



---

## **Final Project**

Ger's Garage

Balazs Barcza

Student Number : SB18002

---

---

*Date :*

*15/06/2019*



# 0 INDEX

---

## Table of Contents

<b>0 INDEX.....</b>	<b>2</b>
<b>1 PROJECT INTRODUCTION:.....</b>	<b>4</b>
1.1 PART 1: OVERVIEW.....	5
1.2 PART 2: PURPOSE .....	5
<b>2 LITERATURE REVIEW:.....</b>	<b>6</b>
2.1 PART 1: CLOUD COMPUTING SERVICES.....	6
2.2 PART 2: GOOGLE CLOUD PLATFORM.....	6
2.3 PART 3: GCP SQL DATABASE.....	6
<b>3 SYSTEM ANALYSIS AND DESIGN:.....</b>	<b>7</b>
3.1 PART 1: FUNCTIONAL REQUIREMENTS.....	7
3.2 PART 2: USER INTERFACE DESIGN.....	7
3.3 PART 3: FUNCTIONAL & DATA DESIGN.....	8
<b>4 IMPLEMENTATION OF THE SYSTEM:.....</b>	<b>9</b>
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
<b>5 TESTING AND EVALUATION:.....</b>	<b>11</b>
5.1 PART 1: FUNCTIONAL REQUIREMENTS.....	11
5.2 PART 2: INSTRUCTIONS FOR USER.....	11
5.3 PART 3: COMMERCIALISATION / MARKETING.....	12
<b>6 CONCLUSIONS:.....</b>	<b>13</b>
6.1 PART 1: EVALUATE THE SUCCESS AND RESULTS OF THE PROJECT.....	13
6.2 PART 2: SUGGESTIONS FOR FURTHER WORK.....	13
<b>7 CODE LISTINGS:.....</b>	<b>14</b>

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:**

A Web developer is a programmer whose development of applications to the World Wide Web ( WWW ). Web developers design website and website application. Web developers are responsible for designing and develop a website and website application. The developer can use many languages (HTML, PHP, JavaScript, JQuery, etc), and can develop with different technology. He or she has to manage the site, 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.

<https://www.idexcel.com/blog/top-10-advantages-of-cloud-computing/>

## 2 Literature Review:

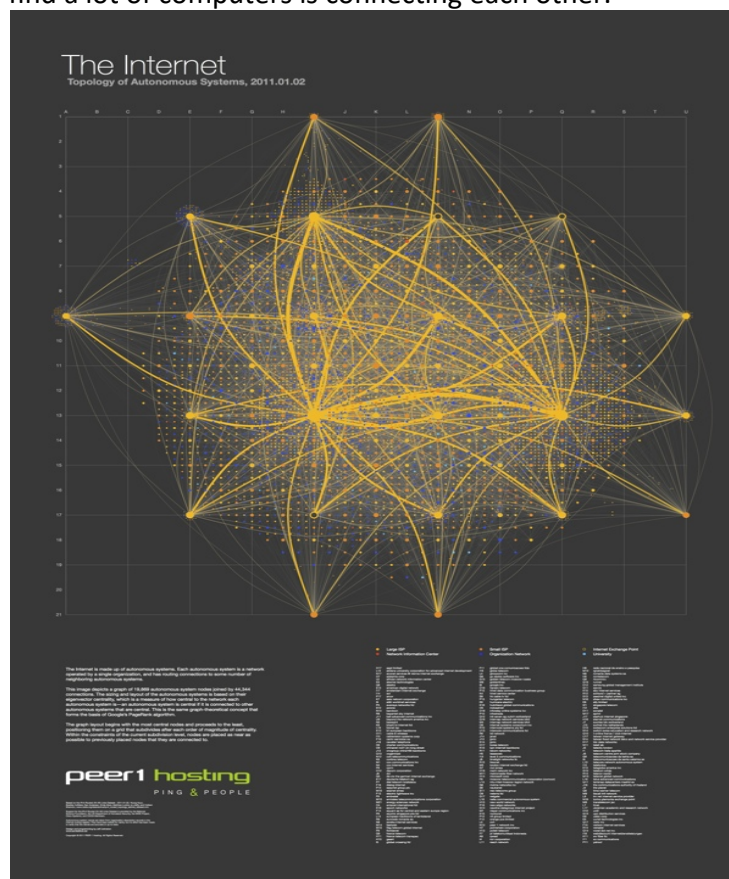
*The aim of this chapter is to present all academic research carried out throughout the project cycle. It is important that learners produce research that defends their justifications for choosing one from of technology or software over another, and other sources of information that have helped inform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

---

### 2.1 Part 1: Cloud Computing Services

What is the cloud computing?

Before we go deeper to cloud computing We have to define the cloud. This is the technology not older than 10 years and nowadays is happening the change. Why the people using the cloud that a logo, Because It is easy to draw it. But If we looking at the big picture we will see cloud computing is similar that the real cloud. In the cloud, We can find a lot of h2o and cloud computing we will find a lot of computers is connecting each other.



The Map of the Internet — Cool Infographics  
<https://coolinfographics.com/blog/2011/4/7/the-map-of-the-internet.html>

*“It’s a layout of all the networks that are interconnected to form the internet. Some are run by small and large ISPs, university networks, and customer networks - such as Facebook and Google. It’s visual representation of all those networks interconnecting with one another, forming the internet as we know it. Based on the size of the nodes and the thickness of the lines, it speaks to the size of those particular providers and the connections.”*

**Randy Krum** Designer | Author | Instructor

<https://coolinfographics.com/blog/2011/4/7/the-map-of-the-internet.html>

Those small computers are working together that a big resource. Those computers are sharing the resource with each other.

Cloud computing is the on-demand availability of computer system resources, especially data storage and computing power, without direct active management by the user. The term is regularly used to describe datacenters available to many clients over the Internet. Big clouds, dominant today, often have roles spread over many locations from central servers. If the connection to the user is relatively close, it may be designated an edge server.

Fundamentally, cloud computing is the delivery of computing services including servers, storage, databases, networking, software, analytics, and intelligence over the Internet (“the cloud”) to offer faster innovation, flexible resources, and economies of scale. You typically pay only for cloud services you use, helping you lower your operating costs, run your infrastructure more efficiently, and scale as your business needs change.

**Cloud computing services have several standard properties:**

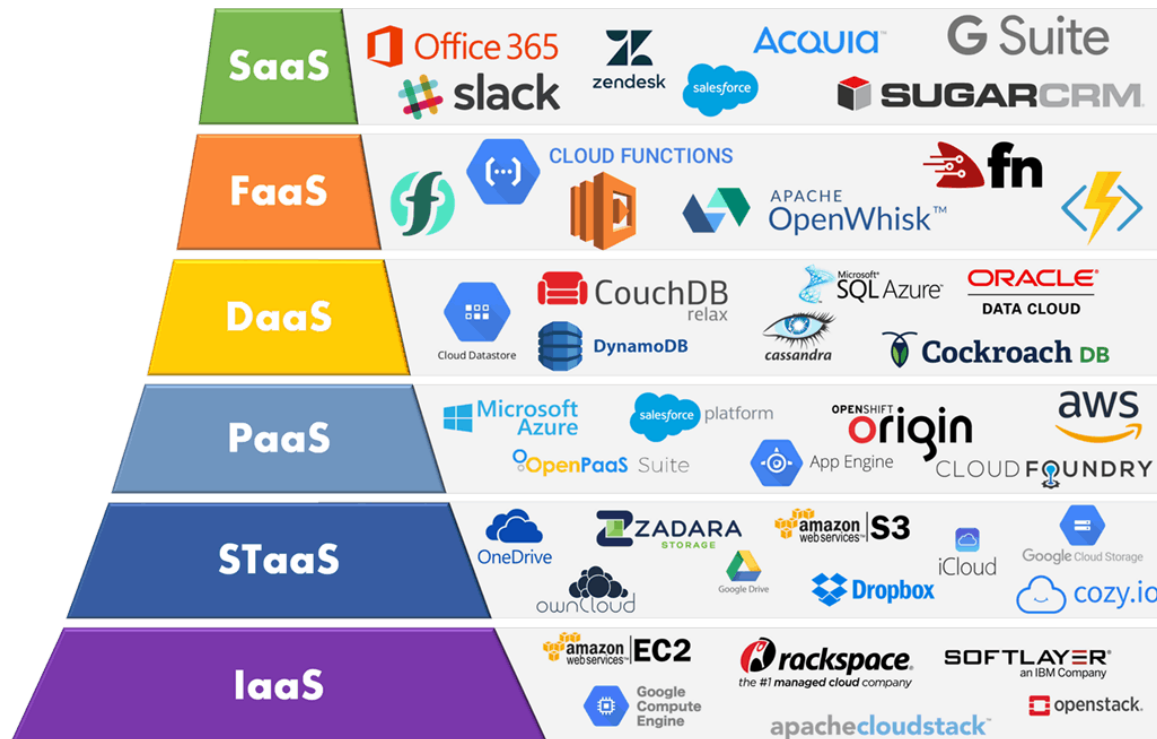
- Virtualization- cloud computing utilizes server and storage virtualization extensively to allocate/reallocate resources rapidly
- Multi-tenancy-resources are pooled and shared among multiple users to gain economies of scale
- Network-access- resources are accessed via web-browser or thin client using a variety of networked devices (computer, tablet, smartphone)
- On demand- resources are self-provisioned from an online catalogue of pre-defined configurations
- Elastic-resources can scale up or down, automatically
- Metering/chargeback-resource usage is tracked and billed based on service arrangement

**Cloud computing services delivered internally or by third-party service:**

- Software as a Service (SaaS) – software runs on computers owned and managed by the SaaS provider, versus installed and managed on user computers. The software is accessed over the public Internet and generally offered on a monthly or yearly subscription.
- Infrastructure as a Service (IaaS)– compute, storage, networking, and other elements (security, tools) are provided by the IaaS provider via public Internet, VPN, or dedicated network connection. Users own and manage operating systems, applications, and information running on the infrastructure and pay by usage.

