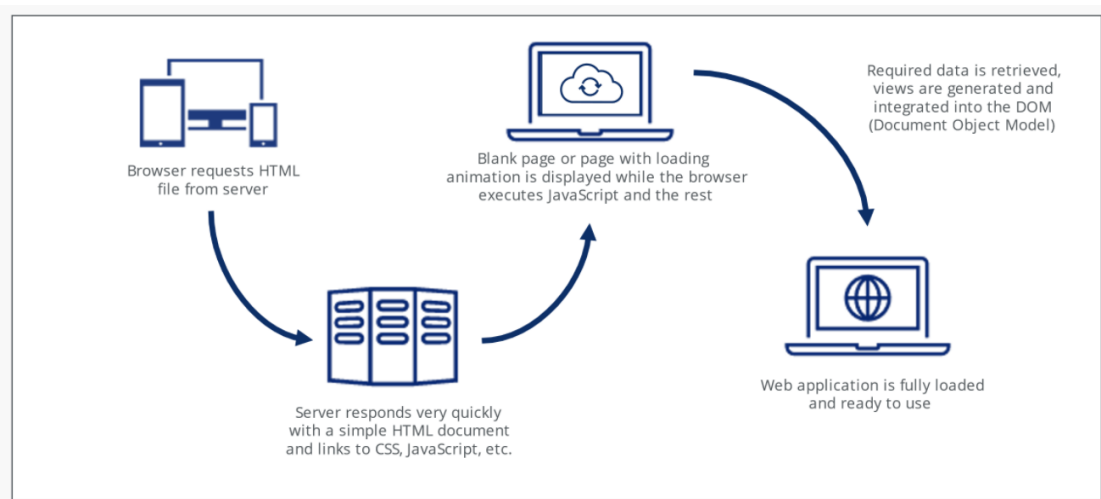
Figure 2.1 - Server-side rendering [13]



Client-side rendering allows the server-side and client-side solutions to be separated. This is a newer approach where the client first requests the code used to generate the website and only makes HTTP requests to retrieve data. This lowers the compute needed by the server as the actual HTML is generated in the user's browser and not on the server. For example, on a page where we want to display the list of products the client makes an HTTP call to return the list of products as JSON and it generates the HTML on the client-side [14].

Client-side rendering is now the norm used by Facebook, Google, Microsoft and most of the other tech giants.

Figure 2.2 - Client-side rendering [13]



2.5.2 Frontend frameworks

A software framework is a development architecture for software applications that was established to make the process of web development easier. It has a number of application programming interfaces (APIs) for getting access to underlying data resources, as well as templating capabilities for displaying data in a browser. It also provides an environment for scripting how information flows. The majority of frameworks also provide tools to help web developers create a content management system (CMS) for managing digital material on websites and the Internet [15].

A web application is a pre-built structure that handles the most repetitive operations and elements of website development. This means that web developers will spend the

majority of their time using code to interface with the various pieces of the web framework.

The most significant advantages for web developers that utilise a web framework are that it is open-source, efficient, has a high level of support, is secure, and offers an integration function.

The efficiency of web developers is another key benefit. This is because web frameworks reduce the amount of repetitious code that developers must write, allowing them to construct websites and apps faster.

A frontend framework is now a crucial component in building a modern web application. Back in the 90' and 20' websites were much simpler and only required simple HTML and JS code. Today this is no longer the case as the websites become more complex and very close to a native experience on most desktop and mobile devices [16].

Some of the most popular front-end frameworks today are:

- JQuery (a JS library used to manipulate HTML);
- Angular (a powerful and mature front-end framework from Google);
- React (a powerful and mature frontend framework from Google).

2.5.3 JQuery



JQuery is a JavaScript-based framework. JQuery was first released in 2006 and is now used by around 80% of all the websites in the world. It allows web developers to add new features to their sites. It's the most used JavaScript library for traversing and manipulating the HTML Document Object Model (DOM). With an easy-to-use API that works in a range of browsers, it's a lot easier. It improves user experience by making your web pages more dynamic and easy. It has a number of aesthetic features, including ease-in and ease-out effects, that make it one of the most popular options among web designers [17].

Most of the other frontend frameworks incorporate jQuery, for example, Angular.

Over the last several years, jQuery has suffered a dramatic drop in popularity. With the development of front-end JavaScript frameworks such as Angular, Vue, and React, jQuery's odd syntax and often-overwrought implementation have fallen behind this new wave of online technology. jQuery, on the other hand, has its uses. It's been used in a variety of applications, from large e-commerce platforms to basic landing sites, for starters [18]. Second, if you don't know CSS, jQuery may still be useful for things like quick prototyping and even animation. jQuery may be old, but it is still used.

2.5.4 AngularJS



AngularJS was first released in 2010 and became very popular in a very short time. In 2016 a new version was released called 'Angular' that was a complete rewrite by the same team at Google.

AngularJS is a web application framework that allows you to create dynamic web apps. It enables you to use HTML as your template language and extend HTML's syntax to represent your application's components easily and succinctly [19]. AngularJS's data binding and dependency injection eliminate a lot of the code you'd otherwise have to write. It's also a wonderful complement for any server technology because everything happens in the browser [17].

AngularJS, on the other hand, adopts a different approach. By adding new HTML components, it tries to reduce the impedance mismatch between document-centric HTML and what an application requires. AngularJS uses a feature called directives to teach the browser new syntax.

Angular uses TypeScript as its programming language and uses "Components" and "Modules" to structure the code. TypeScript is a strongly typed language, which allows developers more familiar with Java or C# to see a familiar syntax [20].

2.5.5 React



React is an open-source JavaScript toolkit for making client interfacing for single-page apps. It's used to oversee the view layer in web and versatile apps. With React, we will also make reusable UI components. React was made by Jordan Walke, a Facebook computer program designer. React to begin with showed up on Facebook's newsfeed in 2011, taken after by Instagram.com in 2012 [21].

React permits designers to form expansive web apps that can alter data without reviving the page. The main reason for React is to be rapid, adaptable, and basic to use. It works on the client interfacing of the application. This must do with the view of the MVC layout. It can be used with other JavaScript libraries or systems in MVC ventures, such as AngularJS . [18].

React uses "Components" and "Props" to handle the interaction between UI and logic. The programming language used is native JavaScript, the interpreted programming language created in the 90' [22].

2.5.6 Backend solutions

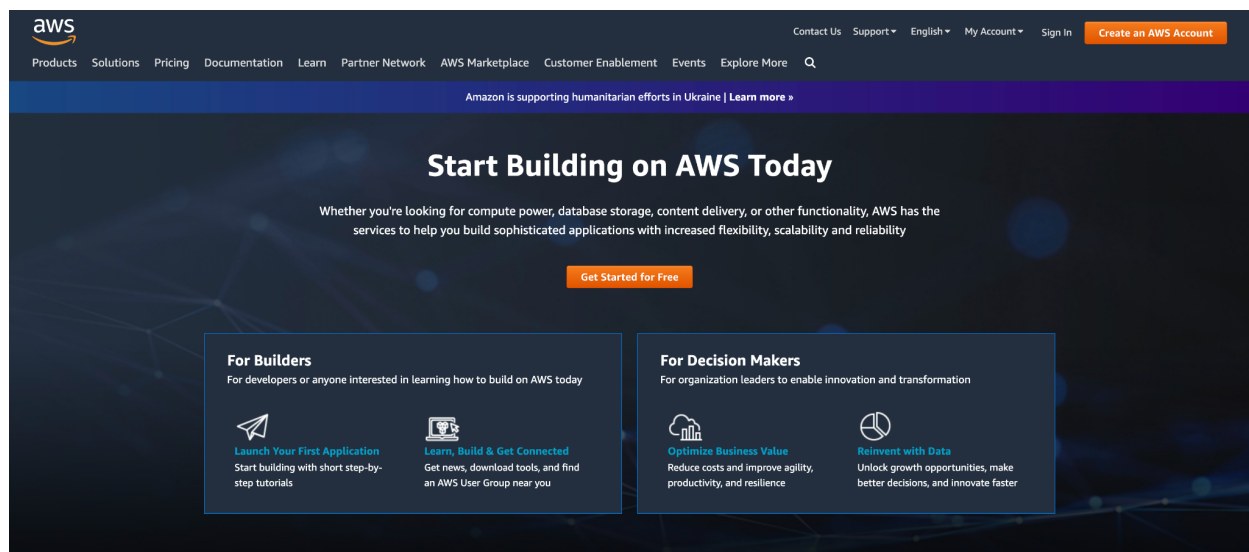
The backend of a web or mobile application is the software that manages the data. It contains all of the functionality necessary to access and handle data that ordinary users are unable to access. It's commonly referred to as the component of an app that the user can't see, and it works hand-in-hand with the frontend to produce the final product.

Any modern platform requires a scalable backend solution. Having in house hardware is no longer a viable option as most of the Cloud-based providers provide a cheap and reliable source of computing and storage [23].

One way to evaluate a Cloud provider is based on their features and costs:

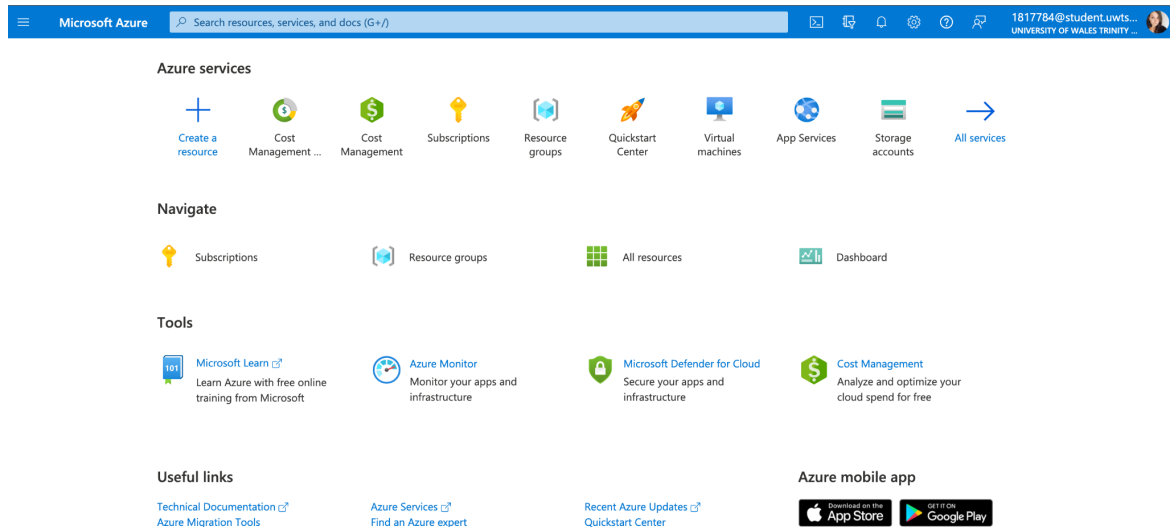
1. Amazon Web Services (the biggest Cloud provider, offers a large selection of hosting solutions for most of the backend frameworks and databases).

