European Scholar Journal (ESJ)
Available Online at: https://www.scholarzest.com
Vol. 2 No. 1, January 2021,
ISSN: 2660-5562

SHUTTLE BUS MOBILE BOOKING APPLICATION

Nur syuhada Idayu Adnan,Firkhan Ali Hamid Ali
Faculty of Computer Science and Information Technology,
UTHM, Johor, Malaysia.

Article history:

Received: 8th December 2020
Accepted: 24th December 2020
Published: 8th January 2021

Abstract:

In the rise of information technology, manual routine, system, and activity has turned out to be useless and a waste of time. In Malaysia, the manual system is still working as daily activity where digital system can be replace it. The lack of knowledge and learning was the main factor of the manual system exist and not being replace.
The key point of proposing this idea is to build a mobile application for shuttle bus booking system. Its goal is to convert the manual routine of shuttle bus booking into a computerized system to ease the routine proses.
The proposed system will serve as a communication platform between drivers and users. The system will allow user to book a shuttle bus to ride and pick a drop-off destination beforehand. The system will also allow the bus driver to see the list of bus stops they need to pick and drop user. The rides information will be record for internal usa

Keywords: Shuttle bus booking system, mobile application, geolocation

1.INTRODUCTION

Transportation services are highly in demand nowadays. People used it daily as their means of transportation to work, classes, and meetings. Public transport was one of the transportations services that are greatly used these days. However, it is hard to know the availability and movement of these services (Xie, C & etc,2010). There are computer system and application that can help in detect vehicle movement, provide a booking system for a ride and a medium of communication between the services and its provider (Chow & etc, 2015). Grab Car application by Grab Holdings is one of the most used mobile transportation services nowadays. Its ability to order a ride from user mobile application and choose the nearest driver have made it popular among transportations services consumers (Ghahramani, N. & Brakewood, C.,2016).

The problem based on this case study is that there are no computerized shuttle bus system or application that can provide drivers early information of which bus stop they need to take and drop student. Drivers only know which stop to take students only by reaching each bus stops. This situation prevents them from planning a safety ride by suddenly stopping at bus stops in order to pick or drop students Be-sides, student also unavailable to notify drivers their wish to ride the bus at certain stop beforehand (Razally, W & etc, 2018). Their only way to alert the drivers is by reaching to the bus stop and wait there. Because there are different routes designated for different busses, student need to wave their hand to the bus they want to ride to alert the driver to stop there. They will miss the bus if they are just standing there without stop the bus. Lastly, there are no method or way to record all the rides history (Rouse, M. , 2017). The system website will only provide the movement of the bus but will not record the ride. In any cases of complaint of unavailability or missing service of one bus can only be check through satellite record of bus movement around the campus and the CCTV camera operate at university main gate.
To overcome the rise issues stated above, a mobile application which is university shuttle bus application is proposed through this paper. This mobile application will be operate using Android system and will contribute in:

1. To design a booking system between busses and students through university shuttle bus application. All ride and drop information to bus stops will be through this application.
2. To develop a system that provide and share real-time information of available busses, location of busses and students to its users for them to plan their ride.
3. To test the system is able to book and record rides between drivers and students.
4. Records can be used for internal audit and further improvement and improvement of the system.

This application will be designed to be develop for university transportation service only. This application will help to contribute to the objective stated and handle the issue from the existing system. The application will allow students to order a ride from the application and choose the closest driver match the routes he/her taken. Drivers will be able to view the ride orders from the students. The application will also record all the rides happened and allow students and drivers to view this history.

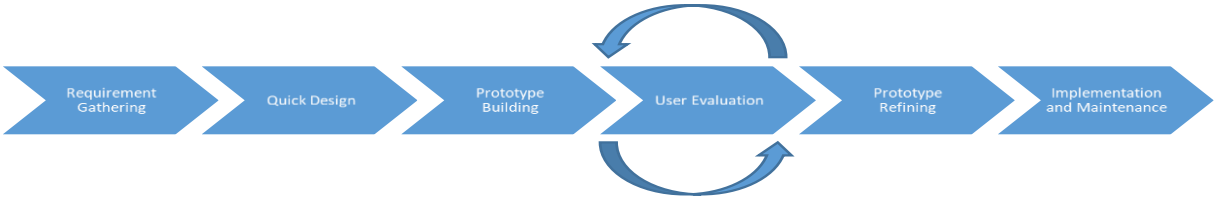
2.MATERIALS AND METHODS:

The materials and methods section, otherwise known as methodology, describes all the necessary information that is required to obtain the results of the study.

