

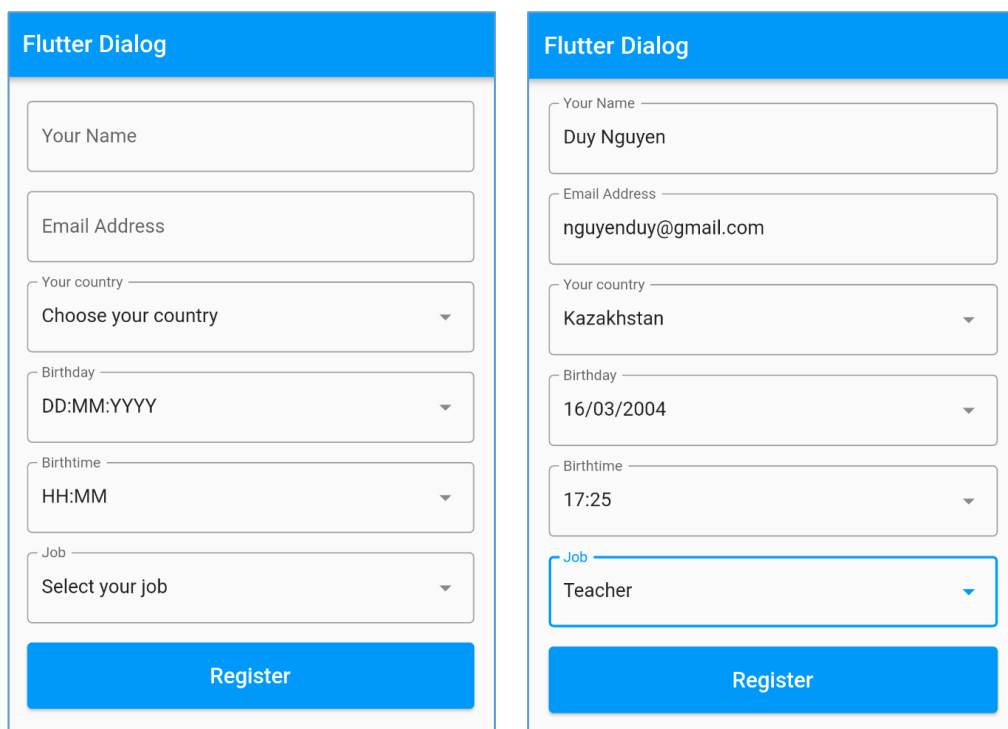
## CROSS-PLATFORM MOBILE APP DEVELOPMENT

(503107)

### LAB 7

#### EXERCISE 1

Create a registration form that allows the user to enter information such as first and last name, email address, country, date of birth, time of birth, and job. The above fields need to be validated carefully, the application displays the appropriate validation message for each specific situation.

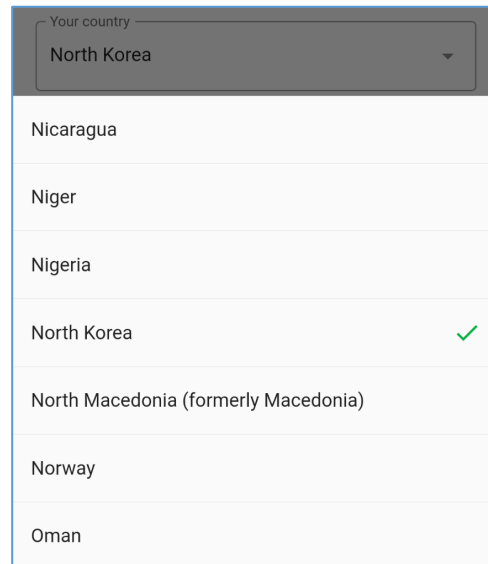


The image displays two side-by-side screenshots of a Flutter registration dialog. Both dialogs have a blue header with the text 'Flutter Dialog'. The left dialog shows the initial state with empty input fields: 'Your Name', 'Email Address', 'Your country' (dropdown menu), 'Birthday' (DD:MM:YYYY), 'Birthtime' (HH:MM), and 'Job' (dropdown menu). The right dialog shows the form filled with sample data: 'Duy Nguyen', 'nguyenduy@gmail.com', 'Kazakhstan', '16/03/2004', '17:25', and 'Teacher'. Both dialogs feature a blue 'Register' button at the bottom.

