UNIVERSITY OF THESSALY

SCHOOL OF ENGINEERING

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

Consulting mobile chat app

Diploma Thesis

Dimitrios Tozakidis

Supervisor: Eleni Tousidou

September 2023

UNIVERSITY OF THESSALY

SCHOOL OF ENGINEERING

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

Consulting mobile chat app

Diploma Thesis

Dimitrios Tozakidis

Supervisor: Eleni Tousidou

September 2023

ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ

ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ

ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

Συμβουλευτική εφαρμογή συνομιλίας μέσω κινητού τηλεφώνου

Διπλωματική Εργασία

Δημήτριος Τοζακίδης

Επιβλέπων/πουσα: Ελένη Τουσίδου

Σεπτέμβριος 2023

Approved by the Examination Committee:

Supervisor **Eleni Tousidou**
Ε.ΔΙ.Π., Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών, Πανεπιστήμιο Θεσσαλίας

Member **Βασιλακόπουλος Μιχαήλ**
Καθηγητής, Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών, Πανεπιστήμιο Θεσσαλίας

Member **Φεύγας Αθανάσιος**
Ε.ΔΙ.Π., Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών, Πανεπιστήμιο Θεσσαλίας

Acknowledgements

Σελίδα με ευχαριστίες του συγγραφέα (προαιρετικό)

DISCLAIMER ON ACADEMIC ETHICS AND INTELLECTUAL PROPERTY RIGHTS

«Being fully aware of the implications of copyright laws, I expressly state that this diploma thesis, as well as the electronic files and source codes developed or modified in the course of this thesis, are solely the product of my personal work and do not infringe any rights of intellectual property, personality and personal data of third parties, do not contain work / contributions of third parties for which the permission of the authors / beneficiaries is required and are not a product of partial or complete plagiarism, while the sources used are limited to the bibliographic references only and meet the rules of scientific citing. The points where I have used ideas, text, files and / or sources of other authors are clearly mentioned in the text with the appropriate citation and the relevant complete reference is included in the bibliographic references section. I also declare that the results of the work have not been used to obtain another degree. I fully, individually and personally undertake all legal and administrative consequences that may arise in the event that it is proven, in the course of time, that this thesis or part of it does not belong to me because it is a product of plagiarism».

The declarant

Dimitrios Tozakidis

Diploma Thesis

Consulting mobile chat app

Dimitrios Tozakidis

Abstract

This thesis focuses on the development of a mobile application to address various challenges individuals face by leveraging personal life experiences. The scientific area of this study lies at the intersection of technology, human interaction, and problem-solving. The purpose is to create a platform that allows users to seek help, receive guidance, and find solutions to their specific problems. The thesis begins by examining existing solutions. A comprehensive analysis of the strengths and limitations of these approaches is conducted. In the second chapter, related work in the field is presented, exploring similar applications and their effectiveness. The subsequent chapter delves into the tools utilized to develop the proposed mobile application. The logical structure of the application is carefully analyzed, highlighting key components that facilitate problem-solving and effective communication. Then it focuses on the practical usage of the application. The core functionality is described, featuring a feed of topics submitted by users seeking assistance. Users with relevant experience and willingness to help can engage in one-to-one conversations. The implementation details and user interface considerations are discussed to ensure a seamless user experience. The main results of this thesis include the development of a mobile application that promotes effective problem-solving by leveraging personal life experiences. By enabling one-to-one interactions, users can receive personalized guidance tailored to their unique challenges. The application fosters a sense of community, connecting individuals with shared experiences. In the final chapter, the thesis concludes by summarizing the findings and discussing the implications of the developed mobile application. It emphasizes the importance of incorporating personal experiences to enhance problem-solving processes and highlights the potential for further improvements and future updates. Overall, this research presents a mobile application that provide a platform to seek help and find solutions to their problems. It showcases the potential for technology to foster collaboration and enhance the quality of life for individuals..

Διπλωματική Εργασία

Συμβουλευτική εφαρμογή συνομιλίας μέσω κινητού τηλεφώνου

Δημήτριος Τοζακίδης

Περίληψη

Αυτή η διατριβή επικεντρώνεται στην ανάπτυξη μιας κινητής εφαρμογής για την αντιμετώπιση διαφόρων προκλήσεων που αντιμετωπίζουν οι άνθρωποι, εκμεταλλευόμενη τις προσωπικές τους εμπειρίες. Ο επιστημονικός τομέας αυτής της μελέτης βρίσκεται στο σημείο τομής της τεχνολογίας, της ανθρώπινης αλληλεπίδρασης και της επίλυσης προβλημάτων. Ο σκοπός είναι να δημιουργηθεί μια πλατφόρμα που επιτρέπει στους χρήστες να ζητήσουν βοήθεια, να λάβουν καθοδήγηση και να βρουν λύσεις στα συγκεκριμένα προβλήματά τους. Η διατριβή ξεκινά με την εξέταση των υπαρχόντων λύσεων. Διενεργείται μια περιεκτική ανάλυση των δυνατοτήτων και περιορισμών αυτών των προσεγγίσεων. Στο δεύτερο κεφάλαιο, παρουσιάζεται σχετικό έργο στον τομέα, εξερευνώντας παρόμοιες εφαρμογές και την αποτελεσματικότητά τους. Το επόμενο κεφάλαιο εξετάζει τα εργαλεία που χρησιμοποιούνται για την ανάπτυξη της προτεινόμενης κινητής εφαρμογής. Η λογική δομή της εφαρμογής αναλύεται προσεκτικά, τονίζοντας τα βασικά στοιχεία που διευκολύνουν την επίλυση προβλημάτων και την αποτελεσματική επικοινωνία. Στη συνέχεια επικεντρώνεται στην πρακτική χρήση της εφαρμογής. Περιγράφεται η κύρια λειτουργικότητα, παρουσιάζοντας ένα ροή θεμάτων που υποβάλλονται από χρήστες που αναζητούν βοήθεια. Οι χρήστες με σχετική εμπειρία και προθυμία να βοηθήσουν μπορούν να συμμετέχουν σε μονομερείς συνομιλίες. Συζητούνται οι λεπτομέρειες υλοποίησης και οι σκέψεις σχετικά με τη διεπαφή χρήστη, προκειμένου να διασφαλιστεί μια ομαλή εμπειρία χρήστη. Τα κύρια αποτελέσματα αυτής της διατριβής περιλαμβάνουν την ανάπτυξη μιας εφαρμογής που προωθεί την αποτελεσματική επίλυση προβλημάτων με τη χρήση προσωπικών εμπειριών. Με τη δυνατότητα μονομερών αλληλεπιδράσεων, οι χρήστες μπορούν να λάβουν προσοποιοποιημένη καθοδήγηση προσαρμοσμένη στα μοναδικά τους προβλήματα. Η εφαρμογή προάγει το αίσθημα κοινονικότητας, συνδέοντας ανθρώπους με κοινές εμπειρίες. Στο τελευταίο κεφάλαιο, η διατριβή ολοκληρώνεται με τη σύνοψη των ευρημάτων και τη συζήτηση των προβλημάτων της ανάπτυξης της εφαρμογής. Τονίζεται η σημασία της ενσωμάτωσης προσωπικών εμπειριών για την ενίσχυση των διαδικασιών επίλυσης προβλημάτων και επισημαίνεται η δυνατότητα για περαιτέρω βελτιώσεις

και μελλοντικές ενημερώσεις. Συνολικά, αυτή η έρευνα παρουσιάζει μια εφαρμογή που παρέχει μια πλατφόρμα για την αναζήτηση βοήθειας και την εύρεση λύσεων σε προβλήματα. Αποδεικνύει την δυνατότητα της τεχνολογίας να προωθεί τη συνεργασία και να ενισχύει την ποιότητα ζωής των ατόμων.

Table of contents

Chapter 1

Introduction

The existence of technology is driven by a multitude of reasons, stemming from human needs, aspirations, and the desire to improve various aspects of life. One of the main reasons of its existence is to improve the quality of life. Technology plays a significant role in enhancing the quality of life for individuals and communities. It provides access to better healthcare, education, transportation, entertainment, and communication. From medical advancements to the Internet, technology has transformed and improved numerous aspects of daily life. Enhancing communication and connectivity while advancing knowledge and research can improve spectacularly human well-being. Technology enables people to connect and communicate across vast distances. From telephones and email to social media and video conferencing, technology has revolutionized the way we interact and share information, fostering global connectivity and collaboration. Moreover it facilitates scientific research, exploration, and discovery. Powerful computing systems, data analysis tools, and sophisticated instruments enable researchers to collect, analyze, and interpret vast amounts of data, leading to new insights and advancements in various fields. The goal of this thesis is intertwined with this ideology.

1.1 Subject of the thesis

The subject of this thesis is to help people with any problems or challenges that they may face. There are many ways to solve our problems. Some times we just don't have enough experience on a topic or we just lack expertise. We may had a bad day, needed help with cooking or just don't know how to use his computer. Of course there are many video on

Youtube with tutorials of how to use a feature of our computer, how to cook and how to exercise to regain confidence. Moreover there are many subreddits where people can share their thoughts and give their own advise to anyone's issue. Finally with the new app Wysa people can enjoy mental health support and emotional well-being assistance by a chatbot that combines artificial intelligence (AI) and empathetic techniques. All these apps are great but each one of them lack something important. Youtube provides little to no interaction between the helper and the person that in need. Subreddit has a forum format so there is limited one to one communication. Finally with Wysa, despite being a great technological achievement, is just a chatbot without any personal life experience. To solve these imperfections the application presented in this thesis relies on exactly this, personal life experience. Mobile phones nowadays are the extend of our hands. So my idea is to make a mobile application where people can find answers and seek help on any issues they face. The core functionality of the application is a feed of many topics that have specific tags. The topics are made from any user that has an issue and need help with it. All the users can view all posted topics, and regarding their experience on this specific issue and the good will to help, can give answers to the editor of the topic. Finally it starts a one to one conversation between the two users that can hold until the problem is solved.

1.2 Organization of the thesis

Related work about the subject of the thesis presented in the second chapter. In the fourth chapter I am looking into the tools of the thesis. Then I analyze the logic behind of the application and afterwards more specifically it's usage, in chapter five. Finally, in the last chapter, I conclude my thesis and I quote some of ideas for potential future updates.

Chapter 2

Related work

Subreddits form an essential component of the Reddit social media platform, contributing to a diverse ecosystem of online communities centered on specific interests, subjects, and conversations. Reddit has evolved into one of the largest and most influential online platforms, boasting millions of active users. Subreddits, in turn, play a pivotal role in shaping its distinct user experience.

