|  |
| --- |
| **An Najah National Universlity**  C:\Users\Wasief\Pictures\logo.gif  **Engineering Faculty**  **Department of Computer Engineering** |
| **Graduation Project Documentation** |
| **Traveler Kit** |
|  |
| **Ammar M.O Haj Hamad (10611827 )**  **Izzat Ahmad Kukhon (10718833)**  **Supervisor : Dr. Luai Malhis**  **De**cember 2011 |

Abstract:

Traveler kit application is an android based application that provides the travelers with services that they might need during their travels, either for a tourist, education or business travels. This application benefits from the services that provided by smart phones such as GPS, internet access, Camera and it’s processer in the image processing techniques.

There are a lot of countries in the world and a lot of them are different from each other in many ways such spoken language, currencies, weather and time differences such as finding a good place to sleep in and communicate with the local people .The traveler will face some difficulties during his trips to other countries, our application provides services designed to reduce these difficulties, which make his trip easier than before and save his time that may be wasted.

This documentation describes our graduation project in four chapters:

* In Chapter one you will find a brief introduction about the project
* Chapter two discuses the application Methodology, the main work done in developing the application ,the software and API’s used in the project .
* Also you will find in this chapter a Review weekly reported works.
* In Chapter three you will find a detailed discretion of the provided services
* Conclusion and future work will be found at the end of the document

**Content List**

**List of Tables/Figures** 1

Acknowledgements 2

Chapter 1: 1

Introduction: 1

Methodology 3

Chapter 2 : 3

Section 1 : Main work 3

Chapter 3: 5

Section 1 Welcome Screen 5

Section 2 Main Menu Activity : 7

Section 2 :Applications languages and API’s used 4

Section 3: OCR Translator 9

Section 4: Server communication: 16

Section 5: Picture and Comment 23

Section 6 : Phrases 28

Section 8 : global clock service 33

Section 9: Weather activity : 35

Section 10:Places service 37

Section 11 : Currency convertor 38

Section 12: Update My Location : 40

Section 13 : User Personal Information: 42

Section 14 :Sending Feed Back Email: 44

Conclusion**:** 45

Future work : 46

References and Bibliography 47

**List of Tables/Figures**

[Figure 1:Welcom Screen Activity 5](#_Toc311943891)

[Figure 2 : Main menu Activit 7](#_Toc311943892)

[Figure 3 Select Image source for OCR Translator 10](#_Toc311943893)

[Figure 4 OCR Trasnlator Activity 16](file:///C:\Users\Ammar\Desktop\An%20Najah%20National%20Universlity.docx#_Toc311943894)

[Figure 5: User Registration form 17](#_Toc311943895)

[Figure 6:Backup Activit 19](#_Toc311943896)

[Figure 7: Backup success 20](#_Toc311943897)

[Figure 8:Restore Activity 20](#_Toc311943898)

[Figure 9: Restore done 21](#_Toc311943899)

[Figure 10: Picture and Comment Main Activity 24](#_Toc311943900)

[Figure 11:Select Picture Source Dialog P&C 25](#_Toc311943901)

[Figure 12:Picture Not Commented yet 26](#_Toc311943902)

[Figure 13: Picture Comment Activity 27](#_Toc311943903)

[Figure 14:Update Comment 27](#_Toc311943904)

[Figure 15:Update Picture Activity 28](#_Toc311943905)

[Figure 16: Main Activit 30](#_Toc311943906)

[Figure 17: Reading Phrases 31](#_Toc311943907)

[Figure 18: Action Menu 32](#_Toc311943908)

[Figure 19: Add New Phrase 33](#_Toc311943909)

[Figure 20: Update Activit 33](#_Toc311943910)

[Figure 21: Global Clock 35](#_Toc311943911)

[Figure 22: Weather Main Activit 37](#_Toc311943912)

[Figure 23: Select Weather Location Activity 38](#_Toc311943913)

[Figure 24: Location Main Activity 39](#_Toc311943914)

[Figure 25: Currency Converter Activity 40](#_Toc311943915)

[Figure 26: Personal information 44](#_Toc311943916)

[Figure 27:User Information Update Activity 45](#_Toc311943917)

Acknowledgements

This project could not be done without Dr. Luai Malhis who not only served as our supervisor but also encouraged and challenged us throughout our academic program. He and the other department staff, Dr. Raed Al Qadi, Dr. Samer Arandi, Dr. Hanal Abuzant, and all other department staff guided us through the learning process, never accepting less than my best efforts. We thank them all.

Chapter 1:

Introduction:

Traveling to other countries became a lifestyle and very common between people for a businesslike, educational or for a tourism purpose. People are traveling to countries which sometimes are completely different than their homes in many ways; spoken language, in climate, currency and obviously in time difference, thus they might face some difficulties in communicating with others and getting what they want especially for the first days during their visits. For example a non English speaker visiting U.K and wants to ask for a place to stay in, or looking for a nearby ATM machine , knowing what is the time back home or translating some words and what is the correct pronouncing for these words i.e. restaurant menu .

This issue and the fact that smart phones became very common, made us thinking to find a good and an easy way to help the travelers and make it easier in their trips than before, and from here we got the idea of our graduation project which is “**Droid** **Travel Kit**”

***Droid*** ***Travel Kit*** *is a smart phone application in general* aims to:

1. Help the traveler to communicate with the local people
2. Help the tourists in finding nearby places such as a hotel
3. Make sure that the travelers will enjoy their time during their travel by making it easier
4. Knowing the common words in some languages and how to pronounce them
5. Saving time and effort for the travelers
6. Minimize the cultures difference challenges
7. Help the user to remember the occasion of the pictures he took during his visit

***Soft Travel Kit*** provides the users with these services:

1. OCR “***Optical Character Recognition***” Translator where the user can use the cell-phone to capture an image contains English text and translate it to any language
2. ***Pronouncing*** words written in English, French and Arabic.
3. ***Common phrases*** Providing the user with the most common words in English ,French and Arabic languages
4. ***Picture and comment*** which is a service where the user can take a picture and write down his comments so that he can remember the occasion at which the image was captured
5. **Global Clock** enabling the user to know the time in his home country and other countries he choose
6. ***Weather***: the user has the ability to check the weather state in his current location and other locations he choose
7. ***Currency converter*** giving the user the ability to convert currency unit to another
8. ***Places***  the user can use this services to know the nearby places such as ATM’s hotels and museums …etc
9. Saving the application data in the ***application’s server*** and restore them on request

Methodology

Chapter 2:

Section 1: Main work

We can divide our application’s main work into main four parts:

1. Learning android developing basics (took three weeks)

Before developing our application we needed to read about android because knowledge about android developing was not sufficient enough to start developing the project.

