ITIS/ITCS 4180/5180 Mobile Application Development
Homework 2
Date Posted: 01/30/2012 at 8:00am
Due Date: 02/07/2012 at 5:00pm

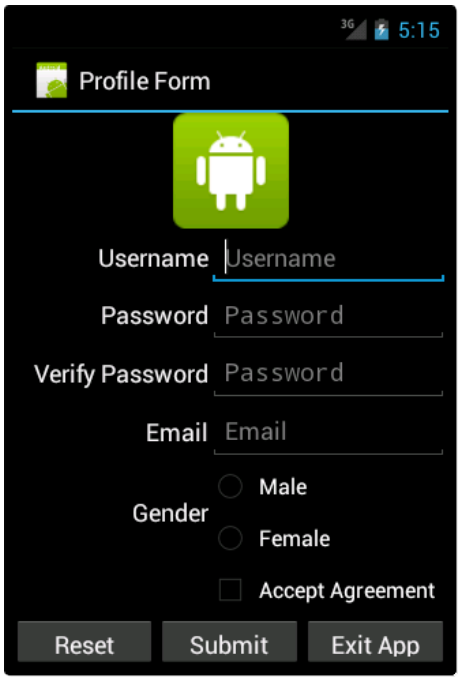
Basic Instructions:

1. In every file submitted you MUST place the following comments:
    a) Assignment #
    b) File Name.
    c) Full name of all students in your group.
2. Each group should submit only one assignment. Only the group leader is supposed to submit the assignment on behalf of all the other group members.
3. Please download the support files provided with this assignment and use them when implementing your project.
4. Export your Android project as follows:
    a) From eclipse, choose "*Export…*" from the *File* menu.
    b) From the Export window, choose *General* then *File System.* Click *Next.*
    c) Make sure that your Android project for this assignment is selected. Make sure that all of its subfolders are also selected.
    d) Choose the location you want to save the exported project directory to. For example, your *Desktop* or *Documents* folder.
    e) The first time you export, make sure you select *Create directory structure for files*.
    f) Click Finish, and then go to the directory you exported the project to. Make sure the exported directory contains all necessary files, such as the `.java` and resource files.
5. Submission details:
    a) When you submit the assignment, compress your exported Android project into a single zip file. The format of compressed file name is:
        i. HW#.zip
    b) You should submit the assignment through Moodle:
        i. Submit the zip file.
6. **Failure to follow the above instructions will result in point deductions.**

Homework 2 (100 Points)

In this assignment you will build your first Android application. You will get familiar with common Android components and how to interact with them. You will build a form that accepts user input, verify the various input values, and show appropriate error messages when needed.

Building the Form (50 Points):
You will build a *Profile Form* application similar to the one shown in the Figure to the right. You will be dealing with the `main.xml`, `strings.xml`, and `drawable-ldpi` files to create the form. The `main.xml` file can be modified through the raw `xml`, or through the GUI tools provided within eclipse (**See Hints at end of this document**). To build the form, you will perform the following tasks:



1- Use a combination of nested *horizontal* and *vertical* `LinearLayout` layouts. These will allow you to organize the form components as seen in the figure.

2- Add a user profile image using a `ImageView`. The image for this `ImageView` is provided within the support files as `android.png`.

3- Use `TextView` components for the *Username, Password, Verify Password, Email, and Gender* labels as seen in the Figure. The actual string values for these labels must be read from the `strings.xml` file provided.

4- Use `EditText` components for the *Username, Password, Verify Password, and Email* Input fields. Each of the inputs will have a `hint` (placeholder) value read from the `strings.xml` file. For example, the *Username* input has a `hint` value equal to Username.

5- For the Gender selections, use a `RadioGroup` with two `RadioButton` components. By default, both `RadioButtons` must be unselected. The labels for each radio button must be read from the `strings.xml` file programmatically (Do this in the `onCreate()` method).

6- Add a `CheckBox` with the label *Accept Agreement* (from `strings.xml`), which by default should be unchecked.

7- Finally, you will add 3 buttons, a *Reset, Submit, and Exit App* button. The text for these buttons must also be read from the `strings.xml` file.

8- The *Reset* button should have a click event listener that will reset all form fields (Clear all input fields, and remove all selections for radio buttons check boxes).

9- The *Exit App* button will also have a click event listener that exits the user from the application and takes him back to the home screen.

10- The *Submit* button will verify all input values and selections as seen in the following part.

Input Verification (50 Points):

Once you build your Profile Form, users are able to input values and make various selections. It is now your responsibility to verify the inputs and selections once the user clicks the *Submit* button. When verification fails, you will be asked to show a proper error message to the user. You will use the error message values provided in the `strings.xml` file.

Form verification should be handled as follows:

1- If any of the *Username, Password, or Email* input fields are empty you will show a corresponding error message using the `setError()` method. For example, if the *Username* field is empty, you will show "Username Not Provided!" (read from the `username_error_message` string in `strings.xml`).

2- If the Password and the Verify Password input fields are not equal, show the "Passwords Do Not Match!" error message using the `setError()` method.

