Software Requirements Specification

Staff Management System

Version 1.0

Feb 23rd, 2021

Gordon Baliles

Tai Quach

Edgar Meza

Tai Pham

Simran Tamang

CS 3321 Software Engineering

1. Introduction

1.1 Purpose

This is a Requirements Specification document for “Staff Management System” software for the chain of restaurants business. This part gives an overview of everything in this document. The purpose of this document is to present all the detailed descriptions of the requirements for the “Staff Management System” software (SMS). This document describes the scope, functional requirements, non-functional requirements, user interface (UI) with the use case.

1.2 Project Summary

Project Name: Restaurants Staff Management System

Project Manager: Dr. Emre Yilmaz, CS 3321 Instructor

Project Developers: Gordon Baliles, Tai Quach, Edgar Meza, Tai Pham, Simran Tamang

Users: Jack Tran, Owner of Mein Restaurant

Mike Tran, Owner of Ishin Udon Restaurant

Victor Ly, General Manager

1.3 Background

Jack Tran and Mike Tran are two siblings who have experience with two restaurants and have taken over their parent’s small restaurant business. In recent months, their small restaurant has grown in popularity and the brothers are wanting to expand. In order to expand their current business, they would like to have a Staff Management System created to assist with the amount of staff they will need to run a much larger, successful restaurant business.

Their current ways of managing staff is very problematic when it comes to expanding:

- Employee’s use manual punch clocks to document hours worked
- Punch cards are manually transferred to logs for time recording purposes.

- Employee files are stored in manual files.

The brothers have requested that a system be created to streamline and digitize the way they manage staff. The new system should allow the Owners and General Manager to access employee's time logs and records, create/delete employee's information as needed. Also, from the time logs and attendance report, owners can modify the payroll. Moreover, modifying the employee's schedule should be easier with the system.

1.4 Project Scope

The scope of the project is a local network-based system that supports multiple users, which includes Owners, General Manager, and employees.

1.5 System Purpose

1.5.1 Users

Owners:

The owners will have complete access to all portions of the system upon implementation of the new system. The Owners/Admin will find it very easy to manage the restaurant's staff members. In addition, they will have complete access to all portions of the system and easy to view, keep track, pull data of all employee work hours for payroll purposes.

General Manager:

The General manager will be able to access all employee portions of the system to make changes, add new or remove an existing staff member.

Employees:

Employees will be able to clock in quickly and accurately with access code as well as log into the system to document their hours.

2. Functional Objectives

2.1 High Priority

1. The system shall allow all users to reliably login. This will allow attendance to be logged, hours worked tracked, and allow for the calculation of the employee's wage by the system, saving x% of the owner or manager's time over manually calculating wages.
2. The system allows the managers and owners to add new employees to its database. This will make sure the system can let the manager or owner to track their attendance, update the employee's schedule and calculate their wages. This will cut the x% amount of time that managers or the owner may have to manually spend looking up employees.
3. The system must allow the manager or owner to register their employee's information. Doing so will save time in contacting the employees in case of emergencies or any other issues.
4. The system must be able to generate a static QR code for each employee. This service will cut security costs and reduce risk.
5. The system will allow the owner and managers to modify the attendance of their employees. This will cut x% amount of time needed to calculate the right payroll in case of human errors.
6. The system will allow the owner to modify all their employee's payroll. This would cut x% amount of time needed to calculate costs if human error occurred. It will also minimize complaints by x amount if a human or system error occurred.

2.2 Medium Priority

1. The system must be able to add to its database the time of attendance and leaving. This will save x% of a manager or the owner's time by keeping a tab of their employee's arrival and departure times. It will also help in calculating their payouts.
2. The system will allow the user to access their timesheet. This will allow the user to view a few important features such as the working times, the hourly rate for the working times, as well as the pay amount going to the selected user. This can save x% amount of time in keeping accurate payroll information as well as save x amount of time for employees to arrive at work instead of checking.

