**Black Order - Project Form 2**

# Summary - Patrick

We are proposing to create an app to improve the use of public transportation in

Darwin. Users purchase their bus ticket through an app on their phones, as well as

allowing them to view bus timetables and maps for their planned route. They can scan

their phones when getting on the bus to pay for their trip, similar to the existing

card-based system. The app will be able to see when the bus will arrive as well as being

able to share routes with friends

**Table of contents**

[**Summary - Patrick**](#_ktopm19hu9bv) **0**

[**Introduction**](#_wn7gje30hxr9) **1**

[Background - Patrick](#_h8iprfs0xgs7) 1

[Market Analysis – needs and existing applications in the market - Aasish and Patrick](#_ef23qx9qmuj0) 1

[App 1: Moovit: Bus Time and Train Time Live Info](#_1e6w78mjosdv) 1

[App 2: Trip View: Bus and Train timetable for NSW](#_9c732oti2eem) 2

[Purpose - Nurul](#_q17pmifzpmjn) 2

[**Project**](#_c420aedrelkq) **2**

[Software description - Nurul](#_hivtnm5gwy4u) 2

[Requirements](#_8o74rfbbli4k) 3

[Functional Requirements - Patrick](#_2ae0mam9uk23) 3

[Non-functional Requirements - Patrick](#_ecw87uz82t1c) 3

[Deliverables](#_i3t9u38mnpu8) 3

[Roles and responsibilities - Joel](#_oqt0b8m4a0ke) 3

[Schedule - Joel, Patrick, Nurul](#_lnb7wh4y0ewl) 4

[Resources - Joel](#_yta47mqkwej9) 4

[Risk Management - Nurul and Patrick](#_gboynrfqp800) 6

[**Github**](#_wggac2ju03ca) **7**

[**References**](#_3pryjs8wgnk5) **7**

# Introduction

## Background - Patrick

Public transportation is widely used in Darwin, as catching the bus is a cheap and efficient way of getting around. It is important that the process of purchasing bus tickets is made as simple as possible to ensure that users save time and avoid hassle when catching the bus. As the majority of the public have a mobile phone, we can take advantage of this to solve the needs surrounding public transport.

## Market Analysis – needs and existing applications in the market - Aasish and Patrick

We conducted research into similar apps in the market to understand what services were already being provided, and what new features we needed to provide. We found two apps that were similar to ours and discussed some similarities and differences.

|  |  |  |  |
| --- | --- | --- | --- |
| **Criteria** | **Moovit** | **NSW Bus/Train Timetable** | **Bus Ticket system (app to be created)** |
| Provides bus schedule | **╳** | **✔** | **✔** |
| View trip history | **✔** | **✔** | **✔** |
| Sharing planned trip | **╳** | **╳** | **✔** |
| Search option by routes, buses, suburbs or stops | **✔** |  | **✔** |
| Online payment | **╳** | **╳** | **✔** |
| Passenger seat availability | **╳** | **✔** | **╳** |
| GPS Location | **✔** | **╳** | **╳** |

Based on our research, the main functionalities that our app provides that other apps don’t include an online payment system, the option to share routes, and a higher focus on bus travel. This will allow us to focus more on the needs of our users, as bus travel is widely used in Darwin.

## Purpose - Nurul

The purpose of app based bus ticketing system to provide a convenient way to purchase bus ticket on online as well as keep track on bus schedule and timing. This will achieve our goal of providing users with a way to purchase bus tickets and browse schedules in a single app.

# Project

## Software description - Nurul

App based bus ticketing system will be an online bus ticket booking application, which allows purchase bus ticket and track timing and schedule in a few easy steps. The responsive designs of this app make sure that it looks great on mobile. It is a powerful bus ticket booking clone for bus owners, bus operators or bus agents to get better prosperity for their bus fleet business. This Bus ticket booking mobile application will be customizable or flexible to add new features.

