Birkbeck, University of London
School of Computer Science and Information Systems

MSc Computer Science
Project Proposal

PATHOGEN TRACKER

AN ANDROID APP TO ASSIST WITH DATA CAPTURE FOR PATHOGEN RESEARCH PROJECTS

Author: Laurent Mignot
Supervisor: George Roussos

TABLE OF CONTENTS

1 ABSTRACT

The Zika Virus was declared a Public Health Emergency of International Concern by the World Health Organisation in February 2016. In recent years, a number of research projects have been launched to investigate the virus, its vectors, origin, and spread.

This proposal identifies the challenges faced by a Zika Virus research project in quickly and accurately accumulating data for genomic analysis. It proposes a cost-effective, practical, and bespoke solution using mobile technology, that can allow researchers in remote locations to quickly collect, validate, and securely transmit data over distance for analysis. It then outlines, justifies and elaborates on the tools and methodology that will be used to implement the suggested solution within the completion deadline.

2 BACKGROUND

2.1 Zika virus, research project, data capture

Zika virus surveillance in human and mosquito populations in Cape Verde [1] or the Zika Project (ZP) is a research project at the London School of Hygiene & Tropical Medicine (LSHTM) investigating the Zika virus (ZIKV) outbreak in Cape Verde. ZP and similar research programs[2] collect and analyse data in order to map the origin, spread, and variants (including drug-resistance) of ZIKV.

ZP's team communicated that their project would benefit from a bespoke technological solution to challenges particular to ZP. A further meeting with ZP's team resulted in broad understanding of high level requirements. A plan to meet those requirements forms the basis of this proposal.

One aspect of ZP involves the on-site capturing of physical samples such as DNA samples from mosquitos and human blood and urine samples. The data and metadata derived from the samples collected on site is transmitted and analysed by ZP's researchers. ZP's team noted that the accuracy and reliability of the data capture process is complicated by factors such as location, distance, and time differences between capture, research and analysis, and data transmission. These factors are explored in section 2.3.

2.2 A virus and its vectors

Zika, originally discovered in Uganda in 1947, recently resurfaced with major outbreaks in Oceania in 2007. In 2015, it gained worldwide notoriety after a large-scale outbreak in Brazil. Between October 2015 and May 2016 thousands of reported cases in Cape Verde led that country to declare a ZIKV epidemic.[1,3–6]

ZIKV is carried by *Aedes* mosquitos. *Aedes aegypti* is found predominantly in tropical areas but *Aedes albopictus* can also be found in southern Europe and northern USA. Consequently, there is concern that ZIKV infected individuals travelling from affected countries such as Cape Verde could carry the virus to the African mainland, Europe and North America.[1,3,5,7,8]
Unfortunately, a number of outbreaks have only been identified as ZIKV after a time delay. It has been suggested[6,9,10] that misdiagnosis may be one of the causes of the time delay as there are similarities between the symptoms of ZIKV and other more common arboviral infections such as Dengue Fever.

2.3 Challenges

2.3.1 How much data do we need?

A report by the PHG Foundation[11] states that, *"pathogen genomes cannot be usefully analysed in isolation"* and that what is needed is to be able to combine genomic data with relevant epidemiological metadata in order to extrapolate meaningful information.[12]

Quickly and efficiently accumulating and combining relevant data and metadata would be a solution towards increasing the effectiveness of genomic data analysis as the effectiveness of this analysis improves in tandem with increases in the quantity of relevant epidemiological data collected.

2.3.2 Time and Distance

ZP's research and sampling is done in Cape Verde but ZP's data analysis is primarily done at LSHTM, some distance away in London. Since new ZIKV outbreaks are primarily detected in remote, tropical locations it is reasonable to assume that conditions of distance between sites of data collection and analysis are likely to apply to other ZIKV research projects.

Widespread internet availability[13] may reduce time delays in data transmission but fails to address problems in data collection, especially when data capture is done in traditional ways such as paper forms and serological samples, etc.

In the diagnosis of infectious disease, whole genome sequencing is now assisting researchers in identifying culprit pathogens with greater precision than when using traditional methods[10,12]. Analysis of collected data is already being used to develop potential vaccines[6], and the search for antiviral solutions is ongoing.[14]

Recent advances in DNA sequencing, including portable sequencers such as the MinION[15], should allow for tools to be developed for researchers to sequence a DNA sample, extract relevant information, attach that information to related metadata, and submit said data electronically to a central data set. The process could almost occur in real-time as wireless connectivity would allow for frequent and rapid data transmission, although cost may factor as roaming cellular data charges can be considerable.

2.3.3 Accuracy

Data can be collected in a variety of ways including from paper forms, tagged serological samples, and digital entries in electronic systems. If meaningful and reliable analysis is to be performed the data collected needs to be as accurate as possible[12]. Human error is a factor and so any acquired information needs a robust process of validation able to identify and minimise errors and improve accuracy.

2.3.4 Cost

Cost should always be considered.

Time and Distance between sampling and analysis can increase transport and transmission costs. The quicker the information can be gathered and transmitted within a given time the more information can be collected, improving any cost-benefit ratio.

The accuracy of data which has been automatically tested and validated could result in savings in personnel and/or the time taken to manually validate information.

