# Contents

[Contents 1](#_Toc375927199)

[List of pictures 2](#_Toc375927200)

[List of tables 3](#_Toc375927201)

[1 Introduction 1](#_Toc375927202)

[1.1 Android platform briefly 1](#_Toc375927203)

[1.2 Widgets 1](#_Toc375927204)

[1.3 History 1](#_Toc375927205)

[1.4 Extensibility 2](#_Toc375927206)

[2 Aims of the work 3](#_Toc375927207)

[3 Existing solutions 4](#_Toc375927208)

[4 Design and architecture 5](#_Toc375927209)

[5 Implementation 6](#_Toc375927210)

[6 Testing 7](#_Toc375927211)

[7 Conclusion 8](#_Toc375927212)

[8 Bibliography 9](#_Toc375927213)

[A. User screeners 10](#_Toc375927214)

[B. CD/DVD contents 11](#_Toc375927215)

# List of pictures

# List of tables

# Naming conventions

Widgets, apps, applications, components, GUI widget

# Introduction

## Android platform briefly

Nowadays one of the most popular operating systems targeted on mobile devices is undoubtedly Android. As of second quarter 2013, almost 80 percent of newly selled devices uses Android as its platform.

This operating system can not be compared to previous versions of desktop platforms, because the one of the key advantages is missing. The touch input. For majority of users, especially the ones who browse the internet a lot, the real breakthrough came with all the touches. In history not so distant, the only possibility to comfortably search the web was to use standard desktop computing inputs – mouse and a keyboard. But what about mobile users? They tend to operate in much more compact environment with limited access to all the gadgets connected with desktop computing. So all users, hungry for browsing the web were sentenced to use arrows or in better cases, some kind of trackball. The real power of fast browsing basically came with three things – wifi/3G internet connection, high definition screens and the ability to input via touches. Its important to stress, that the first two mentioned would not significantly improve the browsing experience itself. But with the connection of the touch screen everything took the whole new direction. Let’s have an example. Filling the form on e-shop – a classic task on desktop computer. What a nightmare on pre-historical mobile device. User was forced to skim through all these visible elements to finally acquire the targeted one. The whole operation was matter of seconds. Compared to touch input, this was very poor performance. Average user, when given low-end Android device, is able to target the form field in a matter of tens or hundreds of milliseconds. And when it comes to filling some input, the screen of the mobile gadget is so large, that a full version of qwerty keyboard fits in.

One of the key values, that caused the major breakthrough of Android is its Linux kernel ,which is hidden under the shiny hood of all the functions, that are presented to the final user. This “glue” is used to offer an interface to all the internal hardware, such as gyroscope, accelerometer and proximity sensors, that are brought to the end user to enrich the interaction with either default or third party applications.

Open source nature of Android, device manufacturers, professional and enthusiast developers are free to update the code undeApache licence.

## Widgets

Synonym for application, or app. Widget is not connected only with Android platform, it can be easily found on desktop PC, as well as on Windows Phone, or iOS. When compared to common software packages, or even large software suites, widget is a lightweight application, that does not necessarily consumes all users attention. Easily imagined example would be widget, that shows the current CPU/memory load, weather forecast app, or calendar.

Among the coding society a wide variety of terms related to widgets is used. Content of this work will be further discussed, but it is targeted on GUI widgets. These elements are previously known from desktop computing under the name components. All the checkboxes, radio buttons, faders and even grouped elements like dialog boxes - these are all GUI widgets.

Android comes with rich set of widgets, that are professionally looking, ready to be used in an out-of-the-box manner. Adequate event-driven mechanism is included, so that these widgets respond accordingly to user actions, like touches, clicks, drags, etc.

Android developers have basically two possibilities to include existing widgets into their apps.

* Specify the layout with all the widgets in XML document (preffered way)
  + Pros and cons,
  + Easily changed without the explicit need to recompile source
* Set the layout programmatically
  + Pros and cons

But what about the situation, when none of the supported widgets fits the developer’s needs and he wants to design a custom one? Of course, there is the possibility to create own components, that might extend the basic functionality of existing ones, or to offer a whole new approach. The question of setting up the initial layout stays almost the same. While there is of course need to specify custom functionality programmatically, once the widget is set up, it can be added to existing layout via XML, or by code change.

