

Requirements and Analysis Document for Yellow

Joakim Agnemyr, Edwin Eliasson, Mona Kilsgård and Viktor Valadi.

TDA 367

2018-10-28

This version overrides all previous versions.

Table of Contents

Table of Contents	1
1. Introduction	2
1.1 Purpose of application	2
1.2 General characteristics of application	2
1.3 Scope of application	2
1.4 Objectives and success criteria of the project	2
1.5 Definitions, acronyms, and abbreviations	3
2. Requirements	3
2.1 User stories	3
2.2 User interface	9
2.2.1 Design structure	9
2.2.2 Design and graphic details	9
2.2.3 Layout and flow	9
3. Domain model	19
3.1 Class responsibilities	19

1. Introduction

1.1 Purpose of application

The aim of this project is to create an inventory application where different groups of people can lend out and keep track of the items in their common inventory. The application will allow the user to know to who, when and during what period the item(s) were lent.

1.2 General characteristics of application

The application will be a desktop application runnable on Mac/Windows/Linux platforms.

The application's purpose is to give members of different societies the possibility to keep track of their inventories and lend out items in these inventories. The users should be able to rent out anything they want by creating their own items. The user has the possibility to give the item a name, a description, a certain amount and add an image. When a customer wants to rent an item the user will be able to find the item in their inventory and rent it out. When an item is available for renting an order is created to give the user an overview of the renting, and input the renters personal info. When the order is created all users who shares the inventory can see the order and the items will be marked as rented during the picked date.

1.3 Scope of application

The application will not support rentals made by private individuals. The rental procedure can only be done by people associated with the group that owns the specific inventory. The renter therefore does not need to create an account and the entire lease is handled by the members of the group. However, this could be a development area for the application. By allowing users to complete their booking themselves, they reduce the workload for the group. But at the present time, the group gets a good opportunity to keep track of their inventory without incorrect and unwanted rentals made by renters.

1.4 Objectives and success criteria of the project

1. It should be possible to create a account.
2. It should be possible to create a group.
3. It should be possible to add items to an inventory.
4. It should be possible to view all the items in the inventory.
5. Users should be able to join different groups by entering a code that is generated when a user creates a group.
6. It should be possible to be part of multiple groups at the same time.
7. The users should be able to see what they have in "stock" and what is rented.
8. It should be possible to edit user information in the application.
9. It should be possible to create orders where all the items that the renter wants to borrow are gathered.
10. It should be possible to add who rents the items while placing an order.

11. It should be possible to view when a item is rented.

1.5 Definitions, acronyms, and abbreviations

- **Committee** - A group of people.
- **Yellow** - The name of this application
- **Student division** - Referring to a group of students with a common inventory.
- **App** - Referring to an application usable on desktops.
- **Item** - Specific object/objects
- **Inventory** - A collection of items.
- **Renter** - The person who lend items from the group who owns the inventory to which the object belongs.
- **Group** - A group in the app users can be a part of and where they can share inventories.
- **User** - The person using the application
- **User Story** - Describes a feature that a user or customer of the system desires.

2. Requirements

2.1 User stories

Listed below are the different user stories for Yellow. All user stories are identifiable by a unique story id and with an explanatory story name. If all the requirements are true the user story is fulfilled.

1. **Story Identifier:** U101

Story Name: Creating a inventory

Description: As a division member I want to be able to create a inventory where I can place items.

Confirmation

- It should be possible to create an inventory with a name.

Functional

- Can I create an inventory?
- Can I add a name for my inventory?

Non-functional

- Is it possible to create an inventory when not in a group?

2. **Story Identifier:** U102

Story Name: See an inventory

Description: As a division member I want to see an inventory with my items because I want to know what items I have.

Confirmation

- It should be possible to see a list of items from a specific inventory.
- It should be possible to see a list of all items in a group with two or more inventories.

Functional

- Can the person see all items from their inventories?
- Can the person see the items for a specific inventory?

3. Story Identifier: U103

Story Name: Create items

Description: As a division member I want to be able to create items while adding my collection of items.

Confirmation

- It should be possible to create an item in my inventory.
- It should be possible to add a name, a short description for the item and enter an amount.
- It should be possible to add a image for the specific item.

Functional

- Can I create an item when in a group?
- Can I create an item with selected inventory?
- Can I create an item with all parameters filled?

Non-functional

- Is it possible to create an item when not in a group?
- Is it possible to create an item if no inventory is selected ("All items" counts as an inventory).

4. Story Identifier: U104

Story Name: Expanding inventory

Description: As a division member I want to be able to expand my inventory because I might want to add items in the future.

Confirmation

- It should be possible to add an item into a selected inventory (already created inventory).
- When creating an item it should be possible to choose which inventory it belongs to.

Functional

- Can I add an item when in a group?
- Can I add an item with a selected inventory?
- Can I add an item with all parameters filled?

Non-functional

- Is it possible to add an item if not in a group?
- Is it possible to add an item if no inventory is selected?

5. **Story Identifier:** U105

Story Name: Several inventories

Description: As a division member I want to create different inventories because I might have items in different places.

Confirmation

- It should be possible to add multiple inventories.

Functional

- Is it possible to create an inventory when in a group?
- Is it possible to have more than one inventory?
- Can I name my inventory?
- Does the inventory belong to a group?

6. **Story Identifier:** U106

Story Name: View inventories

Description: As a division member I want to see items that belongs to a specific inventory or view all my items at the same time.

Confirmation

- It should be possible to see inventories one at the time.
- It should be possible to see all items in a group at the same time.

Functional

- Is it possible to switch between inventories?
- Is it possible to view all items in a group?

7. **Story Identifier:** U107

Story Name: Gathered inventories

Description: As a division member I want to be able to gather my inventories into groups.

Confirmation

- The division member should be able to create a group.
- The created group should have a name and a color.
- The user should be able to choose which group the inventory should belong to while creating the inventory.

Functional

- Can I see the inventories that are in the selected group?
- Can I see the items that are in the selected inventory?

8. **Story Identifier:** U108

Story Name: Sharing inventories

Description: As a division member I want to be able to share my inventory with others because there might be several people sharing an inventory.

Confirmation

- The user is able to invite other users to a group that it is a member of.
- The user should be able to view invite codes to all the groups he/she is a member of.

