Utilizing Andoid for Multipurpose Application Development

Adila Karim, Athira Unnikrishnan, Julna Nazer
UG Scholars
Department of Computer Science and Engineering
M.E.S College of Engineering
Kuttippuram, India

Abstract— ***Mobile computing is the technology by which a user utilizes the capabilities of his/her mobile phones to get their jobs done. Increase in the use of mobile computing has led to an explosion in app development. Android OS is seen as the most advantageous of the platforms for app development. We propose the use of Android to develop a multipurpose app that takes care of the day to day activities of a user. In order to demonstrate the utilization of Android for multipurpose app development, we propose a new application named REGISTROID. This application is first of the form, and bundles diary along with daily photo addition, to-do lists, reminders and an expense tracker. This paper studies how the use of REGISTROID can simplify the user's experience by hosting 5 different applications into one. REGISTROID works as one personal journal and is all one needs to lead an ordered life***.

Keywords—Android, application, multipurpose app, REGISTROID.

I. INTRODUCTION

People are always on the move, keeping up with the time constraints of the work. It is the ease with which one can complete their jobs on the move that has been the reason behind the growth of mobile computing. Technically, mobile computing can be said as the interaction between the devices and the human beings in the course of getting their tasks done. Utilizing the capabilities of their mobile phones, they can access their data, do manipulations, upload or share it or do anything they need. This has been a boon to many busy individuals. The major advantages [3] of mobile computing are

- Productivity growth: Being able to use the capabilities of the devices, one can complete the work on the go. This means that no time is wasted and the productivity level increases. The necessary work can be done anytime, anywhere, provided the requirements are available.
- Increased portability: mobile computing helps one to be on the move and still never missing out any of one's tasks.
- Link up with cloud computing: This means that the users can save the data on an online server and be able to access the same whenever they need.
- Entertainment opportunities: Mobile computing serves entertainment purposes and can even be used to share and produce notes and presentations.

II. ANDROID OPERATING SYSTEM

The popularity of touch screen devices has exploded unbelievably in the recent years. Android operating system can be stated as the reason behind the rising popularity. Android is strengthened by the open source nature of the operating system. Open source nature when added with the advantage of the flexibility it provide to the developers a platform that is affordable. The plethora of features available that helps the developer customize the application to be developed is also an added blessing.

A. Features of Android

- Increased share in the market: Many other carriers have been outlived by Android in the market. The Android Market preinstalled devices ensure that the users just need to search for the application they need in the store. Users have the access to all the applications that are presented in the store. This means that the user can save the time they spent in the internet searching for the right app.
- Open source: Android is an open source software. This Linux based operating system is not the property of a single provider, it can be associated with all the developers. Open source nature of Android is the reason behind the increased share in the market.
- Capability of mash-up: Multipurpose app development benefits a lot from this feature. Mash up means that different application can be combined together to produce new outputs e.g. Location service offered by Android can be combined with the notes written to know the exact location where the note was written. This will help when writing travelogues. Combining different services in new and different ways create exciting possibilities for multipurpose app development.
- Time to market: All the defined features in the operating system results in short time span for the app development. Time to market thus is low as all an app developer needs to do is create and then launch the application.

- Compatibility: Despite the varying size of displays and the resolution they showcase, Android can be used to develop applications on all these platforms. Android also comes bundled with the services that helps the developer create applications that are cross compatible.

B. *Android Applications*

Applications in Android are written in the Java programming language, but with a difference [2]. Instead of using Java Virtual Machine (JVM) to execute the code, with Android we use Dalvik Virtual Machine (DVM). DVM converts the code for the applications and executes the Java byte code. Android application code needs to be converted to be compatible with the DVM before it can be run on devices hosting Android. The Dalvik Executable files are the results different Java files[1]. DVM was developed with the mobile infrastructure in mind. Helping the capabilities of the devices that are lightweight, DVM was developed to boost up app development for mobile devices.

C. *Android as a Platform*

Android is developed on Linux and can be stated as a software stack. All the required tools and the Application Program Interface (API) are provided by the Android Software Development Kit (SDK). As the Android was built on Linux [1], the base layer of the stack is constituted by Linux. The middle layer is populated by Libraries and Android runtime management components. The Android App Framework comes above this layer which in turn comes below the top layer.

