



Project Course

Requirements Documentation



Författare: Team 1

Examinator: Tobias
Andersson

Termin: VT-19

Ämne: Project Course in CS

Kurskod: 1DV005



Index

| | |
|------------------------------|----------|
| Index | 2 |
| Introduction | 3 |
| Purpose | 3 |
| Scope | 3 |
| Definitions | 3 |
| Overview | 3 |
| Overall description | 4 |
| Product perspective | 4 |
| Product functions | 4 |
| Stakeholders | 4 |
| General constraints | 5 |
| Assumptions and dependencies | 5 |
| Specific requirements | 6 |
| User Stories | 6 |
| Functional requirements | 7 |
| Customer | 7 |
| Administrator | 7 |
| System | 8 |
| Non-functional requirements | 8 |
| Security | 8 |
| Database | 8 |
| Requirements analysis | 8 |
| Requirements Risks | 8 |
| Traceability analysis | 10 |
| Risks and Requirements | 10 |
| Requirement dependencies | 11 |
| UML | 13 |
| Use Case Diagram | 13 |
| Routing diagram | 14 |
| Components packages diagram | 14 |
| Sprint 1 Components diagram | 15 |



Introduction

This document will give an overview of the requirement engineering process. Its aim is to provide descriptions and definitions.

Purpose

The aim of this document is to explain all the functional and non-functional requirements. This document version was last modified with date 15/04/2019 Week 15 and will be refined when needed. This document may be used by all the stakeholders working in the project, that includes non-technological stakeholders with a medium level of knowledge in computer science.

Scope

<<NAME OF THE STORE>> will be able to display a number of items, allowing a customer to create a list of items to buy and finally place an order. In the case of an administrator, it will be able to check the orders placed by customers and create new items, modify the items already saved and delete items from the database.

Definitions

Customer: User that is able to use the system to buy items.

Administrator: User that is able to manage the items and orders.

User: Person that navigates and uses the system, independently of their role.

CRUD: Create, retrieve, update and delete. Mostly used to handle objects and items in the system.

Persistence: Data storage, Database.

DB: Database.

Frontend: Functionalities with which the user interacts directly.

Backend: Functionalities that are not directly accessed by the user.

Overview

This document sections will discuss the overall description of the system, as the product perspective, functions, stakeholders, general constraints and other assumptions. Further below, the document elaborates on the specific requirements for this project. Lastly, the system will provide traceability and UML diagrams to get a better understanding of the development.



Overall description

This sections will describe the system, its functionality, constraints and assumptions, and the stakeholders that will use it.

Product perspective

<<NAME OF THE STORE>> will handle the process of buying and managing items by using a web application. The system consists of two parts: a customer system and an administrator system. The customer system will handle all the functionalities of showing the items, selecting items and checking out. The administrator system will add functionalities, as item CRUD and order management. All the information will be stored in persistence, in this case, a Firebase database.

Product functions

Our application has to be able to manage two different user modes, a user-client and a user-administrator. The functionality of both systems will be explained in the next chapters.

- Customer browse and search for products
- Customer order products
- Administrator handle products
- Administrator handle orders
- The shop can contain any number of products
- All products shall be stored in a database
- Every product shall belong to a product category

Stakeholders

- Customer
 - Login into the system
 - Browse the e-commerce
 - Place orders
- Administrator
 - Login into the system
 - CRUD items
 - handle orders
- Maintenance IT
 - Maintain the system.
 - Adapt the system to future changes.
 - Update the persistence
 - Inform the Manager
- Developers
 - Design the system
 - Develop the system



- Manager
 - Get profits
 - Coordinate departments

General constraints

- Time constraints
- Internet constraints

Assumptions and dependencies

All the implementation of a database and authentication will be done using a Firebase account.

