University for Applied Sciences

Informatics Department

Applied Informatics

To be defined

Documentation for the Architecture of an Mobile Application for Preventing Food Waste

Bruno Macedo da Silva
676839
inf3645@hs-worms.de

Supervisor	Prof.Dr. Volker Schwarzer
Working Period:	Summer Semester 2022
Due Date:	31.Juni 2022

Inhaltsverzeichnis

Abbreviations

FAO Food and Agriculture Organization of the United Nations.

UN United Nations.

Glossary

Activity Diagram This kind of diagram shows the behavior of a system, it depicts in a graphical fashion the logic of a single use case. It falls in the category of behavior workflow [Baresi, 2009].

App The word will be used here to reference the mobile application to be developed.

Class Diagram This kind of diagram presents the structure of a system with its classes, attributes, methods and relationships[IBM, 2004].

Client Since we have two different kind of stakeholder that will use the app, the word client will define the stakeholder that place an order in the app.

Federated Login Authentication method in which users use existing accounts to gain access to another domains or systems without the need of creating new credentials. The authenticity of a user is attested by service and granted to another [Robinson, 2019].

Mobile Payment Gateway Those services works as an intermediary between customer, merchant and bank/credit card company. Here a payment request is sent to the gateway and forwarded to the approval instances. The core functionality of those gateways is the cryptography within the communication steps[Vilmate, 2019].

Provider The second kind of stakeholders are those who offer their products. They can begin restaurants, bakeries, pastries and similar.

Stakeholder Describe all kind of potential person or entity that may have interest using the app.

Use Case Diagram This kind of diagram presents the main requirements and functionality of a systems. It displays a simplified overview of core purpose of the application [Waykar, 2015].

1 Introduction and Goals

According to the Food and Agriculture Organization of the United Nations (FAO) in 2019 931 millions tonne of food were wasted [FAO, 2013]. This has environmental, but special social consequences. In a world where approximately 9.9% of the [AAH, 2022] population suffers from hunger that waste percentage sounds paradoxal.

According to United Nations (UN) 5% of the global food loss and waste comes from restaurants [UN, 2022]. The solution for this problem must be locally applied so its effects can be seen in a global structure. To do so we propose to develop a mobile application that connects restaurants, bakeries and or pastries to clients. The former would offer their remaining products, which are still consumable, prior to the closing time, to a small price and the latter would browser in the app to find which shops are offering products.

We as "Clean Up the Word (R)" are a rising StartUp whose main concerns is to find environmental solutions to daily problems. Our portfolio includes projects about management of waste and optimization of household water usage. This product we want to develop targets small communities, like small cities or regions within a big city, to reduce the amount of wasted consumable food.

With our project we want to achieve the following goals:

- Connect provider with clients, so the former can offer products that the latter can purchase
- Collect statistical data about waste reduction within the provider
- Promote reduction of food waste that still could be consumed
- Allow clients to have a different dining experience.
- Allow provider to promote their products and gather new clients.

1.1 Design Purpose

The main purpose of this architecture is creating exploratory prototype of an App. We aim to test it with potential Stakeholder and regions to analyze their general acceptance and wishes [Cervantes and Kazman, 2016] and get a fast feedback.

This prototype will also make it feasible to identify unknown needs an wishes of the potential Stakeholder, so we can eventually increase the scope of functionality. Exploring this domain will also provide us with information regarding the behavior of our Stakeholder when it comes to buying and serving food that would be wasted, but is still consumable.

1.2 Requirement Overview

The following functionalities should provide a brief description of the main requirements of the app:

Id	Requirement	Description
F-1	Register as Client.	A Client can register to the app with its e-mail.
F-2	Login	After registration Client can login into the app.
F-3	Purchase option	A registered Client can purchase an available offer (see F7).
F-4	Filter/search options	A Client can perform filter and search actions for products.
F-5	Register as Provider	A Provider can register with store and add logos and picutres.
F-6	Create offer	A registered Provider can publishes what products they are offering with price and ammount.
F-7	Upload offer	A registered Provider can add, edit or remove offers to his catalog.
F-8	Check orders	A registered Provider can check all existing orders.