1. Optical Character Recognition “OCR” (took a month )
   1. We started in reading about OCR in general and how it works, so that we can use this technique in out project.
   2. Converting the Tesseract OCR engine from C language into java language, to use it because the our developing language for android is java
   3. Testing the OCR engine
2. Server developing Using PHP language : ( took a week )
3. Developing the rest of our application services (took 2 months )
4. Application final testing and interface (took one week )

Section 2: Applications languages and API’s used

**Languages used:**

1. Java language
2. PHP language
3. XML language

**Applications used:**

1. Eclipse IDE on ,Linux copy
2. Net beans , Linux copy
3. Client FileZilla ,Linux copy
4. Android SDK, Linux copy
5. Android NDK ,Linux copy

**API’s and Web services used :**

1. Google Translator API V1 and V2
2. Tesseract OCR engine
3. Sky hook
4. Yahoo weather API
5. Currency web service
6. Google Places

Chapter 3:

Section 1 Welcome Screen

**3.1.1 Overview** :

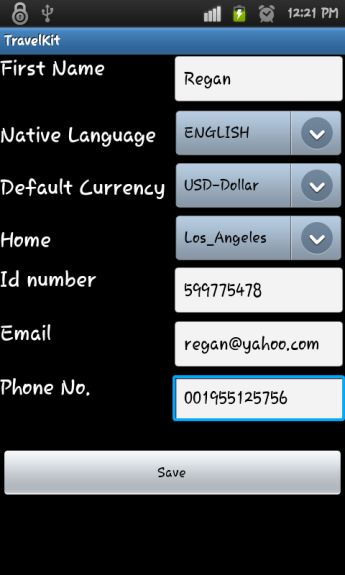


Figure :Welcom Screen Activity

Where the user will enter his personal information needed to run the application in the way that is designed for.

1. The required information:
2. His Name
3. Email Address
4. Home
5. Phone Number
6. Native Language
7. Default Currency
8. ID Number

**3.1.2 . Methodology :**

1.2.1:

The activity consists from four edit text views ,where the user can enter his Name, Email, Phone No. and his ID number . And three Spinners (Compo Boxes) to select his Home, Native Language and default currency from the listed values .

1. The Currency spinner is loaded with the international currency names from corresponding xml file that contains the Currencies names and International currency code in pairs .the selected currency is stored in the data base and used to set the “To “ currency name in the currency converter activity
2. For the Native language spinner , the languages provided for the user from xml file to select one of them. The selected language is used to set the language the user wants to translate to , in the translator , as default language .

c. And for the Home spinner , the values loaded from an array list called Zones which has the zones time , this selected value stored in the data base to set the home clock at the global clock activity .

1.2.2:

The entered information values are stored in the data base . In a table called User Info .the table has the following fields

Table 1 User Info

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ID | Name | Email | Home | Phone\_Number | Native\_Lang | Default\_Currency | ID\_Number | Flag |

1.2.3:

To guarantee that welcome screen activity will not be launched second time we used the Boolean flag field in the previous table , where we set it to True after hitting the save button in the activity , this flag checked every time the application is launched

1.2.4: After that we start the Register Activity then to the main menu, discussed in the next page

Section 2 Main Menu Activity :



Figure : Main menu Activity

**2.1 : Overview**

Is the main activity of the application, from here the user can run any of the application’s services it contains eight buttons and option menu pops up when the user hit the mobile’s menu button .

**The buttons start the following activities:**

1. OCR Translator
2. Weather
3. Global Clock
4. Translator
5. Phrases
6. Picture and Comment
7. Currency Converter
8. Places

**And the option menu Items are :**

1. Feed Back
2. Register the Program
3. Backup
4. Restore
   1. **Methodology:**
5. Before this Activity starts it will check if the application start for the first time or not ,by checking the state of the data base field “Flag” if it is false it will direct the user to the welcome screen, if it is True this activity will start .
6. When the user press one of the buttons in this activity, the corresponding activity will start.
7. By pressing the menu button of the device the option menu items will be displayed from here the user can choose one of these items to start one service.

Note : All these Services are discussed in later chapters .

Section 3: OCR Translator

* 1. **.1: OCR Translator:**

**3.1.1: OCR introducing:**

Optical character recognition serves an important function in today’s world. It allows

Existing paper documents to be easily converted into the digital domain, which enables efficient

Storage, searching, manipulation and distribution of the documents. Optical character

Recognition has other practical uses, such as helping vision-impaired people read printed

Matter. Clearly, this is a compelling software problem .

The approach taken in this section is to use OCR technique to properly extract the printed characters and symbols from images taken by the android based cell-phone’s Camera, converting them into an editable text that can be translated into other languages ,to make it possible for the travelers to translate and pronounce what is written in languages that they can’t read .

Our OCR service uses an open source Tesseract engine which is considered the most accurate free OCR engine in existence. Google has benefited extensively from Tesseract in their projects

3.1.2:Implementation Overview

We developed our own application, called ”OCR Translator ”,to capture and rectify images to feed to the Tesseract OCR engine in location and segmentation phase . OCR Translator allows the users to select an image already stored on their Android devices or use the device’s camera to capture a new image; it then goes through an image Gray scaling and thresholding algorithm ,then passing the input image to the Tesseract service. When the OCR process is completed it produces and returns a string of editable text , displayed translation activity ”OCR Translator” screen, where the user is allowed to edit the results translate to his native language or another and listening to its pronunciation

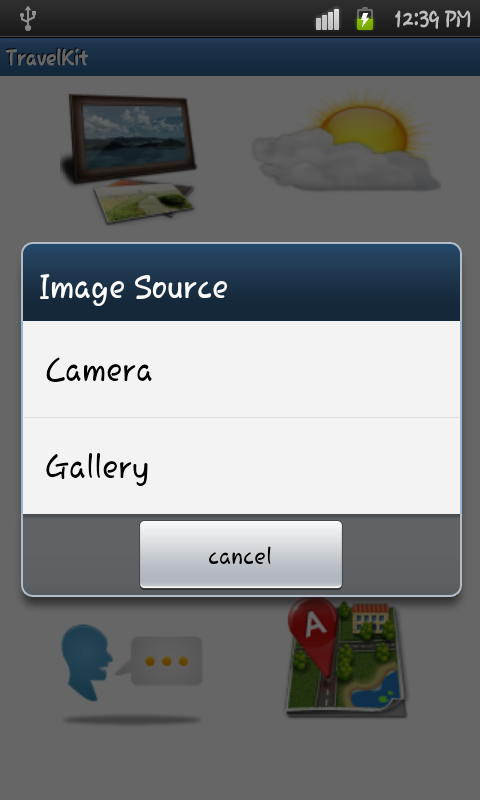


Figure Select Image source for OCR Translator

3.1.3 : OCR Translator process goes through five phases :

1. Image Capturing
2. Gray scaling and Image adaptive thresholding (Otsu’s method)
3. Location and Segmentation (using Tesseract Engine )
4. Feature extraction
5. Translate and pronounce the result

**OCR (Optical Character Recognition )**

3.1.3.1 : **Image Capturing :**

The user can select the Image to be processed by the OCR service either from his mobile SD card or from capturing the Image using his cell-phone’s camera , so that he can capture any image from a restaurant’s menu , or a traffic sign .

3.1.3.2 : **Gray scaling and Image adaptive thresholding ( using Otsu’s method)**

When performing OCR, it is common practice to convert the multilevel image into a bi-level image of black and white. This is called thresholding .

The thresholding process is important as the results of the following recognition are totally dependent of the quality of the bi-level image. Still, the thresholding performed on the scanner is usually very simple. A ﬁxed threshold is used, where gray-levels below this threshold is said to be black and levels above are said to be white. For a high-contrast document with uniform background, a pre chosen ﬁxed threshold can be sufficient. However, a lot of documents encountered in practice have a rather large range in contrast. In these cases more sophisticated methods for thresholding are required to obtain a good result.

