National College of Ireland

BSc in Business & Information Systems

2016/2017

planmyevent.eu

Technical Report

planmyevent.eu

Shauna Drumm

x12383251

shaunadrumm@hotmail.co.uk

National
College *of*
Ireland

Table of Contents

Table of Figures

Table of Tables

Executive Summary

Currently in Ireland there is no online service for booking venues. Planmyevent.eu is a web service that provides this feature. It allows venue owners to register for an account and from here they can add, modify and delete their own venues details meaning they have total control and management of the online presence of their business on third party service. Once venues are added to the service it allows customers to search for a venue based on their chosen criteria. The customers can specify what type of event they're having, how many people and which location and the search results will return a list of venues that match their preferences. From here the customer can view extra details about the venue and then select one of their choice add a date and time and book it. Once the venue is booked the booking will appear in a calendar of bookings for the venue owners to view. They can see the date and time of the booking and the customer's details should they need to contact them. As the bookings increase the venue owners will also be able to utilise a business intelligence dashboard that shows them the number of bookings per month. This will allow them to analyse the most popular times for bookings and make smart business decisions. The venue owners will also be able to see a calendar of bookings and add employee accounts. It will also include a section that will provide advertising space for companies who provide products and services related to events such as balloons, entertainers and flowers. With this functionality planmyevent.eu provides a unique web service for booking and managing venues in Ireland.

Project Scope

The scope of the project is to develop a web service for booking venues. The system shall have a search algorithm that provides the user with a list of search results for venues based on the criteria entered by the user. The user must then be able to book this venue through the web service. The venues will be added by the venue owners themselves reducing the administration of this task for the developer. All requirements are specified below in the relevant sections.

1. System Analysis

Background & Research

Triggered by a personal need for such a service and through Internet research, it appears that currently there is no online booking site for venues in Dublin that allows a user to book directly through the site and allows venue owners to add their own venues through a registered account. The aim of this project is to provide that service. Research highlights two websites of similar topic, but neither showed the intelligent functionality that planmyevent.eu will provide. These websites are typical examples of what is available now. Other websites target specific event types: - Weddings, Birthday parties or they are not much more than a directory listing. There is no intelligence for the customer to efficiently book these venues once selected. There is no sales or marketing campaign cross selling supporting services, catering, entertainers, speakers etc. There is a gap in the Irish market for a web service that books venues directly and allows venue owners to manage their venues and bookings through the website administration console. The two example websites from the research conducted are: -

- www.findavenue.ie
  - o This website is a 'venue finding service'. The user enters the given criteria and that information is sent as an enquiry to the website team. Users experience here is low. Users want an instant response. The opportunity to make revenue is low.
- www.venuesearch.ie
  - o This website is a 'venue directory'. The user enters the venue or event type and location and it returns a list of possible venues with contact details. This has no revenue potential other than random advertisement click revenue, it is manually maintained and has no business intelligence.

Survey Results

To conduct the research further a survey was conducted. The survey was conducted of 8 questions and answered by 35 anonymous people.
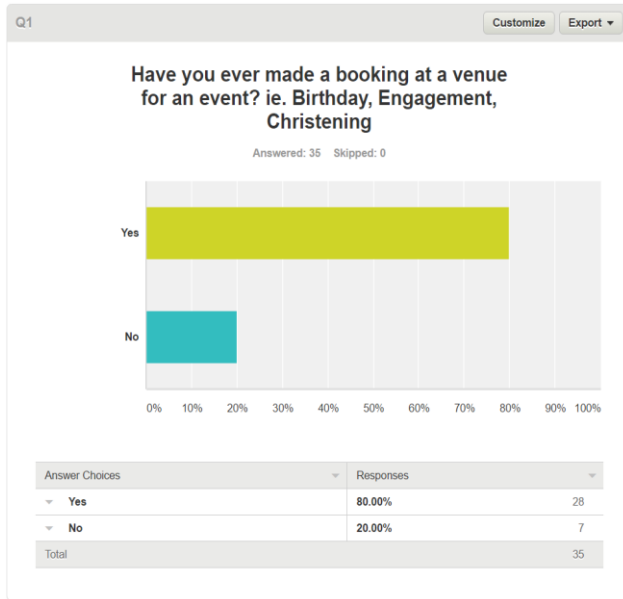
Figure 1: Survey Results 1

Figure 2: Survey Results 2

This result showed that 80% of the people that answered the survey have made a booking before.

This result shows that if there were an online service that allowed the booking of venues 71.43% of the survey respondents would use it. The one person that answered no also answered no to "Have you ever made a booking".
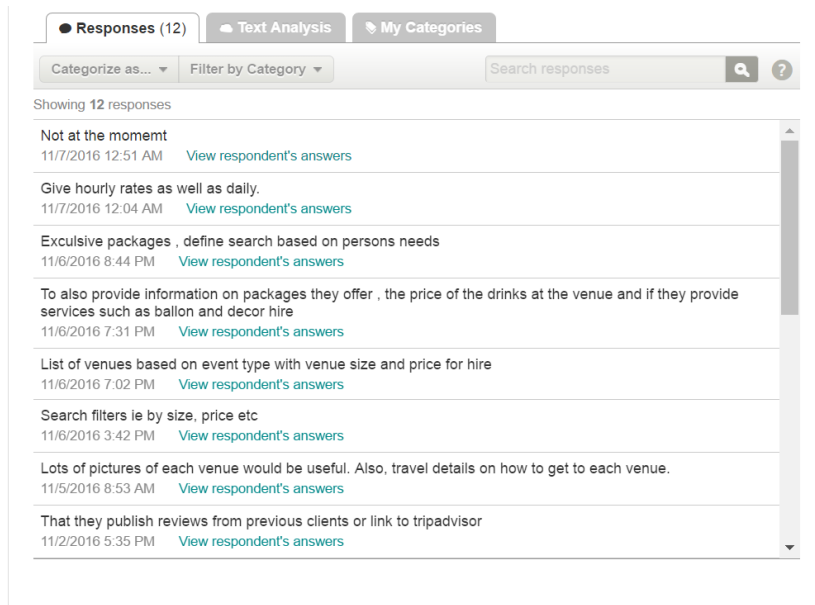
Figure 3: Survey Results 3

These two questions show that planmyevent.eu has a great opportunity for success! Other questions allowed the respondents to make suggestions towards different functionality the web service could offer. These responses will be used to drive the requirements gathering.

System proposal

Aims

The objective of the project is to deliver an Internet Web Service that provides an event management platform for booking venues. Initially it will target small and medium sized enterprises. Revenue will be generated by charging commission on booking fees and placement of targeted advertisements for related services. Its unique selling point will be that it offers services that are currently not available and assist consumers in finding the best match venue in terms of cost, location and feedback. It will go to market with a base set of features and be constantly updated with new features to ensure it retains the interest of the consumers and stays ahead of its competitors. Statistics will be analysed to assess the value in further development of any existing or new features. The project scope will focus on these key areas: -

- The consumer web service
- The Venue Owners administration portal
- The web service must be responsive, resilient and easy to use
- It must aim to meet all functional and non-functional requirements

What planmyevent.eu does for the consumer

Planmyevent.eu will run a search algorithm to return venues that best suit the user's needs. It is one central location for all venues in their chosen area. It will allow them to book directly through the website avoiding the inconvenience of checking multiple websites, making phone calls and submitting multiple enquiries.

Users will be able to: -

- Search for venues based on, location and type of event. View details of this venue including: price, location, reviews and extra features.
- Book venues directly through the website receiving instant confirmation.
- See contact details for venue owners allowing them to submit messages and queries to the venue owners
- Access additional services and products related to events
- Provide reviews and feedback for all venues on planmyevent.eu

What planmyevent.eu does for the business

The venue owners/managers will have their own administration console they can log into via the website. This will allow them to see their bookings populated in a calendar, view sales statistics via a dashboard, manage their venues and venue content such as photos, price etc. The price of their venue will be defined in the site and non-negotiable between buyer and seller. Planmyevent.eu will enable a business intelligence dashboard that only the planmyevent support team can access. This dashboard will contain booking statistics for all venues. Planmyevent.eu will send tips to venue owners on how they can improve business based on features that are most popular. Venue providers will have: -

- Authenticated access to their own administration portal
- Administrate their account
- Add and manage multiple venues. Details for venues can included items such as, headcount, opening hours, bar, typical uses, photos, price.
- View and maintain a schedule of the venue bookings and availability through a calendar in day, week and month view
- Access statistics of consumer bookings, reviews and revenue produced by their venues via a dashboard
- Add and manage employee accounts so they maintain administrator permissions whilst still allowing other employees to view the calendar of bookings
- View any reviews relevant to the venues they own and the option to respond to the customer that added the review.

Advantages

These features make planmyevent.eu a unique service to the Irish Market:

- Providing a unique service to the Irish consumer base and venue owners that isn't currently available.
- Generate revenue based on commission from booking fees
- Generate revenue from advertising space for cross selling services
- Grow its cyber footprint by ensuring high quality user experience in venue booking whilst being very easy to use
- Increase online presence for venues using a third-party service
- Gather and analyse its statistics to help define its future functions
- Provide statistics to venue owners guiding them to make decisions that could increase revenue.
- The service provides little or no training. It should be easy to use, quick and convenient whilst using styles that are attractive and welcoming.
- Prices between customer and venue owner are non-negotiable.
- Employees can access important booking information without accessing confidential business statistics
- The venue owners have an opportunity to respond to reviews and feedback added by customers enabling strong customer loyalty.

Disadvantages

These features negatively impact the succession of planmyevent.eu:

- Failure to grasp the attention of the venue owners, without their interest the web service will not succeed.
- The web service may not be developed in time to keep up with competitors. Losing its differentiation advantage.
- As a start-up visitor traffic will be low. This will be discouraging when inviting service providers to advertise on the page.

Cost Analysis

The following costs must be considered:

Incoming:

- Planmyevent.eu receives €20 euro per month for one advertising space
- Planmyevent.eu receives 15% commission on each booking made

Outgoing:

- Planmyevent.eu offers a free trial to the pilot test group of 2-3 venues for 2-4 months
- Planmyevent.eu offers a 30 day free trial to all new venue owners

Fixed Assets:

Using just a basic plan on Microsoft Azure for the database and web application service the following costs are expected:

Service type	Custom name	Region	Description	Estimated Cost
App Service	App Service	North Europe	1 instance(s), 744 hours, size: b2, basic tier, 0 SNI connection(s), 0 IP connection(s)	$111.60
SQL Database	SQL Database	North Europe	1 basic database(s) x 744 hours, size: b	$4.98
Support			Support	$300.00
			Monthly Total	
			Annual Total	

Disclaimer

All prices shown are in US Dollar ($). This is a summary estimate, not a quote. For up to date pricing information please visit https://azure.microsoft.com/pricing/calculator/
This estimate was created at 5/10/2017 12:01:59 AM UTC.

Table 1: Cost Analysis

This estimate cost includes the standard level of support from Microsoft Azure. This includes the following:

- Technical support for Azure service issues via Resource health
- Unlimited 24x7 billing & subscription support
- Unlimited 24x7 technical support, including non-Microsoft technologies running on Azure
- < 2 hours Initial Response time

This cost is suitable whilst the web service is in production. Once the site goes live planmyevent.eu will require more support from MS Azure and more resources increasing costs. However if it succeeds incoming cash flow should be expected to exceed outgoing cash flow. (Azure, 2017)

2. System Design

Technologies

- **Register365:** This company provides web hosting, DNS and domain registration. (Register365, 2016) Register365 will be used for the domain registration of planmyevent.eu.
- **Microsoft Azure's Web Applications Feature:** A platform for deploying and managing powerful websites and web apps. It supports Windows and Linux platforms. (Azure, 2016) This will be used to host planmyevent.eu
- **WinSCP:** This service provides secure file transfer between a local computer and a remote computer. (WinSCP, 2016) This will be used to securely transfer files and images from the local machine to the hosting server.
- **Bootstrap:** This is a web framework that provides HTML and CSS templates for front end website development. (w3schools, 2016) This will be used for the front end development of planmyevent.eu for aspects such as the form fields, buttons, navigation bar, images and headings.
- **PHP:** Hypertext Preprocessor is a scripting language that can be called from HTML pages. (PHP, 2016) This will be used to provide smart functionality for planmyevent.eu such as gathering data from forms, user logins, and automated emailing.
- **JavaScript:** Is a programming language that can be embedded in HTML pages and is used to create interactive effects within websites. (w3schools, 2016) This will be used to create some interactive styles and features on planmyevent.eu that make the website look modern and attractive to the user. JavaScript libraries such as JQuery will also be used.
- **SQL:** Structured Query Language is a database query language used to extract, filter and modify records. (w3schools, 2016) SQL will be used to create, insert, update and delete all aspects of the planmyevent.eu database. It will also be used for aspects of the search algorithm.
- **GitHub:** Is an open software development repository service. (GitHub, 2016) It will be used for development version control by storing copies of all web files and images and will also be used to retrieve files in the event of a Disaster Recovery.

Requirements

- The requirements specification sets out the variables that must be achieved for the users to gain maximum experience from planmyevent.eu.
- All requirements should be verifiable.
- The requirements are designed so that the system is easy to use and does not require any user training.
- No technical errors should occur. If accidental user errors occur the user can contact planmyevent.eu technical support.
- The MoSCoW technique is used to prioritise all requirements. It allows us to place realistic priorities on project requirements. (Waters, 2016)

"MoSCoW is a technique for helping to understand priorities. The letters stand for:

- *Must Have*
- *Should Have*
- *Could Have*
- *Won't Have this time"*

User Dependencies

The user dependencies are as follows:

Customer:

- The users must have access to a computer, with keyboard and mouse that connects to the internet.
- The users must have some basic knowledge or understanding of using websites.
- The users should require little or no training to use planmyevent.eu but may contact planmyevent.eu support should they require assistance.

Venue Owners:

- The users must have access to a computer, with keyboard and mouse that connects to the internet.
- The users must have some basic knowledge or understanding of using websites.
- The users should require little or no training to use planmyevent.eu but may contact planmyevent.eu support should they require assistance.
- The users must own or manage a real and verifiable venue.

Functional Requirements
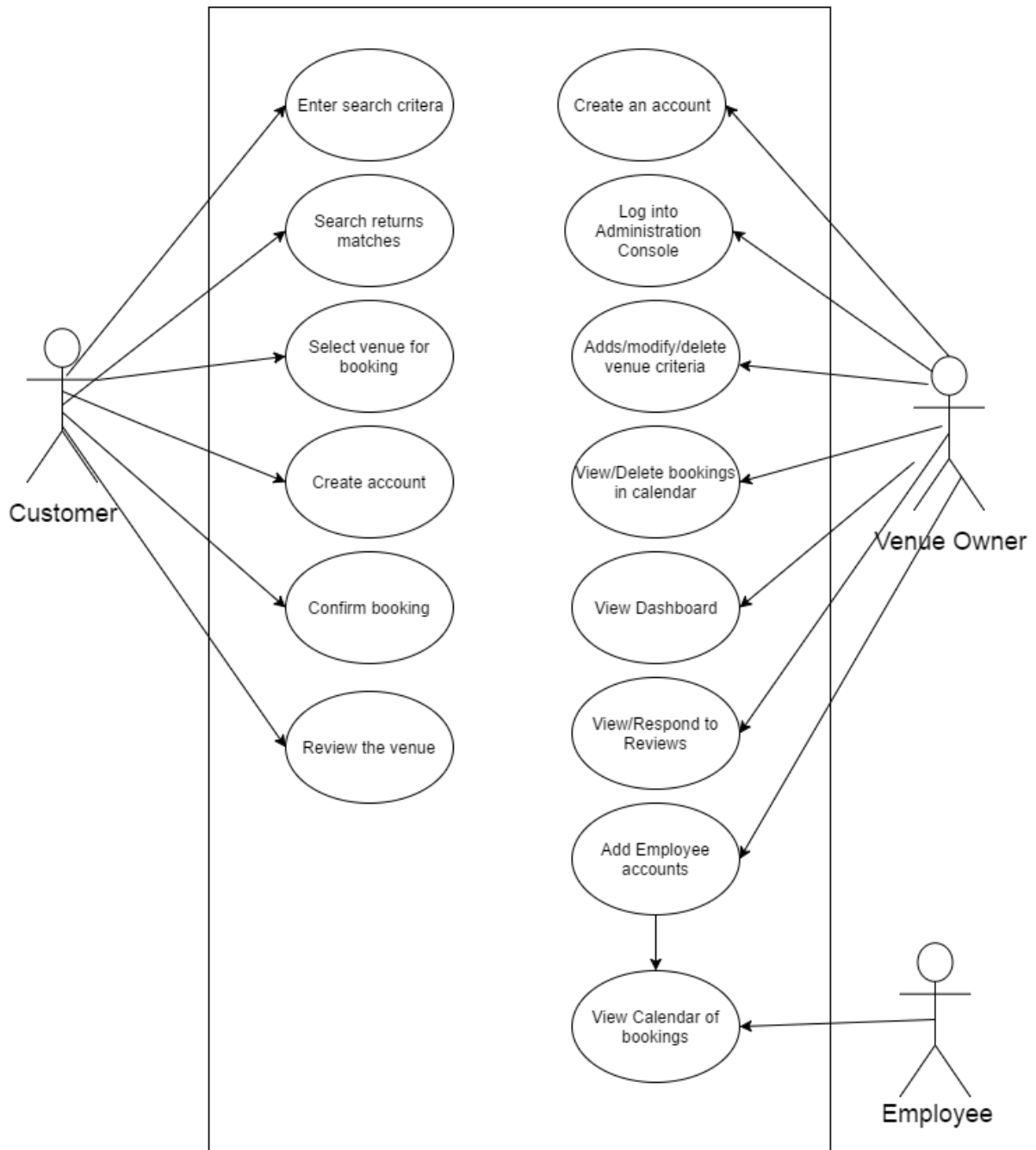
Use Case Diagram

Figure 4: Use Case Diagram

Requirement 1: *FR_001 Venue Owner account creation*

Priority: M – Must have

Description: The following requirement describes that venue owners must be able to create an account with planmyevent.eu that allows them to add, modify and delete (AMD) venues, AMD employee accounts, view or delete bookings, view or respond to reviews and view dashboard. The fields required for venue owner account registration are defined in the ERD (Entity Relationship Diagram).

