

Payment Management Software with input from Smartphone Camera

Group: 19

Wasan Prongsirikul	55070503441	joprongsirikul@gmail.com
Tanakit Mahasrisangpate	55070503418	garto_ekarus@hotmail.com
Sukkajorn Chaitraguntong	55070503450	tuysahaha@hotmail.com

Advisor: Assoc. Prof. Dr. Thumrongrat Amornraksa

I have read this report and approve its content.

Abstract (English)

Presently, in Thailand, there are apartments or mansions growing up every day. Types of apartment can be classified into many levels, ranging from luxury to low level. The rental price and facilities are normally depended on the level of apartment. Most of them belong to low-to-mid level. Thus, this project focuses on real problem solving for low-to-mid level apartments.

A practical problem found on these apartments is a management system which is not usually well developed. Most of them rely on paper-based and manual system. It lacks of implementing up to date technology on the system, and this may lead to a fraud.

In this project which classified as a “Potential Commercial Product”, a mobile application is developed as a product for apartment management system. In particular, we replace the old fashion management system with an up to date mobile application technology which requires less number of apartment staff for operations. The developed application helps the staff to manage tenants, payment system, and give a visual summary in total to apartment owner.

In the payment system, an image processing technology called “OCR” or “Optical Character Recognition” can be used for rental payment calculation. The OCR module developed in this project will recognize number characters in the image, captured by a mobile phone camera, and output the result as numbers directly for the calculation. The input image is taken from numbers appeared on water and electricity meters.

Abstract (Thai)

ณ ปัจจุบันประเทศไทยได้มีการสร้างอพาร์ทเมนต์หรือหอพักชนิดต่างๆ โดยที่จำนวนของอพาร์ทเมนต์เหล่านี้ก็ได้มีการเพิ่มขึ้นในทุกๆวัน อพาร์ทเมนต์ ณ ปัจจุบันนั้นมีความหลากหลายทั้งความหรูหราและสิ่งอำนวยความสะดวกต่างๆ ซึ่งราคาก็แตกต่างกันไปแต่ส่วนใหญ่จะเป็นระดับกลางลงไป โดยในโปรเจกต์นี้เราให้ความสนใจในการแก้ปัญหาของอพาร์ทเมนต์ที่อยู่ในระดับกลางลงไปเป็นหลัก

ปัญหาที่เราพบคือระบบการจัดการหอพักต่างๆไม่มีการพัฒนาให้ดี ผู้ประกอบการยังคงยึดติดกับระบบการจัดการแบบเก่าไม่ได้มีการนำเทคโนโลยีในปัจจุบันเข้ามาช่วยเหลือทำให้ความปลอดภัยของระบบเสื่อมถอยและมีช่องโหว่ของการทุจริตเกิดขึ้น

โปรเจกต์นี้จัดอยู่ในโปรเจกต์สำหรับ“ผลิตภัณฑ์ในเชิงพาณิชย์ที่มีศักยภาพ”เราได้ทำการพัฒนาผลิตภัณฑ์เป็นแอปพลิเคชันบนมือถือสำหรับการจัดการและดูแลระบบของหอพัก เราได้ทำการแทนที่ระบบแบบเก่าด้วยระบบที่มีความทันสมัยที่มีการใช้จำนวนบุคลากรในการจัดการน้อยกว่า แอปพลิเคชันนี้จะช่วยเหลือบุคลากรในการจัดการระบบส่วนของผู้เช่า จัดการระบบการชำระเงิน และสามารถช่วยเหลือเจ้าของหอพักให้มีวิสัยทัศน์ในการจัดการหอพักได้ง่ายขึ้นเนื่องจากเราจะมีการสรุปข้อมูลต่างๆให้ทันที

ในส่วนของระบบชำระเงินได้มีการนำเทคโนโลยีสำคัญคือเทคโนโลยีการจัดการกับรูปภาพแบบ “ไอซีอาร์” คือการแปลงข้อมูลภายในรูปภาพจากการโทรศัพท์มือถือให้กลายเป็นข้อมูลอักขระดิจิทัล ในที่นี้เราจะแปลงรูปภาพมิเตอร์น้ำ-ไฟให้กลายเป็นตัวเลขเพื่อทำการคำนวณกับค่าชำระโดยตรง และสรุปค่าใช้จ่ายออกมาเพื่อส่งไปยังผู้เช่า

Acknowledgements

First, we would like to show our gratitude to Assoc. Prof. Dr. Thumrongrat Amornraksa, our advisor, who sharing his pearls of wisdom with us during the course of this research. We also grateful to all lecturers along this college for well instructed knowledge. For, my senior friends, we are thankful for their helps. Finally, we are impressed with our family for their kind co-operation and encouragement which help us in completion of this project

Table of Contents

Chapter 1.....	3
1.1 Problem Statement and Approach.....	3
1.2 Objectives.....	3
1.3 Scope	4
1.4 Tasks and Schedule.....	5
Chapter 2.....	6
2.1 HTTP Protocol.....	6
2.2 Cordova	6
2.3 OCR (Optical Character Recognition)	7
2.4 jQuery.....	8
2.5 jQuery Mobile	8
2.6 Node.js	9
2.7 WAMP server	10
Chapter 3.....	11
3.1 Feature List.....	11
3.2 Architectural diagram.....	12
3.3 Use case diagram.....	13
3.4 User interface screen layouts.....	16
3.4.1 Mobile application UI	16
3.4.2 Web page (Dashboard).....	20
3.5 Navigation map and site map	22
3.6 Database schema	23
Chapter 4.....	26
Chapter 5.....	29

List of Figures

Topic	Page
Figure 1: Architectural diagram	12
Figure 2 Use case diagrams: describing the action each user type is able to do in the application.	13
Figure 3: Owner home page.....	16
Figure 4: Manage apartment option.....	16
Figure 5: Manage staff option.....	17
Figure 6: Staff's home page.....	17
Figure 7: Manage room option	18
Figure 8: Manage tenant option	18
Figure 9: Payable status page.....	19
Figure 10: Payment confirm page	19
Figure 11: Save meter page.....	19
Figure 12: Tenant's home page	20
Figure 13: Tenant's payable status page	20
Figure 14: Complaint page	20
Figure 15: Dashboard	21
Figure 16: Navigation site map	23
Figure 17 ER diagram: describe database relations	24
Figure 18: OCR success result	27
Figure 19: OCR unsuccess result	27

Chapter 1

Introduction

1.1 Problem Statement and Approach

In Thailand, there are a lot of apartments everywhere with different levels; from luxury apartment to normal apartment. Most of the normal to low level apartments are not have well prepared payment management system. Having one person, a staff, responsible for maintenance and rental charge is very common. The staff will take notes of all utility meter (water and electricity) then calculate with other cost. This process is vulnerable. There is a chance that human-error will be occurred. Furthermore, this process cannot examine that the utility unit has been changed by the staff intently, or not. The fraud is happened frequently because of this unexaminable situation.

