|  |
| --- |
| Comp 3900 industry project |
| Chat Program |
| An Android Chat Application |
|  |
| **Robin Hsieh, German Villarreal & Mateusz Siwoski** |
|  |

|  |
| --- |
|  |

# Abstract

The chat program is a chat program for the Android OS, meant to be used with a wireless device. The program will allow users to send text messages to each other and will display by a custom username who sent the message.

This document describes the functions and features of the program and general operation of the application. The application is meant to be a general chat program and is designed to be capable of supporting additional features such as picture messaging.

# Table of Contents

[Abstract 1](#_Toc373843499)

[Table of Contents 2](#_Toc373843500)

[Introduction 3](#_Toc373843501)

[Features 4](#_Toc373843502)

[Devices & Operating System 4](#_Toc373843503)

[Design: 5](#_Toc373843504)

[Gantt Chart: 6](#_Toc373843505)

[Conclusion 7](#_Toc373843506)

[Instructions for Installations: 8](#_Toc373843507)

[How to install through APK: 8](#_Toc373843508)

[Test Cases 9](#_Toc373843509)

[Figures 10](#_Toc373843510)

[Test 1: Initial Start Up (PASS) 10](#_Toc373843511)

[Test 2: Typing Text (PASS) 10](#_Toc373843512)

[Test 3: Typing multiple text (FAIL) 11](#_Toc373843513)

[Test 4: Custom username (Pass) 11](#_Toc373843514)

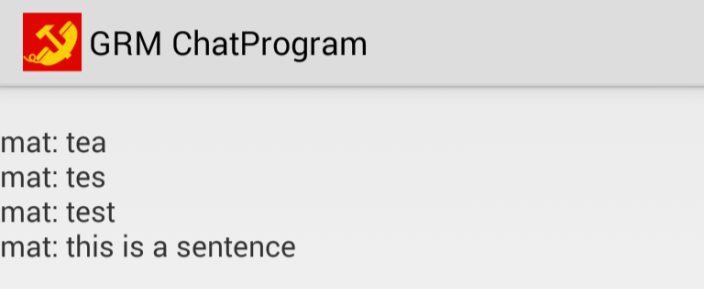
[Test 5: Display typed text correctly (PASS) 12](#_Toc373843515)

# Introduction

For our industry project, we have been tasked to create a chat application for the Android OS. The application uses peer-to-peer to connect with other users and allows for communication between at least two devices. The device is meant for the android OS, allowing for touch screen typing.

# Features

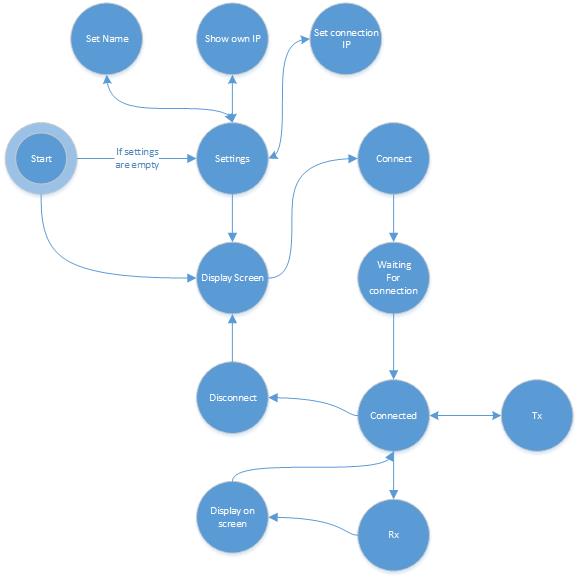
The following features have been implemented:

* Custom Username: Allows, upon start-up of the program, a user to enter a custom username to display in the general chat screen what they have said.   
    
  
* Auto Scrolling of text: As messages have reached the end of the defined window size, the text will be appended to the bottom and the screen will descend as more messages are sent.

# Devices & Operating System

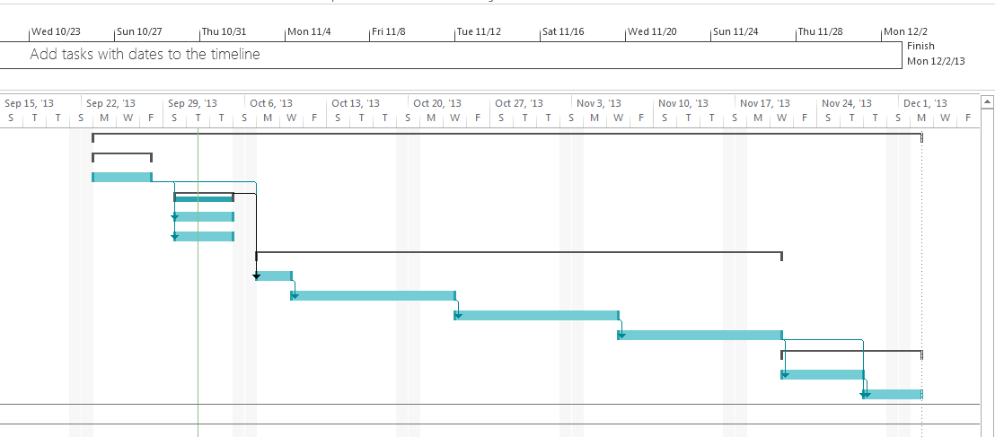
The chat program is dependent on any device that is running the Android OS 2.4 and higher. The devices that were used in testing were the Nexus 4, Nexus 7 and Samsung Galaxy S3.

# Design:



# Gantt Chart:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Task Name | Duration | Start | Finish | Predecessors |
| **Android Chat Program** | **51 days** | **Mon 9/23/13** | **Mon 12/2/13** |  |
| **Implementation** | **5 days** | **Mon 9/23/13** | **Fri 9/27/13** |  |
| Report | 5 days | Mon 9/23/13 | Fri 9/27/13 |  |
| **Design stages** | **5 days** | **Mon 9/30/13** | **Fri 10/4/13** |  |
| Design | 5 days | Mon 9/30/13 | Fri 10/4/13 | 3 |
| Setup | 5 days | Mon 9/30/13 | Fri 10/4/13 | 3 |
| **Application Development** | **33 days** | **Mon 10/7/13** | **Wed 11/20/13** |  |
| Android display screen | 3 days | Mon 10/7/13 | Wed 10/9/13 | 4,3 |
| Establish connection with devices | 10 days | Thu 10/10/13 | Wed 10/23/13 | 8 |
| Read information | 10 days | Thu 10/24/13 | Wed 11/6/13 | 9 |
| Write information | 10 days | Thu 11/7/13 | Wed 11/20/13 | 10 |
| **Testing** | **8 days** | **Thu 11/21/13** | **Mon 12/2/13** |  |
| Testing | 5 days | Thu 11/21/13 | Wed 11/27/13 | 11 |
| Additional Features | 3 days | Thu 11/28/13 | Mon 12/2/13 | 11,13 |



