CouponCar: An Android Based Application to Automate the Street Parking Payment

N. Atifah Hammade[1], Rozaida Ghazali[2, *], Salama A. Mostafa[3], Bashar A. Khalaf[4]

[1,2,3]Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia, 86400, Johor, Malaysia.
[4]Department of Medical Instruments Engineering Techniques, Bilad Alrafidain University College, 32001, Diyala, Iraq.

Emails: nur.atifah95@gmail.com; rozaidabtghazali@gmail.com; salama@uthm.edu.my; bashar@bauc14.edu.iq

***** Correspondence: rozaida@uthm.edu.my

Abstract

In the current system of making a street parking payment in Malaysia, citizens are using a manually paper-based parking coupon that still lacks in terms of the payment process, thus making it difficult for Malaysians to pay for their parking. Therefore, the CouponCar application is proposed to help citizens in Malaysia to pay for their parking with ease. Citizens do not need to buy a parking coupon at the city council or any agents that sell the parking coupon. Instead, the manual system will be replaced by using an android based application where they just need to install the application inside their smartphone, tablet, or any suitable device. This system will also help the officer to check many cars within a short time, whereas they can easily scan the QR code on every car's dashboard. The system was developed using a structured approach and based on activities in the system prototyping model. Overall, this system can facilitate the citizen to make a street parking payments in Malaysia.

Keywords: Couponcar; Street Parking; and Information Technology

1. Introduction

Industrialization and the improvements in the automobile industry had not only resulted in increased wealth, but it had led to traffic congestion and parking problems in cities. In Malaysia, the number of newly registered vehicles has increased tremendously [1]. In the year 2013, the total registered motorcars had increased to 23,819,256 units (Malaysian Ministry of Transportation, 2013) compared to the year 2008, when there were only 17,971,901 units of registered motorcars (Malaysian Ministry of Transportation, 2008). This means there has been an increase of 32.5 percent within a span of 5 years [2]. With the enormous increase in car ownership due to economic development and the growth of urban areas due to urbanization, it is becoming increasingly difficult to find a parking spot in crowded areas [3], [4]. Today's parking systems are therefore clearly an essential component of the transportation system, including the street parking payment system. Unfortunately, many states in Malaysia are still using the manual system, the paper-based parking coupon. Although it is a new improvement from the older parking payment system, the parking meter, the current system still had several disadvantages [5].

Motivated by the situation, CouponCar is developed to effectively make a street parking payments in Malaysia. Citizens do not need to buy a parking coupon at the city council or any agents selling the parking coupon [6]. Instead, the manual system will be replaced by using an android based application where they just need to install the

DOI:https://doi.org/10.54216/FPA.060205

application inside their smartphone, tablet, or any suitable device. Next, there will be no boundaries anymore, whereas citizens will use only one application in different states and even different cities in each state [7]. Besides that, this application can help the citizens park at any available zone inside the app. They can choose which zone they use to park their car at that time. Furthermore, the citizens don't need to worry about the time left to park their car as they are the ones who will start the timer and end it when they are done parking. Moreover, no more wasting money. The system will calculate the time and substitute the total payment from the user's credits balance, so no money is wasted. Finally, this system will help the officer to check many cars within a short time, whereas they can easily scan the QR code on every car's dashboard [8].

2. Related Work

This section explained about literature review that has been conducted for this project. The information gained from the review was needed to strengthen understanding and get a big picture about this project's requirements. This section briefly presents the review of similar existing systems to see the differences and similarities with the proposed system.

Three similar systems were compared with the developed system. The three similar systems are JomParking, ParkBox, and MBJB Spot. It is essential to study these three similar existing systems to identify the advantages and disadvantages of the proposed system with the existing systems. Therefore, by performing this study, it can help to improve the proposed system.

JomParking [9] was developed by AppCable Sdn. Bhd. JomParking is an application to pay for street parking using a mobile phone. JomParking is available at all metered street parking spaces and certain city-owned parking lots in Klang Valley. Main street parking spots will be activated using the JomParking service within the Klang Valley. JomParking logo will be displayed on the designated street parking area. To use this application, users need to register for an account first. During the parking process, users need to enter the zoning code, for example, Ampang, and choose their car plate number. They can add or delete their car plate number in the My Vehicles section. They also need to enter the park duration. This application uses tokens to pay for parking. The token is a payment method in JomParking that will be used to pay the street parking fee. The tokens can be purchased at a minimum of RM6 and can be utilized in JomParking apps. Parking is monitored via automatic license plate readers on specified vehicles and handheld devices. Each time user uses JomParking, users will enter the license plate number, which identifies the parking session. Enforcement personnel will look up the license plate number to determine if it is paid [10].

Digital Sdn Bhd [11] proposes a ParkBox project that aims to develop an automated smartphone street parking payment application. ParkBox is available at all metered street parking spaces and certain city-owned parking lots in Taiping. Main street parking spots will be activated using the ParkBox service within Taiping. To use this application, users need to register for an account first. During the parking process, the users need to enter the parking zone. They also need to enter the park duration. This application uses credit to pay for the parking. The credit is a payment method in ParkBox that will be used to pay the street parking fee. The credit can be purchased and can be utilized in ParkBox apps.