User Stories

| Code | User Story |
|------|---|
| 4722 | As a customer, I can sign in so that I can shop |
| 4723 | As a team, we want to be able to collaborate and merge our code and functionalities |
| 4724 | As a (store) administrator I can sign in so that I can manage products |
| 4725 | As an administrator, I can add a product category so that products can be uploaded to a place where customers can find it |
| 4726 | As an administrator, I can upload a product to a category so that my customers can make purchases |
| 4727 | As an administrator, I can specify how many copies of a product are available so that I don't sell products I don't have |
| 4728 | As an administrator, I can update product details so that it accurately describes the product |
| 4729 | As an administrator, I can delete a product so that I don't show non-existing products |
| 4730 | As an administrator, I want to be able to attach and detach a file to a product, so that my customers have access to additional materials |
| 4731 | As a customer, I can search for products so that it's easier to find what I'm looking for |



| | |
|------|--|
| 4732 | As a customer, I can add a product to my shopping basket so that I can buy stuff |
| 4733 | As a customer, I can view all products in my shopping basket so that I have some idea of how much I'm buying |
| 4734 | As a customer, I can edit my shopping basket so that I only buy what I want |
| 4735 | As a customer, I can rate a product so that I can express my opinion about products I know something about |
| 4736 | As an administrator, I can mark one or more products "deal of the day", which will make it/them show up first on the home page, clearly marked "deal of the day", so that I can promote products to my customers |
| 4737 | As a customer, I can save my address so that I can reuse it for future orders. |
| 4738 | As a user, I can filter search results based on rating (appropriate for the chosen mechanism, thumbs up/down or by number of stars) so that I only see highly rated products |
| 4739 | As a customer, I can place an order so that I can actually get the products |
| 4740 | As a customer, I receive an email notification when the status of an order changes |



Specific requirements

The requirements are organized by the user and in order of importance. The objective is to create a first stage application as soon as possible with the minimum functionality to fulfil the main objective of a webshop: to sell and manage products. All the requirements that are not strictly tied to this objective will be fulfilled in subsequent sprints once the first stage has been reached.

Functional requirements

Customer

- CR1. Register new customer
- CR2. Login
- CR3. Add item to the shopping cart
- CR4. Retrieve the shopping cart
- CR5. Update the shopping cart
- CR6. Checkout
- CR7. Filter items by category
- CR8. Search items
- CR9. Rate items
- CR10. Filter items by rating
- CR11. Save address

Administrator

- AR1. Login
- AR2. Create a category
- AR3. Update a category
- AR4. Delete a category
- AR5. Create an item
- AR6. Update an item
- AR7. Delete an item
- AR8. Add product constrain
- AR9. Retrieve orders
- AR10. Create special offers

System

- SR1. Display items
- SR2. Calculate the shopping cart price

Non-functional requirements

Security

- NFR1. Protect administrator routes from customer accounts
- NFR3. Send e-mail notification when some course information change

Database

- NFR2. A firebase database has to be accessible by the system.



Requirements Risks

1. Performance risks. May create delays in the performance of the system
 - a. Developers are not familiar with the best practices in the selected technologies.
2. Safety and security risks. May affect reaching the requirements of system division between customers and administrator
 - a. Developers were not able to protect administrator routes.
3. Implementation of technology risks. May require using technologies that are not familiar with the developing team.
 - a. Developers are not familiar with the technologies used for the project.
4. Schedule risks. May delay the development of the system.
 - a. Developers are unable to work for personal/health problems.
5. Stability risks
 - a. Requirements may change or disappear during the development

Traceability analysis

The matrices below will be refined in each sprint.

Risks and Requirements

| | Risk 1 | Risk 2 | Risk 3 | Risk 4 | Risk 5 |
|------|--------|--------|--------|--------|--------|
| CR1 | MEDIUM | NONE | MEDIUM | LOW | LOW |
| CR2 | MEDIUM | NONE | MEDIUM | LOW | LOW |
| CR3 | MEDIUM | NONE | MEDIUM | MEDIUM | LOW |
| CR4 | MEDIUM | NONE | MEDIUM | MEDIUM | LOW |
| CR5 | HIGH | NONE | MEDIUM | MEDIUM | LOW |
| CR6 | HIGH | NONE | MEDIUM | MEDIUM | LOW |
| CR7 | HIGH | NONE | MEDIUM | MEDIUM | LOW |
| CR8 | HIGH | NONE | MEDIUM | MEDIUM | LOW |
| CR9 | HIGH | NONE | HIGH | MEDIUM | LOW |
| CR10 | HIGH | NONE | HIGH | MEDIUM | LOW |
| CR11 | HIGH | MEDIUM | HIGH | MEDIUM | LOW |
| AR1 | MEDIUM | MEDIUM | MEDIUM | MEDIUM | LOW |



