FEASIBILITY REPORT

Canada Tourism -Project

SanjayKumar MuthuKumar – B00834742

Mayank Chaturvedi – B00839818

Anshdeep Singh – B00844351

Nihir Shah – B00824490

JANUARY 30, 2020

DALHOUSIE UNIVERSITY

Group - 9

Project Requirement

This project requires the development of a client mobile app and a website that allows information to search on provincial/national parks, major beaches, and important cities. The app should have a cloud-based server where all the validations must be performed. There must be optimum resource utilization and load balancing for the cloud instance. It should be designed using containerization and must be divided into multiple loosely based modules. The app should allow users to create an account and make payments using a dummy card for transactions. Database and other additional tools can be used for communication and analysis. The mobile app should only communicate with the server, submit user queries and display reports (for analytics). All the major functionality should be developed in the cloud server.

Android

Android is the world's most popular operating system which is being used by more than 2.5 billion active devices. There are many reasons for choosing Android as the desired platform for the mobile app. Apart from many users, it has a huge potential for innovative technology. Developers can experiment and try out new technologies easily. Android apps are also cheap to create which ultimately reduces the development cost.

Android apps are written in Java programming language which most developers are familiar with. They are also compatible with many devices which makes the app accessible to different categories of users. The installation of Android apps is also very easy making it a seamless experience for the users. Due to these reasons, creating an Android application is a wise and profitable option for DalCloud5409.

Spring Boot

Spring Boot is a framework that is used to create a stand-alone application that is based on Java and its open source [5]. The major advantages of using Spring Boot is easy to develop, it reduces development time and increases productivity [5]. It takes the pain from the developers to do configuration steps while setting up the project [5]. It follows the MVC architecture.

Spring Boot makes the integration of the database an easy task and it auto-configure Spring Data for database access [5]. We can achieve this by just adding the "spring-boot-starter-data-jpa" module in our project [5]. Spring also provides support for Amazon web services and containerization which makes the development of the website and its hosting easy [5]. It also provides support for authentication and authorization [5].

Spring also provides support for managing the user sessions [5]. It is very essential to manage user sessions while dealing with multiple requests from an application. Spring provides this support using Spring Security API [5].

Spring boot comes bundled with the apache tomcat server [5]. It helps you to run your application as a simple Java Application [5]. It makes the deployment process and gives the option to developers to generate a war file or jar file according to the requirement [5].

Maven (For Website) and Gradle (For Android Application)

A maven is a tool used in Java projects to build and manage it [6]. There is a POM file that is in an XML format that allows maintaining project configuration details which will be used by Maven. It helps us to download the external dependencies that our application will require while running [6].

A Gradle is a build tool which can be used with programming language like Java which provides support for running test case and creating documentation [7]. Similarly, like the POM file, Gradle has to build.gradle file. It helps you to specify the dependencies the same as maven [7].

MySQL

MySQL is a relational database mainly used to maintain web data [8]. It is based on Structured Query Language. We decide to choose MySQL because our data would be structured with clear schema (Tables and Fields) and we would be requiring the multi-row transactions while retrieving certain details like the payment information or while making a booking [8]. Though with large database performance tends to be slow with MySQL we are assuming our database is not that complex and might contain only a few tables with some thousands of records [8].

Firebase

SSL Certificates

In today's world, everybody is concerned about the information that is being transmitted by their interaction over the internet. This leads to the point that users want to visit the website that is secured and wants to make sure that the data that is being transmitted is completely encrypted and secured. So, here comes the SSL certificate into the picture. SSL Certificates protect your sensitive information such as credit card information, usernames, passwords, etc. It also keeps data secure between servers, increases your Google Rankings, builds/enhances customer trust, improves conversion rates.

SSL Certificates are a small data file that carefully ties a cryptographic key to an association's details. A valid SSL certificate provides authentication in addition to encryption. Due to the nature of the Internet, your customers frequently send information via several computers. Any of these machines will claim to be your website and try to send the user's personal information to them. This can only be avoided by a trusted SSL Provider with an SSL Certificate.