Reddit, fundamentally, operates as a user-driven platform where individuals have the liberty to generate content, partake in discussions, and share links within a democratic and decentralized structure. Subreddits 2.1 act as miniature universes within this expansive ecosystem, providing focused arenas for users to connect with like-minded individuals who share their passions, hobbies, or areas of interest. By subscribing to pertinent subreddits, users can personalize their Reddit journey, curating the content they engage with to align with their preferences.

The appeal of subreddits lies in their capacity to accommodate a wide spectrum of topics and communities. Whether one's interests revolve around technology, science, movies, music, video games, fitness, or even more niche subjects like birdwatching or antique collecting, there is likely a subreddit dedicated to it. This decentralized approach empowers users to locate communities that resonate with them, cultivating a sense of belonging and facilitating meaningful interactions.

Each subreddit functions autonomously, governed by its own set of regulations and guidelines established by moderators who volunteer their time to uphold the subreddit's standards and cultivate a positive user experience. Moderators play a pivotal role in curating content, enforcing regulations, and overseeing discussions within their respective communities. They

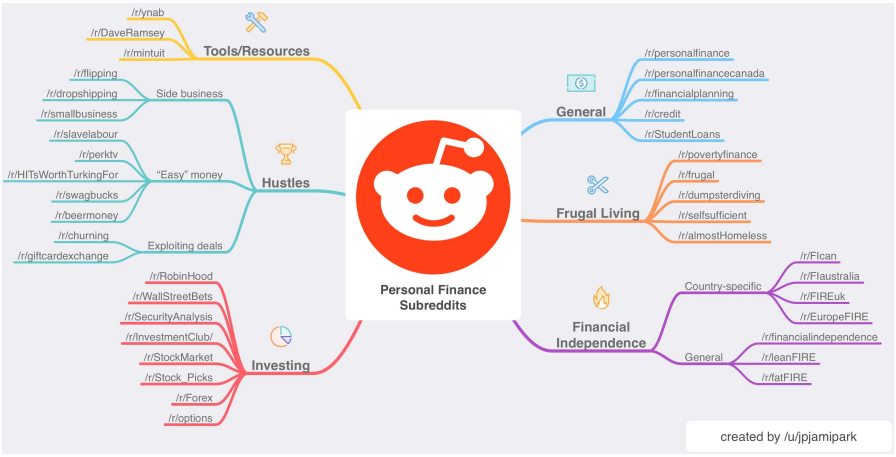

Figure 2.1: Famous subbredits in different categories

possess the authority to eliminate spam, enforce guidelines, and mediate conflicts, all in pursuit of creating a secure and inclusive environment for participants.

The popularity and engagement of a subreddit hinge on user activity and the quality of shared content. Posts that garner upvotes from users ascend in visibility and ascend the subreddit's feed, while those receiving downvotes or unfavorable feedback may be concealed or relegated to obscurity. This voting system, coupled with comment threads, fosters spirited discussions and furnishes a mechanism for the community to collectively curate content.

Subreddits also serve as valuable hubs for information aggregation and news coverage. Numerous subreddits revolve around current events, affording users the opportunity to debate and disseminate news articles, blog entries, or personal viewpoints on particular subjects. The assortment of opinions and backgrounds within these communities can offer a more comprehensive grasp of intricate issues.

In addition to discussions, subreddits frequently host "Ask Me Anything" (AMA) sessions, where individuals—ranging from celebrities and experts to individuals with unique experiences—respond to inquiries from the community. This format promotes direct interaction between users and prominent figures, fostering a sense of proximity and accessibility seldom encountered on other platforms.

Subreddits transcend the virtual realm as well. Many local communities establish their own subreddits dedicated to specific cities or regions, functioning as platforms for residents to discuss local news, events, and share recommendations. These geographically-centered subreddits can be invaluable resources for newcomers or those seeking connections in their locale.

In summary, subreddits serve as the foundational building blocks of Reddit, furnishing focused and vibrant spaces for individuals to connect, share, and deliberate on topics of interest. By embracing the principles of decentralization and user-driven content, subreddits have empowered millions of users to construct communities centered around their passions. They facilitate meaningful interactions, provide access to a multitude of perspectives, and furnish a platform for information sharing and cultural phenomena. Whether one's pursuit is knowledge, entertainment, or a sense of belonging, the expansive world of subreddits offers something for everyone.[1], [2]

2.1 Youtube videos

YouTube has brought about a transformation in the realms of learning, education, and skill enhancement by providing a dynamic and easily accessible platform for individuals to acquire knowledge and cultivate new competencies 2.2. Here are several key reasons highlighting YouTube's contributions in these domains:

1)YouTube boasts an extensive repository of educational material spanning a wide spectrum of subjects and disciplines. Whether one's interests lie in mathematics, science, history, literature, or any other field, YouTube offers a treasure trove of tutorials, lectures, and informative videos to complement the learning journey. This rich array of content ensures that learners can access resources tailored to their specific areas of interest and educational requirements.

2)Visual and Interactive Learning: YouTube's video format enables visual and interactive learning experiences. Many individuals find it considerably easier to grasp complex concepts when they can witness demonstrations, visual aids, and practical examples. With YouTube, learners can immerse themselves in step-by-step tutorials, visual elucidations, and simulations that significantly enhance comprehension and information retention.

3)Self-Paced Learning: One of YouTube's most commendable attributes is the flexibility it offers for self-paced learning. In stark contrast to traditional classroom settings or rigid course schedules, YouTube empowers individuals to learn at their own convenience and progress according to their preferred pace. Learners can pause, rewind, and revisit videos as often as needed to gain a thorough understanding of intricate topics.

4)Global Reach and Diverse Perspectives: YouTube's global platform serves as a meet-

ing ground for content creators and learners hailing from diverse corners of the world. This convergence fosters a vibrant tapestry of varied viewpoints, cultural insights, and alternative approaches to learning. Users have the privilege of exploring content generated by educators, experts, and enthusiasts representing diverse backgrounds, thereby nurturing a global perspective that encourages cross-cultural understanding and appreciation.

5)Practical Skill Development: YouTube serves as an invaluable resource for practical skill acquisition. Whether one aspires to learn a musical instrument, master a foreign language, acquire coding proficiency, enhance culinary skills, or refine artistic talents, YouTube offers a plethora of tutorials, guides, and how-to videos. This platform empowers individuals to acquire and hone real-world skills in a convenient and cost-effective manner.

6)YouTube's algorithm-driven recommendations and personalized playlists enrich the learning journey. Drawing insights from a user's viewing history and preferences, YouTube suggests related videos and channels aligned with their specific interests. This aids learners in discovering fresh content and constructing personalized learning pathways tailored to their unique goals and areas of focus.

7)Accessibility and Inclusivity: YouTube's accessibility features render it a valuable asset for individuals with diverse learning styles, abilities, and language preferences. Features such as captions, subtitles, and auto-translation options enable individuals with hearing impairments or language barriers to engage with educational content. Furthermore, YouTube's compatibility across various devices and network connections ensures that learning opportunities remain accessible to individuals from a wide range of socio-economic backgrounds.

8)YouTube's comment section facilitates engagement and collaboration among learners and content creators. Users can pose questions, seek clarification, and share insights, fostering a sense of community-driven learning. This interactive ambiance encourages discussions, knowledge exchange, and the formation of virtual study groups.

In summary, YouTube's extensive array of educational content, its provision for immersive visual and interactive learning experiences, self-paced learning flexibility, global accessibility, practical skill development resources, personalized learning recommendations, inclusive features, and community-oriented collaboration make it an indispensable tool for learning, education, and skill cultivation. It empowers individuals to broaden their knowledge horizons, acquire fresh skills, and engage with a diverse assortment of educational content to bolster both personal and professional growth.[3]

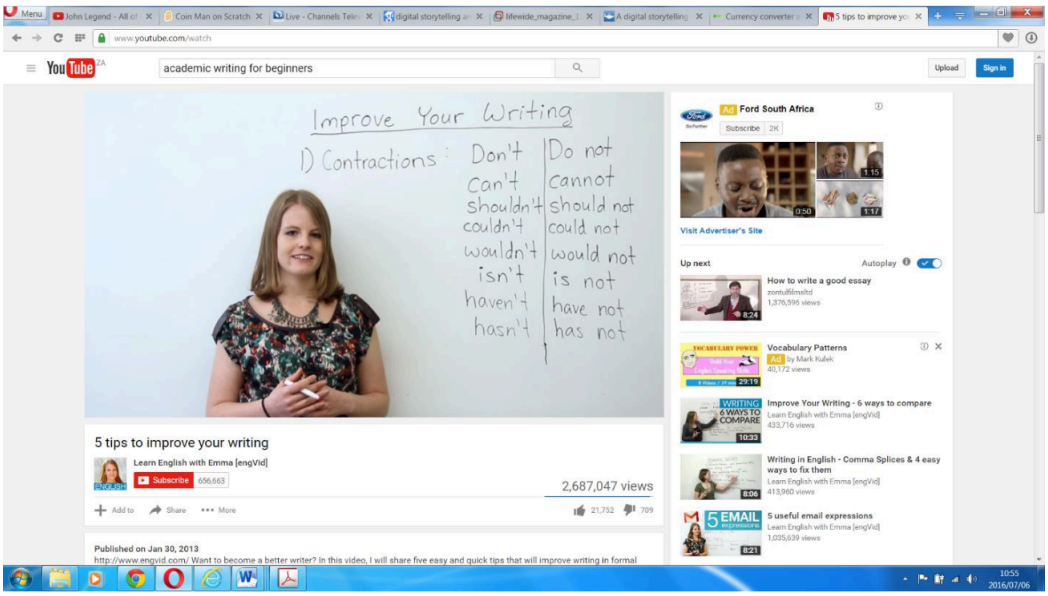

Figure 1: Illustration of one of the videos used to facilitate academic writing group

Figure 2.2: A youtube tutorial to improve your writing

2.2 Wysa chatbot

Wysa stands as a conversational AI chatbot, skillfully crafted to extend mental health assistance and emotional well-being support. Developed by Touchkin, Wysa seamlessly integrates artificial intelligence (AI) with empathetic methodologies, fostering a confidential sanctuary where users can freely articulate their thoughts, emotions, and concerns.

Wysa's central objective revolves around the enrichment of mental well-being, offering both emotional solace and introspective tools. The chatbot skillfully amalgamates research-backed techniques derived from a spectrum of therapeutic models, including cognitive behavioral therapy (CBT), dialectical behavior therapy (DBT), and mindfulness practices. Wysa's AI-driven algorithms meticulously scrutinize user inputs, orchestrating bespoke, empathetic, and non-judgmental responses.

