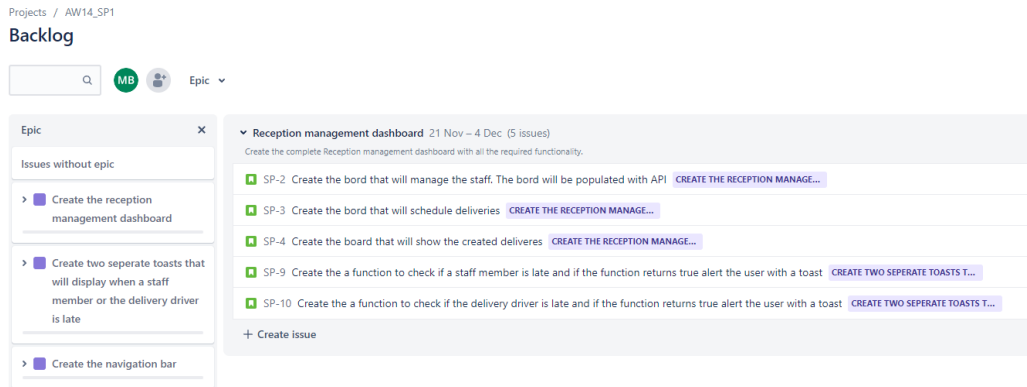


# Reflection Report

## Planning the project

The first sprint created was the “Reception management dashboard” which would contain all the tasks of creating the main part of the project with the most important features.



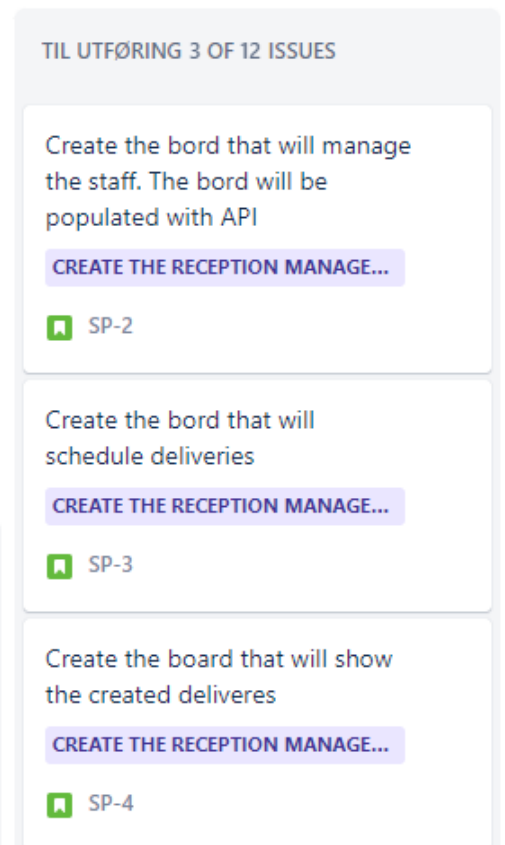
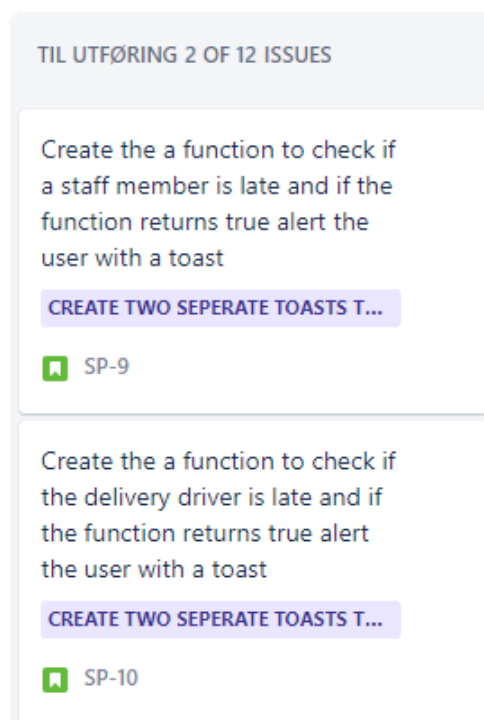
The reason for choosing these tasks first is because the dashboard is by far the most important part of the app and if for some reason the project does not get finished at least the main functions of the app is in place and still usable to some degree.

