National College of Ireland

Year 3 (BSHC3B, BSHTM3)

2017

|  |  |  |
| --- | --- | --- |
| **Member:** | **Student Number:** | **Online names:** |
| **Jessica Bankable** | **x15766059** | **Jbankole22** |
| **Charlene Moore** | **x15412048** | **charlene2225** |
| **Keith Feeney** | **x15015556** | **keithfeeneyNCI** |
| **Angela Darel** | **x15021386** | **Evilrose, Anayela** |

****

GOLDEN YEARS

Technical Report



**Table of Contents**

[Executive Summary 4](#_Toc501811284)

[Definitions, Acronyms, and Abbreviations 5](#_Toc501811285)

[1 Introduction 6](#_Toc501811286)

[1.1 Background 6](#_Toc501811287)

[1.2 Motivation 6](#_Toc501811288)

[1.3 Project Overview 6](#_Toc501811289)

[1.4 Target group 7](#_Toc501811290)

[1.5 Technologies 7](#_Toc501811291)

[1.6 Distribution of tasks 7](#_Toc501811292)

[System 9](#_Toc501811293)

[1.7 Requirements 9](#_Toc501811294)

[1.8 User Requirements Definition 9](#_Toc501811295)

[1.9 Requirements Specification 9](#_Toc501811296)

[1.10 Functional requirements 10](#_Toc501811297)

[1.10.1 Use Case Diagram 10](#_Toc501811298)

[1.10.2 Requirement 1: Populate Event 10](#_Toc501811299)

[1.10.3 Requirement 2: Search Events 14](#_Toc501811300)

[1.10.4 Requirement 4: Create Customer Account 17](#_Toc501811301)

[1.10.5 Requirement 5: Create Business Account 19](#_Toc501811302)

[1.11 Non-Functional Requirements 21](#_Toc501811303)

[1.11.1 Performance/Response time requirement 21](#_Toc501811304)

[1.11.2 Availability requirement 22](#_Toc501811305)

[1.11.3 Recover requirement 22](#_Toc501811306)

[1.11.4 Robustness requirement 22](#_Toc501811307)

[1.11.5 Security requirement 23](#_Toc501811308)

[1.11.6 Reliability requirement 23](#_Toc501811309)

[1.11.7 Maintainability requirement 23](#_Toc501811310)

[1.11.8 Portability requirement 23](#_Toc501811311)

[1.11.9 Usability requirement 24](#_Toc501811312)

[1.11.10 Reusability requirement 24](#_Toc501811313)

[2 GUI 25](#_Toc501811314)

[2.1.1 System Architecture 29](#_Toc501811315)

[2.1.2 System Evolution 30](#_Toc501811316)

[3 Discussions and reflections 32](#_Toc501811317)

[3.1 Learning, Skills & Process 32](#_Toc501811318)

[3.2 Client Feedback Session 34](#_Toc501811319)

[4 Conclusion and Future Work 35](#_Toc501811320)

[5 Further Development 36](#_Toc501811321)

# Executive Summary

This Project is about creating an application of some sort that is relevant today. It could be an Android Application, a Web application, or even a Game application etc. As a group we decided that we wanted to create a Web Application. After conducting a lot of research and creating surveys and having them completed by our elders and consulting with older seniors, our team found out, that there were not much of activities set aside especially for them to take part in.

The solution to this, is the creation of “Golden Years”. This Web Application addresses this niche in the market. It offers much support and guidance for elders, to ensure they live life to the full, no matter what age they are. Application itself will provide the booking through our website offering advertisement for the company and fun and exciting events for customers.” Golden Years” financial plan would be to charge companies for using our website to advertise their events to grow as a company. Part from that, the application would serve as link between transport services and events, ensuring the elderly can travel to the events of their desire. Other services provided by “Golden Years” include but are not limited to: medical guidance on what events we feel suitable for certain health conditions, connections to learning facilities for those who cannot read, write, use technology, swim etc.

## Definitions, Acronyms, and Abbreviations

|  |  |
| --- | --- |
| WA | Web Application |
| GY | Golden Years |
| App | Application |
| OAP | Old Age Pensioner |
| IDE | Integrated development environment |

# Introduction

“Golden Years” is created for those, who would like to find activities close by to their home to enjoy with others, to socialise. Developers aim is to encourage the elderly to not stop living their lives because of their age. This was where company name stemmed from “Golden Years”. Meaning years to cherish and treasure. Application itself is designed to be easy to use by people with little technical knowledge. Our app offers all information that is needed in order to try something new, Medical advice, Transport, Activities and Education (if user need to learn something before participating). There are also plans to simplify the application as much as possible, to ensure ease of use for the users with little technical skills, vision impairment, manual problems etc.

## Background

Original idea was to create a booking website for all sorts of events. One of the group members proposed to concentrate the Project around elderly users, who are often neglected in terms of internet usage. Results of the research were clear: there is many booking websites available already, none of which offers full package for the senior citizens and none of the covers their needs or desires. GY team came up with conclusion, that this is a good market niche that still needs a lot of improvement because are already covered (bookings) and that still have plenty of space for innovations (old fashion dancing, medical trips etc.)

## Motivation

Due to conducted research and clear space for improvement in OAP space of the overall applications market, the group decided to go forward with the idea and developed web app, called “Golden Years”.

## Project Overview

The project lifecycle consists of a total of 13 weeks. During these weeks, following tasks and activities were taking place: coming up with the idea, conducting research, developing project proposal, requirements specifications, mid-point presentation and finally developing the project that we felt relevant for today’s market. The Gantt chart included with this report shows this in more detail.

## Target group

“Golden Years” application is aimed for users age 55 and over (OAP). Due to specific target market, GY team recognize secondary target group, which would consist of family and friends of OAP’s. As secondary market, those users can book or pay for activities and services offered by GY, on behalf of senior users that app is originally designed for (e.g. family booking and paying for grandparents dancing evenings).

## Technologies

To develop the application, following programming languages were implemented:

JavaScript, HTML, CSS, Java, Python, PHP, MySQL and PayPal Developer. Initially team members were using different programming languages and different platforms : MySQL, Notepad ++, Visual Studio. Later on all work was placed on Cloud 9 and further developed using it as primary platform. At the later stages bootstrap was implemented and web application was based on it, to ensure seamless responsiveness.

To book an event, the website sends the event title and price to PayPal and PayPal deals with all payments. An email is sent to the (Golden Years) company with booking information on completion of the order.

## Distribution of tasks

**Charlene:** As Charlene was not the best coder, Charlene did a lot of research to figure out how to do a lot of the coding. Charlene worked on the Login system and worked on sessions and cookies. Charlene and Keith worked on connecting the website to the Database. Charlene also worked on the general aesthetics of the website.

**Jessica:** Jessica worked on the Payment System originally and did a lot of research on this. Jessica tried to get it working over many attempts but was not successful. The group decided to use PayPal instead. Keith and Jessica worked on the Payment system until it was completed. Jessica also worked on the general aesthetics.

