Digital Epidemiology Lab

EPFL

École Polytechnique Fédérale de Lausanne

Enhancing User Engagement in Dietary Tracking Apps through Personalised Interactions and Large Language Models

by Andrea Giovanni Perozziello

Master Thesis

Approved by the Examining Committee:

Prof. Marcel Salathé
Thesis Advisor

Prof. Jacques Fellay
External Expert

EPFL SV GHI - UPSALATHE
AAB 219
Station 19
CH-1015 Lausanne

July 5, 2024

Follow the white rabbit...
— The Matrix

Dedicated to my passion for a more just Food System.

Acknowledgments

I want to thank Fanny Van de Poel for the support I received during my experience in Nairobi, Kenya.
A particular thanks goes to my dearest friends Vittorio Rossi, Alexander Furlong, Jurriaan Schuuring, Henrik Øberg Myhre, Jeremy Do-Dinh, and Tanguy Marbot for standing by me in my decisions these last years at EPFL.
Ana, my partner, with whom I share all the experiences of my life over the past four years.
I want to thank Marcel, my supervisor. His supervision was genuinely inspiring. He made me feel trusted, in control, and curious about learning from the beginning to the end of this Master's thesis journey. A special thanks goes to the other lab members I interacted the most in the office during these four months: DJ, Yannis, Marouane, Rohan, Céline and Geneviève. Nice ambience, nice presence, and good quality moments spent together.
Finally, I want to thank my parents and my sister, who were able to give me all the love I needed and taught me how to love others.

Lausanne, July 5, 2024 Andrea Giovanni Perozziello

Abstract

This thesis explores integrating a large language model (LLM) into the MyFoodRepo app to enhance user engagement through personalised interactions, reminders, and chatbot functionalities.

Despite the widespread use of food-tracking apps, current solutions often lack personalised engagement, reducing user adherence. Existing methods need to provide the tailored interactions necessary to maintain consistent dietary logging.

Our approach leverages the capabilities of LLMs to provide individualised reminders and nutritional advice based on users' dietary habits. We developed a modular architecture that supports personalised user interactions and implemented a robust reminder scheduling system to enhance user engagement.

Our design sets a strong foundation for the future testing and validation of the effectiveness of data from personalised interactions in digital health applications. This work provides a robust basis for further enhancements in the field using LLM technology.

Contents

Chapter 1

Introduction

1.1 Setting the Context

Dietary tracking is crucial, especially when analyzing the correlation between diet and medical conditions. Research institutions and hospitals need comprehensive data to understand, address, and potentially resolve various diseases. By tracking an individual's nutrition, significant information can be gathered, providing valuable insights into the conditions, causes, and possible resolutions of diseases. This process can ultimately lead to new knowledge in the medical field.

Health apps, such as the MyFoodRepo app, can play a significant role in clinical trials and research. While they may not directly contribute to better dietary habits and overall health, their primary contribution lies in enabling cohort managers to track the diet of study participants. When used alongside other clinical markers, these apps can yield significant knowledge about a patient's medical condition, potentially offering insights on improving dietary habits and overall health.

1.2 Identifying the Main Challenge

One of the main challenges users face in maintaining engagement with dietary tracking apps is the tendency to forget to log their meals. This lack of engagement can be attributed to users not perceiving the immediate benefits of using the app, viewing it merely as a tool required for research results.

User engagement is critical for the effectiveness of dietary tracking apps. High engagement rates ensure the collection of valuable data, which is essential for the research and clinical trials linked to the use of the application. When users frequently engage with the app, the data produced is more comprehensive and reliable, significantly enhancing the value of any medical investigation conducted.

1.3 Critique of Related Work

Previous studies, such as *"Kindness makes you happy and Happiness makes you healthy"* [3] and *"Notifications and Customisation in mHealth Apps"* [9], have addressed related problems but with limitations. The first focused on persuasive messages to encourage kind behaviors using Cialdini's principles of persuasion [2], and a decision tree model but did not consider the specific context of dietary tracking. The latter analyzed user attitudes towards notifications but did not implement or test software-based solutions. These studies do not fully address the challenges of maintaining user engagement in dietary tracking apps and lack the integration of advanced AI solutions. These limitations and how my research addresses them will be discussed in more detail in Chapter 6.

1.4 Proposed Solution

MyFoodRepo LLM Experiment Project

The primary goal of the *MyFoodRepo LLM experiment project* is to investigate how to stimulate and enhance user engagement in digital health mobile applications through the integration of a large language model (LLM). This project incorporates a virtual conversational agent (chatbot) to enable various types of user interactions. The successful implementation of this framework will allow the laboratory to evaluate the effects of static persuasive messages, personalised reminder messages, and interactive chatbot guidance on user engagement.

Need for Personalised Interactions

Personalised interactions are hypothesized to significantly enhance user engagement [1, 6–8], and adherence to dietary logging. A stronger connection with the app can be established by allowing users to chat and receive direct feedback and guidance on their logged food. Additionally, personalised reminder messages tailored to the user's dietary habits can further enhance the user experience, increasing the likelihood of consistent food logging.

1.5 Thesis Statement

The essence of this thesis is to develop and test a robust framework integrating a large language model into the MyFoodRepo app to enhance user engagement through personalised interactions and reminders.

1.6 Highlighting Results and Contributions

Preliminary Findings

Preliminary internal testing by lab colleagues and fellow students has shown positive feedback for the developed framework. These results suggest that the integration of personalised interactions and reminders could significantly improve user engagement and retention.

Advancing User Engagement Strategies

This project aims to create a solid framework for testing how personalisation interventions can be effectively implemented in digital health applications. Integrating a chatbot and personalised reminders represents a significant advancement in user engagement strategies within the context of dietary tracking. Traditional methods often lack personalisation and persuasiveness, whereas using LLM and personalised reminders offers tailored and persuasive interactions that can significantly enhance user engagement.