Presgo Gateways Sdn Bhd developed MBJB Spot in [12]. MBJB Spot is an application to pay for street parking using a mobile phone. MBJB Spot is available at all metered street parking spaces and certain city-owned parking lots in Johor Bahru. Main street parking spots will be activated using the MBJB Spot service within Johor Bahru. To use this application, users need to register for an account first. During the parking process, users need to enter the zone, for example, Green Zone, and choose their car plate number. They can add or delete their car plate number in the Vehicles section. They also need to enter the park duration. This application uses reload to pay for the parking. The reload is a payment method in MBJB Spot that will be used to pay the street parking fee. The reload can be purchased at a minimum of RM20 and can be utilized in MBJB Spot apps [13].

Table 1: Comparison between similar existing systems with the developed system

Features/	JomParking	ParkBox	MBJB Spot	The Proposed CouponCar
Platform	Application only	Application only	Application only	Web-based and application
Sign up function	Yes	Yes	Yes	Yes, using the web-based

DOI:https://doi.org/10.54216/FPA.060205

Sign in function	Only in application	Only in application	Only in application	Both web-based and
Manage profile	Yes	Yes	Yes	Yes
Manage car	Yes	Yes	Yes	Yes
Generate report	No	No	No	Yes, using a web-based
Using QR code	No	No	No	Yes
Online payment	Only in application	Only in application	Only in application	Both web-based and
Location	Klang Valley only	Taiping only	Johor Bahru only	Every city and state in
Detect location	No	Yes	No	Yes
Display technology	No	Google Map	No	Google Map
Assuming the time	Yes	Yes	Yes	No
Notification	The time limit is	The time limit is	The time limit is	Low level of credit balance
Accurately calculate the time period used	No, the user still needs to assume	No, the user still needs to assume	No, the user still needs to assume	Yes
iMoved feature	No	No	No	Yes
Well functioned	Yes	Yes	No	Yes
User-friendly	Yes	Yes	No	Yes
Respond period	20s	30s	50s	15s

Based on table 1, there are several similarities and differences between the three existing and developed systems. There are a few similarities between all these four systems, such as all of them provide a function to manage the car's information, manage profile, and have a sign-up function. Meanwhile, the differences that can be obtained from this table are the existing system only can be used in the application while CouponCar can be used in both web-based platforms, and applications with each of them will perform different functionalities. CouponCar also has the function to generate reports, detect the location, and use the QR code and Google Maps technology. Furthermore, CouponCar can be used in every city and state in Malaysia, while others can be used only in a certain area. Besides the online payment that can be made on both platforms, CouponCar can accurately calculate the time period used to park the car and provide notification for low credit balance. Moreover, CouponCar also has an iMoved feature, a user-friendly interface, and well-functioning, and the fastest response period. The results obtained from this study are referred to as a guideline to develop the complete application system.

3. Research Methods and Materials

This proposed system is developed to help Malaysian citizen to pay their street parking payment with ease. CouponCar has two web-based and mobile application platforms, but the main process will be using the mobile application. In a web-based platform, the users will register to create an account. After the users have successfully registered, the system will automatically generate the QR code for every account [3]. Then, users can log in to the system and manage their profile and their car's information. They can also print the QR code and stick it on their car's dashboard and generate parking session reports. Next, users need to buy credits for their accounts using online banking or credit card. During the main process, when users want to park their car, they will be using the application installed inside their smartphone, tablet, or any suitable device. They will open the application, and they need to log in using their account. Then, they need to choose their location and type of zones, and they need to click the button on to start the timer and click the off button to turn it off when they finish parking.

CouponCar system prototyping is selected as the methodology for this project. The main reason system prototyping is selected as this project methodology is because of the short time schedule provided to complete this project and the documentation. Iterative development and system prototyping are excellent choices when timelines are short because they best enable the project team to adjust the functionality of the system on the basis of a specific delivery date [14]. Looking into the conditions of CouponCar development, system prototyping would be the most suitable methodology to be used. Compared to waterfall development, where analysts and users proceed sequentially from one phase to the next [15], system prototyping is iterative which means, a phrase could be repeated if deficiencies are detected in CouponCar. This is the key disadvantage in waterfall development, where the design must be completely specified before programming begins. A long-time elapses between the completion of the system proposed in the analysis phase, and the delivery of the system, and testing is treated almost as an afterthought in the implementation phase.

DOI:https://doi.org/10.54216/FPA.060205

However, phases in the system prototyping could be worked on in parallel with other phases resulting in faster development and documentation of the project. On top of that, by producing an early version of CouponCar to be tested by the users. A better version of CouponCar could be produced at the end of the timeline to assure good feedback for the evaluation of CouponCar. Thus, system prototyping is selected as the methodology for this project. As shown in Figure 1, the prototyping model has five development phases: the planning, analysis, design, prototyping, and implementation phase. The overall implementation of the prototyping in the development schedule is best described in the following.

