BTECH 451
Mid Year Report

Idphoto.co.nz

Hsin-Hua (Suzzie) Yang
6070085

People

Academic supervisor
Xinfeng Ye

Company supervisor
Paul Young

Abstract

The final stage to working towards a Bachelor of Technology (BTECH) degree have been about working alongside with a New Zealand company project team and taking part in their development project. The aim of my project is about developing a functional website platform for online production of passport photos. Aspects regarding to website development and cases relevant to the project are studied and analysed for solutions aiming at a particular problem case. Addressing issues that have arisen during the development process and possible solutions are identified solve or mitigating the impact of the problem.

Chapter 1
Introduction

Information Technology (IT) under BTECH is a 4 year degree program which offers various paper study in the fields of information system and computer science. In the final year of study, it is compulsory for students to undertake a year-long project that is offered in BTECH 451 and worth an accumulative of 45 points for both semesters - BTECH 451A and BTECH 451B respectively. This report summarises the research findings, current work done in the first semester. Future plan of the project is proposed in the report and in depth research regarding to recommended solutions are to be discussed further in the final-report.

For my project I am working on the development of a service-based business website called "idphoto.co.nz" which is an online platform for producing standard ID photos targeting nationwide customers seeking for a quick and low cost photo production.

1.1 The company

WOW Digital Image is a New Zealand photo-service company located in East Auckland. The company started in 1996, since then it has slowly increased its reputation around the area based on their quality assurance standard and has built a large local customer base. Recent years consumer demands in developing physical photos has greatly decreased as the introduction of digital technology spawns and the rapid growth of Internet has allowed people to store digital files and viewing photos on social medias instead of having physical photo printouts stored in albums. As a result, the majority of the customers come for services like passport photo and studio photo taking.

1.2 idphoto.co.nz

In an advance Internet world, It is imperative for businesses today to understand the benefits utilizing this borderless, resource abundant technology. Migrating services online also known as e-commerce has become a mandatory action in terms of broadening the company's targeting customer, efficiency of the service as well as keeping along with the current technology trend to stay in the competitive market. The company therefore came up with the idea of bringing its passport photo service online which scales perfectly with the online passport renewal service that was introduced in 2012 by the New Zealand government Internal Affairs department. The company has offered a price that is relatively low comparing to the current photo lab pricing. On top of that, the time is estimated to decrease to 5 minutes per digital photo product, benefiting both the customers and company expenses. idphoto.co.nz can potentially win over all passport photo taking labs in the future.

1.3 Previous design

Before I join the development team, the front-end and some parts of back-end basic functionalities have already been implemented. List of back-end functions are listed as follows.

- User login
- Store user detail database
- Store customer order detail database
- Back-end management user interface(UI)

The UI was going for a single-page design to keep the service simple with easy scrolling and click action. Information is presented all in one place so that user does not need to navigate to other pages. With single-page design, bandwidth consumption is generally less comparing to a multi-page website which gives the user a better experience with less waiting time but it really depends on the content type and amount.

Judging from a personal view on the design outcome, the UI was not user-friendly and lack professionalism in terms of the look and styling of the website. The company has been considering for a redesign of the front-end website and I was given to task to redesign the entire layout of the website UI structure as well as understanding the already existing back-end implementations.

I think the challenge for me is to able to integrate the already existing back-end functions with the new UI as well as implementing security procedures in the website.