Use Case

Figure 5: FR_001 Use Case Diagram

Scope: to provide venue owners with the ability to create an account

Flow Description:

Precondition	The user must open a web browser and enter the URL planmyevent.eu
Activation	The process starts when the user clicks "Venue Owner Registration" on the navigation bar.
Main Flow	1. The user begins this process by clicking "Venue Owner Registration" on the navigation bar. 2. The user enters the required information on the registration form 3. The user clicks submit 4. An account has now been created
Alternate Flow	1. The user begins this process by clicking "Venue Owner Registration" on the navigation bar. 2. The user enters the required information on the registration form 3. The user clicks submit 4. The user name or email address entered is already taken 5. The passwords entered do not match 6. The required fields have not been completed 7. The user is offered to try registration again 8. The user enters the required information on the registration form 9. An account has now been created
Exceptional Flow	1. The process is terminated before the registration is submitted aborting the use case.
Termination	The process stops when the user presses submit or closes the browser.
Post Condition	User is a registered member of planmyevent.eu and can log in with username and password at any time.

Table 2: FR_001 Flow Description

Requirement 2: *FR_002 Venue Owners can AMD Employee Accounts*

Priority: S – Should have

Description: The following requirement describes how venue owners with registered accounts can add, modify or delete employee accounts. The fields required for adding an employee are defined in the ERD.

Use Case

Figure 6: FR_002 Use Case Diagram

Scope: to provide venue owners with the ability to AMD employee accounts

Flow Description:

Precondition	The user must open a web browser and enter the URL planmyevent.eu
Activation	The process starts when the user clicks "Venue Owner Login"

Main Flow	1. The user begins this process by clicking "Venue Owner Login" on the navigation bar.
	2. The user enters the log in credentials.
	3. The user clicks login
	4. The user is now logged in to their account
	5. The user selects "Employees" section
	6. The user selects "Add Employee"
	    a. The user enters employee details
	    b. The user clicks submit
	    c. The employee account is created
	7. The user selects "Modify Employee"
	    a. The user enters employee details
	    b. The user clicks submit
	    c. The employee account is modified
	8. The user selects "Delete Employee"
	    a. The user selects the employee username
	    b. The user selects confirm
	    c. The employee account is deleted
	9. The employee account is created, modified or deleted

Alternate Flow	1. The user begins this process by clicking "Venue Owner Login" on the navigation bar. 2. The user enters the log in credentials. 3. The user clicks login 4. The user is now logged in to their account 5. The user selects "Employees" section 6. The user selects "Add Employee" d. The user enters employee details

	e. The employee username is already taken f. The passwords do not match g. The user is asked to try again h. The user enters employee details i. The user clicks submit j. The employee account is created
Exceptional Flow	1. The venue owner doesn't have a registered account, the user must register for an account. 2. The process is terminated by the user before the registration is submitted aborting the use case.
Termination	The process stops when the user presses submit/confirm and log out or closes the browser or their log in session times out before they press submit.
Post Condition	An employee account is created, modified or deleted. If added, the employee can log in. If modified the employee account details will be changed. If deleted, the employee is no longer registered and therefore cannot log in.

Table 3: FR_002 Flow Description

Priority: M – Must have

Description: The following requirement describes how venue owners with registered accounts can add, modify or delete venues. The fields required for adding venues is defined in the ERD.
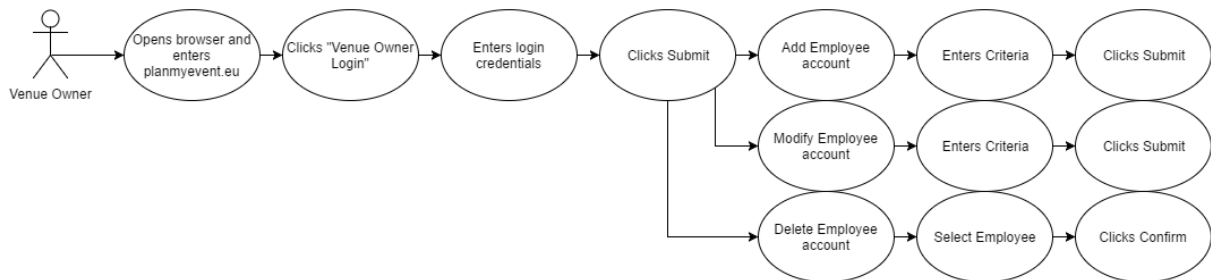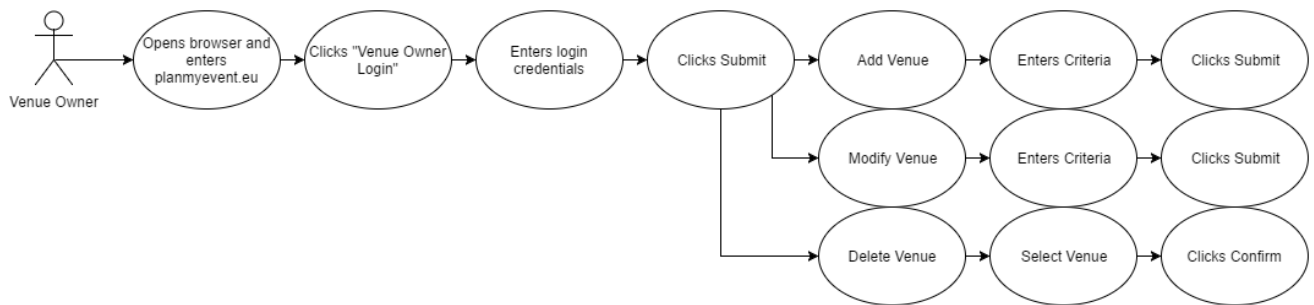
Use Case

Figure 7: FR_003 Use Case Diagram

Scope: to provide venue owners with the ability to AMD venues

Flow Description:

Precondition	The user must open a web browser and enter the URL planmyevent.eu
Activation	The process starts when the user clicks "Venue Owner Login"

Main Flow	1. The user begins this process by clicking "Venue Owner Login" on the navigation bar.
	2. The user enters the log in credentials.
	3. The user clicks login
	4. The user is now logged in to their account
	5. The user selects "Venues" section
	6. The user selects "Add Venue"
	    a. The user enters venue details
	    b. The user clicks submit
	    c. The venue is created
	7. The user selects "Modify Venue"
	    d. The user enters venue details
	    e. The user clicks submit
	    f. The venue is modified
	8. The user selects "Delete Venue"
	    d. The user selects the venue name
	    e. The user selects confirm
	    f. The venue is deleted
	9. The venue is created, modified or deleted

Alternate Flow	1. The user begins this process by clicking "Venue Owner Login" on the navigation bar.
	2. The user enters the log in credentials.
	3. The user clicks login
	4. The user is now logged in to their account
	5. The user selects "Venues" section
	6. The user selects "Add Venue"
	    d. The user enters venue details
	    e. The venue name already exists

	f. The photo is too large g. The photo is the wrong format h. The required fields were not entered i. The user is offered to try again j. The user enters the venue details k. The user clicks submit l. The venue is created
Exceptional Flow	1. The venue owner doesn't have a registered account, the user must register for an account. 2. The process is terminated before the registration is submitted aborting the use case.
Termination	The process stops when the user presses submit/confirm and log out or closes the browser or the user log in session times out.
Post Condition	A venue is created, modified or deleted. If added, the venue can be searched for by the customers. If modified the venue details will be changed. If deleted, the venue no longer exists and therefore cannot be searched for or booked.

Table 4: FR_003 Flow Description

Requirement 4: *FR_004 Venue Owners can view, modify & delete bookings*

Priority: M – Must have

Description: The following requirement describes how venue owners with registered accounts can view, modify and delete bookings in a calendar. The fields required for bookings is defined in the class diagram.
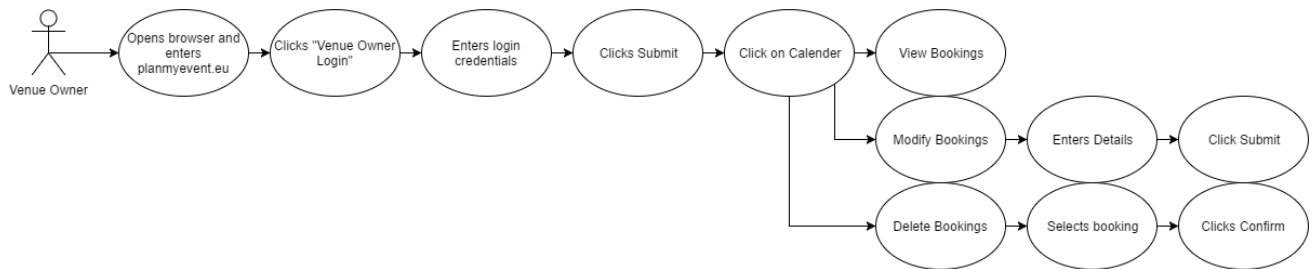
Use Case

Figure 8: FR_004 Use Case Diagram

Scope: to provide venue owners with the ability to view, modify and delete bookings.

Flow Description:

Precondition	The user must open a web browser and enter the URL planmyevent.eu
Activation	The process starts when the user clicks "Venue Owner Login"

Main Flow	1. The user begins this process by clicking "Venue Owner Login" on the navigation bar.
	 2. The user enters the log in credentials.
	 3. The user clicks login
	 4. The user is now logged in to their account
	 5. The user selects "Calendar" section
	 6. The user can view the bookings by day, week, month
	 7. The user selects a booking to modify
	    a. The user enters amendments
	    b. The user clicks submit
	    c. The booking has been modified
	 8. The user selects a booking to delete
	    a. The user selects a booking
	    b. The user selects delete
	    c. The user clicks submit
	    d. The booking has been deleted
	 9. The bookings have been viewed, modified or deleted.

Alternate Flow	Not applicable
Exceptional Flow	1. The venue owner doesn't have a registered account, the user must register for an account. 2. The process is terminated before the action is complete aborting the use case. 3. The user has not added a venue and has no bookings
Termination	The process stops when the user presses submit/confirm and log out or closes the browser or the users log in session times out.
Post Condition	The bookings remain as they were at the point of termination.

Table 5: FR_004 Flow Description

Requirement 5: *FR_005 Venue Owners can view and respond to reviews*

Priority: S – Should have

Description: The following requirement describes how venue owners with registered accounts can view, and respond to reviews of their venues left by customers after the event has happened. The fields required for bookings is defined in the class diagram.
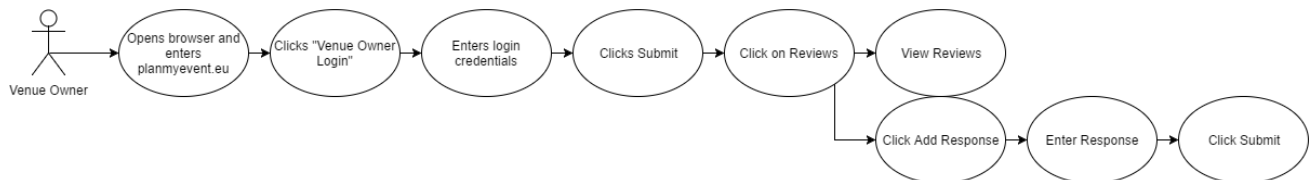
Use Case

Figure 9: FR_005 Use Case Diagram

Scope: to provide venue owners with the ability to view, and respond to customer reviews.

Flow Description:

Precondition	The user must open a web browser and enter the URL planmyevent.eu
Activation	The process starts when the user clicks "Venue Owner Login"

Main Flow	1. The user begins this process by clicking "Venue Owner Login" on the navigation bar.
	 2. The user enters the log in credentials.
	 3. The user clicks login
	 4. The user is now logged in to their account
	 5. The user selects "Reviews" section
	 6. The user can view the reviews
	 7. The user can respond to reviews
	    a. The user selects a review
	    b. The user enters a response
	    c. The user selects submits
	    d. The response is emailed to the customer

Alternate Flow	Not applicable
Exceptional Flow	1. The venue owner doesn't have a registered account, the user must register for an account. 2. The process is terminated before the action is complete aborting the use case. 3. The user has not added any venues 4. The customer has not entered an email address
Termination	The process stops when the user presses submit/confirm and log out or closes the browser or the users log in session times out.
Post Condition	The reviews remain as they were at the point of termination.

Table 6: FR_005 Flow Description

Requirement 8: *FR_006 Venue Owners can view a bookings dashboard*

Priority: M – Must have

Description: The following requirement describes how venue owners with registered accounts can view a dashboard showing five data marts. The data marts show number of each space booked, revenue from each space booked, number of each space booked per month, extras included in bookings and total number of ratings. The fields required for bookings and venues are defined in the class diagram.
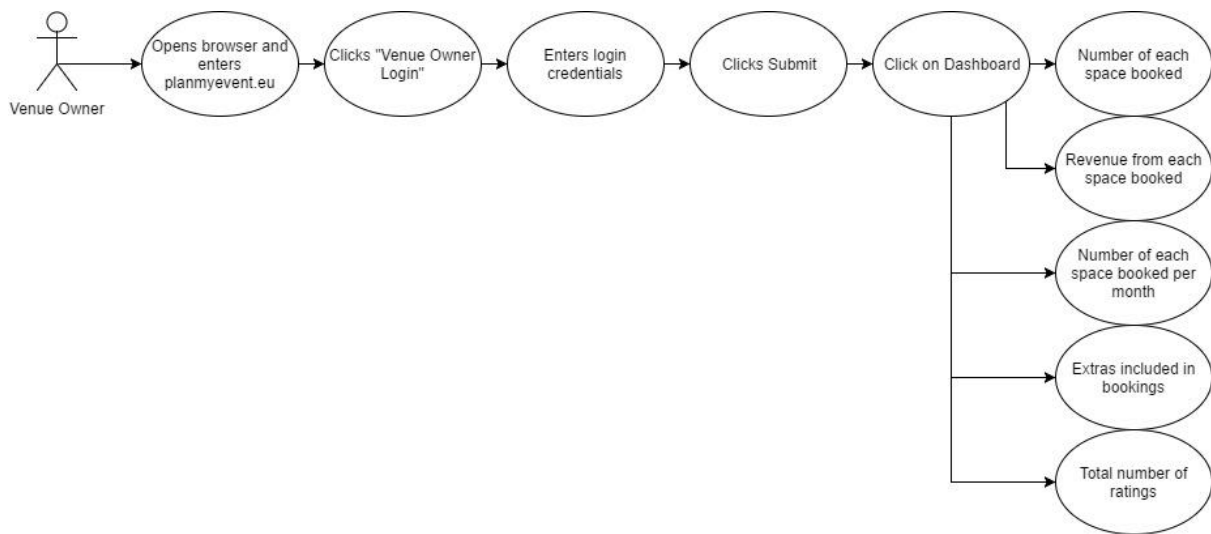
Use Case

Figure 10: FR_008 Use Case Diagram

Scope: to provide venue owners with the ability to view a dashboard showing useful dashboard statistics

Flow Description:

Precondition	The user must open a web browser and enter the URL planmyevent.eu
Activation	The process starts when the user clicks "Venue Owner Login"

Main Flow	1. The user begins this process by clicking "Venue Owner Login" on the navigation bar.
	2. The user enters the log in credentials.
	3. The user clicks login
	4. The user is now logged in to their account
	5. The user selects "Dashboard" section
	6. The user can view the dashboard
	    a. The user can view the dashboard by number of each space booked
	    b. The user can view the dashboard by revenue from each space booked
	    c. The user can view the dashboard by number of each space booked per month
	    d. The user can view the dashboard by extras included in bookings
	    e. The user can view the dashboard by total number of ratings
	7. The user can print the dashboard
	8. The user can print the individual charts

Alternate Flow	Not applicable

Exceptional Flow	1. The venue owner doesn't have a registered account, the user must register for an account. 2. The process is terminated before the action is complete aborting the use case. 3. The user has not added any venues 4. The user has not received any bookings
Termination	The process stops when the user clicks on a different page or logs out or closes the browser or the users log in session times out.
Post Condition	The dashboard will update based on bookings made by customers.