3 PROPOSED SOLUTION

A tailored smartphone application ('App') could facilitate ZP researchers to quickly and accurately collect and transmit information for analysis.

Although the focus will be on tailoring a solution to satisfy ZP's particular requirements, a solution could be designed so that it is applicatory to similar projects with similar needs, but that is beyond the scope of this proposal.

The complete solution will consist of:

1. App for data collection, and
2. a server-side system to handle communication between App and the data persistence layer, plus any user authentication and access control required by ZP ('Server').

Due to time constraints, App will take priority. Server will be met by an appropriate service that satisfies the requirements specified in section 5.2, below.

App could be implemented in different ways. These include:

- A browser-based application: this would work on any portable device with a web browser, including smartphones, tablets and laptop computers
- An application built for personal computer operating systems, including: macOS, Linux, and Windows. Laptop computers run these systems, satisfying the need for portability
- An application built for smartphone and/or tablet devices, including Android and iOS.

Option c has been chosen for App for reasons described in section 3.1, below.

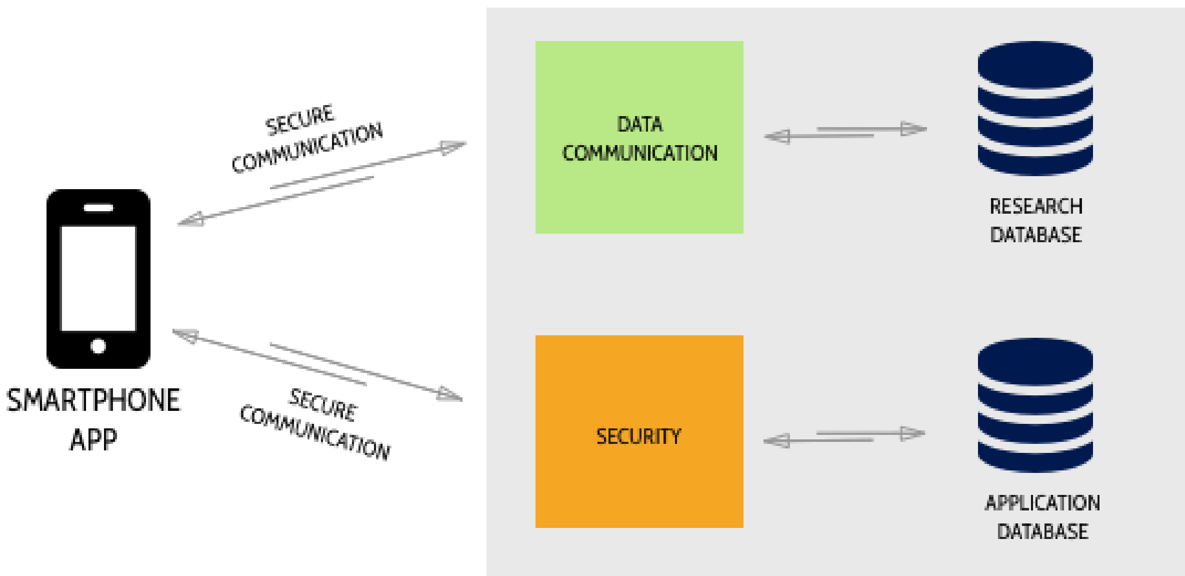

Figure 1 – High-level solution concept.

3.1 Smartphones – ubiquitous, affordable, portable and practical

Ubiquitous

As of Q4 2016, worldwide mobile phone subscriptions totalled 7.5 billion[16]. Roughly 52% of subscriptions are smartphone subscriptions. While subscription and user numbers do not necessarily correlate, the ubiquity of smartphones is unmistakably evident and so the likelihood of a user being in possession of a smartphone is high.

Affordable

The ubiquity of smartphones reduces the likely quantitative cost of purchasing units of new hardware. Budget model smartphones can also be purchased for relatively low prices.

Portable and Practical

Smartphones are more portable than laptops and have longer battery life. This means App can be used for longer periods of time without requiring access to electrical outlets.

Most smartphones include a number of useful sensors and devices. These can allow ZP researchers to capture a broad range of information with just a single device. For instance, a smartphone's camera allows photos to be taken and attached to a data record and ZP researchers could use a smartphone's GPS sensor to give precise geolocation coordinates with the data they have collected.

4 SIMILAR TECHNOLOGICAL SOLUTIONS

This section explores some other smartphone app solutions which have used mobile technologies to assist with research and data capture.

4.1 ResearchKit

Apple's ResearchKit framework (RKF) aims to assist researchers performing medical studies by providing them with tools for enrolling participants, assigning tasks, carrying out surveys and displaying data in a meaningful way. It uses the various sensors in an iPhone to collect objective data relevant to a given study. Since Apple released ResearchKit[17] in 2005, a number of apps using the ResearchKit framework (RKF) have been developed. One website[18] lists 33 RKF based apps as of February 2016. Additionally, frameworks for Android have been released which provide similar features.[19,20] While these are frameworks, not applications, their rapid adoption and use by a number of research projects evidences that there is interest in applying mobile technologies to data collection and research.