Figure - 2.3 Amazon Web Services (43)



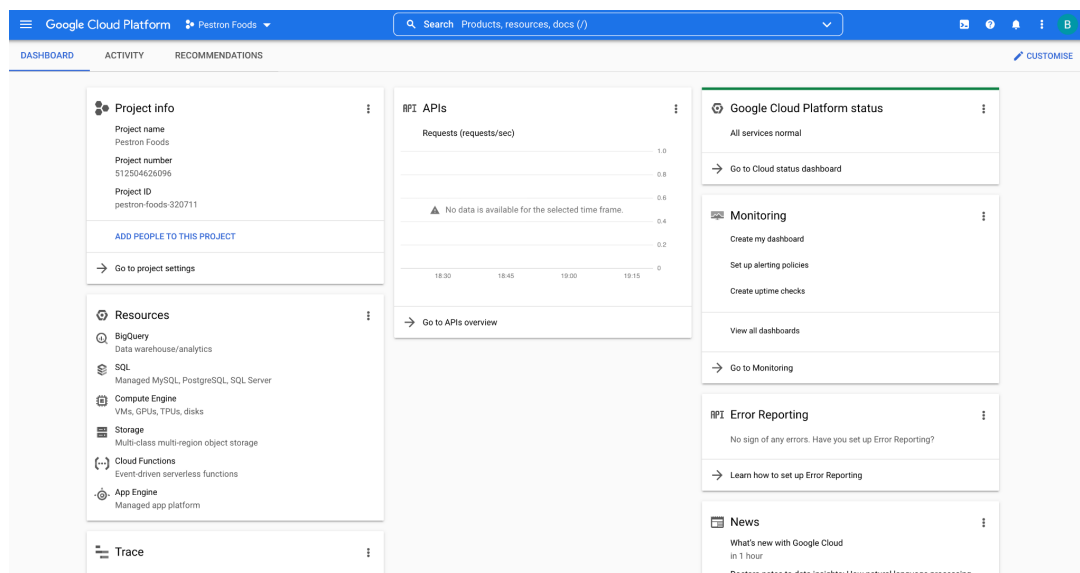
2. Microsoft Azure (the second-largest provider, created by Microsoft, offers mainly solutions around .NET but not only).

Figure - 2.4 Amazon Web Services



3. Google Cloud (the third-largest provider, offers a very flexible environment for serverless applications, like FireBase)

Figure - 2.5 Google Cloud Platform



For a social network, a serverless solution like Firebase is recommended and a NoSQL Database like Firebase Firestore is a great solution as it offers linear query speeds independent from the Database size. [24]

To secure the platform social signin can be used like Sign in with Google, Facebook and Instagram.

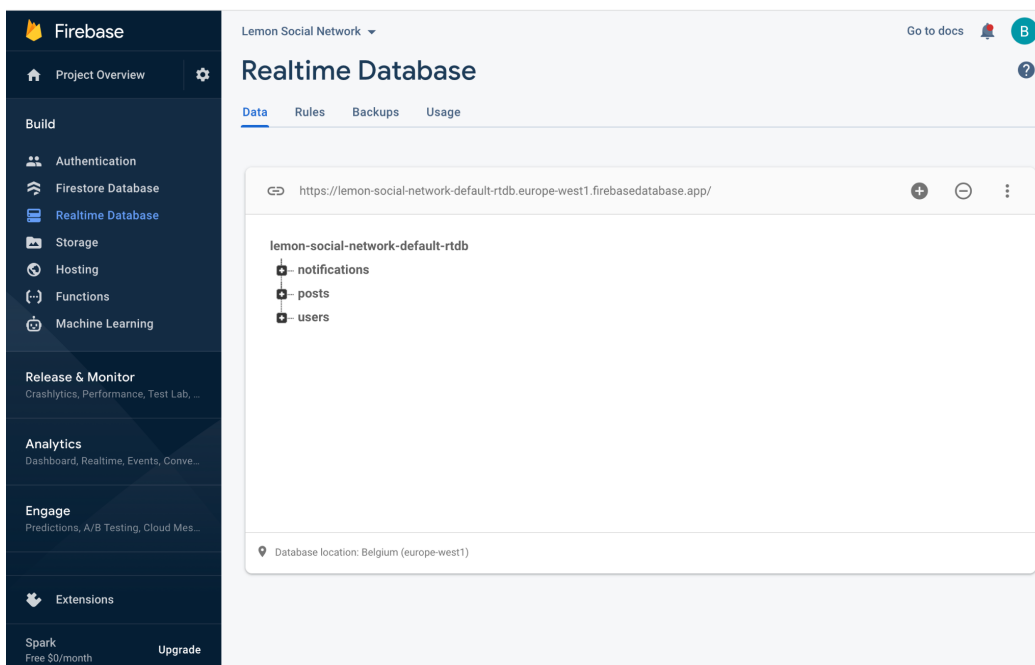
2.5.7 Firebase

The Firebase Realtime Database is a NoSQL cloud-based database with offline functionality that syncs data in real time across all clients. In the Realtime database, data is kept as JSON, and all connected clients share a single instance, which receives automatic updates with the most recent data.

Firebase is a set of devices that permits clients to "build, improve, and grow your app," and the devices it offers cover a part of the administrations that engineers would regularly need to construct themselves but do not since they'd or maybe focus on the app encounter. This incorporates analytics, verification, databases, arrangement, record capacity, thrust informing, and so on. The administrations are cloud-based and versatile, requiring small to no exertion on the portion of the developers.

Traditional app development, on the other hand, as often as possible requires composing both frontend and backend code. The frontend code essentially calls API endpoints that the backend may access, taking off the difficult lifting to the backend code. With Firebase items, the conventional backend is bypassed, putting the work on the client. Each of these items has authoritative access through the Firebase console [25].

Figure 2.6 - Firebase (Lemon Social Network Database)



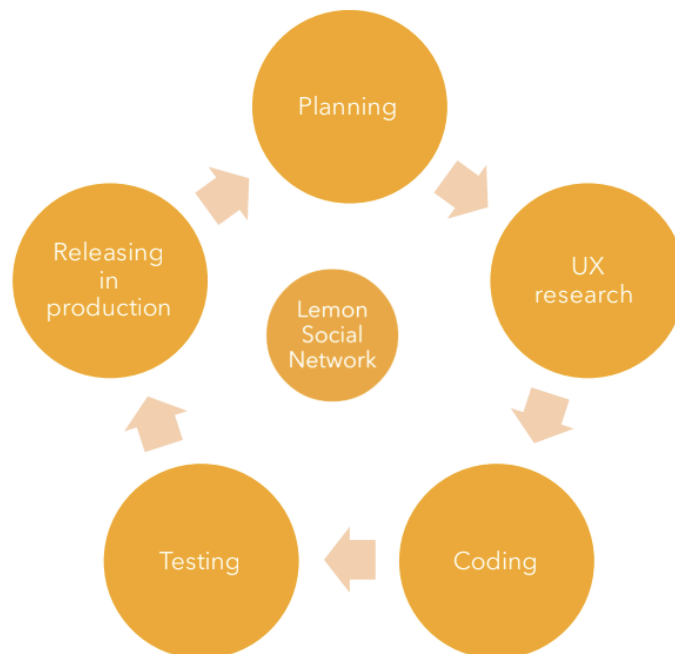
2.6 Software Development Life Cycle

The Software Development Life Cycle (SDLC) is an organised series of steps in software engineering that leads to the development of the desired software product.

To build, test and release a social network requires detailed planning. The development cycle can be divided into

- Planning
- UX research
- Coding
- Testing
- Releasing in production

Figure 2.7 - Agile methodology



This project followed the agile methodology. Agile approach is a strategic planning method that breaks a project into stages. It necessitates regular communication with stakeholders as well as ongoing development at all levels. Once the job begins, teams go through a cycle of planning, executing, and evaluating. By executing continuous iterations of development and testing across the software development life-cycle, the agile methodology may be utilised to provide agility in a software development project. Development and testing go hand in hand in agile software development [26].

The planning step is where the developer estimates all the work and breaks it down into smaller pieces. After that, he conducts UX research to better understand the problem. The coding process follows, with each item being addressed in accordance with the UX design. The "Homepage" and the ability to write a post were the first features to be coded. Following the coding phases, comes the testing phase. During this phase, all of the social network's features will be tested, and feedback will be given. The comments will be utilised to improve the features so that they better meet the needs of the consumers, resulting in a "product-market fit" solution [27]. The release in the production phase is the last stage, in which the frontend and backend implementations are put in the cloud and made available to all users. In order to enhance the product over time, the cycle will be repeated.

2.6.2 UX Research

One of the first steps even before starting the coding phase is the User Experience (UX) research.

The User Experience research phase is an essential aspect of the software development cycle. It includes market research, interviews with users and prototyping. The UX should also include detailed mockups of each individual screen of the platform, a transition between them and UI loading states.

In a typical social network, UX research is done by a team of UX designers. After the initial prototyping of the screens, users are invited to try the mockup screens and give feedback. Once the UX mockups have been validated they can be handed over to the development team [28].

2.6.3 Coding phase

The major goal of the coding phase is to transfer the design of the software system into a programming language format that can be used by a computer. The aim of coding is to create a high-quality system that can be used in any circumstance.

The coding phase starts with breaking down the UX designs into two main parts:

- The server-side logic (usually coded by a backend development team);
- The client-side UI (usually coded by a frontend development team);

Usually, this is handled by different teams working together but it depends on the company size where it can be done by a single team of full-stack developers [29].

The backend team takes the requirements and decides on the technology, the Cloud Provider and the system architecture. After all of this gets decided a software development management tool is used to write requirements tickets. Tools like JIRA and Trello are very popular and fit really well with the software development cycle.

A typical software development cycle is called a “Sprint”. A Sprint is between 2 to 4 weeks long and is a way to organise the work into limited chunks of time with specific deadlines.

The frontend team does a similar thing and breaks down the work in Sprints and works closely with the backend team to make sure the interaction between the web app and the server endpoints matches the requirements.

Once the first sprint of the coding phase is done and code is ready to be tested and is handed over to the QA (Quality Assurance) team [30].

2.6.4 Testing

The testing phase usually is done by a dedicated team of testing engineers. This is typically done in a QA environment, separated from development and from the production environment.

QA Engineers will manually test the code to see if it matches the requirements and also write code to run automated tests that can run on different configurations like different screen sizes or different compute power to make sure the software is as stable as possible [31].

2.6.5 Releasing in production

Once the code has been tested the next phase is the production release. This can be done in rollout phases. For example, the code can be first released to 10% of the users to see any initial reactions to the new or updated features and to catch any bugs before the majority of the users will get to see it.

The release process is usually handled by dedicated software like, for example, Jenkins. Jenkins allows developers to quickly compile, test and release code with the ability to see if the tests break and roll back if a feature causes any issues in production [32].

CHAPTER 3 – ANALYSIS

This project focuses on building a social network related to a healthy lifestyle. The first step is to divide the requirements into:

1. Functional requirements
2. Non Functional requirements

The functional requirements describe how a system should behave and the expected behaviour from a user point of view. This social networking service acts as a way to connect individuals with other people to share information, ideas, and messages about lifestyle [33].

