ITRW 324

Group 18

Phase 2

Mobile Application



Mr P Rossow

15 August 2017

***Web services Technical : Phase 2***

**Team members:**

Ruan Swardt

Jéan Labuschagne

Tiaan Rademeyer

Stefan van Schoor

Ruan van Zyl

Henko van Staden

Charl Barkhuizen

**Demi:**

Thys Wentzel

**Lecturer:**

Mr R Rossow

**Subject:**

ITRW 324

Contents

[Introduction 4](#_Toc493264188)

[How will we do this? 4](#_Toc493264189)

[Technical tools used in this phase: 4](#_Toc493264190)

[Mobile: 4](#_Toc493264191)

[Web service: 4](#_Toc493264192)

[Key struggles: 5](#_Toc493264193)

[The methodology 5](#_Toc493264194)

[Whats next? 5](#_Toc493264195)

[What have we learned throughout this phase: 5](#_Toc493264196)

[Why choose single platform 6](#_Toc493264197)

[Why use Android studio 6](#_Toc493264198)

[Technical aspects 6](#_Toc493264199)

[Push notifications 6](#_Toc493264200)

[Kenmerke van ŉ push notification 7](#_Toc493264201)

[Idee van ‘n push notification in die app 7](#_Toc493264202)

[Log in en Registrasie 7](#_Toc493264203)

[Hoe die log in by die app werk 8](#_Toc493264204)

[Probleme met die push notification 8](#_Toc493264205)

[Probleme met die log in 8](#_Toc493264206)

[Procedure and requirements to publish application in application store 9](#_Toc493264207)

[Workflow: 10](#_Toc493264208)

[Conclusion 11](#_Toc493264209)

# Introduction

After the completion of phase 1, we started developing our mobile application. This application serves a different purpose than our main web application. Where the web application focused more on the physical inventory in storage and the insertion of inventory into the storage facilities the mobile app is designed to carry some of the weight of the employees as well as give structure to the process of developing a vaccine from start to finish.

Some implementations to do this includes a tracker a selection option etc.

How will we do this?

We developed a native mobile application to serve as a tracking device to enable employees to track each batch of vaccines created and be up to date with other sections of the process. Along-side this tracker we have created a basic interface where an employee can initiate a batch and update the database with the new values of current stock levels. By doing this the user will save at least 45-90 mins each time a batch needs to be created by eliminating trivial stuff such as basic calculations.

Technical tools used in this phase:

### Mobile:

* Android studio
  + IDE used to develop the application
* Java (jdk)
  + The main programming language, java was required to create a native app instead of a mobile web application.
* Xml
  + Interfaces, and GUI programming language works with java in android studio.
* JSon
  + Json is needed to parse data from database via the web service(php) and send it to the application file.
* Gradle
  + Android studio tool used to build project
* Emulator
  + Emulators are effective for testing your app easily.

### Web service:

* Php
  + Web service language.
* MySQL
  + Database host.
* Xampp
  + Allows mySQL and Appache.

## Key struggles:

Having no particular app developing skills prior to this phase, one of the key challenges were not only to understand the language in which to program and its syntax but also learning the software (Android studio etc) that goes along with it.

Another key challenge was the combining of the different phases and activities of the project into a coherent application. Although we could not fully accomplish everything we have desired before the deadline we are excited to take on the challenge of creating a coherent cloud-based system in phase 3.

The last major obstacle was to divide the work into small pieces to hand out activities and responsibilities because of the nature of the application. We really strive to handle our planning in a more effective way for the third and final phase.

## The methodology

Initially we went with the agile method model, this means being flexible, iterative and continually producing working software. As a result of a few time wasters we realised we were getting behind schedule. We then took a priority mythology approach by assigning priority values to all uncompleted tasks. We merged this with the iterative nature of the agile method and produced important working software (although it was small parts). The major problem with this was that we transitioned to late and struggled to ‘*fail fast*’. In the next phase, we will aim to implement this fail fast approach more effectively by reducing the time wasted on non-working software and rather change to a more plausible approach.