Table 7: FR_006 Flow Description

Priority: S – Should have

Description: The following requirement describes how employees can log in to view calendar of bookings and add notes for venue owners to see. They should not have any other permissions to amend the bookings or venue. The fields required for employee accounts are defined in the class diagram.
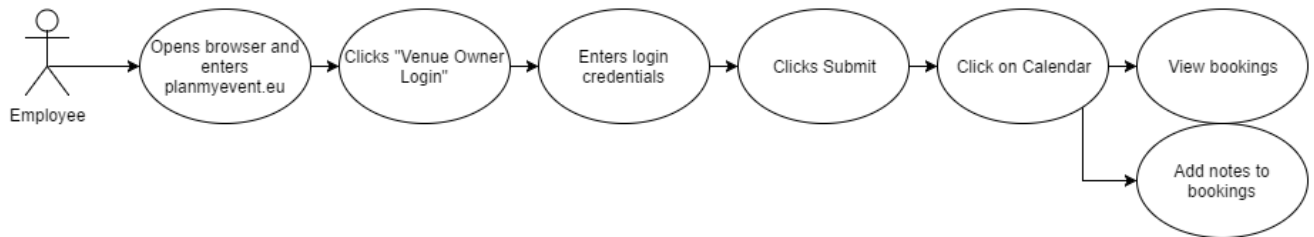
Use Case

Figure 11: FR_007 Use Case Diagram

Scope: to provide employees with the ability to view and add notes to bookings

Flow Description:

Precondition	The user must open a web browser and enter the URL planmyevent.eu
Activation	The process starts when the user clicks "Employee Login"
Main Flow	1. The user begins this process by clicking "Employee Login" on the navigation bar. 2. The user enters the log in credentials. 3. The user clicks login 4. The user is now logged in to their account 5. The user can view the calendar

Alternate Flow	1. Steps 1-4 of main flow
	  a. The user can view the bookings in the calendar in month view
	
	 2. Steps 1-4 of main flow
	  a. The user selects add note
	  b. Types the note
	  c. Clicks submit
	  d. The note is now added alongside the booking

Exceptional Flow	If the process is terminated before it is finalised the user will need to start again.
Termination	The process stops when the user clicks on a different page or logs out or closes the browser.
Post Condition	The calendar of bookings updates based on bookings made by customers.

Table 8: FR_007 Flow Description

Requirement 8: *FR_008 Customers can search for venues*

Priority: M – Must have

Description: The following requirement describes how a customer must be able to search for venues based on their selected criteria without registering for an account. The fields required for the search are outlined in the class diagram.

Use Case

Figure 12: FR_008 Use Case Diagram

Scope: to provide customers with the ability to search for venues

Flow Description:

Precondition	The user must open a web browser and enter the URL planmyevent.eu
Activation	The process starts when the user clicks "Search for a Venue"
Main Flow	1. The user begins this process by entering the search criteria 2. The user clicks search 3. The system returns search matches
Alternate Flow	1. The user begins this process by entering the search criteria 2. The user clicks search 3. The system does not return any search matches 4. The user tries again 5. The system returns search matches
Exceptional Flow	1. The process is terminated before the action is complete aborting the use case.
Termination	The process stops when the user clicks on a different page or closes the browser.
Post Condition	The web service remains the same.

Table 9: FR_008 Flow Description

Priority: M – Must have

Description: The following requirement describes how a customer must be able to book a venue they have just searched for. The fields required for the search and booking are outlined in the class diagram.

Use Case

Figure 13: FR_009 Use Case Diagram

Scope: to provide customers with the ability to book venues after completing a search

Description: to allow customers to enter the web service, enter a selection of features they wish their venue to provide and book a venue based on the search results

Flow Description:

Precondition	The user must open a web browser and enter the URL planmyevent.eu
Activation	The process starts when the user clicks "Search for a Venue"
Main Flow	1. The user begins this process by clicking "Search for a Venue" on the navigation bar. 2. The user enters the search criteria 3. The user clicks search 4. The system returns search matches 5. The user selects the venue they wish to book 6. The user selects 'book this venue' 7. The user enters their personal details and details regarding the booking 8. The user selects confirm 9. The venue is booked
Alternate Flow	1. The user begins this process by clicking "Search for a Venue" on the navigation bar. 2. The user enters the search criteria 3. The user clicks search 4. The system returns search matches 5. The user selects the venue they wish to book 6. The user selects 'book this venue' 7. The venue does not offer features the user wants 8. The user cancels the booking process 9. The user selects a different venue 10. The user selects 'book this venue' 11. The user enters their personal details and details regarding the booking 12. The user selects confirm 13. The venue is booked
Exceptional Flow	1. The user already knows which venue they want to book but it is not available on planmyevent.eu 2. The venue is already booked on their chosen date
Termination	The process stops when the user confirms the booking, clicks on a different page or closes the browser.

Post Condition	The venue is booked for the date and time selected. The booking is added to the venue owners calendar of bookings.

Table 10: FR_009 Flow Description

Requirement 10: *FR_010 Customers can leave reviews after an event*

Priority: S -Should have

Description: The following requirement describes how a customer should be able to leave a review after an event has occurred.

Use Case

Figure 14: FR_010 Use Case Diagram

Scope: to provide customers with the ability to leave reviews after an event has occurred

Description: to allow customers to submit a review of a venue after the booked event at that venue has occurred.

Flow Description:

Precondition	The user must have an email account. The user opens a browser and enters planmyevent.eu
Activation	The process starts when the user selects 'Add Review'
Main Flow	1. The user begins this process by navigating to the 'Add Review' page 2. The user enters the review criteria 3. The user clicks submit 4. The review is submitted
Alternate Flow	Not applicable
Exceptional Flow	1. The user does not have an email address 2. The user does not complete the mandatory fields 3. The user terminates the process before the form has been submitted aborting the use case
Termination	The process stops when the user submits the review
Post Condition	The review is added to the venue details and the venue owners profile to give them the ability to respond.

Table 11: FR_010 Flow Description

Interface Requirements

Graphical User Interface (GUI) Requirements
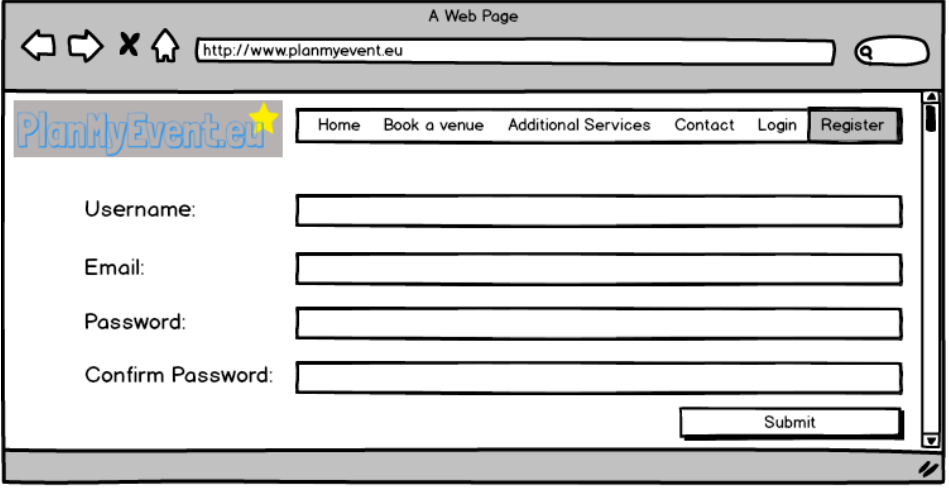
GUI Requirement	GUI_001
References Functional Requirement	FR_001 - Venue Owner account creation
Wireframe	
Description	1. Must contain the following fields: • Title • First Name • Surname • Username • Company Name • Email • Password • Confirm Password 2. Button should say 'Submit Registration' 3. Navbar must be included 4. Footer must be included 5. Style of page must be in planmyevent.eu theme

Table 12: GUI_001

GUI Requirement	GUI_002
References Functional Requirement	FR_002 - Venue Owners can AMD Employee Accounts
Wireframe	
Description	1. Must contain the following fields: • First name • Surname • Staff number • Username • Password • Confirm Password 2. Button should say 'Submit Registration' 3. Navbar must be included 4. Footer must be included

Table 13: GUI_002

GUI Requirement	GUI_003
References Functional Requirement	FR_003 - Venue Owners can AMD Venues
Wireframe	
Description	1. Must contain all fields defined in the ERD

	2. Should be broken up using JQuery Steps
	3. Should be broken into the following sections:
	• Venue Details
	• Event Details
	• Venue Features
	• Booking Policies
	• Photos
	• Agreement
	4. Navbar must be included
	5. Footer must be included
	6. Style of page must be in planmyevent.eu theme

Table 14: GUI_003

GUI Requirement	GUI_004
References Functional Requirement	FR_004 - Venue Owners can view/delete bookings
Wireframe	
Description	1. Calendar should be available in day, week and month view 2. The day should show the booking venue, date and time. 3. Navbar must be included 4. Footer must be included 5. Style of page must be in planmyevent.eu theme

Table 15: GUI_004

GUI Requirement	GUI_005
References Functional Requirement	FR_005 - Venue Owners can view and respond to reviews
Wireframe	
Description	1. Reviews should be divided into: 5 star(green), 4 star(yellow), 3 star(yellow), 2 star(red) and 1 star(red). 2. Review details should be: • Venue Name • Space booked • Star rating • Review

	• Customer name
	• Customer Email
	3. Navbar must be included
	4. Footer must be included
	5. Style of page must be in planmyevent.eu theme

Table 16: GUI_005

GUI Requirement	GUI_006
References Functional Requirement	FR_006 - Venue Owners can view a bookings dashboard
Wireframe	
Description	1. Must show a variation of charts to meet the functional requirement 2. Charts must use colour scheme 3. Navbar must be included 4. Footer must be included 5. Style of page must be in planmyevent.eu theme

Table 17: GUI_006
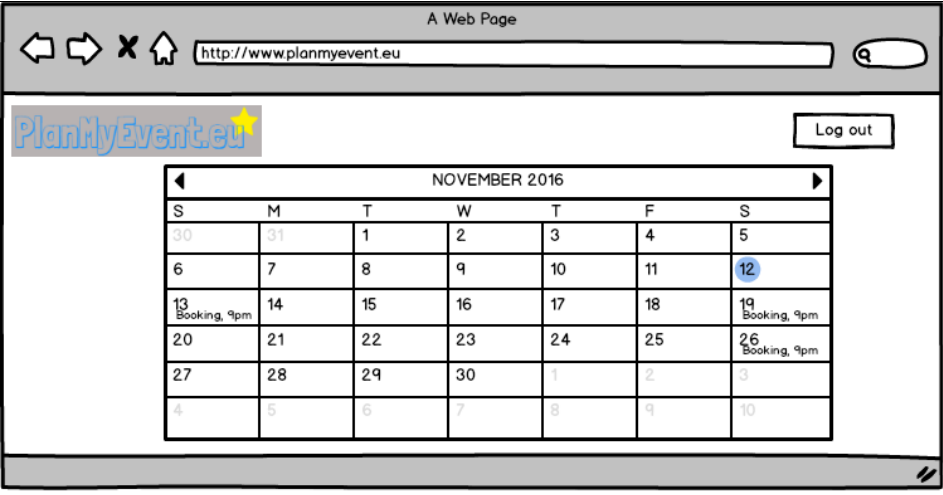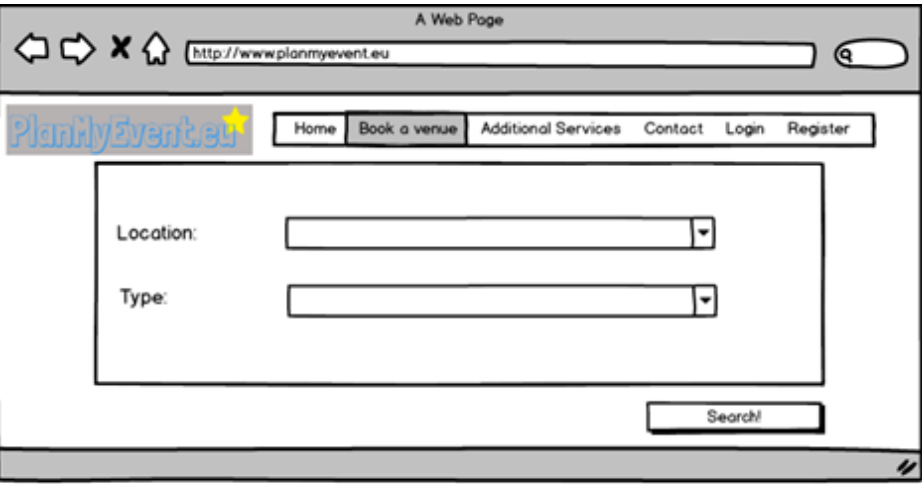
GUI Requirement	GUI_007
References Functional Requirement	FR_007 - Employees can view calendar of bookings
Wireframe	
Description	1. View calendar in day, week, month view 2. The day should show the booking venue, date and time. 3. Navbar won't be included 4. Footer must be included 5. Style of page must be in planmyevent.eu theme

Table 18: GUI_007

GUI Requirement	GUI_008
References Functional Requirement	FR_008 - Customers must be able to search for venues
Wireframe	

Description	1. The search fields must be:
	    • Location
	    • Type of event
	2. The location field could have map
	3. Date could be a field
	4. The button must say 'Search'
	5. Navbar must be included

| | 6. Footer must be included |
| | 7. Style of page must be in planmyevent.eu theme |

Table 19: GUI_008
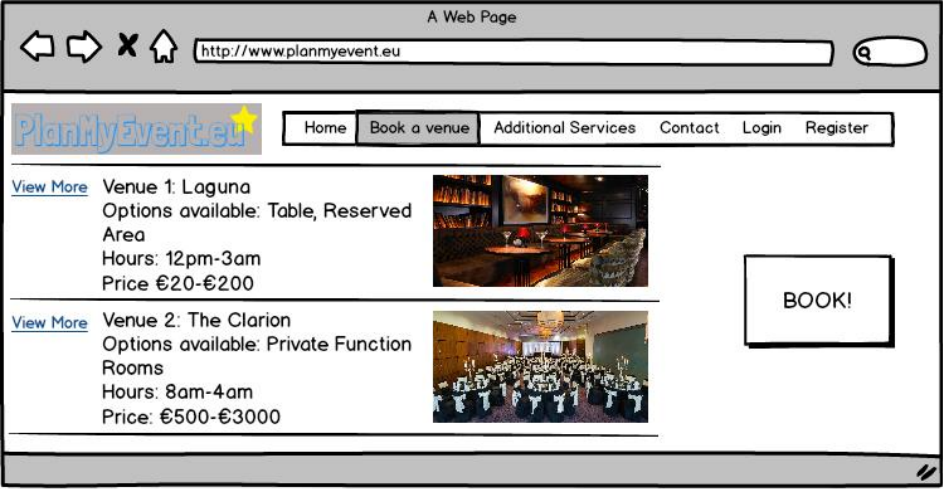
GUI Requirement	GUI_009
References Functional Requirement	FR_009 - Customers must be able to book venues
Wireframe	*Created with Balsamiq - www.balsamiq.com*
Description	1. Search results should include: • Name • Address & Phone number • Photo • Some extra features • Areas for reservation • Price • Location on map • Reviews 2. Clicking on 'Read Reviews' will open the reviews for the selected venue 3. Clicking on 'View on Map' will show that venue's location on a map 4. Clicking on 'Book this venue' will open a form to fill for booking. Fields must include: • Venue Name • Start date & time • End date & time • Reservation area and price • Number of people • Extras available • Special requests • Customer name • Customer email • Customer Phone number 5. Button should say 'Book this venue' 6. Button 'cancel' to go back to search results

Table 20: GUI_009

GUI Requirement	GUI_010
References Functional Requirement	FR_010 - Customers must be able to leave reviews after an event
Wireframe	A Web Page http://www.email.com Inbox Junk Mail Sent Items Deleted Items New Email from planmyevent.eu We hope you had a successful and enjoyable event! Please review the venue by replying to this email. We look forward to seeing you again shortly! Kindest Regards, The planmyevent.eu team :) Reply Forward Created with Balsamiq - www.balsamiq.com
Description	1. Email text should read: "We hope you had a successful and enjoyable event! Please review the venue by replying to this email. We look forward to seeing you again shortly! Kindest Regards, The planmyevent.eu team ☺" 2. Link to planmyevent.eu should be at the end 3. Logo should be at the end

Table 21: GUI_010

GUI Requirement	GUI_011 – Must have a logo
References Functional Requirement	Not applicable
Priority	M
Wireframe	PlanMyEvent.eu
Description	Logo must appear on every page within the navbar or at the top left of the screen.

Table 22: GUI_011

GUI Requirement	GUI_012 Must have a style
References Functional Requirement	Not applicable
Priority	M
Wireframe	Not applicable
Description	Must have an attractive modern style consistent throughout the web service.