4.2 FieldTrip Open

FieldTrip Open[21] is a platform for building mapping and data-collection apps maintained by EDINA[22]. The platform is composed of various components: mobile apps, web-applications for creating and previewing surveys, plus a middleware app to handle data transmission between client and database.

Investigation of the mobile app component's source code – available on GitHub[23] – revealed the approach used to create cross-platform apps (iOS and Android are supported). The solution uses Apache Cordova[24]. Cordova wraps an HTML/JavaScript app into a native container and provides an Application Programming Interface (API) that allows JavaScript code to access a device's native functions and hardware components. The data entry workflow is built from survey configuration files in JavaScript Object Notation (JSON) format. This means workflow can be changed without the need to rebuild and publish the mobile app(s).

4.3 Mobile Data Collection Applications

An internet search for "data collection apps" turned up a fair number of results available commercially. Brief investigation revealed a common set of features including:

- Data entry "design" – a user can design their own forms for use in data collection
- Mobile apps for both major mobile platforms, iOS and Android
- Offline capability – data capture is still possible if the device is offline
- Storage – a user can choose from a fairly large range of online solutions to store the collected information

4.4 Conclusion

Cost implications aside, certain of the apps available commercially could be considered for use for certain of ZP’s requirements. However, the aim of this proposal is to produce a tailored app specifically suited for all ZP's particular needs. The commercially available solutions explored cater to a wide audience and provide a broad set of features, many of which may be superfluous. This could introduce an unnecessary added layer of clutter and complexity for ZP.

Additionally, if a particular feature required by ZP’s team is not available in an off the shelf solution it could mean that certain of ZP’s requirements may need to be sacrificed or compromised, or bespoke work would be carried out, which would introduce additional time and cost.

5 REQUIREMENTS AND DELIVERABLES

ZP's team communicated a considerable number of features they deemed useful for this proposal. Due to the completion deadline, features will be restricted to the Minimum Viable Product (MVP)[25] that meets the priority functions ZP requires. Once the MVP is complete, if sufficient time remains, additional secondary features could be prioritised and implemented.

5.1 App Requirements

Detailed requirements for App will be determined together with ZP's team throughout implementation and documented, but ZP's team have communicated the following as being high-level priority features.

App must:

- Allow for different types of users – One researcher may focus on data related to humans, another on vector data. The requirements for these cases may differ
- Allow a user to view and modify previous entries
- Allow sensor data to be captured from the device and added to an entry
- Allow for data entry even if the device is offline
- Securely store data locally until transfer to the server system is possible, in accordance with LSHTM guidelines for research data[26]
- Be easy to use – a goal of this project is to improve the data capture process
- Read data over Bluetooth – the MinION[15] allows for portable DNA sequencing and connects to a computer via USB. Relevant data could be imported by the app over Bluetooth and attached to an entry.

It should be possible to use the app for other research projects; however, the following considerations apply:

- Data capture requirements may differ, the app would need to be designed so as to enable different workflows, perhaps driven by some sort of "workflow description language"
- Institutions may have different policies related to storing and transmitting research data
- The data's destination will likely be different
- Satisfying data security and reliability requirements for many different projects introduces additional complexity.

5.2 Server Requirements

Due to time constraints, the Server will be provided by an already existing server platform. Server should, where applicable, comply with LSHTM's guidelines for research data[26], and Information Management and Security[27]. ZP's exact requirements will be determined and documented in the report.

Requirements can include but are not limited to:

- **Security and Reliability of data in storage**: the system should protect the data from unauthorised access and corruption and ensure regular backups are performed
- **Security of data in transit:** the data should be transmitted in a secure way

The chosen platform will need to support: a form of access control, user access management, secure and concurrent communication between App and Server, plus data redundancy.

6 IMPLEMENTATION

This section proposes the implementation of App and on the rational justifying the choices made when selecting between relevant options.

6.1 App

While this proposal's aim is to deliver to ZP a working application, tailored to ZP's particular needs and ready for use in September, App has the potential to be applicatory to projects with similar requirements to ZP. Extensibility and flexibility in the design of App will render it suitable for further development.

6.1.1 Smartphone platform

As of Q3 2016, with a combined 99.3% of global smartphone market share[28], Android and iOS are the two dominant mobile operating systems in use and so constitute the choice of platforms.

App must allow for data entry whether or not ZP's smartphones are connected to the internet. Consequently, data should be persisted reliably on a smartphone. A scheduling mechanism can then handle transmission when a suitable internet connection becomes available.

Android and iOS both support a local database using the SQLite Database Management System (DBMS) suitable for persisting data on the device.

Both systems include a way to schedule and execute so-called "background" tasks without user interaction. On iOS this is defined as a "Background activity"[29] and on Android, as "Services"[30].

Either iOS or Android then, are suitable for App but Android has been chosen because Android has a substantially larger market share[28] and due to its lower unit cost average[31]. The programming opportunity is an additional incentive.

6.1.2 Cross Platform and Hybrid solutions

Several frameworks exist which provide a way to write mobile apps that target multiple platforms using a single code-base[24,32–35]. Only cost free solutions have been evaluated.