Chapter 2
Project related research

2.1 New Zealand standard ID photo

According to New Zealand's department of Internal Affairs, the standard ID photo for a passport photo is to meet the requirement of a 35mm x 45mm size physical photo and a head size limited to between 32mm and 36mm with a light blue background colour. The eye height is also required to be within red area. The person will need to display a neutral expression with mouth shut and eyes open. It is also required that face features including eyes, nose, mouth and eyebrows are fully visible. Since 2012, the new service of renewing passports online started to operate on the passport renewing government website and requirements of the photo are slightly changed. The online renewing service only accept photo size of 3-inch x 4-inch and the size of the file must be greater than 500kb and less than 10Mb. The government's website also introduced an auto photo checking system which checked whether the user's passport photo is up to standard that the image is not under/overexposed, size of file and photo. User will need to pass the checking to obtain a valid passport photo for the online renewing service.
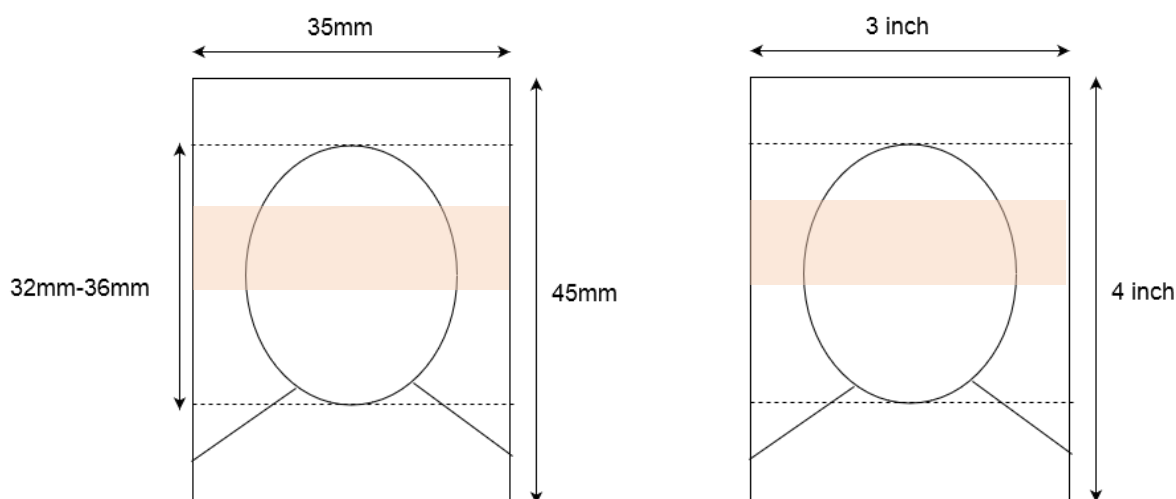
Figure 2.1.1 Size of photo for applying through paper application (left) and online service (right).

2.2 Framework

Web frameworks have been widely used in development of web applications for a variety of device platforms. Developers rely on frameworks for fundamental concepts and to create a standard working website without the need to implement it from scratch, thereby reducing the

overall development time. Front-end framework normally comes with packages of standardised code in HTML, CSS, JQuery, JavaScript languages which aims to ease the development of dynamic websites and web applications. Frameworks that offer various functionality types are determined by the purpose of a specific user application's structure. Regarding to different browser platforms emerging, most of the frameworks available today have solved problems regarding to cross-browser incompatibility.

2.2.1 Structure

Model-View-Controller (MVC) is prominent on web based applications. It offers separation of concern (SoS) in that it separates data model which represents the underlying logic in data structure and view which is the user interface as well as controller that represent the classes acting as a middleman for communication between the model and view. With this structural approach client side is clearly differentiated with server side which provides the benefits of code reuse and allow semi parallel application development as a result greatly reduced development time and complexity of code. In testing perspective, the relationship with components are clearly defined and articulated.

Three-tier organisation revolves around three layers: presentation layer (Client), business logic layer and data access layer (server). It looks similar to MVC however with this architecture, the client tier never contacts directly with the server tier. In large applications MVC is the presentation tier of an N-tier architecture.

2.3 Web Application Programming Interface (API)

Provide a way to connect computer software components. It facilitates interactions between code modules, application and backend IT system. The API specifies the way in which these different software components can interact with each other and enables content and data to be shared between components.

2.4 Front-end

Front-end development is also known as client-side development which the user can see and interact with the business services directly. A successful front-end requires the user to easily navigate through the website and find information or services that are relevant to their search. Current trends in front-end design has taken into account that devices, browsers and operating systems varies widely with respect to the user and thus requires careful planning on the side of the developer.

2.4.1 Design concepts

Design principles for a website are one of the key factors that affects the way people are attracted the website and greatly affects the probability of visitors staying or revisiting the

website. Thus the web traffic into the website. Websites are strategic issue for organisations to increase their revenue for their service

There is a general underlying concept to all successful websites that are simplicity, consistency, clarity of the design.

2.4.2 Programming concepts

Markup languages
It stores information and present them with various styling (fonts, sizes, colour, graphics) which the user can utilise. **HypterText Markup Language (HTML)** is one of the popular language that has been used for over 20 years to create web pages. It is simple to learn and easy to implement, however the problem with HTML is that new tags cannot be customized for developer's needs and communication with client/server is complex and not efficiently handled. **Extensible Markup Language (XML)** solves the HTML problem, which separates the presentation from content and allowing information to be in a ready state for passing between applications. XML are more popularly used in industries today.

Style sheet language
Cascading Style Sheets (CSS) integrates with HTML or XML to describe the presentation of web pages that includes fonts, layout, colours and effects. The primary advantage is that content and presentation are separated with formatting on a separate stylesheet therefore greatly improves the efficiency of implementing the website visual looks by reusing CSS rules across multiple pages. CSS can also render web pages to be correctly displayed on different devices and browsers.

Scripting Language
The use of scripting in web development adds interactivity and functionalities to enhance the website when added to HTML documents. **JavaScript (JS)**, also called the language of the Web, is the most popular programming language used in most if not all websites on the Internet today. Web developers often implement JS for its handy features for a number of reasons.