Table 23: GUI_012

GUI Requirement	GUI_013 Customer Navbar
References Functional Requirement	Not applicable
Priority	M
Wireframe	Not applicable
Description	1. Should include the following: • Home • Additional Services • About • Add Review • Login • Venue Registration

Table 24: GUI_013

GUI Requirement	GUI_014 Venue Owner Navbar
References Functional Requirement	Not applicable
Priority	M
Wireframe	Not applicable
Description	1. Should include the following: • Home • Employees • Venues • Bookings • Reviews • Dashboard • Log Out

Table 25: GUI_014

Application Programming Interfaces (API) Requirements

API Requirement	Name	Description	Prioirty
AP_001	Google Maps	Google Maps API to show venue location. A google maps API key is required for this.	S
AP_002	Google Analytics	Use Google Analytics to gather information about website traffic and use.	M

Table 26: AP

Non-functional Requirements

Performance/Response time requirement

Non-functional Requirement	Name	Description	Prioirty
NF-T_001	Homepage loading	It should take no longer than 5 seconds for the home page to load	S
NF-T_002	Submission loading	It should take no longer than 20 seconds for a form submission to load	S
NF-T_003	Reservations	It must not take longer than 30 seconds for a booking to be confirmed to avoid double booking	M

Table 27: NF-T

Availability requirement

Non-functional Requirement	Name	Description	Prioirty
NF-A_001	Availability	Should be available 99.9% of the time per year	S
NF-A_002	Maintenance	Won't be available between 1am and 5am on Saturday nights due to maintenance	W

Table 28: NF-A

Recover requirement

Non-functional Requirement	Name	Description	Priority
NF-R_001	Disaster Recovery	All files must be recovered within a 24-hour period of notification of the disaster	M

Table 29: NF-R

Security requirement

Non-functional Requirement	Name	Description	Priority
NF-S_001	Secure Communication	HTTPS must be used when handling user data	M
NF-S_002	Admin Passwords	Hosting services and database must be password protected with strong passwords	M
NF-S_003	Password Encryption	Passwords must be encrypted with SHA1 encryption	M
NF-S_004	User Accounts	Usernames and passwords must match before authenticated access is granted to the user	M
NF-S_005	Primary Key	Primary Keys should be unique and randomly generate	S
NF-S_006	Firewall	The web hosting service must provide a firewall	M

Table 30: NF-S

Reliability requirement

Non-functional Requirement	Name	Description	Priority
NF-RE_001	VAT Number	Registered venues must provide a VAT number	M

Table 31: NF-RE

Maintainability requirement

Non-functional Requirement	Name	Description	Priority
NF-M_001	Bugs	Bugs must be logged once reported	M
NF-M_002	Version Control	Version control must be kept	M
NF-M_003	Backups	Regular system backups should be implemented	S

Table 32: NF-M

Portability requirement

Non-functional Requirement	Name	Description	Priority
NF-P_001	Portable	All aspects should be portable should it need to move to a different hosting service	S

Table 33: NF-P

Reusability requirement

Non-functional Requirement	Name	Description	Priority
NF-RU_001	Includes	The navbar, footer and stylesheets must be reusable in all pages	M

Table 34:NF-RU

Resource utilization requirement

Non-functional Requirement	Name	Description	Priority
NF-UT_001	Bandwidth	A minimum of 15GB data per month	M
NF-UT_002	Disk Space	A minimum of 1GB disk space	M
NF-UT_003	Database	A minimum of 200MB database	M
NF-UT_004	Emails	A minimum of 50 email accounts	M

Table 35: NF_UT

Design and Architecture

Register 365 System Architecture

The following diagram describes the system architecture for planmyevent.eu:
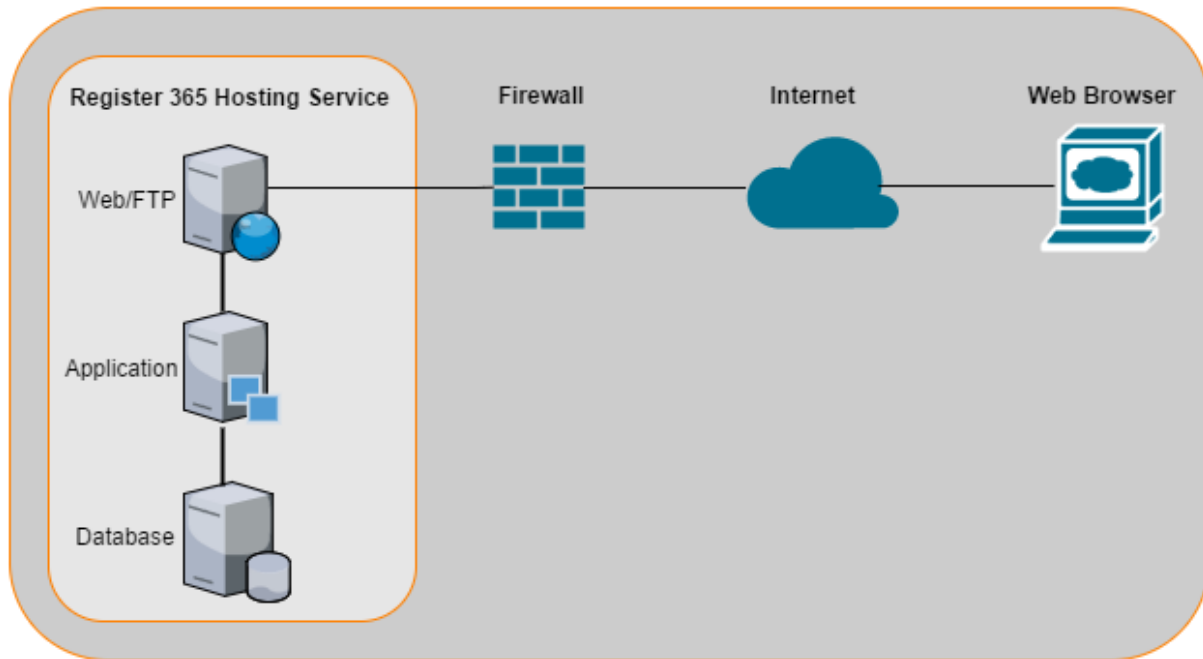
Figure 15: System Architecture

The system architecture is based on a "LAMP" stack (Linux, Apache, MySQL, PHP). (Turkeylinux, 2016) The web hosting server will have four components running on it: -

- An Apache Web service
- A FTP service
- A Mail service
- PHP interpreter
- A MySQL database service

The web server serves the web files such as the HTML, CSS and JavaScript files. The application server runs the PHP application. The database stores the data records: - customer, venue, venue owner, employee and booking details. FTP (File Transfer Protocol) is used to securely transfer files from the local machine to the server. The mail service is used to create mailboxes with the domain name for example: info@planmyevent.eu.

The user launches a web browser such as Google Chrome, Internet Explorer or Mozilla Firefox and enters the url: planmyevent.eu. The browser accesses the internet and resolves the IP address of the site from the DNS server. It will then request the home page. It will access the web server via the Firewall. The firewall will be configured to permit access to the site. After reaching the web server that web server will interact with the PHP code. This code will run queries on the database and produce the results in HTML in the browser to serve the users query.

Microsoft Azure System Architecture

Current System Architecture
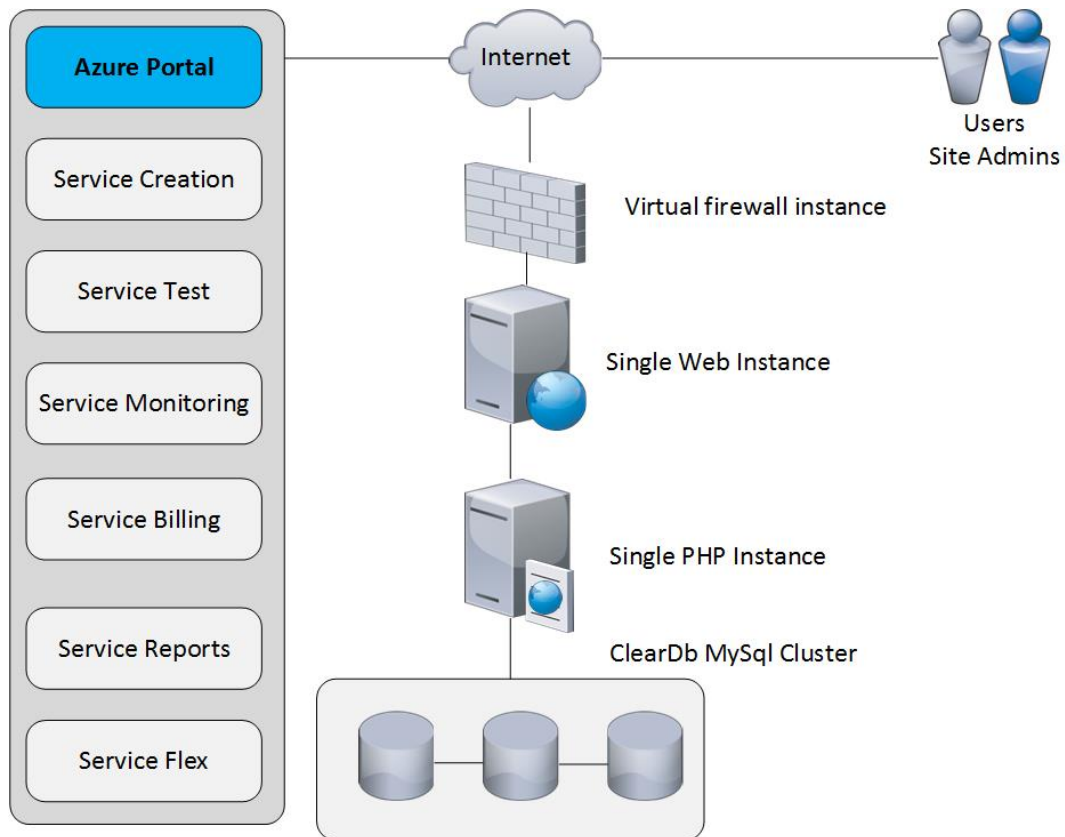
Figure 16: MS Azure Current Architecture

The current system architecture comprises of:

- The MS Azure portal
- A 3-node MySql database cluster
- A single PHP instance
- A single web instance
- A single virtual firewall instance

Scalable System Architecture
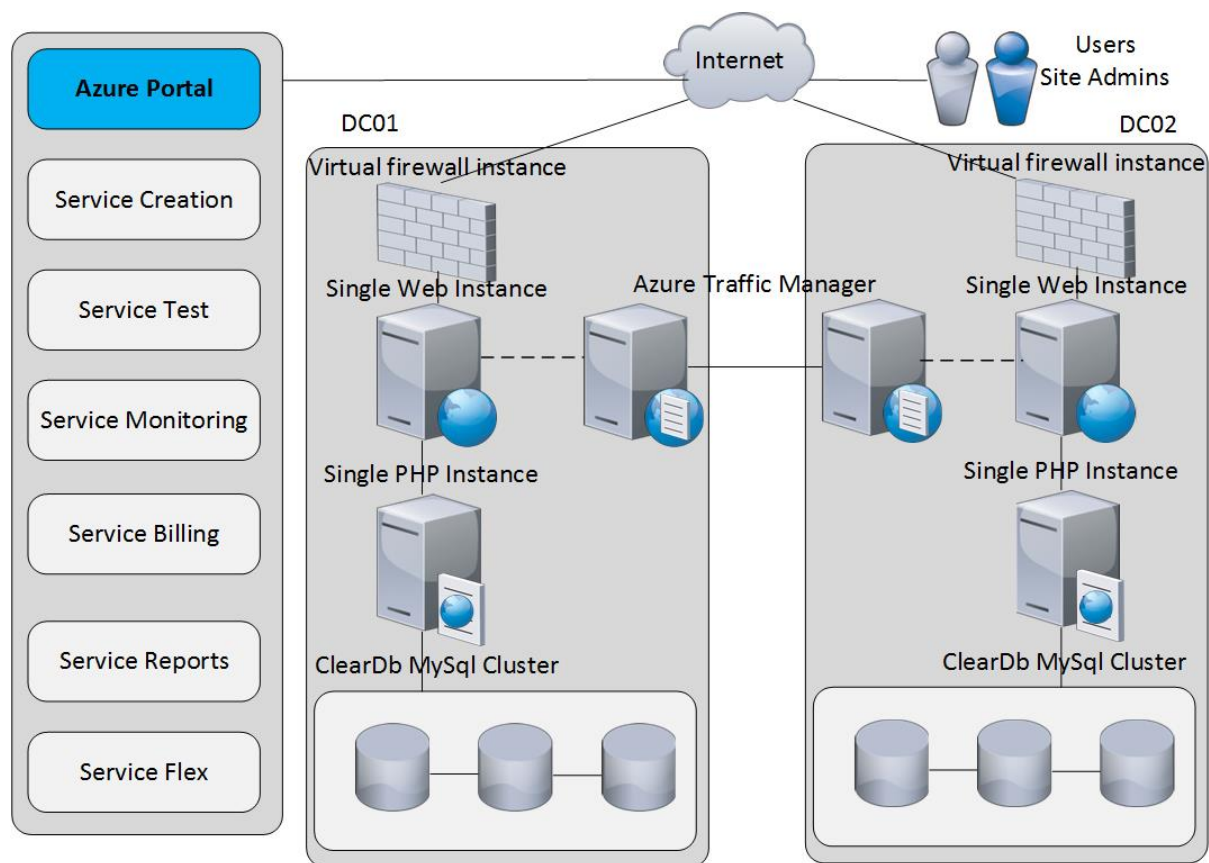
Figure 17: MS Azure Scalable Architecture

The scalable MS Azure Architecture:

- This can provide scale for performance and resilience but also disaster recovery because it is geo-distributed. The key function here for manging the geo load-balancing is the azure traffic manager. It uses a DNS technique for resolving two sites once they are available.
- Moving to this architecture is a very simple, flexible activation from the Azure portal.

