Final Project

Ger’s Garage

Balazs Barcza

Student Number : SB18002

Date : *15/06/2019*

0 INDEX

Table of Contents

0 INDEX....2

1 PROJECT INTRODUCTION:....4

1.1 PART 1: OVERVIEW....5

1.2 PART 2: PURPOSE....5

2 LITERATURE REVIEW:....6

2.1 PART 1: CLOUD DATABASE....6

2.2 PART 2: ANDROID STUDIO....6

2.3 PART 3: BOOTSTRAP....6

3 SYSTEM ANALYSIS AND DESIGN:....7

3.1 PART 1: FUNCTIONAL REQUIREMENTS....7

3.2 PART 2: USER INTERFACE DESIGN....7

3.3 PART 3: FUNCTIONAL & DATA DESIGN....8

4 IMPLEMENTATION OF THE SYSTEM:....9

4.1 PART 1: SOFTWARE ARCHITECTURE IMPLEMENTATION....9

4.2 PART 2: SYSTEM ANALYSIS AND DESIGN REFLECTIONS....9

4.3 PART 3: PROBLEMS CONFRONTED....9

5 TESTING AND EVALUATION:....11

5.1 PART 1: FUNCTIONAL REQUIREMENTS....11

5.2 PART 2: INSTRUCTIONS FOR USER....11

5.3 PART 3: COMMERCIALISATION / MARKETING....12

6 CONCLUSIONS:....13

6.1 PART 1: EVALUATE THE SUCCESS AND RESULTS OF THE PROJECT....13

6.2 PART 2: SUGGESTIONS FOR FURTHER WORK....13

7 CODE LISTINGS:....14

7.1 PART 1: #####.....14
7.2 PART 2: #####.....14
7.3 PART 3: #####.....14
8 APPENDIX B: (OTHER TECHNICAL OR DATA APPENDICES AS REQUIRED):.....15
8.1 PART 1: PROJECT PLANNING 15
8.2 PART 2: REFLECTIVE LEARNING JOURNAL 15
9 LIST OF REFERENCES:.....16

1 Project Introduction:

The first chapter sets the background and motivation for the project. The problem to be solved is stated, with the project aims and a list of specific objectives.
The chapter could include:
- A brief synopsis of the project context (supplied by CCT)
- General areas of computing that project context covers / requires knowledge of
- Brief summary of your initial proposed plan for addressing the project context
- Short section arguing 'why' this is a good project – outline Individual's skills, interests, strengths – they Individual can describe how the project brings together many of the modules they've listed
- Novel aspects – a real world business or organisation or taking advantage of new technology

---

1.1 Part 1: Overview

Ger is a mechanic who runs a small garage. He carries out maintenance checks for all kinds of small to medium vehicles (i.e. motorbikes, cars, small vans and small buses). He has a small number of staff who work with him.

In most cases, a maintenance check will require parts or other supplies (e.g. brake fluid; engine oil). He keeps a stock of common supplies at his garage and sells them to customers if/when needed. He needs an online service to allow customers to book their vehicles in for a check-up or service. He has asked you to build a WEB-BASED [OR MOBILE] application to fulfil his requirements.

Project overview:
He carries out maintenance checks for all kinds of small to medium vehicles (i.e. motorbikes, cars, small vans and small buses). He has a small number of staff who work with him.

In most cases, a maintenance check will require parts or other supplies (e.g. brake fluid; engine oil). He keeps a stock of common supplies at his garage and sells them to customers if/when needed.

He needs an online service to allow customers to book their vehicles in for a check-up or service. He has asked you to build a WEB-BASED [OR MOBILE] application to fulfil his requirements.

Ger's garage minimum requirements:

- A new website for Ger's Garage is created that advertises his garage and his services. [OPTION – YOU COULD CREATE THIS AS A MOBILE APPLICATION INSTEAD]
    1. websites
    2. moblie applications

    Booking services:

Customers can register on the website and book their vehicle in for a service OR a repair.
There must be AT LEAST 4 types of booking:

1. Annual Service
2. Major Service
3. Repair / Fault
4. Major Repair

Login Service:
Registered customers should be able to login on future occasions and the system should remember their details. This would include details of the vehicle they LAST booked in to the garage

Booking checklist:
Customers need to be able to select a DATE for their service. The website should limit the number of bookings allowed per day (you can decide the limit, but this should be realistic for a small garage). If there is no available space for a day, then the customer should not be able to book a service on that day.

