



---

## **Final Project**

Ger's Garage

Balazs Barcza

Student Number : SB18002

---

---

*Date :*

15/06/2019

**Login: admin@gmail.com**

**password:123456**

github link:

**[https://github.com/Balays33/Ger-s\\_Garage](https://github.com/Balays33/Ger-s_Garage)**

website link:

**<https://gres-garage-firebase.web.app/>**

# 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
<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: INDEX.HTML.....	14

7.2	PART 2: LOGIN.HTML & LOGIN.JS.....	14
7.3	PART 3: ADMIN.HTML & ADMIN.JS.....	14
<b>8</b>	<b>LIST OF REFERENCES:.....</b>	<b>16</b>

# *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's garage is the small business, He is a mechanic Who needs a new website or mobile app to manage his business. The website has to help him to manage his business. It has to handle invoice booking and online shop.

His garage maintenance and service small and medium vehicles ( motorbikes, cars, small vans, and small buses).

He is selling car parts so the website has to have an online shop.

His garage has few employees so he has to manage they are roster schedule. The customer can book service if is available. If no available mechanic can not be a book. 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.

### **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:**

Customer can register on the website or log in If the customer has an account. The client can book 4 types of service.

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

**Login Service:**

If the clients have accounted the website has to remember their details.  
The system has to have database support.

**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 has admin access to the site and He can manage the booking and the staff roster.

**Printing services:**

The system has to make an invoice, print out the schedule any particular day.

**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 Fiesta). 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:**

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:**

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.

---

## **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.



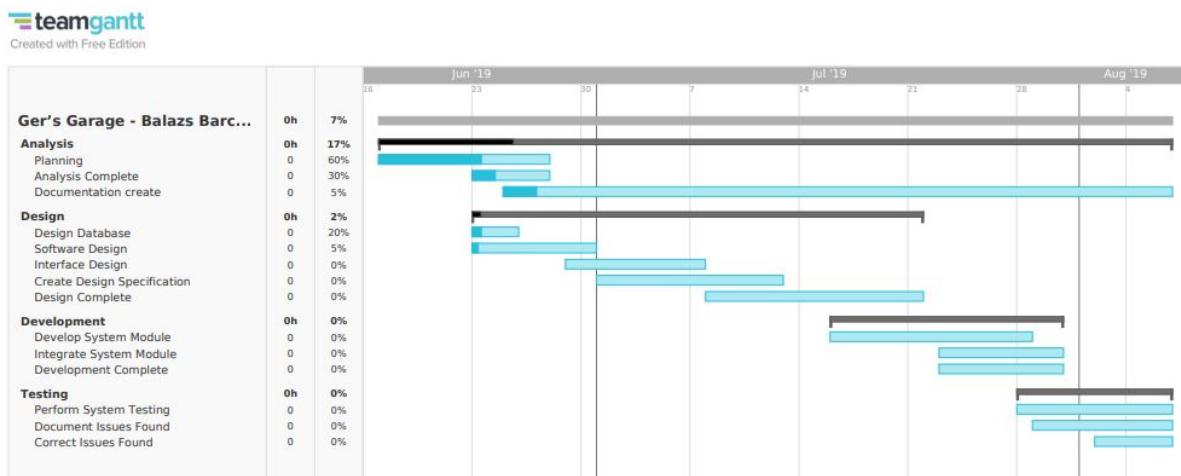
<https://medium.com/level-up-web/developer-roadmaps-all-in-one-place-75c0402db0e0>

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 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/>

Project need knowledge:

**Web design**

HTML, CSS, JSON, hosting,

**Programing skill**

JavaScript

**Database**

SQL, firebase Realtime database and Database integration

**Security**

access control, cybersecurity

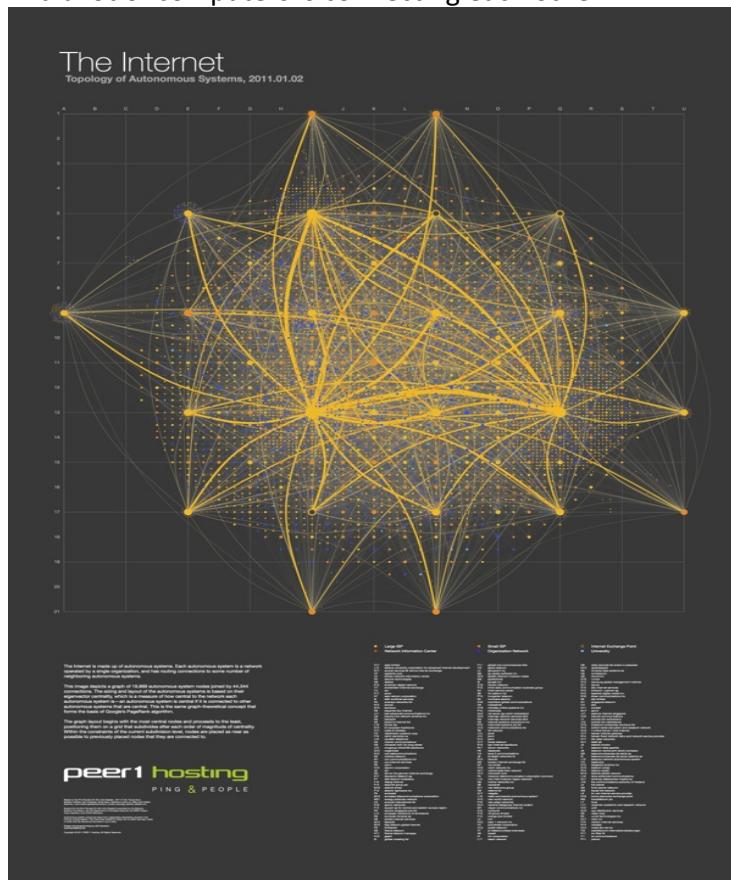
## 2 Literature Review:

*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:**

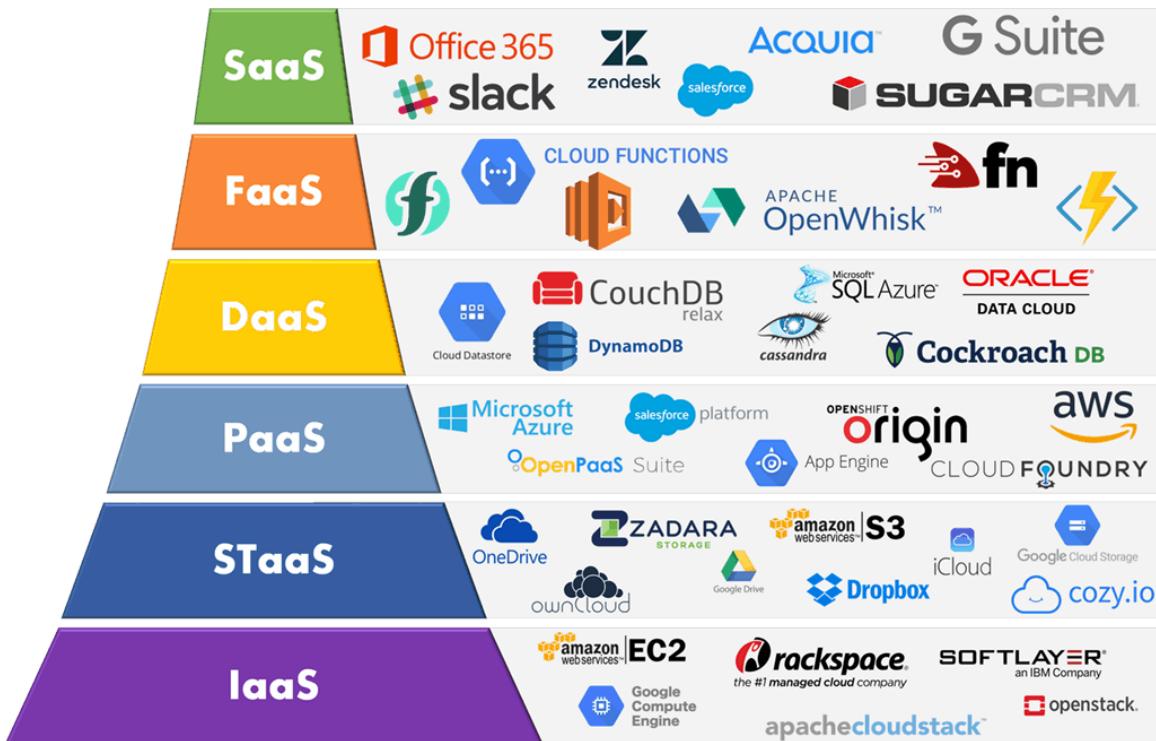
- Virtualization- cloud computing utilizes server and storage virtualization extensively to allocate/reallocate resources rapidly
- Multi-tenancy-resources are pooled and shared among multiple users to gain economies of scale
- Network-access- resources are accessed via web-browser or thin client using 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.

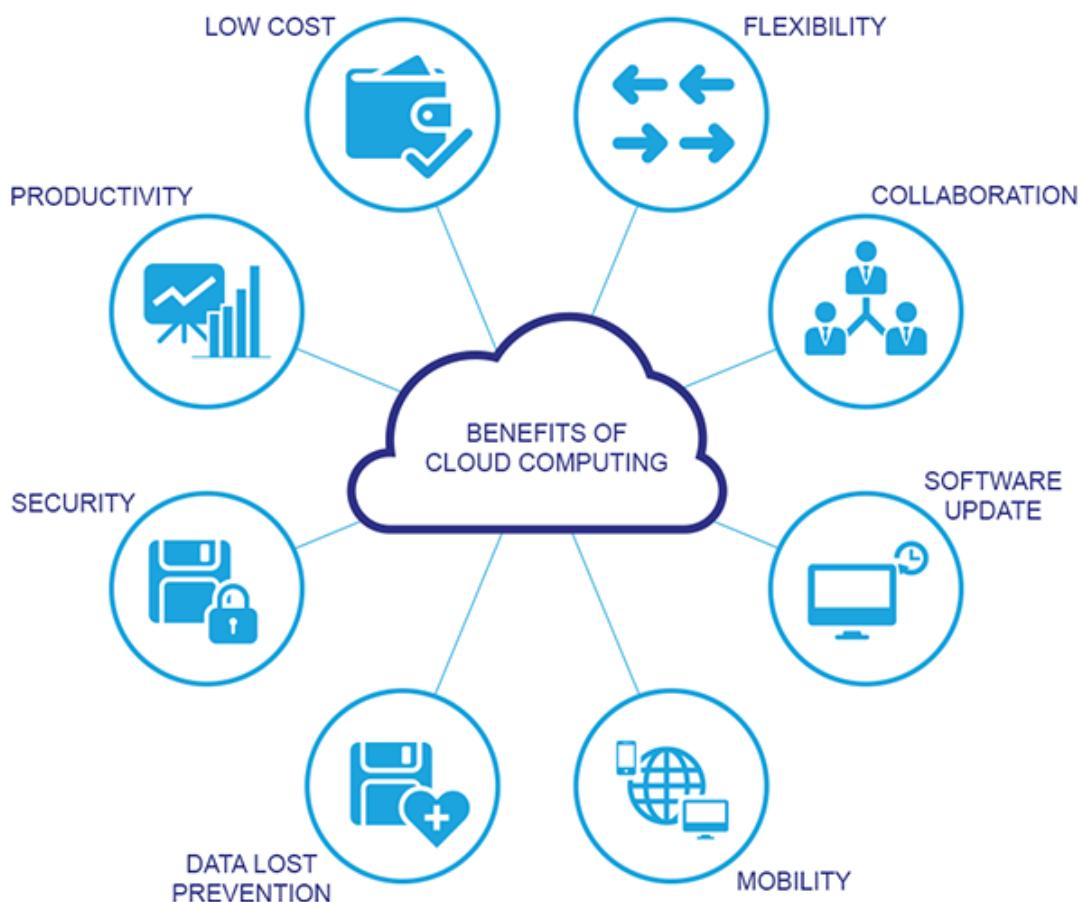
**Cloud computing services offer numerous benefits:**

- 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:**

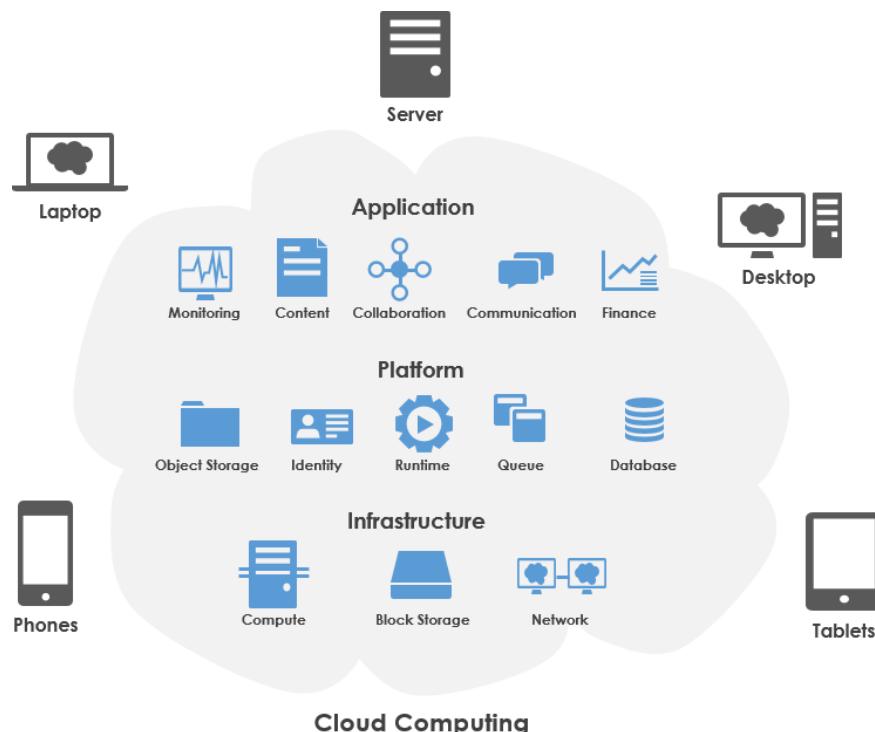
- Cost  
If you use Cloud computing You do not need to buying hardware and software and setting up and running on-site datacenters—the racks of servers, the round-the-clock electricity for power and cooling.
- Productivity  
Datacenters typically require a lot of “racking and stacking”—hardware setup, 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 scaled, so the users can choose the right size of IT resources. Can make computer power stronger, storage size, change location.
- Speed  
The system more flexible, 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.