Chapter 2

Background

2.1 Introduction

This background chapter introduces the necessary technologies and dependencies to understand the design and implementation of the MyFoodRepo LLM Experiment project. It covers key technologies such as large language models, web frameworks, database management, communication platforms, scheduling libraries, and data manipulation tools.

2.2 Key Technologies

Large Language Models (LLMs)

LLMs are advanced systems trained to understand and generate human language based on statistical patterns. These models, trained on extensive datasets, can generate human-like text, making them useful for tasks such as chatbots and personalised communication.

Flask Framework

Flask is a lightweight web framework for Python, chosen for its simplicity, flexibility, and minimal setup requirements. Its modular design allows easy integration of various components, making it ideal for developing the web services needed for the MyFoodRepo chatbot feature.

Database Management with SQLAlchemy and SQLite

SQLite is a lightweight, file-based database system that stores user interactions, meals, and reminders. SQLAlchemy, an ORM (Object-Relation Mapping) library for Python, simplifies database interactions by allowing developers to work with Python objects, enhancing safety and efficiency in managing the SQLite database.

Message Integration with Twilio

Twilio is a cloud communications platform that provides APIs for sending and receiving SMS. In this project, Twilio facilitates SMS communication, enabling the app to send personalised reminders and receive user responses. Our system also supports WhatsApp messaging.

Task Scheduling with APScheduler

APScheduler is a Python library for scheduling jobs and tasks. It is used to schedule reminder messages and synchronize data with the MyFoodRepo API, ensuring timely reminders and up-to-date food log data for all users.

Data Manipulation with Pandas

Pandas is a powerful data manipulation and analysis library for Python. It processes food log data and groups and aggregates it to create textual representations for the LLM, as well as evaluates user engagement data.

Chatbot Integration with OpenAI

OpenAI provides advanced LLM technologies, enhancing user interactions in the MyFoodRepo app. These technologies generate personalised reminder messages and responses to user messages, offering a highly personalised user experience.

Chapter 3

Design

3.1 Design Overview

The design of the MyFoodRepo LLM Experiment project consists of a web service that provides essential backend functionalities and a messaging service capable of formulating messages and responding to incoming messages via webhooks. This dual-component system is designed to test and enhance user engagement through personalised interactions, reminders, and chatbot functionalities. The primary goal is to investigate how to enhance user engagement in digital health applications through the integration of a large language model (LLM).

3.2 System Architecture

The project employs a modular monolithic architecture. This architecture was chosen because it is cost-effective and suitable for the project's primary goal of conducting experiments to decide which features to implement in the MyFoodRepo app currently in production. The modular monolithic architecture allows for a straightforward design, easy maintenance, and scalability without the complexity of managing multiple microservices.

Module Interactions

The system is divided into several modules, each with specific responsibilities:

- Communication Module: Handles the low-level sending of messages (*Twilio Client* see 3.1).

- Response Module: Formulates messages based on user interactions, interfacing with the *Twilio Client* module.

- Config Module: Sets up and manages the database, scheduler, Flask app, and logging mechanism.

- Locales Module: Manages available messages in different languages.

- Scheduler Module: Manages scheduling tasks such as database synchronization with the MyFoodRepo API (*Data Sync* see 3.1)and sending reminders to users.

- OpenAI Client: Manages interactions with the OpenAI API and handles message retrieval and storage (*OpenAI Client* see 3.1).

- Services Module: Provides simple data processing features for handling food log data.

- Utils Module: Contains utility functions to support various operations across the system.

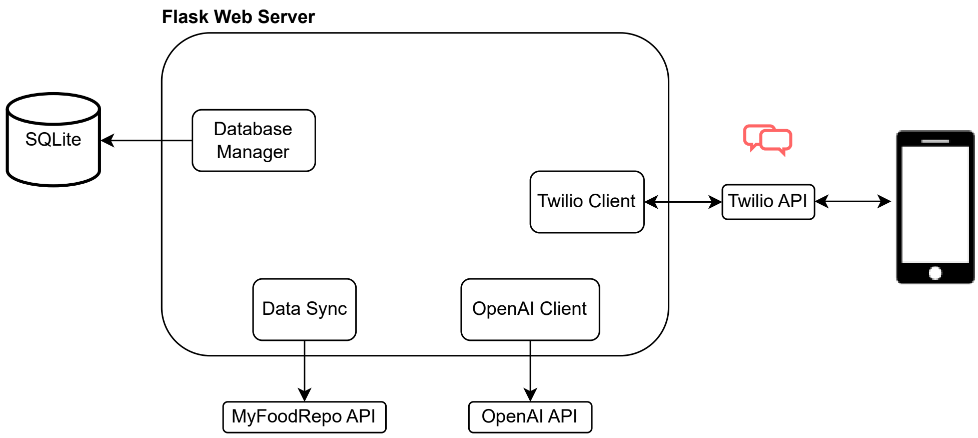

Figure 3.1: Content Diagram of the Application

3.3 User Interaction Design

User interaction flows were designed based on the hypothesis that personalised reminders and chat interactions could enhance user engagement.

Four user groups were created to test this hypothesis (see Figure 3.2):

- Group 0 (Control Group): Receives only a welcome message.

- Group 1: Receives a welcome message and three static persuasive reminders daily (breakfast, lunch, and dinner).
- Group 2: Receives a welcome message and can chat with the bot about their logged food.
- Group 3: Receives a welcome message, can chat with the bot, and receives personalised reminders based on their dietary habits three times a day.

