



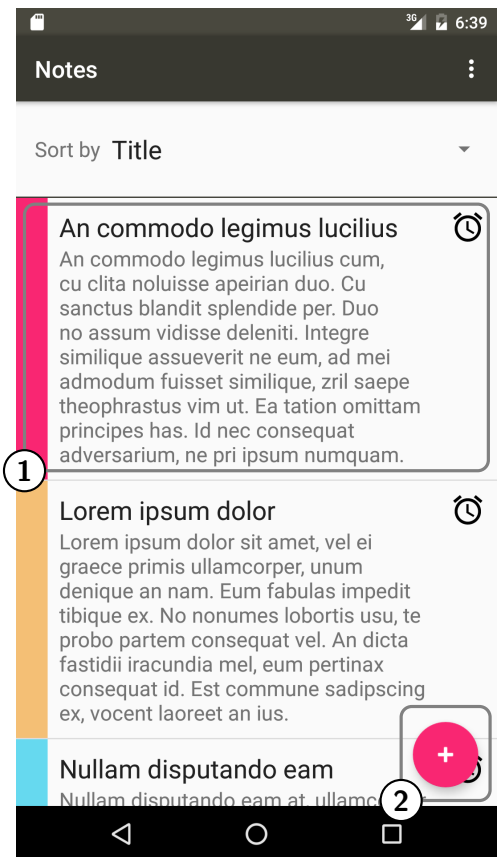
Assignment III

(Due date is on Léa)

In this assignment, you will integrate the activities and fragments from the first two assignments into a single functioning app.

The user will start by interacting with the note list activity. Now, they will be able to create and edit their notes.

- ① Clicking on a list item will allow the user to edit the existing note.
- ② Clicking the `FloatingActionButton` allows the user to create a new note.



1 Integrating Assignments and refactoring

Begin by verifying the following project setup. Refactor as needed. *This project structure will be important when using the layouts I've provided below.*

1. Include your Assignment 1 activity and fragment files. Refactor these files to use the name “Note”:

```
NoteActivity.java
NoteFragment.java
activity_note.xml
content_note.xml
fragment_note.xml
menu_note.xml
```

2. Include your Assignment 2 activity and fragment files. Refactor these files to use the name “NoteList”:

```
NoteListActivity.java
NoteListFragment.java
activity_note_list.xml
content_note_list.xml
fragment_note_list.xml
menu_note_list.xml
```

Include the spinner and list item layouts with whatever name you've chosen.

3. Include all files relating to the database in a package called `model/`.
4. In `content_note.xml`, set the fragment ID to: `note_Fragment`.
5. In `content_note_list.xml`, set the fragment ID to: `noteList_Fragment`.

2 Creating a Note

A note is created in the following way:

1. In `NoteListActivity`, the user clicks the `FloatingActionButton` with the + sign.
2. `NoteActivity` is launched for the user to create their note. The fields are set to their default (blank) values.
3. The user writes their note, then presses the SAVE action button. This creates a new note in the database.
4. The app returns to `NoteListActivity` and the list is refreshed to include the new note.

3 Editing a Note

A note is edited in the following way:

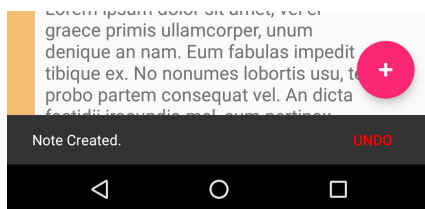
1. In `NoteListActivity`, the user selects an item in the list. Replace the toast you used to for testing this event in your previous assignment.
2. `NoteActivity` is launched for the user to edit their note. The fields are set to the current values of the note.
3. The user edits the note content, then presses the SAVE action button. The note is updated in the database.
4. The app returns to `NoteListActivity` and the list is refreshed to include the update to the note.

4 Notification and Undo

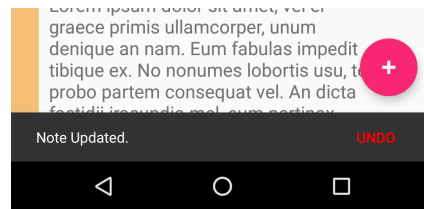
In `NoteListActivity`, use a snackbar to inform the user that the note was either created or edited. The snackbar should be created in `NoteListActivity` in connection to the `FloatingActionButton` above. Here's the code to create a snackbar:

```
Snackbar snackbar = Snackbar.make(fab, hello, world, Snackbar.LENGTH_LONG)
    .setAction(press me, new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            // TODO: event handler
        }
    })
    .setActionTextColor(Color.RED);
snackbar.show();
```

Create an “undo” button on the Snackbar to allow the user to undo either operation.



