T.C. MARMARA UNIVERSITY

FACULTY of ENGINEERING

COMPUTER ENGINEERING DEPARTMENT

CSE4053 Information Systems Analysis & Design

Project #5: Implementation

PARK IT

Ertuğrul SAĞDIÇ - 150116061

Ali ÇETİNKAYA - 150119606

1. System Construction

1.1) Which programming languages(s) have you used for this project?

We used the Flutter framework which uses the Dart programming language to develop our application.

1.2) Which IDEs have you used for this project?

We have used android studio in order to compile and run our written code. We run the application on an android emulator.

1.3)What are the main modules of your program?

We have two modules. One of them is authentication and the other one is the features module. Authentication module includes Sign up module and login module. Features module includes Monitor live density information module, Scan QR code module, and See payments module.

- Sign up module: This module is created in order to sign the new user to the system.
- Login module: This module is created in order to log the user who already signed up to the system.
- Monitor live density module: This module is created in order to see the density of the parking lots.
- Scan QR code module: This module is created in order to check in/out to the parking lot with QR code.
- See payments module: This module is created in order to see the payments history of the customer.

1.4) Which database management system have you used?

We use Firebase as our database system. Due to time Schedule is not flexible, We choose the one free and ready to use database system. Firebase is a NoSQL database where the data is stored as JSON format. Firebase also includes a ready to use authentication system.

1.5) We evaluated our software according to the requirements of the customer and abilities of our team members.

1.6) About hardware we will use customer’s smartphones and Firebase as our cloud server.

2. Complete Test Plan

TEST PLAN

PROGRAM ID: 1	**VERSION NUMBER:** 1.1	**DATE DESIGNED:** 10.06.2021
TESTER: ALİ ÇETİNKAYA / ERTUĞRUL SAĞDIÇ		**DATE CONDUCTED:** 14.06.2021

TEST ID: 1

REQUIREMENT ADDRESSED: Tested parts of application shoud be work properly

OBJECTIVE: TESTING MODULES OF APPLICATION

TEST CASES

INTERFACE ID:	DATA FIELD:	VALUE ENTERED or SHOWED
1.1.1	email, password	"stoneageprogrammer@gmail.com", "123a123"
1.2.1	email, password	"stoneageprogrammer@gmail.com", "123a123"
2.1.1	density, price	"%0", "50"
2.2.1	parking_lot, ent_time, dept_time, tot_price	"A", "10:00","11:00","50,00"
2.3.1	qr_code	"deneme123"

Expected Results and Notes

User should be sign in or sign up to app,
see density of selected parking lots,
see bill details (price, entrance time and leave time and parking lot name.

Results and Notes according to testing

Tests are completed and no errors are encountered.

However, the QR code system is not working right now.

TESTING

i) UNIT TESTING

Black-Box Testing: Potential users are selected and they tested our program according to their expectation by a parking lot app. We agreed that the user experience should be improved. Ho

ii) INTEGRATION TESTING

User Interface and Use Scenario Testing: User interface and use scenario testings are very important for our Project. Because users have very limited time to use a parking lot app and everything should be very clear and understandable. So testing scenarios and user interface is one of the important testing parts of our Project. So we test every scenario and user interface of our application.

Data Flow Testing: In data flow testing the system is calculating density of each parking lot and calculating total price that users should pay. We need to be more careful. Calculating price and calculating density of parking lots are important parts of data flow testing.

System Interface Testing: Data transfer part is one of the important parts of our Project. It should collect data from users and change it to useful datas as customer's spend time in parking lots. When customers are increasing we can test our system more clearly

iii) System testing

System testing includes following parts:

Requirements Testing: This method informs us of unexpected activities that users can do. We can test our application with testers that don't have any information about our application.

Usability Testing: This testing method is for testing Project by Professional analysis that knows user orientations and behaviours and thinking.