Functional

- Can I view the invite codes to my groups?

9. Story Identifier: U109

Story Name: Joining groups

Description: As a division member I want to be able join an already existing group to be a part of that inventory.

Confirmation

- The user should be able to join a group by being invited by a member of that specific group.
- It should be possible to insert the invite code given by another user.
- It should be possible to view all inventories and items specified in the specific group.
- The group should be listed with the users other groups.
- The division member should be able to view the invite code to the specific group after joining it.

Functional

- Can I join a group?
- Can I see the inventories that are in the joined group?
- Can I see the items that are in the selected inventory?
- Can I view the invite code to the group?
- Can I still see my other groups after joining another group?

Non-functional

- Can you join a group without the invite code?

10. Story identifier: U110

Story Name: Keeping groups and inventories private

Description: As a group member I don't want non-group members to see the groups inventory because I want to keep it private within my group.

Confirmation:

- The group member only has access to their own groups and inventories.

Functional

- Can I only see the inventories that I am a part of?
- Can only the people in the group see that specific groups invite code?

11. **Story Identifier:** U111

Story Name: Renting items

Description: As a division member I want to rent out item(s) because I want to meet the customers needs.

Confirmation:

- It should be possible to add items to an order.
- It should be possible to delete items from the order before it is confirmed.
- The order should include information about the renter.
- The order should have a starting date and a return date.

Functional:

- Can I create an order with multiple items?
- Can I create an order with information about the renter?

12. **Story Identifier:** U112

Story Name: Item renting status

Description: As a division member I want to see what items are rented because I want to know if they are available for renting.

Confirmation

- It should be possible to add a renting time while creating a order.
- It should be possible to view when a item is rented in the item calendar.

Functional

- Does every item have a calendar with that specific items renting status?
- Is the status updated while placing a order?

13. **Story Identifier:** U113

Story Name: Sign in/sign up

Description: As a section member I want to have a account where I can find all my inventories.

Confirmation:

- It should be possible to create a account.
- It should possible to choose a name, username, email and password while creating an account.
- It should possible to sign in by inserting chosen username and password.
- It should possible to log out and sign in with the created account.
- The user should be able to add groups, inventories and items while signed in.

Functional:

- Is the account saved after it is created?
- Is the user obligated to insert their username and password?
- Can there be multiple accounts?
- Can only people with a account sign in?

14. **Story Identifier:** U114

Story Name: User Settings

Description: As a section member I want to change my user information because I might want a new username or password.

Confirmation:

- It should be possible for the user to change their username, name, email and password.

Functional:

- Can I save the settings for the next time I login?

Non-functional:

- Can the user change username to any other users username or password?

2.2 User interface

In general, Yellow has two main purposes. It should be possible to view and edit an inventory, and it should be possible to lend out items and know who has them and when. This notification has provided the basis for how Yellow's user interface is designed.

2.2.1 Design structure

The application is mainly structured into two different sections with similar layouts, but with discernible differences. The different sections' purposes is to indicate to the user which functions are available. In the first section the user will be able to view and edit groups, inventories and items while the other section handles the rental part of the application. This breakdown is made because a user is not believed to be interested in adding and changing items at the same time as placing an order for a renter.

2.2.2 Design and graphic details

The overall color scheme for Yellow is a darker background where important functionality is enhanced with a yellow color. Yellow also uses two different background images depending on whether the user is logged in or not. This is to clarify whether the user is in the application and can start using it or not. The background image also has a bit brighter color when the user is signed in. This to indicate even more that the user has finished the first step and now can start using the application.

Most labels are written in a white or light grey color to pop out from the background. Therefore labels and texts placed on a brighter background is displayed in a black or dark grey color.

Yellow also uses colour pickers to make it easier for users to distinguish their groups in the list. In addition to this, the application keeps a fairly neutral and easy-to-understand color use where, for example error messages and lent items are highlighted red, and available items green.

Icons are used by Yellow to clarify some features of the application and also to reduce the usage of texts which in large quantities can make the interface look messy.

2.2.3 Layout and flow

When starting up Yellow the user will get two options, either sign in or sign up (*Figure 1: Welcome screen*). These two options are displayed on the same screen to make it easy for the users to know what is expected of them. If the user chose to sign up, a new, but similar page (*Figure 2: SignUp screen*) will be displayed where the user can create an account by typing in relevant information.