System Flow
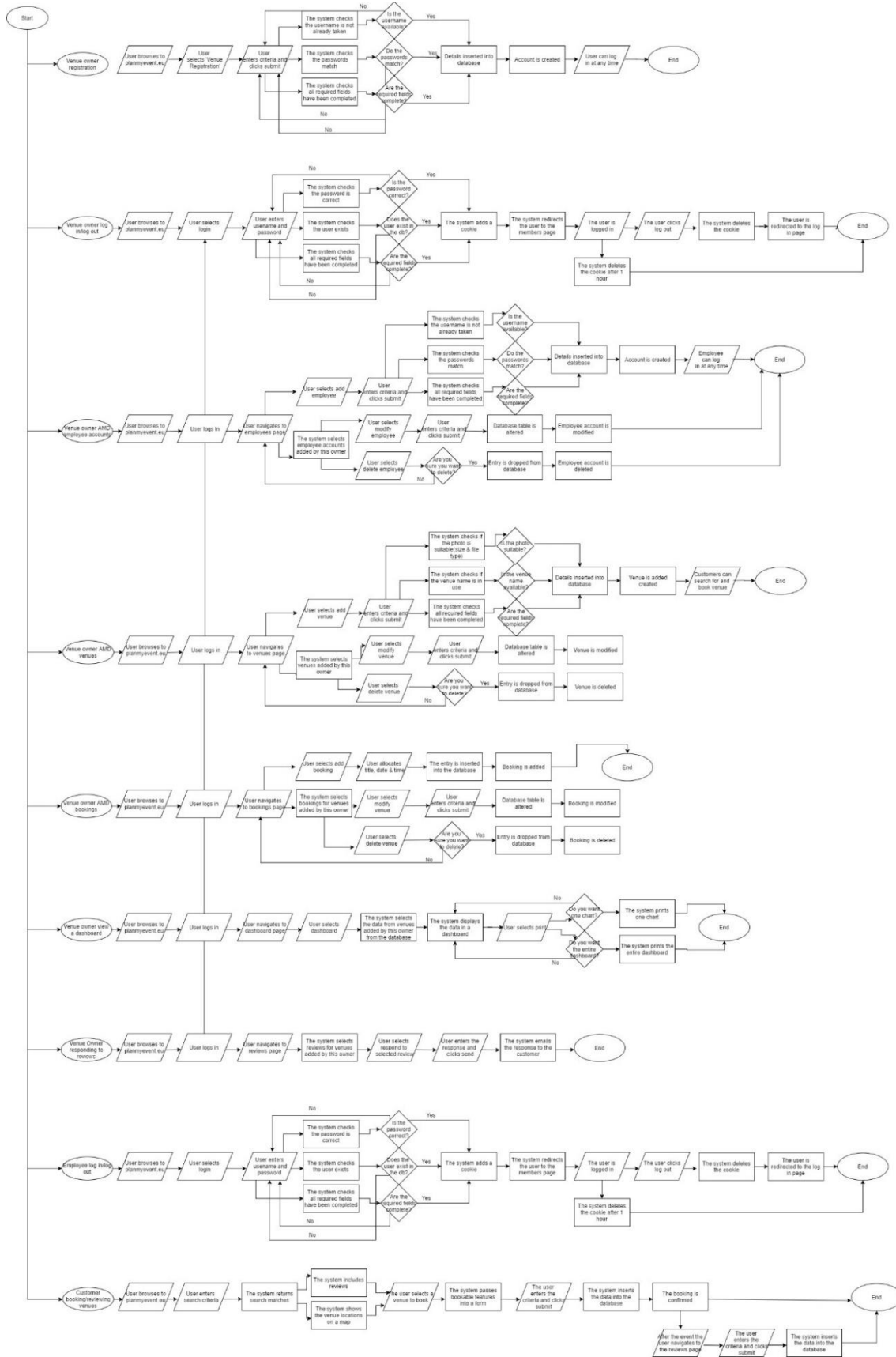
Figure 18: System Flow

Database Design

The ERD is a data model used to show the framework of the database. It contains tables which are joined by one-to-many relationships. The tables contain attributes which will be used to store information that is entered into the forms. The attributes are subject to change during development where needed.
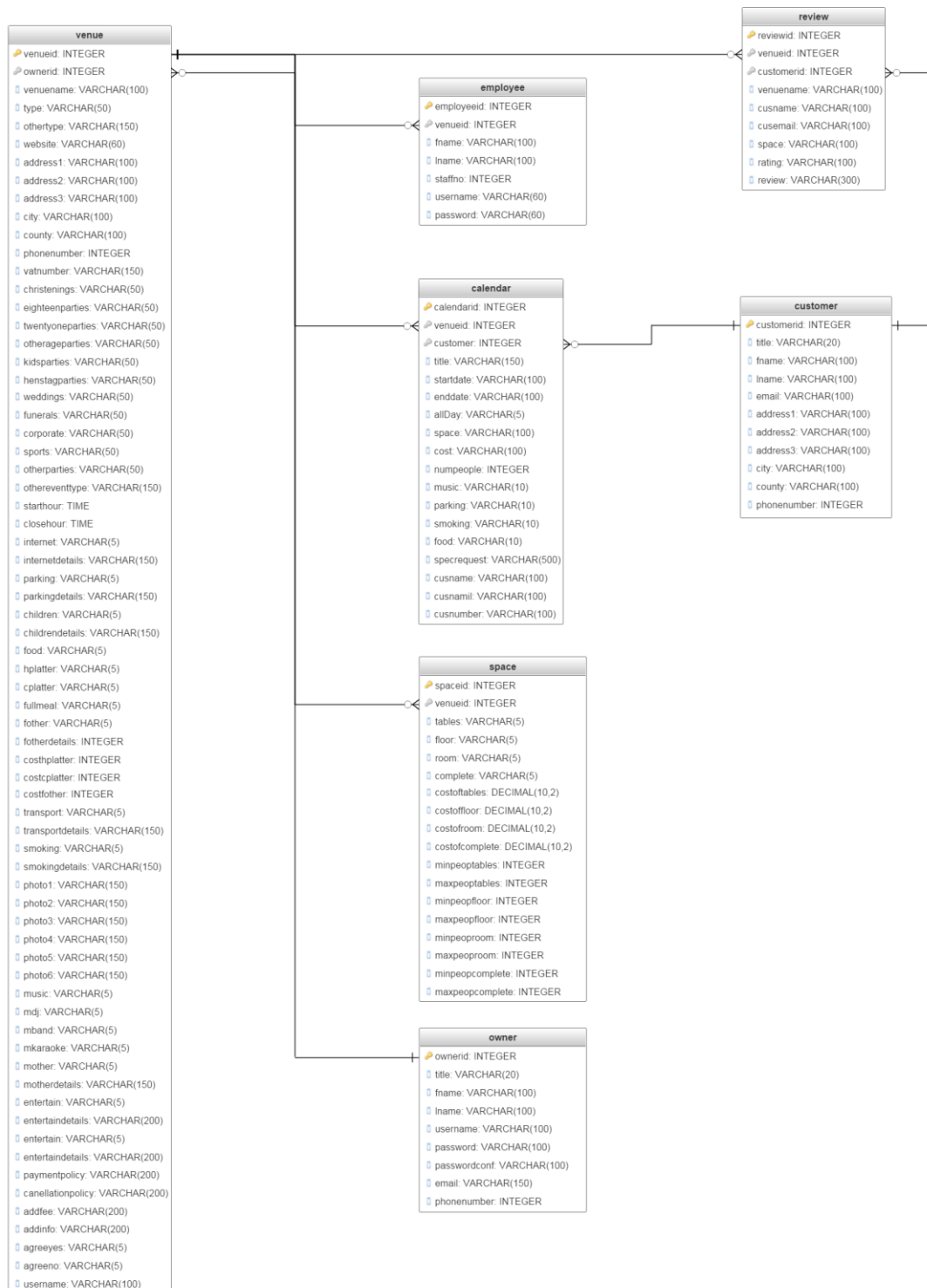
Figure 19: Database Design (ERD)

3.   Programming

Requirements

Functional Requirements

The requirements must be programmed in order of priority. Starting with 'Must' haves then 'Should' haves.

FR_001 Venue Owner account creation – M

Venue owners must register for an account before they can login and add venues. The venue registration page can be accessed via the navigation bar. The form contains 8 fields, 3 of which are mandatory, username, password and password confirmation. At this stage security requirements NF-S003 (Passwords must be encrypted with sha1 encryption) and NF-S004 (Usernames and passwords must match before authenticated access is granted to the user) are met. All code can be found in **register.php.** The code was used from: (ThoughtCo., 2016) The tutorial for this login script uses md5 encryption which is now vulnerable. It has been changed to encrypt the user passwords with sha1 encryption. An example of a string hashed using sha1 is: a94a8fe5ccb19ba61c4c0873d391e987982fbbd3. get_magic_quotes_gpc() is also used to add slashes to strings in preparation for being entered into the database.

```php
//This makes sure they did not leave any fields blank – mandatory fields
if (!$_POST['username'] | !$_POST['pass'] | !$_POST['pass2'] ) {
     die('You did not complete all of the required fields please go back
and try again. </br> <a href="register.php">Click here</a>');
}

// checks if the username is in use – this means the username is unique to
each user that registers
if (!get_magic_quotes_gpc()) {
     $_POST['username'] = addslashes($_POST['username']);
}

$usercheck = $_POST['username'];
$check = mysql_query("SELECT username FROM owner WHERE username =
'$usercheck'")
or die(mysql_error());
$check2 = mysql_num_rows($check);

//if the name exists it gives an error
if ($check2 != 0) {
     die('Sorry, the username '.$_POST['username'].' is already in use.');
}

// this makes sure both passwords entered match
if ($_POST['pass'] != $_POST['pass2']) {
     die('Your passwords did not match. ');
}

// here we encrypt the password and add slashes if needed
$_POST['pass'] = sha1($_POST['pass']);
```

Once the account is created the venue owners can login via **login.php** and are then directed to **members.php.** The members page is purely used for navigation, the slider effect was created using JQuery plugin swiper: (idangero, 2017). The images for the swiper were created using: (LogoMakr, 2017) Here, graphical user interface requirement GUI_014 (Venue Owner Navbar) is met, code can be found in **navbar2.php** and this is included on every page the venue owner accesses when logged in using a php include.
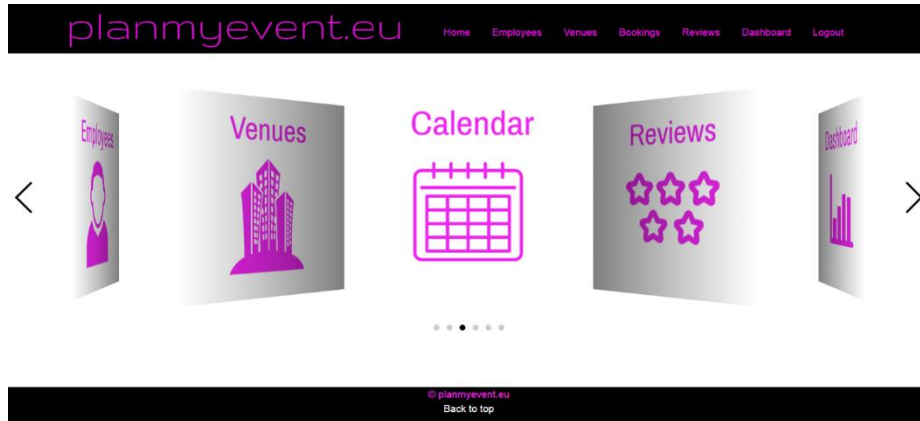
Figure 20: Screenshot: Members Page

FR_003 Venue Owners can AMD Venues – M

Once venue owners have registered for an account they can login and AMD (add, modify & delete) venues.

Add

To create the style for this a JQuery plugin was used: (JQuerySteps, 2017). The code for this page can be found in **vertical.php**. All details that are entered here by the user are added to the database and can then be used to search on. It is broken down into 6 sections:

1. **Venue Details:** Contains the basic details of the venue, the address entered here is used for the search and geocoding feature of the google maps API.
2. **Event Details:** Types of events and details about these events that can be booked/arranged at this venue. The event type is used in the search. Standard prices are also gathered here which are used when booking and are non-negotiable.
3. **Venue Features:** This includes extra features that may be attractive to a customer, interesting statistics are gathered from this showing which features are favorable.
4. **Booking Policies:** Policies outlined by the venue to ensure they are not liable and do not falsely advertise their facilities.
5. **Photos:** This section allows a photo to be uploaded that is produced in the search results for the customer to view.
6. **Agreement:** The agreement acknowledged between the venue and planmyevent.eu. This section is important as it is how planmyevent.eu receives revenue.

Figure 21: Screenshot: JQuery Steps

Once all the information has been entered into the input fields queries are run to insert the data into two tables. The $_POST method is used to collect the form data. In this instance $_POST should be used over $_GET. The $_GET method is insecure as information sent from a form with the GET method is visible in the URL, there is also a limitation on 2000 characters. The $_POST method embeds values within the body of the HTTP request and has no limitations. It is therefore more secure and suitable for this scenario. (w3schools, 2017) mysqli_query() is used to perform a query against the database.

```php
//create two queries
   $query1 = "INSERT INTO venue (venuename, website, address1) VALUES
('".$_POST['venuename']."', '".$_POST['website']."',
'".$_POST['address1']."')";
   $query2 = "INSERT INTO space (tables, floor, room, complete) VALUES
('".$_POST['tables']."', '".$_POST['floor']."', '".$_POST['room']."',
'".$_POST['complete']."')";

//runs the queries
$result = mysqli_query($dbConnection, $query1);
$result = mysqli_query($dbConnection, $query2);

// commits the current transaction for the specified database connection
mysqli_commit($dbConnection);
```

The photos are uploaded to a folder on the web server, and their name is inserted into the database. The code used to create this is from: (w3schools, 2017)

```php
$target_dir= "../images/uploads/"; //specifies the image directory
$target_file= $target_dir . basename( $_FILES['photo1']['name']);
//specifies the path
$uploadOk = 1;
$imageFileType = pathinfo($target_file,PATHINFO_EXTENSION); //holds the
file extension
// Check if image file is a actual image or fake image
if(isset($_POST["submit"])) {
    $check = getimagesize($_FILES["photo1"]["tmp_name"]);
```

```
    if($check !== false) {
        echo "File is an image - " . $check["mime"] . ".";
        $uploadOk = 1;
    } else {
        echo "File is not an image!.";
            echo basename( $_FILES['photo1']['name']);
            $Filename=basename( $_FILES['photo1']['name']);
            echo $Filename;
        $uploadOk = 0;
    }
}
```

Other checks are completed before the image is uploaded:
- Check if file already exists
- Check file size
- Allow certain file formats

Manage – Modify/Delete

All venues added by that owner are selected and listed on **managevenues.php.** From here the
venue owners can view, modify and delete their venues. Delete code can be found in
confirmdven.php and **vendelete.php.** Modify code can be found in **modvendetails.php** and
venmodify.php. The details of the venue selected are passed into a form and from here either
modified or deleted. For the checkboxes if the variable is 'Yes' the checkbox is ticked, if the variable is
'No' the checkbox is not ticked. This means the venue owners to not have to enter every detail when
they modify venues, just what they chose to change.

```
<p> Music:</p>
Yes <input type="radio" name="modmusic"  value="Yes" <?php echo $music ==
'Yes' ? 'checked' : '' ?>/>
No <input type="radio" name="modmusic"  value="No" <?php echo $music ==
'No' ? 'checked' : '' ?>/>
```


Figure 22: Screenshot: Modify Venue

Once venues are added into the database customers must be able to search for them based on type of event and location. Two dropdown menus allow them to select event type and location, code for this can be found in **index.php**. Once they press submit it returns the venues that match the entered criteria. This code can be found in **searchresult.php.**

```
//gets the values from the form using $_POST method
$county = $_POST['county'];
$eventtype = $_POST['eventtype'];

//SQL query
$sql="SELECT venueid, venuename, address1, address2, county, phonenumber,
music, parking, food, smoking, photo1, username FROM venue WHERE
$eventtype='Yes' AND county='$county'";

//runs query
$records=mysql_query($sql);
```

The data is selected from the two tables that the venue information is entered into:

```
while($venue=mysql_fetch_assoc($records)){

//the id of the venues
$num = $venue['venueid'];

//selects everything from the second table that has an id of the venues
selected from the first query
$sql2="SELECT space.spaceID, tables, floor, room, complete, costoftables,
costoffloor, costofroom, costofcomplete FROM space INNER JOIN venue ON
space.spaceID=venue.venueID WHERE space.spaceID = '$num'";

//runs the query
$records2=mysql_query($sql2);

//fetches a result row as an associative array
$venue2=mysql_fetch_assoc($records2);
```

The results are echo-ed into a table provided by: (TableSorter, 2017). The data can be sorted by clicking on the table headers.

Figure 23: Screenshot: Search Results

FR_009 Customers must be able to book venues – M

Once presented with the search results, customers can select a venue to book. The code for this can be found in **bookingdetails.php** and **bookvenues.php.** The $_POST method is used to get the information from hidden form fields on the search results page and echo them into input fields on the booking details page. From here the customer can select what they would like to book. Once they press submit the booking is confirmed and the information is entered into the calendar table. The following details can be filled in:

- **Date/Time:** A beginning and ending date and time can be selected. This was created using: (datetimepicker, n.d.)
- **Space:** This determines what area of the venue the customer reserves and the price.
- **Number of people**
- **Special Requests:** These are a use full statistic to show what changes the venue owners can make based on popular requests.
- **Extra's available:** Based on what is returned from the search results options are made available for the customer to book. For example, if the venue does not provide music then music will not be available for the customer to select. An if statement is used.

Figure 24: Screenshot: Book Venue

```
<tr><td>Extras available:</td><td></br><?php
if ($music == "Yes") {//if the venue provides music produce a checkbox for
music
      echo "<label class='form-check-inline'>
                    <input class='form-check-input' type='checkbox'
name='music' value='Music'> Music
              </label>     &nbsp";
}


if ($parking == "Yes") {//if the venue provides parking produce a checkbox
for parking
      echo "<label class='form-check-inline'>
  <input class='form-check-input' type='checkbox' name='parking'
value='Parking'> Parking Space
</label>      ";
}
```

```php
if ($food == "Yes") {//if the venue provides food produce a checkbox for
food
        echo "<label class='form-check-inline'>
                        <input class='form-check-input' type='checkbox'
name='food' value='Food'> Food
                </label>     &nbsp";
}

if ($smoking == "Yes") {//if the venue provides smoking produce a checkbox
for smoking
        echo "<label class='form-check-inline'>
                        <input class='form-check-input' type='checkbox'
name='smoking' value='Smoking'> Access to smoking area
                </label>     &nbsp";
} ?>
```

FR_004 Venue Owners can view, modify & delete bookings – M

Once the customer has confirmed their booking, the details are inserted into the calendar table in the database and can be viewed from a calendar once a venue owner logs in. The code for this can be found in **bookingcalendar.php** and **process.php**. The calendar will only show bookings for venues that belong to the owner that is logged in:

```
SELECT * FROM calendar WHERE username = '$username
```

From the calendar the venue owners can add, modify and delete bookings. The calendar was developed using JQuery plugin fullcalendar (fullcalendar, 2017) and developed using this tutorial: (jqueryajaxphp, 2015).

Figure 25: Screenshot: Bookings Calendar

FR_006 Venue Owners can view a bookings dashboard – M

Once bookings are entered into the database, useful statistics can be gathered from them. Once venue owners are logged in they can view and print a business intelligence dashboard. The code for this can be found in **dashboard1.php**. To develop these charts, JS Highcharts was used: (highcharts, 2017) The full page can be printed using the window.print() function:

```
<script>
      function myFunction() {
          window.print();
      }
</script>
```


Figure 26: Screenshot: Dashboard

```
//Chart 1
//this query counts the number of bookings that reserved a table at a venue
belonging to the current logged in venue owner
$result1 = mysql_query("SELECT COUNT(space) AS tablecount FROM calendar
WHERE space = 'Table' AND username = '$username'");
$row1 = mysql_fetch_assoc($result1);

//Chart 2
//this query calculates the sum amount cost for the number of bookings that
reserved a table at a venue belonging to the current logged in venue owner
$result5 = mysql_query("SELECT sum(cost) AS tablesum FROM calendar WHERE
space = 'Table' AND username = '$username'");
$row5 = mysql_fetch_assoc($result5);

//Chart 3
//this query counts the number of bookings that reserved a table in January
at a venue belonging to the current logged in venue owner
$result9 = mysql_query("SELECT COUNT(space) AS jantable FROM calendar WHERE
MONTH(startdate) = 01 AND space ='Table' AND username = '$username'");
$row9 = mysql_fetch_assoc($result9);

//Chart 4
//this query counts the number of bookings that included music with their
booking at a venue belonging to the current logged in venue owner
$result60 = mysql_query("SELECT count(music) AS musiccount FROM calendar
WHERE music = 'Music' AND username = '$username'");
```

```
$row60 = mysql_fetch_assoc($result60);

//Chart 5
This query counts the number of each star rating for venues belonging to
the current logged in venue owner
$result64 = mysql_query("SELECT count(review.rating) AS 5rating FROM review
INNER JOIN venue ON venue.venuename = review.venuename WHERE venue.username
= '$username' AND review.rating = '5'");
$row64 = mysql_fetch_assoc($result64);
```

FR_002 Venue Owners can AMD Employee Accounts – S

Venue owners should not allow full access to their accounts and so they can add employee accounts so that employees can view a calendar of bookings but not access the dashboard and manage venues etc.