There are two different approaches to creating cross platform and hybrid solutions:

1. App is created as a web site which runs in an embedded web browser in the app. Adobe PhoneGap[32], an open-source distribution of Apache Cordova[24] as a good example.
2. App is written in a language supported by the framework, (Xamarin[33] uses C#, React Native[34] and Native Script[35] use JavaScript) and converts to native code when compiled for each platform.

Limited experience with C# excludes Xamarin as a time effective option for this proposal. However, extensive web development experience makes Cordova, React Native, or Native Script viable candidates for consideration. Due to time constraints and since cross-platform compatibility is not mandated by ZP, App will be built as a native Android app.

6.1.3 Programming Language

Java is Android's standard programming language but Scala[36] and Kotlin[37] can be used as alternatives. Kotlin is a language created by JetBrains[38] that can be compiled to run on Android. Kotlin claims to reduce the amount of so-called "boilerplate code"[39] required by Java and claims to provide safety mechanisms to prevent common programming errors. It also claims to be fully interoperable with Java code.

When development starts, Kotlin's suitability will be tested, evaluated and determined. Observations will be documented and reported.

6.2 App-Server Communication

The primary function of App is to facilitate collection, validation and transmission of data to a specified destination and any database chosen by ZP's team. If App is designed with a specifiable end-point, its destination could be changed without the need to re-compile and distribute App.

Representational State Transfer (REST)[40] has been widely adopted[41] for use in Web-based services and comprises a set of constraints defining an architectural style for building client-server applications. One of the constraints describes separation of client and server so that the client does not need to declare the format or mechanism for storing data.

REST-based services communicate via Hypertext Transfer Protocol (HTTP) and HTTP Secure (HTTPS). Changing the data's destination could be accomplished by providing APP with a different base URL, so long as the URL specifies a destination with a compatible API that can recognise and process requests to retrieve, modify, and store the data. A REST-based system grants the desired flexibility for App.

6.3 Server platform

ZP are likely to use their own server platform so the server solution proposed will be used as proof-of-concept to demonstrate that App works as intended.

An internet search for the server solution providing the features the proof-of-concept requires, such as: secure database storage, a RESTful API, user management, access control, concurrent client/server communication, revealed a number of possible options, including: Microsoft Azure, Amazon Web Services (AWS), and Firebase.

Firebase[42] is a cloud-based Platform as a Service (PaaS) with an array of features available. Applicable features include: database storage, authentication management, and a REST-based API. Firebase also includes features of particular relevance to app development, including: a test platform and crash and user experience reporting.

While Firebase is a commercial solution, there is a cost-free pricing tier[43] which includes the features required by this project.

Firebase is also a Google platform which means it easily integrates with Android.

Consequently, using Firebase will allow cost-free, early and rapid integration and development of App, and so will meet this proposals time and price constraints.

6.4 Security

For App to be used by ZP, LSHTM's ethical, regulatory, and legal requirements and guidelines[26,44,45] for research data capture and storage must be complied with. ZP team's specific requirements will depend on the nature of the data collected, but the importance of data security in storage and transmission can be assumed.

6.4.1 Security of data in transit

Data transmitted between App and Server should be encrypted.

As Firebase communicates via HTTPS over Transport Layer Security (TLS) which provides end-to-end encryption between client and server, it should provide the necessary level of security.

6.4.2 Security of data on smartphones

There are at least three methods for securing data on Android smartphones:

1. **Encrypting the entire file system**. As this requires the agreement and participation of the app's end user, it could be impractical to enforce. LSHTM may have policies regulating compliance but compliance cannot be assumed and this proposal should ensure that data is secure whether or not LSHTM's policies are complied with.
2. **Encrypt the entire local database**. SQLCipher[46] allows for whole database encryption on Android. However, US export laws related to cryptography need to be considered.[47] And if used, the app would have to undergo relevant compliance processes.
3. **Encrypt and decrypt data as it is stored and retrieved**, using Android's built-in encryption libraries.

Once ZP's team has communicated its requirements the relevant choice should become apparent and will be documented.

