HOST STATIC WEBSITE ON AWS

A Project Work Report

*Submitted in the partial fulfilment for the award of the degree of*

**BACHELOR OF ENGINEERING**

**IN**

**CLOUD COMPUTING**

**Submitted by:**

**ANKIT KUMAR RAI – 20BCS4144**

**ADITYA RAJ– 20BCS4140**

**ISHITA GUPTA – 20BCS4048**

**HARDIK– 20BCS4123**

**Under the Supervision of:**

**Mrs. Geetinder Saini**



**CHANDIGARH UNIVERSITY, GHARUAN, MOHALI - 140413,**

**PUNJAB**

**August – 2022**

**Bonafide Certificate**

This is to certify that the work embodied in this Project Report entitled **“ HOST STAIATIC WEBSITE ON AWS”** being submitted by **“ 20BCS4144 ”, “20BCS4048”, “20BCS4140” and “ 20BCS4123” ,** 5th Semester for partial fulfillment of the requirement for the degree of **“Bachelor of Engineering in Computer Science & Engineering** **with specialization in Cloud Computing”** discipline in “ **Chandigarh University** ” during the academic session August-Dec 2022 is a record of Bonafede piece of work, carried out by student under my supervision and guidance in the **“Department of Computer Science & Engineering ”, Chandigarh University.**

### APPROVED & GUIDED BY:

### DECLARATION

### We, student of Bachelor of Engineering in Computer Science & Engineering with specialization in Cloud Computing, 5th Semester, session: Aug – Dec 2022, Chandigarh University, hereby declare that the work presented in this Project Report entitled “HOST STATIC WEBSITE ON AWS” is the outcome of our own work, is bona fide and correct to the best of my knowledge and this work has been carried out taking care of Engineering Ethics. The work presented does not infringe any patented work and has not been submitted to any other university or anywhere else for the award of any degree or any professional diploma.

**STUDENT DETAILS AND SIGNATURE**

Ishita Gupta (20BCS4048)

Ankit Kumar Rai (20BCS4144)

Hardik(20BCS4123)

Aditya(20BCS4140)

**APPROVED & GUIDED BY**

**HOST STATIC WEBSITE ON AWS**

**A PROJECT REPORT**

**.**

**ACKNOWLEDGEMENT**

It is indeed with an excellent pleasure and immense sense of gratitude that we acknowledge the assistance of those individuals. We are highly indebted to our Supervisor Mrs. Lata Gupta, for the facilities provided to accomplish this main project.

We might prefer to thank our Dr. Aman Kaushik, Head of the Department of Computer science and Engineering, Chandigarh University, for this constructive criticism throughout our project.

We feel elated in manifesting our sense of gratitude tour internal project guide and our co-supervisor Mr. Sant Kumar Maurya ,Department of Computer science and Engineering, Chandigarh University. He has been a relentless source of inspiration for us and that we are very deeply thankful to him for his support and valuable advice.

We extremely grateful to our Departmental staff members, Lab technicians and non-teaching staff members for their extreme help throughout our project.

Finally, we express our heart full thanks to all of our friends who helped us in successful completion of this project.

**Project Associates:**

**ABSTRACT**

It is well-known : It’s almost impossible to overstate the importance of website ownership to businesses, large and small alike. The Internet gives even the smallest of startup companies a chance to connect with millions of potential customers. A well-designed small business website is your best chance of making a lasting impression on your target audience. Nowadays mostly everyone probably knows the importance of having their business on the internet. AWS itself provides a free tier type of access for one year for new users to try different services. They provide almost all types of infrastructure services required for internet-connected audience and businesses. The simplest form of website architecture is the static website, where users are served static content. Some examples include brand microsites, marketing websites, and intranet information pages. Static websites are straightforward, but they can still have demanding requirements in terms of scalability, availability, and service-level guarantees. For example, a marketing site for a consumer brand may need to be prepared for an unpredictable onslaught of visitors when a new product is launched. This paper cover comprehensive architectural guidance for developing, deploying, and managing static websites on Amazon Web Services (AWS) while keeping operational simplicity and business requirements in mind. We also recommend an approach that provides 1) insignificant cost of operation, 2) little or no management required, and 3) a highly scalable, resilient, and reliable website. This paper first reviews how static websites are hosted in traditional hosting environments. Then, we explore a simpler and more cost-efficient approach using Amazon Simple Storage Service (Amazon S3). Finally, we show you how you can enhance the AWS architecture by encrypting data in transit and to layer on functionality and improve quality of service by using Amazon CloudFront.