Add

This code can be found in **addemployees.php.** The process is the same as venue owner registration except when the user logs in the cookies are named differently depending on role so that employees cannot log in as venue owners and venue owners cannot log in as employees.
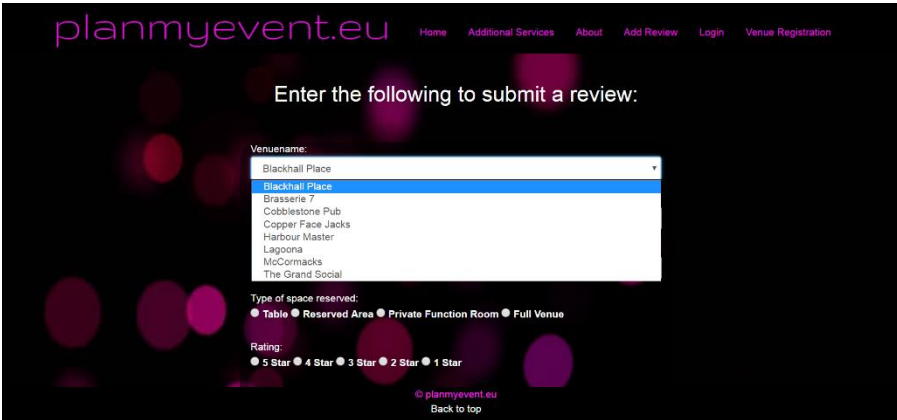
Manage

All employee accounts added by that owner are selected and listed on **manage.php.** From here the venue owners can view, modify and delete their employee accounts. Delete code can be found in **confirmdemp.php** and **empdelete.php.** Modify code can be found in **modempdetails.php** and **empmodify.php.** The details of the employee selected are passed into a form and from here either modified or deleted.

FR_010 Customers should be able to leave reviews after an event – S

A customer should be able to leave a review after they have attended an event at a venue listed on planmyevent.eu. The user must navigate to the 'Add Review' page and submit the form to add a review. The code for this can be found in **reviewadd.php**. The following code selects all venue names and echo's them into a dropdown menu restricting users entering venues that are not listed on planmyevent.eu.

```php
<select class="selectpicker form-control" name="venuename"
style="width:100%;">
    <?php
    $sql="SELECT venuename FROM venue ORDER BY venuename ASC";
    $records=mysql_query($sql);
    while($row=mysql_fetch_assoc($records)){

    echo '<option value="' . $row['venuename'] . '">' . $row['venuename']
. '</option>';
    }
    ?>
</select>
```


Figure 27: Screenshot: Add Review

FR_005 Venue Owners can view and respond to reviews – S

Reviews added by customers will be displayed for the venue owner's venues only once they are logged in, allowing them to respond. The code for this can be found in **reviews.php**, **reviewrespond.php** and **reviewmail.php.**

```
//inner join used to select columns from the review table where the the
venue name in the review table matches the venue name in the venue table
and these venues have a username of the current user logged in.

$sql5="SELECT review.venuename, review.space, review.rating, review.review,
review.cusname, review.cusemail
      FROM review
      INNER JOIN venue ON venue.venuename = review.venuename WHERE
venue.username = '$username' AND review.rating = '5';";


$records5=mysql_query($sql5);
```
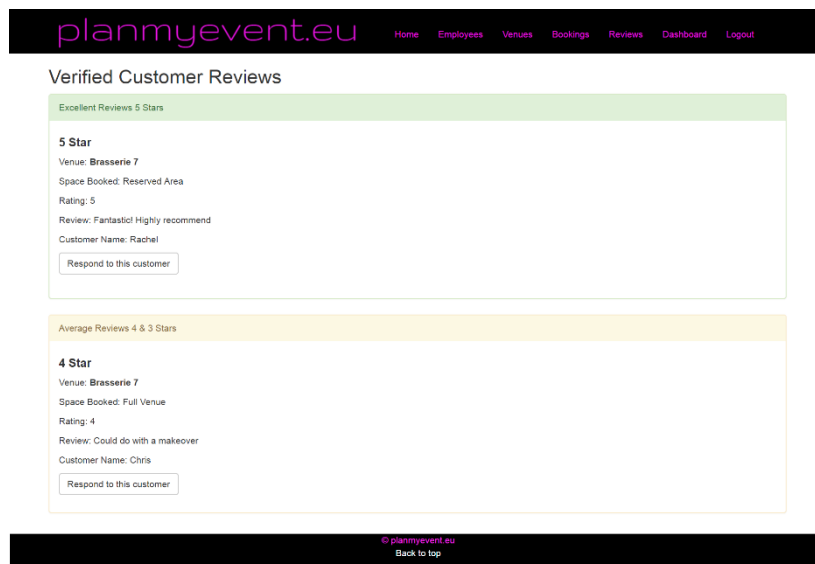

Figure 28: Screenshot: View Reviews

The user completes a form and submits to send a response to the customer by email. The customers email address is passed into the form in a read only field so that it cannot be accidently changed.

```
/* Set e-mail recipient */
$myemail = check_input($_POST['email']);

/* Check all form inputs using check_input function */
$subject = check_input($_POST['subject'], "Enter a subject");
$email = "info@planmyevent.eu";
$message = check_input($_POST['message'], "Write your message");

/* If e-mail is not valid show error message */
if (!preg_match("/([\w\-]+\@[\w\-]+\.[\w\-]+)/", $email))
{
show_error("E-mail address not valid");
}
/* Let's prepare the message for the e-mail */
$message = "
```

```
Please do not reply to this email.  If you have any queries please email:
$email or visit our website for more info: www.planmyevent.eu
Subject: $subject

$message
";


/* Send the message using mail() function */
mail($myemail, $subject, $message);
```

FR_007 Employees can view calendar of bookings – S

Employees can view a calendar of bookings for any bookings, that are for venues belonging to the venue owner that added the current logged in employee account. The code for this can be found in **bookingcalendaremp.php** and **processemp.php**. The employees can view the calendar by day, month or year.

```
//selects the owner username that is associated with the current employee
logged in and creates a variable

$empcal = mysql_query("SELECT username FROM employee WHERE
empusername='$username'");
$row = mysql_fetch_assoc($empcal);
$ownusername = $row['username'];


//Selects from the calendar where that variable is true
$query = mysqli_query($con, "SELECT * FROM calendar WHERE
username='$ownusername'");
```

API Requirements

AP_001 – Google Maps - S

A google maps api is used to pin point where the selected venue is located on a map. The code for this can be found in **geocoding.php.** To make this work the geocoding API was used. This works by converting addresses into geographic coordinates which can then be used to place markers on a map. (googlemaps, 2017)

To do this, in the GoogleMaps API console the geocoding API and JavaScript API. An API key was then provided:

<div align="center">

`AIzaSyD_udpyWPn83FXz5Vpn7cZpRnto07g5FsA`

</div>

The address from the search result is passed into a hidden geocoding form using the $_POST method. When the user clicks on view map, the pin puts a location on the venue.
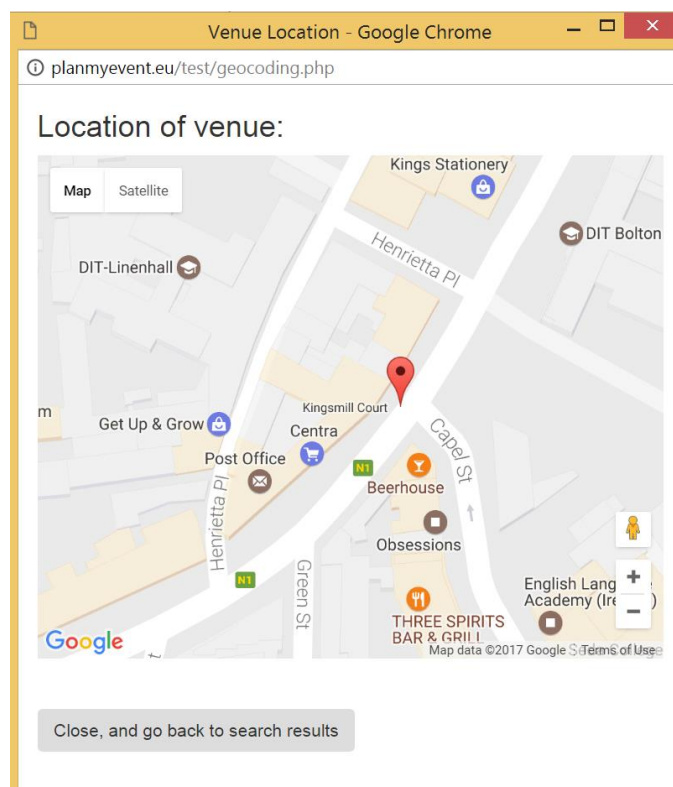
Figure 29: Screenshot: Google Maps

```
<script>
  function initMap() {
    var map = new google.maps.Map(document.getElementById('map'), {
      zoom: 18,
      center: {lat: 53.350140, lng: -6.266155}
    });
    var geocoder = new google.maps.Geocoder();

    $( document ).ready(function() {
            geocodeAddress(geocoder, map);
        });

  function geocodeAddress(geocoder, resultsMap) {
```

```
        var address = "<?php echo $_POST['address'] ?>";
        geocoder.geocode({'address': address}, function(results, status) {
          if (status === 'OK') {
            resultsMap.setCenter(results[0].geometry.location);
            var marker = new google.maps.Marker({
              map: resultsMap,
              position: results[0].geometry.location
            });
          } else {
            alert('Geocode was not successful for the following reason: ' +
status);
          }
        });
      }
        }
    </script>
```

AP_002 - Google Analytics

Google Analytics is used to gather useful statistics from the web service. The following code is included on all pages. And was taken from: (GoogleAnalytics, 2017)

```
<script>

    (function(i,s,o,g,r,a,m){i['GoogleAnalyticsObject']=r;i[r]=i[r]||func
    tion(){
    (i[r].q=i[r].q||[]).push(arguments)},i[r].l=1*new
    Date();a=s.createElement(o),
    m=s.getElementsByTagName(o)[0];a.async=1;a.src=g;m.parentNode.insertB
    efore(a,m)
    })(window,document,'script','https://www.google-
    analytics.com/analytics.js','ga');

    ga('create', 'UA-98752849-1', 'auto');
    ga('send', 'pageview');

</script>
```

4. Testing

Application/Interface Unit Testing

The application/interface unit testing aims to determine if there are any defects present in the interface. This testing primarily focuses on the achievement of the functional requirements.

Customer Tests

Test	Instruction	Results	Testers Comments
Navigation Bar	Click on each link on the navigation bar and determine if it arrives at the expected page.	Pass	
Footer	Click 'Back to Top' on the footer of each page and determine it scrolls back to the top of that webpage	Pass	
Additional Services	Navigate to the additional services page. Click on each title and determine it navigates to that section. Example, click Music and ensure it arrives at music	Pass	
About – Email	Navigate to the about us page. Fill in the form to send an email and ensure it returns 'Email Sent'. Log into the planmyevent email account and ensure the email has been received. Subject line should read "Customer Query". Respond to the email and ensure it is received by the customer test email account.	Pass	
Search	On the home page search for a venue by selecting a location and event type from the drop down. (The database contains venues for Christenings in Dublin 7 and Co. Dublin or Other Age Parties in Dublin 2)	Pass	
Search Result - Table	Determine the search results can be sorted by clicking on the table headers	Pass	
Search Result - Google Maps	Determine the venue location matches the pin point when the user clicks 'View on Map' and it opens in a pop up window. Clicking on 'Close, and go back to search results' closes the pop up window.	Pass	
Search Result - Booking	Clicking on 'Book this Venue' opens in a pop up window. Clicking on 'Cancel, and go back to search results' closes the pop up window. The name of the venue selected appears at the top. Ensure the user can select a start and end date and time. The user must be able to select an area to reserve based on what is produced in the search results. Enter number of people. The extras available checkboxes must only display if that result is true in the search results. Eg. If music is 'yes' the checkbox for music must be there. Enter a special request. Enter name, email and number. You will not be able to continue without entering @ in the email field. Click 'Book this Venue' and it says booking confirmed, name of venue. Clicking on 'Close, and go back to search results' closes the pop up window.	Pass	
Review	Navigate to the add review page. Fill in the form to submit a review and confirm it returns review submitted.	Pass	

Table 36: AT: Consumer Tests

Venue Owner Tests

Test	Instruction	Results	Testers Comments
Venue Owner Registration	Navigate to the venue registration page. Failing to fill out all fields will ask you to return and try again. Entering passwords that do not match will ask you to try again. Entering a username that already exists (sd) will ask you to try again. You will not be able to continue without entering @ in the email field. Enter details and click 'Submit Registration'	Pass	
Venue Owner Login	Login using the account just created via the venue owner/manager login.	Pass	
Navigation bar	Click on each link on the navigation bar and determine if it arrives at the expected page.	Pass	
Navigation Swiper	Click on each link on the navigation swiper and determine if it arrives at the expected page.	Fail	Works intermittently
AMD employees	Navigate to the employee's page. Add an employee account by filling in the form and pressing submit. Navigate to the manage employees page. Delete an employee, refresh the page and determine it is deleted. Modify an employee, refresh the page and determine the modifications have been implemented.	Pass	
AMD Venues	Navigate to the venues page. Add a venue, include adding a photo. Press submit. Navigate to the manage venues page. Delete a venue, refresh the page and determine it is deleted. Modify a venue, refresh the page and determine the modifications have been implemented.	Pass	
View Bookings	Navigate to the bookings page and determine bookings can be viewed by day, week, month.	Pass	
AMD Bookings	Navigate to the bookings page and determine a booking can be deleted by dragging it to the trash can. Determine it can be modified by changing the time and venue name.	Pass	
View/Respond to reviews	Navigate to the reviews page. Determine 5, 4, 3, 2, 1 star reviews can be viewed. Click on any of the reviews to determine they can be responded to. Check the customers test email account to ensure the email has been received. Subject line should read "Review Response"	Pass	
View Dashboard	Navigate to the dashboard page. Determine all charts are visible. Determine all information can be read from charts and can be printed individually and the entire page can be printed. Determine the values in the charts match the values in the database.	Pass	
View account details	Navigate to the 'Your Details' page and ensure the listed details match that of the current logged in user.	Pass	

Table 37: AT: Venue Owner Tests

Employee Tests

Test	Instruction	Results	Testers Comments
Login	Determine the account created by the venue owner is valid, and the user can log in and log out.	Pass	
View Calendar	Determine the user can view a calendar of bookings made for venues owned by the venue owner that created the employee account.	Pass	

Table 38: AT: Employee Tests

Comparison Testing

The comparison test aims to compare planmyevent.eu with the two similar web services discovered during research.

