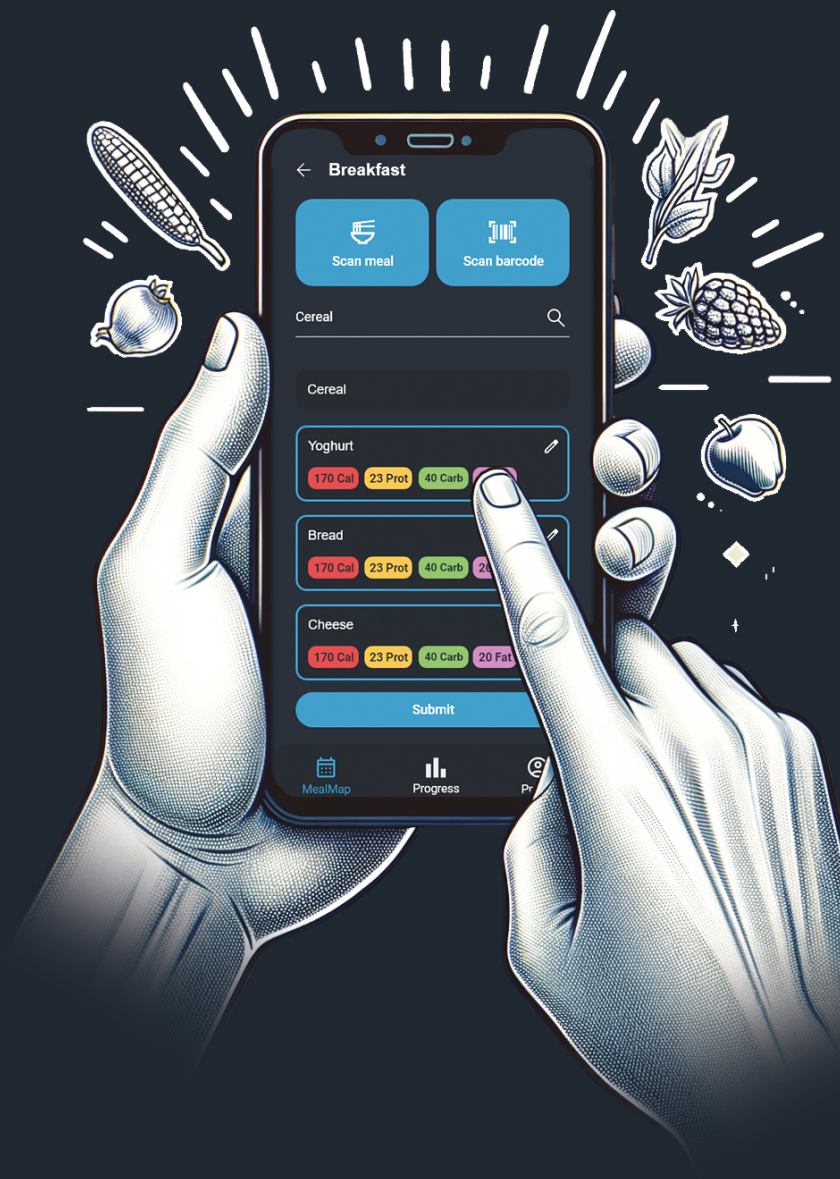
Final work

Erasmushogeschool Brussel
Multimedia & Creative Technology

2023 - 2034

Nourify



Promotor
Johan Van Den Broek

Author
Izzet Sen

Place of publication
Brussels

Preface

This report is about my bachelor's thesis for my degree in Multimedia & Creative Technology at Erasmushogeschool Brussel. My bachelor's thesis is a mobile app about nutrition and food logging. I chose this subject because I am interested in nutrition and going to the gym, but every time I used an app to log my meals, it would take too long, and I would get demotivated. My goal for my app was to make it easy to use with minimal clicks and minimal time consumption.

I want to thank my mentor, Johan Van Den Broek, for his guidance throughout the whole process. I also want to thank my fiancée, Cansu Bilal, an Erasmushogeschool Brussel alumna, for her constructive feedback on the design and logo of my app.

Working on this project has been challenging but also very rewarding.I faced several obstacles, especially trying to make the app both functional and simple to use. These challenges taught me a lot and helped me grow both personally and professionally. I have gained a deeper understanding of mobile app development and user experience design, and this experience has been incredibly enriching.

Content

Abstract

In my bachelor's thesis, I developed a mobile app focused on nutrition and food logging. I chose this project because I'm passionate about nutrition and fitness and found existing meal logging apps too time-consuming and frustrating.

The main goal was to create an easy-to-use app that minimizes the time and effort needed to log meals. The app offers three ways to log meals: a search bar, image recognition, and a barcode scanner, all accessible from a single page. This design makes meal logging quick and straightforward.

To achieve this, I focused on user-friendly design principles. For the image recognition feature, I compared different models and selected the one that best fit the needs of the app. The barcode scanner was implemented using a well-known API to ensure it works with a wide range of food products.

I conducted user testing with one participant to evaluate the app's usability and effectiveness. The results showed that my app significantly reduces the time needed to log meals compared to traditional methods. The user was very satisfied with how easy and efficient the app is.

