

# User Requirements Specification



**Members:**

Airell Rasendriya Bachtiar

Borek Bandell

Ismet Bilir

Tim de Lange

**Tutor:** Stan Hartingsveldt

**Class:** CB-S2-01

**Eindhoven, September 2021**

# Table of Content

<b>Table of Content</b>	<b>2</b>
<b>Agreements with Clients</b>	<b>3</b>
<b>Functional Requirements</b>	<b>3</b>
<b>Use Cases</b>	<b>4</b>
<b>GUI</b>	<b>9</b>
Layout	9
Login	10
Schedule	11
Employee	12
Product	15
Department	18
<b>Website Wireframe</b>	<b>20</b>

# 1. Agreements with Clients

We are going to work on this project for 6 weeks. We agreed on certain goals or requirements that the product will fulfill, however there are some things that don't fit in this scope that can be improved upon later.

We told the client about the way we want to handle the scheduling. We agreed that this will be done with a screen where the user can drag and drop employee names on the schedule. This can also be seen in the GUI part of this document.

# 2. Functional Requirements

ID	A user should be able to	Target Actor	Use Cases	MoSCoW
UR-01	Login	All users	UC-01	Must
UR-02	View an overview of employees	Manager and Administration	UC-11	Must
UR-03	Sort the overview of employees	Manager and Administration	UC-18	Should
UR-04	Filter the overview of employees	Manager and Administration	UC-02	Could
UR-05	Add new employee	Administration	UC-03	Must
UR-06	Edit or delete an employee	Administration	UC-04	Must
UR-07	View an overview of products	All users	UC-12	Must
UR-08	Sort the overview of products	All users	UC-18	Should
UR-09	Filter the overview of products	All users	UC-06	Could
UR-10	Add a product	Manager and Administration	UC-08	Must
UR-11	Edit a product	Depot Worker	UC-07	Must
UR-12	Delete a product	Manager	UC-09	Must
UR-13	View an overview of departments	Administration	UC-13	Could
UR-14	Add new department	Administration	UC-10	Could
UR-15	Delete/edit departments	Administration	UC-14	Could
UR-16	Filter the overview of departments	Administration	UC-15	Won't

UR-17	Sort the overview of departments	Administration	UC-17	Won't
UR-18	View the schedule	Depot Worker and Sales worker	UC-16	Won't
UR-19	Assign schedule	Manager and Administration	UC-05	Must
UR-20	Automatically assign a schedule	Manager		Could
UR-21	Make a restock request	Admin. , Manager and Sales worker	UC-18	Should
UR-22	View restock requests	All users	UC-19	Should
UR-23	Handle restock request	Depot worker	UC-20	Should

### 3. Use Cases

Different actors:

- Manager: Is able to do almost everything except for managing departments and adding/editing/removing employees (manager can see an overview of employees).
- Administrator: Is able to do everything.
- Depot Worker: Almost the same as a sales person but can also change stock details.
- Sales Person: Almost the same as a depot worker but can not change stock details and only view stock details.
- User

With the actor user we mean every user qualified to use the system.

**ID:** UC-01

**Use case:** Login

**User requirement:** UR-01

**Actor:** User

**Prerequisite:** None

**Main success scenario:**

1. User opens the system and sees the login screen.
2. System asks for username and password
3. User fills in credentials and confirms login
4. System logs in and shows the main screen

**Extensions:**

3a. User doesn't enter or enters wrong credentials

1. System shows an error message
2. Password field gets reset and Manager has to enter credentials again

3. Return to step 2

**ID:** UC-02

**Use case:** Filter the overview of employees

**User requirement:** UR-04

**Actor:** Manager and Administration

**Prerequisite:** UC-11

**Main success scenario:**

1. User chooses a filter and confirms
2. System shows list of employees based on the filter

**ID:** UC-03

**Use case:** Add a new employee

**User requirement:** UR-05

**Actor:** Administration

**Prerequisite:** UC-11

**Main success scenario:**

1. Administration chooses add employee option
2. Administration fills all credentials of an employee
3. Administration confirms to add employee

**Extensions:**

- 2a. Administration doesn't fill all credentials of an employee
  1. System shows an error message
  2. Return to step 2