Security testing: We need an infrastructure analyst or professional help to test security however we don't have any funding to apply it. However, because we are using a cloud database management system we will use their security system too.

Performance Testing: We can test our system with over user capacity. However due to parking lot capacity is limited we don't have high troubles about performance.

Documentation Testing: We prepared Project proposal, system request, Feasibility, work plan requirements documents. We can test our documents.

iv) Acceptance testing

We apply the Alpha testing method. We control the system with potential users that can use a parking lot application.

Documentation

You can find our all documentations below:

System Documentation:

1. **Project Proposal:** Project proposal is a letter that says what we want to do.

2. **System request:** Explaining business model of our Project. Requests and values are explained on that paper.

3. **Feasibility:** Feasibility is a paper that compares projects in real and in paper.

4. **Work Plan:**

5. **System Proposal:** Explaining all Project to customer and explaining advantages of our Project.

User Documentation:

1. **User Manual:** We are planning to prepare an user manual

2. **Helpdesk:** We will implement a helpdesk on our application that includes helpful information about application and contacts of developers to ask questions.

Store and Tracking

We use github to store our app. Github allows us to work remotely on one data. We use google drive, google calendar to follow our deadlines, follow our works. We create an excel table to arrange task sharing.

SYSTEM INSTALLATION

- **b) Prepare a conversion strategy/plan according to the following issues.**

  i) Style: direct, parallel.

  Our conversion strategy will be parallel conversion. The parking lot customers will check in/out with our system and also with the old system until the users will get used to the application.

- **ii) Locations: pilot, phased, simultaneous.**

  Since the system allows the customers to check in/out to all parking lots, it will be better to apply simultaneous conversion. Since the customers can park in the parking lot with the old system and new system, it will not be a problem to add all parking lots to our system.

- **iii) Modules: whole-system, module by module.**

We are going to install the whole system. Of course there are modules in the application. However, in our case, it will be wiser to use the whole system instead of module by module because the modules are dependent on each other at some point. For example we can not see the density of the parking lot without the Scan QR code module which will basically change the information on the database of the specific parking lot.

-

- **c) Why have you selected that conversion strategy? Explain in terms of costs, benefits, and risks.**

  We will change the whole system and apply a new system. Old system will not be available when our system is finished. So this conversion strategy is suitable for the kind change that we will apply.

- **d) Prepare a training plan for the system.**
- Training plan will be include of our application. Due to the old ticketing system being very complex and not having visible statistics, training wouldn't be hard.

SYSTEM MAINTENANCE

b) Give information about your support plan.

We will make a contract with the customer. According to the contract we will give support 7/24 or less according to the budget of the customer. First year limited support is included in our project.

c) How do you maintain the system? How do you report the problems/bugs?

We are using the Google play store and App store.

In these kinds of mobile application stores, users can easily report the problems and bugs. Also, the report module will be implemented in order to have this kind of report of the problems and bugs in our hand immediately.

d) Do you use support / ticketing system? Explain briefly.

We will implement a report menu inside of the application. Thanks to that feature, users can easily contact us and give feedback about their experiences and problems.

e) How do you handle project assessment and change requests?

We will make a contract with customers about change requests and assessments. According to the contract we will meet change requests and project assessment demands. Also we will provide 1 year limited support and also full support for bugs and problems.

Post Implementation Review

POST IMPLEMENTATION REVIEW	
DATE	15.06.2021
PROJECT NAME	PARK IT
PROJECT SPONSOR	CUSTOMER
PROJECT MANAGER	MUSTAFA AGAOGLU
PLANNED END DATE	17.06.2021
ACTUAL END DATE	17.07.2021
PLANNED BUDGET	10.000$
PROJECT SUMMARY	The Park It app is prepared for a company that wants to increase their income. Park it app includes calculating the staying payment of customers, density of each parking lot that company have. With the QR code system of the Park It app company will not use tickets that are very good for the environment. With the density of parking lots, customers can go to parking lots that have low density.