In conclusion, the mobile app I developed solves common problems with meal logging. By combining multiple logging methods into one simple interface, the app makes it easy for anyone to keep accurate nutrition records with minimal effort.

Introduction

In today's busy world, keeping track of what we eat and staying on top of our nutrition can be quite a challenge. Many people use mobile apps to help with this, but I've found that a lot of these apps are just too complicated and time-consuming. This frustration, combined with my interest in nutrition and fitness, inspired me to create a mobile app that makes food logging quick and easy.

The main goal of my thesis is to develop a mobile app that solves the problem of complicated meal logging. I wanted to answer the question: "How can a mobile app be designed to make meal logging quick, efficient, and user-friendly?" To tackle this, I focused on a few key questions:

1. **What features make a meal logging app easy to use and efficient?**

2. **How does the design of the user interface affect the app's usability?**

To find answers, I started with a lot of research on user experience design and mobile app development. I looked at other meal logging apps to see what works and what doesn't. Then, I designed and built my app, making sure it was simple and user-friendly.

The project included testing the app with one user to see how well it worked and to get feedback. Although the feedback was positive, I know that more testing with a larger group of users is needed to get a fuller picture of how well the app performs and where it can be improved.

In the following chapters, I'll discuss the theory and research behind my project, explain how I developed the app, describe the design and implementation process and talk about what I found. By the end of this thesis, I hope to show how my app can make a real difference in how people log their meals and manage their nutrition.

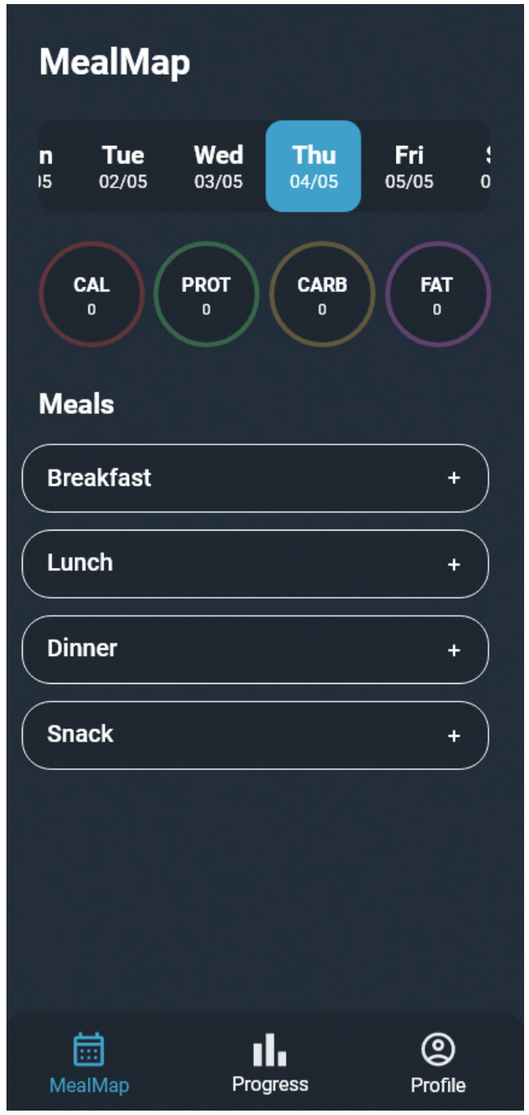
Market analysis

Competitive analysis

To understand the strengths and weaknesses of current meal logging apps, I examined three popular ones: MyFitnessPal, Lose It!, and Cronometer. This analysis informed the design of my own app, Nourify, to address common user frustrations.

From this analysis, several key takeaways emerged. Users need a quick, easy way to log meals, indicating the importance of simplicity. A balanced feature set is essential to avoid overwhelming users. A modern, intuitive interface enhances the user experience, and it's important that key features are accessible without requiring a subscription.

MyFitnessPal has the advantage of a large database, making it easy for users to find and log foods. However, its complexity can be overwhelming for new users. Lose It! offers a clean, easy-to-use interface, which enhances user-friendliness, but many of its features require a subscription, limiting the functionality available to free users.



Applying these insights to Nourify, I focused on streamlined logging, offering three easy methods—search, image recognition, and barcode scanning—on a single page. The design is user-friendly, with a simple, intuitive interface. Additionally, core features are available for free, ensuring that all users can benefit from the app's essential functionalities.

MoSCoW analysis

The MoSCoW analysis is a prioritization technique used in project management and software development to determine the importance of various features and requirements. The acronym "MoSCoW" stands for: Must-have, Should-have, Could-have and Won't-have.

By using MoSCoW analysis, project teams can focus their efforts on what is truly essential, ensuring that the project meets its core objectives and delivers the necessary functionality to users.

Must-have

- Meal Logging and Nutritional Tracking: Users need to be able to log their meals and track their nutritional intake.
- Barcode Scanning and Image Recognition: Important features to make meal logging quicker and more efficient.

Could-have

- Integration with Wearables:Optional feature to sync with fitness trackers and other health devices.