**ID:** UC-04

**Use case:** Edit or delete an employee

**User requirement:** UR-06

**Actor:** Administration

**Prerequisite:** UR-11

**Main success scenario:**

1. Administration sees list of all employees
2. Administration chooses an employee to edit
3. Administration changes changeable credentials
4. Administration saves the changes

**Extensions:**

- 3a. Administration leaves important credentials empty
  1. System shows an error message
  2. Administration fills all empty important credentials
  3. Return to step 4
- 3b. Administration chooses to remove an employee
  1. System removes chosen employee
  2. End of use case

**ID:** UC-05

**Use case:** Assign schedule

**User requirement:** UR-09

**Actor:** Manager and Administration

**Prerequisite:** The software is opened on the main screen

**Main success scenario:**

1. User opens schedule assigning screen
2. User chooses employee to work on certain day and moment
3. System assigns employee to the day and moment

**Extensions:**

- 3a. Employee has already been assigned on this moment/day
  1. System shows error message to user
  2. Return to step 2
- 3b. Employee daily or weekly hour has been fulfilled
  1. System shows error message to user
  2. Return to step 2

**ID:** UC-06

**Use case:** Filter the overview of products

**User requirement:** UR-09

**Actor:** All User

**Prerequisite:** UC-12

**Main success scenario:**

1. User chooses a filter and confirms
2. System shows list of products based on the filter

**ID:** UC-07

**Use case:** Edit a product

**User requirement:** UR-11

**Actor:** Depot Worker

**Prerequisite:** UR-12

**Main success scenario:**

1. Depot worker sees list of all products
2. Depot worker chooses a product to edit
3. Depot worker changes changeable information
4. Depot worker saves the changes

**Extensions:**

- 3a. Depot worker leaves important information empty
  1. System shows an error message
  2. Depot worker fills all empty important information
  3. Return to step 4
- 3b. Depot worker chooses to remove a product
  1. System removes chosen product
  2. End of use case

**ID:** UC-08

**Use case:** Add a product

**User requirement:** UR-10

**Actor:** Manager and Administration

**Prerequisite:** UR-12

**Main success scenario:**

1. Manager chooses add product option
2. Manager fills all information of a product
3. Manager confirms to add product

**Extensions:**

- 2a. User doesn't fill all information of a product
  1. System shows an error message
  2. Return to step 2

**ID:** UC-09

**Use case:** Delete a product

**User requirement:** UR-12

**Actor:** Manager

**Prerequisite:** UR-12

**Main success scenario:**

1. Manager sees list of all products
2. Manager chooses a product to edit
3. Manager chooses to remove a product
4. System removes chosen product

**ID:** UC-10

**Use case:** Add new department

**User requirement:** UR-14

**Actor:** Administration

**Prerequisite:** UR-13

**Main success scenario:**

1. Administration chooses to add new department
2. Administration fills the credentials
3. Administration confirms to add new department

**Extensions:**

- 2a. Administration doesn't fill the credentials
  1. System shows an error message
  2. Return to step 2

**ID:** UC-11

**Use case:** View an overview of employees

**User requirement:** UR-02

**Actor:** Administrator and manager

**Prerequisite:** User should be logged in as an administrator or manager.

**Main success scenario:**

1. User clicks on the employee overview screen
2. Systems shows an unordered list of employees.

**Extensions:**

- 2a. User wants to sort the list
  1. Go to UC-02

**ID:** UC-12

**Use case:** View an overview of products

**User requirement:** UR-07

**Actor:** Administrator and manager

**Prerequisite:** User should be logged in as an administrator or manager.

**Main success scenario:**

1. User clicks on the products overview screen.
2. Systems shows an unordered list of products.

**Extensions:**

- 2a. User wants to filter the list
  1. Go to UC-06

**ID:** UC-13

**Use case:** View an overview of departments

**User requirement:** UR-13

**Actor:** Administrator

**Prerequisite:** User should be logged in as an administrator.

**Main success scenario:**

1. User clicks on the department overview screen.
2. Systems shows an unordered list of departments.

**Extensions:**

- 2a. User wants to filter the list
  1. Go to UC-06

**ID:** UC-14

**Use case:** Delete/edit departments

**User requirement:** UR-15

**Actor:** Administrator

**Prerequisite:** UC-13

**Main success scenario:**

1. User selects a department
2. User clicks edit or delete button

**Extensions:**

- 2a. User clicks delete
  1. Department gets deleted from the database
  2. The list of departments refreshes



2b.