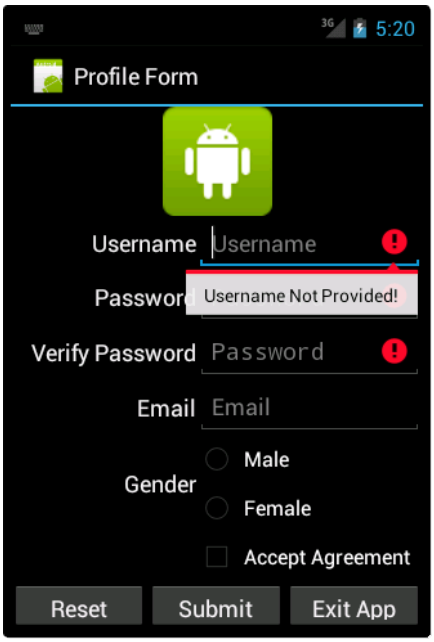
3- If the email entered by the user is invalid show the "Email Format Invalid!" message using the `setError()` method. For verifying the email, use the Java `Pattern` and `Matcher` classes (**See Hints**).

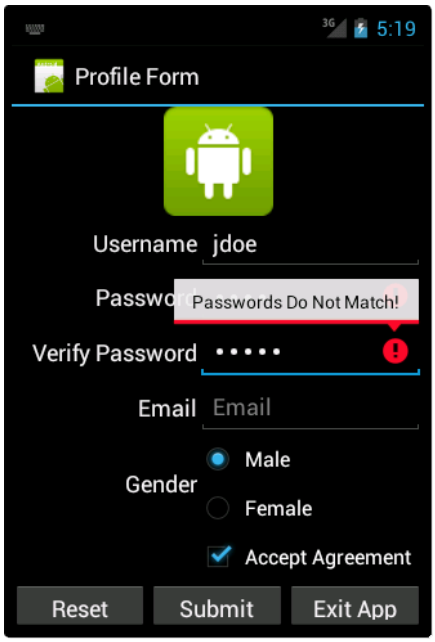
4- If no gender is selected, show the "No Gender Selected!" message. To show this message use Android's Toast class (**See Hints**).

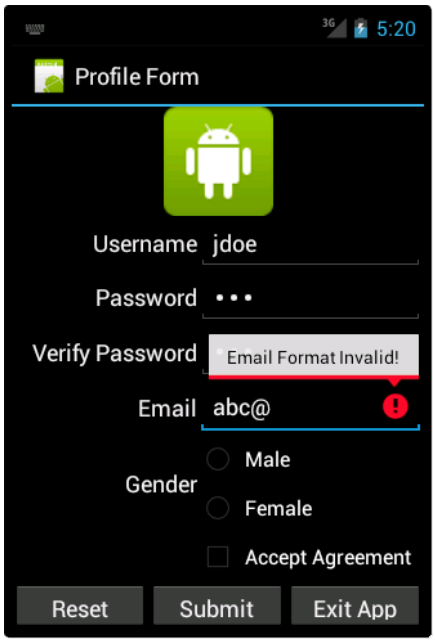
5- Users must check the Accept Agreement checkbox. If they do not, show the message "You Must Accept the Agreement!". Use Android's Toast class.

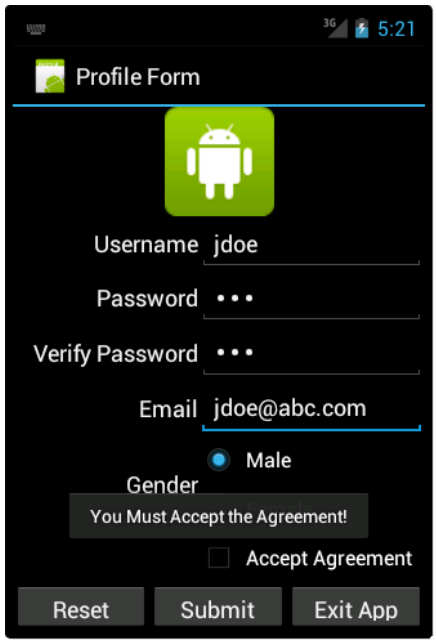
6- When all of the above cases are verified and all inputs are correctly input and all selections have been made, you will show the message "Verification was Successful!". Use Android's Toast.

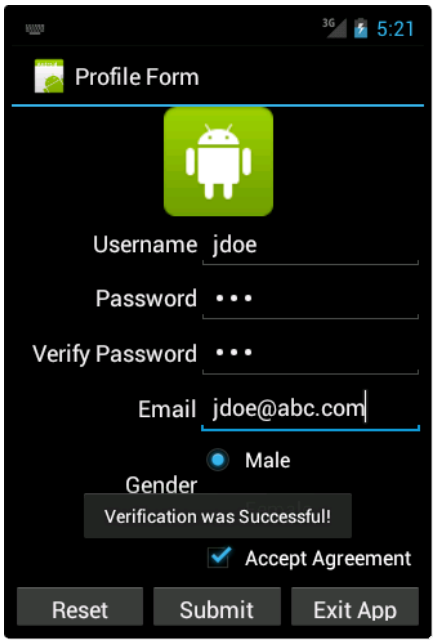
Example Screen Shots











Hints

Modifying the main.xml file:

1- For GUI based editing, right click on the main.xml file and select `Open With` >> `Android Layout Editor`.

2- For aligning UI components vertically or horizontally use the `gravity` property.

3- Use Medium sized TextViews.

Opening the strings.xml file:

1- For GUI based editing, right click on the strings.xml file and select `Open With` >> `Android Resource Editor`.

Verifying Email Addresses:

1- To verify if a certain string follows the email format, you can the `Pattern` Class provided by android.util.

Android Toast Notifications:

1- Toast notifications show up for a limited time and disappear automatically. The following shows you how to show a Toast.