



Bilkent University

Department of Computer Engineering

Senior Design Project

InPackt

High-Level Design Report

December 24, 2021

Şükrü Can Erçoban - 21601437

Kadir Mert Laleci - 21601960

Vural Doğan Akoğlu - 21602479

Kutsal Bekçi - 21602163

Fadime Selcen Kaya - 21801731

Supervisor: Asst. Prof. Dr. Hamdi Dibeklioglu

Jury Members: Asst. Prof. Dr. Shervin Arashloo, Asst. Prof. Dr. Hamdi Dibeklioglu

This report is submitted to the Department of Computer Engineering of Bilkent University in partial fulfillment of the requirements of the Senior Design Project course CS491/2.

Table of Contents

Introduction	3
Purpose of the System	4
Design Goals	4
Availability	4
Accuracy	5
Response Time	5
Extendibility	5
Scalability	5
Privacy	6
License	6
Maintainability	6
Usability	6
Reliability	7
Definitions, Acronyms, and Abbreviations	7
Overview	8
Current Software Architecture	9
Open Food Facts	9
Proposed Software Architecture	10
Overview	10
Subsystem Decomposition	10
Presentation Layer	11
Application Layer	12
Data Layer	13
Hardware/Software Mapping	13
Persistent Data Management	14
Access Control and Security	14
Global Software Control	15
Boundary Conditions	15
Initialization	15
Termination	15
Failure	16

Subsystem Services	16
Logic Layer	16
User Manager	16
Database Manager	17
Search Manager	17
Product Manager	17
Product Recognition Engine	17
Product Suggestion Engine	18
Data Layer	18
Cloud Database	18
Pretrained Models	18
UI Layer	18
Login/Sign Up View	18
Home Page View	19
Product Scanning View	19
Product Search View	19
Product Detail View	19
Preference List View	20
Settings View	20
Consideration of Various Factors in Engineering Design	20
Public Health Considerations	20
Public Safety Considerations	21
Public Welfare Considerations	21
Global Considerations	21
Cultural Considerations	21
Social Considerations	22
Environmental Considerations	22
Teamwork Details	23
Contributing and Functioning Effectively on the Team	23
Helping Creating a Collaborative and Inclusive Environment	24
Taking Lead Role and Sharing Leadership on the Team	25
References	26

1.0 Introduction

After the industrial revolution, the number of packaged products has increased drastically. Nowadays markets are full of packaged products and comparing the products is a time consuming activity in our lives. Reading the table of contents for each product we bought, understanding chemicals in the product and deciding which one to put into the shopping cart became a growing problem as the chemicals and number of packaged products increased.

Health is without question the single-most important value in life. Without sound health, nothing else can be important. Therefore, great importance must be given to how we fuel our bodies. Many food items, especially packaged ones, contain additives that the average person is not even able to pronounce, let alone know the effects. People need to be educated about these ingredients in order to make informed choices about their most basic need.

Based on these issues, InPackt will offer a quick and healthy shopping experience and help to compare between products. Thanks to algorithms and developing technologies, we can compare and decide the best products for us without looking at their table of contents directly. The user only needs to point their phone camera at the product and InPackt will take care of everything from there. It will display to the user the ingredients, packaging information, dangerous content warnings as well as similar product recommendations based on user preferences.

In this report, the purpose of the system with an overview, current software architecture and subsystem services will be explained. The details of the and will be proposed software architecture illustrated with presentation, logic and data layers. In addition, consideration of various engineering factors in the project will be discussed. Finally, the

details of the teamwork throughout the project will be presented with tables as contributing and leading.

1.1 Purpose of the System

The main purpose of the InPackt application is to ensure that the user can shop in a healthy and safe way. InPackt informs the user about the suitability of the product for the health and preferences of the user according to the product content by using the real-time object recognition, and in case of any unsuitability, it guides the user by suggesting an alternative product to the user.

1.2 Design Goals

The design goals are listed below.

1.2.1 Availability

- InPackt application should be available 24 hours a day, 7 days a week for all users who have devices with iOS or Android operating systems.
- InPackt application should be available mainly for users who live in Turkey, and ideally worldwide.

1.2.2 Accuracy

- InPackt application should be able to display the ingredients of the product scanned by user with at least %90 accuracy.
- InPackt application should be %90 accurate when recommending products according to the personal characteristic specified by the user while creating a profile for the product scanned by the user

1.2.3 Response Time

- InPackt application should display the ingredient of the product in less than 3 seconds after the user scans.
- InPackt application should display the alternative products in less than 3 seconds.