Whats next?

The next phase of the project includes launching our mobile and web application on a cloud based server such as Amazon, Azure etc. The next phase also includes fixing bugs for previous phases as well as integrating the mobile and web application to function side by side.

## What have we learned throughout this phase:

Phase two was fun and exciting but at the same time terrifying and frustrating. With our basic java background the Java coding for the app was not that difficult, but we had to adapt to some new syntax and layout properties.

The most important things we learned throughout this phase include:

* Creating a native app from scratch
* XML programming language
* The function of JSON in mobile development
* Learning how to fail fast (still a working progress)
* Working as a team to discover and solve problems

# Why choose single platform

We decided to go with single platform because it is easier to get free development tools for android other than for iOS. It is also easier to test the application on cell phones when using android studio. Most development tools are not free to use, especially development tools doing cross platform development.

We don’t have as much knowledge and experience in creating native applications, and therefor decided to go for single platform, such as using device features such as sensors and network access.

# Why use Android studio

We are using android studio because it already has emulators to test the application. Android studio can also set the screen size of the application and layout of the application much easier than other development tools.

Xamarin is lacking third party support for each platform and it is expensive to get. While Android studio has third party support and is free. Android studio is faster than eclipse updates are easy to install. It also uses less memory on the computer that gives faster performances. Android studio also has a lot of extra features such as templates and debug features.

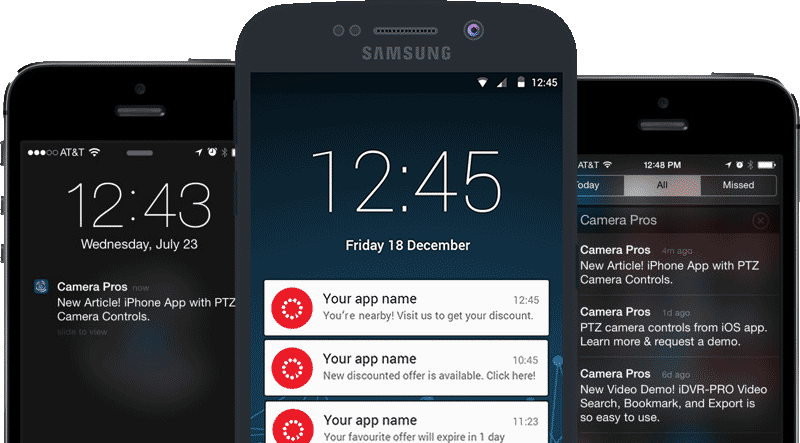
The virtual machines of Android studio works much better that on Xamarin. Also working cross platform on Xamarin the process does not go as smoothly as desired. Android studio is a good development tool to start with and then later other tools can be used for full scale development.

# Technical aspects

We have a login screen for the clients and employees and from there they are moved to the main activity page. On the main page they can choose to see information on where the vaccine is and how long it will take still. There is also a page to see the inventory levels. There is also a page that shows information on viruses and which vaccine is used for the virus. We use a home bar to move between the pages.

# Push notifications

Partykeer as ‘n persoon ‘n app aflaai dan word dit baie gebruik in die begin en later vergeet die person dalk daarvan. ‘n Push notification maak dit moontlik om met sekere gebeure ŉ klein boodskap te stuur wat dan die persoon of app se naam wys van wie dit afkom, die tyd asook meer inligting oor wat gebeur.



In die skets hierbo is daar ŉ paar voorbeelde van hoe vandag se push notifications moontlik kan lyk. Daar kan ander funksies ook bygevoeg word waar die gebruik kan kies om die boodskap of “event” te ignoreer of kan ook op die notification klik wat mens dan na die app toe vat na ŉ sekere bladsy toe.