**Keith**: Keith created the database originally in MySQL Workbench. Keith used PHP to connect the website to the database with the help of Charlene. This was difficult, and Keith struggled to get it working. Keith assisted Charlene in general coding and the aesthetics of the website.

**Angela:** Angela created the mock-ups for the website. Angela also completed a very basic version of the website. This was for the Mid-Term Presentation. Angela created a version of the Booking system that was not used in the final version of the website. Angela helped with general aesthetics of the website.

# System

## Requirements

## User Requirements Definition

The client has requested a Web Application that they wish to use on touch screens in public areas, such as O’Connell Street.

They have requested that the Web Application target is for people over the age of 65, but that anybody can use the Application.

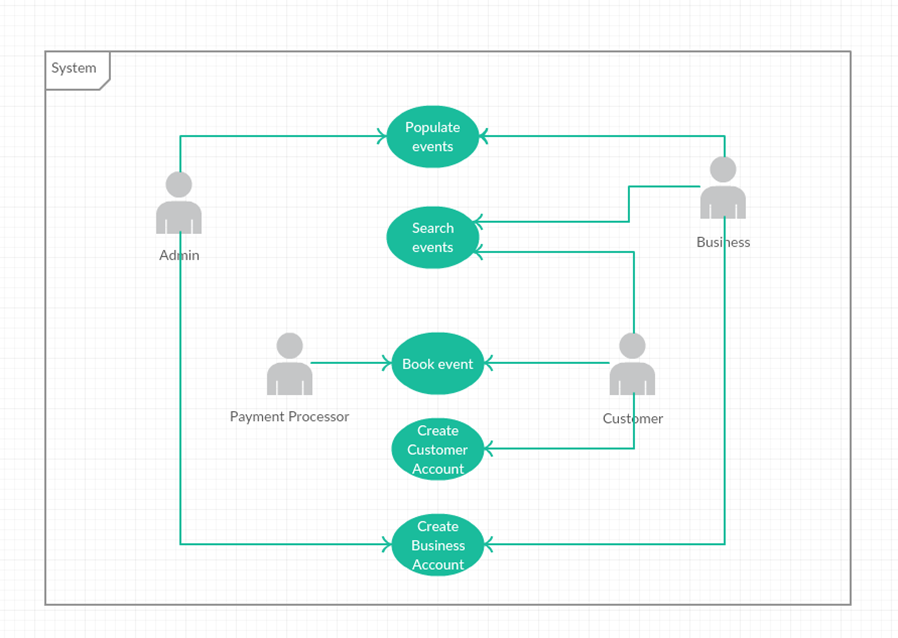
The client has specified that the Web Application is for creating and booking events, but has the capabilities of any other sections such as Medical advice or Education.

## Requirements Specification

An Admin must have an understanding of Privacy and Security. Admins must have a basic computing knowledge, specialising in the Windows OS. Training will be provided to Admins, that will not exceed one hour.

## Functional requirements

### Use Case Diagram



### Requirement 1: Populate Event

This is where a Business can add an Event so that users can select one.

#### Description & Priority

This is a vital part of the Web Application. These Events will be added by the Business so that a Customer can select them and book them. Without this requirement, the application would not work

#### Use Case

Populate Event

**Scope**

The scope of this use case is to for a Business to create Events on the Web Application. When created, the Business can create a new Event or edit their own Events or view other public Events that have been previously added and approved.

**Description**

This use case describes how a Business can add an Event by inputting relevant data into text fields, or selecting the likes of a calendar and time drop-down menu. All new Events must be validated by an Admin (for likes of payment, profanity, etc.).

**Flow Description**

**Precondition**

* Admins must approve new events before they are live on the Web Application.
* Admins are always logged in
* Admins must check new Events for profanity, errors and misspellings and/or confirm changes with Business.
* Business have previously created an account.
* The communication between the Server/Internet and the Web Application must be connected (via mobile Wi-Fi).
* If required, the Server and Web Application need to have the latest updates.
* When an Event is “live”, this means that the event is public for the Customer.
* Many pages exist within the app
* A Database stores the events.

**Activation**

This use case starts when a Business wishes to sign in to their previously created account and create a new event.

**Main flow**

1. The System is currently in a wait state on the Main Page, waiting for a button to be tapped. The buttons are; “Log in/Sign up”, “Continue as Guest” and “Business Log in/Sign up”.
2. The Business selects “Business Log in/Sign up”.
3. The Business enters in their username and password.
4. System displays buttons “Create new event”, “Edit event” and “Search live events”
5. The Business selects “Create new event”.
6. System displays “New event page”. The page contains text boxes for the Business to input data.
7. The Business inputs the title.
8. The Business selects how many places are available. If left blank, it’s unlimited.
9. The Business inputs the Price of the Event.
10. The Business selects whether payment can be made at Event. (Yes/No)
11. The Business enters Price
12. The Business inputs the address.
13. The Business checks the Google Maps marker and edits it if necessary.
14. The Business inputs the phone number and/or email for event enquires.
15. The Business selects a date from a calendar.
16. The Business selects a time from a drop-down menu.
17. The Business selects whether Transport is included (Yes/No option)
18. <See A1>
19. The Business selects “No” option.
20. The Business selects Save. <See E1>
21. System sends data to Database on the server.
22. System displays message “Your Event has been received and will be checked by an admin before going live”.
23. The Business can log out or browse their own events or other public events.
24. Admin receives notification there’s a new event.
25. Admin checks event for profanity, spelling errors, etc.
26. Admin checks for payment from Business (outside System)
27. Admin approves Event provided there’s payment (outside system).
28. Event is live.

**Alternate flow**

A1 : Transport selection is “Yes”

12. The Business inputs Transport company.

13. The Business inputs pickup location and time.

14. The Business inputs drop-off location (if different from event)

15. The Business inputs return pickup location (if different from event) and time.

16. The Business inputs return drop off (if different from original pick up location).

17. <Returns to Step 13 in Main flow>

**Exceptional flow**

E1 : Network connection lost when Business is creating Event

1. The System is unable to connect to the Server or the Internet due to (e.g.) Server being upgraded.
2. The System saves the current page and the inputs from the Business.
3. The System send an email to Admin with error message.
4. Business is logged out by System.
5. Network connectivity returns.
6. Business logs back in.
7. System displays the page the Business was on.
8. <Returns to Step 12 in Main Flow>

**Termination**

This use case is terminated when the Admin has posted the Event.

**Post condition**

The system goes into a wait state and is ready for input, or for the Admin or Business to sign out of their account.

### Requirement 2: Search Events

#### Description & Priority

Search Events is where a Customer can search for events within the Web Application. This is a very important part of the Application, as without it, Customers would not be able to proceed to booking events.

#### Use Case

Search Events

**Scope**

The scope of this use case is allowing Customers to search for events by searching via text and Google Maps. When the Customer selects an Event they wish to attend, the use case ends.