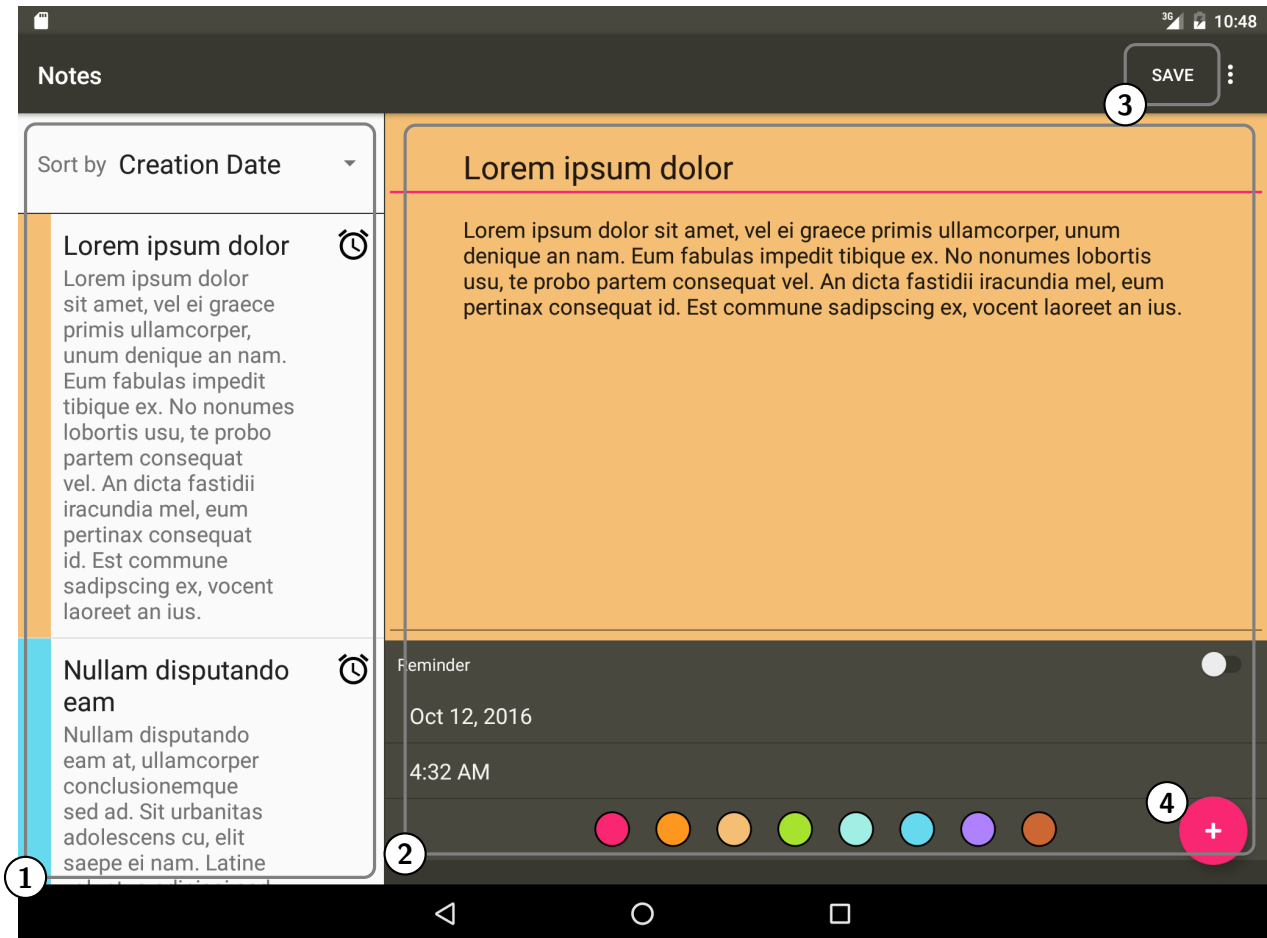
Note created with undo.



