GitHub Username: UmbertoColozza

Medicine tracking

Description

This application aims to know in real time the amount of medicines you have at home. With the possibility of receiving alerts on the expiry of the medicine.

Intended User

How many times have we forgotten to have a medicine? How many times have we found expired medicines in our first aid kit? Now there will be no more problems because with "Medicine tracking" we monitor the medicines available to the family. But it can also be useful for companies that need to keep the first aid kit up-to-date.

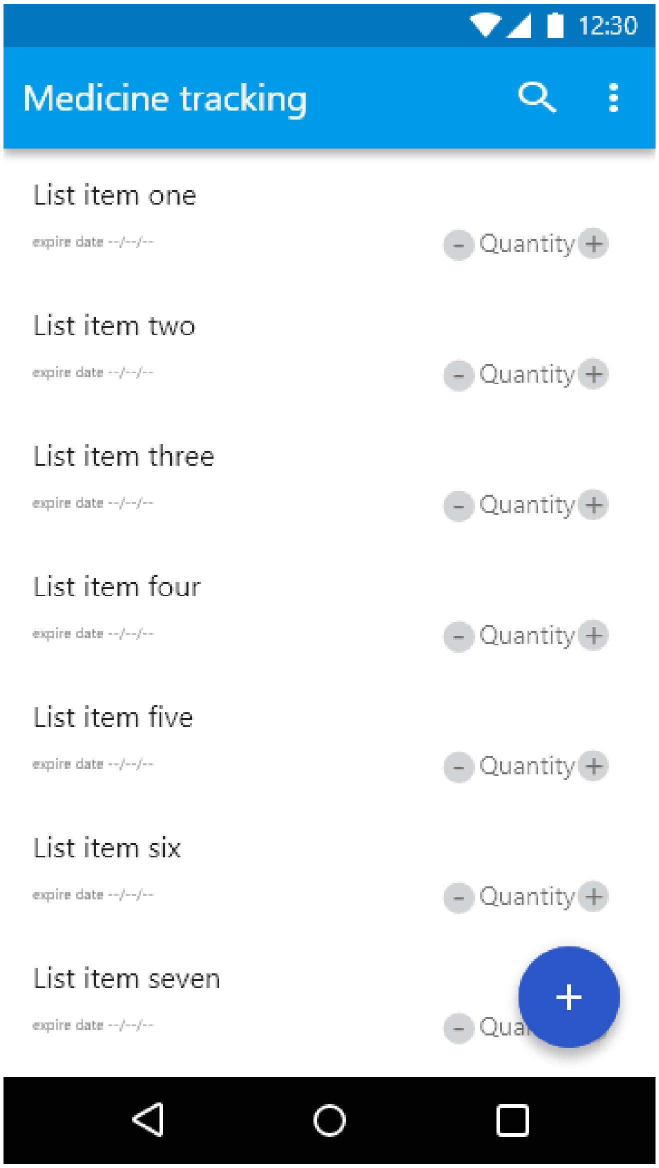
Features

List the main features of your app.

- List of medicines;
- Medicine card containing photo, name, description, quantity, expiration date;
- Insert / update / delete medicine card;
- Medicine expiration alert;
- Backup/Restore online (Google Drive) / offline (SD card), this task start in asynctask
- Alarm manager: schedule task that warns you even if the app is not working if there is some expired medicine sending a notification
- App will be solely written in the Java language.
- All of the strings within the app will be stored within a *strings.xml* file, making localization easily possible.
- Easily accessible for RTL languages by providing appropriate layout features - *start* and *end* XML attributes will be used throughout the app.