3. The system should be able to generate an attendance report. This allows the manager and owner to check for any mistakes they have noticed while working. This would save x% amount of time when checking if the payrolls are accurate.
4. The system should allow the users to access their payroll reports. Any disputes with hours worked and wages can have x amount of time saved if the user can check what they are getting paid.

2.3 Low Priority

1. The system will allow the user to view their attendance record information. This information will allow the employee to make sure they are logging in the correct amount of time into the system as well as the manager and owner are able to check as well. This can save x% amount of time in settling complaints or mismatching information.
2. The system should allow the users to change their passwords. The user would save x% amount of time if they cannot login and just want to change their password to login. It would also add some protection in case the QR code was taken.

3. Non-Functional Objectives

3.1 Performance

- The system should be able to support x number of users.
- The time taken for a user to view any section of the system they have access should be between x to y seconds.

3.2 Reliability

- The system operational time should be no less than x%.at all times.
- If the system fails, down time should not exceed more than x hours.

3.3 Security

- The system will require users to scan their QR code and enter a password for user verification/system use.
- Passwords will have to meet a strict complexity requirement.

3.4 Maintainability

- System must have above x% maintainability for 24 hours, in other words, a downed component must be fixed x% of the time within 24 hours.

3.5 Compatibility

- The system must be working on the latest and last x amount of OS versions for Windows, Mac, and or Linux.

3.6 Usability

- The UI for the system should be easily understood by all.
- Users should be able to navigate/access the required system areas with minimal training.

4. User Interface Requirements

This section presents all the primary functions of the system. Also, it will show the basic user interface of the system. Only clock in/out authorized for employees, the rest is authorized for owner and manager.

4.1 Login page.

When the owners start the shift of the day and open the application, see Figure 1. It requires the username and password. Of course, just owners and manager access only. Our team will create only one account for owners, we will require security questions, email addresses, phone numbers with some owner's information for privacy.

Figure 1 - Login Page

4.2 Home page

After logged-in, the system will direct to the home page, see Figure 2. The homepage is where the owner controls everything. Employees are only allowed to use the clock in/out tab. The rest is owner and manager access only. We will go through each function on the homepage.