Note updated with undo.

5 Tablet

Using the techniques from class, update your app to work on a tablet device. The app will now consist of a single activity `NoteListActivity` with both of the fragments:



Tablet consists of both fragments.

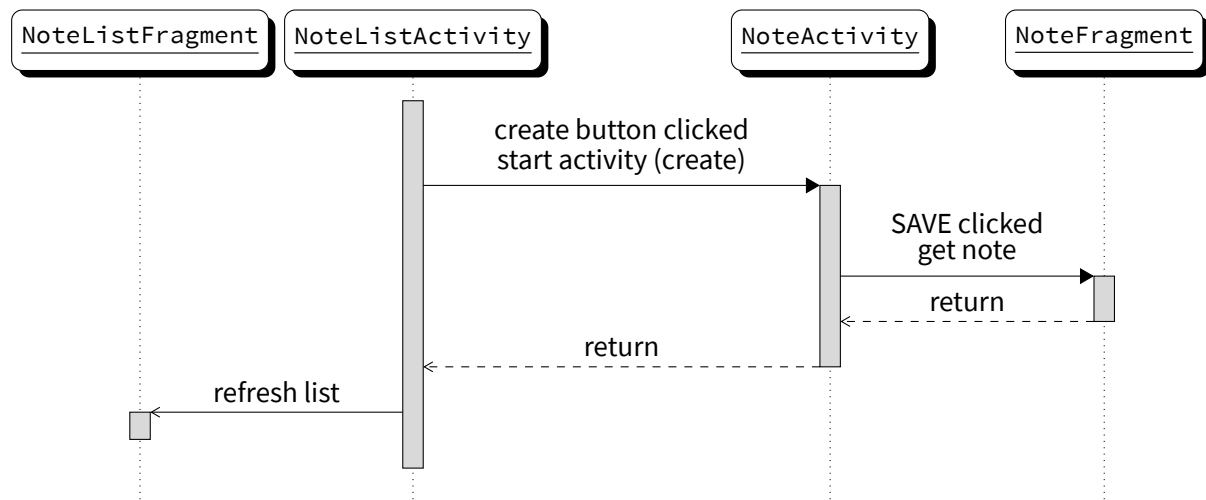
- ① `NoteListFragment` works as it did before except now clicking the items will allow them to be edited in the fragment on the right.
- ② `NoteFragment` works as it did before.
- ③ The SAVE action will save the new or edited note and refresh the list.
- ④ The `FloatingActionButton` still allows the user to create a new note, this time in the fragment on the right.

To help you out, I've included two layouts:

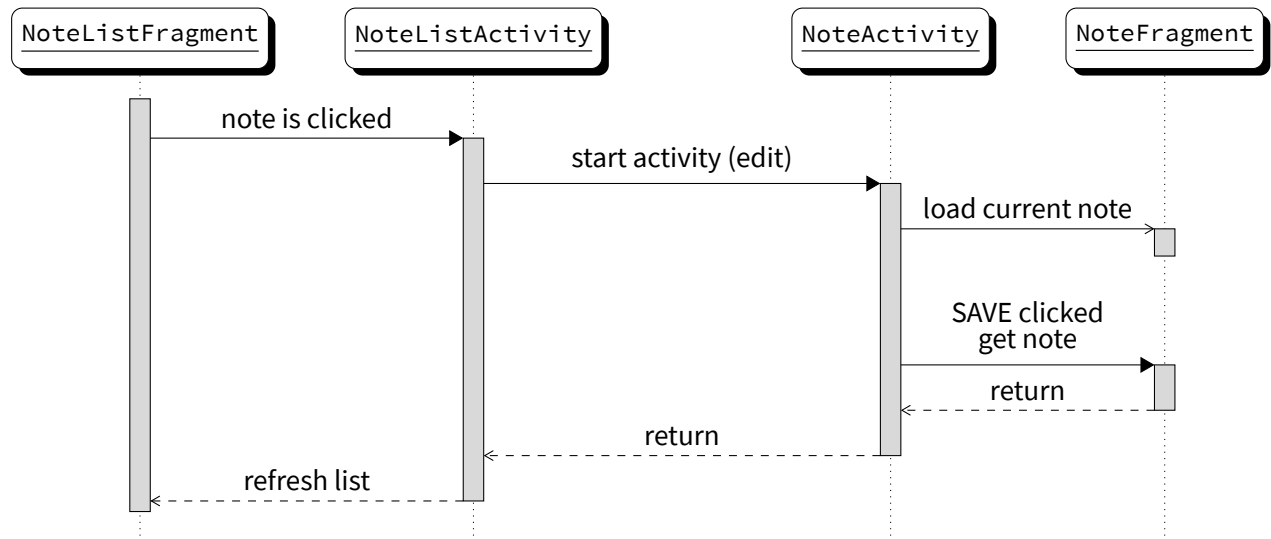
```
res/layout-large/content_note_list.xml
res/menu-large/menu_note_list.xml
```