**TABLE OF CONTENT**

Title Page i

Bona fide Certificate ii

Declaration iii

Acknowledgement v

Abstract vi

Table of content vii

1. INTRODUCTION Introduction ……………………………………………………………………………… 11

1.1 Salient Features of our AWS…………….………………………………… 12

1.2 What are the services offered by existing system…………………… . 16

1.3 Problem Definition…………………………………….…….……………. 16

2. REQUIREMENTS SPECIFICATION

2.1 Introduction……………………….…………………………………………. 18

2.2 Hardware requirements……………………………………………………… 19

2.3 Software requirements………………………………………………………. 19

3. ANALYSIS

3.1 Existing System ………………….……………………………………….…. 22

3.2 Proposed System ……………………………………………………………. 23

3.3 Feasibility study ……………………………..……………………………… 24

3.4 Software specification ………………………………………………………. 25

4. DESIGN

4.1 System Design ……………………………………………………………….. 31

4.1.1 Introduction to AWS……..………………………………………………… 31

4.1.2 How Amazon S3 works ……………………………………………... 31

5. SAMPLE SCREENSHOTS 42

6.CONCLUSION 48

7. FUTURE SCOPE 50

8. BIBLOGRAPHY 52

9. REFERENCES 54

10. APPENDIX 58

CHAPTER 1

**INTRODUCTION**

**1.1 Introduction:**

Content on a static website is shown in the same format that it is stored. There is no need for server-side code execution. For instance, a static website that only has HTML documents displaying images would transmit the HTML and images to the browser in their original form. Web browsers on computers, tablets, or mobile devices can receive static websites. A variety of HTML pages, photos, videos, CSS style sheets, and JavaScript files are frequently included. Static websites can include client-side interactivity, so they don't have to be dull. You can create rich user experiences that are interactive and engaging by combining HTML5 with client-side JavaScript technologies like jQuery, AngularJS, React, and Backbone. Examples of static websites are as follows:

* Marketing Websites
* Product Landing pages
* Microsites that display the same content to all users
* Team Homepages
* A website that lists available assets (e.g., image files, video files, and press releases) allows the user to download the files as-is
* Proofs-of-concept used in the early stages of web development to test user experience flows and gather feedback

Since material is delivered as-is and can be cached by a content delivery network, static websites load rapidly (CDN). There is no application logic or database querying required on the web server. They are also rather affordable to create and host. Without automated methods, maintaining huge static websites can be difficult, and static websites are unable to provide individualized information.

The best situations for static websites are those where the material is rarely changed. It is best to take into account a dynamic website design whenever the content increases in complexity or has to be often updated, customized, or dynamically created. However, for this project, we'll use AWS to host a static website.

This project is a component of an evaluation of cloud computing technology. The Foundation for Cloud Research is looking on ways to employ information technology more often. The project mainly involves hosting a static travel and tour website on Amazon Web Services.

We have a lot of advantages running in AWS, including security, scalability, pricing, and agility advantages. The top-notch infrastructure and security operations of AWS are advantageous for this design. The website becomes prepared for traffic surges with the usage of the Auto Scaling function, enabling website hosts to be ready for product launches and the potential for their website to go popular in the future. Additionally, AWS uses a pay-as-you-go model, which essentially enables us to use the resources effectively and without any issues. Additionally, the improved agility of the host is a result of the demand-based nature of AWS services. Since the infrastructure management is not required of the site hosts, more time and resources can be used to develop value that differentiates their business.

AWS upends conventional IT thinking and makes new "cloud-native" architectures possible. A contemporary static website may be built without using a single web server.