| | | | | | |
|------|------|--------|--------|--------|-----|
| AR2 | HIGH | MEDIUM | MEDIUM | MEDIUM | LOW |
| AR3 | HIGH | MEDIUM | MEDIUM | MEDIUM | LOW |
| AR4 | HIGH | MEDIUM | MEDIUM | MEDIUM | LOW |
| AR5 | HIGH | MEDIUM | MEDIUM | MEDIUM | LOW |
| AR6 | HIGH | MEDIUM | MEDIUM | MEDIUM | LOW |
| AR7 | HIGH | MEDIUM | MEDIUM | MEDIUM | LOW |
| AR8 | HIGH | MEDIUM | HIGH | MEDIUM | LOW |
| AR9 | HIGH | MEDIUM | MEDIUM | MEDIUM | LOW |
| AR10 | HIGH | MEDIUM | HIGH | MEDIUM | LOW |
| SR1 | HIGH | MEDIUM | MEDIUM | MEDIUM | LOW |
| SR2 | HIGH | MEDIUM | MEDIUM | MEDIUM | LOW |
| NFR1 | HIGH | MEDIUM | MEDIUM | MEDIUM | LOW |
| NFR2 | HIGH | MEDIUM | MEDIUM | MEDIUM | LOW |
| NFR3 | HIGH | MEDIUM | MEDIUM | MEDIUM | LOW |

Requirement dependencies

| | Requirements | Stories |
|------|--------------|---------|
| CR1 | - | 4722 |
| CR2 | CR1 | 4722 |
| CR3 | - | 4732 |
| CR4 | CR3 | 4733 |
| CR5 | CR3 | 4733 |
| CR6 | CR3 | 4739 |
| CR7 | AR2, AR5 | - |
| CR8 | AR5 | 4731 |
| CR9 | AR5 | 4735 |
| CR10 | CR9, AR5 | 4738 |
| CR11 | CR2, CR6 | 4737 |