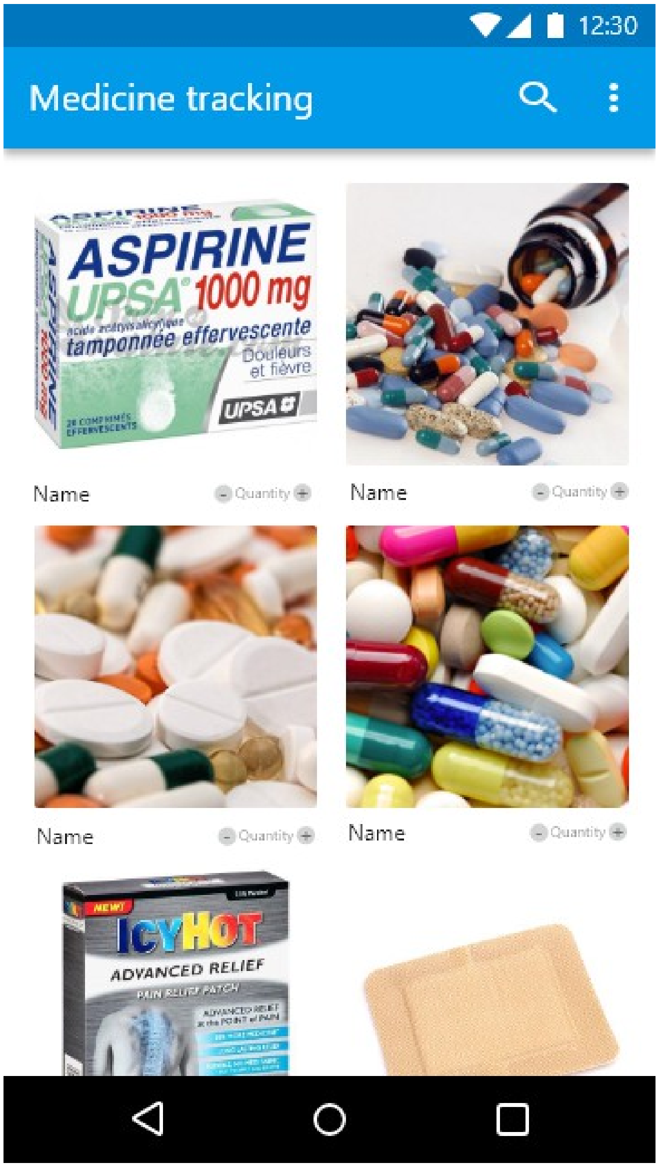
User Interface Mocks

Screen 1



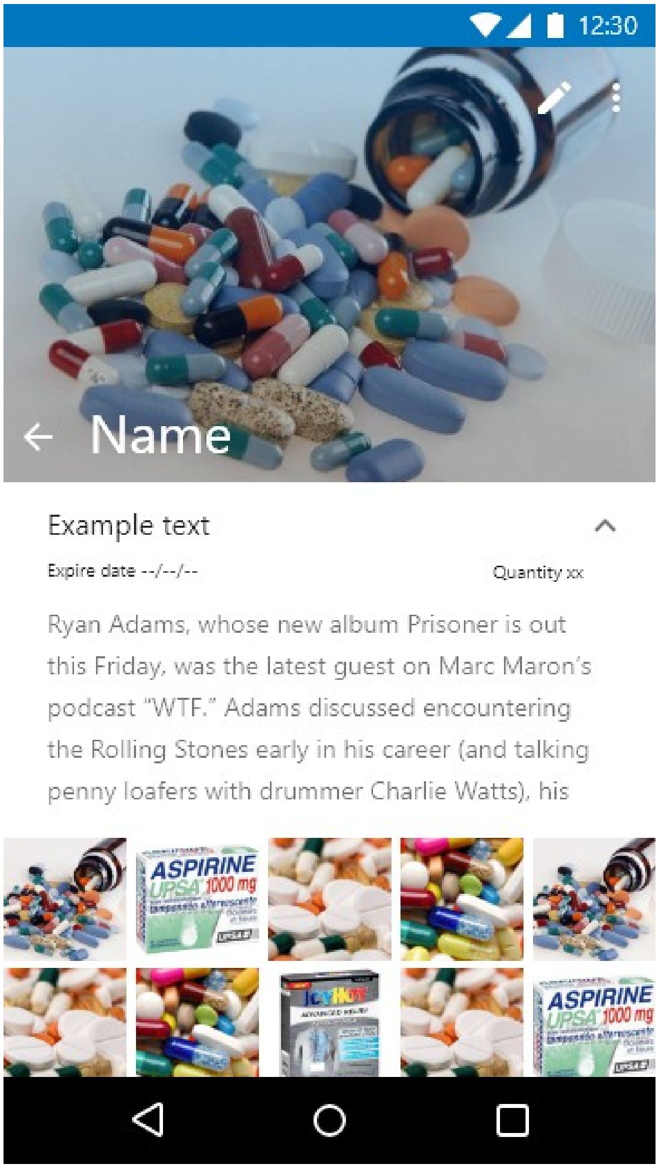
List of medicines with expiration date and possibility to change the quantity with +/-