Ger's garage is closed on Sundays for service bookings. DO not allow customers to select a Sunday.

Information about the vehicles:
Customers will need to be able to provide some basic information about themselves and their vehicles:

- Customer name & contact details (mob phone essential)
- Vehicle type & make
- Vehicle licence details
- Vehicle engine type
  - diesel
  - petrol
  - hybrid
  - electric
- Booking Required
  - Annual Service
  - Major Service
  - Repair / Fault
  - Major Repair
- Customer Comments (to allow customer add any notes they want to add, such as a description of the problem)

Ger administration:
Ger needs to have an admin access to the site to view bookings for any particular day OR week, so that he can plan his work & staff rostering. You can assume Ger has at least 4 mechanics available on any one particular day (you can change this, but be realistic – this is a small garage!)

He needs to be able to allocate a mechanic to each vehicle. You can assume that all of his staff are able to carry out any type of service / repair and that each mechanic could carry out AT MOST 4 services/repairs in one day. If the booking is a Major Repair then this would count double.

Printing services:
He needs to be able to print the schedule for any particular date.

He needs to be able to allocate costs to each booking. You can assume a basic fixed cost (e.g. an Annual Service might cost €200 minimum).

Items/Parts:
Ger needs to be able to add to this the cost of any item/parts that were needed to fix/service the vehicle. For example, if a tyre needed to be replaced then the cost of the tyre would be added and the cost of carrying out a "wheel balancing" would be added.

You can decide the cost of each item/part and what types of items/parts you include, but you should be realistic. You do not have to provide an exhaustive list of parts, but the more you provide the more realistic your final product will be. AT A MINIMUM you should provide 40 different parts/items for the garage.

Note that the customer has to provide details of the type and make of car (e.g. Car - Ford Feista). The site MUST allow for motorbikes, cars, small vans and small buses. You can decide how many different makes to provide as default, but you MUST provide at least 30 in total. Make sure to allow the user choose "other" if their vehicle is not in the list.

Booking statuses:
Ger needs to be able to set each booking to one of 5 possible statuses:

1. Booked – this is the default status when a booking is made
2. In Service – when the vehicle arrives at the garage
3. Fixed / Completed – when the vehicle is ready for collection
4. Collected – When the customer has taken the vehicle away and paid their bill
5. Unrepairable / Scrapped – when the fault cannot be fixed; in this case the car has either been taken away by the customer or has been sent for scrap.

Invoice:
Ger needs to be able to print an "invoice" or bill for each customer when the service/repair is complete. This should provide an itemised bill for the customer. For example:

CUSTOMER:
Joe Bloke
Mob No: 085 02140201
Vehicle: Peugeot 406
Licence: 12 G 123456

Annual Service €189

Mini Valet €39
Car mat €17
TOTAL DUE €245

Payment due on collection.

You can assume that the actual payment is handled by Ger at his checkout / on another system. You do not need to process payments or issue a receipt. To achieve a distinction grade you should aim to include additional functionality that has not been specified here but which would make sense within the project brief.

REPORT GUIDANCE:
For the design section of your report, you should aim to include the following
ESSENTIAL:

- Wireframe designs for the website. Include key pages – you may not need to include every single page.
- Functional Requirements – set out the various requirements in a structured manner and specify a level of priority (e.g. Must have, etc.)
- Data Requirements – set out the data you will need to store using a 'data dictionary' or similar structured layout. Make sure you specify data types, restrictions and any other important details
- Database tables – should be normalized to 3NF minimum

DIAGRAMS

- Include design diagrams to detail your design. Examples would include: a class diagram, and E-R Diagram, Use-cases.

JUSTIFICATION
Why did you choose this wireframe design? How did you determine your data requirements? How did you construct your class diagram and/or use-cases?

1.2 Part 2: Purpose

Why need Gerg a website?