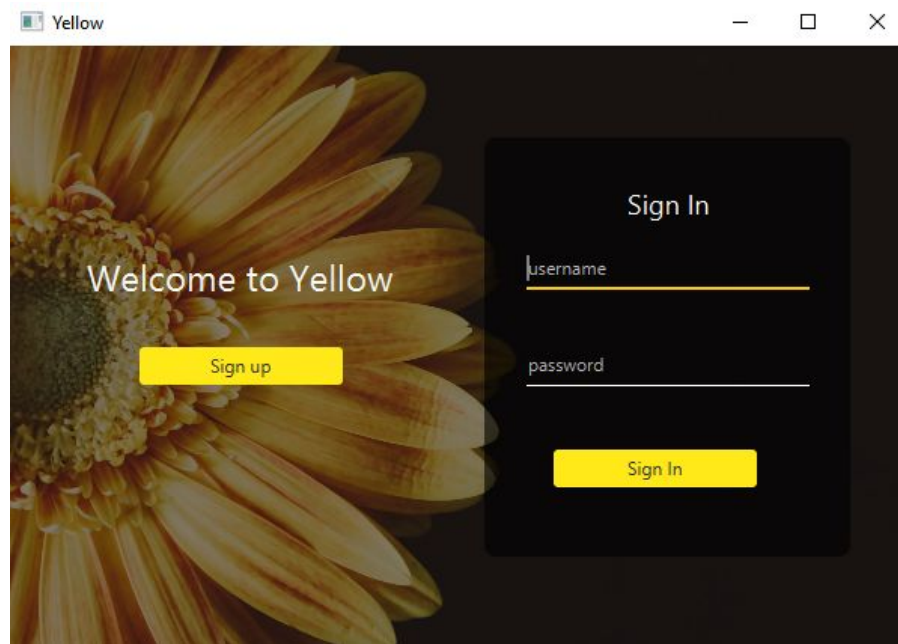


Figure 1 : Welcome screen

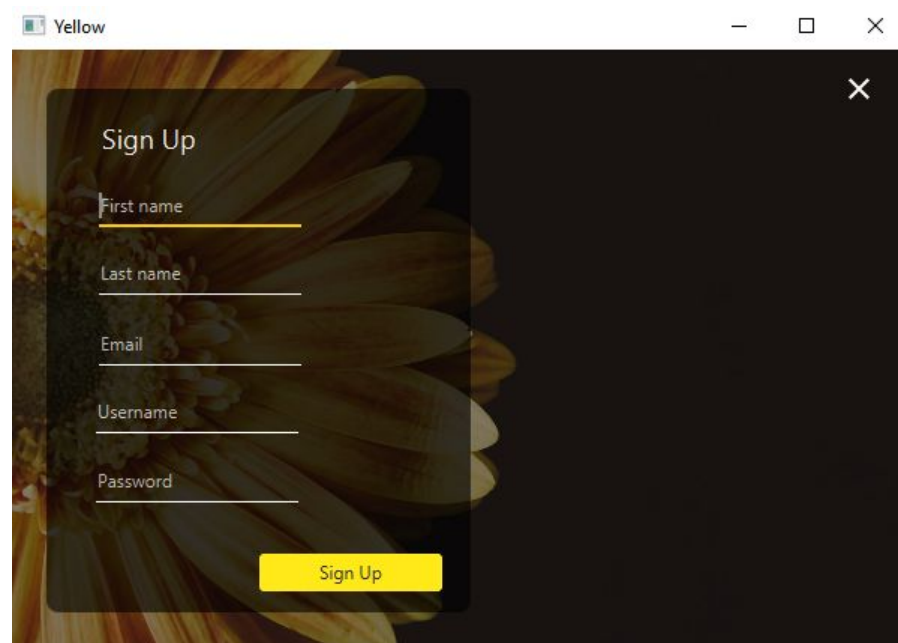


Figure 2: SignUp screen

After the user finished any of these two steps, a new screen with four new options will be displayed (*Figure 3: Yellow screen*). These options are believed to be relevant while first going into the application. By restricting the choices here, Yellow makes it easier for the user to find what they are looking for in the application.

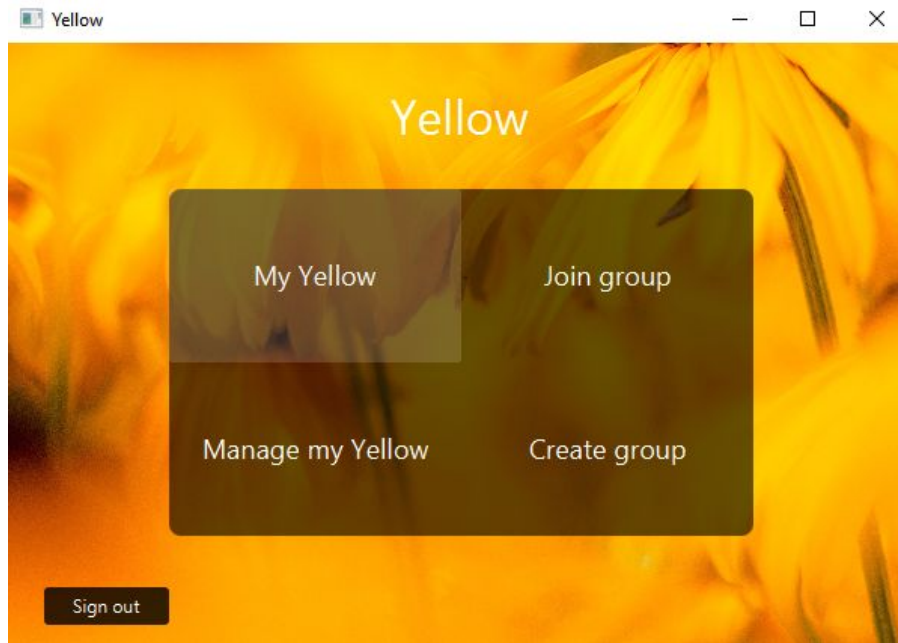


Figure 3: Yellow screen

In most cases the user have been using Yellow before and are already a part of a group. But if that is not the case, the options to either create a group or join a group is available here. After doing any of these two, the user will be taken to the “Manage my Yellow” page (*Figure 3: Manage my Yellow*). This is where you edit existing groups, inventories and items, create new ones or go directly to “My Yellow” (*Figure 9: My Yellow*), which is the part/section of the application where it is possible to lend out items.

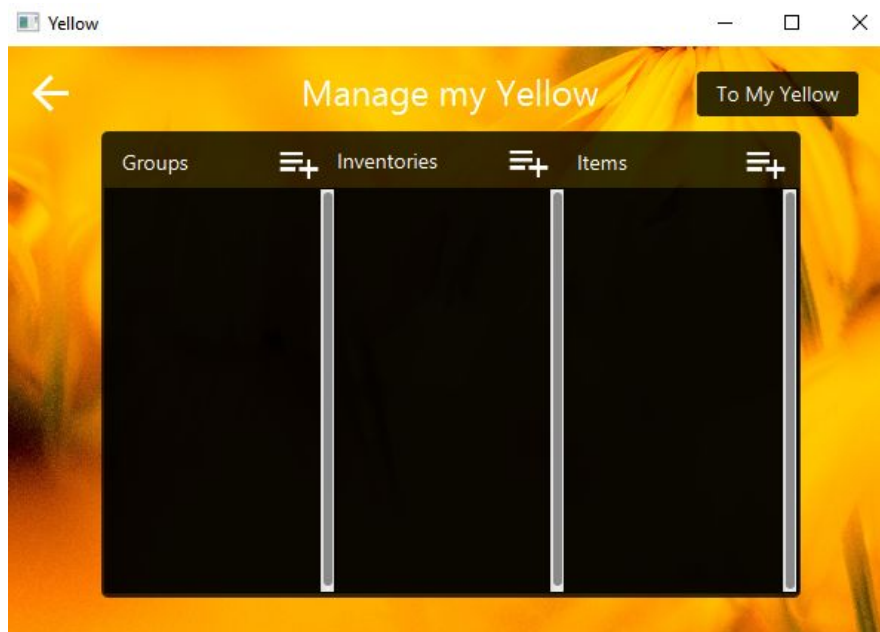


Figure 4: Manage my Yellow

If you are a previous user of Yellow and are familiar with the structure of the application, you probably already know if you want to edit your inventory or lend out items when signing in. Therefore

it is possible for the user to go directly from “Yellow screen” into “My Yellow” or “Manage my Yellow” after signing in (*Figure 3: Yellow screen*).