6 Sequence Diagrams

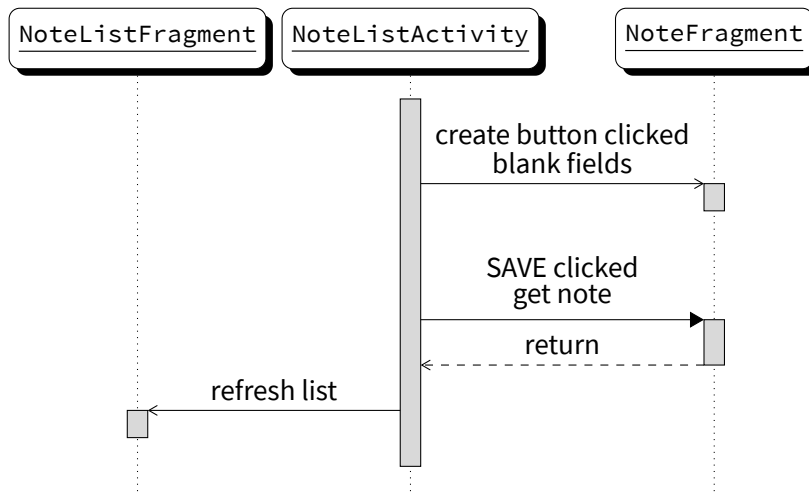
Note is created (handset).



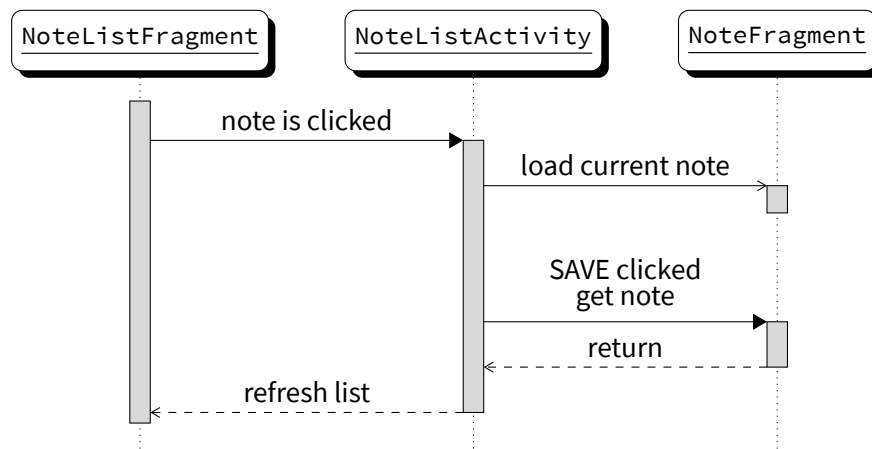
Note is edited (handset).



Note is created (tablet).



Note is edited (tablet).



7 Requirements

- Your program should be clear and well commented. It must follow the “420-616 Style Guidelines” (on Léa).
- Work from your previous assignment, or create a new Android project with minimum SDK 21 or later.
- Your project structure (class names, etc...), match those from Section 1.
- Note creation and editing are done as outlined in Sections 2 and 3. The snackbar with undo works as outlined in Section 4.
- Your app works for both handset and tablet devices as a single project. The class `NoteListActivity` is implemented to work with both device types.
- Submit your project using git. Follow the Git Submission instructions on Léa.