List of Figures

CONTENTS

Abstract

Transportation plays the major role in the part of humans. Nowadays, it is very difficult to manage the time and distance without vehicles. The population is increasing rapidly which leads to increase in vehicles but as the number of vehicle are increasing it leads to increase in pollution. Government are encouraging citizens to use less private vehicle and go for public transport. Public transportation (Bus Services) is the main source of travelling for inter and intra cities in India. In metropolitan cities like Bangalore, Mumbai, Pune and many others cities where people are travelling in buses daily. Today in the era of Digital India and Cashless Economy, public transport needs to adopt the technology, there is a need of smart and trusted system. The major problem experienced by the passengers are getting the perfect tender for the ticket fare and waiting time. This project proposes about the Mobile app for booking the journey ticket as well as the conductor will be able to scan the booked ticket from the system provided to the conductor by depot. This will lead to fast and paper less ticketing system.

Keywords: Transportation, Ticket, Government

1. Objective:

The main objective of the app is to provide a smart mode of payment for the passengers in the field of transportation. A passenger app to book the tickets by selecting the relevant source and destination with a payment option. After successful payment the ticket is generated in the form of a QR code. The QR code is then saved to the passenger device in case of network failures. The QR code is then scanned by the conductor who uses another app to verify the validity of the ticket. Every ticket id generated by the passenger app is automatically expired after a certain time making it time constraint.

2. Requirements:

Hardware requirements:
- Android Phone
- PC

Software Requirements:
- Android Studio
- Android OS 5.0 or higher
- Amazon AWS EC2
- AWS MySQL

Functional Requirements:
- Administrative functions :

The Depot Manager can use Android application or HTML Webpage to add or remove new user and if they found any complaint or discrepancy from passenger then they will use to Webpages and android application to analyze and solve the issue
- Authentication:

The user will able to login if registered and user need to provide Email Address and Password to login. Various users with different privileges able to login with their requirement of the use such as Passenger, Conductor, Depot manager.
- Authorization levels:

To store the QR code in device it requires additional permission from device owner. Conductor application requires additional permission to access Camera to scan QR Code
- External Interfaces

Passenger and Conductor Application have Mobile Android interfaces which helps to book the tickets and scan the QR code respectively
- Historical Data:

Webpage will help Depot Manager to trace and to check the conductor details and passenger details.

Non-Functional Requirements:

- Accessibility:

Passenger can access the mobile application anywhere and can book ticket. Passenger will also get the benefits of Cash-less payment

- Capacity:

Multiple users can connect at a time and book the tickets

- Efficiency:

AWS Database is utilized as it is powerful in the computation and highly optimized

- Quality:

No loss of data

- Reliability:

Passenger data is secured and all the data is stored with all the security

- Security:

Amazon Web Services provides various level of authentication which provides data security to passenger and it helps to safely manage the mobile application and networks

3. Stakeholders:

A stakeholder is a person or organization that has rights, shares, claims, or interests concerning the system or its properties meeting their needs and expectations. To put it more simply, the interests of stakeholders have some influence on the project, so their opinion should always be taken into account.

Passengers: Ease of access while booking ticket

Conductor: Authentication of Ticket to analyze the validity of ticket

Depot Manager: To manage employee and issues of passenger using Web Pages or Android Application

4. Scheduling

The Figure 1, gives a brief idea about the time taken for the different modules to be completed varying from finalizing the requirements to the final review scheduled on 8th November.
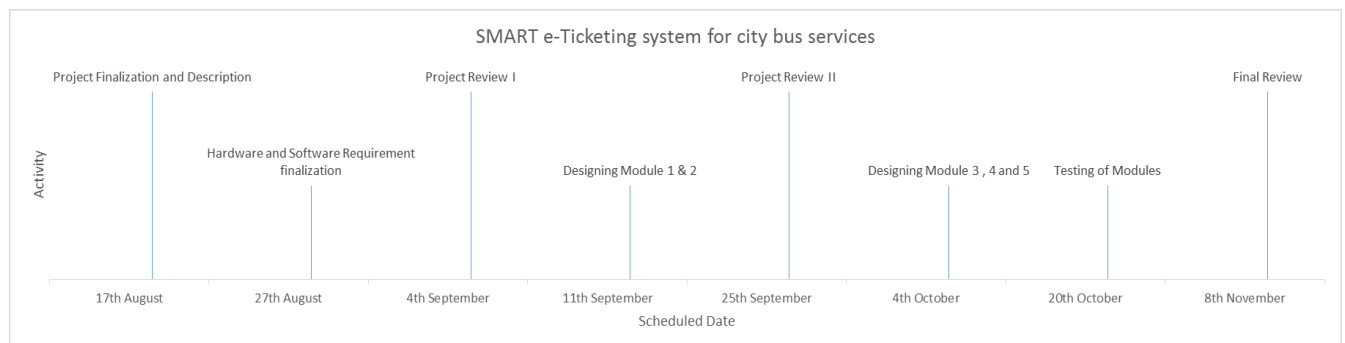

