URS-RASH

# Introduction:

Date: 09/09/2021

Location: MS Teams

Members:

* Rawan Alahabsi
* Stefan Popescu
* Ana Kowalewska
* Hristo Tanchev

Tutor: Georgiana Manolache

# Revision table:

|  |  |  |  |
| --- | --- | --- | --- |
| Version/ Status | Description | Author | Date |
| 0.1 | First draft of the Project Plan, was not completed yet | Stefan Popescu, Rawan Alahabsi, Ana Kowalewska, Hristo Tanchev | 09/09/2021 |

Table of content:

[Introduction: 1](#_Toc82183186)

[Revision table: 2](#_Toc82183187)

[Agreements with client 4](#_Toc82183188)

[Functional requirements C# application 6](#_Toc82183189)

[Functional requirements Website 7](#_Toc82183190)

[Use cases 8](#_Toc82183191)

[GUI C# Application () 22](#_Toc82183192)

[GUI Website 29](#_Toc82183193)

Agreements with client:

- Agreed to schedule meeting via Ms Teams based on the client’s program

The project is going to have 3 phases but the one we are going to focus on is the first one:

- The first phase has to be focused on:

- Delivering an URS document containing a revision table, a table of contents, an agreements with client section, functional requirements, use cases, and also a GUI

- Coming up with a logo and also a team name

- The start of coding for an intermediate version

- Focus on delivering the add/remove (but still keeping his old data)/update employee function and schedule(assign/edit/view) of the employees and also see the statistics of the employees

-The 2nd phase will be divided in multiple Iterations starting with the first one we decided to focus on:

-Creating a website available for the employees working for Mediabazaar

-The website should present the following features: login, see schedule, statistics and the feature to edit information about employees (name, phone number, address)

-In the 2nd iteration we decided to focus on:

-Creating the product management system that presents the following features: add/view/edit/remove products, see stocks, add stock requests, sell products

-Creating the form for the deport workers in which they will see the stock requests and either approve or deny them, moreover they can see the stock request history.

-A 2nd version of the website in which the employee can say when they are not available

-In the 3rd iteration we decided to focus on:

-Developing an automated scheduling system that will take into account different parameters such as: contract type, days off

-Creating the department management system in which the admin can create/view/edit/remove departments and also assign employees to it.

Functional requirements C# application:

* MUST:

C-FR01- Admin must be able to add employees in the system

C-FR02- Admin must be able to remove an employee from the system

C-FR03- Admin must be able to edit information about employees

(contract type, wages, username, password, when you start working, when you stop working, birthday, name, e-mail, gender, home address)

C-FR04- Admin must be able to see a list of all employees

C-FR05- Admin must be able to assign work shifts for the employees

C-FR06- Admin must be able to modify the schedule of the employees

C-FR07- Admin must be able to see the schedule

C-FR08-Admin must be able to add a product

C-FR09-Admin must be able to edit a product

C-FR10-Admin must be able to delete a product

C-FR11-Admin must be able to view product details

C-FR12-Admin must be able to sell a product

C-FR13-Admin must be able to add a stock request

C-FR14-Depot worker must be able to handle a stock request

C-FR15-Depot worker must be able to view stock request history

C-FR16-Admin must be able to add a department

C-FR17-Admin must be able to view a department

C-FR18-Admin must be able to edit a department

C-FR19-Admin must be able to remove a department

C-FR20-Admin must be able to assign an employee to a department

C-FR21-Admin must be able to unassign an employee to a department

C-FR22-Admin must be able to view the employees who are assigned and to which department

Functional requirements Website:

* MUST

W-FR-01- Employee must be able to login with the username and password from the database

W-FR-02- Employee must be able to logout

W-FR-03- Employee must be able to see his working schedule on the site

W-FR-04- Employee must see his profile page with his information

W-FR-05- Employee must be able to edit some specific information about himself.

W-FR-06- Employee must be able to change his password

W-FR-07- Employee must be able to change his username

W-FR-08 - Employee must be able to send a message via the website with a specific datetime when he or she isn’t available

W-FR-09 - Employee must be able to call in sick via the website

Use cases:

**Add employee to the system (C-FR-01):**

-Actor = Admin:

**Requirement:**

The user must be in the “add employee” form.

Main success scenario:

1. Types the contract type, wages, username, password, when you start working, when you stop working, birthday, name, e-mail, gender, home address
2. Presses the “Add” button
3. The user presses the “Yes” button at the appearance of the confirmation pop-up
4. The registration is complete and a success message will pop up

Extensions:

4a. The fields have not been filled with the appropriate information.

.1: The system sends an “error” message and so the fields need to be edited

.2: Return to step 1

Remarks: The action can be canceled at any time by pressing the “X” button from the top right corner of the screen

**Remove employee from the system (C-FR-02):**

-Actor = Admin:

**Requirement:**

The user must be in the “view all employees” form.

Main success scenario:

1. Selects the required employee from an employee list
2. Presses the “Remove” button
3. A pop-up message requiring a confirmation for the action
4. Presses “Yes”
5. The removal is complete and a success message will pop up

Extensions:

4a. At the press of the “No” button of the confirmation pop-up

.1: End of use cases

Remarks: The action can be canceled at any time by pressing the “X” button form the top right corner of the screen

The remove button is unavailable unless an employee has been selected.

**Edit employee information (C-FR-03):**

-Actor = Admin:

**Requirement:**

The user must be in the “view all employees” form.

Main success scenario:

1. Selects the employee from an employee list
2. Modifies the desired fields
3. Presses the “Edit” button
4. The edit is complete and a success message will pop up

Extensions:

2a. The fields have not been filled with the appropriate information.

.1: The system sends an “error” message and so the fields need to be edited

.2: Return to step 2.

Remarks: The action can be canceled at any time by pressing the “X” button form the top right corner of the screen

The edit button is unavailable unless an employee has been selected.

**Add work shift (C-FR-05):**

-Actor = Admin:

**Requirement:**

A date must already be selected.

Main success scenario:

* + - 1. Select a desired working-shift which will open a tab containing all the available employees
      2. Select an employee from a list of available employees
      3. Press “Assign” button
      4. A message asking the user to confirm the action will pop up
      5. The “Confirm” button is pressed

6. The assignment is complete and a success message will pop up

Extensions:

6a: At the press of the “No” button of the confirmation pop-up

.1: End of use case

Remarks: The action can be canceled at any time by pressing the “X” button from the top right corner of the screen

The user cannot assign an unavailable employee (already works on that day/ he has reached the maximum amount of work for the week)

**Remove schedule(C-FR-06):**

-Actor = Admin:

Main success scenario:

1. Select a date from the calendar
2. Select the desired employee
3. Press the “Remove” button
4. A message asking the user to confirm the action will pop up
5. The removal is complete and a success message will pop up

Extensions:

1a: The user selects a date which has no employee assigned to

.1: Return to step 1

4a: At the press of the “No” button of the confirmation pop-up

.1: End of use case

Remarks: The action can be canceled at any time by pressing the “X” button from the top right corner of the screen.

The user cannot press the “Remove” button unless he chose an employee.

**View schedule(C-FR-07):**

-Actor = Admin:

Main success scenario:

1. Selects a date from the calendar
2. A new window appears showing all the working employees and their working shift

**Add product to the system (C-FR-08):**

-Actor = Admin:

**Requirement:**

The user must be in the “product management” form.

Main success scenario:

1. Types the necessary data
2. Presses the “Add product” button
3. The registration is complete and a success message will pop up

Extensions:

4a. The fields have not been filled with the appropriate information.