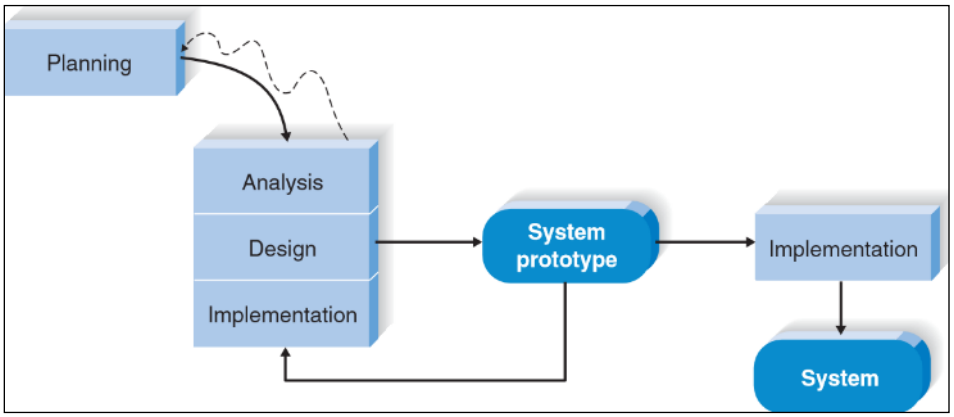

Figure 1: System prototyping [7]

Table 2 shows the Workflow process for the system prototyping model. Based on table 2, each phase has its own assignment and output that need to produce during the entire project development. Besides that, the output had been completed within the specific days that have been given.

Table 2: Methodology phases and their activities

Phase	Activity	Output
Planning	• Identify the project background, problem statement, objectives, and scope of the project. • Review on related field • Proposed the project • Determine the project schedule, activities, and output	• Project proposal • Develop Gantt chart
Analysis	• Produce the interview questions and conduct an interview session with the users • Produce the questionnaire and distribute it to the users • Analyze the requirements • Design the context diagram and data flow diagram • Design the flowchart • Design the entity-relationship diagram	• Interview questions paper • Questionnaire • As-is model and to-be model • Context diagram • Data flow diagram • Flowchart • Entity Relationship diagram
Design	• Design user interface • Design database	• User interfaces designed • Database designed
Implementation	• Write a code for each system module and connect it with the database • Confirm the proposed system fulfills the user requirements	• Developed system • Test case
Prototyping	• Build prototype version 1 • Build prototype version 2 • Build prototype version 3	• Prototype version 1 • Prototype version 2 • Prototype version 3

DOI:https://doi.org/10.54216/FPA.060205

The system calculates the time period and automatically substitutes the total payment from the user's credit balance. Furthermore, in the application, the user also can manage their profile and their car's information, buy credits for their account and view the history of their previous parking session. Meanwhile, the officer can scan the QR code to check whether the user had activated their apps and whether they had parked their car in the correct zone or not. In order to do that, the officer also needs to have a smartphone or any suitable device to scan the QR code. In addition, this application will give a notification to the user if their account has reached a certain low level of credit balance in the account. It will alert the user to reload their credits for their account. This application also has the iMoved feature, which detects that the user may have driven away without ending the parking session and reminds the user to stop the current parking session. Figure 1 simplifies how system prototyping is conducted.

4. System Design

This section explains the result of the analysis and design of CouponCar. A structured approach was used to analyze and design this system. The analysis result was visualized in several diagrams such as a flowchart, Context Diagram (CD), Data Flow Diagram (DFD), and Entity Relationship Diagram (ERD). Interface diagrams were also produced in the design phase. Besides that, a data dictionary was built to store all the data for this system.
The system design determines the system's overall system architecture by defining the architecture, components, modules, interface, and data for a system to satisfy specified requirements [16], [17]. The system architecture is the conceptual design that defines the structure and behavior of a system. It also includes both interface design and database.

4.1. User's Flowchart

Figure 2 shows the flowchart for a user in the web-based platform. Users need to register first to create an account. After the user has successfully registered, the system will automatically generate the QR code for every account. Then, users can log in to the system with their registered username and password. If the login is successful, the user will be directed to the dashboard [18]. Otherwise, they have to log in again. They can manage their profile and also their car's information. They also can print the QR code and stick it on their car's dashboard. Next, users can buy credits for their accounts using online banking or credit card. Furthermore, they can also generate parking session reports to see their previous parking session.