“Manage my Yellow” (*Figure 4: Manage my Yellow*) is divided into three different flowpanes/lists where groups, inventories and items are listed. The lists is structured in a way that the different inventories is shown based on which group is chosen and items based on chosen inventory. By doing this the user can easily see which inventories and items that belong to which group (*Figure 5: Manage my Yellow with groups, an inventory and a item*).

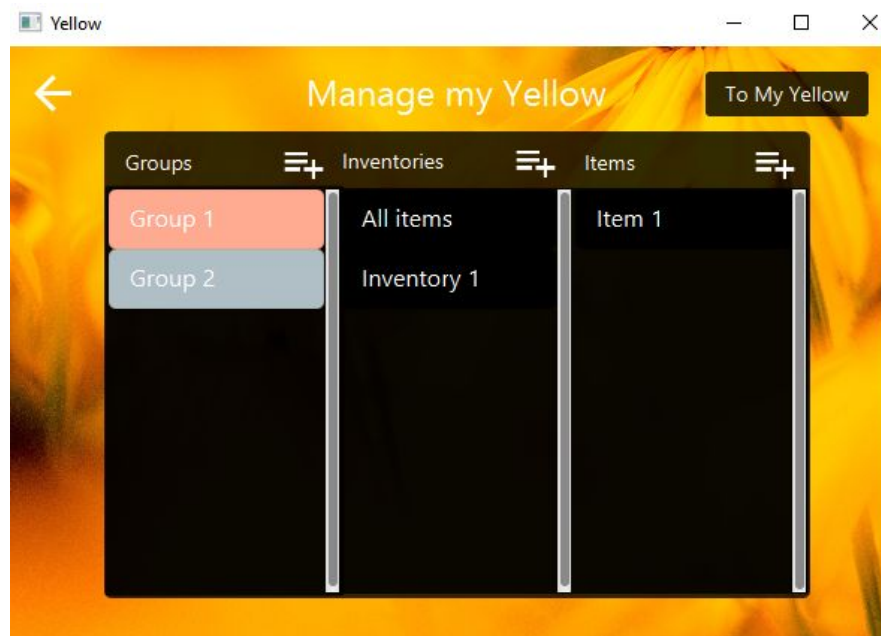


Figure 5: Manage my Yellow with groups, an inventory and a item.

In the “Manage my Yellow” window the user can also add new groups, inventories and items. This is easily done by clicking on the adding symbol placed next to the associated label (*Figure 5: Manage my Yellow with groups, an inventory and a item*). By this action a new dialog is displayed where the user can insert relevant information. All the dialog screens in Yellow has the same layout which hopefully results in the user recognize himself making it easier to navigate around the application (*Figure 6, Figure 7 & Figure 8*).

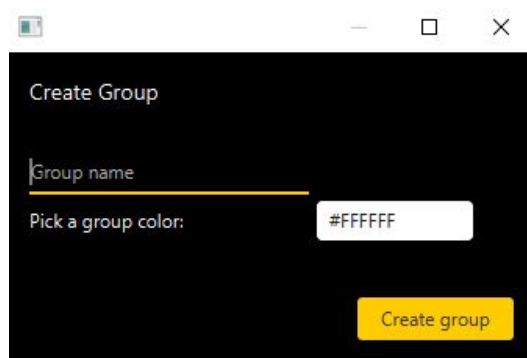


Figure 6: Create group

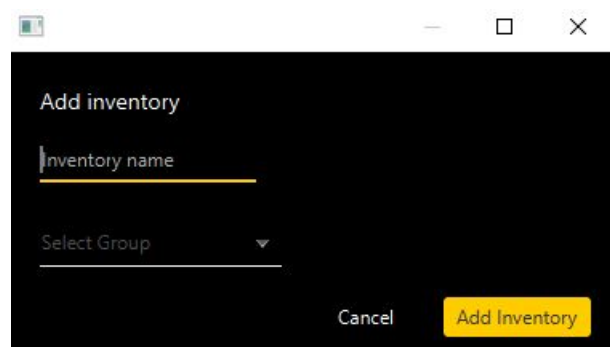


Figure 7: Add inventory

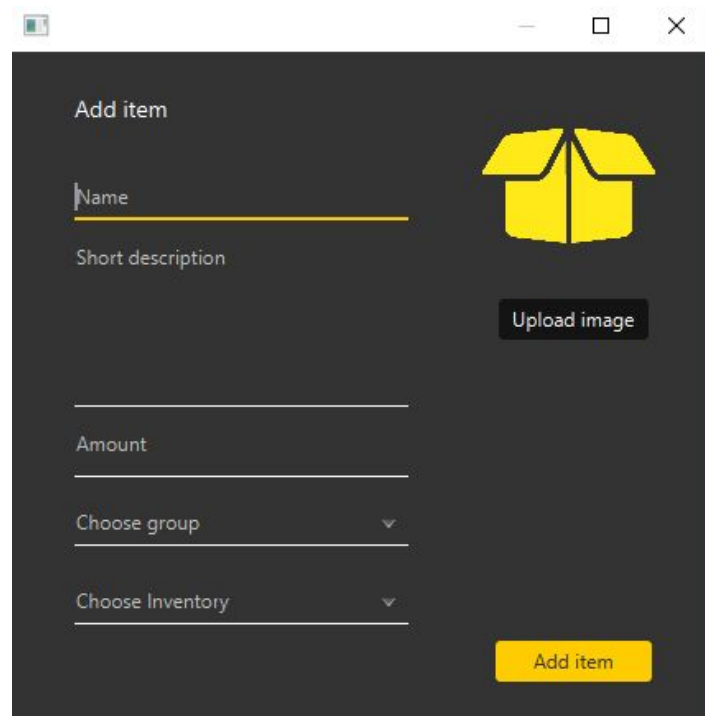


Figure 8: Add item

By choosing a group (by clicking on it) in “Manage my Yellow” an “edit” panel is revealed that takes the user to a almost identical dialog screen as the one that is shown while adding a new group. However, the differences in this view is that the user has the option to change already given information or deleting the object. This structure applies to both groups, inventories and items.