Small medium or large size of the business has to have a website or mobile app because this is the key to making sales and contact with the customers. Consumer behavior changes over time to adapt to modern technology, and consumer behavior has changed to adapt to the digital age.
At the 21 century, the key is to get real-time and communication with the customer. The 1930s, advertising in the Yellow Pages. It was a standard operating procedure for most businesses.
Then came the world wide web (WWW) and a new disruption to the commercial status of the digital transformation. As more and more consumers realized they could find what they needed online faster and more effectively than a phone book, behavior moved away from using the printed directory. Nowadays at 2019, and the number of purchasers that go online to find a local business has jumped to 97 percent. If you want them to choose your company, you need to be found online—meaning you need a website.

"Websites work. No matter what your business or profession, a website can generate business, promote goodwill among customers and prospects, and deliver strong marketing messages - whether your business is small, large or in-between, well-established or brand-new.

> People use the Web in greater and greater numbers, more and more every day. Even if you are a completely local small business, service, contractor or consultant, odds are people have used search engines to look for your web site - and if you don't have a web site... well, you get the picture.

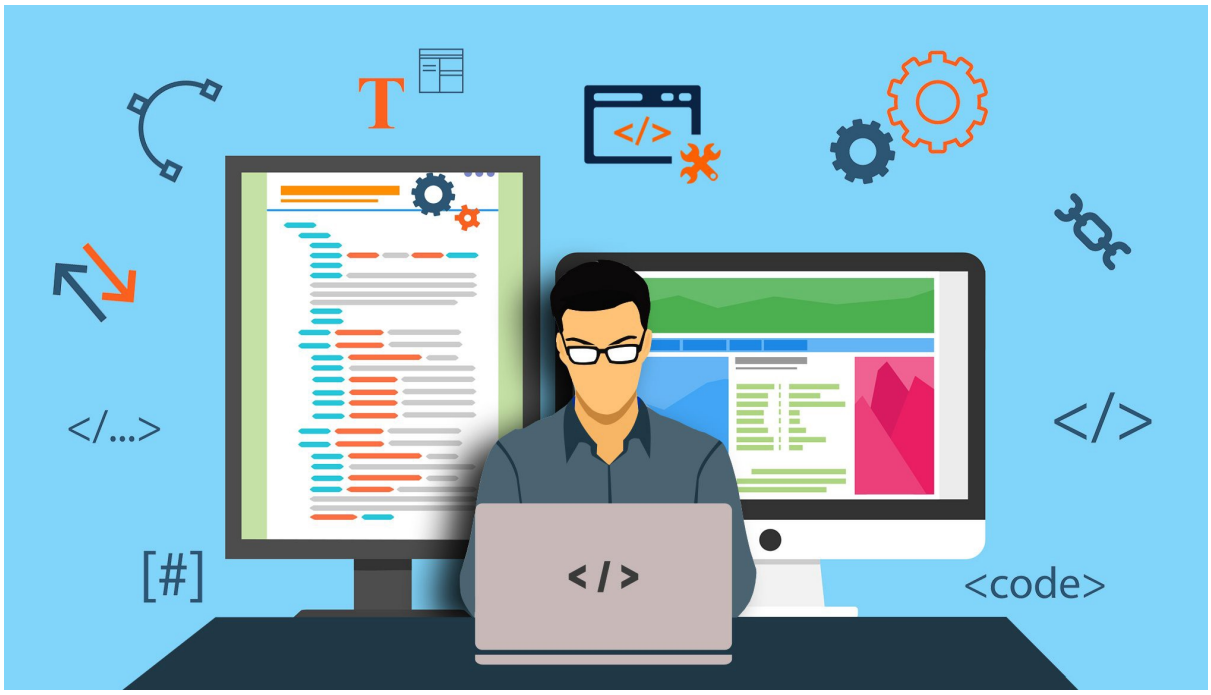
" http://www.networksolutions.com/education/business-needs-website/

The is a couple of the reason why small business needs a Website

- The customers expect it.
  The customer has to trust a business
- It provides social proof.
  The potential customers are already looking for you online, including customer recommendations on your site is a great way to impress potential clients.
- Influence the clients
  You can control the communication channel
- You can control the communication channel
- The competitors all have company websites.
- You will be visible on the Google search results
- Display your products or outline your services in detail