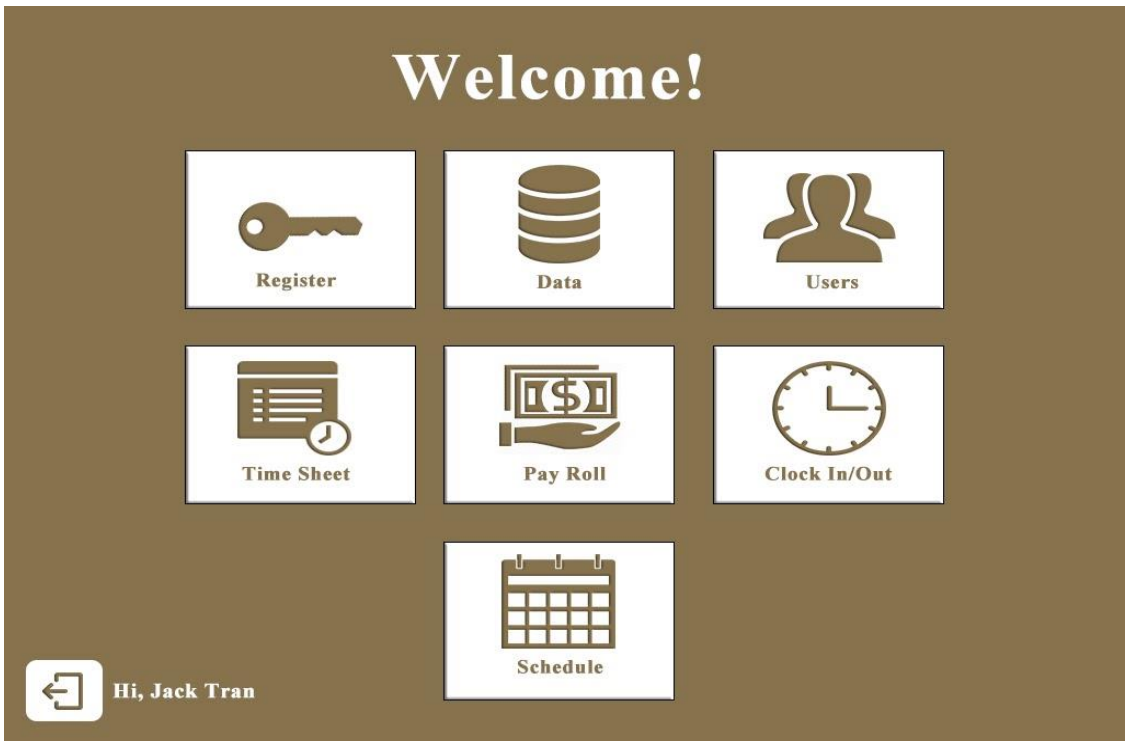

Figure 2 – Home Page

4.3 Clock In/Out

Every employee must clock in and clock out at every single shift. The attendance time will be record on the timesheet. Owner and manager can manage it and then generate the attendance report and payroll.

4.4 Access Code UI

Every user who use the system must enter their own code for every function if they want to access, see Figure 3. This is a very basic “Access Code” user interface for the system.

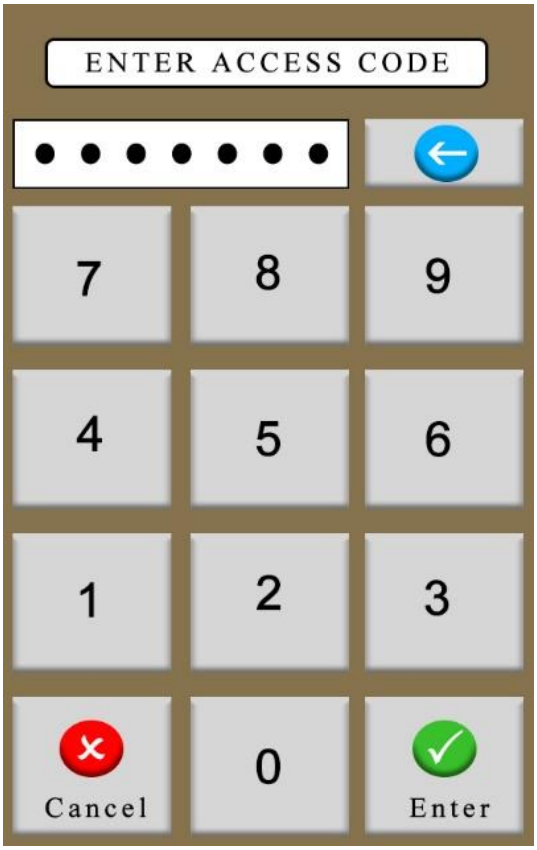

Figure 3 – Access Code UI

4.5 User UI

This function allows owners manage employees in the restaurant. It has a list of employees with their role. They will able to generate the ID card and QR code for employees. Also, if employee forgot to clock in or clock out, they can modify attendance, edit timecard easily. Moreover, owners or manager able to use “Add User” function to add new employee, and they can modify each of employee on the “Actions” tab.

