Individual Project Report

Web-based Management System for Hair Salon

By: Isabel Tiong Shu Kie (G1578672N)

Supervisor: Bobby Chung

Submission Date: 17th June 2018

Abstract

Management plays an important role as can boost the entire operational efficiency in any business enterprise or company. The managing is referring here can be time management, staff management or sales management and other different areas. It makes sure that the company is well in place and does not get into insufficient trouble easily.

Although we knew that managing can be done easily nowadays by using POS system or other business managing software. There are still quite some companies that has not implement them and still doing manual managing.

For this final year project, I will be helping a small medium enterprise that works as a hair salon, to help them to switch from manual managing to automate managing. Proposed solutions were made for them based on the problem they are facing and their business perspective.

All the details of this process are written in this project report, from the beginning when we get to know the enterprise itself to the end when we delivered the final product for them.

Table of figure

Table of content

1.0 Introduction

Building webpage or web-based application has always been a huge interest for me that wished to learn more about. This task for the final year project was actually assigned by the supervisor, Mr. Bob, in charge, since I have no idea what to do at the beginning. When Mr. Bob briefly talk about this project, I was actually very happy that I can get a chance to actually build an application on my own.

Even so, my skill of programming has yet not considered as good nor compatible but somehow, I was able to learn and implemented in during the process. Also, this is the first time to work on a project that is integrating with real world, which I find troublesome but acknowledging new stuff along the way.

This project has been the hardest assignment I have ever done which technically used most PHP and HTML codes. Since I am only familiar with Java, I spend most of the development time learning these two languages. Working with the cooperating enterprise and the lecturers has also became a great experience and a very tough process for me as I have difficulty talking to people whom I am not familiar with.

2.0 About the cooperating enterprise

2.1.1 User stories

Figure 1 G2 by Govin logo

G2 by Govin is a hair salon located in Far East Plaza, where it competes with 15 others more hair salon in that single building. The current cooperating hair salon is actually their second outlet, first one located in Johor Bahru. G2 by Govin has around 7 staffs excluding the manager in charge, Mr. Tomo whom is the client of this project. Currently, G2 by Govin is using an information system that they paid monthly for management, which has caused series of issues to it.

2.1.2 Challenges

The issues that G2 by Govin facing for the operational efficiency is having **too much paperwork**.

When the information system is yet implemented, G2 by Govin recorded their payment, appointment timetable and other reports manually in papers. Causing a huge pile of papers which is hard to referred to when needed, also it is hard to organised with paper files.

Since it is done manually, human mistakes are lightly to happen, such as forgetting to record or writing the wrong information. Also, there is no appointment timetable, the staffs have to record their appointment in their own calendar after checking with the paper-based appointment list at the counter.

When G2 by Govin implement the information system, they paid it monthly but still it does not solve the issues above.

The reason why is it so is because the information system itself is an off the shelf package. It has way **too many useless features** and has a **low usability.**

2.1.3 Future direction

As for the hair salon future business direction, the CEO of G2 by Govin is planning to **establish his third outlet in Singapore in the next 3 years**. When they do so, managing the inventory and the sales report of all three outlets will be a hard work, if they are still using paper works.

Also, the CEO hope to **enhance communication between the outlets** for a better planning of resources, whether it for sales or used in the salon.

So, the strategy for G2 by Govin is to replace the current information system with a basic, easy-to-use management system.

That is why the manager contact the supervisor and then assigned the task for this project.

2.1.4 Proposed Solution

After acknowledging the issues and future direction of G2 Govin, three solutions are proposed during the discussion.

- Solution 1: Make a new management system, only focusing on the appointment system and some simple managing features, this may have no advanced features that they hope, but will be delivered on time.
- Solution 2: Directly make a new management with every implementation that they requested to have but the time will have to be much more than the expected time they assigned.
- Solution 3: Buy an existing open source POS system which has a lower cost of the current information system, then refactor the code to fit G2 by Govin structure.

Giving some times for the manager and the staffs to discuss, they come to end of choosing solution 1 due to its simplicity, low cost and because their need for a better appointment system is the highest. So, this project is started based on solution 1 at the beginning, focusing on the appointment systems.

3.0 Literature review

Agile methodology (used in)-

Agile software development is a methodology that is used the manage the software engineering process. Agile contains different approach such as Scrum, the Crystal Methods, Lean Methodology and many more.

Most methods of Agile uses iterations, which divided the timeline into tiny pieces, usually around 1~3weeks time, in order to keep track of the user requirement and to boost the software quality. In these tiny timelines are the software features, which include the progress of designing, building, testing and refactoring. And during the end of each iteration, a meeting should be made with the client, user, supervisor or team members, to check the result.

Agile methodology focused more on integrating with the team members by communicating and also satisfying the clients by attending frequent stand-up meeting. The clients are likely to be satisfied when using the Agile Methodology, as changes can be make instantly and the final result will be in high quality. For team member, stand-up meeting is also required, to make sure all members are still in track in the progress and checking whether they occurred in any difficulties.

Most of the time, Agile methodology are used in group, where there are the programmers, technical writers, testers and scrum master, whom was in charge to keep an eye of everyone’s progress.

But Agile methodology can actually be used in solo too. The purpose for using Agile is to do sprint cycle during the development, which can also be done by solo developers. The main reason that people assume that Agile is not meant for solo is because there are some Agile approaches that cannot be done alone, such as Extreme programming which requires pair coding, but if there are no such approaches, Agile methodology can still be used in solo software engineering.

ERD diagram (used in)-

ERD basically stands for Entity Relationship diagram, it shows the data and the integration, which can also mean relationship, between one another in a database. Inside the ERD diagram, data with primary key or foreign key will also be shown. This is to have a clearer image of the linking through the databases.

The progress will be much easier when building the linking between databases while referring to the Entity Relationship diagram.

Data dictionary-

Data Definition Matrix, which is also referring to the Data dictionary mentioned here. Data dictionary provides the detailed data structure of the tables in the database. This includes the name, type, defining whether it is primary key or foreign key, checking if it is NOT NULL and other parts of the data structure. To simplify it, data dictionary is a table of the details of the ERD diagram's data. Normally, ERD diagram is used for business concept and the data dictionary are used by the software engineered while programming.

Data dictionary is a tool that allows the technical team to communicate to the business clients more efficiently in order to built a better relational database to fulfil the user requirements.

Gantt chart-

A Gantt chart is a visual view of tasks that needs to be carried out for the development. Since Agile methodology is implemented, the table of the Gantt chart will be separated according to sprint, which inside contains features that are need to be delivered within that sprint. Also containing the most important information which is the expected start date, end date and duration of days that is needed to finish that feature.

Gantt chart helps the developer to check whether he himself is on same timeline that is in the chart. This is because the expectation date that is written in the Gantt chart is the ideal timeline of development. So, by using Gantt chart in time management, there is less chance for the developer to be unable to deliver the product due to time consuming.

Technical parts

PHP-

PHP is a scripting language and is one of the languages that works the best for building web application, which also known as server side. PHP mainly is to develop web application or shell application with command line. PHP is a programming language to develop dynamic website, you can use it as an interface to connect to the database. It takes the information from HTML, turns it into forms and helps to make connections.

HTML-

HTML stands for Hypertext Markup Language, it is used for making the static pages for webpage. Which means it does not do any dynamic functions. HTML main focus is to design and built the template of the webpage, also by implementing some css files.

JavaScript-

JavaScript is also a scripting language like PHP, but unlike PHP, JavaScript can built functions and used APIs to make advanced features. JavaScript used by adding it inside the HTML files using <scripts> or you can just key in the link to the code of the scripts.

4.0 Requirement

This session will be elaborating the functional and non-functional requirement for this project. Especially for functional requirement, this is needed to implement into the Moscow method for the focus of the development.

4.1.1 Functional

Based on the solution we chose for G2 by Govin, there are several functional requirements after the discussion, here these requirements are sorted by their priority, so that we can arrange the workflow.

The **MUST HAVE** requirements.

1. Security system- Only staff is able to access to the system, so a login page must be required. No registration is available, if new user need to be added, it can only be added directly through the user database.
2. The features button should be shown on the header of the page.
3. The entire page layout design must be simple and easy to understand.
4. A working appointment schedule system.
5. A working payment system.
6. List that shows inventory and staff information.

So, the features that is mentioned above will be the main focused when building the application. Other requirement below will still be made if time allows.

The **COULD HAVE** requirements

1. CRUD of staff information and inventory.
2. Separate the selling items and the services.
3. Automate generated sales report after payment.
4. Obvious colours for appointment timetable.

4.1.2 Non-functional

The non-functional requirements are referring to the technical part of this project, which are the hardware and software that are needed to make the web-based management application.

Hardware:

Platform- Window 10

Server- Apache 2.4.33

Web browser capable- Internet Explorer, Google Chrome, Firefox, Safari

Software:

Programming language- PHP 7.0, JavaScript, HTML

6.0 Software design

6.1.1 Architecture design

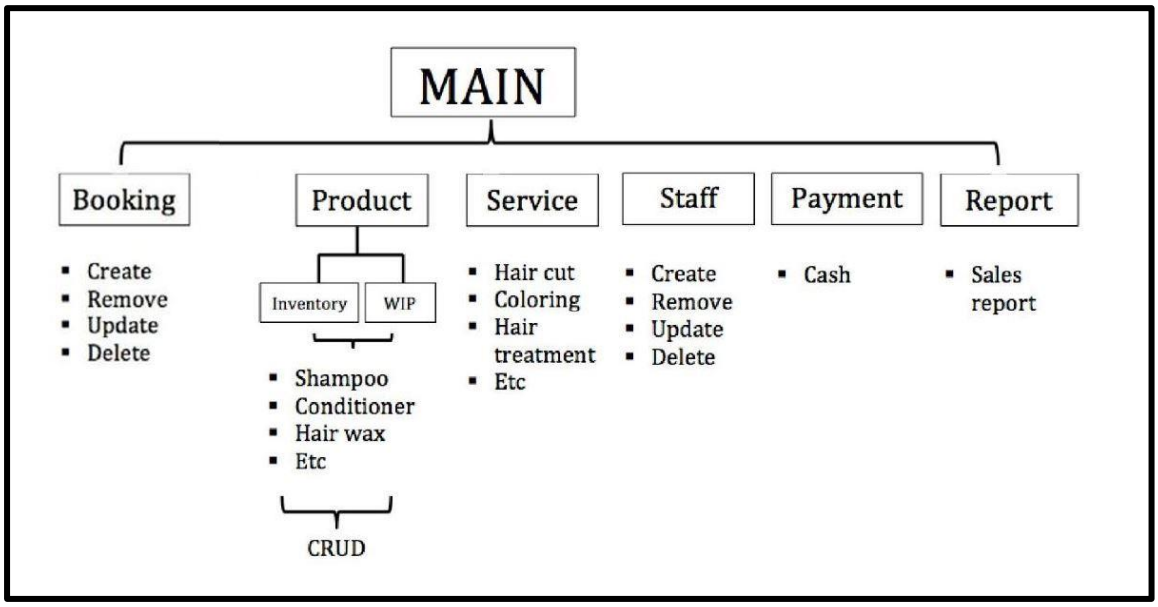

Figure 2 modular design

The above figure shows the modular design of the application. The product and services are actually under the same features, Sales. The 'inventory' and 'WIP' under Product means whether it use for service purposes or being sold as an item. The modular design figure above will be the architecture blueprint for the application.