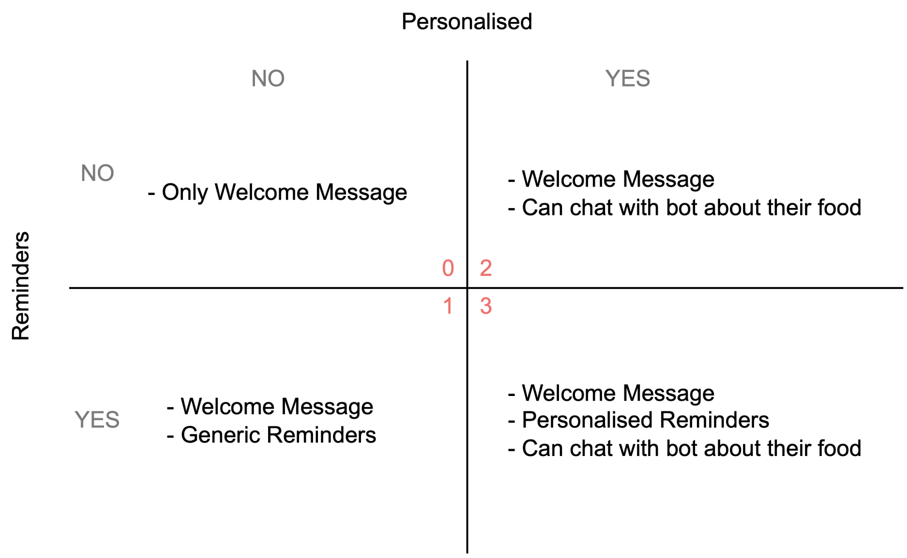

Figure 3.2: Study Groups table

3.4 Personalised Reminders and Chatbot Functionality

Personalised Reminders

Personalised reminders were implemented to enhance user engagement by providing dynamic and tailored content. These reminders take into account individual data such as age, gender, dietary goals, recent meal history, and the type of meal for which the reminder is crafted (breakfast, lunch, dinner). The aim is to be more persuasive and relevant to the user, increasing the likelihood of meal logging (implementation details discussed in 4.6).

Chatbot Functionality

The integration of the OpenAI-powered chatbot was designed around three main components:

- System Prompts: Provide the LLM with structured conversation guidelines and contextual dynamic information about the user, such as last meals consumed, dietary goal, age and gender.
- User Prompts: Represent real user interactions received as messages.
- Assistant Prompts: Are the chatbot's responses generated by the LLM based on the input prompts.

The main challenge in this design was engineering the System Prompts to ensure the model received the correct contextual information at the right time.

3.5 Database Design

The database schema (see Figure 3.3) consists of four tables: Users, Messages, Meals, and Reminders.

- Users Table: Stores user information and links to their message history, meals and reminders.
- Messages Table: Contains the history of all messages (System, User, and Assistant Prompts) for each user.
- Meals Table: Logs the food entries made by users in the MyFoodRepo app, grouped within a 10-minute window of their consumption. The *description* column holds the name of the meal, whereas the *macros* column holds a JSON object with the aggregated macros.
- Reminders Table: Lists all reminders for each user, specifying the time and type of meal (breakfast, lunch, snack, dinner).

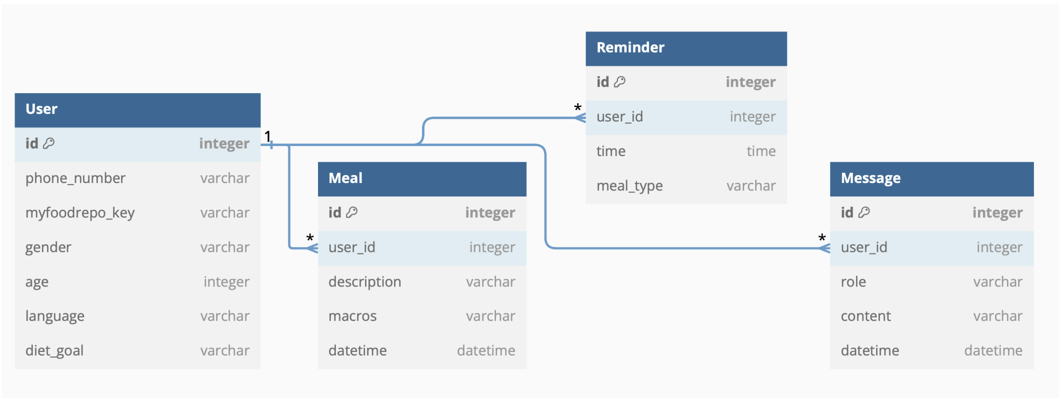

Figure 3.3: SQLite database schema

3.6 Scheduling and Data Synchronization

Scheduling Mechanisms

Scheduling tasks using APScheduler was crucial for managing reminders and data synchronization jobs. The scheduler initiates upon server startup, scheduling two main tasks: fetching meal data from the MyFoodRepo API and scheduling user reminders. The scheduling mechanism ensured that reminders were sent at the correct times and that the system's data remained up-to-date with the MyFoodRepo data app.

Data Synchronization

Data synchronization between the MyFoodRepo app and the system was designed to occur through scheduled jobs. These jobs fetched new data from the MyFoodRepo API, processed it into a Pandas dataframe, and updated the database. The system ensured data consistency by aggregating food log annotations into meals and calculating macro-nutrient totals. In this way, our system had access to the most recent meal history of each user.

3.7 Scalability and Maintainability

The design emphasizes scalability and maintainability through modularity and good coding practices:

- Modularity: The system is divided into well-defined modules, each responsible for specific functionalities.
- Good Coding Practices: The codebase features short functions, separation of concerns, consistent naming conventions, and self-descriptive variable and function names. Doc-strings are used instead of comments to keep the code self-explanatory.

Although the project was designed for a relatively small number of participants (a few hundred), the modular design and good practices ensure that it can be easily scaled and maintained for larger deployments.

Chapter 4

Implementation

4.1 Setup and Initialization

Before starting the development of the chatbot, exploratory data analysis (EDA) was performed on the food log datasets from the MyFoodRepo app's test environment. This analysis, conducted using Python notebooks, helped in understanding the data structure and identifying potential operations. The EDA revealed various details about the data stored by MyFoodRepo, such as understanding the structure of the food logs, how to group them to represent meals, aggregating their macro-nutrients, and creating informative descriptions. These summaries are later prompted to the LLM to provide information about the user's diet.