# Conclusion

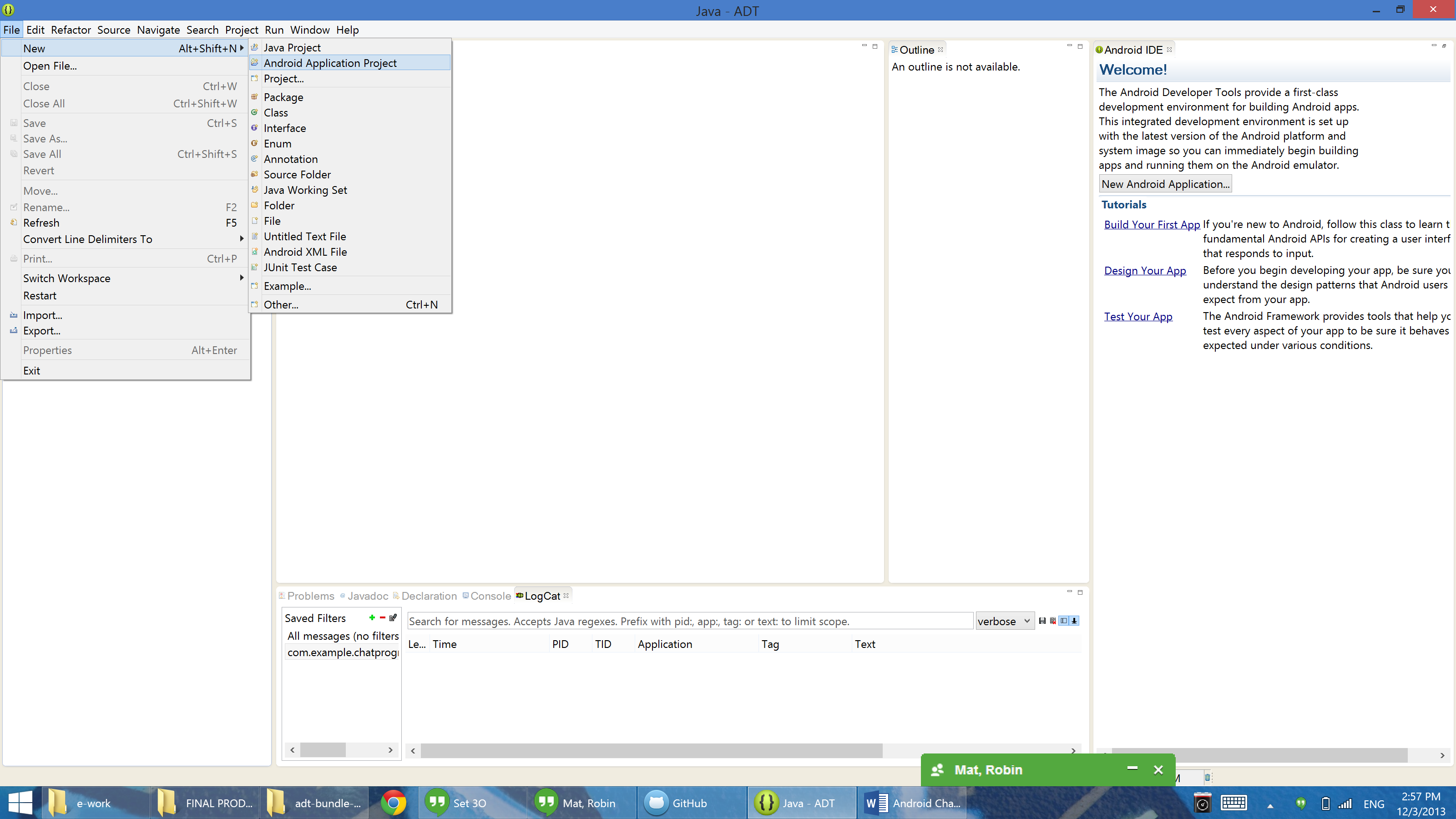
Asdfa

# Instructions for Installations:

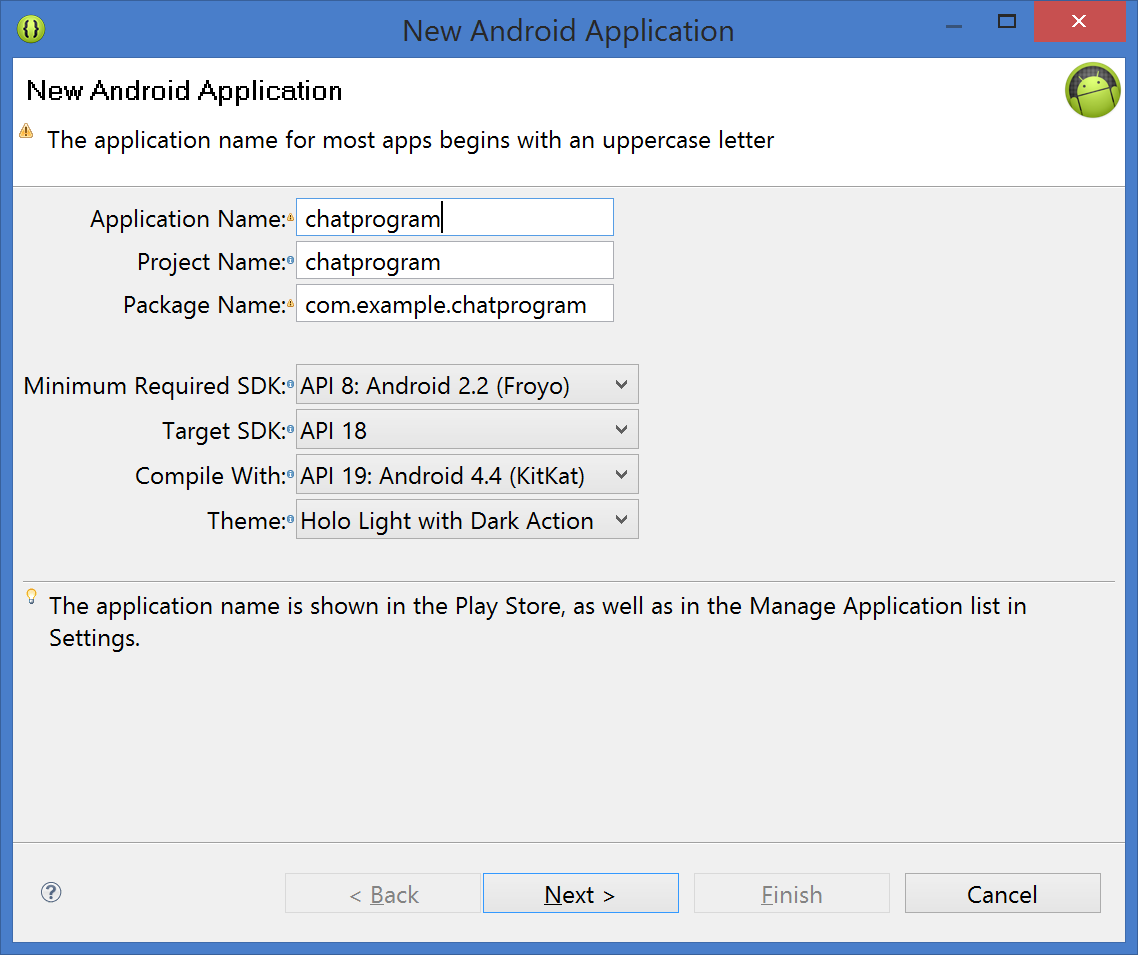
## How to install through Eclipse:

We assume you have the newest version of Eclipse and ADT plugin before starting with these instructions. If you do not have it installed you can find the bundle here: <http://developer.android.com/sdk/index.html> with tutorials on setting up your environment.

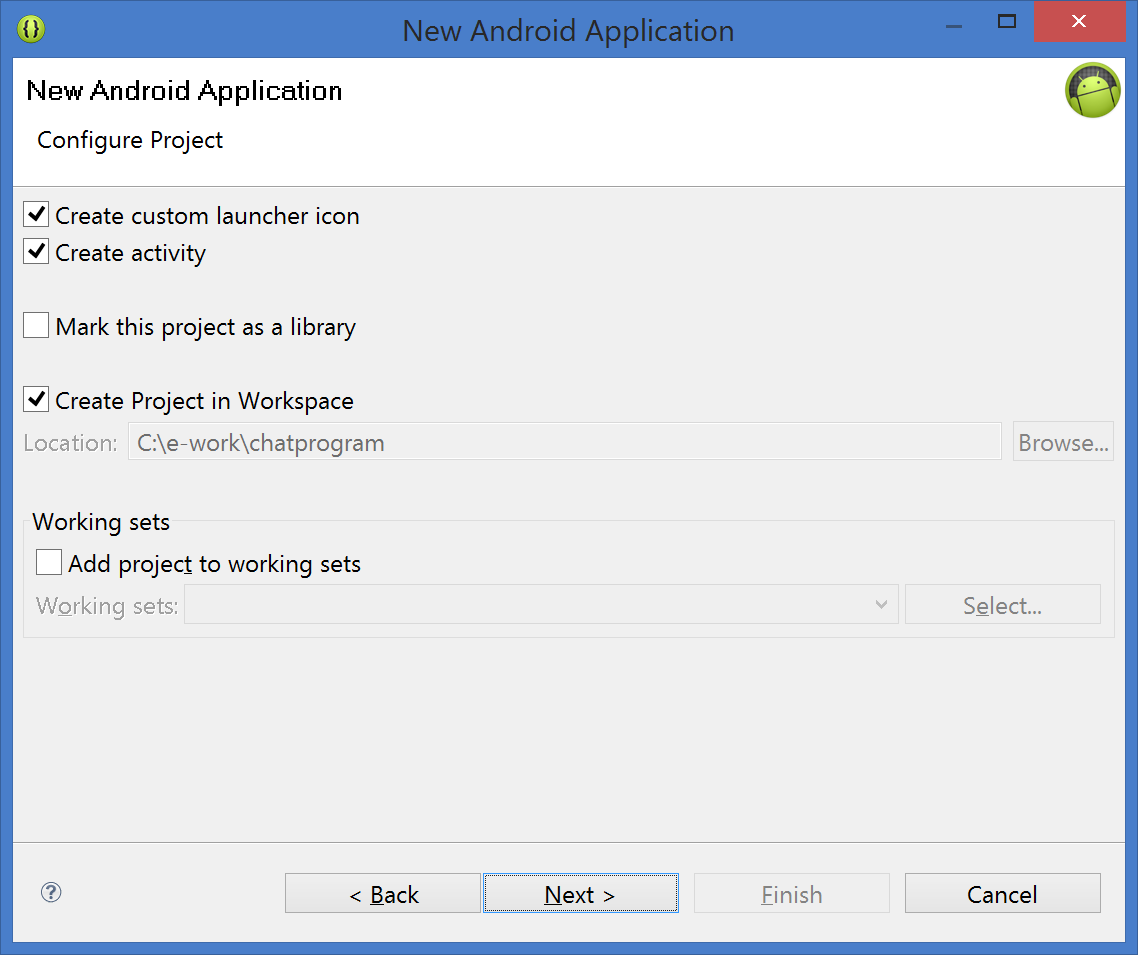
1. Open Eclipse and create a new Java Android

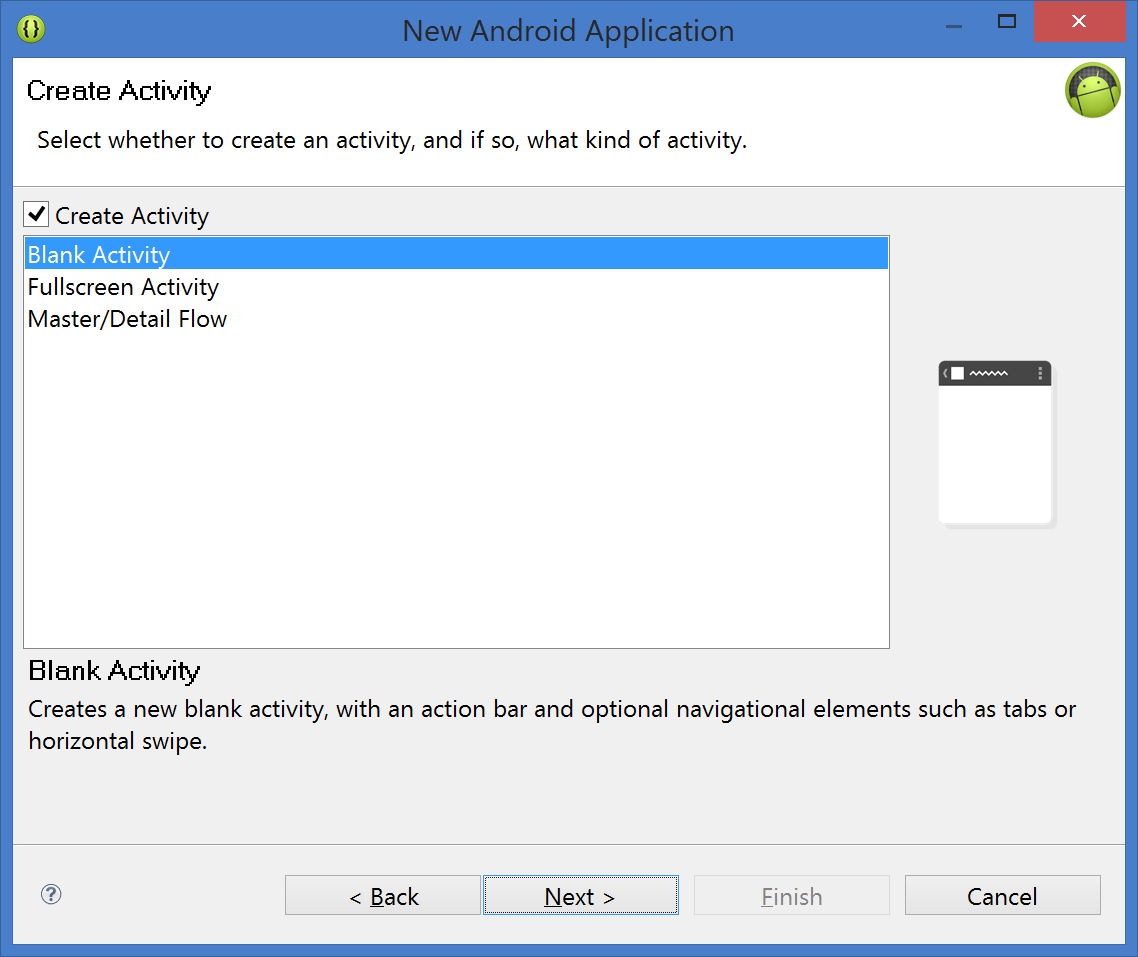
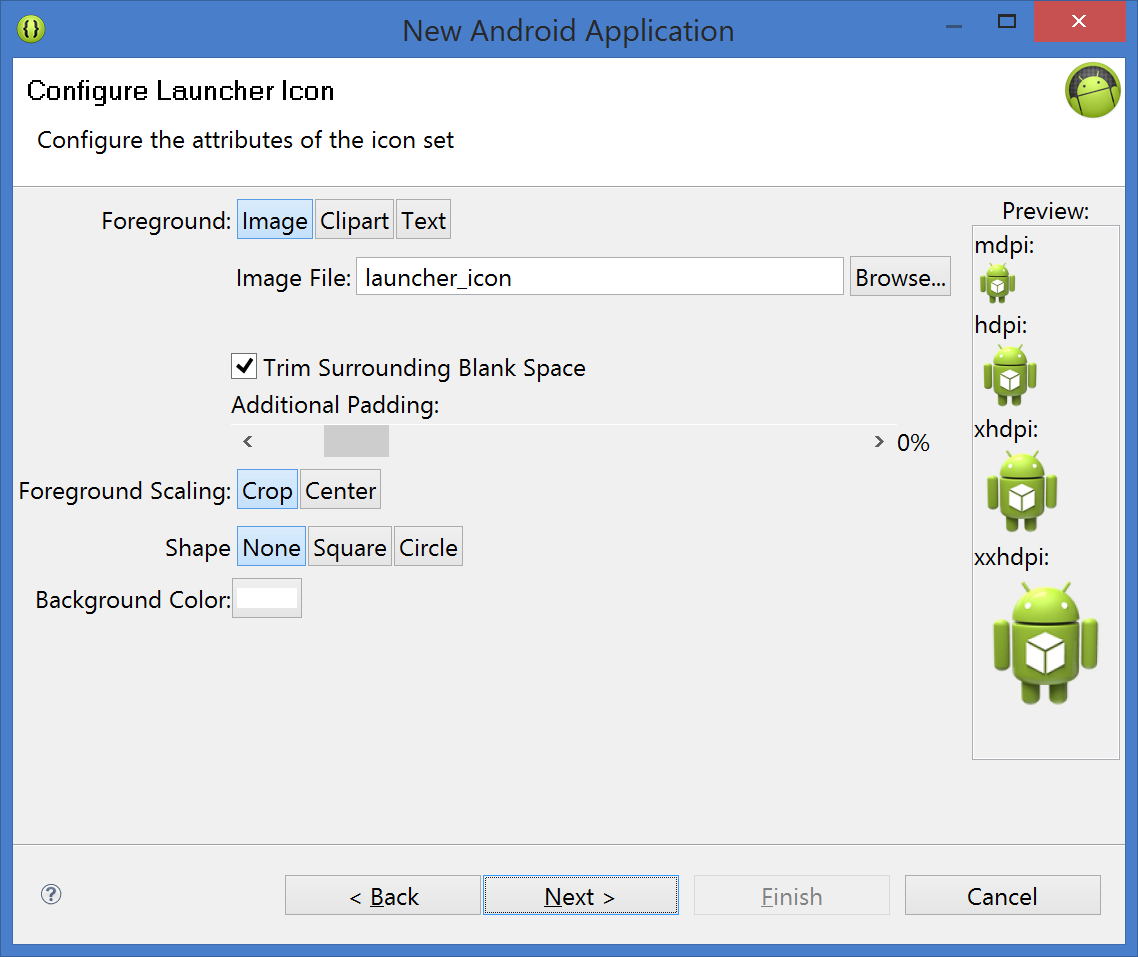