In order of developing this application, three existing systems and mobile application has been analyzed and studied to learn its features and the benefits of it. Those applications are known as System A, System B and System C for privacy issue. Besides, ideas, technologies and preventive measures also can be collected from these studies. Comparison of applications' features and benefits with the proposed application was shown as following.

Features	System A	System B	System C	Proposed System
Bus Real-Time Location and Movement	This system does provide this feature. However, it does not differentiate bus that are active and not active. Busses location are shown depending on the route user choose.	This system does provide this feature.	This system does provide this feature. Bus that are active are label with green while non-active bus is label with red.	This system does provide this feature. Active busses will be shown on map while non-active busses will not be shown.
Bus Stop Location	Bus stop locations are shown on the map. The locations of the bus stops are shown depending on the route user choose.	Bus stop locations are shown on the map. The application also provided the direction of bus through each stop.	Bus stop locations are provided on the map. The locations of bus stop will be shown based on which routes user choose.	Bus stop locations are provided on the map. The bus stops shown are depend on user routes choose.
Login and User Personalization	-	-	The application provided the login page.	The application provides login page for its user.
Time Estimation	-	-	Time estimation are provided in this application. It estimated the first closest bus to arrive at one bus stop. Then, it estimated the next closest bus after the first one.	Time estimation are provided in this application. Bus have been assigned to user will be calculate its estimated time to arrive at user location.
Bus Booking	-	-	-	Booking bus are provided in this application.

In developing this system, software model development that have been implement in prototyping model. In prototyping model, prototype of the application is being built, tested, and reworked. These processes are being done repeatedly until the out-put is acceptable by the client (Arora, I., & Gupta, A., 2012). Phases that are included in this model is requirement gathering, quick designing, built prototype, prototype testing or user evaluation, prototype refining, and implementation and maintenance as the following figure. All the project planning and requirements and specification are gathered in requirement gathering phase. Based on the requirement gathered, a quick design of the application is being done in quick design phase (Popescu, A., 2010). Then, a prototype was built from the design created. The prototype then was being tested and evaluated by user before going to prototype refining phase. The evaluation and prototype refining phase are being repeat until the prototype have reached the best outcome that is acceptable by user (Boell, S. K., & Cecez-Kecmanovic, D., 2014). The final changes of the prototype then the being implemented and maintain in implementation and maintenance phase.



There are four type of prototype software model development that is rapid throwaway prototype, evolutionary prototype, incremental prototype, and extreme prototype. Type of prototype model used in developing this application is evolutionary prototype. In evolutionary prototyping model, it is the most time and effort saving rather than the other three. It is gradually built based on user feedback on each version of prototype. After developing one prototype, user feedbacks are collected, and a new version of prototype are being built from the previous version. The cycles are repeated until the prototype fully acceptable by user. This type of prototype model is helpful if the requirements are not fully understood in the initial stage of developing. Besides, it is the most beneficial model to use if each functionality of a complex project is needed to be checked. All this phase had been done as following table.

Phase	Task	Output
Requirement Gathering	Planning requirement gathering technique	Requirements technique choose:
	Conduct an interview session with stakeholder	Interview
	Prepare and submit referral letter requesting for interview	Research existing system
	Prepare interview question	User observation
	Held an interview with stakeholder	System requirements are gathered.
	Research and study existing system	Understand system flow.
	System A	Knowledge about technology used in the application.
	System B	Learn technology and technique encouraged for the proposed system and avoid
	System C	Learn user behavior towards the existing system
	User observation	List of specification and requirements for the application
	Observing user activity and action on the existing system	
	Sorting and determine user requirements.	
Quick Design	Design UML diagrams for system	Use case diagram
	Designing database system	Sequence diagram
	Determine data, attributes, and key fields for database	Activity diagram
	Perform data normalization	Completed database system design, tables and attributes.
	Design system's database	ERD diagram visualizing database and its entities relationship.
	Design a conceptual model and ERD	User interface design for students
	Designing User Interface	User interface design for drivers
Prototype Built	Develop application prototype based on system design.	First version of prototype
User Evaluation	Testing the user interface.	User evaluation and feedback on the application interface and its functionality.
	Testing the application functionality.	Receive feedback on functions, design, and interface to be improve.
Prototype Refining	Repair and refining the existing prototype.	New and refined version of prototype.
Implementation and Maintenance	Implement the accepted prototype to a final product.	Final product released.
	Maintenance.	