Should-have

- Progress Visualization: Adding charts or graphs to show users' progress over time to keep them motivated.

Won't-have

- Displaying Micronutrients: This feature won't be included to keep the app simple and focused on the main functions.

Functional analysis

The functional analysis outlines the key features and functionalities that Nourify will provide to effectively meet user needs.

Nourify includes several core functions to ensure it meets user needs effectively. First, users can log their meals and monitor nutritional intake using a search bar, barcode scanning, and image recognition. This makes meal logging quick and easy. The barcode scanning feature allows users to quickly log packaged foods by scanning barcodes to retrieve nutritional information, making the process more convenient. Additionally, the image recognition feature enables users to log food items by taking pictures, which is especially useful for fresh or homemade meals.

The app also includes a search bar for food logging, which provides a way to log meals by searching for food items in a database, offering flexibility in logging different types of foods. The user interface is designed to be clean and intuitive, ensuring that all core functions are easily accessible from one page, making the app easy to use. Finally, progress visualization allows users to track their nutritional progress with charts and graphs, helping them stay motivated by showing their progress over time.

Process

Mindmap

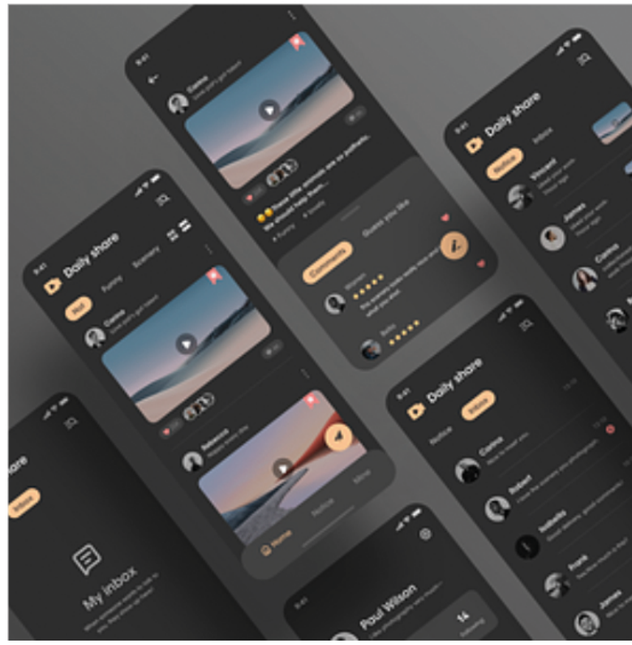
To organize and visualize the key features and functionalities of Nourify, I created a mind map. This helped in structuring the main components of the app, focusing on what users need and the overall user experience.

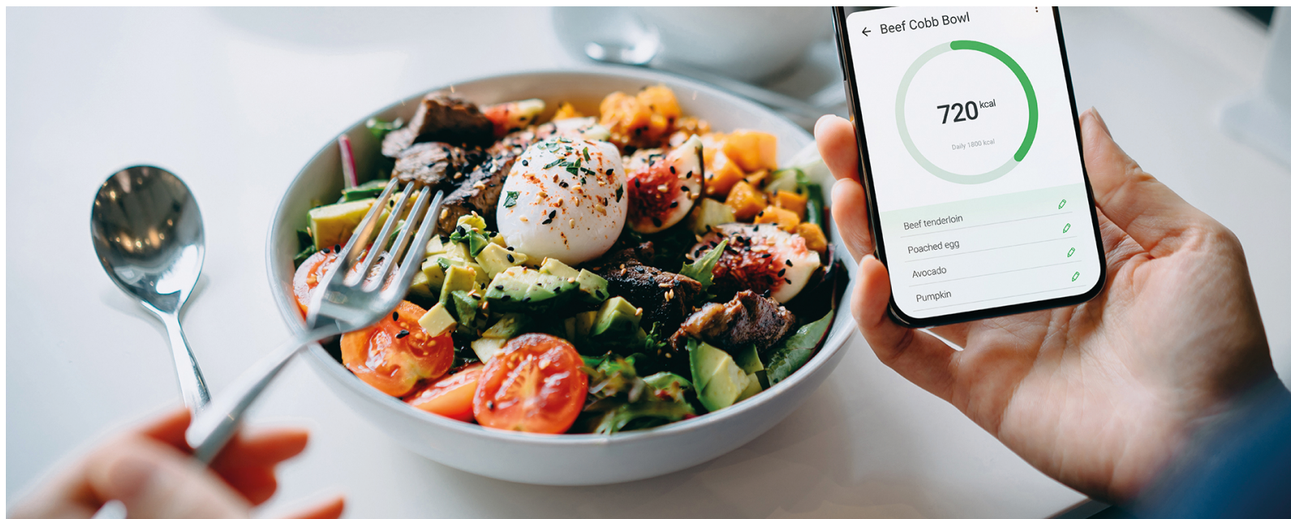
The mindmap breaks down the features into categories like Progress Tracking and Product Scanning, with specific functionalities such as logging meals, logging weight, image recognition, and barcode scanning. It also highlights the User-Centric Approach and Customization, considering factors like age, gender, activity level, and customizing macros.