The user-friendly chat interface makes Wysa accessible at any time and from any location. It's accessible through a mobile app 2.3, granting individuals the flexibility to interact with the chatbot discreetly and at their convenience. A friendly penguin avatar serves as Wysa's visual persona, fostering an inviting and approachable ambiance.

Through thoughtful dialogues, Wysa gently nudges users to delve into their thoughts and feelings, offering guidance during challenging moments and equipping them with coping strategies. It assists users in recognizing counterproductive thought patterns, challenging pessimistic thinking, and nurturing healthier outlooks. Furthermore, Wysa extends an array of

Figure 2.3: The Wysa opening screen

activities and exercises aimed at promoting relaxation, stress alleviation, and mindfulness.

Privacy and data security remain paramount for Wysa. It diligently adheres to stringent confidentiality protocols, refraining from storing any personally identifiable information. Users can confidently confide their emotions and apprehensions with the chatbot, assured of the safeguarding of their privacy.

Beyond its core functionality, Wysa has diversified its repertoire to encompass a myriad of self-help tools and resources. It boasts guided meditation sessions, breathing exercises, mood tracking features, and techniques for enhancing sleep quality. Users have the ability to set goals, monitor their progress, and receive motivational prompts.

While Wysa serves as a valuable ally for mental health support, it's crucial to underline that it should not be viewed as a substitute for professional therapy or medical counsel. Its purpose lies in complementing existing mental health services, furnishing a supportive and easily accessible resource for individuals seeking emotional well-being assistance.

In summation, Wysa emerges as an AI-driven chatbot, extending emotional support and introspective aids, thereby assisting individuals in fortifying their mental well-being. With its user-centric interface, evidence-informed strategies, and unwavering commitment to user privacy, Wysa provides a secure and flexible platform for users to explore their emotions, counter detrimental thought patterns, and cultivate healthier coping mechanisms.[4]

Chapter 3

The theory behind the app idea

3.1 The problem to be solved

In today's rapidly evolving technological landscape, there are numerous ways to harness the power of technology to enhance our lives. From fostering communication through email, instant messaging, video calls, and social media platforms to embracing online education via courses, educational apps, and digital libraries, we have witnessed a profound transformation in the way we interact with the digital world. Moreover, the realm of health and fitness has seen a significant shift, with wearable devices like smartwatches and fitness trackers aiding individuals in monitoring their health and fitness goals. Health apps have further empowered us to track our dietary habits, exercise routines, sleep patterns, and overall well-being.

Amid this technological abundance, the central question emerges: how can we consolidate these diverse facets of our digital lives to make them more accessible to everyday people? The answer to this question germinated from the realization that while there are many applications and platforms providing valuable assistance to users, there remains a gap in the market. An analysis of existing applications, as detailed in the previous chapter, unearthed certain shortcomings and inefficiencies. These imperfections prompted the conceptualization of an innovative solution, one that aspires to be both holistic and deeply personal.

The vision crystallized around the idea of developing an application that transcends the impersonality often associated with platforms like YouTube. It seeks to offer comprehensive guidance, eschewing the limitations of mere forums, as seen on platforms like Reddit. Additionally, this application aims to provide a distinctly human touch, differentiating it from AI-driven advisors such as Wysa. Thus, the quest for a truly exceptional and empathetic dig-

ital assistant began.

3.2 The solution

While the problem was apparent, devising a solution proved to be a creative journey. An unlikely source of inspiration emerged from observing the dynamics of Subreddit, a community-driven platform. It became evident that people have an innate desire to help one another, often without expecting any tangible rewards. This phenomenon of altruism, existing in a world occasionally marred by cruelty, offers a glimmer of hope.

A significant insight from personal experience underscored the importance of individuals who have personally navigated through specific situations. These individuals possess a unique depth of knowledge and empathy that can profoundly impact those currently facing similar challenges. Moreover, those who have successfully overcome adversity are often driven by a strong sense of compassion, having firsthand knowledge of the difficulties involved. This insight coalesced into the concept of an application where people could seek answers and advice from those with direct experience in the problem they wish to solve.

Users are empowered to create anonymous accounts, facilitating open and honest conversations. They can draft articles soliciting help for a variety of issues, inviting other users to engage in one-on-one chats until the problem is resolved. Detailed functionality of the application is elaborated upon in Chapter Five.

Moreover, it was evident that the application needed to be mobile-centric. In an age where modern smartphones have become ubiquitous, these devices are a constant companion for individuals, making them the ideal platform for accessing this valuable resource.[5], [6]

3.3 The target group

The envisioned application is poised to cater to a diverse target group of individuals seeking personalized guidance and support in various aspects of their lives. This target group includes:

1) Individuals in need of emotional support: My application aims to provide a more personal and empathetic approach compared to existing platforms. Therefore, the target group would include individuals who are seeking emotional support, advice, or guidance from oth-

ers who have experienced similar situations. This could encompass a wide range of personal or mental health issues.

2) Those seeking practical advice and solutions: My application intends to offer practical solutions and guidance based on the experiences of others. This would appeal to individuals who are looking for specific answers or advice on topics such as relationships, work, education, health, fitness, and other areas of personal development.

3) People who prefer one-on-one interaction: The one-to-one chat feature of my application would be attractive to individuals who prefer direct communication and a more personalized experience. They may appreciate the opportunity to have a private conversation with someone who has firsthand experience in dealing with similar challenges.

4) Individuals who value anonymity: My application allows users to create anonymous accounts, which would be appealing to those who wish to seek help without disclosing their identity. This would provide a safe and non-judgmental environment for people to open up about their problems and seek assistance.

5) Tech-savvy mobile users: Since my application is designed as a mobile application, the target group would consist of individuals who are comfortable using smartphones and are likely to have access to them throughout the day. This could include people of different age groups, as smartphones have become a ubiquitous device across various demographics.

In summary, the target group encompasses individuals seeking a supportive and personalized platform for seeking advice, guidance, and solutions to their problems, with a strong emphasis on human connection and shared experiences. The envisioned application strives to bridge the gap between technology and genuine human support, creating a community of empathetic individuals ready to assist each other on their respective journeys.

Chapter 4

The tools

This application could not be carried out, without the usage of the following valuable tools.

4.1 Android Studio

Android Studio stands as a potent integrated development environment (IDE) exclusively crafted for the creation of Android applications. Developed under the Google umbrella, it boasts an extensive array of tools and functionalities designed to streamline the app development process. Here, we will provide an overview of Android Studio, highlighting its core attributes and capabilities.

Android Studio presents a user-friendly interface 4.1.catering to both seasoned developers and newcomers alike. It offers a comprehensive toolkit that simplifies app development, including a code editor featuring intelligent code completion, syntax highlighting, and debugging prowess. These attributes empower developers to compose code with greater efficiency, reducing the likelihood of errors and bolstering overall productivity.

A standout feature of Android Studio is its advanced emulator. This emulator allows developers to rigorously test their applications on virtual Android devices, faithfully simulating a range of screen sizes, hardware configurations, and Android versions. This capability ensures that the app functions seamlessly across various devices, thereby mitigating compatibility concerns.

Another noteworthy aspect of Android Studio is its support for multiple programming languages. While Java has traditionally dominated Android app development, Android Studio

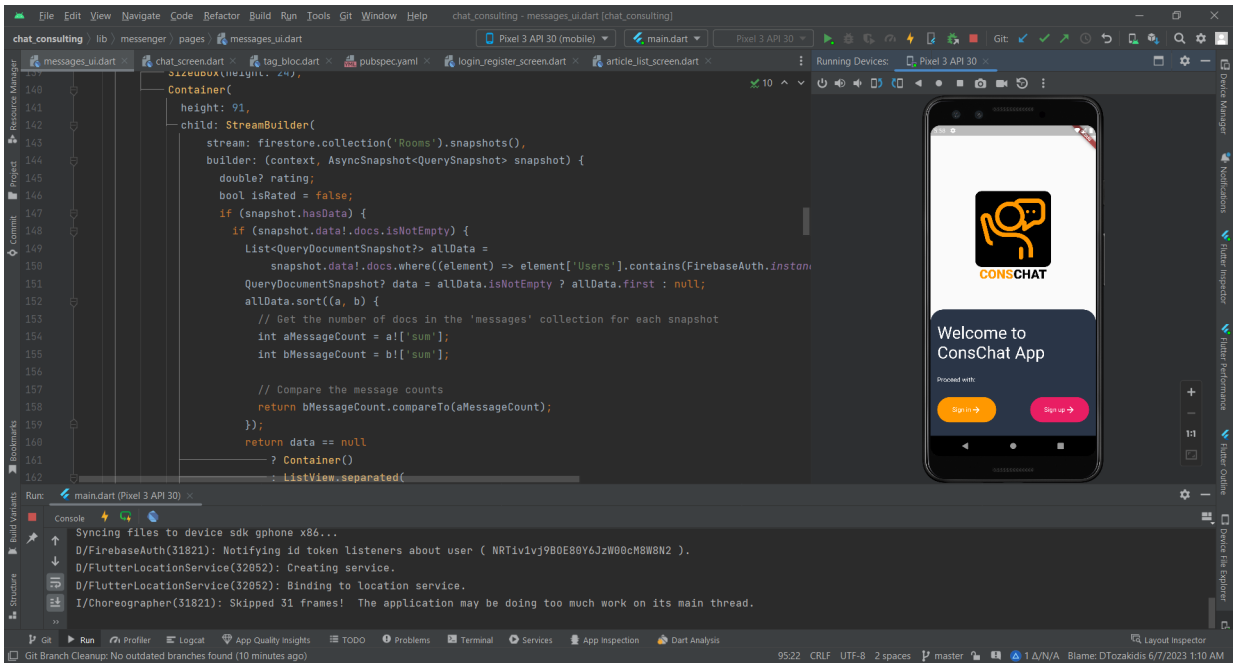

Figure 4.1: The android studio enviroment while developing the application

also embraces Kotlin and Dart. Developers are free to choose their preferred language based on their expertise and project-specific requirements.

Android Studio comes equipped with a suite of debugging tools that greatly assist in error identification and resolution. This includes a debugger enabling developers to step through their code and scrutinize variables, breakpoints, and memory usage. Furthermore, Android Studio offers performance profiling tools to optimize app performance and pinpoint potential bottlenecks.

The IDE seamlessly integrates with the Android Software Development Kit (SDK), which encompasses libraries, frameworks, and APIs, thereby equipping developers with a rich assortment of functionalities for their applications. Android Studio simplifies SDK management through an intuitive interface for installing, updating, and managing these critical components.

For collaborative development endeavors, Android Studio conveniently incorporates support for version control systems such as Git. This enables multiple developers to work on the same project concurrently, facilitating efficient collaboration and simplified code management.