- Platform as a Service (PaaS)– All software and hardware required to build and operate cloud-based applications are provided by the PaaS provider via public Internet, VPN, or dedicated network connection. Users pay by use of the platform and control how applications are utilized throughout their lifecycle.

#### Cloud services delivery models



<https://imelgrat.me/cloud/cloud-services-models-help-business/>

#### Models of cloud computing:

Not all clouds are the same, different users, different type of cloud computing system right for them. Different models, types, and services offer the right solution. There are three different ways to deploy cloud services: on a public cloud, private cloud, or hybrid cloud.



<https://www.allcovered.com/the-learning-center/cloud-revolution-or-evolution-582>

- **Public cloud**  
Public clouds are owned and operated by a third-party cloud service providers, Public cloud provides computing resources, like servers and storage, over the Internet. The cloud provider owned and managed all hardware, software and other infrastructure.
- **Private cloud**  
A private cloud belongs to cloud computing resources used exclusively by a single business or organization. It can be located on the company’s data center, or companies can pay third-party service providers to host private cloud.
- **Hybrid cloud**  
Hybrid clouds join public and private clouds. The technology permits applications and data to be shared between them. The data and applications can transfer between private and public clouds, a hybrid cloud gives to the business adaptability, more deployment options, and helps optimize your current infrastructure, security, and compliance.

**Cloud computing services offer numerous benefits:**

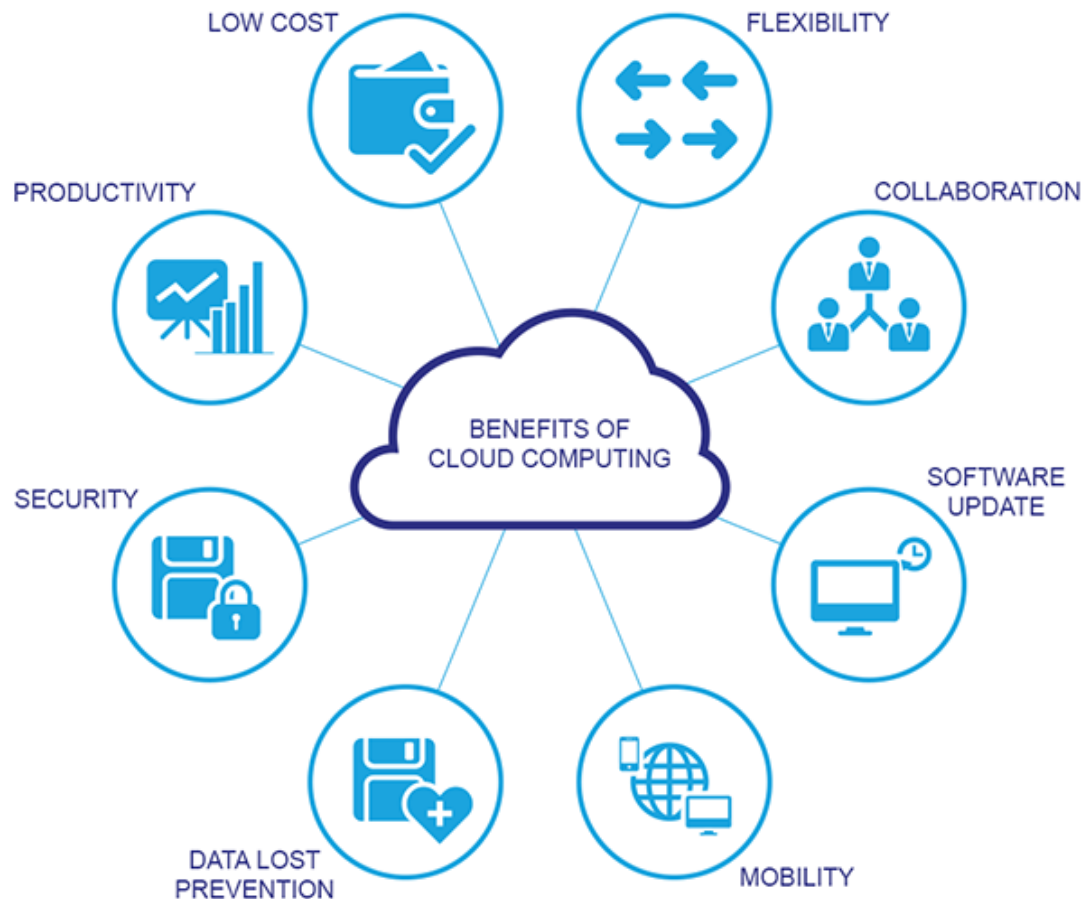
- Faster implementation and time to value
- Anywhere access to applications and content
- Rapid scalability to meet demand
- Higher utilization of infrastructure investments
- Lower infrastructure, energy, and facility costs
- Greater IT staff productivity and across organization
- Enhanced security and protection of information assets

**Cloud computing in the business view:**

- **Cost**  
If you use Cloud computing You do not need to buying hardware and software and setting up and running on-site datacenters—the racks of servers, the round-the-clock electricity for power and cooling.
- **Productivity**  
Datacenters typically require a lot of “racking and stacking”—hardware setup, software patching, and other time-consuming IT management chores. Cloud computing eliminates the need for many of these jobs, so IT teams can spend time on achieving more important business goals.
- **Security**  
The cloud providers offer methods, technologies, and controls and security on your products. It is helping protect your data, apps, and infrastructure.
- **Performance**  
The most significant cloud computing services run on a worldwide network of secure datacenters, which are usually improved to the latest generation of fast and efficient computing hardware. This offers several benefits over an individual corporate datacenter, including reduced network latency for applications and greater economies of scale.



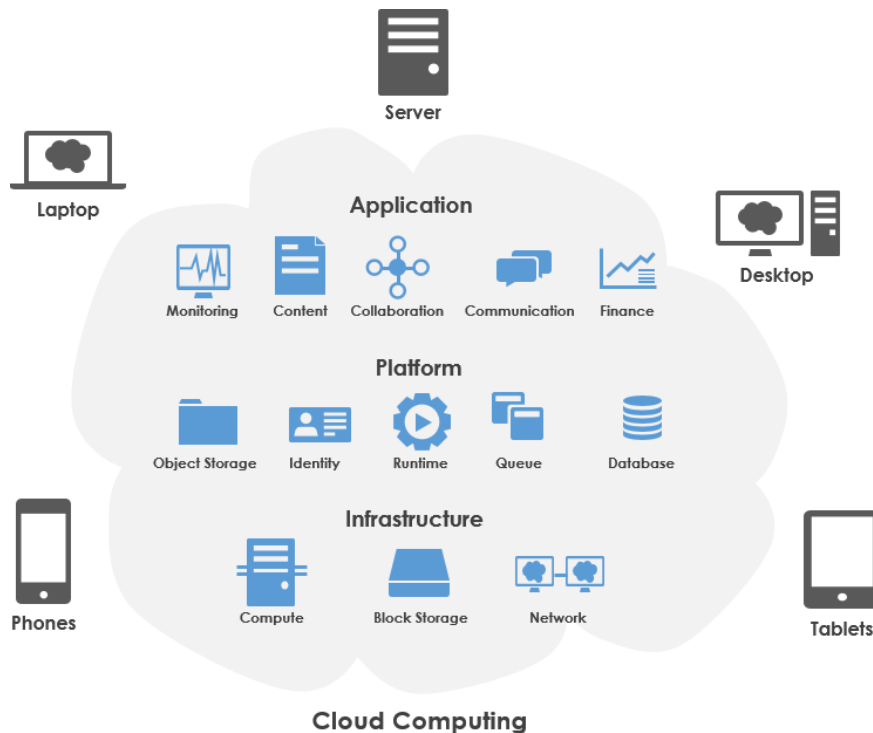
- Global scale  
The cloud computing service can be scale, So the users can choose the right size of IT resources. Can make computer power stronger, storage size, change location.
- Speed  
The system more flexibility, few minutes and the user can create service.
- Reliability  
Cloud computing makes data backup, recovery, faster and save because data can be copied at multiple locations.



<http://www.visiontechme.com/cloud-computing.php>

You can use an online service to send an email, edit documents, watch movies or TV, listen to music, play games, or store pictures and other files, it's likely that cloud computing is making it all possible.





<https://www.visual-paradigm.com/guide/cloud-services-architecture/what-is-aws-architecture/>

#### Create new apps and services

Quickly build, deploy, and scale applications—web, mobile, and API—on any platform. Access the resources you need to help meet performance, security, and compliance requirements.

#### Test and build applications

Reduce application development cost and time by using cloud infrastructures that can easily be scaled up or down.

#### Store, back up, and recover data

Protect your data more cost-efficiently—and at massive scale—by transferring your data over the Internet to an offsite cloud storage system that's accessible from any location and any device.

#### Analyze data

Unify your data across teams, divisions, and locations in the cloud. Then use cloud services, such as machine learning and artificial intelligence, to uncover insights for more informed decisions.