6.1.2 Data design

6.1.2.1.1 ERD diagram

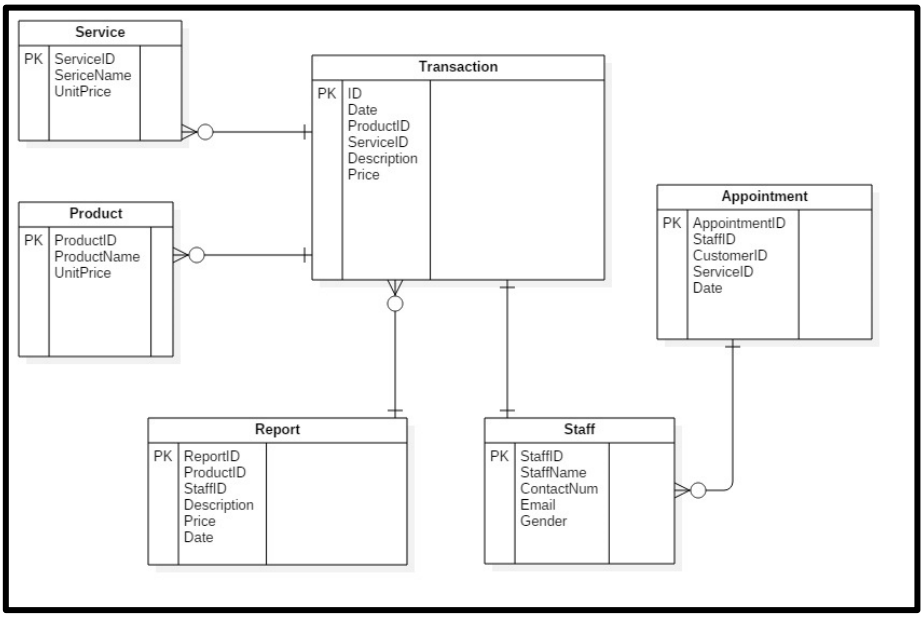

Figure 3 ERD diagram

The figure above shows the ERD diagram of the management application. As shown, most tables all connected with the transaction table in the middle. Which means that the transaction, which is the payment system, will be the feature considering the most complicated database linking among other features. Although the primary key in each table are labelled as 'PK', the foreign key however will not be shown in the ERD diagram. For data structure referring, please check on the data dictionary below.

6.1.2.1.2 Data dictionary

Entity	Attribute	Data Type	PK	FK	Unique	NOT NULL	Description
Transaction	ID	INTEGER(10)	YES	NO	YES	YES	A unique code to represent each transaction
	Date	DATE	NO	NO	NO	YES	Date of transaction made
	ProductID	INTEGER(10)	NO	YES	YES	YES	A unique code to represent each product
	ServiceID	INTEGER(10)	NO	YES	YES	YES	A unique code to represent each service
	Description	VARCHAR(255)	NO	NO	NO	YES	Description for transaction if needed
	Price	VARCHAR(8)	NO	NO	NO	YES	Price of the total transaction

Entity	Attribute	Data Type	PK	FK	Unique	NOT NULL	Description
Appointment	AppointmentID	INTEGER(10)	YES	NO	YES	YES	A unique code to represent each appointment
	StaffID	INTEGER(10)	NO	YES	YES	YES	A unique code to represent each staff
	CustomerID	INTEGER(10)	NO	YES	YES	YES	A unique code to represent each customer
	ServiceID	INTEGER(10)	NO	YES	YES	YES	A unique ode to represent each service
	Date	DATE	NO	NO	YES	YES	Date of appointment

Entity	Attribute	Data Type	PK	FK	Unique	NOT NULL	Description
Report	ReportID	INTEGER(10)	YES	NO	YES	YES	A unique code to represent each report
	ProductID	INTEGER(10)	NO	YES	YES	YES	A unique code to represent each product
	StaffID	INTEGER(10)	NO	YES	YES	YES	A unique code to represent each staff
	Description	VARCHAR(255)	NO	NO	NO	YES	Description of report
	Price	VARCHAR(8)	NO	YES	YES	YES	Price of the total transaction
	Date	DATE	NO	YES	YES	YES	Date of transaction made

Entity	Attribute	Data Type	PK	FK	Unique	NOT NULL	Description
Staff	StaffID	INTEGER(10)	YES	NO	YES	YES	A unique code to represent each staff
	StaffName	VARCHAR(20)	NO	NO	YES	YES	Name of staff
	ContactNum	VARCHAR(20)	NO	NO	NO	YES	Staff contact number
	Email	VARCHAR(255)	NO	NO	NO	YES	Staff email address
	Gender	VARCHAR(6)	NO	NO	NO	YES	Staff gender

Entity	Attribute	Data Type	PK	FK	Unique	NOT NULL	Description
Service	ServiceID	INTEGER(10)	YES	NO	YES	YES	A unique code to represent each service
	ServiceName	VARCHAR(255)	NO	NO	YES	YES	Name of service
	UnitPrice	VARCHAR(20)	NO	NO	NO	YES	Service unit price

Entity	Attribute	Data Type	PK	FK	Unique	NOT NULL	Description
Product	ProductID	INTEGER(10)	YES	NO	YES	YES	A unique code to represent each product
	ProductName	VARCHAR(255)	NO	NO	YES	YES	Name of product
	UnitPrice	VARCHAR(20)	NO	NO	NO	YES	Product unit price

6.1.2.1.3 Use case diagram

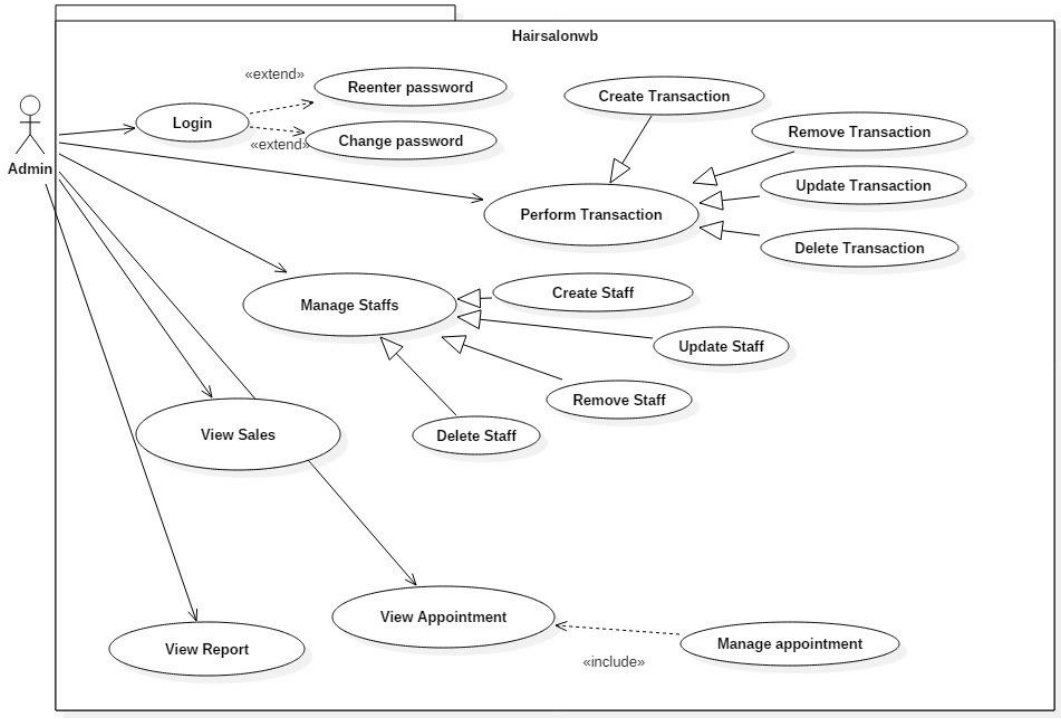

Figure 4 use case diagram

The figure above shows the use case diagram for this management application. As shown, most of the use cases are CRUD, which means create, remove, update and delete, but some features such as view reports is not able to do any changes with the data.

6.1.3 Prototype

Before coding has started, simple prototype images of the application have been drawn. This is made so that it will make the progress easier, because referring can be done when building the HTML template.