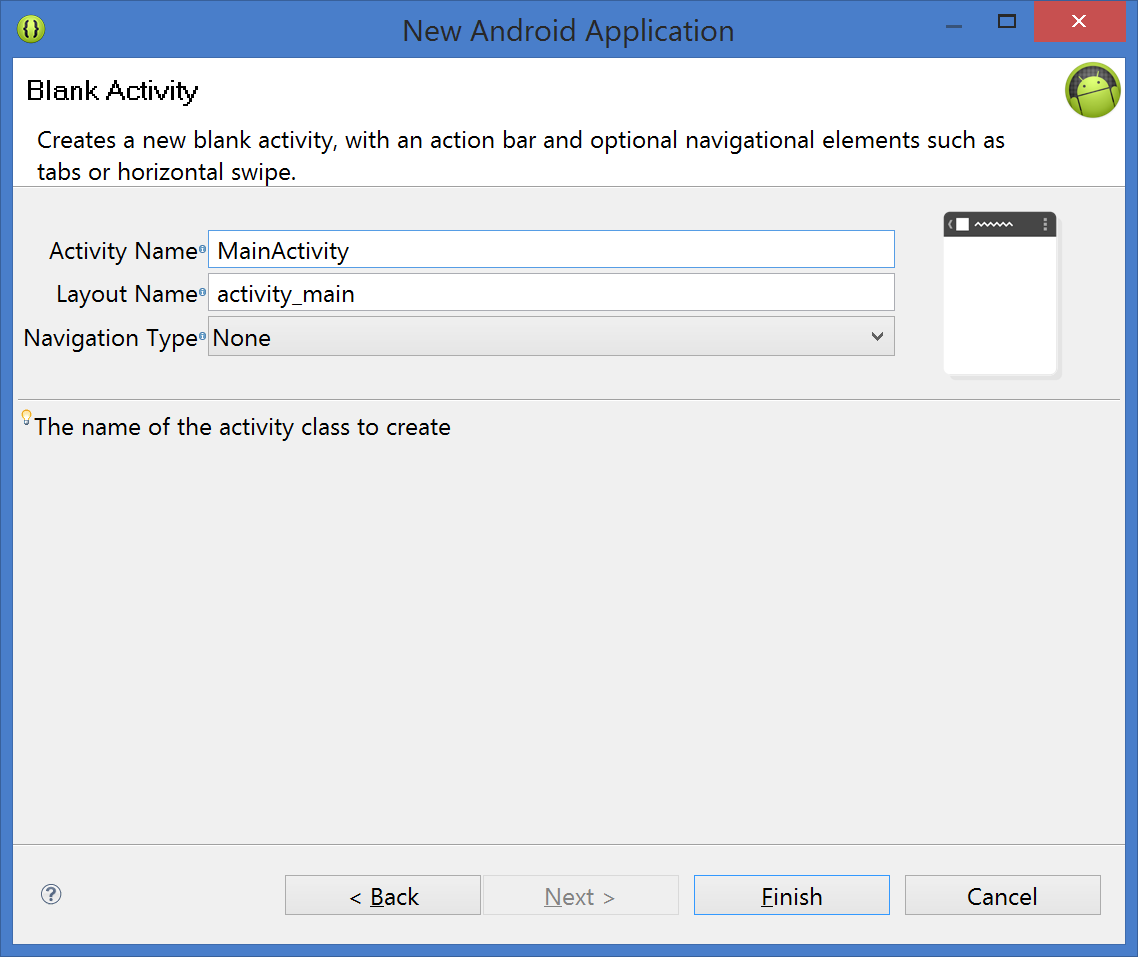
1. Name it “chatprogram” and the rest should automatically be filled in; click **Next**.