#### Stream audio and video

Connect with your audience anywhere, anytime, on any device with high-definition video and audio with global distribution.

#### Embed intelligence

Use intelligent models to help engage customers and provide valuable insights from the data

captured.

### Deliver software on demand

Also known as software as a service (SaaS), on-demand software lets you offer the latest software versions and updates around to customers—anytime they need, anywhere they are.

---

## 2.2 Part 2: Google Cloud Platform

### 1. Cloud Service Providers

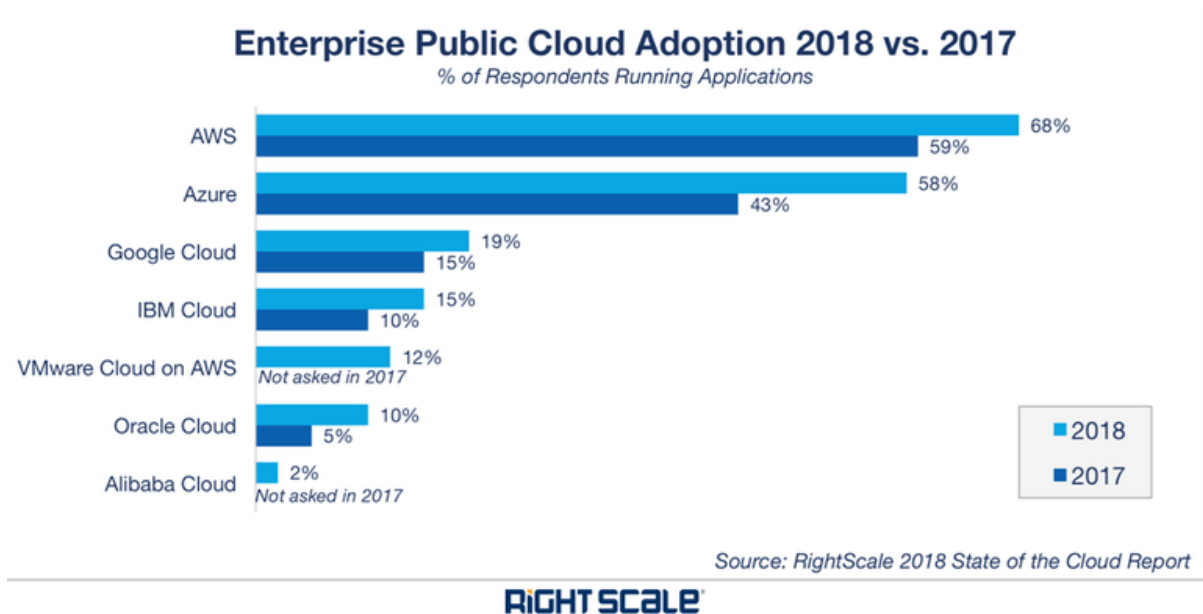
Cloud Computing Services are Information Technology (IT) as a service over the Internet. Cloud computing is a term which is used for storing and accessing data over the internet. It doesn't store any data on the hard disk of your PC. Cloud computing helps you to access your data from a remote server.

Cloud computing services range from full applications and development platforms to servers, storage, and virtual desktops. There's are various types of cloud computing services are available in the market.



<https://www.edureka.co/blog/what-is-azure/>

Market share 2018



<https://www.zdnet.com/article/top-cloud-providers-2018-how-aws-microsoft-google-ibm-oracle-alibaba-stack-up/>

## 2. What is the GCP and Why should We choose it.

GCP short name of the Google Cloud Platform.

Google Cloud Platform provides infrastructure as a service, platform as a service, and serverless computing environments. Google, is a cloud computing service provider that runs on the same infrastructure that Google uses internally for its end-user products, such as Google Search and YouTube, Gsuite.

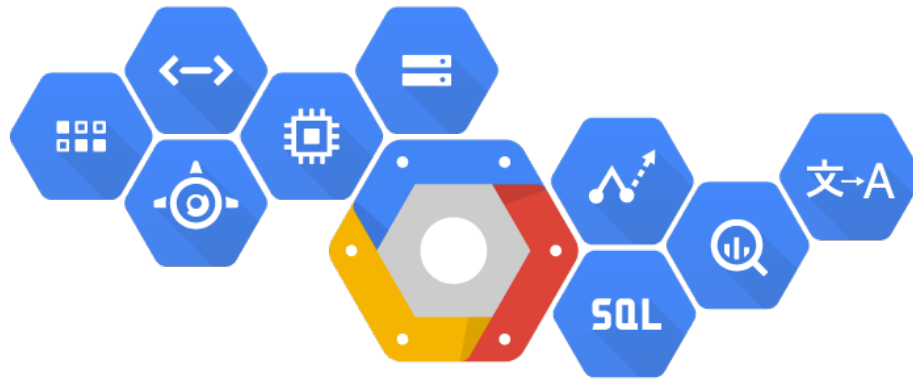
This is developing dynamically and improving and the 3rd bigger provider.

Google offers free 300 dollars free credit, to the user. The clients can use this credit to test the system before a charge.

Google has been in the cloud computing race for much less time than the incumbent leader AWS. Amazon Web Services rakes in about \$6 billion per quarter — still way ahead of Google Cloud, but from market trends, Google Cloud seems to be the fastest growing cloud platform today.

Google Cloud is differentiated and place GCP ahead of the other service providers in the industry.

- Billing per second  
Google Cloud charges customers for usage of its compute engine instances in one-second increments having a one-minute minimum.
- Big Data  
GCP has next-gen frameworks for data warehousing, advanced machine learning, and visual analytics give it an edge in data processing and analytics.
- User-friendly interface  
It's CloudShell, great tutorials, clear communication, and well-integrated services.
- Kubernetes leadership  
It is the creator of Kubernetes, the container management platform.



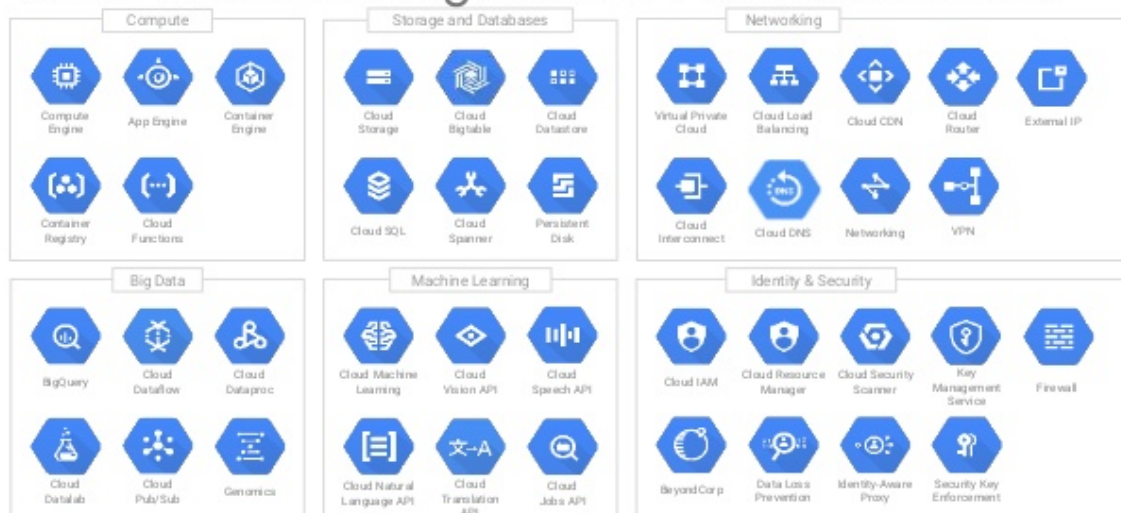
# Google Cloud Platform

<https://medium.com/google-cloud/deploying-websockets-cluster-to-gcp-with-lets-encrypt-certificates-5ebb7fc1e245>

Few companies that are using Google Cloud Platform (Compute Engine or App Engine) include Spotify, HSBC, Home Depot, Snapchat, HTC, Best Buy, Philips, Coca Cola, Domino's, Feedly, ShareThis, Sony Music, Ubisoft, and Apple.

## 3. Google Cloud Platform Products

### More than 60 Google Cloud Platform services



[https://www.slideshare.net/Hadoop\\_Summit/running-apache-hadoop-on-the-google-cloud-platform](https://www.slideshare.net/Hadoop_Summit/running-apache-hadoop-on-the-google-cloud-platform)

Google Cloud Platform products span the following categories:

- API management: API Analytics, API Monetization, Cloud Endpoints, Developer Portal, Cloud Healthcare API
- Artificial intelligence & machine learning: AI Hub (beta), Cloud AutoML (beta), Cloud TPU, Cloud Machine Learning Engine, Diagnostics Enterprise Edition, Cloud Natural Language, Cloud Speech-to-Text, Cloud Text-to-Speech, Cloud Translation, Cloud Vision, Cloud Video Intelligence, Cloud Inference API (alpha), and more
- Compute: Compute Engine, Shielded VMs, Container security, App Engine, Cloud Functions, GPU, and more
- Data analytics: BigQuery, Cloud Dataflow, Cloud Dataproc, Cloud Datalab, Cloud Dataprep, Cloud Composer, and more
- Databases: Cloud SQL, Cloud Bigtable, Cloud Spanner, Cloud Datastore, Cloud Memorystore
- Developer tools: Cloud SDK, Container Registry, Cloud Build, Cloud Source Repositories, Cloud Tasks, and more, as well as Cloud Tools for IntelliJ, PowerShell, Visual Studio, and Eclipse
- Internet of Things (IoT): Cloud IoT Core, Edge TPU (beta)
- Hybrid and multi-cloud: Google Kubernetes Engine, GKE On-Prem, Istio on GKE (beta), Anthos Config Management, Serverless, Stackdriver, and more
- Management Tools: Stackdriver, Monitoring, Trace, Logging, Debugger, Cloud Console, and more
- Media: Anvato, Zync Render
- Migration: Cloud Data Transfer, Transfer Appliance, BigQuery Data Transfer Service, Velostrata, VM Migration, and more
- Networking: Virtual Private Cloud (VPC), Cloud Load Balancing, Cloud Armor, Cloud CDN, Cloud NAT, Cloud Interconnect, Cloud VPN, Cloud DNS, Network Service Tiers, Network Telemetry
- Security: Access Transparency, Cloud Identity, Cloud Data Loss Prevention, Cloud Key Management Service, Cloud Security Scanner, and more
- Storage: Cloud Storage, Persistent Disk, Cloud Filestore, and more

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.

