FACULTY OF INFORMATION TECHNOLOGY CTU IN PRAGUE

Assignment of bachelor's thesis

Title:	Design of Task Management System for Chain Stores
Student:	Alena Kužmová
Supervisor:	Ing. Marek Suchánek
Study program:	Informatics
Branch / specialization:	Web and Software Engineering, specialization Software Engineering
Department:	Department of Software Engineering
Validity:	until the end of summer semester 2021/2022

Instructions

Many chain stores are dealing with task management issues. The goal of this thesis is to analyze this domain and design a system to support work efficiency. It will support key activities of chain stores staff and allow them to oversee their workload.

- Analyze and describe task management in chains stores. Create a conceptual model to summarize the key aspects of the domain.
- Specify functional and non-functional requirements. Furthermore, model use cases for both store employees and managers.
- Design the architecture concerning the requirements and analysis. Consider also legal obligations related to personal data (e.g. GDPR).
- Design UI for different users of the systems and different types of devices used by employees.
- Evaluate the benefits brought by the system to chain stores and propose possible future steps.

Electronically approved by Ing. Michal Valenta, Ph.D. on 18 October 2020 in Prague.

FACULTY
OF INFORMATION
TECHNOLOGY
CTU IN PRAGUE

Bachelor's thesis

Design of Task Management System for Chain Stores

Alena Kužmová

Department of Software engineering

Supervisor: Ing. Marek Suchánek

May 11, 2021

Acknowledgements

I would like to thank my supervisor Ing. Marek Suchánek, without his assistance and dedicated involvement in every step throughout the process, this thesis would have never been accomplished.

Declaration

I hereby declare that the presented thesis is my own work and that I have cited all sources of information in accordance with the Guideline for adhering to ethical principles when elaborating an academic final thesis.

I acknowledge that my thesis is subject to the rights and obligations stipulated by the Act No. 121/2000 Coll., the Copyright Act, as amended, in particular that the Czech Technical University in Prague has the right to conclude a license agreement on the utilization of this thesis as school work under the provisions of Article 60(1) of the Act.

In Prague on May 11, 2021 .....

Czech Technical University in Prague

Faculty of Information Technology

Citation of this thesis

Kužmová, Alena. *Design of Task Management System for Chain Stores.* Bachelor's thesis. Czech Technical University in Prague, Faculty of Information Technology, 2021.

Abstrakt

Mnoho obchodních řetězců čelí problémům spojeným se správou úloh a časovým managementem svých zaměstnanců. Právě z tohoto důvodu si tato bakalářská práce klade za cíl navrhnout softwarový nástroj, který bude takové funkce poskytovat a umožní zefektivnění práce. První část je věnována analýze aktuálního stavu v obchodním řetězci Lidl, pro který byl tento nástroj primárně vyvíjen, což následně umožnilo navrhnout požadavky týkající se funkcionality i uživatelského prostředí samotné aplikace. V dalších kapitolách je rozebírána implementace tohoto návrhu a následné vyhodnocení, ve kterém jsou shrnuty silné stránky aplikace, ale také její limitace, případně možnosti budoucího rozšíření.

Klíčová slova obchodních řetězec, správa úloh, statistiky uživatel, pracovní efektivita, statistika úloh, analýza, UML, wireframes

Abstract

Many chain stores face challenges related to the task and time management of their employees. For this reason, this bachelor's thesis aims to design a software tool that will provide such functions and enable more efficient work. The current state of workflow analysis in Lidl retail chain is described in detail in first part of this thesis. Consequently to that analysis, this tool was designed to fulfil all requirements regarding the functionality and user environment of the application. The following chapters discuss the design and the subsequent evaluation, which summarizes the application's strengths and its limitations or possibilities for future expansion.

Keywords chain store, task management, user statistics, work efficiency, task statistics, analysis, UML, wireframes

Contents

List of Figures

Introduction

My brother once came to me with the idea of some task manager for grocery store employees and management. Right away, I told them about Jira [1] or Trello [2] as I work with them daily, and they are commonly used for workflow management in school and work projects. To my surprise, I found out these tools are used mostly or even only by IT departments of companies even though there is a considerable need for this kind of technology in other departments as well. [3][4][5]

After consulting the idea of a task manager with many managers and employees of specific companies, a few friends and I decided to create a start-up for task management development. This app will be divided into the management part and the user part because there are many general data protection regulations. Also, there is a great potential in using employees' phones as to this day they are forbidden in-store, but also if they do not want to use them, there is always a way to use the app on the device in the back office.

Goals

This thesis aims to analyze, describe task management in chain stores, and create the first part of the plan for creating the application, which will describe the software development process and also appearance and the functionality of the application

The thesis contains an analysis and description of task management in chain stores. Information collected from the analysis will create a conceptual model to summarize the domain's critical aspects from the managers' requirements and analysis of the current solution's main problems, functional and non-functional requirements for the application. The use cases will describe the potential scenario in which a system receives an external request and responds to it for both employees and managers. The application will divide into two separate parts: first one for managers and team leaders and another one for

employees. A management application should work just within companies VPN, and there will be a predefined number of devices that can be used.

The thesis will consist of designing the architecture from the requirements and analysis, considering legal obligations related to personal data. Also of creating user interface for different types of devices and users, and lastly the evaluation of the benefits brought by the system and the possible future steps for the system.

Structure

Structure of this thesis follows the software engineering approach. In the beginning, Chapter 1 is focused on the Lidl and analysis of its inner employee hierarchy and processes. Following the description of main problems and the current solutions. The next chapter, Chapter 2 focuses on the analysis of the company's specific requirements for the application. These requirements are divided between functional and non-functional requirements. Chapter Chapter 3 describes the tools and technology used in this thesis and their advantages and disadvantages. Chapter 4 defines structural and behavioral models that were created from the system requirements. Following withChapter 5 which shows design and wireframes of the application for the different devices and users. Chapter 6 describes the benefits of the application and the possible future features that can be considered. [6]

Chapter 1

Analysis

1.1 Lidl chain stores

Lidl Stiftung and Co. KG is a significant discount supermarket chain with over 10 000 stores, mainly in Europe and a few in the United States. Founded in 1973 by Josef Schwarz and is part of the Schwartz Group, like Kaufland, which is also one of Europe's biggest supermarket chains. Lidl is approaching zero waste and no-frills, they show most of their products in original packaging to minimize staff, and if the carton is empty, it is easy to replace. Lidl is also currently developing the first zero-carbon building in Sweden. [7] [8]

1.2 Company hierarchy

As mentioned earlier, Lidl is trying to minimize the staff and be as efficient as possible with the lowest staff quantity. This software will be designed for the specific requirements of Lidl. All of this information was obtained during many meetings and discussions with Lidl representatives to fully understand and describe such a big company's processes. With the complication of GDPR, which means only managers on specific devices with Lidl's VPN can access information about users, it is essential to understand the company's structure. In our case, the company has stores across Europe and the United States, so each country also has different rules and policies.

In this thesis, the Slovak part of the company and its hierarchy will be discussed. The Head of the company in the Slovak region is the state manager that overlooks three regional managers, as shown on the image Figure 1.1 below. Each Regional manager has assigned a specific region with warehouses. One Regional manager oversees ten Regional store managers, each of them has to take care of three to six stores in the region. The number of stores differs on the sales in each store. Each store has a store manager and assistant manager, the number of employees in each store is different, but mostly it is around 12. Employees work on various tasks during the shifts, such as running from the

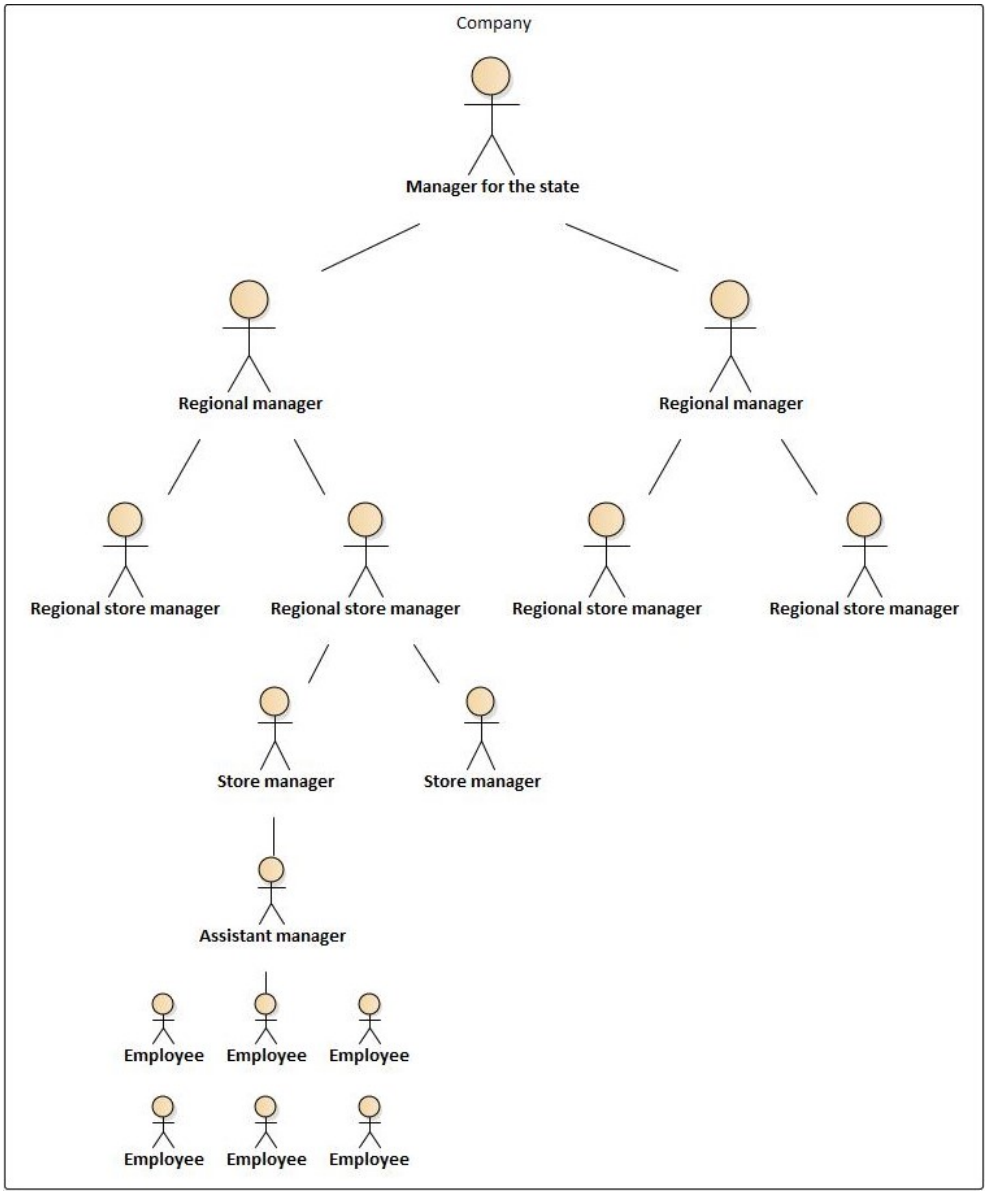

Figure 1.1: Company hierarchy

bakery point to the cash register or unload new goods.