# Kenmerke van ŉ push notification

Die volgende packages moet import word om te begin werk aan die notification :

import android.app.Notification;

import android.app.NotificationManager;

import android.view.View;

Die notification.Builder() word gebruik om jou notification te skep, daar word gekyk na 3 stappe om die notification op te stel:

.setContentTitle(“Title”) – Die title word gebruik meestal om te wys van af die “event” kom.

.setSmallIcon(R.drawable.image) – Dit is die klein prentjie wat op die status bar vertoon word.

.setContentText(“Meeting in 5min”) – Dit wys ŉ boodskap wat die persoon vir jou stuur in kleiner text as die title

Daar is baie ander .set metodes wat gebruik word om die notification te verander.

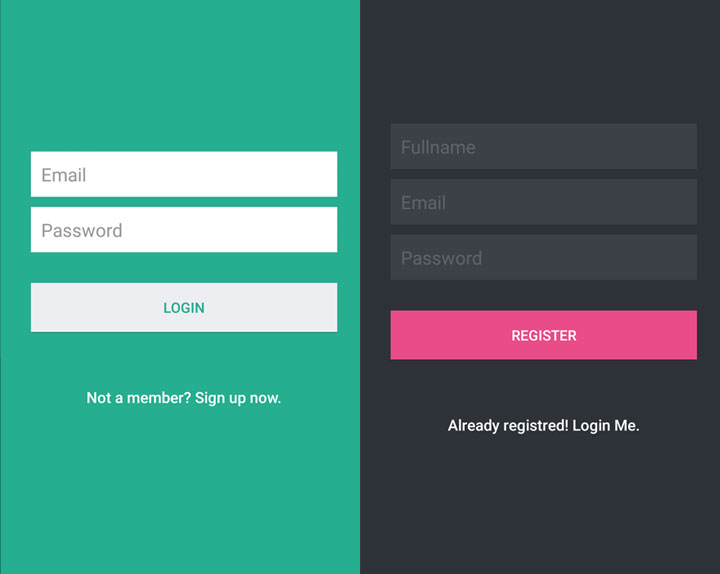
Mens moet ook ŉ notificationManager maak wat dan beheer oor waar die notification gaan. Daar moet eers ŉ int uniqueNum gestel word. Die notificationManager gebruik dan die .notify(uniqueNum, notification) om die notification te stuur na die gebruik wat daardie uniqueNum bevat.

# Idee van ‘n push notification in die app

In die app is daar ŉ bladsy waar mens dan data verkry van die databasis af oor die mense wat geregistreer is m.b.v ŉ spinner. Onder die spinner is daar twee edit text waar die een gebruik word vir die huidige gebruiker se naam en die ander wat dan vra vir ŉ boodskap. Sodra die spinner op die persoon is vir wie jy die push notification wil stuur, moet dit dan soek vir die persoon se employee nommer en dit dan gebruik as die notification se unieke nommer(uniqueNum). Sodra dit dan so gevind word sal daardie persoon die notification ontvang.

# Log in en Registrasie

Sekuriteit speel ŉ groot faktor in vandag se lewe, nie net in persoon nie maar sowel as aanlyn. As mens gebruik maak van ŉ “log in and registration” form verbeter dit alklaar die stelsel se sekuriteit omdat daar nie onnodige mense dan toegang kan verkry tot die app nie.



In die skets hierbo is daar ŉ voorbeeld van hoe ŉ envoudige log in skerm sal lyk. Met vandag waar daar tonne kuberkrakers is het mense aan beter maniere gedink om te gebruik vir sekuriteit. Een van die voorbeelde kan wees sodra mens aan teken om toegang te kry na die app toe, word daar eers ŉ “confirmation email/sms” met ŉ kode op na jou foon toe gestuur en dan dit eers in tik voordat jy na die app toegelaat word.