The best methods for thresholding are usually those which are able to vary the threshold over the image adapting to the local properties as contrast and brightness. However, such methods usually depend upon a multilevel scanning of the document which requires more memory and computational capacity. Therefore such techniques are seldom used in connection with OCR systems, although they result in better images.

We developed theThresholding phase based on adaptive algorithm “Otsu’s algorithm “.

3.1.3.3 : **Location and Segmentation (using Tesseract Engine )**

Segmentation is a process that determines the constituents of an image. It is necessary to locate the regions of the document where data have been printed and distinguish them from figures and graphics. For instance . Applied to text, segmentation is the isolation of characters or words. The majority of optical character recognition algorithms segment the words into isolated characters which are recognized individually. Usually this segmentation is performed by isolating each connected component that is each connected black area. This technique is easy to implement, but problems occur if characters touch or if characters are fragmented and consist of several parts.

Tesseract Engine is written in the C++ programming language, so it is not a trivial task to use it on the Java-based Android OS. The C++ code needs to be wrapped in a Java class and run natively via the Java Native Interface (JNI). Though there is some effort involved, one great benefit to running Tesseract natively is that C++ is substantially faster than Java. After implementing our own rudimentary wrapper.

While default Tesseract settings are configured in OCR Translator”, the user has limited control over the settings via a preference panel, where processing speed intervals can be set and training data and languages can be modified .

**The main problems in segmentation may be divided into four groups:**

1- Extraction of touching and fragmented characters :

Such distortions may lead to several joint characters being interpreted as one single character

2- Distinguishing noise from text. :

Dots and accents may be mistaken for noise, and vice versa.

3- Mistaking graphics or geometry for text.

This leads to non-text parts being sent to recognition

4- Mistaking text for graphics or geometry.

In this case the text will not be passed to the recognition stage. This often happens if characters are connected to graphics

3.1.3.4 : Feature extraction

The output of the previous phase goes through feature extraction to identify each symbol

The objective of feature extraction is to capture the essential characteristics of the symbols, and it is generally accepted that this is one of the most difficult problems of pattern recognition. The most straight forward way of describing a character is by the actual raster image. Another approach is to extract certain features that still characterize the symbols .

The techniques for extraction of such features are often divided into four main groups, where the features are found from:

• The distribution of points.

• Transformations and series expansions.

• Structural analysis.

• Template-matching techniques

The different groups of features may be evaluated according to their sensitivity to noise and deformation and the ease of implementation and use.. The criteria used in this evaluation are the following:

Robustness:

1) **Noise**: Sensitivity to disconnected line segments, bumps, gaps, filled loops etc.

2) **Distortions**: Sensitivity to local variations like rounded corners, improper protrusions, dilations and shrinkage.

3) **Style variation**: Sensitivity to variation in style like the use of different shapes to represent the same character or the use of serifs, slants etc.

4) **Translation**: Sensitivity to movement of the whole character or its components.

5) **Rotation** :Sensitivity to change in orientation of the characters.

Practical use:

1) Speed of recognition.

2) Complexity of implementation.

3) Independence : The need of supplementary techniques.

We choose template-matching techniques because it is simple and easy to implement, where we compare the stored characters templates with the extracted characters by the program and choosing the one with the best match result

3.1.3.5: Translate and pronounce the result

After the text has been obtained from the processed image we send it to android activity where it can provide the user with the ability to translate the text into the language he choose and pronounce it.

We used a paid Google translator API V.2 to translate the text to the chosen language, to use this API we had to buy it first and then obtain the key they give to use to request the translation , there are two method to do the request Using REST directly or Using REST from , we used REST directly method which explained **bellow** :

**There are four parameters in the request**

1. The key obtained after buying the service from Google
2. Translate , which is the source text from source language to target language
3. Languages ,(List the source and target languages supported by the translate methods)
4. Detect , detect language of source text

The specific format for the single Google Translate API URI is:

<https://www.googleapis.com/language/translate/v2?parameters>

Where **parameters** are any parameters to apply to the query.

Here is an example of how this works in the Translate API.

https://www.googleapis.com/language/translate/v2?key=INSERT-YOUR-KEY&q=hello%20world&source=en&target=ar

And the result will be like :

200 OK // means the translate done correctly

{

"data": {

"translations": [

{

"translatedText": "مرحبا"

}

]

}

}

After that we will take the result parsing it and then display it to the user .

For more details you can go :to

<http://code.google.com/apis/language/translate/v2/using_rest.html#WorkingResults>

For the prouncounication service we used Google text to speech library (provided with android systems) to translate the text into speech.

**Note : We used the Google Translation API in a separate activity so that the user can translate any word immediately in case he wants to enter the text manually**

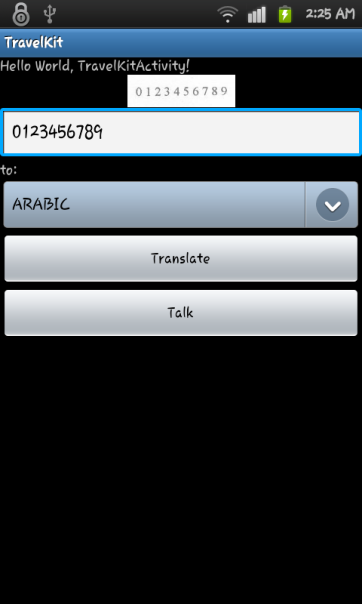


Figure OCR Translator Activity

Section 4: Server communication:

4.1 : Overview :

Most of the mobile application nowadays tends towards a client server structure .

Where this structure links the users (clients) to the service provider so they will be notified with all the changes and the new features added to the product , and making backup copy of the clients data to protect data to be lost .

Our server provides our users with the ability to back up and restore their data in this version of our product the user can backup and restore his information only ,and we have the plans to provide him the with more features such as backup the images and the phrases .

Where the users have to first register the product in the server to be able to access these service .

* 1. :Methodology :
     1. **Client side :**

We used HTTP Post protocol to communicate with our server

* 1. .1.1.Product Registration :

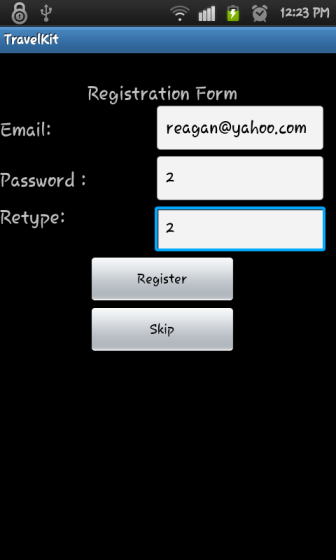


Figure : User Registration form

The user can register his product either immediately after the welcome screen which starts on the installation, or by pressing the register menu item (of the main menu ) .

4.2.1.1.A: This activity asks the user to enter his email address and password with confirmation the email address text box will be loaded by the user email address stored in the data base .

4.2.1.1.B: when the user presses the register button we send his Email , Password and his device MAC Address to the server . The MAC address used to make sure the user can’t create more than one account from his device (from one MAC), and also to guarantee that he will not create more than one account with his email address in other device .

4.2.1.1.C: After sending the information , our server will register the product and this explained in the server side section

4.2.1.1.D: After the response from the server we inform the client that his product has been registered .