1.3 Current solution

We will take a closer look at one of the stores and its inner processes during the regular shift. At the beginning of the shift, the store manager checks the employee list on the tablet connected to Lidl's VPN. Who is assigned to work and checks what tasks should be done that day. The store manager knows if

some goods will be refilled from the warehouse that day and how many pallets are supposed to be unloaded. In additionally that, sometimes palettes come destroyed, not in specific quality or quantity as expected.

The store manager assigns tasks in the morning, but it is difficult to chase employees to change the task if anything happens. The employee can forget what the tasks were, and in this situation,they has to find a store manager. Employees are switching and interrupting tasks as they can be asked to go from unloading shipment to opening cash registers because there are too many customers waiting in line. In this case, the store manager does not know why it takes that long to unload the shipment and thinks that the employee is inefficient. It is uneasy for the Store managers and Assistants to track the employees' workflow and quality.

1.4 Main problems

Example of one employee's day estimation by the assignation of tasks:

- 6:00 - 6:50 bakery point
- 7:00 - 8:00 fruit and vegetable quality check and reorganising
- 9:00 - 11:00 unloading shipment
- 11:30 - 15:00 refilling goods

Employees actual day:

- 6:00 - 6:50 bakery point
- 7:00 - 7:30 cash registry
- 7:30 - 8:00 fruit and vegetable quality check and reorganising
- 8:00 - 8:30 customer stops employee and asks for refill of specific goods
- 8:30 - 10:00 too many customers in line, so the employee is required to open another cash registry
- 10:00 - 10:30 shortage of bakery goods- the employee has to go to the bakery point to refill the shelves
- 10:30 - 13:00 unloading shipment, three employees are assigned to this task, but two them have to rush to the cash registry and refill the goods
- 13:30 - 15:00 refilling goods

After analyzing the workflow of employees and comparing estimation with the actual day of the employee, it is clear that there are many problems:

- manager cannot oversee the workflow of the employees,

- employee could not comment on the task completion if there were some problems such as the palettes were damaged,
- employee has no way of notifying the manager that the task was interrupted,
- manager has a hard time changing tasks during the shift.

Based on these problems and requirements specified by Lidl employees a system of requirements for the application has been created and can be found in Chapter 2.

The application should solve these problems and help optimize the workflow during the shifts, it will also help manager to control workflow. Employees will use their private phones for the application, so Lidl does not have to buy new devices, creating less carbon footprint, which goes in hand with Lidl's eco-friendly policy. [9]

The manager or assistant will be able to assign tasks for the employees before and during the shift. Employees can start one of the tasks assigned to them. The task's timer starts to count the time employee has spent on the task by starting the task.

The employee can pause the current task, which will also stop the timer and start the different tasks. This feature will solve employees' problems running from task to task, as shown in the employees actual day without the manager notification.

After completing the task, the employee can write comments on the task, mentioning any problems during the completion.

Chapter 2

System requirements

After the consultation with the Lidl managers, the system requirements description to solve the main problems and to optimize to current workflow can be seen in the analysis of Chapter 1 and optimize the current workflow. The requirements describe the features and usage of the system.

These requirements also describe the business processes, system behavior, and the critical performance parameters. These parameters need to be met by the system for the customer to be satisfied. This thesis will further describe the functional and non-functional requirements with the following of the use cases created from these requirements. [10]

A functional requirement is a description of necessary software behavior, including calculations and data input; it is how the system does or does not respond to input. [10]

Non-Functional Requirements in Software Engineering presents a systematic and pragmatic approach for building quality into the software systems. They define system attributes such as performance, security, scalability, usability, modifiability, and many more. [11] [10]

Requirements have a specific hierarchy using the MoSCoW prioritization technique to understand and manage the priorities of the requirements. This technique separates requirements to Must-have, Should-have, Could-have, or Won't-have this time. This classification provides a good indication of the requirement and its expected completion. [10][12]

2.1 Functional requirements

2.1.1 'Must-have this' requirements

2.1.1.1 FR01 Division of permissions

Employee

- user can see only tasks assigned to them

- user sees assigned tasks and adds comment after finishing
- user can change state of task
- user end server contains no information about users
- user does not see user info only UID
- user has read only notifications privileges
- log-in through generated token
- only one device

Management

- manager coordinates all of the tasks and users that they oversee
- managing personnel
- manager can create and control tasks
- manager is able to see reports and statistics of specific employee and store
- manager assigning users to tasks
- manager creates notifications
- manager is connected to Lidl database with user info
- log-in only on VPN and specific devices provided by Lidl

2.1.1.2 FR02 Log in to the app

Employee downloads the app on the smartphone and has their microchip. An employee scans the microchip in the scanner and gets 'Log in pin code', which will be disabled after usage.

Employee types 'pin code' to the application's login page on the smartphone and logs in. Log in will last for three months. After this period, employees will be logged out and redeem the new 'Log in pin code', and enter it into the application.

Manager has two ways of logging in:

1. Username and password - Manager has a 'manager tablet' with the application and uses username and login information that have been generated by store Lidl information system.
2. Log in pin code - same as Section 2.1.1.2

2.1.1.3 FR03 Tasks management

Manager creates the task and assigns it to one or more employees with the name of the task and description. Managers can add priority to a task for employees to know what they should do first. The manager would have some basic tasks prepared to assign or can create new ones and save them if they are used frequently.

Employee can decide which of the assigned tasks to start first or have a specific priority.

2.1.1.4 FR04 Task workflow

Tasks can be in states:

1. *Created*
2. *Started*
3. *Declined*
4. *Paused*
5. *Finished*

Task is finished after all employees finish or decline the task

2.1.2 Should-have this requirement

2.1.2.1 FR05 Creating statistics

Manager will see information's about time that employee spent on the tasks.

2.1.2.2 FR06 Device ID

The back-office device generates the pin, and after login, the id is saved, and the employee does not have to log in again until the employee wants to use a different device. The employee has only one device, and it is the last logged in with a pin.

2.1.3 'Could-lave this' requirement

2.1.3.1 FR07 Shift attendance

The manager assigns people to a specific shift at a specific time. There are pre-prepared times to assign people which will work that day in-store or change/delete that person, assigning people to tasks at a specific time to see who is that time at the store.

After assigning people to specific shift, they are visible on top of the list of people assignable to tasks

2.1.4 'Won't-have this' time

2.1.4.1 FR08 Creating Reports

Generated reports are from statistics, and they will be defined by the company what they want to see specifically. An example of the report compares

the employee efficiency on the same task from the employees throughout ten different stores.

2.2 Non-Functional requirements

2.2.1 NF01 Devices

There are 3 types of devices that will be used: smart phone, tablet and computer

2.2.2 NF02 Platforms

1. Android 4 and higher - depends on implementation,
2. iOS,
3. web browser,
    a) Internet Explorer 9, 10, and 11
    b) Chrome 49 and higher
    c) Mozilla Firefox 52 and higher

2.2.3 NF03 GDPR

All user info is stored in Lidl database and only employees with company VPN and permissions have the rights to see it.

2.2.4 NF05 Response time

There will be an icon of sending until its delivered. The sending does not have to happen right away, it can be delayed for about 2–3 minutes.

2.2.5 NF06 Security

Only managers with permissions and specific device granted by Lidl can access information about employees.

2.2.6 NF07 Documentation

User documentation is required for assisting end user to use the application. This documentation consists of employee and manager user documentation.

2.2.7 NF08 Localizations

Application will be available in Slovak and English and ready for additional future translations.

2.2.8 NF09 Scalability

For the further system scaling will be used horizontal scaling. System should be able to hold 1 000 employees at the time but it is designed to increase numbers of users to more that 100 000.

Chapter 3

Tools and technology

3.1 Modeling

Unified Modeling Language(UML) is a standardized modeling language which is mostly used for analysis, design, implementation of software based systems and modeling business processes. UML is not a software development tool, it is a modeling language. [13]

There are two most broad types of UML diagrams:

1. Behavioral UML
    a) Activity Diagram
    b) Use case Diagram
    c) Interaction Overview Diagram
    d) Timing Diagram
    e) State machine Diagram
    f) Communication Diagram
    g) Sequence Diagram
2. Structural UML
    a) Class Diagram
    b) Object Diagram
    c) Component Diagram
    d) Composite Structure Diagram
    e) Deployment Diagram
    f) Package Diagram

g) Profile Diagram

All of the diagrams in this thesis are described more in Chapter 4. In this thesis i use UML to model and document requirements for the application.

For modeling software, I choose Enterprise Architect mostly because of its transparency of relations among models, which helps create big projects. There were many other options such as Draw.IO which I am familiar with from school projects, but due to presence of lagging and lack of tools required for more complex projects, this was not considered suitable option for needs of this thesis. [14] [15] [16] [17]

3.2 Design

For the wireframe designing and prototyping, I was deciding between two software options, Adobe XD and Axure RP 8. Personally, I used both of them, Axure RP 8 at work and Adobe XD for personal and school projects. Both of these programs are great tools for prototyping, but they are, in my opinion, pretty different in the way of usability. [18] [19]

Axure RP 8 is complex software with unlimited options for animating elements and making them interact with each other, but the learning curve is moderate. The designer can be creative with the conditional flows in the interactions, but the complex conditions are time-consuming, and even easy animation changes can consume much time. Axure can drag shapes onto a canvas and build specific designs, but with moderately complex design, the tool becomes hard to overview.

