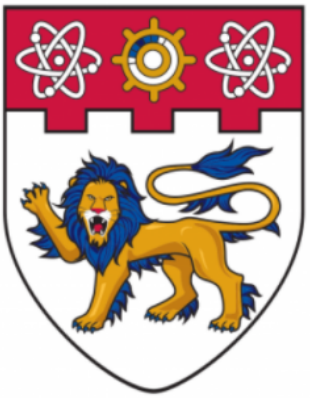


NANYANG TECHNOLOGICAL UNIVERSITY
SINGAPORE

REPORT
ON
INDUSTRIAL ATTACHMENT
WITH
ELEOS WEB PTE.LTD

Xiong ChenYu
U1521516C
EEE

May 16, 2017

Contents

List of Tables

List of Figures

1 Abstract

This report describes the internship I spent at the ELEOS WEB PTE.LED. The ELEOS WEB PTL.LES is a web company provide solution for retailer management. They provide lots of products.

- Octopus™ Retail POS :

  A Comprehensive Retail Management Solution Complete, easy-to-use, stand-alone POS solution specially designed for small and medium scale retail businesses such as clothing stores, electronic computer stores, pharmacies drug stores and almost any other retail stores. Integrates perfectly with your desktop, laptops or mobile devices.

- Octopus™ Food POS :

  A "Smart" Cafe-Experience
  Octopus™ Food POS is an ideal choice for F&B outlets. Connecting the Point-of-Sales device with the rest of your other devices – OctoWaiter, OctoMenu and self-service kiosk seamlessly, Octopus™ Food POS provides a modern seamless concept and high efficiency for your F&B operations.

- E-Commerce :

  Break Geographical Barriers. Reach Out to Your Potential Customers. **Octopus™ E-Commerce** replicates your offline business into online business literally in minutes. Armed with a powerful management interface, Octopus™ E-Commerce allows users to manage their own virtual stores easily, including management of images, setting up promotion activities and monitoring of your inventory and customers from an integrated standpoint.

  – YOUR BUSINESS NEVER SLEEPS :

  Get your business to work for you even when you are asleep or on vacation. Publish your products / services and start an online presence.

  – GO BORDERLESS :

  Break out from your market and bring your business to the international level. See how our solutions can enhance your market reach and customer base.

  – POS-CONNECTED HASSLE-FREE PRODUCT INPUT :

  Add in product easily from your existing POS product database. Or opt to input new or online-exclusive products via web access. Makes inventory management easy.

  – FUSS-FREE CONTENT MANAGEMENT :

Choose your preferred theme, customise the layout and you are ready to sell online. Simple interface allows you to insert your company logo; create new pages, page navigation and product blogs easily. No prior programming skills needed.

- Companion Apps :
  - Octopus™ Mobile POS :

Point of Sales with Mobility in Mind

POS is now even more compact for your phone. Octopus™ Mobile POS is a complete all-in-one solution that is ideal for retailers and consumer merchants alike.

  - Octopus™ Admin Dashboard :

Pulse for Your Business Octopus™ Admin Dashboard gives the business owners their key performance indexes - whether it is their sales revenue, best sellers or what colour sells best in which season.

With Octopus™ Admin Dashboard, business owners have the pulse of their business at their finger tips.

2 Acknowledgment

First I should thank Nanyang Technology University gives me a change to have the internship experience.
Second I should thank my supervisor Mr. Drake, he gives me a lot of freedom to explore and learn new frameworks.
Thirdly I should thank Prof. Jong Ching Chuen who gave me a lot of advise during the Internship.
Lastly I should thank all my fellows working in Eloes Web interns and formal stuff. They help me a lot during the internship.

3 Chapter One

3.1 Introduction

During my internship period, I work as Full-stack developer. My duty is to develop a royalty and promotion system. The system consist 2 parts, one is the android and ios application running on customers mobile device. They can use the application to favorite the shops and get royalty points. What's more they can also receive the E-stamps and get the new promotion notifications. Another part is a AdminDashboard for the shop holders. They can use this to add the new customers to their loyalty system and group them to distribute the e-stamps, e-vouchers and push the promotion notifications.