**Description**

This use case describes the how the Customer will input text and/or select a location from Google Maps. Nearby events will show on the map for the Customer to select.

**Flow Description**

**Precondition**

* The Customer is logged in
* Guest account is defined as a Customer. Guest accounts are given the username “Guest” followed by a random, unique number, e.g. Guest5256
* The network connectivity between the Web Application and the Server and/or Internet is connected.
* The user’s location is pre-set
* “Search page” is where the System shows the following buttons: “Search events” and “Your events”. Other buttons are present but not important for this Use Case.
* “Maps page” is where an embedded, interactive Google Maps map with markers or nearby events is shown along with a short list of the most popular apps.
* “Event page” is where information about an Event is shown for the Customer. It also contains “Return” and “Book” buttons.
* It is assumed that the Customer can only book one Event per booking or “transaction”.

**Activation**

This use case starts when a Customer taps “Search Events” within the Application.

**Main flow**

1. The System identifies that the Customer is logged in and that the network remains connected.
2. System displays buttons on “Search page”.
3. The Customer taps “Search events”.
4. The System changes page to “Maps page”. <See E1>
5. Customer selects an event from the list or select a marker on the map.
6. System shows Event Page.

<See A1>

1. Customer selects “Book”. Customer has successfully searched for an event

**Alternate flow**

A1 : Customer does not like Event, searches for another one.

1. The Customer selects “Return”

<Returns to Step 5 in Main Flow”

**Exceptional flow**

E1 : Google Maps server down

1. System displays message “Cannot connect to Google Maps”.
2. System removes Map and gives list of local towns/cities.
3. System keeps checking for Google Maps connection

<Returns to Step 5 in Main Flow>

**Termination**

When the Customer selects “Book”, this Use Case ends.

**Post condition**

The System waits for Customer input.

### Requirement 4: Create Customer Account

#### Description & Priority

Customer account is where a Customer can create an account. It is not necessary for a Customer to create an account, but it is a useful tool.

#### Use Case

Create Customer Account

**Scope**

The scope of this use case allows Customers to create an account.

**Description**

This use case describes the how the Customer is able to create an account. If the Customer is a Guest, they can create an account from any page, e.g. Events Page, Booking Page, etc.

**Flow Description**

**Precondition**

* The network/Internet is connected.
* Create Customer Account page is a page where a Customer can input their information.

**Activation**

This use case stars when a Customer selects “Sign up” from the main page.

**Main flow**

<See A1>

1. Customer selects “Sign up” from the Main Page.
2. System displays “Create Customer Account” page.
3. Customer inputs their name (required)
4. Customer inputs their address, city
5. Customer inputs their email and/or phone number (one required)
6. Customer selects “Save”
7. System sends data to database.

<See E1>

1. System sends password and confirmation to Customer’s email and/or phone.

**Alternate flow**

A1 : Customer is a Guest and wishes to create an account from any page.

1. Customer selects “Sign up”.
2. System saves current page in current state.
3. System displays “Create Customer Account” page

<Returns to Step 3 in Main Flow>

1. System displays saved current page.
2. System deletes saved page from memory.

**Exceptional flow**

E1 : System cannot communicate with Database.

1. System displays message “Sorry, an error occurred!”
2. Customer clicks “OK”.
3. System returns to Main Page. (Account cannot be created)

**Termination**

When the Customer has created an account, this Use Case ends.

**Post condition**

The System waits for input.

### Requirement 5: Create Business Account

#### Description & Priority

Business account is where a Business can create an account. If a Business wishes to create an Event, a Business Account is required.

#### Use Case

Create Business Account

**Scope**

The Scope of this Use Case is to allow a Business to create an account, so that they can Create Events or Search live events.

**Description**

This use case describes the how the Business is able to create an account. All Business Account creations need to be approved by an Admin, for security and permission security.

**Flow Description**

**Precondition**

* The network/Internet is connected.
* Create Customer Account page is a page where a Customer can input their information.
* Admins must approve new Business Account
* Admins are always logged in
* A Business Account must be created from the Main Page and cannot be accessed when logged in as a Customer (or Guest)
* Business Sign Up Page is where a Business inputs their information. This is different from a Create Customer Account Page.

**Activation**

This use case stars when a Business selects “Business Sign Up”.

**Main flow**

1. Business selects “Business Sign Up”
2. Business inputs Company Name
3. Business inputs Representative’s name
4. Business inputs Representative’s email and/or phone number
5. Business selects Save

<See E1>

1. System sends data to Database.

<See A1>

1. Admin approves Business Account
2. System sends password and confirmation to Business’s email and/or phone.

**Alternate flow**

A1 : Admin does not approve Business Account.

1. Admin does not approve Business Account
2. System sends email/SMS to Business to inform that their Business Account was not created.

**Exceptional flow**

E1 : System cannot communicate with Database.

1. System displays message “Sorry, an error occurred!”
2. Business clicks “OK”.
3. System returns to Main Page. (Account cannot be created)

**Termination**

When the System sends communication to the Business, this Use Case Ends.

**Post condition**

The System returns to the Main Page and waits for input.

## Non-Functional Requirements

### Performance/Response time requirement

There are no standards for response time, therefore Golden Years should response within 300-500ms. Ideal page response/ loading time would be 200-300ms but 500 is still acceptable for the purpose of the project.

### Availability requirement

Application run time would be 24h, also booking system would run 24h. As application presents users with both- events information and booking system, it should be available throughout bank holidays and any other holidays.

Golden Years should be available to access from any device, connected to the internet with wire or via Wi-Fi.

### Recover requirement

Recovery time after a downtime, shouldn’t be longer than 24 hours. Maximum accepted offline time - RTO (recovery time objectives) shouldn’t pass 48h. RPO (recovery point objective) ideal time shouldn’t pass 8h, as this would be the maximum accepted time when data can be lost from Golden Years.

Backup for inserted transaction data should be every 5 minutes since the payment information is sensitive and will also rely on payment systems such as bank or PayPal.

Recovery and availability requirement are connected in terms of creating backup server and creating back up data, to ensure fluent run of the server in the case of major disaster such as system down, power down, and the like.

### Robustness requirement

Listener should reject API requests, that are invalid.

Unexpected flood of requests (happen due to errors or attack), should be resisted by listener.

Listener or database sessions shouldn’t be able to create amount of processes exceeding or reaching maximum session limit. It should limit them instead.

Application should terminate and provide error message whether an error occurred (i.e. lack of connection to database) or application is not available (i.e. due to chosen payment method issues – bank application maintenance).

Database should be protected from reaching session limit.

### Security requirement

Security of application should be incorporated into its developing process. Good authentication process (possibly tied to user ID and/or IP), encrypting and backing data and databases, access protection for data and databases, right handling of the session etc. Login and password should be encrypted, same as sensitive data such payments. Timeout message should pop up on the screen after 5 minutes of user inactivity. There should be confirmation timeout button, so the user is aware and can log back in before performing another action.