On the other hand Adobe XD is much more straightforward, but there are not many options for animating elements or element interaction in the Axure RP8. Adobe XD has fewer features but with a much better User Experience. I chose this tool because of its pricing, most of its plugins are also free of charge, and UI Kits are available, such as Material Design.[20] [21] [22]

Chapter 4

System design using UML

4.1 Conceptual model

The goal of conceptual model is to capture relationships among objects. It is an abstract model with three most common objects:

1. Boundary objects,
2. Controller objects,
3. Entity objects.

In the diagram, Figure 4.6 I used Entity objects for describing the principles of the application. [23]

Application should solve problem of the current management in the chain stores.

4.1.1 Security

Employees will be identified by id, to take care of GDPR side of the issue. Only manager in manager mode and connected to the Lidl VPN will be able to see any of the employees information. As can be seen in Figure 4.1

The primary focus is on using employees' phones, but they can connect only one device to their account. With the restriction of only one device, there is a sense of security for the application. If an employee's device gets stolen, there is no information obtainable that can hurt the chain store, and the employee can easily connect a new device.

There can be an issue if the manager device is stolen, but this case is improbable because of Lidl's policy on devices connected to the internal network. Managers are forbidden from taking devices out of the store. To Log in to the application, there is a need to fill in log in credentials generated by Lidl. After a short period of inactivity, the user is logged out.

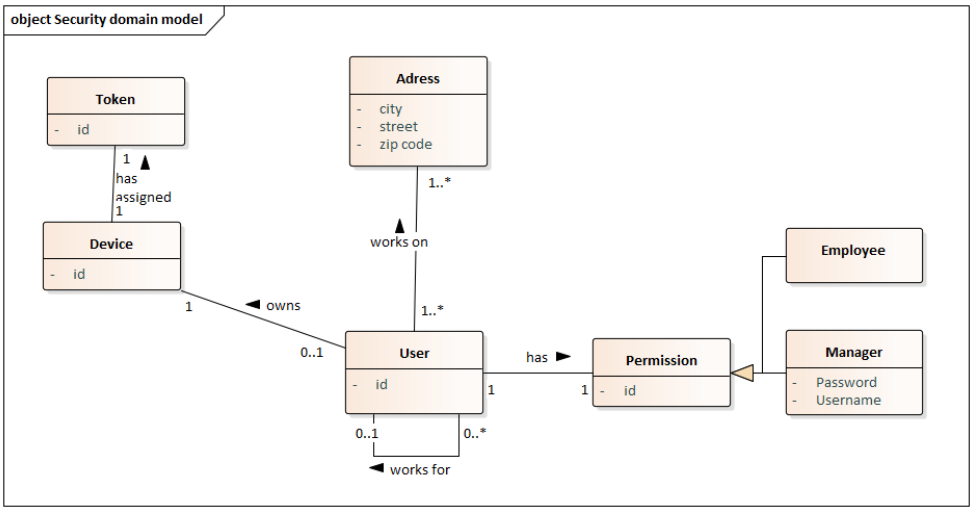

Figure 4.1: Security conceptual model of the task management system

Devices granted by Lidl are connected only when they are in-store or in headquarters. Because of this, there is a limitation for the managers where they can use manager mode. If the manager wants to log in out of the internal network area, they are denied access to the application, and the application will switch to Employee mode. The same thing happens if the manager wants to log in to the manager mode on their devices.

4.1.2 Shift assignations

The manager needs to know who will be on the shift to make task assignations easier and more practical. For the manager, it is more practical to see on the top filtered employees that are that day on the shift. As can be seen in diagram Figure 4.2.

There is a shift assignation that keeps track of employees' shifts as a solution to this problem. Manager at the start of the week or week before assigning employees to specific days and times, e.g. Monday and morning shift. There are already prepared most used time structures to choose from, e.g. morning and evening shift, but there is an option to add a specific time in need. In case the employee cannot come that day to the shift, the manager can delete that employee from the shift and add someone else.

4.1.3 Assigning employees to stores

The database will be updated with the pieces of information of employees and their working location daily. Each employee is assigned to one store. There can be a situation when an employee changes between stores e.g., the new store is being established, and the manager moves some employees from one store to another.

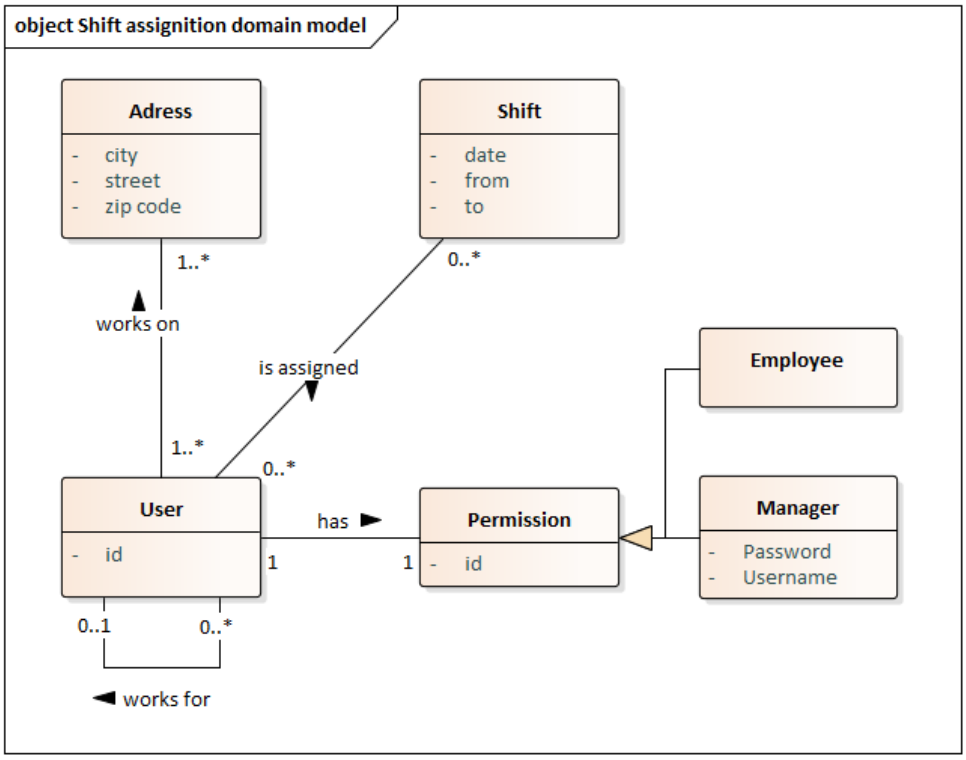

Figure 4.2: Shift assignition conceptual model of the task management system

Manager can be assigned to one or more stores; it depends on the manager position, which can be seen in Figure 1.1. In the case of regional managers, they are assigned to oversee at least three stores.

4.1.4 Statistics and reports

After the process of creation and assignation of tasks Figure 4.4, is completed a statistics is taken from every task. Statistics consists of pieces of information about employees and their activities during the completion of the task. Because all of the tasks are broken into the sub-tasks, as shown in Figure 4.4, they are informed about every employee assigned the to task.

These statistics can be seen only by the manager and depend on the manager's position in the Figure 1.1. The manager can see only the statistics of the stores and employees that are overseen by them.

Reports are generated from statistics at the request of the manager. The manager has to choose what statistics need to be shown and what kind of information. It depends on the manager to decide what is essential for them to see.

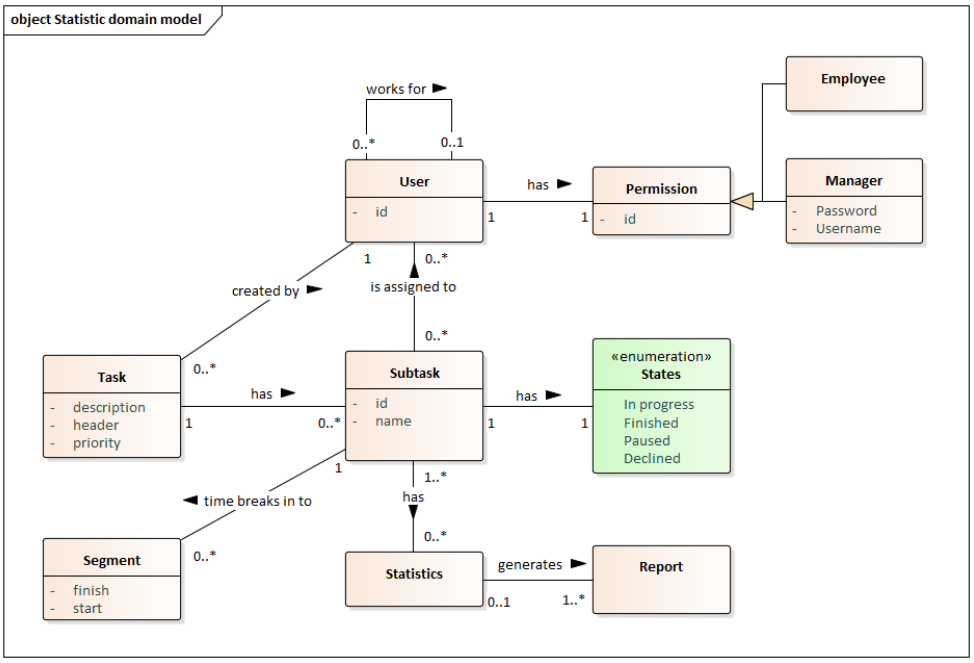

Figure 4.3: Statistics and reports conceptual model of the task management system

Reports for managers can differ by every store or level of hierarchy in Lidl. For example, the Regional manager wants to compare two stores in the time it takes to unload the specific shipment. However, the store manager needs to know how many tasks each employee completed each day throughout one week.

4.1.5 Assigning tasks

If the manager wants to create a task, they must be logged in to the application with manager mode turned ON. The manager selects from the pre-created tasks that are most common or can create a new one and save it to the common ones.

After picking or creating the task's name, the manager adds a description to the task and either assigns a task to the employees right away or leaves it for later. Tasks can be created and unassigned for a while; this gives managers the freedom to plan shifts.