3.RESULT AND DISCUSSION

The final product is being develop by using Android Studio using Java language. The database that are being used is a Firebase Real-Time Database. It is a cloud hosting database. It used NoSQL query to query the data and saved the data in string in JSON format. The product then being test by using functional testing method (Yakura, E. K., 2002).

Functional testing module are used to test all the module functionality in the application. There are six test plan that have been conduct according to the modules in the application which is login module test plan, registration module test plan, student map module test plan, driver map module test plan, user profile module test plan and ride history module test plan. The table of test plan can be seen in the following tables.

Item	Test Case	Expected Result	Actual Result	Pass/Fail	Correction to Be Done
1	Insert email and password	Login success-fully and directed to main module.	Same as expected result.	Pass	None
2	Not inserting email or password.	Alert text show "Do not leave any of the field empty!".	Same as expected result.	Pass	None
3	Login with un-registered account.	Alert text show "Sign in error!"	Same as expected result.	Pass	None
4	Enter wrong password for the email.	Alert text show "Sign in error!"	Same as expected result.	Pass	None

Item	Test Case	Expected Result	Actual Result	Pass/Fail	Correction to Be Done
1	All the field are inserted correctly.	Register success-fully. Alert text show "Register successfully! Go back to login."	Same as expected result.	Pass	None
2	All the field are empty.	Alert text show on top each of the empty field.	Same as expected result.	Pass	None
3	One of the field empty.	Alert text show on top of the empty field.	Same as expected result.	Pass	None
4	Enter wrong format for email.	Alert text show "Sign up error!" and cannot register account.	Same as expected result.	Pass	None
5	Password are not longer than six character.	Alert text show on top of the empty field saying "Password need to longer than six character".	Same as expected result.	Pass	None

Item	Test Case	Expected Result	Actual Result	Pass/Fail	Correction to Be Done
1	Choosing bus route.	The bus stop location will show correctly as designated for the route.	The bus stop location does not change.	Fail	Correcting the code for radio button.
2	Click on the bus stop marker.	Bus stop location will be display on top of the marker and on top of Call Bus button.	Same as expected result.	Pass	None
3	Choose bus stop marker and click Call Bus button.	Successfully request ride. Student location, destination name and destination location will be update in the database. Fetching Bus button showed.	Same as expected result.	Pass	None
4	Not choosing bus stop marker and click Call Bus button.	Alert text show "Choose drop destination" and cannot book bus.	Crashed and exit the application.	Fail	Fix the coding in Call Bus button.
5	If there are bus available.	Student request sent to bus driver.			
6	User cancel the ride.	Driver detail and location will be disappeared. Call Bus button will be shown.	Same as expected result.	Pass	None
7	Click on the Profile button.	User directed to user profile interface.	Same as expected result.	Pass	None
8	Click on the History button.	User directed to user history interface.	Same as expected result.	Pass	None
9	Click on the Sign Out button.	User will be signed out from the application and directed to login interface.	Same as expected result.	Pass	None

Item	Test Case	Expected Result	Actual Result	Pass/Fail	Correction to Be Done
1	Upon login and directed to this interface.	Driver id and location updated on the database as driver available.	Same as expected result.	Pass	None
2	Received student request.	Student detail and location shown on the interface. Pick Up button shown on the interface.	Same as expected result.	Pass	None
3	Clicking Pick Up button.	Driver id and location will be removed from driver available to driver working. Drop off button are shown.	Same as expected result.	Pass	None
4	Clicking Drop Off button.	Driver id and location will be removed from driver working to driver available. Student location and destination will be cleared from map.	Same as expected result.	Pass	None
7	Click on the Profile button.	User directed to user profile interface.	Same as expected result.	Pass	None
8	Click on the History button.	User directed to user history interface.	Same as expected result.	Pass	None
9	Click on the Sign Out button.	User will be signed out from the application and directed to login interface.	Same as expected result.	Pass	None