We have a solution to improve the system by using a mobile application. The application will replace the notes taking and calculation process. Instead of taking notes with a paper, we capture photos of the meter. From help of the optical character recognition (OCR), the application will receive the utility unit from the transformed image. The received number will be stored and used in bill calculation.

As a result, this approach can reduce the system vulnerability compared to the old system. The human-error is exterminated since the software will do the calculation. The error becomes examinable that the staff intends to defraud, or not.

1.2 Objectives

The objective of this project is to develop an apartment management system by including technology into the system. Since, our target user is the owner and staff of normal to low level apartment, the application is designed with user friendly concept. We keep the application to be convenient to use. However, the application remains reliable and gives better performance of managing process.

1.3 Scope

This project is based on mobile application platform. We start project with android operating system. This project is apartment management application. The application consists of three main parts: room management, account management, and payment management.

Room management is the part where the structures of apartment are managed. The structures of apartment are apartment name, room, and cost. Managing includes create, edit, and remove function.

Account management is the part where user account is registered. Users in this application are classified into 3 types:

- Owner
- Staff
- Tenant

These three types of user are doing different rows. Owner is the person who has permission to create and manage their apartment and staff. Staff is the person who is responsible for managing room, payment, and tenant. Tenant is the customer in this service by renting the room.

Payment management is the part where the payment procedure is managed. The OCR module is implemented in this part. The bill calculation is done automatically. Furthermore, payment status and history is observable.

1.4 Tasks and Schedule

Term1	SEP	OCT	NOV	DEC	
Proposal					
Proposal presentation					
Background					
Procedure of collecting utility cost in apartment					
Content in the common report					
Study programs to be used					
Study image processing (OCR)					
Design 1					
Architecture Design					
Project overview					
System flowchart					
Database Design					
Input/Output data					
Relate database together					
Query					
Interface design					
User Interface					
Report & Presentation 1					
Report 1					
Edit Report 1					
Report 2					
Edit Report 2					
Presentation 1					
Term2	JAN	FEB	MAR	APR	MAY
Design 2					
Interface design					
User Interface					
Mock up and site navigation					
Subsystem flowchart					
Implementation 1					
Create Database					
Create web application					
Coding web application					
OCR module					
Testing 2					
Overall modules					
Debugging					
Implementation 2					
Security system					
Testing 3					
Debugging 3					
Report & Presentation 2					
Report 1					
Edit Report 1					
Report 2					
Edit Report 2					
Report 3					
Edit Report 3					
Presentation 2					

Chapter 2

Background, Theory and Related Research

2.1 HTTP Protocol

HTTP or Hypertext Transfer Protocol is set of rules for accessing World Wide Web (WWW) to exchange information from web browser [1]. Now, when accessing to the WWW, we do not need to include “http” in front of the Uniform Resource Locator (URL) anymore because it is already a standard method for communication.

The concept of HTTP protocol is that there is a HTTP client, the one who access via web browser, and a web server, a hardware device serving a requested file upon request. When user send request to web server, whether entering the URL or click on the hypertext link, the browser makes a HTTP request and send to the Internet Protocol address (IP address) that is indicated by URL. The destination server will locate the requested file(s) and send back to the client [2].

HTTP protocol is a stateless protocol. The command is executed independently meaning that having no information about what occurred previously. A new web page is processed without any knowledge of the previous page requested [3].

2.2 Cordova

Apache Cordova is a mobile development framework that allows standard web technologies to cross-platform development. This is the best way to develop mobile application for multiple platforms from HTML, CSS and JavaScript without re-implement it with each platform’s language and tools. Applications execute within wrappers targeted to each platform, and rely on standards-compliant API bindings to access each device's capabilities such as sensors, camera, data, etc. [4]

Cordova Service Platform:

Apple iOS

BlackBerry

Google Android

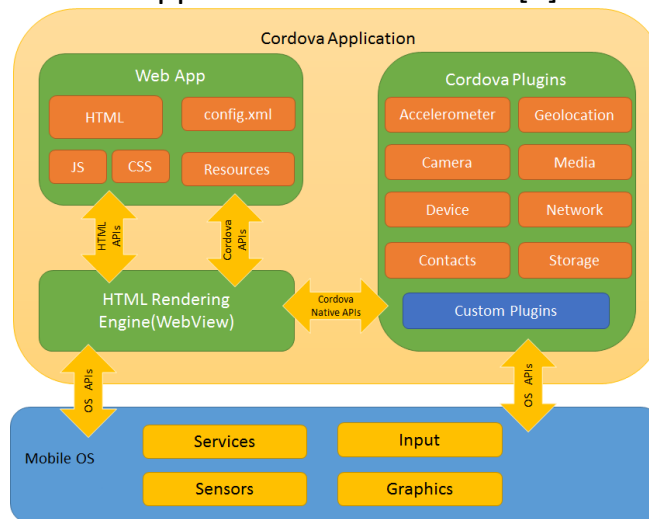
Microsoft Windows Phone

Touch

Bada
Firefox OS
LG webOS
Nokia Symbian OS
Tizen
Ubuntu

Architecture:

There are several components to a Cordova application. The following diagram shows a high-level view of the Cordova application architecture. [4]



Retrieved from: <https://cordova.apache.org/docs/en/latest/guide/overview/index.html> [Accessed: 03- May- 2016]

2.3 OCR (Optical Character Recognition)

Optical Character Recognition is one of the image processing techniques allowing you to transform different types of documents, such as scanned paper document, PDF files, or digital photo into editable and searchable data [5].

OCR is a combination of detect and recognize algorithm. It uses edge detection for searching and wanted object and then trying to recognize it. During the process, image will be enhanced with various type of filtering so that it is easier for the machine to understand.

2.4 jQuery



jQuery is a warehouse of JavaScript library. The purpose of jQuery is to make it easier to implement the web site with JavaScript. It wraps those lots of common task that require many lines of code into one simple line of code [6]. jQuery simplifies HTML document traversing, event handling, animating, and Ajax interactions for rapid web development [7].