Here is some drawing of the application features before programming:

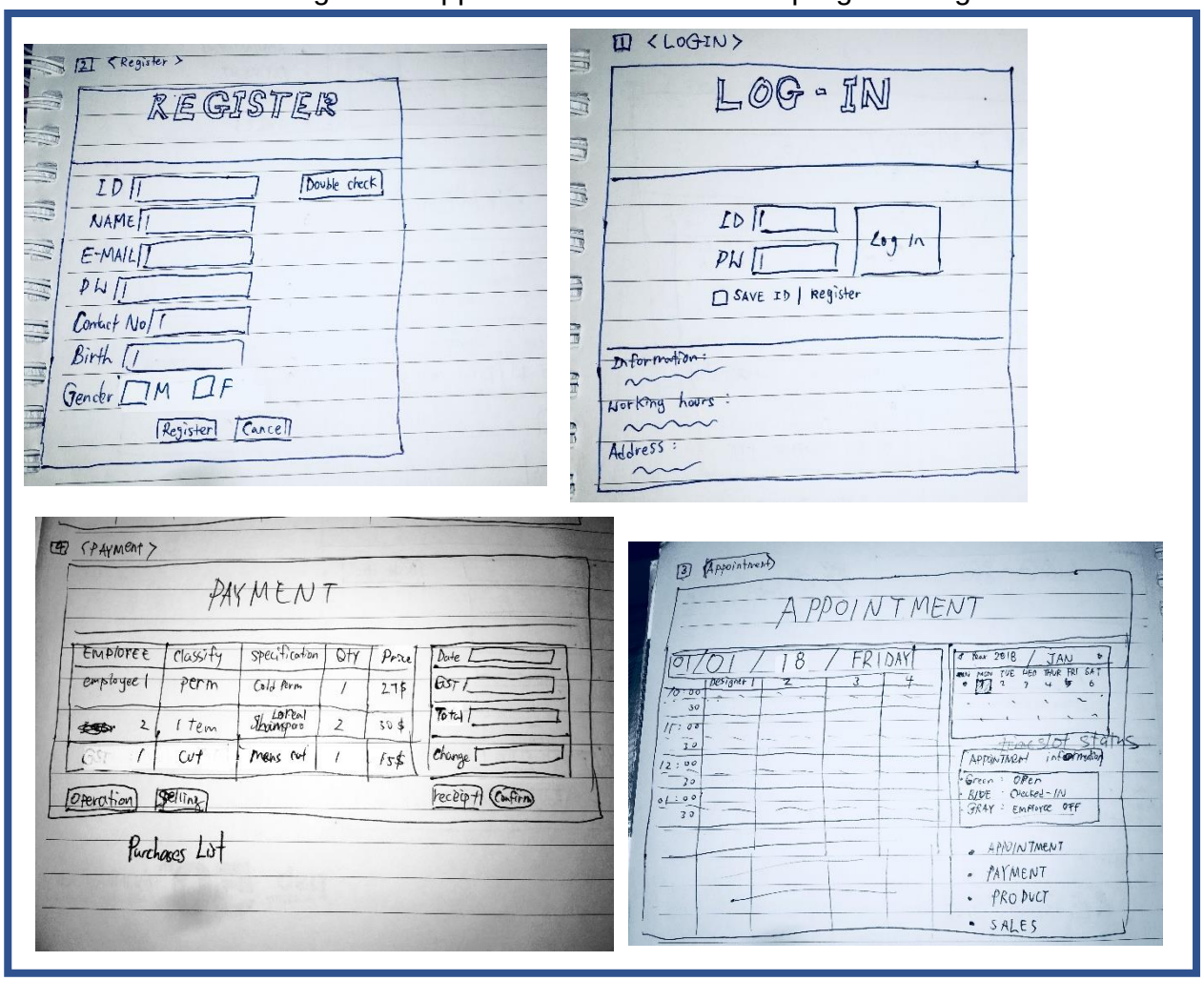

Figure 5 prototype drawing

see the prototype, not only that they want to see the drawing but also the static HTML pages, which is not delivered when I am asked for prototype during the meet up.

7.0 Software quality assurance

7.1.1 Unit test

Software testing is required in order to check the functionality of the application. First of all, **unit testing** is carried out to test if every feature is working perfectly.

First unit: Managing appointment schedule

No.	Test case	Expected result	Pass/ Fail
1	User click on ‘Schedule’ button on the header	Directed to the appointment schedule page	Pass
2	User click on left right button	Left will show the previous month, right will show the next month	Pass
3	User click on the date	Input fields for appointment details and time will show up	Pass
4	User click on save changes in input table	Appointment date duration will be saved and create an event tag, shown in the timetable	Pass
5	User click on the event tag	Input tables that allowed user to edit the details or time will show up. Also showing the delete option	Pass
6	User click on the delete option	Appointment data will be deleted	Pass
7	User click on save changes	Appointment information will be updated	Pass
8	User click on ‘week’ button	Timetable will be shown in weekly format	Pass
9	User click on ‘day’ button	Timetable will be shown in daily format	Pass

Second unit: Admin login logout

No.	Test case	Expected result	Pass/ Fail
1	Admin login with correct combination of username and password	Page will login to welcome page	Pass
2	Admin login with incorrect combination of username and password	Error message will popup considering the wrong combination of username and password	Pass
3	Admin press logout button	Page will jump back to the login page	Pass

Third unit: Welcome page

No.	Test case	Expected result	Pass/ Fail
1	User click on ‘Get Started’ button	Directed to next page	Pass

Forth unit: Products information handling

No	Test case	Expected result	Pass/ Fail
1	User click on ‘Product Overview’	User can view current product information list	Pass
2	User click on ‘Add Product’	User can add a new product with his or her information	Pass
3	User enter “service” as type	“service” typed name will be shown on the service list in ‘Product Overview’	Pass
4	User click on the ‘Edit’ button in the ‘Product Overview’	User can edit the information of the staff	Pass
5	User click on the ‘Delete’ button in the ‘Product Overview’	A confirmation for deletion will popup	Pass
6	User click on ‘confirm’ in delete pop up window	Product information will be deleted	Pass

Fifth unit: Staffs information handling

No	Test case	Expected result	Pass/ Fail
1	User click on ‘Staff Overview’	User can view current staff information list	Pass
2	User click on ‘Add Staff’	User can add a new staff with his or her information	Pass
3	User click on the ‘Edit’ button in the ‘Staff Overview’	User can edit the information of the staff	Pass
4	User click on the ‘Delete’ button in the ‘Staff Overview’	A confirmation for deletion will popup	Pass
5	User click on ‘confirm’ in delete pop up window	Staff information will be deleted	Pass

Sixth unit: Payment making

No.	Test case	Expected result	Pass/ Fail
1	User click on ‘Payment’	Page with table of product, service and order will shown	Pass
2	User enter quantity number and press ‘Add’	The information of the product will be added to the order list	Pass
3	User add same product that had already existed in the order list	A popup saying that the item has already been added will appeared and the item will not be added to the order list	Pass
4	User add a lot of different products	Order list will count the total price	Pass
5	User press ‘Remove’ in order list	Product will be removed from the order list	Pass
6	User press ‘Confirm’ in order list	Information of payment and current date will be saved to the sales report	Pass

Seventh unit: Sales report

No	Test case	Expected result	Pass/ Fail
1	User click on the ‘Report’ button	Directed to page with sales report	Pass

7.1.2 Version control

In this project, GitHub has been used in order to keep track of the evolution and development for the coding. The branches are categories into features part so that I can focused in just only one in each branch. After checking the features are working, they are merged to the master branch. But still there are some issues with the header file while merging as different features has different contents for their header file, so the header file is modified in the master branch itself.

Here is the GitHub link for the application:
https://github.com/isabeltiongsk/hairsalon

8.0 Project Management

8.1.1 Project Methodology

The management methodology that is used for this project is the Agile Methodology as I think it is best method that fits.

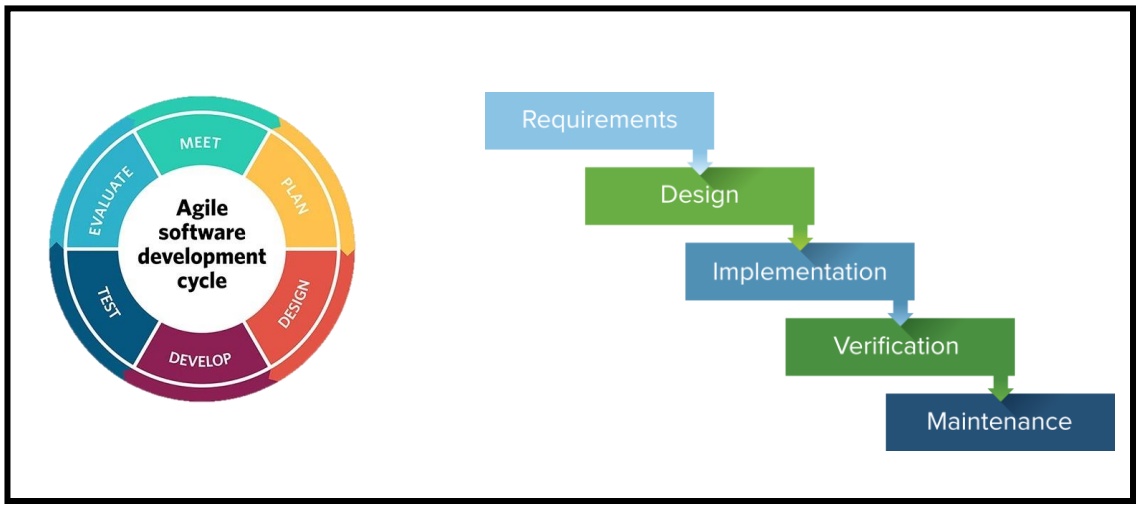

Figure 6 Agile(left) Waterfall(right)

Comparing Agile(left) with other methodology such as Waterfall(right), Agile can shows a series of close updates of the product. The client will be updated with changes or progress in every meeting that has been carried out. Unlike Waterfall where there are no meetings with the client during the entire progress.

By updating with the manager of G2 by Govin, I can be sure that I am doing the correct feature for them and also be able to listen to any of their new suggestions or idea and make changes. Also, in my opinion, since I am working with a small medium enterprise, which is not hard to handle, so I mind as well fulfil their satisfaction as much as possible.

8.1.2 Project time management

For time managing, Gantt chart is used for a reminder and also a time tracker.