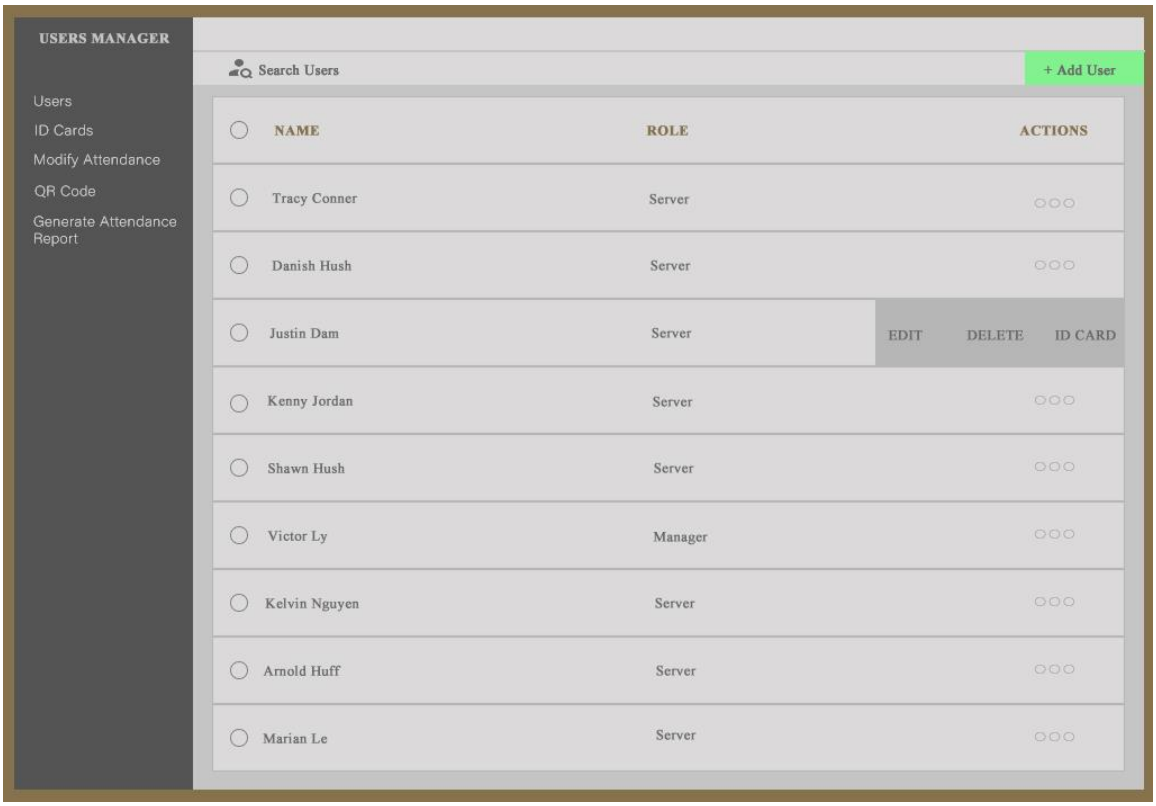

Figure 4 – User UI

4.6 Time Sheet UI

This function will allow the owner modify employee's time sheet to generate the payroll. It will track time in time out each day working of each employee. In the time sheet view, it will display the number of working hours, pay rate, and pay amount for each employee.

	Mon Feb 8	Tue Feb 9	Wed Feb 10	Thu Feb 11	Fri Feb 12	Sat Feb 13	Sun Feb 14	Hours	Rate	Amount
Tracy Conner	-	6:30	-	-	8:00	6:30	6:30	26:9	$10/h	$269
Danish Hush	-	6:30	-	-	8:00	6:30	6:30	26:9	$10/h	$269
Justin Dam	-	6:30	8:00	6:30	-	-	6:30	26:9	$10/h	$269
Kenny Jordan	-	-	-	6:30	8:00	6:30	6:30	26:9	$10/h	$269
Shawn Hush	-	-	8:00	-	-	-	-	8:00	$10/h	$80
Victor Ly	-	8:00	-	8:00	8:00	8:00	8:00	40:00	$20/h	$800
Kelvin Nguyen	-	-	8:00	-	-	6:30	8:00	22:30	$10/h	$223
Arnold Huff	-	6:30	-	6:30	-	-	8:00	21:00	$10/h	$210
Marian Le	-	-	8:00	6:30	6:30	-	-	21:00	$10/h	$210

Figure 5 = Time Sheet UI

4.7 Pay Roll UI

Pay roll is authorized for owner and manager only, they can access to “View Pay Roll” to track the pay sheet for each employee. It will compute net pay base on worked hours and pay rate, deduct the tax for each employee. Also, it will display the status for each pay slip. Moreover, owner and manager can be able to generate the pay roll report to word, excel or any office application.