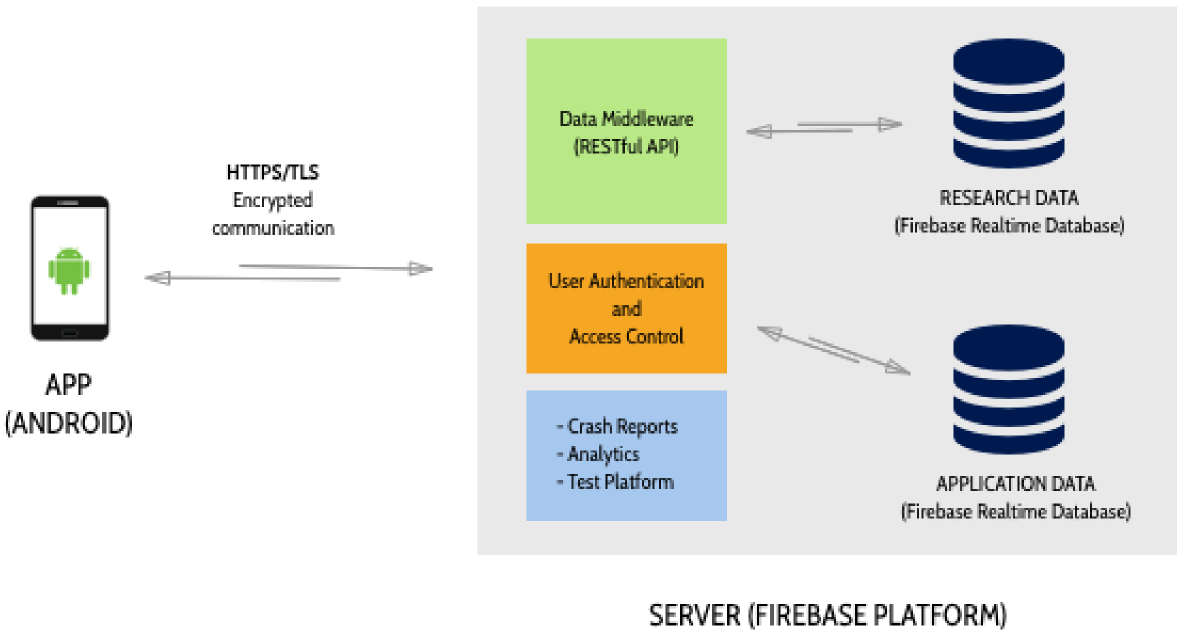

Figure 2 - Conceptual solution architecture using Firebase Platform. Android icon by KarthiK[56]

7 METHODOLOGY AND SCHEDULE

Formal adherence to team-focused Agile methodology is not practical as App is being created by a single developer.

Although ZP's input is necessary in order to ensure this proposal is tailored to their requirements, too much reliance on ZP's involvement could delay implementation and this proposal has a hard completion deadline.

Relevant principles of Agile such as: focussing on delivering required features, restricting work in progress to smaller tasks, and an iterative approach to development to accommodate changing requirements will be applied to implementing this proposal.

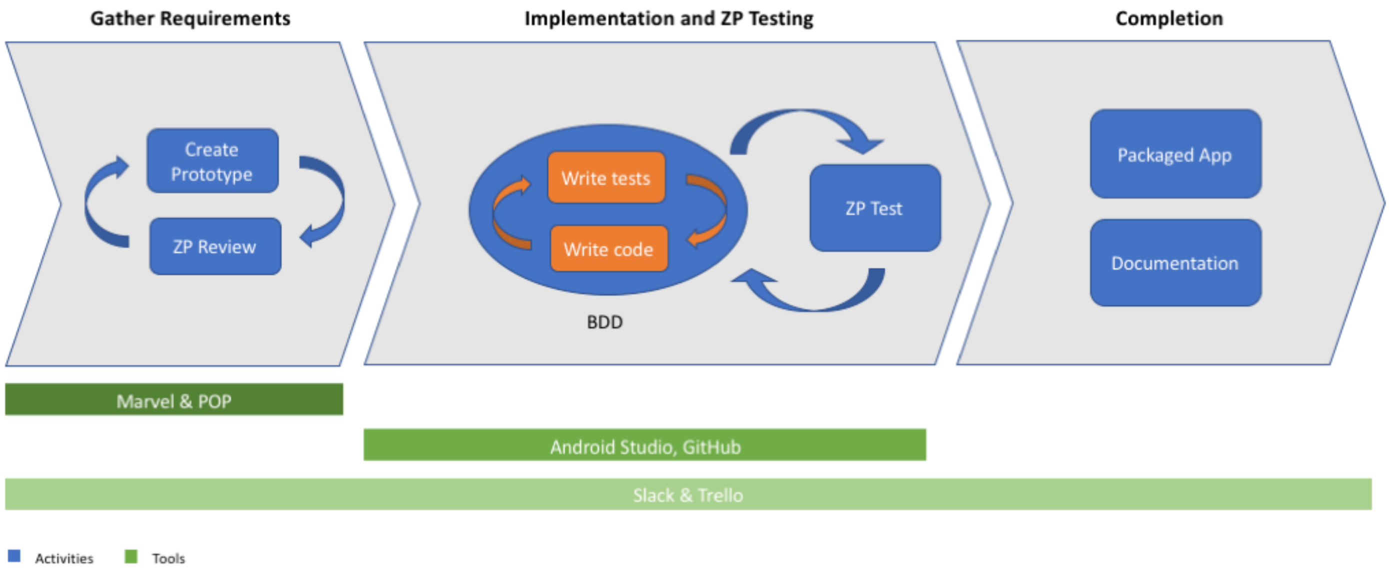

Figure 3 - Overview of implementation process and tools

7.1 Interactive prototyping

In order to solicit the collection of regular feedback from ZP's team on design variations, a suitable prototyping tool will be selected and used to create interactive prototypes of the user experience. ZP's feedback will result in further prototyping until their requirements are met.

A number of web-based prototyping applications provide the required functionality, such as: InVision[48], Marvel[49], Mockingbot[50], Proto.io[51]. Marvel has been selected due to its included smartphone app POP[52].

POP will allow paper sketches of the UI to be photographed and used to create the interactive prototypes.

7.2 Development and Testing

Rather than writing code first and then testing it, a Test Driven Development (TDD) approach breaks the software development cycle into a series of tasks such as:

1. Write tests against one specific requirement
2. Write the code to pass these tests
3. Repeat

Following TDD is likely to result in cleaner code and a reduction of programming errors, reducing the time spent debugging.