Sprint		Task	Start	Duration	End	Status
1			1st Feb 2018	14days	14th Feb 2018	In progress
		Admin login	2/1/2018	6	2/6/2018	Done
		Welcome Page	2/7/2018	2	2/8/2018	Done
		Page Template	2/9/2018	6	2/14/2018	Done
2			23th Feb 2018	53days	16th Apr 2018	Not Started
		Timetable part one	2/23/2018	20	3/14/2018	In progress
		Timetable part two	3/15/2018	33	4/16/2018	Not Started
3			17th Apr 2018	14days	30th Apr 2018	Not Started
		Staff overview	4/17/2018	4	4/20/2018	Not Started
		Staff CRUD	4/21/2018	3	4/23/2018	Not Started
		Product overview	4/24/2018	4	4/27/2018	Not Started
		Product CRUD	4/28/2018	3	4/30/2018	Not Started
4			1st May 2018	14days	14th May 2018	Not Started
		Payment system	5/1/2018	10	5/10/2018	Not Started
		Sales Report	5/11/2018	4	5/14/2018	Not Started

Figure 7 Gantt chart table

The Gantt chart above was screenshotted when the first sprint was done, so that is why the other status are ‘Not Started’.

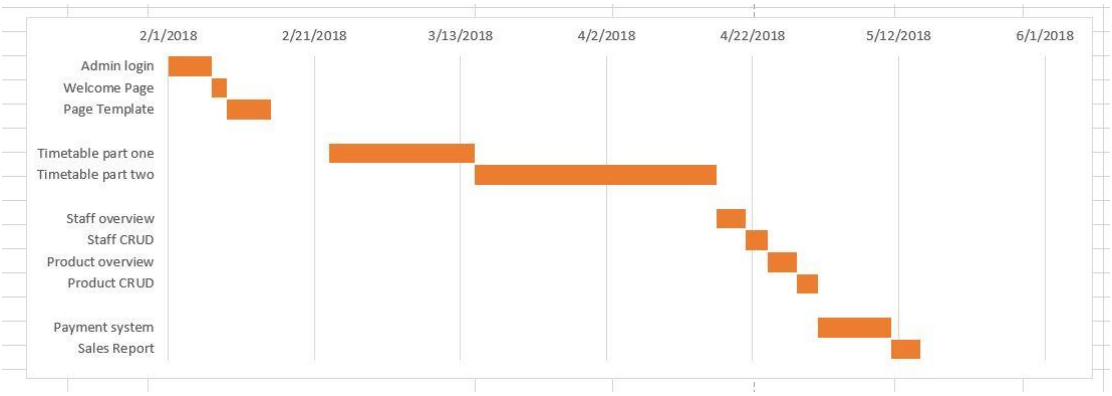

Figure 8 Gantt chart flow chart

Although the final day should be around June, as you can see from figure above, the last date is the middle of May. That is because during March and April, it is very hard to meet up with the cooperating manager due to busy business time. So, the feature should be done early as possible and then the next meeting available will only be in May. Since there is a lot to show, there must be a lot to changed, which requires times, that is why there is still half month of May to work on any changes.

8.1.3 Project communication

Communication is important as this is a cooperative project, so it is needed to integrate with the manager or the staff for stand-up meetings. Even so, the business time for hair salon is not fixed and it is hard to set a specific time.

So, in my case, by referring to their timeline which is sure to be busy, I will just drop by any time I want during the non-busy time. Normally I met up with the manager or his assistant, and go through the same three progress every time:

- Define what has been done
- Ask for review or suggestion
- Discuss compatibility of changes

I have also keep contact with the manager, Mr. Momo, through Whatapps. This is used when I need to confirm some features or Mr. Momo come up with some other ideas and decided to change.

8.1.4 Project Goal

Knowing the project goal helps to focus on the main aspect and not to move too far from it. The main goal for this project is to make **a working and simple web-based management application**, following by the Must Have requirements.

9.0 Conclusion

9.1.1 Future implementation

Here is the future implementation that are to be developed if this project is carry on:

- Customer Tracking

This refers to the future service, adding customer information, which is like the member of the salon. Information of the customer will be recorded, such as basic personal info, purchase history and appointment history.

- Advanced features

One feature that needed to be advanced now is the sales report. Users should be able to view sales report according to specific staff, customer or product. Also, the user should be able to read in monthly, weekly or yearly view.

- Different access platform

Different platform for manager and normal staff, Staff should not be able view the sales report, whereas only the manager can do so. So different platform should be made.

- Official website

A simple hair salon official website, but with an add on online booking system. In that way, customer can book the appointment themselves and then the data will be sent to the database and notice the staffs automatically.

9.1.2 Critical evaluation

Technical issues

The two most difficult features in this project are the appointment schedule table and the payment system. For the appointment schedule, I am using the JavaScript Event Calendar called full calendar.

You can see the what it originally looked like from here: https://fullcalendar.io/ , it is just a simple raw drag and drop calendar, which is not quite suitable to be used as an appointment timetable. So, I rewrote the JavaScript and make additional features to the calendar such as:

- Input fields will appear when adding new event, this include event name and the time and even the colour of the event tag on the timetable.
- Events can be edited when clicked together with the delete option.

The original calendar script does not have colour tags and is unable to edit, which is a main feature that is needed for this system. Also, there is minor issues when coding the payment system since I got really confused when linking the order list to the product and service table and then to the database for sales report.

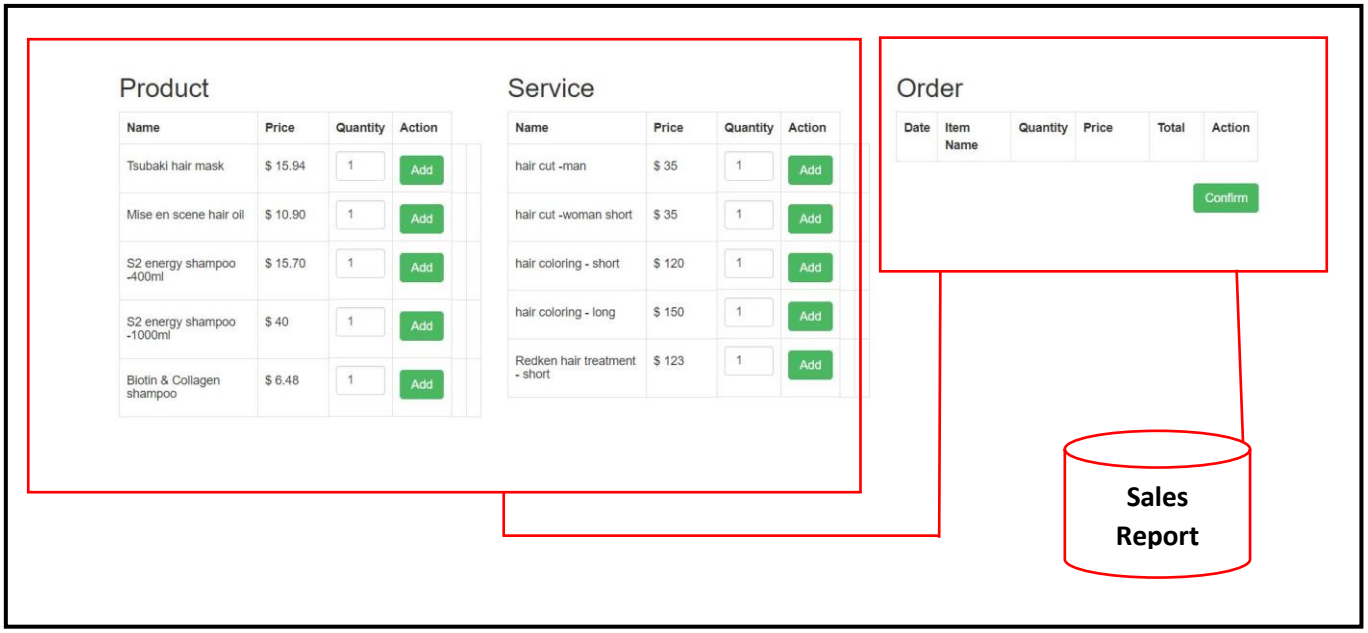

Figure 2 Payment system with database

During the progress of building the payment system, lots of errors were occurred, still I am able to manage the errors by referring to it one by one. But there is one issue that had me stuck completely. This is when working on the feature where the information of the order list will be saved to the database when user click confirm. The page can be run successfully but the data is not added to the database. I have check the database name and the SQL command and found no errors in it until I ask my lecturer for help. This is when I know that “order” is not a suitable table name as the SQL will define it as the keyword. So, instead of using the name order, I changed the table name to record.

So, in the progress for programming in this project, I wished that I have even more time to learn more about PHP and JavaScript so that I would not have conducted so many issues, which can save quite some times.

Real world challenge

For the real-world challenge, the biggest challenge for me is to dealing with sudden changes and flexible time.

Normally, I worked according the instruction and arrangement, following step by step very closely and would not dare to go out of tracks when working. In my expectation the workflow should be like below:

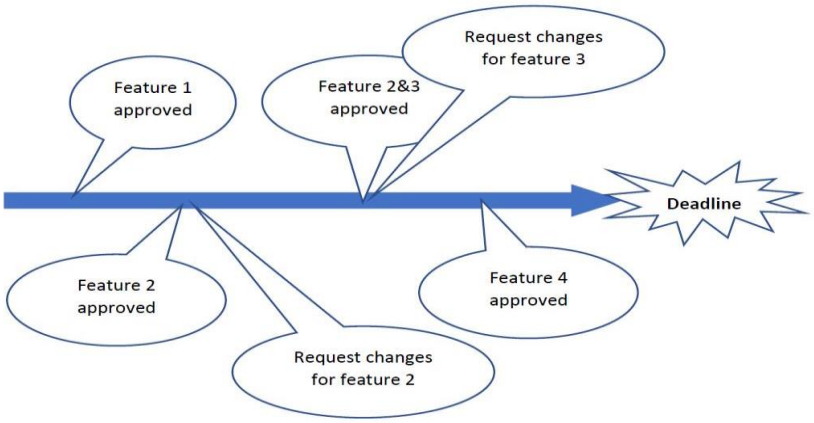

Figure 10 expected timeflow

But what happened in the real world was:

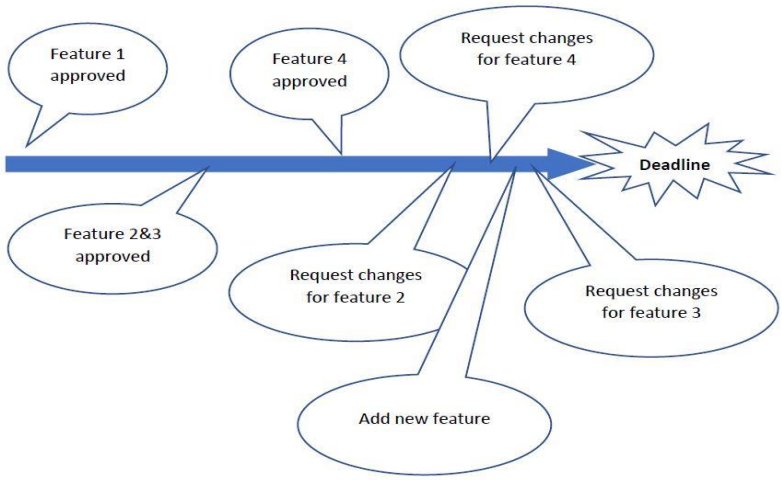

Figure 3 real world timeflow

The sudden changes messed up my expected timeline in my Gantt chart and panicked to stress when all these changes are requested at the time very near to the day of deadline. For then, time sacrifice is made in order to finish all the tasks and thankfully I was able to finish most of the features. So, dealing with abrupt changes and flexibility of time management are the main issues I faced in the real-world, and also the facts that I have learned from it.