4.1.6 Task activity

The employee gets the task assigned to them and has two options to react. One of the options is to decline the task. With this option, the employee has to comment on what was the reason for rejecting the task. If the employee chooses to start the task, they can work on it for a while and finish it with

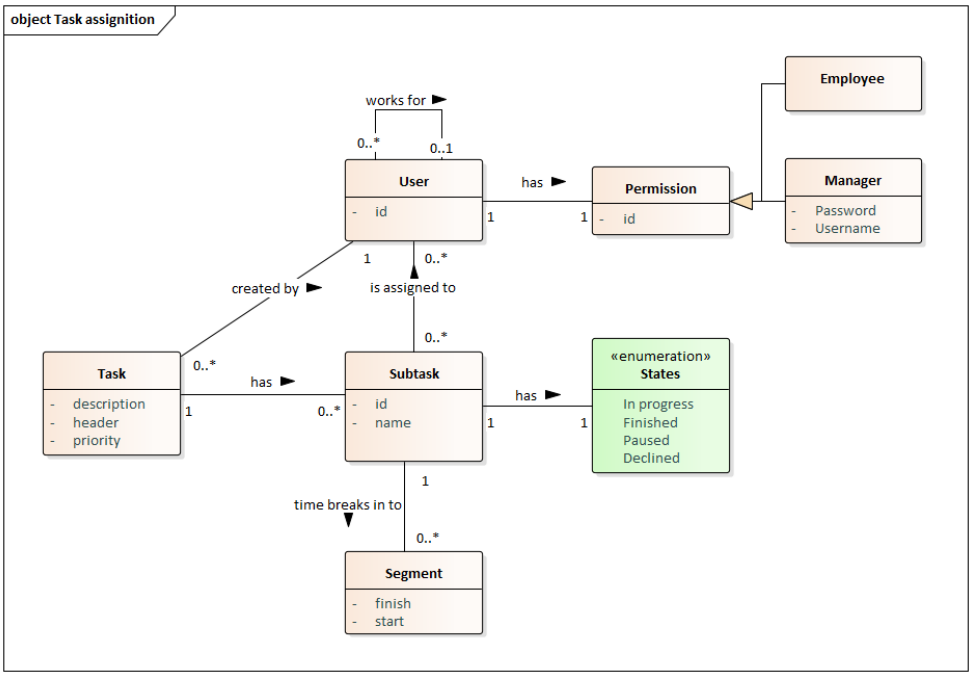

Figure 4.4: Task assignition conceptual model of the task management system

the comment if something went wrong or there is an option to pause started work.

After pausing the task, the employee can choose a different task, unpause this task, or finish the task if completed while they had it paused. After the task is being paused, it is broken into the time segments of start and finish time for each segment. When the task was started and paused, started and finished, or even paused and finished. The task can be assigned to more than one employee. In this case, sub-tasks are created to solve the problem if one of the employees finishes before the other.

With this solution, the task is not in the state of done until all of the employees assigned to this task finish. If an employee forgets to finish the task, it is on the manager to check why all of the participants did not finish the task.

4.1.7 Notifications

Notifications work to notify employees that something happened and they need to shift their attention from the current task to something else as can be seen in Figure 4.5.

Only manager can send notifications. The notification has a header, which is the name of the notification and description. The description contains vital information that the manager wants employees to know.

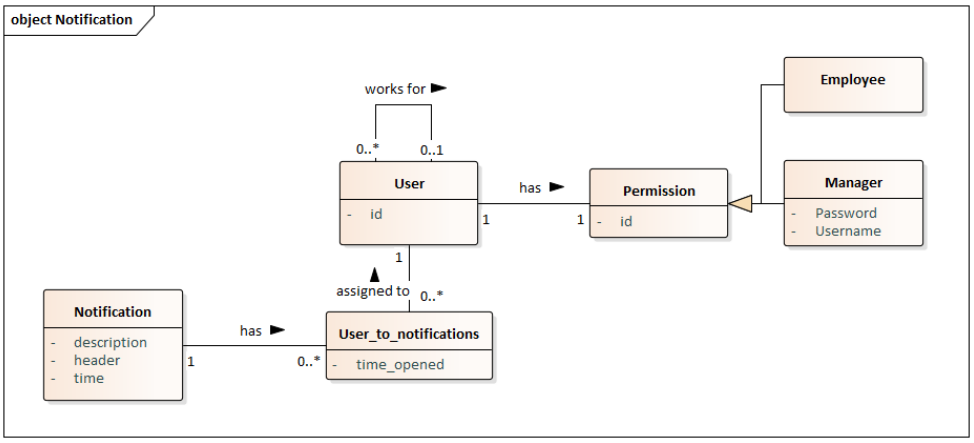

Figure 4.5: Notification conceptual model of the task management system

The manager can choose whom to send the notification to, and it can be from one employee to all of the employees on the shift. The manager can see who has already opened the notification and at the what time

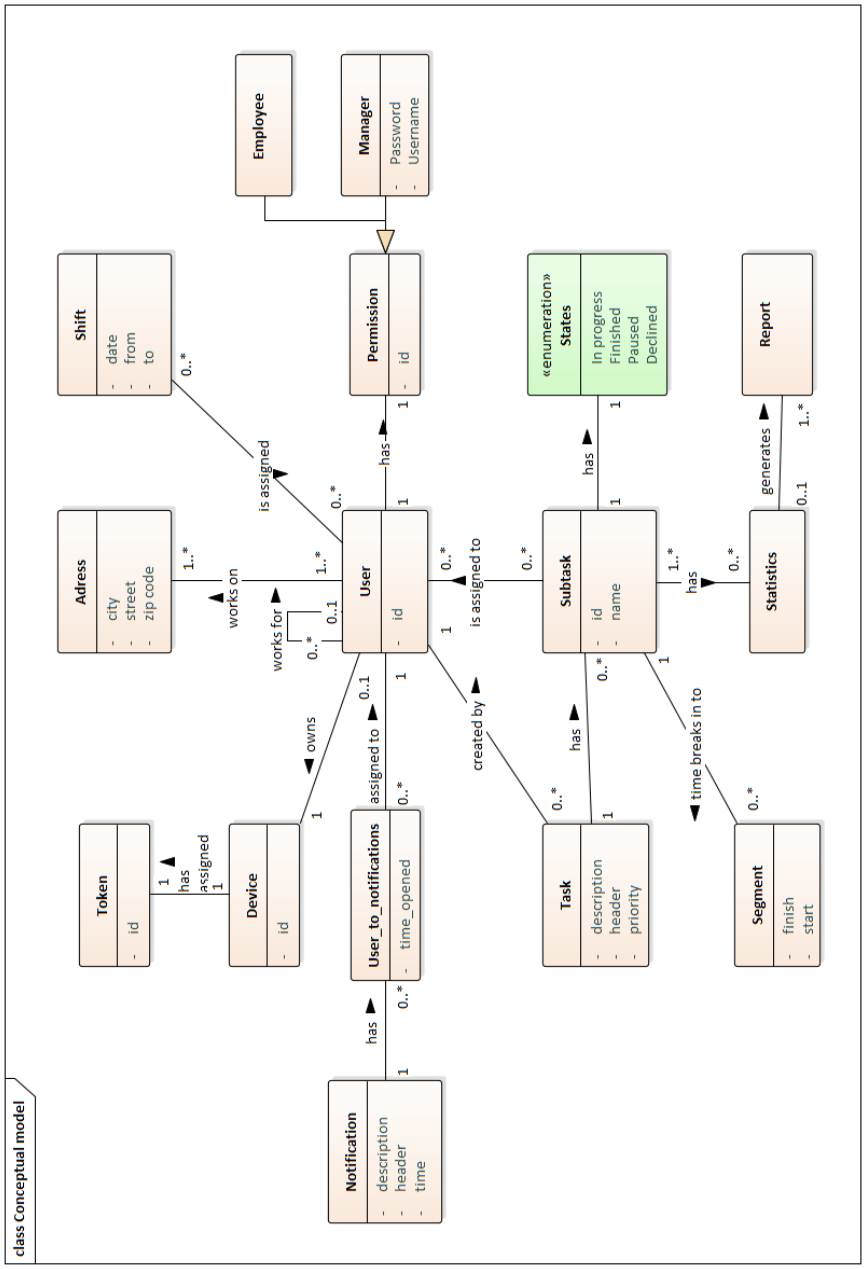

Figure 4.6: Conceptual model of the task management system

4.2 Use case model

The diagram in Figure 4.7 shows how the actor uses the system. However, only the names of activities are given and further details need to be specified: there is no other option than to fill in the text specifications of individual use cases.

From the functional requirements of the system, it is possible to break roles into two parts:

1. *Employee*
2. *Manager*

Manager can edit employees on the shift, create tasks, see statistics, generate reports. The employee can only see tasks that have been assigned to them by the manager. Even the employee is able to create simplified version of a task, which in contrast to the task created by manager contains only short description of work that was/will be done.

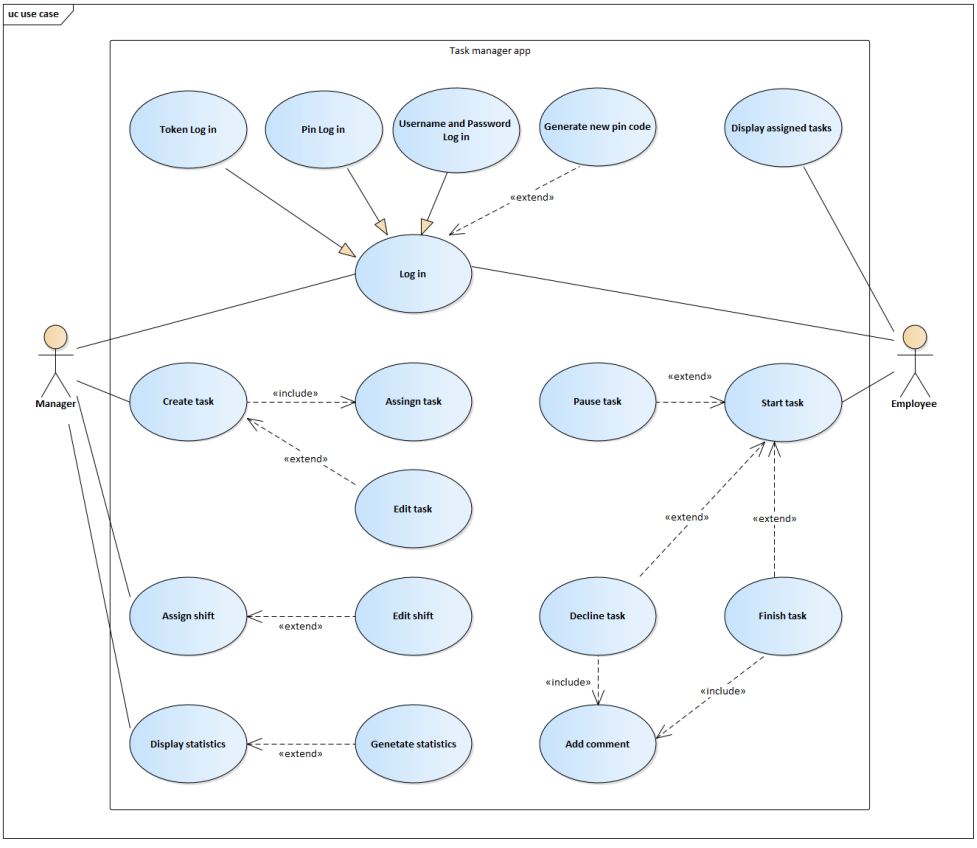

Figure 4.7: Use case diagram of the task management system

4.2.1 Use cases

The use case is an activity that is done by the user of the application. Use cases are sorted by function requirements defined in Chapter 2.