Figure 1: Scheduled Timeline
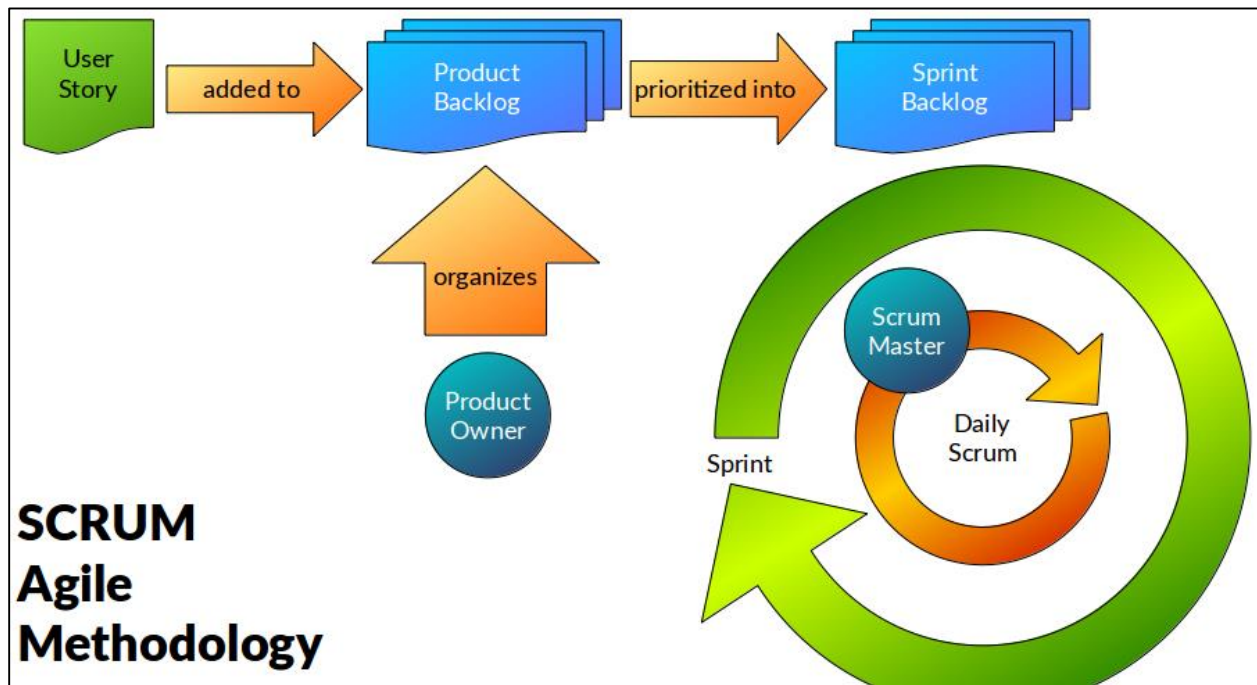
5. **Process Model:**

Figure 2 : Implementation of Process Model

The project takes less time to develop since it contains less modules. The modules are small in size and takes less amount of time with minimal number of developers. Agile is the best option in this situation since every error can be immediately verified and rectified in a short amount of time with minimal resource usage. Changing and adapting to different technologies is what agile does best and since the feasibility analysis returned in favor of MySQL than firebase the change in structure was simple and easy to adapt. Every week had scrum meetings on the progress of each individual's work in the project and the integration steps to be taken in the project development.