DOI:https://doi.org/10.54216/FPA.060205

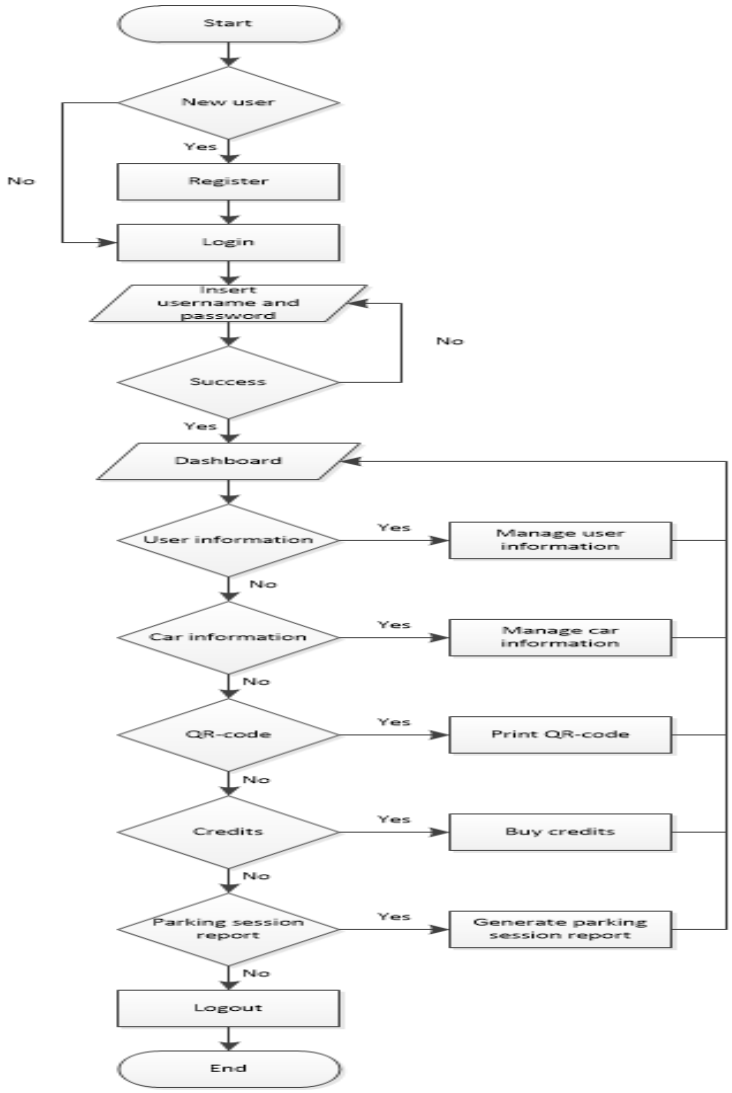

Figure 2: User's Flowchart in Web-based

Figure 3 shows the flowchart for user perspective in the application platform. Users can log in to the system with their registered username and password. If the login is successful, the user will be directed to the dashboard. Otherwise, they have to log in again. Users can manage their profile and their car's information, buy credits for their account, and view the history of their previous parking sessions history. To start the parking session, users need to choose their location and type of parking zone, and they need to click the button to start the timer and click the off button to turn it off when they finish parking. The system will calculate the time period and automatically substitute the total payment from the user's credit balance.

DOI:https://doi.org/10.54216/FPA.060205

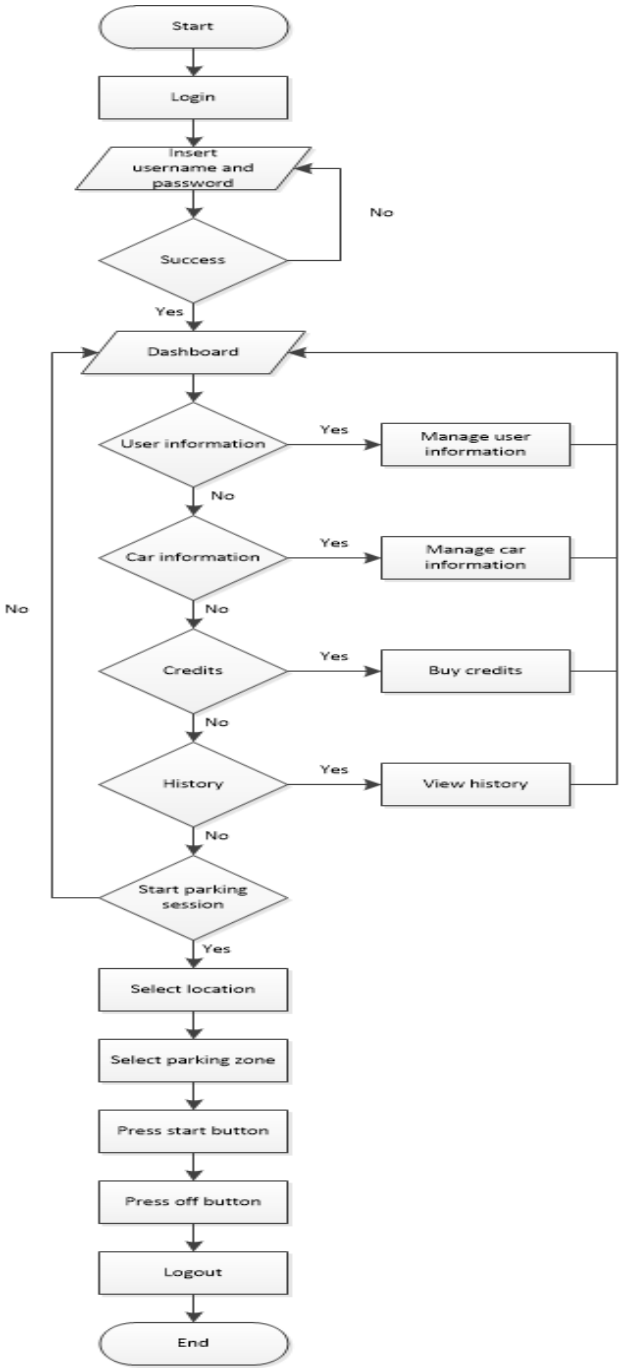

Figure 3: User's Flowchart in Application

4.2. Officer's Flowchart

Figure 4 shows the flowchart for the officer. Officers can log in to the system with their registered username and password. If the login is successful, the officer will be directed to the dashboard. Otherwise, they have to log in again. Officers can scan the QR code to check whether the user had activated their apps and whether they had parked their car in the correct zone or not. If the user has activated their apps and the other required information is correct, the officer can check another user's car. Otherwise, the user will be fined by the officer.