Moodboard

To define the visual style and overall feel of Nourify, I created a moodboard. This was crucial in setting the tone for the app's design and ensuring it matched the user experience goals.

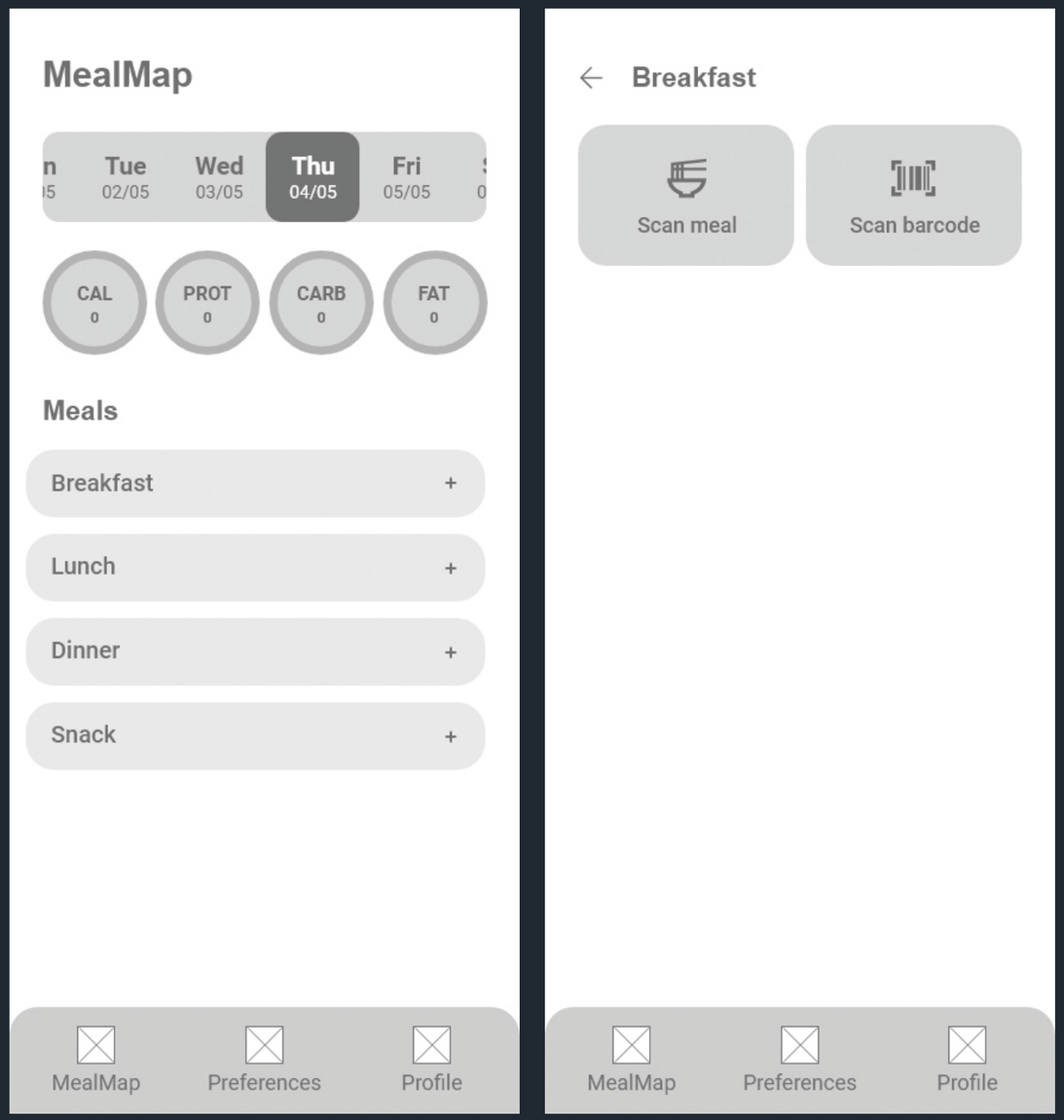
The moodboard includes images that reflect the intended look and feel of Nourify, like barcode scanning, image recognition, nutritional charts, and app interface designs. These elements together convey a modern, user-friendly, and efficient approach to meal logging and nutritional tracking.





