**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.

## 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.

### App 1: Moovit: Bus Time and Train Time Live Info

|  |  |
| --- | --- |
| **Similarities** | **Differences** |
| Provides bus schedules | No in app payment system |
| Provides maps and routes | No option to share routes |
| History of recent trips | Uses GPS to locate buses |
| Provides estimated times of arrival | Has options for trains, bikes, taxi, etc |

### 

### App 2: Trip View: Bus and Train timetable for NSW

|  |  |
| --- | --- |
| **Similarities** | **Differences** |
| Have live tracking and delay notification service | No online payment |
| Possible to view trip history | Shows actual journey cost |
| Sharing planned trip with friends and known people | Full version being paid only |
| Maps and journey path can be viewed | Doesn’t support voiceover on IOS devices |
| Search option by routes, buses, suburbs or stops | Our app won’t provide passenger seat availability on viewed trip. |

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

· Enter location, time, route

· Pay for the ticket in the app

· Path of the bus for route

· How long the bus will take, when it will arrive

· Share route with friends

· History of trips

· Save trips

· Scanner for ticket

### 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
* User details should be kept as secure as possible

## 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** | **Deliverable** |
| 5 | Complete Project Form 2, setup Github, Google Drive |
| 6 | Paper/digital prototype of app, start writing tests for TDD |
| 7 | Start development process(interface creation, backend development) |
| 8 | Continue development and tests |
| 9 | Frontend prototype complete, continue backend development |
| 10 | Start documentation, complete prototype |
| 11 | Finished app and documentation, submission |
| 12 | Complete 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

Risk management helps the software developer team to manage uncertainty and understand the risks or problems can affect a software development project. Generally, risks are considered as probability that might occur in some circumstances.

**App based bus ticketing system projects risk can be categorized as:**

**· Financial Risk**

**· Security Risk**

**· Technology Risk**

**· User Risk**

**Risk Assessment:**

Risk assessment is considered as a first step in procedure of risk management. It is an important step to protect and understand risks of app based ticketing system project and helps to focus on those risks that really important. It also helps to understand project team capacity to tackle and identify the future risks and provide input towards developing adoption plan.

**Risk Control:**

Second step of risk management process is risk control, which is a technique to utilize the findings from risk assessment and reduces risk by implementing changes. These changes are implemented during the development of the project as risks are found.

**Risk Identification:**

This step identifies the specific risks that may occur during the project. Financial risk is a type of risk that may occur when the team runs over budget during development. Securing user data is important to minimise security risks. Network infrastructure may not be able to support many concurrent users, which may negatively impact the performance of the app. The user interface may not be user friendly, which impacts the amount of users who would use the app. These risks need to be considered during development in order to create risk management strategies to minimise risk.

**Risk Prioritisation:**

After identifying the relevant risks, they are given priorities based on their overall impact on the project. The highest priority risk in this project is the financial risk, as if the team runs over budget, they will not be able to continue work on the app. The next prioritised risk is the security risk, as users rely on the app to secure their details. Technology risk is next prioritised, as the performance of the app directly affects usability. User risk is the least prioritised as the functionality of the app is more important than the accessibility, however it is still an important aspect to consider.

# 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>

### 