First task in the sprint would be the staff board where we will take the make the API requests that initialize when the app is launched.

The bord titles, row/columns and buttons are not specified since I will just use the pdf file to refer to. Specifying that I need to apply the functionalities to the button and placement would be make the task to much broken down and make it messier in my opinion.

Second and third tasks are pretty much the same. Both will have boards with titles, row/columns and buttons which I can just look at the pdf for reference.

The toasts which are less important but still relevant to the dashboard Was put last in the sprint.





The second sprint created was the “Nav-bar, styling, readme” which would contain all the tasks of creating the content that that was less important like the styling which is important in a way but does not perform the functions which the client must be able to do at all costs.

The navigation bar was not considered important because of app just having one page. The timer was not considered important either because you could still perform all the necessary tasks in the app.


Projects / AW14\_SP1


## Backlog


  Epic ▾


Epic ×

Issues without epic

>  Create the reception management dashboard

>  Create two separate toasts that will display when a staff member or the delivery driver is late


>  Create the navigation bar


>  Apply all styling and correct visual content


Reception management dashboard 21 Nov – 4 Dec (5 issues)


Nav-bar, styling, readme 4 Dec – 18 Dec (7 issues)


Create the navigation bar and dynamic calendar/clock. Apply all the correct styling to the app. Add the logo and adjust all images to fit. Finally update the readme file to include instruction how to install the finished app.


 SP-5 Create the options in the navigation bar [CREATE THE NAVIGATION BAR](#)


 SP-16 Make sub-options in nav bar hidden when not interacted with [CREATE THE NAVIGATION BAR](#)

 SP-17 Clicking on the option called dashboard will navigate to the main Reception Management Dashboard page [CREATE THE NAVIGATION BAR](#)

 SP-6 Create the dynamic calendar/clock

 SP-7 Apply the correct styling to the boards buttons and navigation bar [APPLY ALL STYLING AND CORREC...](#)

 SP-8 Add the logo and adjust all images to fit [APPLY ALL STYLING AND CORREC...](#)

 SP-11 Update the read me file to include instructions on how to install the app

First in the list is the navigation bar. This one taken before the timer since the navigation bar barely requires functionality to it.

Then we can create the dynamic calendar/clock so that we can finish all the functionalities and make sure everything works.

After that we can finally apply all the styling to the app, the colours, font styles and animations.

At the end we write the instructions and do a final test of the app.

TIL UTFØRING 3 OF 12 ISSUES

Create the options in the navigation bar

CREATE THE NAVIGATION BAR

SP-5

Make sub-options in nav bar hidden when not interacted with

CREATE THE NAVIGATION BAR

SP-16

Clicking on the option called dashboard will navigate to the main Reception Management Dashboard page

CREATE THE NAVIGATION BAR

SP-17

Create the dynamic calender/clock

SP-6

Update the read me file to include instructions on how to install the app

SP-11

TIL UTFØRING 2 OF 12 ISSUES

Apply the correct styling to the bords buttons and navigation bar

APPLY ALL STYLING AND CORREC...

SP-7

Add the logo and adjust all images to fit

APPLY ALL STYLING AND CORREC...

SP-8

Then after having both sprints ready and the dates and roadmap setup, the planning was completed.

Projects / AW14\_SP1

Roadmap

MB

Status category

Epic

	T	NOV	DEC
Sprints			Reception mana... Nav-bac styling, rea...
SP-13 Create the reception managemen...			
SP-14 Create two separate toasts that w...			
SP-15 Create the navigation bar			
SP-18 Apply all styling and correct visua...			
+ Create Epic			

Projects / AW14\_SP1

Backlog

MB

Epic

Epic

Issues without epic

Create the reception management dashboard

Create two separate toasts that will display when a staff member or the delivery driver is late

Create the navigation bar

Apply all styling and correct visual content

+ Create Epic

Reception management dashboard 21 Nov - 4 Dec (5 Issues)

Create the complete reception management dashboard with all the required functionality

SP-2 Create the board that will manage the staff. The board will be populated with API

SP-3 Create the board that will schedule deliveries

SP-4 Create the board that will show the created deliveries

SP-9 Create the a function to check if a staff member is late and if the function returns true alert the user with a toast