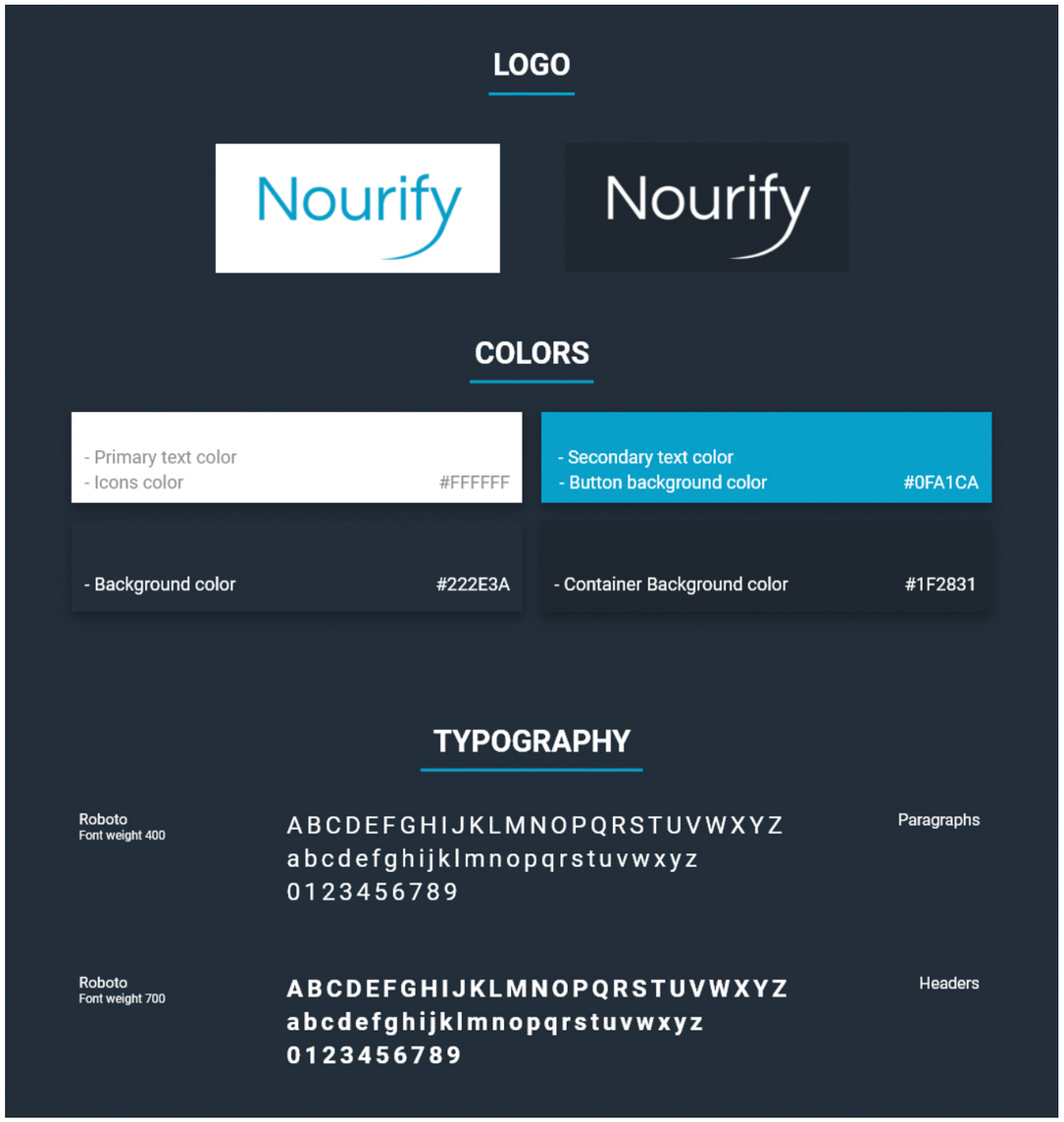
Wireframe

I began by creating wireframes to ensure the app was user-friendly and easy to navigate. This step was crucial in designing a smooth user experience, allowing me to visualize the layout and functionality before moving on to detailed design and development. The wireframes helped identify potential usability issues early and ensured that all core features were easily accessible.



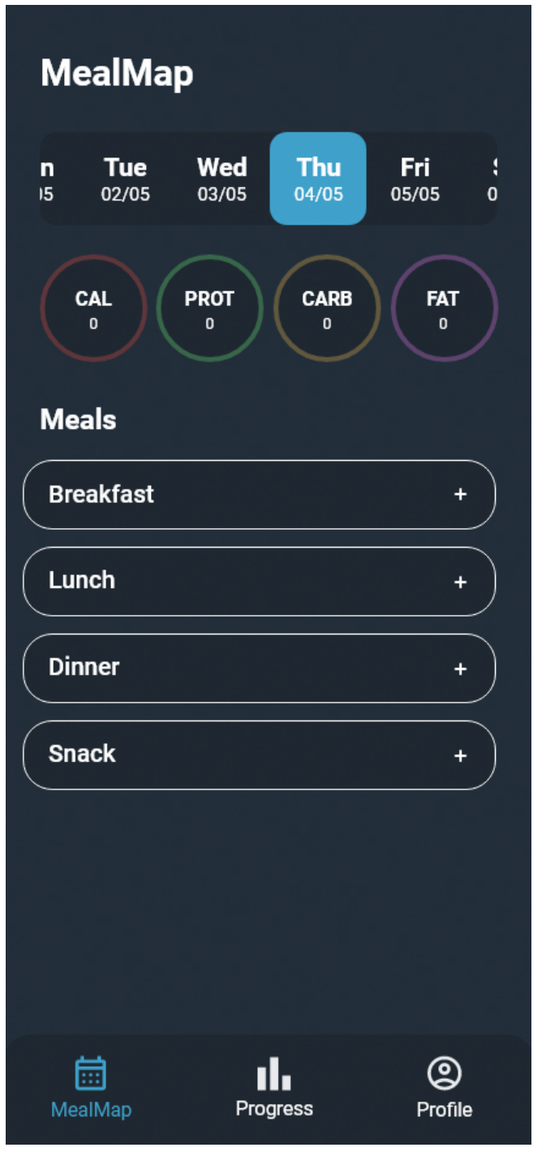
Styleguide

The second step was to define a comprehensive styleguide. I chose a dark theme for the app, aiming to create a visually appealing and modern look. The styleguide included color schemes and typography to maintain consistency throughout the app. This helped ensure a cohesive design and an intuitive user interface, enhancing the overall user experience.