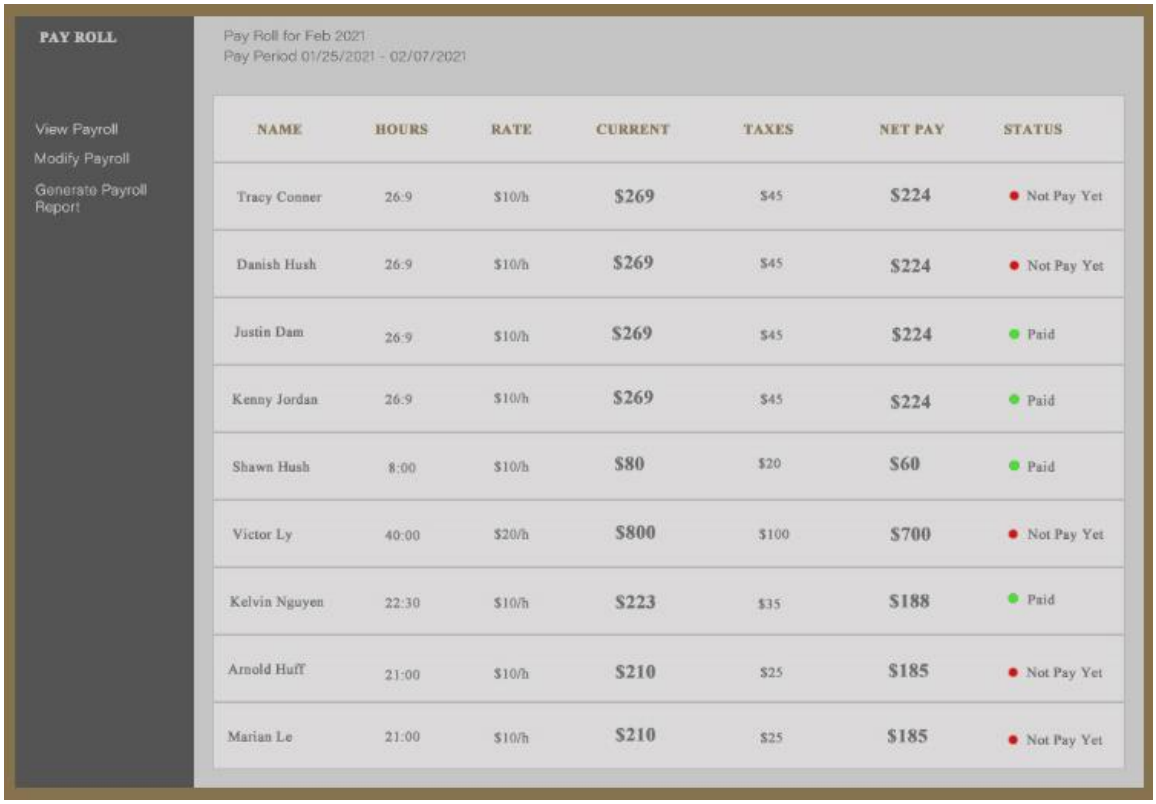

Figure 6 – Pay Roll UI

4.8 Schedule UI

Owner and manager be able to work on schedule for staff on this function. It will display working day for each employee. Also, they can modify schedule in case of emergencies easily. It will also. show day by day, week by week, month by month schedule.

Figure 7 - Schedule UI

5. The Use Case Model

5.1 Use Case Diagram

5.2 Use Case Descriptions

Login

Use Case Name: Login

Summary: Login options are available for owners and manager. Once they logged in, the system will direct to the home page. Employees will be allowed to clock in/out and view work hours's report, but they are required to have their access code. The Owners and Manager will have

additional access to personnel information and display attendance reports, payroll information, etc.

Basic Flow:

1. All users need to log in the system initially.
2. The system asks for QR code and password.
3. The user gives the information.
4. The system checks against all registered users.
5. The system begins the login process and allows users to access their level of information.