The specifications are not required at beginning and changing frequently. The bus ticket fare changes according to the area and sometimes the increment in fare which results in changing of fare in the application and database can be modified easily. The cost of project remains as estimated but the delivery of new components are based on the requirements and their priority. This model provides us very high success rate, as the modules as distributed in chunks and developed as per the requirement which helps while integrating it in whole application which further leads to detect the error in the system after completely loading the module in the application.

6. Module identification

The project is divided into three major parts. One is the android app for passenger where the passenger can book the e-ticket and the other android app for the conductor to verify the e-ticket generated by the user. The third part is the admin page for the depot to control the staff details and payment information. Based on the functionality, the project is divided into the following modules

1. Authentication and registration:

The authentication module consists of two parts where one is the registration and the other is the login part. Registration process for the Passenger app requests the passenger to enter the details such as the name, email ,address and contact no etc. along with password which is then stored in the database. After the registration the login activity is started and the user enters the login details to access the app home interface. It's the same for the conductor app and the admin page.
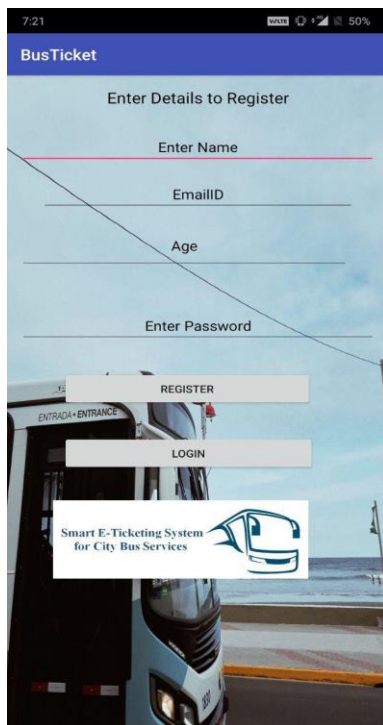
Figure 3: Login Application Page

Figure 4: Registration Application page

2. Journey ticket creation:

After the successful authentication, the app's homepage is the place where passenger can select the journey details and book the ticket. The passenger will be able to select the source stop, destination bus stop and the date of the journey which is mostly used with the system time.
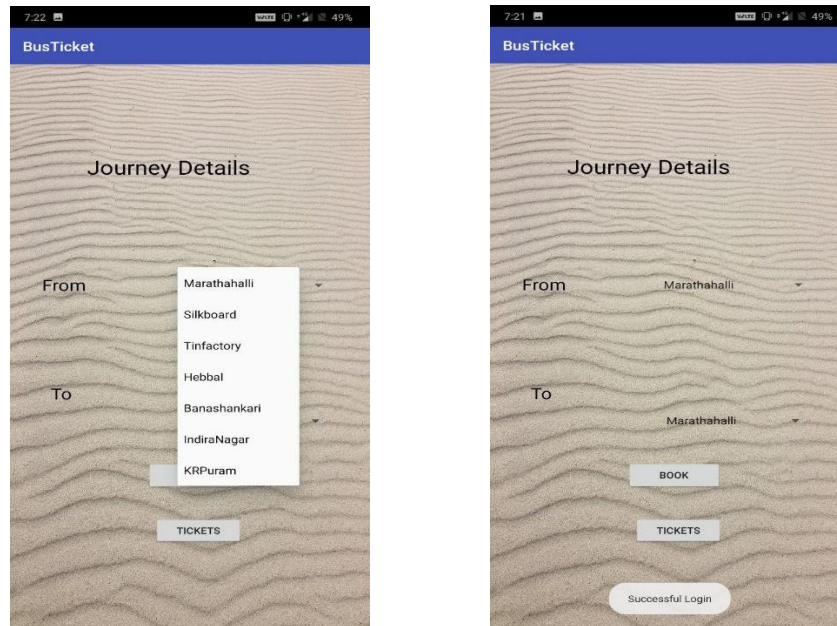