Mock-up

The third step was to combine the wireframe with the styleguide to create mock-ups. This meant taking the layout and navigation from the wireframes and applying the dark theme and visual elements from the styleguide. The mock-ups gave a clear and detailed preview of the app, helping to fine-tune the design and make sure everything looked and worked well together before moving on to development.



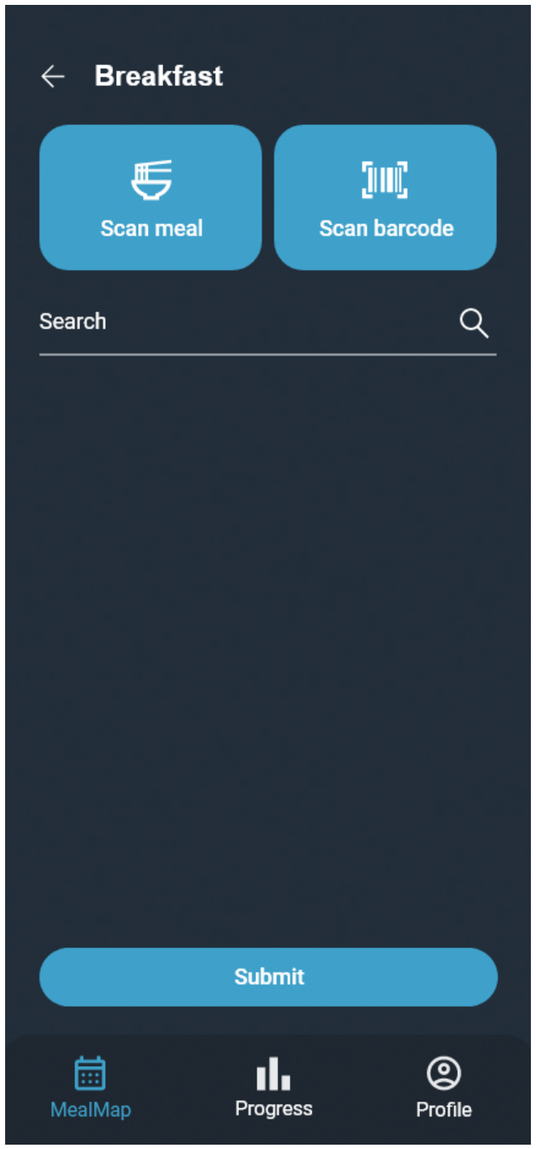


Image recognition

During my specialization course, "Expert Lab", I explored various image recognition technologies. I researched different options and decided to prototype two of them: Google Cloud Vision API and Clarifai.

Google Cloud Vision API was chosen because it can identify a wide range of objects, making it very versatile. On the other hand, Clarifai had a specialized model for recognizing food products, which was perfect for my focus on nutrition and food logging.

I created prototypes with both technologies to see which one worked best for Nourify, ensuring that the chosen solution would make meal logging quick and efficient for users.

Comparison between Google Cloud Vision API and Clarifai

PARAMETER	GOOGLE CLOUD VISION API	CLARIFAI
IMAGE RECOGNITION ACCURACY	Versatile in identifying many types of objects	Specialized models for food recognition
EASE OF INTEGRATION	Well-documented, integrates smoothly with various platforms	User-friendly API with good documentation
COST	Usage-based pricing, can get expensive with high volumes	Flexible pricing plans, possibly more cost-effective
COMMUNITY SUPPORT	Large community with lots of online resources	Smaller, but actively growing community
CUSTOMIZATION AND TRAINING	Limited customization, uses pre-trained models	Allows customization and training of models
PERFORMANCE	Generally fast and reliable	Fast performance, optimized for specific usae cases
SECURITY	High level of security and industry compliance	Emphasizes data security and compliance
TRUSTWORTHINESS	Backed by Google's reputation and infrastructure	Trusted in the industry with a solid track record
DOCUMENTATION	Extensive and well-documented	Clear and comprehensive documentation
SCALABILITY	Highly scalable, good for large-scale applications	Scalable, but may need additional considerations

This table highlights the differences between Google Cloud Vision API and Clarifai, showing why Clarifai was chosen for Nourify due to its specialized food recognition model.