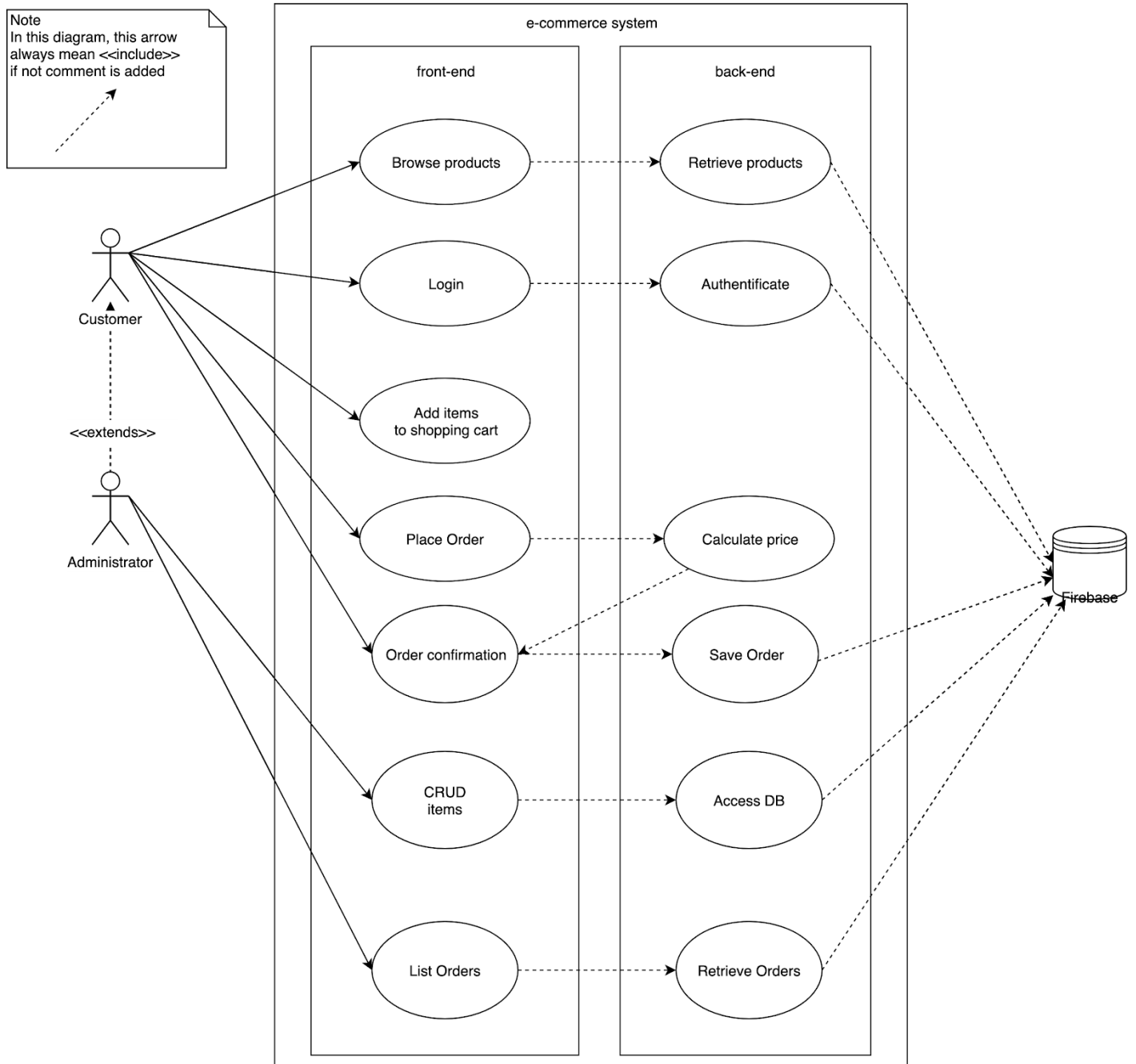


| | | |
|------|---------------------|------|
| AR1 | - | 4724 |
| AR2 | - | 4725 |
| AR3 | AR2 | 4725 |
| AR4 | AR2 | 4725 |
| AR5 | - | 4726 |
| AR6 | AR5 | 4728 |
| AR7 | AR6 | 4729 |
| AR8 | AR5 | 4727 |
| AR9 | CR6 | - |
| AR10 | AR5 | 4736 |
| SR1 | NFR2, AR5 | - |
| SR2 | CR3 | - |
| NFR1 | CR1, CR2, AR1, NFR2 | - |
| NFR2 | - | - |
| NFR3 | CR1,CR6 | 4740 |