.1: The system sends an “error” message and so the fields need to be edited

.2: Return to step 1

**Add product to the system (C-FR-09):**

-Actor = Admin:

**Requirement:**

The user must be in the “edit product” form.

Main success scenario:

1. The admin loads the old data in and makes the changes he desires
2. Presses the “Edit” button
3. The edit is complete, and a success message will pop up

Extensions:

4a. The fields have not been filled with the appropriate information.

.1: The system sends an “error” message and so the fields need to be edited

.2: Return to step 1

**Remove product to the system (C-FR-10):**

-Actor = Admin:

**Requirement:**

The user must be in the “product management” form.

Main success scenario:

1. Admin selects the id of the product he wants to remove
2. Presses the “Delete product” button
3. The removal is complete, and a success message will pop up

**View product stocks (C-FR-11):**

-Actor = Admin:

**Requirement:**

The user must be in the “product management” form.

Main success scenario:

1. Admin clicks on the update view
2. The system displays the stock with a color scheme

**Sell products (C-FR-12):**

-Actor = Admin:

**Requirement:**

The user must be in the “product management” form.

Main success scenario:

1. Admin selects the name of the desired product he wants to sell and also the quantity he wants to sell
2. Presses the “Sell product” button
3. The selling is complete, and a success message will pop up

Extensions:

1a Admin puts a bigger quantity than it is available.

.1: The system sends an “error” message and so the quantity need to be edited

.2: Return to step 1

**Add product stock (C-FR-13):**

-Actor = Admin:

**Requirement:**

The user must be in the “product management” form.

Main success scenario:

1. Admin selects the name of the product he wants to restock and also the quantity he wants
2. Presses the “Add stock request” button
3. A confirmation message will show.

**Accept/Deny stock request (C-FR-14):**

-Actor = Depot worker:

**Requirement:**

The user must be logged in as a depot worker form.

Main success scenario:

1. Depo worker selects the stock request he wants to accept/deny
2. Presses the desired button
3. The stock is updated is complete, and a success message will pop up

**View stock history (C-FR-15):**

-Actor = Depot worker:

**Requirement:**

The user must be logged in as a depot worker form.

Main success scenario:

1. Depo worker presses the view history button

**Add department (C-FR-16):**

-Actor = Admin:

**Requirement:**

The user must be logged in as admin in the “Add department” form.

Main success scenario:

1. Admin fills in the name of the department
2. Presses the “Add department” button
3. The department will be added
4. A success massage will pop up

Extensions:

2a. The fields have not been filled with the appropriate information.

.1: The system sends an “error” message and so the fields need to be edited

.2 Return to step 1

**View department (C-FR-17):**

-Actor = Admin:

**Requirement:**

The user must be logged in as admin in the “Add department” form.

Main success scenario:

1. Admin presses the “view department” button
2. The system loads the departments

**Edit department (C-FR-18):**

-Actor = Admin:

**Requirement:**

The user must be logged in as admin in the “Add department” form.

Main success scenario:

1. Admin selects the department he wants to modify
2. Admin changes the name of the department
3. Admin presses the “Edit” button
4. A success massage will pop up

Extensions:

2a. The fields have not been filled with the appropriate information.

.1: The system sends an “error” message and so the fields need to be edited

.2 Return to step 1

**Remove department (C-FR-19):**

-Actor = Admin:

**Requirement:**

The user must be logged in as admin in the “Add department” form.

Main success scenario:

1. Admin selects the department he wants to remove
2. Presses the “Delete department” button
3. The department will be deleted
4. A success massage will pop up

Extensions:

2a. Admin didn’t select a department

.1: The system sends an “error” message

.2 Return to step 1

**Assign an employee to a department (C-FR-20):**

-Actor = Admin:

**Requirement:**

The user must be logged in as admin in the “Assign to department” form.

Main success scenario:

1. Admin selects the desired employee and the department he wants to assign to
2. Presses the “Assign” button
3. The assignment will be added
4. A success massage will pop up

Extensions:

2a. The fields have not been filled with the appropriate information.

.1: The system sends an “error” message

.2 Return to step 1

**Unassign an employee to a department (C-FR-21):**

-Actor = Admin:

**Requirement:**

The user must be logged in as admin in the “Assign to department” form.

Main success scenario:

1. Admin selects the desired employee and the department he wants to unassign
2. Presses the “Unassign” button
3. The action will be complete
4. A success massage will pop up

Extensions:

2a. The fields have not been filled with the appropriate information.

.1: The system sends an “error” message

.2 Return to step 1

**View the assignment of an employee to a department (C-FR-22):**

-Actor = Admin:

**Requirement:**

The user must be logged in as admin in the “Assign to department” form.

Main success scenario:

1. Admin clicks the “View relations” button
2. The system loads the relations

**Login to the site (W-FR-01)**

Actor: Employee

Main success scenario:

1. The Employee inputs the credentials (username, password) and presses the login button.
2. The System checks the username and password
3. The Employee is redirected to his profile page

Extensions:

2a: The username/password are incorrect

1. Invalid credentials message pops up
2. Return to step 1

**Logout (W-FR-02)**

Actor: Employee

Main success scenario:

Precondition: Employee is logged in

1. Employee clicks on the logout button
2. The system disconnects him
3. Employee is redirected to login page

**View employee schedule (W-FR-03)**

Actor: Employee

Main success scenario:

Precondition: Employee is logged in.

1.Employee selects Schedule in the left sidebar.

2.The System redirects the user to a new page

3. The System loads the user’s schedule

**View the profile page of the employee (W-FR-04)**

Actor: Employee

Main success scenario:

Precondition: Employee is logged in.

1.Employee clicks on the profile page in the left menu of the page

2.The System redirects the user to a new page

3. The System loads the user’s information

**Change password (W-FR-06)**

Actor: Employee

Main success scenario:

Precondition: Employee is logged in.

1. Employee clicks on the profile page in the left menu of the page
2. Employee clicks on the change password button
3. Employee inputs his old password, the new password and the confirmation of the new password
4. Employee clicks on the modify button
5. “Your password was updated” message pops up

Extension:

4a The old password is incorrect

1. Old password incorrect message pops up
2. Return to step 3

4b Confirmation password and password are different

1. Your passwords don’t match message pops up
2. Return to step 3

**Change username (W-FR-07)**

Actor: Employee

Main success scenario:

Precondition: Employee is logged in.

1.Employee clicks on the profile page in the left menu of the page

2.Employee clicks on the change username button

3.Employee inputs his old username and the new username

4.Employee clicks on the modify button

Extension:

4a The old username is incorrect

1. “Old username is incorrect” message pops up
2. Return to step 3

**Employee unavailablity (W-FR-08)**

Actor: Employee

Main success scenario:

Precondition: Employee is logged in.

1. Employee clicks on the unavailability page in the left menu of the page
2. Employee chooses a date and daypart for when he or she is unavailable with a specific reason
3. Employee clicks send button
4. System stocks the unavailability

Extension:

4a 1. The wrong date has been picked

4a2.Return to step 2

**Employee’s sickness (W-FR-09)**

Actor: Employee

Main success scenario:

Precondition: Employee is logged in.