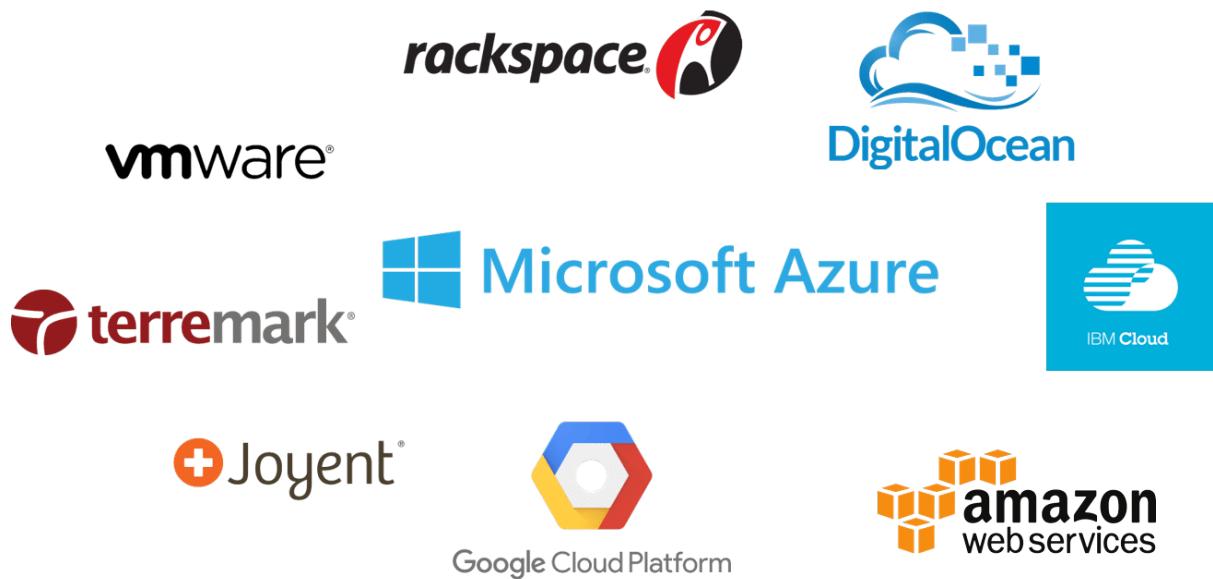
---

## 2.2 Part 2: Google Cloud Platform

### 1. Cloud Service Providers

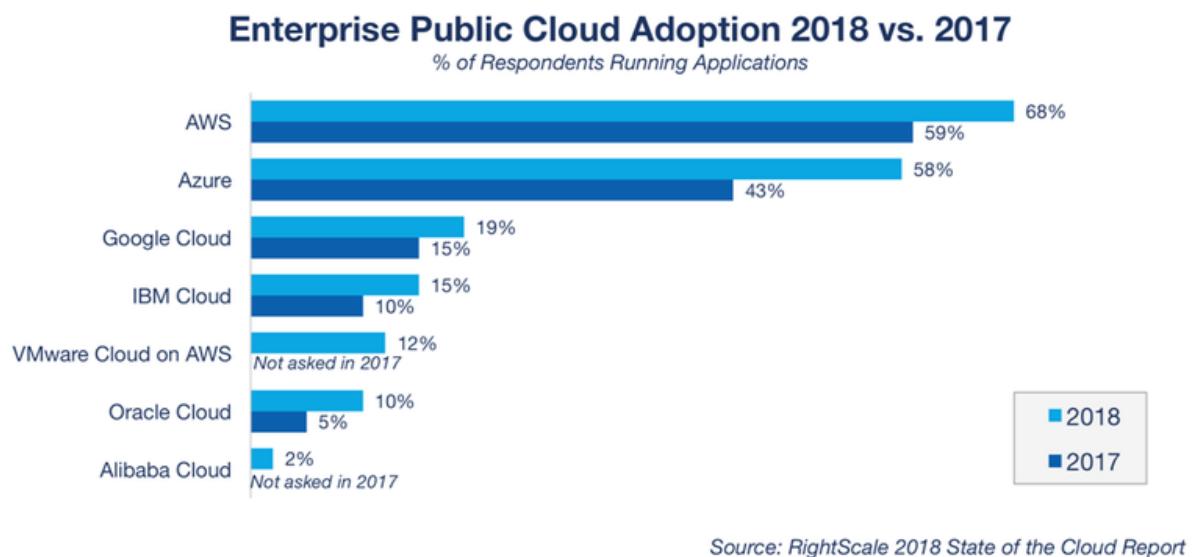
Cloud Computing Services are Information Technology (IT) as a service over the Internet. Cloud computing is a term which is used for storing and accessing data over the internet. It doesn't store any data on the hard disk of your PC. Cloud computing helps you to access your data from a remote server.

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



[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, Diagflow 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?

Object	In-memory	Non-relational	Relational	Warehouse		
 Cloud Storage Binary or object data Images, media serving, backups	 Cloud Memorystore (beta) Web/mobile applications, gaming Cache, game state, user sessions	 Cloud Datastore Hierarchical, mobile, web User profiles, Game State	 Cloud Bigtable Heavy read + write, events AdTech, financial, IoT	 Cloud SQL MySQL, PostgreSQL CMS, eCommerce	 Cloud Spanner RDBMS+scale, HA, global Transactions, Ad/Fin/MarTech	 BigQuery Enterprise Data Warehouse Analytics, Dashboards
						

<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 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.

## 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

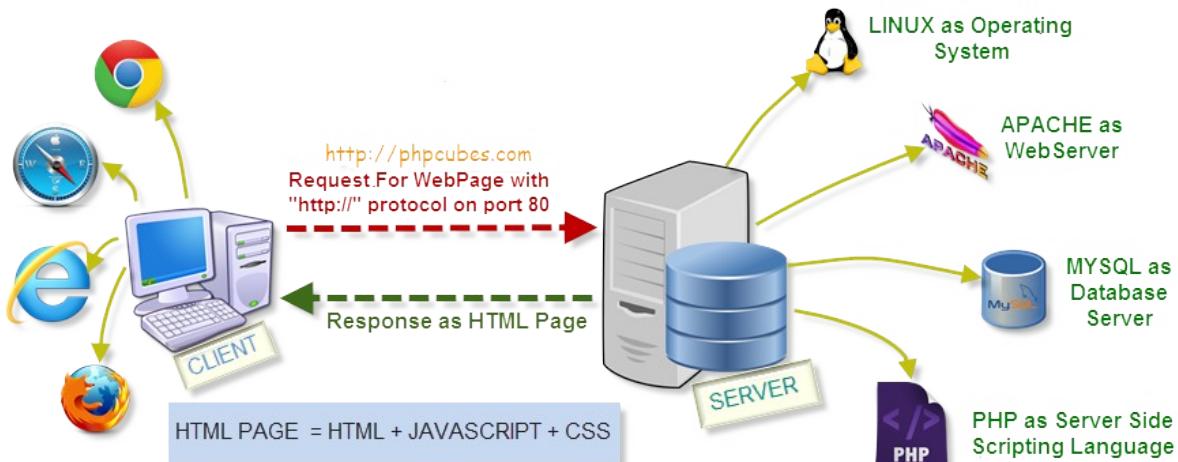
o Diagrams – Normalised database tables

---

### 3.1 Part 1: Functional Requirements

*Analysis:*

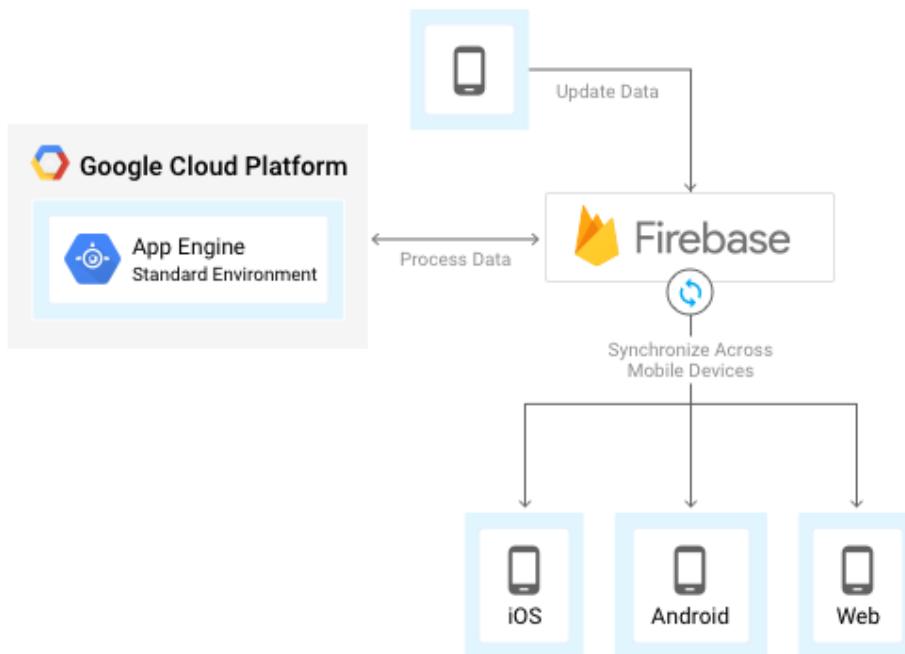
*Gers has a small service company, and He is selling parts and maintenance vehicle. He needs to be online to organize the booking, and the costumers have to manage and book service. The thesis project is aiming for developing a useful tool for project website and application to manage the online booking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

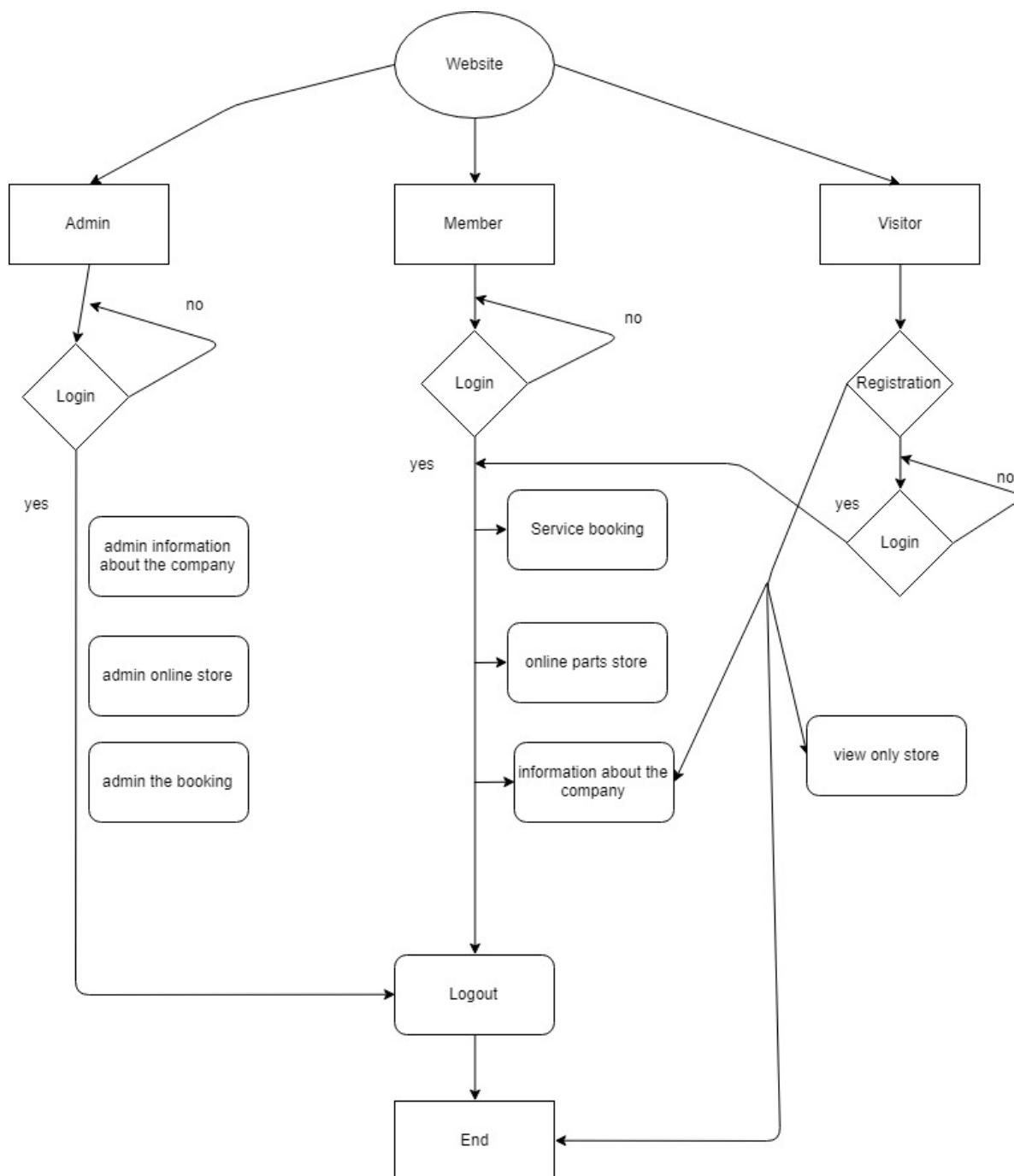
The screenshot shows the Firebase Documentation website. The navigation bar includes links for Products, Use Cases, Pricing, Docs (which is underlined), and Support. A search icon is also present. The main menu has tabs for Overview, Guides (which is selected and highlighted in blue), Reference, Samples, and Libraries. On the left, a sidebar titled 'Guides' contains sections for 'Get started with Firebase' (with 'Add Firebase to an app' expanded) and 'Manage your Firebase projects'. Under 'Add Firebase to an app', 'Web' is selected and highlighted in blue. Other options include iOS, Android, Flutter, and a general 'Add Firebase' section. The main content area shows the title 'Add Firebase to your JavaScript project' with a breadcrumb trail: Firebase > Docs > Guides. Below the title is a note: 'Follow this guide to use the Firebase JavaScript SDK in your web app or as a client for end-user access, for example, in a Node.js desktop or IoT application.' A warning message in an orange box says: 'To set up administrative access from privileged environments (such as servers) [set up the Firebase Admin SDK instead](#).' The 'Before you begin' section lists steps: 'Install your preferred editor or IDE.', 'Open your JavaScript project (web or Node.js).', and 'Sign into Firebase [using your Google account](#)'. A note at the bottom says: 'If you don't already have a JavaScript project, you can download one of our [quickstart samples](#) to try out Firebase.' The page has a rating of five stars.