1. Continue through the setup wizard with the default settings

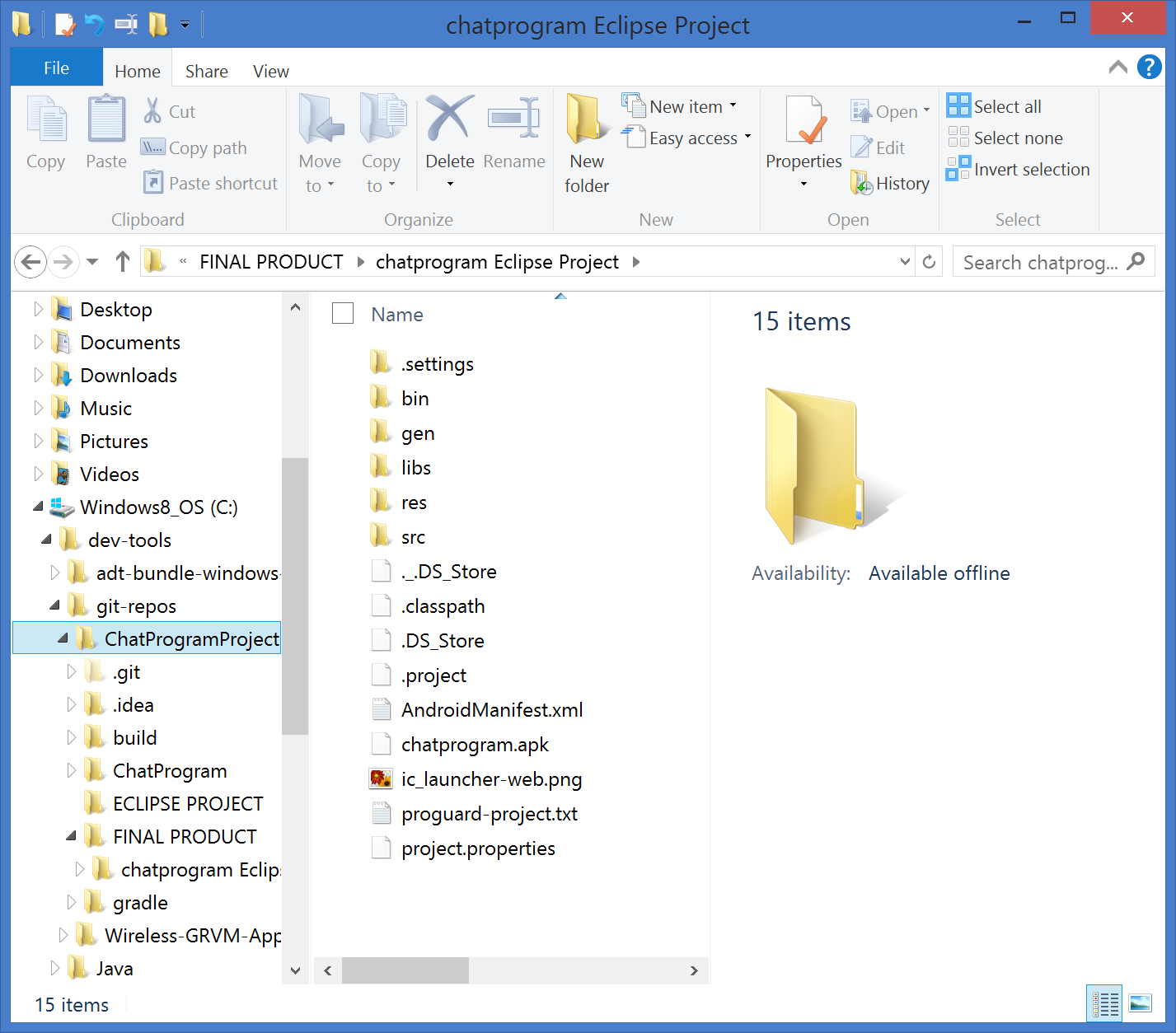




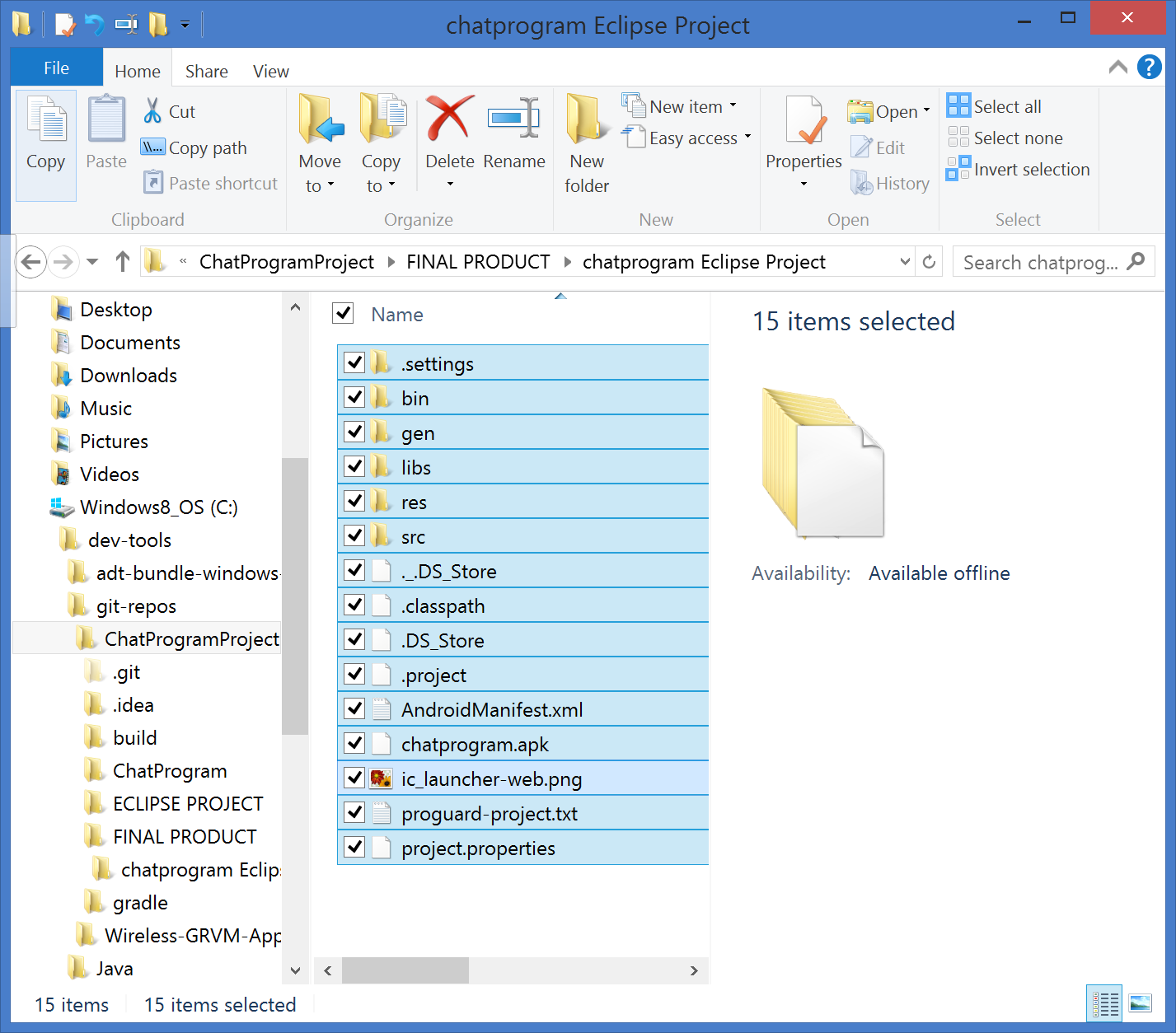


Your project should appear in the “Package Explorer” view and the default MainActiviy.java may open.

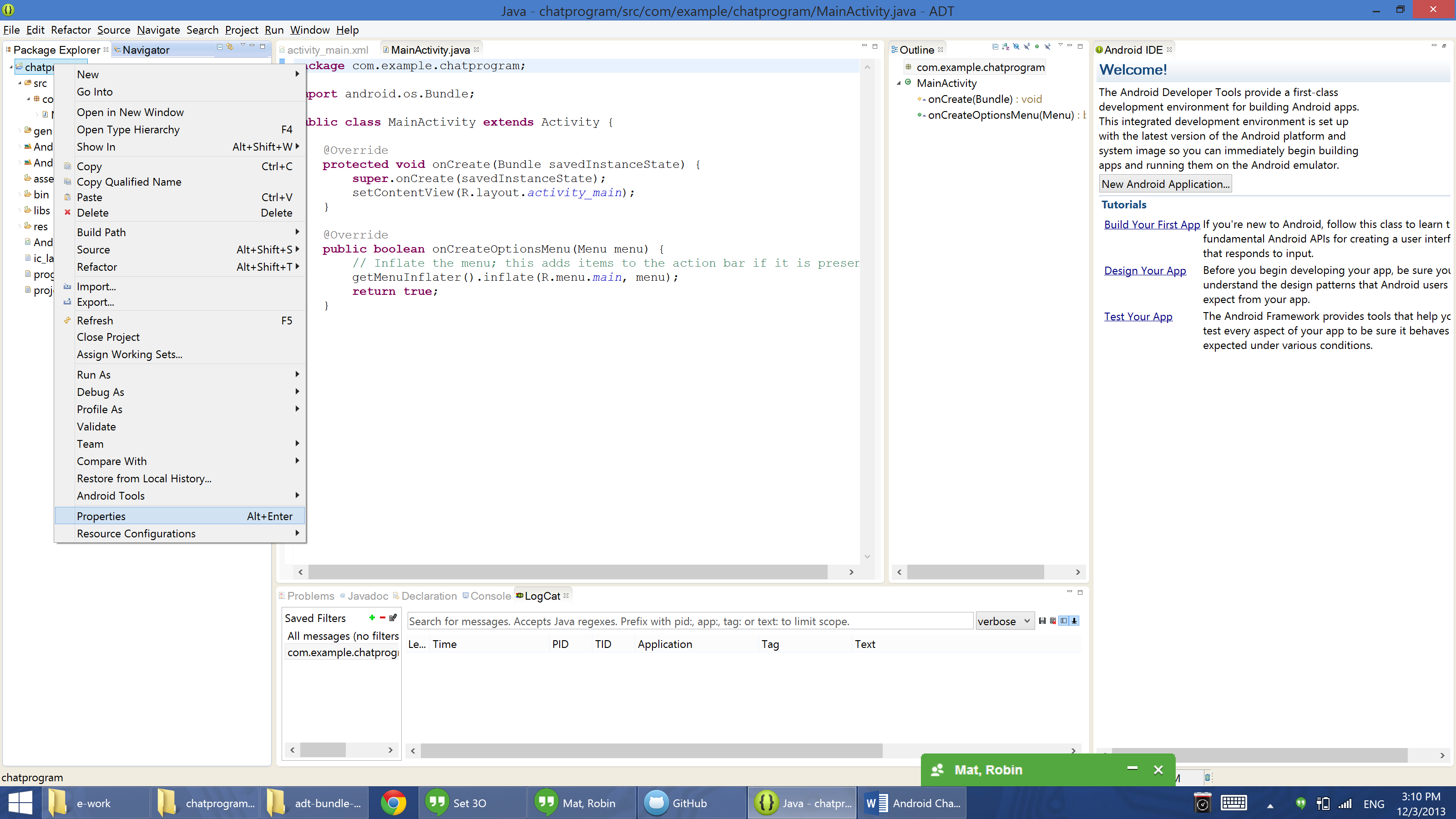
1. In windows explorer locate the final version of the chatprogram application