Technologies

To build the Nourify app, I used the following technologies:

- **Expo SDK51 (React Native):** This framework was chosen for its efficiency in building cross- platform mobile apps.
- **Firebase & Firestore:** These were used for the authentication system and to store data securely.
- **Android Studio:** Utilized for the emulator to test the app on different devices.
- **Visual Studio Code:** The primary code editor for writing and managing the app's codebase.
- **Clarifai:** Implemented for image recognition to identify food items from photos.
- **Open Food Facts:** Used to retrieve data for scanned barcodes, providing nutritional information.
- **CalorieNinjas API:** Integrated to get data for the search bar, allowing users to find nutritional information for various foods.

Maintenance documentation

To maintain and run the Nourify app, follow these steps:

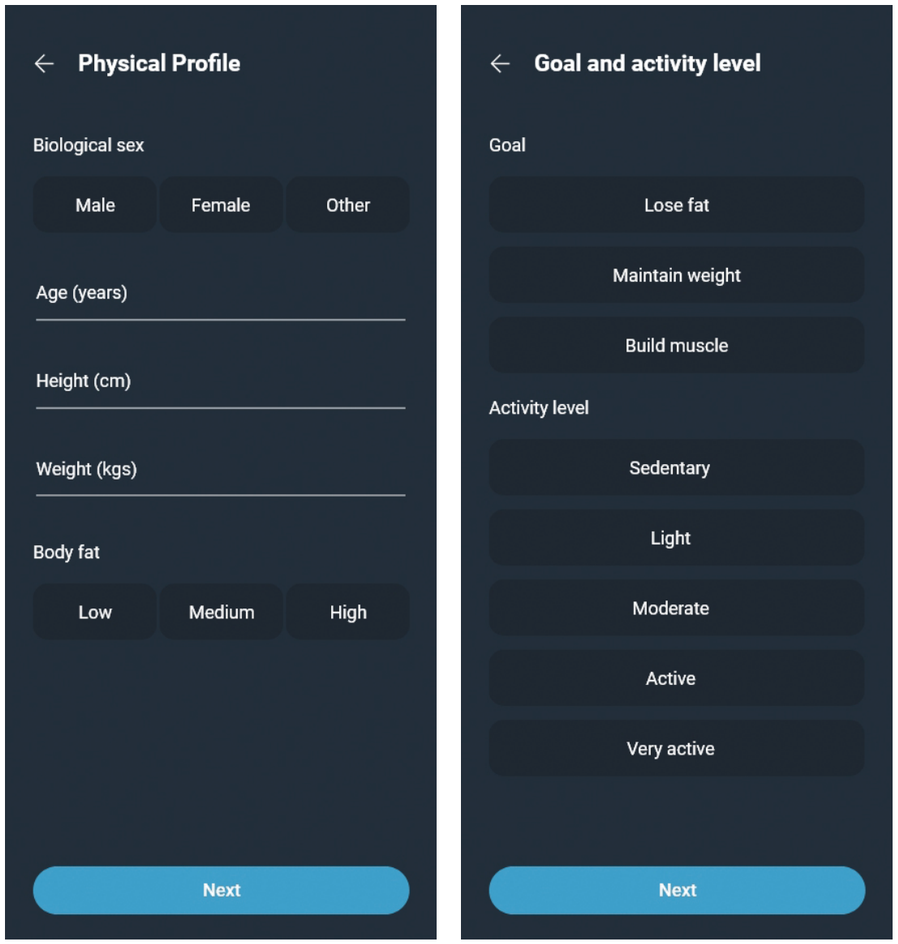
1. **Clone the Git Repository:**
    - Start by cloning the repository to your local machine.
2. **Install Packages:**
    - Navigate to the project directory and run the following command to install all necessary packages: npm install
3. **Create an Environment File:**
    - Create a new environment file named `.env` in the project root directory.
4. **Set Up Firebase:**
    - Create a Firebase application at https://console.firebase.google.com/
    - Obtain your Firebase configuration details (API Key, Auth Domain, Project ID, Storage Bucket, Messaging Sender ID, App ID).
5. **Get API Keys:**
    - Obtain API keys for Clarifai and CalorieNinjas.
6. **Configure Environment Variables:**
    - Add the following information to the newly created `.env` file, replacing `[to fill]` with your actual keys and IDs:
        - EXPO_PUBLIC_FIREBASE_API_KEY=[to fill]
        - EXPO_PUBLIC_FIREBASE_AUTH_DOMAIN=[to fill]
        - EXPO_PUBLIC_FIREBASE_PROJECT_ID=[to fill]
        - EXPO_PUBLIC_FIREBASE_STORAGE_BUCKET=[to fill]
        - EXPO_PUBLIC_FIREBASE_MESSAGING_SENDER_ID=[to fill]
        - EXPO_PUBLIC_FIREBASE_APP_ID=[to fill]
        - EXPO_PUBLIC_CALORIENINJAS_API_KEY=[to fill]
        - EXPO_PUBLIC_CLARIFAI_API_KEY=[to fill]
7. **Run the App:**
    - Download and install the Expo Go app from the Play Store on your mobile phone.
    - In the project directory, run the following command to start the Nourify app: npx expo start
    - A QR code will appear in the terminal. Scan this QR code using the Expo Go app on your phone.
8. **Ensure Network Connection:**
    - Make sure your mobile phone and the computer running the Nourify app are connected to the same Wi-Fi network.