10.0 Appendix

10.1.1 Implementation

10.1.1.2 User manual

Admin login

Step 1: Key in user name and user password.

Here you can use this account (username: admin, password: admin)

Step 2: Press enter or click the login button

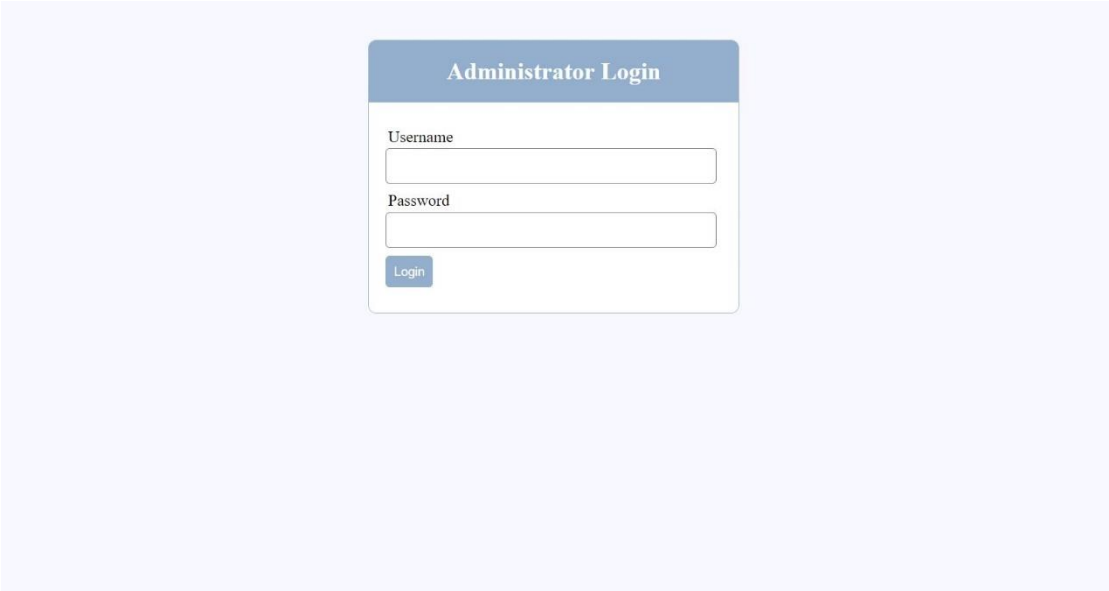

Figure 12 login page

Welcome page

Click ‘Get Started’ button

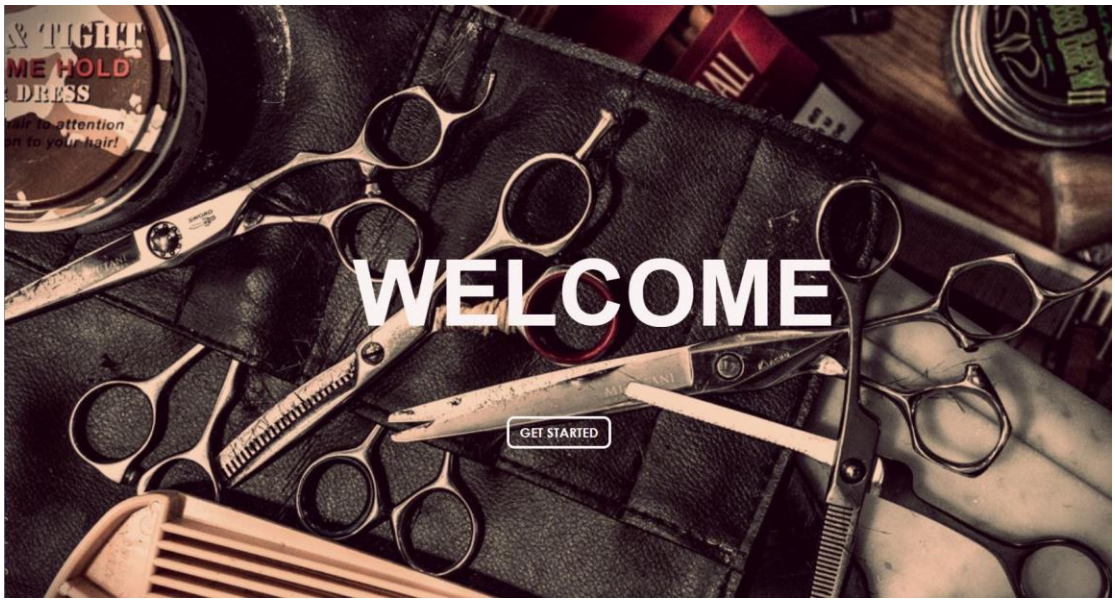

Figure 14 welcome page

Admin logout

Click on the ‘Logout’ button

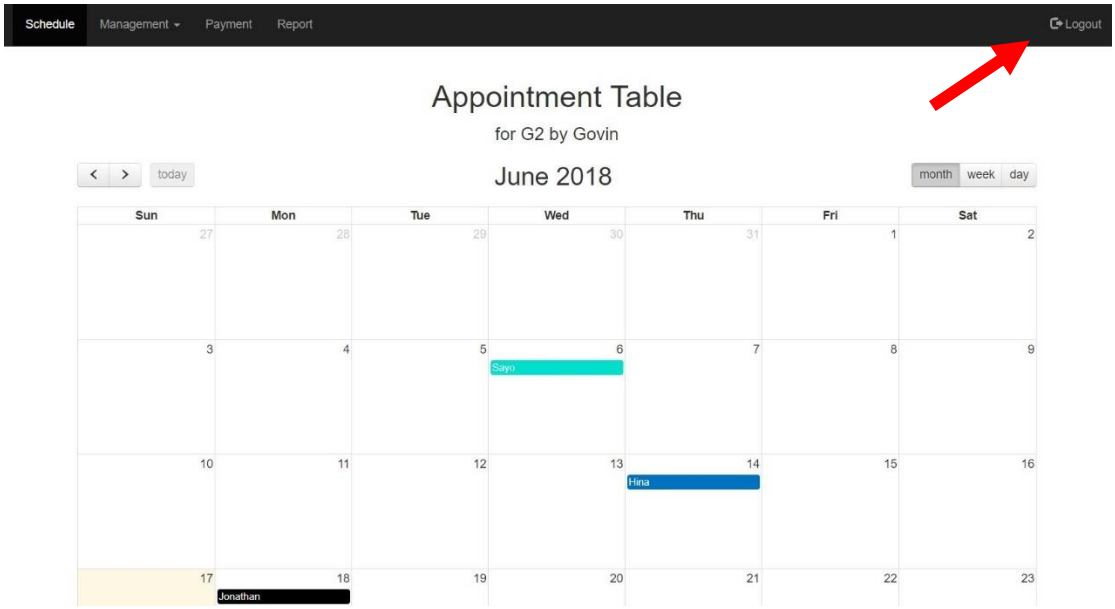

Figure 15 homepage

Make appointment

Step 1: Click on the date that you wished to make appointment

Step 2: Fill in the Title, which is for customer name

Step 3: Select colour, which represent the staff

Step 4: Enter date and time

Step 5: Click ‘Save Changes’