Alternative Flow:

Step 4: if the user provides an invalid QR code go back to step 2

Step 4: if the password is wrong, the system asks for a new one and continues step 4 with the new password

Preconditions: User must be registered

Postconditions: The user can access their level of information

Register

Use Case Name: Register

Summary: In order to work for the business and access system items, the user must be registered. Register case is located in the homepage and can only be accessed by the owner and manager.

Basic Flow:

1. The user must indicate they wish to register on the homepage.
2. The system asks for a QR code and password.
3. The user provides the QR code and password the owner provides.
4. The system checks for duplicates on the QR code and any password requirements.
5. The system asks for a name, street address, zip code, phone number and email.

6. User enters their information.
7. The system stores the information and determines their access level.
8. The system starts the login session and allows the user to access their level of information.

Alternative Flow:

Step 3: If the user provides a duplicate QR code, the system displays a message and goes back to Step 2.

Step 4: if the user provides an inadequate password the system goes back to Step 2.

Step 5: if the user does not provide some information, the system displays notice and repeats step 5.

Preconditions: Owners or managers only

Postconditions: The user can access their level of information

Add Attendance

Use Case Name: Add Attendance

Summary: When the system is logged in, it updates their record displaying up the work. Employees have access to only log in/out but in order to add attendance, view attendance or modify attendance then the system needs the manager or owner authorization.

Basic Flow:

1. The user logs in.
2. The system updates with their time and day of login.
3. The system will then keep track of time of employee's attendance until they clock out.
4. The user selects clock out on their user tab.
5. The system ends time tracking and stores the length of time the employee worked that day into its database.

Alternative Flow:

Step 1: If the user forgets to login, others may access and update the information manually

Step 4: The user forgets to clock out, resulting in the system sending a notification to the manager or owner and requires manual modification

Preconditions: User logins

Postconditions: Their attendance report is updated

View Attendance

Use Case Name: View Attendance

Summary: The user may see if they are showing up for work as required and see days they have missed or forgot to enter

Basic Flow:

1. The user logs in and selects the view attendance option.
2. The user selects the option to view their attendance report.
3. The system fetches the stored information and represents it to the user.

Alternative Flow: none

Preconditions: User must be logged in and executing the use case View Attendance

Postconditions: The user may ask a manager or the boss to update their information if they forgot or had exemptions like injuries

View Timesheet

Use Case Name: View Timesheet

Summary: Viewing time sheet is authorized to manager and owner only, however each employee can view and track their own time sheet. The report will display the number of hours that the employee worked that week and his paycheck report but cannot modify or change since they don't have access.

Basic Flow:

1. The use case begins when the user selects the “View Timesheet” option in the login page.
2. The system grabs the preset schedule made by the manager.
3. The system presents the time schedules, calculates and presents the hourly rate and pay.

Alternative Flow: none

Preconditions: The user is executing the use case View Timesheet

Postconditions: User may not view their time schedules for the week

Generate Attendance Report

Use Case Name: Generate Attendance Report

Summary: In order to see who clocked in as well as manage staff, the manager and owner may generate such reports to see if their employees are coming to work

Basic Flow:

1. The manager or owner selects the use case “Generate Attendance Report”.
2. The system asks for the length of time, by week or month.
3. The system creates a spreadsheet of all the days the employee showed up for work and the days they have missed.

Alternative Flow:

Step 2: if the owner or manager inputs an invalid value, the system repeats step 2

Preconditions: The owner or manager executes the Generate Attendance Report use case

Postconditions: The owner or manager may now see if their employees are showing up for work

Access Payroll Report

Use Case Name: Access Payroll Report

Summary: The owner or manager may see if they are paying the proper amount to their employees.

Basic Flow:

1. The owner or manager selects the use case “Access Payroll Report”
2. The system brings up all the registered employees for the manager, or the employees and managers for the owner as well as their pay for the week beside their names

Alternative Flow: none

Preconditions: The owner or manager executes the use case Access Payroll Report

Postconditions: The user may now view how much their employees are being paid

Change Password

Use Case Name: Change password

Summary: In case of breaches or lapse in memory, users are able to change their passwords to continue logging in for work

Basic Flow:

1. The user executes the use case “Change password” .
2. The system asks for the old password and then asks for the new one twice to confirm the new password.
3. The system is updated with the new password as part of the login information for the user.