It is necessary to display several different dialogs in the following situations:

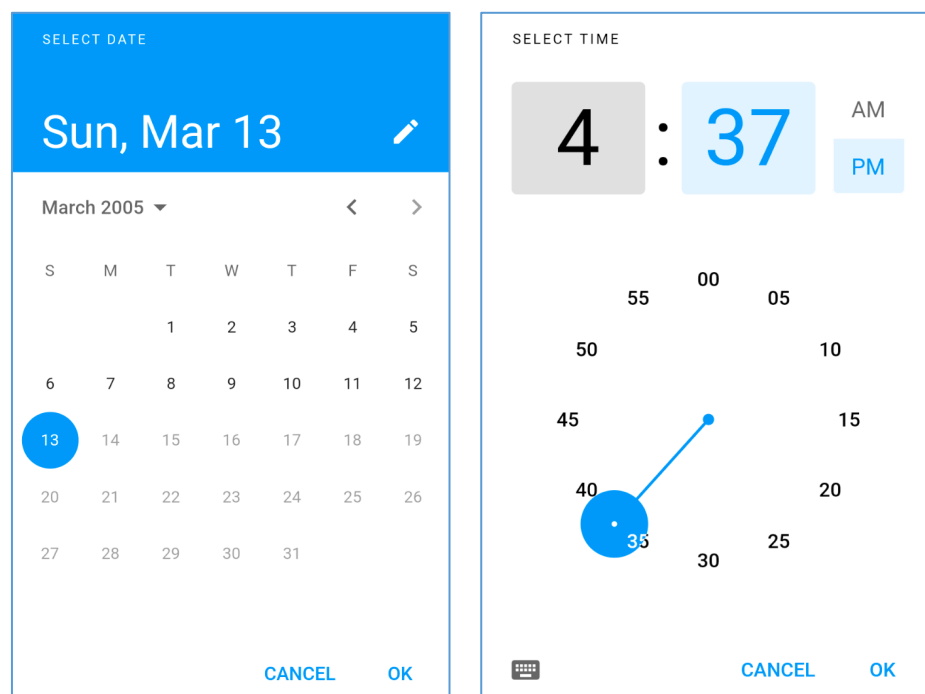
- **Country field:** the user cannot manually enter the value, instead a bottom sheet will be displayed, providing a list of countries in a listview widget for the user to choose from. In subsequent dialog displays, the listview should automatically scroll to the previously

selected country position and display a check mark icon next to the currently selected country name.



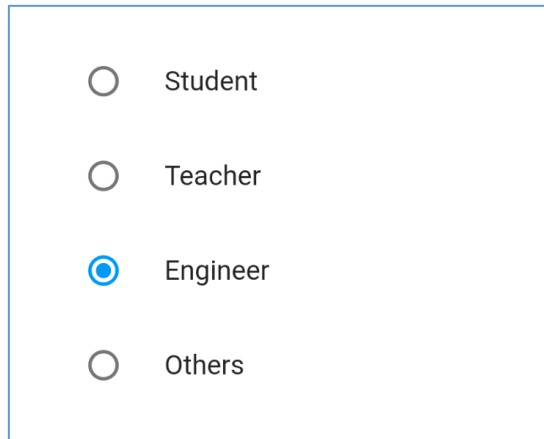
A dropdown menu titled "Your country" with "North Korea" selected. The menu is open, showing a list of countries: Nicaragua, Niger, Nigeria, North Korea (with a green checkmark), North Macedonia (formerly Macedonia), Norway, and Oman.

- **Birthday and birth time fields:** The app displays the date picker and the time picker dialog respectively. For date picker, only values in the range 18-99 years old can be selected.



Two side-by-side screenshots of mobile app dialogs. The left dialog, titled "SELECT DATE", shows a calendar for March 2005 with the date "Sun, Mar 13" selected. The right dialog, titled "SELECT TIME", shows a time picker set to 4:37 PM. Both dialogs have "CANCEL" and "OK" buttons at the bottom.

- **Job Field:** Users are selected 1 of 4 career values provided in the form of single choice dialog. Dialogs are displayed in the form of stateful widget, meaning that whenever they tap into a choice, the value of the field and dialog will be updated immediately.