## Requirements

### Functional Requirements - Patrick

* User can view bus timetables
* User can access bus maps
* User can view routes of each bus
* User can enter current location
* The app will provide an estimated time of arrival for the bus
* User can pay for the ticket within the app
* The app is able to be scanned by bus drivers to verify tickets
* User can access a history of previous bus trips
* User can save trips for later viewing
* User can share trip routes with others

### Non-functional Requirements - Patrick

* The application should be fast and responsive
* The application should be available as long as there is an internet connection
* The application should be simple and easy to use
* The application should not be intrusive in data collection
* The application should use encryption to secure user data

## Deliverables

* Main application - bus ticket app
* Final Report
* Presentation

## 

## Roles and responsibilities - Joel

Aasish - lead developer, backend

Joel - team leader, backend, lead software tester

Patrick - frontend, scrum master

Nurul - frontend, documentation

## Schedule - Joel, Patrick, Nurul

|  |  |  |
| --- | --- | --- |
| **Week** | **Task description** | **Deliverable** |
| **5** | setup Github, Google Drive | · complete Project Form 2  · complete Project plan |
| **6** | Paper/digital prototype of app, start writing tests for TDD | · complete first stage paper/Digital prototype  · complete project form 3 |
| **7** | Start development process (interface creation, backend development) | · complete finalized digital prototype  · Database, server, setup  · Version1 front end design |
| **8** | Continue test case development | · Backend google map, routes and location setup |
| **9** | Frontend prototype complete, continue backend development, Implementing test cases and software development | · V2 Frontend design complete  · Stage 1- Implementation of functionality (time, pay, scan and history)- stage 1 |
| **10** | Start documentation, Implementing test cases and software development | · Stage 2 -Implementation of functionality (time, pay, scan and history)    · Refactoring and enhance performance |
| **11** | app and documentation, submission | · Stage 3 -complete functionality (time, pay, scan and history)  · Documentation completion |
| **12** | Preparation of presentation | ·Project submission and final presentation |

## Resources - Joel

**Development environment**

**Cloud base bus ticket system- spine-tail server connection**

**File encodings**

1. IDE encoding:UTF-8

2. Project encoding:UTF-8

3. Properties files encoding:UTF-8

**Naming convention**

**1. Follow basic android naming convention(link:**

[**https://android.jlelse.eu/java-coding-standards-ee1687a82ec2**](https://android.jlelse.eu/java-coding-standards-ee1687a82ec2) **)**

**2. Class Naming**

Activity class uses Activity as a suffix, such as: LoginActivity

Fragment class uses Fragment as a suffix, such as: ShareDialogFragment

Service class uses with Service as a suffix, such as: DownloadService

Adapter class uses Adapter as a suffix, such as: CouponListAdapter

Tool class uses Util as a suffix, such as: EncryptUtil

Model class uses BO as a suffix, such as: CouponBO

Interface implementation class uses Impl as a suffix, such as: ApiImpl

**3. Method Naming**

Initialization method, prefix:init, example: initView

button click method, prefix:to, example: toLogin

setting method, prefix:set, example: setData

get method, prefix:get, example: getData

asynchronously loading method, prefix:load, example: loadData

judgement method , prefix: is or has, or a word with logical meaning such as equals, example: isEmpty

**TOOLS and FRAMEWORK (version of framework will be specific by Maven or Gradle)**

**1.**

JDK: jdk Version: 1.8.0\_131

IDE: android studio Version: 3.0.1

**2.**

Continuous integration: Jenkins(we may need a server…)

Version control: github

Dependency injection: butter knife

Security API: OWASP Enterprise Security API

Build management: Gradle (need learn Grovvy ) , or Maven

Database ORM: ORMLite

Json: Gson, GsonFormat

Logging: Timber

UI test: UiAutomator

Unit test: junit、 mockito、 robolectric

## Risk Management - Nurul and Patrick

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Risks | Description | Impact | Possibility | Solution |
| Security | User data might be leaked. | HIGH | HIGH | Use encryption to secure user data |
| User | The user interface may not be user friendly, which impacts the amount of users who would use the app | HIGH | MEDIUM | Do UX research, to ensure the application is accessible to all our target user. |
| Functionality | Network infrastructure may not be able to support many concurrent users, which may negatively impact the performance of the app | HIGH | LOW | Use efficient algorithm. |
| No communication within the group | Group members fail to report their progress | MEDIUM | MEDIUM | Organise regular meetings in person to ensure everyone is on track |
| Group members not knowing how to use certain resources | Group members may not understand programming languages, libraries, etc | MEDIUM | HIGH | Get group members knowledgeable in the resource to teach the rest of the group, as well as find solutions together |
| Finish the project on time | There might be a delay to finish the project on time. | HIGH | LOW | Follow Schedule strictly. |

# Github

https://github.com/PRT452Black-order/Bus-Ticket-System

# References

Moovit: Bus Time & Train Time Live Info: <https://play.google.com/store/apps/details?id=com.tranzmate>

Bus and Train timetable for NSW:

<https://transportnsw.info/apps/tripview>

### 