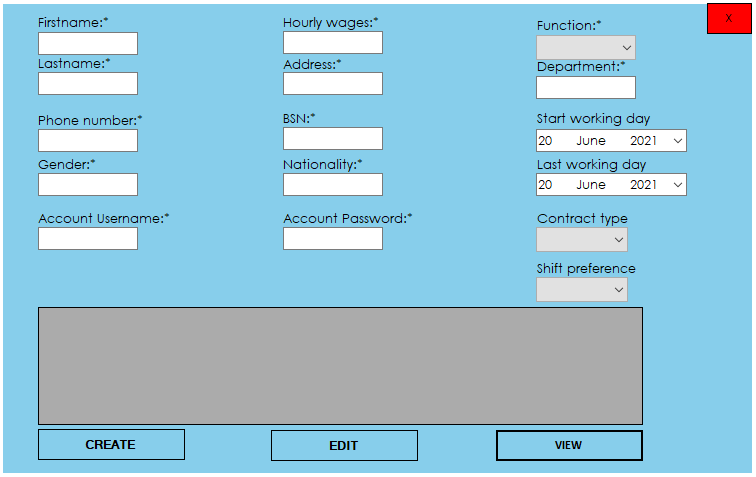
1. Employee clicks on the sickness page in the left menu of the page
2. Employee chooses a date and daypart for when he or she is unavailable with symptoms
3. Employee clicks send button
4. System stocks the request

Extension:

4a1.The wrong date has been picked

4a2.Return to step 2

GUI C# Application ()



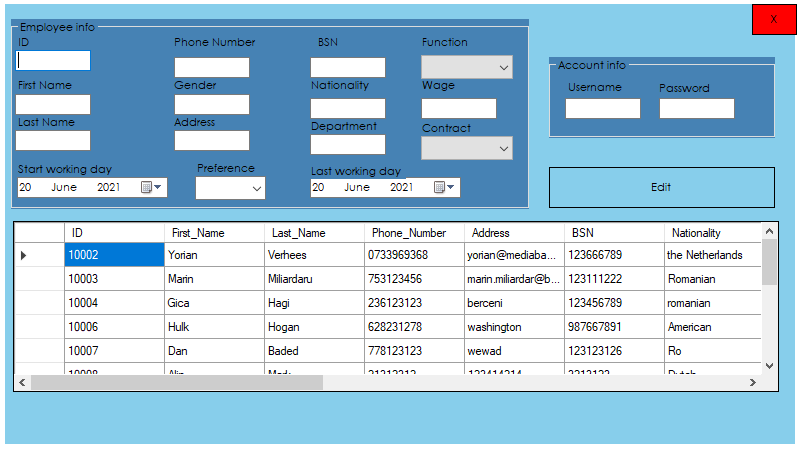
This is the add employee form in which the admin can register an employee in the system

\*The contract type is split in 3:

- 0 hour contract

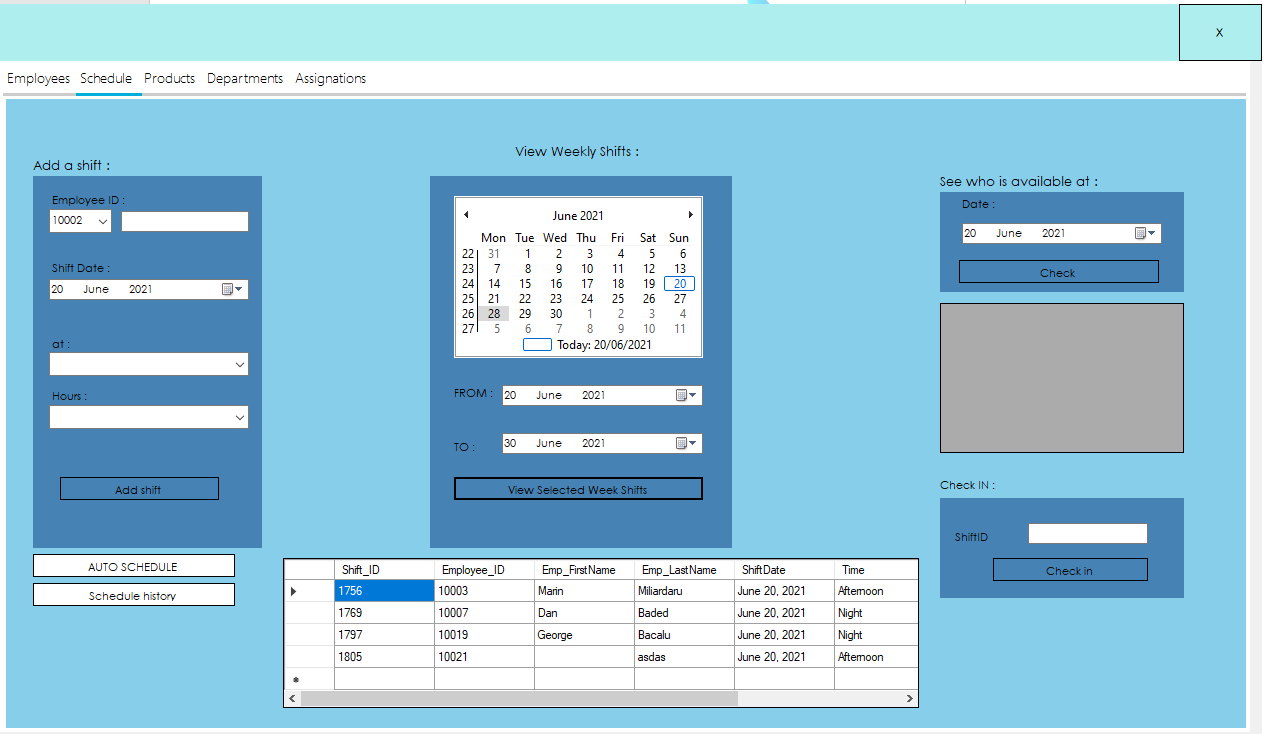
- Part time: 32 hours of work/week

- Full time: 40 hours of work/week

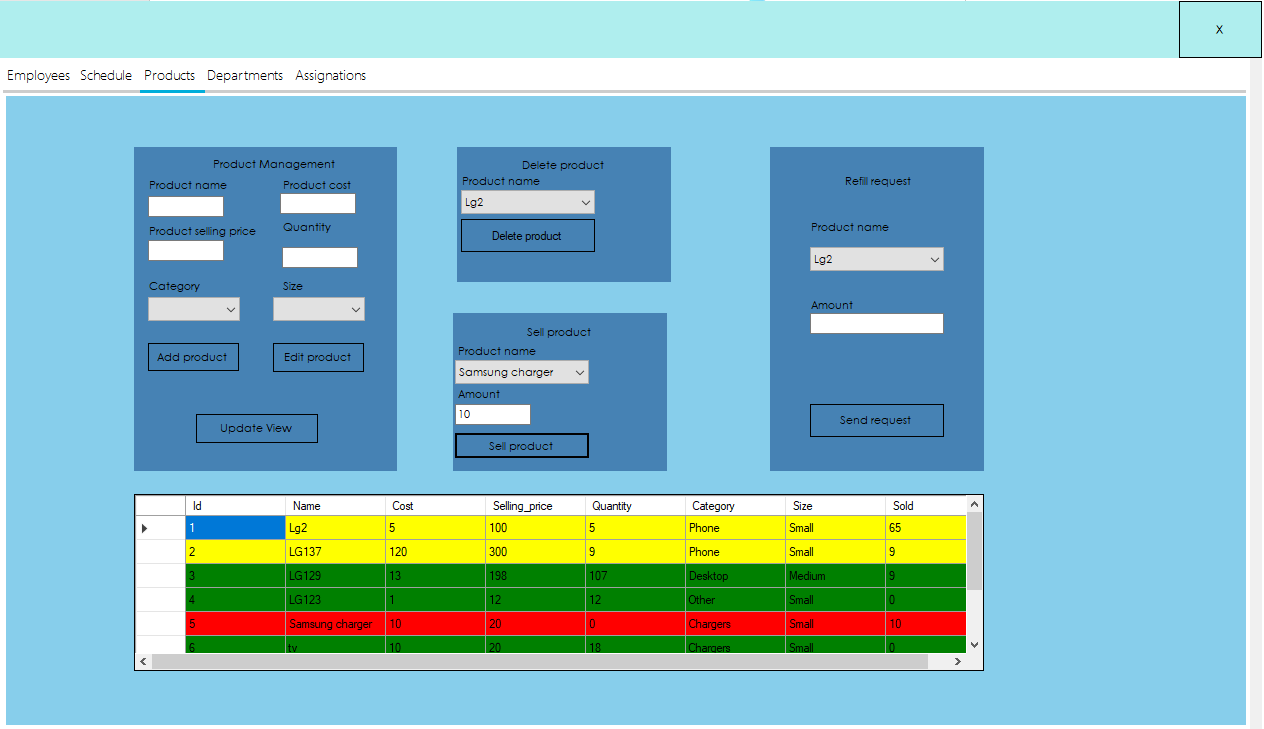


This is the tab where the admin can edit the information about a specific employee.

Once you click on one employee all their data will be loaded

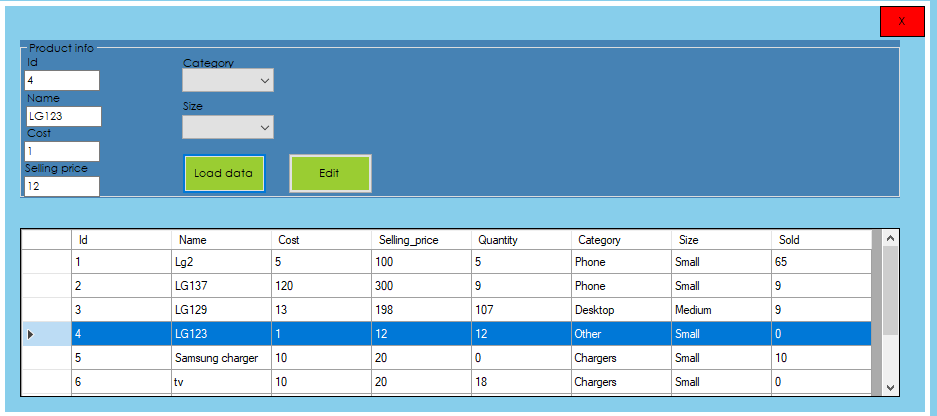


The schedule tab allows the admin to add shifts and check in the employees to work, check for employee availability on a specific date, auto schedule every employee and also he can view the shifts

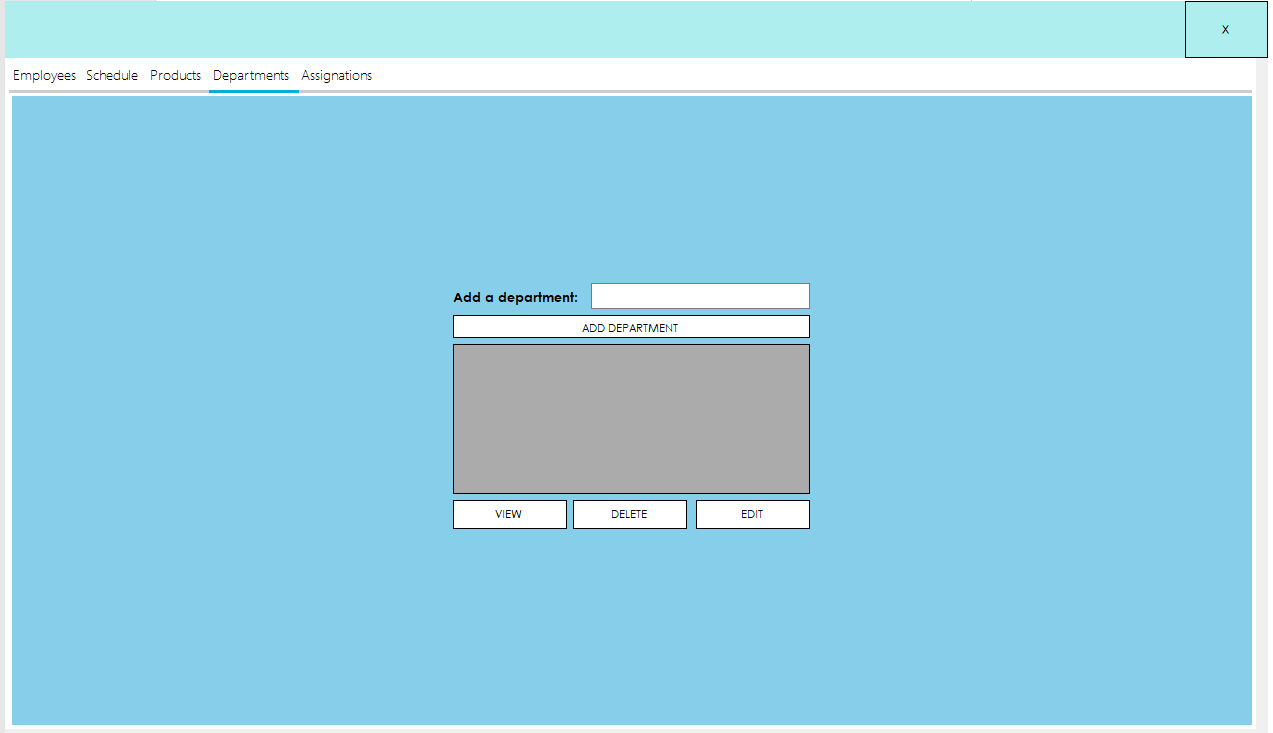


This is the product manager tab where the admin can CRUD products and also he can sell products and request a refill of products