1. The add department screen opens with the current known credentials already filled in.
2. User changes credentials and clicks confirm.

**ID:** UC-15

**Use case:** Filter the overview of departments

**User requirement:** UR-16

**Actor:** Administrator

**Prerequisite:** UC-13

**Main success scenario:**

1. User chooses a filter and confirms
2. System shows a ordered/filtered list of the departments

**Extensions:**

None

**ID:** UC-16

**Use case:** View the schedule

**User requirement:** UR-18

**Actor:** Depot worker

**Prerequisite:** Depot worker is logged into the website

**Main success scenario:**

1. User clicks the show my schedule button.
2. The website displays the schedule of the depot worker.

**ID:** UC-17

**Use case:** Sort the overview of employees

**User requirement:** UR-03

**Actor:** Manager and Administration

**Prerequisite:** UC-11

**Main success scenario:**

1. User chooses something to sort on and confirms.
2. System shows a list of employees sorted by the selected thing to sort.

**ID:** UC-18

**Use case:** Sort the overview of products

**User requirement:** UR-08

**Actor:** All users

**Prerequisite:** UC-12

**Main success scenario:**

1. User chooses something to sort on and confirms.
2. System shows a list of employees sorted by the selected thing to sort.

**ID:** UC-17

**Use case:** Sort the overview of departments

**User requirement:** UR-17

**Actor:** Administrator

**Prerequisite:** UC-13

**Main success scenario:**

1. User chooses something to sort on and confirms.
2. System shows a list of departments sorted by the selected thing to sort.

**ID:** UC-18

**Use case:** Make a restock request

**User requirement:** UR-21

**Actor:** Admin. , Manager and Sales worker

**Prerequisite:** User should be in the restock requests overview, UC-18

**Main success scenario:**

1. User clicks the new restock request button.
2. User adds the details for the restock request.
3. User confirms.

**Extensions:**

2a. User filled out wrong details. The system shows an error message and clears the wrong fields.

**ID:** UC-19

**Use case:** View an overview of all restock requests

**User requirement:** UR-22

**Actor:** All users

**Prerequisite:** User is logged in and clicks the view restock requests button.

**Main success scenario:**

1. A list of all restock requests are shown to the user.

**Extensions:**

None

**ID:** UC-20

**Use case:** Handle a restock request

**User requirement:** UR-23

**Actor:** Depot workers

**Prerequisite:** User should be in the restock requests overview, UC-18

**Main success scenario:**

1. User selects a restock request and clicks on handle.
2. User selects the right stock for the product in the depot and store.
3. User confirms.
4. The system sets the state of the restock request to handled.

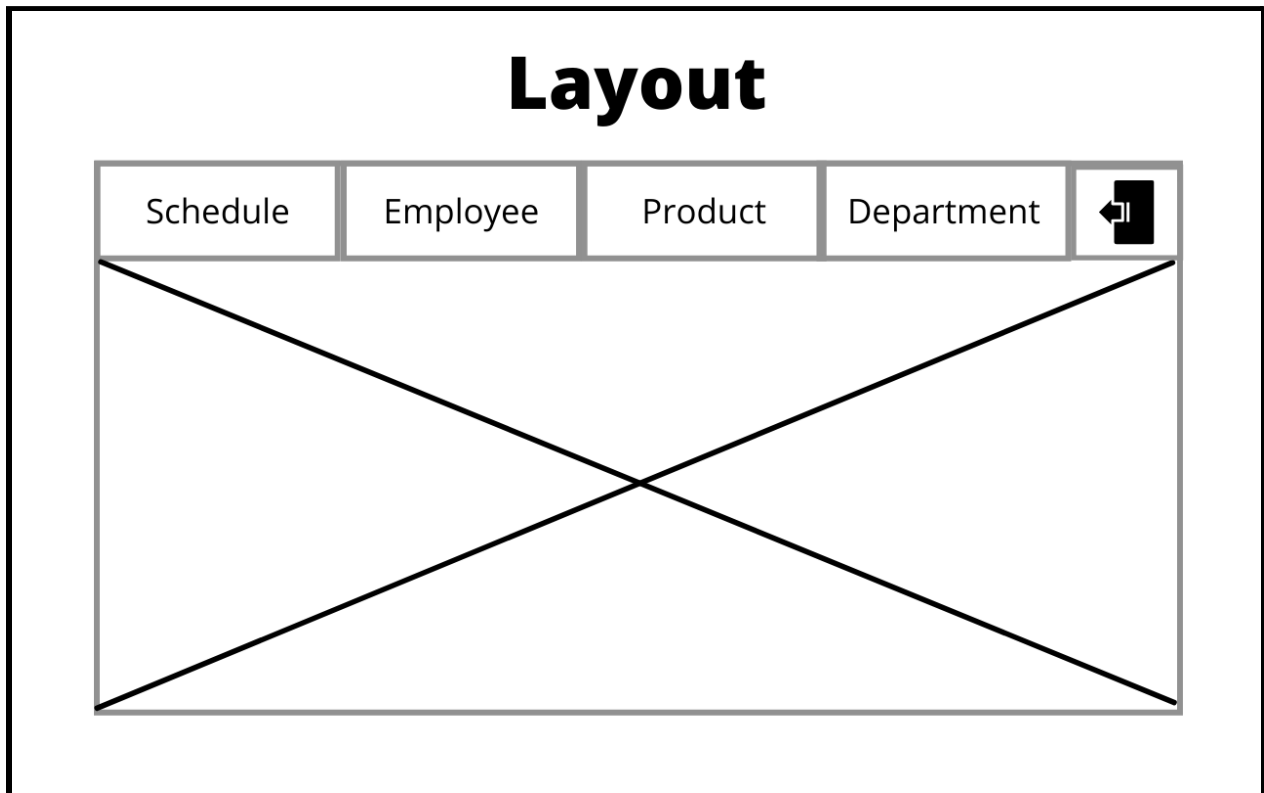
**Extensions:**

None

## 4. GUI

### 4.1. Layout

The wireframe below is the layout of the application. On top there are 4 different options you can choose between schedule, employee, product and department. The content will be shown in the 'X' area. The logo on the top right is the logout button.



Enables disabled

## 4.2. Login

# Login

Username:

Password:

Login

- Before entering the program, this page will be shown first
- User can enter their username and password
- User then can press the login button to proceed

### 4.3. Schedule

## Schedule

Monday

Tuesday

Wednesday

Thursday

Friday

<- Monday - 06/09/21 ->

Morning	Afternoon	Evening

Names

- User can choose a day to assign shift for that day
- User can drag the name of their employees to assign to their 3 different shifts
- Green and red colored boxes indicate the availability of that employee
- Note: After assigning the employees for this day at the top there is a percentage that tells the user how many employees can still be assigned for the rest of the week.
- Box on the right shows the status of each day. Green means available and red means full.

## 4.4. Employee

### Employee Overview

Filter: ▼

Sort: ▼

Name:	Role:	Contract Type:	Phone Number:

Fire Employee

Add Employee

- User can choose a filter to see more specific kind of employees such as: roles, currently working, currently employed
- User can sort the list by: name, role, contract type
- More information of the employee can be found by double clicking the user
- Add button is to add new employee
- Fire button to terminate the contract in the system
- If the user clicks on a user once, then small bits of information are shown on the right side.

# Employee Register

First name:	<input type="text"/>	BSN:	<input type="text"/>
Surname:	<input type="text"/>	Function:	<input type="text"/>
Username:	<input type="text"/>	End Date:	<input type="text"/>
Password:	<input type="text"/>	Contract Type:	<input type="text"/>
Phone Number:	<input type="text"/>	Hourly Wage:	<input type="text"/>
Email:	<input type="text"/>	Note:	<input type="text"/>

**Cancel**

**Add**

- This GUI is the design of a form that lets you add an employee
- User must insert the important detail(all except Note) before confirms it to add an employee
- User can cancel the registration and it will bring you back to the overview

# Employee Info

Edit

First name:	<input type="text"/>	BSN:	<input type="text"/>
Surname:	<input type="text"/>	Function:	<input type="text"/>
Username:	<input type="text"/>	End Date:	<input type="text"/>
Password:	<input type="text"/>	Contract Type:	<input type="text"/>
Phone Number:	<input type="text"/>	Hourly Wage:	<input type="text"/>
Email:	<input type="text"/>	Note:	<input type="text"/>

Back

- This page is shown if user pressed the more button in employee overview
- It will shows user all the information of the chosen employee
- User can edit employee by pressing the edit button