Android Studio goes a step further in enhancing productivity by boasting an extensive plugin ecosystem. Developers can extend the IDE's capabilities by installing plugins that provide additional features like design templates, code generation, and integration with third-

party libraries.

To expedite app deployment, Android Studio offers a straightforward process for generating signed APKs (Android application packages) or app bundles. These packaged applications can be readily published on the Google Play Store or be distributed directly to end-users.

In conclusion, Android Studio emerges as a robust IDE thoughtfully tailored for Android app development. Its user-friendly interface, potent toolset, and extensive feature set render it an indispensable companion for developers. Whether one is embarking on their app development journey or is a seasoned expert, Android Studio equips them with the essential tools and resources to efficiently craft Android applications.

4.2 Flutter

Flutter 4.2 stands out as an open-source UI framework, thoughtfully crafted by Google, offering developers the capability to construct top-notch, native-like applications across various platforms, all from a single codebase. Its soaring popularity among developers owes itself to its adaptability, performance prowess, and user-friendly nature.

At its core, Flutter embraces a reactive programming model wherein the user interface takes shape through a blend of widgets. Widgets serve as the fundamental building blocks of a Flutter application, seamlessly combinable and nestable to craft intricate user interfaces. Flutter bestows an expansive array of pre-designed widgets tailored for diverse purposes like buttons, text fields, images, and more. These widgets come with a high degree of customizability, enabling developers to weave together distinctive and visually captivating interfaces.

A standout advantage of Flutter lies in its cross-platform agility. Developers wield the power to script code once and dispatch it across multiple platforms, encompassing iOS, Android, web, and desktop, thereby saving precious time and effort. This feat is accomplished through Flutter's adept use of a rendering engine that interfaces directly with the native components of each platform, resulting in performances that are high-caliber and uncannily akin to native experiences.

Flutter's hot-reload functionality emerges as another prominent feature. Developers can swiftly modify code and instantly witness the changes materialize in the app sans the rigamarole of recompilation or application restart. This iterative development process significantly

elevates productivity, expedites testing, and streamlines debugging.

Flutter also opens the gateway to a treasure trove of libraries and packages through its package manager, Pub. This diverse array of packages spans across a multitude of functionalities, encompassing networking, database integration, state management, and more, enabling developers to harness pre-existing solutions and fast-track development.

Moreover, Flutter is firmly aligned with Android's Material Design guidelines and iOS's Cupertino design guidelines, guaranteeing that Flutter-crafted apps mirror the native look and feel of each platform. This dedication fosters a harmonious user experience spanning various devices and platforms.

In terms of performance, Flutter orchestrates the compilation of apps into native machine code, effectively quelling the performance overhead traditionally associated with cross-platform frameworks. The outcome is fluid animations, nimble rendering, and a responsive user interface.

Flutter notches up its appeal further with commendable developer tools and an active community. The Flutter SDK is bundled with the robust integrated development environment (IDE) christened Flutter Studio, flaunting an array of features like code autocompletion and debugging. Meanwhile, the Flutter community thrives on knowledge sharing, package creation, and support provision across various forums and channels.

In summation, Flutter emerges as a versatile and potent framework for the development of cross-platform applications. Armed with an array of widgets, hot-reload capabilities, performance enhancements, and the backing of a vibrant community, Flutter has emerged as a preferred choice for developers aiming to efficiently craft top-notch applications. Its knack for delivering native-like experiences across an array of platforms cements its status as an excellent choice for businesses and developers keen on reaching a diverse user base.[7]

4.3 Dart

Dart stands as a programming language 4.3 crafted by Google, engineered to deliver speed, efficiency, and adaptability. It lies in the creation of high-performance applications spanning web, mobile, and desktop domains. Dart operates within the realm of object-oriented programming, sporting a C-style syntax that's both legible and writable. It presents a toolkit replete with features and utilities that streamline the development journey.

Figure 4.2: The flutter logo

A standout attribute of Dart lies in its capacity to be compiled into native code for a multitude of platforms. In essence, this implies that developers can craft code once and execute it across diverse platforms such as web browsers, mobile devices, and desktops. This intrinsic versatility establishes Dart as a formidable contender in the realm of cross-platform development, dramatically curtailing the investment of time and effort.

Dart flexes its muscles with a robust type system, wherein variables and expressions sport distinct types scrutinized during compile-time. This setup serves as an early error detection mechanism, elevating code reliability and maintenance ease. Yet, Dart also extends its support for type inference, sparing developers the necessity of invariably declaring types.

Another commendable facet of Dart revolves around its adeptness in asynchronous programming. Dart parlays the concept of futures and the async/await syntax to deftly manage asynchronous operations, be it sending network requests or reading from files. This framework empowers developers to craft responsive code that doesn't obstruct concurrent task execution.

Dart's utility is further augmented by its extensive standard library encompassing a gamut of functionalities ranging from collections to networking and file I/O. Complementing this is the presence of a package manager known as Pub, facilitating seamless library sharing and utilization, courtesy of the contributions from the developer community.

Dart serves as the language of choice for constructing applications utilizing the Flutter framework. The amalgamation of Dart and Flutter has been met with acclaim, thanks to their user-friendliness, hot-reload feature, and expeditious development cycles.

In summary, Dart emerges as a potent and adaptable programming language, armed with a bevy of features and tools for erecting high-performance applications. Its cross-platform

```dart
import 'package:flutter/material.dart';

Run | Debug
void main() {
  runApp(MaterialApp(
    home: Center(
      child: Text("Hello world"),
    ), // Center
  )); // MaterialApp
}
```

Figure 4.3: Simple Dart code that prints ”Hello world at the center of the screen

capabilities, robust type system, prowess in asynchronous programming, and comprehensive standard library bestow developers with a fertile ground for efficient and productive application development.[8]

4.4 Firebase

Firebase stands as a potent and all-encompassing platform crafted by Google, equipped with a plethora of tools and services engineered to aid developers in the construction and expansion of web and mobile applications. It offers a backend infrastructure that adeptly manages a spectrum of tasks, including data storage, authentication, hosting, and real-time synchronization, thereby affording developers the luxury of channeling their efforts towards the core functionality of their applications. In this elucidation, we shall delve into the salient features and advantages of Firebase.

At the heart of Firebase resides the Realtime Database 4.4, a NoSQL cloud-hosted database that empowers developers to store and synchronize data in real-time across multiple clients. It adopts a JSON-like data structure, rendering data organization and retrieval a straightforward endeavor. Leveraging the Realtime Database, developers can engineer applications that seamlessly propagate changes across diverse devices, delivering a fluid user experience.

Firebase Authentication emerges as another pivotal feature, bestowing applications with a secure and user-friendly authentication system. It extends support for various authentication methods, encompassing email and password authentication, social media logins, and even custom authentication mechanisms. By integrating Firebase Authentication, developers

can effortlessly oversee user authentication, authorization, and user account management, obviating the need to construct these components from scratch.

Cloud Functions occupies a prominent role within Firebase, furnishing developers with the capability to execute serverless code in response to events and triggers. This functionality streamlines task automation, data processing, and integration with other services, enabling the creation of potent serverless architectures without the encumbrance of server or infrastructure management.

Firebase also presents a formidable suite of tools catering to app testing, monitoring, and performance enhancement. Firebase Test Lab, for instance, affords developers the means to subject their apps to testing on real devices hosted in the cloud, ensuring compatibility and quality across various platforms. Firebase Performance Monitoring supplies in-depth insights into app performance, facilitating the identification and rectification of bottlenecks. Furthermore, Firebase Crashlytics extends real-time crash reporting, enabling developers to promptly detect and address issues, thereby upholding a stable user experience.

The Hosting feature of Firebase offers developers a dependable and scalable infrastructure for hosting web applications. It encompasses static and dynamic content hosting, automatic provisioning of SSL certificates, and integration with Content Delivery Networks (CDNs), ensuring swift and secure content delivery to users spanning the globe.

In summation, Firebase serves as an exhaustive platform, replete with a diverse array of tools and services designed to simplify the development and administration of web and mobile applications. It boasts robust features for real-time data synchronization, authentication, database storage, serverless computing, testing, monitoring, and hosting. By harnessing Firebase, developers can direct their energies toward molding the core functionality of their applications while relying on Google's sturdy backend infrastructure. Firebase's user-friendliness, scalability, and synergy with other Google Cloud services have propelled it into the echelons of popularity among developers aiming to expedite their application development endeavors and deliver top-notch user experiences.[9] [10]

4.4.1 Firestore

Firestore stands out as the quintessential tool offered by Firebase for my application's needs. Firestore, a NoSQL document database within the Google Cloud Platform's repertoire, plays a pivotal role in storing and harmonizing data across various devices and platforms. This

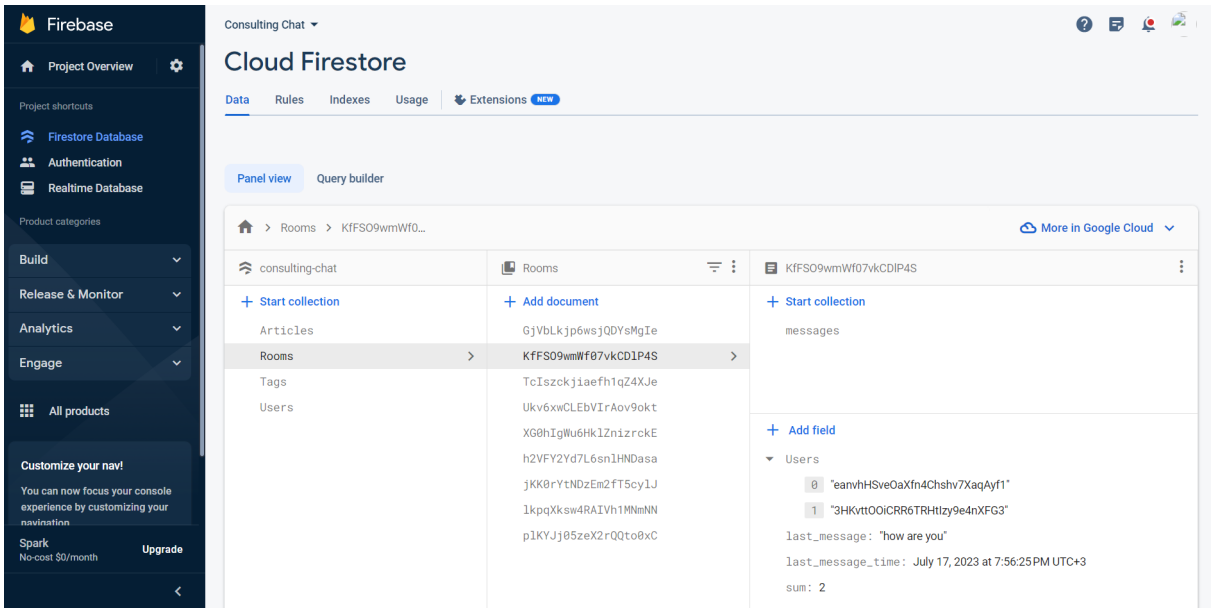

Figure 4.4: The Firestore Database of the application

makes it the perfect choice for constructing scalable, real-time applications.