Screen 2



Screen variant 1. Display with grid that can be selected from the settings

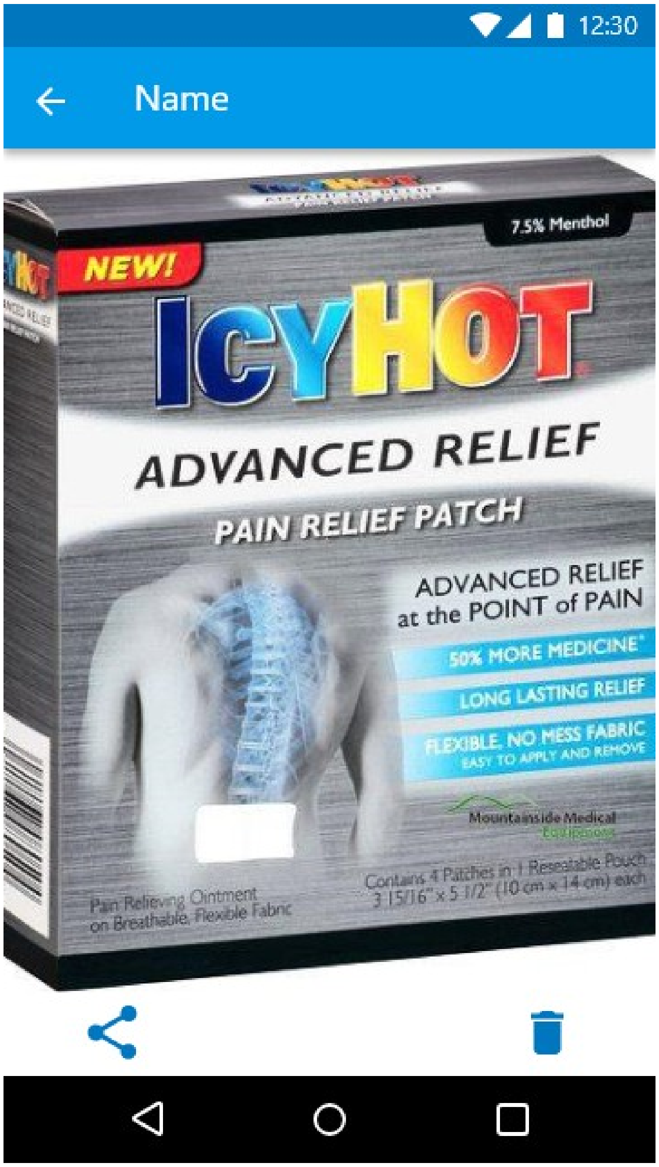
Screen 3



Ryan Adams, whose new album Prisoner is out this Friday, was the latest guest on Marc Maron's podcast "WTF." Adams discussed encountering the Rolling Stones early in his career (and talking penny loafers with drummer Charlie Watts), his