FR01 Division of permissions

- **UC00** *Employee* has permission to only see tasks assigned to them.

FR02 Log in to the application

- **UC01** *Employee* logs in to the application with token.
- **UC02** *Manager* logs in to the application with pin code.
- **UC03** *Manager* logs in to the application with username and password.
- **UC04** *Manager* logs in to the application in employee mode.

FR03 Task management

- **UC05** *Manager* creates task and assigns it to one employee.
- **UC06** *Manager* creates task and assigns it to multiple employees.
- **UC07** *Manager* deletes task.
- **UC08** *Manager* edits task.
- **UC09** *Employee* has tasks assigned.

FR04 Task workflow

- **UC10** *Employee* starts task and finishes it.
- **UC11** *Employee* starts task and pauses it to works on other task.
- **UC12** *Manager* creates task that is not assigned to any employee.
- **UC13** *Employee* declines task.
- **UC14** *Employee* unpauses task and finishes it.
- **UC15** *Employees* assigned to the same task finish it.

FR05 Creating statistics

- **UC16** *Manager sees statistics of workflow.*

FR06 Creating Reports

- **UC17** *Manager generates report from the statistics.*

FR07 Shift attendance

- **UC18** *Manager* assigns *Employee* to shifts.
- **UC19** *Manager* deletes *Employee* from shifts.

FR08 Device ID

- **UC20** *Manager* changes devices and generates new pin code.
- **UC21** *Employee* changes devices and generates new pin code.

4.3 Activity diagram

The activity diagram is one of the behavioral UML diagrams. Diagram illustrates the flow control in the system and the flow of the steps from the use case. Activity diagrams model sequential and concurrent activities. We can describe particular events and their flow from the start to the end with an activity diagram.

Many times the activity diagram is confused with the flow chart diagram. The flow chart is used to illustrate the workflow and express most business processes. With an activity diagram, it is easier to understand the workflow. [24]

4.3.1 Notifications

Notifications are a crucial part of the system; therefore, they help the manager inform employees in case of emergency or change of the tasks. As shown in the diagram Figure 4.8, the process of sending a notification is conditioned by the manager.

The manager is logged in to the application in manager mode and wants to create a new notification. The manager chooses to create a notification. Before the notification can be sent, manager needs to add a description and a header.

After the description is finished, the manager can add notification recipients to the next step. The manager can choose from all of the employees that are overseen by them. At the top of the list of employees are those who are currently assigned to the shift. When recipients are added, the manager can send the notification to the employees. The employee receives a notification and opens it.

4.3.2 Task

The manager creates a task with the header and description as can be seen in Figure 4.9. Assigning employees is similar to assigning notifications in Figure 4.8. After the task is assigned, the employee decides which of the assigned tasks has the highest priority. The employee can accept the task or decline it.

After acceptance of the task, the task is started, and the time counter starts. If the employee decides that there is another more important task, they can pause the current task and counter stops. When the task is paused, the employee can choose another task and start it.

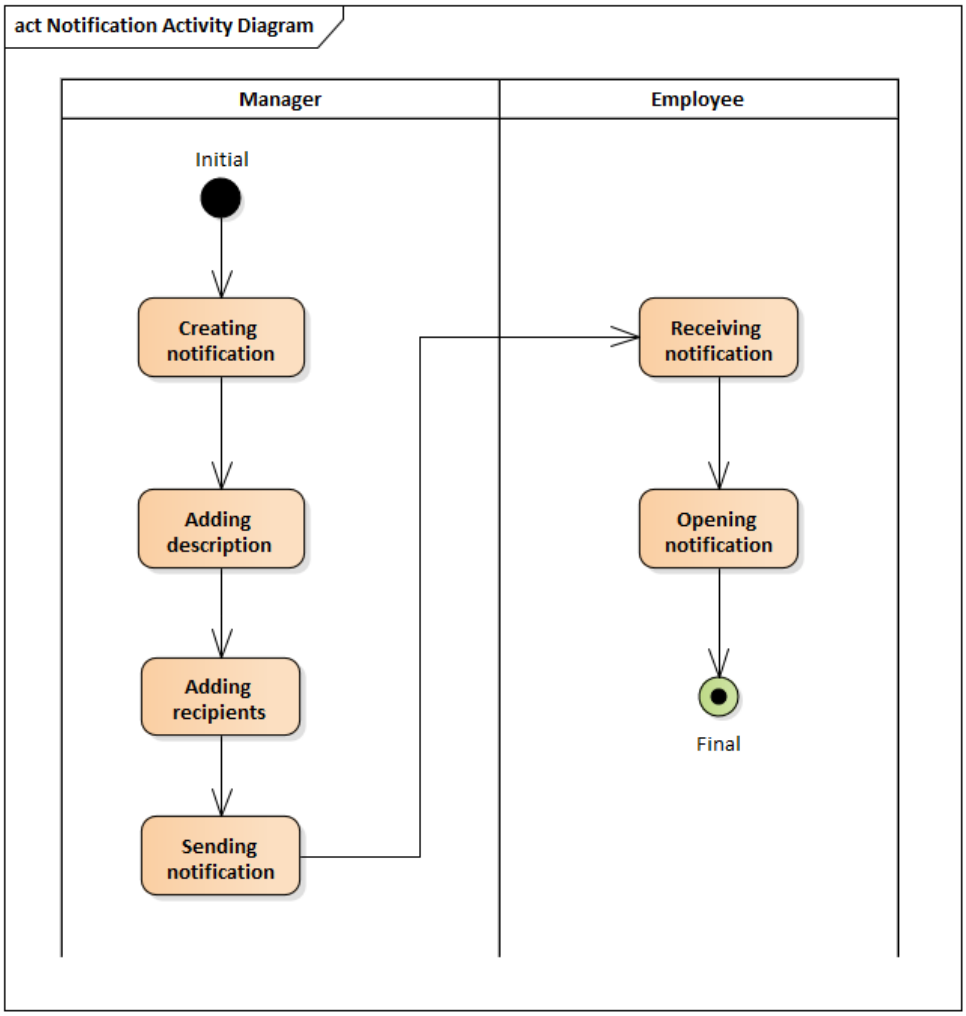

Figure 4.8: Notifications activity diagram

The employee can start the paused task again, and the timer starts again. After un-pausing the task employee goes through the process of completing the task or pausing it again.

If employees do not have any assigned tasks or are not assigned the task that is important to complete, it can be created by the employee. The manager only sees the employee's task, and it is like a notification for the manager.

After the employee decides to decline the task, the task is removed from the employee's active task list.

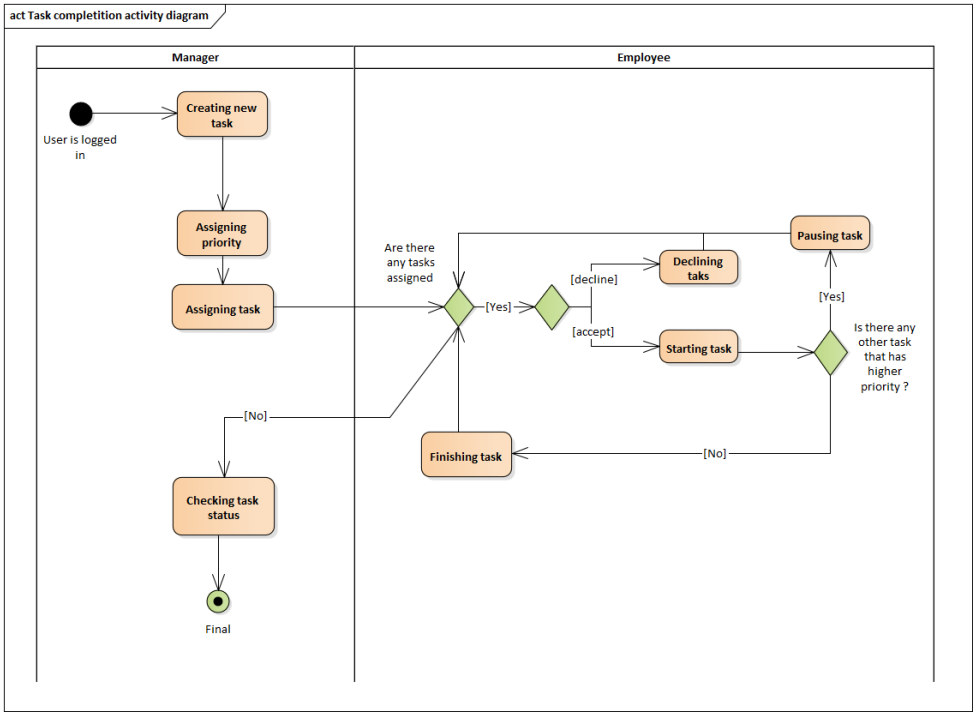

Figure 4.9: Task activity diagram

4.4 State diagram

State diagram represents an abstract description of the behavior of the system. The state machine expresses the states of a particular object and the transitions between these states. [25]

In the Figure 4.10, states are modeled showing a task in several stages, starting with the created task waiting to be assigned to the employees. The manager can assign one or more employees to the task.

After the task is assigned, the employee can choose to accept or decline the task. By declining the task, employee can not access the task anymore, and the process is finished. However, if the employee accepts the task, it changes state to 'in progress'. From there, employee has two options, to pause the task or to finish it.