1. Simplicity: Comprises syntaxes that are close to English and the use of Document Object Model (DOM) for easy manipulation of elements and content.

2. Speed: Codes are executed on the client-side instead of the web server meaning the site is much more responsive with less processing strain on server.

3. Versatility: JS can be executed on web pages that uses other programming languages to extend functionalities.

However, JavaScript have issues with cross-browser compatibility so the display of the website varies between different browsers and become unpredictable to web developers. Security issue has also arisen in the use of JS on website which will later be discussed in the security part of the section. There are few other scripting languages such as Python, Perl provide the same functions of JS but in the end industries have converge to JS as the one platform that meets the ECMAScript standard specification.

2.4.3 Framework Application

The most commonly used front-end framework is **Bootstrap**. It is known for offering a lightweight, beautiful, intuitive web design kit that contains CSS styles, Grid classes and JS components as well as supporting a large range of basic responsive functionalities. Open source Bootstrap-only template are made available for public access which greatly simplifies the development process and complexity. The most important feature that makes it so popular is that developers are allows to modify and customise the code bringing the developers a touch of their own preference in design aspects.

AngularJS is a JavaScript MVC framework that is operating not only with HTML/CSS as well as data on the web pages. Comparing to early MVC approach of having all data manipulated on the server-side, AngularJS allows MVC components and application logic to execute partly on the client providing greater server performance. It automatically synchronises data from the view with the model through data-binding. Client-side navigation, validation and deeplinking with hashbang urls or HTML pushState are all important parts of providing a greater user experience. In testing phase, AngularJS comes with pre bundled test cases. Built in dependency injection feature allows a clear understanding of how application is wired and testing components can be easily replaced by another according to the developer's needs aligning with user's requirements. Without having to manipulate DOM directly and requests are simulated, test assertions can be tested in isolation.

jQuery was developed to change the traditional way people worked with JavaScript into a framework that allows manipulation of DOM at ease. On top of that libraries were released that offers many common scripting functions written in the common language JS. It has been optimized to work with a variety of browsers automatically.

Nowadays in which website is the main source of where people obtain information and more platforms for websites to be more of the frameworks have been developed to meet different criteria of what the developer and stakeholders look for.

2.5 Back-end

In general cases, the backend also called server-side consist of three primary parts: a server, an application and a database. These components interact by having an application which connects with a database to look up, store, update and return information to the user in the form of front-

end code. The processing of these actions happen in the server which the database runs on. From this it is obvious that the back-end is responsible to handle data and therefore need to look into aspects of protection, management and the structure of the data in order to provide a secure dynamic site that is updated in real-time.

2.5.1 Application Building

In terms of dynamic web pages, efficient communication between back-end components are essential. The language chosen needs to consider the database used, server platform and software, budget of the organisation and what the organisation value the most. There is no one platform that is the 'best' but the chosen language and design components should be the most suitable one in the organisation's perspective.

PHP Hypertext Processor (PHP) is a general-purpose server-side scripting language. It works great with HTML by embedding PHP codes that works interchangeably within the page. With PHP's high flexibility in terms of compatibility with operating systems, web servers and database, it has the freedom to choose which one suits best. The chief advantage is that it has a large open source community which people contribute in making PHP resource files for easier development and support in making the website that uses combinations of various web template systems, frameworks or content management. Therefore, PHP is a language that beginner programmers can pick up easily.

Ruby on rails is the framework that makes Ruby an easy choice for a back-end development language. The mature and stable framework has lessened development time of high quality products for client and more maintainable which is why many large companies uses Ruby. In comparing with PHP's large amount of framework which risk not having continuous support in the future, Ruby is a much safer choice in the long run.

.Net Framework
ASP.net is a set of technologies in the Microsoft .NET framework for building Web applications and XML Web Services. The approaches are Web forms, ASP.NET MVC and ASP.NET. Each of these approach will vary according to development style. Web Forms offers controls and components for building a UI-driven site with data access. MVC that offers clear separation of concerns. Web Pages working with Razor syntax for a fast and lightweight approach to combine server code with HTML. It differs from PHP in that it is on Object Oriented Programming (OOP) paradigm rather than a scripting paradigm.

2.4 Security measures

Website security is a major section to be focused on and planned carefully to prevent web vulnerabilities being exploited. There are common security approaches to mitigate or eliminate the possibility of an attack conducting on websites especially on that consist of valuable databases. The following are lists of ideas that should be used for user login process.

1) Login pages should be encrypted: The basic way is the use of SSL after user authentication which may work back 20 years ago but with technologies becoming more sophisticated, attackers' method to retrieve user's login details are easier as well. Encrypting the session after login is useful to prevent malicious security cracker to gain access to sensitive data.