# Hoe die log in by die app werk

Daar was gebruik van volley om die log in meer te vergemaklik. Die volgende stukkie kode moet eers by gradle dependencies van jou Android projek se app module bygevoeg word voordat jy die volley metodes kan gebruik :

compile 'com.android.volley:volley:1.0.0'

Daar was ŉ mySQL databasis geskep op 000webhost.com sodat daar nie heeltyd localhost op die emulator gehardloop moet word nie. Die php dokumente is in die route directory van 000webhost gesit sodaar daar gekonnekteer kan word van die databasis af met die app. Daar is gebruik van wagwoord encryption met die metode md5 sodat dit die sekuriteit kan verhoog. As die gebruiker nie op die databasis is nie en jy gaan na die registrasie vorm toe dan voeg die app jou by die databasis in, maar as daar ŉ gebruiker is met dieselfde username as ŉ ander gebruiker gaan daar ŉ error boodskap uitgegooi word. Sodra die gebruiker klaar geregistreer is vat die app jou na die log in vorm toe. Sodra jy dan daar jou username en wagwoord in tik vat die app jou na sy “Main Activity” toe anders sal daar weer ŉ error boodskap op kom.

# Probleme met die push notification

In fase 2 is die hele bladsy opgestel van die push notifications. Die grootste probleem is om die data te verkry van die databasis af sodat die employeeNum gelyk gestel kan word aan die uniqueNum. Daar was ook aan ander idees gedink om die notifications te gebruik om te wys wanneer een proses klaar maak of as dit besig is, en nie net gebruik te word om met mekaar te kommunikeer nie. In fase 2 werk die persoon se naam van wie dit afkom, die boodskap wat saam die notification gestuur word. Die spinner is die groot probleem waar daar nog ŉ probleem is dat die databasis nie data terug wil stuur nie.

# Probleme met die log in

Die registrasie werk reg en kan data in die databasis in stoor maar sodra mens op die log in klik gebeur daar niks nie. Al is al die goed reg ingetik gebeur daar nogsteeds niks nie. So die stuur van data is nie die probleem nie maar wel die onttrek van die data van die databasis na die app toe.

# Procedure and requirements to publish application in application store

Because we are designing and android application we will have to publish our application in Google Play Store. Understand and adhere to the developer program policies. There are consequences for breaking policies, it is to protect the users and Play store.

A developer account is needed to publish the application. Set up a merchant account if you want to publish paid for applications.

Use the quality guidelines to test the application for different devices.

* Standard android design guidelines: should not reinvent expected functions such as the home button, should not replace system icon when it triggers standard UI behaviour.
* Navigation: all dialogs are dismissible using the back button, pressing the home button navigates to home screen.
* Notifications: Multiple notifications are stacked into one where possible, persistent in ongoing events such as music play back, uses notifications only to indicate change relating it the user personally.

Functionality requirements

* Functional behaviour appropriate level of permissions.
* Permissions: request absolute minimum permissions needed to support core functionality.
* UI and graphics: Needs to support landscape and portrait (if possible).
* User/Application state: Should not leave any services when in background, unless related to core capability.

Build a release ready apk.

Plan the applications Play store listing.

Prepare descriptions, promotional graphics, screenshots and videos for the play Play store’s page.

Include link to privacy policy.

Define application’s device compatibility.

Opt in to the right distribution options.

Set in application products and subscriptions.

Determine the applications content rating.

# Workflow:

|  |  |
| --- | --- |
| **Member** | **Explination of responsibilities and duties** |
| Jean | Group leader, overall assistance as well as front and back end developing including php, Java, xml. Arranging meetings, keeping up team moral and similar administrative duties also falls under my responsibilities. |
| Stefan |  |
| Ruan |  |
| Ruan |  |
| Charl | Design User interface |
| Tiaan |  |
| Henko | Design User interface |

# Conclusion

In conclusion, we are excited to continue to phase 3 where we will be able to apply the knowledge that we have obtained to fix all previous mistakes and present a fully functional system that consists of a web application, mobile application and cloud integration.

We have made many mistakes throughout these previous phases and one key change we would like to make is to fail faster, this will enable us to spend more time on the working parts of the system while giving lower priority to those areas that can hold you back with little fruits to bare.