

# **Software Quality Engineering (SE-471)**

## **Semester Project**

### **Bus Tracking System**

#### **Group Members**

Muhammad Shahrar Malik (3799-BSSE/F18, [muhammad.bsse3799@iiu.edu.pk](mailto:muhammad.bsse3799@iiu.edu.pk))  
Atif Maqbool (3749-BSSE/F18, [atif.bsse3749@iiu.edu.pk](mailto:atif.bsse3749@iiu.edu.pk))  
Mahd-ul-Haq (3872-BSSE/F18, [mahd.bsse3872@iiu.edu.pk](mailto:mahd.bsse3872@iiu.edu.pk))  
Muhammad Zeeshan Yousaf (3910-BSSE/F18, [muhammad.bsse3910@iiu.edu.pk](mailto:muhammad.bsse3910@iiu.edu.pk))

# Table of Content

## Section 1

### 1. Brief Background

1.1.	Introduction.....	1
1.2.	Problem Statement.....	2
1.3.	Objective.....	3
1.4.	Project Scope.....	4

## Section 2

2.	System Architecture.....	6
2.1.	Diagram.....	6
2.2.	Description.....	6
3.	Requirements.....	7
3.1.	Tactics / Quality Attributes	
3.1.1.	Availability.....	8
3.1.2.	Performance.....	10
3.1.3.	Modifiability and Resuability.....	11
3.1.4.	Portability.....	12
3.1.5.	Security.....	13

# **1. Brief Background**

## **1.1. Introduction**

Mostly people in our country as well as in other countries prefer bus service instead of their own mean of transport. Because it is considered cheap and reliable mean of transportation. Public prefer it because of various reasons such as traffic jam, parking fee, safety of own vehicle, and avoidance of traffic challan. Alongside the quality of this service is very poor. Public do not know exact arrival time and also do not know about current position of bus. This problems is due to the reason that bus service did not adopt tracking technology. If they use tracking technology, this problem can be overcome and the quality of service will be increase.

For this reason, Auckland Regional Council(ACR) wants such system that tracks buses. It wants to add GPS to all of its buses so that it can track where they are to within 100 metres. They will use this information to provide estimated arrival times of buses at each major bus stop.

## **1.2. Problem Statement**

In the current situation people prefer their own transportation instead of buses, the reason is that if they use bus service, it will consume more time if they wait for a bus. In morning time, Student and other professionals needs to wait for a bus without knowing what time the bus will arrive actually. Sometime, they might feel anxious and impatient when they wait for a bus in case if they did not know what exactly the bus arrival time is. This situation waste lot of time, and the time wasted here can be spend on other purpose. Examples to describe this issue, a student have to take the 7.30am bus to attend an 8am morning class and the class is important and cannot be late. The scheduled time for the bus is 7.30am, but the arrival time is not the same as scheduled time because of the traffic jam in some area. If the bus arrives at exactly 7.30am, the student will not late for the class. If the bus late 5 to 10 minutes, the student may miss or may be late for the class. Thus, the student unable to make decision whether to wait for the 7.30am bus or walks in to campus before 7.30am because if they wait for bus they will miss their class.

This will also cause dissatisfaction of public to bus management team, and because the management team are not aware of that situation. Also they are unable to update latest bus information to users.

### 1.3. Objective

The objective of this system is to provide good quality service and easiness to user/public. ACR do not wants to waste user's time and their reputation. So they wants bus tracking system.

Few points of their proposed system are

- Wants to track all buses
- Track bus with in 100 meters
- Global Positioning System (GPS) for buses.
- Components on bus includes
  - GPS receiver
  - Radio transmitter
  - Hardware
  - Software
- GPS receiver determines position within 10 meter at each second
- Planned display that contains
  - Radio receiver
  - Display 4 or 5 buses and times
  - Determines buses whose estimated arrival time is in the next hour.

Above mentioned few hardware are required for this project.