### Reliability requirement

In order to provide reliability of the application:

• Time between application failures should not exceed 1 per 6 months.

• Components failure should not occur more than 1 per 6 months.

• Possible errors inducted by maintenance of the system, should not exceed 2 per year

### Maintainability requirement

Golden Yeas will go through testing faze before launching. Testing would ensure discovery and elimination of possibly occurring errors (errors can occur in design, code or logic level of application).

Support after the launch would be handled by both the Support Team and the Development Team for a period of 1 month, to allow for a fast response and fixing of any core issues. After that period support queries should be handled solely by the Support Team.

### Portability requirement

Golden Years is an online based application. Therefore, it requires an internet connection to ensure proper functioning. It's not a standalone application and can' be for example downloaded and stored (i.e. on USB drive).

### Usability requirement

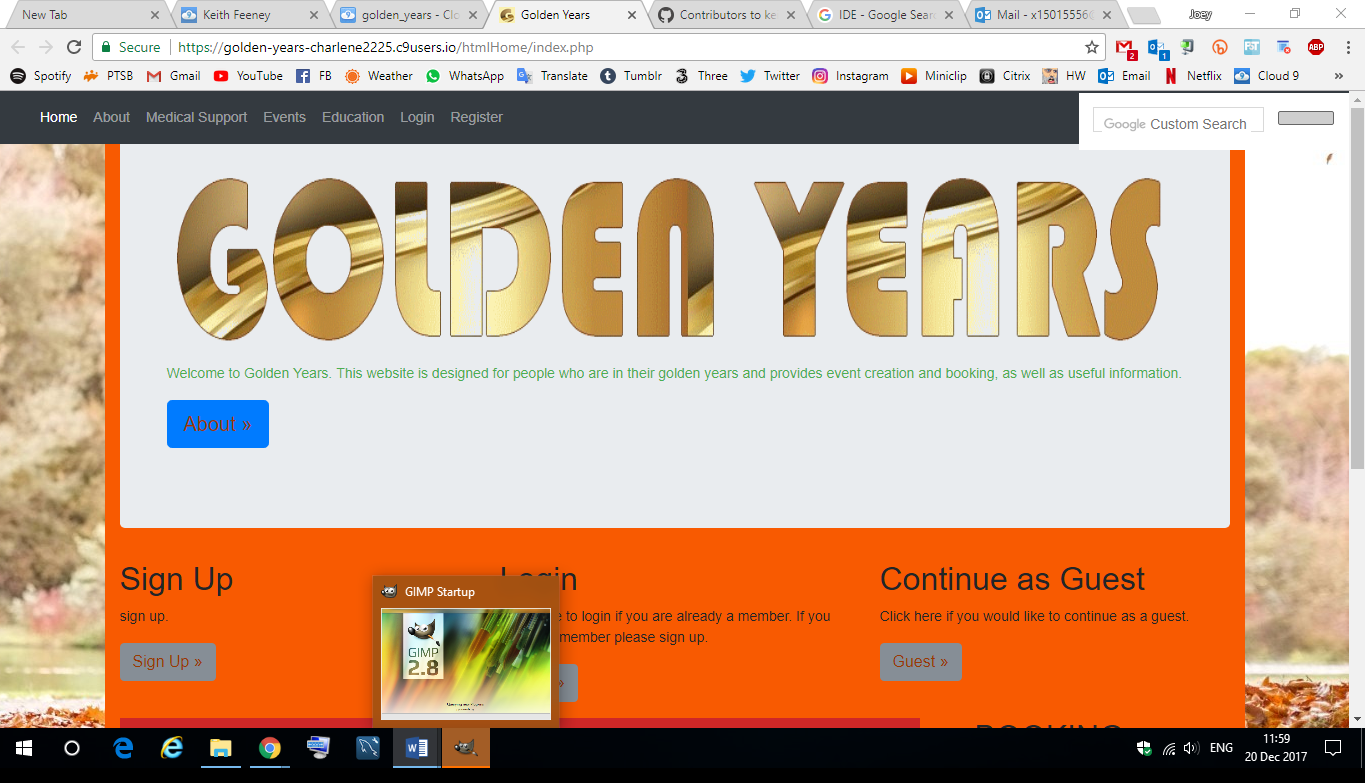
As special purpose application, that is to server senior users, the application should have clear and simple interface. Also, to ensure clear view and ease of the usage, Golden Years content should be presented with bigger than standard font size and contain big, easy to see and click buttons.