Figure 5: Journey Selection

3. QR Code:

With the successful payment the ticket id generated is converted into a QR Code for the conductor to verify the ticket using the conductor android app. The QR code generator uses the Zxing library from the android studio to both generate and scan the code from within the app.
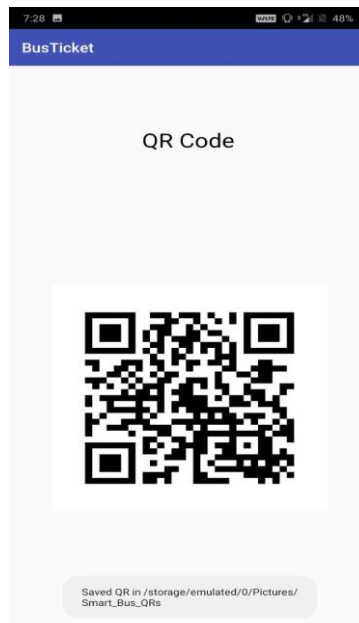

Figure 6: Booking Summary in QR Code

Smart E-Ticketing System for City Bus Services   5

4. Ticket summary:

The ticket booked is shown on the summary page for the conductor verification and on the conductor app this section is used to show whether the booking after the QR code scanning is valid or not.
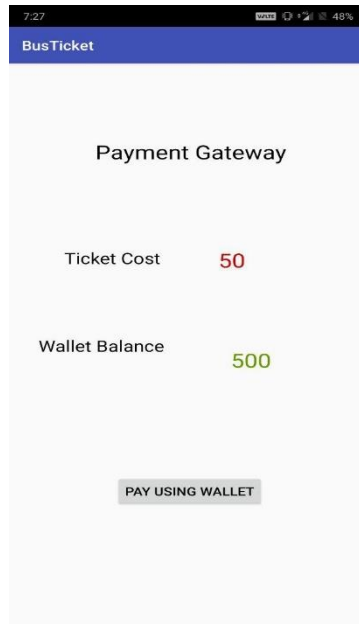

Figure 7: Payment Summary

5. Verification Method:

After booking the ticket, the Bus conductor could verify the ticket. The validity for a single ticket to be used is 10 minutes. After every 10 minutes tickets are invalid. The logo with Green and Red Mark helps conductor to understand the ticket with visual analysis.
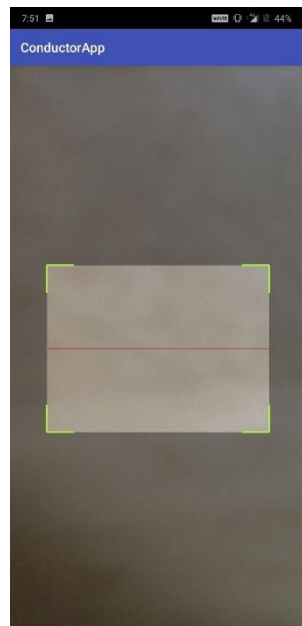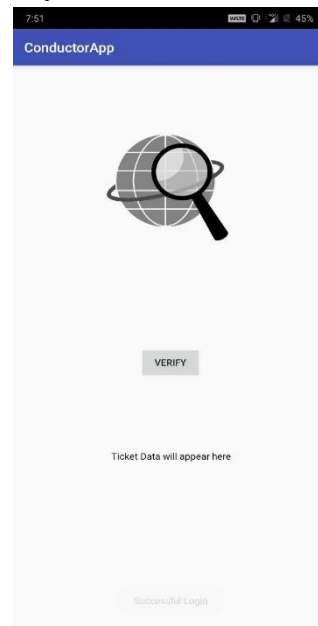

Figure 8: QR Code Scanner

Figure 9: Conductor Verification

Smart E-Ticketing System for City Bus Services   6

6. Admin page:

The admin page is used for the system controlling for the depot. It consists of allocation of monthly passes, wallet for the depot, driver and bus details. The monthly pass is only issued from the depot portal and can't be done from within the app. The wallet option in the web page shows the budget and the number of monthly passes available for distribution. Admin page is used mainly by the depot to store the details of the driver, conductor, bus details and their routes.