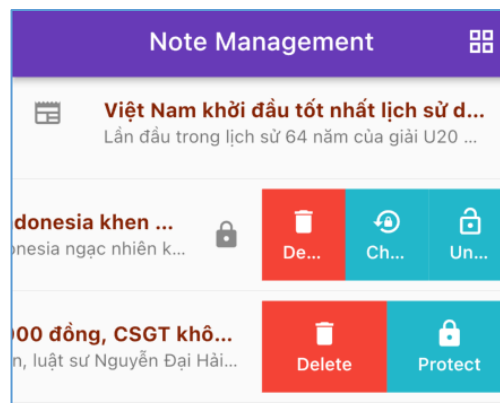
A single choice dialog box with four radio button options:

- ☐ Student
- ☐ Teacher
- ☒ Engineer
- ☐ Others

## EXERCISE 2

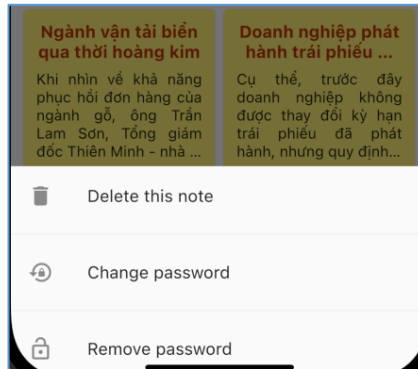
Continue working on the note management exercise from last week's lab. Add dialogs to make the application more interactive. Please fulfill the following specific requirements:

- In the listview mode, add the flutter\_slidable package to set the action menu items on the notes. When you swipe on any note, you will see options including: delete, set password, [change password, delete password]...



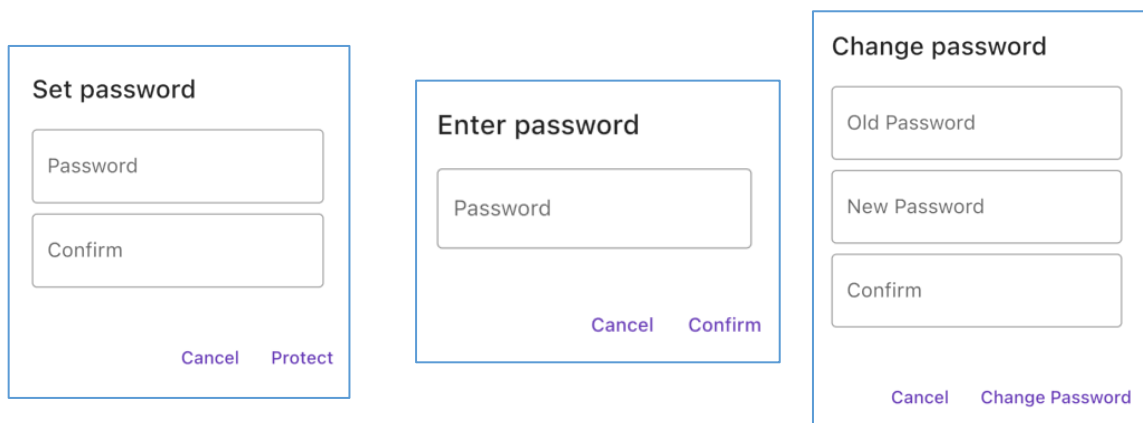
*If password is enabled, the swipecard menu will display "change password" and "remove password"*

- In the grid view mode, instead of using flutter\_slidable, you need to display the bottom sheet every time you do a long press on the notes, the bottom sheet needs to provide the same options as the swipable menu in the listview mode.



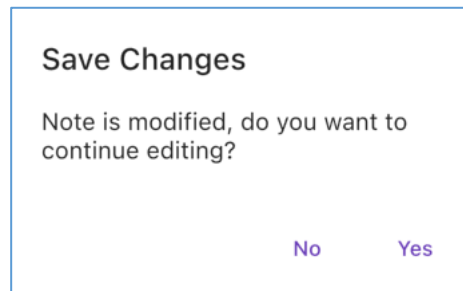
*Bottom sheet is used in gridview mode*

- Before officially deleting a note, you need to display an alert dialog to confirm with the user. After deleting the note, display a snackbar message to notify the user along with an "undo" text button on the snackbar for the user to restore if desired.
- When setting a password for the note, a custom alert dialog will be displayed asking the user to enter the password twice. After the note has been passworded, a lock icon should be displayed next to the note so that users can distinguish which notes are password protected. When reading notes that are protected by a password, an alert dialog will appear asking the user to enter the password, only if the correct password is entered will they be able to view the contents of the note.



*Different types of dialogs are used for the password protection feature*

- Adjust the note update function to display an alert dialog asking the user if he wants to save the content before exiting, in case the user presses the back button to exit while the note content has been edited.



*A dialog is displayed to warn you when the user exits and forgets to save.*

- Use flutter\_colorpicker package to allow user to change note background color (in gridview mode, default color is yellow). Then, in the note adding / editing interface, there will be an additional section that allows users to select or adjust the background color of the note.
- Added the ability to change password and remove password for a password protected note. When changing the password, the user needs to enter the old password and enter the new password twice. If the password is deleted, the user must re-enter the password to confirm before deleting.