Pausing the task will stop the progression of the task for the employee. The task will be waiting to be un-paused. When the task is restored, it goes again to the state of 'in progress'. From there, it can go again to a paused or finished state.

If the employee completes the task, it will go to the state of 'finished', which ends the task.

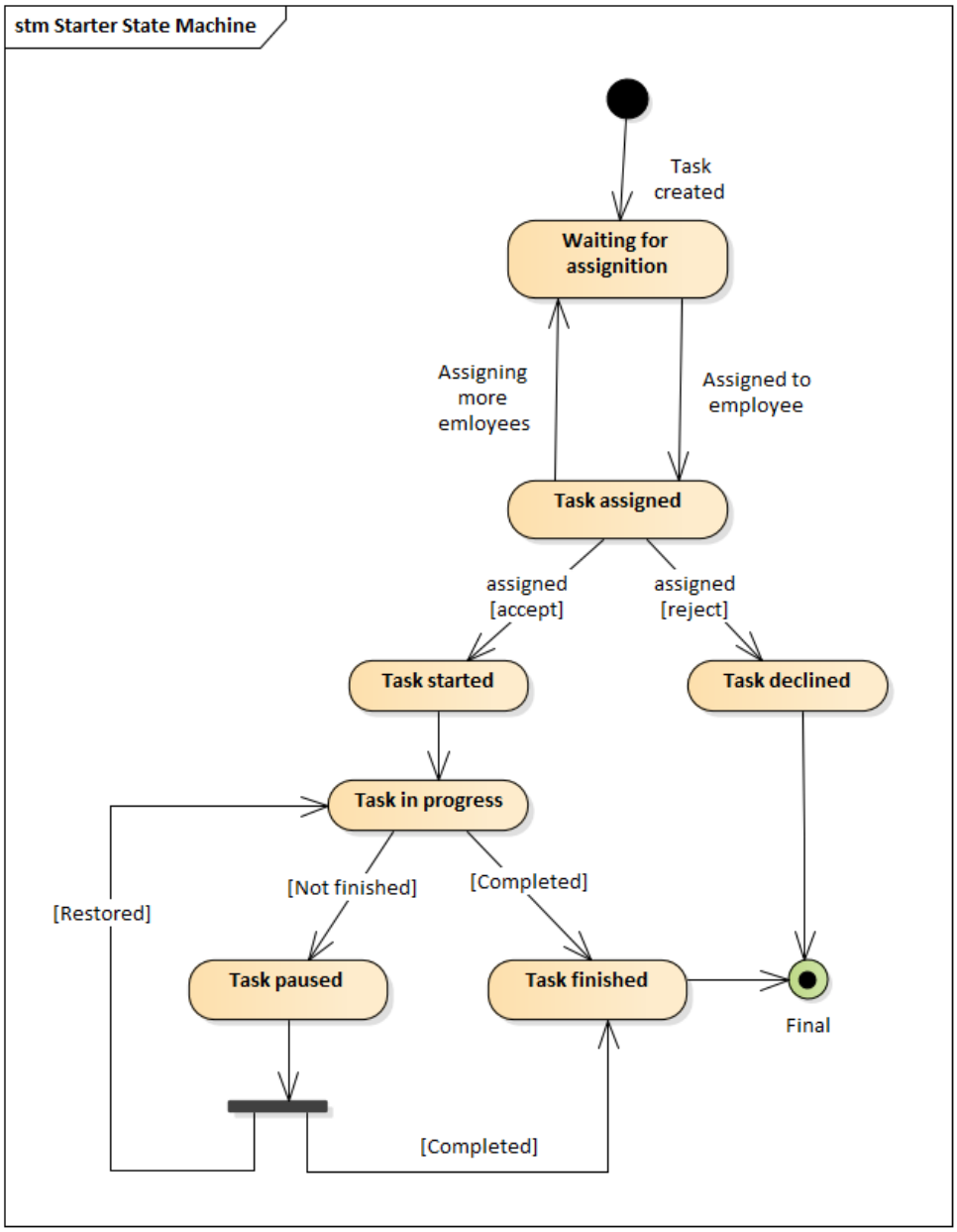

Figure 4.10: Task state diagram

Chapter 5

User interface design

'Wireframe is a two-dimensional skeletal outline of a webpage or app. Wireframes provide a clear overview of the page structure, layout, information architecture, user flow, functionality, and intended behaviors.' [26]

5.1 Ten Usability Heuristics

During the designing of the wireframes, I followed Ten Usability Heuristics by Jakob Nielsen, general principles for User Interface design. [27] [28] [29]

1.Visibility of System Status: The system should always keep users informed about what is going on, with appropriate visual feedback within a reasonable time. Feedback can be associated with some action and can also be provided by changing color, timer, or graphics. An example of this principle can be a loading bar or sound when the task is completed. This principle is used in Figure 5.9 with the tasks that have been finished or in Section 5.2.2 when the task is finished.

2.Match between system and the real world: The system should speak user's language in the mean of familiar concepts such as icons and phrases. An example of the practice is the icon of the shopping basket for online purchases. This principle can be seen in the application icons.

3.User Control and Freedom: The user has the freedom to navigate and perform actions, e.g. undo the previous accidental actions or step back in the process. This principle can be seen in the undo icon.

4.Consistency and Standards: This principle means using the same look across the application, for example, the same color of finish button in Figure 5.6.

5.Error Prevention: Preventing the user from doing the action that can create problems. An example of this principle is a confirmation before ending the task in Figure 5.7.

6.Recognition rather than recall: The system should minimize the memory load of pieces of information the user has to remember. The user should not have to remember information from one part of the dialogue to another. An example of this principle is in the Task completion process.

7.Flexibility and Efficiency of use: Accelerators can be unseen by novice users and can speed up the process for expert users. The system caters to both experienced and inexperienced users. An example of this principle is creating and saving tasks in Section 5.2.5. The user can save the task into the predefined tasks.

8.Aesthetic and minimalist design: The user interface design should focus on essentials, information visible for the user should be relevant. This requirement should minimize the ratio of relevant to irrelevant information in the user interface.

9.Help users recognize, diagnose, and recover from errors: The error message should be expressed in plain language and suggest the possible solution. The principle should help the user understand the problem. An example of this is an error while logging in to the application on Figure 5.12.

10.Help and Documentation: The design's goal is intuitive enough for the user to navigate without the use of documentation. In the case of user difficulties, there should be user documentation to provide additional help for the use.

5.2 Mobile UI design

5.2.1 Log in page

The idea is to create as simple UI design as possible for users to understand. Employees can log in to the application with a pin code generated by the back office device. Figure 5.1

Manager can log with username and password which can be seen in Figure 5.2.

Log in page will be same for all types of devices.

5.2.2 Dashboard

After a successful login, an employee sees the 'Task dashboard', Figure 5.3, where are all of the tasks assigned to them. The red task represents a higher priority task.

Every task has a name, description, and time when it was assigned. The description of the task is shortened and can be seen fully after tapping or clicking on the task.

The yellow task represents a paused task. After clicking on the task, the task acceptance page as seen on Figure 5.4, will show up to continue the task.

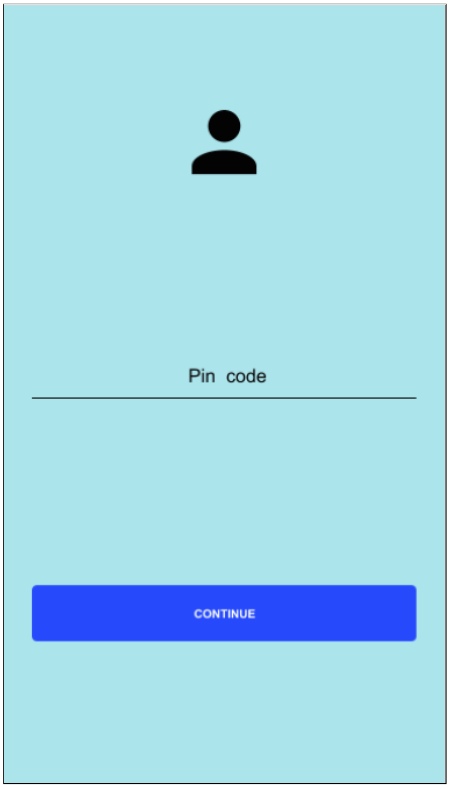

Figure 5.1: Employee log in page

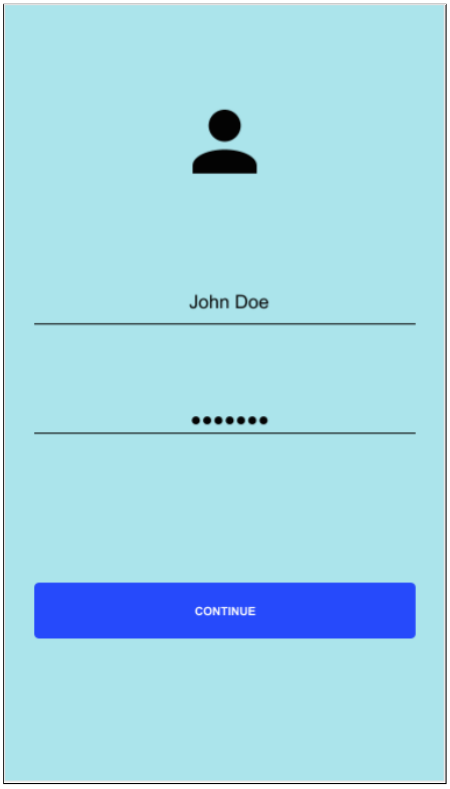

Figure 5.2: Manager log in

5.2.3 Task completion process

The employee can choose to accept or decline the assigned task, Figure 5.4. By declining the task employee is transferred to Figure 5.7.

Accepting the task will take the employee to the Figure 5.5, the progress page. On the progress page employees can choose to pause or finish the task. By pausing the task, employee will be transferred to task dashboard, Figure 5.3.

If the employee decides to finish the task, the application will transfer them to the finishing page, Figure 5.6.