Firestore adopts an organizational structure centered around collections, akin to tables in conventional relational databases. Within each collection, multiple documents can reside, with each document comprising a collection of key-value pairs, akin to JSON objects. Firestore employs a versatile data model, permitting the storage of intricate data structures and nested objects within documents.

A standout feature of Firestore is its prowess in real-time data synchronization. It leverages the concept of listeners, which can be configured to receive updates whenever data within a collection or document undergoes alterations. This empowers developers to craft responsive and collaborative applications, where changes effected by one user instantaneously propagate to other connected devices or users sharing the same dataset.

Firestore also bestows potent querying capabilities upon developers. It enables the retrieval of documents based on specific criteria, such as filtering based on field values or sorting results. This empowers efficient data retrieval and manipulation, even when dealing with extensive collections.

Furthermore, Firestore guarantees robust data consistency. It ensures that data modifications transpire atomically and in isolation, signifying that changes to multiple documents are either wholly applied or not at all. This bolsters data integrity and averts race conditions or conflicts.

Firestore seamlessly integrates with other Google Cloud services, encompassing Firebase Authentication, Cloud Functions, and Cloud Storage. This facilitates the construction of end-to-end solutions by harnessing supplementary services for authentication, serverless computation, and file storage.

Firestore also fortifies security features. It allows for the establishment of granular access controls to specify who possesses read, write, or modification privileges over documents or collections. Firestore actively supports Firebase Security Rules, a declarative language for delineating security policies, enabling the imposition of fine-grained authorization rules rooted in user roles and attributes.

Firestore scales adeptly and autonomously manages the foundational infrastructure necessary to cater to the demands of your application. It adeptly manages high read and write loads, assuring optimal application performance even under the burden of heavy traffic.

In summation, Firestore emerges as a commanding and adaptable NoSQL document database, renowned for real-time synchronization, formidable querying capabilities, data integrity, and effortless integration with other Google Cloud services. It empowers developers to craft applications that are both scalable and responsive while delivering robust security and scalability attributes.

Chapter 5

The application’s logic

5.1 The devices i used for application testing

The development and testing of the application by making use of two devices. I used it on physical device and on an emulator . The emulator is a feature of the Android Studio that provides real time testing without requiring the physical device. However it is not as stabilized as a physical device is. Therefore at the final stage of the application was also run on a physical device to be sure that everything works as intended.

5.1.1 Google Pixel 3

First I used the Google Pixel 3 device 5.1 inside of my emulator. The Google Pixel 3 is a smartphone released by Google in October 2018. When running the Google Pixel 3 emulator, it aims to replicate the software and user experience of the physical device on a computer. While I cannot provide a real-time demonstration, I can describe the general features and specifications of the Google Pixel 3. The Pixel 3 runs on the Android operating system, specifically the stock Android version that is developed by Google. The emulator would replicate the specific version of Android that the Pixel 3 was running at the time of its release. The Pixel 3 is powered by a Qualcomm Snapdragon 845 processor and 4 GB of RAM, which ensures smooth performance and multitasking capabilities. The emulator would aim to replicate this hardware configuration to provide a similar experience. As a flagship device from Google, the Pixel 3 offers a range of software features. This includes Google Assistant integration, the Pixel Launcher, and access to various Google services and apps. The emulator would strive to replicate these features, allowing users to interact with Google

Figure 5.1: The google pixel 3 device

Assistant and other software components. It supports various connectivity options, including Wi-Fi, Bluetooth, NFC, and USB-C. The emulator would emulate these connectivity options, allowing simulated interactions with these features. It's important to note that the emulator provides a simulated environment and may not perfectly replicate the exact user experience of using a physical Pixel 3 device. However, it allows developers and users to test and interact with the software and features of the Pixel 3 without requiring the physical device.

5.1.2 Redmi Note 9 Pro

The physical device I used was the Redmi Note 9 Pro 5.2 a smartphone produced by Xiaomi under its Redmi sub-brand. Released in March 2020, it offers a range of features and specifications at an affordable price point. Here's a description of the Redmi Note 9 Pro based on the features we need for our application: Design: The Redmi Note 9 Pro features a modern and sleek design with a glass front and back, protected by Gorilla Glass 5. It has a large 6.67-inch IPS LCD display with a resolution of 2400 x 1080 pixels. The phone has slim bezels and a centrally positioned hole-punch cutout on the front for the selfie camera. The back of the device houses a quad-camera setup and a fingerprint sensor. Performance: Under the hood, the Redmi Note 9 Pro is powered by the Qualcomm Snapdragon 720G chipset,

which offers excellent performance for everyday tasks and gaming. It is available in different memory configurations, typically ranging from 4GB to 6GB of RAM and 64GB to 128GB of internal storage. The device also supports expandable storage via a microSD card. Battery and Charging: One of the standout features of the Redmi Note 9 Pro is its massive 5,020mAh battery, which ensures long-lasting usage on a single charge. The phone supports 18W fast charging, allowing you to quickly replenish the battery when needed. It also has a USB Type-C port for charging and data transfer.Software and Features: The Redmi Note 9 Pro runs on MIUI, Xiaomi's custom Android-based operating system. MIUI offers a range of customization options and features, including a system-wide dark mode, gesture navigation, and various pre-installed apps. The device also includes a headphone jack, an IR blaster for controlling compatible devices, and a dedicated microSD card slot for expandable storage. Overall, the Redmi Note 9 Pro offers a balanced combination of design and performance, making it a popular choice for users seeking a budget-friendly smartphone with good features. As I already mentioned this application made with flutter, so with one codebase, it is available for both android and ios applications. Though I have no access to an ios device for testing. That's why the release of the application is only for android devices.

5.2 Application's Architecture

5.2.1 Firebase Auth

The architecture of this application is based on some very important key points. Starting from the main screen, the authentication is made with the library Firebase Auth. Firebase Auth provides a robust and secure authentication system for Flutter apps. It allows users to sign in or sign up using various authentication methods such as email and password, phone number, Google Sign-In, Facebook Login, and more. It offers seamless integration with Flutter apps, providing a simple and straightforward way to implement authentication features. The Firebase Auth package in Flutter provides a set of APIs and widgets that make it easy to handle user authentication. Also, it handles user authentication securely, implementing industry-standard encryption and security measures to protect user data. It ensures that user credentials are encrypted during transmission and securely stored in the Firebase backend. It offers various customization options, allowing developers to tailor the authentication flow to their app's specific requirements. It supports custom UIs, email templates, and user man-

Figure 5.2: The Redmi note 9 pro device

agement features, providing flexibility and control over the authentication process. Finally simplifies the management of user authentication states in Flutter apps. It provides built-in methods and callbacks to handle login, logout, and user state changes, making it easier to handle different app scenarios and user experiences. Overall, Firebase Auth in Flutter provides a powerful and flexible authentication solution, enabling developers to implement secure user authentication quickly and efficiently in their Flutter apps.[11]

5.2.2 Bloc

Fundamental role in the app's development was the design pattern bloc (short for Business Logic Component) that is used for managing state and handling user interactions. It follows the principles of separation of concerns and helps in organizing and maintaining clean code architecture. The most critical feature is that bloc separates the business logic and state management from the user interface. It helps in maintaining a clear separation between UI components and the logic that operates on the data. A bloc receives events or actions from the

UI and processes them to produce states. Events represent user interactions or other triggers, and states represent the resulting state of the application. Blocs typically use streams to communicate events and states. The bloc emits new states as the application's state changes, and the UI updates accordingly. The state in a bloc pattern is often represented by immutable data classes. Whenever a state change occurs, a new instance of the state class is created to reflect the updated state. Blocs promote testability by decoupling business logic from the UI layer, making it easier to write unit tests for the logic. Additionally, blocs can be reused across multiple parts of an application, reducing code duplication and improving maintainability. Flutter provides packages like flutterbloc and bloc that offer ready-to-use implementations of the bloc pattern. In the application I used flutterbloc. These packages provide utilities and base classes to simplify the implementation of blocs in Flutter applications. In conclusion, by using the bloc pattern, Flutter developers can effectively manage application state, handle user interactions, and maintain a structured codebase that is easier to test and maintain.[12], [13]

5.2.3 Firestore

The feature of the chat, a feature that relies on the whole success of the application, was held with the help of Cloud Firestore. Cloud Firestore is a NoSQL document-based database provided by Firebase, a backend platform by Google. It offers real-time data synchronization and offline capabilities, making it suitable for building reactive and collaborative mobile and web applications. In the context of Flutter, Cloud Firestore provides automatically synchronized data in real time across multiple devices, ensuring that changes made by one device are immediately reflected on all other connected devices. This enables real-time collaboration and updates. It offers offline support, allowing your Flutter app to access and modify data even when there is no network connection. The local changes made by the app are automatically synchronized with the server once a network connection is reestablished. You can perform complex queries on your data using Cloud Firestore's querying capabilities. It supports queries based on multiple fields, sorting, filtering, and pagination. This allows you to retrieve specific subsets of data efficiently. Cloud Firestore organizes data into documents, which are stored in collections. You can model your data in a hierarchical structure and easily create, update, and delete documents and collections using Flutter and Firestore's API. Also it provides built-in security rules that allow you to control read and write access to your

data. Additionally, it integrates with Firebase Authentication, enabling you to authenticate users and enforce fine-grained access control based on user identity. On of the lost important features that I used planty times throughout the process of the application was listeners and snapshots. With this feature, you can listen to changes in the database and receive real-time updates as data changes. This allows you to build reactive Flutter UIs that automatically update when the underlying data changes. Furthermore, it scales automatically to handle large amounts of data and concurrent users. It provides strong consistency guarantees and offers automatic data backups, ensuring the reliability and durability of your data. By leveraging Cloud Firestore in my Flutter app, I built a responsive, collaborative, and scalable application that can work both online and offline while benefiting from real-time data synchronization.