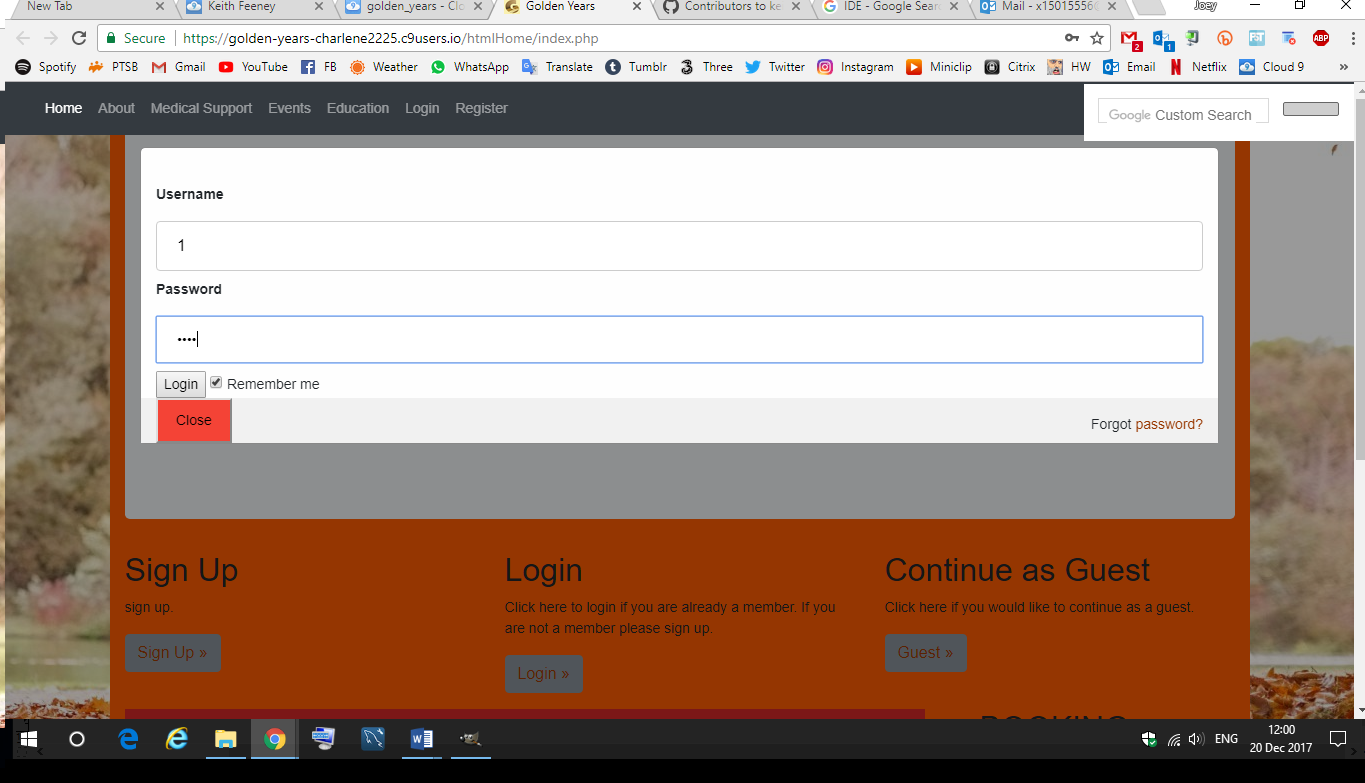
### Reusability requirement

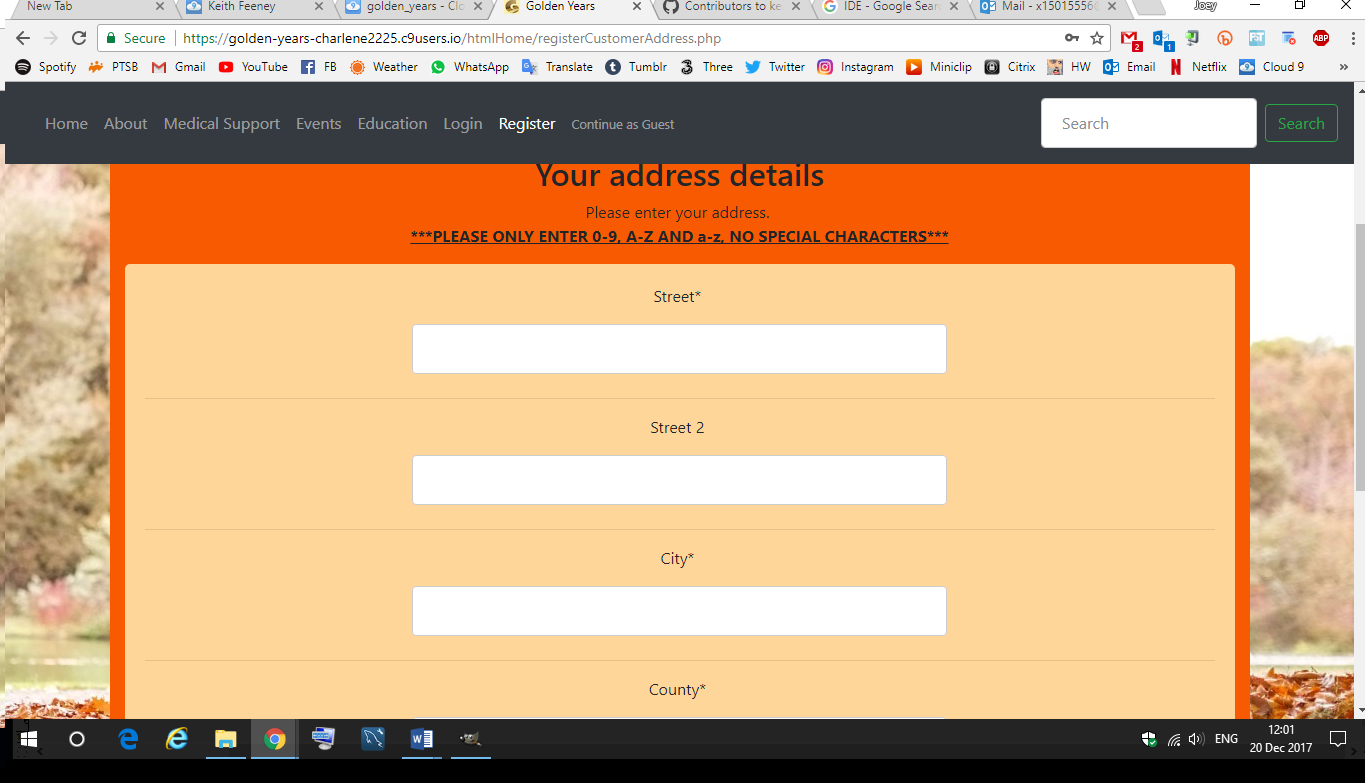
As special purpose application, that is to server senior users, the application should have clear and simple interface. Also, to ensure clear view and ease of the usage, Golden Years content should be presented with bigger than standard font size and contain big, easy to see and click buttons.