1.2.4 Extensibility

- The design of the InPackt application should be extensible for future updates covering additional features according to the needs of the users.
- The user data of the InPackt application should be stored to be extensible for a Web application or other platforms.

1.2.5 Scalability

- InPackt application database should be scalable according to the increased number of users. Initially, at least 100 different devices can access simultaneously.

1.2.6 Privacy

- User information should not be stored and used outside the user's device.
- User information should be encrypted to store their own devices.
- The InPackt application should ask the user's permission for the alternative product recommendation system according to the private user information that has been created by the user.

1.2.7 License

- All used databases, libraries and the like must be licensed.

1.2.8 Maintainability

- The InPackt application should have a design that will allow the maintainability of the system without any problems with future updates and ensuring that some changed parts are not affected to others functionality.

1.2.9 Usability

- The application aims to present to the user what is in the content of the product that the user wants to buy, whether it is suitable for the profile created by the user, and if it is not suitable, what other alternative products are available to the user in a fast, easy and reliable manner.
- Tutorial and guideline for how to use the InPackt app is available within the app.

1.2.10 Reliability

- The information displayed on the application must be accurate.
- There must be a brief information page about most used and unknown harmful ingredients to be displayed when there is no internet connection for informing users when they need to be informed.

1.3 Definitions, Acronyms, and Abbreviations

Flutter: Flutter is an open source framework used for developing applications on mobile devices such as android ios and web applications as well.

PostgreSQL: PostgreSQL is an open source relational database management system to store the product data and object images which are used for object training.

Tensorflow: It is an end-to-end open source platform developed by Google used in machine learning and object recognition.

Starflot: It is a plugin that supports other languages such as Python, Java, Ruby, Rust etc. to use in Flutter.

Anaconda: It is a distribution of R languages and Python that simplifies package management and deployment. We use the open source packages for object recognition.

Training: For object detection, Yolov4-tiny model is used as a pre-trained model and it is trained in darknet framework.

Machine Learning Model: Machine learning is a model that learns to reach certain results by using a set of data with an algorithm provided to it.

API: Application programming interfaces help software program development by enabling programs to transfer functions and data safely and easily.

1.4 Overview

InPackt will be a mobile application developed for android and ios users and will be available in Google Play Store and App Store. InPackt will enable users to eliminate risky products specifically catered for them by scanning the product's package to see the harmful ingredients and further information about the contents. Aforementioned products can be of the types food, and if time and resources allow us, cosmetics, cleaning products or supplements.

The determination of the “risky” product or ingredient will be customized depending on the choice of the user. Customization can include specific unwanted ingredients, allergants as well as products that are gluten or lactose free, vegan, vegetarian or halal. The goal of our project is to provide an easy way to see the ingredients of the market products and shorten the decision process of choosing between products by individualized ingredient detector. A user will be able to look at alternative products to the one that they scanned depending on their customization criteria.

InPackt will also present the container type and information of any product with a package to further inform the user. If they want, the users will be able to factor in the sustainability and the recyclability of the packaging in their recommendations.

In the product information page, among the ingredients and container type, there will be calorie specifications as well if the product is a food, and the user will be able to choose to be recommended lower-calorie items.

InPackt will utilize real-time object recognition in order to identify the product. The user will point their camera toward the product and the recognized item will pop up on the screen. If time and efficiency allow us, we want to implement an Augmented Reality feature so that the instant that the app recognizes a product, it is displayed on the user's screen on the product. The object recognition is the main focus, and if it fails, the writings of the package will be read to match with a product. If that fails as well, the options of voice command, barcode scanner or manual product entry will be displayed.

The project will definitely be available for users in Turkey, in Turkish and English, and if the time and technology is sufficient, be available for users worldwide in English. The products will need to be searched in their own language, however apart from that the app can be used in both languages.

2.0 Current Software Architecture

The architecture of the product similar to our project is explained below.

2.1 Open Food Facts

Open Food Facts is an open-source, worldwide, collaborative project that aims to store information about branded food such as its ingredients, nutrition facts, and packaging, organized and divided by country [1]. The application uses a layered architectural approach. Since this approach is built around the database and all of its side applications use the database, MVC (Model View Controller) model is one of the best architectures for them. Moreover, Open Food Fact is kept developing by multiple people

through github repositories through each layer. Therefore, thanks to MVC modal, it is easier to keep and check new changes ideas in layers.

3.0 Proposed Software Architecture

The architecture of InPackt is described below.