2) Data validation should be done server-side: Web forms usually comes with data validation however this will not be enough to prevent attacks as the attackers can easily gain access to client-side and validate the data by themselves. If it is done at server-side then it is harder for attackers to gain access and compromise the server.

3) Key-based authentication is better over password authentication: password authentication is easier to guess if the user is thinking the password logically. Cryptographic key-based authentication is more difficult to crack, keeping authorized system to authorized users.

Studies have shown that browser based attacks are closely tied with JavaScript, which is a very popular scripting language for website development on the client's side. A great deal of attention has been paid to the JS related security vulnerabilities such as cross-site scripting that could directly lead to security breaches. Insecure JS practices may not necessarily result in direct security breaches, but could definitely cultivate the creation of new attack vectors. Insecure JavaScript inclusion and insecure JavaScript dynamic generation.

Insecure JavaScript inclusion is the practice of using the src attribute of a <script> tag to include a JavaScript file from an external domain into the top-level document of a webpage. An easy way to avoid this form of attacks formation is the separation of JavaScript code with HTML mark-ups as well as keeping external sources at a low number.

With JavaScript dynamic generation, it refers to unawareness of using dangerous JavaScript functions including the eval() to dynamically generate new scripts. It makes the website prone to attacks including cross-site scripting and cross-site request forgery which both are used to conduct malicious activities. The use of Document Object Model methods can replace the use of eval() to dynamically generate and execute (DOM) various JavaScript statements.

Chapter 3
Current work of idphoto.co.nz

In this section, idphoto.co.nz's website will be explained in detail, discussing the service flow and features of different aspects of the website. The front-end functional offers as well as the back-end database construction are explained in detail. The issues regarding to the previous website design are analysed and solutions are conducted to improve the not only the overall performance and also how the website is presented visually. Semester 1 project work progress is mainly focused on the front-end development and improvements made.

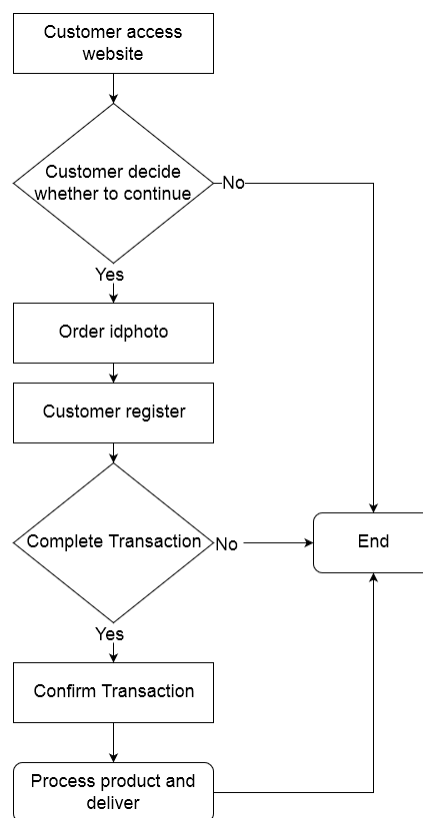## 3.1 Functionalities of front-end

Figure 3.1.1 Service flow

The primary service of idphoto.co.nz is selling standard New Zealand ID photos that is guaranteed to be valid to the application the customer is using the photo for. The main flow starts off that customers may or may not continue to start placing an order. Once the customer has decided to continue, the service displayed the form of application the customer is applying for (online/paper application). From this step, as soon as the user selected their option, the system has 2 cases:

1)With online form the photo order should be producing a standard 3-inch x 4-inch digitalised photo to the customer.

2) With paper application, the photo order should be producing a 35mm x 45mm size photo that is made available for printing and cutting into standard size.

With either one of the option, the customer will need to upload a portrait photo that meets our specified image requirement and agree to our terms of conditions in order to proceed to the next step. Then the customer will need to register an account with their personal details including name, email address, contact number and address. After successful registration upon entering valid information, the customer need to pay for the product through an online transaction system and depending on the outcome of the transaction, it will show as either of the two states:

1) Paid: Successful transaction, the order will be placed into the customer's account as well as the system database.

2) Not Paid: Unsuccessful transaction or an unexpected error, the order will still be placed into the customer's account with an unpaid status which the customer can choose to continue for payment in the future by logging into their account.

Any order details entering the system database will be orders that are already paid and be processed by our employee for producing the product.

Customer account

Upon creation of an account, the customer can login to view their order status and conduct actions according to their need.

Unpaid status: For unpaid case, the customer can either continue the transaction by selecting pay or edit the order (change type/ re-uploading photo) or delete the order.

Action	Payment status	Amount	Order date	Type	Uploaded photo	Your ID photo
Pay Edit Delete	Unpaid	-	-	-	-	-

Paid status: For paid case, the customer can view the current state of the ID photo and download the original photo file at her/his will but cannot make any changes once the order is paid.