---

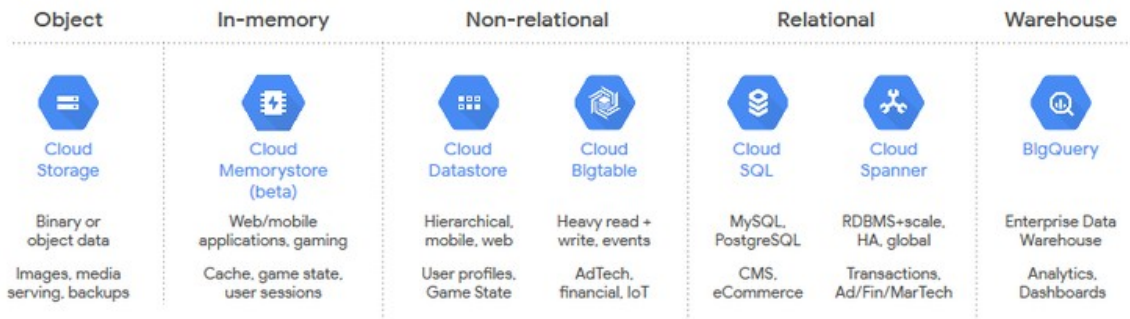
## **2.3 Part 3: GCP SQL Database**

Google Cloud database services are fully managed, scalable database services to support all your applications. GCP is rounding out its stable of managed database services as it on boards more large enterprises.

Managed database services are increasingly popular as enterprises aim to abstract the

underlying infrastructure and connect with databases via application programming interfaces.


## Where do I store my stuff?



<http://www.knowstuff.org/2018/04/google-cloud-platform-adds-more-managed-database-services/>

### Cloud SQL


Google Cloud Database Services comes as part of Google's cloud platform services, and is the last of the big name cloud providers we'll mention here. There are a lot of options for users to choose from, not relational vs non-relational ones. The Cloud SQL option provides managed support for PostgreSQL & MySQL, while the BigTable option provides a petabyte-scale, fully managed NoSQL database service for large analytical and operational workloads.



#### Cloud SQL

Cloud SQL is a fully managed database service that makes it easy to set up, maintain, manage, and administer your relational MySQL and PostgreSQL databases in the cloud.


[View cloud sql →](#)



#### Cloud Bigtable

NoSQL database service for use cases where low latency reads and high throughput writes, scalability, and reliability are critical.


[View cloud bigtable →](#)



#### Cloud Spanner

Cloud Spanner is a mission-critical, scalable relational database service, built to support transactions, strong consistency, and high availability across regions and continents.


[View cloud spanner →](#)



#### Cloud Memorystore

Cloud Memorystore is a fully managed in-memory data store service for Redis built on scalable, more secure, and highly available infrastructure.


[View cloud memorystore →](#)



#### Cloud Firestore

Cloud Firestore is a fast, fully managed, serverless, cloud-native NoSQL document database that simplifies storing, syncing, and querying data for your mobile, web, and IoT apps at global scale. Cloud Firestore is the next generation of Cloud Datastore.

[View cloud firestore →](#)



#### Firebase Realtime Database

The Firebase Realtime Database is a cloud-hosted NoSQL database that lets you store and sync data between your users in real time.

[View firebase realtime database →](#)

<https://cloud.google.com/sql/>

Google control your database, so you can concentrate on your applications. Cloud SQL is ideal for WordPress sites, e-commerce applications, CRM tools, and any other application that is compatible with MySQL, PostgreSQL, or SQL Server.

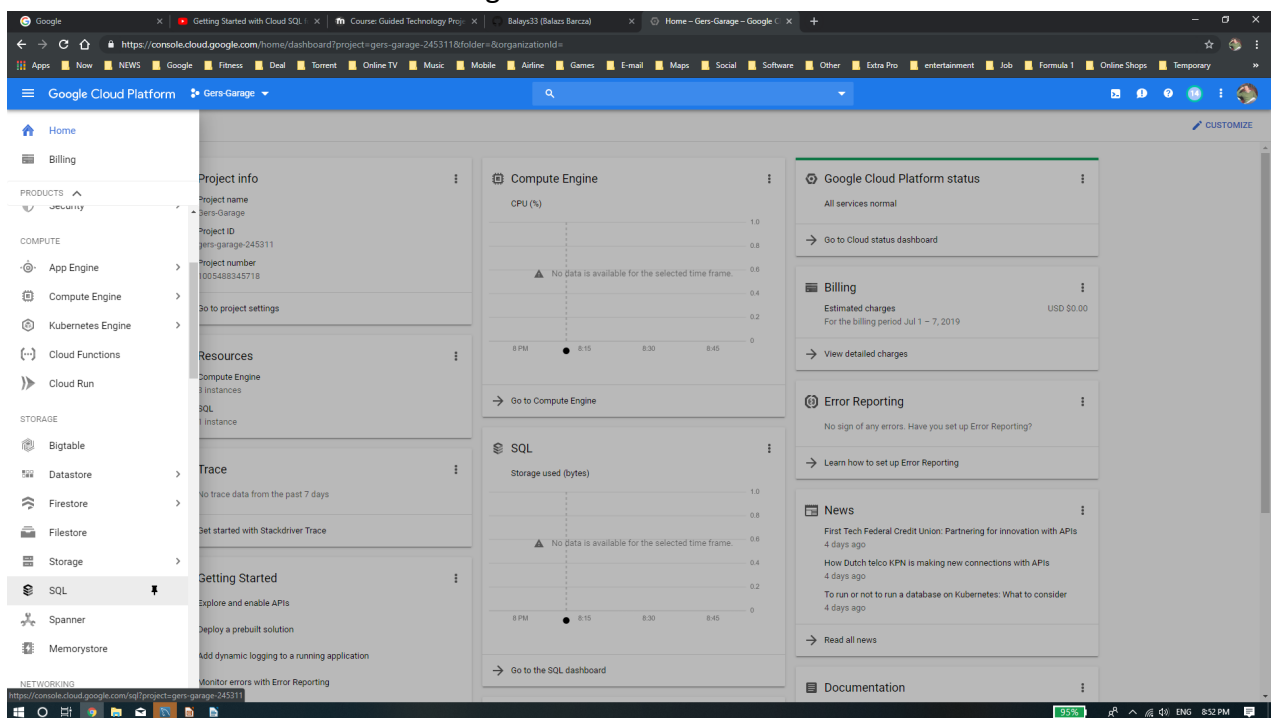
Cloud SQL is simple to use. It does not need any software installation. It automates all your backups, replication, patches, and updates.

Cloud SQL delivers high performance and scalability with up to 10 TB of storage capacity.

Simply configure replication and backups to protect your data. The database is highly available, and automatically encrypted.

## Step by step setting up cloud SQL:

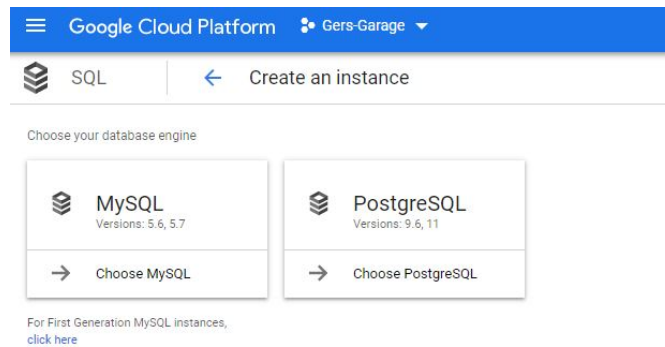
### 1. Create SQL instance with Google Cloud



Google Cloud navigation bar SQL icon

The google give to use to choose different applications, We can create virtual machines or storage etc. This interface is easy to use and clear design. I like because I get lager working place and If I want to use other service I can find the navigation bar always. The platform easy to set up and personal.





We can create MySQL instance or PostgreSQL instance. In my project I will use MySQL because I have experience and I have created my database with MySQL workbench.

We can setup basic parameters with this panel. We can give name of the instance. We can setup password, and the location of the SQL database. We should choose the location but We should leave out the zone any and the Google Cloud will choose the best fit to the project. We can setup the connections (private and public IP).

We can create size of the storage and configurations of the machine. (bigger size better but price is higher) The storage automaticity increase.

Other impotent We can set up when will want to make backup and Failover replica.