Admin Management

| Employee Data | Register Employee | Budget/Transactions | Contact |

Please Enter the Employee Details

Enter Name
Email ID
Contact No.
Age
Username
Password

Register

Figure 10: Depot Manager Web Page

7. Test Case Report

1. AWS Database Tables Creation:

The below figure shows the table which are developed in AWS MySQL using EC2 instance. The below tables contains: Journey fares, User details, Employee Details, Ticket history, Ticket data, etc.
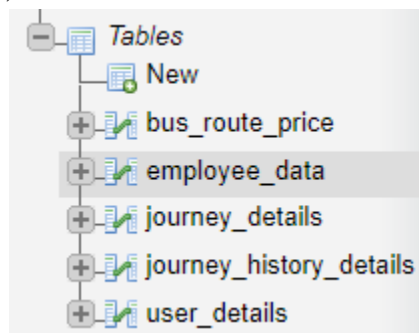

Tables
New
bus_route_price
employee_data
journey_details
journey_history_details
user_details

Figure 11: Database Generated Tables

2. **Registration:**

The below figure 12 shows the process of registration of user and after registering the user the password will be securely stored in the database. The figure 13 shows about the user registration row in the table.
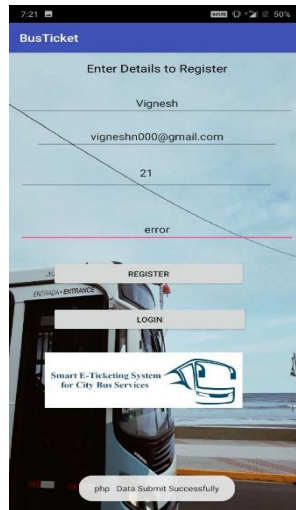

Figure 12: User Registration
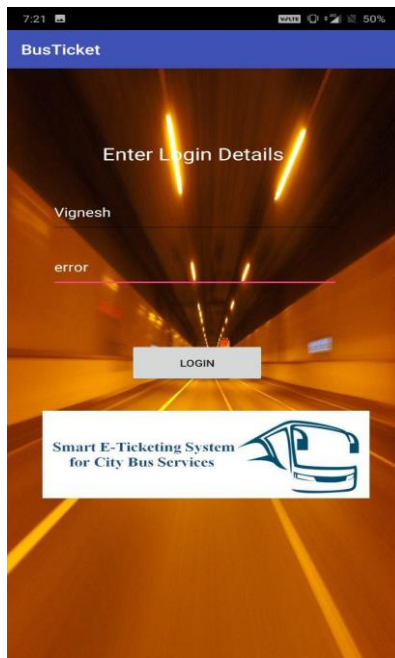

Figure 13: Database User data

3. **Login:**

Figure 14: Sign in page for user

The figure 14 shows the Log In page for user this page will require Login Id and Password to be entered in the application for further ticket booking process. If password fails to be entered correctly then the Login fails.

4. Ticket Booking:

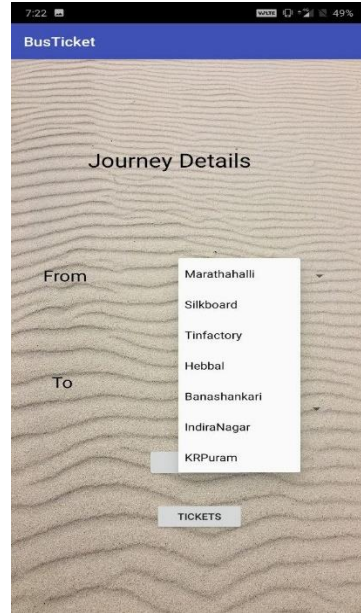The figure 15, shows us the interface for Journey detail selection

Figure 15: Journey Selection

Figure 16, shows the rows in the database after successful completion of ticket booking

			jid	username	from_address	to_address	ticketdata	timestamp
☐	✎ Edit	⌗ Copy ⊖ Delete	1	Vignesh	KRPuram	Tinfactory	KRPuramTinfactory07112019192152	2019-11-07 13:52:08
☐	✎ Edit	⌗ Copy ⊖ Delete	2	Vignesh	Tinfactory	Silkboard	TinfactorySilkboard07112019192152	2019-11-07 13:52:16
☐	✎ Edit	⌗ Copy ⊖ Delete	3	Vignesh	KRPuram	Marathahalli	KRPuramMarathahalli07112019192152	2019-11-07 13:52:25
☐	✎ Edit	⌗ Copy ⊖ Delete	4	Vignesh	Tinfactory	Marathahalli	TinfactoryMarathahalli07112019192152	2019-11-07 13:52:44
☐	✎ Edit	⌗ Copy ⊖ Delete	5	Vignesh	KRPuram	Marathahalli	KRPuramMarathahalli07112019192743	2019-11-07 13:57:59