The following Use Case Diagram displays an overview of the primary functionality of the app:

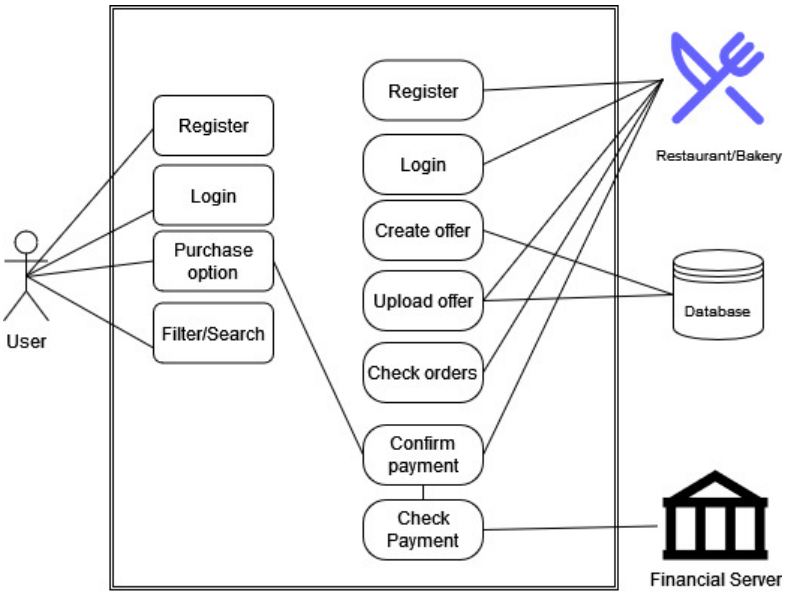

Abbildung 1: Preliminary functions

1.3 Stakeholders

The main stakeholders of this app are described in the table below:

Stakehoder	Description
Providers	Owner of a restaurant, bakery or pastry.
Clients	Person who wants to buy last minute products from a provider.
Developers	Team in charge of creating the application using existing tactics and creating new solutions.
Boarding Committee of "Clean Up the Word (R)"	Members of the management team who wants to delivery environmental solution do daily problems
Environment Activist	Part of the society who aims to find environmental solutions to daily problems.

2 4+1 Architectural View Model

In this section we will describe the App using the 4+1 Architectural View Model. With this model we will represent the App using five different views, which should focus on specific elements of the project. Each view provide a different purpose [Kruchten, 1995]. For this project we will provide the 3 following views of the 4+1 Architectural View Model:

- **Scenario view**: simple description for the end user
- **Behaviour view**: description of the existing processes
- **Structural view**: object-oriented decomposition

The scenario view was presented in the section **??** of this project.

2.1 Behaviour view

The following Activity Diagram depicts the register and login procedure within the app.