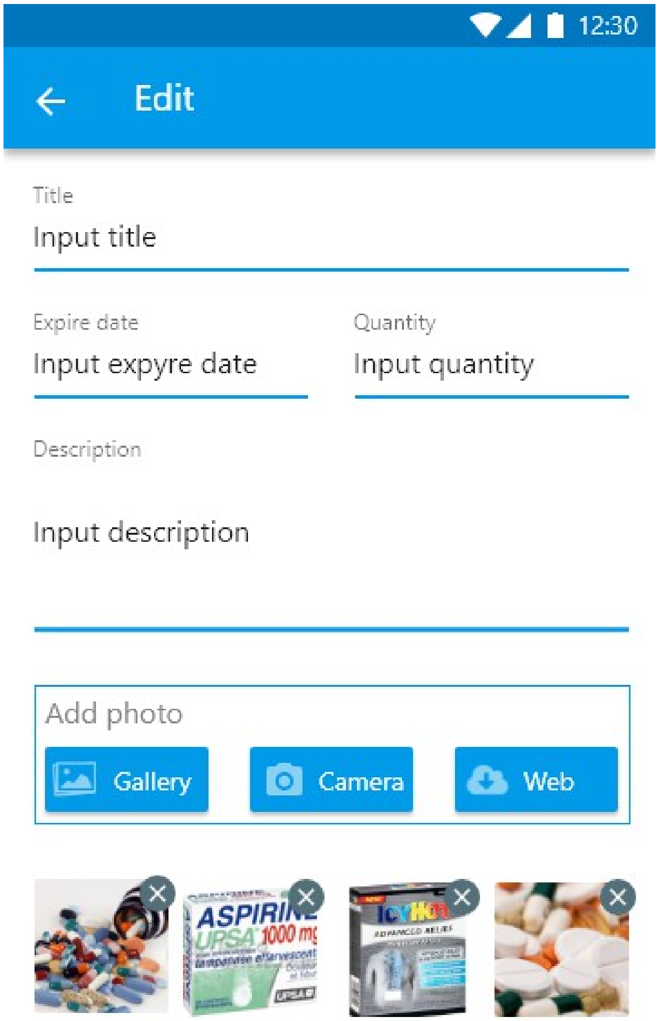
Detail sheet. Displays the name, expiration date, description, quantity and a grid of preview photos related to medicine

Screen 4


12:30
Name
7.5% Menthol
NEW!
ICYHOT
ADVANCED RELIEF
PAIN RELIEF PATCH
ADVANCED RELIEF at the POINT of PAIN
50% MORE MEDICINE*
LONG LASTING RELIEF
FLEXIBLE, NO MESS FABRIC
EASY TO APPLY AND REMOVE
Mountainside Medical Equipment
Pain Relieving Ointment on Breathable, Flexible Fabric
Contains 4 Patches in 1 Resealable Pouch
3 15/16" x 5 1/2" (10 cm x 14 cm) each