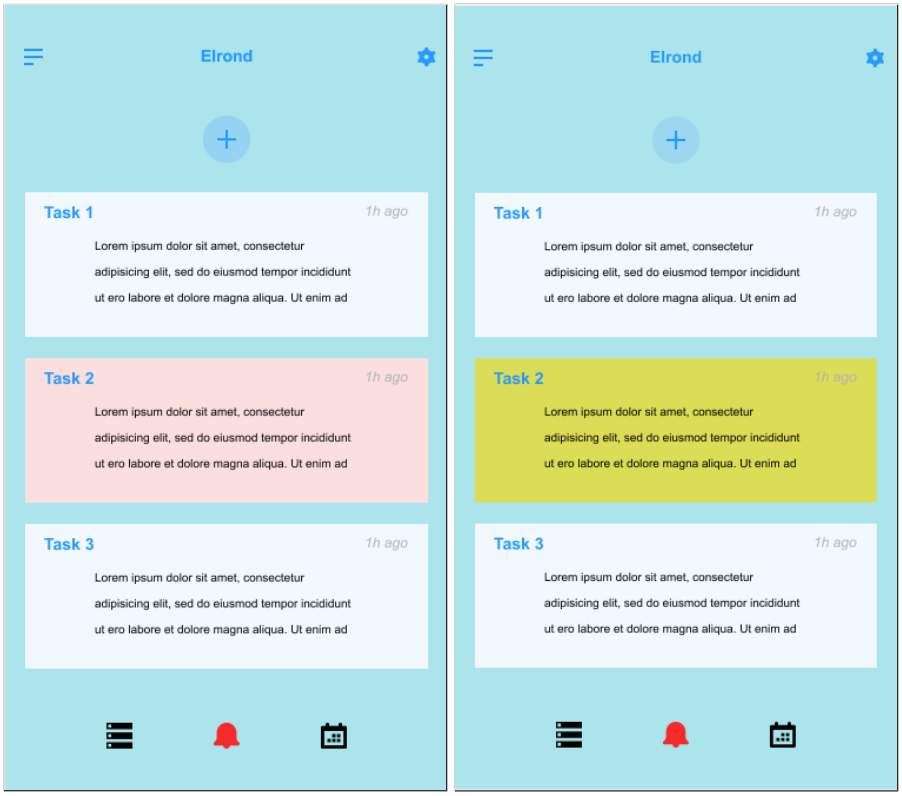

Figure 5.3: Dashboard priority and paused task

5.2.4 Finishing and declining task

On the finishing page employees can add a comment on the task. The comment should describe if any problem occurred during the completion of the task and in the case of declining the task why the employee decided to do so in Figure 5.7.

If an employee finished the task and there was some issue during the task completion, Figure 5.6, an employee can leave comment.

Figure 5.4: Task acceptance page

Figure 5.5: Task progress page

5.2.5 Tasks creation

Managers can create the task by going to the task creation page and pick from pre-created tasks or create a new one. When creating a new task, the manager needs to add a heading and description. After doing so, manager is transferred to the next page and assign employees to the task.

5.3 Web UI design

The design will be similar to the Section 5.3.1 with the same functionalities as the mobile application.

5.3.1 Statistics

Manager can look at statistics of the employees or store and can choose what statistics they wants to see. E.g., total tasks assigned and completed.