For Lemon Social Network we can define the following functional requirements for the end-user:

Table 3.1 - The functional requirements

Requirement	Feature	Implementation
As a user I should be able to login with Google and Facebook	Login	Frontend and Backend
As a user I should be able to navigate	Navigation	Frontend
As a user I should be able to see the home page with all posts	Activity stream	Frontend
As a user I should be able to scroll and see more posts after the 5th post	Activity stream	Frontend and Backend
As a user I should see the products page	Activity stream	Frontend and Backend
As a user I should see the diets page	Activity stream	Frontend and Backend
As a user I should see the profile page	User Profile	Frontend and Backend
As a user I should see the list of notifications	Notifications page	Frontend and Backend
As a user I should be able to create a new post	Creating post	Frontend and Backend

Lemon Social Network also has an admin side responsible for monitoring and curating the content created by users. The administrator functional requirements are:

Table 3.2 - The Administrator functional requirements

Requirement	Impact
As an admin user I should be able to delete any user	User accounts
As an admin user I should be able to delete any post created by any user	Content
As an admin user I should restrict access to any registered user	User access
As an admin user I should see the last deployment of the frontend and backend	Deployments

The non-functional requirements define the quality of the product and are particularly important as our product is customer-facing. In our case, as a social network, the non-functional requirements are:

Table 3.3 - The non-functional requirements

Requirement	Impact
Users should be allowed to use other social logins like Google and Facebook to reduce security risks	Login security
The system should use a JWT token exchange	Backend security
The JWT short-lived token should expire in 60 seconds	Backend security
The backend should be capable to handle at least 1000 simultaneous users or 1000 requests per second	Backend scalability
The Angular frontend should be tested against SQL injections	Frontend security
The Angular frontend should use HTTP only cookies	Frontend security

The Angular frontend should be accessible from any mobile device with Android 4.4 or iOS 8 and above	OS requirements
The Angular frontend should adapt to any screen size above 470dp x 320dp	Responsiveness
The Angular frontend should be accessible on browsers like Chrome, Edge and Safari	Browser compatibility

Both functional and non-functional requirements are an essential step in building a scalable and a high standards software product. To architect a system capable of scaling and self-recovery in case of failure we can define a set of **system requirements** that specify what are the hardware and the software requirements [34].

Table 3.4 - The system requirements

Requirement	Type
The database should be a NoSQL type	Database type
The database should be backed up daily at 2 am	System recovery
The system should use Firebase Cloud Functions to run its code	System framework
The system should run a cron job daily to check its status	System operability
The frontend should be written in Angular and Typescript	Frontend programming language and framework
The backend should use Cloud Functions to execute backend code in JavaScript	Backend programming language and framework

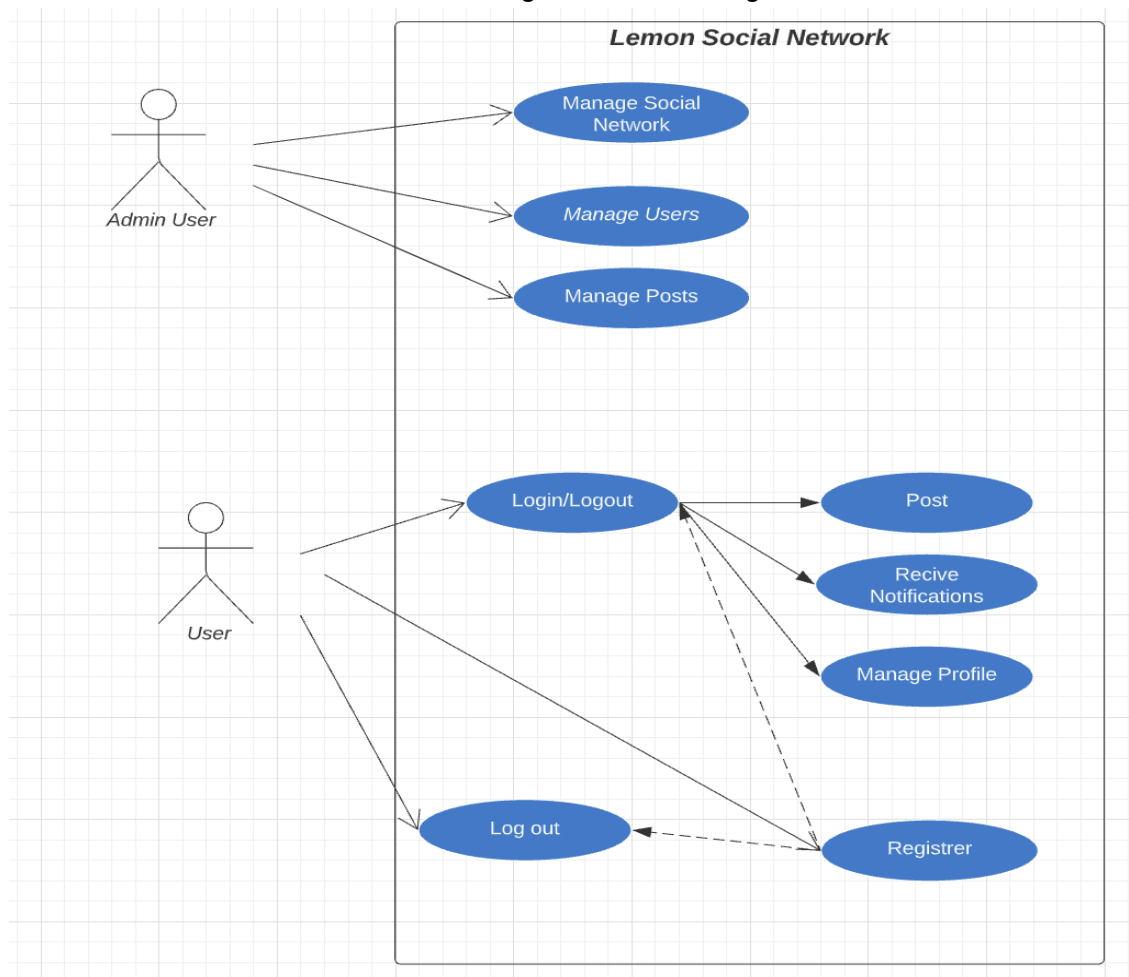
3.1 UML Diagrams- use case

The interaction between the components of Lemon Social Network is graphically represented in this Use Case Diagram [35]. It represents the system analysis process for identifying, clarifying, and organising Lemon Social Network. In this Use Case Diagram, the major actors of Lemon Social Network are the Admin User, New User, and Existing User.

The relationship between and among the actors and the uses cases of Lemon Social Network are the following:

- Admin User Entity. The use cases of an Admin User are: Manage Social Network, Manage Users and Manage posts.
- Users Entity. The use cases of New Users Entity are: Register, Post, Manage Profile, Receive Notification, Login/Logout

Figure 3.1 UML diagram



3.2 System Requirements

The requirements for a Cloud-native platform like Lemon can be divided into hardware and software requirements.

The hardware requirements refer directly to the machine specifications in terms of RAM memory, CPU clock speed and count, hard disk space and type. In some specific cases, a physical firewall requirement might be necessary [36].

A typical Cloud-native hardware requirement might look like this:

Table 3.5 - Cloud-native hardware requirement

Type	Spec
CPU	1 Core, 3.5GHZ
RAM	3 GB
Hard Disk	SSD, 30GB

The evaluations of such requirements must be done directly looking into the platform estimated usage and costs. A lower latency system with lighter specs will give a better user experience but at a higher cost.

These are some examples of pricing in Google Cloud from Compute Engine instances.

Table 3.6 - Google Cloud pricing

Machine Type	CPU Count	Memory	Price per month
e2-standard-2	2	8GB	\$48.91
e2-standard-4	4	16GB	\$97.83

Compute Engine instances are a standard way of measuring compute power in the Google Cloud Platform. There are similar technologies in other Cloud Providers like Amazon Web Services or Microsoft Azure.

Some of the criteria for evaluating the hardware requirements are:

- Number of requests per second (how much load the system will have in a real production environment)
- Location of the hardware (this will impact the cost, hardware hosted in colder regions costs less)
- Price per CPU/RAM/Storage per hour or per month
- The ability to physically access the hardware or not (most Cloud Providers don't offer the ability to physically access the Data Centres)

In some instances the platform might be hosted using 'serverless' technology. In this case, the hardware requirements might be similar but the ability to finetune could be more difficult. For example, Firebase offers a serverless technology called Cloud Functions. Cloud Functions allow developers to write code without worrying about the underlying hardware in order to streamline the software development process. Because of this abstraction, some compromises have to be made in terms of specification options [37].

An example of a Cloud Function pricing model is as follows:

Table 3.7 - Cloud Function pricing

Memory	vCPU	Price/100ms
128MB	.0833 vCPU	\$0.000000231
256MB	.1666 vCPU	\$0.000000463

In this scenario the pricing is not constant but varies based on usage (pay as you go business model). So the 128MB RAM and 0.8vCPU are billed for every 100ms of usage.

When deciding the hardware specifications between a classical virtual server and a 'serverless' technology, careful consideration is required especially on the pricing model.

The software requirements refer to the platform's needs for different types of software to operate, from operating systems to frameworks and programming languages.

For a social network, the software requirements can be broken down into different categories:

- Hosting requirements
- Continuous integration requirements

- c. Development requirements
- d. Security requirements

The table below shows an example of requirements and their usage.

Table 3.8 - Software requirements

Type of software	Name	Used by
Operating system	Debian 11	By the Cloud servers
Firewall	GCP Firewall	By the Cloud server
Continuous integration	Jenkins	Used as continuous integration solution to test and deploy code
Frontend framework	Angular SDK	To code the platform's UI and UX
Backend framework	Java SDK	To code the code the platform's backend
Antivirus	Norton 360	Entire company team members

The software for the Cloud servers are usually provided by the provider but needs to be regularly maintained by the company.

The Software Development Kit or SDK is usually used under licence by the software developers and typically developed by a different organisation. For example, the Angular SDK is developed by Google.

The Antivirus and Firewall software is typically installed on local machines or cloud machines based on requirements and updated frequently.

The continuous integration software typically has hardware requirements as well that impact the performance of the deployment of the software. Some companies choose to host their own continuous integration solutions while others prefer to use a service provider like Bitbucket Pipelines or Gitlab Deployments [38].

Both the **hardware** and **software** requirements represent an important step in creating a robust software platform. It impacts the cost, development time and resources and requires a constant re-evaluation as the business grows or as the technology changes.

3.3 Risk analysis

When building a complex software platform like a social network a robust risk analysis. We can divide the risk analysis in multiple verticals as: risk, probability of risk, impact and mitigation. One of the biggest concerns for a software platform, like Lemon, is a backend security breach [39].

The table below lists the potential risks for the platform, showing the level of risk and the impact. The mitigation column shows what steps should be taken in case of a critical situation.