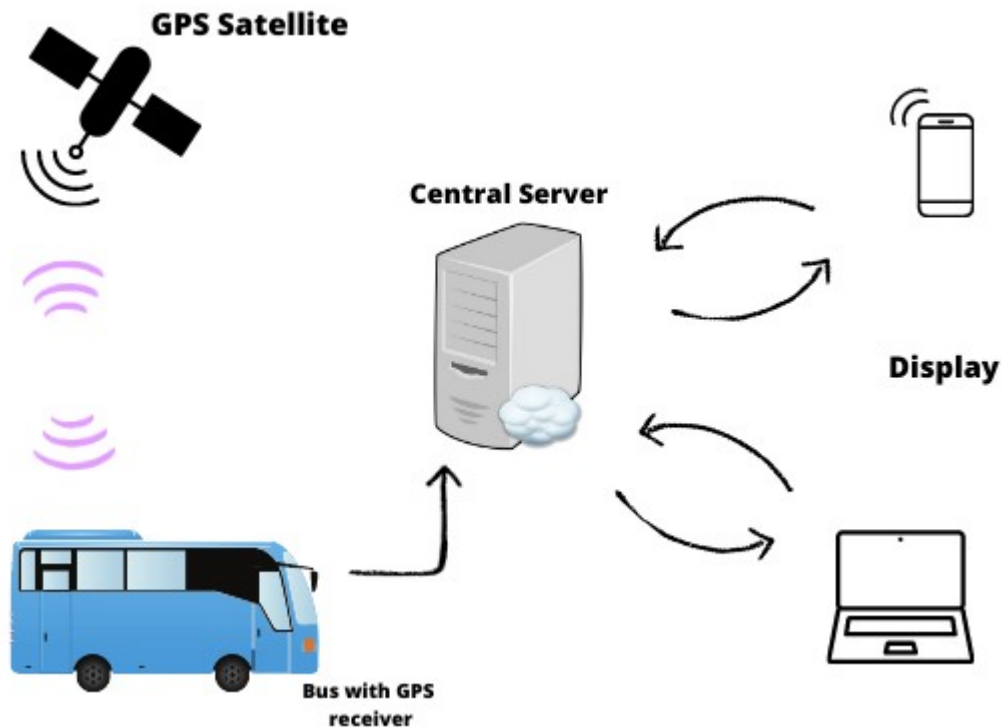
## 1.4. Project Scope

- This system is especially designed for Auckland Regional Council(ACR). ACR wants a system which track their buses.
- This purpose is achieved by using GPS along with some other necessary hardwares and softwares.
- Along with this they also wants website and mobile application service for their users to check information about buses.
- During designing of system we will make its architecture in such a way that it is portable and can adopt changes.
- Hence by doing some modification we can use this for others Regional Council as well.

## **Section 2**

## 2. Architectural Diagram of Bus Tracking System

### 2.1. Diagram



### 2.2. Description of system architecture

- Single radio receiver for a signal city
- Bus hardware transmits
  - Unique bus ID
  - Time stamp
  - Location (every second)
- Time stamp and location comes from GPS receiver after request
- Display has unique ID number. They receive information from central system and not directly from bus.
- There is no direct communication between display and a bus. They communicate through central server.



- Also display and central do not communicates. Display just receive information from central system and display it.

### 3. Requirements

#### Functional

Some functional requirements of system are:

- Verify information source
- Show current time
- List the bus details
- Clarify to users that when to rely on information and when not to.
- Estimated arrival time of bus at stop
- List buses that stops at specified stop

#### Non Functional

This is related to quality of a system. Some quality attributes that we will discuss in detail in tactics are:

- Availability
- Performance
- Modifiability
- Security
- Reusability
- Portability

*In this project only Performance, Modifiability, Resuaility and Portability quality attribute are required but we also implement Availability, Security attributes along with required on bus tracking system because it is an also important quality attributes*

## **3.1. Tactics**

### **3.1.1. Availability**

In this quality attributes we have to make sure that the system must always be available to the user. If system is not available to the user its means that fault occurs. For this we have to consider different scenario like how we can detect fault and if fault occur then how we can recover it and what are the necessary prevention steps to avoid fault becoming failure.

### **Availability tactic for bus tracking system**

#### **Fault detection**

- As from the requirements, the GPS receiver which is present in the bus can determine its position within 10 meters at every second. So we implement such system in hardware of bus that can give signals to GPS in each second (like heartbeat) which means that the system is available and no failure occur.
- We can also detect fault by implementing architecture in a way that bus continuously send availability signals to display through central server and vice versa (ping/echo) in specific time period. If the respond is in particular time period its means no fault occur.

## **Fault prevention**

- For determining position of buses, at least 4 GPS are required. But we will use more than 4. In this way if failure occurs in one GPS we can disconnect it from the system and use other GPS.
- We can determine failure using an internal process monitor which we can implement in a central server

## **Fault recovery**

If a fault occurs we can recover it by using many techniques like

- We can switch server to other. This takes very little time and it has no effect on the performance of the whole system.
- We can maintain logs of the system. This will help us to recover from a fault and set the system state to the previous one.

By implementing the above tactics we can ensure that the system is available to the user and if a fault occurs it will mask it or at least reduce it. As a fault occurs when the system does an abnormal exit so by implementing these tactics we can reduce the probability of a fault becoming a failure.

### 3.1.2. Performance

This quality attribute focus on how quickly the system responds back to a particular event. In this case study we have to make such system whose responds time is minimum and display frequently shows the next buses.