**4.2.1.2: Data Backup :**

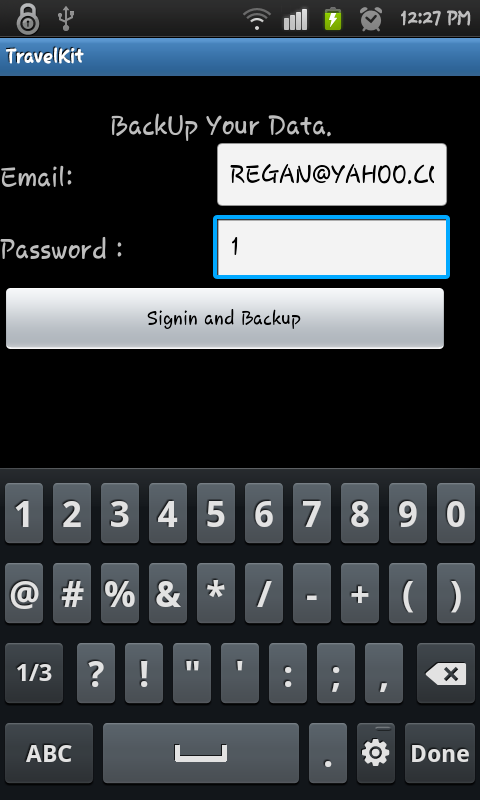


Figure 6:Backup Activity

4.2.1.2.A : For the user to back up his data he must sign in first using his email address and password and that checked on server side ,and we inform the user with the response of the server.

4.2.1.2.B : When the user pressed the backup button , his data from the data base will be uploading up to the server

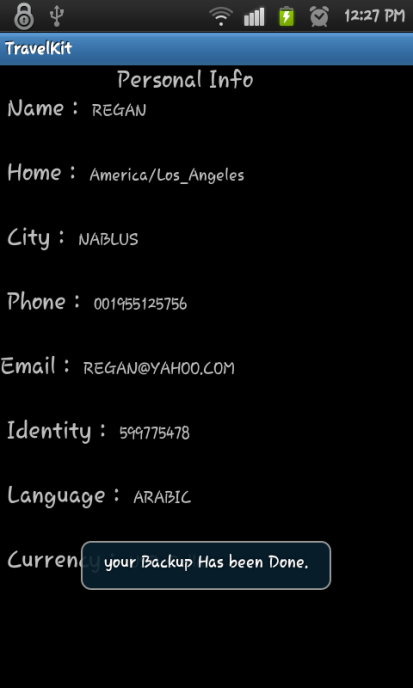


Figure : Backup success

4.2.1.**3 Data restore :**

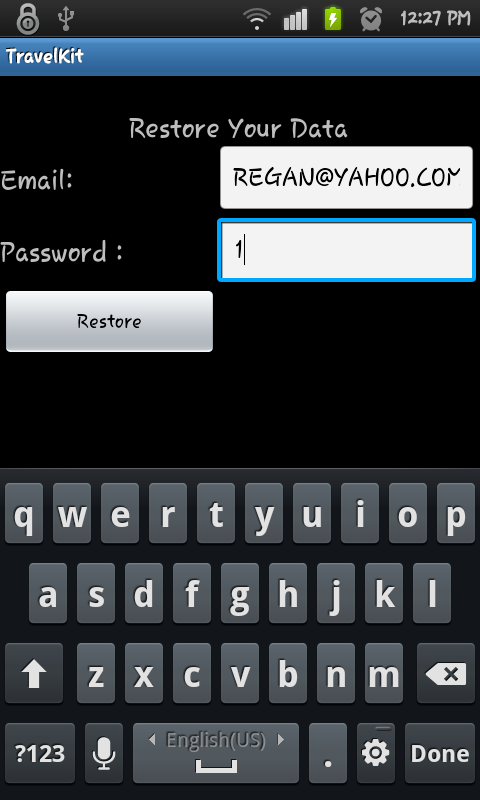


Figure :Restore Activity

4.2.1.3.A : For the user to back up his data he must sign in first using his email address and password and that checked on server side ,and we inform the user with the response of the server.

4.2.1.3.B : When the user pressed the restore button , his data will be downloaded from the server to his data base .

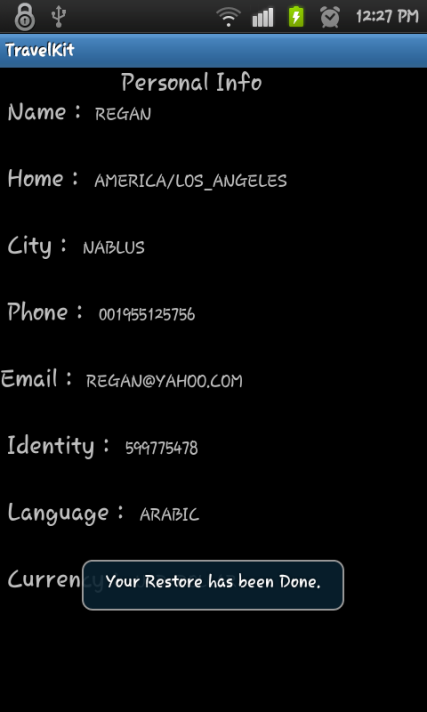


Figure : Restore done

* + 1. **: Server Side :**

The server side has four PHP pages, register, Login, backup and restore.

4.2.2.1 **Registration page**:

4.2.2.1.A: When the server receives a Registration request from the client associated with Email , Password and the MAC address , it will create a new record in the server data base in Users Info table as bellow :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Email | Password | MAC address | Backup\_Flag |

**Note** : The Backup\_Flag used in the backup step , where it is False by default

4.2.2.1.B: after creating the user record in the data base we create new folder for the each client where the name of the folder is using the ID as a unique name so that it will be like /UserFolder\_ID/

4.2.2.1.C: After creating the folder and the user record the server will send response to the client to inform him that his product has registered , and the server send a message to the user’s email

4.2.2.2 **Login Page** :

* + - When the user request to login he will send his email address and password , with the type of the Action he wants to do either backup or restore .
    - The server will check the email and password if it is correct it will start a session for the user and store the user name and the ID of the user retrieved from the server database to use it as a path name to reach to backup files . If it is false the server will inform the client side that its password is wrong or he is not registered yet
    - After the user login succeed the server will check the type of the action request by the client if it is
      * + **Restore** :

The server will check the backup flag if it is false the server will send a response telling the user that he doesn’t have backup file yet . Else it will go through the next step

If the flag is true the server will use the path stored at the session variable to reach the user backup copy and send it to the client

* + - * + **Backup** :

The server call the backup page that receives the user information sent with the backup request

The backup page will use the path obtained from the user ID that stored in session variable to locate the user’s folder on the server so in which we store the user’s data in

After saving the data the server set the backup flag to true so that the user can restore his data

Also the server will send a response to the client telling him that the backup process has done successfully

* + - After finishing the Activity the server will destroy the sessions to make sure no one will access the data except the user

Section 5: Picture and Comment

5.1: Introduction

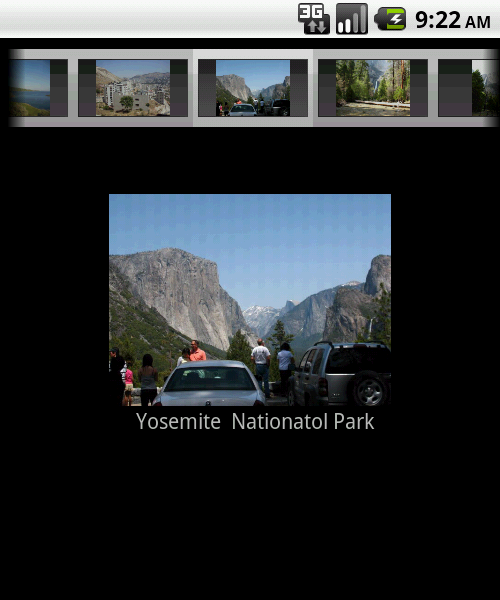


Figure : Picture and Comment Main Activity

People always take pictures during their day life in order to remember some of their life's moments and beautiful scenes that could be used as a scenario of their life, and to be a remembrance to them and their sons and they always prefer for these pictures to be texted or commented with the related historical events or the Occasions at which they were captured or even during their travel to a beautiful places and the nature sights while they are traveling around the world.