Web Developer:

Web developers are at the forefront of the Internet age. The websites we browse, the gifts we order and the news we read online are all made possible by the web developers who design, build and implement Internet websites. Web developers are responsible for designing and develop a website and website application. The developer can use different languages (HTML, PHP, JavaScript, JQuery, etc), and can develop with different technology, and manager site functionality, implement application features and manage security.



https://medium.com/level-up-web/developer-roadmaps-all-in-one-place-75c0402db0e0

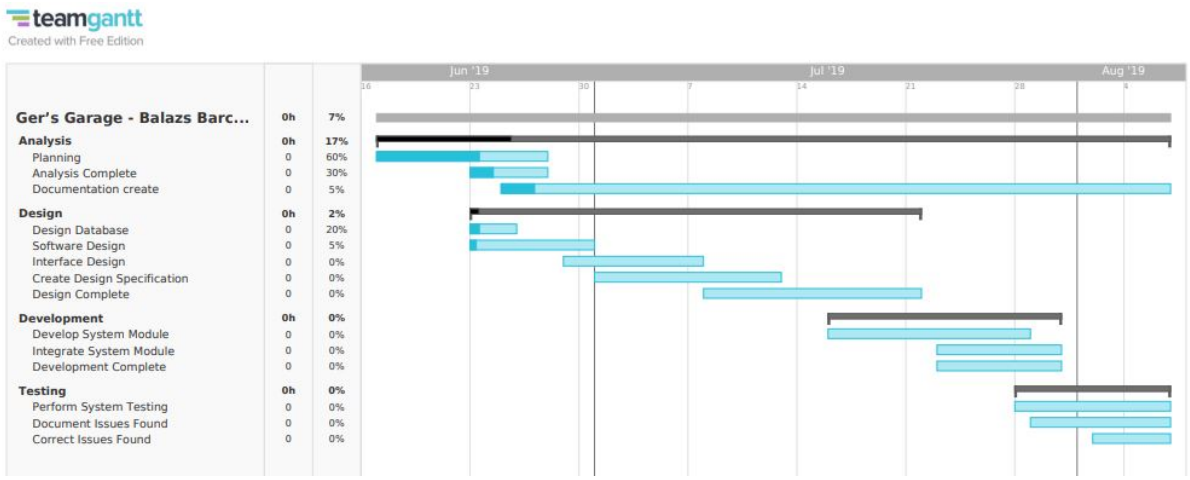
The 9 Web Technologies Every Web Developer Must Know in 2019

1. Browsers
2. HTML & CSS
3. Web Development Frameworks
4. Programming Languages
5. Protocols
6. API

7. Data formats
8. Client (or Client-side)
9. Server (or Server-side)

More information https://tms-outsource.com/blog/posts/web-technologies/

My origin project plan.



I have chosen this project because I wanted to build an app (website) and I wanted to use the Google cloud platform. I think this is the future more and more project will move to the cloud. It is a game changer because It is more playable and advances.

1. Less Costs

The services are free from capital expenditure. There are no huge costs of hardware in cloud computing. You just have to pay as you operate it and enjoy the model based on your subscription plan.

2. 24 X 7 Availability

Most of the cloud providers are truly reliable in offering their services, with most of them maintaining an uptime of 99.9%. The workers can get onto the applications needed basically from anywhere. Some of the applications even function off-line.

3. Flexibility in Capacity

It offers flexible facility which could be turned off, up or down as per the circumstances of the user. For instance, a promotion of sales is very popular, capacity can be immediately and quickly added to it for the avoidance of losing sales and crashing servers. When those sales are done, the capacity can also be shrunk for the reduction of costs.

4. All over Functioning

Cloud computing offers yet another advantage of working from anywhere across the globe, as long as you have an internet connection. Even while using the critical cloud services that offer mobile apps, there is no limitation of the device used.

5. Automated Updates on Software

In cloud computing, the server suppliers regularly update your software including the updates on security, so that you do not need to agonize on wasting your crucial time on maintaining the system.

You find extra time to focus on the important things like 'How to grow your businesses.