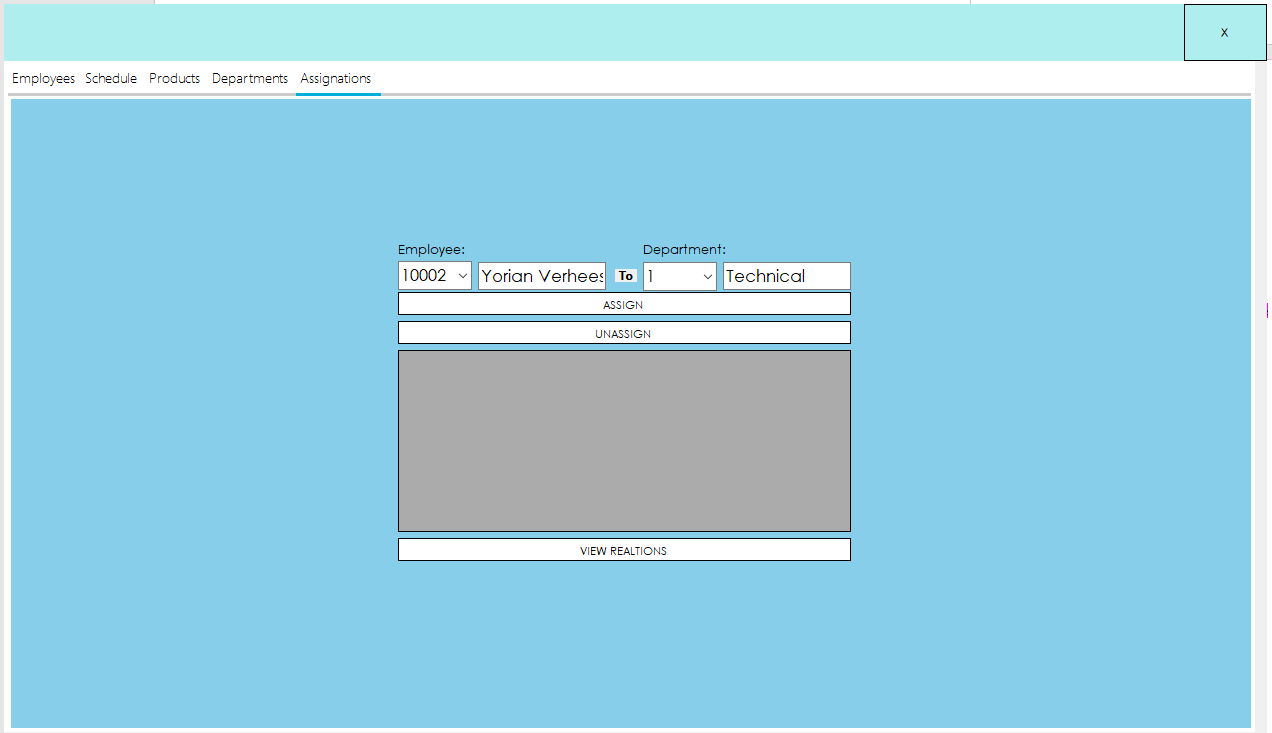
There is also a color scheme(green for 10+, yellow for 5 to 10 products and red for 0 to 5 products)



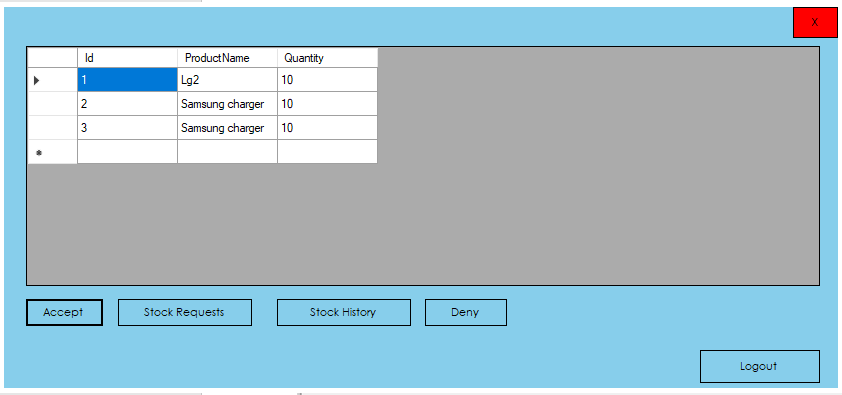
This is the edit product tab where you can select a product press load data and then you edit the desired information



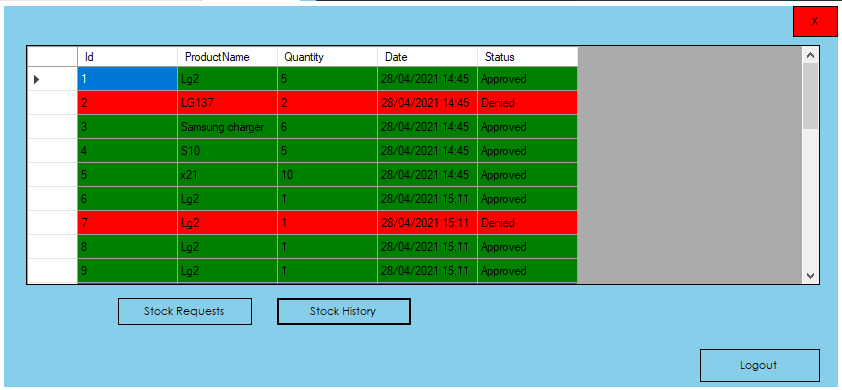
This is the department manager tab where the admin can CRUD departments



This is the assignation tab where the admin can assign employees to specific departments



This is the depot worker view where he can see the stock requests and he has the decision to either accept or deny them



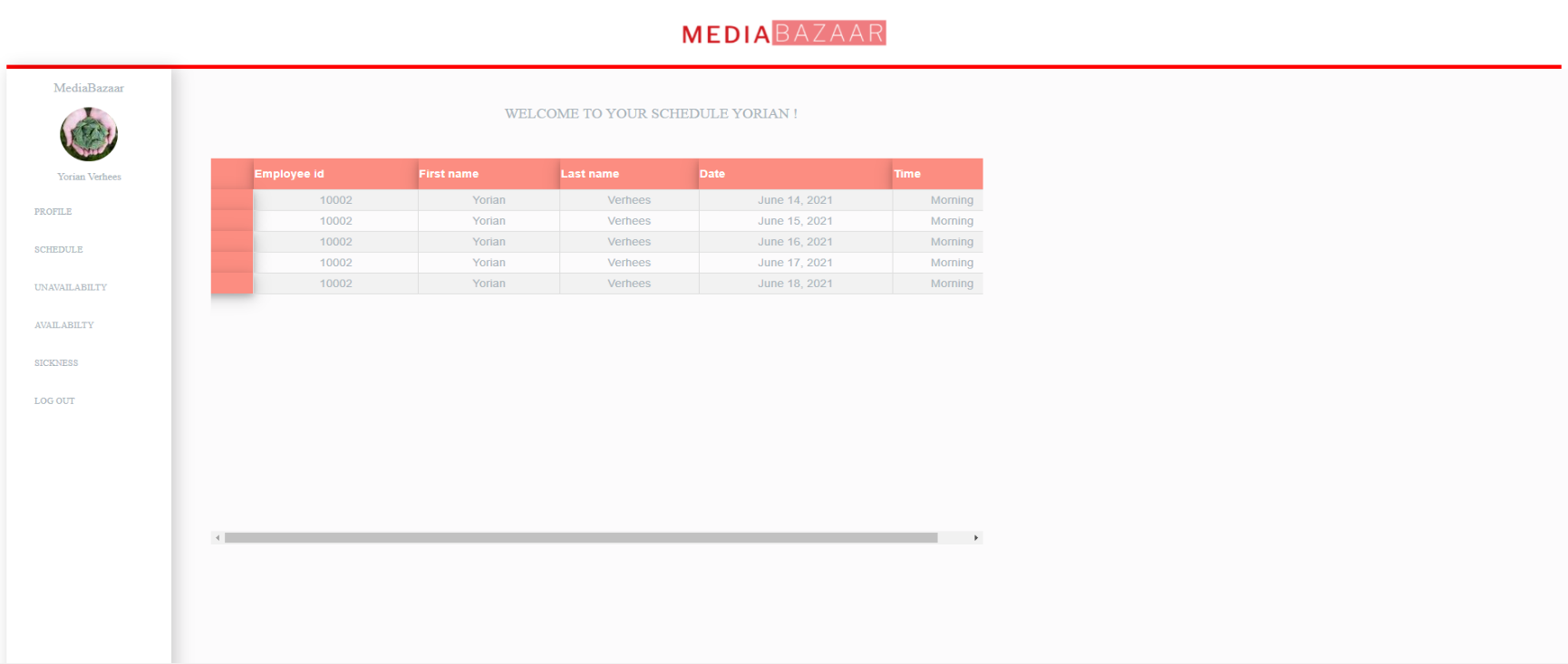
This is the stock request history where the depot worker can see the history of requests the approved ones are with green while the denied are marked with red