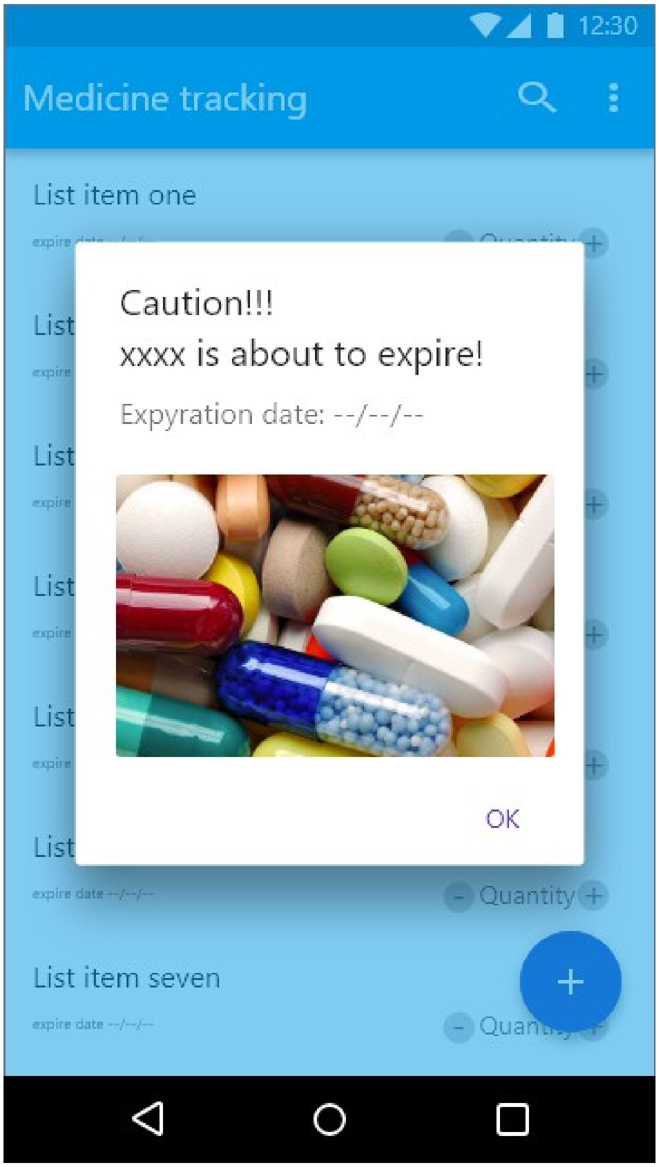
Provide descriptive text for each screen
Photo zoom of the screen 3

Screen 5



Possibility to modify, update or delete the medicine card with addition photo from gallery, camera or download from url.

Screen 6

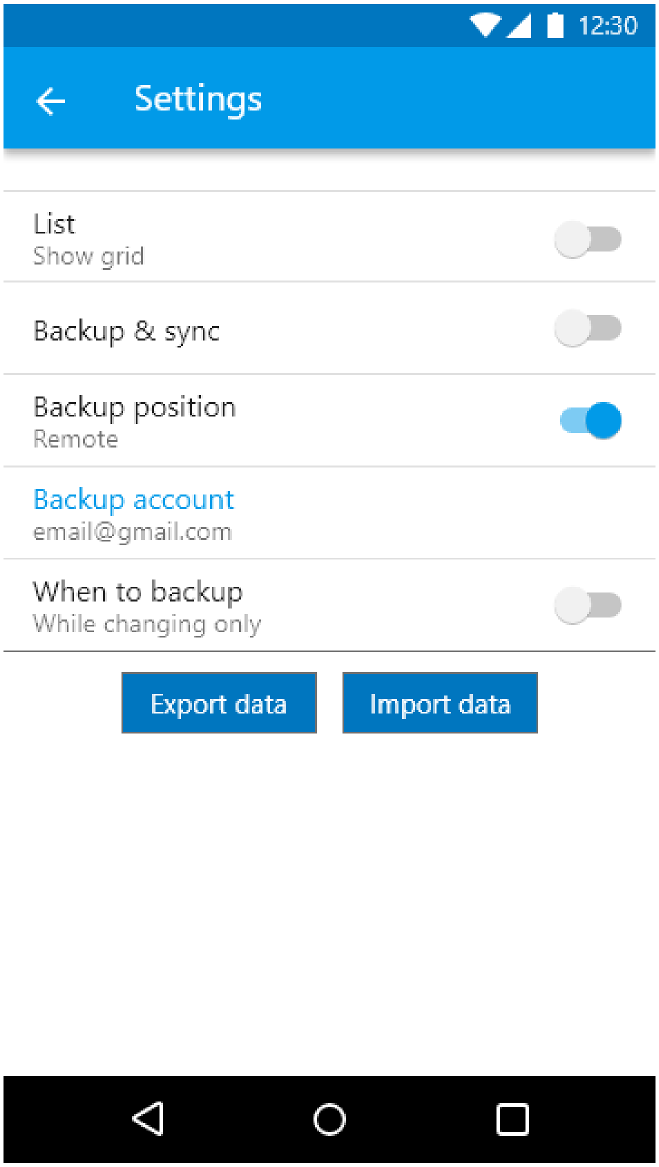


Caution!!!
xxxx is about to expire!