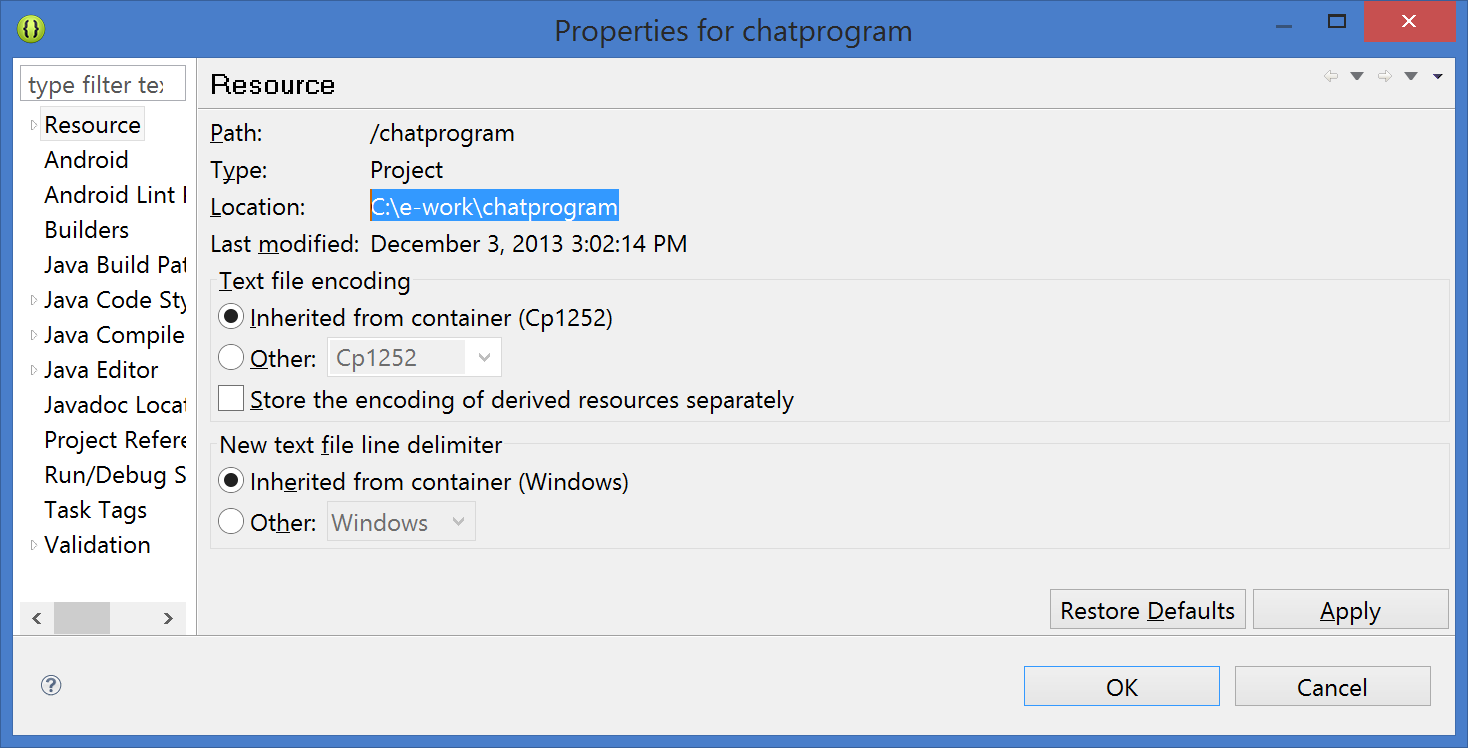


1. Select all the files, right click and click **Copy**

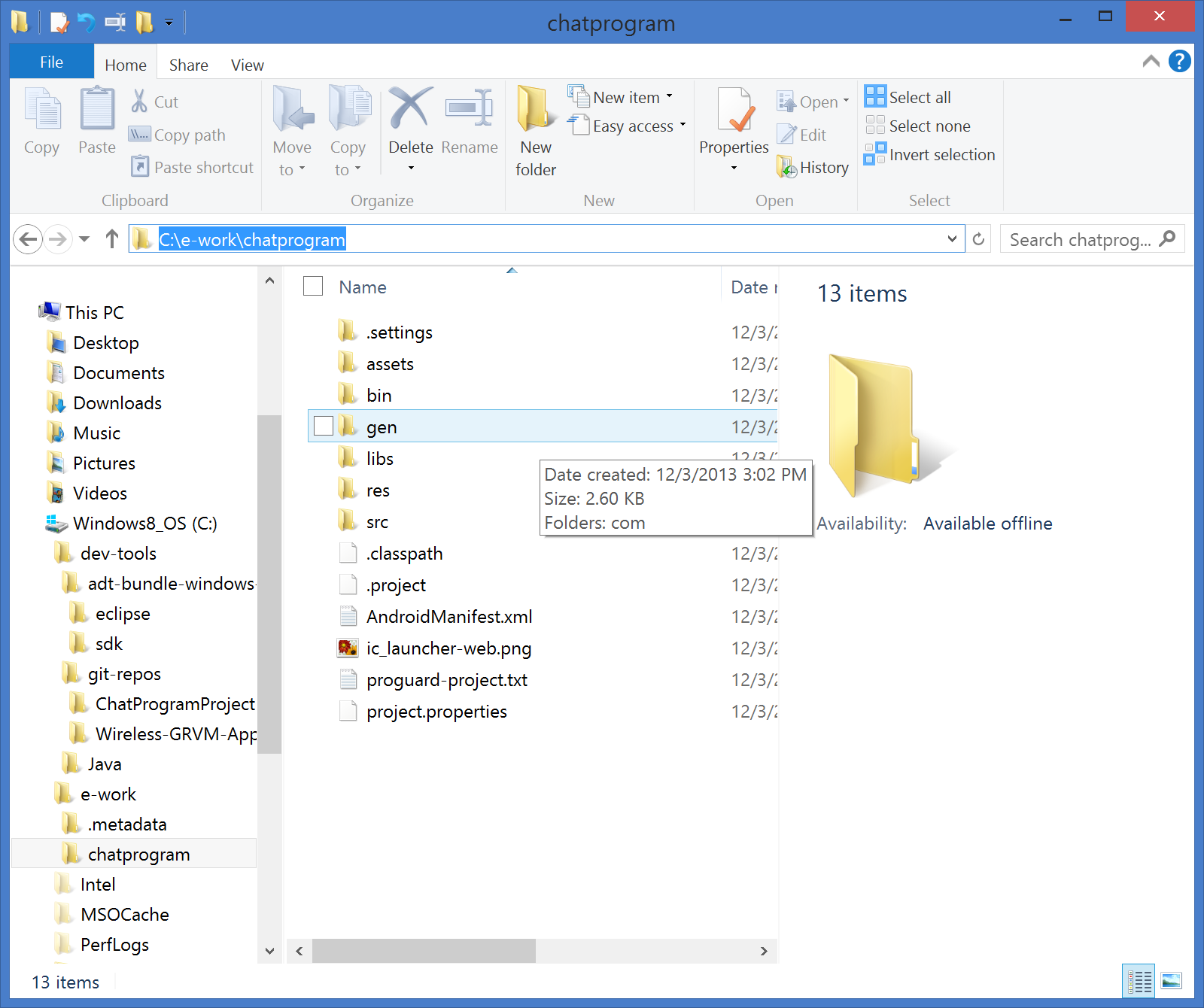


1. Locate your chatprogram project created in Step 1 by right clicking the project in Eclipse and click **Properties**.

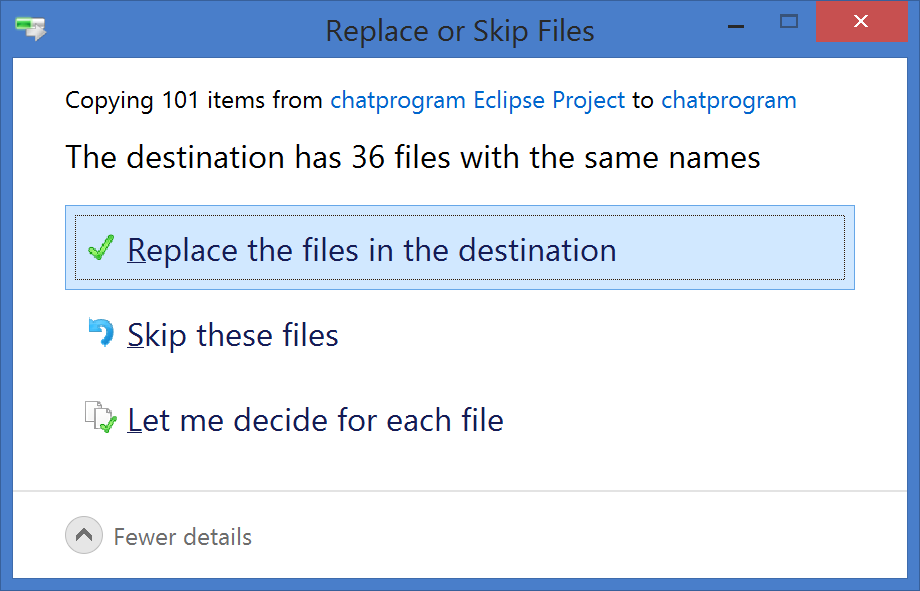
The following window will open stating the location of your project.

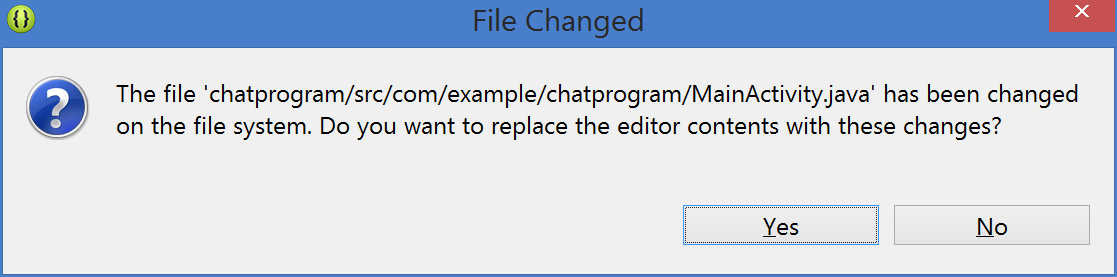


1. Navigate to the location of this project in Windows Explorer.



1. **Paste** all of the files copied in Step 5.
2. Click **Replace the files in the destination**.



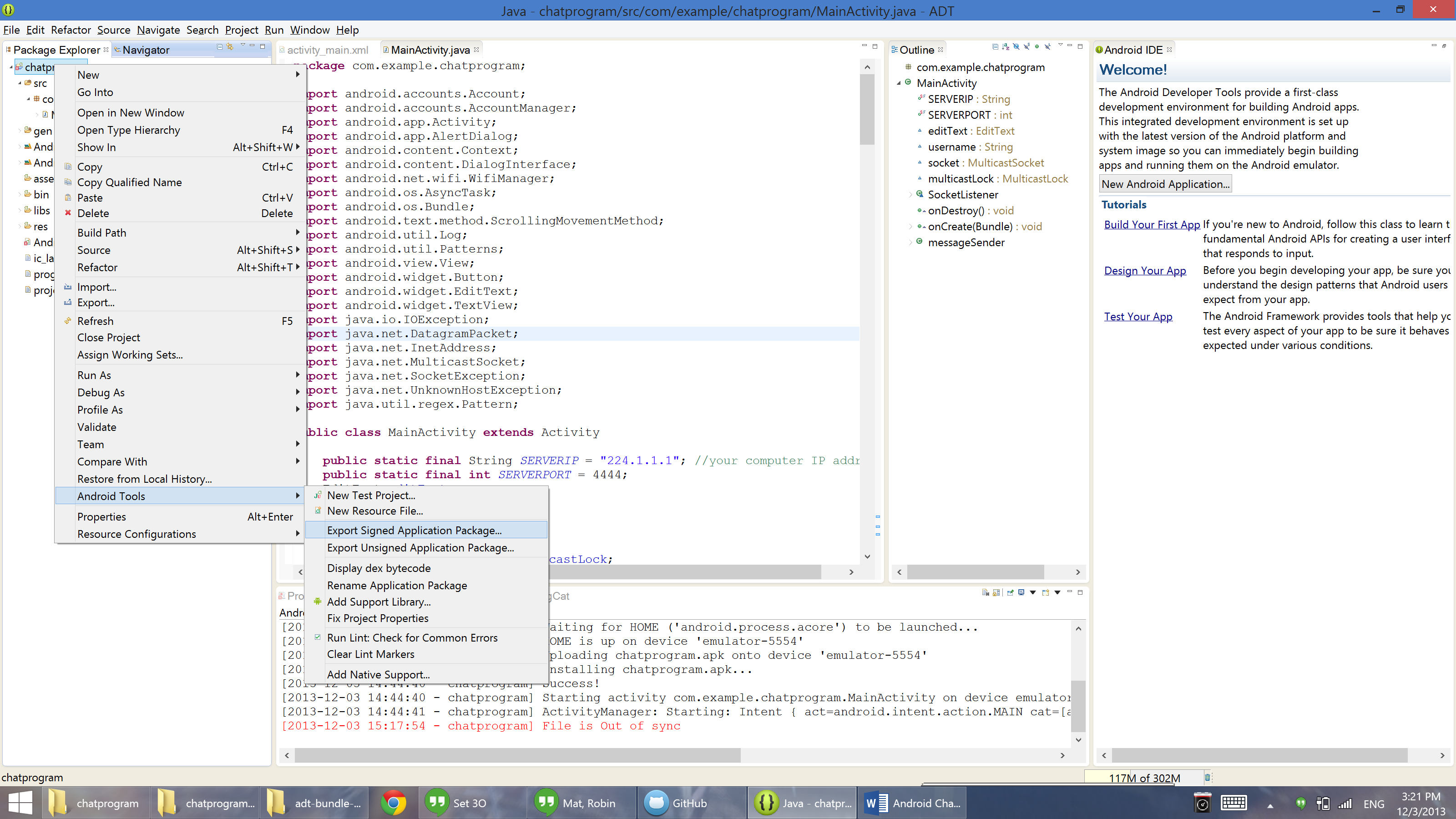
1. Eclipse will notify you there were changes to your project. 

Click **Yes**.Your MainActivity.java that was open will reload displaying our finished file.

## How to create your APK using Eclipse:

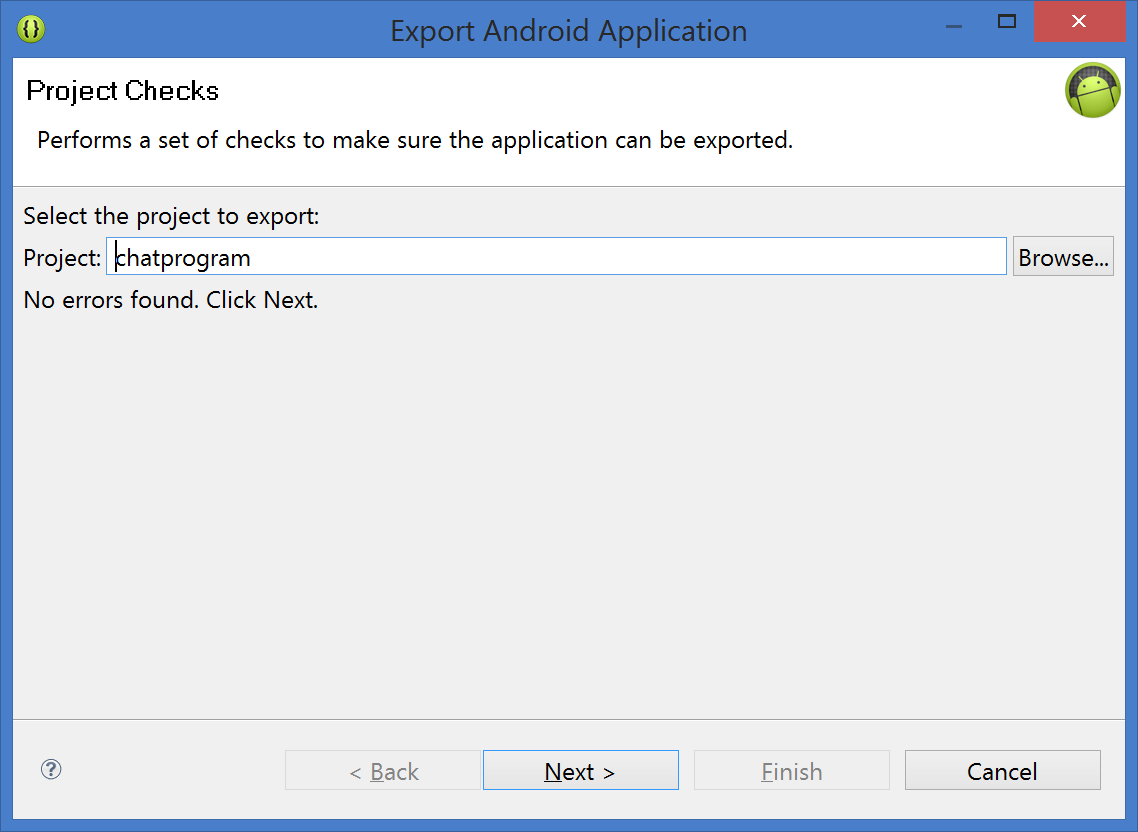
The APK is the file used to install the program into all Android devices. We must create this file from the source code derived in the previous steps of this tutorial. This tutorial assumes the previous instructions were successfully completed.