DOI:https://doi.org/10.54216/FPA.060205

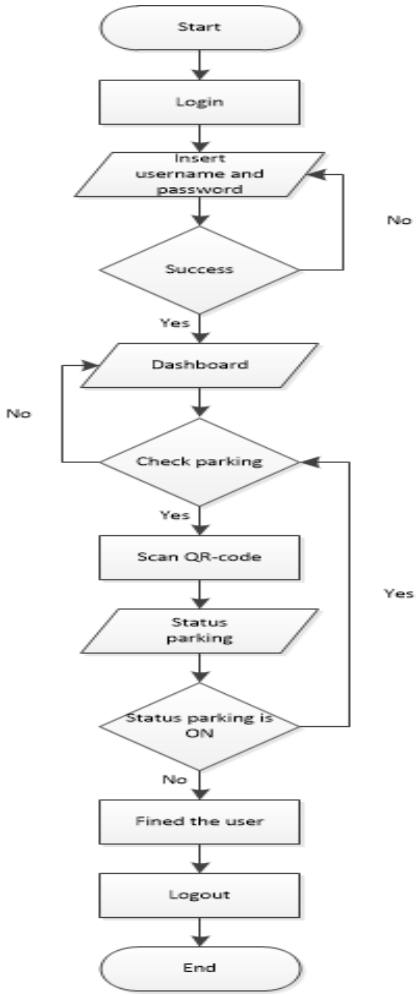

Figure 4: Officer's Flowchart

4.3. Database Discretion

The database design is determined by the characteristics of the data kept in the database. The characteristics are the attributes, type of the data, size, type of key, and the description of the data. The following are the tables from the database that have been designed and extracted from the ERD. They were created using HeidiSQL 9.4.

Table 3: User table

Attribute	Data Type	Size	Key	Description
user_id	int	11	Primary key	Id for each user
user_first_name	varchar	255		User first name
user_last_name	varchar	255		User last name
user_email	varchar	255		User email

DOI:https://doi.org/10.54216/FPA.060205

user_username	varchar	255		Username for the user to login into the system
user_password	varchar	255		Password for the user to login into the system
user_credits	double			Total user credits in their account
user_qr_code	longblob			User QR code for their account

As shown in Table 3, the user table is used to store all the user's information, including first name, last name, email, username, and password. user_id was used as a primary key to identify the specific user's information.

Table 4: car table

Attribute	Data Type	Size	Key	Description
car_id	int	11	Primary key	Id for each car
car_plate_no	varchar	255		Car plate number
car_name	varchar	255		Car name
car_color	varchar	255		Car color

As shown in Table 4 was used to store all the car's information, including plate number, car name, and car color. car_id was used as a primary key to identify the specific car's information.

Table 5: parking table

Attribute	Data Type	Size	Key	Description
parking_id	int	11	Primary key	Id for each parking
car_plate_no	varchar	255	Foreign key	Car plate number
user_credits	double		Foreign key	Total user credits in their account
parking_location	varchar	255		Parking location
parking_zone	varchar	255		Parking zone type
parking_price	double			Parking price
parking_date	date			Parking date
parking_time	double			Parking time period
parking_total	double			Total parking payment

DOI:https://doi.org/10.54216/FPA.060205

parking_status	varchar	255		The status of the user's parking

As shown in Table 5 was used to store all the parking information, including parking location, parking zone, parking price, and parking date. parking_id was used as a primary key to identify the specific parking information while car_plate_no and user_credits are the foreign keys for this table.

4.4. Interface Design

The interface design was built based on the result of the analysis. The following are the interfaces that have been designed based on DFD. They are designed by using Brackets and Sublime Text 3. Figure 5 shows the homepage interface of CouponCar. Any user can view this page without the need to login into the system. This interface provides updated news on CouponCar and other useful information.