The Core features [8]:

1. Access elements in a document.
2. Modify the appearance of a web page.
3. Alter the content of a document.
4. Respond to a user's interaction.
5. Animate changes being made to a document.
6. Retrieve information from a server without refreshing a page.
7. Simplify common JavaScript tasks.

2.5 jQuery Mobile



jQuery Mobile (JQM) is an addition of jQuery technology. Since, devices that we used now are having so many types with different resolution and functionality. These problems will be solved with JQM by creating a responsive web design. It is a framework for user interface experience based on jQuery technology that works across all phones, tablet, or even desktop platform. [9]

The JQuery Mobile is compatible with the following by-default browsers:

iOS	Symbian S60
Symbian UIQ	Symbian Platform
BlackBerry OS	Android
Window Mobile	webOS
Bada	Maemo
MeeGo	

Main Features [10]

jQuery Mobile started in August 2010 as a modern framework, including many patterns and best practices for multiplatform development. The main features of the framework are:

- Cross platform, cross device, and cross browser
- UI optimized for touch devices
- Theme able and customizable design -Usage of nonintrusive semantic HTML5 code only, without the need of any Java- Script, CSS, or API knowledge
- AJAX calls automatically to load dynamic content
- Built on the well-known and supported jQuery core
- Lightweight size, 12Kb compressed -Progressive enhancement
- Accessibility support

2.6 Node.js

Node.js is a server-side runtime environment platform for developing web applications. It is open-source software written in JavaScript built by Google and can be run on multiple platforms: OS X, Windows, Linux. [11] Node.js is fast processing that make the number of application that work with node.js is increasing that include the application that will help to develop [12]

Features [11]:

Asynchronous and Event Driven - All APIs of Node.js libraries are asynchronous that is, non-blocking. It essentially means a Node.js based server never waits for an API to return data. The server moves to the next API after calling it and a notification mechanism of Events of Node.js helps the server to get a response from the previous API call.

Very Fast - Being built on Google Chrome's V8 JavaScript Engine, Node.js library is very fast in code execution.

Single threaded but highly scalable - Node.js uses a single threaded model with event looping. Event mechanism helps the server to respond in a non-blocking way and makes the server highly scalable as opposed to traditional servers which create limited threads to handle requests. Node.js uses a single threaded program and the same program can provide service to a much larger number of requests than traditional servers like Apache HTTP Server.

No buffering - Node.js applications never buffer any data. These applications simply output the data in chunks.

License - Node.js is released under the MIT license.

2.7 WAMP server

WAMP server is a Windows web development environment. It is including program that use to create a website. It will help about configuration that normally need to install each one and hard to configuration that may cause problem to waste time. By using WAMP, it can create a web application and test on the virtual server environment provided by WAMP. [13] [14]

WAMP included:

Apache – working as webserver

PHP – Coding language for create website

MySQL – managing database

PHPmyadmin – program created with PHP for easily to working with MySQL via web browser

Chapter 3

Design and Methodology

3.1 Feature List

Since, our approach is to develop the old payment management process in normal to low level apartment. We changed the way to manage apartment by using smartphone.

From recording meter unit with paper, we use a phone camera to take a photo of the meter and processed with OCR module. We provide complaint feature so that it is easier for tenant to report problem. Payment tracking can help staff to acknowledge the payable status of each tenant. History tracking is for tenant who wants to know his own history. Dashboard concludes data for owner.

Features	Descriptions
Apartment management	<ul style="list-style-type: none">• Create and manage apartment structure (staff, rooms, residents, etc.)• Set costs (room cost, central fee, electricity cost, water cost)
Electricity and Water meter unit processing	<ul style="list-style-type: none">• Taking meter photos and its unit.• OCR (Optical Character Recognition)• Recognize number in the photo then transform it to digital number. Store it in database for bill calculation
Complaint	<ul style="list-style-type: none">• Report technical broken or other problem
Dashboard	<ul style="list-style-type: none">• Conclude information of all active apartments
Payable status tracking	<ul style="list-style-type: none">• View payable status
History Tracking	<ul style="list-style-type: none">• Show payment history

Table 1: Features list

3.2 Architectural diagram

This diagram shows what components were used and communicated with the application. It also shows what data was transferring between modules.

There are 3 types of user: Owner, Staff, and Tenant. Each user type can do different tasks. Owner is responsible for setting up apartment and staff. Staff is responsible for setting up room, tenant, payment, and complaint. Tenant can observe status and history. In addition, tenant can report problem through the application.

Dashboard and Print bill is for owner which will access via web page. Facebook module represents the Facebook page.