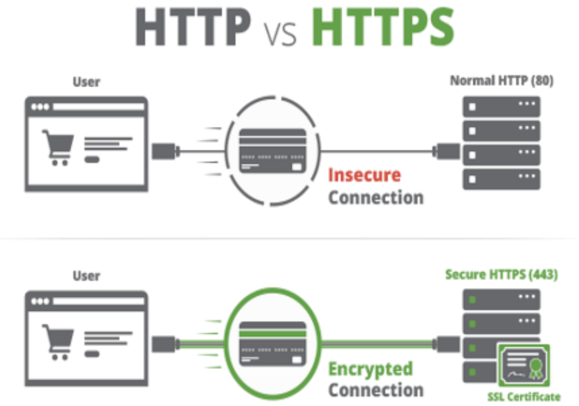


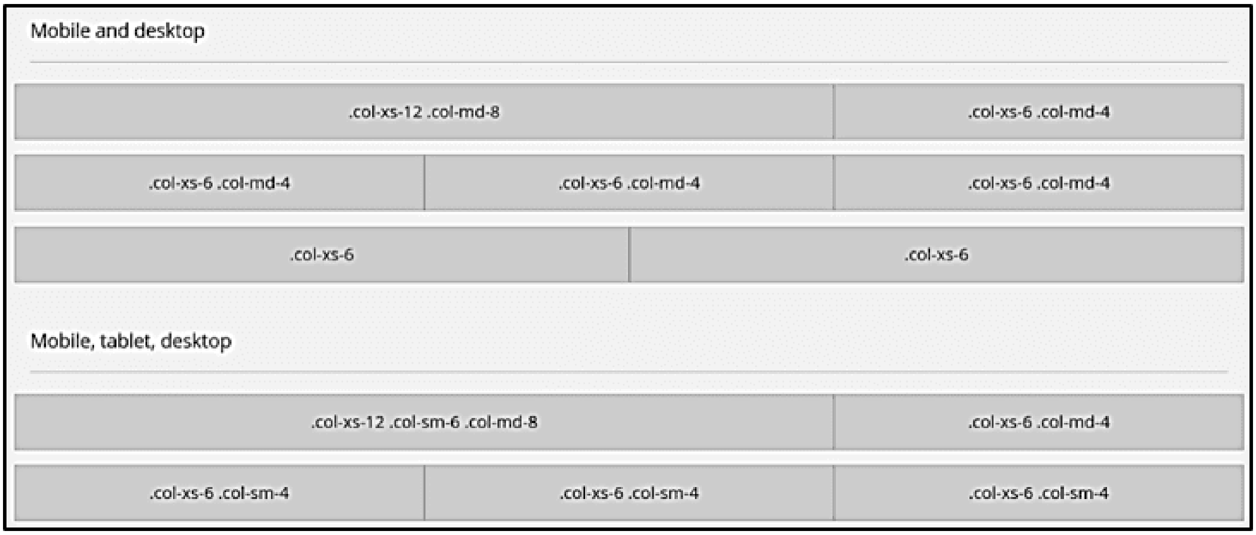
Figure 1 : HTTP vs HTTPS

SSL Certificates are a small data file that carefully ties a cryptographic key to an association's details. A valid SSL certificate provides authentication in addition to encryption. Due to the nature of the Internet, your customers frequently send information via several computers. Any of these machines will claim to be your website and try to send the user's personal information to them. This can only be avoided by a trusted SSL Provider with an SSL Certificate [1]. Figure 1 shows an overview of the difference between HTTP and HTTPS (secure) connection.

Responsive Design: HTML and CSS with BootStrap

The launch of several new web-enabled devices would change the way web systems are developed. There was and still has a distinct mobile user interface for web applications in the past. Responsive Development is a cloud-based approach that usually makes web pages on a variety of devices and screen sizes from mobile to desktop and optimizes browsing.