High availability

Recommended for all production instances to improve fault tolerance. Failover replica is hosted in a different zone from the master and is billed as a separate instance. [Learn more](#)  
Create failover replica

Enter an ID for your failover replica. ID is permanent. Use lowercase letters, numbers and hyphens. Start with a letter.



## Guided Technology Project – Ger's Garage | SB18002 Balazs Barcza

The screenshot shows the 'Instance details' page for a MySQL instance named 'database-ger-garage'. At the top, there are action buttons: EDIT, IMPORT, EXPORT, RESTART, STOP, DELETE, and CLONE. Below these, a status bar indicates 'Instance is being updated. This may take a few minutes. While this operation is running, you may continue to view information about the instance.' The main content area has tabs for OVERVIEW, CONNECTIONS, USERS, DATABASES, BACKUPS, REPLICAS, and OPERATIONS. The OVERVIEW tab is active, showing a CPU utilization graph (currently empty with 'No data for this time interval'), a 'Connect to this instance' section with public IP address '35.204.130.200' and instance connection name 'gers-garage-245311:eu-west-4:database-ger-garage', and a 'Configuration' section showing 1 vCPU, 3.75 GB memory, 10 GB SSD storage, MySQL 5.7 database version, and auto storage increase enabled. On the right, the 'Operations and logs' section shows a table of recent updates, all with a status of 'Update finished'.

This page gives information about the instance details. We can start or stop the instance. We can connect to the Google shell. There are many ways to connect to the instance. Within GCP, for example, we could use App Engine, Compute Engine or Container Engine. We can connect anywhere by authorizing IP addresses or using Cloud SQL proxy. Google shell provides command line access from the browser.

The screenshot shows a Cloud Shell terminal window with the prompt 'gers-garage-245311'. The user has entered the command 'gcloud sql connect ger-garage2 --user=root --quiet'. The terminal output shows the welcome message and the project name.

This is the command shell.

“

Welcome to Cloud Shell! Type "help" to get started. Your Cloud Platform project in this session is set to **gers-garage-245311**.  
gers-garage-245311.barczabalazs@cloudshell:~ (gers-garage-245311)\$  
gcloud sql connect ger-garage2 --user=root --quiet^C

“

`gcloud sql connect ger-garage2 --user=root --quiet`

The screenshot shows a MySQL terminal window with the prompt 'MySQL [test]'. The user has entered the command to create a table 'test.customers' with columns 'contact\_id', 'first\_name', 'last\_name', 'birthday', and 'customer\_password'. The output shows 'Query OK, 0 rows affected (0.04 sec)'.

create table in GCP shell.

The screenshot shows a MySQL terminal window with the prompt 'MySQL [test]'. The user has entered the command to insert data into the 'test.customers' table. The output shows 'Query OK, 1 row affected (0.02 sec)'.

Insert data into the Database

```
MySQL [test]> select * from customers;
+-----+-----+-----+-----+-----+-----+-----+-----+
| contact_id | first_name | last_name | birthday | Mobile_Number | customer_Email | customer_Comment | customer_password |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | Justin | Miller | 2008-07-04 | -8028 | JustinMMiller@dayrep.com | Just For Feet | 66174090 |
+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)

MySQL [test]> █
```

select \* from customers;

## 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*

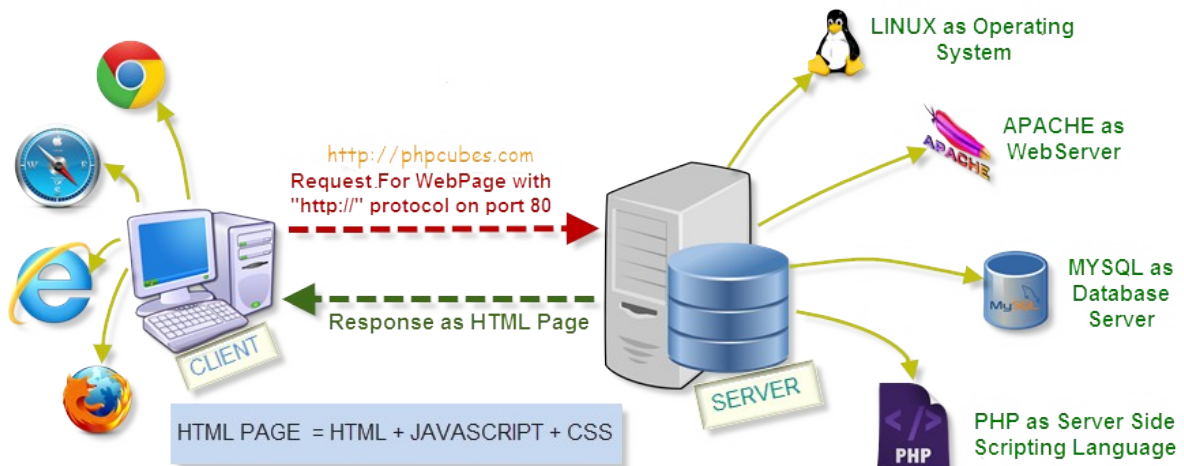
- o Diagrams – Normalised database tables*

---

### 3.1 Part 1: Functional Requirements

*Analysis:*

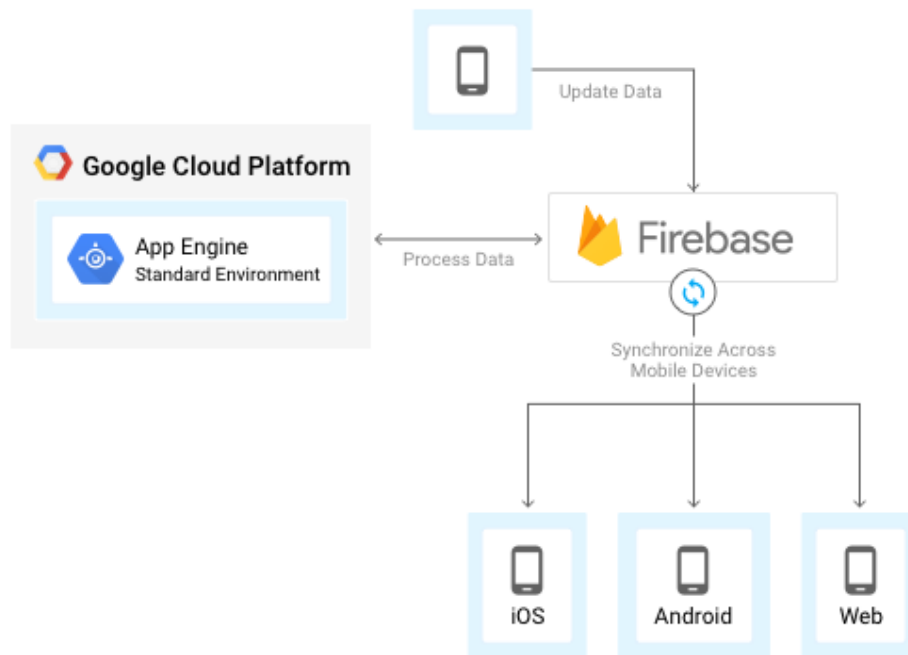
*Gers has a small service company, and He is selling parts and maintenance vehicle. He needs to be online to organize the booking, and the costumers have to manage and book service. The thesis project is aiming for developing a useful tool for project website and application to manage the online booking system.*



<https://people.rit.edu/~agy5732/140/proj2/http>

*This application has to increase sales and has to provide what the customers needs. We have to break down the application into subsystems. I have used the Firebase service and I have broken up small components. I have to select a design pattern. Building a backend service for a mobile app is similar to building a web-based service, with some additional requirements:*

- Limit on-device data storage.
- Synchronize data across multiple devices.
- Handle the offline case gracefully.
- Send notifications and messages.
- Minimize battery drain.

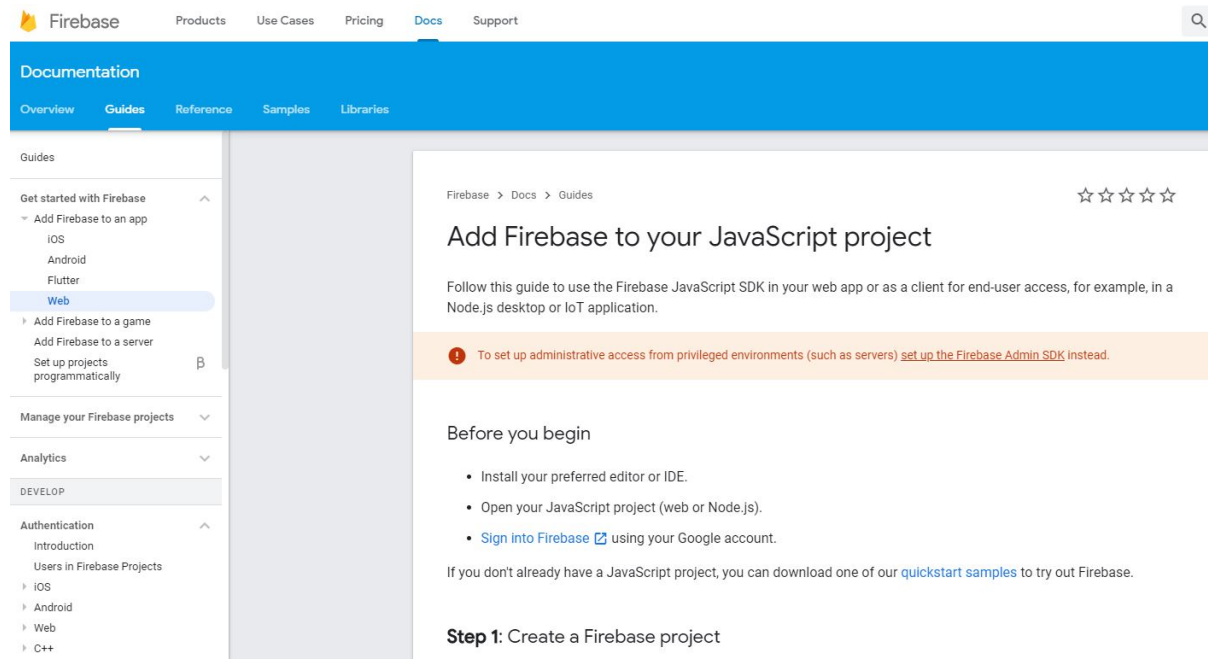


<https://www.slideshare.net/KasperLoevborgJensen/introduction-to-firebase-with-android-and-beyond>

This is web-based design but the website is developed to the mobile-first.