# GUI Website

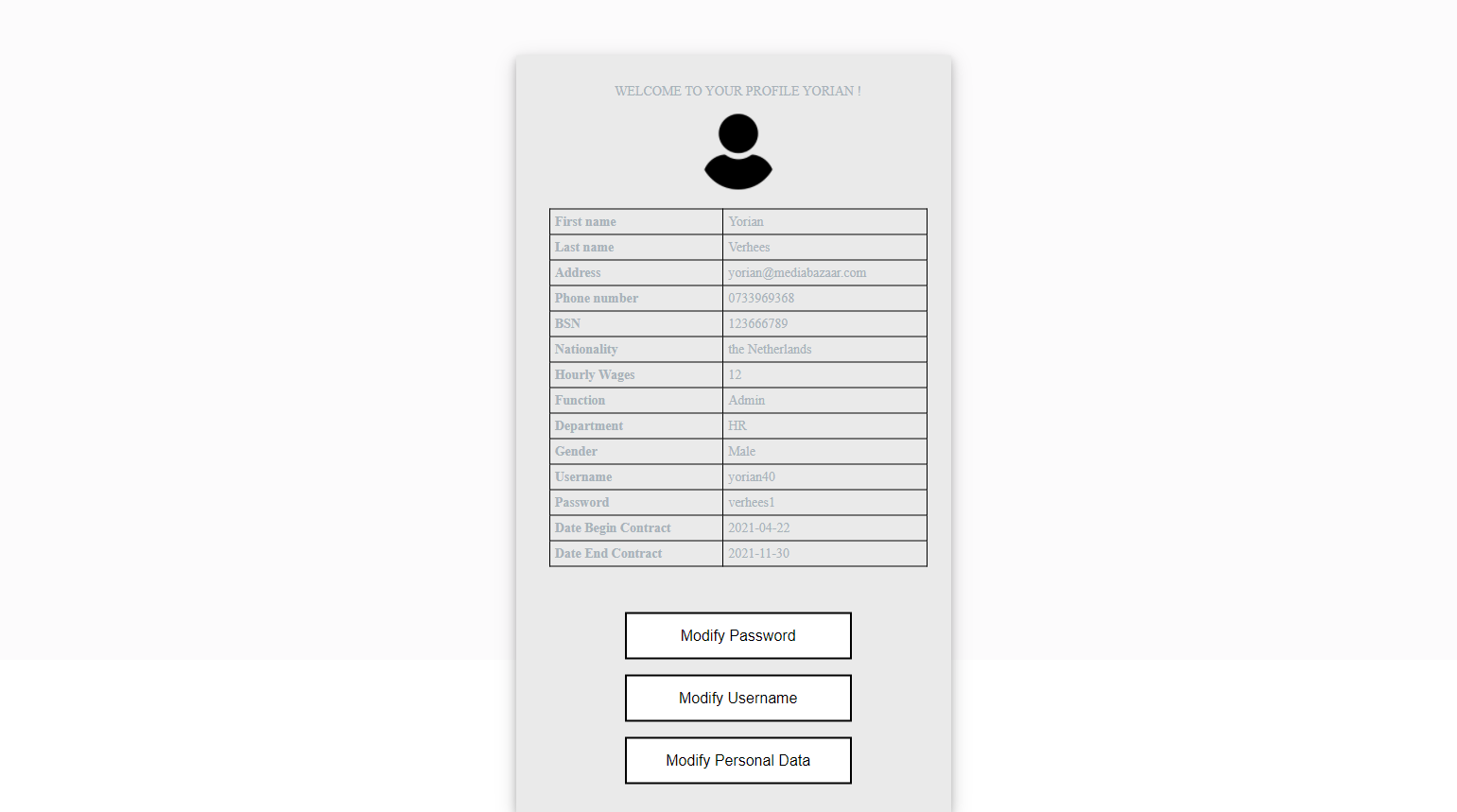
Graphical user interface, application

Description automatically generated

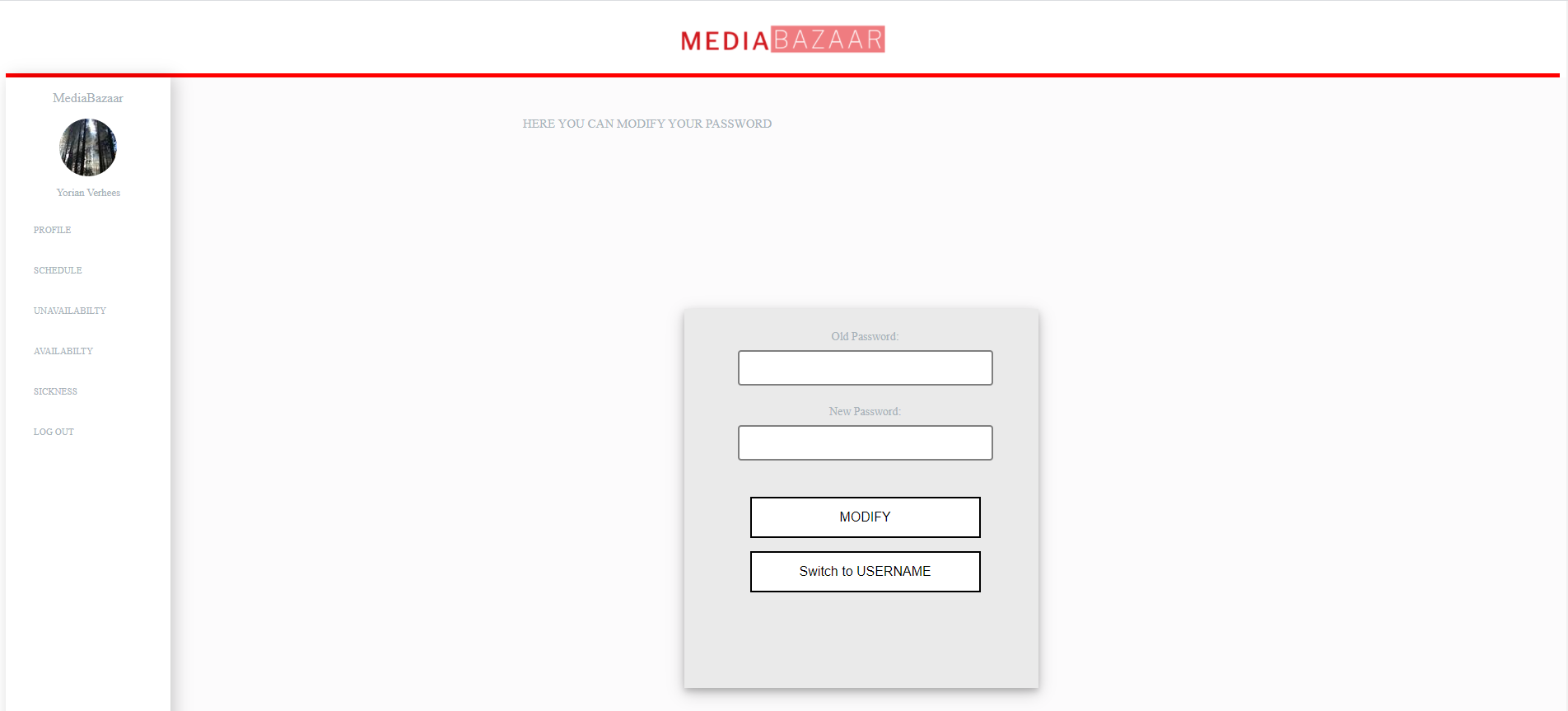
This is the login page where the employee enters his credentials