# Edit Employee

**Remove**

First name:	<input type="text"/>	BSN:	<input type="text"/>
Surname:	<input type="text"/>	Function:	<input type="text"/>
Username:	<input type="text"/>	End Date:	<input type="text"/>
Password:	<input type="text"/>	Contract Type:	<input type="text"/>
Phone Number:	<input type="text"/>	Hourly Wage:	<input type="text"/>
Email:	<input type="text"/>	Note:	<input type="text"/>

**Back**

**Edit**

- This page is shown if user press edit button from employee info
- User can edit all the information of the selected employee
- Back button is to cancel
- User can remove/fire employee by pressing remove button

## 4.5. Product

### Product Overview

Filter: ▼

Sort: ▼

Name:	Product code	Store Stock:	Depot Stock:

Remove Product

Add Product

- Users can choose a filter to see only that category of products. For example: brand, type,
- Double clicking on a button will go to a page that contains more information about that product
- Add button is to add a product
- Remove a product from the list
- On the right there is small information that you can see if you press a product once to select it

### Product Info - New Product

Product Name: <input type="text"/>	Depot stock: <input type="text"/>
Brand: <input type="text"/>	Store stock: <input type="text"/>
Dimension: <input type="text"/>	Depot Location: <input type="text"/>
Length: <input type="text"/>	Department: <input type="text"/>
Height: <input type="text"/>	Description: <div style="border: 1px solid black; height: 80px; width: 100%;"></div>
Width: <input type="text"/>	
Selling Price: <input type="text"/>	
Price without VAT: <input type="text"/>	

- This GUI is the design of a form that lets you add a product
- User must insert the important detail(all except Description) before confirms it to add a product
- User can cancel the registration by pressing cancel button

### Product Info

Product Name: <input type="text"/>	Depot stock: <input type="text"/>
Brand: <input type="text"/>	Store stock: <input type="text"/>
Dimension: <input type="text"/>	Depot Location: <input type="text"/>
Length: <input type="text"/>	Department: <input type="text"/>
Height: <input type="text"/>	Description: <div style="border: 1px solid black; height: 80px; width: 100%; background-color: black;"></div>
Width: <input type="text"/>	
Selling Price: <input type="text"/>	
Price without VAT: <input type="text"/>	

- This page is shown if user pressed the more button in product overview
- It will shows user all the information of the chosen product
- User can edit product by pressing the edit button

### Product Info - Editing

Product Name:	<input type="text"/>	Depot stock:	<input type="text"/>
Brand:	<input type="text"/>	Store stock:	<input type="text"/>
Dimension:	<input type="text"/>	Depot Location:	<input type="text"/>
Length:	<input type="text"/>	Department:	<input type="text"/>
Height:	<input type="text"/>	Description:	<div style="background-color: black; height: 15px; width: 100%;"></div>
Width:	<input type="text"/>		<div style="background-color: black; height: 15px; width: 90%;"></div>
Selling Price:	<input type="text"/>		<div style="background-color: black; height: 15px; width: 95%;"></div>
Price without VAT:	<input type="text"/>		<div style="background-color: black; height: 15px; width: 95%;"></div>

- This page is shown if user press edit button from product info
- User can edit all the information of the selected product
- Back button is to cancel
- User can remove product by pressing remove button

## 4.6. Department

### Department Overview

Filter: ▼

Sort: ▼

Name:	Code:	Number of employees	Manager:

Remove Department

Add Department

- User can choose a filter to filter on the list
- Sort through the list
- Double click to view more information
- Add button is to add a department
- Remove department button
- Right side there is small information that is shown after selecting a department

# Department Register

Name:

Description:

**Cancel**

**Add**

- This GUI is the design of a form that lets you add a department
- User must insert the important detail(all except Description) before confirms it to add a department
- User can cancel the registration by pressing cancel button

# Department Info

Name:

Description:

**Back**

**Edit**

- This page is shown if user pressed the more button in department overview
- It will shows user all the information of the chosen department
- User can edit department by pressing the edit button

# Edit Department

Name:

Description:

**Back**

**Remove**

**Edit**

- This page is shown if user press edit button from department info
- User can edit all the information of the selected department
- Back button is to cancel
- User can remove department by pressing remove button

## 5. Website Wireframe

As of now there is no website wireframe because the website currently does not fit in the scope of the project.