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.



Programming Language:

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

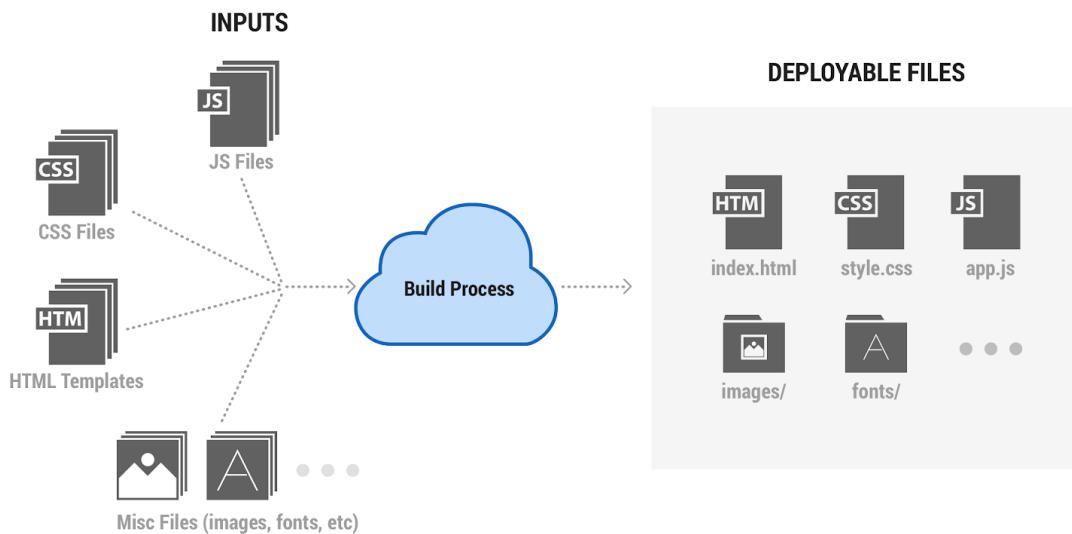
have to write a client-side web application.

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

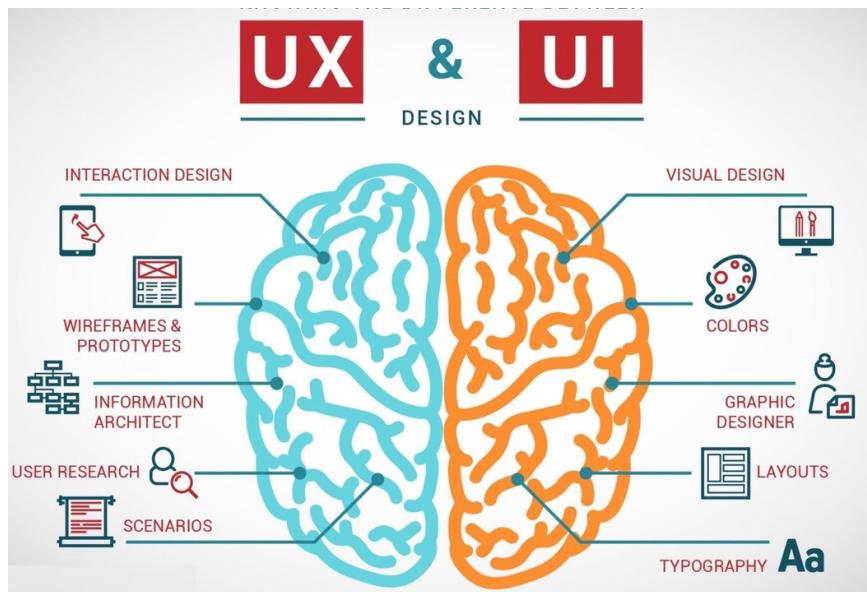
11. This should contain an argument as to how this suggested interface supports each of the use cases specified in the analysis

- o Diagrams – Screen designs, either pen-and-paper or computer drawn of how the user interface will appear

UI (user interface ) is the process of designing interface in software and It is focused on looks or style. The well-designed software easy to use and the user will spend more time on the app. The UI design is graphical user interfaces but it can be voice-controlled.

The main access points to the software are the user interface, where the users interact with designs. Graphical user interfaces (GUIs) are designs' control panels. It van be voice-controlled interfaces, It is going to be the future because the users can control the software without using the graphic interface. We can make 2d or 3d graphical interface but my case it is only 2D.

I should create the illusion that users aren't interacting with a website so much as they're trying to attain goals directly and as effortlessly as possible. My design should have as many enjoyable features as are suitable.

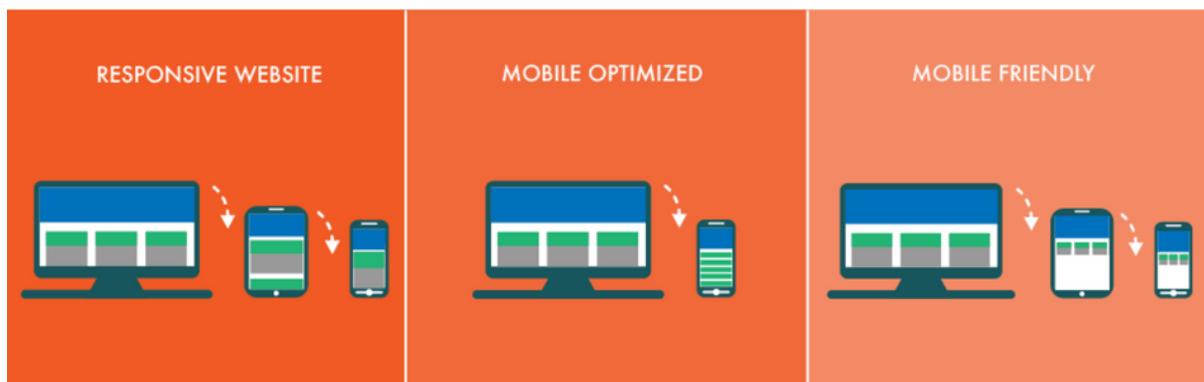


<https://uxdesign.cc/why-ux-and-ui-should-remain-separate-7d6e3addb46f>

Interface design is one of the key elements of the project. If the website or the app is too slow the user won't use the service. If the user interface is not easy to understand We will have the same problem. It has to design user-friendly and clear design. It has to be mirror the company profile so It was important to use the company logo colours. I have chosen the white and blue colour to make the web app visible.

We have to be careful and decide what type of website we would like to design.

#### MOBILE RESPONSIVE VS. OPTIMIZED VS. FRIENDLY: WHAT'S THE DIFFERENCE?



<https://compulse.com/mobile-responsive-vs-optimized-vs-friendly-whats-difference/>

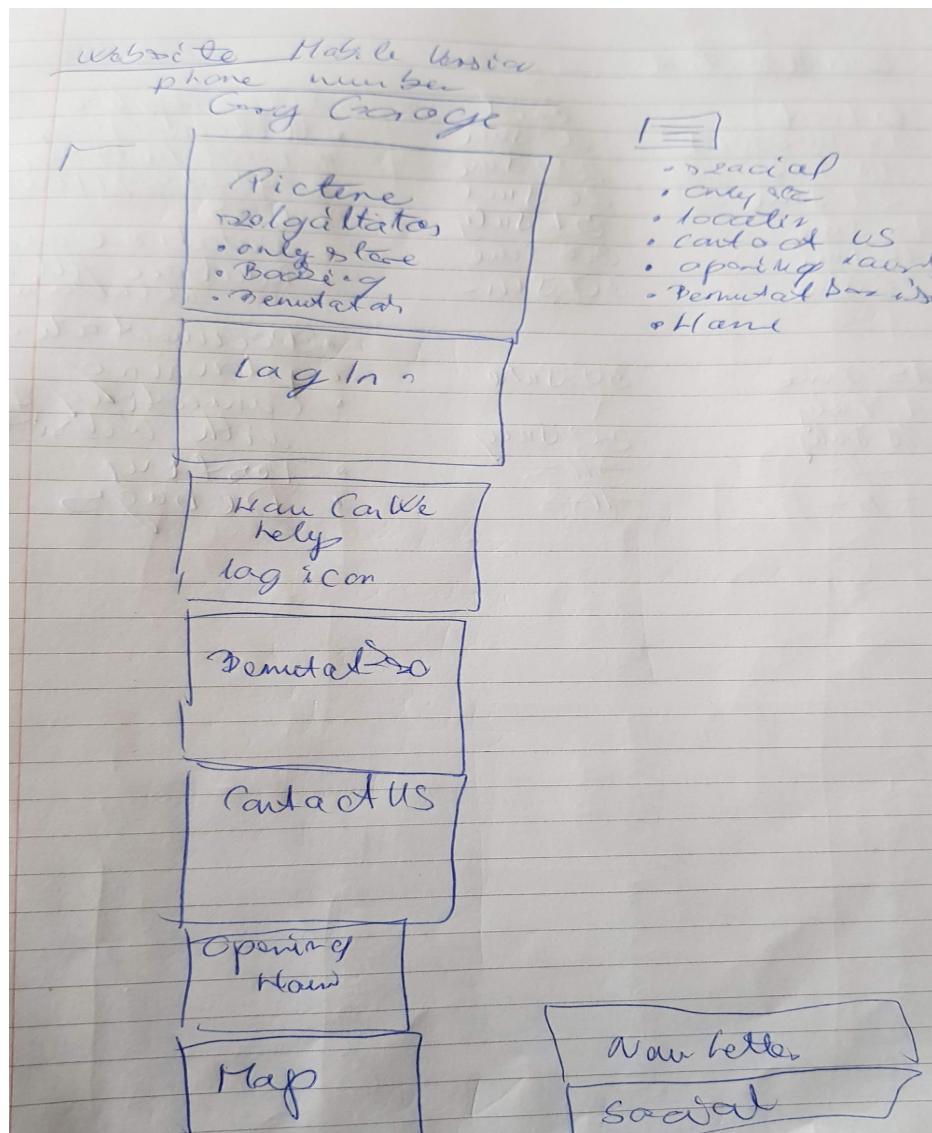
The successful businesses have to have a user-friendly website in 2019. More the more people use mobile phones than ever before, so the website has to be the response to the smartphone. I can design the website to mobile devices: mobile responsive, mobile-optimized and mobile-friendly.

1. The mobile responsive website is the most user-friendly. This type of website is design to a different size. This is focused on the mobile and tablet and PC interfaces.

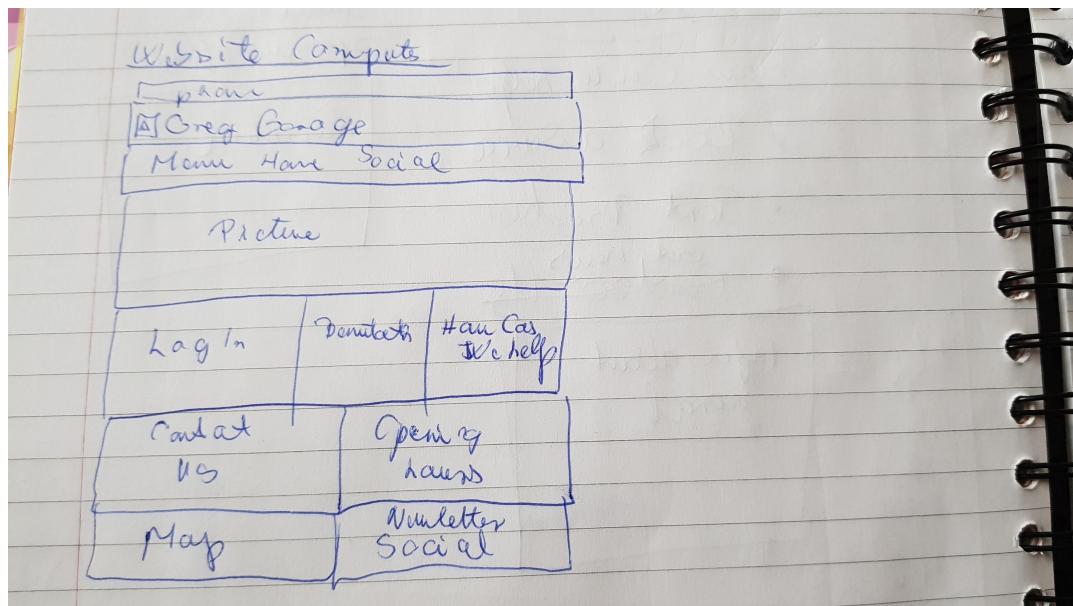
2. A mobile-optimized site is build to the mobile user. It can use the PC platform but not the best user-friendly.
3. A mobile-friendly design is the most limited “friendly”. These websites will still be convenient to mobile onlookers, but they aren’t designed for the optimal user activity.