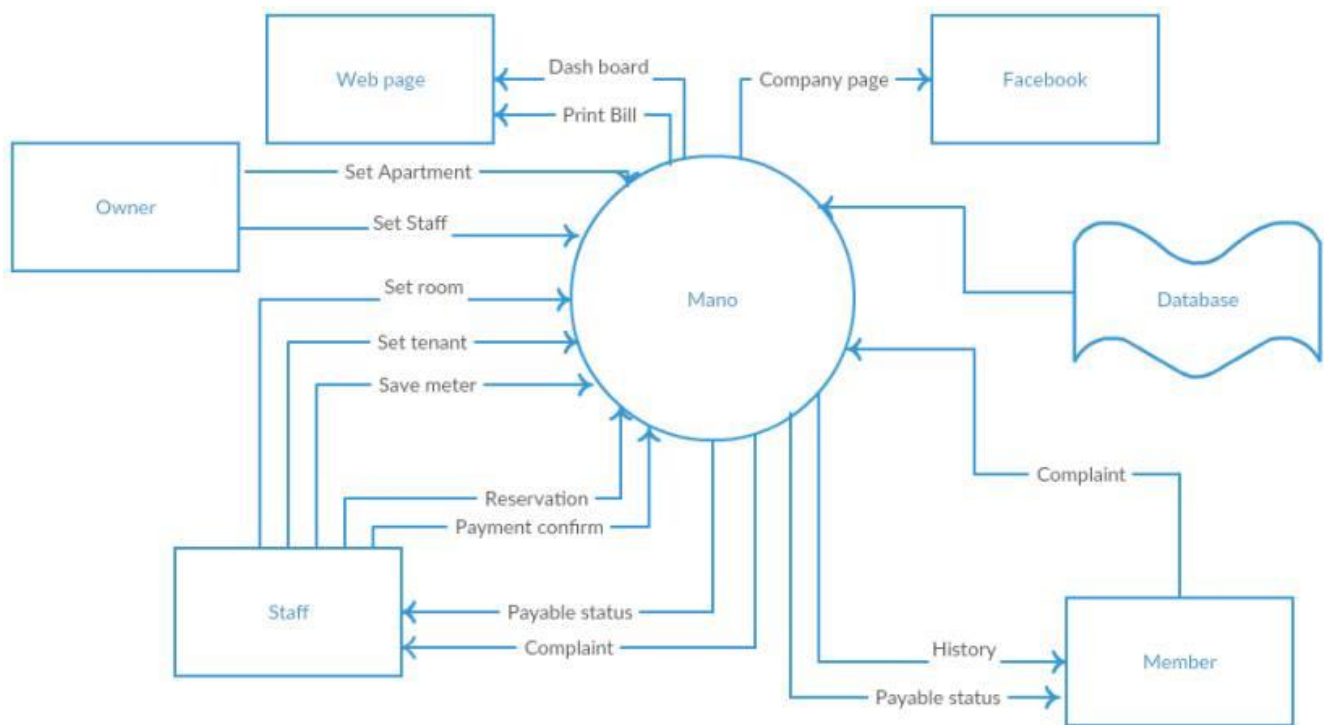


Figure 1: Architectural diagram

3.3 Use case diagram

Use case diagram describe the usability of each user type. The project has 3 types of user: Owner, Staff, and Tenant. Each user is able to do different tasks according to their type. This diagram displays actions each user type can do. The descriptions of each action are provided below.

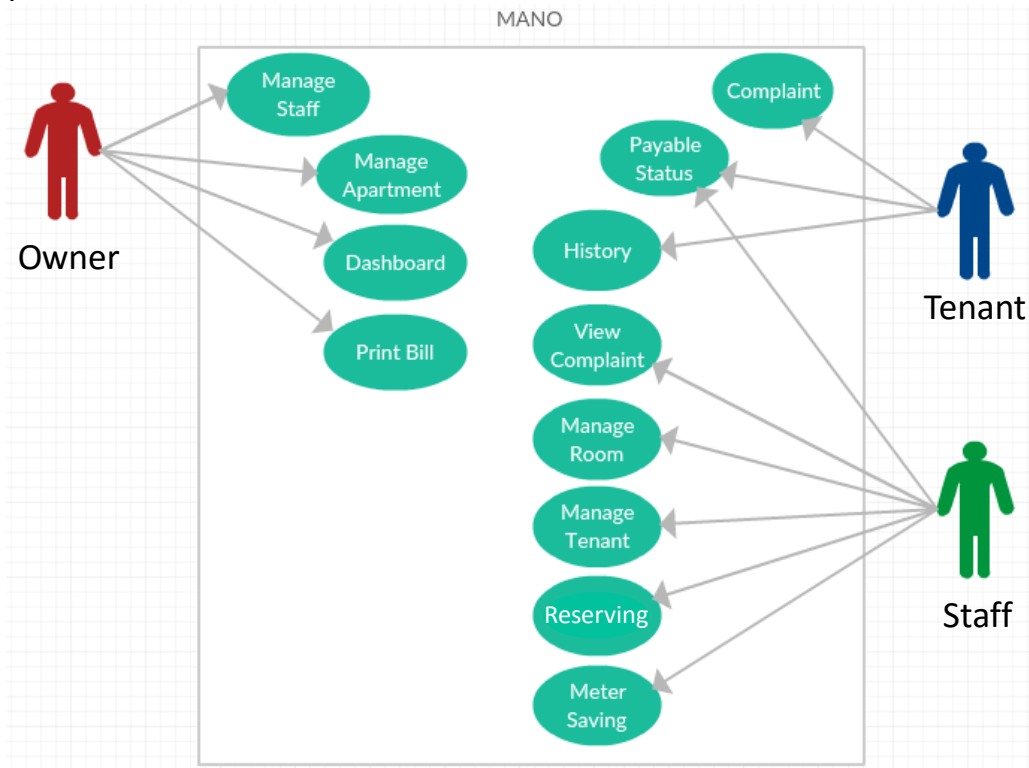


Figure 2 Use case diagrams: describing the action each user type is able to do in the application.

3.3.1 Owner

Owner is responsible for managing the apartments and staffs. There are four actions owner can do. First, manage apartment, owner must create their apartment into the system. Second, manage staff, owner need to create staff and assign the apartment each staff will be in-charged. Third, dashboard, owner can observe status of their apartments with dashboard. Dashboard concludes data from all active apartments that the owner created. Last, print bill, at the end of the month, owner is able to print bill for tenants.

Cases	Descriptions
Manage apartment	<ul style="list-style-type: none"> • Create apartment and set the central fee. • Edit apartment information.
Manage Staff	<ul style="list-style-type: none"> • Create staff account • Assign apartment that staff will be in-charge • Edit staff account
Dashboard	<ul style="list-style-type: none"> • View dashboard
Print bill	<ul style="list-style-type: none"> • Print bills

Table 2: Owner use case action

3.3.2 Staff

Staff is responsible for managing rooms, tenants, and payment in the assigned apartment. There are six actions staff can do. First, manage room, staff must create room and set the cost of each room. Second, manage tenant, staff will create tenant account when tenant book the room. In addition, booking the room has 2 stages: reserving and instant booking. The second step is an instant booking. Third, reserving, when staff agree with tenant who want to reserve the specific room before doing instant booking, staff can reserve the room for tenant. Forth, payable status, staff can observe the payable status of all tenants. Fifth, meter saving, this feature is used when staff take a picture of the utility meter at the end of the month. The picture is processed with OCR module after being captured. Last, view complaint, staff is responsible for reviewing complaint from tenants.

Cases	Descriptions
Manage room	<ul style="list-style-type: none"> • Add rooms and set room cost • Edit room status (FULL/EMPTY/RESERVE) and cost

Manage tenant	<ul style="list-style-type: none"> • Create tenant account • Assign room for tenant • Edit tenant information
Reserving	<ul style="list-style-type: none"> • Reserve the room
Payable status	<ul style="list-style-type: none"> • View all tenant's payable status if they paid or not
Meter saving	<ul style="list-style-type: none"> • Take photo of meters • Record electricity unit and water unit for calculation
View complaint	<ul style="list-style-type: none"> • View complaint from tenant

Table 3: Staff use case action

3.3.3 Tenant

Tenant is the customer of this service. There are three actions tenant can do. First, payable status, tenant can view their own payable status each month. Second, history, tenant can view recorded payment that had been paid. Last, complaint, tenant can complain any problem to staff.