Table 3.9 - Risk analysis

Risk	Probability of the risk happening	Impact	Mitigation
Backend API failure	Low	High impact, the work and data will be lost.	The risk could be avoided by taking regular backups or saving the content online as well.
Database failure	Low	High impact, the platform will stop working, no data will be lost	Regular backups or saving the content online as well.
Backend security breach	Very low	User data, for example passwords hashes leaked	Security checks every 6 months or every big feature update
Backend credentials lost	Very low	High impact, the work and data will be lost or leaked.	Credentials like usernames and password should be kept secure
Any hardware failure	Low	Medium impact, work and data will be lost	Backup devices like computers or cloud servers
Personal illness or injury	Medium	Project can be delayed	Assign additional personnel to replace duties

CHAPTER 4 - DESIGN

This chapter explains how the lemon social network was designed and implemented. The design outlines the framework's needs as well as the rationale for teaching and employing specific design patterns. This chapter's implementation part explains and describes how the lemon social network was created.

4.1 Introduction

In today's market with continually evolving interfaces, UX design has become increasingly important. The primary purpose of user experience design is to create a system that gives a favourable user experience. As a result, user experience design, usability, accessibility, information architecture, and human-computer interaction are all incorporated within UXD.

When interacting with a system, the user experience (UX) refers to how a person feels. Any sort of human-device interaction, such as a website, web application, desktop programme, and so on, falls under this category.

UX is important since it aims to meet the user's needs. Its mission is to provide positive experiences that inspire customers to continue with a product or brand. Furthermore, a rich user experience allows you to design customer journeys that are most conducive to economic success on your website [40].

InVision Freehand was used to create all of the social network's interfaces.

InVision Freehand is a cloud-based digital product design tool that aids developers and businesses of all sizes in prototyping and managing their products. People can discuss, test, and experiment with producing templates and concepts on a digital dashboard.

4.2 Design

Every screen that will be coded has been planned out. The flow and actions that can be done are shown by the red arrows.

The log in pager will be the first page the user sees. A user will be able to log in to the platform using google or the login button. The "Create new account screen" is the following screen, where a person can register using an email and a password.

The main page, the diets page, and the recipes page follow. Diets, recipients, and items in the healthy-lifestyle category are the several screens that a user may view on the site. A user can like a post of other users.

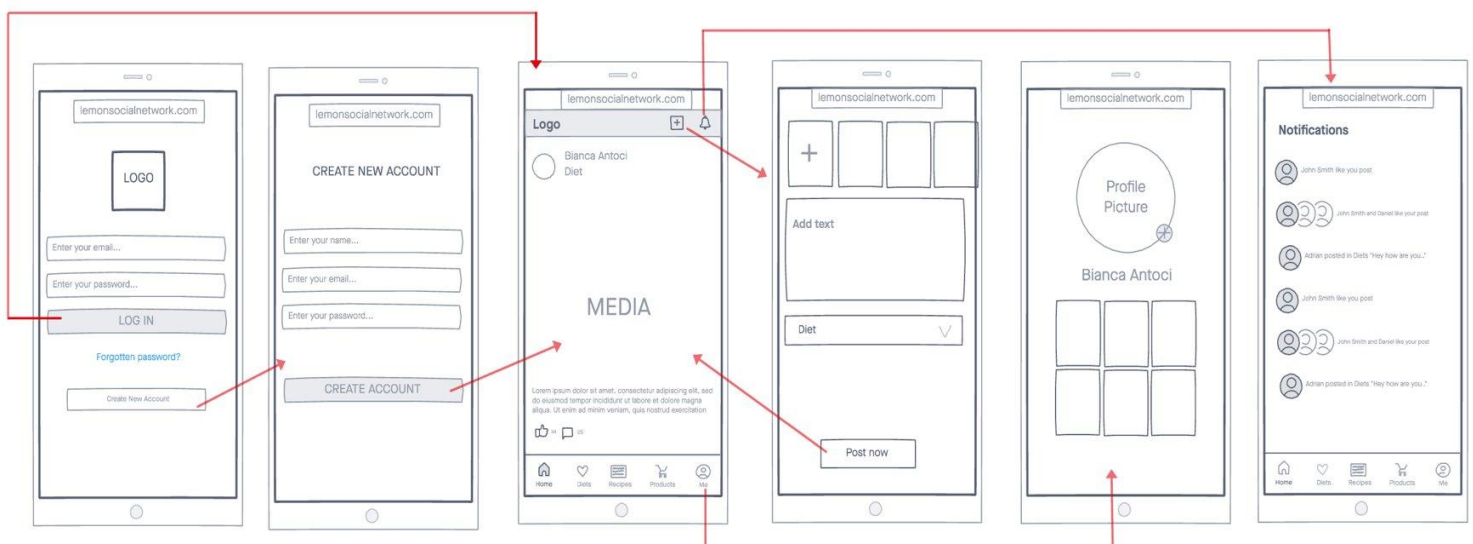
By selecting the plus icon, a user may create a new post by simply uploading a photo and writing a short description. It will also provide the option of placing the post on the diets, recipes, or products pages.

By clicking the notification icon, users can view all of the people who liked his post.

There's also a profile page where users may personalise their appearance on the network and where they can see all the posts that they created.

Also, the profile page allows users to log out and delete their accounts.

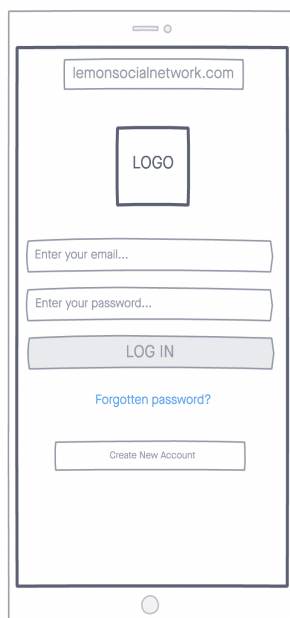
Figure 4.1 - UX design



This diagram shows the entire user journey on the platform. Once logged in the user can navigate to all the other features in no particular order.

In the next section, each screen is analysed in detail in order to better understand the user experience:

Figure 4.2 - UX design



A mobile app screen design for login and registration. At the top, there is a header bar with the URL 'lemonsocialnetwork.com'. Below the header is a square placeholder for a 'LOGO'. The main content area contains three input fields: 'Enter your email...', 'Enter your password...', and a 'LOG IN' button. Below the 'LOG IN' button is a link for 'Forgotten password?' and a 'Create New Account' button.

The first screen that the user will see is the login and register screen. From here there are three main options: to log in, to register and to reset the password.

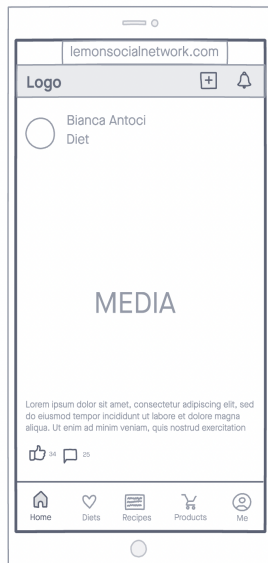
Figure 4.3 - UX design



A mobile app screen design for user registration. At the top, there is a header bar with the URL 'lemonsocialnetwork.com'. Below the header is a section titled 'CREATE NEW ACCOUNT'. The main content area contains three input fields: 'Enter your name...', 'Enter your email...', and 'Enter your password...'. Below the input fields is a 'CREATE ACCOUNT' button.

The screen is the user registration screen. From here the user can enter their details and create a new account. A very simple user experience.

Figure 4.4 - UX design

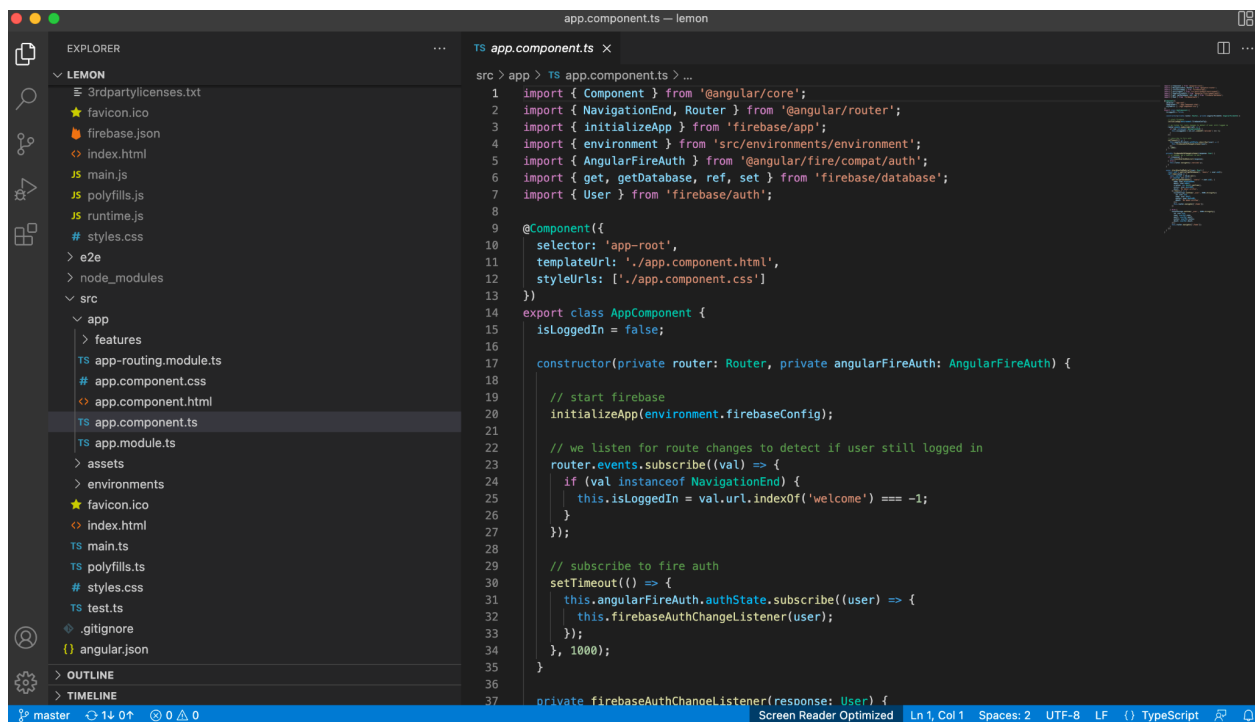


The main screen of the platform is the home page with the bottom navigation. The bottom navigation allows users to quickly navigate the main features as: diets, recipients, products and profile. The main feed is 'scrollable', meaning that users can scroll through a list of posts created by other users.

Chapter 5 - IMPLEMENTATION

The software development tool used to implement Lemon Social Network is Visual Studio Code.

Figure 5.1 - Visual Studio Code



The implementation of Lemon social network has been broken down into two separate parts. The first is the platform's interface and the second is the server-side implementation. Both parts have to work in conjunctions and have to match the UX/UI requirements.