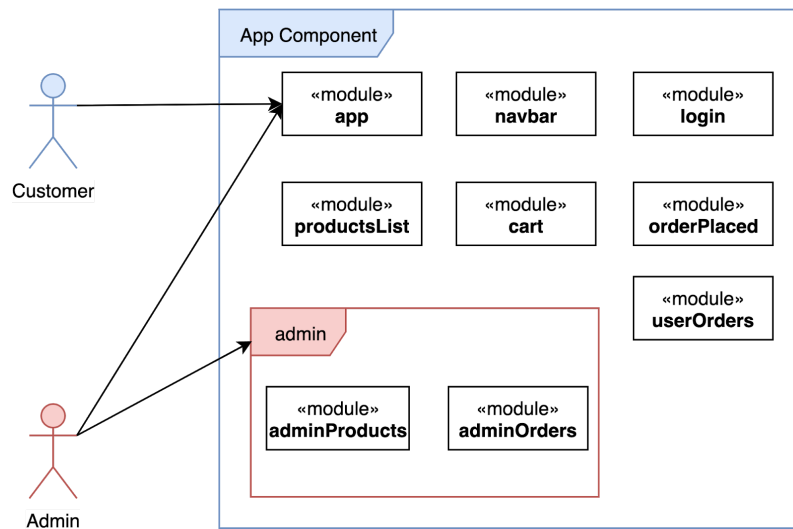
UML

Use Case Diagram

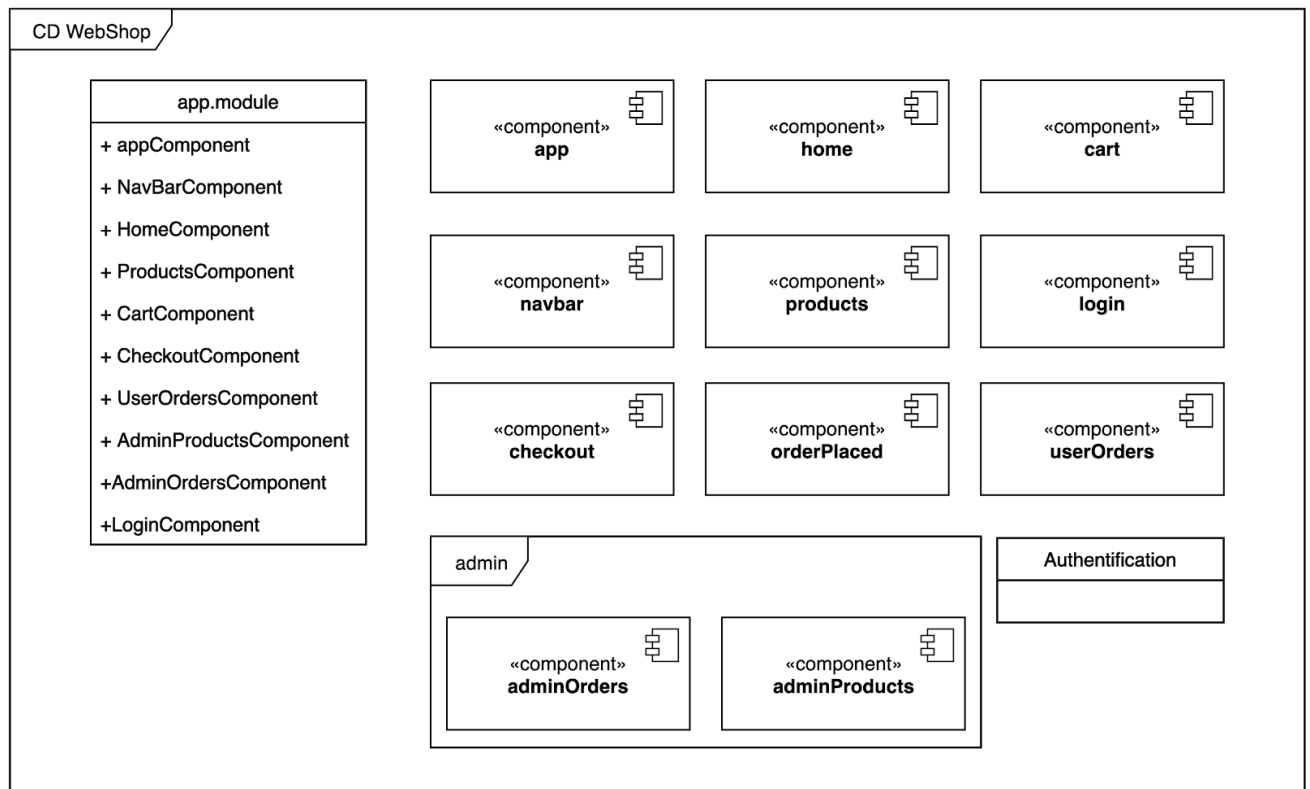




Routing diagram



Components packages diagram





Sprint 1 Components diagram