3.2 Background

I decide to use the most baised framework metoer witch have the most fastest develop speed,so the can test the prototype easily,and make changes,and we use the mongodb cause it is not schema based that means we can change the database structure easilily and have the less side effect,despire we have to create the customize schema on application level on my own .

The learning curve is good,first I do the 3 todo app on the official web and I found the react version is more suitable for the project cause in the future we need the mobile version on android and ios so the react native is a better choice.So build the react to our skill stack will save the future.

I need to develop the ios app,so I can not skip the mac os.But not all my notebook is mac to I need a stable system to develop the backend and frontend of the system.I used to be a Arch Linux user but It is just a platform for play with the linux, the community is quite ambicious.They always take the most latest as the rolling distribution.Another popular choice is geetoo wich also very clean.But the geetoo is souce based rather than binary based compared to arch which means the you can control the compiling process on you own so you need a high proformeance computer.

So I use elementary OS witch is based on ubuntu but it more like mac OS.The installing is very easy.Follow the official and you will get it easily. I decide to choose a editor for my life change.The choice is firstly vim and then change to the nvim for the better economy of the synchronized api which makes the editor much faster and never block the ui.

I list all the all thins I used to create the application from front-end to backend.

Table 1: The Usages and tools

Usages	Tools
Editors	Neovim, Emacs
Debugger	Google-Chrome
Backend Framework	MeteorJs
Front-end Framework	ReactJs
Front-Ui Framework	Meterial-Ui
Front-State Framework	Redux
Distribution Framework	Webpack 2
Database	MongoDb
Documentation	GitBook

4 Chapter Two

4.1 Team Work

We use Github as the version control system to do the team work. We do the different parts together and merge it to a single files. We use the github branch to manage the new feature testing and gh-pages page to use the gihub page to make documentation.

4.2 Mobile Application

It is the Android and Ios application running on customers *mobile device*. They can use the application to *favorite the shops and get royalty points*. What's more they can also receive the E-stamps and get the new promotion notifications.

4.3 Dashboard

AdminDashboard for the shop holders. They can use this to add the new customers to their loyalty system and group them to distribute the *e-stamps*, *e-vouchers create* and push the *promotion notifications*

4.4 Review of theory and previous work

Below shows all my requirement and the successful implementation during the internship period.