This is often to happen to take different kind of pictures especially when we bring our mobiles while we are in travel or tours.

In this service (Picture with comment) we provide the user with ability to digitally comment on his pictures and photos so that he can record his own notes and comments and each photo is there with its own noted so the user need not to worry about remembering when I took this photo? Why it was captured? Where is that place?

So, all these information about the time at which our photos were taken or in which places or Circumstances we have taken can be recorded with the photo itself on your mobile.

5.2: Implementation Overview

***This service was implemented as the following***

5.2.1: **Taking the picture**:

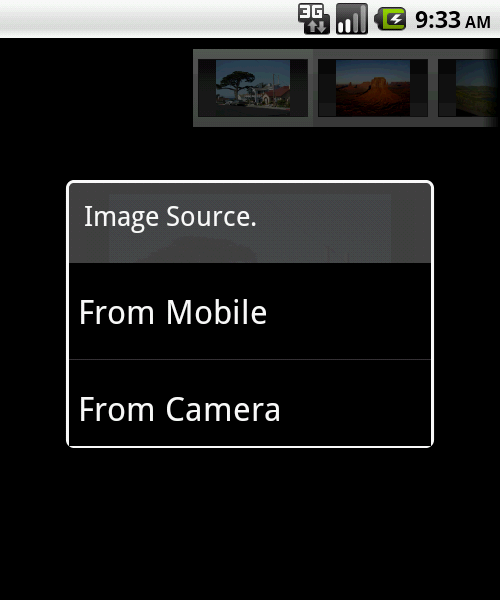


Figure 11:Select Picture Source Dialog P&C

Here when the user enter the main activity of this service we give the user the choice to capture a new photo by his cell – phone's camera or just by browsing his mobile's memory to select a pre captured image by pressing his mobile menu buttons a menu item(**Add new photo**) will give him this choice and ask him choose either use a camera to capture new image or let him browse his mobile memory and he can select any picture to move to the main activity to be involved in this service.

Here once the user click on the selected image a dialog will ask him to confirm that he will really involve this image in this service and a copy of the image will be stored in the new path location i.e. in the folder related to the application .

If the use press the cancel of that dialog the operation will be aborted, and before the user get out of this sub activity of selecting to bring a photo from his memory , he can click on another photo to bring so he can bring more than one photo and he will confirm the process for each one .

In the case that the same picture is located in advance in the folder of our application we will ask him either to overwrite the photo with the old comment or to cancel the operation and then the image will be stored in pre-defined path locations according to our implementation such that we stored all pictures that has been commented or to be commented by the user in a specified folder where all pictures related to this service are stored.

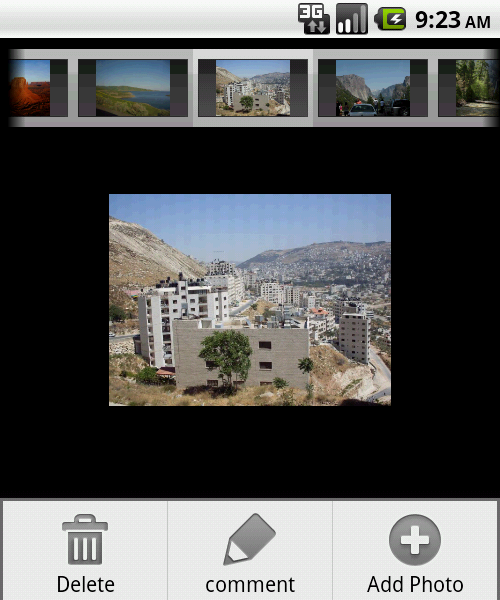


Figure :Picture Not Commented yet

5.2.2: **Manipulating selected photo:**

5.2.2.1: **Commenting new picture**

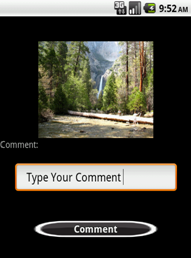


Figure : Picture Comment Activity

After that we view all pictures that has been captured by cell phone or brought in the main activity of this service in which the user can select a photo and write down his new comment on it by just pressing the comment button or by check the menu item (Comment) from the phone menu and anew activity will let him to write down his comment and save it in the database of the application.

5.2.2.2: **Updating pre commented picture (re -commenting)**

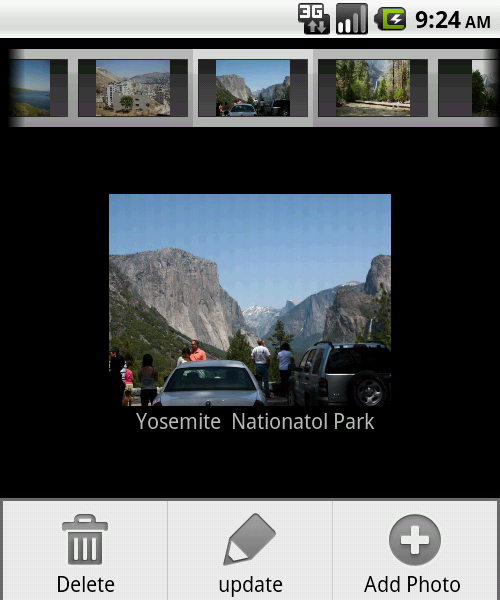


Figure :Update Comment

The user can also modify his previous comments on any picture, by selecting the photo and then either pressing the update button in the bottom of the activity or by choosing the update menu item from the mobile menu.

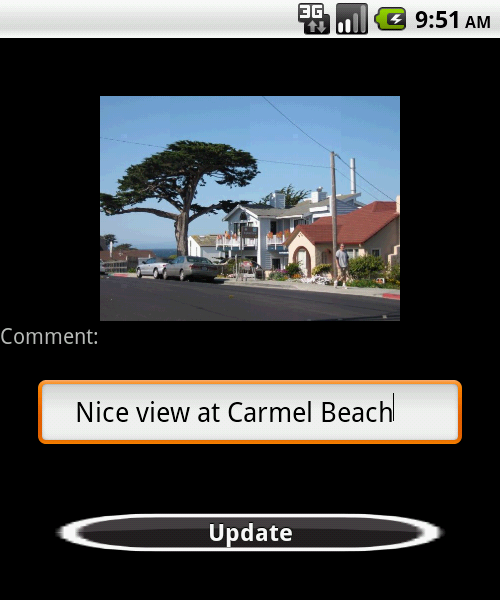


Figure :Update Picture Activity

Then the update activity will start and the user can modify his previous comment on the selected picture.