#### Why You Need a Mobile-Friendly Website in 2019?

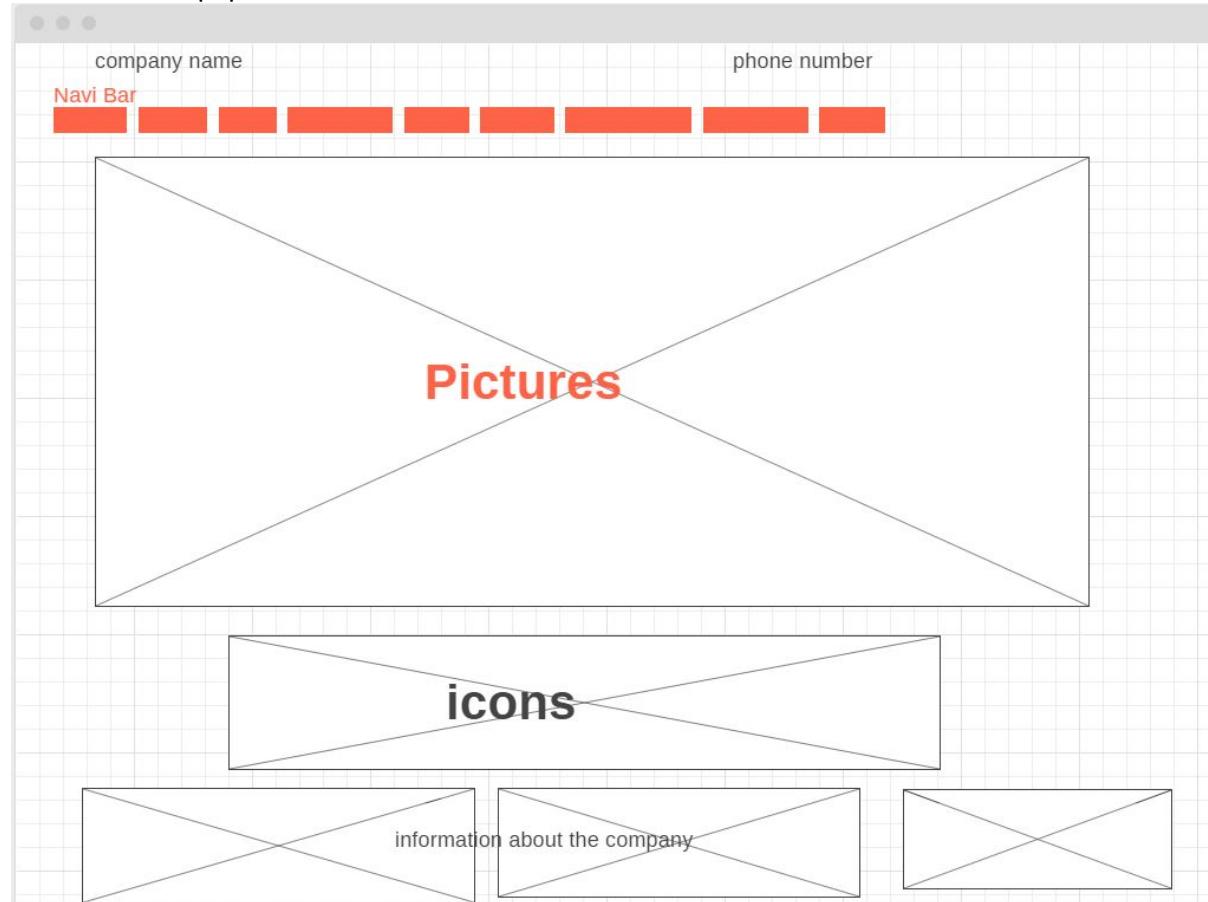
- A mobile-friendly design makes a website more available.
- Improved user experience.
- Responsive website architecture is cost-effective.
- A mobile-friendly website increases sales.
- A mobile improved site engages Google ranking.



This is the mobile -friendly version design. My website has to be adjusted to the different screen size so I have focused the mobile version.



Scratch on the paper wireframe.



This is the early plan basic scratch about the website.

I did not plan too many separate pages, I wanted to give all of the information on the main page and a couple of pages to future growth.

I have to integrate the Navi bar all of the pages because it is helps the user easy navigation.

When we are designing a website we have a couple of great tools to speed up the design progress. We can use the CSS and we can program all of the small design parts of the website. We can use CSS framework directed at responsive , or we can use those optional together. I have used two frameworks, one of the bootstrap and another one the google material design.

### Bootstrap:

*Bootstrap is a free and open-source CSS framework directed at responsive, mobile-first front-end web development. It contains CSS- and (optionally) JavaScript-based design templates for typography, forms, buttons, navigation and other interface components. 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.*



The screenshot shows a snippet of HTML code for a Bootstrap navbar. The code includes a logo icon and the word "Bootstrap". The code is syntax-highlighted with colors for different elements like tags and class names.

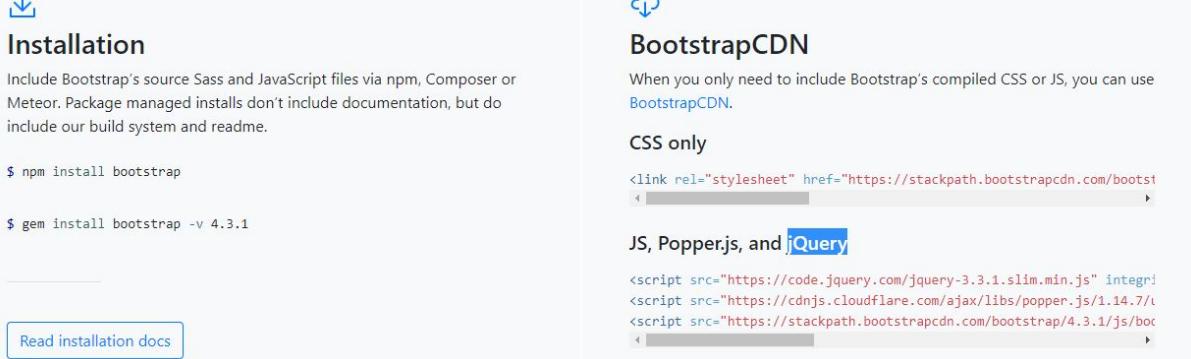
```
<!-- Image and text -->
<nav class="navbar navbar-light bg-light">
  <a class="navbar-brand" href="#">
    
    Bootstrap
  </a>
</nav>
```

Adding images to the .navbar-brand will likely always require custom styles or utilities to properly size.

<https://getbootstrap.com/>

You can use the bootstrap in two ways. You can install to your local computer or you include in your HTML file the bootstrap components.

style-sheet and the jQuery.



The screenshot shows two sections of the Bootstrap documentation. On the left, under "Installation", it shows how to install Bootstrap via npm or gem. On the right, under "BootstrapCDN", it shows how to include Bootstrap's compiled CSS or JS using a CDN. Both sections include sample code snippets.

**Installation**

Include Bootstrap's source Sass and JavaScript files via npm, Composer or Meteor. Package managed installs don't include documentation, but do include our build system and readme.

```
$ npm install bootstrap
$ gem install bootstrap -v 4.3.1
```

[Read installation docs](#)

**BootstrapCDN**

When you only need to include Bootstrap's compiled CSS or JS, you can use [BootstrapCDN](#).

**CSS only**

```
<link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css" integrity="sha384-ggOyR0iXCbMQv3F4QOEHQBZJ3jATkI1+RJ0BSJ8nXuHZGCEPtDwqH+o...
```

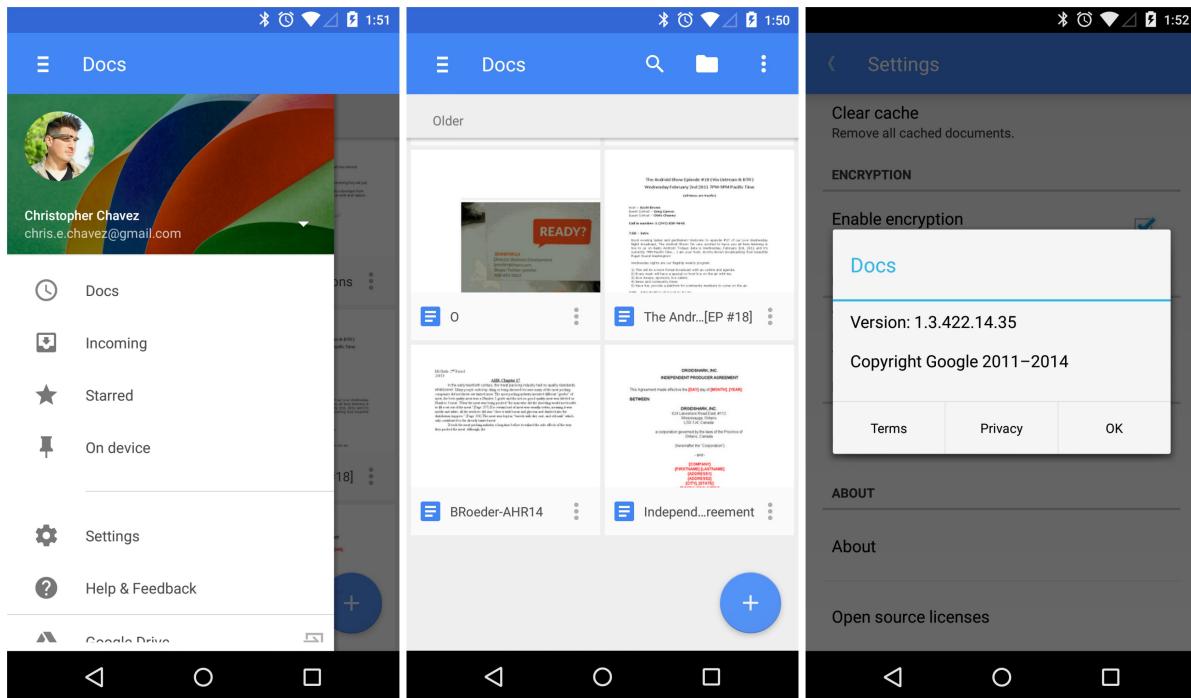
**JS, Popper.js, and jQuery**

```
<script src="https://code.jquery.com/jquery-3.3.1.slim.min.js" integrity="sha384-q8i/X+9jwhV3D2O+tbhMbbA1h0DTSSo3UZJLZw2MPHEgkx/LvqK2f...">
<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js" integrity="sha384-UOJZtSDW3ZC7ZUJw1hZUjZ" ...>
<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js" integrity="sha384-...
```

### Google material design:

Material Design is a design language created for Google Android's new OS. It is originally focused on touch-based mobile app design it should be possible to separate the same ideas into web design. It has online documentation and it is very detailed. Google has created a material design language. This design is focus to make the Android app more user-friendly. It helps the developer make faster android interface. Mobile interfaces are created out of layered material objects like rectangular bars or circular buttons. Content(text, imagery, video) is laid flat onto

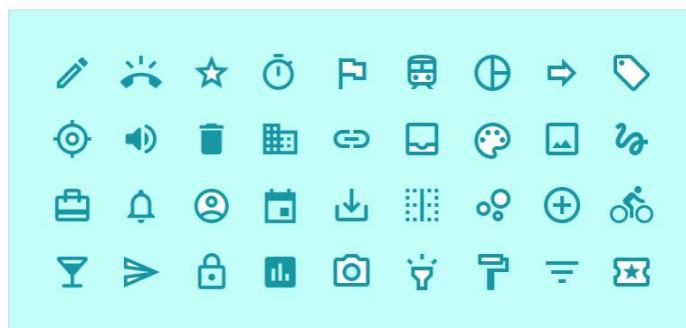
the material. If you use the Google material design tools you will find a lot of icons, buttons, text box, the navbar. You can make the app.



<https://phandroid.com/2014/10/29/google-drive-docs-slides-and-sheets-all-updated-with-material-design/>

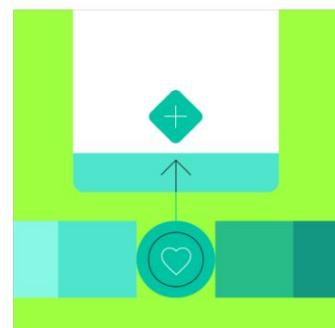
### Material articles

Make progress faster, with these helpful articles



System icons

System icons symbolize common actions, files, devices, and directories. Each icon is reduced to its minimal form, expressing essential characteristics.



Generate custom color palettes

Craft a unique color scheme for your brand with this online tool.

<https://material.io/design/>

How you can access the Material Lite CSS.

You can include the code all of the HTML page or you can download and host yourself.

- HOSTED

Just add the following `<link>` and `<script>` elements into your HTML pages (27kB gzipped):

```
<link rel="stylesheet" href="https://fonts.googleapis.com/icon?family=Material+Icons">
<link rel="stylesheet" href="https://code.getmdl.io/1.3.0/material.indigo-pink.min.css">
<script defer src="https://code.getmdl.io/1.3.0/material.min.js"></script>
```

- DOWNLOAD

- BUILD
- BOWER
- NPM

### 3.3 Part 3: Functional & Data Design

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 Name	Contact number	E-mail
First Name	Family name	mobile

**Vehicle**

Type	Engine Size	Fuel	Body Type
Spass	1359 CC	Petrol	5 door Hatchback
Motorcycle	?	Diesel	?
Ci	?	Hybrid	?
Car	?	Electric	?
Colour	Vehicle licence Number	Date	Car Brand
grey	05-D-12345	2001	Opel
7	131-W-12345	2049	tata min
			40 Brand

**Customer Component**

Staff number	Staff name
2	
3	
4	

**Date / Time**

Day	Time
Monday	8-9 62
Tuesday	10-10 72
Wednesday	10-11 84
Friday	11-12 9
Saturday	13-14 16

**Part / ITEM**

Part number	type	cost
1	Brake shoe	
2		
!		40
40		

**Note:** invisible status

This is the basically plan in the notebook.