Action	Payment status	Amount	Order date	Type	Uploaded photo	Your ID photo
-	Paid	-	-	-	Download	In process

Completed order status: For complete order case, the ID photo is uploaded to the customer's account and made downloadable to the customer. The original uploaded file is also downloadable as well. The customer can also place a feedback regarding to the order and the overall service for the website's future improvement references.

Action	Payment status	Amount	Order date	Type	Uploaded photo	Your ID photo
Feedback	Paid	-	-	-	Download	Download

3.2 System Implementation

The front-end module is developed in the integrated development environment (IDE) called Brackets. Brackets carry features and functionalities that made building website projects much more productive, eliminating redundant tasks that takes up too much time the development's daily basis. It supports HTML/CSS/JavaScript and additional language supporting through extensions. On top of that, quick edits in CSS are displayed clearly showing the corresponding HTML section. For my development I have used JQuery and Bootstrap css framework components for the UI of the website. Instant changes in the CSS implementation are reflected instantly on browser with the use of live preview that Brackets has offered.
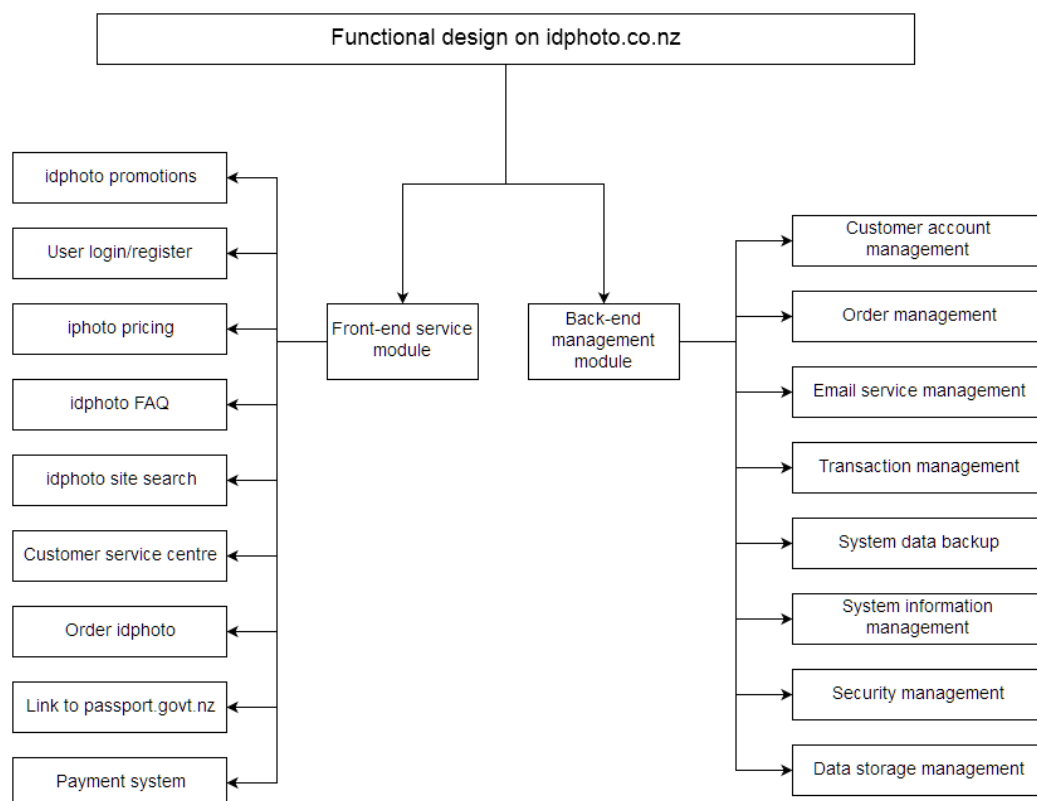

Figure 3.2.1 Functional design of idphoto.co.nz

The website pages will include all necessary information about the working of the idphoto.co.nz. Current promotions for ordering ID photo, pricing of the idphoto, FAQ page for the most popular questions are displayed, easy navigation throughout idphoto.co.nz are provided by the site map, a customer service centre page containing the organisation's contact details for customer query or issue solution, a external link button to the passport renewing online service for customer's convenience and the idphoto ordering service containing the online payment system.

For the back-end management module, the registered customer personal information, order details should be managed in databases, transaction management should be secure with log records for future references, system information should be up to date, system data should be backup at time intervals in case of some desaturase event or attack happening to the system server and erased all customer data and data cookie for users that did not complete their ordering progress or change of mind situation.