This proposal will follow a Behaviour Driven Development[53] (BDD) approach. BDD expands on TDD with a set of best practices for writing tests, and will ensure reflexivity to ZP's objectives by encouraging that tests are written against expected software behaviour rather than against implementation detail.

7.3 Tools

Slack[54] will be used as a tool to facilitate communications between individuals involved in implementing this proposal. Trello[55] integrates directly with Slack and will be used to priorities tasks throughout implementation. Android Studio, the official Integrated Development Environment (IDE) for Android will be used for developing App, and GitHub will be used for version control.

7.4 Work schedule

This proposal aims to be completed by the defined deadline. Completion means:

1. App has been developed and has passed all ZP's requirement tests
2. Documentation of App is complete

(Figure 4) is an exemplification of how implementation could proceed but changes in ZP's requirements may result in changes in the work schedule.

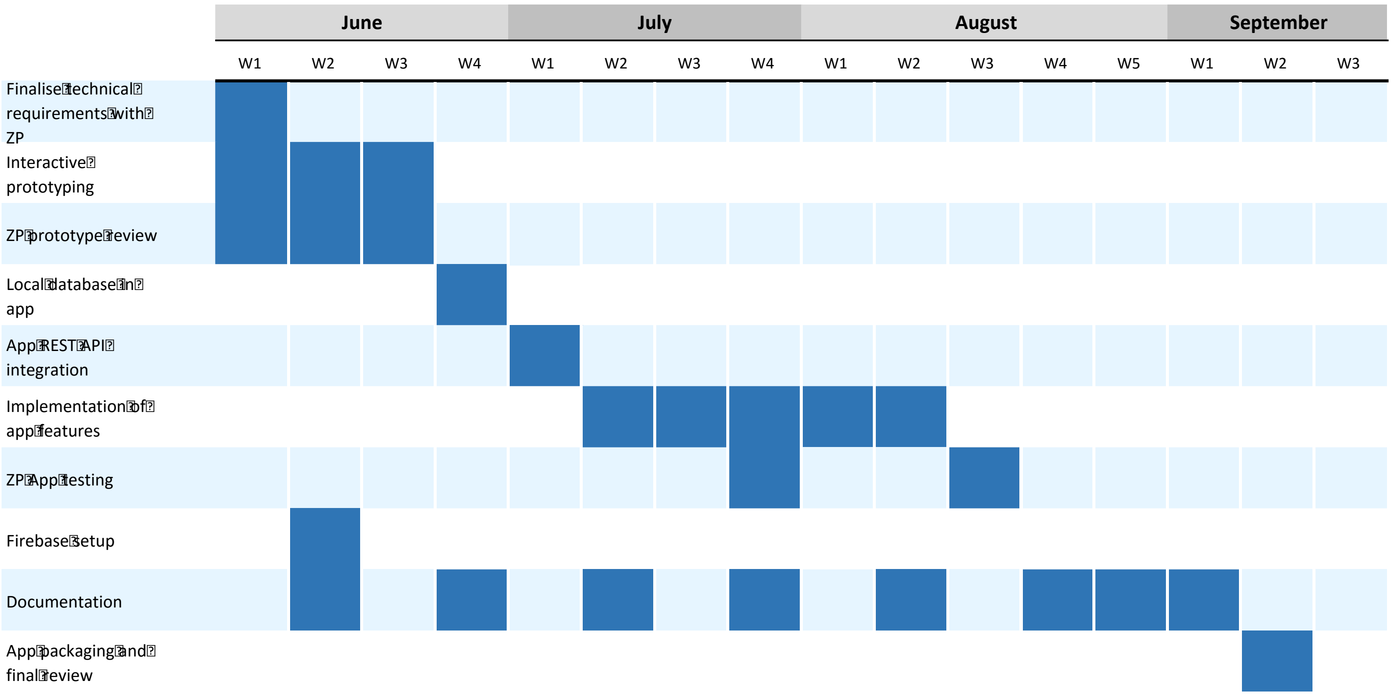

Figure 4 – Work Schedule

BIBLIOGRAPHY

1 London School of Hygiene & Tropical Medicine (2016) '"Zika virus surveillance in human and mosquito populations in Cape Verde"'. , p. 9777170. [online] Available from: http://www.bloomsbury.ac.uk/studentships/studentships-2017/zika-virus-surveillance-in-human-and-mosquito-populations-in-cape-verde (Accessed 14 December 2016)

2 ScienceDaily (n.d.) 'Zika Virus Research News -- ScienceDaily'. [online] Available from: https://www.sciencedaily.com/news/plants_animals/zika_virus_research/ (Accessed 31 March 2017)

3 Hennessey, Morgan, Fischer, Marc and Staples, J. Erin (2016) 'Zika Virus Spreads to New Areas — Region of the Americas, May 2015–January 2016'. *MMWR. Morbidity and Mortality Weekly Report*, 65(3), pp. 55–58.

4 Faria, N. R., Azevedo, R. d. S. d. S., Kraemer, M. U. G., Souza, R., et al. (2016) 'Zika virus in the Americas: Early epidemiological and genetic findings'. *Science*, 352(6283), pp. 345–349.