This is a help to make the website scale and It helps to manage the different size of the screen.

I hosted the website on Firebase because It is free and It is well documented.

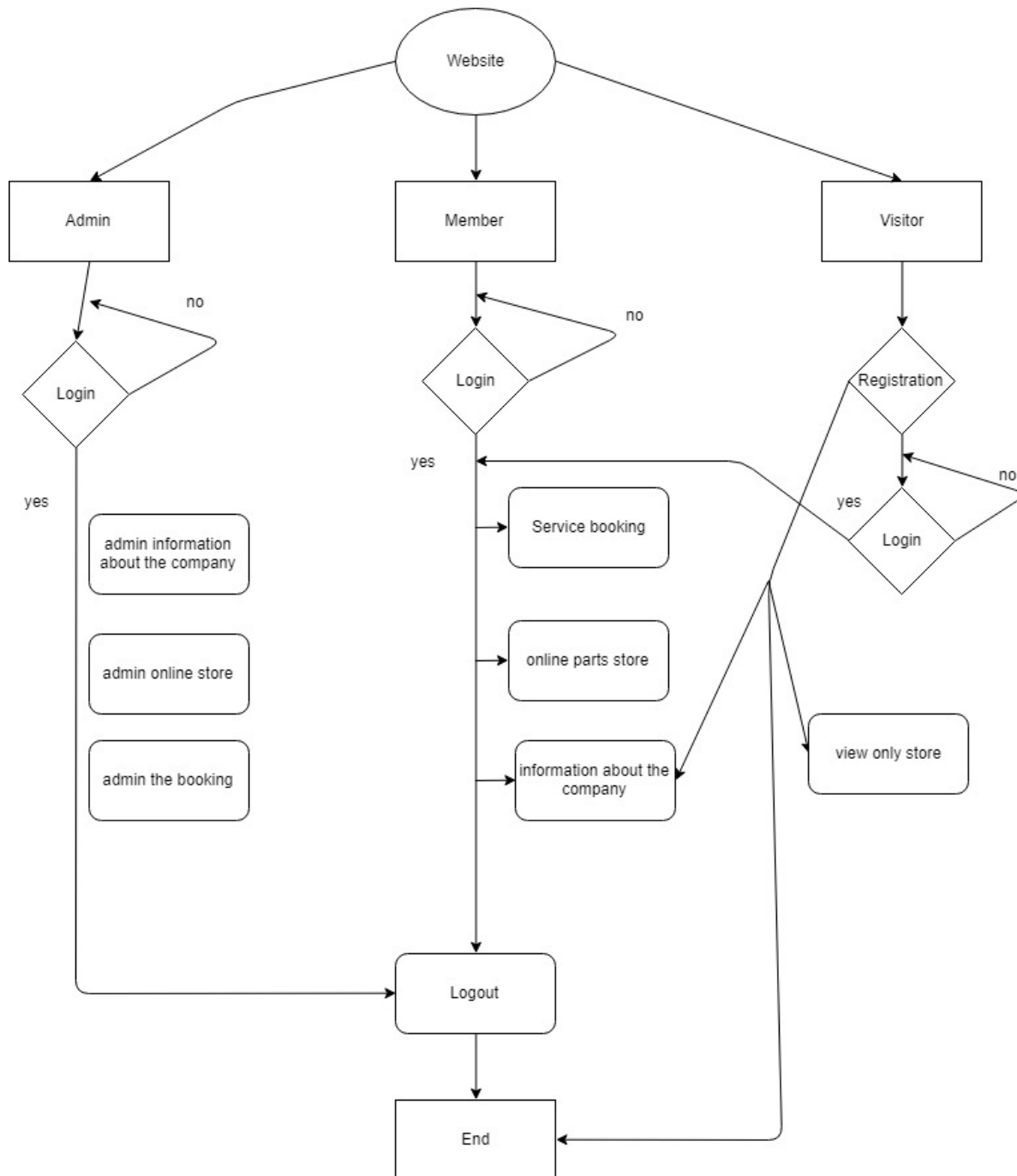


Firebase documentation.

I want to host the SQL in the Google Cloud Platform GCP so It has a support to connect the database with the main website. The firebase easy to scale and give more functions to the Web app.

Firebase is a fully managed platform for building iOS, Android, and web apps that implements automatic data synchronization, authentication services, messaging, file storage, analytics, and more. Starting with Firebase is an efficient way to build or prototype mobile backend services.

- Firebase apps that need a backend service to modify the synchronized data.
- Backend services that run regularly to process or analyze Firebase data.



This graph shows how the website can be accessed with different users.

Three types of user can have the web app.

admin: This person can be Ger or his colleagues who have admin access.

They can modify the website and get information, modified about the booking status and can change the online store manager the staff.

visitor: This user who does not have registration so He or she only can get basic information about the company, can check the service and price and only store, but can not make online shopping and booking.

Member: This person has an account and can log in and out. The person can make a booking and check the online store and buy products.

Programming Language:

I had to choose the programming language. My choose was the HTML and CSS to create webpages. I

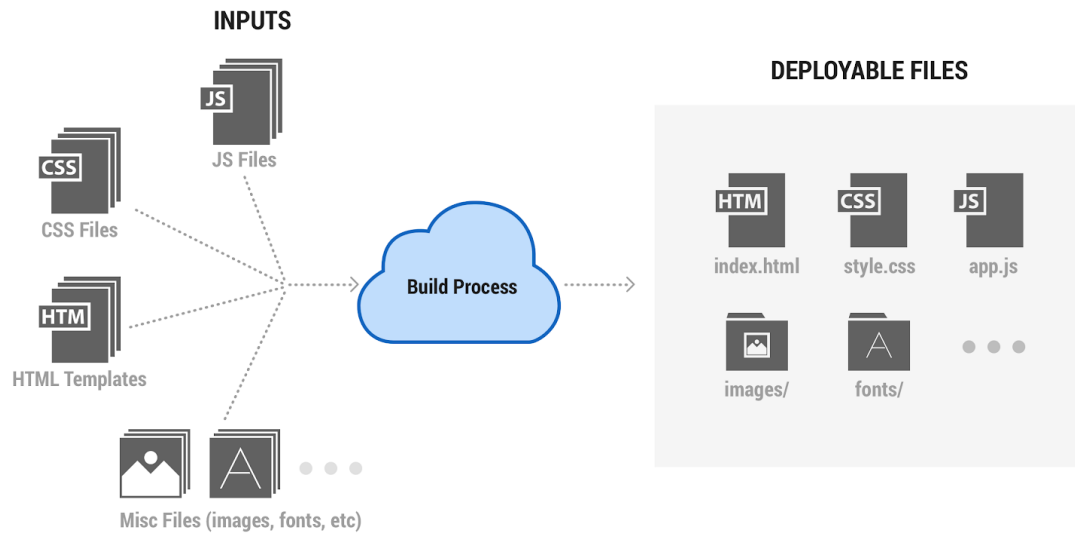
have to write a client-side web application.

JavaScript and PHP good chose. If you want to use the latest and greatest features of the language before they are supported in all browsers.

I have used MySQL to create the database.

Build Process:

The main tasks of the build process are code compilation, files. The end result will be an index.html and the required JavaScript, CSS, and other static assets, like images and fonts, that are needed to deploy the web application.



---

### 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

What is the database?

The software used to manage and manipulate that structured information is called a DBMS (Database Management System). This is one component of DBMS. It is basically a simple list of information. The first step I have to find the entities and relationships. The entities I store information about the database, and the relationships are the links between these entities. I have used in my project the excel to create tables and I have designed the ER diagram with MySQL workbench.

Normalization Rule

Normalization rules are divided into the following normal forms:

First Normal Form

Second Normal Form

Third Normal Form

BCNF

Fourth Normal Form

Data Base			
Customer			
Customer Name		Contact number	E-mail
First Name	Family name	mobile	
Vehicle			
Type	Engine Size	Fuel	body type
Opel	1309 CC	Petrol	5 door Hatchback
Motorbike	?	Diesel	?
Car	?	Hybrid	?
Van	?	electric	?
Vehicle licence		Make date	Colour/brand of
Colour	Number		
grey	05-D-12345	2001	Opel
?	131-W-12345	2019	tata min 40 Brad
+ other not at the list			
Booking Required		Base Cost	Extra cost
1	• Annual Services	200€	
2	• Major Services	-	
3	• Repair/Fault	-	
4	• Major Repair	-	

Customer Complaint	
Staff number	Staff name
2	
3	
4	

Rate/Time	
Day	Time
Monday	9-18
Tuesday	10-10
Wednesday	10-11
Thursday	11-12
Friday	13-14

Post / ITEM		Cost
Post number	type	
1	tyre	
2	brake fluid	
1		40
40		
Cost, impossible status		

This is the basically plan in the notebook.