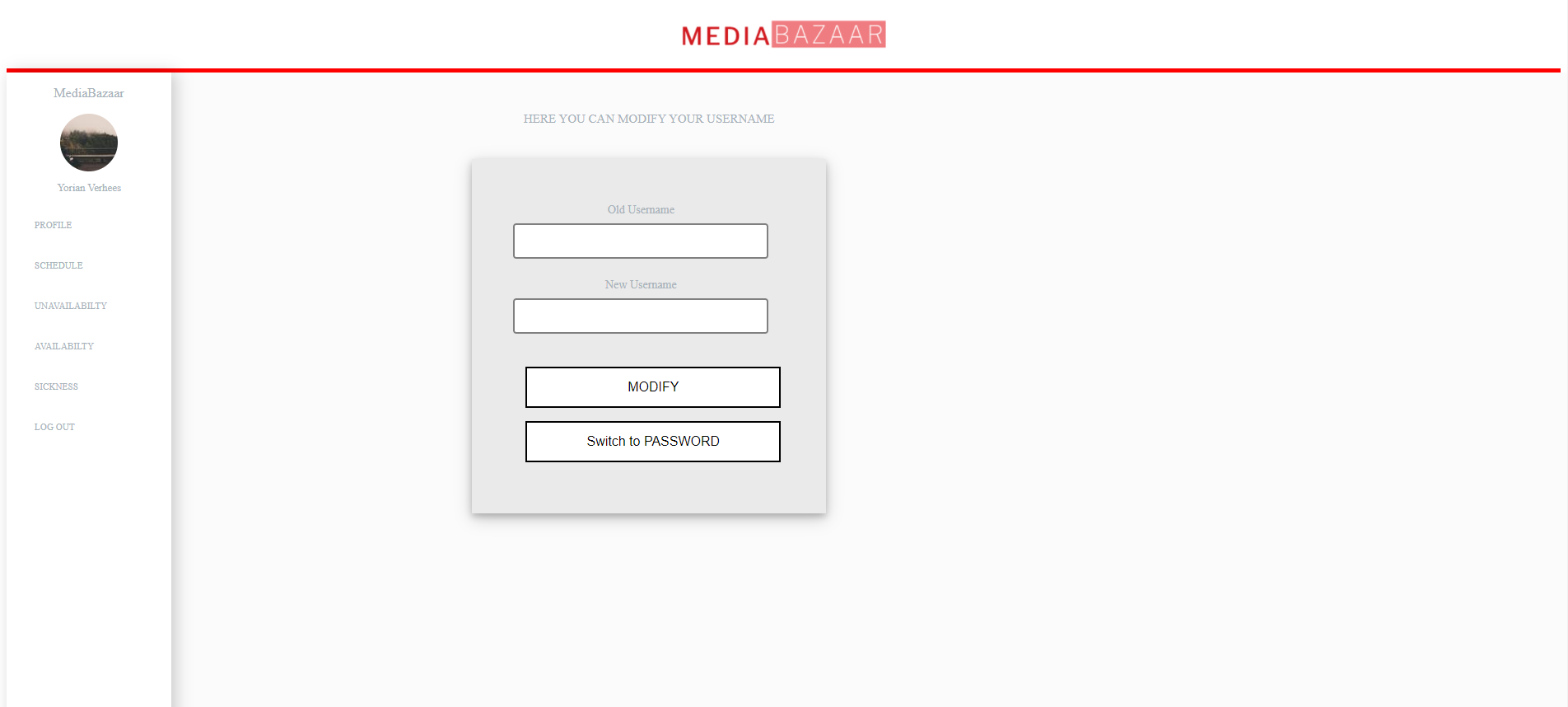
After the employee logs in the first page to see is his own schedule



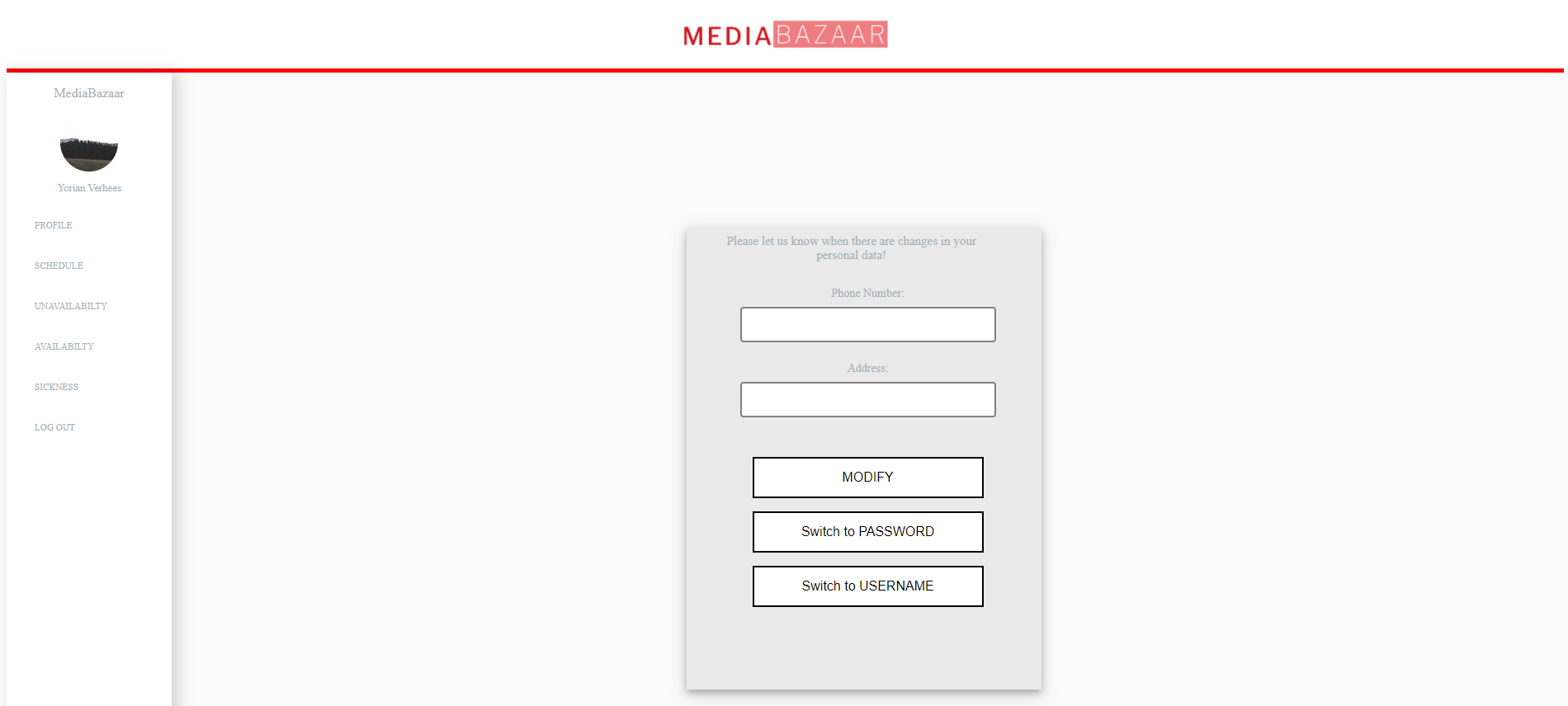
This is the employees’ profile page where he can see the data about himself and also chose to modify information