Chapter 6

The apps usage

6.1 Welcome page

First of all, the user must own an android device. With this device he is able to install the application. With the completion of the installation, he can now use the application. The first time he opens the application, he faces the welcome screen of “ConsChat” 6.1, that is the name of the app. The app welcomes him and offers two choices. The user can either create a new account or login with an existing account 6.2. If he select the first option, he must provide a username, his personal email and a password. The password must have at least 6 characters and the email address must be in the correct form and cannot already used by another another user. By selecting the second option, he must provide his credentials in order to login successfully, his email and password. After completing any of this actions, he can press the enter button to navigate inside the application. If there is an issue with his credentials, an appchip will inform him that something went wrong with the specified error message.

6.2 The option of chatting

6.2.1 Messages screen

After connecting successfully, the user gets navigated to the messages screen 6.3. There he can view any messages that have been exchanged with other users. If it is the first time he enters the application, he will see a message that informs him that he hasn’t exchanged

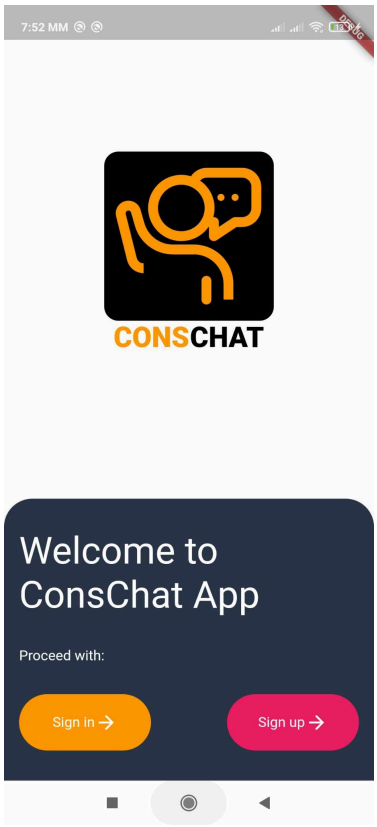

Figure 6.1: The welcome page

any messages yet. The chat screen consists of the ‘favorite users’ and the latest messages. On the row of the favorite users, he will see them in descending order on the number of the exchanged messages. Every user on the favorite list has a rounded avatar with the first letter of the user’s name inside. Has under the avatar there is his full name. On the list of the latest messages every user button consists of four items. First there is the same avatar with the first letter inside, followed by the user’s name. Under the user’s name he can read the last message that had been exchanged inside that chat room. Finally at the top of each message he can see the exact time that the last message was sent.

6.2.2 Chat screen

The user can now press any button either from the favorite or the last messages list. This will navigate him inside a chat room. Every chat room consists of two users. On the top of the screen the current user can preview the name of the other user of the room. Under the name the user can see the rating of this user in stars. The rating range is one to five and is the average of the ratings other users gave to this one 6.4. If the current user hasn’t rated

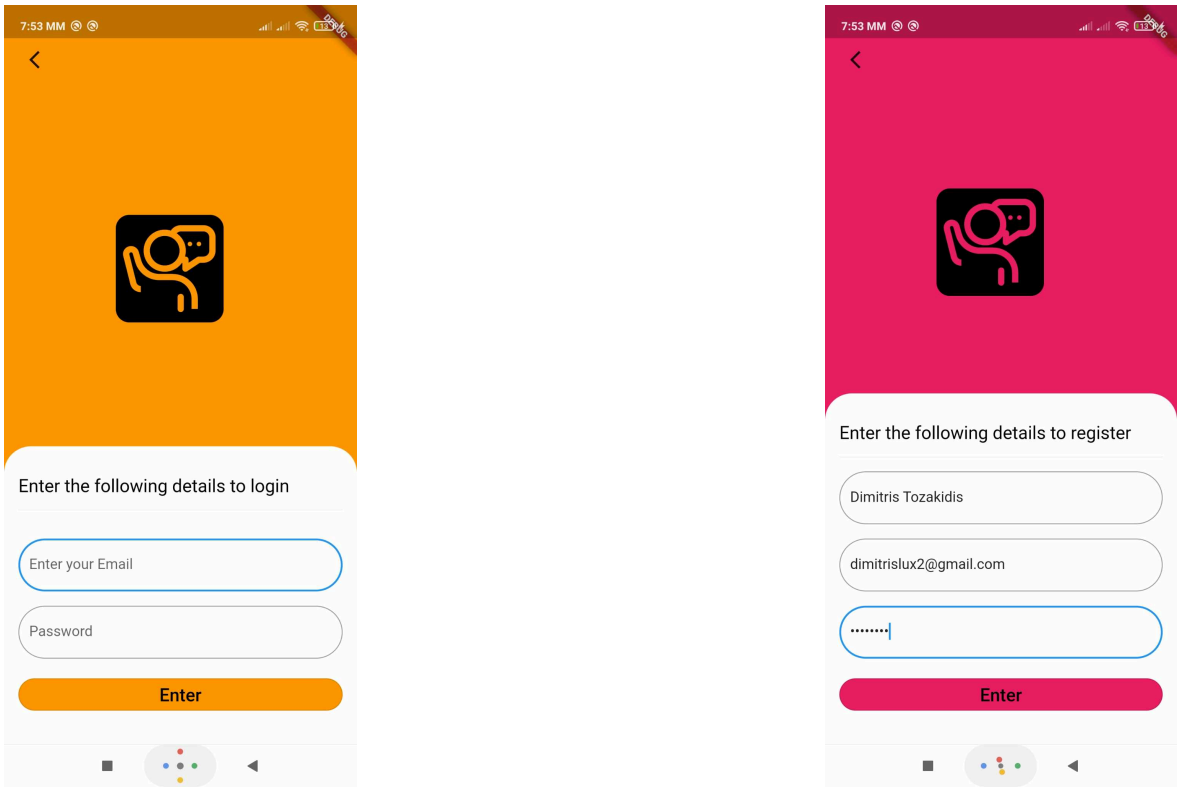

Figure 6.2: Login and register page

this user yet, then the stars will have yellow color and he is able to rate him between one to five stars. After rating him the stars become blue and the rating gets locked. The next time he will enter the room the rating to be blue and changed with the new average rating. Under the rating bar, the current user can preview the messages of this chat room. On the left are the other user messages and, on the right, his own messages. Over every message, there is the time that the specific message was sent. In the case of two or more messages that have the same sending time, the sending time is only visible on the first message. On the bottom of the screen, there is a text field where the user can type a new message. On the right is the send button, which he can press to send the message. The messages are updated live by the use of a stream provided by firestore. All the information for the specific room is stored inside a document of the collection Rooms. It consists of the two users id, the last message, the last message time , the amount of messages and a collection called messages. Inside this collection is a document for each message. This document has the message, the time that got sent and the sender id.

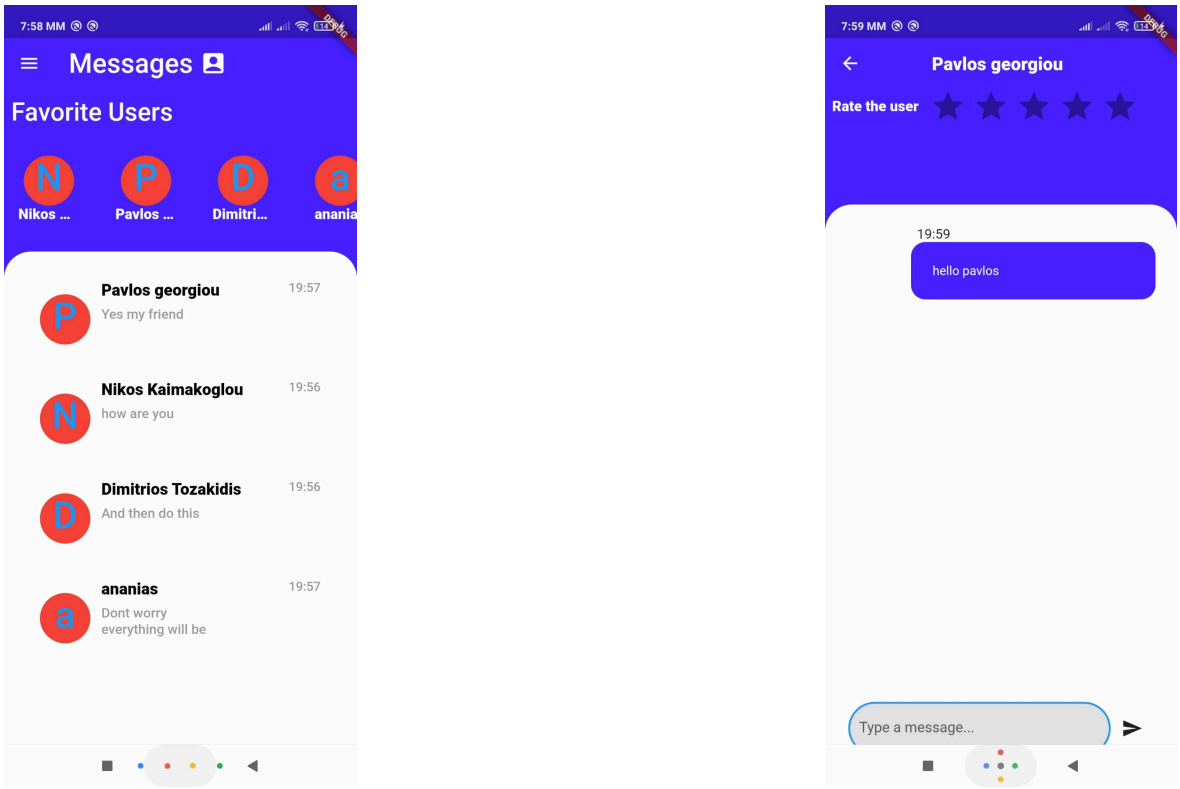

Figure 6.3: The messages screen and a chat room with user that has no ratings

6.3 Profile and log out

On the top left of the messages screen is the menu button. By pressing this button, the user can preview the options he has inside the application. After the messages option comes the profile option 6.5. Inside this screen the user can see his personal name so that he knows how other people see him. Another option inside this screen is to sign out or delete his account permanently. The last option for the user is the Logout button. By pressing this button the user navigates to the opening screen of the application. Now he can again Sign in or create a new account. But before the sign out option there is the most important option, the ads.