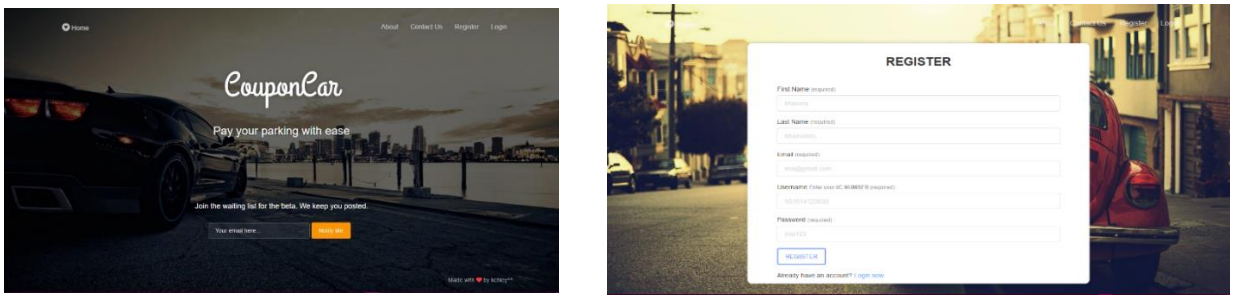

Figure 5: CouponCar homepage interface

Figure 5 also shows the user registration interface. This interface is for the user only. The new user has to fill in the information needed, such as first name, last name, email, username, and password. The user then needs to click the register button to store the user's information in the database.

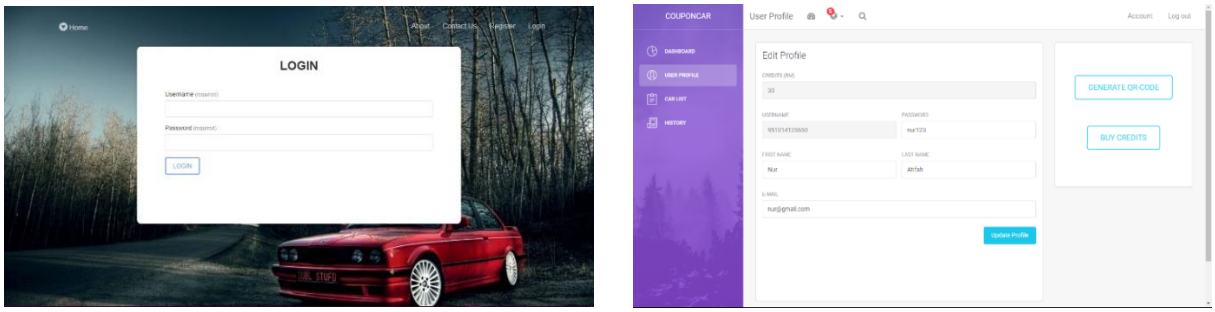

Figure 6: Web-based login interface

Figure 6 shows the login interface in a web-based platform. End-users involved in this system, such as users and admin, need to first log in to the system. To start the login process, they must enter the correct username and password. Next, the user must press the login button to login into the system. After login success, they will be navigated to their own homepage. Figure 6 also shows the user's profile interface. This interface is for the user only. Users can insert, update, delete and view their information. They also can generate the QR code and buy credits for their account.

DOI:https://doi.org/10.54216/FPA.060205

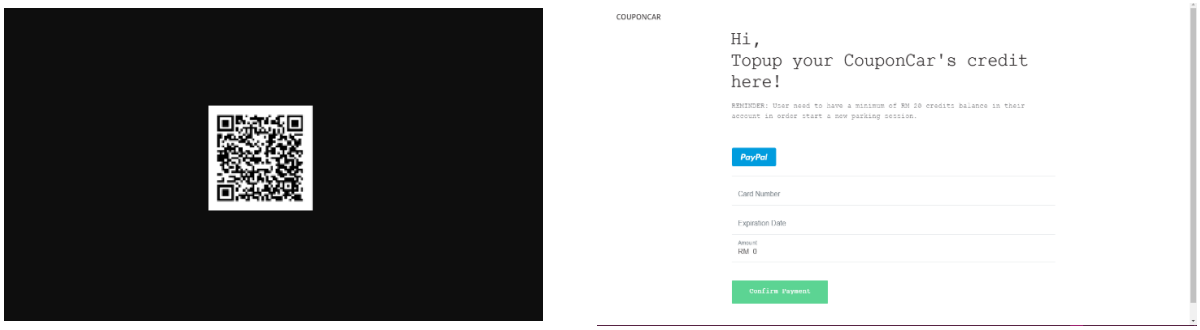

Figure 7: (a) Generate QR code interface, (b) Buy credits interface.

Figure 7(a) shows the generated QR code interface. This interface is for the user only. After users click the button to generate a QR code, they can display the QR code, save it as an image, print the QR code, and stick it on their car's dashboard. Figure 7(b) shows the buy credits interface. This interface is for the user only. Users can buy and top up their credits by using online payment. They need to enter several details such as card number, expiration date, and amount of credits that they want to buy. After that, they need to click the button to confirm their payment.

Figure 8: (a) Buy credits success interface, (b) Application login interface.

Figure 8 (a) shows the buy credits success interface. After users click the confirm payment button, they are directed to this page if the transaction is successfully processed. Then, their credit balance will be automatically updated by the system. Whereas, Figure 8 (b) shows the login interface in the application platform. End-users involved in this system such as users and officers need to log in to the system first. To start the login process, they must enter the correct username and password. Next, they must press the login button to login into the system. After login success, they will be navigated to their own homepage.