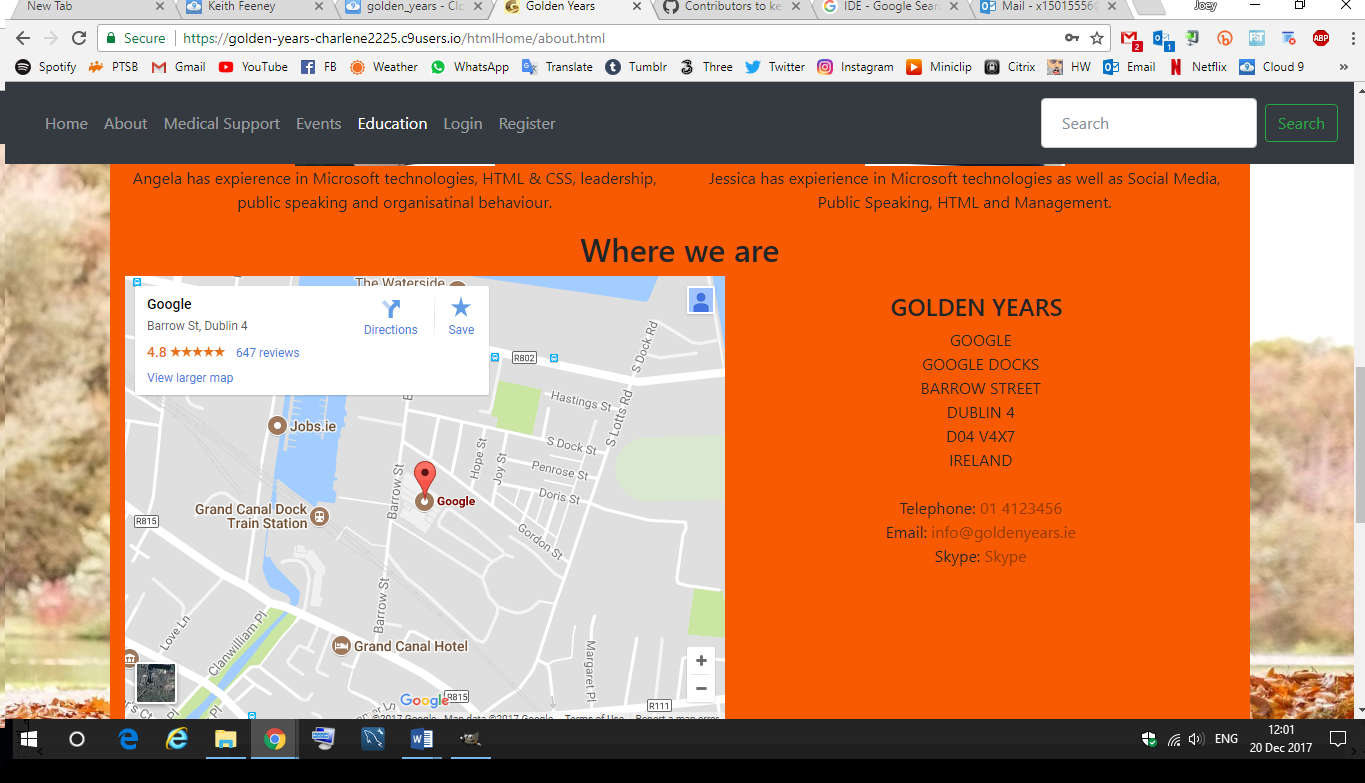
# GUI

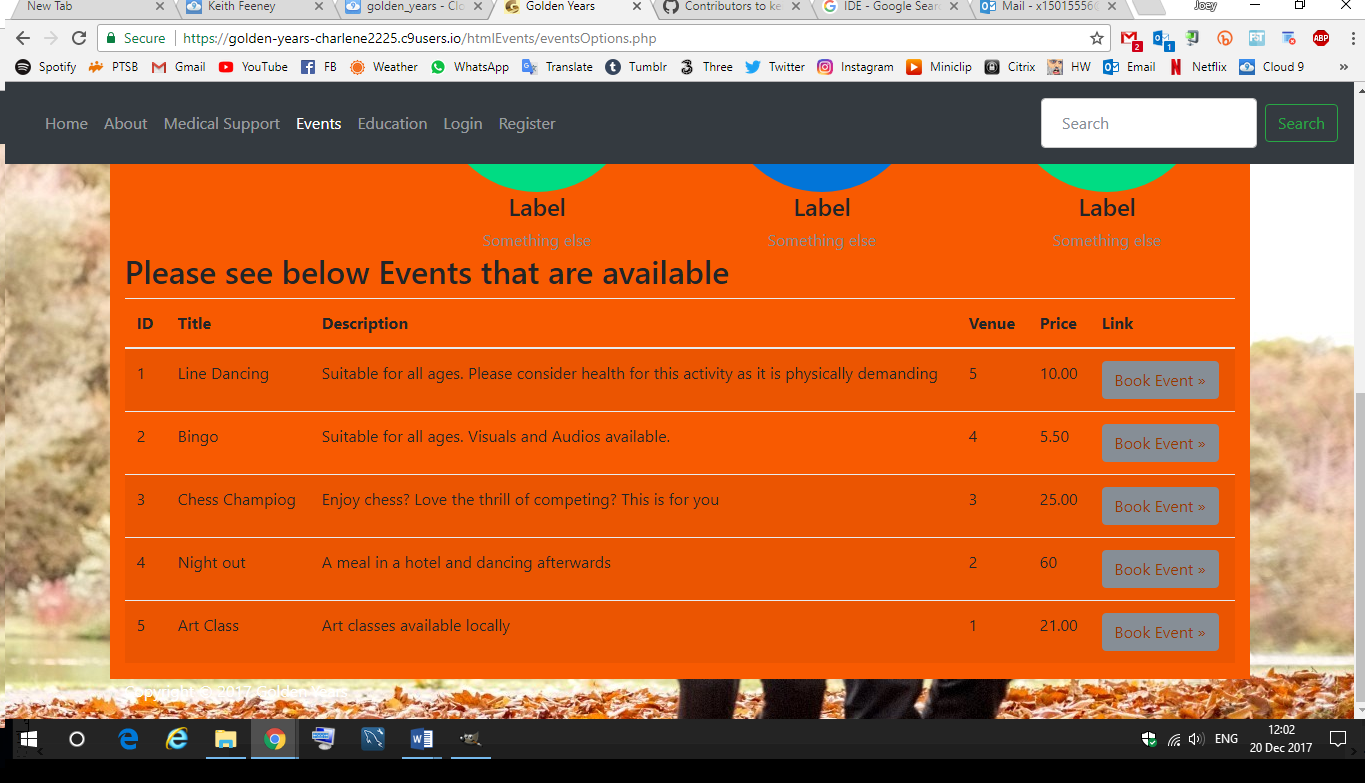
These images show a selection of pages on the website.