Test	Instruction	Plan my event	Find a venue	Venue search
Search	Does the web service provide a search feature that allows the user to search for venues?	✓		✓
Search Criteria	Does the web service allow the user to specify event type or location when searching?	✓		✓
Search Map	Does the web service allow you to search for a venue location via a map?			✓
Booking	Does the web service provide the ability to book venues directly through the site?	✓		
Reviews	Does the web service provide the ability to add a review for a venue?	✓		
About	Does the web service provide an about us, contact or FAQ page?	✓	✓	✓
Registration	Does the web service allow venue owners to register for an account?	✓		✓
Manage Venue	Does the web service allow venue owners to manage their own venue?	✓		✓
Manage Venues	Does the web service allow venue owners to manage multiple venues?	✓		
Usability	Is the web service attractive, modern and easy to use?	✓	✓	✓
Venue Location	Does the web service pin point the venue location on a map?	✓		✓
Business Intelligence	Does the web service provide useful business intelligence statistics for the venue owner?	✓		
Calendar	Does the web service provide a calendar of bookings for the venue owner?	✓		
Advertising	Does the web service provide advertising for additional event related services?	✓		

Table 39: Comparison Tests

Software Performance Testing

A performance test was conducted on the Azure web application. The load was 250 users and the duration was 1 minute. The results show that the average response time is 0.03 seconds. The CPU was at 3.69 which is significantly below the maximum threshold available. Memory usage was at 3.02, this is also significantly below the maximum threshold. The overall results indicate that the service is more than performant.
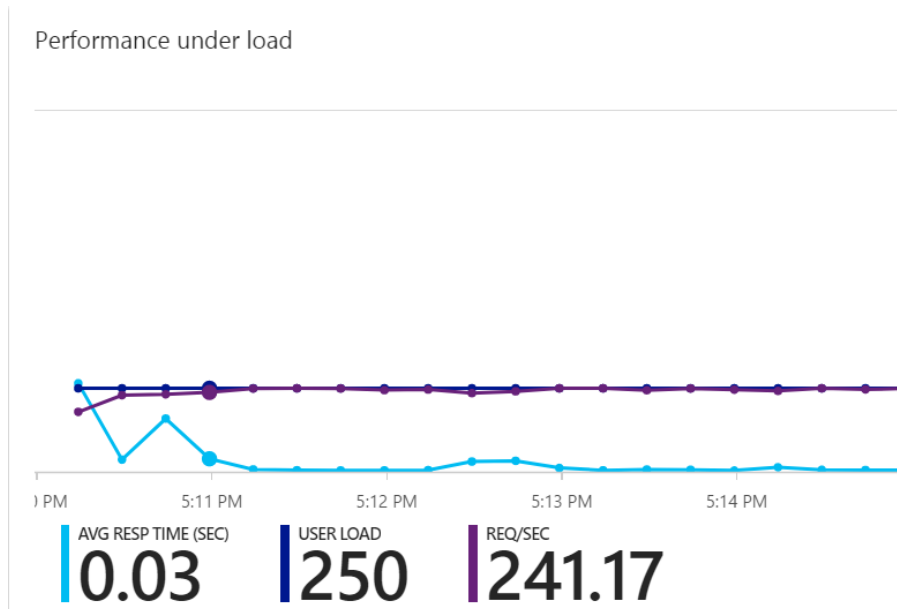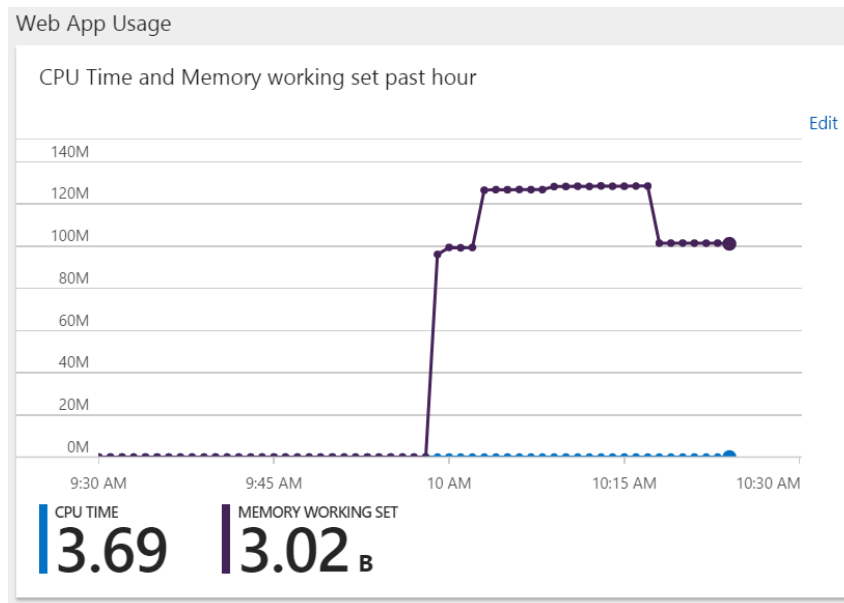

Figure 30: Screenshot: PerformanceTest1

Figure 31: Screenshot: PerformanceTest2

User Acceptance Testing

Customer Tests

Test	Instruction	Result	Testers Comments
Access the web service	Can you easily access the web service using the URL provided?	Pass	
Use the search	Did you find the search easy to use? And did it return what you were expecting?	Pass	Yes, but it would be nice if you could search on other things as well such as price.
Search Results	Could you read the search results clearly and understand them?	Pass	
Google Maps	Could you view the venue location easily on the map? Was the pin located where you would have expected it to be?	Pass	
Reviews	Could you access the reviews easily and read them clearly?	Pass	
Book Venue	Could you book the venue you wanted easily?	Pass	Yes, but the date and time picker was difficult to use. It would also be great if I could pay for it at the time of booking.
Add Services	Could you find the additional services page and use it with ease?	Pass	
Add Review	Could you add a review with ease	Pass	
About	Could you find the about page add ease? Was the information provided on it useful to you? And if not, were you able to submit a question with ease?	Pass	

Table 40: UA: Consumer Tests

Venue Owner Tests

Test	Instruction	Result	Testers Comments
Registration	Could you register for an account with ease?	Pass	Yes, however the username I had chosen was already taken. When I went to try again everything I had entered was empty meaning I had to enter it all again.
Login/Logout	Could you log in and log out of your account easily?	Pass	
Swiper/navbar	Could you navigate easily using the navbar or swiper menu	Fail	The menu at the top worked fine but the swiper menu only works sometimes
Add Employees	Could you add employee accounts easily?	Pass	
Manage Employees	Could you manage employees at ease? Including deleting and modifying.	Pass	

Add Venues	Could you add venues as expected? Did it include all the fields you would need to add the details of your venue?	Pass	Would be handy to have a special promotions section.
Manage Venues	Could you use the manage venues feature with ease? Including modify and delete.	Pass	
Bookings	Could you view and utilise the bookings calendar as expected?	Pass	It would be useful to have more information related to the booking.
Reviews	Could you read and respond to reviews easily?	Pass	
Dashboard	Could you use the dashboard as expected? Including reading information from all charts and printing.	Pass	As I use this service it would be great to see more charts

Table 41: UA: Venue Owner Tests

Employee Tests

Test	Instruction	Results	Testers Comments
Log in/log out	Could you log into your account and log out easily?	Pass	
View Calendar	Could you view and use the calendar with ease?	Pass	

Table 42: UA: Employee Tests

5.   Conversion

Infrastructure Conversion

There are two sets of infrastructure:

- Staging site
- Live site

Initial Development: All development once completed is tested on a staging site before it is released to the live site. Initially it was developed on register365 hosting space, this is the staging site and can be accessed at planmyevent.eu/test. Once complete and fully tested it is then migrated to Microsoft Azures Web Applications allowing for scalability of the web service should it be required, this is the live site and can be accessed at pme12.azurewebsites.net/test/ . Hosting on Microsoft Azure will provide a service that will comply with the availability requirements.

Continued Development: Any updates or developments will always be fully developed and tested on the staging site before being migrated to the live site. By doing this any outages due to bugs or any disruption during development and testing will be contained to the staging site and therefore does not cause any interruption to the service of the live site. Any updates to the live site will be implemented between 1am and 5am on Saturday nights meeting non-functional requirement NF-A_002.

Migration Process:

1. **Export existing database:** Within register365 the database is exported into a SQL script. This script is used to create the schema for the new database and populate it with any data that resides in the existing database.
2. **Create Azure database:** To connect the MySQL database on Azure by logging in via MySQL workbench was used. Once logged in the script can be run.
3. **Transfer web pages:** Download all webpages using FTP from the staging web application and upload them to the live web application also using FTP.
4. **Update connection strings:** Update the connection string in **dbconnection.php** to point to the Azure database.
5. **Test:** Test all functionality is working as expected.

Business Conversion

Planmyevent would hope to obtain 2-3 venue owners to take part in the initial pilot. The pilot would ideally run for 2-4 months. Any business conversion issues would be addressed during the pilot.

As a start-up, the most suitable method of conversion is the parallel method. Parallel conversion strategy can be quite costly as it involves running two systems at once. For this reason, planmyevent.eu offers 2-4 months' free trial to venue owners taking part in the initial pilot. This allows venue owners to keep their current service running whilst comparing features and usability with planmyevent.eu.

Once the trial is complete, a direct cutover strategy will be used to move venue owners from their old systems and fully implement planmyevent.eu as the sole system for their venue and booking management. A 30-day free trial is offered to new venue owners who wish to trial the system.

6. Production & Maintenance

Production

Once the site is live it is in production. All production statistics can be analysed via google analytics. Stats show how the website is growing, what locations are accessing it, what browsers are being used to access it and much more. This data is important for determining the business value of the service, if it is meeting the correct volume of users and the appropriate volume of business is on the correct growth path. This will give good indication of what parts of the service needs further enhancements to further enrich the development of the service and subsequently grow more business.

Figure 32: Screenshot: Google Analytics

Maintenance

The software lifecycle management will be coordinated through GitHub. The infrastructure management will be managed using Microsoft Azure. The database is currently hosted on a clearDB 3-node cluster provided and managed through the Azure portal. Providing a single management platform.

Figure 33: Screenshot: ClearDB

Scalability Options

A number of options are available within Azure that can be enabled to automatically scale up or scale out the web service when it reaches a pre-defined performance threshold.
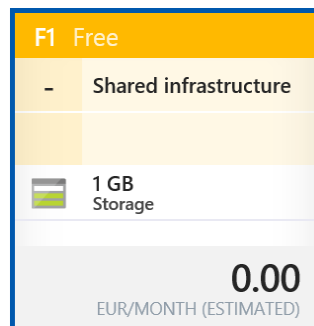
Currently the web app is in this plan:

Figure 34: Screenshot: WebAppPlan

For local performance scalability of the web app one of the following options is available:

Figure 35: Screenshot: ScaleWebApp1

For high resilience and high scalability including disaster recovery and business continuity features for the web app one of the following options is available: (Traffic manager is a global load balancer and uses a DNS technique to load balance between sites in different geo locations. One of these options must be selected to implement a disaster recovery plan)

Figure 36: Screenshot: ScaleWebApp2

Currently the database is on this plan:

Figure 37: Screenshot: DBPlan

To scale up the database, the following options are available: (Venus or Saturn plan would be required for Disaster Recovery)

Figure 38: Screenshot: ScaleDBPlan1

System Evaluation and Further Development

On evaluation, the system architecture is fully scalable and resilient. For this reason, this will not be the focus of the next version. Based on feedback from user acceptance testing and from the pilot deployment group the web service requires focus on improving user functionality. This will be the primary focus for the short-term system evolution road map. If user acceptance is high this will increase use of the web service and lead to scalability being the focus of the long-term road map.

System Evolution Road Map

The system road map outlines the short term and long term goals for the development of planmyevent.eu. Over time the system will be developed to have the following features:

Version 2

- **Secure payment:** method through the service such as verified by visa.
- **Extra Details in calendar:** When customers book extra features, they are included in the booking calendar for the venue owner to view.
- **Double Booking:** Venues become unavailable to book if they have already been booked for the specified date and time.

Version 3

- **Venue Pages:** Individual pages for each venue.
- **Business Intelligence:** As each venue will have their own page and website traffic increases, targeted advertising based on user searches and page visits.
- **Searching Efficiency:** Increased efficiency in the search results, including filters.

Version 4

- **Customer registration:** to make booking and payment faster
- **Application 1:** An application that can be installed on venue owner's computers
- **Application 2:** An application available to customers on all mobile phone operating systems app stores.

Bibliography

Azure, 2016. *Web Apps.* [Online]
Available at: https://azure.microsoft.com/en-us/services/app-service/web/
[Accessed November 2016].

Azure, 2017. *Azure Pricing Calculator.* [Online]
Available at: https://azure.microsoft.com/en-gb/pricing/calculator/

Balsamiq, 2016. *Balsamiq.* [Online]
Available at: https://balsamiq.com/
[Accessed November 2016].

datetimepicker, n.d. *Bootstrap Datetime Picker.* [Online]
Available at: http://tarruda.github.io/bootstrap-datetimepicker/

Draw.io, 2016. *Draw.io.* [Online]
Available at: https://www.draw.io/
[Accessed November, December 2016].

fullcalendar, 2017. [Online]
Available at: https://fullcalendar.io/

GitHub, 2016. *GitHub.* [Online]
Available at: https://github.com/
[Accessed November 2016].

Google, 2016. *Google Images.* [Online]
Available at: https://images.google.com/
[Accessed November/December 2016].

GoogleAnalytics, 2017. *GoogleAnalytics.* [Online]
Available at: https://www.google.ie/analytics/#?modal_active=none

googlemaps, 2017. *Geocoding.* [Online]
Available at: https://developers.google.com/maps/documentation/geocoding/intro

highcharts, 2017. [Online]
Available at: https://www.highcharts.com/

idangero, 2017. *Swiper.* [Online]
Available at: http://idangero.us/swiper/#.WQn6vVXyvX4

jqueryajaxphp, 2015. *fullcalendar-crud-jquery-php.* [Online]
Available at: https://jqueryajaxphp.com/fullcalendar-crud-jquery-php/

JQuerySteps, 2017. [Online]
Available at: http://www.jquery-steps.com/

Logomakr, 2016. *logomakr.* [Online]
Available at: https://logomakr.com/
[Accessed November 2016].

LogoMakr, 2017. [Online]
Available at: https://logomakr.com/

PHP, 2016. *What is PHP.* [Online]
Available at: http://php.net/manual/en/intro-whatis.php
[Accessed November 2016].

Register365, 2016. *Home.* [Online]
Available at: https://www.register365.com/
[Accessed November 2016].

SurveyMonkey, 2016. *SurveyMonkey.* [Online]
Available at: https://www.surveymonkey.com/
[Accessed November 2016].

TableSorter, 2017. [Online]
Available at: http://tablesorter.com/docs/

ThoughtCo., 2016. *PHP login script.* [Online]
Available at: https://www.thoughtco.com/php-login-script-p2-2693850#step2

Turkeylinux, 2016. *lampstack.* [Online]
Available at: https://www.turnkeylinux.org/lampstack
[Accessed December 2016].

w3schools, 2016. *Bootstrap - Get Started.* [Online]
Available at: http://www.w3schools.com/bootstrap/bootstrap_get_started.asp
[Accessed November 2016].

w3schools, 2016. *JavaScript Introduction.* [Online]
Available at: http://www.w3schools.com/js/js_intro.asp
[Accessed November 2016].

w3schools, 2016. *SQL Introduction.* [Online]
Available at: http://www.w3schools.com/sql/sql_intro.asp
[Accessed November 2016].

w3schools, 2017. *php file upload.* [Online]
Available at: https://www.w3schools.com/php/php_file_upload.asp

w3schools, 2017. *php forms.* [Online]
Available at: https://www.w3schools.com/php/php_forms.asp

Waters, K., 2016. *2009.* [Online]
Available at: http://www.allaboutagile.com/prioritization-using-moscow/
[Accessed November 2016].

WinSCP, 2016. *Introducing WinSCP.* [Online]
Available at: https://winscp.net/eng/docs/introduction
[Accessed November 2016].

Appendix

Glossary of Terms

AMD	Add, Modify, Delete
ERD	Entity Relationship Diagram
API	Application Programming Interface
GUI	Graphical User Interface
DNS	Domain Name Server
MS	Microsoft
FTP	File Transfer Protocol

Table 43: Glossary of Terms

Project Proposal

Project Scope
The scope of the project is to develop a web service for booking venues. The system shall have a search algorithm that provides the user with a list of search results for venues based on the criteria entered by the user. The user must then be able to book this venue through the web service. The venues will be added by the venue owners themselves reducing the administration of this task for the developer. All requirements are specified below in the relevant sections.

Project Plan & Design
The system must be fully planned and designed before any implementation. We must understand the time scale, the resources available, the project scope and the risks involved. The project design must satisfy all specified functional and non-functional requirements. The services must be defined and the steps necessary to deliver them. The system architecture be defined by a use case diagram, entity relationship diagram and class diagram. These diagrams will explore and display the logic associated with the business intelligence of the service.

The infrastructure will be based on a multi-tier system architecture consisting of 3 tiers; presentation, logic and data. The presentation tier is the top-most level of the application. It will be used to display the user interface such as the search results. The logic tier will coordinate the application by processing commands and performing calculations. The search algorithm will work in this tier. The data tier includes the database that is used to store information such as the venue or user details. This information is utilised by the logic tier and then presented to the user. The web service will be hosted via Microsoft Azure's Web Application Service.

Documentation
All aspects of the system functionality and design, planning and implementation of the system must be documented.
The following documents will be provided: -

- Project Proposal including project plan
- Reflective Journal
- Requirements Specification including functional and non-functional requirements, use case diagram, Entity Relationship diagram, class diagram and flow description
- Technical Report including implementation details and testing results.

Special Resources Required
- Project Supervisor
- Infrastructure for testing and live platforms
- A fast and reliable computer for development

Technical Details
The website domain will be registered with register365 and the website will be hosted via Microsoft Azure's Web Applications Feature. WinSCP will be used to transfer files onto the server known as File Transfer Protocol (FTP). PHP, JavaScript, jQuery and Bootstrap will be used to develop the front end of the website. SQL will be used to create, insert, update and delete all aspects of the database. A search algorithm will be used to return results that best match the user's specifications. GitHub will be used for development version control.

## Evaluation
The system will be evaluated with the methods outlined in the technical approach. System Integration Testing will be done continuously throughout the development to ensure the maximum benefit is obtained from a multi-tier system architecture. Beta Testing, Software Performance Testing and Comparison Testing will be implemented during and after development before the system goes live.

User acceptance testing can be considered the most important test. It will describe the overall goal which is how the user interacts with the system. If this test fails, the development and implementation has failed.