4.4.1 System Scope(Functional Requirement)

1. Customer Side
    - ☒ **Basic Feature**
        - ☒ User Register System(Third Party Oauth)

Figure 1: Login Page

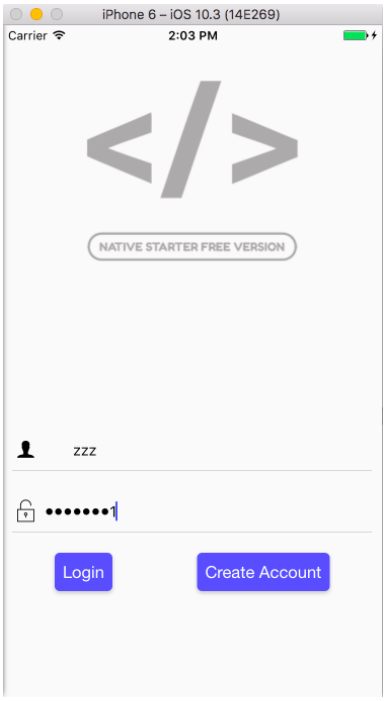


Figure 2: Mobile Functions

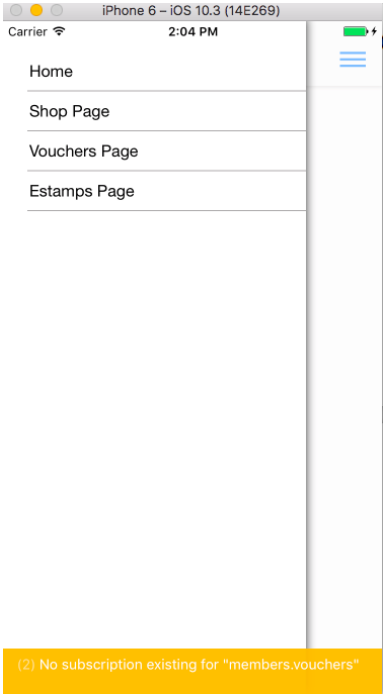


Figure 3: Dashboard for the shop holders

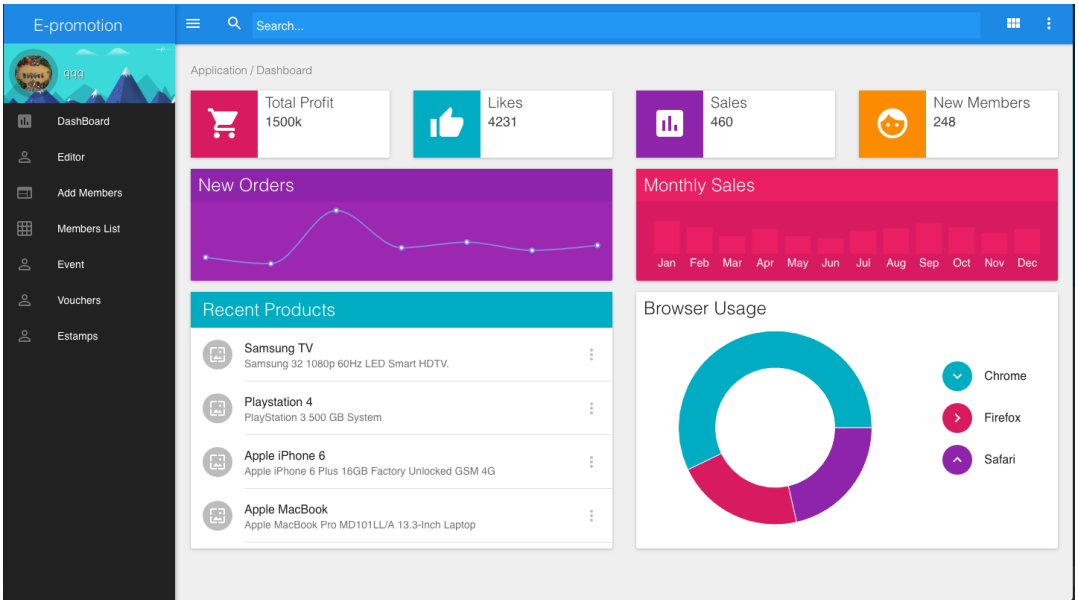


Figure 4: Editor to write new notifications

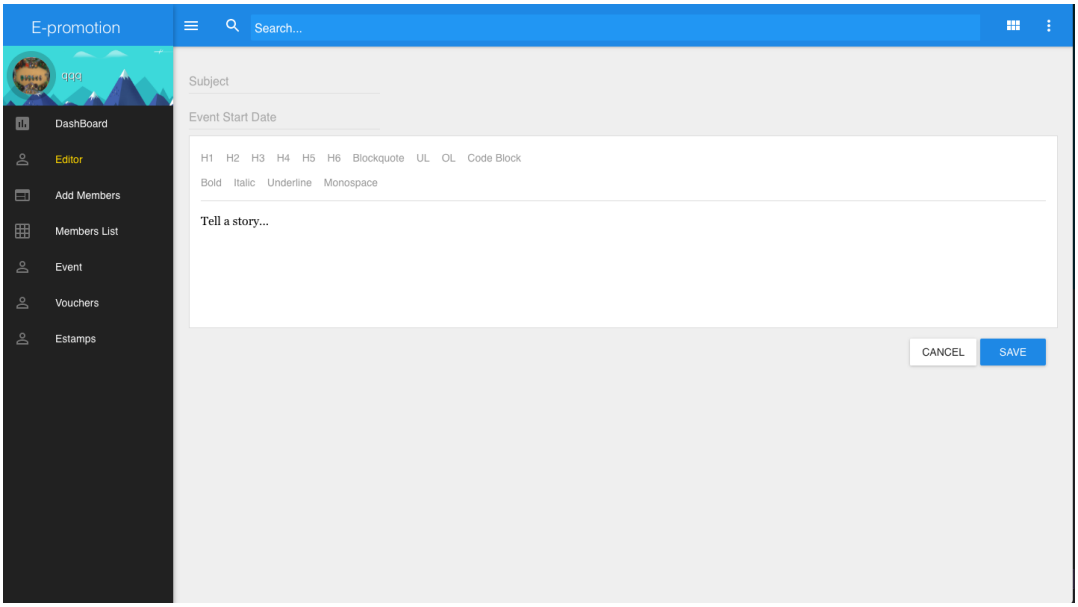


Figure 5: Form to create new vouchers