Figure 16: Table for QR Code generation and validation

Event:

The below figure 17 shows the event which will trigger in the database and delete the tickets from the table. The tickets can be deleted after 10 minutes of ticket generation. According to the requirements the successful payment bus ticket should be valid for 10 minutes.

Figure 17: QR code reject event

After the Event trigger all the entries which are generated for 10 minutes or more than 10 minutes are removed as shown in figure 18.

Figure 18: Empty QR code table

The figure19 is showing the table which contains all the journey details with all the historic data maintained for any future issue or discrepancies.

Figure 19: Permanent table for user details
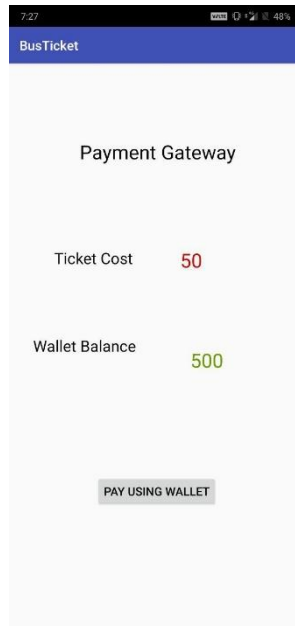
5. Payment Gateway

Figure 20: Payment Summary

The figure 20 shows the Payment Summary and the payment mode for purchasing the ticket

6. QR Code Generator

Figure 21: QR Code

The figure 21 shows the QR Code generated after successful payment of ticket. The QR code contains Start and Destination data with date and time the ticket payment has been completed.

7. Conductor QR Scan
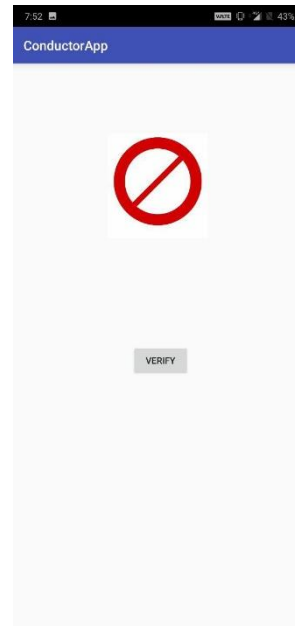

Figure 22: Approved Ticket

Figure 23: Rejected Ticket

The figure shown above is the interface of conductor to validate the ticket condition. If the ticket generation is done within 10 minutes then the ticket status will be "Approved", if the ticket generation time is more than 10 minutes then the ticket is "Rejected" and to buy a new ticket.

8. Conclusion:

The Android app developed for the passengers makes it easier to book tickets and eliminates the hassle of paying in crowded places and often not getting back the balance amount taken by the conductor. In this scenario the passenger can book the ticket 30 minutes before their commute which is then expired after a certain period of time. The conductor who has the scanner app scans the ticket of the passenger eliminating the need for paying back the balance in a crowded environment.

9. Future Enhancement:

The customer app can be improved in many ways further enhancing the capabilities of the customer in ease of managing the journey details.

- A fingerprint recognition module can be added to the app replacing the traditional login method for a faster and secure way of authentication.
- Adding support for more third party payment services including PayPal, UPI etc.
- Including more features such as one tap booking from history which takes the previous booking history and repeats the booking process, easier and best for daily commuters.

10.References

[1]. https://developer.android.com/docs

[2]. https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/install-LAMP.html

[3]. https://stackoverflow.com/questions/38217414/android-studio-tutorial

[4]. https://www.journaldev.com/18198/qr-code-barcode-scanner-android

[5]. https://www.tutorialspoint.com/android/android_php_mysql.htm