5 Barreiro, Pablo (2007) 'Evolving RNA Virus Pandemics: HIV, HCV, Ebola, Dengue, Chikunguya, and now Zika!' *AIDS reviews*, 18(1), pp. 54–5.

6 Barzon, Luisa, Trevisan, Marta, Sinigaglia, Alessandro, Lavezzo, Enrico and Pal??, Giorgio (2016) 'Zika virus: From pathogenesis to disease control'. *FEMS Microbiology Letters*, 363(18), pp. 1–17.

7 Kraemer, Moritz U G, Sinka, Marianne E., Duda, Kirsten A., Mylne, Adrian Q N, et al. (2015) 'The global distribution of the arbovirus vectors Aedes aegypti and Ae. Albopictus'. *eLife*, 4(JUNE2015), pp. 1–18.

8 Pybus, Oliver G., Tatem, Andrew J. and Lemey, Philippe (2015) 'Virus evolution and transmission in an ever more connected world'. *Proceedings of the Royal Society B: Biological Sciences*, 282(1821), p. 20142878.

9 Ioos, S., Mallet, H.-P., Leparc Goffart, I., Gauthier, V., et al. (2014) 'Current Zika virus epidemiology and recent epidemics'. *Médecine et Maladies Infectieuses*, 44(7), pp. 302–307.

10 Faye, Oumar, Faye, Ousmane, Diallo, Diawo, Diallo, Mawlouth, et al. (2013) 'Quantitative real-time PCR detection of Zika virus and evaluation with field-caught Mosquitoes'. *Virology Journal*, 10(1), p. 311.

11 PHG Foundation (n.d.) 'PHG Foundation | About us'. [online] Available from: http://www.phgfoundation.org/about (Accessed 28 March 2017)

12 Luheshi, Leila, Raza, Sobia, Moorthie, Sowmiya, Hall, Alison, et al. (2015) *Pathogen Genomics Into Practice*,

13 Internet World Stats (n.d.) 'World Internet Users Statistics and 2017 World Population Stats'. [online] Available from: http://www.internetworldstats.com/stats.htm (Accessed 31 March 2017)

14 Deng, Yong-Qiang, Zhang, Na-Na, Li, Chun-Feng, Tian, Min, et al. (2016) 'Adenosine Analog NITD008 Is a Potent Inhibitor of Zika Virus'. *Open Forum Infectious Diseases*, 3(4), p. ofw175.

15 Oxford Nanopore Technologies (n.d.) 'MinION'. [online] Available from: https://nanoporetech.com/products/minion (Accessed 11 February 2017)

16 Ericsson (2016) 'ON THE PULSE OF THE NETWORKED SOCIETY Ericsson Mobility Report'.

17 Apple Inc. (n.d.) 'Apple - Press Info - Apple Introduces ResearchKit, Giving Medical Researchers the Tools to Revolutionize Medical Studies'. [online] Available from: http://www.apple.com/pr/library/2015/03/09Apple-Introduces-ResearchKit-Giving-Medical-Researchers-the-Tools-to-Revolutionize-Medical-Studies.html (Accessed 29 March 2017)

18 Tourraine, Vincent (n.d.) 'shazino - List of all ResearchKit apps'. [online] Available from: http://blog.shazino.com/articles/science/researchkit-list-apps/ (Accessed 29 March 2017)

19 Cornell Tech (n.d.) 'ResearchStack'. [online] Available from: http://researchstack.org/ (Accessed 29 March 2017)

20 Patel, Chintan (n.d.) 'ResearchDroid – An Android ResearchKit Library to Create Medical Research Apps | Applied Informatics'. [online] Available from: http://blog.appliedinformaticsinc.com/android-researchkit/ (Accessed 29 March 2017)

21 EDINA (n.d.) ‘Home | Fieldtrip’. [online] Available from: http://fieldtrip.edina.ac.uk/ (Accessed 29 March 2017)

22 EDINA (n.d.) ‘home | Edina’. [online] Available from: http://edina.ac.uk/ (Accessed 29 March 2017)

23 EDINA (n.d.) ‘edina/fieldtrip-open’. [online] Available from: https://github.com/edina/fieldtrip-open (Accessed 29 March 2017)

24 The Apache Software Foundation (n.d.) ‘Apache Cordova’. [online] Available from: https://cordova.apache.org/ (Accessed 29 March 2017)

25 Lenarduzzi, Valentina and Taibi, Davide (2016) ‘MVP Explained: A Systematic Mapping Study on the Definitions of Minimal Viable Product’, in *2016 42th Euromicro Conference on Software Engineering and Advanced Applications*, IEEE, pp. 112–119.