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

Ger's Garage UNF																							
Database																							
Booking code number	Customer Code	Customer Password	Customer Name	Customer number	Customer mobile	Customer e-mail	Customer Bday	Comments	Vehicle type	Vehicle Made	Vehicle year	Vehicle licence details	Vehicle engine type	Booking Required	Booking Cost	Items / Parts	Item Cost	Date / Day	Time	Number of day	Date Month	Year	Staffs
								1. motorbikes 2. cars 3. small vans 4. small buses					1.diesel 2.petrol 3.hybrid 4.electric	1. Annual Service 2. Major Service 3. Repair / Fault 4. Major Repair					1. Monday 2.Tuesday 3.Wednesday 4.Thursday 5.Friday 6.Saturday 7.Sunday		1, 2, 3, 4,		
1	123	apple	Richard A. Addis	843-371-9785	RichardA.Addis@gergarage.com	02/27/89	etc	2	Ford Fiesta	2014	14D2345	1	2	200	12	23	2	10	22	3	2019	3	
2	222	citron	Joe M. Rosales	900-867-4863	JoeM.Rosales@gergarage.com	02/27/89	etc	2	skoda fabia	2019	19ID2343	4	5	300	10	100	3	11	21	3	2019	2	
3	123	apple	Richard A. Addis	843-371-9785	RichardA.Addis@gergarage.com	02/27/89	etc	2	BMW	2014	182D4345	3	1	190	33	150	1	10	22	3	2019	4	
4	322	manta	Shawn C. Anderson	570-592-3221	Shawn.C.Anderson@gergarage.com	05/10/57	etc	12	yamaha sp6	2016	16W1234 17E467	22	32	123 500	22 44 55	123 32 56	5 2	12 1	12 12	3	2019	31	

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

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

Ger's Garage 1NF																							
Database		First normal form (1NF) is a property of a relation in a relational database. A relation is in first normal form if and only if the domain of each attribute contains only atomic (indivisible) values, and the value of each attribute contains only a single value from that domain.																					
Booking code number	Customer Code	Customer Password	Customer Name	Customer mobile number	Customer e-mail	Customer Bay	Customer Comments	Vehicle type	Vehicle Made	Vehicle year	Vehicle licence details	Vehicle engine type	Booking Required	Booking Cost	Items / Parts	Item Cost	Date / Day	Time	Number of day	Date Month	Year	Staffs	Vehicle statuses
1	123	apple	Richard A. Addis	843-371-9785	RichardA.Addis@gersgarage.com	02/27/89	etc	2	Ford Fiesta	2014	14D2345	1	2	200	12	23	2	10	22	3	2019	3	2
2	222	citron	Joe M. Rosales	909-867-4863	JoeM.Rosales@gersgarage.com	02/27/89	etc	2	skoda fabia	2019	191D2343	4	5	300	10	100	3	11	21	3	2019	2	1
3	123	apple	Richard A. Addis	843-371-9785	RichardA.Addis@gersgarage.com	02/27/89	etc	2	BMW	2014	182D4345	3	1	190	33	150	1	10	22	3	2019	4	4
4	322	malina	Shawn C. Anderson	570-592-5221	Shawn.C.Anderson@gersgarage.com	05/10/57	etc	1	yahama	2016	16W1234	2	3	123	2255	123	5	12	12	3	2019	31	3
4	322	malina	Shawn C. Anderson	570-592-5221	Shawn.C.Anderson@gersgarage.com	05/10/57	etc	2	opel	2107	17E467	2	2	500	55	56	2	1	12	3	2019	31	5

## 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>

Ger's Garage 2NF																								
Database		Second normal form (2NF) is a normal form used in database normalization. 2NF was originally defined by E.F. Codd in 1971. A relation is in the second normal form if it fulfills the following two requirements: (1.) It is in first normal form (1NF)																						
Booking code number	Customer Code	Customer Password	Customer Name	Customer mobile number	Customer e-mail	Customer Bay	Customer Comments	Vehicle type	Vehicle Made	Vehicle year	Vehicle licence details	Vehicle engine type	Booking Required	Booking Cost	Items / Parts	Item Cost	Date / Day	Time	Number of day	Date Month	Year	Staffs	Vehicle statuses	Booking code number
1	123	apple	Richard A. Addis	843-371-9785	RichardA.Addis@gersgarage.com	02/27/89	etc	2	Ford Fiesta	2014	14D2345	1	2	200	12	23	2	10	22	3	2019	3	2	
2	222	citron	Joe M. Rosales	909-867-4863	JoeM.Rosales@gersgarage.com	02/27/89	etc	2	skoda fabia	2019	191D2343	4	5	300	10	100	3	11	21	3	2019	2	1	
3	123	apple	Richard A. Addis	843-371-9785	RichardA.Addis@gersgarage.com	02/27/89	etc	2	BMW	2014	182D4345	3	1	190	33	150	1	10	22	3	2019	4	4	
4	322	malina	Shawn C. Anderson	570-592-5221	Shawn.C.Anderson@gersgarage.com	05/10/57	etc	1	yahama	2016	16W1234	2	3	123	2255	123	5	12	12	3	2019	31	3	
4	322	malina	Shawn C. Anderson	570-592-5221	Shawn.C.Anderson@gersgarage.com	05/10/57	etc	2	opel	2107	17E467	2	2	500	55	56	2	1	12	3	2019	31	5	

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

2NF 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	RichardAAddis@j0uprade.com	02/27/89	etc											
222	citrom	Joe M. Rosales	909-867-4863	JoeMRosales@journpide.com	02/27/89	etc											
322	maina	Shawn C. Anderson	570-592-5221	ShawnCAnderson@armyspy.com	05/10/57	etc	2 NF B										
2 NF B																	
Customer Code	Vehicle type	Vehicle Made	Vehicle year	Vehicle engine type	Booking Required	Booking Cost	Items / Parts	Item Cost	Date / Day	Time	Number of day	Date Month	Year	Staffs	Vehicle statuses	Booking code number	
123	2	Ford Fiesta	2014	14D2345	1	2	200	12	23	2	10	22	3	2019	3	2	1
222	2	skoda fabia	2019	191D2343	4	5	300	10	100	3	11	21	3	2019	2	1	2
123	2	BMW	2014	182D4345	3	1	190	33	150	1	10	22	3	2019	4	4	3
322	1	yamaha	2016	16W1234	2	3	123	22	123	5	12	12	3	2019	31	3	4
322	2	opel	2107	17E467	2	2	500	55	56	2	1	12	3	2019	31	5	4

### Second Normal Form (2NF)

For a table to be in the Second Normal Form,

*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	RichardAAddis@j0uprade.com	02/27/89	etc							
222	citrom	Joe M. Rosales	909-867-4863	JoeMRosales@journpide.com	02/27/89	etc							
322	maina	Shawn C. Anderson	570-592-5221	ShawnCAnderson@armyspy.com	05/10/57	etc							
3NF C													
Vehicle licence details	Vehicle type	Vehicle Made	Vehicle year	Vehicle engine type	Customer Code								
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
16W1234	1	yamaha	2016	2	322	5	12	12	3	2019	31	3	4
17E467	2	opel	2107	2	322	2	1	12	3	2019	31	5	4
3NF D													
Items / Parts	Item Cost	3NF E					3NF F						
Vehicle licence details	Date / Day	Time	Number of day	Date Month	Year	Staffs	Vehicle statuses	Booking code number					
14D2345	2	10	22	3	2019	3	2	100					
191D2343	3	11	21	3	2019	2	1	200					
182D4345	1	10	22	3	2019	4	4	300					
16W1234	5	12	12	3	2019	31	3	400					
17E467	2	1	12	3	2019	31	5	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.*

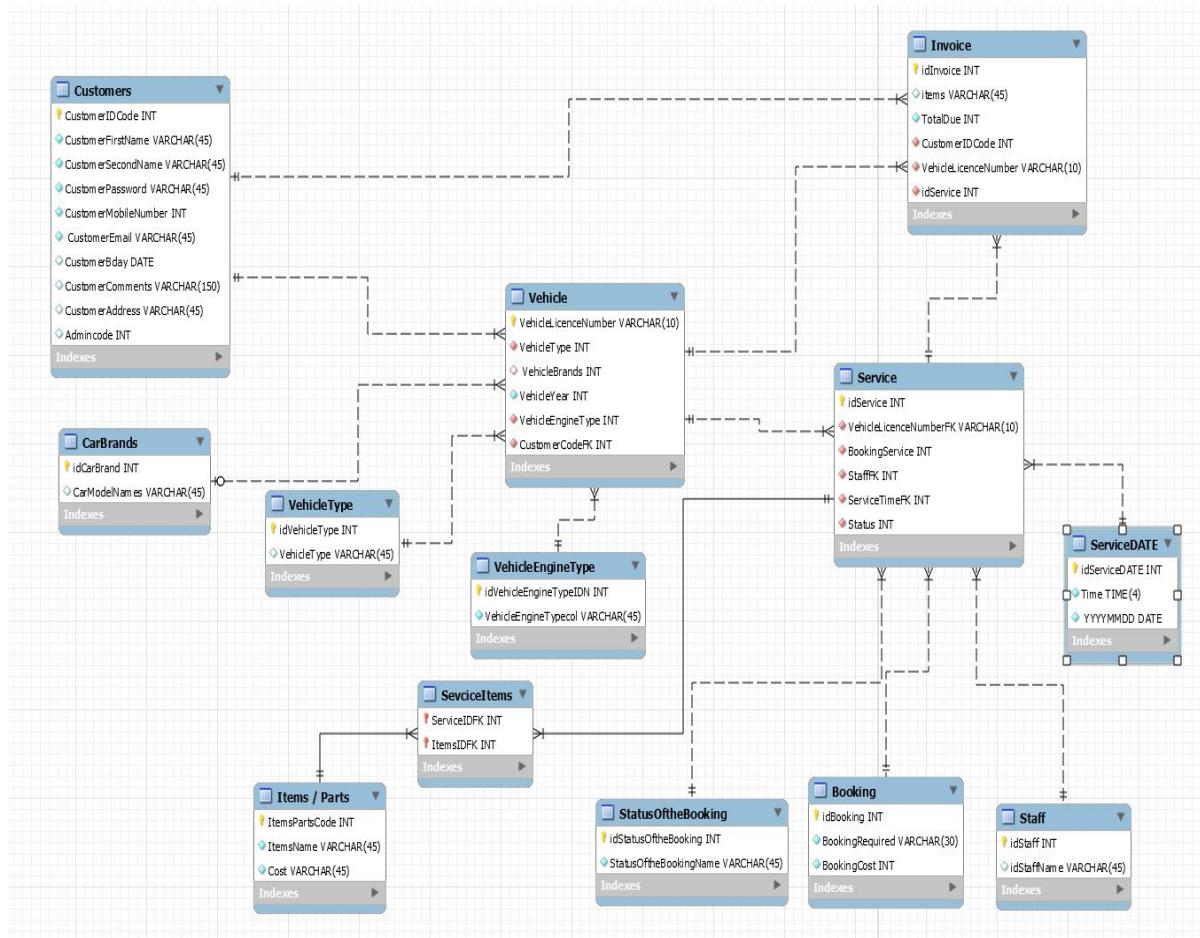
<https://www.studytonight.com/dbms/database-normalization.php>

I had to make ER diagram( Entity Relationship Diagram) to see how is look like my database.  
Why we should use ER?

Entity-relationship diagram (ERD) shows the relationships of entity sets stored in a database. ER Diagrams are most often used to design or debug relational databases in the areas of software engineering, business information systems, education, and research. ER diagrams also are often used in connection with data flow diagrams (DFDs), which map out the flow of information for processes or systems.

I have used MySQL workbench to create my ER diagram. This software is really easy to use and very visible of the table connection.

The design shows the primary keys and foreign keys. I had a little issue when I have added data to my database because when I created the DB system I could not transfer all of the data.



## 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.

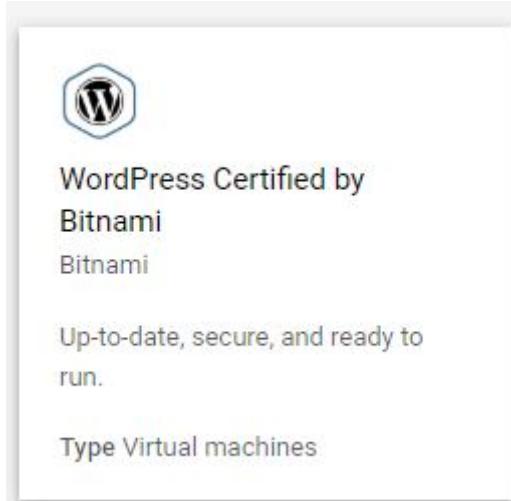
Step by step Setting up a WordPress site on Google Cloud.

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.

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

[←](#) New WordPress Certified by Bitnami deployment

Deployment name

Zone

Machine type  1.7 GB memory

Boot Disk

Boot disk type

Boot disk size in GB

Networking

Network

Subnetwork

External IP

Firewall

Add tags and firewall rules to allow specific network traffic from the Internet

Allow HTTP traffic

Source IP ranges for HTTP traffic

Allow HTTPS traffic

Source IP ranges for HTTPS traffic

 **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)

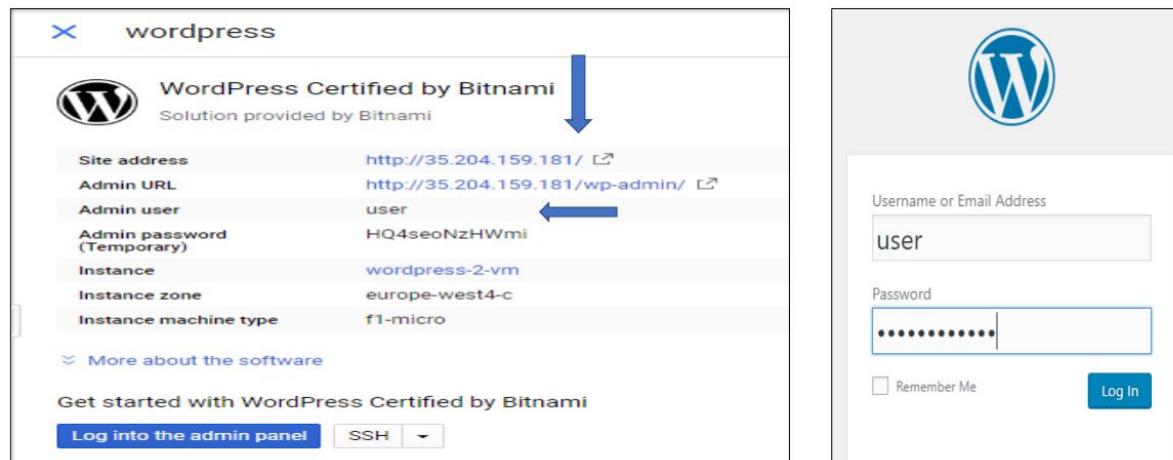
**Documentation**

- [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 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



**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

Admin password (Temporary)

Instance

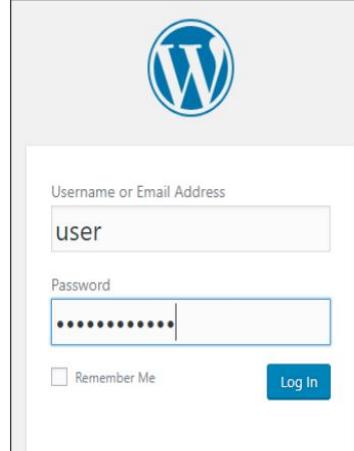
Instance zone

Instance machine type

[More about the software](#)

[Get started with WordPress Certified by Bitnami](#)

[Log into the admin panel](#)



Create easy website.