3.1 Overview

The software architecture design is based on the Model View Controller pattern, and has three layers corresponding to the layers of the pattern. Each layer further develops on its own as each of them are explained below.

3.2 Subsystem Decomposition

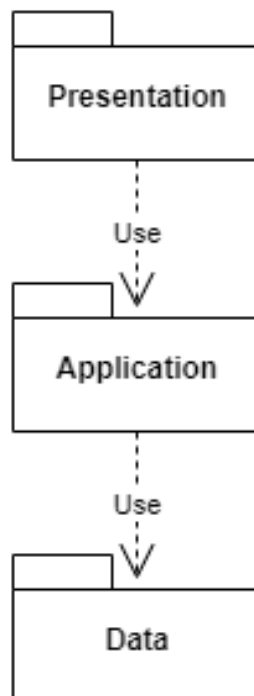


Figure 1: The Subsystem Decomposition of the project.

The project follows a Model-View-Controller pattern with a three-layered design. The presentation layer can only access the application layer, and the application layer uses the data layer. The reason we chose this model was to have a low coupling and high coherence among packages. The layers are hierarchical, and have as little connections as possible in order to achieve the goal of low coupling.

3.2.1 Presentation Layer

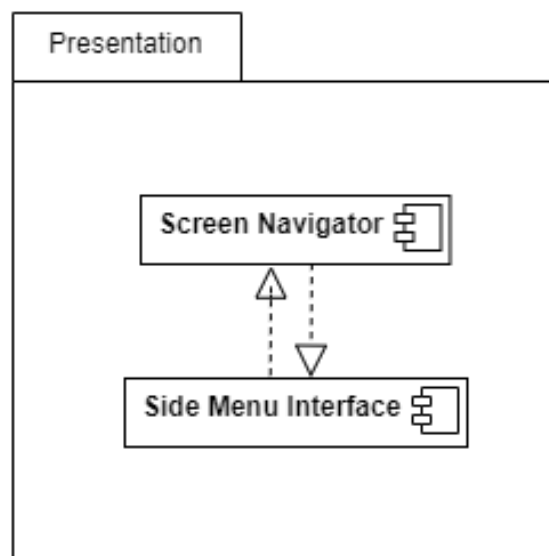


Figure 2: Presentation Decomposition.

The presentation layer represents the user interface or the view layer of the MVC pattern. In this package, there exists two components: the screen navigator and the side menu interface. The screen navigator component contains all complete page views which are the login/sign up view, home page view, product scanning, search and details view, profile view, preference list view as well as the settings view. The side menu interface contains the overlay of the side menu which can be accessed from nearly all of the listed pages above which means any of them can access this interface, and through the interface some views can be accessed.