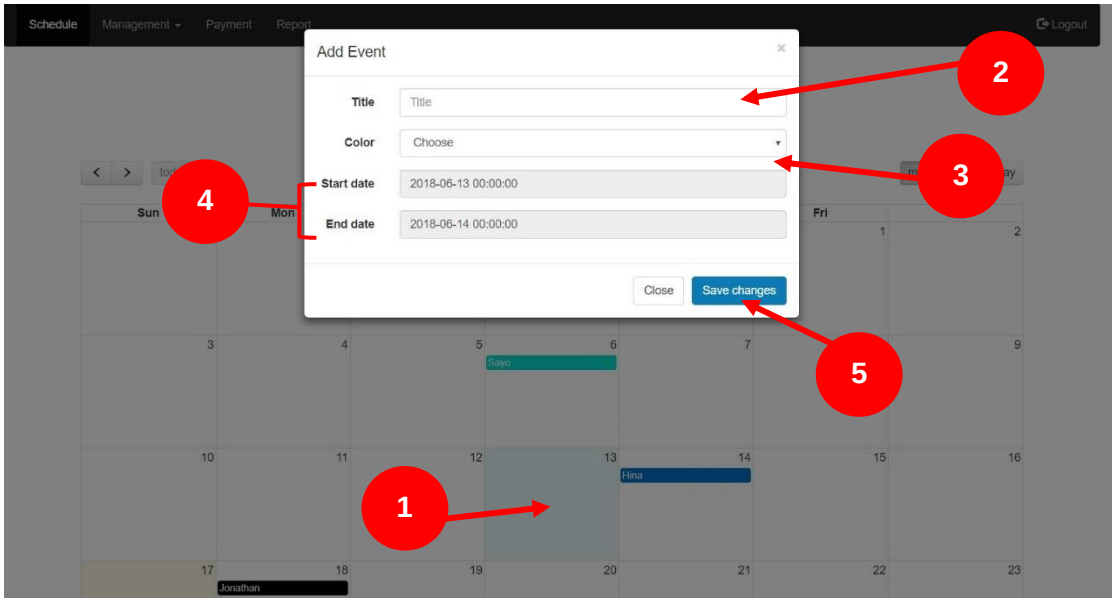

Figure 16 popup window add appointment

Edit appointment title or colours

Step 1: Double click on any appointment

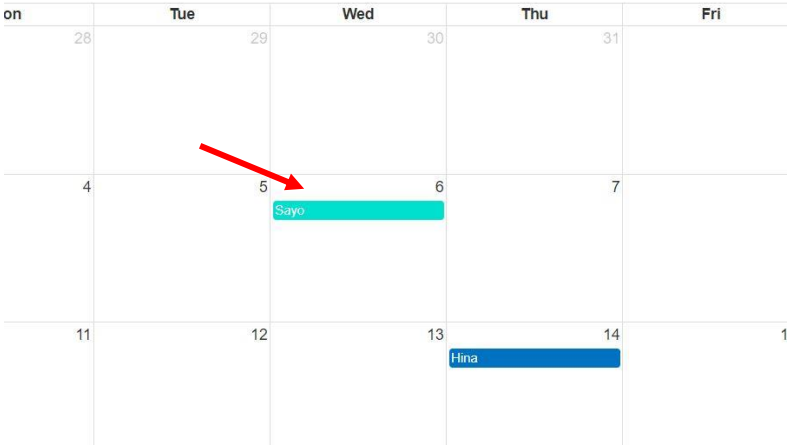

Figure 17 appointment selected

Step 2: Key in new title name or select different colours

Step 3: Click ‘Save Changes’

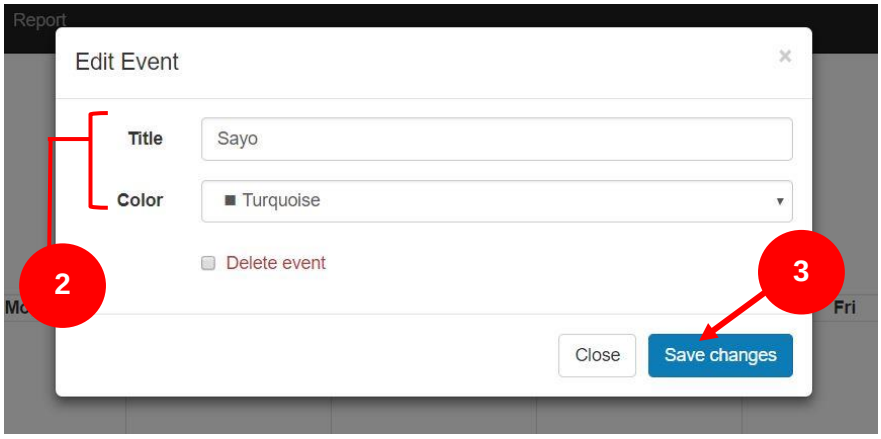

Figure 18 edit appointment

Delete appointment

Step 1: Double click on any appointment

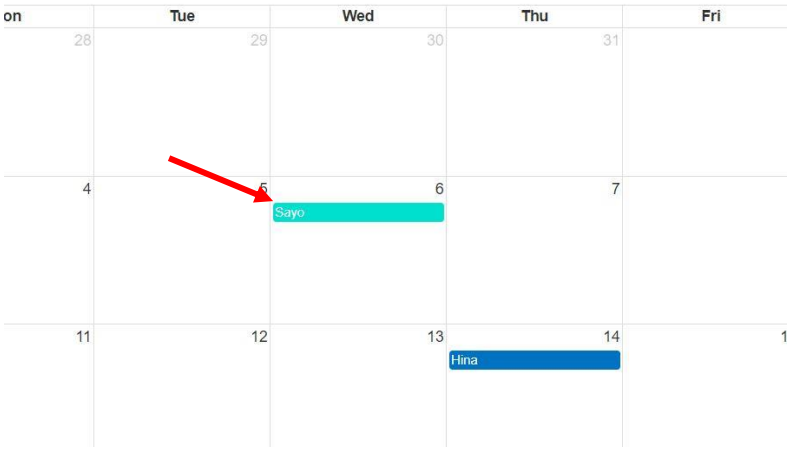

Figure 19 appointment selected

Step 2: Click ‘Delete event’

Step 3: Click ‘Save changes’

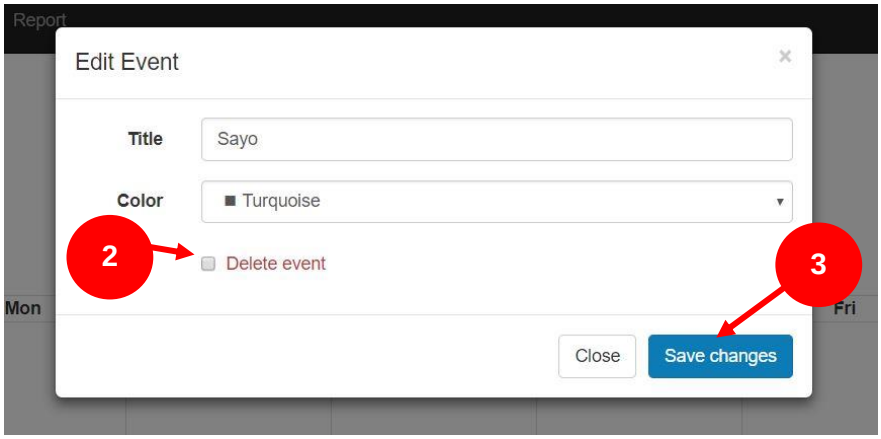

Figure 110 delete appointment

Edit appointment timeline

Change date duration-

Step 1: Put cursor on selected appointment

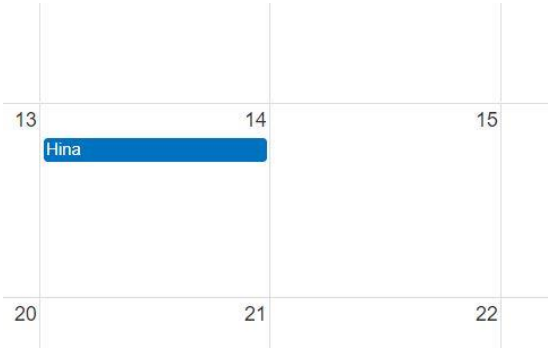

Figure 11 select appointment

Step 2: Pull to desire duration date

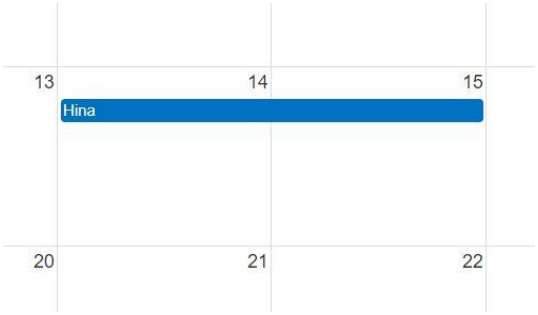

Figure 12 drag tag

Change date- Drag and drop selected appointment to desire date

See screenshot gif demo:
https://media.giphy.com/media/ZNYs1b3fGkevHSmoh5/giphy.gif

Change time-

Step 1: Click either ‘week’ or ‘day’

Figure 13 month week day

Step 2: Change the duration of time by dragging the tag bar

Figure 23 before edit

Figure 24 after edit

View timetable for previous and next month

1. Previous month 2. Next month

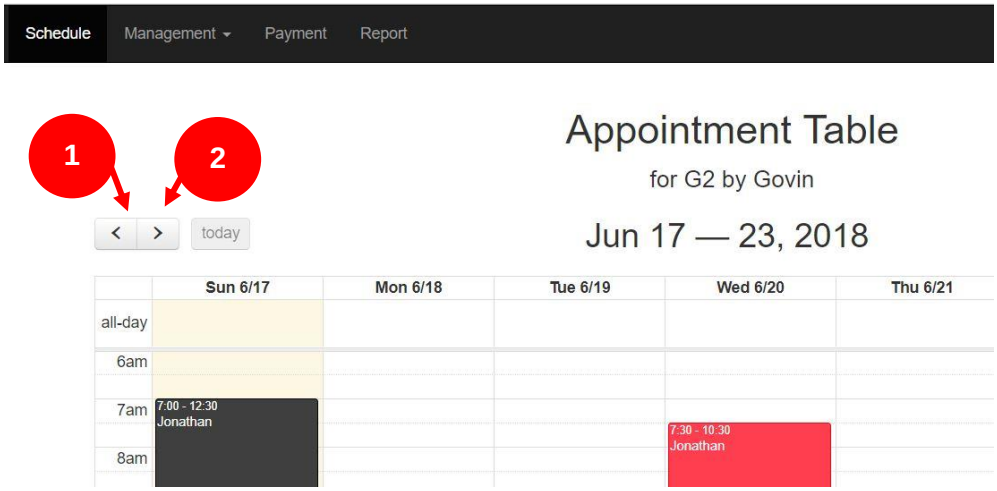

Figure 25 select month

View Staff list

Click on 'Staff overview'