Expyration date: --/--/--

Alert screen. It is displayed when there is some medicine expiring

Screen 7



List
Show grid

Backup & sync

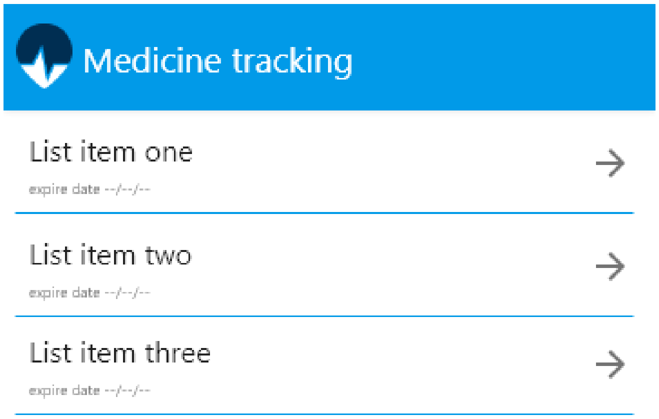
Backup position
Remote

Backup account
email@gmail.com

When to backup
While changing only

Setting screen. Here you can change the layout of the medicine list on the home screen, enable / disable automatic backup, or manually start backup or restore

Screen 8 (Widget)



Medicine tracking

List item one
expire date --/--/--

List item two
expire date --/--/--

List item three
expire date --/--/--

Key Considerations

How will your app handle data persistence?

The app will use SQLite Database with Room, Live Data and View Model .
In addition it is possible to backup / restore data online

Describe any edge or corner cases in the UX.

I thought about making this app because I noticed that there are no others, there are similar apps that warn you when to take the medicine but no one that warns you when they expire.

Describe any libraries you'll be using and share your reasoning for including them.

For example, Picasso or Glide to handle the loading and caching of images.

Room	provides an abstraction layer over SQLite to allow fluent database access while harnessing the full power of SQLite.	1.1.0
Butterknife	Binding XML views to *View* objects within Java	8.8.1

Google Drive	Read, write, and sync files stored in Google Drive	15.0.1
Google Account Login	Identity tools	15.0.1
Picasso	Image loading and caching	2.71828

Describe how you will implement Google Play Services or other external services.

Use Google Account Login and Google drive

Next Steps: Required Tasks

Task 1: Project Setup

- Download and configure the Google Play services SDK, which includes the Google Drive Android API
- Create a project in the Google API Console and get a signing certificate for the app.
- Add Drive services
- Configure libraries

Task 2: Implement UI for Each Activity and Fragment

Activity:
- Main activity
- Description activity
- Photo gallery activity
- Edit activity
- Setting activity

Fragments:
- List fragment
- description fragment
- Photo gallery fragment
- Edit fragment (also includes downloading photos via url with Picasso)
- Setting fragment

View:
- Custom alertdialog

Task 3: Implement Room

- Create model classes with entity, and viewmodel class
  Two classes are needed, one for the description of the medicine with the name "medicine" and another with the list of the photos with the name "photo"

  Class medicie contains: ID, Name, Description, Expyre data, Quantity, Show alert

  Class photo contains: ID, Medicine Id, File name

- Create Data Access Objects (DAOs)

- Create the database

Task 4: Test of database

Create unit test of database

Task 5: Widget

Create the widget with which the user can see the list of expiring drugs at home