Example of programmatic layout setup and XML-based

## History

Based on Linux kernel

Originally developed by Android Inc. during its development phase supported by Google and in 2005 bought by Google

Reveiled in 2007 as well as Open Handset Alliance – consorctium of telecommunication , hardware and software companies, used to advance open standards for mobile devices. Also first device, that used Android as a primary operating system was HTC Dream. Critical reviews stated, that the phone itself looks both futuristic and retro, but it is not an “iPhone killer.” On the other hand Android as an operating system was received quite well and bright future was yet to appear.

Android corresponds to real world actions, like touching, sliding, pinching, reverse pinching, etc. Direct manipulation in general.

Other visual stimuli are presented through the ability to receive events through internal hardware devices, like gyroscope, accelerometer, proximity sensors, GPS module and many others.

Android was initially released in 2007, first device, that used Android as an operating system was HTC Dream.

## Extensibility

Android is a very popular platform for developers. As of April-May 2013, 71% developers targeted their applications on Android.

Easy to start bundle, free to use even for commercial development, mention more than ADT bundle, but also the IDE based on IntelliJ IDEA (compare this one to Eclipse, user satisfactory, usage)

What is necessary to perform to write custom apps

Bridge between hardware devices(gyro, sensors, …) easy used through API

## History of Android releases

### Android 1.5 Cupcake

First major Android release was introduced April 30, 2009. Noticeable features, which were presented in this release were: newly integrated home screen widgets, folders visible on home screen, stereo Bluetooth support, ability to copy/paste within web browser, and video recorder with playback. Devices worth mentioning were ie. HTC Hero, Samsung Moment, or Motorola Backflip.

### Android 1.6 Donut

Donut release presented the quick search box, interface for camera was updated, as well as Android Market, and gallery. Battery usage indicator and text-to-speech engine were added. Devices, which benefited from these new features were LG Optimus, Samsung Galaxy, or Sony Xperia X10.

### Android 2.0 – 2.1 Eclair

Google Maps Navigator in beta version was firstly presented with the Eclair release. Other features were: update browser based on Chrome engine, support for multiple accounts, or improved keyboard. High-end devices, which used Eclair were HTC Desire, Samsung Galaxy S and Motorola Droid.

### Android 2.2 – 2.3 Froyo

This release finally added support for Adobe flash, as well as multiple keyboard languages. Certain actions were taken and the whole system was optimized towards higher performance with slightly decreased computational demands. Noticable devices of this period are Nexus One or HTC Evo.

### Android 2.3 – 2.3.7 Gingerbread

As a minor release, Gingerbread offerred noticable UI refinements, support for NFC and faster text input. From all the devices from this period, Nexus S, HTC Sensation or Samsung Galaxy S II stand out the most.

### Android 3.0 – 3.2 Honeycomb

Honeycomb release meant a breakthrough mainly for table users, because holographic UI design was added, as well as action bar, improved multi-tasking or updated Android apps, which came in the bundle. Among others, Motorola Xoom, Eee Pad Transformer or Samsung Galaxy Tab were presented.

### Android 4.0 Ice Cream Sandwich

Release 4.0 meant improved multi-tasking, ability to unlock the screen by face detection, resizeable widgets and the option, that soft buttons can replace hard keys. This era presented Galaxy Nexus, HTC One or Sony Xperia T.

### Android 4.1 – 4.3 Jelly Bean

Jelly Bean came with streamlined UI, multi-user profiles for tablets, lock screen widgets and OpenGL ES 3.0. Devices worth mentioning are Samsung Galaxy S3 and S4, HTC One or Sony Xperia Z.

### 

# Aims of the work

# Existing solutions

## What is the difference between this solution and UI protocol?

# Design and architecture

## Protocol

## Modularity

# Implementation

# Testing

# Conclusion

## Causes of higher latency + suggestions on how to decrease it

## Future development

# Bibliography

1. A brief history of Android releases. *AndroidGuys.* [Online] 2013. [Cited: December 20, 2013.] http://www.androidguys.com/2013/10/10/key-moments-history-android-releases/.

2. Android. *Wikipedia.* [Online] 2013. [Cited: December 20, 2013.] http://en.wikipedia.org/wiki/Android\_(operating\_system).

# User screeners

# CD/DVD contents