3.2.2 Application Layer

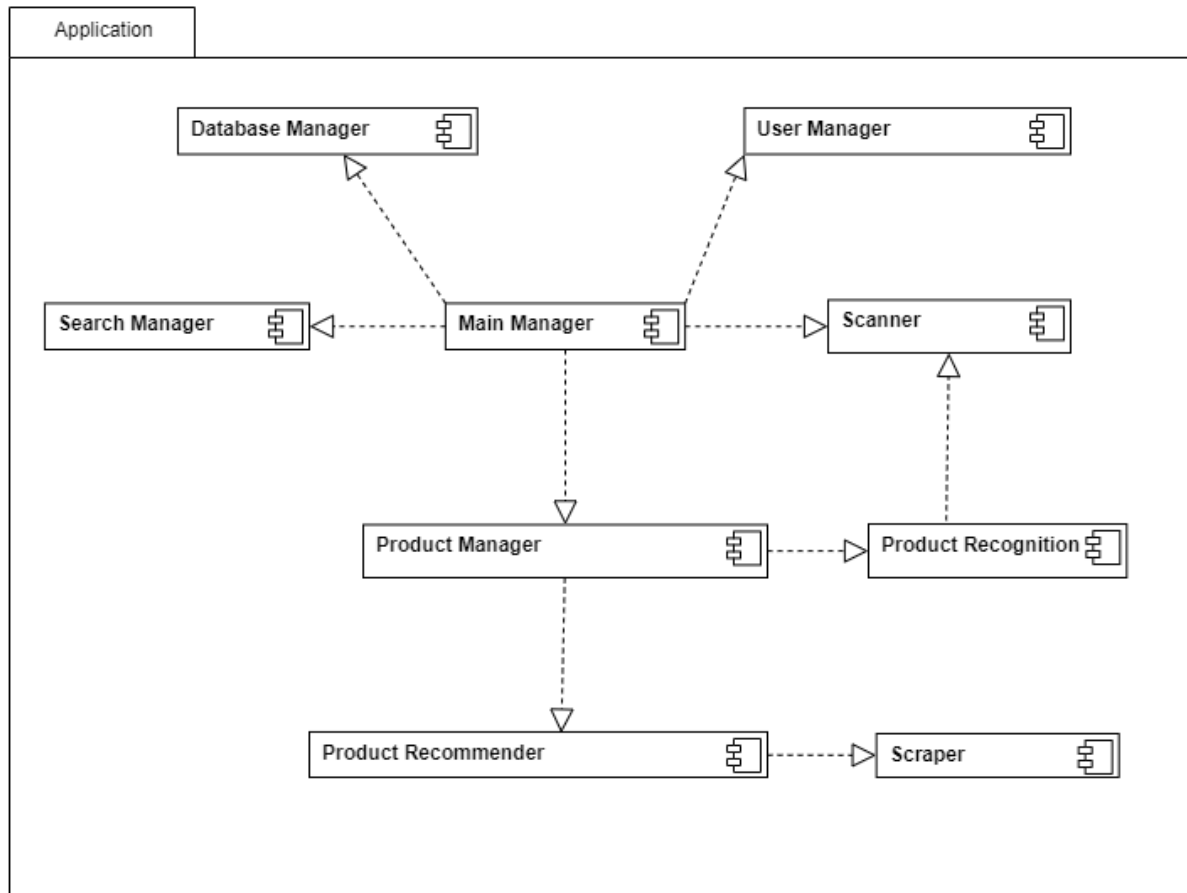


Figure 3: Application Decomposition.

The application layer is made of many different components that are implemented with one or more classes. The main manager controls most of the application's features and services. It uses the database manager, user manager, search manager, scanner and the product manager. The product manager itself uses the product recognition engine and the product recommender engine, which uses the web scraper.

3.2.3 Data Layer

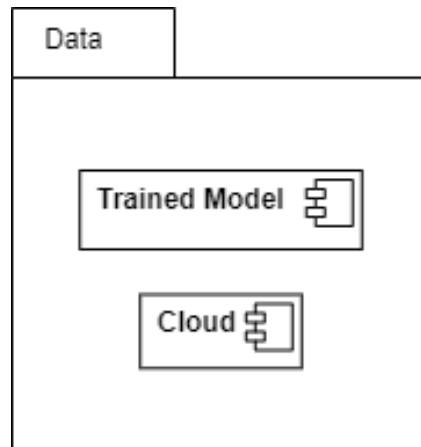


Figure 4: Data Decomposition.

The data layer consists of two different components for storing data: the trained model and the cloud databases. The data for the trained model consists of needed data for the object recognition model. The cloud database stores the user information as well as any and all preferences and history. It also stores information about all of the products and ingredients that can be accessed through the app, through scanning or searching, and their relevant information.

3.3 Hardware/Software Mapping

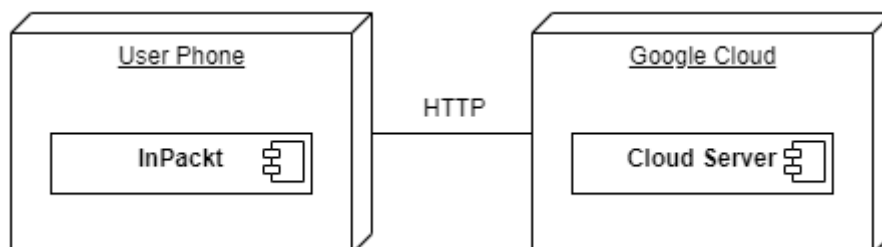


Figure 5: Hardware/Software Mapping.

The application will be run on the user's phone, whether Android or Iphone, and will be connected to the Cloud server when data extraction is required.

3.4 Persistent Data Management

The data regarding users, products and ingredients is going to be stored on a cloud server. The service that we are using is Google Cloud. We chose our region as Europe Central, specifically Warsaw. This was the option closest to us which will provide a more efficient application.

3.5 Access Control and Security

InPackt has two types of usages: one with signing in and one without. The users who sign in need to first create a profile with their email and a password. This information will of course be kept private, and off access to everyone. These users will be able to keep track of their action history and favorites as well as unwanted. The application also allows users to use the services it provides without the need for an account, except for the history and saved products services. The user will access the data of the products and ingredients through the cloud and nothing about the user will be stored in return. This type of user is called a guest user, as they do not affect the database, just access it. These will provide our application with security and reliability.