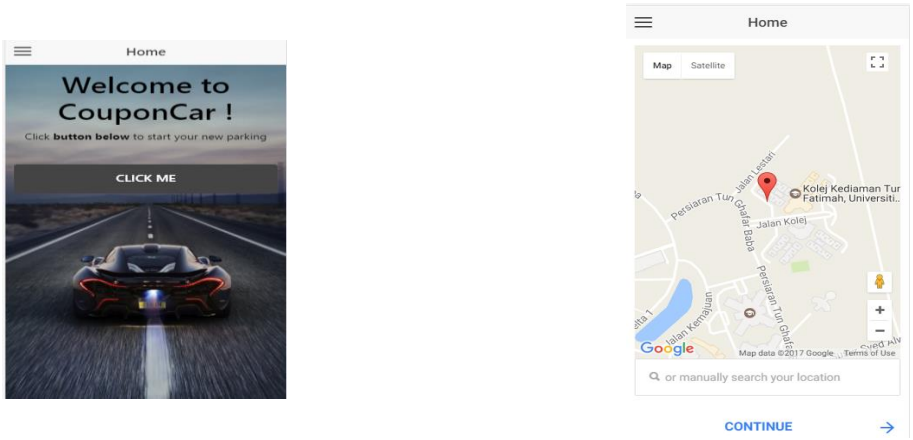

Figure 9: User's home interface, b) Parking session: page 1 interface

DOI:https://doi.org/10.54216/FPA.060205

Figure 9 (a) shows the user's home interface. This interface is for the user only. Users can click the click me button to start a new parking session. Figure 9 (b) shows the parking session interface for page 1. This interface is for the user only. This is when the user wants to park their car. GPS will automatically detect the user's location, and they need to click the continue button to proceed to the next step.

5. Implementation and Results

The implementation phase involves developing the final version of the system and the database based on the data flow diagram level 0 from the previous chapter and the requirements that had been gathered. PHP has been used as a programming language for the development of the system. Microsoft SQL Server has been used to develop the database and as a tool to connect the database to the system interface. Furthermore, each interface had been developed and tested during development. This is to make sure that the database is connected with the interface and can be manipulated from the interface without having to insert it from Microsoft SQL Server. In addition, this will help to find any error or bug and resolve it while developing the system.
The testing phase was conducted during and after the development of the CouponCar. A set of test cases was created based on the requirements of the system. In addition, the system was tested with users. Before that, the system was tested by the developer. This is to make sure that the system will be freed from any defects, errors, and bugs and performed as intended by the users.

5.1. Test Cases

Each test case should be traceable to the requirements stated in the previous section. Actual results for each test case also had been recorded. The details for each test case are listed below. The admin does user acceptance testing. Test results are shown in Table 6:

Table 6: User (Admin) Test Case

Allocated	Test Case	Result
SRS_ REQ_200	Login function	Pass
SRS_ REQ_300	Manage admin function	Pass
SRS_ REQ_400	Manage officer function	Pass
SRS_ REQ_500	Manage user function	Pass
SRS_ REQ_600	Manage parking lot function	Pass
SRS_ REQ_700	Manage website function	Pass
SRS_ REQ_800	Generate report function	Pass

The citizen does user acceptance testing. Test results are shown in Table 7:

Table 7: User (Citizen) Test Case

Allocated	Test Case	Result
SRS_ REQ_100	Register function	Pass
SRS_ REQ_200	Login function	Pass
SRS_ REQ_900	Manage user profile function	Pass
SRS_ REQ_1000	Generate QR code function	Pass
SRS_ REQ_1100	Buy credits function	Pass
SRS_ REQ_1200	New parking session function	Pass
SRS_ REQ_1300	Manage car list function	Pass

DOI:https://doi.org/10.54216/FPA.060205

SRS_ REQ_1400	Generate history function	Pass

The officers do user acceptance testing, and the test results are shown in Table 8:

Table 8: User (Officer) Test Case

Allocated	Test Case	Result
SRS_ REQ_200	Login function	Pass
SRS_ REQ_900	Manage user profile function	Pass
SRS_ REQ_1500	Scan QR code function	Pass

6. Analysis and Discussion

This section discussed the overall conclusion of the project including the advantages and disadvantages of the system and the recommendations for future use. Finally, a summarization of this project was also described. Based on the results of testing that has been conducted, some advantages can be identified in this system. The advantages of CouponCar are as follows:

1) User can easily pay their street parking payment using the apps that have been installed on their smartphone.
2) The system will exactly calculate the time period and substitute the total payment from the user's credit balance.
3) Users use only one application despite being in different states and even different cities.
4) Able to help the officer to check many cars within a short time, whereas they can easily scan the QR code on every car's dashboard.
5) Users can use the same QR code with the same registered account even if they have more than one car.
6) The system can automatically detect the location of the user by using GPS technology.

Based on the results of testing that has been conducted, there are some limitations that can be identified in this system. The limitations of CouponCar are as follows:

1) As this is only a prototype, this project can't implement the real process of online payment that requires a registered company.
2) CouponCar only provides three types of parking zone with their respective price for an hour.