I did not use the WordPress I have found a better solution to my program. I have chosen the Firebase because It offers better support and documentation to my project. It is open up new opportunity to develop better website plus in the future I can make native android and ios app.

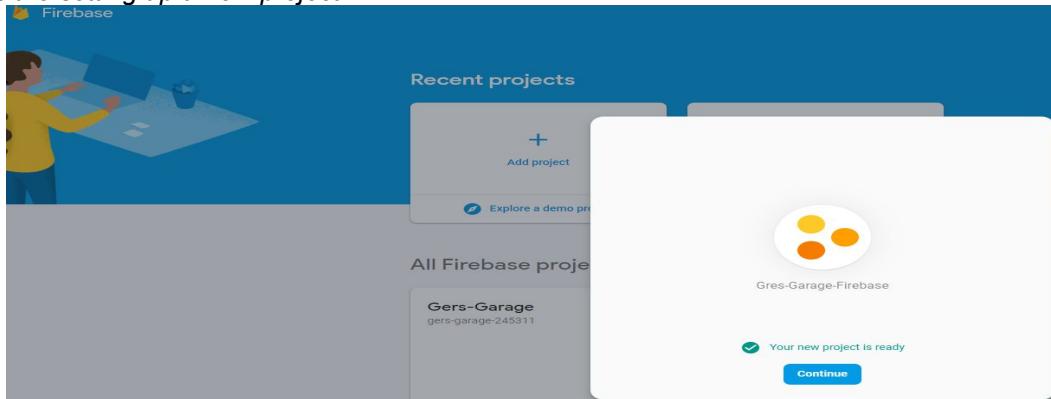
## 2. Firebase:

*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 Virtual 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.firebaseio.js"></script>

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

<!-- 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.*

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

The screenshot shows the VS Code interface with the following details:

- SIDE BAR:** Shows files in the project: public, .firebaserc, .gitignore, and firebase.json.
- TOP BAR:** Shows file operations: New file, Open folder..., Add workspace folder....
- RECENT TAB:** Shows recent items: PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL (selected).
- TERMINAL:** Displays the following text:

```
Waiting for authentication...
+ Success! Logged in as barczabalazs@gmail.com
PS C:\Users\barcz\Documents\CCT_FinalProject\Ger-s_Garage\WebApp -3> firebase init hosting

#####
# #   # #   # #   # #   # #   # #   # #
#####   # # #####   #####   #####   #####   #####
# #   # #   # #   # #   # #   # #   # #
# #   #####   #####   #####   #####   #####   #####
# #   # #   # #   # #   # #   # #   # #
```

You're about to initialize a Firebase project in this directory:  
C:\Users\barcz\Documents\CCT\_FinalProject\Ger-s\_Garage\WebApp -3  
? Are you ready to proceed? Yes

==== Project Setup

First, let's associate this project directory with a Firebase project.  
You can create multiple project aliases by running `firebase use --add`,  
but for now we'll just set up a default project.

? Select a default Firebase project for this directory: gres-garage-firebase (Gres-Garage-Firebase)  
i Using project gres-garage-firebase (Gres-Garage-Firebase)

==== Hosting Setup

Your public directory is the folder (relative to your project directory) that  
will contain Hosting assets to be uploaded with `firebase deploy`. If you  
have a build process for your assets, use your build's output directory.

? What do you want to use as your public directory? public  
? Configure as a single-page app (rewrite all urls to /index.html)? Yes  
+ Wrote public/index.html

i Writing configuration info to `firebase.json`...  
i Writing project information to `.firebaserc`...  
i Writing gitignore file to `.gitignore`...

+ Firebase initialization complete!

*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>*

```
PS C:\Users\barcz\Documents\CCT_FinalProject\Ger-s_Garage\WebApp -3> firebase serve  
--- Serving from 'C:\Users\barcz\Documents\CCT_FinalProject\Ger-s_Garage\WebApp -3'...  
i  hosting: Serving hosting files from: public  
+ hosting: Local server: http://localhost:5000
```

The screenshot shows a landing page for Firebase Hosting setup. At the top, the word "Welcome" is displayed in orange. Below it, the text "Firebase Hosting Setup Complete" is centered in large, bold black font. Underneath this, there is a message: "You're seeing this because you've successfully setup Firebase Hosting. Now it's time to go build something extraordinary!". At the bottom of the page is a large blue button with white text that reads "OPEN HOSTING DOCUMENTATION".

`firebase serve` <http://localhost:5000/>

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

`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.firebaseio.google.com/project/gres-garage-firebase/overview>

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

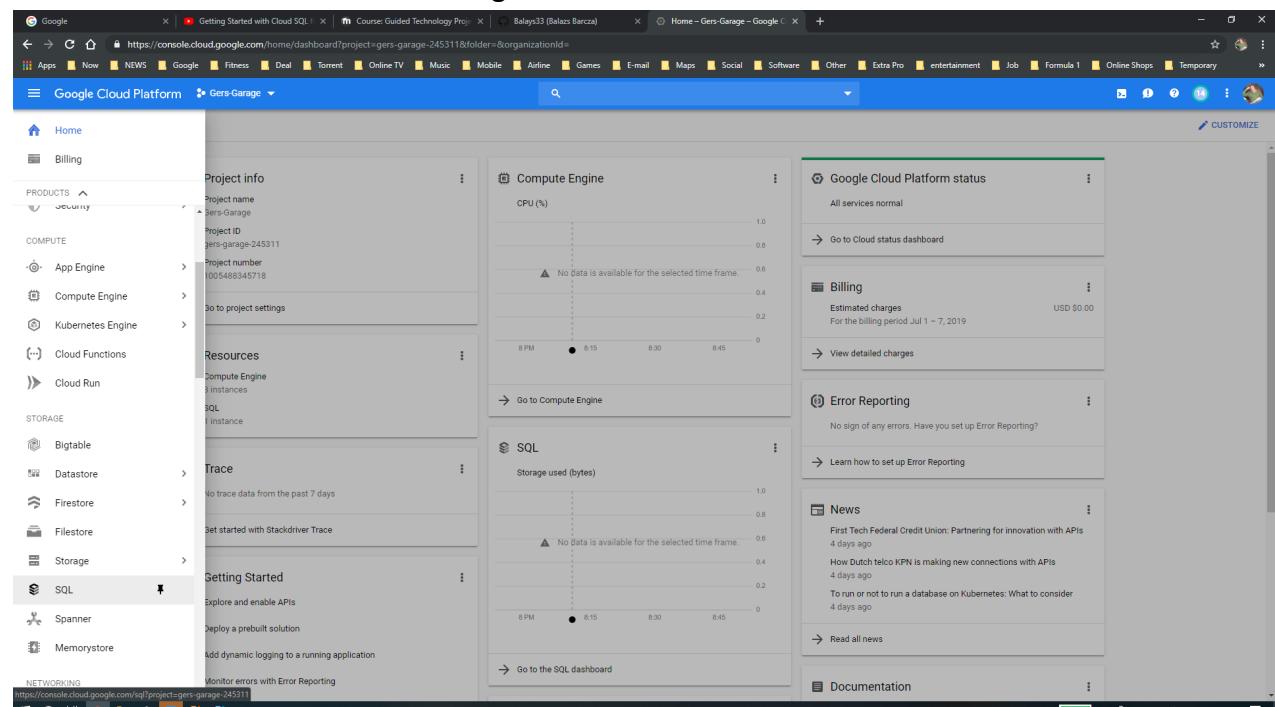
PS C:\Users\barcz\Documents\CCT\_FinalProject\Ger-s\_Garage\WebApp -3>

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

This is the URL where I can find the website.

### 3. Step by step setting up cloud SQL:

#### 1. Create SQL instance with Google Cloud



The screenshot shows the Google Cloud Platform dashboard for the project 'Gers-Garage'. The left sidebar navigation bar has a 'SQL' icon highlighted. The main content area displays the 'Compute Engine' and 'SQL' sections. The 'Compute Engine' section shows CPU usage over time, with a note that no data is available for the selected time frame. The 'SQL' section shows storage used in bytes over time, also noting no data is available. Other sections visible include 'Google Cloud Platform status', 'Billing', 'Error Reporting', 'News', and 'Documentation'.

Google Cloud navigation bar SQL icon

The Google Cloud Platform interface allows users to choose different applications, such as virtual machines or storage, etc. This interface is easy to use and has a clear design. I like it because I get a larger working place and if I want to use other services, I can find the navigation bar always. The platform is easy to set up and personal.

The screenshot shows the Google Cloud Platform SQL interface. At the top, there is a blue header bar with the text "Google Cloud Platform" and "Gers-Garage". Below the header, there is a navigation bar with "SQL" and a link to "Create an instance". The main area is titled "Choose your database engine" and contains two options: "MySQL" (Versions: 5.6, 5.7) and "PostgreSQL" (Versions: 9.6, 11). Each option has a "Choose MySQL" or "Choose PostgreSQL" button respectively. A small note at the bottom says "For First Generation MySQL instances, click here".

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.

The screenshot shows the "Create a MySQL Second Generation instance" form. It includes fields for "Instance ID" (ger-garage2), "Root password" (linuxbb33), "Location" (Region: europe-west4, Zone: Any), "Database version" (MySQL 5.7), and "Configuration options". The configuration options are expanded to show:

- Set connectivity**: Public IP enabled
- Configure machine type and storage**: Machine type is db-n1-standard-1, Storage type is SSD, Storage size is 10 GB, and will automatically scale as needed.
- Enable auto backups and high availability**: Automatic backups enabled, Binary logging enabled, Not highly available.
- Add database flags**: No flags set
- Set maintenance schedule**: Updates may occur any day of the week, Cloud SQL chooses the maintenance timing.
- Add labels**

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 and the Google Cloud will choose the best fit for 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 automatically increases.

Other important settings: We can set up when we 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](#)

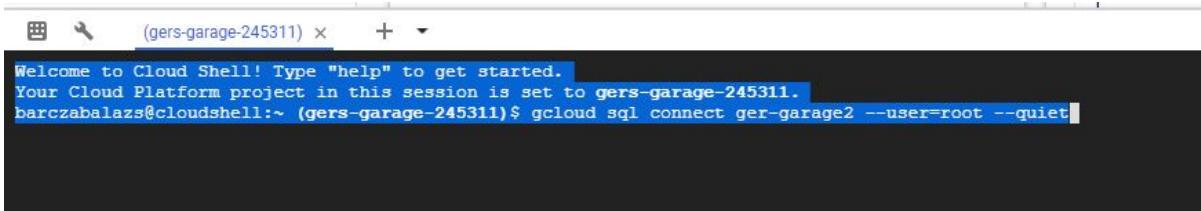
Enter an ID for your failover replica. ID is permanent. Use lowercase letters, numbers and hyphens. Start with a letter.

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

The screenshot shows the 'Instance details' page for a MySQL instance named 'database-gere-garage'. The instance is a 'MySQL Second Generation master'. A note indicates it is being updated. The 'OVERVIEW' tab is selected, showing CPU utilization (no data for the current time interval) and configuration details (1 vCPU, 3.75 GB memory, 10 GB SSD storage). The 'OPERATIONS' tab displays a log of recent updates.

Date/Time	Type	Status
Jul 2, 2019, 5:32:57 PM	Update	Update finished
Jul 2, 2019, 5:30:57 PM	Update	Update finished
Jul 2, 2019, 5:26:46 PM	Update	Update finished
Jul 2, 2019, 5:23:39 PM	Update	Update finished
Jul 1, 2019, 7:28:47 AM	Update	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, Computer 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.



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**.  
barczabalazs@cloudshell:~ (gers-garage-245311)\$ gcloud sql connect ger-garage2 --user=root --quiet^C

"

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

```
MySQL [test]> Create table test.customers( contact_id INT NOT NULL AUTO_INCREMENT, first_name VARCHAR(25), last_name VARCHAR(30) NOT NULL, birthday DATE, har(200), customer_password int(7), primary key (contact_id) );  
Query OK, 0 rows affected (0.04 sec)  
MySQL [test]>
```

Create table in GCP shell.

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

```
MySQL [test]> INSERT INTO test.customers ( first_name, last_name, birthday, Mobile_Number, customer_Email, customer_Comment, customer_Password )
->   VALUES
->   ( 'Justin', 'Miller', '2008-7-04', 936-447-8517, 'JustinMMiller@dayrep.com', 'Just For Feet', 66174090 );
Query OK, 1 row affected (0.02 sec)

MySQL [test]> █
```

### Insert data in to 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;

The google cloud database is connected with the MySQL workbench.

I can modify the data in the local machine with workbench or I can manage the database with the GCP shell console.

The google cloud database is connected with the MySQL workbench.

I can modify the data in the local machine with workbench or I can manage the database with the GCP shell consol.

In my project, I have used Google other ecosystems to store my data. It is a Firestore (firebase). This data can be modified on the firebase website or the GCP website.

### 4. Excel data add to Firebase.

I am building a service vehicle shop and the key part of the project the website should have an online store.

The customer can check the online store and can find different items with price. I have found a list from the online and I have download the excel sheet.

<https://www.autodoc.co.uk/>

I have to implement to my website. I have to use the Firebase cloud service to make access to the user. I want to upload the CSV file to the cloud. I have a few optional I can use on the firebase storage but and I can let to the user to download the price list or I can store the Firebase database. I firebase not support the excel sheet so I have to convert to JSON or XML. I used the <https://codebeautify.org> website to change the excel to JSON or XML.

#### STEPS TO UPLOAD PRICE LIST IN FIREBASE:

1. download items list
2. convert it into csv.
3. convert CSV into JSON from (<https://codebeautify.org/csv-to-xml-json#>).
4. validate JSON from <https://jsonlint.com/>
5. save in a file say test.json.

6. open firebase with login.
7. select the database.
8. click on (...) in verticle, beside + - sign,
9. click on import.
10. browse and select pricelist.json file
11. import

convert CSV into JSON from (<https://codebeautify.org/csv-to-xml-json#copy>).