3.6 Global Software Control

InPackt application is designed according to client-server software architecture. The new request from the server is needed when a new account is created, or edited in the preferences, such as the product is added to the favorites, and the ingredient is added to the favorites etc. All the information about the products and users are stored in Google Cloud services that's why no storage is needed and ensure the synchronization of these separate parts of the project whenever a request is needed from the server the response will be received and that keeps Inpackt project able to maintain balanced workload.

3.7 Boundary Conditions

3.7.1 Initialization

Depending on the platform users need to download the app either from the Google Play Store or App Store. Users can use the app directly after installing as a guest however, they need to register with a valid email address in order to benefit from personalized recommendations. The internet connection is a must since the application uses the cloud server.

3.7.2 Termination

The users are able to log out from the side menu view, and be directed to the login and sign up page. If they are not logged out when they exit the application, they will be logged in the next time it starts. When a user exits the app, all the working

services are terminated and the application stops, meaning it does not continue working in the background. The camera does not record anything when it is not purposely used by the user.

3.7.3 Failure

If the internet connection is lost during scanning or searching when the application is not able to get the data from the cloud, the application will present a popup to the user. This popup will inform the user that there is not internet connection, and their action resulted unsuccessfully.

4.0 Subsystem Services

4.1 Logic Layer

4.1.1 User Manager

This manager is responsible for controlling the actions that are performed in the UI layer for end-users and it can be authentication for profiles, editing the user profile, creating and updating preference list, favoriting or disliking products. When these actions happen, this manager should update the database with the database manager.

4.1.2 Database Manager

The Database Manager is responsible for processing the data in the data layer to create or update actions which are performed in other subsystems.

4.1.3 Search Manager

The search manager is responsible for searching products or ingredients which are wanted from the end user. The end user can filter his search for specific searches. For each search change, the manager updates the results from the database.

4.1.4 Product Manager

Product manager handles every action for product based ones with the help of other subsystems. The actions can be searching, scanning, favoriting, disliking. The used subsystems are user manager for favorites and dislikes, product recognition engine for detecting objects from scanning and product suggestion engine for suggesting a product according to preference list.

4.1.5 Product Recognition Engine

This engine connects the phone camera and according to the scanned product, it recognizes the product with a trained custom object detection model and the highest similarity of the product will be given as detected.

4.1.6 Product Suggestion Engine

The engine suggests a product according to the user's preference list and this engine works when a product is searched or scanned. It suggests the same category product from the database, if there is a better product in terms of user preference.

4.2 Data Layer

4.2.1 Cloud Database

The database stores user information, product ingredient details, favorites, unwanted and preference list of the user and past scans which are performed earlier. These datas are used with the managers in the logic layer.

4.2.2 Pretrained Models

This model contains the dataset for the object recognition for detecting the product by similarity. These are trained with many images to recognize the product.

4.3 UI Layer

4.3.1 Login/Sign Up View

The page for user login/sign up with their information. According to their actions, the user manager is used to perform registration or logging in.

4.3.2 Home Page View

This page is the opening page after login/signup action and the page is like a dashboard which contains the user's action in the application. You can see the last scanned product, favorite/unwanted products and ingredients.

4.3.3 Product Scanning View

The view is for scanning products and the page contains the phone's camera view. In this page, the product recognition engine is used to detect the objects.

4.3.4 Product Search View

The view is simply for searching products without scanning. The page contains search bar, filter tab and the results according to search.

4.3.5 Product Detail View

The page can be reached from scanning or selecting from searched results. The page contains the ingredients and the suggested product and for these actions product suggestion engine and product manager is used.

4.3.6 Preference List View

In the preference list page, the user can edit his favorite/unwanted list and he can enter his preference in this page. User manager is used from the logic layer.

4.3.7 Settings View

This page enables users to customize the application to themselves, these changes can be font size of the application, theme and account settings which is the user information.

5.0 Consideration of Various Factors in Engineering Design

As the name of the application implies, our project has an impact on many keystones of human life such as health, safety, welfare, global, cultural, social, environmental and economical. Effects of the algorithm in various categories are discussed below.

5.1 Public Health Considerations

One of the main uses of the InPackt is to show the users unhealthy ingredients and help users to avoid preferable ingredients. Therefore, public health will be enhanced by InPackt as the user scanning products. However, Inpackt aims at 90% success in the product recommendation, it may suggest a wrong product to a user with an allergic condition, which may cause health problems for the user.

