

## Mobile Computing

### 4. Lab Assignment: Fragments and Universal App

30.05.2024

**Presentation of results on: 13.06. (A) / 20.06. (B)**

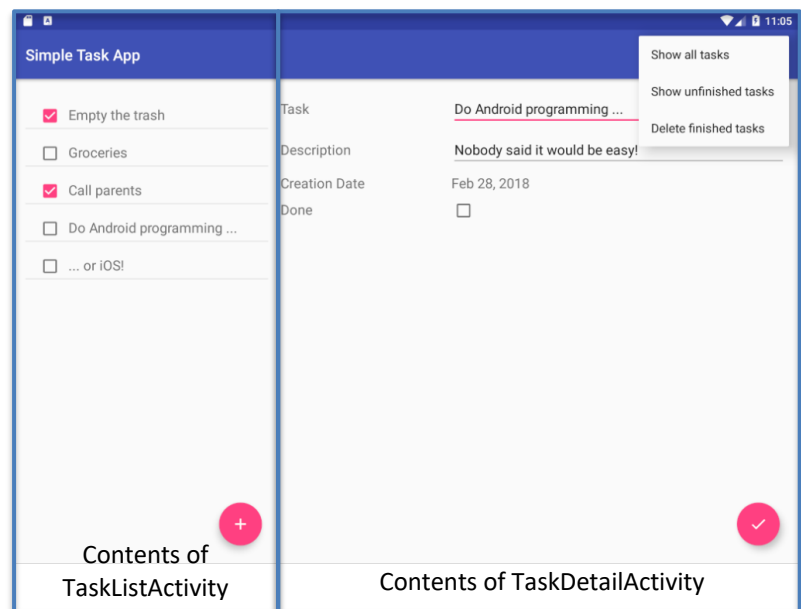
**Please note: your final work results (standard track) have to be submitted on these dates.**

### Objectives

Working with Fragments

### Your Task

Your app should now run both on a smartphone (as before) and a tablet. If run on a tablet, the task details should be shown next to the tasks list (cf. figure).



To achieve this goal, you need to *roughly* do the following:

- Extract the UI logic and application logic from the activities into two fragments.
- Make the activities use these fragments, i.e. the activities should “talk” to the fragments and the fragments should return information (events, e.g. task edited, task created) back to the activities using dedicated callback interfaces. (Now, the app should run on a smartphone as before).
- Assign an alternative layout to the TaskListActivity which is used, if the app is run on a tablet (or a smartphone in landscape mode, if you like). This layout should combine the two fragments.

Make sure that the app’s complete functionality works both on a smartphone and a tablet.

See page 2 for bonus tasks.

---

## Bonus Tasks

If you finished the above task, you may gather bonus points by adding one or more of the following features to your app:

1. Architectural improvements: Try to improve your app's architecture and data flow by regarding the following concepts and technologies:
  - Make use of Androids *LiveData* and allow to query tasks from the repository / database only once and observe any changes to the resulting task list in your fragments (observable queries).
  - Implement a *ViewModel* for the task list and task detail UIs, i.e. both UI screens should get their data from a specific *ViewModel* instance and forward user interactions such as submitting a task details form or creating a new task (or task list) to these instances. For the task detail UI, it makes sense to use data binding to establish this connection.
  - **Note:** for this bonus task – if done well – you will be granted a larger bonus than for others.
2. Implement swipe gestures in your list view for deleting items (maybe also for finishing tasks).
3. Support manual sorting of items within the list view.