Alternative Flow:

Step 1: If the user forgets their password to login, they may choose to change it outside their profile interface.

Step 2: The system asks for the user to provide an email.

Step 3: If the email matches, the user’s email is sent a link to change their password.

Step 4: The system is then updated with their new selected password.

Preconditions: The user must be registered or logged in

Postconditions: The user may login with their new password

Modify Schedule

Use Case Name: Modify Schedule

Summary: In case of emergencies and other human and natural issues, the owner and manager are able to change their employee’s schedules to meet such issues.

Basic Flow:

1. The owner or manager executes the “Modify Schedule” use case.
2. The system retrieves all the employees the user’s level is able to access.
3. The system then allows the user to access and edit the timesheets of any certain individual.
4. The timesheets stored by the system would then be updated.

Alternative Flow:

Step 3: if conflicting times are given, the system would provide an error message and not save the changes and repeat step 3

Preconditions: The user executes the Modify Schedule use case

Postconditions: The user may now edit their employee's schedules if necessary

Modify Attendance

Use Case Name: Modify Attendance

Summary: Users may forget to login during work, especially during rush hours or after an emergency or they may have to leave.

Basic Flow:

1. The owner or manager executes the "Modify Attendance" use case in their view.
2. The attendance sheet stored by the system is displayed.
3. The user may then edit it as necessary.
4. The system then saves the changes made to the attendance sheet.

Alternative Flow:

Step 3: If the user creates invalid additions like repeated days or negative times the system outputs an error, does not save the changes and repeats step 3.

Preconditions: The user executes the use case Modify Attendance

Postconditions: The user may now properly correct the attendance sheet

Modify Payroll

Use Case Name: Modify Payroll

Summary: Unexplained absences, overtime, and many other scenarios may require the owner to change the payroll of his employees.

Basic Flow:

1. The user executes the "Modify Payroll" use case.
2. The system brings up the payroll data stored in it.
3. The user selects the person being modified.
4. The system asks the user to change the value.
5. The new payroll value is saved by the system.

Alternative Flow:

Step 4: if the user inputs an improper value or outside preset limits, the system outputs error and repeats step 4

Preconditions: The user executes the Modify Payroll use case

Postconditions: The payroll is now changed due to some situation

Generate QR Code

Use Case Name: Generate QR Code

Summary: Each employee needs a QR code to identify them and the owner is able to produce them for their employees.

Basic Flow:

1. The user executes the “Generate QR Code” use case.
2. The system generates a QR code for the user.
3. The system asks for the employee who will be assigned the QR code.
4. The user selects the employee.
5. The system saves the QR code to the employee profile.

Alternative Flow:

Step 2: if the QR code is somehow the same as another, it is checked and a new one is generated before moving to step 3

Step 4: If the user selects a non-existing employee or provides not matching information stored in the system, an error is given, and step 4 is repeated

Preconditions: User executes the Generate QR Code use case

Postconditions: The owner can now store QR codes to his employees to both identify them for attendance and such as well as set their basic user settings

Add User

Use Case Name: Add User

Summary: The owner needs to be able to add his employees to the system so that their attendance, payrolls, timesheets, and other various information may be given and tracked

Basic Flow:

1. The user executes the “Add User” use case.

2. The system asks for the basic information, a name, a QR code, and some password.
3. The user inputs the required information.
4. The system generates fields for the user such as address and email that employees may enter when they register.
5. The user profile is saved to the system.

Alternative Flow:

Step 3: if the user does not provide the necessary fields, an error message is output and step 2 is repeated

Step 3: if the user provides some faulty information like numbers only for a name, an error message is output, and step 2 is repeated

Preconditions: The user executes Add user use case

Postconditions: A profile is made for an employee