Back-end working flow

The employee is responsible for producing a valid NZ ID photo upon receiving an order for a certain type of application. They will be working with 2 image-editing softwares that is installed in the organisation computer. 1)Photoshop CS2. 2)CPAC Imaging Pro. The reason for using an older version of Photoshop is because the newest Photoshop CC have removed certain features such as the extract filter that is used to changing the background of the image. The use of CPAC Imaging Pro is for a faster performance of skin smoothing. First thing is to check whether the uploaded photo meets image requirement rules. This image requirement is specified in the way that would decrease the probability of making an ID photo that does not pass the checking of the government's photo checking system and also the effort for employees to spend adjusting the uploaded image before applying SOP. For example, the person's hair would be too puffy to fit within the specified size of the head and eye height. In this case, the person's hair will need to be retouched on to make it smaller to the point that everything fits to standard. This action could take some time and it will not be reasonable to be spending a relatively long time on an order that can be prevented through the customer her/himself by following the requirements like tying up hair. The later step is going through the SOP for producing the ID photo.

SOP:
1. Check for fitting and crop to size with the right resolution.
2. Apply CPAC Imaging Pro smoothing skin effect.
3. Double check with the resolution and size of the photo.
4. Upload photo to the customer's order database.

3.3 Improvements made to previous design

Front-end
The colour scheme has been recombined, pairing dark blue and white/grey colour and minor orange/light blue elements for a more vibrant and professional look to the website.

Content layout issue: The flow of the page was arranged in a way that was confusing and not easy to follow. Upon accessing the website, the user is given the description requirement for applying for a certain type of service. From the user's point of view, presented a chunk of work description in the beginning not only decreases the overall simplicity of the website and also force the user to read something that might not be necessary for them.

Solution: Upon entering the website, the requirement descriptions are displayed according to user's hovering selection. This way, the descriptions are displayed when required and keeping the overall website clean and simple. The buttons are replaced with image icons for easy identification of the service selections.
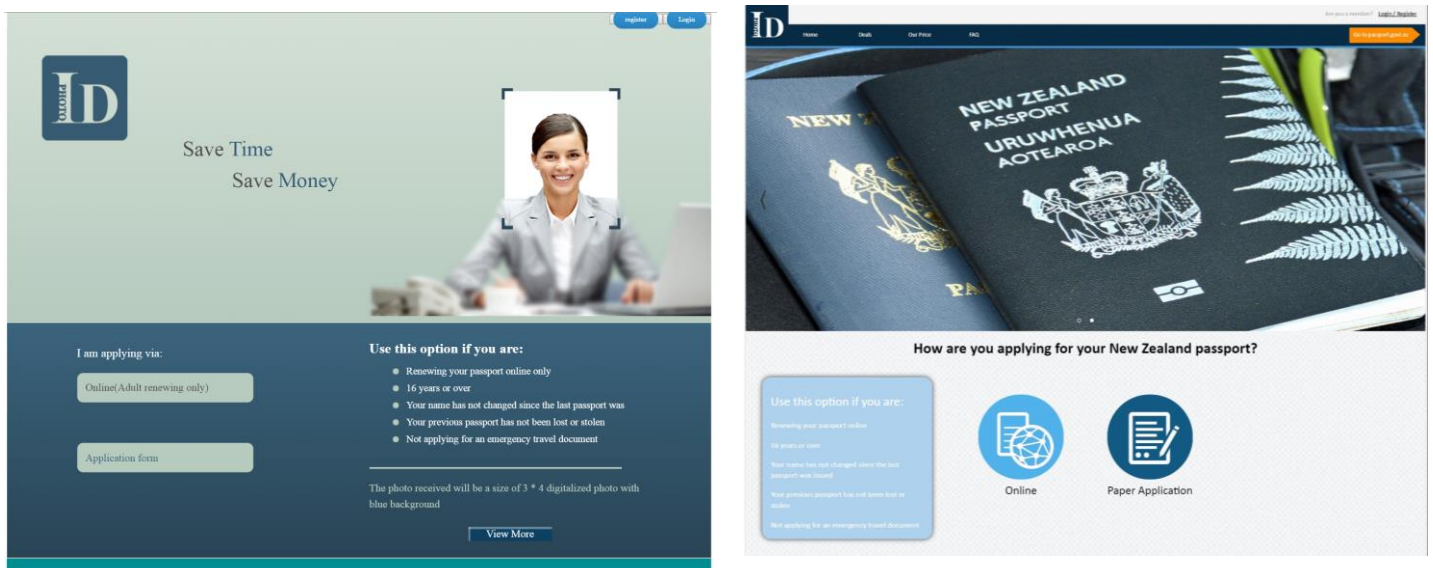

Figure 3.3.1 Screenshot of the previous design and the hovering effect of the new design.