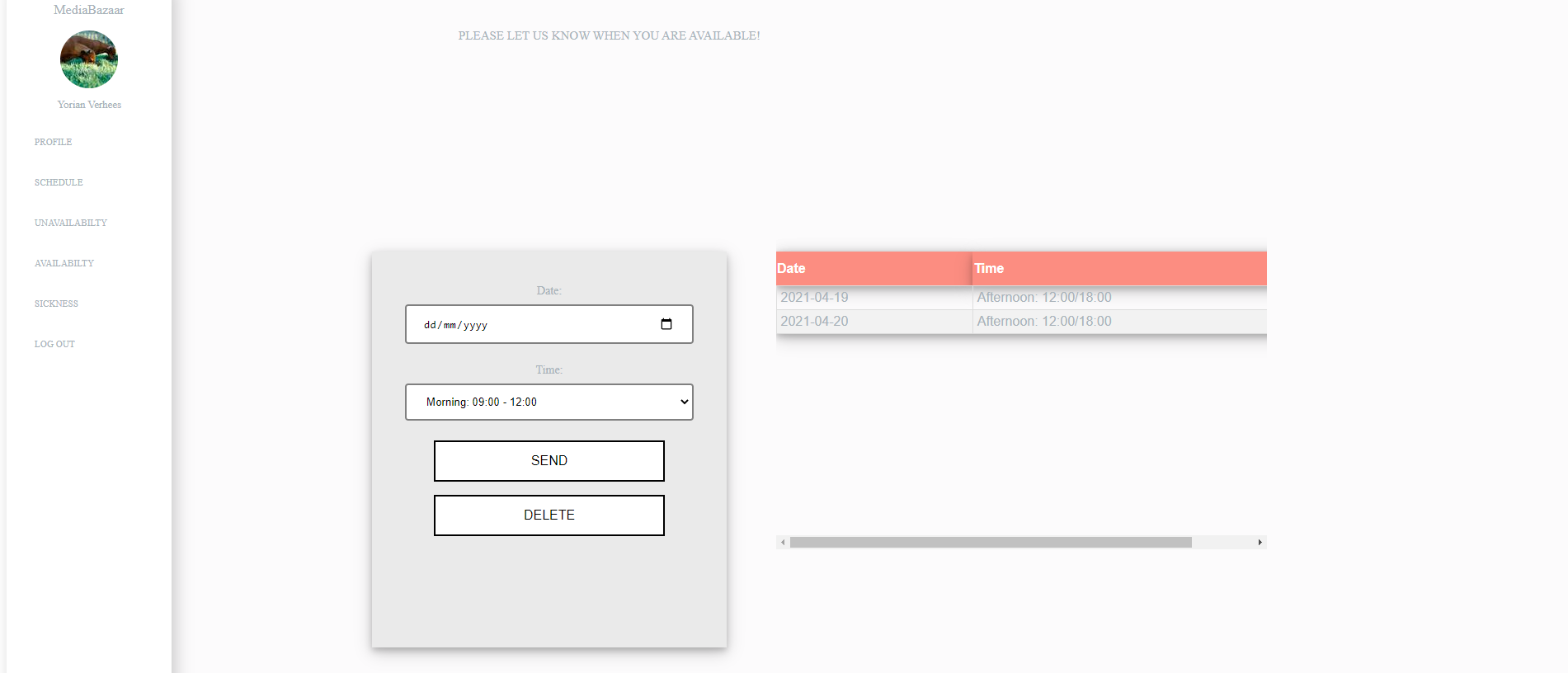
This is the tab where he can either modify his password or chose to go and modify his username



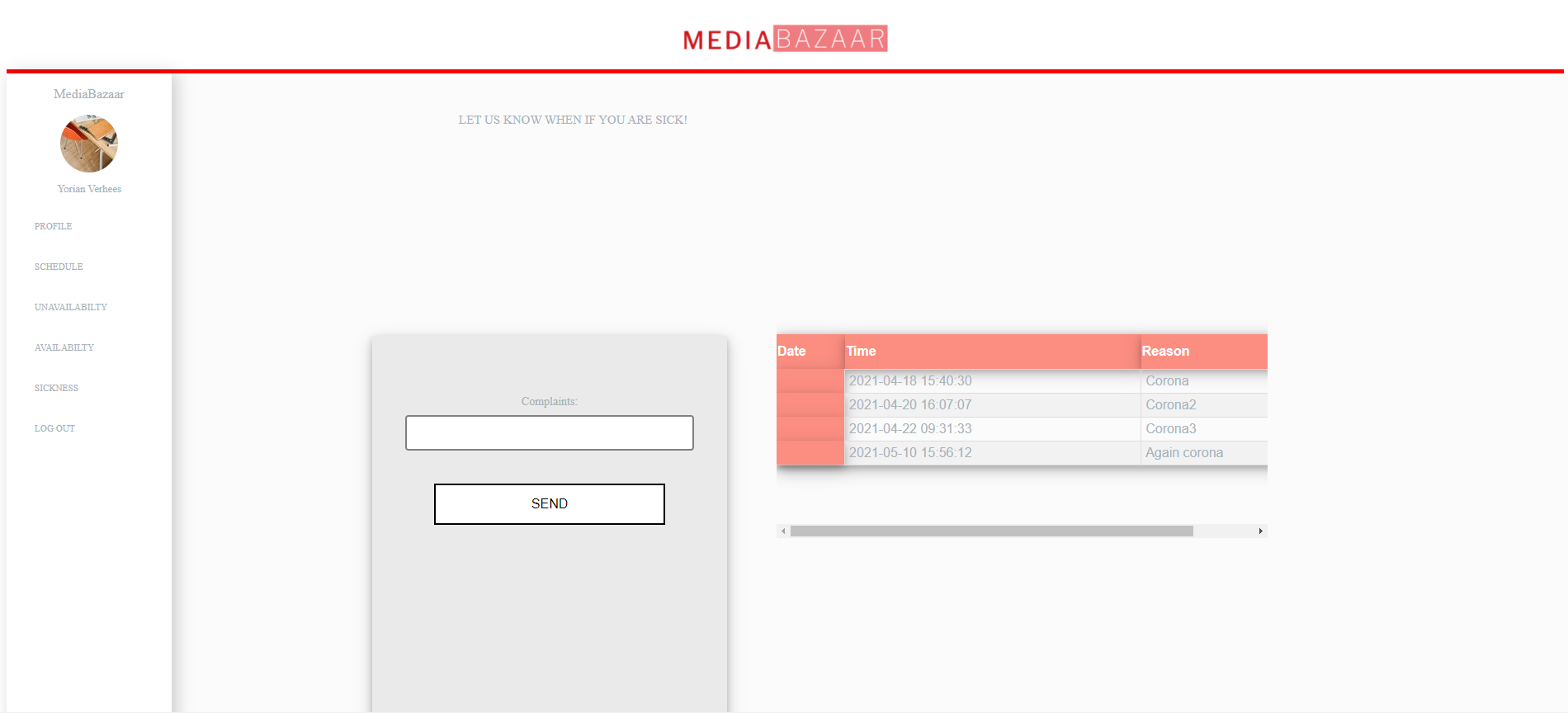
This is the tab where he can either modify his username or chose to go and modify his password



The employee can also decide to modify his phone number and address if he desires to.



This is the availability tab where the employee can mark which are his preferred shifts.



In the sickness tab the employee can mark when he will miss work due to sickness

Graphical user interface

Description automatically generated

In the unavailability tab the employee can say when he is not available to work.