The project began by creating a repository within the *Digital Epidemiology Lab GitHub* organization. A `'.env'` file was created to load environment variables using the `load-dotenv` library. A virtual environment was set up with Conda, and a *'requirements.txt'* file was added to manage dependencies. A robust logging mechanism (defined inside `'src/config/logging_config.py'`) was configured early on, and built upon the standard Python *logging* library allowing users to log different events with varying granularity and handle two output streams: console log and a *'myfoodrepo.log'* file.

4.2 Database Configuration

The database configuration is defined and managed inside a module named `'data_manager/models.py'`, which contains the schema and relationships of the database tables (see Figure 3.3). SQLAlchemy facilitated this process by providing a smooth interface for managing database operations in Python. At server startup, if the database did not already exist, it was created and initialized.

A scoped session manager was implemented using *sessionmaker* to improve efficiency and safety. This singleton approach allowed simpler and safer database operations by centralizing the management of transactions. Consequently, all functions interacting with the database did not have to manage database abstractions such as *commit* or *rollback* operations.

4.3 API Development

Several API endpoints were developed to enable communication between different system components.

The main endpoints include:

- *sms_reply*: Handles all incoming messages.
- *add_all_reminders*: Adds reminders to the database for all users in Groups 1 and 3.
- *init*: Populates the database from a CSV file derived from Google Form data.
- *start*: Sends welcome messages to all participants.

These endpoints facilitate interactions between the web service and messaging components, allowing the system to process user inputs, generate appropriate responses and initialise some components during project setup.

4.4 Integration of OpenAI

Integrating OpenAI's API involved setting up the *'openai_api_key'* environment variable. A dedicated module was created to manage interactions with OpenAI, enabling functionalities such as initializing conversations with users, managing messages by role (System, User, Assistant), and creating chat completions based on conversation history.

The system leverages OpenAI's Chat Completions to generate both chatbot messages and personalised reminder messages. For chatbot messages, the Chat Completions feature utilizes the conversation history to provide coherent and contextually relevant responses, ensuring a seamless and engaging interaction for the user. For personalised reminders, specific prompts are crafted using user data, such as dietary habits, preferences, and past interactions, which are then processed by OpenAI to generate tailored and persuasive reminders.

This integration allowed the system to generate personalised messages and chatbot responses, significantly enhancing the user experience by providing timely, relevant, and engaging interactions.

4.5 Message Handling and Scheduling

Message Handling

Message handling was implemented using Twilio for SMS and WhatsApp communication. To accomplish this, an abstraction for the two messaging services was implemented in the *'communication'* module. This abstraction allowed the rest of the code to avoid dealing with the specific low-level details of sending different types of messages. This class is defined inside the '`src/communication/messaging_service.py`' file.

Environment variables such as *account_sid*, *auth_token*, and the Twilio phone number were required for configuration.

Three main functions were developed: one for handling incoming messages and generating AI-based replies, another for sending messages (like reminders) to users, and finally, one to send a welcome message to all users. All message-related interactions use the user's phone number as both a recipient and an identification mechanism.

Scheduling Tasks

Scheduling mechanisms were designed using APScheduler to ensure timely execution of tasks. The scheduler initiates a routine when the server starts, performing two main tasks:

1. Fetching all reminders for users of groups 1 and 3 (study groups with the Reminder Feature activated, see 3.2) from the database and scheduling them as jobs, using '`cron`' (line 27 of Listing 4.1).
2. Scheduling database synchronization jobs multiple times a day to keep the data updated, using '`intervals`' (line 13 of Listing 4.1).

```
1 logger = Logger('myfoodrepo.scheduler').get_logger()
2 scheduler = BackgroundScheduler()
3 timezone = pytz.timezone('Europe/Zurich')
4 
5 
6 def fetch_meal_data():
7     update_database()
8 
```

```python

def schedule_fetch_meal_data():
    scheduler.add_job(
        fetch_meal_data,
        'interval',
        hours=DB_UPDATE_TIME_INTERVAL
    )


def schedule_reminders(app):
    logger.info("Scheduling reminders for all users...")
    all_users = User.query.filter(User.study_group.in_([1, 3])).all()

    for user in all_users:
        for reminder in user.reminders:
            logger.debug(f"Scheduling Reminder:{reminder.id} for User:{user.id}")
            scheduler.add_job(
                send_reminder,
                'cron',
                hour=reminder.time.hour,
                minute=reminder.time.minute,
                args=[user, reminder.meal_type, app],
                timezone=timezone,
                id=f'reminder_{reminder.id}'
            )
    logger.info("Scheduling reminders DONE")


def start_scheduler(app):
    schedule_reminders(app)
    schedule_fetch_meal_data()
    scheduler.start()

    atexit.register(stop_scheduler)


def stop_scheduler():
    scheduler.shutdown()
```

Listing 4.1: Scheduler module code

4.6 Personalised Reminder Generation

Personalised reminders were generated using data such as the user's age, gender, dietary goals, and recent meal history. The system utilized OpenAI to create personalised messages based on these factors. This process involved generating a specific prompt (see Listing 4.2) for OpenAI that included relevant user information, which OpenAI then used to craft a personalised reminder message. This approach ensured that reminders were timely and relevant to users, increasing the likelihood of consistent dietary logging.

```
REMINDER_PROMPT_TEMPLATE = """
    You are a helpful assistant tasked with generating personalized reminders for users of a food diary app.
    Below is the information about the user and their dietary habits.
    Use this information to create a personalized reminder message.
    This {meal_type} reminder should encourage the user to log their meals, considering their dietary goals, and meal history.

    User Information:
    - Gender: {gender}
    - Age: {age}
    - Dietary Goal: {dietary_goal}
    - Recent Meals: {recent_meals}

    Your reminder should be friendly, motivating, and specific to the user's dietary needs and goals. Consider their recent meals to suggest what they might log next or highlight patterns you see. Provide general observations about their diet and how it aligns with their dietary goals.

    Use the principles of persuasion, primarily Commitment, and occasionally Scarcity, to make the reminders more compelling and less repetitive between one and another.

    Here are some example reminders:

    1. **Commitment**: "Hi there! Don't forget to log your {meal_type}. You've been doing an excellent job staying consistent with your goal of {dietary_goal}. Keeping track of your meals like (some meals that well align with the user's dietary goal) shows your dedication. Keep it up!"
    :
    :
    5. **Scarcity**: "Hey! Don't miss logging your {meal_type}. Each meal logged is a crucial step towards achieving {dietary_goal}. You are doing great, and each log is a valuable part of your journey!"

    Use these examples to guide your responses, but feel free to be creative and adjust based on the user's recent meals and specific situation. Generate one personalized reminder message for the user."
    Any response you formulate MUST NEVER exceed 1100 characters, otherwise the user will never receive it!
    Any message you formulate should be analytical, creative, concise and helpful for the user! **NEVER be repetitive!**
"""
```