5.2.2.3: **Deleting the picture.**

We add the deletion option in our activity because we will prevent the user to delete the images from our own application folder and the user will not be able to view the content of that folder without using our application, he also can't delete the whole folder i.e. the folder will protected against any modification without our application.

The user can delete any unwanted pictures from the application folder just by selecting the image and pressing the delete button in the bottom of the activity or choosing the delete menu item and the user need to confirm this operation and upon user agreement of deletion we delete the corresponding comment from the databases.

Section 6 : Phrases

6.1 :Overview :

Phrases service provides the travelers with the most common words used in three languages ,Arabic , English , and French .and give him the ability to listen how they pronounced .

This service has three main activities (pages ) , the main activity , the add new phrase activity and the update activity .

In the main activity consist from three main parts , the phrases list , the option menu and the action dialog.



Figure : Main Activity

In the main activity the phrases are displayed to the user in a list , where the list contains the each phrase in the three languages , by pressing one phrase the service will pronounce the phrase in the language he choose . The user can choose the language from the option menu by pressing the menu button in his device

The user can update and delete the phrase from the action dialog by holding the phrase he want to delete or update , also he can add new phrase by pressing the button in the top of the list or to make it easier we included the add service in the action dialog .

6.2 : Methodology :

**6.2 .1 Reading the phrases :**



Figure : Reading Phrases

We created a phrases class that has the four variables (id, Arabic ,English , and French )

Phrases are stored in data base (Sqllight data base ) in phrases table where it consists from 4 columns:

1. ID integer (auto increment )
2. Arabic Word
3. English Word
4. French Word

Selecting all of the rows from the table using sql queues and save them into an array list of phrase class

Putting them in an adapter ,putting the adapter to the main list . Each adapter consists from three text view one for each language and has an on click listener that call the pronounce function , where we used Google Text to speech API to pronounce the word , in the language the user choose .

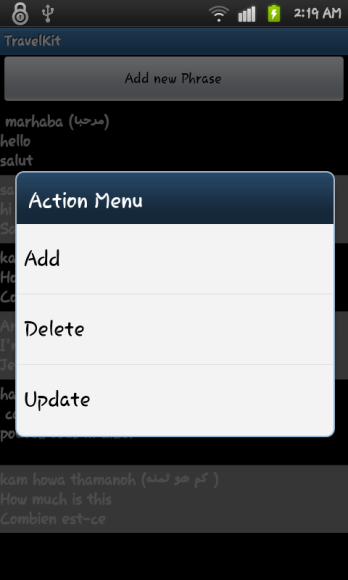


Figure : Action Menu

6.2 .2: Adding new phrase :

When the user select the adding service (using the add button or from the action dialog) an activity that has 4 edit text view with a save and listen button will appear , the user can enter the word one of these text views , and the program will translate the word he entered to the other languages (using Google translator API V.2) and fill the text views so the user can edit them if he wants , but for the Arabic word the user has to enter the word in Arabic letters (can be obtained from the translated word ) and how it sounds in English letter because the API does not support the Arabic language .

Before saving the phrase into data base the user can listen to the word to make sure that he entered the write word .

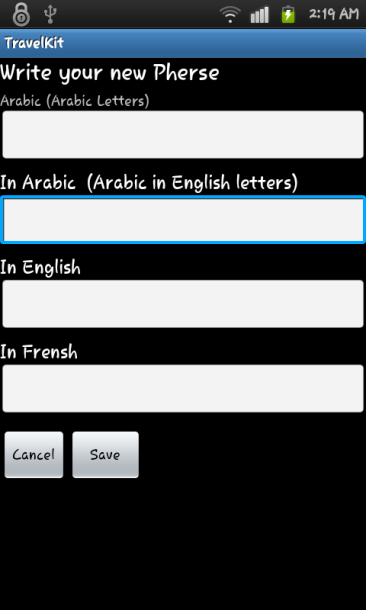


Figure : Add New Phrase

6.2 .3: Updating the Phrases :

Updating the word is similar to adding new word except the update activity will filled from the data base and the user will change the words to a new ones , similar for adding activity it has save and listen button also .

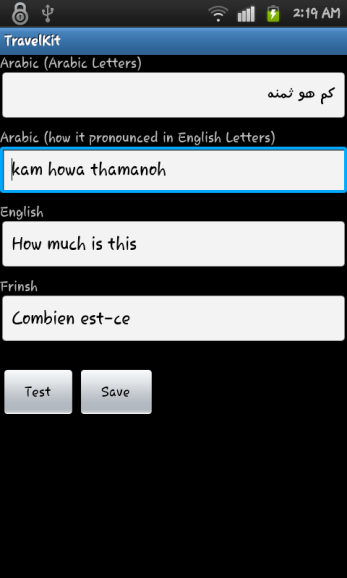


Figure : Update Activity

**6.2.4: Deleting the Phrases :**

Deleting Process is starts when the user select a phrase from the main list ,by selecting the delete option from the action menu , the phrase ID is obtained and sent to deleting activity to delete the phrase from the data base

Section 8 : global clock service

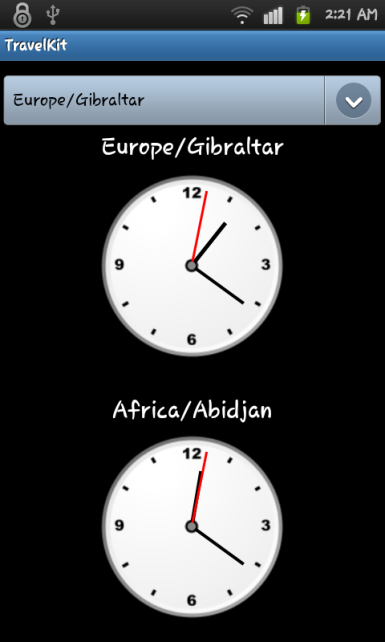
****

Figure : Global Clock

When the user travel to other county that has a time difference with his home country he might need to know what is the time back home or in other country , we provided an activity with two clocks one to show the home time and another shows the time of a country he choose from the spanner of countries in this activity , assuming the user changed his device clock to the country he visited ,the user has three clocks in his device , even if he didn’t change his device clock he can know the time by selecting the visited county from the spanner .

To display the clock of the country that the user choose, we obtained the time zones from the device ,where each android device has this array stored within it , we moved the time zones into a String array and filled the spinner using this array , when the user select one of these time zones we take the time from the time zone and by math calculations we will have the time in seconds, minuet and hours .