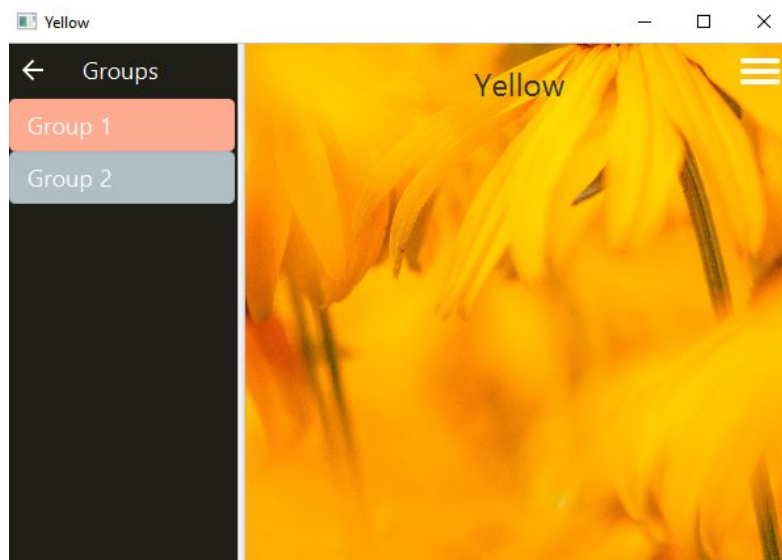


Figure 9: My Yellow

“My Yellow” (*Figure 9*) is the view where it is possible to create orders and view which items that are available. It is structured into two main sections. One list view and one bigger view where things such as items and orders are displayed. The list view has the same structure as “Manage my Yellow” and

by clicking on an available group, that groups inventories will be displayed while its items are shown in the main window.

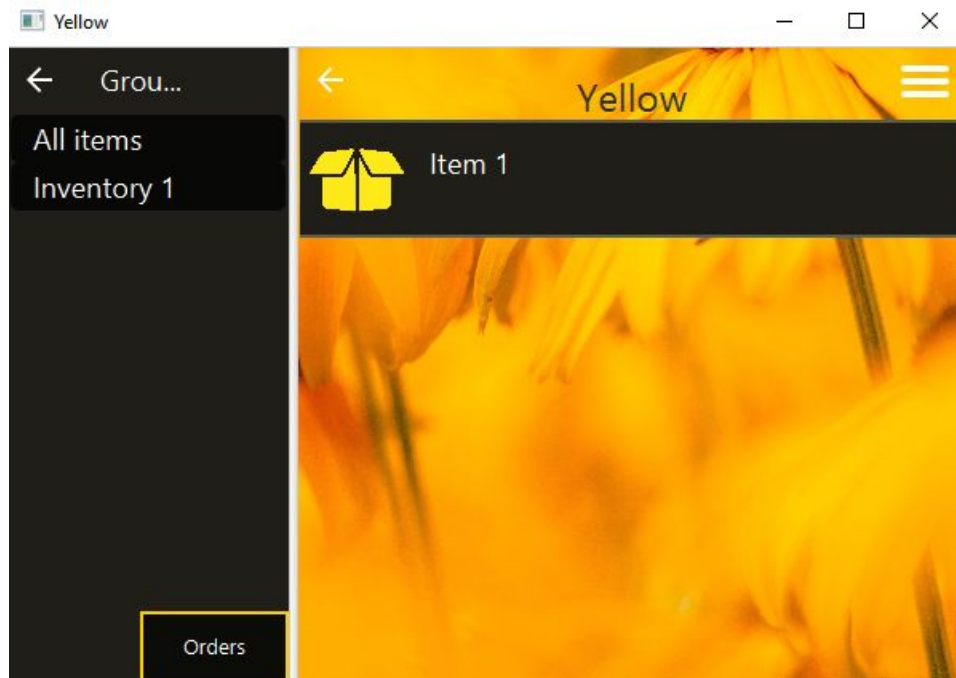


Figure 10: My Yellow → Chosen Group

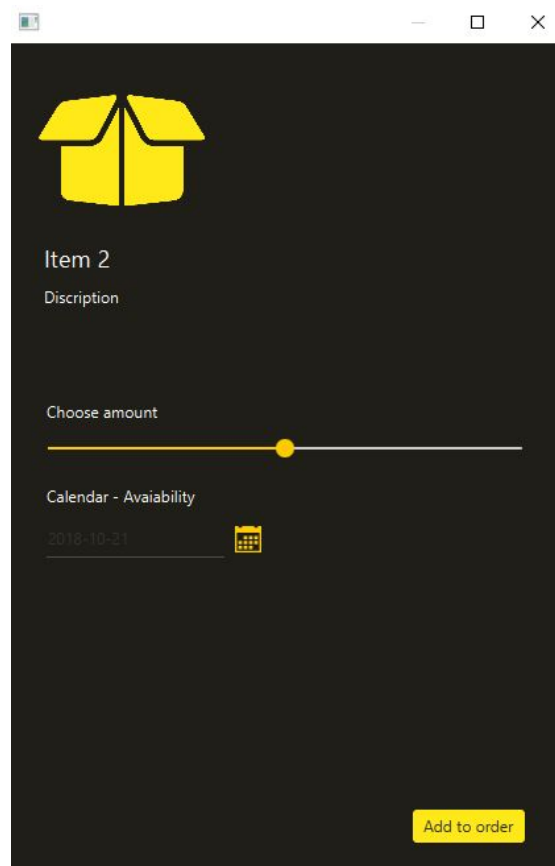


Figure 11: View Item

By clicking on a item in the list the user access the detailed information about that specific item which is shown in a dialog (*Figure 11: View Item*). It is also in this view that it is possible to add a item to the active order and view when, and how many items that is available.

While in the “My Yellow” window and a group is chosen from the list, one new button named Orders will reveal in the left bottom corner (*Figure 10: My Yellow → Chosen Group*).