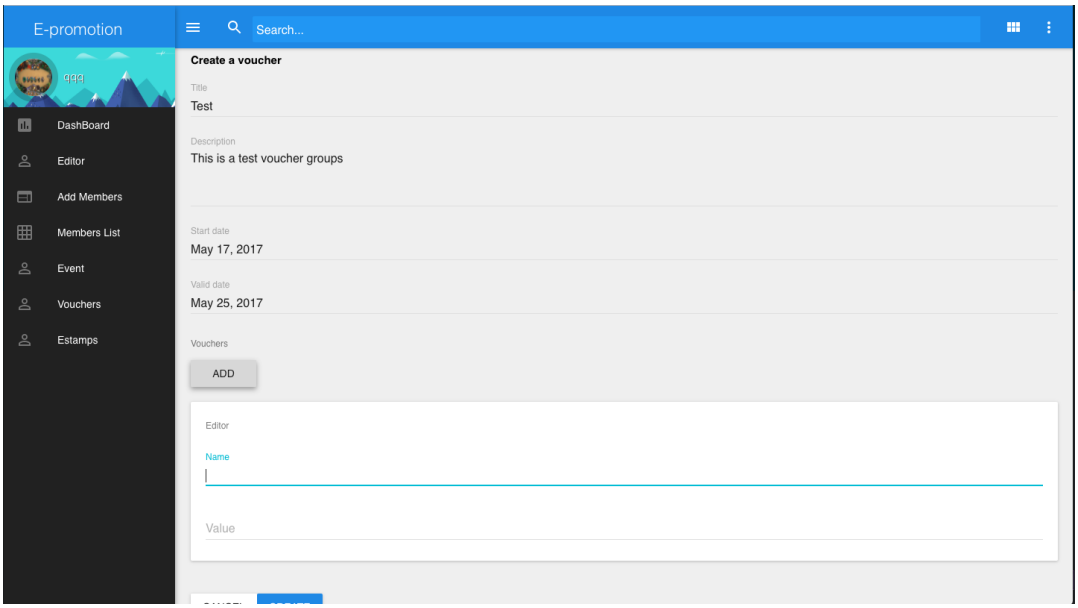


Figure 6: Estamps page

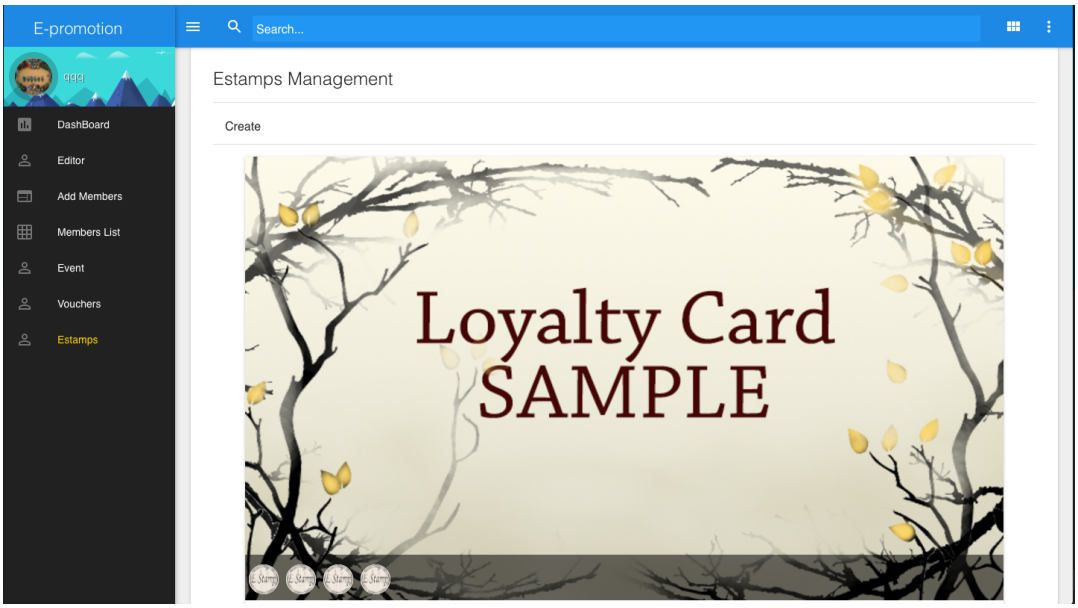


Figure 7: Page to add new member

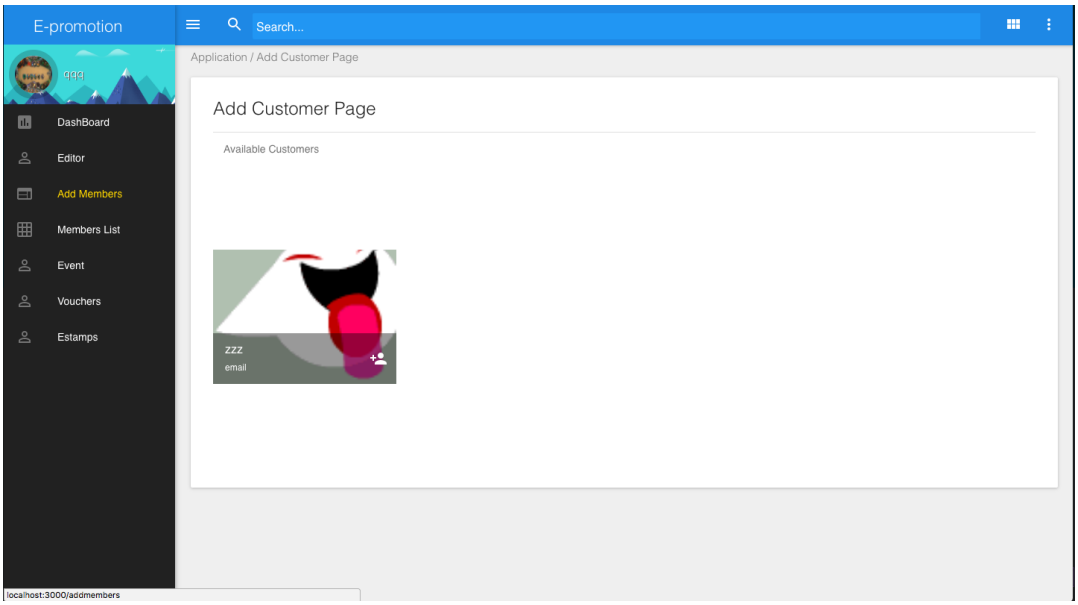
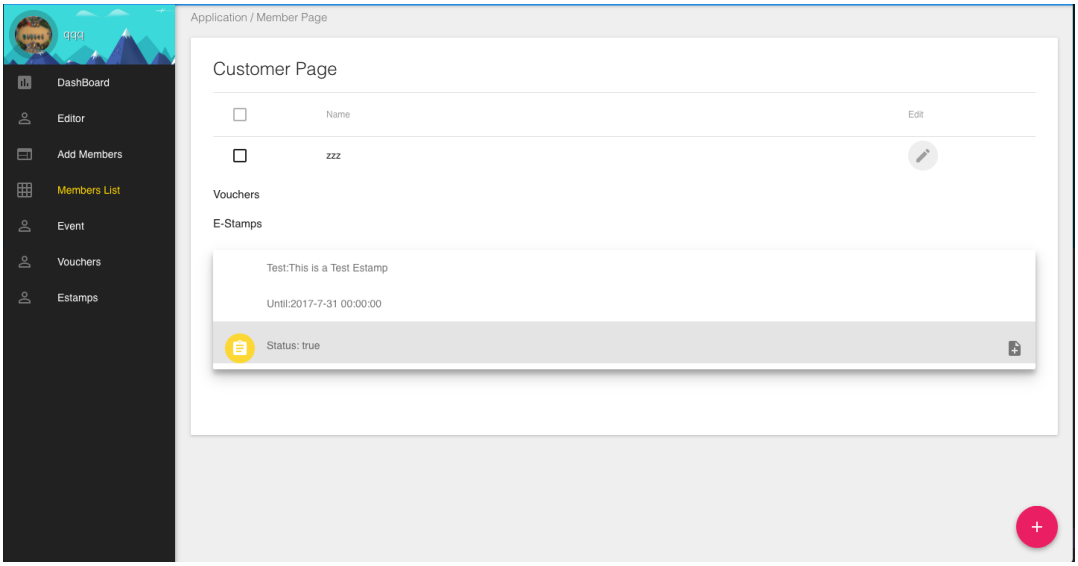