We use the grid structure of Bootstrap (Figure 2) to display content responsively for the Tourism Website. The monitor width of the unit is divided into 12 parts. As shown in, the content can take space from 1 and 12 parts (columns) of scale. The benefit of grouping the content in this facility is that both mobile and desktop users have the same user experience. From the viewpoint of the developer and the client, we must build a website for all users that decreases production time and development costs considerably [3].



Mobile and desktop

.col-xs-12 .col-md-8 | .col-xs-6 .col-md-4

.col-xs-6 .col-md-4 | .col-xs-6 .col-md-4 | .col-xs-6 .col-md-4

.col-xs-6 | .col-xs-6

Mobile, tablet, desktop

.col-xs-12 .col-sm-6 .col-md-8 | .col-xs-6 .col-md-4

.col-xs-6 .col-sm-4 | .col-xs-6 .col-sm-4 | .col-xs-6 .col-sm-4

Figure 2: Bootstrap Grid Layout

Containerization: Overview to Kubernetes

With the dawn of the Container Era, when every application deployed was containerized. It became important to have a system that manages the containerized workload. Since, if in a production system a container goes down, another one should be restarted. Automation of such a use case is important to maintain minimum downtime and hence high availability of services. So, to achieve high efficiency, availability of services we would be using Kubernetes which manages containerized workload and services. It facilitates both declarative functionalities and workload automation. It provided an infrastructure that allowed it to be more stateless and self-healing. Self-healing being its most important aspect [4].

Kubernetes components include a set of machines called nodes that forms a cluster. It has a master node and at least one worker node. While the master node manages the worker nodes and pods in the cluster. It is these worker nodes that host the post, the component of an application, which is eventually managed by Kubernetes.

Load Balancer

Most application servers are based on threads. Threads are light-weight processes that are creating within the same user memory. They are used to serve parallel requests from the client which increases the concurrency of the application. However, threads perform blocking IO operation. When a thread is processing a request, it cannot process another request while it is idle. Thus, when requests are overloaded, they are blocked from being served when all other threads are bound to other requests (even if they are idle). Thus, most application servers a threshold as to how many requests they can serve. To increase the capacity of the application, resources have to be scaled horizontally, depending upon the requirement. Requests can be shared among these servers using a Load Balancer.

The Load Balancer that we will use for the application is Nginx. Nginx is a popular reverse proxy software that performs Load Balancing between servers. Nginx provides several load balancing algorithms to choose from, i.e., round-robin, least connections, source hashing, etc. Nginx also performs SSL offloading where the encrypted secure connection is terminated by Nginx, and the proxied backend connection takes place in plain text. Nginx also provides security measures such as rate-limiting, Web Application Firewall, tuning configurations to block application-layer attacks, etc.

High availability of Load balancer is ensured by using 2 Nginx instances in parallel. Both these instances run in a master-master configuration, by handling requests in parallel. This architecture avoids a single point of failure of the load balancer. Requests between the 2 Nginx servers will be distributed using DNS load balancing. The DNS request for the domain name will return alternating answers containing the 2 instance’s IPs.

Analytics

Basic HTTP analytics such as live connections, live requests will be obtained using Nginx’s stub status module. These will be aggregated and displayed in a GUI based graph console. The console will provide transparency on the system at a glance. It shall facilitate business decisions on scaling the system.