Stateless issue: Once the user start placing their order, if they happen to close the window by accident before reaching the submit button on the customer register step, the data entered before is lost and the user will need to start again. If the user want to visit other pages during their order placement, hitting the back button will return them to the page where the information was entered before but start again with the order placement process.
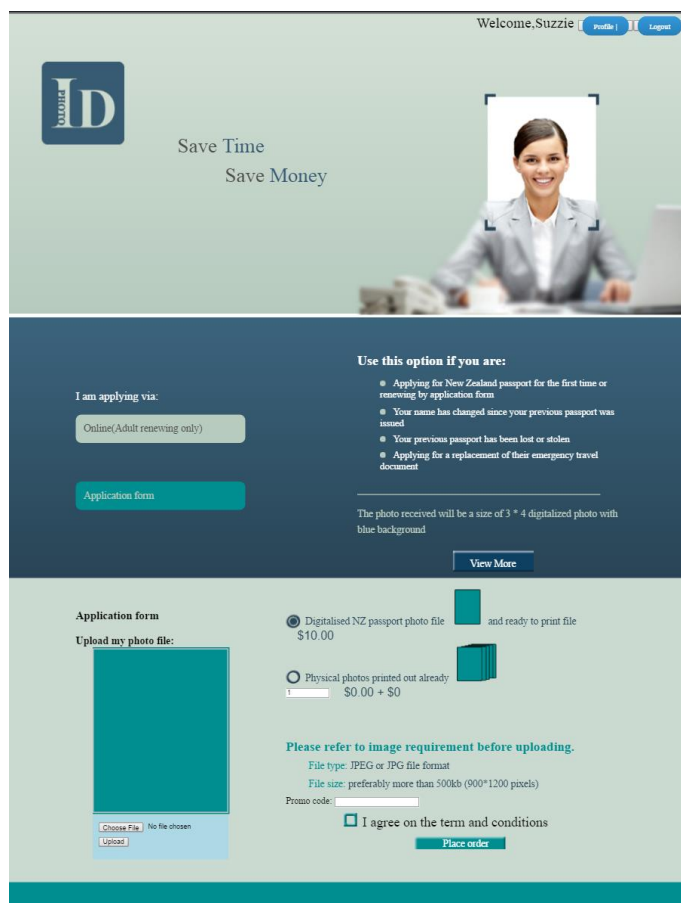
Solution: Implementing cookie for remembering stateful information that is able to record user's browsing activity and the information user have previously entered into form fields.

Redundancy in web page files: A new web file is created when there is a click event to the next ordering section. The new web file is being reused in other web files and web files are loaded for every click event. Although the performance difference is almost negligible however, the slight flicking from reloading pages is still visible from the user's view.

Solution: There are 2 ways to approach this.

1) With one html file, provides a smoother UI transition by changing what's different with JavaScript. There is no page refresh, no flicks as compared to the previous design.

2) AngularJS, it breaks dynamic elements into partials, and even with multiple separate files, it makes them all concise together, and there is no need for JavaScript at all.

I have gone for a one html file rendering with JS as it is more understandable working with JS comparing to my limited knowledge to AngularJS coding syntaxes. Specific tags required to write in order to use certain functions.


```javascript
function nextStep(elmnt){

    document.getElementById("passportselect").style.display="none";

    document.getElementById("onlineForm").style.display="none";
    document.getElementById("appForm").style.display="none";

    if(elmnt.className=="icon-online")
    {

        document.getElementById("onlineForm").style.display="block";

    }else if(elmnt.className=="icon-app"){

        document.getElementById("appForm").style.display="block";

    }else if(elmnt.id=="onlineNext"||elmnt.id=="appNext"){
        document.getElementById("upload").style.display="block";

    }
}
```

Figure 3.3.2 The left are 3 different web files being connected together upon click event. On the right is the use of JS to manipulate between page clicks.

Navigation improvement:

A progress bar will be implemented to the website's order placing flow. The user is able to navigate to previous stages to change details and navigate back to where they left off.

Chapter 4
Future work

4.1 Plan of action

I will be focused on finishing off the front-end design before the start of semester 2. In the upcoming semester, I aim to gain a thorough understanding of the existing database and weighing pros and cons with alternatives to approaching to security implementation. Understanding aspects of how to protecting customer's sensitive data and prevent violation of consumer privacy acts. There are a fair bit of existing PHP coding for back-end implementation and with my lack of knowledge in PHP, it is expected to be a really time consuming process to understanding the existing code therefore It is possible that implementing a new database is a better option. Security aspects should be integrated along with the development of the database. After the development of back-end and security, test cases should be created for evaluating the readiness of idphoto.co.nz going online. By the end of the year I hope of a successful launch of idphoto.co.nz and continue to make improvements through feedbacks from customers.