Cases	Descriptions
Payable status	<ul style="list-style-type: none"> • View their payable status
History	<ul style="list-style-type: none"> • View payment history
Complaint	<ul style="list-style-type: none"> • Report technical broken or other problem

Table 4: Tenant use case action

3.4 User interface screen layouts

3.4.1 Mobile application UI

3.4.1.1 Owner

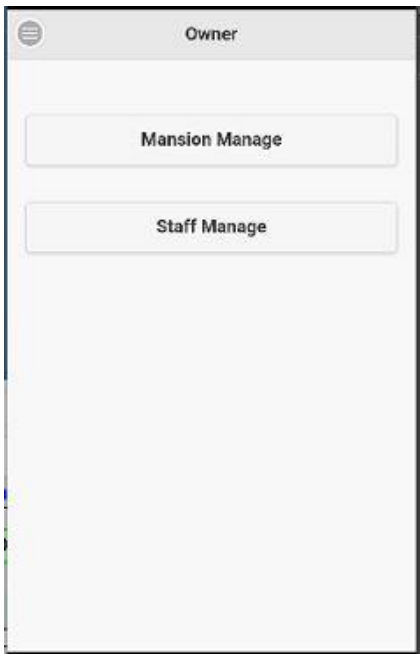


Figure 3: Owner home page

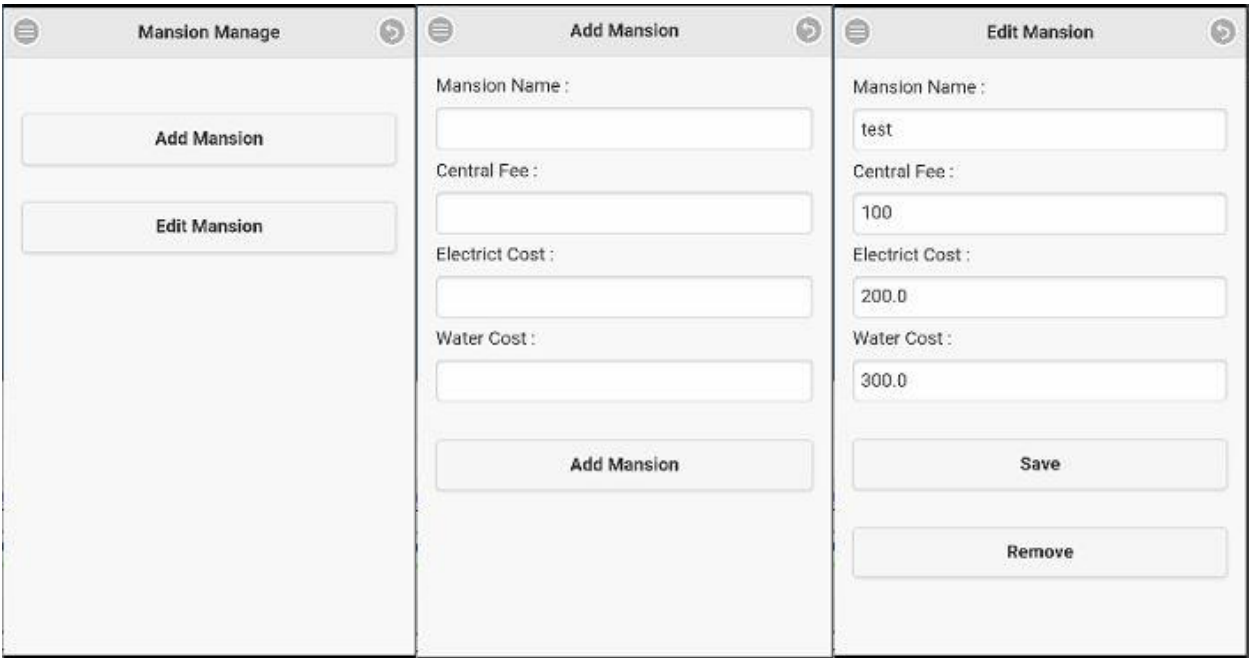


Figure 4: Manage apartment option

The figure displays three mobile application screens for staff management:

- Staff Manage:** Features two buttons, "Add Staff" and "Edit Staff", for navigating to the respective functions.
- Add Staff:** A form for adding a new staff member with fields for Name, Last Name, User Name, Password, and a Mansion Select dropdown (currently showing "Ap1"). It includes an "Add Staff" button at the bottom.
- Edit Staff:** A form for editing an existing staff member. Fields include Name (pre-filled with "Wasan"), Last Name (pre-filled with "Prongsirikul"), User Name (pre-filled with "staff01"), Password (pre-filled with "staff01"), and Mansion Name (pre-filled with "test"). It includes "Save" and "Remove" buttons at the bottom.

Figure 5: Manage staff option

3.4.1.2 Staff

The figure shows a mobile application screen titled "Staff" with a list of management options:

- Room Manage
- Tenant Manage
- Payment Status
- Payment Confirm
- Save Meter
- Reservation

Figure 6: Staff' s home page

Room Manage	Add Room	Edit Room
<div>Add Room</div> <div>Edit Room</div>	Room No. : <input type="text"/> Price : <input type="text"/> <div>Add Staff</div>	Room No. : <input type="text" value="001"/> Price : <input type="text" value="5000"/> Electric Meter : <input type="text" value="0"/> Water Meter : <input type="text" value="0"/> Status : <div> <div>Empty</div> <div>Full</div> <div>Reserv</div> </div> <div>Save</div> <div>Remove</div>

Figure 7: Manage room option

Tenant Manage	Add Tenant	Edit Tenant
<div>Add Tenant</div> <div>Edit tenant</div>	Name : <input type="text"/> Last Name : <input type="text"/> User Name : <input type="text"/> Password : <input type="text"/> Room No. : <input type="text"/> <div>Add Tenant</div>	Name : <input type="text" value="test"/> Last Name : <input type="text" value="test"/> User Name : <input type="text" value="test"/> Password : <input type="text" value="testpw"/> Room No. : <input type="text" value="011"/> <div>Save</div> <div>Remove</div>

Figure 8: Manage tenant option

หมายเลขห้อง	วันที่	สถานะ	วันที่ชำระ
staff6	2016-02-25	YES	2016-03-17

Figure 9: Payable status page

Bill ID :

Search

Figure 10: Payment confirm page

Room No. :

Electric Meter :

Water Meter :