Item	Test Case	Expected Result	Actual Result	Pass/Fail	Correction to Be Done
1	Upon directed to this interface.	User detail in database are retrieved and shown here.	Same as expected result.	Pass	None
2	Click the profile image field.	User be able to choose image from their device.	Same as expected result.	Pass	None
3	Not changing any data on the field and click Update button.	Data in the database will not be updated and changed.	Same as expected result.	Pass	None
4	Click the Update button.	User detail are updated and saved in database. User detail on the interface was updated.	Data was saved on the database. However, user is directed to the previous interface.	Fail	Need to fix code in Update to avoid bring using back to the previous screen after updating.
5	Clicking Back button.	User will be directed to map interface.	Same as expected result.	Pass	None

Item	Test Case	Expected Result	Actual Result	Pass/Fail	Correction to Be Done
1	Upon directed to this interface.	All ride history related to user listed here.	Same as expected result.	Pass	None
2	Click one of the listed histories.	A single page history should be display with all the detail of the ride.	Error. The page show and closed instantly.	Fail	Check nodes name in database if the nodes name stated correctly or not. Check the value has been pass correctly or not.
3	Clicking Back button.	User will be directed to map interface.	Same as expected result.	Pass	None

4.CONCLUSION

In conclusion, Shuttle Bus application is built to ease the booking and ride between busses and students through booking system provide by this application. Besides, this application also helps in providing and sharing real-time information of available busses, location of busses and students to its user. Aside from that, this application can help in diminish unsafety way of stopping bus at bus stop by booking a ride from this application. This also help in arranging student movement in riding the bus in a mannerly way without pushing and forcing into the bus. It is because for each student ride request, busses will be assigned to them. However, there are a lot of improvement can be done in future. One of the suggested improvements is adding the notification module. This module will help to notify student upon bus arrival. Besides, it is also recommended to include a Full button for driver to indicate that the bus is full and cannot received any student request. Finally, development of this application is hoped to cater the problem stated before and will be beneficial for its user.

REFERENCES

1. Xie, C., Gong, H., & Wang, F. (2010, June). A solution for the last mile problem of the beijing rapid transit network: Local shuttle bus system. In 2010 18th International Conference on Geoinformatics (pp. 1-6). IEEE. https://doi.org/10.1109/GEOINFORMATICS.2010.5567502
2. Chow, S., Okazaki, M., Watanabe, T., Segawa, K., Yamamoto, T., Kurogi, H., ... & Mochioka, N. (2015). Light-sensitive vertical migration of the Japanese eel Anguilla japonica revealed by real-time tracking and its utilization for geolocation. PLoS One, 10(4). https://doi.org/10.1371/journal.pone.0121801
3. Ghahramani, N., & Brakewood, C. (2016). Trends in Mobile Transit Information Utilization: An Exploratory Analysis of Transit App in New York City. Journal of Public Transportation, 19(3), 9. http://doi.org/10.5038/2375-0901.19.3.9
4. Razally, W., Sulaiman, N., Rohani, M. M., Daniel, B. D., Radzuan, I. S. M., & Junaid, T. M. (2018). Reengineering sustainability at Universiti Tun Hussein Onn Malaysia (UTHM): travel & transportation. In E3S Web of Conferences (Vol. 48, p. 07003). EDP Sciences. Retrieved from https://www.e3s-conferences.org/articles/e3sconf/abs/2018/23/e3sconf_iwgm2018_07003/e3sconf_iwgm2018_07003.html
5. Rouse, M. (2017, March). Fleet Management [Blog post]. Retrieved from https://whatis.techtarget.com/definition/fleet-management
6. Arora, I., & Gupta, A. (2012). Cloud databases: a paradigm shift in databases. International Journal of Computer Science Issues (IJCSI), 9(4), 77. Retrieved from http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.402.3638&rep=rep1&type=pdf
7. Popescu, A. (2010). Geolocation api specification. World Wide Web Consortium, Candidate Recommendation CR-geolocation-API-20100907. Retrieved from https://www.immagic.com/eLibrary/ARCHIVES/TECH/W3C/W081222P.pdf
8. Boell, S. K., & Cecez-Kecmanovic, D. (2014). A hermeneutic approach for conducting literature reviews and literature searches. In Communications of the Association for Information Systems, 34(1), 12. https://doi.org/10.17705/1CAIS.03412
9. Yakura, E. K. (2002). Charting time: Timelines as temporal boundary objects. Academy of Management journal, 45(5), 956-970. https://doi.org/10.5465/3069324