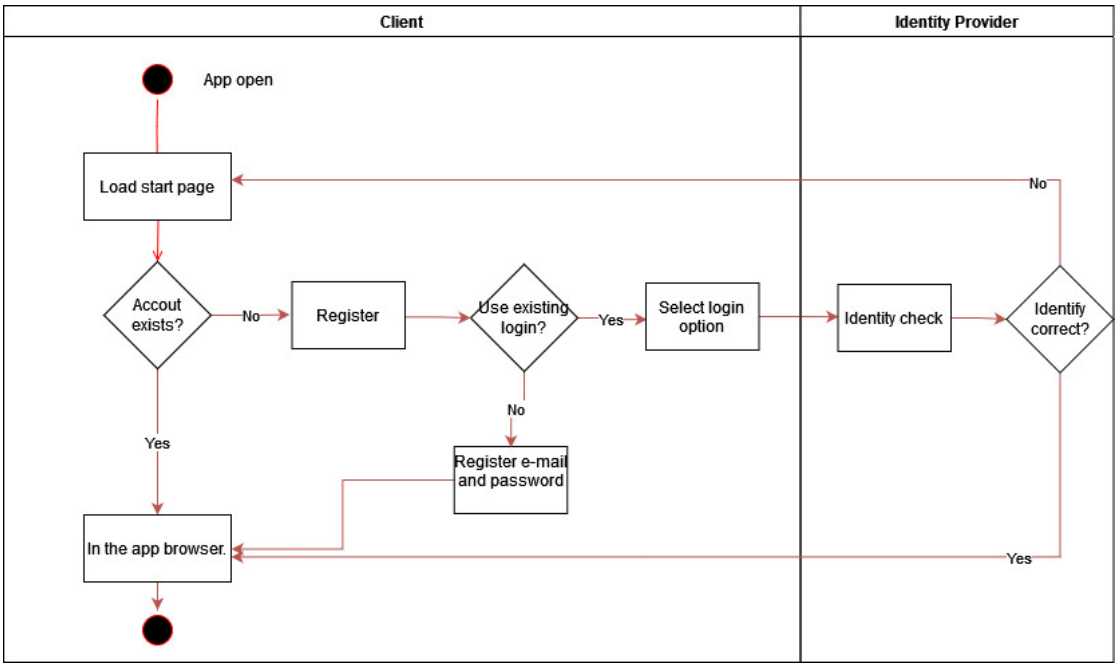

Abbildung 2: Login procedures

2.2 Structural view

To describe this view we choose a Class Diagram. With it we may provide a static description of elements within the structure of our system. They can also be used during the programming process to display what is needed to be done.

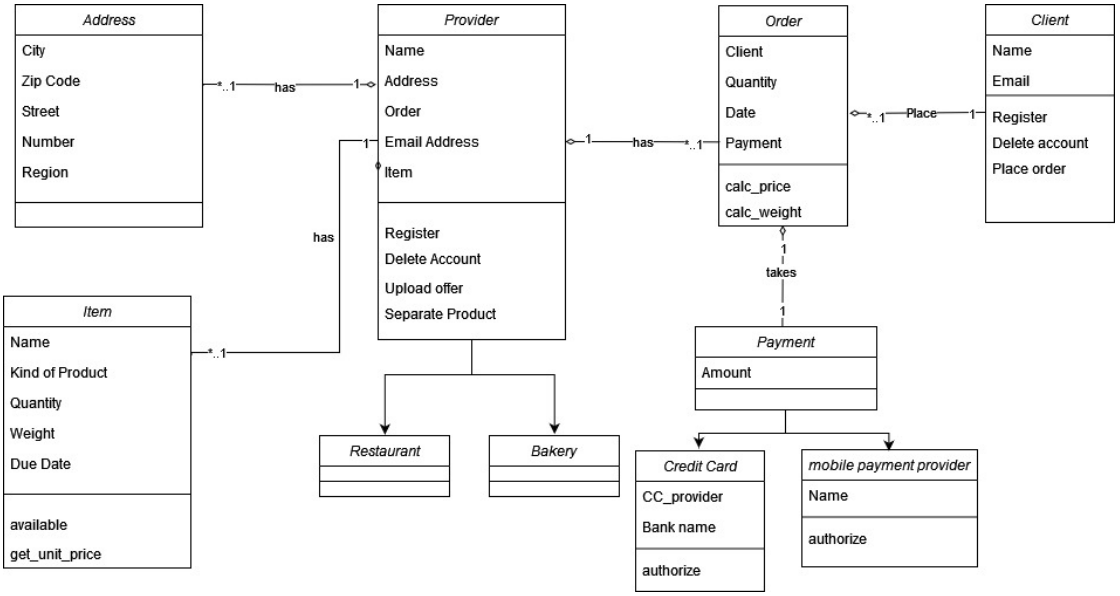

Abbildung 3: Classes of the project

3 Architecture Constraints

In this project we must distinguish between Technical and Organizational Constraints. The former describes specific elements of the project, like programming language, released platform (e.g. operational systems) and technical decisions related to the functionalities. The latter deals with management elements[Franzen and Thoms, 2020] (e.g time, budget and team). The following tables describes the technical and the organizational constraints of this project:

Technical		
Id	Constraint	Description
CT-T-1	Programming Language	A multilanguage (Java, Kotlin, iOS, Swift) approach increases the maintainability burden and consequently the costs (see CT-B-4). It can also interfere with compatibility with different kind of device.s
CT-T-2	Platform	Offering the application for different platforms (iOS and/or Android) increases costs for maintainability and requires a bigger team. Since the prototype should run during the first year mainly to gather information about consumer behavior the costs in this test phase can increase rapidly if we decide to develop for the most common platforms.
CT-T-3	Payment	One the one hand creating an own payment framework can gives full control of the application, but on the other hand it will required specialized team and increases costs and time (see CT-B-4).
CT-T-4	Payment gateway	Using existing Mobile Payment Gateway reduces development time, but demands fully Interoperability of the app with the existing gateways. It may also be a problem if the Client don't use this kind of payment method.
CT-T-5	Login	Using existing Federated Login decreases development time, but like CT-T-4 demands fully interoperability of the app with appliances. It may also be a problem if the Client don't trust this kind of login.