1. Right click your project, navigate down to **Android Tools** and click on **Export Signed Application Package...**

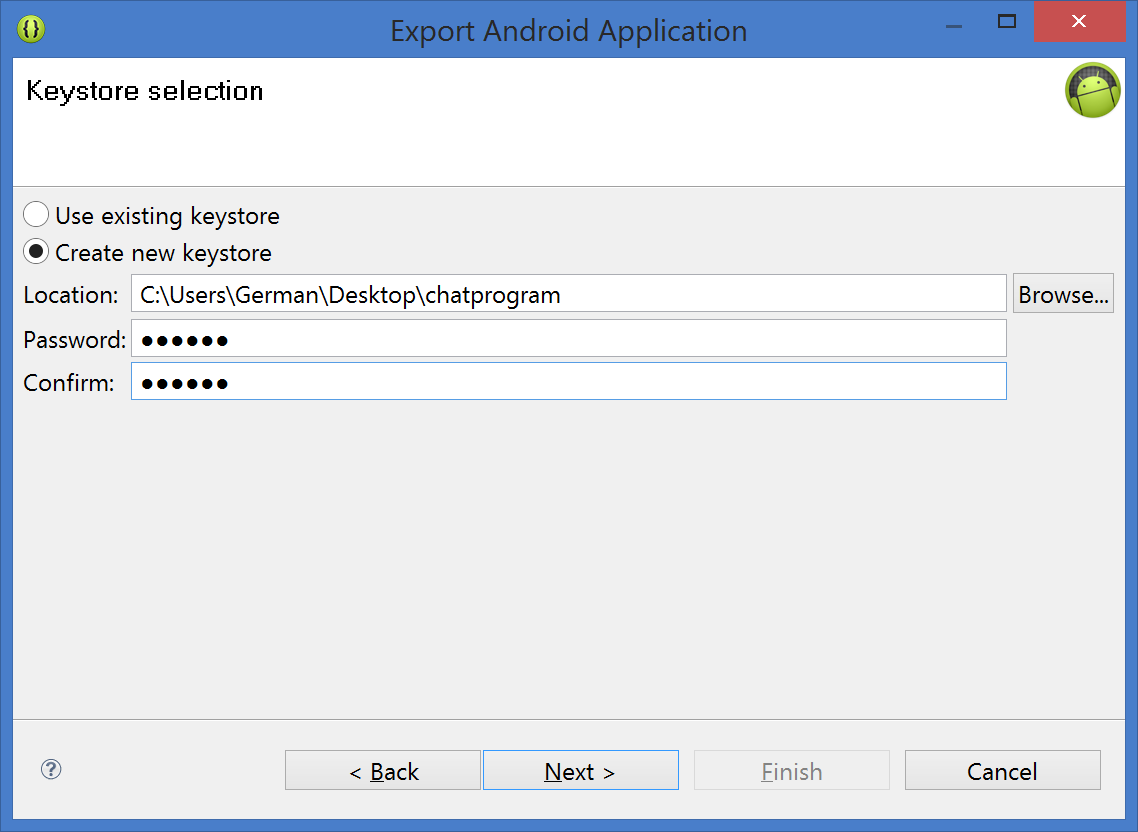


We use the *Signed* as opposed to the *Unsigned* version in order to be able to install the application on other devices. If you export an *unsigned application package* your application may not install on certain devices.

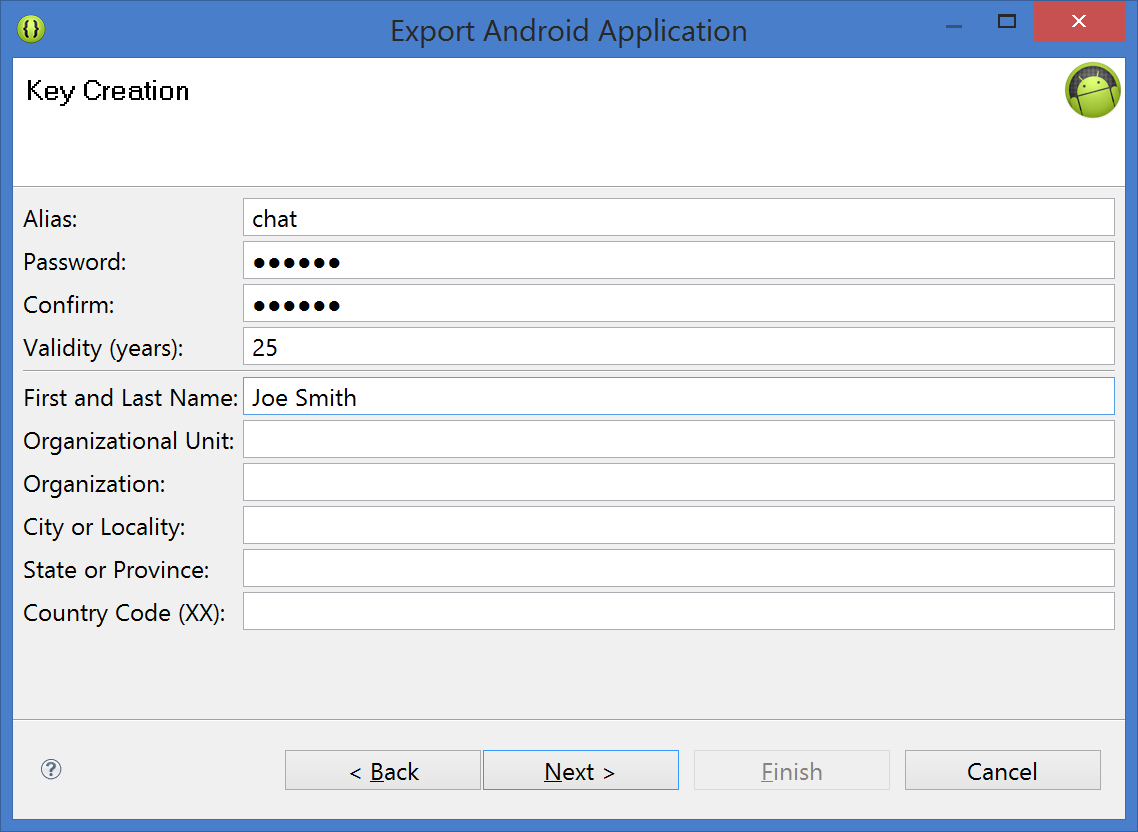
1. The following message box should open, click **Next**.



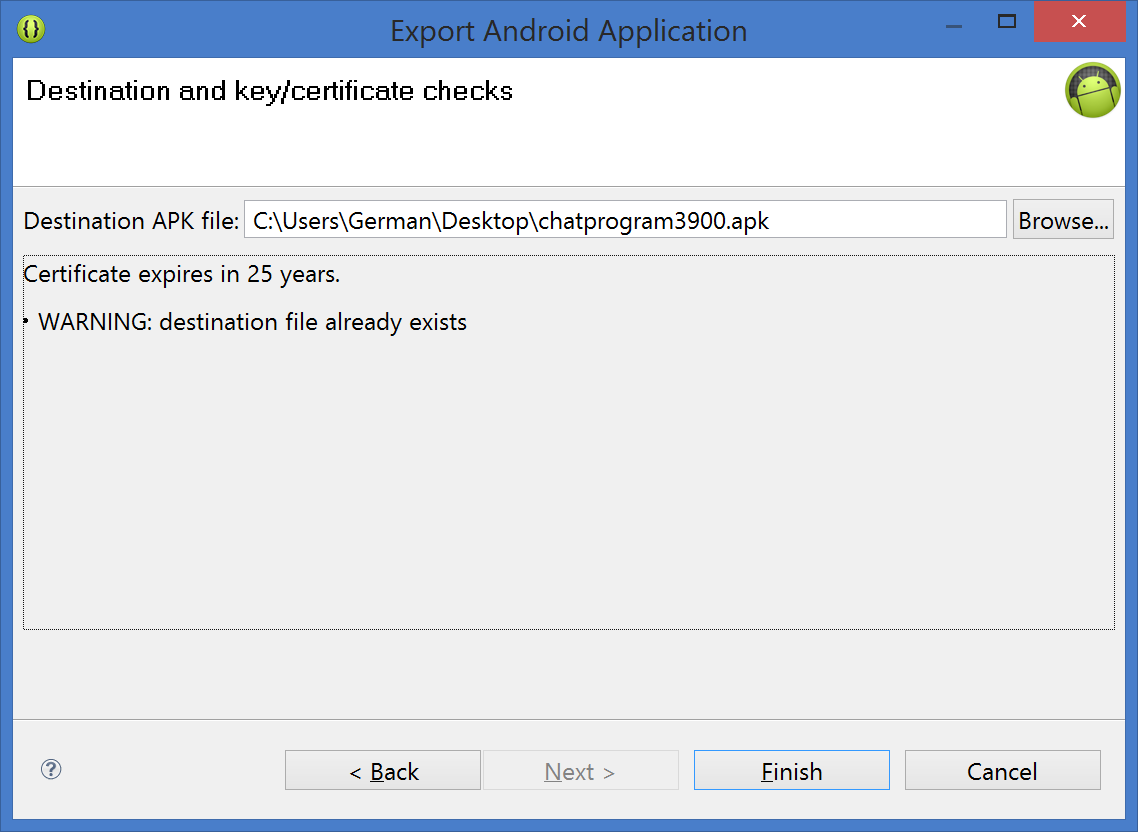
1. Create a new keystore and enter the information required. The location will be where the keystore will be saved, make this an easy location to access such as the Desktop. Also use a password that will be easy to remember. Click **Next**.



1. Insert the required fields to Sign the application with. Click **Next**.



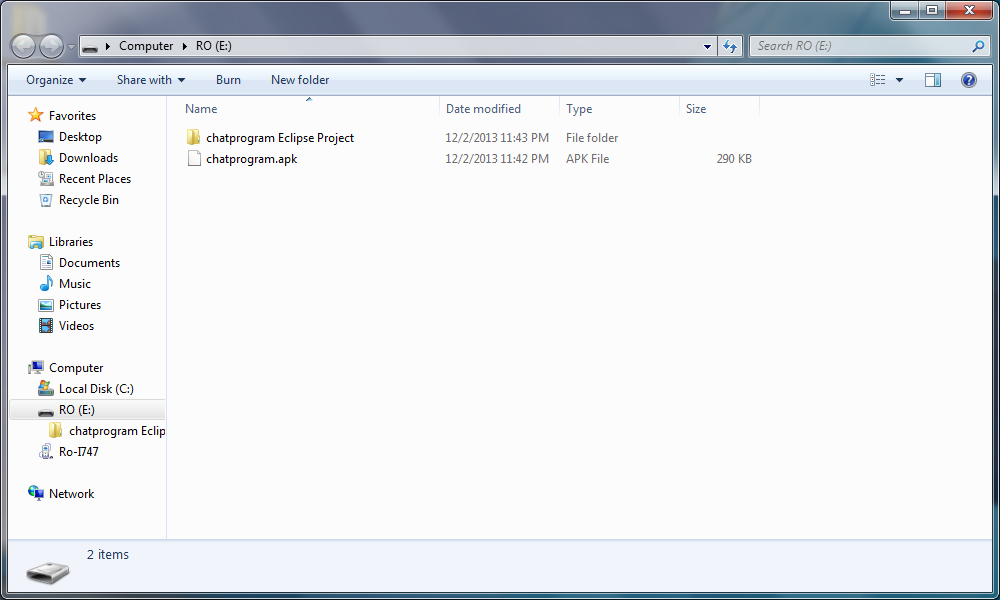
1. Select where to export the actual APK file that will be used to install the program on Android devices. It is recommended to make this file easily accessible. Click **Finish**.