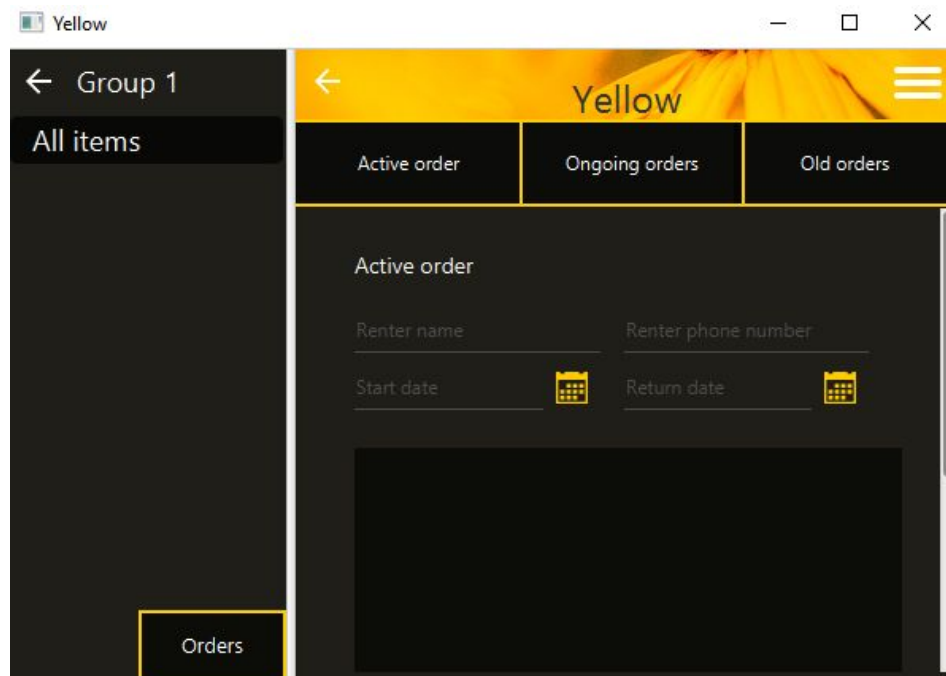


Figure 12: Active order

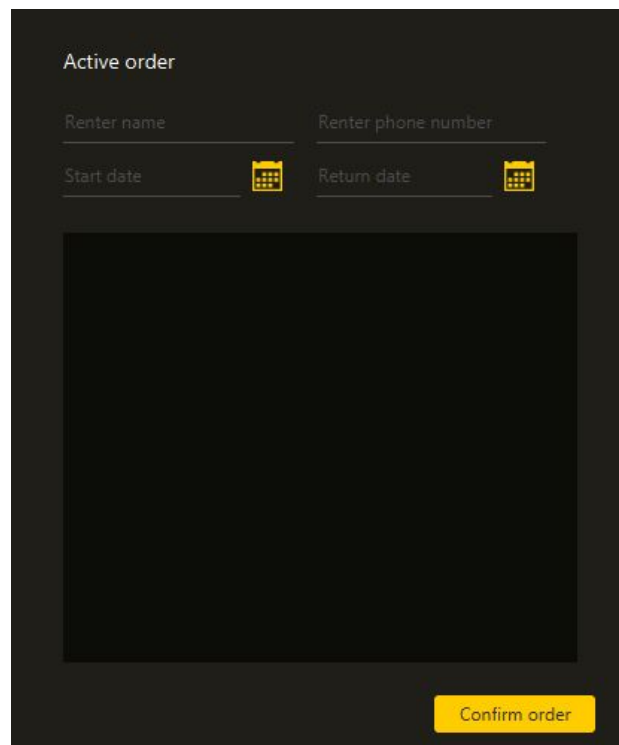


Figure 13: Active order full view

The order view is divided into three different views: Active order, Ongoing orders and Old orders (Figure 12: Active order & Figure 13: Active order full view). The active order shows all the items that is added to the order and is shown there until the order is confirmed (Figure 14: Items in active order).

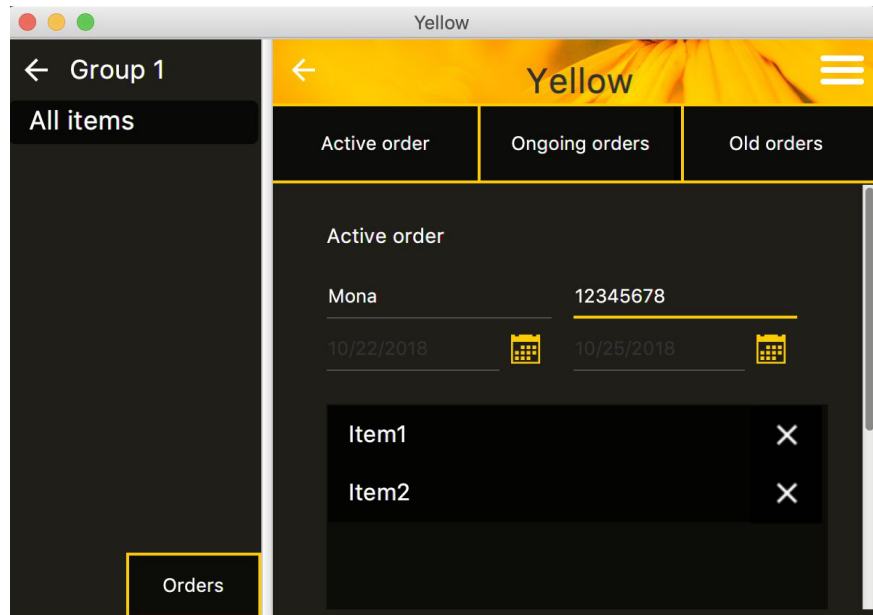


Figure 14: Items in active order

For an order to be confirmed, a renter must be entered and all the items must be available during the chosen rental period. If the items are not available during the picked date, a snackbar error message will pop up telling the user what items are not available and the order will not be placed. It is also not possible to add items from two different groups in the same order due to the fact that orders are linked to a specific group.