Ger's

## Database

This is the database Unnormalized Form. This table hold all of the entities

- customers
- vehicle
- booking required
- staff
- date/time
- parts/items
- booking status
- etc

F

First Normal Form

*First Normal Form (1NF)*

*For a table to be in the First Normal Form, it should follow the following 4 rules:*

*It should only have single(atomic) valued attributes/columns.*

*Values stored in a column should be of the same domain*

*All the columns in a table should have unique names.*

*And the order in which data is stored, does not matter.*

<https://www.studytonight.com/dbms/database-normalization.php>

Seco

Second Normal Form (2NF)

For a table to be in the Second Normal Form,

## Guided Technology Project – Ger's Garage | SB18002 Balazs Barcza

It should be in the First Normal form.  
And, it should not have Partial Dependency.

<https://www.studytonight.com/dbms/database-normalization.php>

### Ger's Garage 3NF

Database

The third normal form (3NF) is a normal form used in database normalization. 3NF was originally defined by E.F. Codd in 1971. Codd's definition states that a table is in 3NF if and only if both of the following conditions hold: The relation R (table) is in second normal form (2NF)

3NF A

Customer Code	Customer Password	Customer Name	Customer mobile number	Customer e-mail	Customer Bday	Customer Comments
123	apple	Richard A. Addis	843-371-9785	RichardA.Addis@jo urapide.c om	02/27/89	etc
222	citrom	Joe M. Rosales	909-867-4863	JoeMRosales@jo urna pide.com	02/27/89	etc
322	malna	Shawn C. Anderson	570-592-5221	ShawnCA nderson@ armyspy.c om	05/10/57	etc

3nf C

Vehicle license details	Vehicle type	Vehicle Make	Vehicle year	Vehicle engine type	Customer Code	Date / Day	Time	Number of day	Date Month	Year	Staffs	Vehicle statuses	Booking code number
14D2345	2	Ford Fiesta	2014	1	123	2	10	22	3	2019	3	2	1
191D2343	2	skoda fabia	2019	4	222	3	11	21	3	2019	2	1	2
182D4345	2	BMW	2014	3	123	1	10	22	3	2019	4	4	3
18W1234	1	yamaha	2016	2	322	5	12	12	3	2019	3 1	3	4
17E467	2	opel	2107	2	322	2	1	12	3	2019	3 1	5	4

3NF D

Items / Parts	Item Cost
12	23
10	100
33	150
22 55	123
55	56

3nf E

Vehicle license details	Date / Day	Time	Number of day	Date Month	Year
14D2345	2	10	22	3	2019
191D2343	3	11	21	3	2019
182D4345	1	10	22	3	2019
18W1234	5	12	12	3	2019
17E467	2	1	12	3	2019

3NF F

Booking Required	Booking Cost
1	100
2	200
3	300
4	400

Third Normal Form (3NF)  
A table is said to be in the Third Normal Form when,

It is in the Second Normal form.  
And, it doesn't have Transitive Dependency.

<https://www.studytonight.com/dbms/database-normalization.php>

## 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.

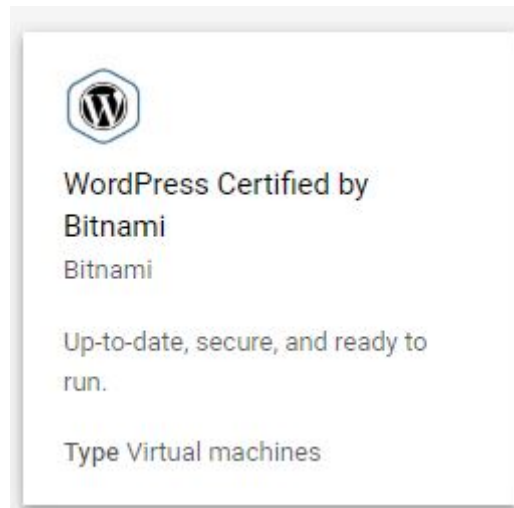
Step by step Setting up a WordPress site on Google Cloud.

Step 1:

I have created a project

Step 2:

Marketplace I have created a virtual 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.

### ← New WordPress Certified by Bitnami deployment

Deployment name  
wordpress-1

Zone  
us-central1-f

Machine type  
small (1 shared ... 1.7 GB memory Customize

Boot Disk  
Boot disk type  
Standard Persistent Disk

Boot disk size in GB  
10

Networking  
Network  
default

Subnetwork  
default (10.128.0.0/20)

External IP  
Ephemeral

Firewall  
Add tags and firewall rules to allow specific network traffic from the Internet  
☒ Allow HTTP traffic  
Source IP ranges for HTTP traffic  
0.0.0.0/0, 192.169.0.2/24  
☒ Allow HTTPS traffic  
Source IP ranges for HTTPS traffic  
0.0.0.0/0, 192.169.0.2/24

Deploy



#### WordPress Certified by Bitnami overview

Solution provided by Bitnami

\$13.61 per month estimated

Effective hourly rate \$0.019 (730 hours per month)

Details

#### Software

Operating System	Debian (9)
Software	Apache (2.4.39)
	ImageMagick (6.9.8)
	lego (2.6.0)
	MySQL (8.0.16)
	OpenSSL (1.0.2s)
	PHP (7.3.6)
	phpMyAdmin (4.9.0.1)
	SQLite (3.28.0)
	Varnish (6.0.3)
	WordPress (5.2.2)
	WP-CLI (2.2.0)

#### Documentation

[Access using SSH](#)

Configure SSH keys to access the application as the user "bitnami".

[Using SFTP](#)

Use this guide to upload files using SFTP.

[MySQL access credentials](#)

Use username "root" and the temporary password to access MySQL.

[Change your MySQL root password](#)

Change your temporary MySQL root password by following these instructions

[Accessing phpMyAdmin](#)

Access phpMyAdmin via an SSH tunnel using this guide.

[Adding plugins with privileges](#)

Some plugins need privileged access to install. Edit privileges with this guide.

[Installation directory structure](#)

Learn how application files, libraries and configuration files are organized.

[Terms of Service](#)

#### Step 4:

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

wordpress

WordPress Certified by Bitnami  
Solution provided by Bitnami

Site address <http://35.204.159.181/>

Admin URL <http://35.204.159.181/wp-admin/>

Admin user user

Admin password (Temporary) HQ4seoNzHWmi

Instance wordpress-2-vm

Instance zone europe-west4-c

Instance machine type f1-micro

More about the software

Get started with WordPress Certified by Bitnami

Log into the admin panel SSH

WordPress logo

Username or Email Address  
user

Password  
.....

☐ Remember Me

Log In

Create easy website.

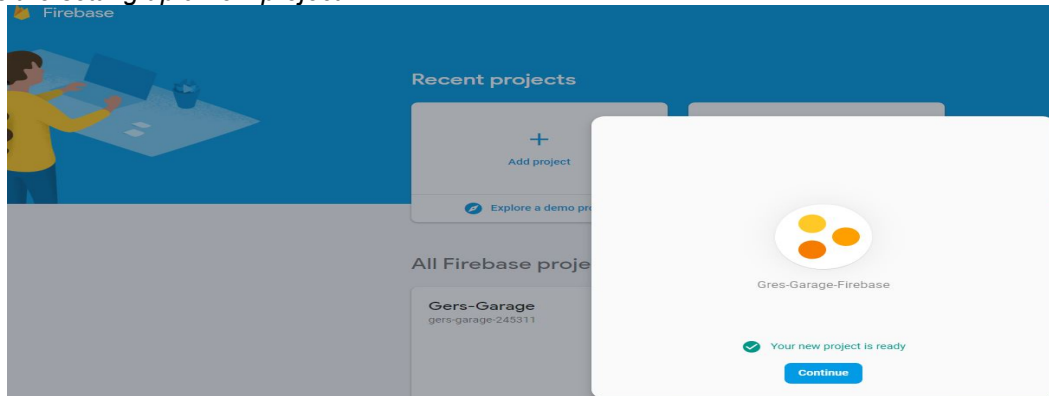
## 2. Firebase:

Firebase is a mobile and web application development platform developed by Firebase, Inc. in 2011, then acquired by Google in 2014.

ADD MORE INFO

I will use this platform to host my website.

This is the setting up a new project.



The project ready to use.

I am using the Visual Studio Code software to develop my application. I have to install the firebase in the terminal.

`npm install -g firebase-tools` This is Node.js® is a JavaScript runtime built system.