The screenshot shows the Code Beautify interface. On the left, there is a 'CSV Input' section containing a list of items with numbers, descriptions, and prices. In the center, there are buttons for 'Load Url', 'Browse', 'CSV to XML', and 'CSV to JSON'. Below these buttons is a small advertisement for Malsoft. On the right, the 'Result : CSV TO JSON' section displays the converted JSON code. The JSON output is as follows:

```

1+ [
2+   {
3+     "number": "1",
4+     "Items": "A/C CLUTCH",
5+     "price": "16.99 EUR"
6+   },
7+   {
8+     "number": "2",
9+     "Items": "A/C SINGLE LINE",
10+    "price": "16.99 EUR"
11+   },
12+   {
13+     "number": "3",
14+     "Items": "A/C DUAL+ LINE",
15+     "price": "15.99 EUR"
16+   },
17+   {
18+     "number": "4",
19+     "Items": "A/C COMPRESSOR",
20+     "price": "30.99 EUR"
21+   },
22+   {
23+     "number": "5",
24+     "Items": "A/C CONDENSER",
25+     "price": "29.99 EUR"
26+   },
27+   {
28+     "number": "6",
29+     "Items": "A/C EVAPORATOR",
30+     "price": "25.99 EUR"
31+   },
32+   {
33+     "number": "7",
34+     "Items": "A/C FILTER-ENGINE",
35+     "price": "11.99 EUR"
36+   },
37+   {
38+     "number": "8",
39+     "Items": "A/C FILTER-PIPER",
40+     "price": "11.99 EUR"
41+   },
42+   {
43+     "number": "9",
44+     "Items": "A/C FILTER-PIPER",
45+     "price": "11.99 EUR"
46+   },
47+   {
48+     "number": "10",
49+     "Items": "A/C FILTER-PIPER",
50+     "price": "11.99 EUR"
51+   },
52+   {
53+     "number": "11",
54+     "Items": "A/C FILTER-PIPER",
55+     "price": "11.99 EUR"
56+   },
57+   {
58+     "number": "12",
59+     "Items": "A/C FILTER-PIPER",
60+     "price": "11.99 EUR"
61+   },
62+   {
63+     "number": "13",
64+     "Items": "A/C FILTER-PIPER",
65+     "price": "11.99 EUR"
66+   },
67+   {
68+     "number": "14",
69+     "Items": "A/C FILTER-PIPER",
70+     "price": "11.99 EUR"
71+   },
72+   {
73+     "number": "15",
74+     "Items": "A/C FILTER-PIPER",
75+     "price": "11.99 EUR"
76+   },
77+   {
78+     "number": "16",
79+     "Items": "A/C FILTER-PIPER",
80+     "price": "11.99 EUR"
81+   },
82+   {
83+     "number": "17",
84+     "Items": "A/C FILTER-PIPER",
85+     "price": "11.99 EUR"
86+   },
87+   {
88+     "number": "18",
89+     "Items": "A/C FILTER-PIPER",
90+     "price": "11.99 EUR"
91+   },
92+   {
93+     "number": "19",
94+     "Items": "A/C FILTER-PIPER",
95+     "price": "11.99 EUR"
96+   },
97+   {
98+     "number": "20",
99+     "Items": "A/C FILTER-PIPER",
100+    "price": "11.99 EUR"
101+   },
102+   {
103+     "number": "21",
104+     "Items": "A/C FILTER-PIPER",
105+     "price": "11.99 EUR"
106+   },
107+   {
108+     "number": "22",
109+     "Items": "A/C FILTER-PIPER",
110+     "price": "11.99 EUR"
111+   },
112+   {
113+     "number": "23",
114+     "Items": "A/C FILTER-PIPER",
115+     "price": "11.99 EUR"
116+   },
117+   {
118+     "number": "24",
119+     "Items": "A/C FILTER-PIPER",
120+     "price": "11.99 EUR"
121+   },
122+   {
123+     "number": "25",
124+     "Items": "A/C FILTER-PIPER",
125+     "price": "11.99 EUR"
126+   },
127+   {
128+     "number": "26",
129+     "Items": "A/C FILTER-PIPER",
130+     "price": "11.99 EUR"
131+   },
132+   {
133+     "number": "27",
134+     "Items": "A/C FILTER-PIPER",
135+     "price": "11.99 EUR"
136+   },
137+   {
138+     "number": "28",
139+     "Items": "A/C FILTER-PIPER",
140+     "price": "11.99 EUR"
141+   },
142+   {
143+     "number": "29",
144+     "Items": "A/C FILTER-PIPER",
145+     "price": "11.99 EUR"
146+   },
147+   {
148+     "number": "30",
149+     "Items": "A/C FILTER-PIPER",
150+     "price": "11.99 EUR"
151+   },
152+   {
153+     "number": "31",
154+     "Items": "A/C FILTER-PIPER",
155+     "price": "11.99 EUR"
156+   },
157+   {
158+     "number": "32",
159+     "Items": "A/C FILTER-PIPER",
160+     "price": "11.99 EUR"
161+   },
162+   {
163+     "number": "33",
164+     "Items": "A/C FILTER-PIPER",
165+     "price": "11.99 EUR"
166+   },
167+   {
168+     "number": "34",
169+     "Items": "A/C FILTER-PIPER",
170+     "price": "11.99 EUR"
171+   }
172+ ]

```

<https://codebeautify.org/csv-to-xml-json#copy>

validate JSON from <https://jsonlint.com/>

The screenshot shows the JSONLint interface. At the top, there is a navigation bar with links like NEWS, Google, Fitness, Deal, Torrent, Online TV, Music, Mobile, Airline, Games, E-mail, Maps, Social, Software, Other, Extra Pro, entertainment, and Jobs. Below the navigation bar, there is a logo for Big Nerd Ranch and a message: 'Not happy with your current app or digital product? Let's discuss your needs.' A 'Try the New Pro' button and a 'More Developer Tools' link are also present. The main area contains the JSON code, which is identical to the one shown in the previous screenshot. At the bottom, there are buttons for 'Validate JSON' and 'Clear', and a 'Support JSONLint for \$2/Month' button.

<https://jsonlint.com/>

upload the firebase realtime database

The screenshot shows the Firebase Realtime Database interface. The database structure is as follows:

```

gres-garage.firebaseio
  +-- 0
    |   Items: "A/C CLUTCH"
    |   number: "1"
    |   price: "16.99 EUR"
  +-- 1
    |   Items: "A/C SINGLE LINE"
    |   number: "2"
    |   price: "10.99 EUR"
  +-- 2
    |   Items: "A/C DUAL+ LINE"
    |   number: "3"
    |   price: "15.99 EUR"
  +-- 3
    |   Items: "A/C COMPRESSOR"
    |   number: "4"
    |   price: "30.99 EUR"
  +-- 4
    |   Items: "A/C CONDENSER"
    |   number: "5"
    |   price: "80.99 EUR"
  
```

## 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.

The screenshot shows the Firebase Authentication settings page. The sidebar menu includes Project Overview, Authentication (selected), Database, Storage, Hosting, Functions, and ML Kit. The main content area is titled "Authentication" and shows the "Sign-in method" tab selected. The "Sign-in providers" section lists the following providers:

Provider	Status
Email/Password	Enabled
Phone	Disabled
Google	Disabled
Play Games	Disabled
Game Center	Beta
Facebook	Disabled
Twitter	Disabled
Github	Disabled
Yahoo	Disabled
Microsoft	Disabled
Anonymous	Disabled

This screenshot is showing the other sign-in methods.

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

## 2. Calendar:

The website should have a calendar to manage the booking. I have found a couple of optional. I could choose the google calendar or other pay optional. Google gives lots of support to use the API but I was not sure It is right to me If I have more time I would like to integrate the google calendar API. I have found a simple optional java base calendar. I got a Github repository.

<https://github.com/GramThanos/jsCalendar>

The screenshot shows the homepage of the jsCalendar project. At the top, there is a navigation bar with links for Home, Docs, Themes, Demos, and Github. Below the navigation, there is a large button labeled "Download version 1.4.4". To the right of the button is a calendar for August. The calendar is a grid with columns for Sunday through Saturday. The days of the month are numbered from 28 to 31, then 1 to 31 again. The 11th of August is highlighted with a blue circle. The days of the week are labeled S, M, T, W, T, F, S.

The screenshot shows the jsDelivr CDN page for loading jsCalendar. At the top, there are three tabs: Local, JsDelivr CDN, and Unpkg CDN. The JsDelivr CDN tab is selected. Below the tabs, there is a heading "Load jsCalendar using jsDelivr, an open source CDN!". Underneath the heading is a code block containing the HTML and JavaScript code needed to load the jsCalendar library via jsDelivr. The code includes meta tags, a title, and script tags for both the main calendar file and its CSS file. At the bottom of the code block, there is a link: "Full jsDelivr example with all the files and their integrity hashes [here](#)".

<https://gramthanos.github.io/jsCalendar/>

It was easy to implement the javascript code.

---

### 4.3 Part 3: Problems confronted

#### Admin Authentication

I had issues to set up the admin I found a lot of documentation about Authentication in the firebase website but I did not have time to implement this future to my system so I have chosen a simple and not the best secure way to create admin page. If I have more time I can update the website with the firebase SDK.

##### **Add the Firebase Admin SDK to Your Server**

The Admin SDK lets you interact with Firebase from privileged environments to perform actions like:

- Read and write Realtime Database data with full admin privileges.
- Programmatically send Firebase Cloud Messaging messages using a simple, alternative approach to the FCM server protocols.
- Generate and verify Firebase auth tokens.
- Access Google Cloud Platform resources like Cloud Storage buckets and Firestore databases associated with your Firebase projects.
- Create your own simplified admin console to do things like look up user data or change a user's email address for authentication.

Here is a feature matrix showing what Firebase features are supported in each language:

Feature	Node.js	Java	Python	Go	C#
Custom Token Minting	✓	✓	✓	✓	✓
ID Token Verification	✓	✓	✓	✓	✓
User Management	✓	✓	✓	✓	✓
Control Access With Custom Claims	✓	✓	✓	✓	✓
Refresh Token Revocation	✓	✓	✓	✓	✓
Import Users	✓	✓	✓	✓	✓
Session Cookie Management	✓	✓	✓	✓	✓
Generating Email Action Links	✓	✓	✓	✓	✓
Managing SAML/OIDC provider configurations	✓				
Realtime Database	✓	✓	✓	✓ *	
Firebase Cloud Messaging	✓	✓	✓	✓	✓
FCM Multicast	✓	✓	✓	✓	✓
Manage FCM Topic Subscriptions	✓	✓	✓	✓	✓
Cloud Storage	✓	✓	✓	✓	✓
Cloud Firestore	✓	✓	✓	✓	
Project Management	✓	✓	✓		

<https://firebase.google.com/docs/admin/setup>

I have created an admin user (admin@gmail.com password:123456)

I have used the javascript function to let the admin log in the admin page

JavaScript HTML DOM - Changing HTML

Changing the Value of an Attribute

To change the value of an HTML attribute, use this syntax:

`document.getElementById(id).attribute = new value`

*file.html*

```
<div class="Logged-in" style="display: none;">
<a id="adminlogin2" href="./nopermission.html">

<h6 style="color:blue;">Admin</h6>
</a>
</div>
```

*file.js*

```
function admininf(user){
console.log(user.email);
if (user.email == "admin@gmail.com"){
console.log("yes");
document.getElementById("adminlogin").href="./admin.html";
document.getElementById("adminlogin2").href="./admin.html";
}
}
```

## Fix CarQuery API

The documentation requires to choose a different car model. I did not want to create a new database and upload lots of data to the could so I have chosen an outside provider. I have found a website who have API to get lots of different car model with information about the car.

CarQuery API is an easy to use JSON based API for retrieving detailed car and truck information, including year, make, model, trim, and specifications. Using CarQuery API is as simple as including a javascript file, and inserting a few lines of script in your page. You can also write your own javascript or server-side API interactions to use the available data any way you like.

<http://www.carqueryapi.com> API

This website is free and gets a lot of data about the car. Unfortunately, it does not provide data about the bus or van or motorbike so I have to build extra placeholder to save other vehicle models.

I thought it is run well and I will integrate the API easily. I got all of the code and documentation from the website. When I have run the test on my local machine It was fine. I have deployed my website to the could and I have found out I have an issue with the page. I got an error message because of a security issue.

✖ Mixed Content: The page at [active-mixed-content.html:1](https://googlesamples.github.io/web-fundamentals/samples/discovery-and-distribution/avoid-mixed-content/active-mixed-content.html) was loaded over HTTPS, but requested an insecure script '<http://googlesamples.github.io/web-fundamentals/samples/discovery-and-distribution/avoid-mixed-content/simple-example.js>'. This request has been blocked; the content must be served over HTTPS.

# This is an error message

⚠ Mixed Content: The page at [passive-mixed-content.html:37](https://googlesamples.github.io/web-fundamentals/samples/discovery-and-distribution/avoid-mixed-content/passive-mixed-content.html) was loaded over HTTPS, but requested an insecure image '<http://googlesamples.github.io/web-fundamentals/samples/discovery-and-distribution/avoid-mixed-content/puppy-thumb.jpg>'. This content should also be served over HTTPS.

**Mixed content** occurs when initial HTML is loaded over a secure HTTPS connection, but other resources (such as images, videos, stylesheets, scripts) are loaded over an insecure HTTP connection. This is called mixed content because both HTTP and HTTPS content are being loaded to display the same page, and the initial request was secure over HTTPS. Modern browsers display warnings about this type of content to indicate to the user that this page contains insecure resources.

I had to fix because The user will have trouble to access data and book service. So I have to research on the web and I have found the website has to change to secure way HTTP to HTTPS and I have to implement code to my HTML file.

<http://www.carqueryapi.com> --> <https://www.carqueryapi.com> API

```
<meta http-equiv="Content-Security-Policy" content="upgrade-insecure-requests">
<!--http://www.carqueryapi.com API-->
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js"></script>
<script type="text/javascript" src="https://www.carqueryapi.com/js/jquery.min.js"></script>
<script type="text/javascript" src="https://www.carqueryapi.com/js/carquery.0.3.4.js"></script>
```

This is good no more mixed contact on my page.