Listing 4.2: System Prompt for Personalised Reminders

Where the {recent_meals} would contain a list of food items like this one:

```
Food Item: Wholemeal pasta egg-free, cooked in salted water (uniodised), Green
    pea, steamed (without addition of salt), Tomato sauce
Time: 2024-06-15 14:04
Macros:
- Calories: 357.45
- Water_ml: 162.87ml
- Protein_g: 15.19g
- Carbs_g: 60.68g
- Fat_g: 2.90g
- Sat_fat_g: 0.48g
- Fiber_g: 14.20g
```

Listing 4.3: Example of recent meals content

4.7 Data Synchronization and User Interaction Flow

Data Synchronization

Effective data synchronization between our system and the MyFoodRepo API is ensured by scheduling sync jobs at regular intervals (default is every two hours, defined by 'DB_UPDATE_TIME_INTERVAL=2' inside 'src/constants.py'). These jobs are meant to fetch data from the MyFoodRepo API (*Data Sync* see 3.1), process it using Pandas to extract relevant information, and store it in the database.

This ensures that our system always has up-to-date information about each user's meals logged on the MyFoodRepo app.

User Interaction Flow

User interaction flows were managed with checks within the code to determine which functions were allowed for each user group or to filter messages from phone numbers from users not registered. For example, the *sms_reply* endpoint (see Listing 4.4) checked the user's study group before performing operations. For features like reminders, jobs were scheduled only for users in study groups 1 and 3, with differentiation in the type of reminders sent. This method ensured that each user received appropriate interactions based on their assigned group, allowing for effective testing of different engagement strategies.

```
@app.route("/sms_reply", methods=['GET', 'POST'])
def sms_reply():
    body = request.values.get('Body', None)
```

```python
    from_number = convert_from_whatsapp_number_format(request.values.get('From', None))

    if not is_user_registered(from_number) and not is_user_allowed_to_chat(from_number):
        logger.warning(f"Receiving unexpected messages from: {from_number}")
        return "User not registered in the program, or not allowed to chat!"

    logger.info(f"Received a message from: {from_number} with content: {body}")

    response_data = {'response': None, 'sent': False}
    timer = threading.Timer(5.0, send_interim_response, [from_number, response_data])
    timer.start()
    threading.Thread(target=process_openai_response, args=(
        app, from_number, body, response_data, timer
        )).start()

    return "Message Sent"
```

Listing 4.4: *sms_reply endpoint code*

4.8 Testing and Debugging

Currently, automated tests have not been implemented. Testing has been carried out manually through various endpoints that test all functionalities of the program. Although this is not ideal, these manual tests have been crucial in identifying and fixing issues to meet the project requirements. While automated testing is planned for the future, the current manual testing approach has helped ensure that the system functions correctly and meets the desired specifications.

Chapter 5

Evaluation

5.1 Introduction

The primary purpose of this evaluation chapter is to convince the reader that the design works as intended. This includes describing the evaluation setup, the designed experiments, and how the experiments showcase the individual points we want to prove. The key hypothesis is that the introduction of a virtual conversational agent can enhance user engagement in the context of digital health applications like MyFoodRepo. This evaluation is crucial as it lays the groundwork for testing the effectiveness of such interventions, aligning with the broader goals of the thesis to improve user engagement through innovative design.

5.2 Evaluation Setup

Experiment Overview

The purpose of the experiment is to investigate whether the use of a virtual conversational agent can improve user engagement in digital health applications. The key components include the project infrastructure, the obtainment of ethical consent, user recruitment via a Google form, the experimental pipeline, and data analysis tools to evaluate user engagement and retention.

Groups and Participants

The experiment consists of four groups:

- **Group 0 (Control Group)**: Uses the existing app without any new features.
- **Group 1**: Receives static reminders three times a day (breakfast, lunch, dinner) using persuasive principles based on Cialdini's principles of persuasion (commitment and scarcity) [2].
- **Group 2**: Has access to chat with the bot about their logged food, receiving nutritional guidance and diet advice.
- **Group 3**: Receives personalised reminders three times a day in addition to the chatbot interaction, with reminders tailored based on age, gender, dietary goals, and recent meals.

The group sizes of at least 50 individuals per group are chosen to ensure statistical significance, providing a robust sample size for reliable results. The final sample size will account for approximately 200 persons, where each person will have been randomly assigned a number from 0 to 3 which will determine the study group. For convenience, participants have been selected based on their possession of a Swiss phone number.

Environment and Tools

The experiment will be conducted in a real-world setting where participants use their smartphones to interact with the MyFoodRepo app. The backend infrastructure is hosted on a managed web service from *Infomaniak*, which supports the project by running the program continuously for the duration of the experiment (two weeks).

5.3 Experiment Design

Group 0 (Control Group)

Participants receive a welcome message with a link to download the app and an activation key. This group serves as the baseline, expected to exhibit the typical behavior of MyFoodRepo app users.

Group 1

Participants receive a welcome message and three static persuasive reminders daily at breakfast, lunch, and dinner times. These reminders are crafted using Cialdini's principles of persuasion. For example:

- **Commitment reminder**: "Stay on top of your goals by logging your meal. Consistency is key!"