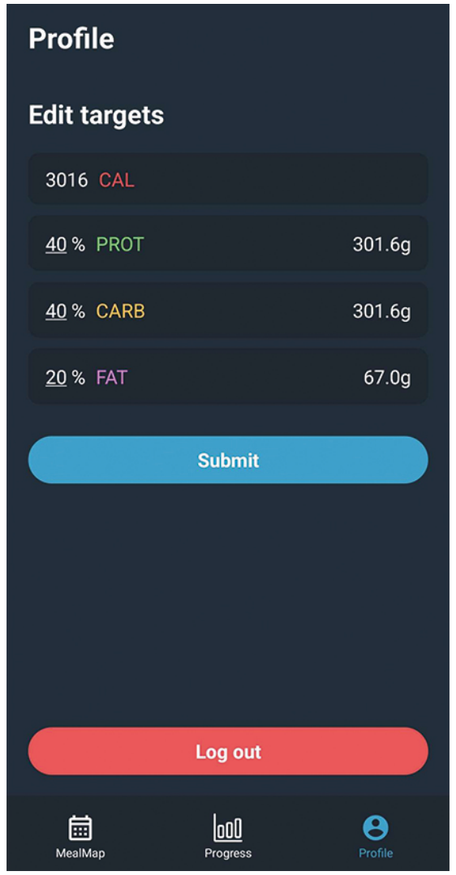
Following these steps will allow you to successfully run and maintain the Nourify app on your device.

User documentation

After opening the Nourify app, follow these steps to get started and use its features effectively:

1. **Sign Up:**
   - Open the app and sign up for a new account.
   - Provide personal information such as age, gender, height, weight, activity level, and goal. This information is used to calculate your daily calorie intake and macronutrient targets.
2. **MealMap Screen:**
   - After signing up, you will be taken to the MealMap screen.
   - Here, you will see different meal sections: breakfast, lunch, dinner, and snacks.
   - Tap on any meal section to add products using the search bar, barcode scanner, or image recognition.
   - You can also edit the amount of each product. After adding a product, it will appear on the screen.





3. **Viewing Your Daily Intake:**
   - All logged meals for the day will be displayed, showing the total calories and macros you have consumed.
   - You can navigate back to previous dates to see your nutritional intake on those days.
4. **Progress Screen:**
   - Use the bottom navigation to go to the Progress screen.
   - Here, you can visualize your weight journey and enter your current weight.
5. **Profile Screen:**
   - Navigate to the Profile screen using the bottom navigation.
   - In the Profile screen, you can customize your daily macronutrient targets and log out of your account.

By following these steps, you can effectively use Nourify to track your meals, monitor your progress, and manage your nutritional goals.

Most of the choices I made during the development process were effective. Using technologies like Expo SDK51, Firebase, Firestore, Clarifai, Open Food Facts, and CalorieNinjas API provided a solid foundation for the app. These tools helped implement the core functionalities well. However, there are a few areas that could have been approached differently for a better outcome.

One area for improvement is user testing. Although initial user feedback was positive, testing with a larger and more diverse group of users would have provided better insights into usability and functionality issues. Also, while the app's design is user-friendly, there is always room to refine the interface based on user feedback.

Throughout this project, I have learned valuable lessons about mobile app development, user experience design, and the importance of testing and feedback. The process has also deepened my understanding of integrating various technologies to create a cohesive product.

Conclusion

After working on this project for a long time, I'm glad I could bring it to a successful conclusion. While not everything is perfect and there is room for improvement, I have learned a lot about new technologies and improved my ability to create user-friendly interfaces with a strong focus on user experience.

My main goal was to create a mobile app that makes meal logging quick, efficient, and user- friendly. I believe this goal has been achieved. The app's user-friendly interface, combined with features like barcode scanning, image recognition, and a simple search bar, has made the logging process intuitive and efficient.

In conclusion, while there are areas for improvement, I am proud of what has been achieved with Nourify. The project has met its main objectives and provided a strong foundation for future enhancements. The experience has been incredibly enriching, and I am confident that the skills and knowledge gained will be invaluable in future projects.

Literature

Adobe. (2024). Moodboard tool. https://new.express.adobe.com/
Lucid. (2024). Mindmap tool. https://www.lucidchart.com/
Expo. (2024). Expo camera documentation. https://docs.expo.dev/versions/latest/sdk/camera/
Expo. (2024). Expo documentation. https://docs.expo.dev
Firebase. (2024). Database & authentication documentation. https://firebase.google.com/docs/
Clarifai. (2024). Image recognition model. https://clarifai.com/clarifai/main/models/food-item-recognition
CalorieNinjas. (2024). Nutrition API documentation. https://calorieninjas.com/api
Open food facts. (2024). Scanned products API documentation. https://wiki.openfoodfacts.org/Documentation
OpenAI. (2024). Image generation. https://chatgpt.com/

erasmus
HOGESCHOOL BRUSSEL

Nourify