6. Security
Cloud computing offers great security when any sensitive data has been lost. As the data is stored in the system, it can be easily accessed even if something happens to your computer. You can even remotely wipe out data from the lost machines for avoiding it getting in the wrong hands.

7. Carbon Footprint
Cloud computing is helping out organizations to reduce their carbon footprint. Organizations utilize only the amount of resources they need, which helps them to avoid any over-provisioning. Hence, no waste of resources and thus energy.

8. Enhanced Collaboration
Cloud applications enhance collaboration by authorizing diverse groups of people virtually meet and exchange information with the help of shared storage. Such capability helps in improving the customer service and product development and also reducing the marketing time.

9. Control on the Documents
Before cloud came into being, workers needed to send files in and out as the email attachments for being worked on by a single user at one time ultimately ending up with a mess of contrary titles, formats, and file content. Moving to cloud computing has facilitated central file storage.

10. Easily Manageable
Cloud computing offers simplified and enhanced IT maintenance and management capacities by agreements backed by SLA, central resource administration and managed infrastructure. You get to enjoy a basic user interface without any requirement for installation. Plus you are assured guaranteed and timely management, maintenance, and delivery of the IT services.

2 Literature Review:

The aim of this chapter is to present all academic research carried out throughout the project cycle. It is important that learners produce research that defends their justifications for choosing one from of technology or software over another, and other sources of information that have helped inform the individuals thinking, planning and delivery of the project.

2.1 Part 1: Cloud Database

1. A cloud database is a database that typically runs on a cloud computing platform, and access to the database is provided as-a-service. Database services take care of scalability and high availability of the database. Database services make the underlying software-stack transparent to the user.

2.2 Part 2: Android Studio

11. Android Studio is the official integrated development environment for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development. It is available for download on Windows, macOS and Linux based operating systems.

2.3 Part 3: Bootstrap

17. Build responsive, mobile-first projects on the web with the world's most popular front-end component library.

Bootstrap is an open source toolkit for developing with HTML, CSS, and JS. Quickly prototype your ideas or build your entire app with our Sass variables and mixins, responsive grid system, extensive prebuilt components, and powerful plugins built on jQuery.

3 System Analysis and Design:

The overall aim of this chapter is to answer the questions – exactly what is the application supposed to do? It can include the following, where relevant:
- Functional Requirements
o Detailed description of the functionality of the proposed system. This should be comprehensive and exact, break up the application into subsystems.
o Diagrams – use Case diagrams, Wire frames, with text descriptions
- Data Requirements
o An overview of the entities and data in the system, and what data needs to be stored
o Diagrams – an Entity-Relationship Diagram
- User Interface Design
o This should contain an argument as to how this suggested interface supports each of the use cases specified in the analysis
o Diagrams – Screen designs, either pen-and-paper or computer drawn of how the user interface will appear
- Functional Design
o Functional design should model both the structure of each software component in the systems, and also how they interact with each other.
o Diagrams – detailed class diagram and an Interaction Diagram to show the interaction between objects in the system
- Data Design
o Whether to be implemented as a database or some other central data repository, a detailed design of the data storage components should be presented
o Diagrams – Normalised database tables

3.1 Part 1: Functional Requirements

1. Using virtualization software, **install** 2 virtual machines.
Use *Server 2012 R2 (GUI).) or Server 2008R2 (GUI)* (Both Server OS not Client OS)

One VM will act as the Server and the other VM will act as the Client. [Client will also be Web Server]

3.2 Part 2: User Interface Design

11. This should contain an argument as to how this suggested interface supports each of the use cases specified in the analysis
o Diagrams – Screen designs, either pen-and-paper or computer drawn of how the user interface will appear

3.3 Part 3: Functional & Data Design

17. Functional design should model both the structure of each software component in the systems, and also how they interact with each other.

o Diagrams – detailed class diagram and an Interaction Diagram to show the interaction between objects in the system

Whether to be implemented as a database or some other central data repository, a detailed design of the data storage components should be presented

o Diagrams – Normalised database tables

4 Implementation of the system:

This chapter should detail how the learner implemented a working system based on their design. This should include the technologies used (languages, APIs, frameworks etc.) and how the system was implemented, based on the user and functional requirements identified during the analysis and design phase. This chapter should address any potential problems that could arise in the system and suggested or implemented solutions.
Possible areas for discussion in this chapter are:
o Architecture considerations - e.g. are there specific functional requirements that will influence the software architecture implementation.
o Technologies used - operating systems, databases, computer languages, frameworks, API's etc.
o Implementation of the system - main body of work for the chapter. This will discuss precisely how the system was developed, based on the analysis and design considerations.
o Problems encountered - any issues that may have arisen during the implementation phase, e.g. the project's cross-platform compatibility between different operating systems.