- **Scarcity reminder**: "Seize the moment! Log your meal before it passes by."

The expected outcome is that the reminders will help users associate logging meals with their meal times, potentially improving logging consistency.

Group 2

Participants receive a welcome message and have the ability to chat with the bot about their logged food. This interaction is expected to make the app usage more interactive and informative, potentially increasing user engagement by providing personalised nutritional advice.

Group 3

Participants receive a welcome message, can chat with the bot, and receive personalised reminders. These reminders are generated using an OpenAI model, considering factors such as meal type, gender, age, dietary goals, and recent meals (see Sections 3.4, and 4.6). This group is expected to show the highest levels of engagement and retention due to the combined effects of personalisation and interaction.

5.4 Evaluation Metrics and Criteria

User Engagement

User engagement will be measured by tracking the frequency of app use, interaction with the chatbot, and logging consistency. Specific metrics include the number of chatbot interactions and the regularity of meal logging.

User Retention

User retention will be measured by analyzing the number of meals logged, time gaps between logs, and consistency in calorie logging. Metrics include meals logged, time gap analysis, and calorie logging anomalies.

User Satisfaction

User satisfaction will be assessed through a feedback form at the end of the study, using a 5-point Likert scale to rate overall experience and specific features.

5.5 Expected Results and Hypotheses

We hypothesize that the introduction of personalisation components, such as chatbots and personalised reminders, will positively impact user engagement, retention, and satisfaction. Specifically, we expect that groups with access to the chatbot (Groups 2 and 3) will show higher engagement levels. Additionally, we anticipate that personalised and persuasive reminders will improve user retention, with Group 3, which has the most comprehensive features, exhibiting the highest retention rates. Finally, we hypothesize that personalised and persuasive elements will enhance user satisfaction, with users in Group 3 likely to report the highest satisfaction levels due to the tailored experience.

5.6 Analysis Plan

Data Collection and Processing

Data will be collected through automated logs of food entries and chatbot interactions. This data will be processed to group food logs into meals per day and to count the number of chatbot interactions.

Statistical Analysis

We propose using ANOVA (Analysis of Variance) to compare the means of engagement and retention metrics across the four groups. ANOVA is appropriate for this analysis because it allows us to determine if there are statistically significant differences between the groups' means, thereby assessing the overall impact of the personalisation components. For categorical data analysis, such as user satisfaction ratings, we will use Chi-square tests. The Chi-square test is suitable for examining relationships between categorical variables, enabling us to evaluate whether the distribution of user satisfaction ratings differs significantly among the groups.

Interpretation of Results

Results will be interpreted by examining correlations between our hypotheses and the collected data. The success of the experiment will be judged by the ability to demonstrate whether the intervention impacts user engagement, retention, and satisfaction.

5.7 Validation of Design

Theoretical Support

The theoretical basis for this experiment is supported by studies on the effectiveness of persuasive technology and the role of personalised interactions in enhancing user engagement in digital health applications.

Existing Studies and Literature

Relevant studies include research on Cialdini's principles of persuasion in health behavior change and the use of conversational agents in improving user engagement in health apps (see Section 6.2).

5.8 Discussion

Potential Performance

Based on theoretical analysis, the design is expected to perform well by significantly improving user engagement and retention through personalised and persuasive interactions.

Strengths and Limitations

Key strengths include the distinct groups that allow for detailed analysis of the impact of personalisation and reminders. Another strength relies on the link that the project has to the MyFoodRepo app, which differentiates it from similar apps for its excellent personalised nutrition and food tracking features. Potential limitations involve the current capabilities of LLMs in handling personalised interactions, the usage of SMS in delivering notifications instead of *push notifications* and the need for more complex interaction handling in future versions.

Lessons Learned

Lessons learned include the need to diversify LLM-generated reminders to avoid repetitiveness and the importance of integrating user feedback to adapt reminders and interactions. The experiment's design has revealed the potential of LLMs in providing basic nutritional guidance and recipe suggestions.

Insights for Future Work

Future work should focus on conducting the proposed experiment to validate these hypotheses and explore more complex interaction scenarios. Enhancing the personalisation and adaptability of the reminders based on user feedback will be crucial.

Chapter 6

Related Work

6.1 Overview of the Field

The general field of my research is persuasive technology for digital health applications, specifically within the context of personalised nutrition. Key areas of focus within this field include user engagement strategies, personalisation techniques, and the effectiveness of digital interventions to promote healthy behaviors. This section aims to highlight related work, how it solved relevant problems, and why these problems are different from the ones addressed in this thesis. This discussion provides the reader with a broader context of the problem that our research aims to address.

6.2 Identifying Closely Related Work

Relevant Studies and Projects

Two significant studies closely related to my work are from the *Persuasive Technology 2023 Springer* book [5]. Ciocarlan et al. (2023) [3] investigates the use of persuasive messages, while Pretolesi et al. (2023) [9] examines the efficacy of notifications in digital health applications. These studies are recent; both were conducted in 2023.

Problem Addressed by Related Work

Ciocarlan et al. (2023) address the communication problem with users, aiming to persuade them to perform certain tasks through digital behaviour change interventions using persuasive messages. Pretolesi et al. (2023) focus on the lack of proven efficacy of notifications in digital health applications, seeking to improve the personalisation and effectiveness of such notifications.

The primary objectives of these studies are, for Ciocarlan et al. (2023) to enhance user interaction personalisation to increase persuasiveness and to design app features that improve user experience, engagement, and retention Pretolesi et al. (2023).

6.3 Approaches and Solutions

Ciocarlan et al. (2023) conducted a randomized controlled experiment to evaluate two digital behavior change interventions (personalised and non-personalised) that used persuasive messages to encourage kind behaviors. They employed Cialdini's principles to motivate participants to complete activities [2]. A machine learning model was developed to predict the most effective principle for individuals based on their personality traits using the Five Factor Model (FFM)[10]. The ID3 algorithm for decision tree learning was implemented using the decisiontree 0.5.0 Ruby library.