For the home time, we take the time zone from the data base where the home time zone stored from the welcome activity , for more information about welcome activity you can go to page 7

To draw the analog clock we must have the time zone which obtained as described above , and then using the Canvas class we will draw each clock hand , where each clock hand is a Canvas line that its parameters are obtained from these formulas :

*1- For Hours hand*

canvas.drawLine(x, y, (float)(x+(r-15)\*Math.cos(Math.toRadians((hour / 12.0f \* 360.0f)-90f))), (float)(y+(r-10)\*Math.sin(Math.toRadians((hour / 12.0f \* 360.0f)-90f))), mPaint);

*2- For Minutes hand:*

Hand canvas.drawLine(x, y, (float)(x+r\*Math.cos(Math.toRadians((min / 60.0f \* 360.0f)-90f))), (float)(y+r\*Math.sin(Math.toRadians((min / 60.0f \* 360.0f)-90f))), mPaint);

1. *For Seconds hand:*

canvas.drawLine(x, y, (float)(x+(r+10)\*Math.cos(Math.toRadians((sec / 60.0f \* 360.0f)-90f))), (float)(y+(r+15)\*Math.sin(Math.toRadians((sec / 60.0f \* 360.0f)-90f))), mPaint);

Section 9: Weather activity :

9.1: Overview :

To make it easy for the user so he doesn’t need to go to the internet entering the weather website and check the weather in certain country , we provided him with this service , where the default location’s weather is the place he is in now (the location obtained from the longitude and latitude using GPS or Wi-Fi , which is also stored at the data base when the user update his location for more information go to page ) .

9.2: Activity structure :

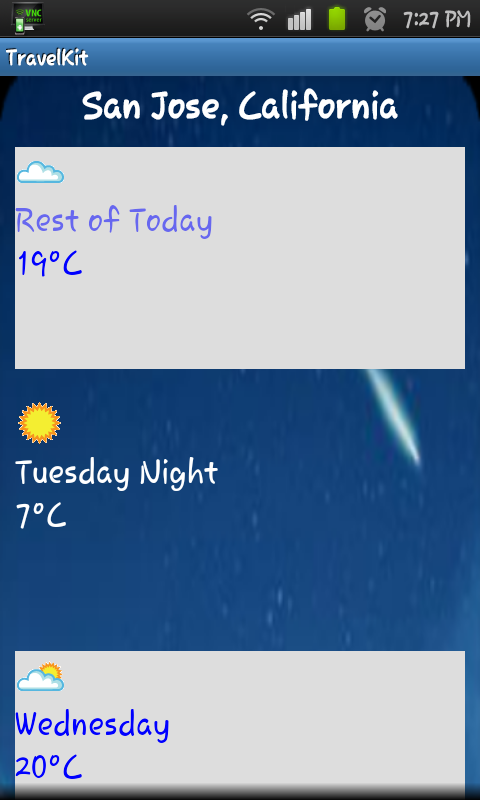


Figure : Weather Main Activity

The weather activity has two main activities , information activity where the temperature and the humidity are displayed , and the entering the location activity , where the user can enter the location name to get the weather information about it.

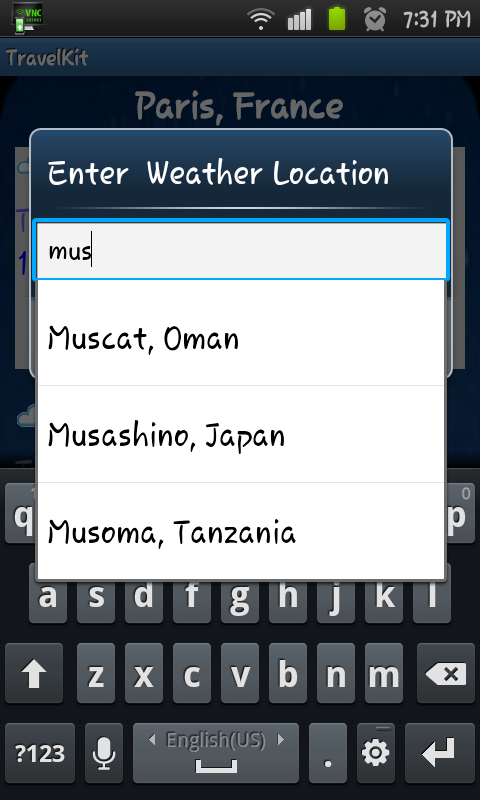


Figure : Select Weather Location Activity

We used wunderground weather API to get the weather information by sending the name of the location and retrieving the information from the server , with the auto complete text box

Section 10:Places service

10.1: Overview

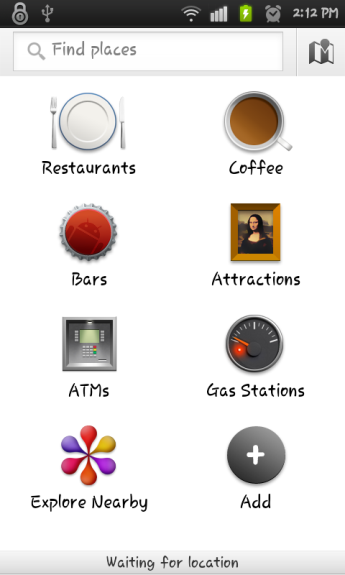


Figure : Location Main Activity

The traveler need to know where to go to sleep , to have some food or to go to a museum or ATM machine , to make it easy for him we included in our project Google service called places which is one of the services that are in Google maps package .

10.2: Methodology :

1. Checking if the user has the places application in his device by searching the packages names that installed in the device .
2. If the application is not installed you will open the link of the application in the android market to install It .
3. From there the user can enter the name or the type of the place he is looking for and the program will list the places according to the nearest place or other specification the user choose

Section 11 : Currency convertor

11.1: Introduction

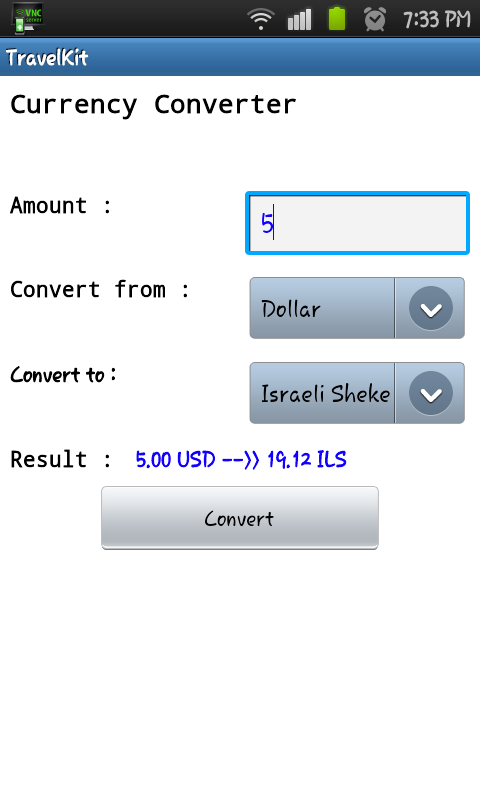


Figure : Currency Converter Activity

People who travels very frequently like Travelers and tourists are always travel to countries that have different currencies.

We know that almost each country has its own accredited currency, moreover some countries have more than one credited currency.

And for those people to do their Business transactions or buy anything while they are in those countries, they sometimes face some difficulties to deal with those new currencies and conversion between them.

11.2: Implementation

So they actually needs a way to convert between currencies and for that reason we have developed a Currency convertor based on currency convertor web service of <http://www.webservicex.net//CurrencyConvertor.asmx>.

And we use an http post connection protocol to connect to the web service and getting an up to date conversion rate between the two currencies based on user choices.

11.2.1: Steps of operation

1- First our user will write in the text box the amount of units of the original currency that need to be converted.

2-Now our user need to choose using a spinner or a combo box the currency to convert from and using another spinner he will be able to choose the one to convert to .