7. Conclusion

CouponCar application proposed to help citizens in Malaysia to pay for their parking with ease. Citizens do not need to buy a parking coupon at the city council or any agents that sell the parking coupon. Instead, the manual system will be replaced by using an android based application where they just need to install the application inside their smartphone, tablet, or any suitable device. CouponCar is able to operate as planned from the first phase of the software development. For future work, it is recommended to make an online payment so users can buy and use the credits to pay using online banking, credit card, or other online payment services. Also, provides a variety of parking zone that have their own price for it.

REFERENCES

[1] Baranovski, I., Stankovski, S., Ostojić, G., & Horvat, S. (2020, March). Support for Self-service Automated Parking Systems. In *2020 19th International Symposium Infoteh-Jahorina (Infoteh)* (pp. 1-5). IEEE.

[2] TJB, D. D., Subramani, A., & Solanki, V. K. (2017). Smart City: IOT Based Prototype for Parking Monitoring and Management System Commanded by Mobile App. *Annals of Computer Science and Information*

DOI:https://doi.org/10.54216/FPA.060205

Systems, *10*, 341-343.
[3] Mosman, A. B. A., Mostafa, S. A., Mustapha, A., Shamsudin, A. U., & Ahmed, M. (2019). An Optimized Method for Automated Stadium Attendance Management System. *International Journal of Engineering & Technology*, *8*(1.7), 112-118.
[4] Mainetti, L., Patrono, L., Stefanizzi, M. L., & Vergallo, R. (2015, December). A Smart Parking System based on IoT protocols and emerging enabling technologies. In 2015 IEEE 2nd World Forum on Internet of Things (WF-IoT) (pp. 764-769). IEEE.
[5] Dennis, A., Wixom, B. H., & Roth, R. M. (2012). Systems Analysis and Design (5th ed.). London: Pearson Education.
[6] Mohd, N. A., Mostafa, S. A., Mustapha, A., Ramli, A. A., Mohammed, M. A., & Kumar, N. M. (2020). Vehicles Counting from Video Stream for Automatic Traffic flow Analysis Systems. *International Journal*, *8*(1.1).
[7] Nelson, G. D., & Bruess, C. E. (1986). Assessment of a worksite safety belt promotion program. *Health education*, *17*(5), 63-67.
[8] Irwin, J. R., & McClelland, G. H. (2002). Heuristiques trompeuses et modèles de régression multiple avec variable modératrice. *Recherche et Applications en Marketing (French Edition)*, *17*(2), 87-101.
[9] JomParking. (2016). Hard to Pay Parking? Retrieved from http://jomparking.com/apps Malaysian Ministry of Transportation. (2008). Transport Statistics Malaysia 2008. Retrieved from http://www.mot.gov.my/en/Official%20Publications/Malaysia%20Transport%20Statistics%20Year%202008.pdf
[10] Mainetti, L., Palano, L., Patrono, L., Stefanizzi, M. L., & Vergallo, R. (2014, September). Integration of RFID and WSN technologies in a Smart Parking System. In *2014 22nd International Conference on Software, Telecommunications and Computer Networks (SoftCOM)* (pp. 104-110). IEEE.
[11] ParkBox. (2015). "It's the Faster Way When You are in Rush". Retrieved from http://www.parkbox.my/
[12] MBJB Spot. (2016). Mobile Parking App. Retrieved from https://mbjbspot.com/ ParkBox. (2015). "It's the Faster Way When You are in Rush". Retrieved from http://www.parkbox.my/.
[13] Kasym, K., Sarsenen, A., Segizbayev, Z., Junuskaliyeva, D., & Ali, M. H. (2018, June). Parking gate control based on mobile application. In *2018 Joint 7th International Conference on Informatics, Electronics & Vision (ICIEV) and 2018 2nd International Conference on Imaging, Vision & Pattern Recognition (icIVPR)* (pp. 399-403). IEEE.
[14] Khalaf, B. A., Mostafa, S. A., Mustapha, A., Ismaila, A., Mahmoud, M. A., Jubaira, M. A., & Hassan, M. H. (2019). A simulation study of syn flood attack in cloud computing environment. *AUS journal*, *26*(1), 188-197.
[15] Evans-Cowley, J. S., & Kubinski, B. (2015). There's an App for That: Mobile Applications That Advance Urban Planning. In *Emerging issues, challenges, and opportunities in urban E-planning* (pp. 33-45). IGI Global.
[16] Kim, H. J., & Modell, J. (2012). Mobile app design tool for smartphones: A tutorial. *International Journal of Soft Computing and Engineering (IJSCE)*, *2*, 237-243.
[17] Mostafa, S. A., Mustapha, A., Shamala, P., Obaid, O. I., & Khalaf, B. A. (2020). Social networking mobile apps framework for organizing and facilitating charitable and voluntary activities in Malaysia. *Bulletin of Electrical Engineering and Informatics*, *9*(2), 827-833.
[18] Mainetti, L., Palano, L., Patrono, L., Stefanizzi, M. L., & Vergallo, R. (2014, September). Integration of RFID and WSN technologies in a Smart Parking System. In *2014 22nd International Conference on Software, Telecommunications and Computer Networks (SoftCOM)* (pp. 104-110). IEEE.

DOI:https://doi.org/10.54216/FPA.060205