Pretolesi et al. (2023) surveyed a diverse population on their attitudes and practices concerning notifications in health and well-being applications. The survey design was inspired by existing literature and included the affinity for technology interaction (ATI) scale and demographic information [4]. Data were processed and analyzed using R 4.1.2, with response frequencies calculated for descriptive analysis. Cumulative link models were computed using the ordinal package to assess customization preferences.

6.4 Comparison and Contrast

Differences in Problem Scope

While both studies address issues within digital health, they do not specifically analyze how to effectively increase user adherence in such applications. Additionally, the introduction of Large Language Models (LLMs) and the analysis of survival rates in applications were not considered in these studies.

Differences in Approach

The methodologies of these related works differ from mine in several ways. Ciocarlan et al. (2023) used a digital behavior change intervention with a machine learning model but did not build an extensive software system. Pretolesi et al. (2023) used surveys and did not involve any software development. My approach involves developing a comprehensive software framework for running experiments with a personalised chatbot system integrated into an existing digital health app (the MyFoodRepo mobile application).

Strengths and Weaknesses of Related Work

The strengths of Ciocarlan et al. (2023) include its in-depth analysis of user personalisation and the application of persuasive principles. However, the model had limitations due to noisy training data and overfitting. Pretolesi et al. (2023)'s strength lies in its thorough analysis of notification preferences, though its reliance on surveys limits its applicability to real-life contexts.

Both studies evaluated their solutions through randomized controlled trials and surveys, providing valuable insights but also highlighting the need for more practical, software-based interventions.

6.5 Contributions and Improvements

Addressing the Gaps

My research addresses the implementation of a personalised chatbot system within a specific, tested app context, providing a practical solution that combines personalisation and real-time interaction, which was not explored in Ciocarlan et al. (2023) or Pretolesi et al. (2023).

Innovations and Trade-offs

The innovative aspect of my approach is the integration of an LLM (ChatGPT-4o) for providing personalised interactions. The trade-offs include potential biases introduced by the LLM and the less tailored, one-size-fits-all approach compared to the highly specific personalisation techniques used in previous studies.

Evaluation and Results

I plan to evaluate my solution through a combination of user engagement metrics, retention rates, and satisfaction surveys, similar to the methods used in related works but with the added dimension of real-time interaction analysis. Preliminary internal testing has shown positive feedback, suggesting the potential effectiveness of my approach.

6.6 Discussion and Reflection

The related works collectively contribute to the field by advancing our understanding of persuasive technologies in digital health applications. My work builds on these contributions by focusing on personalised nutrition and motivation within the broader context of digital health, using cutting-edge AI solutions.

Future research directions suggested by related works include more in-depth sentiment analysis, larger demographic sampling, and the implementation and testing of new findings. My research opens new avenues by providing a framework to test the impact of personalised conversational agents on user engagement in digital health applications.

It is important to highlight the value of related works while distinguishing my contributions. The in-depth analysis and methodologies of these studies provide a solid foundation for my research. By building on their strengths and addressing their limitations, my work contributes to advancing the field of persuasive technology and personalised digital health interventions.

The main takeaways from the review of related work are that extensive efforts have been made to understand and improve user engagement in digital health through personalisation and notifications. These studies provide valuable insights and highlight the importance of continuous innovation. This related work sets the stage for my research by illustrating the need for practical, software-based solutions that integrate advanced AI technologies like LLMs.

Chapter 7

Conclusion

The primary objective of this thesis was to create a framework for testing how the introduction of personalisation and large language models (LLMs) can impact user engagement in digital health applications. The main result of this research is the successful implementation of such a system, which will enable this testing scenario. This aligns perfectly with the original goals, as the system was fully developed.

The key findings of this research include the successful implementation of LLMs for personalisation within the MyFoodRepo app framework. Although we have not yet conducted experiments to evaluate the impact of these personalised interactions, we are optimistic that AI-powered chatbots and personalised reminders will enhance user engagement. The integration of LLMs has streamlined the process of personalizing reminders and automating message replies, providing a robust foundation for future evaluations of their effectiveness.

This system improves upon existing solutions in dietary tracking and user engagement by integrating LLMs in a personalised context for mobile applications in digital health. This project provides a tangible way to test such interventions, setting it apart from related studies. The innovative aspects of this design include the use of LLMs and the linkage to a powerful food tracking app like MyFoodRepo, which can accurately analyze detailed food log data, including nutrients and quantities [11]. This research addresses gaps in existing studies by offering a framework to test the impact of virtual conversational agents on user engagement in digital health applications.

The framework developed in this thesis demonstrates the feasibility of implementing personalised reminders and chatbot interactions in health apps to improve user engagement. These interactions can be implemented using communication systems like Twilio and virtual conversational agents like the ones offered by OpenAI. The potential benefits for users include a more attentive and engaging journey toward their nutrition goals, with healthcare providers benefiting from improved patient engagement and more consistent data for research purposes.

This research offers new insights by combining a cutting-edge food-tracking app with personalisation components to enhance user engagement. This combination is unique in the field of persuasive technology applied to digital health. The framework developed in this research can help understand the effectiveness of different engagement strategies, contributing to the broader understanding of user engagement in health applications.

The main limitations of this study include the demographic focus on the Swiss population, the short duration of the designed experiment (two weeks), and the reliance on current versions of LLMs and SMS communication. These limitations might affect the generalizability and longevity of the findings. Future research should test the experiment in different settings and with more diverse audiences, for a longer period of time, using updated LLM versions, and employing more immersive communication systems like in-app push notifications and WhatsApp.

The next steps for research in this area involve running the experiment and analyzing the results. The framework can be expanded to different scenarios, using various communication methods and applying personalisation to other types of digital health applications.