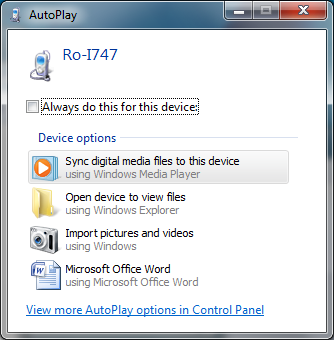


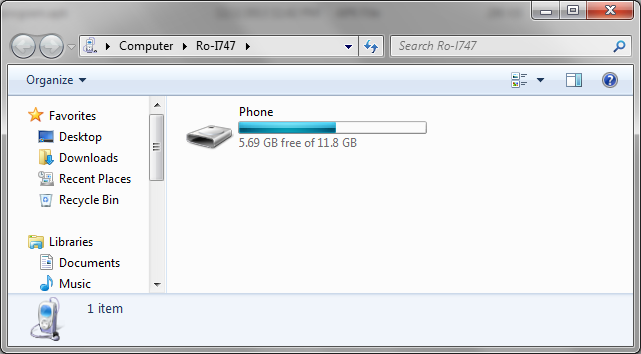
Congratulations! Your APK has now been created and is ready to be installed on compatible Android devices. The following set of instructions will guide you in installing your APK.

## How to install through APK:

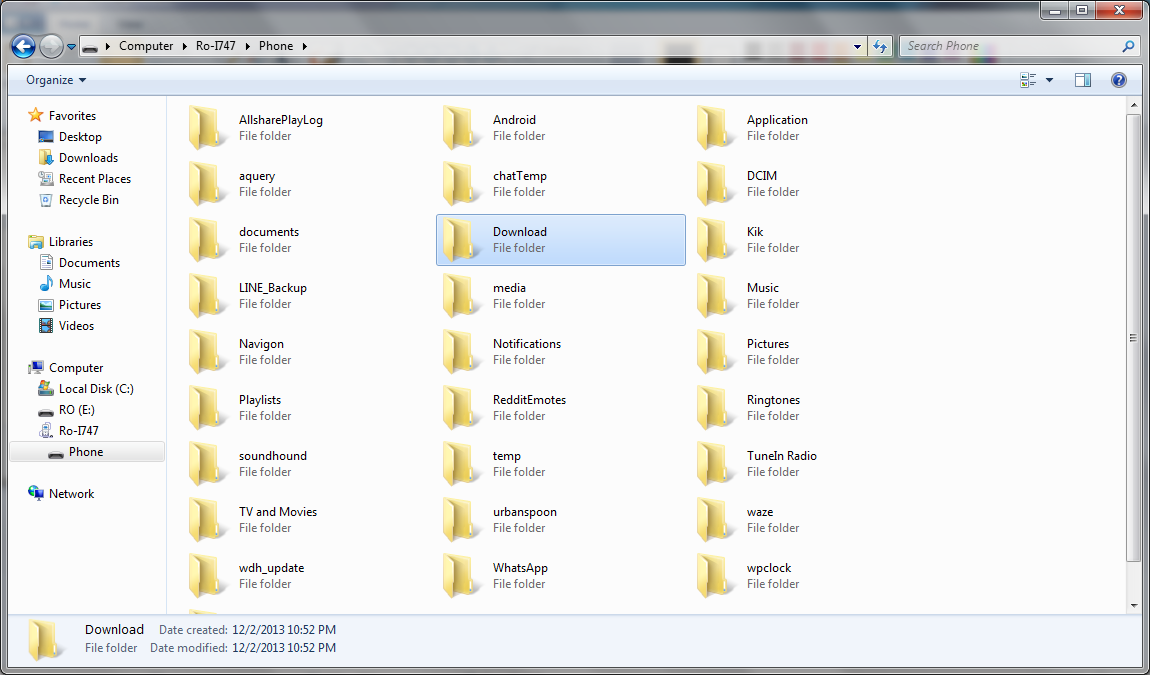
Before anything, to ensure being able to install our application, look online for tutorials on how to install Android Applications without the market if you are unfamiliar with installing an application without the Google PlayStore (Ex. <http://stackoverflow.com/questions/8795244/installing-apk-app-on-android-without-market>).

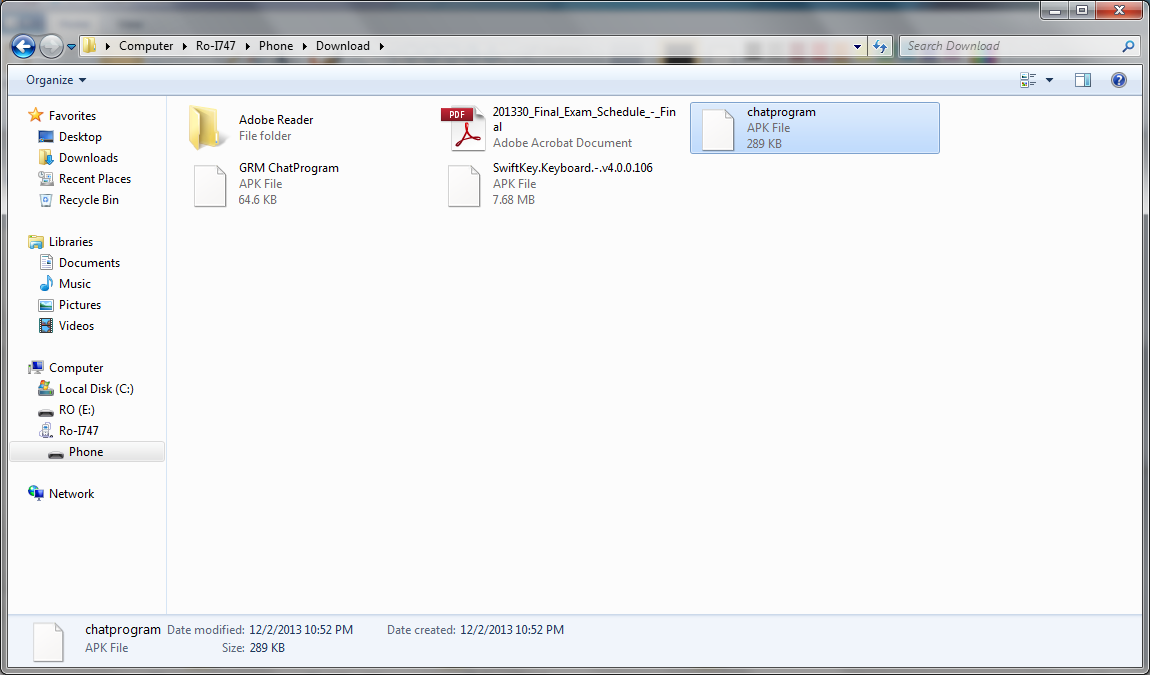
1. Navigate to the APK in your company.
2. Connect Android device through USB, and open devices to view files.



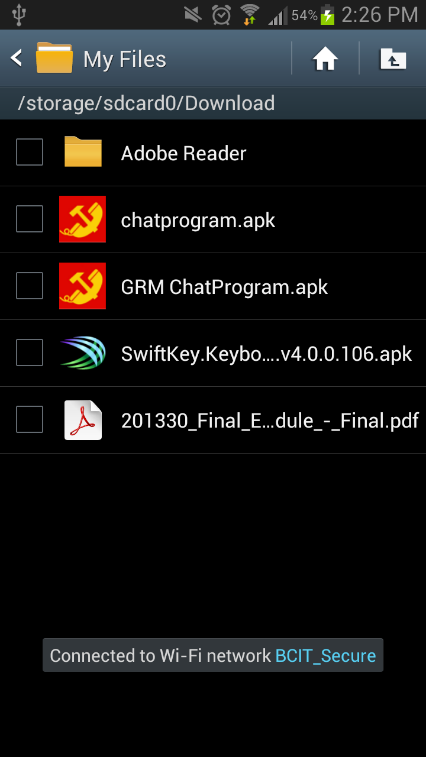
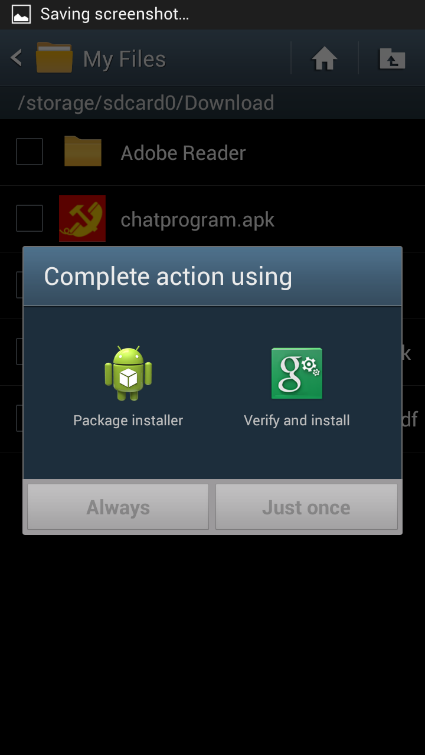
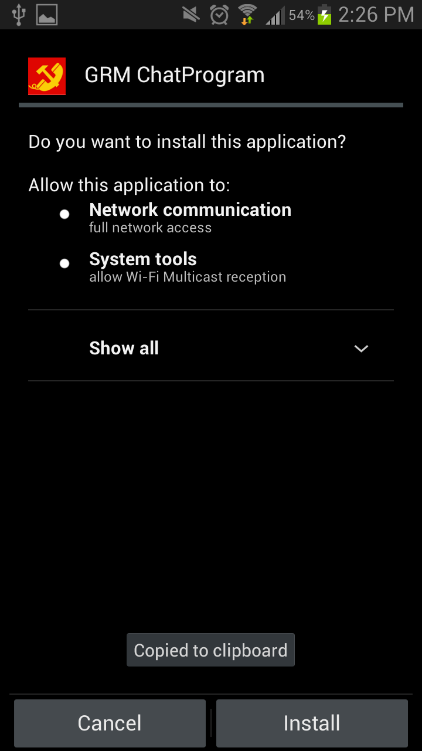


1. Navigate to folder of your choice, and place the APK into the folder.





1. Using a file manager application (Any file manager downloaded from the App Store, Ex. ASTRO File Manager, etc.) Locate the APK within your phone.