4.2 Additional features

Auto check for photos

Current method to verify whether the customer's uploaded photo is valid to be made into a proper NZ ID photo that passes the government's online checking system and officials is my manually checking the photo and the hope of the customers follow the guidelines specified in the image requirement. If there is a way to implement an automated checking for the photos the user uploads, then the efficiency of the service will be improved a lot. From some research papers studied, it may be possible to implement a sensor web which is based on quick query algorithm and digital image processing (DIP) technique. DIP can be used in detecting differences in the standard image in terms of contrast level, file size and whether the person's head fits into a certain region of the image. Each one of these aspects have a standard level presented as a numeric value, when tested image is processed, the pixels will give the numeric data and is stored in a database which is compared to the standard level. If the data falls out of the specified level using some algorithm then the output will give out a not pass to the image, vice versa.
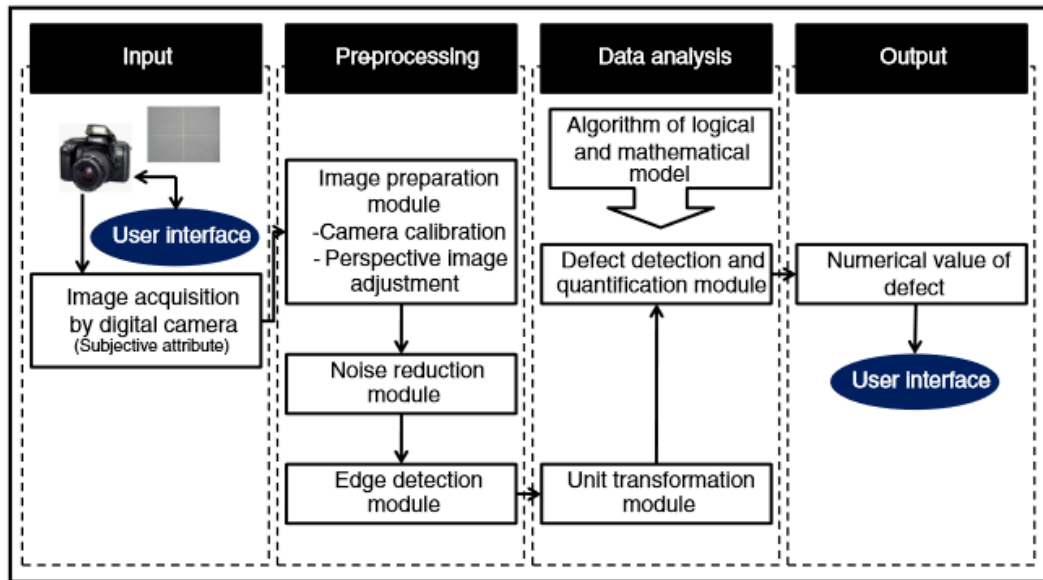
Figure 4.2.1 Stages of defect detection and quantification system.

References

AngularJS. (n.d.). Retrieved June 7, 2016, from https://docs.angularjs.org/guide/unit-testing

Bi, C. G., Xia, L. B., & Lu, Y. M. (2014). *Based on bootstrap and wordpress to develop personal blog* doi:10.4028/www.scientific.net/AMM.687-691.2423

Costa, P. (2008). Evaluating web site design. *SIGDOC 2008 - Proceedings of the 26th ACM International Conference on Design of Communication,* 265-266. doi:10.1145/1456536.1456591

Ferguson, N. (31 May, 2016) What's the difference between frontend and backend? Retrieved from http://blog.careerfoundry.com/web-development/whats-the-difference-between-frontend-and-backend

Media college.com. (n.d.). Retrieved June 7, 2016, from http://www.mediacollege.com/internet/javascript/pros-cons.html

Microsoft ASP.NET (n.d.). Get started with ASP.NET. Retrieved June 7, 2016, from http://www.asp.net/get-started

Microsoft Developer Network. (n.d.). Maintaining Session state with Cookies. Retrieved from https://msdn.microsoft.com/en-us/library/ms526029%28v=vs.90%29.aspx

PHP (n.d.). Retrieved June 7, 2016, from http://php.net/manual/en/introduction.php

Technology Conversations. (n.d.). Retrieved June 7, 2016, from https://technologyconversations.com/2014/07/10/server-vs-client-side-rendering-angularjs-vs-server-side-mvc/#comments

Toll, W. (5 June, 2014) Top 48 Integrated Developer Environments (IDEs) & Code Editors. Retrieved from https://blog.profitbricks.com/top-integrated-developer-environments-ides/

Toshniwal, R., & Agrawal, D. P. (2004). Tracing the roots of markup languages. *Communications of the ACM, 47*(5), 95-98. doi:10.1145/986213.986218

Yalezo, S., & Thinyane, M. (2013). Architecting and constructing an SOA bridge for an MVC platform. *Proceedings - 2013 4th World Congress on Software Engineering, WCSE 2013,* 45-49. doi:10.1109/WCSE.2013.11