This research was an exciting endeavor, and I am hopeful that this framework will allow the scientific community to understand how LLMs in digital health applications can drive user engagement. The future of AI and personalised interactions in digital health applications holds immense potential, albeit with significant ethical and practical considerations. I hope this research inspires others to build upon these findings and continue exploring the possibilities of AI-driven personalisation in health technology.

In closing, the potential of this research is substantial, yet it is crucial to acknowledge its limitations. I encourage readers to consider both the possibilities and the challenges posed by integrating AI in digital health. The ultimate goal is to inspire future research and development that will refine and enhance user engagement strategies in this critical field.

Bibliography

[1] "Influencing Individually: Fusing Personalization and Persuasion". In: *ACM Transactions on Interactive Intelligent Systems* 2.2 (June 2012). Ed. by Shlomo Berkovsky, Jill Freyne, and Harri Oinas-Kukkonen, pp. 1–8. ISSN: 2160-6455, 2160-6463. DOI: 10.1145/2209310.2209312. URL: https://dl.acm.org/doi/10.1145/2209310.2209312 (visited on 07/03/2024).

[2] Robert B. Cialdini. *Influence: science and practice*. 4th ed. Harlow: Pearson Higher Education, 2001. 262 pp. ISBN: 9780321188953.

[3] Ana Ciocarlan, Judith Masthoff, and Nir Oren. "Kindness Makes You Happy and Happiness Makes You Healthy: Actual Persuasiveness and Personalisation of Persuasive Messages in a Behaviour Change Intervention for Wellbeing". In: *Persuasive Technology*. Ed. by Alexander Meschtscherjakov, Cees Midden, and Jaap Ham. Vol. 13832. Series Title: Lecture Notes in Computer Science. Cham: Springer Nature Switzerland, 2023, pp. 198–214. ISBN: 978-3-031-30932-8 978-3-031-30933-5. DOI: 10.1007/978-3-031-30933-5_13. URL: https://link.springer.com/10.1007/978-3-031-30933-5_13 (visited on 07/03/2024).

[4] Thomas Franke, Christiane Attig, and Daniel Wessel. "A Personal Resource for Technology Interaction: Development and Validation of the Affinity for Technology Interaction (ATI) Scale". In: *International Journal of Human–Computer Interaction* 35.6 (Apr. 3, 2019), pp. 456–467. ISSN: 1044-7318, 1532-7590. DOI: 10.1080/10447318.2018.1456150. URL: https://www.tandfonline.com/doi/full/10.1080/10447318.2018.1456150 (visited on 07/03/2024).

[5] Jaap Ham, Alexander Meschtscherjakov, and Cees Midden, eds. *Persuasive Technology: 18th International Conference, PERSUASIVE 2023, Eindhoven, The Netherlands, April 19-21, 2023, proceedings*. Lecture notes in computer science 13832. Meeting Name: PERSUASIVE. Cham: Springer, 2023. 426 pp. ISBN: 978-3-031-30933-5 978-3-031-30932-8.

[6] Maurits Kaptein, Boris De Ruyter, Panos Markopoulos, and Emile Aarts. "Adaptive Persuasive Systems: A Study of Tailored Persuasive Text Messages to Reduce Snacking". In: *ACM Transactions on Interactive Intelligent Systems* 2.2 (June 2012), pp. 1–25. ISSN: 2160-6455, 2160-6463. DOI: 10.1145/2209310.2209313. URL: https://dl.acm.org/doi/10.1145/2209310.2209313 (visited on 07/03/2024).

[7] Maurits Kaptein, Panos Markopoulos, Boris De Ruyter, and Emile Aarts. “Personalizing persuasive technologies: Explicit and implicit personalization using persuasion profiles”. In: *International Journal of Human-Computer Studies* 77 (May 2015), pp. 38–51. ISSN: 10715819. DOI: 10.1016/j.ijhcs.2015.01.004. URL: https://linkinghub.elsevier.com/retrieve/pii/S1071581915000075 (visited on 07/03/2024).

[8] Judith Masthoff, Floriana Grasso, and Jaap Ham. “Preface to the special issue on personalization and behavior change”. In: *User Modeling and User-Adapted Interaction* 24.5 (Dec. 2014), pp. 345–350. ISSN: 0924-1868, 1573-1391. DOI: 10.1007/s11257-014-9151-1. URL: http://link.springer.com/10.1007/s11257-014-9151-1 (visited on 07/03/2024).

[9] Daniele Pretolesi, Lenart Motnikar, Bieg Till, and Jakob Uhl. “Notifying Users: Customisation Preferences for Notifications in Health and Well-being Applications”. In: *Persuasive Technology*. Ed. by Alexander Meschtscherjakov, Cees Midden, and Jaap Ham. Vol. 13832. Series Title: Lecture Notes in Computer Science. Cham: Springer Nature Switzerland, 2023, pp. 410–424. ISBN: 978-3-031-30932-8 978-3-031-30933-5. DOI: 10.1007/978-3-031-30933-5_26. URL: https://link.springer.com/10.1007/978-3-031-30933-5_26 (visited on 07/03/2024).

[10] Thomas A. Widiger and Cristina Crego. “The Five Factor Model of personality structure: an update”. In: *World Psychiatry* 18.3 (Oct. 2019), pp. 271–272. ISSN: 1723-8617, 2051-5545. DOI: 10.1002/wps.20658. URL: https://onlinelibrary.wiley.com/doi/10.1002/wps.20658 (visited on 07/04/2024).

[11] Claire Zuppinger, Patrick Taffé, Gerrit Burger, Wafa Badran-Amstutz, Tapio Niemi, Clémence Cornuz, Fabiën N. Belle, Angeline Chatelan, Muriel Paclet Lafaille, Murielle Bochud, and Semira Gonseth Nusslé. “Performance of the Digital Dietary Assessment Tool MyFoodRepo”. In: *Nutrients* 14.3 (Feb. 1, 2022), p. 635. ISSN: 2072-6643. DOI: 10.3390/nu14030635. URL: https://www.mdpi.com/2072-6643/14/3/635 (visited on 07/04/2024).