- From requirements, if the bus is in 1 km, the estimated arrival time of bus at each bus stop must be within 2 minutes of actual arrival time. Else system should display “best effort”. This can be achieved by calculating length between display and the GPS receiver. As GPS receiver which is in the bus and display cannot communicate directly, they communicate through central server so the GPS receiver continuously sends signal to central server and the central server forward it to display. If the distance is within 1km the display show that bus arrives within 2 mintue. If this distance is greater then 1km, then display shows “best effort”. This whole process just take few seconds. By implementing this tactic we can achieve performance.
- If the bus arrive late due to traffic jam, then the display show it that the expected arrival time is not clear and don’t rely on that time.
- As mobile and website are also required to bus company for there users, so we also need to design architecture in such a way that effectively utilize resources (mobile memory). This will also increase performance of system.
- We can also improve performance by implementing efficient algorithm in which latency rate is minimum.

### 3.1.3. Modifiability and Reusability

Modifiability is an important design quality attribute. This quality attribute focus on how easily a system can adopt changes. As the requirements kept changing, so system must be capable to easily adopt changes without effecting other modules of a system

In our project i.e Bus Tracking System, bus company also would like to allow bus users to get arrival time of all buses at all times through web site or mobile application. So the website and mobile application must be modifiable because it may be possible that in future the requirements will change and not only bus arrival time but device will display **bus number** along with other details like **driver name, bus condition** etc. So our mobile application or website must able to adopt these changes easily without effecting other module.

For doing this we use different modifiability tactics which will allows us to achieve modifiable system.

- While designing architecture of website and mobile application, we will make independent module which will not depends on module of other system like central server etc. But if one module need to collaborate with other we can, we can some **design patterns** to achieve this goal. By doing this we will maintain semantic coherence and reduce coupling.
- We can anticipate expected changes and create seperate module for in which future changes is possible
- Preventing from ripple effect is another tactic to achieve modifiability. We can achieve this by using intermediary. In mobile application or in website if two module has dependency, we can insert intermediary such as the facade, bridge, mediator, strategy, proxy, or factory design patterns that provides intermediaries that convert the syntax of service from one form into another.
- We can also allow end user to do change at run time, it will be cost effective compare to modification done to developer. For this we will provide user manual for guidance of the user.
- **Reusability** is achieved by using techniques such as modularity, high cohesion and loose coupling.
- While designing architecture of system we divide the software into mutiple independent modules. This will make code simple and easy to understand also maintainable. So that other can easily understand the architecture of system.
- If the coherence is high, this will also helps to increase reusability.
- Coupling between different modules must is loose, Like components in website and mobile application must be loose. If they are tightly coupled this will effect

in achieving reusability. So the components of system must be loosely coupled. This will also help us to achieve resuability.

### **3.1.4. Portability**

Portability is another important quality attribute. It is the ability of program or software to run on different platform like same program on different OS with little or no modification.

- If the architecture of system is properly designed and it is modifiable, then the portability is automatically achieved, because portability is just a specialized form of modifiability.
- The portability of a software system depends on:
  - Degree of hardware independence
  - Implementation language .
- In our Case, we will javascript, python because these languages have many built-in modules that will help us in improving the quality of our system and ultimately reduce the cost.

## **Hardware properties**

- In Our Bus Tracking System, we use those hardware which is commonly used in market and has a better performance.
- Our software supports hundreds of trackers currently used in the market.
- End user will use our website and mobile application via device which are commonly available in market.
- As , we know that there are various operating system in the market , so in order to make it compatible for all the devices , we will us React Native ( Framework for developing the android and ios platforms ).
- This reduces our maintenance cost and have many built in modules, that will help us in making bus tracking system portable.

### 3.1.5. Security

Security is an important quality attributes which are mandatory in a system now a days, because if we not focus on system security it will cause damage. So in our Bus Tracking System it is necessary for us to design an architectural tactic for security.

We have to ensure the security of bus tracking system. The attacker may do GPS spoofing. This will cause incorrect information, timing information and incorrect navigation. If not tackle it, this will lead to serious damage. Therefore it is important to detect this types of attacks on bus tracking system.

Also while designing the architecture, we will ensure about data confidentiality because as the bus hardware transmits unique bus ID, timestamp and location, if unauthorized person will access this data, it will harmful for tracking system.

To prevent this from attackers, we will use hardware such as obscuring antennas or install decoy antennas to throw off attackers and add sensors that can detect spoofing signals and send alert to monitoring team, also install antennas in different locations, which catches the anonymous signals which are from attackers tell monitoring team about this situation.

Also to make system available and restricting of any DOS attack we will deploy anti DDOS hardware and software module.

Also to do some prevention steps, we will use web application application firewall(WAF) this will monitor, filter and block suspicious data packets to central server.

WAFs also provide efficient protection from a number of malicious security attacks such as:

- SQL injection
- Cross-Site scripting
- Session Hijacking
- Distributed Denial of Service (DDOS)
- Cookies Poisoning
- Parameter Tempering

So by applying above tactics we can ensure that our system is protected (not fully but at maximum level) from attackers by providing service to authorized users.