Save

Electric Meter

Water Meter

Figure 11: Save meter page

3.4.1.3 Tenant



Figure 12: Tenant's home page



Figure 13: Tenant's payable status page



Figure 14: Complaint page

3.4.2 Web page (Dashboard)

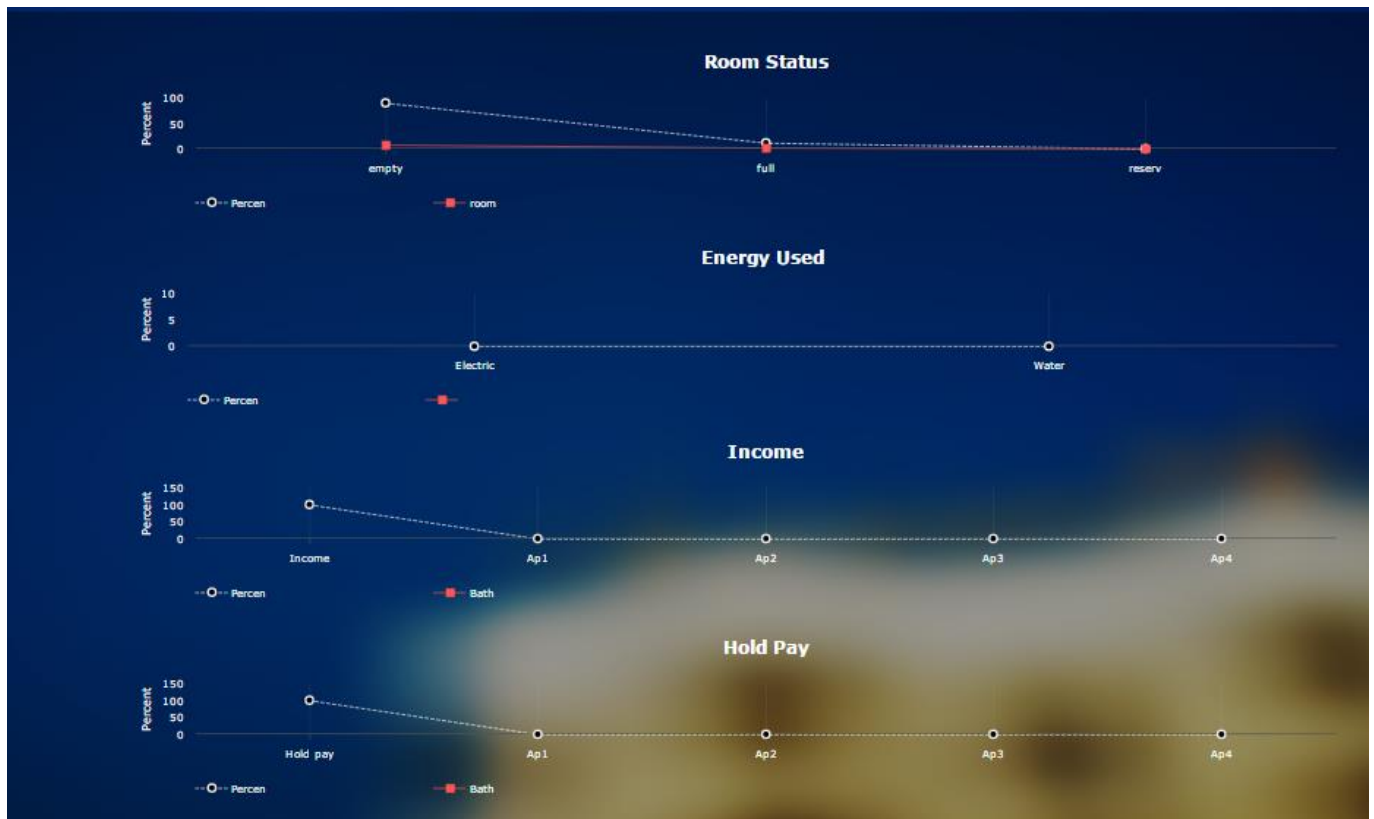


Figure 15: Dashboard

3.5 Navigation map and site map

Navigation map shows overview of how user accesses the application. First, after logged-in, the application will locate to the home page of each user type.

Owner has only 2 menus in the mobile application: Manage apartment and Manage staff. In manage apartment, owner can add a new apartment or edit the existed apartment. Same as Manage apartment, manage staff also has add and edit option. In addition, owner has to assign staff into an apartment that staff will be in-charged. Dashboard and Print bill page have to access through the web browser.

Staff has several menus for working in the application. Manage room, as same as Manage apartment, is used to add or edit room. Manage tenant, as same as Manage staff, is used to book the room for tenant and add, or edit tenant account. Payment status is for checking payable status of all tenants. Payment will be confirmed with Payment confirm. Save meter used when taking photo of the meter at the end of the month. Reservation menu is for reserving the room.

Tenant has only three menus. Check payable status notify tenant for unpaid payable. Payment history is used to view paid payment. Report and Complaint is for complaining problems.