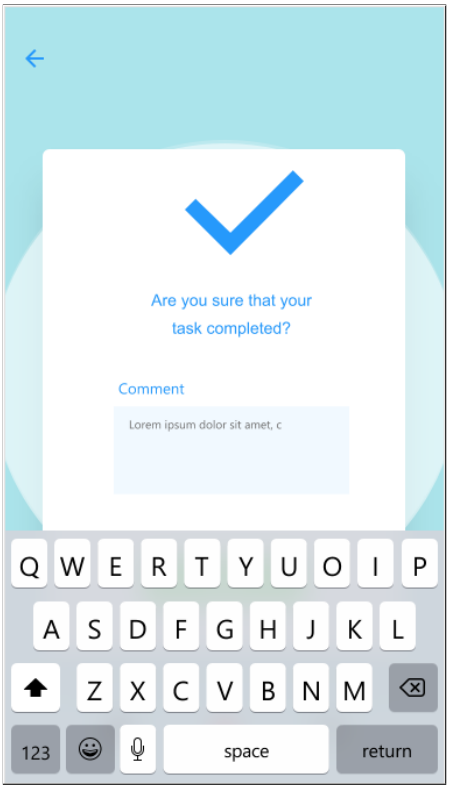

Figure 5.6: Task acceptance page

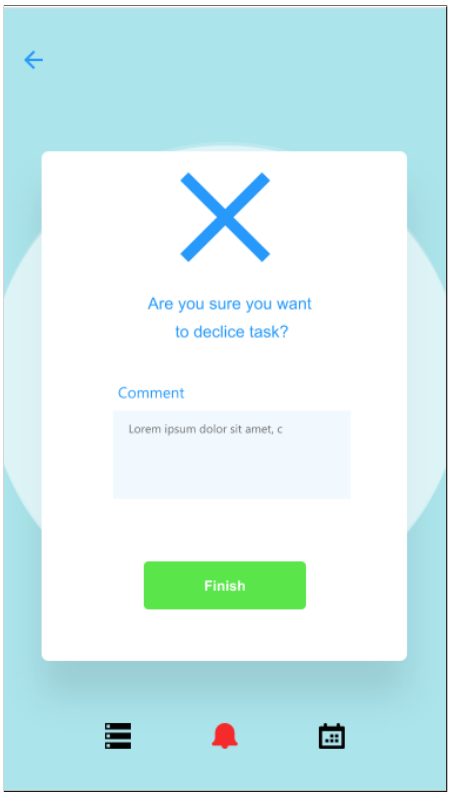

Figure 5.7: Task decline page

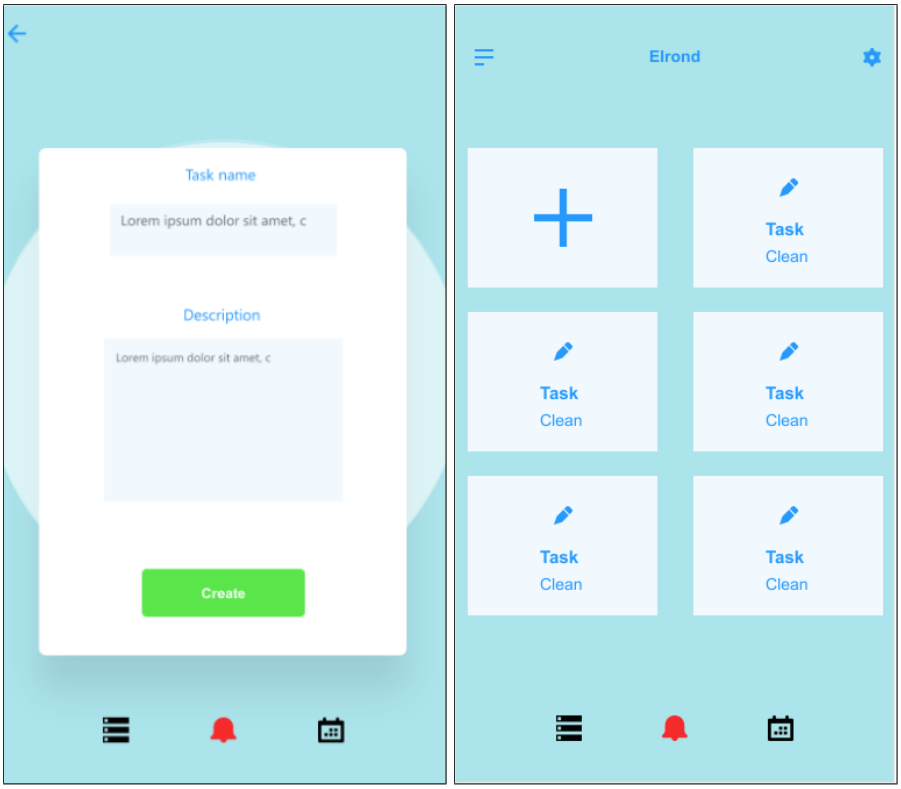

Figure 5.8: Task creation

Figure 5.9: Task history

Figure 5.10: Navigation bar

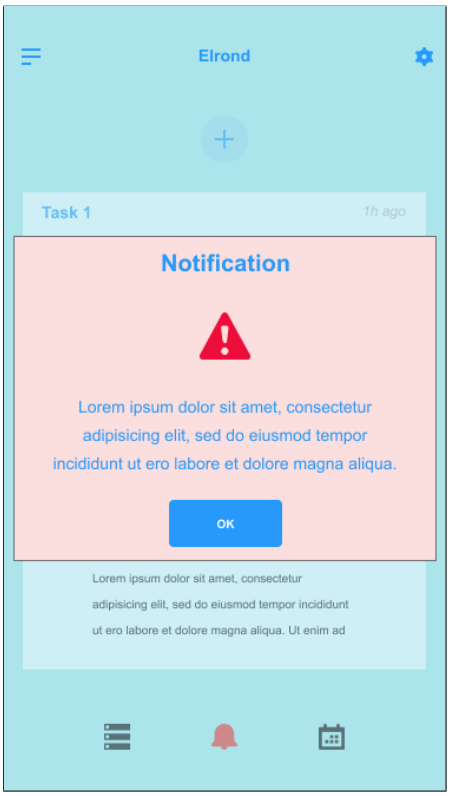

Figure 5.11: Notifications

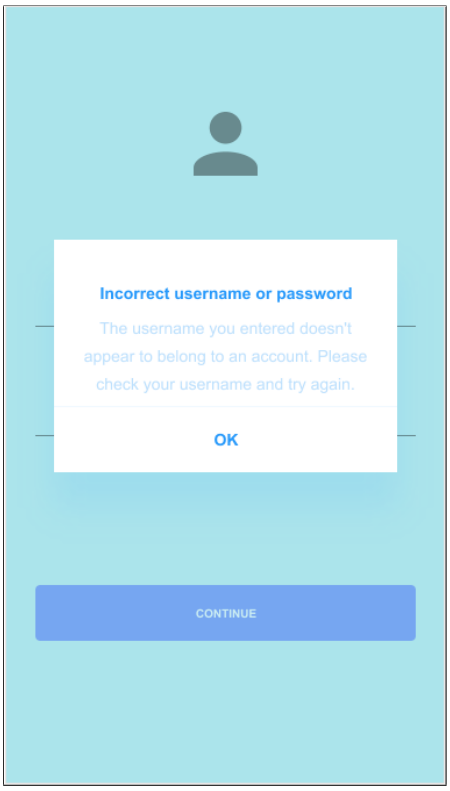

Figure 5.12: Log in error

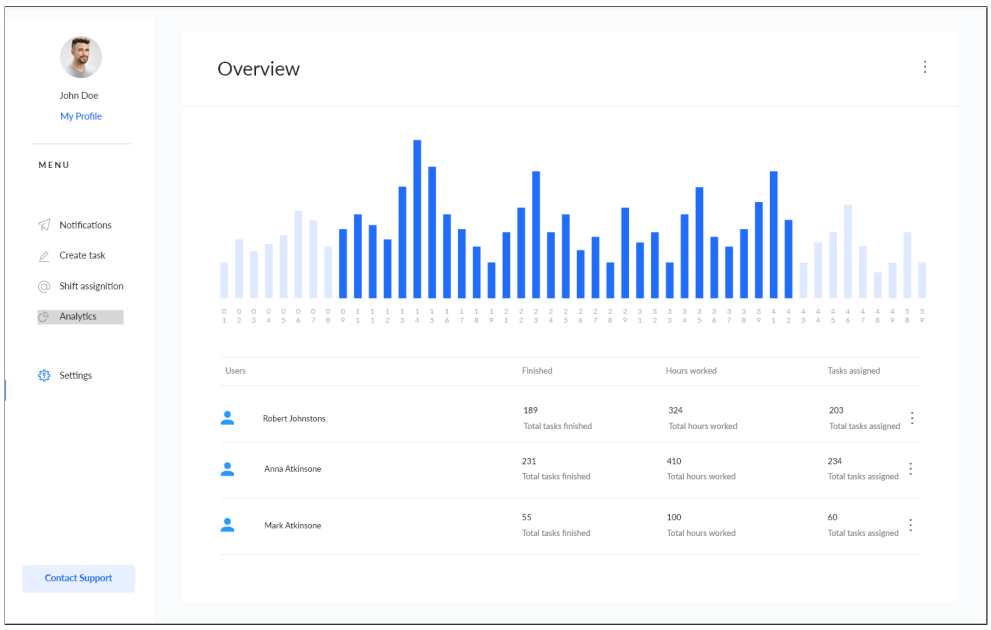

Figure 5.13: Web task statistic

Chapter 6

Design Evaluation

6.1 Benefits

Application is designed to help the management of the chain store to work more efficiently and keep track of the workflow of the employees in the stores. Manager can easily assign tasks and shifts to employees to significantly shorten the time of current solution where the manager walks to the employee throughout the store to assign the task in person. As mentioned earlier, Lidl is pursuing the no-frills and zero-waste approach, the application should help follow this goals. The statistic will help create clear view of the workflow and workload of the employees during the shifts.

Using the **PDCA** cycle method helped me improving the architecture and design of the application. This method consists of four steps:
Plan: This is the first step of the cycle for the identification and the analysis of the problem.
Do: Developing and implementing the solution.
Check: Evaluation of the results and checking if the desired goal was achieved.
Act: This step is for the standardization of the solution. After this step the cycle repeats. [30]

6.2 Architecture

The software architecture and design plays huge role in the application development process. The architecture was designed to be able to meet the requirements of the Lidl and the possible future application scaling. Example of this can be creating and managing permissions.

The design is ready for the new positions that will require specific combinations of permissions. [31]

6.3 User interface

The User Interface was tested by the use cases from Chapter 2 and using Jakob Nielsen Usability Heuristics evaluation mentioned in Chapter 5. Thanks to this feedback it was possible to design and create better wireframes.

6.4 Possible future steps

Future development could be focused on expanding functionality of the application. One of those possible extensions could be calendar, where employees would be able to request vacations or days off. Another possibility is release of the system in other countries. Of course, after deployment and user testing, emergence of new requirements is also possible.

Conclusion

All of the requirements set for this thesis were successfully met. As a part of the thesis, the analysis of the current solutions in a chain store and the main problems during the shifts were described. From this analysis and specific requirements set from the company's management, the functional and non-functional requirements were created. These requirements also described the business processes, system behavior, and critical performance parameters. The requirements were prioritized using MoSCoW classification.

After careful consideration, the UML was used to create models and diagrams. Using the information gathered in analysis and requirements specification, the conceptual model was created to capture the principles and relationships among the application. Use case model was created to explain how the actors use the system. To better describe particular events, flows, and processes in the system, the activity, and state diagrams were created.

The user interfaces for the web and mobile applications were created. For the wireframes and prototype, I used Adobe XD. During the creation of the wireframes, I followed the Jakob Nielsen Usability Heuristics evaluation methods. The benefits of the system and the design were evaluated, and the possible future steps were described.

Bibliography

[1] 2020, [Cited 2020-07-12]. Available from: https://www.atlassian.com/software/jira

[2] 2020, [Cited 2020-07-12]. Available from: https://trello.com/

[3] Fisher, J.; Koning, D.; et al. Utilizing Atlassian JIRA for large-scale software development management. Technical report, Citeseer, 2013.

[4] Fylaktopoulos, G.; Goumas, G.; et al. An overview of platforms for cloud based development. *SpringerPlus*, volume 5, no. 1, 2016, doi:10.1186/s40064-016-1688-5.

[5] Kaur, A. App Review: Trello. *Journal of Hospital Librarianship*, volume 18, no. 1, 2018: p. 95–101, doi:10.1080/15323269.2018.1400840.

[6] Hull, M.; Taylor, P.; et al. Software development processes — an assessment. *Information and Software Technology*, volume 44, no. 1, 2002: pp. 1–12, ISSN 0950-5849, doi:https://doi.org/10.1016/S0950-5849(01)00158-6. Available from: https://www.sciencedirect.com/science/article/pii/S0950584901001586

[7] Lidl – Sweden's first zero-carbon building. Available from: https://linkarkitektur.com/en/Projects/Lidl-Sweden-s-first-zero-carbon-building

[8] Lidl. Protecting the Environment. 2020, [Cited 2020-30-12]. Available from: https://www.abettertomorrow-lidl.ie/environment/

[9] Sloma, M. Carbon footprint of electronic devices. In *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, *Society of*

Photo-Optical Instrumentation Engineers (SPIE) Conference Series, volume 8902, July 2013, p. 890225, doi:10.1117/12.2030271.

[10] Wohlin, C.; et al. *Engineering and managing software requirements*. Springer Science & Business Media, 2005.

[11] Chung, L.; Nixon, B. A.; et al. *Non-functional requirements in software engineering*, volume 5. Springer Science & Business Media, 2012.

[12] Berander, P.; Andrews, A. Requirements prioritization. In *Engineering and managing software requirements*, Springer, 2005, pp. 69–94.

[13] A., K.; SAVAGE, W. J. *Object-oriented design with UML and Java*. New Delhi: Elsevier, 2012, 3-10 pp.

[14] Available from: `https://sparxsystems.com/`

[15] BEST 28 UML Tools in 2021. Available from: `https://www.guru99.com/best-uml-tools.html`

[16] Khaled, L. A Comparison between UML Tools. In *2009 Second International Conference on Environmental and Computer Science*, 2009, pp. 111–114, doi:10.1109/ICECS.2009.38.

[17] Security-first diagramming for teams. Available from: `https://www.diagrams.net/`

[18] Adobe XD Tool. Available from: `https://www.adobe.com/products/xd/solutions/prototyping-tool.html`

[19] Axure RP 9 - Prototypes, Specifications, and Diagrams in One Tool. Available from: `https://www.axure.com/`

[20] Axure RP vs. Adobe Xd. Jul 2020. Available from: `https://humix.be/blog/axure-rp-vs-adobe-xd/`

[21] Axure vs Adobe XD 2021 - Feature and Pricing Comparison on Capterra. Available from: `https://www.capterra.com/prototyping-software/compare/168423-18297/Adobe-XD-vs-Axure-RP`

[22] Adobe XD vs Axure RP. Available from: `https://www.trustradius.com/compare-products/adobe-xd-vs-axure-rp`

[23] Fowler, M. *UML distilled: a brief guide to the standard object modeling language*. Addison-Wesley Professional, 2004.

[24] Jain, A. Unified Modeling Language (UML) — Activity Diagrams [online]. February 2018, [Cited 2020-07-12]. Available from: https://www.geeksforgeeks.org/unified-modeling-language-uml-activity-diagrams/?ref=rp

[25] A., K.; SAVAGE, W. J. *Object-oriented design with UML and Java*. New Delhi: Elsevier, 2012.

[26] Hannah, J. What Exactly Is Wireframing? A Comprehensive Guide [online]. AUGUST 2019, [Cited 2020-02-12]. Available from: https://careerfoundry.com/en/blog/ux-design/what-is-a-wireframe-guide/

[27] Nielsen, J. Ten usability heuristics. 2005.

[28] Nielsen, J. How to conduct a heuristic evaluation. *Nielsen Norman Group*, volume 1, 1995: pp. 1–8.

[29] Jain, A. 10 Heuristic Principles – Jakob Nielsen's (Usability Heuristics). Available from: https://www.uxness.in/2015/02/10-heuristic-principles-jakob-nielsens.html

[30] Johnson, C. N. The benefits fo PDCA. *Quality Progress*, volume 35, no. 5, 2002: p. 120.

[31] Vishnyakov, A.; Orlov, S. Software Architecture and Detailed Design Evaluation. *Procedia Computer Science*, volume 43, 2015: pp. 41–52, ISSN 1877-0509, doi:https://doi.org/10.1016/j.procs.2014.12.007, iCTE in Regional Development, December 2014, Valmiera, Latvia. Available from: https://www.sciencedirect.com/science/article/pii/S1877050914015750

Appendix A

Acronyms

GDPR General Data Protection Regulation

GUI graphical user interface

ID Identification Data

IT Information Technology

UI user interface

UID unique identifier

UML Unified Modeling Language

VPN virtual private network

Appendix B

Contents of enclosed CD

- `readme.txt` ..... the file with CD contents description
- `text` ..... the thesis text director
  - `thesis.pdf` ..... the thesis text in PDF format
  - `source` ..... LaTeX source codes of the thesis
- `models` ..... UML models in PNG format directory
  - `png` ..... UML models directory
  - `models.ea` ..... UML models in Enterprise Architect format
- `UI` ..... the directory with wireframes
  - `wireframes.pdf` ..... PDF format of wireframes
  - `png` ..... directory with PNG format of wireframes