Before writing any code a good practice is to create a new GIT repository. This should act as a code repository and a versioning control system. In Lemon's case, Github was chosen and the source code can be viewed here [41]:

<https://github.com/bianca-antoci/Lemon-Social>

The frontend implementation

The frontend framework used to build Lemon is called Angular and it was created by Google. The first step is to create a new project using the command line:

ng create lemon-social-network

The project should be created in the current folder and comes pre-loaded with a simple test interface. The next step is to create a basic project structure based on the UX:

/features/footer

/features/home

/features/diets-page

/features/footer

/features/header

/features/home-page

/features/new-post-page

/features/notifications-page

/features/post-list

/features/products-page

/features/profile-page

/features/recipes-page

/features/single-post

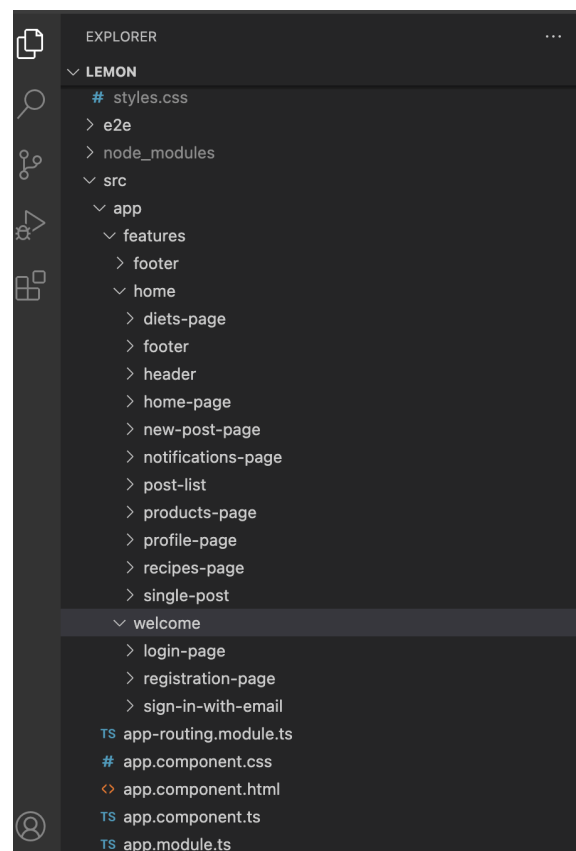
/features/welcome

/features/login

/features/registration-page

/features/sign-in-with-email

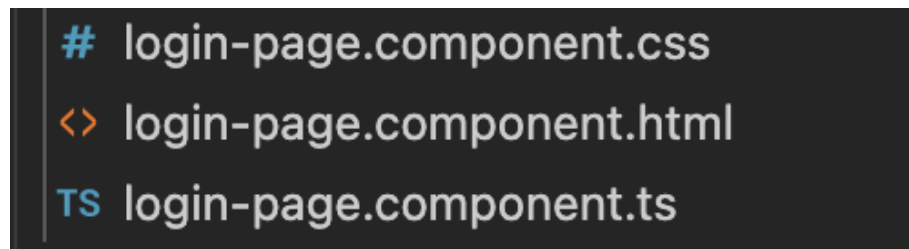
Figure 5.2 - Features



Each folder represents a feature in the platform. In Angular, each page is considered a “Component” and each component should be in a separate folder. Each folder should contain at least 3 source files:

- a TypeScript file containing the business logic of the component
- an HTML file containing the UI related code
- an SCSS file containing the component’s styling

Figure 5.3 - Example of Source Files



The coding phase can be tackled in any order but a good practice is to start with the typical user flow. For example, a user logs in and checks the latest posts. In our case the order can be as follows:

a. The login page

The first component (page) that will be implemented is the login page. The login page allows a user to register with email and password or via Google Sign In.

The Google Sign-In is a standard component we can use to embed on the HTML page but the rest of the features should be coded manually.

Figure 5.4 - Sign in with email and password

```
8      <span class="title">Sign in with email</span>
9    </mat-toolbar>
10  </p>
11  <div>
12    <div class="login-container">
13      <form [formGroup]="userForm" #formDir="ngForm">
14        <div class="form-group input-field">
15          <mat-form-field appearance="fill">
16            <mat-label>Enter your email</mat-label>
17            <input id="email" matInput placeholder="E.g. me@email.com" formControlName="email" email="true"
18              required>
19            <mat-error *ngIf="email.invalid && (email.dirty || email.touched)" class="alert alert-danger">
20              <div *ngIf="email.errors?.['required']">
21                Email is required.
22              </div>
23              <div *ngIf="email.errors?.['email']">
24                Email is invalid.
25              </div>
26              <div *ngIf="email.errors?.['minlength']">
27                Name must be at least 2 characters long
28              </div>
29            </mat-error>
30          </mat-form-field>
31        </div>
32        <div class="form-group input-field">
33          <mat-form-field appearance="fill">
34            <mat-label>Enter your password</mat-label>
35            <input id="password" type="password" matInput placeholder="Use a strong password"
36              formControlName="password" required>
37            <mat-error *ngIf="password.invalid && (password.dirty || password.touched)"
38              class="alert alert-danger">
39              <div *ngIf="password.errors?.['required']">
40                Password is required.
41              </div>
42              <div *ngIf="password.errors?.['minlength']">
43                Name must be at least 8 characters long
44              </div>
45            </mat-error>
46          </mat-form-field>
47        </div>
48      </form>
49    </div>
50  </div>
```

b. The footer

Once the user logs in the footer component should be displayed on every page. The UI should contain a list of all navigation options. For example:

```
<footer class="footer">
  <mat-toolbar colour="primary">
    </mat-toolbar>
</footer>
```

Within the “mat-toolbar” tag we can add all the navigation options, like the list of recipes, the list of products and profile.

Figure 5.5 - Home footer implementation

```
1 <footer class="footer">
2   <mat-toolbar color="primary">
3     <button (click)="onItemClick('home')" mat-icon-button color="basic"
4       aria-label="Example icon button with a home icon">
5       <mat-icon class="mat-icon-selected">home</mat-icon>
6     </button>
```

c. The diets page

This page shows the list of diets created by users and sorted by created date. Each diet will be approved by an admin before being visible to all users. The code looks as follows:

```
<div>
```

```
  <app-home-post-list-component [type]="diet' [title]="Latest
  diets"></app-home-post-list-component>
```

```
</div>
```

Notice the 'type' parameter that indicates what kind of posts should the component load.

Figure 5.6 - Diets page

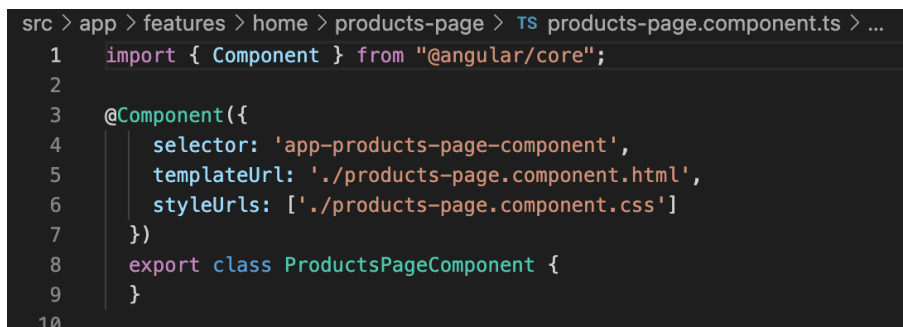
```
1 <div>
2   <app-home-post-list-component [type]='diet' [title]='Latest diets'></app-home-post-list-component>
3 </div>
```

d. The products page

The products page shows the list of health-related products recommendations posted by other users. The layout is similar to the 'Diets page' but the page header will be different. The products page content can be: text only, text and picture or only picture. This is how the products TypeScript component looks like:

```
@Component({
  selector: 'app-products-page-component',
  templateUrl: './products-page.component.html',
  styleUrls: ['./products-page.component.css']
})
export class ProductsPageComponent { }
```

Figure 5.7 - Products TypeScript



```
src > app > features > home > products-page > TS products-page.component.ts > ...
1  import { Component } from "@angular/core";
2
3  @Component({
4    selector: 'app-products-page-component',
5    templateUrl: './products-page.component.html',
6    styleUrls: ['./products-page.component.css']
7  })
8  export class ProductsPageComponent {
9  }
10
```

e. The user profile page

After registration, each user has the ability to edit their name and picture, browse old posts and log out. All these actions are available from the 'User Profile' page, which was designed with the objective to allow users to see each other's profile and for personal information edits.

The profile page, photo upload frontend feature was developed as follows:

```
<div>
    <div class="avatar-pencil" (click)="fileSelect.click()"> <mat-icon>edit</mat-icon>
</div>
<div>
    
    <input style="display: none" #fileSelect accept="image/*" class="input-file" type='file'
    (change)="onFilePick($event);" />
</div>
```

In the first block we allow the user to click the 'Edit' button with 'fileSelect.click()' and once picked the picture will be displayed using the 'img' tag.

Figure 5.8 - Profile Page

```
<div class="container feature">
  <div>
    <div class="avatar-pencil" (click)="fileSelect.click()">
      <mat-icon>edit</mat-icon>
    </div>
    <div>
      
      <input style="display: none" #fileSelect accept="image/*" class="input-file" type='file'
      (change)="onFilePick($event);" />
    </div>
  </div>
</div>
```

f. Creating a new post

This feature allows the user to create a new post. The post can be of multiple types:

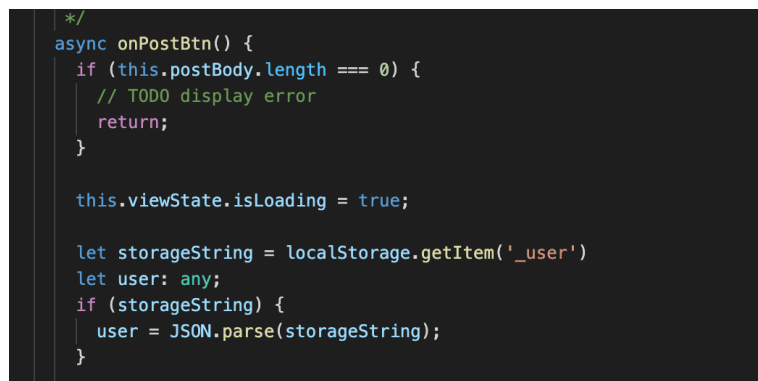
- A diet post
- A product post
- A recipe post

In the code below we define a function that will create a post in the database and redirect to home page:

```
async onPostBtn() {  
    if (this.postBody.length === 0) return;  
    this.viewState.isLoading = true;  
    let storageString = localStorage.getItem('_user')  
    let user: any;  
  
    if (storageString) {  
        user = JSON.parse(storageString);  
    }  
    insetPostToDB(this.postBody, user, this.photos);  
    this.router.navigate(['/home']);  
}
```

This function first checks if there is a post description, if not an error will be displayed. Local storage service is interrogated to check if the user is still logged in and finally a new database entry is created.

Figure 5.9 - Creating a post in the database



```
*/  
async onPostBtn() {  
    if (this.postBody.length === 0) {  
        // TODO display error  
        return;  
    }  
  
    this.viewState.isLoading = true;  
  
    let storageString = localStorage.getItem('_user')  
    let user: any;  
    if (storageString) {  
        user = JSON.parse(storageString);  
    }  
}
```

The backend implementation

The platform backend is hosted in Firebase. The interface that allows the Angular frontend to communicate with the backend is called Firebase Frontend library. The library allows the frontend to directly make database calls with inserts, updates and delete operations. For example to insert a new post:

```
import { getDatabase, ref, set } from "firebase/database";
import { getStorage, uploadBytes } from "firebase/storage";
import { ref as StorageRef } from "firebase/storage";
-----
set(ref(this.fireDB, 'posts/' + new Date().getTime()), {
  body: this.postBody,
  type: this.selectedPostType,
  images: this.imageURLs,
  userName: user.name,
  profilePicture: user.avatar,
  approved: false,
  likeCount: 0,
  likedBy: [],
  createdBy: user.id,
});
```

All the backend related operations are implemented in a similar way. The browser local storage functionality is used in conjunction to store information about the user session; e.g. is the user logged in or not?

