RacingCalendar

Description

With RacingCalendar motorsport fans will never miss a race again. The app will contain the calendars for all major motorsport series with all sessions (free practice, qualyfing and races).
User will be able to follow their favorite series and personalize their notifications (get notified for all sessions of a series, just for some sessions etc...).

Intended User

This app helps motorsport fans, who follow a lot of a series, to have a single organized point to know when a session is on.

Features

Main features of the app:

- Showing a list of all available series (where you can filter them based on type).
- Showing a list of all series marked as favorites.
- Showing a list of all upcoming sessions (for favorite series).
- Customize notifications for each series or event.
- Notify the user when a session he is interested in its starting (~10 minute before).

User Interface Mocks

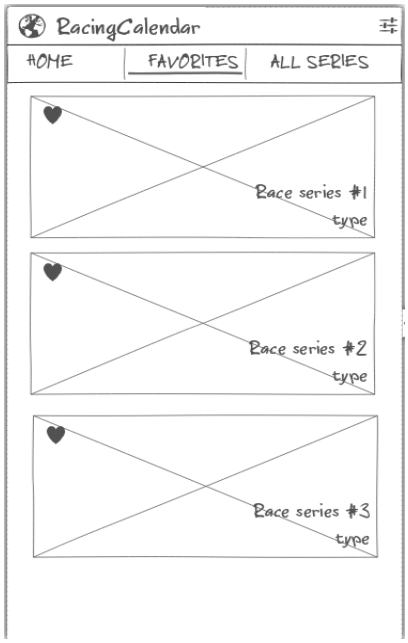
MainActivity: Home tab

In this tab user can see the upcoming sessions for its favorite series. In each event card is shown:

- A little icon representing the series type
- The name of the event
- The time at which the event will start and end (if available)
- A small “clock” icon

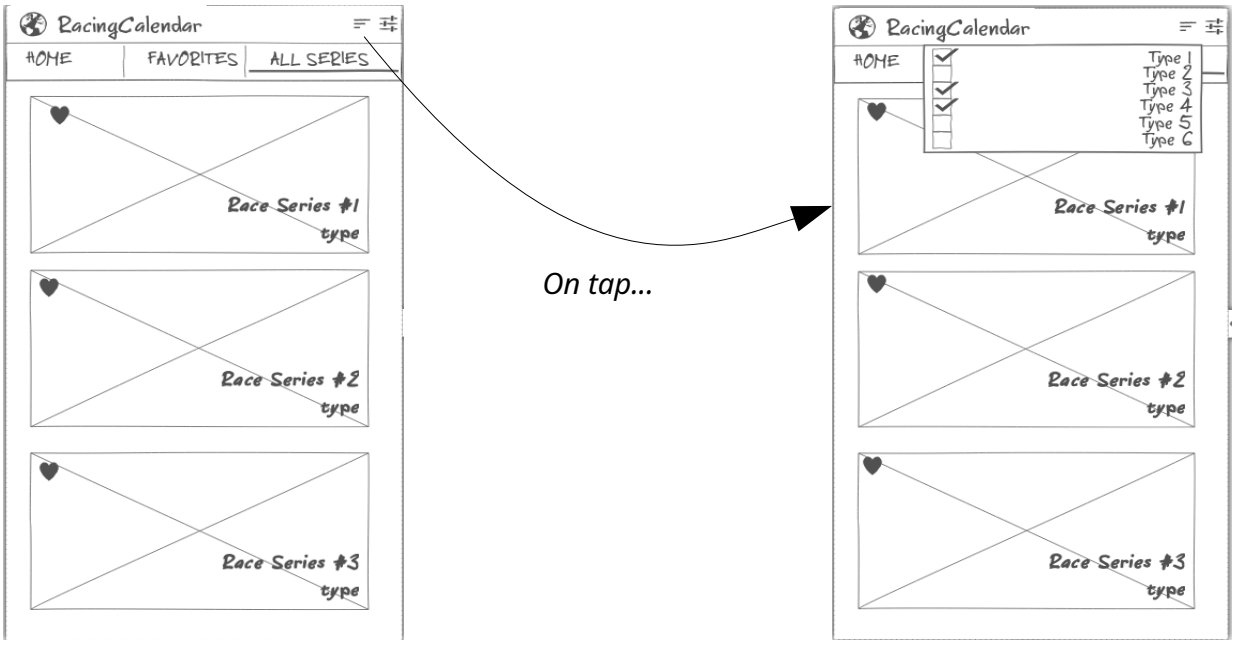
By tapping on the clock icon the user can toggle the notification option for that session.