SP-10 Create the a function to check if the delivery driver is late and if the function returns true alert the user with a toast

+ Create Issue

Nav-bac styling, readme 4 Dec - 18 Dec (7 Issues)

Create the navigation bar and dynamic calender/clock. Apply all the correct styling to the app. Add the logo and adjust all images to fit. Finally update the readme file to include instruction how to install the final

SP-5 Create the options in the navigation bar

SP-16 Make sub-options in nav bar hidden when not interacted with

SP-17 Clicking on the option called dashboard will navigate to the main Reception Management Dashboard page

SP-6 Create the dynamic calender/clock

SP-7 Apply the correct styling to the bords buttons and navigation bar

SP-8 Add the logo and adjust all images to fit

SP-11 Update the read me file to include instructions on how to install the app

## **Sprint 1 - Reception management report**

I started off with the first sprint which I deemed to most important one. Getting the all the functionalities work and tested. If I were to stop after finishing this sprint I at least would have a






demo to show.

Dashboard Inventory Orders

Search Search

Add Add

Remove Remove

Picture	Name	Surname	Email address	Status	Out Time	Duration	Expected	Return Time
	پوریسا	پوریسا	pwry.prs@example.com					
	Manuel Pulido		manuel.pulido@example.com					
	Carmen Gutiérrez		carmen.gutierrez@example.com					
	June Barnett		june.barnett@example.com					
	Paula Santos		paula.santos@example.com					

In

Out

Vehicle:	Name	Surname	Telephone	Delivery Address	Return time
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Add

Vehicle: Name Surname Telephone Delivery Address Return time

Clear

13 12 2022  
15:0:44

## Sprint 1 - ISSUES

I first got the **staff board** populated with the class objects. I ensured that the board would be cleared every time we would populate the table, this would make it so that we don't accidentally override data.

Then I made it so the **delivery board** that would be populated with the class objects which we would create with the **schedule delivery board**. I added some basic conditions to the schedule delivery board input fields like the length of what a telephone number could be. I ensured that the minimal input validation was met, all fields are filled in and the time format is correct.

Then I added the functionality to select objects and change them. With the **staff table** I made it so you can the status from in to out with a duration of how long they should be out. With the **delivery board** I added a clear button that after selecting one of the delivery boards object would be deleted with a confirmation by the user.

Then finally I added the **toasts** which with a timer (the time selected from the user converted into milliseconds) that once ended would check if the current object is still relevant and decide wither to display a toast.


One thing that was a bit challenging was when changing or deleting a row how do you know what object to also change or delete?

Initially I thought I could give the rows and objects id's that would be the current date down to the milliseconds but I don't know if that would be allowed with how the diagram was set up for the class objects. So instead, I matched the email address since email addresses should never be allowed to be changed and they are always unique.






When it came down the delivery board there is no email. So I checked and matched each and every field to get a match with the object. Then I can delete the object and the row.

## Sprint 2 – Nav-bar, styling, readme

Dashboard Inventory Orders

Reception Management Dashboard

Staff

Picture	Name	Surname	Email address	Status	Out Time	Duration	Expected Return Time
	Reinhild	Maurer	reinhild.maurer@example.com	In			
	Selma	Nielsen	selma.nielsen@example.com	In			
	Scarlett	Robinson	scarlett.robinson@example.com	In			
	Patrick	Walker	patrick.walker@example.com	In			
	Umang	Ramesh	umang.ramesh@example.com	In			

Out

In

Schedule Delivery

Vehicle:	Name	Surname	Telephone	Delivery Address	Return time
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Add

Delivery Board

Vehicle:	Name	Surname	Telephone	Delivery Address	Return time
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Clear

16 12 2022 19:49:17

## Sprint 2 - ISSUES

I first started with centring everything. Then I added the forms like the rounded edges on the boards after that I started with the colouring and font styles.

Last I ensured that the plain text like the headlines or the digital clock was in the right font and font colour.

Having completed the styling I started on the “readme” file which I wanted to have three main subjects Title of project, location of code files, libraries used and how to install and run the app.

## Quality insurance

Then after finishing the project to the standard requested, I then went through the check list and made sure that each step was met and if I could change some of the methods used in the script to more elegant methods and clean up the code. I had quality insurance for each issue created in the sprint, but I wanted to make a final quality insurance with everything put in place at once.