## Add Firebase to your web app

1 Register app

2 Add Firebase SDK

Copy and paste these scripts into the bottom of your `<body>` tag, but before you use any Firebase services:

```
<!-- The core Firebase JS SDK is always required and must be listed first -->
<script src="/__/firebase/6.3.0/firebase-app.js"></script>

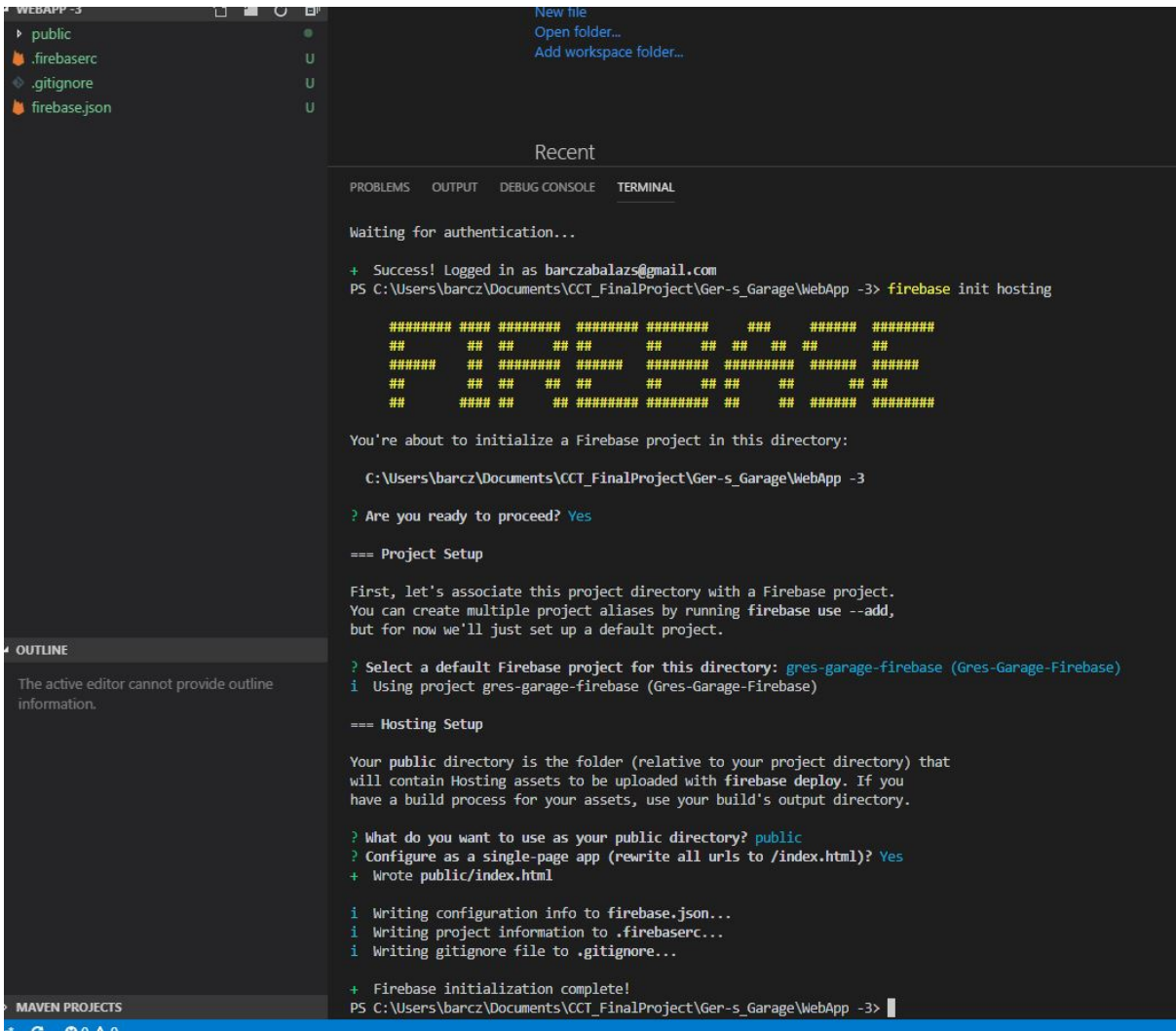
<!-- TODO: Add SDKs for Firebase products that you want to use
https://firebase.google.com/docs/web/setup#reserved-urls -->

<!-- Initialize Firebase -->
<script src="/__/firebase/init.js"></script>
```

Learn more about Firebase for web: [Get Started](#), [Web SDK API Reference](#), [Samples](#)

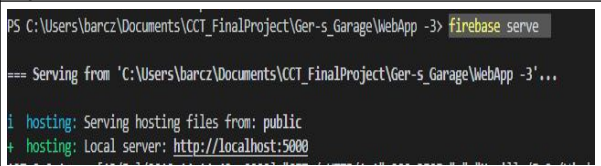
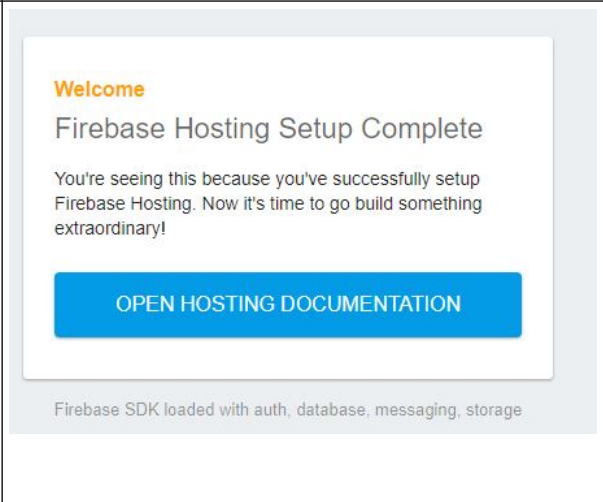
This is the code has to copy to the html file.

I have created a new project folder on my computer. I have a login to the Firebase. Now The firebase created all of the necessary files I need to my website.



After I have run the command All of the project files is ready to use.

Next step to run the app locally. I have used the command  
firebase serve  
After I have run the command I can check my website is running locally.  
<http://localhost:5000>

	
firebase serve	<a href="http://localhost:5000/">http://localhost:5000/</a>



*firebase deploy*

*This command help to host the website in the cloud.*

```
.36"
Shutting down...
Terminate batch job (Y/N)?
Terminate batch job (Y/N)? y
PS C:\Users\barcz\Documents\CCT_FinalProject\Ger-s_Garage\WebApp -3> firebase deploy

=== Deploying to 'gres-garage-firebase'...

i deploying hosting
i hosting[gres-garage-firebase]: beginning deploy...
i hosting[gres-garage-firebase]: found 1 files in public
+ hosting[gres-garage-firebase]: file upload complete
i hosting[gres-garage-firebase]: finalizing version...
+ hosting[gres-garage-firebase]: version finalized
i hosting[gres-garage-firebase]: releasing new version...
+ hosting[gres-garage-firebase]: release complete

+ Deploy complete!

Project Console: https://console.firebase.google.com/project/gres-garage-firebase/overview
Hosting URL: https://gres-garage-firebase.firebaseio.com
PS C:\Users\barcz\Documents\CCT_FinalProject\Ger-s_Garage\WebApp -3> |
```

Hosting URL: <https://gres-garage-firebase.firebaseio.com>

This is the URL where I can find the website.

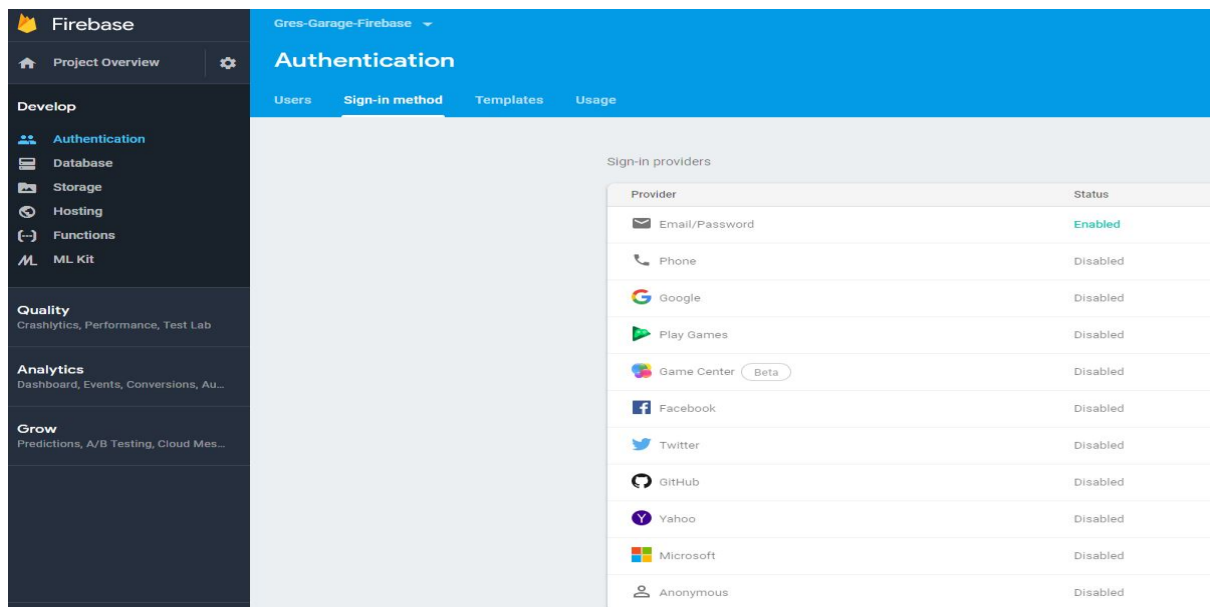
---

## 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.

1. Login:

I have chosen the firebase because It gives lots of different options to create a login method. The user can be authentication different way. It is easy to set up and implement this future to the program. I have chosen the email way because I can use the email to send the marketing message to the user. I can implement another way If I have more time.



This screenshot is showing the other sign-in methods.

- Email/Password
- Phone
- Google
- Play Games
- Game Center Beta
- Facebook
- Twitter
- GitHub
- Yahoo
- Microsoft
- Anonymous

---

### 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