Organizational		
Id	Constraint	Description
CT-B-1	Time to first prototype release	How much time is acceptable from starting the project until we have a functional prototype that can be used by our user?
CT-B-2	Development Team	The existing team can cover the main existing platforms, but their availability may be restricted to due work on other projects Specially for the maintainability of the app it can represents a problem.
CT-B-3	Analytical Team	During running phase of the prototype it will be necessary to have a team in charge of evaluating and interpreting the collected data, to find out if the goals are being achieved.
CT-B-4	Budget	Since this application falls in the category "'middle app" according to [SPD LOAD, 2019] the available budget of US$ 150.000 should cover the development of the main functionaliy and the data analisys (see CT-B-3)

4 Quality Requirements

4.1 Quality Tree

4.2 Evaluation Scenarios

From the requirements, 1.2, we could develop the following uses cases and depict the main quality attributes of this project.

Use Case	Description
UC-1: Register as Client	The Client registers an e-mail address.
UC-2: Login	The Client logins in to the system.
UC-3: Places an order	The Client chooses a Provider.
UC-4: Register payment	The Client registers a payment method.
UC-5: Register as Provider	The Provider registers their facility and products.
UC-6: Update availability	The Provider uploads their product catalog.

With the given use cases we will then be able to define the major quality attributes that are involved in the development of this application. We want those qualities to be measurable and testable so we can verify if the system meets the needs our stakeholders [Cervantes and Kazman, 2016].

ID	Quality Attribute	Scenario	Associated Use Case
QA-1	Performance	A Client registers his/her e-mail address and can immediately browse in the app.	UC-1
QA-2	Performance	A Client opens the app and he can immediately browse in the app.	UC-2
QA-3	Performance	A Client chooses a Provider and places his order. After the confirmation of payment, a push-message is displayed in the app confirming the purchase.	UC-3
QA-4	Usability	A Provider is able to register his company, specify the kind of products he offers and upload a logo or picture of his shop.	UC-5
QA-5	Usability	A Provider is able to update in the app if he is offering for that day any product.	UC-6
QA-6	Interoperability	A Client can register his e-mail using another account (Google, Microsoft, Facebook) in a Federated Login	UC-1
QA-8	Interoperability	A Client can pay the order using a Mobile Payment Gateway (e.g. Stripe, Square, PayPay, SecurePay)	UC-4

The defined quality attributes are represented in the following scenarios:

Performance	
Scenario	Value
Source	Client
Stimulus	wishes to create an account
Artifact	app
Environment	weekend between 3 and 7 PM
Response	immediate access to the app
Response Measure	time between confirmation and access
Source	Client
Stimulus	wants to search for a Provider
Artifact	app
Environment	peak period, between 6 and 7 pm on a Friday
Response	immediate access to the offers
Response Measure	how quick does client's device get update of availabilities
Source	Client
Stimulus	places an order
Artifact	platform
Environment	peak period, between 6 and 7 pm on a Friday
Response	confirmation of payment / payment declined
Response Measure	How long did take until the client get the confirmation/declined of payment?

Usability	
Scenario	Value
Source	Provider
Stimulus	wants to offer his remaining products in the app
Artifact	app
Environment	working time, during afternoon
Response	offer available in the app
Response Measure	How long did the registration and upload process take? How many and what kind of error messages did the Provider get?
Source	Registered Provider
Stimulus	wants wants to make a last minute offer
Artifact	app
Environment	peak period, between 4 and 7 pm on Friday
Response	immediate availability of the offer in the app
Response Measure	How long did it take to upload an offer? How many and what kind of error messages did the Provider get?
Source	Registered Client
Stimulus	wants to search/filter offers
Artifact	app
Environment	peak period, between 4 and 7 pm on Friday
Response	display of the filter/search input
Response Measure	What kind of inputs did the user has to place until he/she finds what he/she wants? Did he have to type anything or were filter/search options available? How long it takes until the client finds a product?