Figure 5.10 - Backend implementation

```
1 import { Component } from "@angular/core";
2 import { Router } from "@angular/router";
3 import { getDatabase, ref, set } from "firebase/database";
4 import { getStorage, uploadBytes } from "firebase/storage";
5 import { ref as StorageRef } from "firebase/storage";
6
```

Figure 5.10 - Backend implementation

```
    }

    set(ref(this.fireDB, 'posts/' + new Date().getTime()), {
      body: this.postBody,
      type: this.selectedPostType,
      images: this.imageURLs,
      userName: user.name,
      profilePicture: user.avatar,
      approved: false,
      likeCount: 0,
      likedBy: [],
      createdBy: user.id,
    });

    this.router.navigate(['/home']);
  }
}
```

CHAPTER 6 - TESTING

The testing scenario has been performed on the deployed version of Lemon Social Network hosted in Google Cloud. The testing was conducted using an iPhone X and on Google Chrome browser using different platform users and one system administrator [42].

The tests conducted follow the typical user flow on the platform and include every feature of the platform as seen in the 'Test Cases' column. Every test result is matched by a screenshot, proving the result of the test.

Table 6.1 Testing

Test Cases	Expected result	Actual result	Discrepancies	Test result
As a user I should be able to register	Users should be able to see the registration page	As expected	No	Figure 6.1
As a user I should be able to register	Users should be able to click the register button	As expected	No	Figure 6.2
As a user I should see an error message if the password is too short	User should be able to see the error message when inserting a password	As expected	No	Figure 6.3
As a user I should be able to sign in with Google	User should be able to click the sign-in with Google button	As expected	No	Figure 6.4
As a user I should be able to sign in with an email and password	User should be able to click the sign in with Google button	As expected	No	Figure 6.5
As a user I should be able to add a new post	User should be able to click the add icon	As expected	No	Figure 6.6
As a user I should be able to upload a new photo when creating a post	User should be able to click the add photo icon	As expected	No	Figure 6.7
As a user I should be able to add a description on my post	User should be able to type	As expected	No	Figure 6.8

As a user I should be able to choose the post type	User should be able to see the list of post type	As expected	No	Figure 6.9
As a user I should be able to see the notification	User should be able to click the notification button	As expected	No	Figure 6.10
As a user I should be able to navigate the bottom navigation bar	User should be able to click the different bottom navigation icons	As expected	No	Figure 6.11
As a user I should be able to like a post	User should be able to click the like icon	As expected	No	Figure 6.12
As a user I should be able to change my profile picture	The user should see the update picture	As expected	No	Figure 6.13
As a user I should be able to change my name	The user should be able to see the updated name	As expected	No	Figure 6.14
As a user I should be able to type something about me	User should be able to type something	As expected	No	Figure 6.15
As a user I should be able to log out	User should be able to click the logout button	As expected	No	Figure 6.16
As an admin i should be able to approve a post	Admin should be able to approve any posts,recipes and products.	As expected	No	Figure 6.17

CHAPTER 7 - EVALUATION

7.1 Personal Evaluation

Developing a software platform, from its research phase, implementation, testing and releasing is a challenging journey as the skills required are quite varied.

The research phase challenges the personal assumptions that one can have about a particular topic, in this case, what social networks are and why they are so popular. Also from a creative perspective, it helps to learn how platforms evolve and create new features.

The implementation phase helps to develop the actual coding skills required for creating a product. Because the platform's backend and frontend are separated, different coding skills are required for each. On the server side, knowledge about database management, how tables schema are created and managed is essential. On the client frontend side, knowledge about TypeScript programming language is required as well as a good understanding of the Angular frontend framework.

The testing and releasing phase require a good attention to detail, as every part of the web application needs to be tested thoroughly. When an issue is found a "bug" ticket gets created and estimated and will be prioritised based on its severity. A particular length of time will be spent on each bug and, in some cases, the issue will be postponed if no solution has been found in a meaningful amount of time.

The time management skill plays an important role when working on any type of project. In this particular case, time has to be allocated in such a way that allows for a constant shift between tasks that are progressing simultaneously.

Learning about a new programming language like TypeScript and a new frontend framework like Angular has its challenges. In most cases, the documentation of the programming language is a good way to solve most of the problems encountered. Also a good alternative to find solutions to programming problems is a website called StackOverflow, which is a place where the software developers community is most active

7.2 System Evaluation

The most difficult aspect of creating a software platform is the user experience or UX. It necessitates a thorough knowledge of the customer's needs. When creating a UX design, users are typically divided into 'Personas'. A persona represents a collection of attributes that describe a specific type of user.

As a new programming language, typescript proved difficult to learn. It was particularly intriguing since, in comparison to JavaScript, it is a highly typing-based programming language, resulting in cleaner code. The frontend development for the project was done using Angular, which has Typescript built in.

Page routing, modules vs components, and CSS vs SCC are all tough concepts in Angular. As Typescript is built on top of JavaScript, knowing how to write an Angular application requires a fundamental comprehension of the language.

Considering Firestore Database is a NoSQL database, no JOIN queries are permitted.

CHAPTER 8 - FURTHER ENHANCEMENTS

Lemon social network is primarily focusing on sharing healthy lifestyle related posts, products and recipes. It allows users to share and connect in a meaningful way.

The platform can be further enhanced by integrating more ways for people to connect and trust each other.

1. Profile with a 'Trust Score'

One of the fundamental concepts in a social network is the trust that a user has in the platform and the trust it has in other users. More trust a user has, the more it will interact with others. A further enhancement of the platform could be the introduction of a 'trust score'. The trust score will start at 1 and it can go up and down based on user interactions. If a user interaction with the platform and its users is positive the trust score will go up, and vice versa.

2. A messaging system end to end encrypted

Any social network that wants to connect its users in a meaningful way has a messaging system. An encrypted end to end system allows user messages to be obfuscated to other people or entities, so only the participants in the conversation can read the conversation in plain text. This is a widely known technique adopted by other messaging platforms like Whatsapp or Telegram.

3. Posting a comment as video

Lemon needs to inspire trust and the video medium is probably the best way. The traditional commenting feature has been via text. By allowing video comments people can post more genuine and trustworthy videos.

4. Inviting friends and get rewarded

One of the best ways to achieve growth is by organic growth. Meaning that users will invite users if they see that the platform benefits them in any way. The platform should allow users to easily invite friends and family by sharing a customizable link. This action should be rewarded by Lemon with prizes like discounts from local shops or Nectar points.

5. Groups

Users naturally tend to connect with others by joining similar interest groups. The platform should allow the formation of groups where users can create any type of interest group as long as the underline focus is the healthy lifestyle.

6. Weekly newsletter

A weekly newsletter is a common way to keep users engaged with weekly updates on posts on the platform as well as the latest trends. In practice this means that an email newsletter will be sent to all users that opt in to this feature.

7. Advanced super-admin feature

In the current version of the platform, an admin user with access to the Cloud hosting, can add, edit and delete any records. An improvement can be made and a dedicated web page where admins can monitor and curate the platform is a necessary improvement.

CHAPTER 9 – CONCLUSION

The aim of Lemon platform is to connect people in a meaningful way by promoting a healthy lifestyle, ranging from recipes, products and discussions. It allows users to exchange ideas and promote their own progress in becoming healthier. The research conducted concluded that most of the existing social network sites don't have a fundamental focus on a healthy lifestyle.

The user experience research played a fundamental role in understanding the key user types, their behaviour and directly impacted how Lemon looks and feels to the end users.

The platform has been conceptualised and built as a Cloud Native platform, hosted in Google Cloud, the hosting provider with one of the biggest data centres in the world. Angular, the frontend framework used to build Lemon, is one of the most flexible and modern frameworks and it provides a solid foundation to build additional features as the user base grows. The Firebase Database is a robust and scalable database allowing the platform to connect users from all over the world in real time.

A modern software platform has to put security as one of its top priorities. A robust risk analysis has been conducted to ensure that from a hardware and software standpoint Lemon meets all the security requirements. Prior to the platform being released, all requirements went through a testing phase, where every part of the platform was tested against predefined scenarios to ensure that all the functional and nonfunctional requirements were met.

The content that a social network produces has to be constantly moderated. Lemon has been structured in such a way that administrators have the ability to moderate any content created on the platform before it is presented to all the other users. The administrators can also remove any user from the platform if their content doesn't meet the accepted standards.

Lemon is set to be an innovative social networking platform with a clear objective of promoting a lifestyle and creating meaningful connections. This study and the implementation of the Lemon platform are the foundations of a new digital product in a niche market.

CHAPTER 10 – REFERENCES

- [1]T.-T. Wu, “Using smart mobile devices in social-network-based health education practice: A learning behaviour analysis,” *Nurse Education Today*, vol. 34, no. 6, pp. 958–963, Jun. 2014, doi: 10.1016/j.nedt.2014.01.013.
- [2]M. Simeone and D. Scarpato, “Sustainable consumption: How does social media affect food choices?,” *Journal of Cleaner Production*, vol. 277, no. 11, p. 124036, Dec. 2020, doi: 10.1016/j.jclepro.2020.124036.
- [3]M. A. Pachucki, P. F. Jacques, and N. A. Christakis, “Social Network Concordance in Food Choice Among Spouses, Friends, and Siblings,” *American Journal of Public Health*, vol. 101, no. 11, pp. 2170–2177, Nov. 2011, doi: 10.2105/ajph.2011.300282.
- [4]L. Pilař, L. Kvasničková Stanislavská, and R. Kvasnička, “Healthy Food on the Twitter Social Network: Vegan, Homemade, and Organic Food,” *International Journal of Environmental Research and Public Health*, vol. 18, no. 7, p. 3815, Apr. 2021, doi: 10.3390/ijerph18073815.
- [5]M. Scotti, F. Ciocchetta, and F. Jordan, “Social and landscape effects on food webs: a multi-level network simulation model,” *Journal of Complex Networks*, vol. 1, no. 2, pp. 160–182, Sep. 2013, doi: 10.1093/comnet/cnt013.
- [6]C.-O. Kim, “Food choice patterns among frail older adults: The associations between social network, food choice values, and diet quality,” *Appetite*, vol. 96, pp. 116–121, Jan. 2016, doi: 10.1016/j.appet.2015.09.015.
- [7]M. J. Walsh and S. A. Baker, “Clean eating and Instagram: purity, defilement, and the idealisation of food,” *Food, Culture & Society*, vol. 23, no. 5, pp. 570–588, Oct. 2020, doi: 10.1080/15528014.2020.1806636.
- [8]T. J. Zagenczyk, K. D. Scott, R. Gibney, A. J. Murrell, and J. B. Thatcher, “Social influence and perceived organisational support: A social networks analysis,” *Organisational Behaviour and Human Decision Processes*, vol. 111, no. 2, pp. 127–138, Mar. 2010, doi: 10.1016/j.obhdp.2009.11.004.
- [9]F. Agneessens and G. (Joe) Labianca, “Collecting survey-based social network information in work organisations,” *Social Networks*, vol. 68, pp. 31–47, Jan. 2022, doi: 10.1016/j.socnet.2021.04.003.