4.1 Part 1: Software Architecture Implementation

Setting up Google Cloud Platform
My website will be on the cloud so I have chosen GCP (Google Cloud Platform)

Google Cloud Platform, offered by Google, is a suite of cloud computing services that runs on the same infrastructure that Google uses internally for its end-user products, such as Google Search and YouTube.
Alongside a set of management tools, it provides a series of modular cloud services including computing, data storage, data analytics and machine learning. Google Cloud Platform provides infrastructure as a service, platform as a service, and serverless computing environments.

Why we should transform a business with Google Cloud.
Modernize workloads on Google's global, secure, and reliable infrastructure.

Develop and run applications using open source and other software without operations staff.

Get insights from data with a full suite of analytics and ML tools.

- **Reduce risk with world-class security**

Your most challenging security scenarios are protected by the same secure-by-design infrastructure, global network, and built-in safeguards that Google uses.

- **Flexible hybrid and multi-cloud options**

Our managed, cloud-native solution means you can write an application once, then run it on-premises, on GCP, or on other clouds with no change in infrastructure

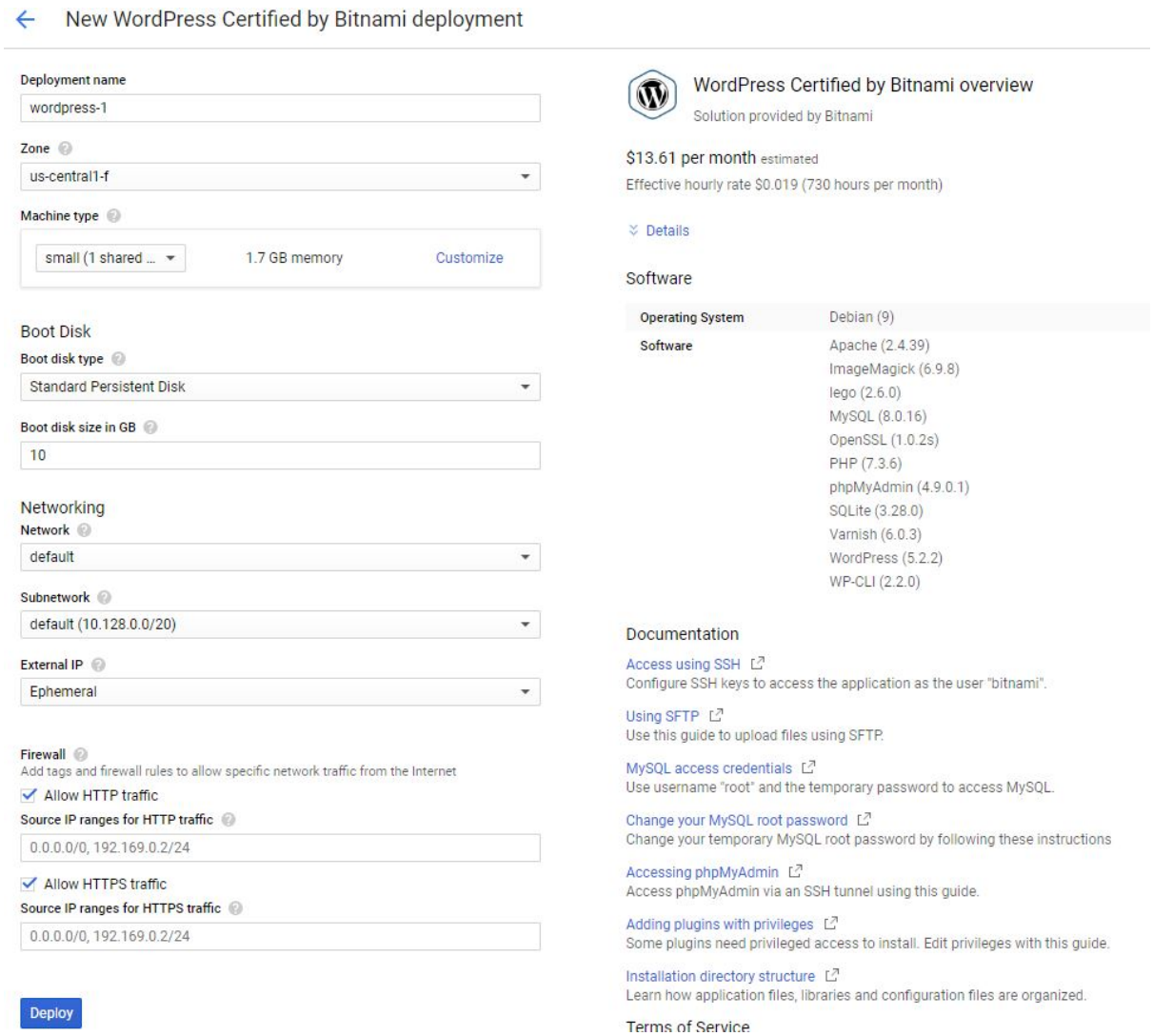
- **Power innovations with AI and ML**

Our easy-to-use artificial intelligence and machine learning capabilities are embedded in our core solutions, making them accessible and easily deployed.

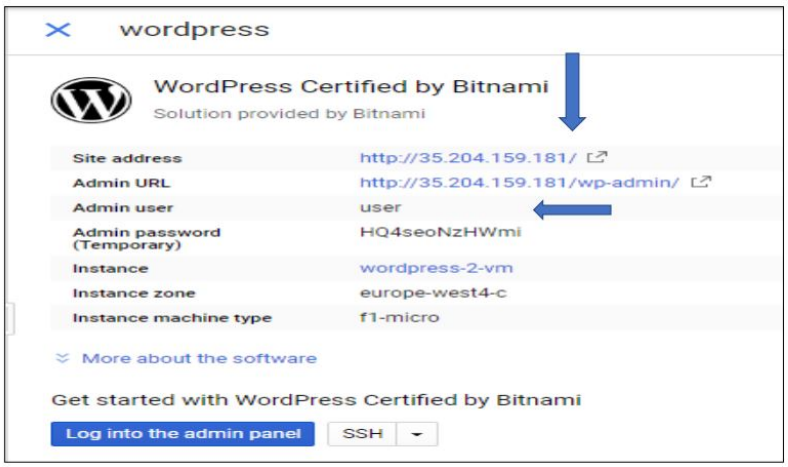
Step by step Setting up a WordPress site on Google Cloud.

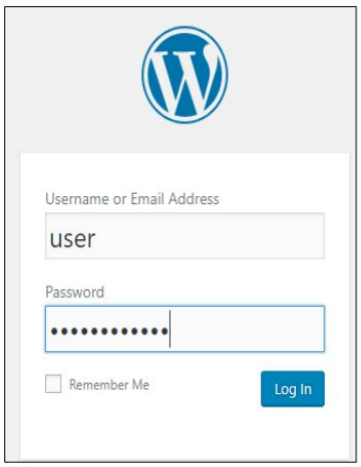
Step 1:
I have created a project
Step 2:
Marketplace I have created a virtula machine

Step 3:
I have to give my WordPress site a name and set up a (0.6GB) micro instance in the Europe West Zone .
Small instance is cheaper.



Step 4:
Once to site is complete I can log-in using the username and password given. Note This is the new site has an external IP address





Create easy website.

4.2 Part 2: System Analysis and Design Reflections

11.This will discuss precisely how the system was developed, based on the analysis and design considerations.

4.3 Part 3: Problems confronted

17.Problems encountered - any issues that may have arisen during the implementation phase, e.g. the project's cross-platform compatibility between different operating systems.