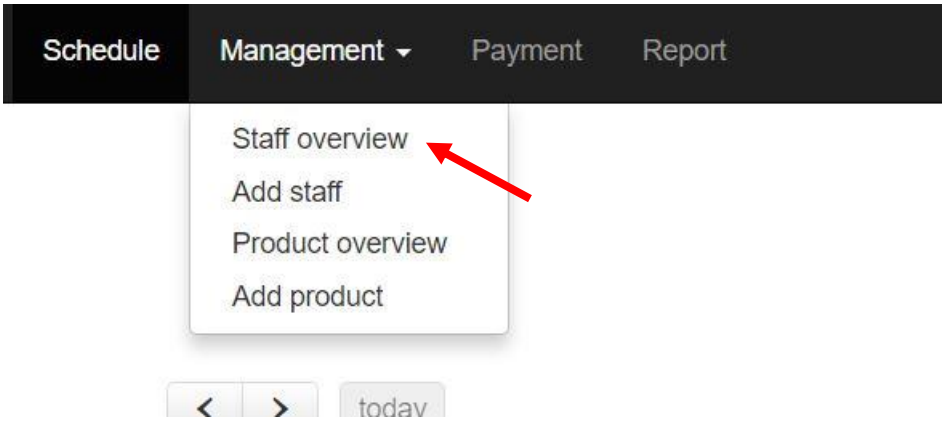

Figure 26 staff overview

View Product List

Click on 'Product overview'

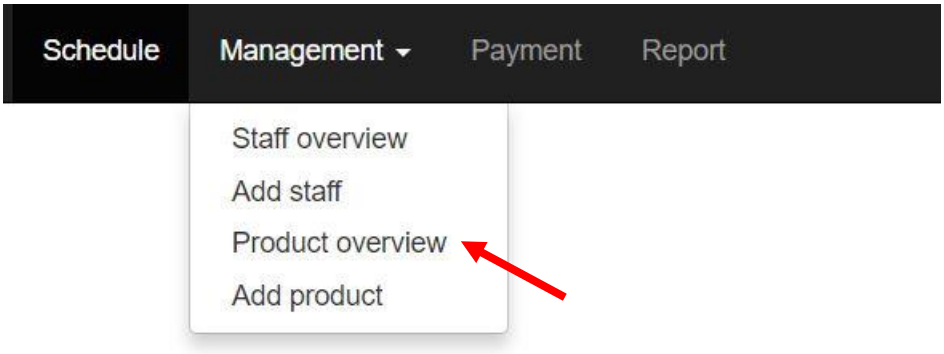

Figure 14 product overview

Edit staff information

Step 1: Click on ‘Staff overview’ to show staff list

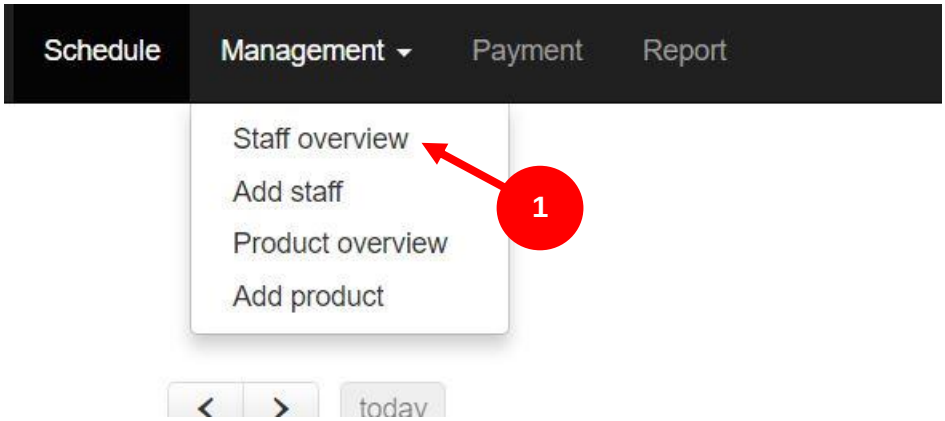

Figure 15 staff overview

Step 2: Click on ‘Edit’

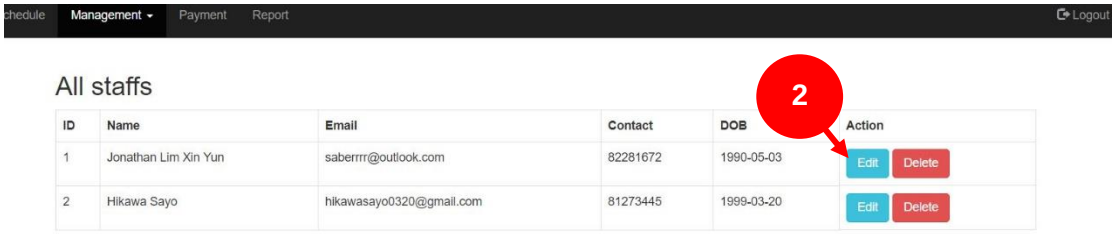

Figure 16 edit staff

Step 3: Key in the changes in the input fields

Step 4: Click ‘Update person

Figure 17 update staff

Edit product information

Step 1: Click on 'Product overview' to show product list

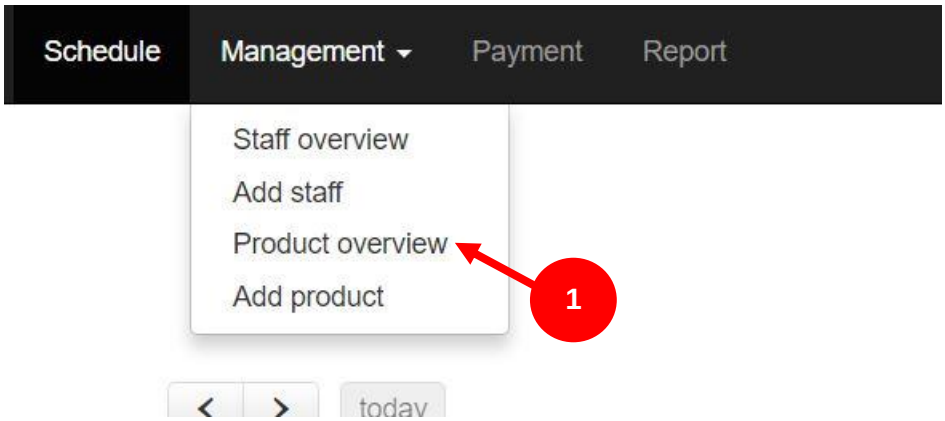

Figure 18 product overview

Step 2: Click on 'Edit'

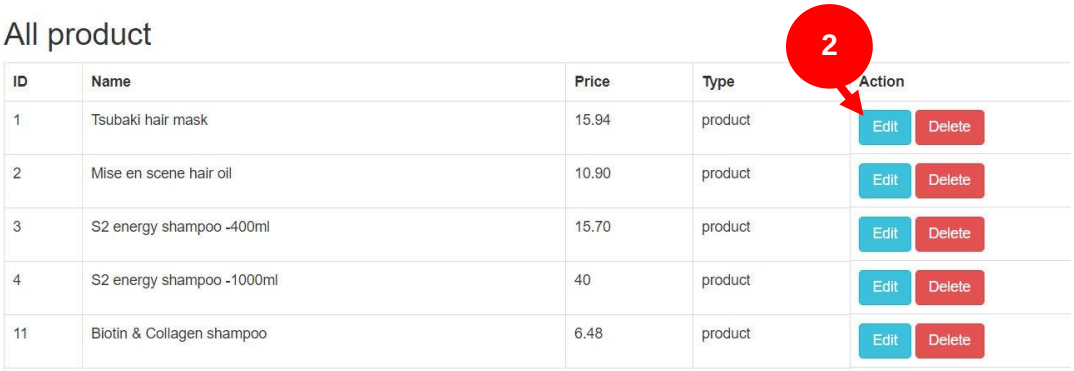

Figure 19 edit product

Step 3: Key in the changes in the input fields

Step 4: Click 'Update item

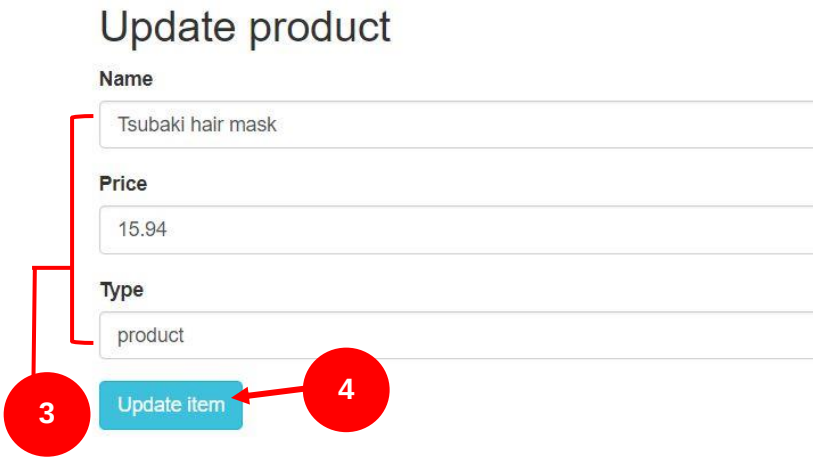

Figure 20 update product

Delete staff information

Step 1: Click on ‘Staff overview’ to show staff list

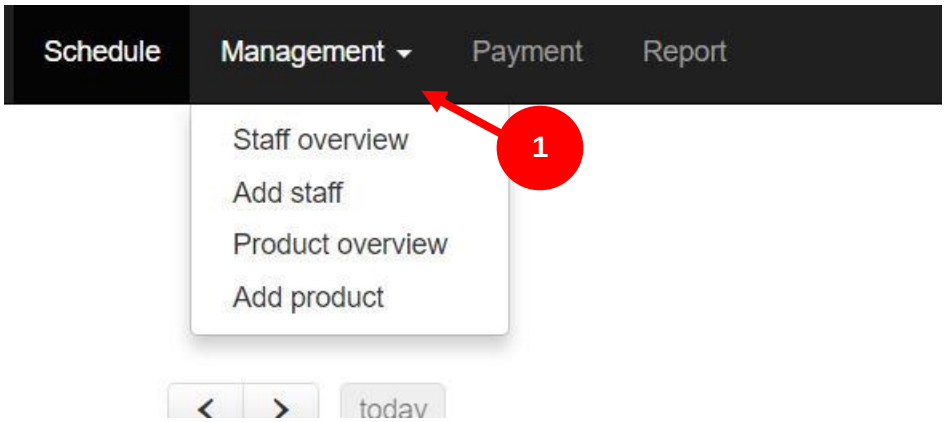

Figure 21 delete staff

Step 2: Click on ‘Delete’