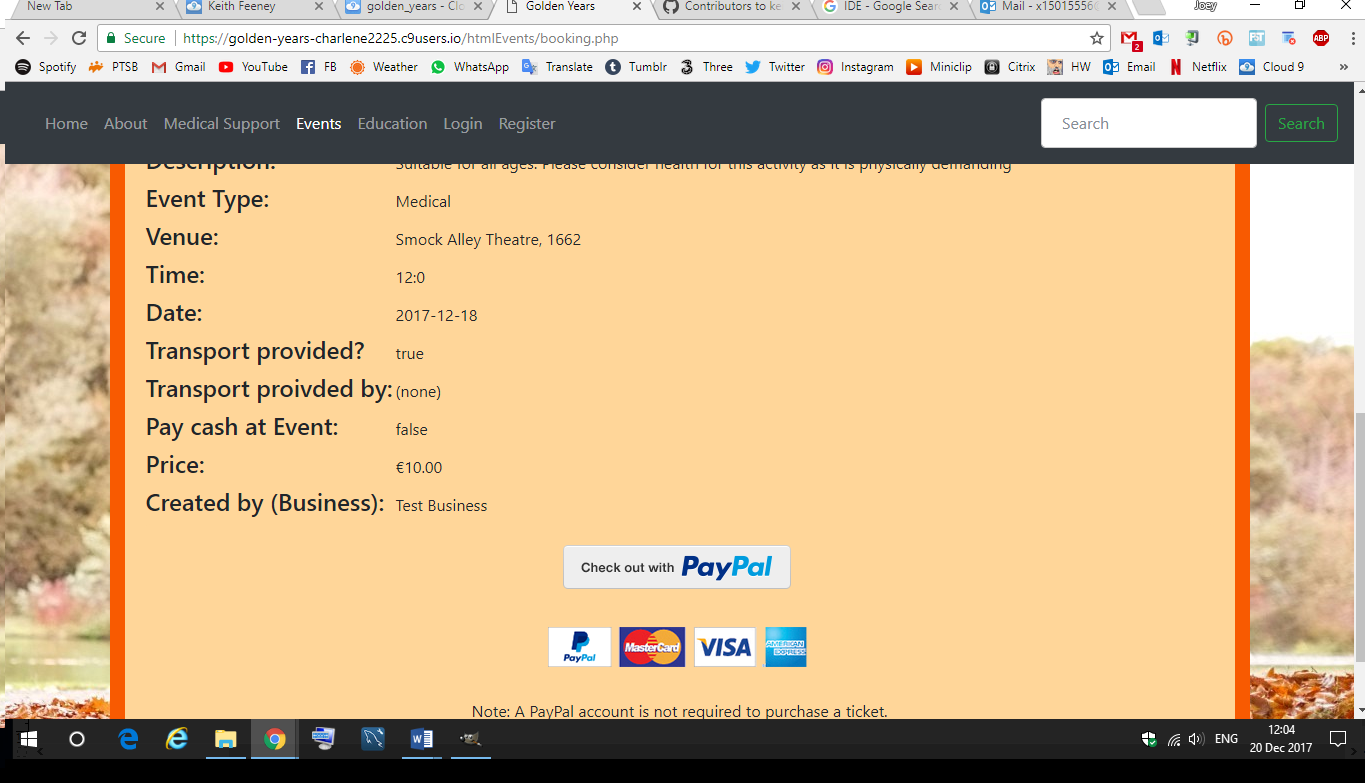




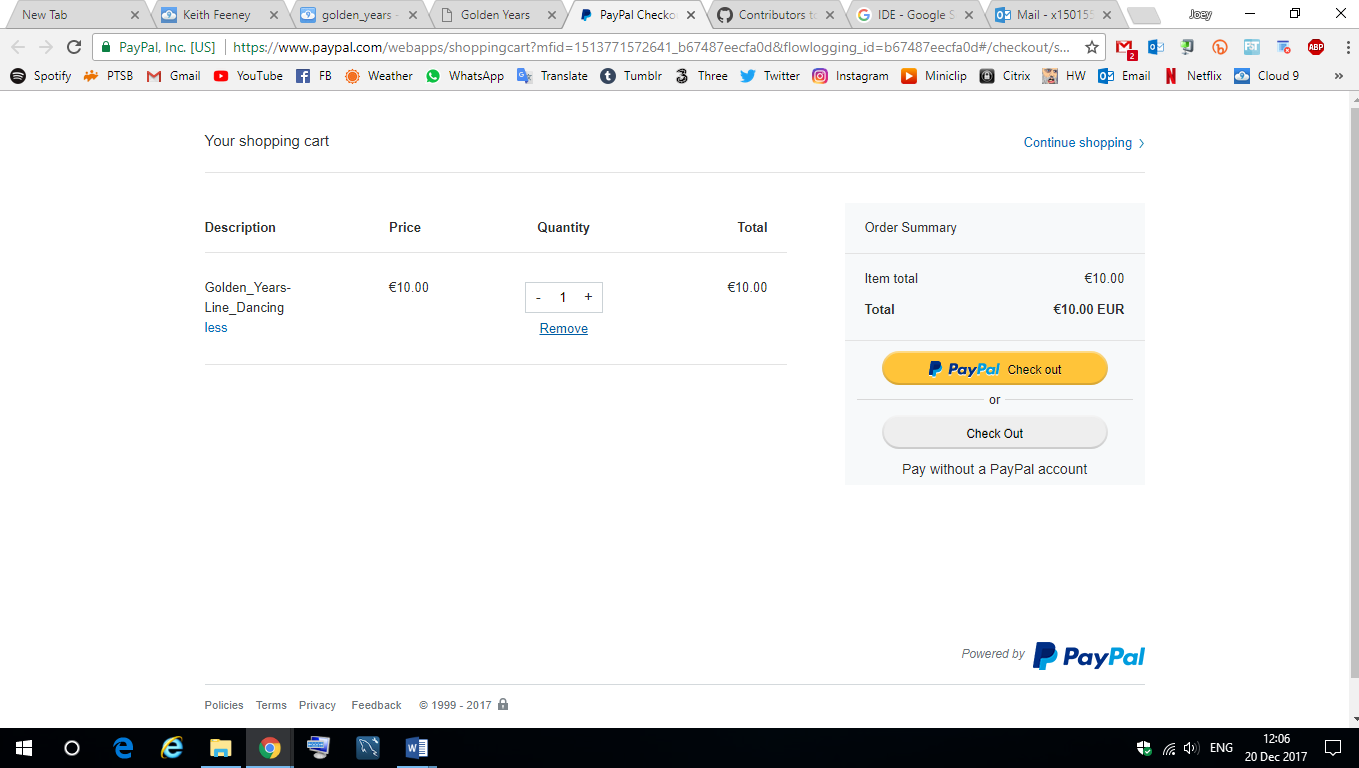


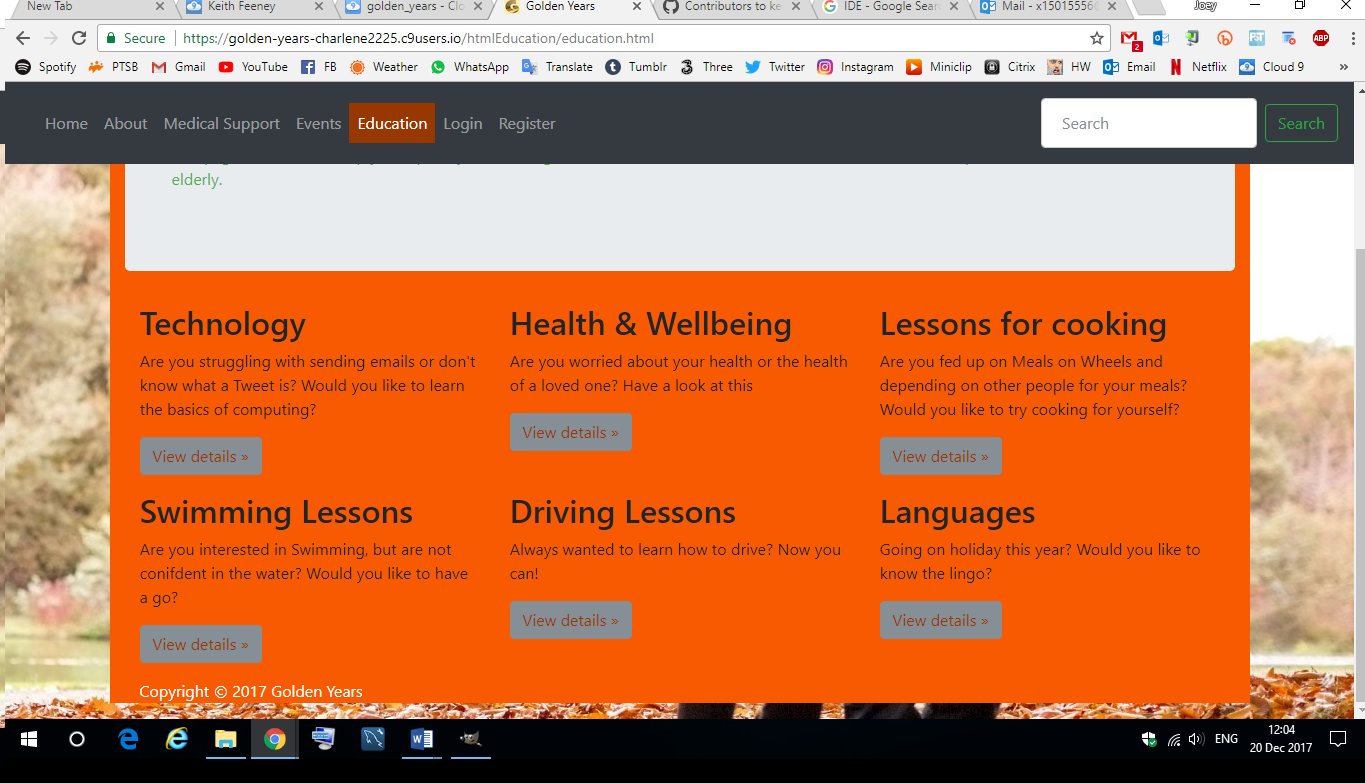






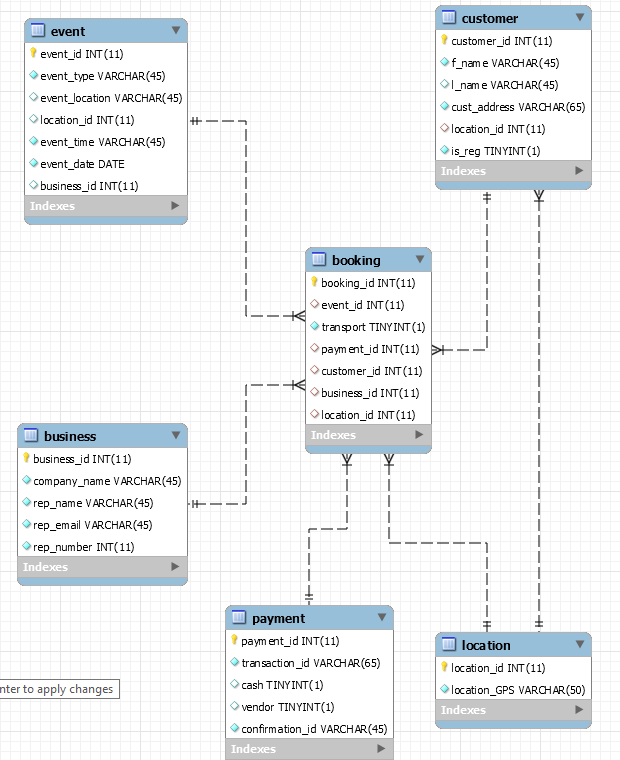
[The PayPal page is just to show the event name and price is sent to PayPal]





### System Architecture

The below class diagrams illustrate the communication between our Web Application. How the different requirements of one process connects with another. For example, our booking system connects with our log in system. As our customers may log in or continue as a guest to continue. All our pages will have relationships. The diagram starts with the Main Page. Here you can see what we offer our customers. For example, Education, Events and Transport. The arrows show direction and the relationships between each class.



### System Evolution

Idea 1: The System could be evolved to be just a Web Site instead of a Web Application. If the touch screens in public areas was not to work, it could become a website so that anyone, anywhere could use it, not just the people standing in public areas.

Idea 2: The System could be transformed into a mobile application. It would be easy to use and could introduce OAPs to the world of technology.

Idea 2a: Because Virtual Reality is becoming popular. The System could incorporate Virtual Reality (VR) headsets with the mobile app. For example, instead of having to go to the location of an event. The event could be started in VR. The OAPs are not as mobile could see attend events

# Discussions and reflections

## Learning, Skills & Process

During the time the group started working on the project, the person chosen as a manager had the responsibility of planning, preparing, and executing the project.

At the beginning of the project, there was a Team Manager and that the manager would change on a weekly basis. After the second week, this was not a viable idea for our group. After this, there was no Team Manager.

At times there will be an issues, delays, or problems, the project manager is the person to call upon to work with that person to review how to fix it. It does not mean that he won't be doing other work required too, but they make sure progress is being made and keeps everyone on track. However, since our group did not have a manager, the group decided to call a meeting in Week 8 to discuss everything and discuss what the group needed to do. During this time, the group developed these skills further:

* Communication skills
* Leadership skills
* Management skills
* Problem-solving skills.

The group have developed the skills in using the project tool interpretation, even though it is not necessary for a team to create work breakdown or a Gantt chart. For our project, the group were able to use these tools, so that the group understand the direction of the project and to be clear how our tasks will fit into the final results. Other skills that the group acquired during the time working on the project, will be communication skills. The group were able to explain our own ideas, express feelings in an open but non-threatening way. The most important one is that we can ask the question to clarify our ideas and emotion and also sense how others in the team feel based on their non-verbal communications. For us to work together on our project successfully, we had to demonstrate a sense of cohesion.