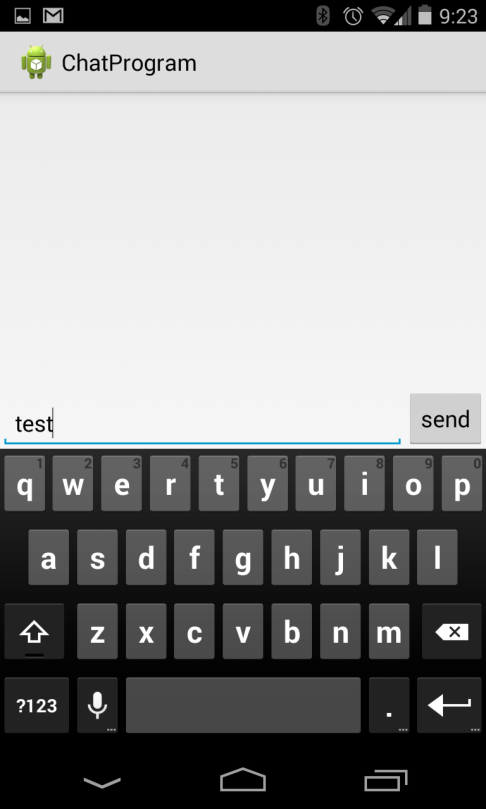


# Test Cases

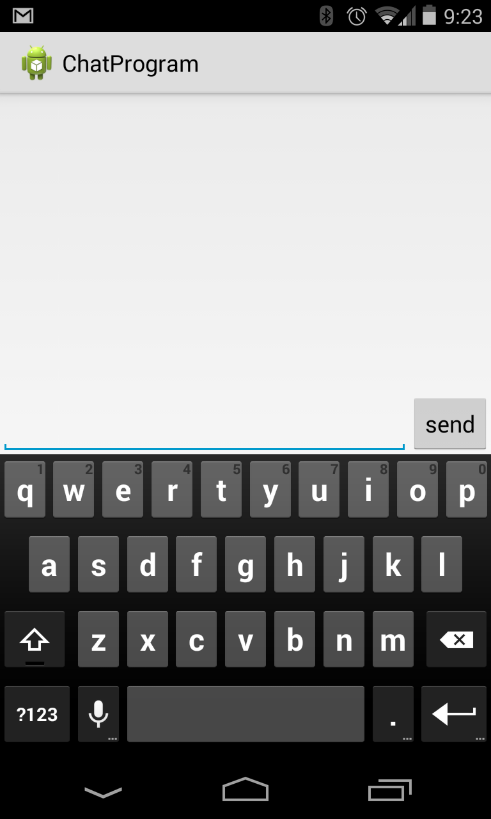
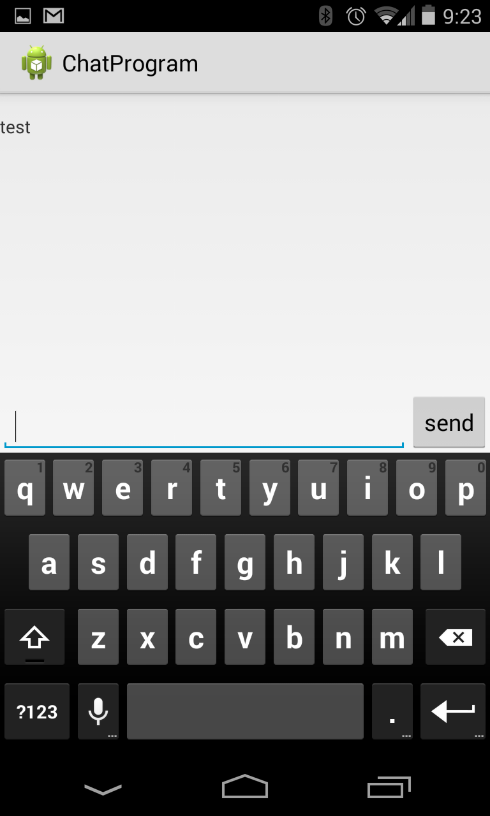
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test** | **Tests Description** | **Tools Used** | **Expected Result** | **Pass/Fail** | **Notes** |
| 1 | Initial Start Up | Nexus 4 | The program should open on an Android device | Pass | See Figure 1 |
| 2 | Typing Text | Nexus 4 | Text written with the keyboard should display | Pass | See Figure 2 |
| 3 | Typing multiple lines | Nexus 4 | Lines of text would be on separate lines. | Fail | See Figure 3 |
| 4 | Custom Username | Nexus 4 | Able to add a custom username to conversation | Pass | See Figure 4 |
| 5 | Displayed typed text correctly | Nexus 4 | Text is on its own line each time entered and not overwritten | Fail | See Figure 5 |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

# Figures

## Test 1: Initial Start Up (PASS)

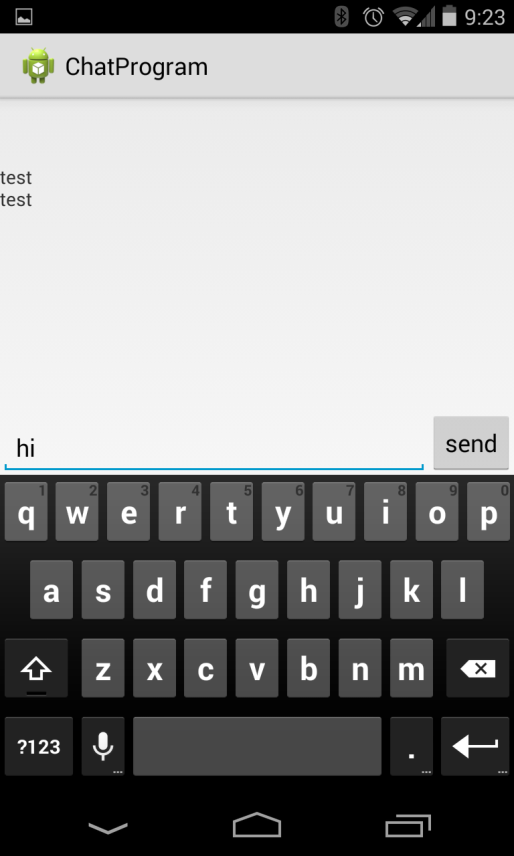
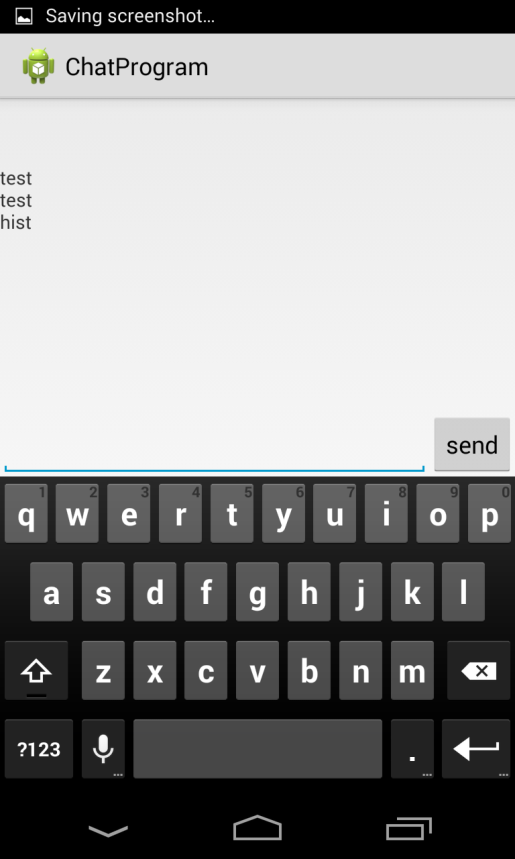


## Test 2: Typing Text (PASS)

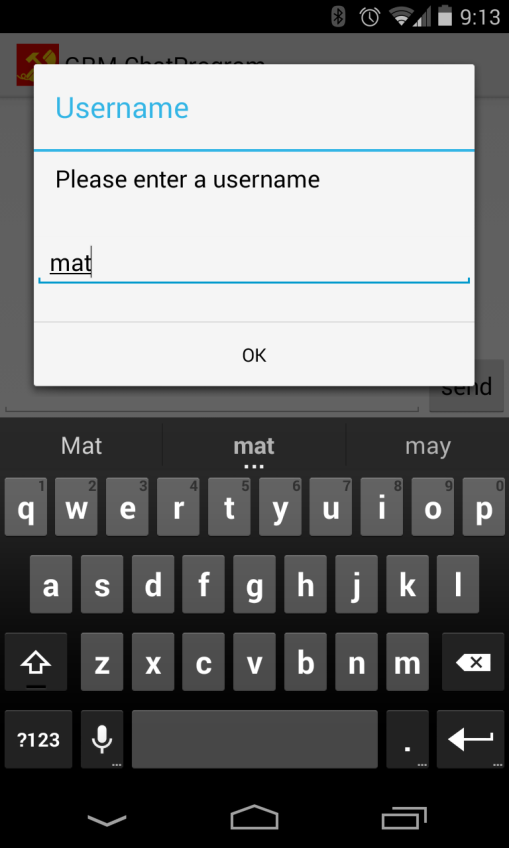
1. (b)

## Test 3: Typing multiple text (FAIL)

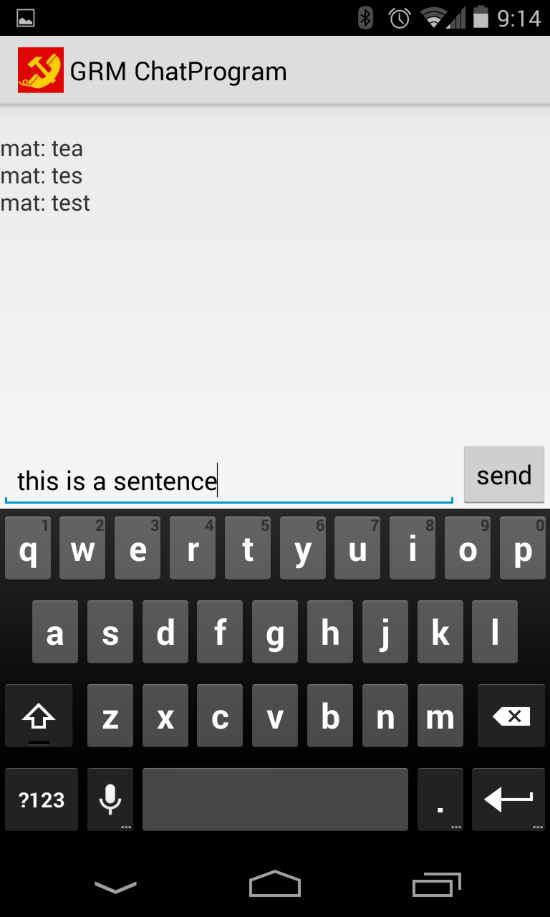
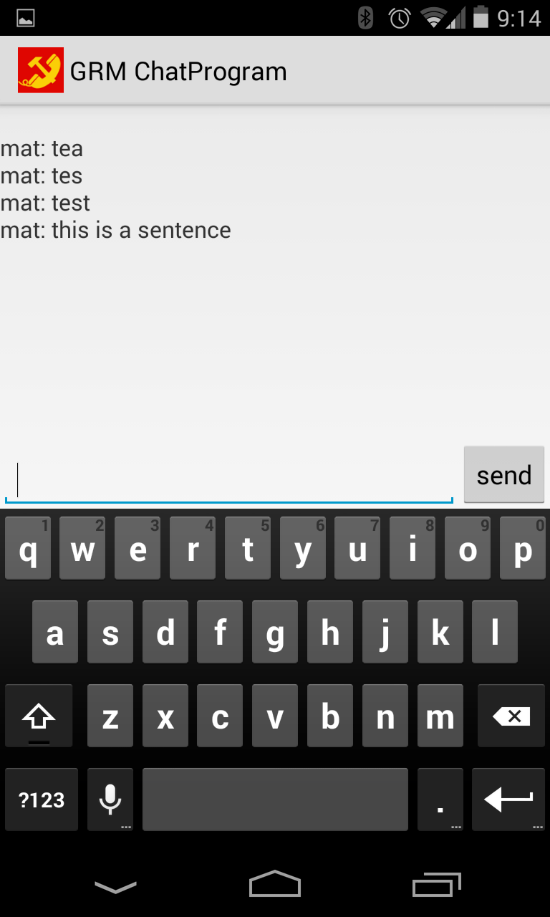
 

(a) (b)

## Test 4: Custom username (Pass)



## Test 5: Display typed text correctly (PASS)

(a) (b)