Figure 8: Page to distribute vouchers



        - ☒ Browse membership to different shops(Sorting,Ranking)
        - ☒ Check the royalty system in shop(Point,E-stamp)
        - ☒ Promotion System(Vouchers,Sales notification)
    - ☒ **Medium Feature**
        - ☒ Redeem gift
    - ☐ **Advanced Feature**
        - ☐ E-wallet(Paypel,Wechat-Pay,ALi-Pay)
        - ☐ E payment deduct voucher value

2. Merchant Side
    - ☒ **Basic Feature**
        - ☒ Regist Shop
        - ☒ Selcet templete from template royalty system
        - ☒ Distribute Membership to Customer(IC,Scan QR code)
        - ☒ Dashboard to display the info of promotion status
        - ☒ Modify shop detail
    - ☐ **Intemeida Feature**
        - ☒ Define Member Teir system Based on points
        - ☐ Define User Groups
        - ☐ Api to combine our system to the exitsting Pos Systme
        - ☒ Create Voucher
    - ☐ **Advanced Feature**
        - ☐ Costumize The vouchers(Animation) and so on.
        - ☐ Costumize tier metal

4.4.2 Execution & Evolution qualities(Non-functional Requirement)

- ☐ **Deploy & Opertate**
    - ☐ Docker
- ☐ Backup
- ☒ Unit Testing
- ☐ Pressure Tesing

4.5 Self explore

I did learn a lot from this internship. Cause it is the first time I am in charge of a huge project and explore every things the frameworks extra. So I will talk more on this in different respect.

4.5.1 Literate writing

When I start to write documentation, I pay a lot of attention to the iterate writing. This kind of skills set help me to write documentations, reports, and academic papers a lots. Because you can write code when you write things and it gives you a lot of efficiency. I use this skills to write this reports. The tables of content and the table, figures graphs all auto-generate. And you can do more. You can find a demo in Python Plot

```
import pylab as pl
from numpy import sin, pi, linspace
t = linspace(0, 2*pi, 100)
pl.plot(t, sin(t))
pic = 'img/myfig.png'
pl.savefig(pic)
return 'img/myfig.png'
```

4.5.2 Category theory

During the last month, I pay a lot time to learn the category mathematics. Because I find myself lack of this kind of knowledge since I am a electronic student not computer science. While I decide to be a programmer in the future. I love programming and I find the category theory help me a lot to build real world model to a software model.

`Primitive` : No property Abstraction `Composition`
`Identity` f . $\mathrm{id_a}$ = f
`Associativity` h . (g . f) = (h . g) . f
`Function` : A mapping of values to values

1. Monoid NOTE *[2017-05-09 Tue 18:27]* mappend maps an element of a monoid set to a function acting on that set
2. Category NOTE *[2017-05-09 Tue 23:17]* `Kleisli category` a category based on a monad.
3. Initial Object NOTE *[2017-05-09 Tue 23:36]* The initial object is the object that has one and only one morphism going to any object in the category.