Project Plan

The following Gantt Charts shows the tasks involved in this project against the time in which they must be completed. The names of the tasks are shown on the left and the dates at the top. Figure 1.1 shows the Gant Chart in day view. Figure 1.2 shows the Gant Chart in week view.
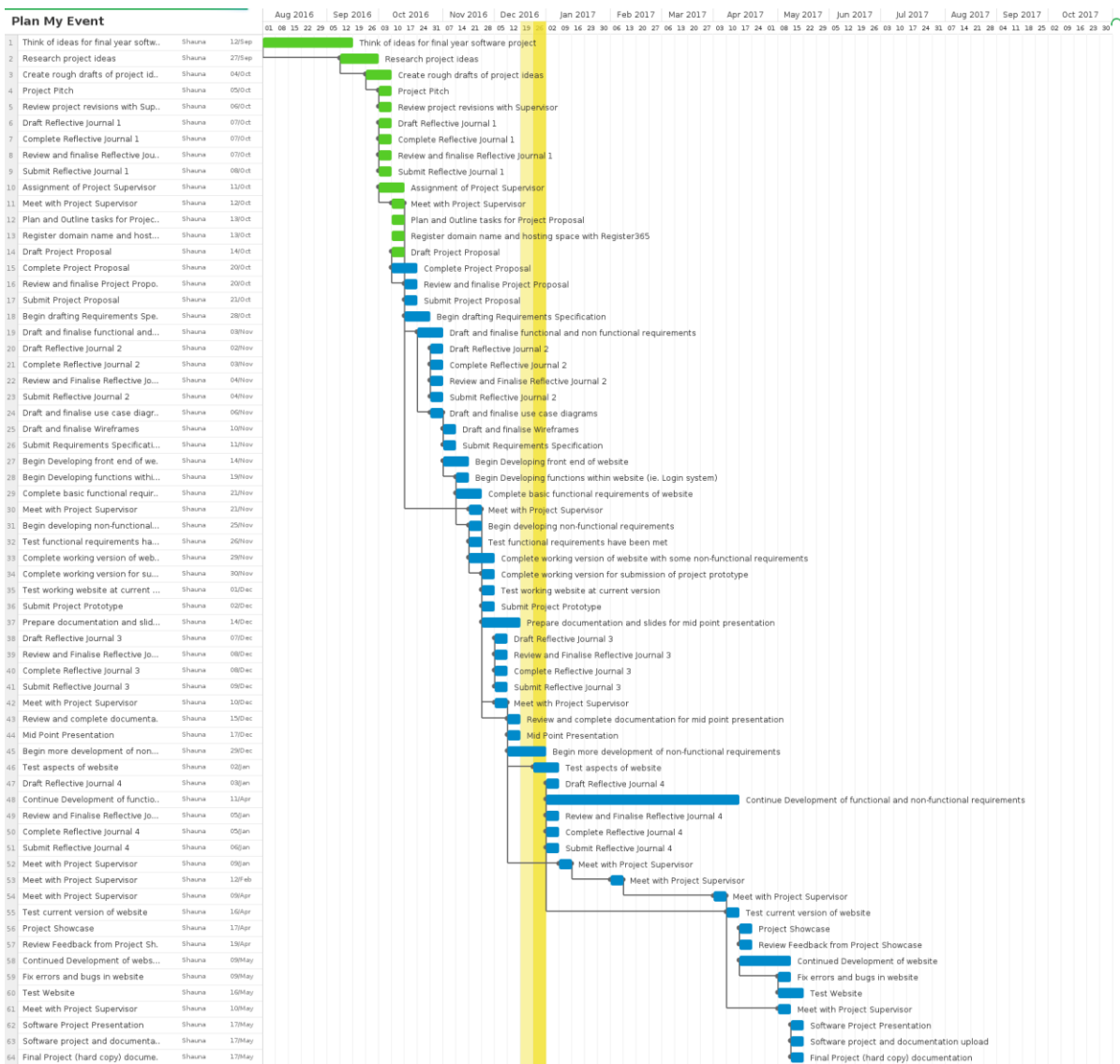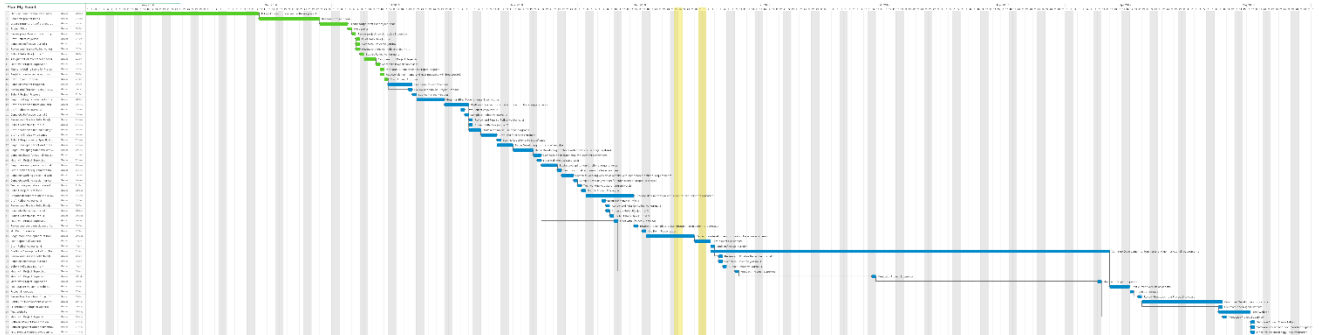
Green   **Tasks Completed**
Blue   **Tasks to be Completed**
Red   **Tasks overdue**
Yellow   **Days allocated to holiday**

Monthly Journals

Reflective Journal 01 - 07th October 2016

The following journal will be completed by me each month to show a reflection of my objectives, efforts, achievements and thoughts on my final year software project.

Introduction

I have spent the last two years (at least) thinking and planning ideas for my final year software project. It is not something that I have looked forward to as I know it is going to be a big challenge for me, maybe my biggest challenge yet. I originally started my time at NCI in the Computing programme, however I struggled to find my passion for writing code and unfortunately my grades reflected this. In 3rd year I became very sick and missed a lot of classes. Because of this I made the decision to defer part of my year. With this in mind, and with the assistance of computing support I also made the decision to change to the Business and Information Systems programme. Since then I haven't looked back and are starting to enjoy my lectures and assignments. It has given me the opportunity to complete two years of work placement and to start recognising my own strengths and weaknesses.

Objectives & Achievements
This month I set the following objectives:

- Revision, revision, revision! - Review some tutorials on YouTube and any relevant information so I am ready to begin my project.
- Select an idea. – Write down all project ideas and select which one I want to do.
- Meet with a lecturer to discuss my ideas.

This month I achieved the following:

I met with computing support in August to discuss my ideas. I had two main ideas for my project, one is a website for booking and managing events and the other is a workflow for joiners and leavers within organisations. I decided to write down both ideas from start to finish with wireframes for both so I knew exactly what I wanted to do. Whilst the second idea seemed more interesting I realised I did not have enough skills and experience with Active Directory to create this project in a fully working order. So I decided to go with my other idea for the event management website. I am really excited about this idea, it is something I've been thinking about for a long time and as a user something I would use quite regularly.

I met again with computing support to finalise my idea and now I am really excited and enthusiastic about it! I began watching bootstrap and PHP tutorials and also using Balsamiq to design my website.

This month I also had my project pitch. I am normally very confident and good at public speaking but for some reason I was very nervous about this. For my project pitch I had, Keith Maycock, Adriana Chis and Joe Molumby. All 3 were very helpful and gave me very positive feedback and suggestions about my project. I am still awaiting to hear whether my project has been accepted or not. This is quite frustrating as it is already week 3. I would have preferred if the project pitches were done in week 1 or early week 2 so that we can begin working on them.

Reflection and Intended Changes

So far I am moderately pleased with the work I have done on my project however I am already quite stressed. I am looking forward to the tasks that I have been set to complete in 4th year however I am finding the amount of work that is expected to be completed within the time frame given to be very overwhelming. I am finding it hard to balance being in full time education for the first time in two years, whilst also in part time employment and maintaining my health. I intend to work on improving my time management skills for setting and completing tasks within a specific time and I hope that by this time next month I will have completed more of these tasks.

Reflective Journal 02 - 2nd November 2016

The following journal will be completed by me each month to show a reflection of my objectives, efforts, achievements and thoughts on my final year software project.

Objectives & Achievements
This month I set the following objectives:
- Continue to revise content using online tutorials and videos.
- Complete my project proposal.
- Meet with my project supervisor.
- Complete one function of my project

This month I achieved the following:
Once my project was approved I met with my project supervisor Sam Cogan on Monday 17th October. We discussed the functionality behind the idea of my project and the requirements it must satisfy. It is important to have a website that has a good style but the functionality is more important, so I began working on a basic template. So far, the website has a navigation bar, registration form, login and database. I designed the database knowing exactly what tables and columns I would need however I struggled a small bit with noting the relationships. I revised my database notes for this part and was able to complete my design. With my database implemented I was able to grow my registration function on my website and push and pull data from my database. So far, I have really enjoyed this part of the development.

This month I also completed my project proposal. It felt great to get the ideas out of my head and onto paper in a format that anyone can read and understand. However it also made me realise the challenges I have ahead for the development of my project. I know it is important for me to stay on top of my tasks and on time with deadlines.

Reflection and Intended Changes
This month I am very satisfied with the work completed so far on my project. I have managed to complete all the set tasks whilst also completing work for other modules meaning my time management skills have slightly improved. Something I am struggling with is the design of my webpages. Technically, I know how to make it work but I am lacking the artistic flair needed to recognise good colour palettes and logos. Whilst I am aware this part is not the most important I think a successful website is one that easy on the eye for the user as part of good functionality. So far a lot of my time has been spent on adjusting shades of colour and resizing fields, my intended changes for next month is to prioritise aspects and requirements of my site so that my time is allocated correctly. I hope this will be outlined in completion of my projects requirements specification.

Reflective Journal 03 - 9th December 2016

The following journal will be completed by me each month to show a reflection of my objectives, efforts, achievements and thoughts on my final year software project.

Objectives & Achievements

This month I set the following objectives:

- Continue to revise content using online tutorials and videos.
- Complete my requirements specification.
- Meet with my project supervisor.
- Complete one function of my project

This month I achieved the following:

This month I found it difficult to find time to work on my project with the deadlines for other subjects. However, I did a small bit on my prototype and got my requirements completed. I struggled at first with the project requirements because I knew exactly what I wanted my system to do but I needed to break it down into individual components. It also made me recognise the elements I hadn't considered and that I will now need to add them to my project scope. I met with my project supervisor Sam Cogan on Thursday 8th December to finalise the details of what is expected for the mid-point presentation. The results of the meeting defined what needs to be completed for the technical report and project prototype. My technical report is in draft and is almost complete, I am nervous to upload it just yet as it is worth so many marks. I want to make sure its 100% correct and to the highest standard I can produce. I've done a small amount of work on my project prototype since data is already going into the database and being pulled out I've mainly been working on styles and graphics and using CSS and jQuery to implement this. It's been fun but also challenging as web design involves skill in taste and graphic design. For this reason, I have kept the styles basic for the prototype. With it being so close to mid-point it is important for me to stay on top of my tasks and be on time with deadlines.

Reflection and Intended Changes

This month I am somewhat satisfied with the work completed so far on my project I would have liked to spend a little more time on it but it was difficult with the time constraints and workload included in the other modules. However now that I have completed assignments for other modules I will be able to focus solely on my software project. It has been difficult to understand what is expected from us in terms of work and documentation. Each student is being told different things and it is very confusing. My intended change for this is to focus on my own work and if I need any assistance ask Sam or Eamon rather than guess amongst my peers. I am looking forward to the mid-point being complete and hopefully I'll have a lot more time to focus on the development of my project in January after the exams.

Reflective Journal 04 – 6th January 2017

The following journal will be completed by me each month to show a reflection of my objectives, efforts, achievements and thoughts on my final year software project.

Objectives & Achievements

This month I set the following objectives:

- Complete project prototype
- Finalise Technical Report
- Meet with my project supervisor.
- Prepare for midpoint presentation

This month I achieved the following:

This month was a crucial milestone for my project. My midpoint presentation was scheduled for the 19th December and so I was under a lot of pressure to have everything ready. I started with my mid-point technical report. I thought this wouldn't take too long as it would just involve combining my project proposal and requirements document but it actually took a lot longer than I thought and

required a lot more detail. It is very unclear what is expected of the mid-point technical report. In the slides on Moodle it just says a "Detailed Requirements Document" is required but in fact in the marking scheme we are also marked on our project proposal. Also, with not much clarity students are left to guess and this leads to a lot of confusion where we all receive different advice from our supervisors.

I then began working on finalising my project prototype. This was tough as it involved styling a lot of pages that didn't contain any real content and therefore involved spending a lot of time on something that is just going to be removed after the presentation. This is very frustrating. I also had nearly every page designed and 3 strong pieces of functionality but lost makes on my prototype for not having every single detail included in the prototype. Over all I am satisfied with my result but I feel I could have achieved higher had more clarity been provided in terms of what was expected from us for the prototype and technical report.

Reflection and Intended Changes

This month I am somewhat satisfied with the work completed so far on my project. I will find it difficult to focus on my project next month as my top priority will be exam revision. I hope that in the week after the exams that we have off college I can progress a little further. The search algorithm is what scares me the most and I see it as one of my biggest challenges in this project. For that reason I would like to work on that next.

Reflective Journal 05 – 10th February 2017

The following journal will be completed by me each month to show a reflection of my objectives, efforts, achievements and thoughts on my final year software project.

Objectives & Achievements

This month I set the following objectives:

- Review mid-point results
- Set targets for next elements of functionality
- Meet with my project supervisor.

This month I achieved the following:

This month I found it very difficult to get motivated to work on my project. I am so tired after the exams and let down by my midpoint presentation that I am finding it hard to get back into routine. I received 73% in my midpoint presentation. Whilst this is a good mark I was unprepared for some of the questions and left out some of the finer details of my requirements. I am annoyed at this because the specification for the requirements was very unclear and we were told to keep it top level, by not going into enough detail I lost marks. I will work to improve this for the final technical report. After receiving my exam results and achieving over 70% in everything it has really pushed me to aim for a 1:1. I began back working on my project getting the search algorithm working. For the moment, I am just attempting to create a full text search that returns venue names. Once this is working I will be able to add extra features that allow the user to search based on certain criteria. I also intend to set up the calendar of bookings. This will be a new challenge for me but I am looking forward to it.

This month I will also focus on the security elements of my project. Currently passwords are encrypted using MD5 encryption, this is vulnerable and will therefore need to be changed. I also hope to implement an SSL certificate meaning that my website will use https rather than http.

Reflection and Intended Changes

I feel now that I am a little bit behind since I found myself very unmotivated after the exams. For this reason I will start setting weekly targets to ensure I stay on top of my project. I was aiming for a 2:1 but since the results of the exams if I push myself I am capable of a 1:1.

Reflective Journal 06 – 10th March 2017

The following journal will be completed by me each month to show a reflection of my objectives, efforts, achievements and thoughts on my final year software project.

Objectives & Achievements

This month I set the following objectives:

- Aim to meet as many functional requirements as possible
- Complete full booking system
- Meet with my project supervisor.

This month I achieved the following:

This month has been the most difficult for me so far. I constantly feel against the clock, I was making great progress with my project but now must leave it and to try and focus on exams. We are starting to have our last classes ever which is quite a strange feeling. I am continuing to focus on meeting all functional requirements and have a full booking system completed before the exams. For this reason, I have spent a lot of time on the booking calendar. This has probably been the toughest technical challenge so far. Working on selecting the individual venues, inserting them into the database and then selecting them for the calendar. I have also only needed to select them for the venue owner that is logged in, this was tough.

I have had to spend a lot more time this month on other subject assignments and focus for exams.

Reflection and Intended Changes

I must work to complete everything within the timeframe given to us. I am struggling to keep up the speed as I have not had a coding class in a long time and feel like I am having to learn everything from the start again. I definitely intend to stick to the timeline, and achieve as many functional requirements as I can. I need to focus more on the business intelligence aspect rather than presentation and user friendly features.

Reflective Journal 07 – 7th April 2017

The following journal will be completed by me each month to show a reflection of my objectives, efforts, achievements and thoughts on my final year software project.

Objectives & Achievements

This month I set the following objectives:

- Complete all top priority functional requirements
- Complete dashboard
- Meet with my project supervisor.

This month I achieved the following:

Now that exams are coming to an end I will be able to spend all my time focusing on my project. The full booking system is now complete with entries going into the database and appearing in the calendar. From this I can pull useful statistics and add them to a dashboard. I am going to aim to make 5 charts and will use JS Highcharts to make them! I have also began working on my technical report, it is very handy that so much had to be completed for the mid-point presentation leaving less work for now. So far, I've spent time refining my requirements and also working on the layout to ensure it has a continuous flow. I am using the Software Development Lifecyle as a framework for my project development and technical report. So far I have completed 8/10 of my functional requirements giving me time to focus on my non-functional requirements and technical document.

Reflection and Intended Changes

Coming to the end of my project I do not have any reflection or intended changes. I am really just working the most I can to complete my project in time.

Other Material Used

- **Survey Monkey:** Used to create a survey for research. (SurveyMonkey, 2016)
- **PowerPoint:** Used to create presentation.
- **Word:** Used to create technical report.
- **Notepad++:** Used to edit web files such as, PHP, HTML, CSS.
- **Logo maker:** Used to create the logo for planmyevent.eu. (Logomakr, 2016)
- **Google images:** Used to select images for website styling. (Google, 2016)
- **Draw.io:** Used to create the use case diagrams, ERD and system architecture diagram. (Draw.io, 2016)
- **Balsamiq:** Used to create the Wireframes. (Balsamiq, 2016)