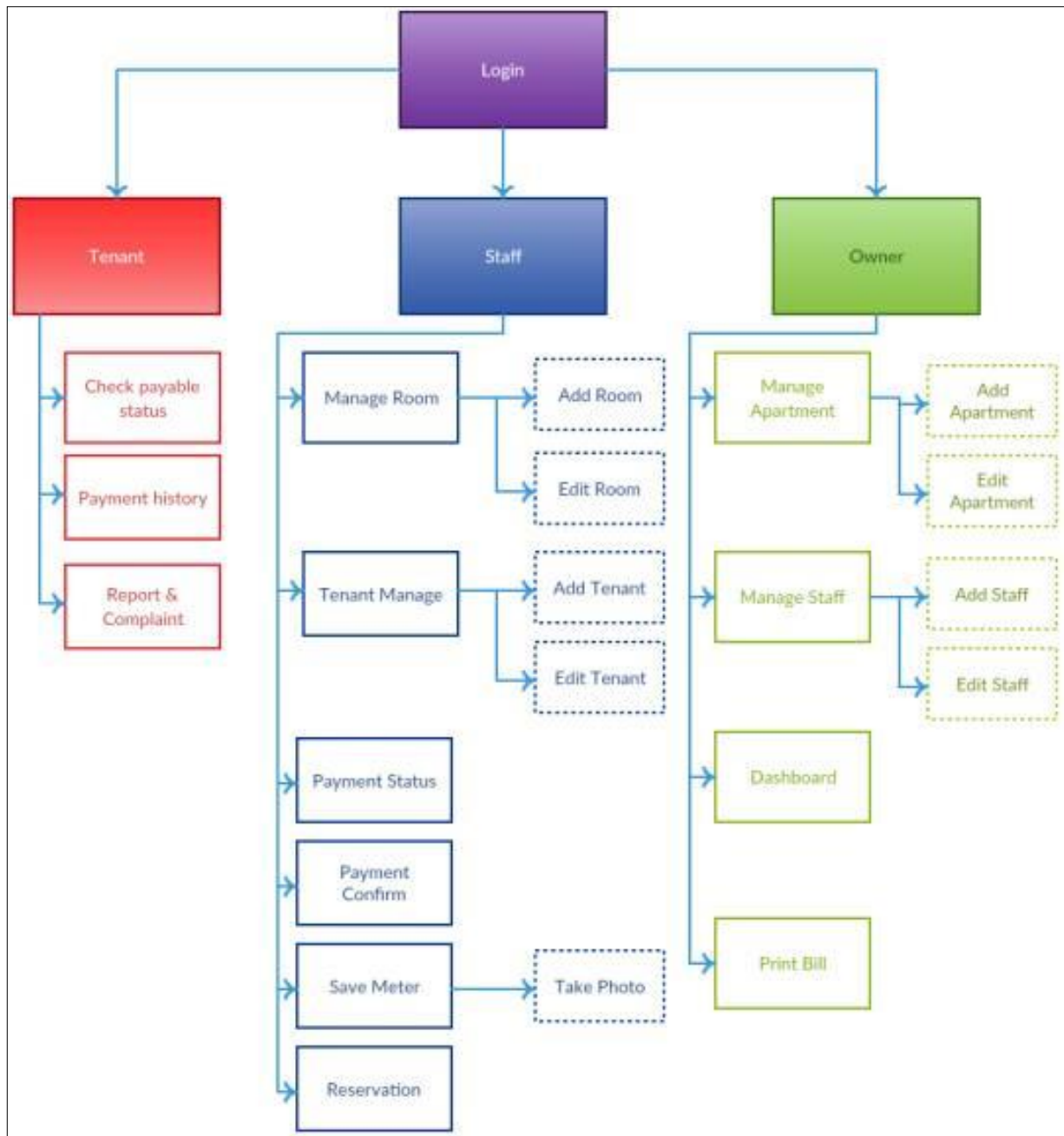


Figure 16: Navigation site map

3.6 Database schema

In order to store data from the application, database is required. This diagram presents how data in the application stored within database, and show how modules relate to each other.

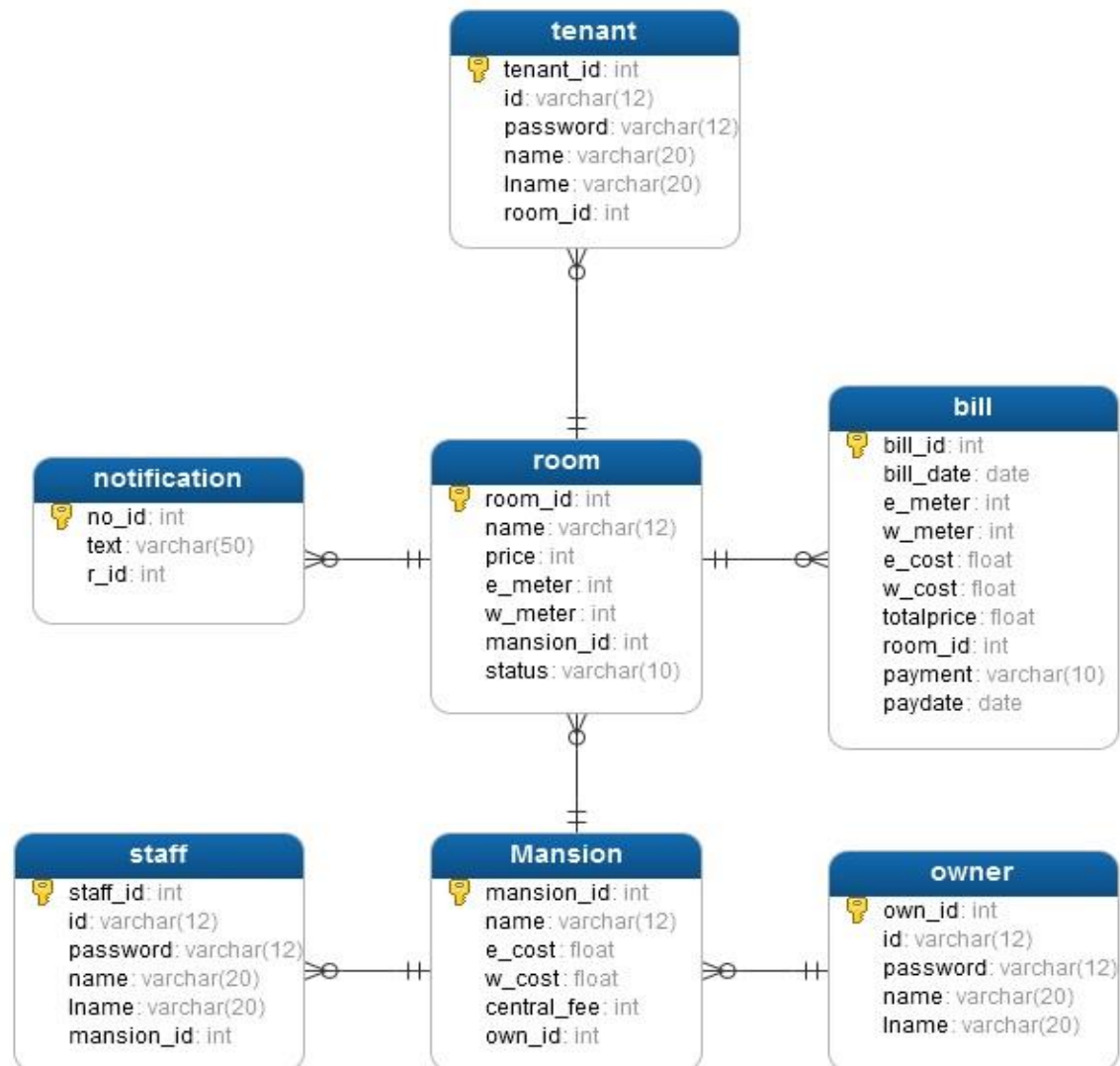


Figure 17 ER diagram: describe database relations

Tenant table – this table contain tenant account and its information. It also recorded the room id the tenant stayed.

Staff table – this table contain staff account and its information. It also recorded the mansion the staff stayed.

Owner table – this table contain owner account and its information.

Mansion table – this table store mansion details and owner id for identify the owner.

Room table – this table store the information of the room.

Bill table – this table contains all bill details.