D. *Developing Applications in Android Framework*

To help start developing an Android application, we have all that we need that is provided by the Android SDK. It bundles all the necessary libraries and the API. The mash- up feature ensures plethora of possibilities. When permission that is appropriate is granted to different components, apps can share data with one another securely [1]. The Android app framework provides all the services. The components in the application framework are activity manager, window manager, content provider, view system, package manager, resource manager, location manager and notification manager.

By clicking on the app icon on the home screen, what the user does is that they send request a call to startActivity().This is the call for the activity manager. It is also a service that provides navigation throughout the complete application [5].
Window Manager part of the framework decides which part of the component in the display goes where. This is responsible for what a user sees and ensures that the display is not obscured by any other decorations [7].

Content provider is that part of the framework that has the access to large repository of data [6]. It works with the data of the application and provides the same to other apps when in need. Often, it has its own user interface and is the source of secure app data access.

Different views that the developer creates to constitute the app are all part of the view system. The app basically is navigation through all the views that are provided [7]. Package Manager is the framework component [7] with which the developer can navigate around the installed packages. The packages and the permissions to be obtained are all handled by the package manager. The devices that host these apps are ultimately a phone. So there needs to be a part that handles all the telephony related process. Telephony system part does just that. The details related to calls, the listening service are all handled by this part of the framework.

Resource Manager, like the name says, manages all the resources [7]. System location services are part of this class. This enables the app developer to get the details about the geographical location of the system and use it for the development of the app. Notification Manager part of the app framework is dedicated to letting the user know in case of any events that has occurred. The notification can be a persistent icon, an alarm sound or by flashing lights.

E. *Android App Models*

Android application model constitutes the components that make up the application [1]. It is the smooth working of all these components that a good app is born. The different components are activity, service, receiver and provider.

The app is a series of views one after another. The navigation from one to another is the result of the activity. Any process that is initiated is called an activity. Service is the component that is used for communication interface and is the component that remains active when the windows are changed. Receiver, like the name implies, listens for any event and in case of it, it does the corresponding actions. Provider component provides data that are needed for the communication between different applications. Like its name, it provides the necessary data.

III. EXISTING SYSTEM

App development has seen tremendous increase that the user can find an app for any of his needs. Different providers offer different platform to showcase their application. Google Play is where we can find all the Android based applications. They are so flexible that nowadays they can even be downloaded in laptops and tablets. App developers have only thought about exploiting the features of Android for fine-tuning their applications that are dedicated to a purpose. This is the mindset that needs to be changed. The disadvantages of the existing system can be overcome by developing multipurpose apps.

For journal keeping purposes, the apps available are again large. The digital diary applications available in the store serve the purpose well. So is the case with the other journal apps like the to-do lists, reminders and the expense tracker. The expense tracker available in the store now is a dedicated app that is developed with a fixed or variable income in mind. The application works as for an earning individual.

A. *Disadvantages*

The present scenario is such that the categories are large. Each category has to itself thousands of different apps. But the problem is that each of these apps is dedicated to a single purpose. Sure, there are apps that do a thing or two. But, when it comes to journal keeping apps, this is a serious disadvantage.

The digital diary applications that populate the store do have daily photo sync option, but these photos do not have their own existence. They are tied to that particular diary entry. If these photos are to be viewed, those particular entries have to be approached.

Reminders that exist provide a tag to label the reminder and remind the users at the defined time or day. These reminders are not familiar or are not comfortable with the different categories of reminders that can exist. Anniversary r schedule is all they provide for classification.

To-do lists that populate the stores act just like the reminder apps. They remind the user to do particular tasks just like a reminder. The distinction here blurs.

Expense trackers are built on the basis on a fixed or variable income. We do not find a provision to add different bank account and the amount we withdraw for our expenses. There is a need for a flexible expense tracker.

IV. PROPOSED SYSTEM

REGISTROID is a single integrated multipurpose app. In this app, we have integrated five different apps into a single personal app. The intention behind the integration is to give the user all that he needs to keep a register, under a single roof. Digital diary keeps the record of each day of the user's life and allows photo addition to describe the day. It does not allow the users to edit or delete past entries. The current date can be picked and be edited to create the diary entry for the day.

Expense tracker takes care of the task of keeping a register for all the in out flow of the money that the user has. The first screen displays two options – add account and view my account. Add account gives the user the provision to add details as to show how he spent or lent the money. The view account option lists all the expenditure details, provide pop-up messages to indicate insufficient funds in the bank account and wallet.