MainActivity: Favorite series tab



In this tab user can see a full list of all its favorite series and remove them from favorites by tapping on the corresponding heart icon.

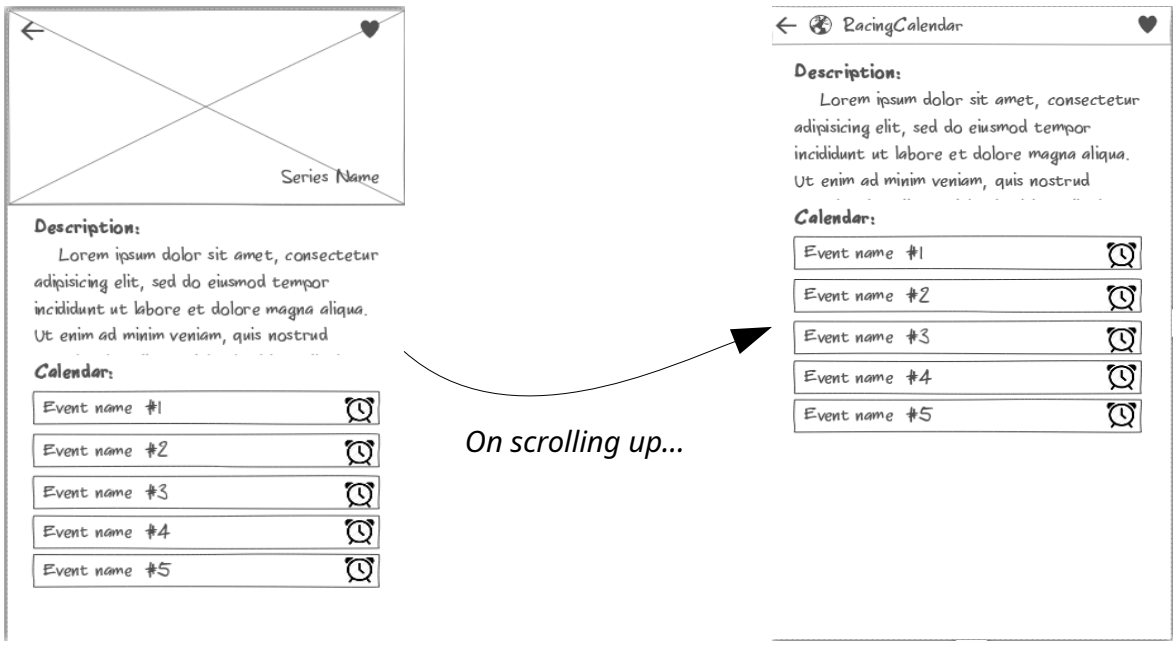
MainActivity: Series list tab



In this tab user can see all the series available on the app and select its favorites (by clicking on the corresponding heart icon).

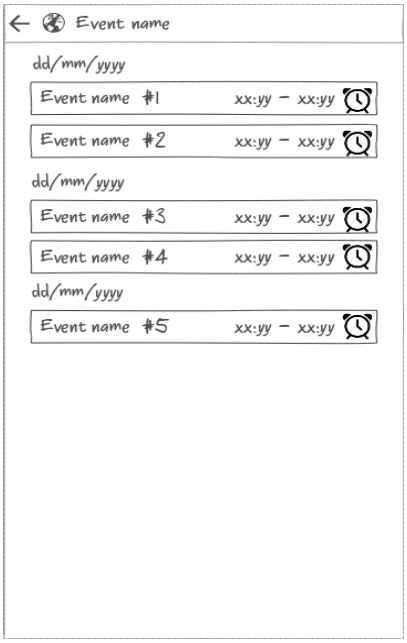
Furthermore, when it taps on the filtering action it can select which series type to show (by default all of them).