5 Testing and Evaluation:

Details of the learner's test plans, test results, user evaluations and discussion of these results in detail and in summary.
Possible entries in this chapter might include:
- Functional correctness
o Set of tasks system should be able to perform – part of requirements specification of system and include a focus on efficiency
o Set of inputs and correct outputs
o Set of 'test scripts'
- Objective of test / statement of which part of systems is being tested
- Input data/situation
- Correct output data / state / behaviour
- Need to show actual results of test – screen shots
- Evaluation - if actual matches correct then working
- Usability
o List of usability requirements
- set of tasks user should be able to perform
- Have a set of tasks for each type of user
- System Response times
- Time for user to complete a task
- Aesthetic
- Acceptable navigation of site and layout
o Set of 'test scripts'
- Instructions for user
- Observation / measure time / evaluate success of task
- Analyse results to come up with usability result
o Can also measure qualitative usability aspects with questionnaires / structured interviews etc.
- Commercialisation / marketing
o Requirements – registration on web search engines, direct marketing – discuss real commercialisation aspects of project
o Evaluation – have set of key words / phrases for targeted websites

---

5.1 Part 1: Functional Requirements

1. *Set of tasks system should be able to perform – part of requirements specification of system and include a focus on efficiency*
o Set of inputs and correct outputs
o Set of 'test scripts'
- Objective of test / statement of which part of systems is being tested
- Input data/situation
- Correct output data / state / behaviour
- Need to show actual results of test – screen shots
- Evaluation - if actual matches correct then working
- Usability

5.2 Part 2: Instructions for user

11. **CompuTech has two departments. Using Active Directory Users and Computers**

(ADUC), create 2 departments (Organizational Units) called Accounting-Dublin and Sales-Dublin. Inside the **Accounting-Dublin** OU and in the **Sales-Dublin** OU create user accounts, and network groups as specified below:
12. Inside the **Accounting-Dublin** OU create a Global Security group called **Accounting** and then create the 5 **Accounting** users accounts shown below, using the names supplied.

5.3 Part 3: Commercialisation / Marketing

17. Requirements – registration on web search engines, direct marketing –discuss real commercialisation aspects of project Evaluation – have set of key words / phrases for targeted websites

6 Conclusions:

The Individual needs to review the entire project against their problem context, aims and objectives, and evaluate project success and results. This may also include a section for suggestions for further work.
Appendix A: Code Listings
This should be a link to a cloud resource (such as GitHub) where the project code is maintained. Students should have only included selected code fragments or algorithm summaries in the main chapters, otherwise the project report can become a monotonous technical manual rather than a story of what they did and why they did it.

6.1 Part 1: Evaluate the success and results of the project

1. Using virtualization software, **install** 2 virtual machines.
Use *Server 2012 R2 (GUI).) or Server 2008R2 (GUI)* (Both Server OS not Client OS)

One VM will act as the Server and the other VM will act as the Client. [Client will also be Web Server]

6.2 Part 2: Suggestions for further work

11. **CompuTech has two departments. Using Active Directory Users and Computers**

(ADUC), create 2 departments (Organizational Units) called Accounting-Dublin and Sales-Dublin. Inside the **Accounting-Dublin** OU and in the **Sales-Dublin** OU create user accounts, and network groups as specified below:
12. Inside the **Accounting-Dublin** OU create a Global Security group called **Accounting** and then create the 5 **Accounting** users accounts shown below, using the names supplied.

---

7 Code Listings:

This should be a link to a cloud resource (such as GitHub) where the project code is maintained. Students should have only included selected code fragments or algorithm summaries in the main chapters, otherwise the project report can become a monotonous technical manual rather than a story of what they did and why they did it.

7.1 Part 1:

7.2 Part 2:

11. **CompuTech has two departments. Using Active Directory Users and Computers**

7.3 Part 3:

8 Appendix B: (other technical or data appendices as required):

If you have additional technical data to showcase it should be included in this appendix, you can also use this appendix to present the raw data of empirical research carried out (questionnaires, interviews etc.)

8.1 Part 1: : Project Planning

1. U

8.2 Part 2: Reflective Learning Journal

11. **C**

---

9 *List of References:*

All citations used within the report should include their full reference using the Harvard referencing style. A reference list should be included in this section of the report.

---

17. U