5.2 Public Safety Considerations

The required data which are username, email password will not be shared anywhere and the safety of the information will be considered. The required data will be used for storing the preference list of the user.

5.3 Public Welfare Considerations

InPackt doesn't have negative impact on public welfare. On the other hand, it can impact positively because InPackt will guide the users to do better shopping for themselves. Moreover, it will be a free application. Therefore, the application might increase public welfare with its guidance and affordability.

5.4 Global Considerations

InPackt will focus on as many markets as possible in the world, so products will not be bought only from Turkey. Moreover, the application language will be English and this helps to be adapted easily to other countries. Therefore, by focusing on global market and main language of the application, InPackt has a global perspective as well.

5.5 Cultural Considerations

InPackt will not be affected by cultural factors because it does not have cultural features to affect.

5.6 Social Considerations

A typical InPackt user interacts directly with the data instead of other users like social networks. Therefore, InPackt will not be affected by social factors.

5.7 Environmental Considerations

InPackt will show the product packaging type which will be related to recycling materials and harmful material for the environment. Therefore, InPackt increase the awareness of the environmental issues and also it increases the sustainability of the products.

In the table below, each factor's levels are indicated and each factor is evaluated between 0 and 10.

	Effect Level	Effect
Public Health Factors	9	Health problem due to allergies
Public Safety Factors	8	Leak of user information
Public Welfare Factors	5	Increase public welfare
Global Factors	3	Balance product standard globally
Cultural Factors	0	No effect
Social Factors	0	No effect
Environmental Factors	5	Recycling material in package

Table 1: The factors and its levels

6.0 Teamwork Details

6.1 Contributing and Functioning Effectively on the Team

Member Name	Contribution in Work Package
Fadime Selcen Kaya	Specification Report Analysis Report High Level Design Report Database Development Data Processing
Vural Doğan Akoğlu	Specification Report Analysis Report High Level Design Report User Interface Frontend Development
Şükrü Can Erçoban	Specification Report Analysis Report High Level Design Report Object Recognition Development
Kadir Mert Laleci	Specification Report Analysis Report High Level Design Report User Interface Frontend Development
Kutsal Bekçi	Specification Report Analysis Report High Level Design Report Object Recognition Development

Table 2: Contribution in Work Package

Critical decisions about the project were taken as a group and divided into certain sub-sections and distributed to the group members as a similar workload. First of all, we divided the project into 3 different sub-areas. The first of these areas was the user interface and program pages related to front end development. Our group members Mert and Vural were appointed as the sub-group to deal with this part. In order to develop object recognition

development, which is one of the other main functions of the InPackt, Kutsal and Şükrü were appointed as a subgroup. Finally, our group member Selcen took the part related to database development and data processing.

In project documentation, group meetings were held in each of the specification, analysis and high-level design reports and shared with the group members in parts and in checkpoints which we determined before, put all of the parts together and discussed about the topics and repeat the process as following distribution of task among group member, meeting on checkpoints date and discussed.

For the project demo, the subgroups brought together the tasks they had done and started testing.

6.2 Helping Creating a Collaborative and Inclusive Environment

We have used the Jira environment which is a planning and tracking program for software applications. We have assigned tasks with deadlines to keep track of the process. On the other hand, we used github to check and pull the current process of the project. In our project, we have a github for each project workforce that are frontend, backend, data processing. In the end we are planning to gather all of our sources in one github. Before each task, we met to make work sharing among us and save all of the meetings in a spreadsheet.

6.3 Taking Lead Role and Sharing Leadership on the Team

WP#	Work package title	Leader	Members involved
WP1	Analysis and High-Level Design Decisions UI API connection	Kadir Mert Laleci	Everybody
WP2	Database Development and Data Processing	Fadime Selcen Kaya	Everybody
WP3	Object Recognition Development	Şükrü Can Erçoban	Everybody
WP4	Machine Learning For Product Suggestion	Kutsal Bekçi	Everybody
WP5	Low-Level Design	Kadir Mert Laleci	Everybody
WP6	User Interface - Demo	Vural Doğan Akoğlu	Everybody

Table 3: Leader Role of Work packages

We have decided to assign each member a leadership role to be fair. The roles assigned to each person are presented in the table above. Other than the leader, each member contributed to each of the work packages, with the guidance and assistance of the leader.

7.0 References

[1] “The Open Food Facts Team.” Open Food Facts. Accessed December 24, 2021

<https://world.openfoodfacts.org/who-we-are>.