6.4 The ads

6.4.1 Ads screen

Ads is the main feature of this application. By selecting this option, the user navigates to the screen with all the ads 6.6. On the top of the screen, there is a text field. There the user can search to find any ad inside the application. The key factor of the search is the title of

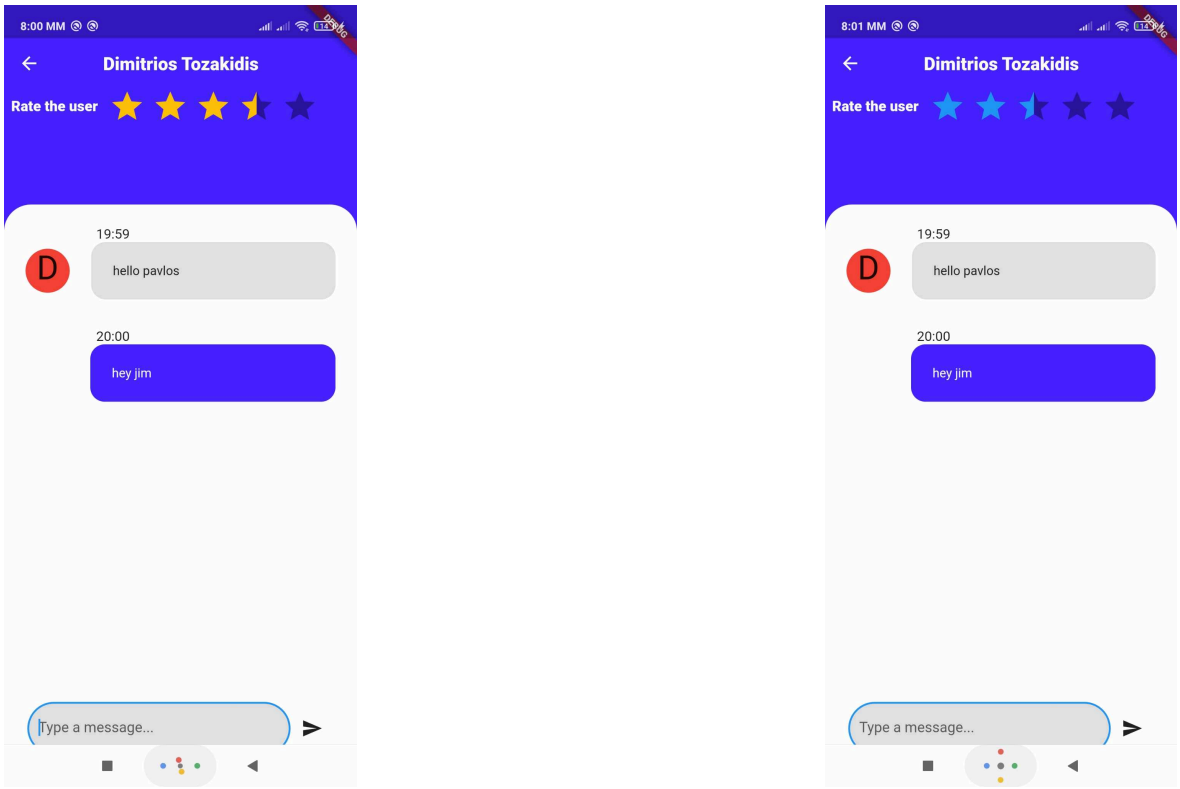

Figure 6.4: On left a chat room with user that has 3.5 rating on the right the current user just rated him with 2.5 stars

every ad. Following the search bar, come the tags. The tags are specific sections that any user can set to his ad. Some examples are sports, anxiety, cooking, technology etc. Every user who creates an ad can set any number of tags inside that ad so other users can understand what this ad is related to. All the existing tags are inside a horizontal list. There a user can select any amount of them in order to preview only ads with the selected tags. The search bar and the filter tags feature, work alongside. Under the tags comes another list but this time vertical. Inside that list there are all the existing ads, in alphabetic order. Every ad consists of a rounded container. This container is dark orange if nobody read this ad and is light orange otherwise. On the top left of the container, is a picture that describes the first tag selected by the author of this ad. Next to this picture is the title of the ad and under that just a part of the text inside this ad that describes the author's issue. On the top right of the container, there is a icon of a trashcan. If the ad doesn't belong to the current user this button is disabled. Otherwise by pressing this button, it opens a dialogue that asks the user if he wants to delete this article 6.7. By selecting the "yes" button the article gets deleted. Lastly on the bottom left, there is a text button that offers the author to edit this ad. The author by pressing it can

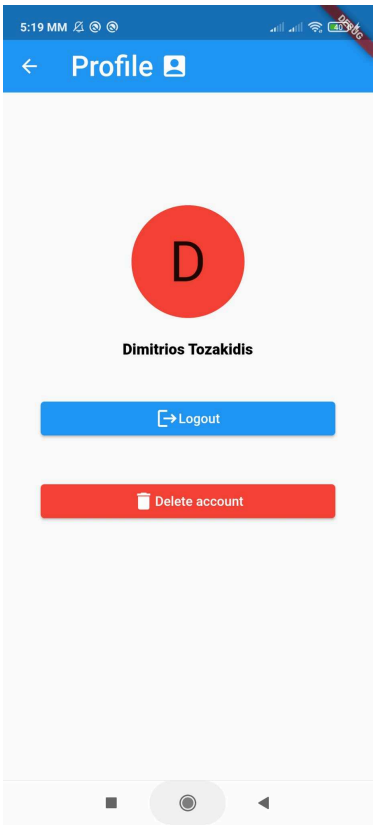

Figure 6.5: The profile screen

change the title, the description and the selected tags of this article. Finally on the bottom right of the screen, there is the elevated button with the plus icon. There the user can add a new article by entering a title, a description and select the tags of this ad. The ad must have a title and at least one tag, the description is optional.

6.4.2 Opened ad

Every ad container is tappable, despite the author. By pressing a container, the user gets navigated to a screen with the ad in full screen 6.8. Inside this screen a user can preview all the information of the ad. If the user is the author of the ad he opened, he can press the menu button on the top right. This opens menu that provides the option add new add, edit this add and delete this ad. The top half of the screen consist of the image that describes the first and most important tag of this ad. Over this image, on the left is the avatar of the author. Under the avatar the user can see the current date. Then comes the title of the ad, with a big font. Finally, the last object that is above the image is the name of the author. Under the image now, there are the tags of this ad. The tags again are tappable and if the user press it get navigated

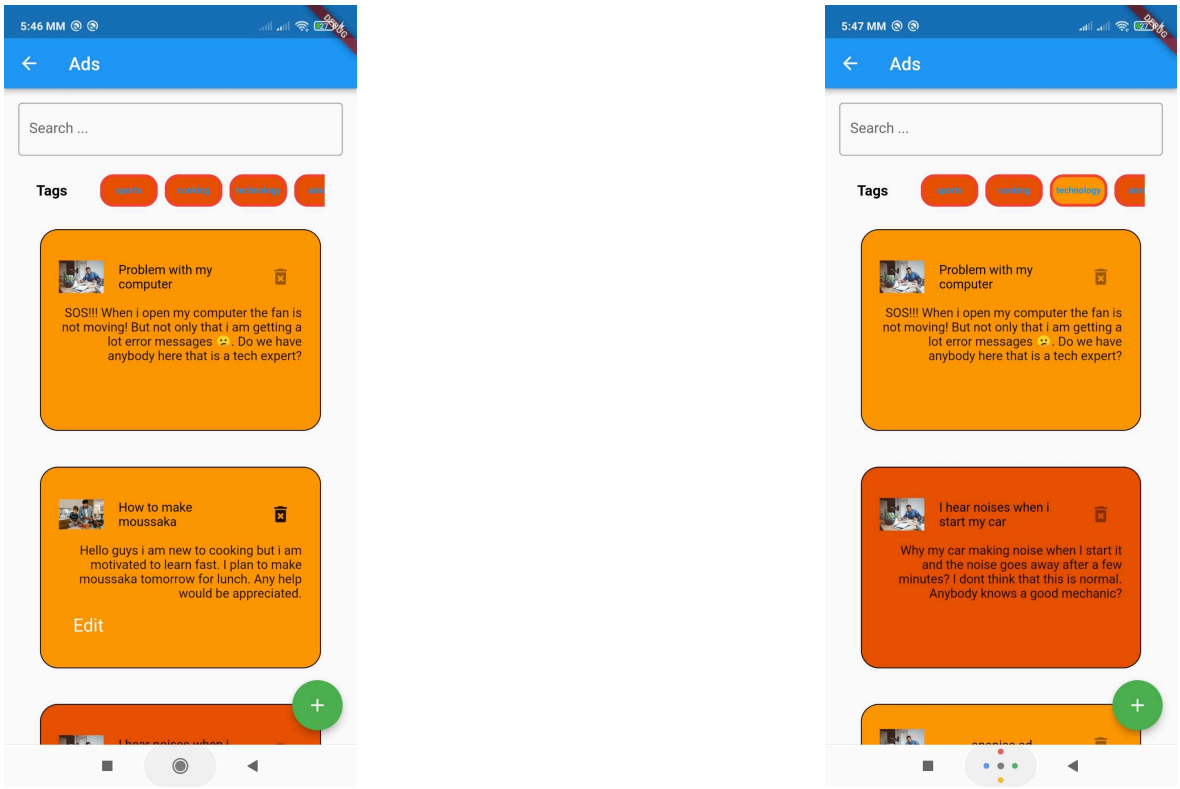

Figure 6.6: The ads screen. On the right the filter technology selected

to the previous screen with the selected tag already selected as a filter. The most important content is under the tags, that is the description. The user can read the ad carefully and only if he wants to help the author, can press the answer button on the bottom right. This button navigates him to a chat room with the author of the specific ad, and there the conversation begins. Every ad document consists of the title, the description, the isRead Boolean, the list with the tags, the writer's name and finally the writer id.[14], [15]