3- Then the user will obtain the result of conversion after he press the convert button.

Here in this service we obtain the up to date conversion rate between the needed currencies using the http post protocol as the following:

3.1- First we send an http post request as the like that:

POST /CurrencyConvertor.asmx/ConversionRate HTTP/1.1

Host: www.webservicex.net

Content-Type: application/x-www-form-urlencoded

Content-Length: length

* 1. Then we get an http pose response from the web service itself and we use it and parse it in order to obtain the actual conversion date which is need to be up to date as the web service provide it to us.

FromCurrency=string&ToCurrency=string

HTTP/1.1 200 OK

Content-Type: text/xml; charset=utf-8

Content-Length: length

<?xml version="1.0" encoding="utf-8"?>

<doublexmlns="http://www.webserviceX.NET/">double</double>

* 1. Now after we get the needed rate we use it to make the currency conversion as the user needs and we view the result for the user on the activity so that the user can see the result of his conversion.

Section 12: Update My Location :

12.1: Overview :

Get location is a service that retrieves the longitude and the latitude of the user’s location , converting them into the city name .

There are three ways to retrieve location information :

1. GPS
2. Cell-ID
3. Wi-Fi

We used GPS and Wi-Fi to retrieve the location’s longitude and the latitude. We used both GPS and Wi-Fi methods because GPS gives more accurate coordinates than Wi-Fi , but it doesn’t work inside doors , thus if the user is inside doors or his device doesn’t have the GPS service he can use Wi-Fi method which is give less accurate coordinates but this information are enough to convert them into city name .

12.2 Methodology :

1. The First method used to retrieve the location information is by GPS , but if that is not possible either because the user is indoors or his device doesn’t support the GPS service we used Wi-Fi.

When the user chooses update my location from the main menu of the application the application will check the possibility of using the GPS method .

1. **Using GPS :**

**A**: To use GPS service of the android smart phone you need to get the permission to access this service and this can be done by including this code in the manifest file of the application

<uses-*permission android*:name="*android*.*permission*.INTERNET" />

**B**: Retrieving the longitude and the latitude of the location is done by using the android Location manager class , you can review the code of this step in the attached codes

**C**: After having the longitude and the latitude the name of the location retrieved by using the Reverse Geocode from Google map’s API . By giving the location coordinates to” requestLocationUpdates” function .To do that you need to have the access to the internet service in the device by using this code in the manifest :

<uses-permission android:name="android.permission.INTERNET"></uses-permission>

1. **Using Wi-Fi :**

**A**  .Using the skyhook API ,and to use this API you need to have your own realm and a username ,which can owned by register their website .

**B**  . by entering the realm and username you can ask for the information of your location such as location name .

1. After retrieving the location name and the longitude and the latitude they will be stored in the data-base in a table called Info table , to be used in the weather activity as a default location

Table \_User Info

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Name | Home | Current city | Phone No. | Email | Native Language | Default Currency | ID Number | Flag | Longitude | Latitude |

Section 13 : User Personal Information:

Activity overview:

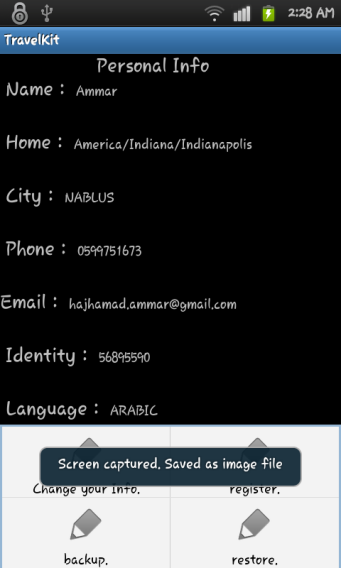


Figure : Personal information

**1-** This activity is implemented in order to provide the user with the ability to change his frequently changed information such as his current location, phone number, his email address.

This edit text boxes and spinners of this activity are loaded with the information that entered by the 0user in the welcome screen and stored in the database and this information is:

Table : User Info

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Name | Home | Current city | Phone No. | Email | Native Language | Default Currency | ID Number | Flag | Longitude | Latitude |

1. Name.
2. Home.
3. City (current city location)
4. Phone Number.
5. Email Address.
6. Native Language.
7. Default Currency.
8. ID Number
9. Flag.
10. Longitude.
11. Latitude\*.

**2**- The user needs to enter the values for each Name,Phone Number,City, and Email Address, and he will select one of the values viewed in the spinners (Check Boxes) for each of the Home,Default Currency and Native Language.

**3**- We load the spinner of Native Language with the values stored in a xml file contains all the international languages using a special spinner adaptor , and the spinner of currency is also loaded with the values stored in another xml file that contain all international currencies names using a special spinner adaptor.

So that the user can select any language to be used as his native one from the spinner and choose any currency code to be used as his home currency code.



Figure :User Information Update Activity

**4**- After which the user has entered his info and pressed the update button, we performed some validation on the entered info like the email address that come in its special format, and we also check if the user has left empty values of any of his info, and upon that we enforce him to enter all needed info in the correct format, then the new entered info is stored in the database and the old one is modified with the new info.

**Note**: Longitude and Latitude are entered automatically by the application based on the GPS Service and the user is unable to enter or modify their values.

Section 14 :Sending Feed Back Email:

14.1: Overview

We give the user the ability to contact us using the email feedback activity ,such that he can provide us with his impression about the application and he can send his comments any desires about the application and this point may help us to modify our current specification or add new services if more users have request.

Moreover, the user may misunderstand the some details of some services we provide and he can ask for some further illustration during this activity.

14.2: This activity consists from :

1. A multiline Edit text in which the user can write down his request, impression and his feedback about the application.
2. A send button that the user press after he finish his feedback body.
3. Then we send the mail body entered by the user to the default android mail application.
4. And we put our email address company as a default addressee and the subject as the Feedback subject.
5. Then we respond to the user on his email address with our response.

Conclusion

Reliably interpreting text from real-world photos is a challenging problem due to variations in environmental factors. Even the best open source OCR engine available (Tesseract) is easily thwarted by uneven illumation, off-angle rotation and perspective, and misaligned text, among others.

Smart phones that supported android or other Operating systems can be used to develop a lot of useful applications that serves the user in his daily life activities

People can use our application as a tool during their English , Arabic and French languages using the Phrase ,Translation and pronunciation provided services

Future work :

1. Upgrade Serve Services
2. Providing more languages for OCR and pronunciation
3. Provide the interface with other languages
4. Adding more services for the travelers

References and Bibliography

**Websites :**

<http://www.developer.android.com/>

<http://code.google.com/apis/language/translate/v2/getting_started.html>

<http://code.google.com/p/tesseract-android-tools/>

<http://www.webservicex.net/ws/wsdetails.aspx?wsid=10>

<http://www.stackoverflow.com>

**Books**:

Professional Android Application Development , Reto Meier (book )

Professional Android Application2 Development , Reto Meier (book )

Pro Android 2 Sayed Hashimi , Satya Komatineni and Dave MacLean (book)

**Articles** :

* Image diﬀerence threshold strategies and shadow detection , Paul L. Rosin and Tim Ellis
* OCR Optical Character Recognition , Line Eikvil
* Discovery of Optical Character Recognition Algorithms using Genetic Programming , Polina K. Spivak