Reminders can be viewed as lists. Users are allowed to add new reminders in each category. The categories are user defined and is set according to his/her preferences. The settings include the option to set alarm time and also date at which the reminder needs to remind the user. Reminders no longer needed, can be deleted. We have provided a pop-up message which says 'A task needs to be reviewed'. This ensures more privacy.

To-do lists are implemented in the boxed list format. The tasks are shown in the order in which they are added in a checked list format. A task is considered to be done once the box is ticked and it no longer appears in the list view.

Photos that are added to each diary entry are taken from the gallery. The photos added in this manner find their way to the photo application which is a part of REGISTROID. These photos can be viewed as a slideshow.

A. *Advantages*

REGISTROID ensures much more security and privacy since it provides password. Through this app, we have tried to overcome the disadvantages of the existing system to a limit. We have made the app user friendly and attractive. It can be used by anyone. Through a single click, the main features come closer to the user. In this app, each photo in the diary entry can be viewed as slideshow. Categorized view of the reminder is possible. To-do lists pops-up at certain intervals. We can set the intervals for the pop-ups. In expense calculator, the users have the provision to add his/her corresponding bank account details. These features ensure the user a personal effect.

V. MODULE DESCRIPTION

REGISTROID contains five modules, each of which is independently executing apps. The modules are digital diary, expense tracker, smiles, to-do and reminders. The modules are divided so as to complete each part of the application in its full form.

A. *Digital Diary*

Digital diary, with its attractive user interface intends to keep the record of each day of user's life. The provision to add photos to describe each day is provided. The diary part of REGISTROID is such that users cannot edit or delete past entries. They can only just view what they had entered on previous days. The current date can be picked from the calendar and can be edited to create the diary entry for the day.

Photos for the diary description are taken from the phone gallery. For a day, user can add as much photos as he wants as long as it's available in the phone gallery. There is no limitation for the user to add the photos and diary description of the current day. That is, once written the current diary entry can be modified until the next day. All the entries of the user are stored within the application and can be viewed as a list. This is done by taking the advantage of the list view provision in Android. Through REGISTROID, the photos get an individual existence and goes into the photo application part – "Smiles".

B. *Expense Tracker*

Expense tracker takes care of the task of keeping a register for all the in and out flow of cash. The app is designed such that the user can add the amount he has in the bank account and withdraw from that account into his wallet for his daily expense. The first screen displays two options- add account and view my account. Add account gives the user the provision to add details as to show how he spent or lent his money. With each spent and lent entry, the amount from the wallet gets decreased. The minimum amount that the user can keep in his wallet is predefined. The view account option lists all the expenditure details. The list view of the expenditure details comes with the current and bank status. Pop-up messages indicate insufficient funds in the bank account and wallet.

C. *Reminder*

Reminder part of REGISTROID is designed such that the user can set categories of his reminders. This option is provided since the categories that each individual has differs from person to person. Users are then allowed to add new reminders in each category. The settings include the option to set alarm time and date at which the reminder needs to do its task. Reminders no longer needed, can be deleted. A pop-up message which says “A task needs to be reviewed” will be shown. This ensures more privacy and makes sure that the users are constantly reminded until the given date is over. The reminders which are not deleted are shown to the user in the form of lists from where if its no longer needed, can be deleted.

D. *To-do Lists*

To-do lists are implemented in a boxed list format. Tasks that a user has for the day are entered in a text box and saved as the same. Once the entered task is saved, it goes to the checked list that indicates all the tasks that is not yet done. The tasks are shown in the order in which they are added in the checked list format. A task is considered to be done once the box is ticked and it no longer appears in the list view. In short, the task that the user entered appears in the checked list until the box is ticked and hence considered to be done.

E. *Smiles*

Smiles is dedicated to the photos that were added to each entry of the diary by the user. These photos that were added to each entry were taken from the galley. The photos thus added find their way to Smiles. This app makes sure that the photos added to dairy entry has an individual existence as photos and is not restricted to its corresponding entries. The photos can be viewed as a slideshow.

VI. SYSTEM IMPLEMENTATION

The REGISTROID kicks off with a splash screen. The timeout is given as 5 seconds. Once the splash screen disappears, the user is confronted with a screen that hosts five icons each representing each part of REGISTROID. The diary part, if clicked, is kicked off with a screen that displays a calendar with the current date represented in a different color. On clicking the particular date, the user is navigated to the page where he can enter that day’s entry. In case he clicks other dates, he will be directed to the entry for the day. He is restricted from editing previous entries. There is an option added in the entry page to add photos that describe the entry. The add image button directs the user to the gallery. The added photos automatically go to the ‘Smiles”, which is the photo gallery part of REGISTROID.

To-do lists are implemented in a boxed list format. Tasks that a user has for the day are entered in a text box and saved as the same. The tasks are shown in the order in which they are added in the checked list format. Smiles is dedicated to the photos that were added to each entry of the diary by the user. These photos that were added to each entry were taken from the galley. This app makes sure that the photos added to dairy entry has an individual existence as photos and is not restricted to its corresponding entries. The photos can be viewed as a slideshow. Reminder part of REGISTROID is designed such that the user can set categories of his reminders. This option is provided since the categories that each individual has differs from person to person. Users are then allowed to add new reminders in each category. The settings include the option to set alarm time and date at which the reminder needs to do its task. Reminders no longer needed, can be deleted. A pop-up message which says “A task needs to be reviewed” will be shown. This ensures more privacy and makes sure that the users are constantly reminded until the given date is over. The reminders which are not deleted are shown to the user in the form of lists from where if its no longer needed, can be deleted.

Expense tracker kicks off with options add account and view my account. Add account option lets the user add the amount he has in his account. Withdrawals from the account go to wallet. It is from the wallet that further expenses are deducted. Users can tag each and every expense. There is also a provision to add the amount that user lends to his friends. Hey can tag this too. The view option offers a summary of the transactions that the user has to his benefit.

VII. PERFORMANCE ANALYSIS

When analyzing the performance of dedicated apps and those that are multipurpose, the advantages of multipurpose apps are huge. Photos that are a part of the REGISTROID can be treated as an object and thus due to the multipurpose nature, a single photo added to the diary entry gets an individual existence. This was not the case in the existing system. When we take the case of reminders the multipurpose app gets more points because it lets the user customize the categories according to their preference. The dedicated apps provide predefined categories and then let the user add the entries. Multipurpose to-do part of REGISTROID offers pop-up at regular intervals to constantly remind the user. This is in contrast with the dedicated app scenario. The expense tracker has the biggest advantage of all. It is first of the for m to be embedded in a multipurpose app. Hosts all the features needed to keep a complete journal.

VIII. CONCLUSION

REGISTROID is an integrated multipurpose app. It takes advantage of all the features that Android provides in developing a multipurpose app. The integration gives the user a perfectly personal useful app. Housing five different apps into a single app, a lot of time gets saved and thus becomes popular with the user. All the main features are made available through a single click. The paper addresses the disadvantages of not taking the helpful features of Android seriously. The mash up feature of android is taken to its best advantage.

Future expansion of the app includes automatic updating via the cloud and direct link with the corresponding banks. The link ups with the corresponding bank will help in up to date notifications from the bank.

REFERENCES

1. Suhas Holla and Mahima M Katti "Android based mobile application development and its security" in International Journal of Computer Trends and Technology, vol. 3, pp. 486.
2. Android Architecture 2010[R/OL]. http://www.cnmsdn.com
3. Advantages and Disadvantages of Mobile Computing http://mobilecomputingproject.wordpress.com /2012/10/11/advantages-and-disadvantages-of-mobile-computing/
4. 3G Mobile Terminal Development Trend of the operating system [M/OL]. http://pda.c114.net/32/c4948.html, 2007 .
   Activity Manager http://anatomyofandroid.com/2013/10/16/activity-manager
5. Content Provider
6. http://developer.android.com /guide/topics/providers/content-providers.html
   Android Development Glossary
7. http://developer.android.com
8. J. Dongjiu Geng, Yue Suo, Yu Chen, Jun Wen, Yongqing Lu, "Remote Access and Control System Based on Android Mobil Phone", vol.2. Journal of Computer Applications, 2011, pp. 560-562.
9. J. Li Lin, Changwei Zou, "Research on Cloud Computing Based on Android Platform," vol.11. Software Guide, 2010, pp.137-139 .
10. J. Burns." Mobile application security on Android". Blackhat, 2009.
11. B. Chess and G. McGraw. "Static analysis for security. Security & Privacy", IEEE, 2(6):76{79, 2004.