- [10]S. P. Borgatti, M. G. Everett, and J. C. Johnson, Analysing social networks. Los Angeles: Sage Publications, 2018.
- [11]S. Cipolletta, C. Malighetti, C. Cenedese, and A. Spoto, “How Can Adolescents Benefit from the Use of Social Networks? The iGeneration on Instagram,” International Journal of Environmental Research and Public Health, vol. 17, no. 19, p. 6952, Sep. 2020, doi: 10.3390/ijerph17196952.
- [12]MBuild it better, “Introduction to the server side,” MDN Web Docs, Mar. 18, 2019. https://developer.mozilla.org/en-US/docs/Learn/Server-side/First_steps/Introduction.
- [13]“Server side rendering, client side rendering, and static site generation at a glance,” IONOS Digitalguide. <https://www.ionos.co.uk/digitalguide/websites/web-development/server-side-and-client-side-scripting-the-differences/> (accessed Mar. 15, 2022).
- [14]“What Do Client-Side and Server-Side Mean? | Client Side vs. Server Side | Cloudflare UK,” Cloudflare.
- [15]S. Settapat, T. Achalakul, and M. Ohkura, “Web-based 3D medical image visualisation framework for biomedical engineering education,” Computer Applications in Engineering Education, vol. 22, no. 2, pp. 216–226, Apr. 2011, doi: 10.1002/cae.20548.
- [16]“A WEB DEVELOPMENT FRAMEWORK FOR TEAM-BASED PROJECT COURSES,” Issues In Information Systems, 2009, doi: 10.48009/1_iis_2009_237-245.
- [17]Bear Bibeault, Yehuda Katz, and Aurelio De Rosa, JQuery in action. Shelter Island, Ny: Manning, 2015.
- [18]R. Nicholus, Beyond jQuery. Berkeley, California: Apress, 2016.
- [19]P. Kassenaar, Angular. Culemborg: Van Duuren Media, Februari, 2020.
- [20]B. Dayley, B. Dayley, and C. Dayley, Learning Angular : a hands-on guide to Angular 2 and Angular 4. Boston: Addison-Wesley, 2018.
- [21]B. Eisenman, Learning React Native : building mobile apps with JavaScript. Sebastopol, Ca: O’reilly Media, 2018.
- [22]Stoyan Stefanov, REACT - UP & RUNNING : building web applications. S.L.: O’reilly Media, 2020.

- [23]S. Touati and Benoit Dupont De Dinechin, Advanced backend code optimization. London ; Hoboken, Nj: Iste Ltd, Wiley, 2014.
- [24]T. Barker, High performance responsive design : building faster sites across devices. Sebastopol, Ca: O'reilly Media, 2014.
- [25]Priyanka Tyagi, PRAGMATIC FLUTTER : building cross-platform mobile apps for android, ios, web & desktop. S.L.: Crc Press, 2021.
- [26]P. Flewelling, The Agile developer's handbook : get more value from your software development : get the best out of the Agile methodology. Birmingham, Uk: Packt Publishing, 2018.
- [27]David Chenho Kung, Object-oriented software engineering : an agile unified methodology. New York, Ny: Mcgraw-Hill, 2014.
- [28]R. Wang and J. Yang, "User Experience (UX) Matters: What are the Most Desired Skills in the UX Designer and UX Researcher Job Ads," Journal of Communication Technology, vol. 4, no. 2, Jun. 2021, doi: 10.51548/jotec-2021-011.
- [29]D. Zheng and F. Da, "Phase coding method for absolute phase retrieval with a large number of codewords," Optics Express, vol. 20, no. 22, p. 24139, Oct. 2012, doi: 10.1364/oe.20.024139.
- [30]J. B. Anderson, Bandwidth efficient coding. Piscataway, Nj: Ieee Press ; Hoboken, New Jersey, 2017.
- [31]L. XU, L. CHEN, and B.-W. XU, "Testing Web Services Based on User Requirements," Chinese Journal of Computers, vol. 34, no. 6, pp. 1029–1040, Sep. 2011, doi: 10.3724/sp.j.1016.2011.01029.
- [32]D. A. Gratton, Developing practical wireless applications. Editora: Amsterdam ; Boston: Elsevier Digital Press, 2007.
- [33]P. Rome, "What are Non Functional Requirements — With Examples," Perforce Software, Aug. 11, 2020.
<https://www.perforce.com/blog/alm/what-are-non-functional-requirements-examples>.
- [34]U. Eriksson, "Functional Requirements vs Non Functional Requirements," ReQtest, Apr. 05, 2012.
<https://reqtest.com/requirements-blog/functional-vs-non-functional-requirements/>.

- [35]Podyachev, “UML diagram based TTCN tests implementing,” SPIIRAS Proceedings, vol. 0, no. 4, p. 155, Mar. 2014, doi: 10.15622/sp.4.10.
- [36]T. Myers, G. Dromey, and P. Fritzson, “Comodeling: From Requirements to an Integrated Software/Hardware Model,” *Computer*, vol. 44, no. 4, pp. 62–70, Apr. 2011, doi: 10.1109/mc.2010.270.
- [37]W. A. Aziz and M. Awad, “Network Function Virtualization over Cloud - Disaster Recovery Solution over Hybrid Cloud,” *International journal of simulation: systems, science & technology*, Dec. 2020, doi: 10.5013/ijssst.a.21.04.15.
- [38]N. Ali and R. Lai, “A method of software requirements specification and validation for global software development,” *Requirements Engineering*, vol. 22, no. 2, pp. 191–214, Nov. 2015, doi: 10.1007/s00766-015-0240-4.
- [39]L. A. T. Cox, “Improving Causal Inferences in Risk Analysis,” *Risk Analysis*, p. n/a-n/a, May 2013, doi: 10.1111/risa.12072.
- [40]J. Allen and J. Chudley, *Smashing UX design : foundations for designing online user experiences*. Chichester, West Sussex: John Wiley & Sons, 2012.
- [41]Achilleas Pipinellis, *GitHub essentials : unleash the power of collaborative development workflows using GitHub*. Birmingham, U.K., Packt Publishing Limited, 2018.
- [42]R. Sonwane, “Time and Cost Effective Web Testing on AWS,” *International Journal Of Engineering And Computer Science*, Sep. 2016, doi: 10.18535/ijecs/v5i9.69.
- [43]“Free Cloud Computing Services - AWS Free Tier,” *Amazon Web Services, Inc.* https://aws.amazon.com/free/?trk=d5254134-67ca-4a35-91cc-77868c97eedd&sc_channel=ps&sc_campaign=acquisition&sc_medium=ACQ-P (accessed Mar. 1, 2022).

APPENDICES

Useful links

Deployed website in the cloud:

<https://lemon-social-network.firebaseio.com/welcome>

Github project source code:

<https://github.com/bianca-antoci/Lemon-Social>

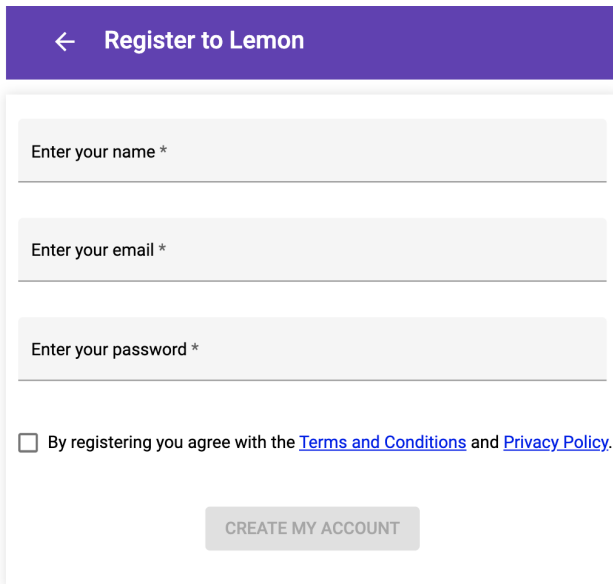
Link to Firebase Database:

<https://console.firebase.google.com/project/lemon-social-network/database/lemon-social-network-default-rtdb/data>

How to run the code:

- Unzip the Lemon folder
- Make sure you have installed NodeJs 12 and Angular SDK
- Open readme.md
- Follow the instructions

Testing screenshots



← Register to Lemon

Enter your name *

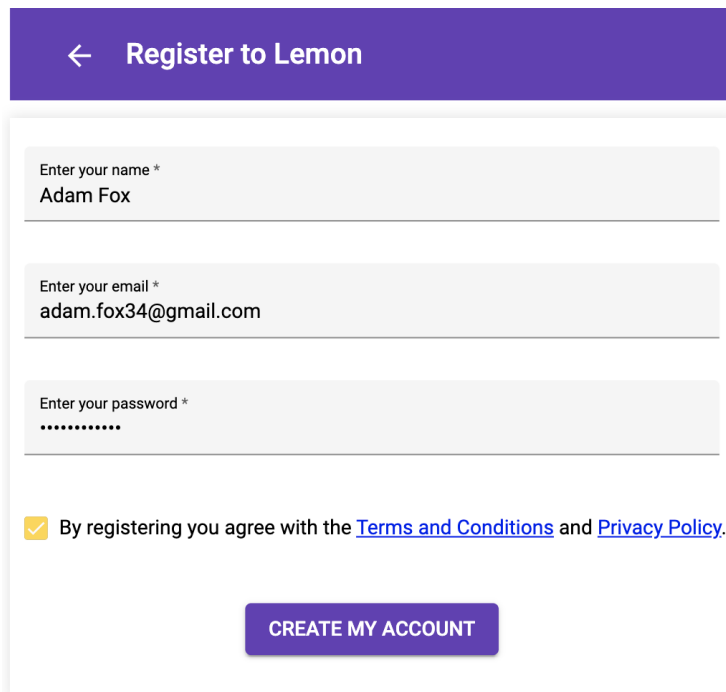
Enter your email *

Enter your password *

☐ By registering you agree with the [Terms and Conditions](#) and [Privacy Policy](#).

CREATE MY ACCOUNT

Testing - Figure 6.4



← Register to Lemon

Enter your name *
Adam Fox

Enter your email *
adam.fox34@gmail.com

Enter your password *
.....

☒ By registering you agree with the [Terms and Conditions](#) and [Privacy Policy](#).