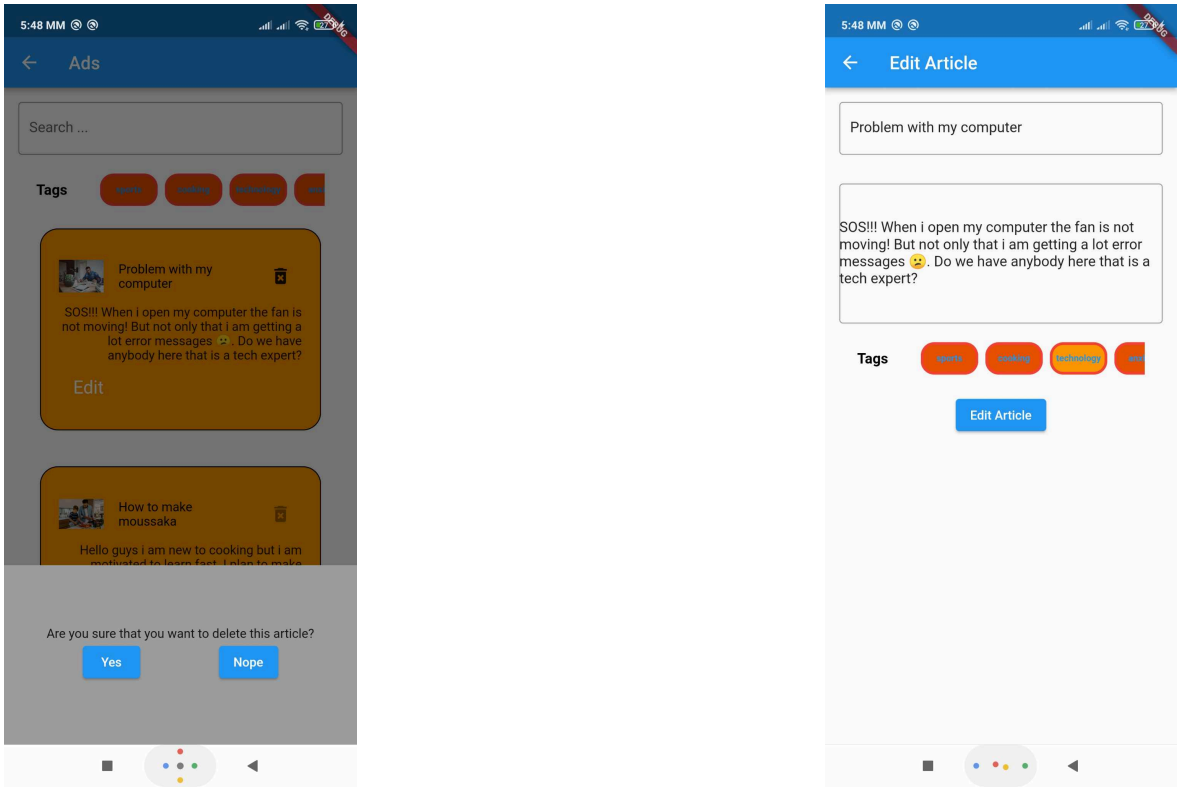

Figure 6.7: On left the delete action dialog. On the right the edit screen

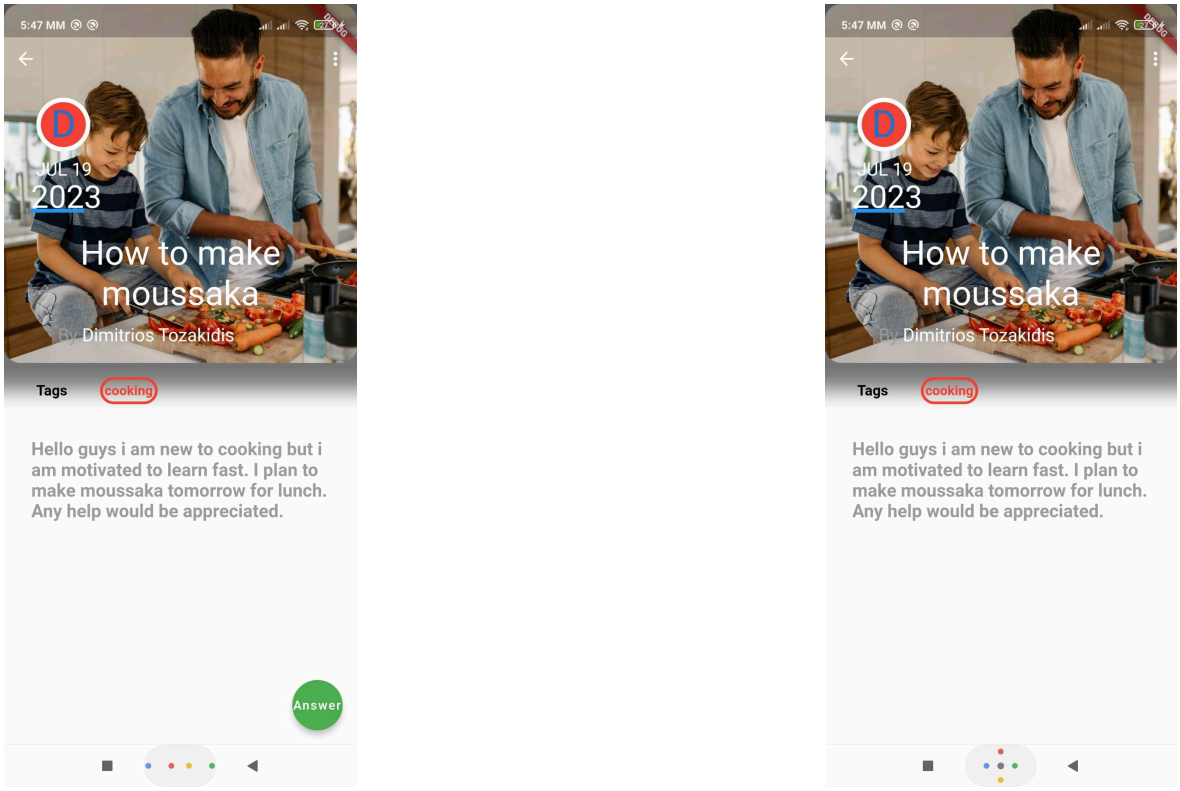

Figure 6.8: On left the preview of an opened ad. The right picture is the preview i see, if the ad is mine

Chapter 7

Conclusion

7.1 Summary and conclusion

Summarizing, the study focused on the development of a mobile application called ”ConsChat,” which aimed to provide a platform for anonymous chatting and ad posting. The application utilized various tools and technologies, including Android Studio, Flutter, Dart, Firebase, and Firestore, to create a robust and user-friendly application.

Throughout the development process, several key features and functionalities were implemented. These included authentication using Firebase Auth, state management using the Bloc pattern, real-time chat functionality using Firestore, and a comprehensive ad system. The application’s logic was designed to provide a seamless and intuitive user experience, with different screens and options for chatting, viewing ads, and managing profiles.

The testing phase involved using physical devices and emulators to ensure the application’s compatibility and functionality across various platforms. The Google Pixel 3 emulator and the Redmi Note 9 Pro physical device were used for testing, providing insights into the application’s performance and user experience on different devices.

Based on the development and testing process, several conclusions can be drawn. The application successfully implemented the desired features, allowing users to create accounts, chat anonymously, post ads, and manage profiles. The utilization of Firebase and Firestore facilitated real-time data synchronization and secure authentication, ensuring a reliable and efficient application.

Positive conclusions can be drawn regarding the usability and functionality of the application. The user interface was designed to be intuitive and user-friendly, allowing users

to navigate through different screens and access various features easily. The integration of different technologies and tools, such as Flutter and Firestore, contributed to the seamless performance of the application.

However, some challenges and limitations were also encountered throughout the study. The lack of access to an iOS device limited the application's availability to Android users only. Additionally, the emulator used for testing, while helpful, may not perfectly replicate the user experience of a physical device. These limitations could be addressed in future iterations of the application.

In conclusion, the thesis made a significant contribution to addressing the problems outlined in the introduction. The development of the ConsChat application provided a platform for anonymous chatting and ad posting, offering users a secure and user-friendly experience. The utilization of various technologies and tools showcased the potential of Android Studio, Flutter, Dart, Firebase, and Firestore in developing robust and efficient mobile applications. The study's positive outcomes and insights lay the foundation for further improvements and advancements in anonymous chatting and ad systems.

7.2 Future extensions

During the development process found some ideas for further extension of this application. One way to extend the diploma project is to develop compatibility for iOS devices. Currently, the application is only available for Android devices. By expanding its compatibility to iOS, you can reach a wider user base and provide a seamless experience for iOS users. This would involve adapting the application code and ensuring compatibility with the iOS platform.

Another extension idea is to allow users to add a profile picture or avatar to their accounts. This would enhance personalization and user identification within the application. Users could upload a picture from their device or choose from a selection of pre-defined avatars. Implementing this feature would involve integrating image uploading functionality and updating the user interface to display avatars alongside user profiles and chat rooms.

Currently, the application supports text-based messaging. To enhance the user experience and offer more communication options, adding voice chat functionality could be a valuable extension. Voice chat would allow users to have real-time conversations using audio instead

of solely relying on text messages. This would require integrating voice recording and playback capabilities, as well as incorporating user interface elements for initiating and managing voice chat sessions.

To improve the ad posting and browsing experience, implementing a recommendation system based on user activity and preferences could be an effective extension. The recommendation system would analyze user behavior, such as the ads they view, the tags they interact with, and their messaging history. Based on this data, the system could suggest relevant ads to users, increasing the likelihood of finding relevant content. Developing a recommendation system would involve data analysis, machine learning algorithms, and integrating the recommendation engine into the application's backend.

These extension ideas would enhance the functionality, user experience, and reach of the application. By expanding compatibility to iOS devices, allowing avatar pictures, incorporating voice chat, and implementing a recommendation system, the diploma project could offer a more comprehensive and personalized platform for anonymous chatting and ad posting.

Bibliography

[1] https://www.reddithelp.com/en/categories/reddit-101/reddit-basics/what-are-subreddits subreddits: What they are and how to use them.

[2] Studying reddit: A systematic overview of disciplines, approaches, methods, and ethics nicholas proferesnaiyan jonessarah a. gilbertcasey fieslermichael zimmer.

[3] Youtube as a source of information about physical exercise during covid-19 outbreak r. vanciniricardo borges viana+4 authors b. knechtle.

[4] Evaluating the therapeutic alliance with a free-text cbt conversational agent (wysa): A mixed-methods study c. beattyt. maliks. mehelic. sinha.

[5] The impact of social media usage on depression cognition and help-seeking behavior: A study based on grounded theory yijing guo, yunjuan cai.

[6] What triggers online help-seeking retransmission during the covid-19 period? empirical evidence from chinese social media c. luo, yuruo li, anfan chen, yulong tang.

[7] https://flutter.dev/ the official flutter website provides documentation, tutorials, and resources for learning and developing with flutter.

[8] https://dart.dev/ the dart website provides documentation and resources for learning the dart programming language.

[9] https://firebase.google.com/docs the official firebase documentation offers comprehensive information about firebase services, including firestore and firebase authentication.

[10] https://www.packtpub.com/product/firebase-essentials-second-edition/9781789809184 this book covers essential firebase concepts and features, including firestore and firebase authentication.

[11] Designing mobile-based chat application k. a. nugraha, restyandito, danny sebastian, nicholas christianto wijaya.

[12] https://pub.dev/packages/flutter_bloc the flutter bloc package documentation provides information on implementing the bloc pattern in flutter applications..

[13] Bloc design pattern priya tyagi.

[14] https://material.io/resources/icons/ material icons - material design.

[15] https://www.geeksforgeeks.org/user-interface-ui-design-in-software-engineering/ user interface (ui) design in software engineering.