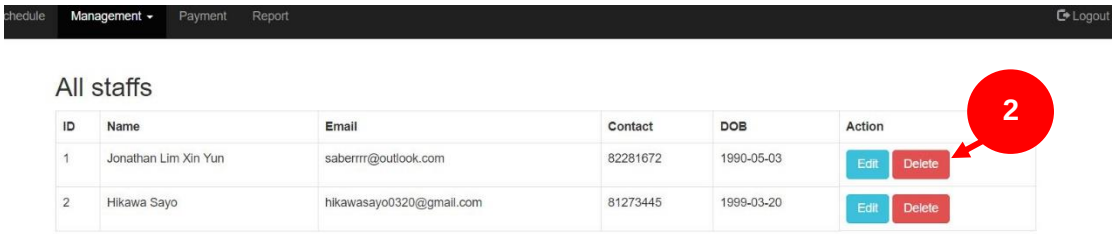

Figure 22 delete

Step 3: Press ENTER or click ‘OK’

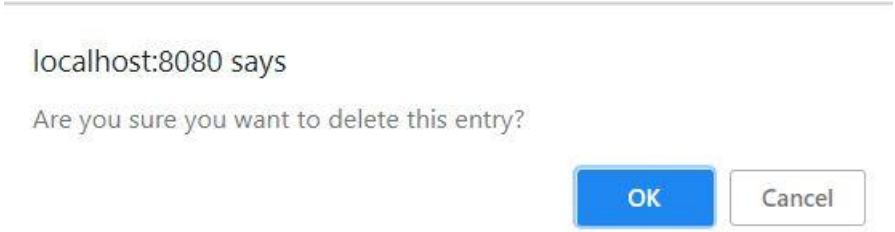

Figure 23 confirmation

Using the payment system

Step 1: Click ‘Payment’ to be directed to payment page

Step 2: Key in the quantity of sales product/ service.

Step 3: Click ‘Add’ to add product/ service details to order list

Step 4: Total Amount will be calculated

Step 5: Click ‘Confirm’ to save the payment data

Step 6: Click ‘Remove’ to remove product/ service from the order list

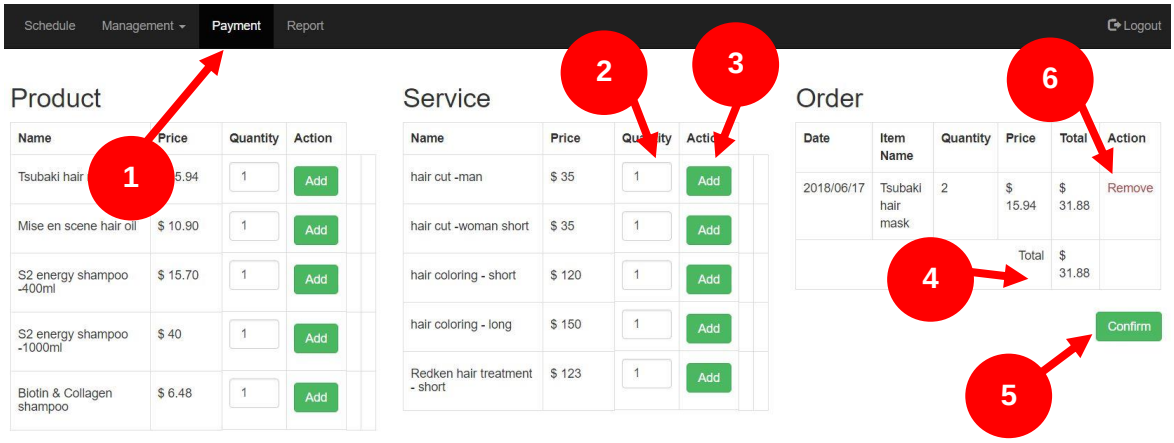

Figure 24 payment system

View Sales Report

Click on ‘Report’ button

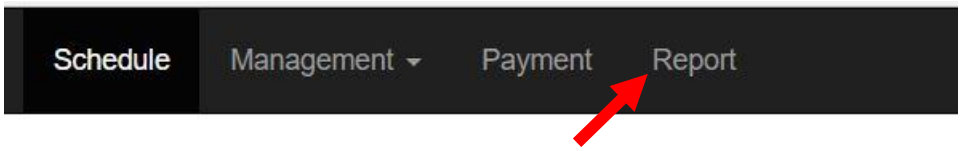

Figure 38 report

10.1.1.3 Installation manual

Note: This software installation is based on using WAMP server.

Step 1: Install WAMP

Step 2: Download the package from GitHub:
https://github.com/isabeltiongsk/hairsalon

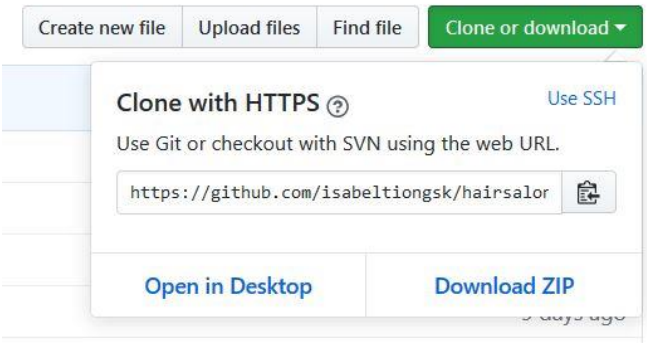

Figure 25 download from github

Step 3: Copy folder under wamp/www/folder

Figure 26 copy to folder

Step 4: Login into PhpMyAdmin

Step 5: Create three databases named ‘calendar’, ‘hairsalondb’ and ‘registration’.

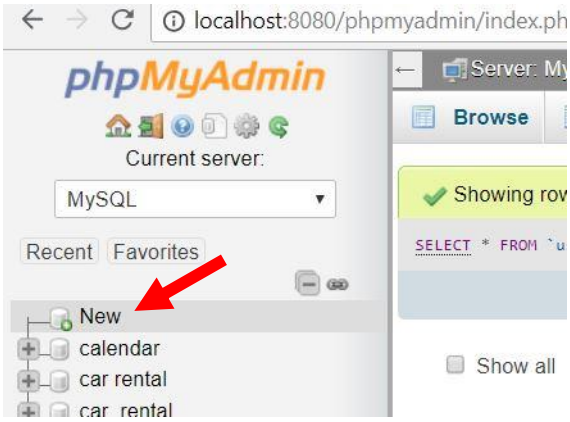

Figure 27 add new database in phpmyadmin

Step 6: Insert the SQL files from the package that you downloaded in step 1 accordingly based on the filename. (For example: input calendar.sql to calendar database)

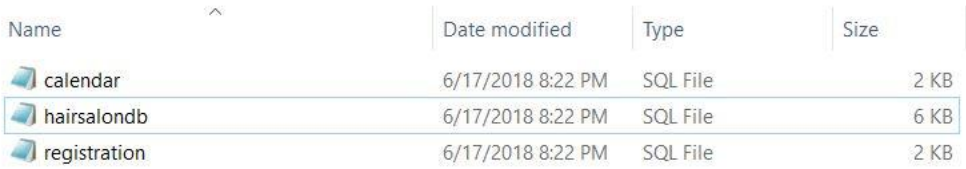

Figure 28 sql files

Step 8: Access to WAMP localhost

Step 9: Select the file and run

10.1.1.3 Error handling manual

This is the manual for solving the errors that might occurred during installation.

1. Error message showing database is not connected

- Check spelling correction of your database name, it should be 'calendar', 'registration' and 'hairsalondb', make sure that you insert the correct SQL files into the databases.
- Check database user account information, this software is using the default user account, which is:
  Hostname: localhost
  Username: root
  Password: (none)
  If you are not using the same as above, please switch to the default user account in your database.

2. Error message showing 'mysqli…'
- Check your PHP version, make sure that it is version 5.6 or above.

3. WAMP service is not running
- Try using other port, to do so:
  1. Right click on the WAMP icon in the toolbar
  2. Select 'tools'
  3. Select 'Use a port other than…'
  4. Enter other port number (example:8080)

If any other error occurred during installation, feel free to contact me by email: isabeltiongsk@gmail.com and tell me about the error.

10.1.2 Project Proposal

Link for project proposal: https://www.scribd.com/document/381951188/Fyp-Proposal

11.0 References

Brandenburg, L. (2016). *Beginner to data dictionary*. [online] Laura Brandenburg. Available at: http://www.bridging-the-gap.com/data-dictionary/ [Accessed 3 Jun. 2018].

Charlie.L. (2015). *Open Sourced Point of Sales Web Client*. [online] Available at: https://www.youtube.com/watch?v=3PyVSeU-JiE [Accessed 17 May 2018].

Codewife. (2017). *Create PHP Admin Panel In Just 5 Minutes*. [online] Available at: https://www.youtube.com/watch?v=OXwYU03FeEY [Accessed 21 May 2018].

Frappe. (2017). *frappe/erpnext*. [online] Available at: https://github.com/frappe/erpnext [Accessed 27 May 2018].

Fullcalendar.io. (2017). *License | FullCalendar*. [online] Available at: https://fullcalendar.io/license [Accessed 3 Apr. 2018].

Itinfo.am. (2014). *Software Development Methodologies*. [online] Available at: http://www.itinfo.am/eng/software-development-methodologies/ [Accessed 9 Apr. 2018].

Nose.Gne. (2016). *Make and admin panel - PHP integration*. [online] Available at: https://www.youtube.com/watch?v=Oo18wDiswRI [Accessed 22 May 2018].

Php.net. (2016). *PHP for beginner*. [online] Available at: http://php.net/manual/en/intro-whatis.php [Accessed 17 Feb. 2018].

Smartdraw. (2015). *Entity Relationship Diagram - Everything You Need to Know About ER Diagrams*. [online] Available at: https://www.smartdraw.com/entity-relationship-diagram/ [Accessed 20 May 2018].

Template.net. (2017). *Tutorial for Software Plan Template*. [online] Available at: https://www.template.net/business/plan-templates/software-plan-template/ [Accessed 4 Mar. 2018].