26 LSHTM (n.d.) ‘Research Data Management Policy | London School of Hygiene & Tropical Medicine | LSHTM.’ [online] Available from: http://www.lshtm.ac.uk/research/researchdataman/rdm_policy.html (Accessed 3 April 2017)

27 LSHTM (n.d.) ‘Information Management and Security Policy | London School of Hygiene & Tropical Medicine | LSHTM.’ [online] Available from: http://www.lshtm.ac.uk/its/informationsecurity/policy/ (Accessed 3 April 2017)

28 IDC (n.d.) ‘IDC: Smartphone OS Market Share 2016, 2015’. [online] Available from: http://www.idc.com/promo/smartphone-market-share/os (Accessed 30 March 2017)

29 Apple Inc. (n.d.) ‘NSBackgroundActivityScheduler - Foundation | Apple Developer Documentation’. [online] Available from: https://developer.apple.com/reference/foundation/nsbackgroundactivityscheduler (Accessed 30 March 2017)

30 Android.com (n.d.) ‘Services | Android Developers’. [online] Available from: https://developer.android.com/guide/components/services.html (Accessed 30 March 2017)

31 Statista (n.d.) ‘The Smartphone Price Gap | Statista’. [online] Available from: https://www.statista.com/chart/4954/smartphone-average-selling-prices/ (Accessed 1 April 2017)

32 Adobe Systems (n.d.) ‘PhoneGap’. [online] Available from: http://phonegap.com/ (Accessed 2 April 2017)

33 Xamarin (n.d.) ‘Mobile App Development & App Creation Software - Xamarin’. [online] Available from: https://www.xamarin.com/ (Accessed 31 March 2017)

34 Facebook (n.d.) ‘React Native | A framework for building native apps using React’. [online] Available from: https://facebook.github.io/react-native/ (Accessed 31 March 2017)

35 NativeScript (n.d.) ‘NativeScript’. [online] Available from: https://www.nativescript.org/ (Accessed 31 March 2017)

36 École Polytechnique Fédérale de Lausanne (n.d.) ‘The Scala Programming Language’. [online] Available from: http://scala-lang.org/ (Accessed 30 March 2017)

37 JetBrains s.r.o. (n.d.) ‘Kotlin Programming Language’. [online] Available from: http://kotlinlang.org/ (Accessed 30 March 2017)

38 JetBrains s.r.o. (n.d.) ‘JetBrains: Development Tools for Professionals and Teams’. [online] Available from: https://www.jetbrains.com/ (Accessed 30 March 2017)

39 Anon (n.d.) ‘Boilerplate code - Wikipedia’. [online] Available from: https://en.wikipedia.org/wiki/Boilerplate_code (Accessed 30 March 2017)

40 Fielding, Roy Thomas (2000) ‘UNIVERSITY OF CALIFORNIA, IRVINE Architectural Styles and the Design of Network-based Software Architectures’.

41 Pingdom (n.d.) ‘REST in peace, SOAP - Pingdom Royal’. [online] Available from: http://royal.pingdom.com/2010/10/15/rest-in-peace-soap/ (Accessed 31 March 2017)

42 Firebase (n.d.) ‘Firebase | App success made simple’. [online] Available from: https://firebase.google.com/ (Accessed 6 April 2017)

43 Firebase (n.d.) ‘Pricing | Firebase’. [online] Available from: https://firebase.google.com/pricing/ (Accessed 6 April 2017)

44 LSHTM (2015) ‘LSHTM Information Management and Security Policy Supporting policy 9: Data Classification and Handling Policy’.

45 LSHTM (n.d.) ‘LSHTM Research Data Access Policy [Draft – for comment]’.

46 Zetetic (n.d.) ‘SQLCipher - Zetetic’. [online] Available from: https://www.zetetic.net/sqlcipher/ (Accessed 6 April 2017)
47 Zetetic (n.d.) ‘Export Requirements for Applications Using SQLCipher - SQLCipher / SQLCipher FAQ - Zetetic Community Discussion’. [online] Available from: https://discuss.zetetic.net/t/export-requirements-for-applications-using-sqlcipher/47 (Accessed 6 April 2017)
48 InVision (n.d.) ‘Free Web & Mobile Prototyping (Web, iOS, Android) and UI Mockup Tool | InVision’. [online] Available from: https://www.invisionapp.com/ (Accessed 24 March 2017)
49 Marvel (n.d.) ‘Turn sketches, mockups and designs into web, iPhone, iOS, Android and Apple Watch app prototypes.’ [online] Available from: https://marvelapp.com (Accessed 24 March 2017)
50 MockingBot (n.d.) ‘MockingBot - Mobile App Prototyping Tool Without Headache’. [online] Available from: https://mockingbot.com/ (Accessed 24 March 2017)
51 Protoio (n.d.) ‘Proto.io - Prototypes that feel real’. [online] Available from: https://proto.io/ (Accessed 24 March 2017)
52 Marvel (n.d.) ‘POP - Prototyping on Paper | Mobile App Prototyping Made Easy’. [online] Available from: https://marvelapp.com/pop/ (Accessed 6 April 2017)
53 North, Dan (2006) ‘Introducing BDD’. *Better Software Magazine*.
54 Slack (n.d.) ‘Slack: Where work happens’. [online] Available from: https://slack.com/ (Accessed 6 April 2017)
55 Trello (n.d.) ‘Boards | Trello’. [online] Available from: https://trello.com/ (Accessed 6 April 2017)
56 KarthiK (n.d.) ‘Android, smart phone icon | Icon search engine’. [online] Available from: https://www.iconfinder.com/icons/211118/android_smart_phone_icon (Accessed 12 April 2017)