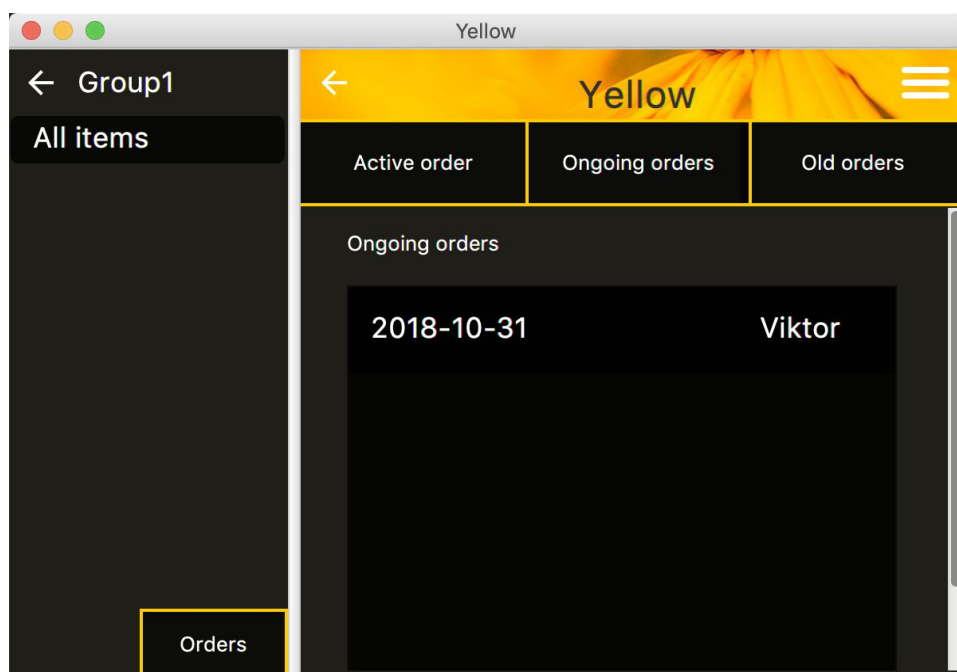


Figure 15: Ongoing orders list view

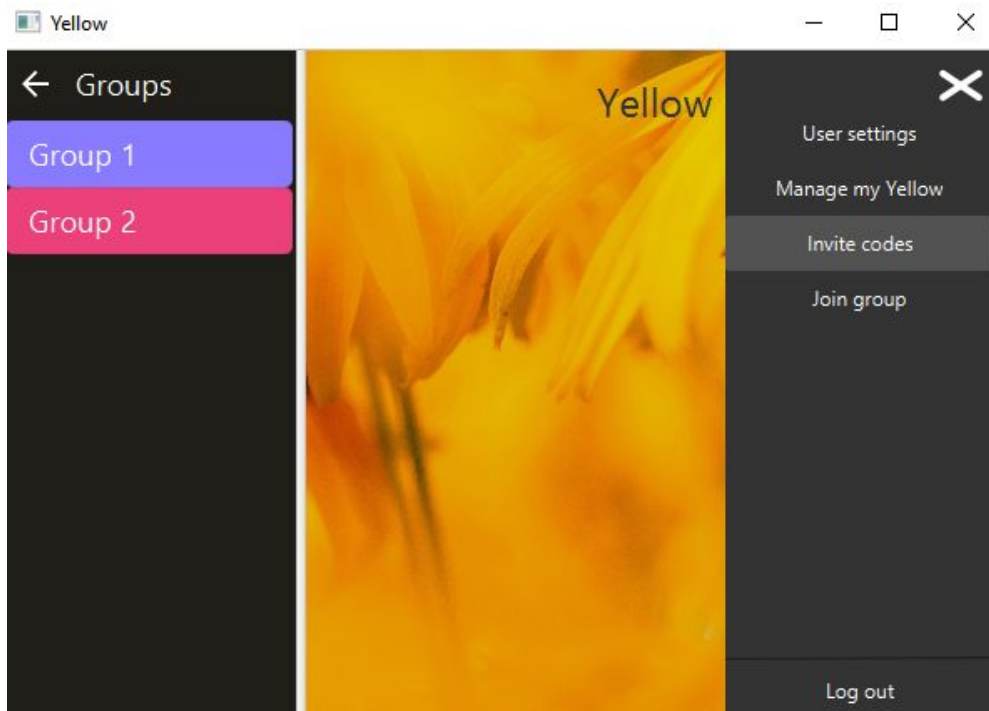


Figure 18: Drawer

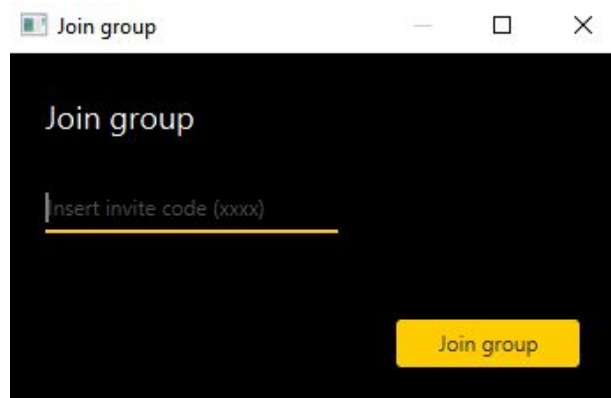


Figure 19: Join group dialog

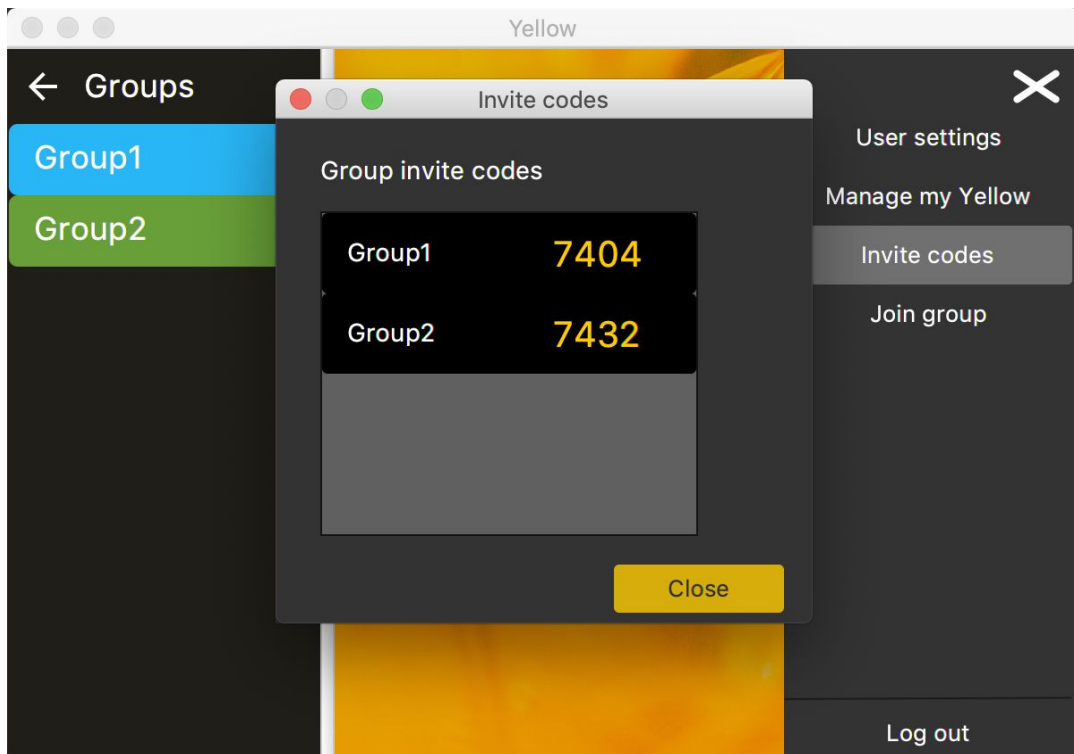
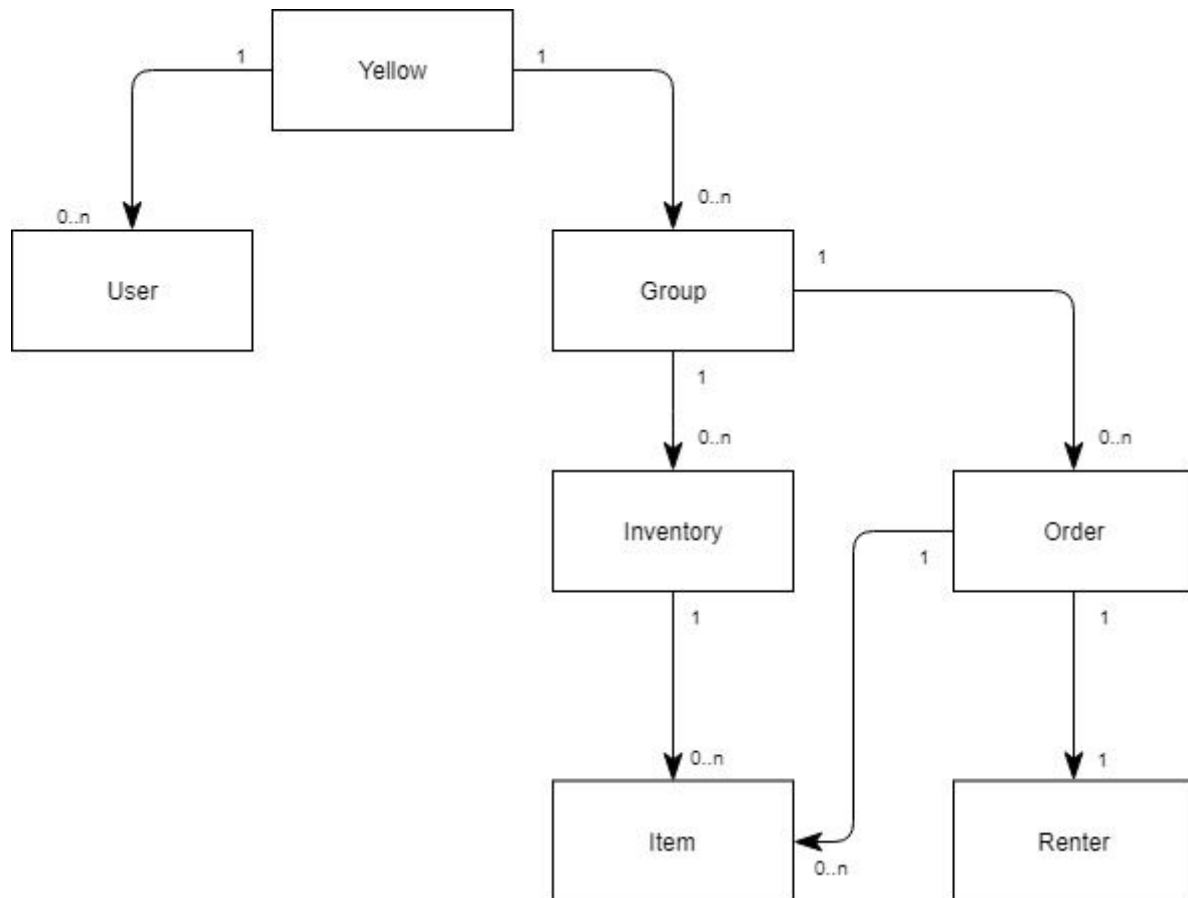


Figure 20: Show invite codes

Another essential view of “My Yellow” is the pop out menu that always is available in the upper right corner in “My Yellow” (*Figure 9: My Yellow*). By clicking on the icon a menu with multiple functions appear (*Figure 18: Drawer*). These functions are considered to be relevant to the user no matter where in “My Yellow” the user is located. This menu contains things such as user settings, a shortcut to “manage my yellow”, join group function (*Figure 19: Join group popup*), logout function and a button to show a dialog screen with the invite codes to the users groups (*Figure 20: Show invite codes*).

3. Domain model



3.1 Class responsibilities

User: Represents the users of the app. It contains information about the user, such as name, username, password and what groups they belong to in the form of an id.

Yellow: Handles the logic and management of all the data. It's the bridge between User and Group as well as the manager of the model, delegating tasks to the other classes.

Group: Represents the groups users take part of and also contains what inventories and orders are in the group.

Inventory: Represents an inventory which users can put items in. Inventories are connected to a specific group.

Item: Represents the item that can be added into an inventory.

Order: Is created to get an overview when an item/items are rented and to get information about the renting.

Renter: Represents the person who rents the item/items.