Figure 9: Composition

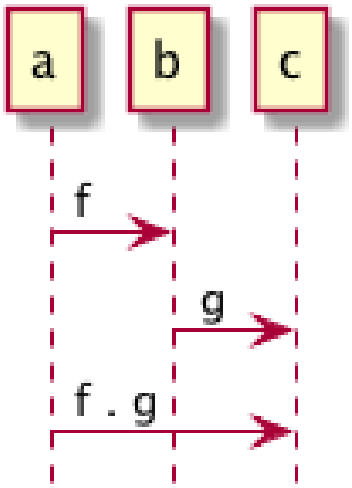


Figure 10: Identity

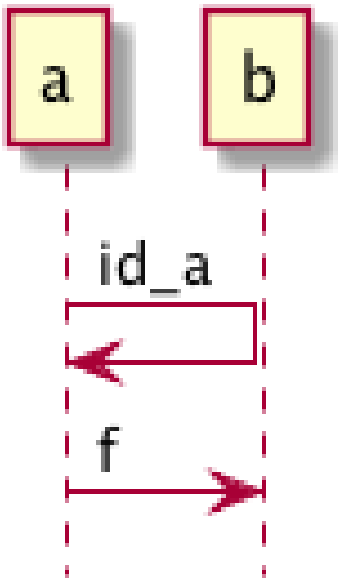
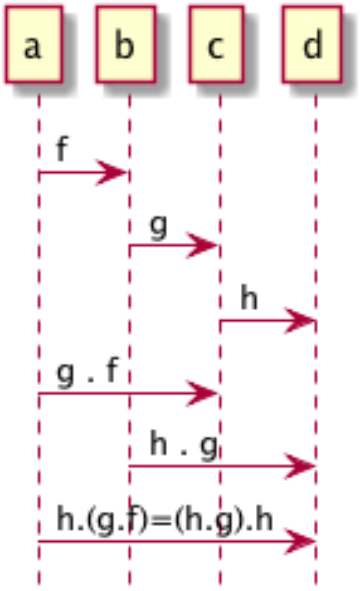


Figure 11: Associativity



5 Chapter Three

5.1 Summary and Conclusion

I explore a lot good stuff during the internship, and make lots of life change movement. Below show all my tools chains and they working together fantastic.

I did love and enjoy the key-bindings of vim, from the day I switch from the emacs, but I switch back shortly because of the 2 main reasons.

1. Emacs plugins works on my windows but the my vim got some problem and I have to dirty my configuration files so much to fixed.

2. I need to use playframework for development and the emacs support ensime very well.

I move from emacs to vim then to neovim and now I am using the spacemacs, I think the configuration for everyone will become the same after long period of learning and copying from others, so it is good to give the community driven software a try.

Besides that, I also earn lots of the develop experience in the web area. And I have the confidence to do well in the final year project.

Table 2: My life Change tools

Usage	Application
Email	imap:mbsync smtp:msmtp
Crypt	gnu2
Editor	Emacs, neovim
Ide	Intellij
Shell	Zsh
Shell manager	Oh-my-zsh
Searcher	fasd, ag
Termial	Iterm2
Chat	QQ, Wechat, Erc
Presentation	Reveal.js
GTD	Org Agenda
Notes	Onenote -> Org Mode
Graph Draw	Plantuml, Ditaa, Gnuplot
Documentation View	Dash
Book Writing	Gitbook
Finance	Ledger
Auto Deploy	Pm2, Docker
Academic Writing	Pandadoc, Org mode
Desktop	Xmonad

6 References

6.1 My Book List

1. **DONE** Learn Your Haskell A Great Good
2. **HOLD** Introduction to algorithms HOLD
   - State "HOLD" from "TODO" *[2017-05-04 Thu 13:12]*
     Stop at black and red tree
3. **WAITING** Computer System (A Programmer Perspective) WAITING
   - State "WAITING" from "TODO" *[2017-05-04 Thu 10:33]*
     Boring

7 Appendix A

7.1 Gantt Table

```
Project starts the 9th of Jan 2017
[Research and Learn] lasts 31 days and is colored in Lavender/LightBlue
[Prototype Design] lasts 31 days and is colored in Coral/Green
[Prototype Design] starts 14 days after [Research and Learn]'s end
[Test prototype] lasts 31 days and is colored in Red
[Test prototype] starts 2 days before [Prototype Design]'s end
[Write tests] lasts 5 days and ends at [Prototype Design]'s end
[Document] lasts 31 days and ends at [Write tests]'s start
[Init and write tests report] is colored in Coral/Green
[Init and write tests report] starts 31 day after [Test prototype]'s start
[Self explore] lasts 25 days and is colored in Black
[Self explore] starts 4 days before [Init and write tests report]'s end
```

Figure 12: Gantt Table