Architecture

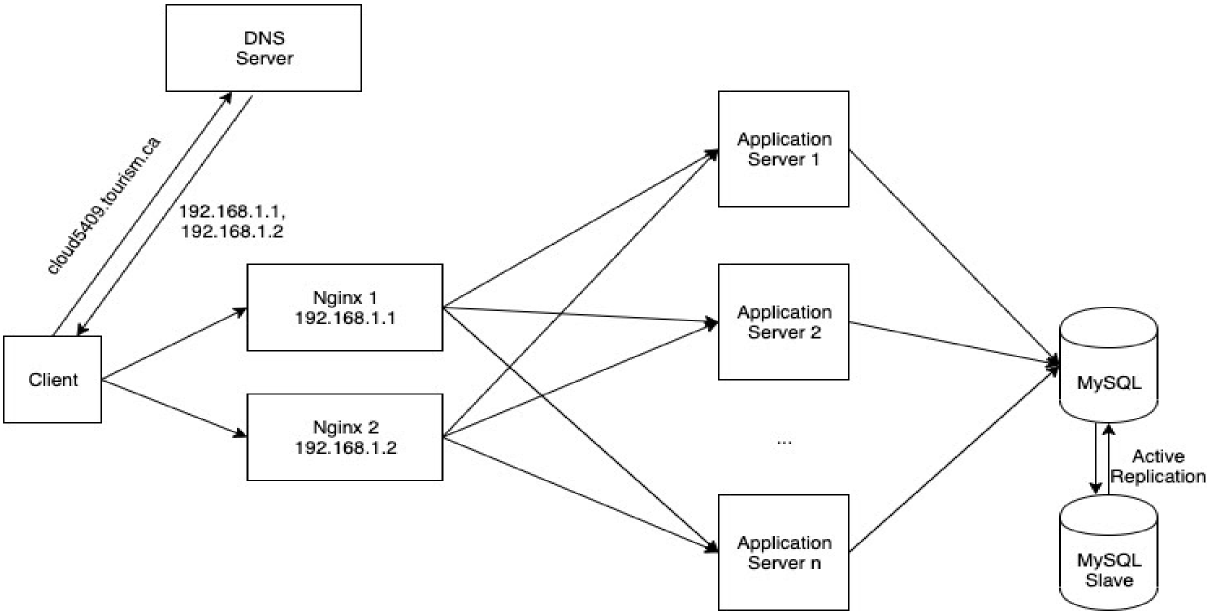


Figure 3: Architecture

Challenges

1. LoadBalancing - By default, AWS provides a distinct IP and domain name for each instance. To perform DNS load balancing, it is necessary to point the same domain name to 2 different IPs.
2. Discover what would be the difference between the API calls between the websites and mobile platform.

References

[1] "What is an SSL Certificate?," *Globalsign.com*, 2018. [Online]. Available: https://www.globalsign.com/en/ssl-information-center/what-is-an-ssl-certificate/. [Accessed: 28-Jan-2020].

[2] "How to Install a Free or Paid SSL Certificate for Your Website | Sucuri," *Sucuri*, 2019. [Online]. Available: https://sucuri.net/guides/how-to-install-ssl-certificate/. [Accessed: 28-Jan-2020].

[3] M. Otto, "Grid system," *Getbootstrap.com*, 2020. [Online]. Available: https://getbootstrap.com/docs/4.1/layout/grid/. [Accessed: 28-Jan-2020].

[4] "What is Kubernetes," *Kubernetes.io*, 2014. [Online]. Available: https://kubernetes.io/docs/concepts/overview/what-is-kubernetes/. [Accessed: 28-Jan-2020].

[5] "Spring Projects", *Spring.io*, 2020. [Online]. Available: https://spring.io/projects/spring-boot. [Accessed: 30- Jan- 2020].

[6] B. Porter, J. Zyl and O. Lamy, "Maven – Welcome to Apache Maven", *Maven.apache.org*, 2020. [Online]. Available: https://maven.apache.org/. [Accessed: 30- Jan- 2020].

[7] G. Features, C. Plugins, G. Kotlin, M. Gradle, M. Resources and G. Enterprise, "Gradle Build Tool", *Gradle*, 2020. [Online]. Available: https://gradle.org/. [Accessed: 30- Jan- 2020].

[8] "MySQL :: Oracle MySQL Cloud Service", *Mysql.com*, 2020. [Online]. Available: https://www.mysql.com/cloud/. [Accessed: 30- Jan- 2020].