Interoperability	
Scenario	Value
Source	Client
Stimulus	wants register using a Federated Login
Artifact	app and Federated Login provider
Environment	peak period (on the context of the Federated Login provider)
Response	authentication succeed or failed
Response Measure	How much data was transmitted and how much was queued?
Focus	System overload [Kasunic and Anderson, 2004]
Source	Client
Stimulus	wants to pay using existing mobile payment account
Artifact	app and Mobile Payment Gateway
Environment	peak period (on the context of the gateway)
Response	confirmation / declined
Response Measure	Total amount generated data in the app that are transferred and processed by the gateway
Focus	Connectivity and System overload [Kasunic and Anderson, 2004]
Source	Client
Stimulus	click on purchase
Artifact	app and Mobile Payment Gateway
Environment	peak period (on the context of the gateway)
Response	confirmation / declined
Response Measure	Time from clicking purchase until time of confirmation/declined
Focus	Data latency [Kasunic and Anderson, 2004]

Literaturverzeichnis

[AAH, 2022] AAH (2022). World hunger: Key facts and statistics 2022. *actionagainsthunger.org*. https://www.actionagainsthunger.org/world-hunger-facts-statistics, Accessed 18th May 2022.

[Baresi, 2009] Baresi, L. (2009). *Activity Diagrams*, pages 41–45. Springer US, Boston, MA. https://doi.org/10.1007/978-0-387-39940-9_9, Accessed 26th May 2022.

[Cervantes and Kazman, 2016] Cervantes, H. and Kazman, R. (2016). *Designing Software Architectures: A Practical Approach*. Pearson Education, Boston.

[FAO, 2013] FAO (2013). Food wastage: Key facts and figures. *fao.org*. https://www.fao.org/news/story/en/item/196402/icode/, Accessed 18th May 2022.

[FAO, 2022] FAO (2022). 17 *fao.org*. https://www.fao.org/food-loss-reduction/news/detail/en/c/1378973/, Accessed 18th May 2022.

[Franzen and Thoms, 2020] Franzen, E. and Thoms, M. S. (2020). Architectural drivers in modern software architecture. https://medium.com/@janerikfra/architectural-drivers-in-modern-software-architecture-cb7a42527bf2, Accessed 18th May 2022.

[IBM, 2004] IBM (2004). What is Class Diagram? https://developer.ibm.com/articles/the-class-diagram/, Accessed 18th May 2022.

[Kasunic and Anderson, 2004] Kasunic, M. and Anderson, W. (2004). Measuring systems interoperability: Challenges and opportunities. Technical Report CMU/SEI-2004-TN-003, Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA. http://resources.sei.cmu.edu/library/asset-view.cfm?AssetID=6871. Accessed 1st June 2022.

[Kruchten, 1995] Kruchten, P. (1995). The 4+1 View Model of architecture. *IEEE Software*, 12(6):42–50. https://doi.org/10.1109/52.469759, Accessed 18th May 2022.

[Robinson, 2019] Robinson, M. (2019). What does federated login mean? a simple, detailed answer. https://carvesystems.com/news/what-does-federated-login-mean-a-simple-detailed-answer/. Accessed 1st June 2022.

[SPD LOAD, 2019] SPD LOAD (2019). How much does it cost to develop an app in 2022? cost breakdown. https://carvesystems.com/news/what-does-federated-login-mean-a-simple-detailed-answer/. Accessed 1st June 2022.

[UN, 2022] UN (2022). Stop food loss and waste, for the people, for the panet. *un.org*. ht

tps://www.un.org/en/observances/end-food-waste-day, Accessed 18th May 2022.

[Vilmate, 2019] Vilmate (2019). Payment gateway integration in a mobile application. https://vilmate.com/blog/payment-gateway-integration-in-a-mobile-application/, Accessed 31st may 2022.

[Waykar, 2015] Waykar, Y. (2015). role of use case diagram in software development. *International Journal of Management and Economics*. https://www.researchgate.net/publication/322991847_role_of_use_case_diagram_in_software_development. Accessed 1st June 2022.