Notification table – is the table store the complaint from the tenant.

Chapter 4

Results and Discussion

Now, the application is finished. Each feature is already implemented and ready to use. In this chapter, we will present the result of our design and implementation including problem of the requested design.

Multiple apartment assigned problem

Allowing one staff into multiple apartments could cause some problem. If one staff can be in different apartments, then the 3 staff account limitation is pointless. The owner will be able to use the same staff account to manage several apartments instead. So, we decided not to let one staff can be in multiple apartment but only one staff per apartment.

Room cost structure

Providing several room types for different cost to support different apartment might not be very effective. Since, there is no standard for different cost of each room. One apartment could have more than five, six, or even more different cost for each room depended on the number of floors, view from the room, location, size, etc. Providing not enough room type might cause problem. So, we decided to let the user to set the cost for each room freely instead.

OCR module

During saving meter unit procedure, OCR module is executed. It will process the image into different form and enhancing the image to be more machine friendly. This mean the color will be changed. The edge will be sharpening. So, the algorithm can find the character in the image.

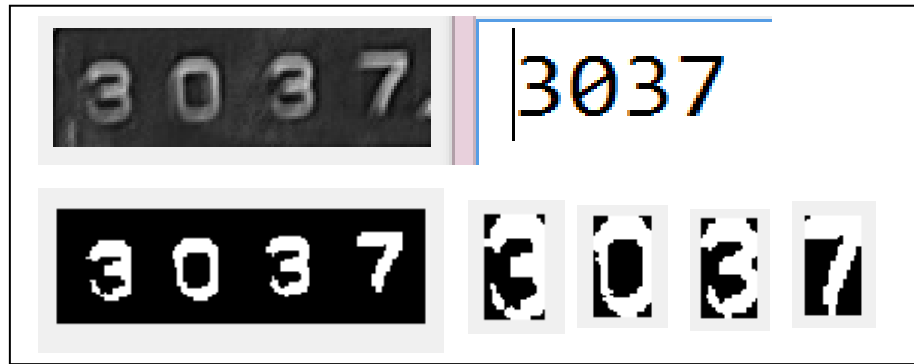


Figure 18: OCR success result

From figure 18, it is shows that OCR module works effectively. Its accuracy is around 80%. The top-left picture is original image. The bottom-down image is processed image. The image is changed to binary form and the edge of the character had been sharp. The result is shown on the top-right



Figure 19: OCR unsuccessful result

From figure 19, it is shows that the number is transformed correctly but there is an extra “0” in front of the line. After trying to solve the cause, we discovered that the “0” the module got is the full image. The module understands that the full image is “0” because there is a reflection surrounded the numbers. This make the OCR module misunderstood. However, this can be solved by selecting the correct region of interest (ROI). So, selecting proper region of interest is also important.

Dashboard

From figure 15, dashboard can conclude all data. We conclude data in to four fields: Room status, Energy used, Total income, and Hold pay.

Room status shows the room availability of each apartment. Energy used tells the unit of both water and electricity used. Total income concludes all income received from all apartments. Hold pay displays the number of tenants who does not paid the rent.

Chapter 5

Conclusion

In order to help normal to low levels apartment to have more effective apartment management, we have designed a mobile application to serve the user need. The application is made to be user friendly. So, it is simple and easy to use and compatible for all users.

Our application can solve the vulnerability of the old system and exterminated the human-error. Furthermore, it is also reduced the time used in calculation process since the application calculate the bill automatically. The OCR module works perfectly. Even though some design had been changed, the result remains the same.

At last, our project is already finished. The project is met the requirement. In case of developing in the future, we looking forward to make the application will be able to do the financial payment with bank, and having more feature that will make the application more interesting.

References

- [1] "HTTP" computerhope.com, (n.d.).
Retrieved from: <http://www.computerhope.com/jargon/h/http.htm>
[Accessed: 20- Jan- 2016]
- [2] Rouse, Margaret "HTTP (Hypertext Transfer Protocol) definition"
searchwindevelopment.techtarget.com, July 2006.
Retrieved from: <http://searchwindevelopment.techtarget.com/definition/HTTP>
[Accessed: 20- Jan- 2016]
- [3] Beal, Vangie "HTTP - Hypertext Transfer Protocol" webopedia.com, (n.d.)
Retrieved from: <http://www.webopedia.com/TERM/H/HTTP.html>
[Accessed: 20- Jan- 2016]
- [4] "Overview" cordova.apache.org, (n.d.)
Retrieved from: <https://cordova.apache.org/docs/en/latest/guide/overview/index.html>
[Accessed: 27- Jan- 2016]
- [5] "What is OCR and OCR Technology" abbyy.com, (n.d.)
Retrieved from: <http://www.abbyy.com/finereader/about-ocr/what-is-ocr/>
[Accessed: 05- Feb- 2016]
- [6] "jQuery Introduction" w3shcool.com, (n.d.)
Retrieved from: http://www.w3schools.com/jquery/jquery_intro.asp
[Accessed: 05- Feb- 2016]
- [7] "What is jQuery?" tutorialspoint.com, (n.d.)
Retrieved from: <http://www.tutorialspoint.com/jquery/jquery-overview.htm>
[Accessed: 24- Feb- 2016]
- [8] Chaffer, J., Swendberg, K., 2006, " Better Interaction Design and Web Development With Simple JavaScript Techniques", Learning jQuery 1.3, p5-18.
[Accessed: 24- Feb- 2016]

- [9] "Introduction" jquerymobile.com, (n.d.)
Retrieved from: <http://demos.jquerymobile.com/1.4.5/intro/>
[Accessed: 05- Mar- 2016]
- [10] Firtman, M. ,2012, "jQuery Mobile: Up and Running", jQuery Mobile, p 1-24
- [11] "Node.js – Introduction" tutorialspoint.com, (n.d.)
Retrieved from: http://www.tutorialspoint.com/nodejs/nodejs_introduction.htm
[Accessed: 05- Mar- 2016]
- [12] SURANART NIAMCOME, 'Node.js', July 06, 2014,
Retrieved from: <http://www.siamhtml.com/introduction-to-node-js/>
[Accessed: 03- May- 2016]
- [13] Bourdon, Romain "WAMP server" wampserver.com, (n.d.)
Retrieved from: <http://www.wampserver.com/en>
[Accessed: 03- May- 2016]
- [14] 'WAMP server', Sep 28, 2016,
Retrieved from: <http://facebook-sdk.blogspot.com/2015/09/wamp.html>
[Accessed: 04- May- 2016]