SeriesDetailActivity



This activity is started whenever the user taps on a series card in the MainActivity (both in the favorite series and series tabs).
The user will see a description of the series followed by this year calendar of events. The user is able to toggle the favorite status of the series by tapping on the heart button in the top right corner, or to toggle notification status for a whole event by tapping on the corresponding clock icon.
As you can see on scrolling app the top bar reduces in size, the background became a solid color (the main one of the image shown) and the series logo pops up on the top left.

EventDetailActivity



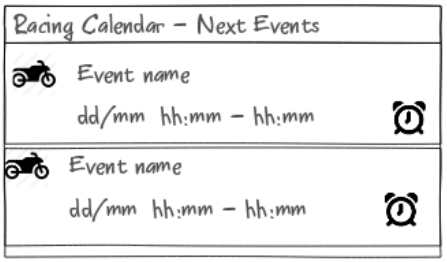
This activity is started whenever the user taps on an event on the SeriesDetailActivity. The user can tap on the corresponding clock icon to get notified of an event.

SettingsActivity

The classic settings activity where the user can set the app. The settings available in the app will be:

- Default behavior of favorite series (notify all events, notify only qulifying and races, notify just races).

Next Event Widget



The app will provide a widget which will show the next events in time order.
Again by tapping on the clock icon the user will toggle notification status. If it taps on one vent the EventDetailActivity is started. If it taps on the top bar the MainActivity, with the Home tab selected, will be started.

Key Considerations

How will your app handle data persistence?

The app will store the calendars of favorite series in a database by using Room.
Data will pull data from server (for non favorite series) when requested using an AsyncTask.

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

The libraries i plan to use are:

- Butterknife (v. 8.8.1), to bind the views in my activities.
- Parceler (v. 1.1.6), to parse objects to activities.
- Picasso (v. 2.5.2), to load the images from the urls.
- Gson (v. 2.85.5), to parse json response from the server.
- OkHttp (v. 3.9.0), to perform asyncronous http requests.
- EasyAdapter (v. 2.0), to manage recycler views.
- Room+LiveData (v. 1.1.0), to manage the database of favorite series

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

To store the calendars for all the series i will develop a simple RESTful api in php and deploy it on altervista.org, since i have a bit of experience in doing so from other projects.

Next Steps: Required Tasks

Task 1: Create RESTful API

1. Develop a simple RESTful API in PHP to which you can ask:
   - The list of series:
     http://<hostname>/series/
   - The details of a series:
     http://<hostname>/series?name=<series_name>
   - The current series calendar:
     http://<hostname>/calendar?name=<series_name>
   - The sessions in an event:
     http://<hostname>/event?id=<event_id>
2. Deploy the API to altervista.org

Task 2: Develop a Java library to retrieve calendars

1. Develop a java library which will use OkHttp and Gson to perform queries to the RESTful API created in Task 1.
2. Add it as dependency for the project.

Task 3: Develop an Android library to save calendars of favorite series

1. Develop an Android library which will use Room+LiveData to save and remove favorite series for a database.
2. Add it as dependency for the project.

Task 4: Develop an Android library to get subscribe and get notified before an event

1. Develop an Android library which uses AlarmManager to subscribe to an event and that sends a notification when the event is imminent.
2. Add it as dependency for the project.

Task 5: Create the main activity divided in three tabs

1. Develop an activity with three tabs
2. Develop layout and behavior of “All Series tab”
3. Develop layout and behavior of “Favorites tab”
4. Develop layout and behavior of “Home tab”

Task 6: Create the series detail activity

Task 7: Create the event detail activity

Task 8: Create the next events widget