Some examples include:

**Openness:** A team member got to know other team members and the fact that each member all have different interests and ideas. The group were able to develop trust, to the fact we could share our ideas and feelings to each one of us as a team member.

**Support:** The team members show support to each other to for the group to accomplish its main goals of completing the project.

**Respect:** The team members communicated their opinions in a way that was respectful of others, focusing on what the group can learn together, and not looking for who to put the blame on when things go wrong. As a group, the team members have enhance our technological skills

## Client Feedback Session

After testing the completed project on a few people, these people have some issues and some positive aspects with the website.

Some issues include:

* Orange background colour is too vibrant, needs toning down.
* Some users did not like that the Checkout was on PayPal, some would have preferred it to be within the website itself.
* Some users would have preferred that the Medical and Education pages were more informative and not just lists.
* Some users didn’t like that when a user logged in, that their name didn’t stay in the navbar.

If the group had more time, these issues would have been addressed.

Some positive aspects include:

* Users found it was easy to create an account
* Users found it was very easy to book an event if they wished.
* Users liked the overall concept.

# Conclusion and Future Work

In conclusion, this project has really improved the group’s thinking about elderly people, after doing so much research on them. The group believe that this project gives our seniors more activities to partake in and improve their medical check-up, their daily living activities and to progress in education, if they wish to.

The web application is completed to attract users who are senior citizens to be more active.

The group have put a lot of work into this Project, which include new technologies used to make the web application accessible and user friendly.

# Further Development

If more time was provided for the project, the further could be implemented:

* Mobile Application:
  + A Mobile Application could be created, so that users would not need to be at a computer to use this service.
* Further Development:
  + The group could use the time to create an index of Medical issues and Education resources, so that the pages do not just link to other sites.
* Address issues from Client Feedback:
  + If there was more time, the issues from the Client Feedback would be addressed.
* Search
  + Create an independent search bar that is used for the website, that does not use Google Search,