CREATE MY ACCOUNT

Testing - Figure 6.2

[←](#) Register to Lemon

Enter your name *

Adam Fox

Enter your email *

adam.fox34@gmail.com

Enter your password *

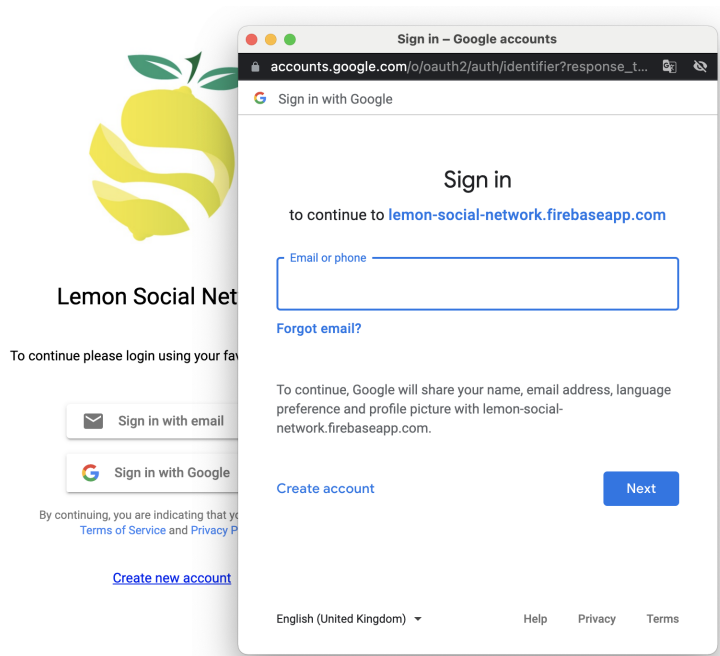
....

Name must be at least 8 characters long

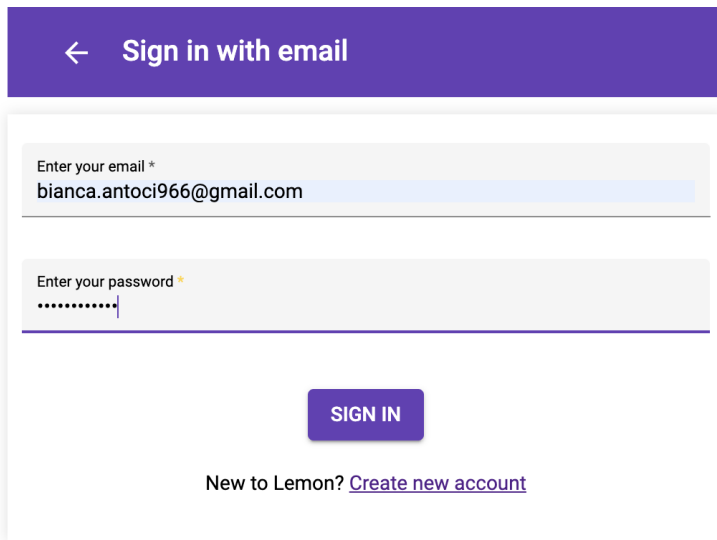
☒ By registering you agree with the [Terms and Conditions](#) and [Privacy Policy](#).

CREATE MY ACCOUNT

Testing - Figure 6.3

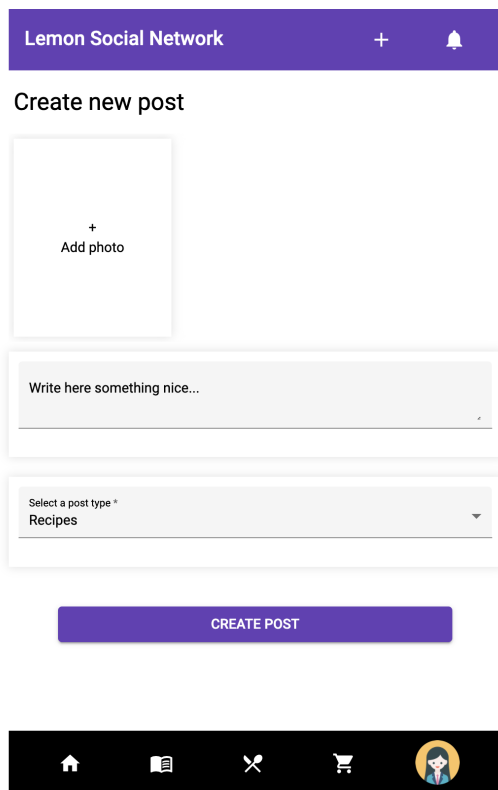


Testing - Figure 6.4



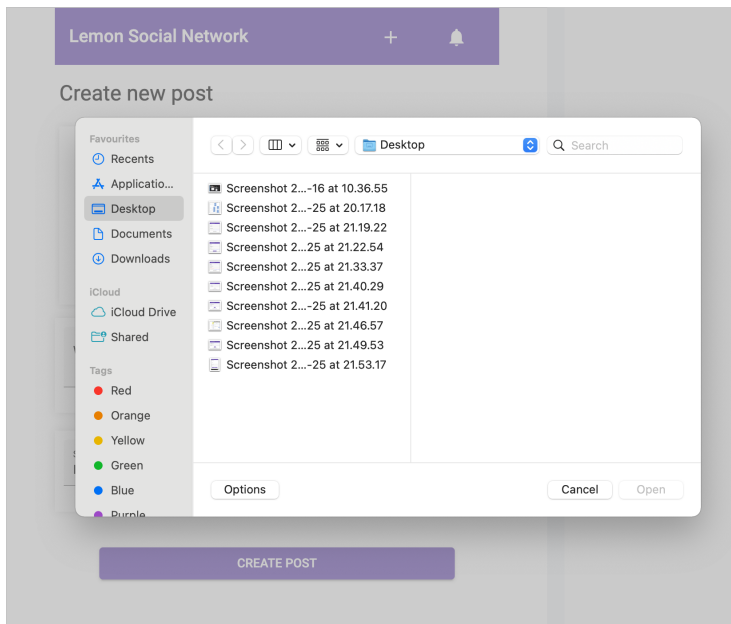
A mobile app sign-in screen with a purple header bar containing a back arrow and the text "Sign in with email". Below the header is a white card with two input fields. The first field is labeled "Enter your email *" and contains the text "bianca.antoci966@gmail.com". The second field is labeled "Enter your password *" and contains a series of dots. Below the input fields is a purple "SIGN IN" button. At the bottom of the card, there is a link that says "New to Lemon? [Create new account](#)".

Testing - Figure 6.5

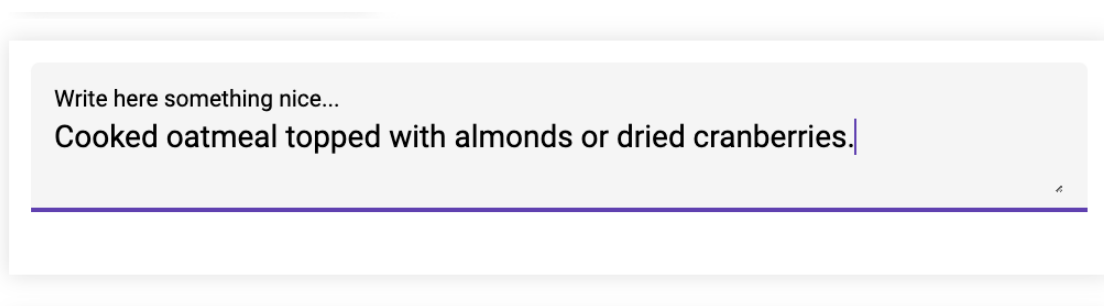


A mobile app "Create new post" screen. At the top is a purple header bar with the text "Lemon Social Network", a plus icon, and a bell icon. Below the header is the text "Create new post". There is a white card with a large square area for a photo, containing a plus icon and the text "Add photo". Below the photo area is a text input field with the placeholder text "Write here something nice...". Below the text input field is a dropdown menu labeled "Select a post type *" with the selected option "Recipes". Below the dropdown menu is a purple "CREATE POST" button. At the bottom of the screen is a black navigation bar with five icons: a home icon, a book icon, a scissors icon, a shopping cart icon, and a profile icon.

Testing - Figure 6.6



Testing - Figure 6.7



Testing - Figure 6.8

Write here something nice...

Cooked oatmeal topped with almonds or dried cranberries.

Recipes

✓ Diets

Products

CREATE POST

Testing - Figure 6.9

Lemon Social Network

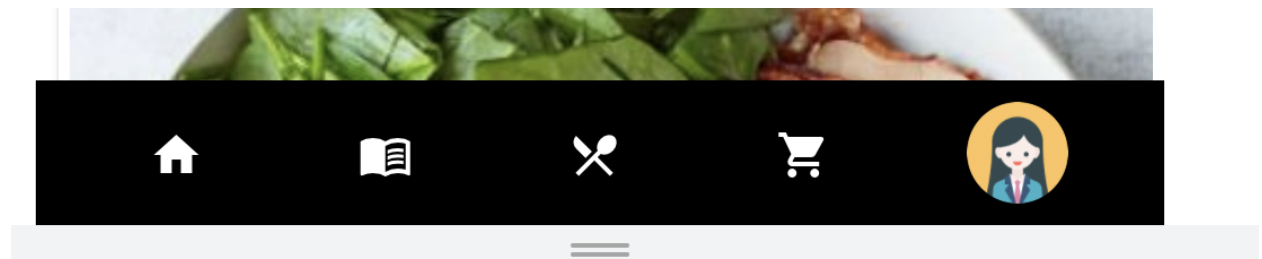
+

Latest notifications

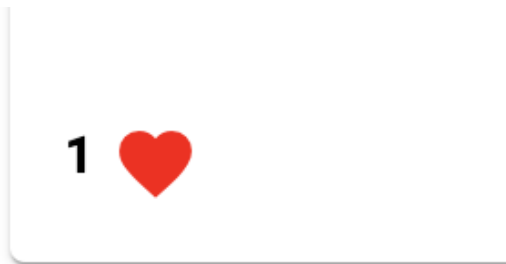
Adrian Antoci likes your post

Mark Smi likes your post

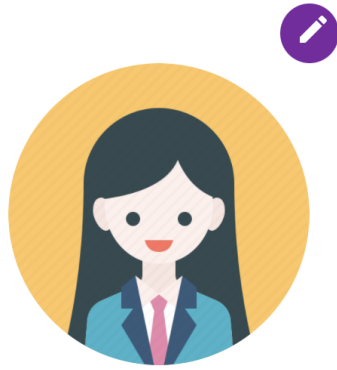
Testing - Figure 6.10



Testing - Figure 6.11



Testing - Figure 6.12



Testing - Figure 6.13

A modal dialog box titled "Edit name" in bold black text. Below the title is a text input field with a light gray background. Inside the field, the text "My name" is in a small purple font, and "Bianca Smith" is in a larger black font. A purple horizontal line is positioned below the text in the input field. At the bottom of the modal, there are two buttons: "CANCEL" and "UPDATE", both in black uppercase letters.

Testing - Figure 6.14

About me
Hello I'm Bianca and this is my profile

Testing - Figure 6.15

Other options

LOG OUT FROM LEMON

Testing - Figure 6.16

posts

```
1644697242674
  approved: true
  body: "\nQuick chicken hummus bowl\nMake our healthy chi..."
  createdBy: "tSeAAuIYohTy3Qw3d4C02U18uFh1"
  images
  likeCount: 1
  likedBy
  profilePicture: "https://lh3.googleusercontent.com/a/AATXAJy5uQV..."
  type: "recipe"
  userName: "Bianca Antocis"
```

Testing - Figure 6.17