This is link to the [Troubleshooting mixed content errors](#)

<https://support.cloudflare.com/hc/en-us/articles/200170476-Troubleshooting-mixed-content-errors>

## 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

The screenshot shows a web browser interface with a toolbar at the top. The 'LOG IN/SIGN UP' button is highlighted with a blue border. Below the toolbar, there is a message: 'Login to view Car Service Info'. In the bottom right corner of the browser window, the developer tools' 'Console' tab is open, displaying the message 'user logged out'.

user logged-out or first time visit the websites

The clients has to log in or register on the website to use all of the future.

The screenshot shows the Firebase Authentication page under the 'Authentication' section. It lists several users with their details:

Identifier	Providers	Created	Signed In	User UID
test@yahoo.com	✉	Aug 7, 2019	Aug 8, 2019	96WcUwWfjIxyYlqxZJMXwzXuBcB2
mark@yahoo.com	✉	Aug 11, 2019	Aug 11, 2019	9eyzV3RRznS23NJBZKVS5dp17x2
ger@admin.com	✉	Jul 13, 2019	Jul 13, 2019	Bc7G45j980ZMKUegBQUYEUAisty2
balazs@gmail.com	✉	Jul 15, 2019	Jul 15, 2019	D00Nh9AfJ6NMEdVQ94oPEilatem1
justyna@gmail.com	✉	Jul 28, 2019	Jul 28, 2019	EI3siid8cISJBBoowNspBN1Euwz1
kylenorth@rhyta.com	✉	Jul 13, 2019		Gqyxp8nak4QelNzkSHpIRXSlvV1
henrysmurrell@armyspy.com	✉	Jul 13, 2019		HL0j1lkYDVbCH7wjkgk2nrK9Qp2
david@yahoo.com	✉	Aug 10, 2019	Aug 11, 2019	LP5kY9eF6aSC015XRNrOc2WPvi12

This is the firebase-auth page. I can manage the users email here.

Delete or disable account or reset the password

The screenshot shows a website interface. At the top, there is a search bar with suggestions: 'test here', 'Balazs', 'David', 'find all', 'Mario', and 'realtime'. Below the search bar, there is a sidebar with links: 'BOOK A SERVICE', 'BUY CAR PARTS', and 'LOG OUT'. On the right side, there is a developer tools' 'Console' tab showing logs related to user authentication and network requests.

after the login. The clients can access the services. If the user is not admin can not visit the administration page.

The screenshot shows a form titled "Personal info" with fields for First and last name (Balazs Barcza), Phone number (851097776), Birthday (mm/dd/yyyy), Address (2 Lanesborough Square, Lanesborough Road, Finglas), and a Comment field. A "Submit" button is at the bottom. To the right, a log window shows the following entries:

```
> {value: "balazs.barcza@yahoo.com"}
balazs.barcza@yahoo.com
test here balazs.barcza@yahoo.com
send data up
Document successfully written!
```

Book services. The client updated his personal information.

## Admin Page

First Name Last Name Mobile number

First Name Last Name Mobile number

**Get Customer Data**

Service status Booked

Service mechanic Peter

**update status** **update mechanic** **Create “invoice”**

david@yahoo.com

Annual Service

Collected

Tue Aug 27 2019 00:00:00 GMT+0100 (British Summer Time)

15:00-16:00

David

balazs.barcza@yahoo.com

Major Service

Unrepairable / Scrapped

Tue Aug 20 2019 00:00:00 GMT+0100 (British Summer Time)

14:00-15:00

Paul

test@yahoo.com

Annual Service

Collected

Tue Aug 20 2019 00:00:00 GMT+0100 (British Summer Time)

10:00-11:00

noname

mark@yahoo.com

Annual Service

Collected

If the administration logged in can check the bookings. Ger can updatingthe services statuses and can add mechanic.

```
admin.js:63
▼ {CustomerAddress: "et33", CustomerBday: "1988-02-15", CustomerComments: "wetrwet", CustomerEmail: "mark@yahoo.com"
", CustomerFirstName: "Mark", ...} ⓘ
  CustomerAddress: "et33"
  CustomerBday: "1988-02-15"
  CustomerComments: "wetrwet"
  CustomerEmail: "mark@yahoo.com"
  CustomerFirstName: "Mark"
  CustomerMobileNumber: "34435"
  CustomerSecondName: "Kink"
▶ __proto__: Object
admin.js:62
getting data
admin.js:63
▼ {CustomerAddress: "2 Lanesborough Square, Lanesborough Road, Finglas", CustomerBday: "", CustomerComments: "", C
ustomerEmail: "test@yahoo.com", CustomerFirstName: "Balazs", ...} ⓘ
  CustomerAddress: "2 Lanesborough Square, Lanesborough Road, Finglas"
  CustomerBday: ""
  CustomerComments: ""
  CustomerEmail: "test@yahoo.com"
  CustomerFirstName: "Balazs"
  CustomerMobileNumber: "851097776"
  CustomerSecondName: "Barcza"
▶ __proto__: Object
```

Json from the cloud-based database

---

### 5.3 Part 3: Commercialisation / Marketing

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

Ger wants to get more business, the website has to improve the google search position and he has to buy Webhost service.

It has to get on the domain name as well.

## 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

The project is working but a lot of small bugs so has to be check and fix them and add more function to the website.

I feel I made good progress and I have learned a lot of new skills on the journey. Unfortunately, I did not have enough time to finish the project I had a lot of issue with the debugging. If the future I have to make a similar project I can use the knowledge from this project.

---

### 6.2 Part 2: Suggestions for further work

I have to make a deep test to be sure all of the data connection is working. I would like to connect my firebase website with the Google Cloud Platform. If I have to restart I will use strat the Firebase because I can get all of the main document to be successful.

---

## 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.

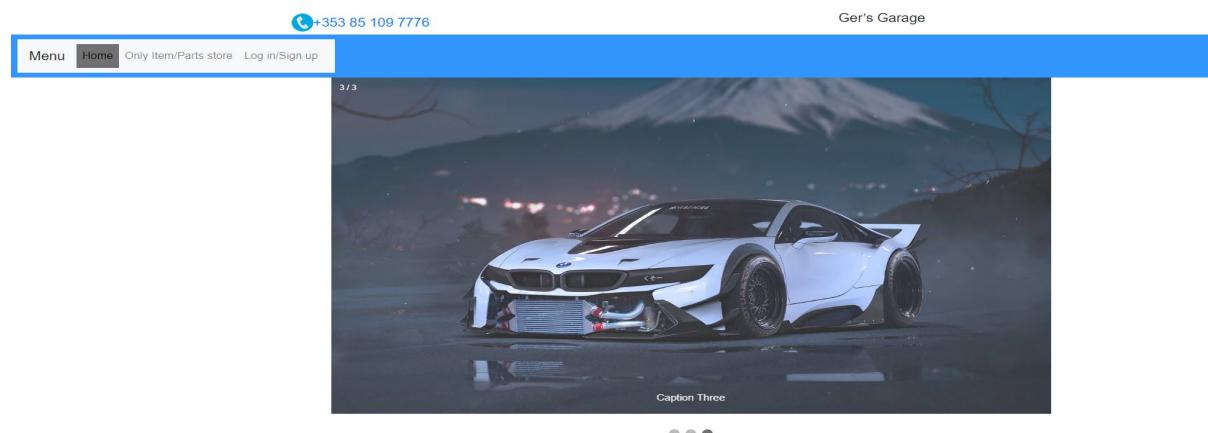
[https://github.com/Balays33/Ger-s\\_Garage](https://github.com/Balays33/Ger-s_Garage)

### 7.1 Part 1: Index page

Navibar : bootstrap navibar

```
<nav class="navbar navbar-light" style="background-color: #3396FF;">
  <!-- Navbar content -->
  <nav class="navbar navbar-expand-lg navbar-light bg-light">
    <a class="navbar-brand" href="#">+353 85 109 7776
    <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarNavAltMarkup"
           aria-controls="navbarNavAltMarkup" aria-expanded="false" aria-label="Toggle navigation">
      <span class="navbar-toggler-icon"></span>
    </button>
    <div class="collapse navbar-collapse" id="navbarNavAltMarkup">
      <div class="navbar-nav">
        <a class="nav-item nav-link active" href="#">Home <span class="sr-only">(current)</span></a>
        <!-- <a class="nav-item nav-link" href="#">Introduce</a> -->
        <a class="nav-item nav-link" href="#">Only Item/Parts store</a>
        <!-- <a class="nav-item nav-link" href="#">Service</a> -->
        <!-- <a class="nav-item nav-link" href="#">Contact Us</a> -->
        <a class="nav-item nav-link" href="#">Log in/Sign up</a>
      </div>
    </div>
  </nav>
</nav>
```

index page with navibar



```
this code sliding pictures script
<script>
var slideIndex = 0;
showSlides();

function showSlides() {
var i;
var slides = document.getElementsByClassName("mySlides");
var dots = document.getElementsByClassName("dot");
for (i = 0; i < slides.length; i++) {
slides[i].style.display = "none";
}
slideIndex++;
if (slideIndex > slides.length) {
slideIndex = 1
}
for (i = 0; i < dots.length; i++) {
dots[i].className = dots[i].className.replace(" active", "");
}
slides[slideIndex - 1].style.display = "block";
dots[slideIndex - 1].className += " active";
setTimeout(showSlides, 3000); // Change image every 2 seconds
}
</script>
```

---

## 7.2 Part 2: login.html + login.js

Implement of the Firebase , Jquery and bootstrap  
database implement

```
<!-- Optional JavaScript -->
<!-- jQuery first, then Popper.js, then Bootstrap JS -->
<script src="https://code.jquery.com/jquery-3.3.1.slim.min.js"
       integrity="sha384-q8i/X+965Dz0rT7abK41JStQIAqVgRVpbzo5smXKp4YfRvH+8abTE1Pi6jizo"
       crossorigin="anonymous"></script>
<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js"
       integrity="sha384-U2eT0CpHqdSJQ6hJty5KVphtPhzWj9W01clHTMGa3JDZwrnQq4sF86dIHNDz0W1"
       crossorigin="anonymous"></script>
<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js"
       integrity="sha384-JjSmVgyd0p3pXB1rRibZUAYoIIy6OrQ6VrjIEaFF/nJGzIxFDsf4x0xIM+B07jRM"
       crossorigin="anonymous"></script>

<script src="https://www.gstatic.com/firebasejs/5.6.0.firebaseio-app.js"></script>
<script src="https://www.gstatic.com/firebasejs/5.6.0/firebase-auth.js"></script>
<script src="https://www.gstatic.com/firebasejs/5.6.0/firebase-firebase.js"></script>
<script src="https://www.gstatic.com/firebasejs/3.1.0.firebaseio.js"></script>

<script>
  // Initialize Firebase
  var firebaseConfig = {
    apiKey: "AIzaSyA_MvxhxRgubmZXleweD6tFG1UCP89tjjs",
    authDomain: "gres-garage-firebase.firebaseio.com",
    databaseURL: "https://gres-garage-firebase.firebaseio.com",
    projectId: "gres-garage-firebase",
    storageBucket: "gres-garage-firebase.appspot.com",
    messagingSenderId: "979259985070",
    appId: "1:979259985070:web:acaece0ce407193e"
  };
  // Initialize Firebase
  firebase.initializeApp(firebaseConfig);

  // make auth and firestore references
  const auth = firebase.auth();
  const db = firebase.firestore();
```

firebase include api key and authentication database url and project hosting name

```
// what can you see on the navibar
const setupUI = (user) => {
  if (user) {
    adminf(user);
    // toggle user UI elements
    loggedInLinks.forEach(item => item.style.display = 'block');
    loggedOutLinks.forEach(item => item.style.display = 'none');
  } else {
    // toggle user elements
    loggedInLinks.forEach(item => item.style.display = 'none');
    loggedOutLinks.forEach(item => item.style.display = 'block');
  }
};
```

This code shows who can see the login page navibar. Who logged-in can see account page who did not login can not access to pages.

### 7.3 Part 3: Admin.html and Admin.js

```
db.collection('customers').get().then(snapshot => {  
  
snapshot.docs.forEach(doc => {  
console.log("getting data");  
console.log(doc.data());  
});  
});
```

This code access to the db(database) collection : cutomers and get all of the data print out to the console.

```
// real-time listener  
  
db.collection('servicetime').onSnapshot(snapshot => {  
let changes = snapshot.docChanges();  
changes.forEach(change => {  
console.log(change.doc.data());  
if (change.type == 'added') {  
renderservicetime(change.doc);  
} else if (change.type == 'removed') {  
let li = carList.querySelector('[data-id=' + change.doc.id + ']');  
carList.removeChild(li);  
}  
});  
});
```

This is the real-time listener If the admin modify the data, The code will update real-time the website.

```
function updatetestatus() {  
  
console.log('update status');  
var ServicestatusUpdate = document.getElementById("validationServer11").value;  
console.log(ServicestatusUpdate);  
var cutomerEmailG = document.getElementById("CustomerEmail").value;  
console.log(cutomerEmailG);  
  
db.collection('servicetime').get().then(snapshot => {  
snapshot.docs.forEach(doc => {  
console.log("get cutomer personal details");  
console.log(doc.data());  
console.log(doc.id);  
var userid = doc.id;
```

```
console.log(userid);
if (doc.data().CustomerEmail == cutomerEmailG) {
  console.log("-----");
  db.collection("servicetime").doc(doc.id).update({
    Status: ServicestatusUpdate,
  })
  .then(function () {
    console.log("Document successfully written!");
  })
  .catch(function (error) {
    console.error("Error writing document: ", error);
  });
}
});
```

The code updating the collection of the services statuses. The code get information from the administration and updating the database.

## *8 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.

---

<https://github.com/jdorffman/awesome-json-datasets>

<https://github.com/iamshaunjp/firebase-auth>

<https://github.com/GramThanos/jsCalendar>

<https://www.youtube.com/channel/UCW5YeuERMmlnqo4oq8vwUpg>

<https://www.youtube.com/user/Firebase>

<https://firebase.google.com/>

<https://getbootstrap.com/>

<https://getmdl.io/index.html>

<https://jsonformatter.curiousconcept.com/#>

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

<http://www.carqueryapi.com/>

<https://www.w3schools.com/>