# **URS**



Location:

Members: Kwangjin Lee, Evaldas Drasutis, Emilian Alexandru, Florin Deleanu

Tutor: Stan S. van Hartingsveldt

Date: 03/09/2020

Table of Contents:

1. Title page
2. Agreements with client
3. Functional requirements
   1. What should the system be able to do?
   2. Group them by application/website (if applicable)
   3. Prioritize them using MoSCoW (must, should, could, won’t)
4. Use Cases
5. GUI
   1. Images of the various views.
   2. A brief description per view how it works
6. Website wireframe (when applicable)
   1. The structure of your website with a brief description of its intent.

Agreements with Client:

1. Employee administration is the most important part of the program, followed by stock
2. The following will be the information on employees:

* First name
* Surname
* Username (account)
* Password (account)
* Phone number
* Address
* Email address
* Spouse (and contact details)
* Date of birth
* BSN (Social security number)
* Functions (at least one, possibly more functions the employee can serve as e.g. sales representative)

1. Employees will manage their password as well as information that is subject to change( example: Spouse, address, etc…) They will be able to also see the unchangeable information like BSN.
2. In order to search for employees you can use either their ID or name
3. Application will be build on Windows
4. Employees will check in when arriving and out when leaving. Tracking break time is not needed
5. An employee can have 1 or 2 shifts per day. In the latter case they must be consecutive
6. All employees will use the website but they will only see the specific data they need
7. Website pages should not be bloated and have a clear focus
8. The main statistics tracked are the number of employees, the average number of shifts, the number of hours compared to their contract, and the illness rates. Department wishes the client would like to have the total number of employees and products. About stock, he would like to see the stock levels, avg time between replenishment of actions and stock.

Functional requirement:

1. Administration should be able to add new employees.
2. Administration should be able to remove employees.//I think this should be removed
3. Administration should be able to add employees to the schedule.
4. Administration should be able to update the employees schedule.
5. Administration should be able to dismiss an employee. //not fully save the reason why fired)
6. Administration should be able to assign an employee to a department.
7. Sales representative should be able to add stock.
8. Sales representative should be able to remove stock.// idk how should i name dissabled stock
9. Sales representative should be able to mark employees' attendance.
10. Sales representative should be able to update stock.
11. Sales representative should be able to make requests for stock refills.
12. Sales representative should be able to view employees information.
13. Sales representative should be able to view employees statistics.
14. Sales representative should be able to search for specific employee.
15. Depot should be able to accept stock refill request.
16. User should be able to log in into the application.
17. User should be able to log into the website.(web)
18. User should be able to edit profile information.(web)
19. User should be able to view working shifts.
20. User should be able to view data on stock.
21. User should be able to logout of the application
22. User should be able to logout of the website

Moscow prioritization table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Administration** | | | | |
| **Functional Requirements** | **M** | **S** | **C** | **W** |
| Login | X |  |  |  |
| Create employee account | X |  |  |  |
| Edit Employee account information | X |  |  |  |
| Assign a rank to an employee | X |  |  |  |
| View inactive/removed employee’s information |  |  |  | X |
| Delete employee account |  |  |  |  |
| Search for employee | ? |  |  |  |
| Deactivate active employee account |  |  | X |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Manager** | | | | |
| **Functional Requirements** | **M** | **S** | **C** | **W** |
| Login | X |  |  |  |
| View employee information | X |  |  |  |
| View employee’s attendance | X |  |  |  |
| Assign a rank to an employee | X |  |  |  |
| Add new items to stock |  |  |  |  |
| Delete items from stock |  |  |  |  |
| Search for employees | ? |  |  |  |
| View statistics regarding stock changes | X |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Depo** | | | | |
| **Functional Requirements** | **M** | **S** | **C** | **W** |
| Login | X |  |  |  |
| Check stock | X |  |  |  |
| See incoming shelf stock |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Use Cases:

Potential end users are company’s staff that maintain and view the information in the software and website.

In the application the following actors should be able to:

Actor:Administration()

Actor: Sales representative ()

Actor: Depot()

Use case: View data of stock

Actor: Depot

Main Success Scenario:

1. Employee selects the refresh
2. System returns the most recent data.
3. Data displayed in the application

Use case: Add employee

Actor: Administrator

Main Success Scenario:

1. User goes to a button to add a new employee.
2. System requests for employees information.
3. Administrator fills in employees personal information (name,age,gender…)//should I add in everything? // no;
4. System shows a message box of confirmation to add a new employee.

Extensions:

1. Not all mandatory fields are filled in.
   1. System displays not all fields filled in the message.
   2. End of use case.

Use case: Login to the application or website

Actor: Administration, (Management?), Depot Worker, Sales representative

Trigger:

1. Employee starts the application
2. Employee logs out

Main Success Scenario:

1. Employee fills in credentials and confirms
2. System confirms the credentials exist in the database
3. System redirects the employee to the main page

Extension:

2a: System rejects the credentials

1. System displays an error message informing the user of the wrong provided information.
2. End of use case

Use Case:Adding new products in stock

Actor: Sales representative

Main Success Scenario:

1. Employee goes to the button labeled add stock
2. Employee fills in the new product information, including the amount in stock and confirms
3. System creates a new product, adding it to the list of available products

Extension:

2a: Employee fills in information not matching the systems criteria

1. System displays a wrong information message, informing the employee of the correctness and location of the problem
2. Return to MMS step 2

2b: Employee fills in already existing information

1. System displays an already existent product message
2. The employee is returned to the stock overview page
3. End of use case

Use Case: Making a product unavailable

Actor: Sales representative

Pre-condition: User must modify an existing product

Main Success Scenario:

1. User clicks a radio button in order to make product unavailable
2. User confirms the changes
3. System displays a confirmation message and mark the product as unavailable

Use Case: Modify a product

Actor: Sales representative

Pre-condition: User must select an item from the product list

Main Success Scenario:

1. User goes to the modification tab by pressing a button
2. System displays the current product information
3. User makes changes to the product and confirms
4. System displays a confirmation message and saves the changes

Extension:

4a: Filled in information does not match the system criteria

1. System displays an error message informing the user of the problem
2. Return to MMS step 2

Use Case: Make a stock refill request

Actor: Sales representative

Main Success Scenario:

1. User selects a product from the list and confirms by pressing a button
2. System process the order and displays a confirmation message

Extension:

2a: Stock is already at max capacity

1. System displays an error message and cancels the order
2. End of use case

User Case:

Actor:

Main Success Scenario:

Extension:

GUI:

Website wireframe:













