**u n i q** technologies

**Internship Report on**

**ANDROID APPLICATION DEVELOPMENT**

Submitted By : S.AARTHI MAHALAKSHMI

N.NIVEDITHA

**CANDIDATE NAME : S.AARTHI MAHALAKSHMI**

**N.NIVEDITHA**

**COLLEGE NAME : MADRAS INSTITUTE OF TECHNOLOGY,ANNA UNIVERSITY**

**DEPARTMENT : COMPUTER SCIENCE AND ENGINEERING**

**DOMAIN OF INTERNSHIP : ANDROID APPLICATION DEVELOPMENT**

**DURATION : 15 DAYS**

**PROJECT NAME : KIDZ WORLD**

STUDENT PROJECT GUIDE

**ANDROID :**

**Android** is an [operating system](http://en.wikipedia.org/wiki/Mobile_operating_system) based on the [Linux kernel](http://en.wikipedia.org/wiki/Linux_kernel) with a [user interface](http://en.wikipedia.org/wiki/User_interface) based on [direct manipulation](http://en.wikipedia.org/wiki/Direct_manipulation_interface), designed primarily for [touchscreen](http://en.wikipedia.org/wiki/Touchscreen) mobile devices such as [smartphones](http://en.wikipedia.org/wiki/Smartphone) and [tablet computers](http://en.wikipedia.org/wiki/Tablet_computer). The operating system uses touch inputs that loosely correspond to real-world actions, like swiping, tapping, pinching, and reverse pinching to manipulate on-screen objects, and a [virtual keyboard](http://en.wikipedia.org/wiki/Virtual_keyboard). Despite being primarily designed for touchscreen input, it also has been used in televisions, [games consoles](http://en.wikipedia.org/wiki/Games_console), [digital cameras](http://en.wikipedia.org/wiki/Digital_camera), and other electronics.

Recent Statistics seem to indicate viral growth of the android platform, it accounts for 78 % of all smartphone sales as shown above, android being Open Source offers numerous benefits to Developers and OEM’s over Closed Source Alternatives

The availability of numerous Low-Cost Android devices and extensive developer documentation and a huge Developer Community make it an attractive choice for customers and developers alike.

**DAY 1: SOFTWARE/TOOLS INSTALLATION**

Eclipse

Android SDK

Java JDK

**DAY 2: Overview Of Software :**

**The Software employed in our project for the development of the android application is the Java- ECLIPSE.**

* What is Eclipse?

In computer programming, **Eclipse** is an integrated development environment (IDE). It contains a base workspace and an extensible plug-in system for customizing the environment. Written mostly in Java, **Eclipse** can be used to develop applications.

* Purpose of using an IDE

An **integrated development environment** (**IDE**) or **interactive development environment** is a software application that provides comprehensive facilities to computer programmers for software development. An IDE normally consists of a source code editor, build automation tools and a debugger. Most modern IDEs offer Intelligent code completion features.

Some IDEs contain a compiler, interpreter, or both, such as Net Beans and Eclipse; others do not, such as Sharp Develop and Lazarus. The boundary between an integrated development environment and other parts of the broader *software development environment* is not well-defined. Sometimes a version control system and various tools are integrated to simplify the construction of a GUI. Many modern IDEs also have a class browser, an object browser, and a class hierarchy diagram, for use in object-oriented software development.

* Use of Eclipse

In [computer programming](http://en.wikipedia.org/wiki/Computer_programming), **Eclipse** is an [integrated development environment](http://en.wikipedia.org/wiki/Integrated_development_environment) (IDE). It contains a base [workspace](http://en.wikipedia.org/wiki/Workspace) and an extensible [plug-in](http://en.wikipedia.org/wiki/Plug-in_(computing)) system for customizing the environment. Written mostly in [Java](http://en.wikipedia.org/wiki/Java_(programming_language)), Eclipse can be used to develop applications. By means of various plug-ins, Eclipse may also be used to develop applications in other [programming languages](http://en.wikipedia.org/wiki/Programming_language): [Ada](http://en.wikipedia.org/wiki/Ada_(programming_language)), [ABAP](http://en.wikipedia.org/wiki/ABAP),[C](http://en.wikipedia.org/wiki/C_(programming_language)), [C++](http://en.wikipedia.org/wiki/C%2B%2B), [COBOL](http://en.wikipedia.org/wiki/COBOL), [Fortran](http://en.wikipedia.org/wiki/Fortran), [Haskell](http://en.wikipedia.org/wiki/Haskell_(programming_language)), [JavaScript](http://en.wikipedia.org/wiki/JavaScript), [Lasso](http://en.wikipedia.org/wiki/Lasso_(programming_language)), [Natural](http://en.wikipedia.org/wiki/NATURAL), [Perl](http://en.wikipedia.org/wiki/Perl), [PHP](http://en.wikipedia.org/wiki/PHP), [Prolog](http://en.wikipedia.org/wiki/Prolog), [Python](http://en.wikipedia.org/wiki/Python_(programming_language)), [R](http://en.wikipedia.org/wiki/R_(programming_language)), [Ruby](http://en.wikipedia.org/wiki/Ruby_(programming_language)) (including [Ruby on Rails](http://en.wikipedia.org/wiki/Ruby_on_Rails) framework), [Scala](http://en.wikipedia.org/wiki/Scala_(programming_language)), Clojure, [Groovy](http://en.wikipedia.org/wiki/Groovy_(programming_language)), [Scheme](http://en.wikipedia.org/wiki/Scheme_(programming_language)), and Erlang. It can also be used to develop packages for the software [Mathematica](http://en.wikipedia.org/wiki/Mathematica). Development environments include the Eclipse Java development tools (JDT) for Java and Scala, Eclipse CDT for C/C++ and Eclipse PDT for PHP, among others.

* Android SDK

The Android SDK provides you the API libraries and developer tools necessary to build, test, and debug apps for Android.

**DAY 5: PROJECT DESCRIPTION WITH MODULES**

Modules Involved:

* Module 1 : LAYOUT DESIGN :

The basic idea of the application to be developed was discussed among the team members and explained to the project coordinator.Layouts for the application were designed accordingly. The various elements of the layout like buttons,image switcher, text views, spinners were added. multimedia elements required to make the application more attractive like images,audio and videos were also downloaded and edited as per the requirement thus giving an overall picture of the application.

* Module 2 Core Functionalities:

The transition of the activities in the application was coded and the several layout elements as mentioned above were populated with content. The switching of the activities was achieved using elements like buttons and swipe actions. Different methods were programmed to implement the smooth flow of the application.

* Module 3 UI design:

The app's user interface is everything that the user can see and interact with. Android providing a variety of pre-build UI components such as structured layout objects and UI controls has allowed us to build the graphical user interface for our app, keeping the end users i.e., kids in our mind.

**PROJECT IMPLEMENTATION**

**DAY 6:**

The welcome screen was designed with a attractive background design and appropriate sound effects. The welcome screen was given a swipe action so as to move to the next page giving a better interface for the user.

Code Snippet:

public class MainActivity extends Activity implements SimpleGestureListener {

private SimpleGestureFilter detector;

TextView v1;

MediaPlayer mp;

Button b1;

@Override

public void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

mp = MediaPlayer.create(MainActivity.this, R.raw.baby);

mp.setLooping(false);

mp.start();

detector = new SimpleGestureFilter(this, this);

}

@Override

public boolean dispatchTouchEvent(MotionEvent me) {

// Call onTouchEvent of SimpleGestureFilter class

this.detector.onTouchEvent(me);

return super.dispatchTouchEvent(me);

}

@Override

public void onSwipe(int direction) {

Intent intent;

switch (direction) {

case SimpleGestureFilter.SWIPE\_RIGHT:

// Toast.makeText(this, "right swipe", Toast.LENGTH\_SHORT).show();

intent = new Intent(this, MainActivity.class);

startActivity(intent);

break;

case SimpleGestureFilter.SWIPE\_LEFT:

// Toast.makeText(this, "left swipe", Toast.LENGTH\_SHORT).show();

intent = new Intent(this, MainActivity1.class);

startActivity(intent);

break;

}

}

public void onDestroy() {

mp.stop();

mp.release();

}

@Override

public void onDoubleTap() {

}

}

Difficulties were faced in implementing the SimpleGestureFilter actions, but then they were easy enough to overcome.

Many video tutorials helped our way.

Day 7 :

SlideShow was tried and implemented successfully using ViewFlipper.Several android tutorial sites were at help to us.

Code Snippet:

ViewFlipper flipper = (ViewFlipper) findViewById(R.id.*flipper1*);

**if**(mFlipping==0){

/\*\* Start Flipping \*/

flipper.startFlipping();

mFlipping=1;

mButton.setText("start slideshow");

}

**else**{

/\*\* Stop Flipping \*/

flipper.stopFlipping();

mFlipping=0;

mButton.setText("stop slideshow");

}

Day 8:

Various videos needed to make the app more attractive were downloaded,trimmed and format converted and inserted in the application.

A gallery was made to project the image switches in few activities. The predefined class ImageSwitcher was used and several methods were coded to make it functional.

Code Snippet:

**public** View getView(**int** arg0, View arg1, ViewGroup arg2) {

ImageView iView = **new** ImageView(ctx);

iView.setImageResource(imgs[arg0]);

iView.setScaleType(ImageView.ScaleType.*FIT\_XY*);

iView.setLayoutParams(**new** ~~Gallery~~.LayoutParams(200, 150));

**return** iView;

}

}

**public** View makeView() {

ImageView iView = **new** ImageView(**this**);

iView.setScaleType(ImageView.ScaleType.*FIT\_CENTER*);

iView.setLayoutParams(**new** ImageSwitcher.LayoutParams

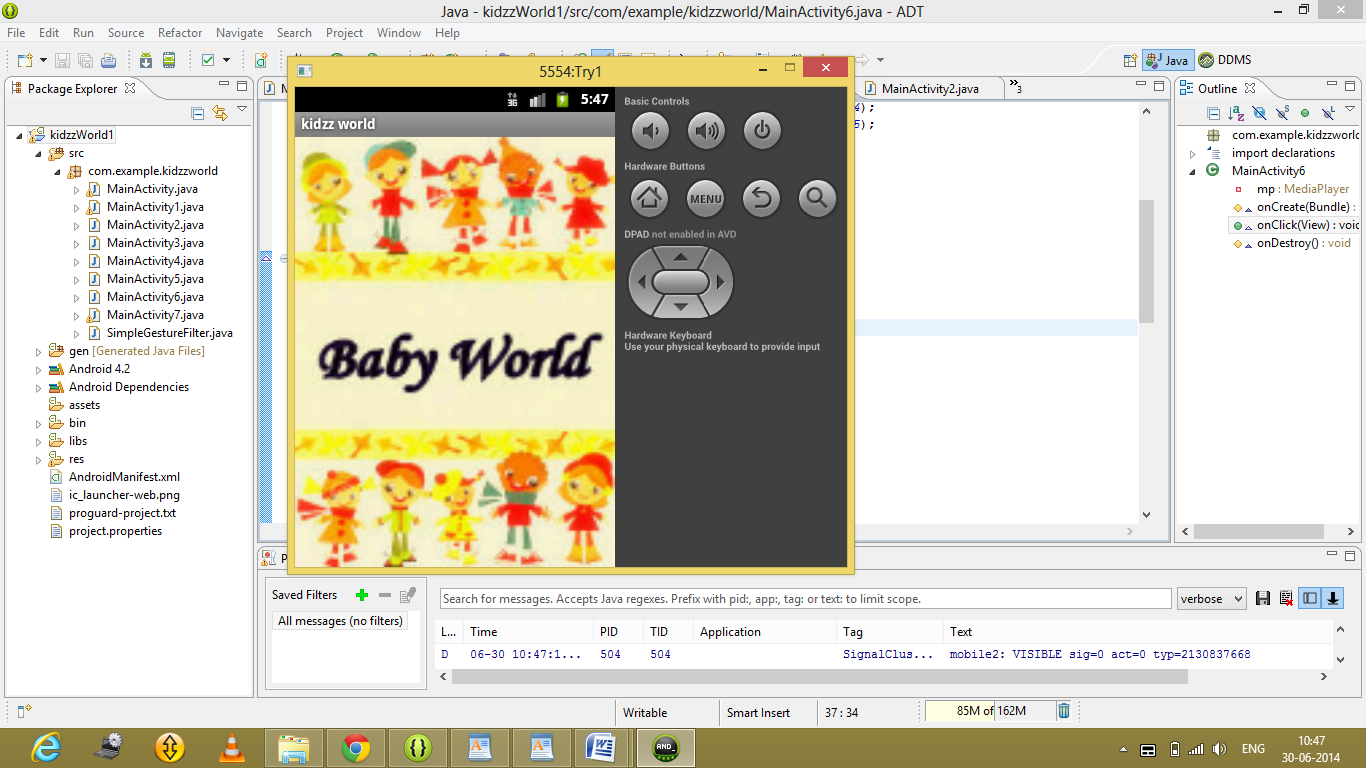
(LayoutParams.*MATCH\_PARENT*,LayoutParams.*MATCH\_PARENT*));

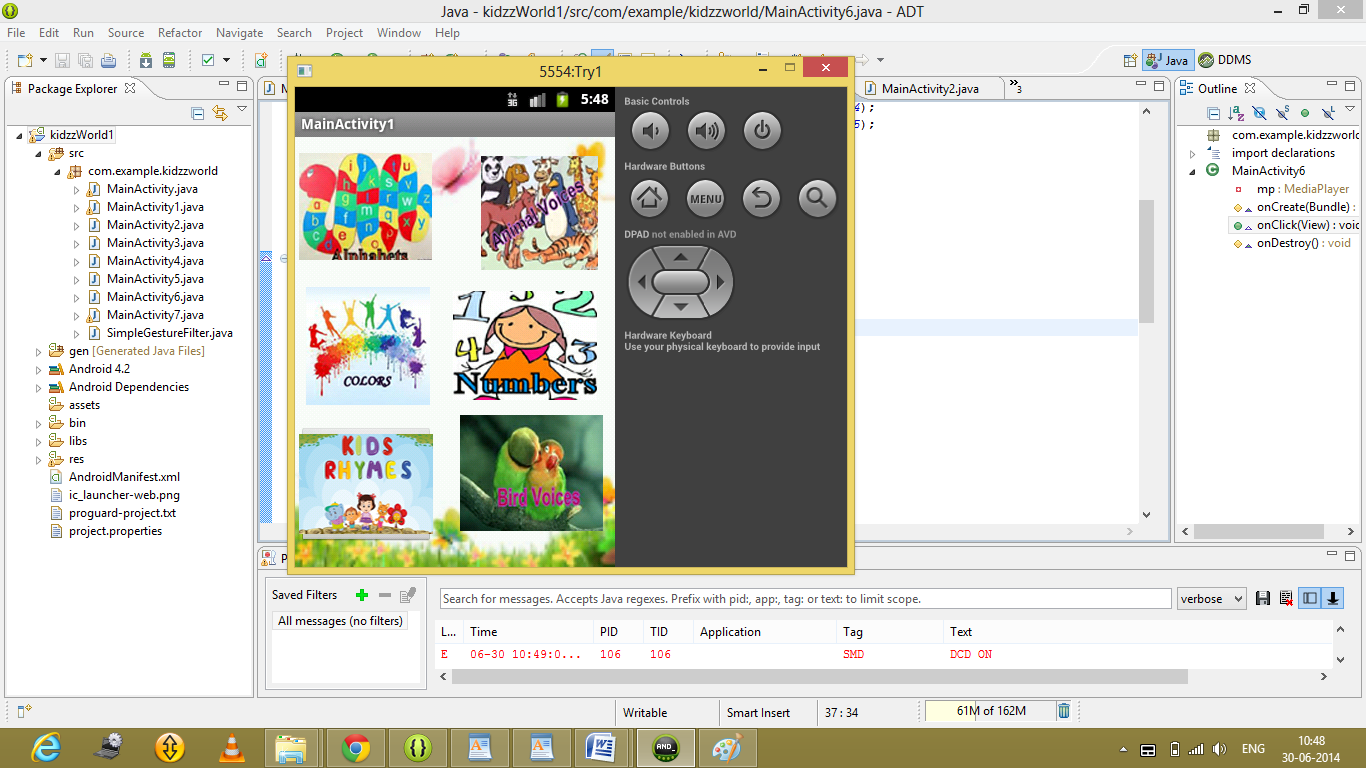
iView.setBackgroundColor(0xFF000000);

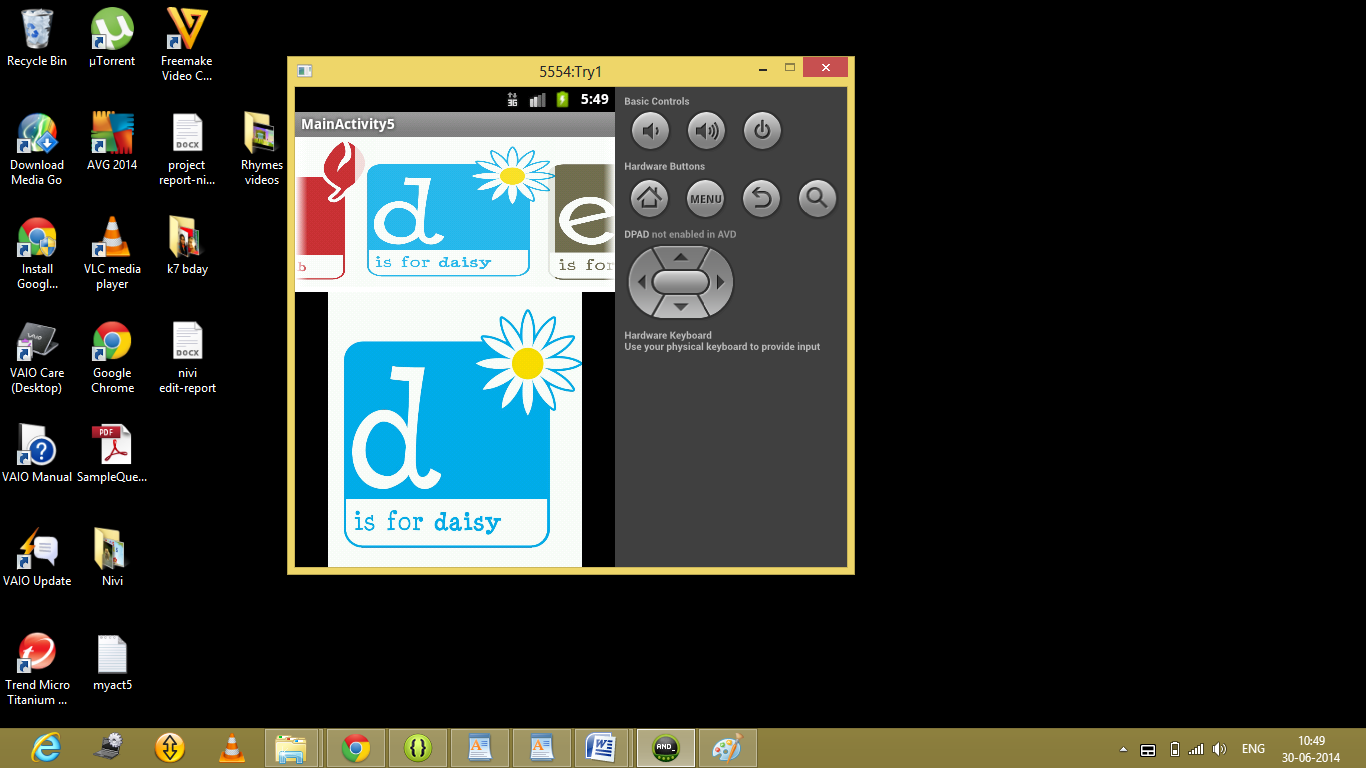
**return** iView;

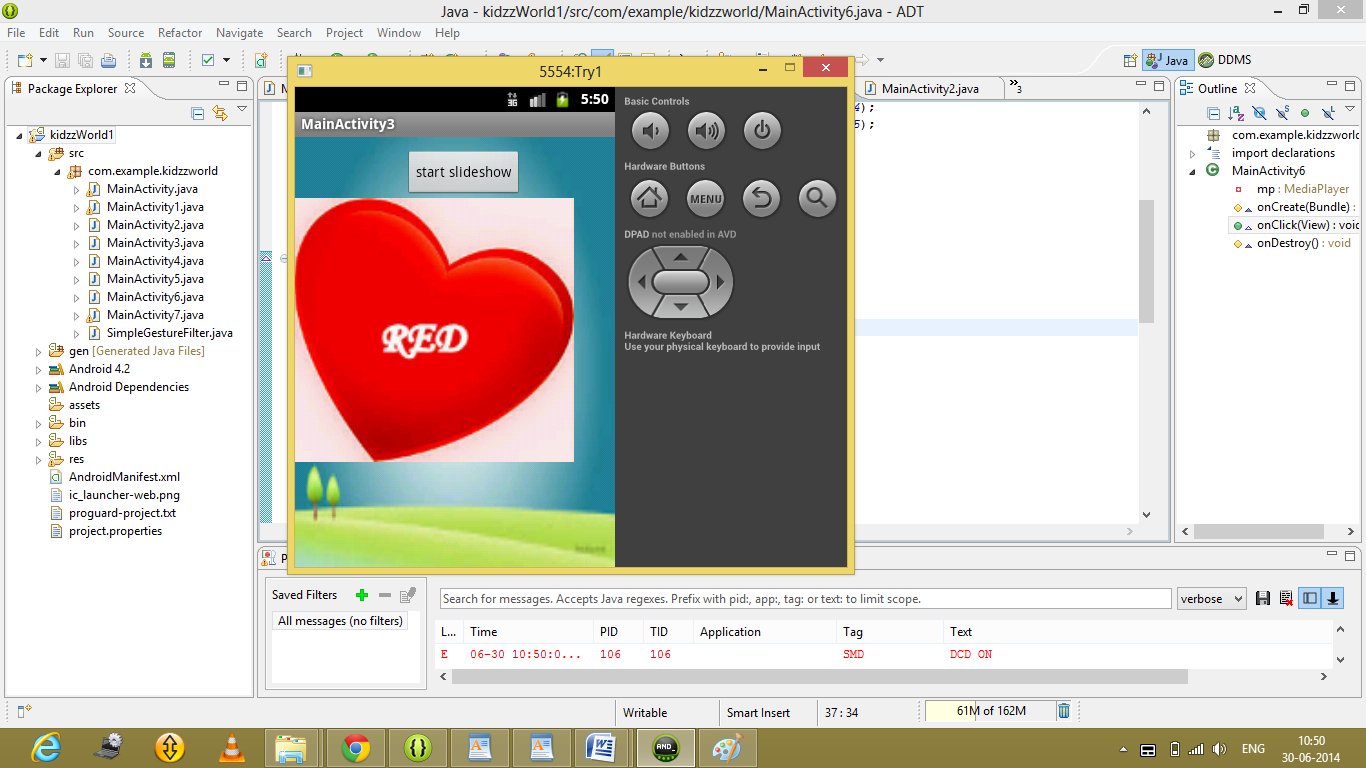
}

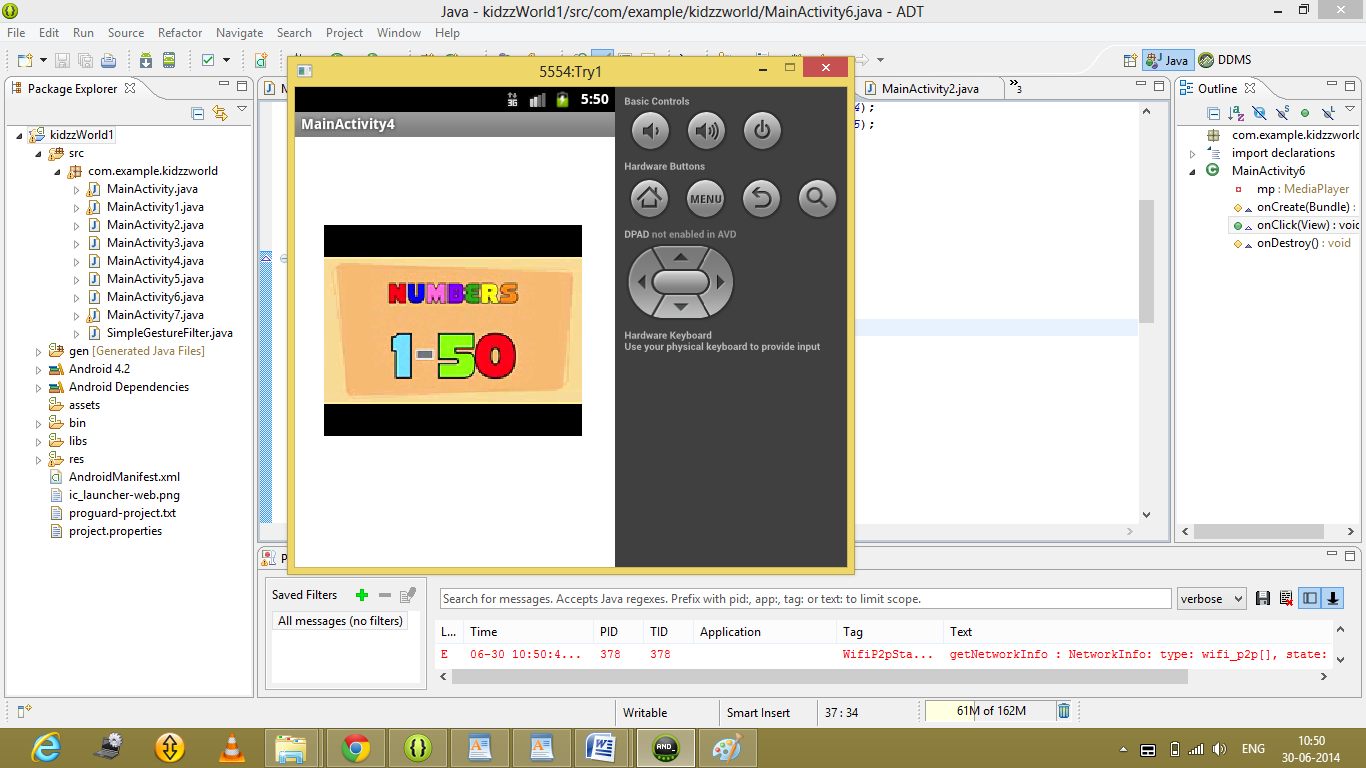
Day 9:



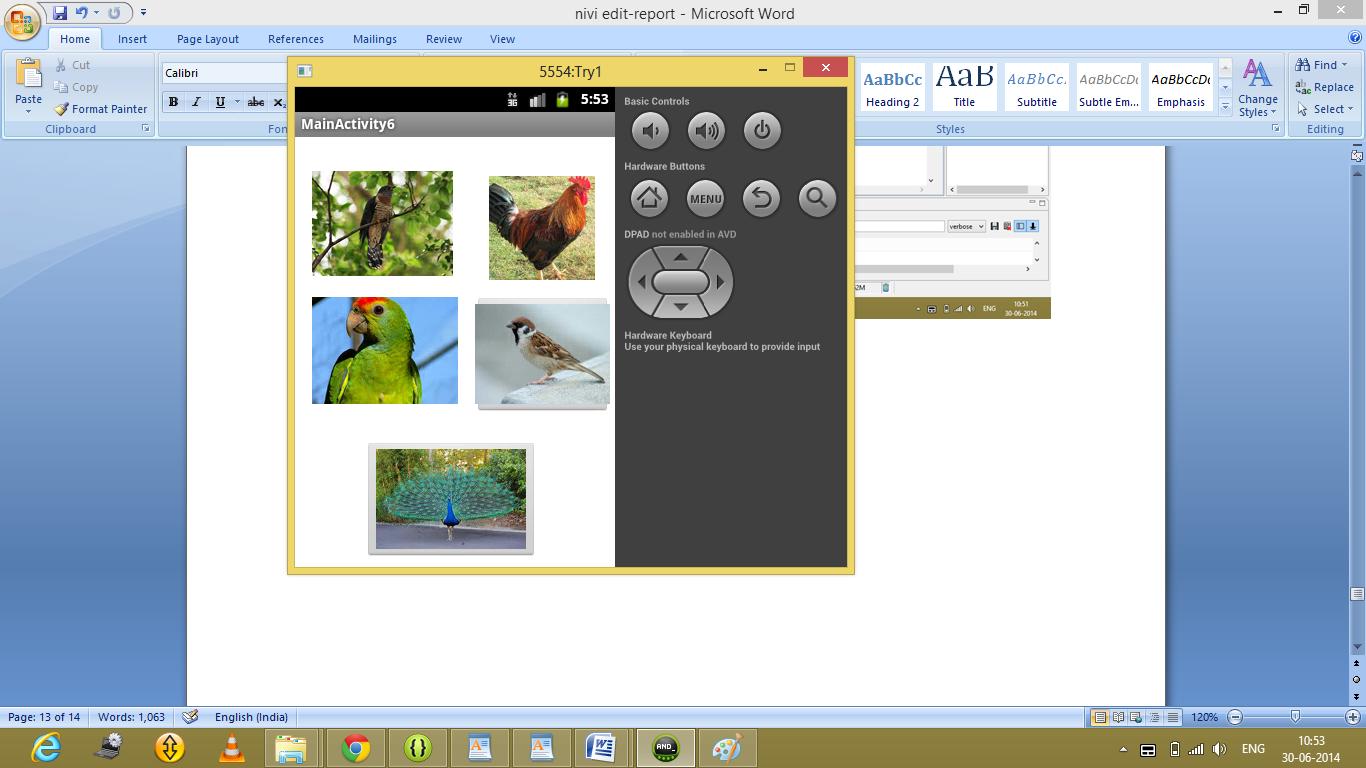


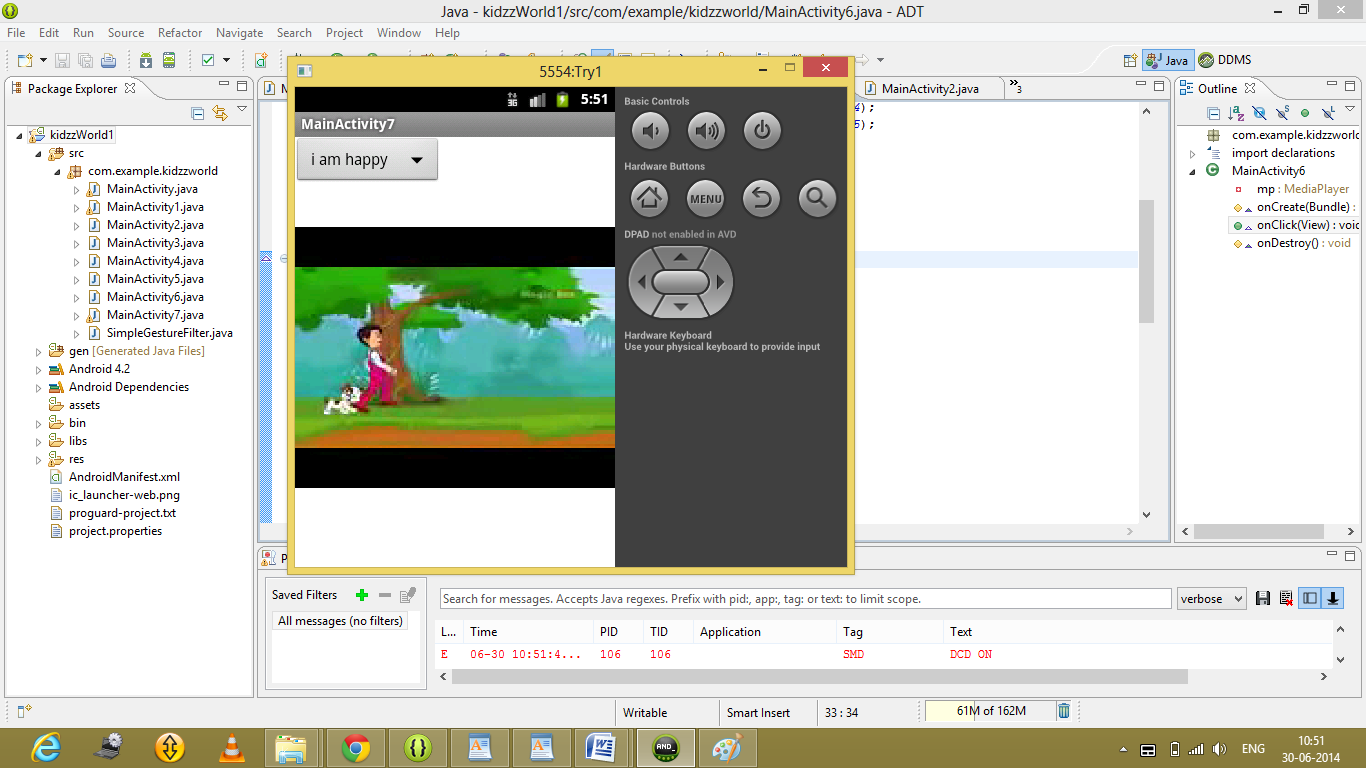






Day 10 :





**Conclusion:**

The application has come out very well. The devepoment process was really fun filled and informative. It instilled many valuable knowledge in our minds. We would like to pursue android programming further.

**Verified.**

**Project Coordinator**