* 1. **SAILENT FEATURES OF AWS**

You must first establish a bucket, give it a name, and choose an AWS Region in order to store your data in Amazon S3. Then, you upload your data as Amazon S3 objects to that bucket. Each item has a key (or key name), which serves as the item's specific bucket identification.

You may customize the functionality that S3 offers to support your unique use case. For instance, you may utilize S3 Versioning to store several versions of an object in the same bucket, enabling you to recover any mistakenly deleted or overwritten items.

Buckets and the items they contain are private and may only be accessed with permissions that you expressly provide. To regulate access, you may make use of bucket policies, IAM policies from AWS, access control lists (ACLs), and S3 Access Points.

**Buckets**

A bucket is a storage location for items in Amazon S3. A bucket may hold unlimited number of items, and your account can hold up to 100 buckets.

When you create a bucket, you give it a name and select the AWS Region in which it will be stored. The name and region of a bucket cannot be changed after it has been created. The guidelines for bucket naming must be followed. Additionally, you may set up a bucket to leverage S3 Versioning or other storage administration tools.

Buckets also:

* Organize the Amazon S3 namespace at the highest level.
* Identify the account responsible for storage and data transfer charges.
* Provide access control options, such as bucket policies, access control lists (ACLs), and S3 Access Points, that you can use to manage access to your Amazon S3 resources.
* Serve as the unit of aggregation for usage reporting.

**Objects**

The fundamental units kept in Amazon S3 are called objects. Metadata and object data make up an object. A collection of name-value pairs that describe the item makes up the metadata. These pairs contain both typical HTTP metadata, such Content-Type, and some default metadata, like the modification date. Custom metadata can also be specified as the object is being saved.

**Keys**

A bucket's objects are uniquely identified by object keys, also known as key names. A bucket has exactly one key for each item within. Each item is uniquely identified by a bucket, an object key, and, if S3 Versioning is enabled for the bucket, optionally, a version ID. As a result, you may consider Amazon S3 to be a simple data map between "bucket + key + version" and the actual item.

Through the combination of the web service endpoint, bucket name, key, and optionally, a version, every object in Amazon S3 may be individually addressed. For instance, DOC-EXAMPLE-BUCKET is the name of the bucket, and photos/puppy.jpg is the key, in the URL https://DOC-EXAMPLE-BUCKET.s3.us-west-2.amazonaws.com.

**S3 Versioning**

To retain many versions of an object in the same bucket, utilize S3 Versioning. You may save, retrieve, and restore any version of any item saved in your buckets using S3 Versioning. Both accidental user behavior and programmed problems are simple to fix.

**Bucket Policy**

In order to provide access permissions to your bucket and the objects inside of it, you may use a bucket policy, which is a resource-based AWS Identity and Access Management (IAM) policy. A policy can only be linked to a bucket by its owner. All of the items in the bucket that belong to the bucket owner are subject to the permissions associated with the bucket. Policies can only be up to 20 KB in size.

The common JSON-based access policy language used by bucket policies is used throughout AWS. For the items in a bucket, you may add or remove rights using bucket policies. Based on the parts in the policy, such as the requester, S3 activities, resources, and characteristics or circumstances of the request, bucket policies either approve or reject requests (for example, the IP address used to make the request). You might, for instance, design a bucket policy that permits cross-account uploading of things to an S3 bucket while guaranteeing that the bucket owner has complete control over the uploaded items.

**S3 Access Points**

Network endpoints with specific access restrictions that specify how data may be accessed via that endpoint are known as Amazon S3 Access Points. You can utilize Access Points to carry out S3 object operations like GetObject and PutObject since they are connected to buckets. For shared datasets in Amazon S3, access points make controlling data access at scale simpler.

The access point policy varies depending on the access point. For each access point, you may alter the settings for Block Public Access. You may also set up any access point to only accept requests from a virtual private cloud in order to limit access to Amazon S3 data to a specific private network (VPC).

**Access Control Lists(ACLs)**

For certain buckets and objects, you may use ACLs to provide authorized users read and write capabilities. An ACL is a subresource that is connected to each bucket and item. The kind of access and the AWS accounts or groups to whom it is given are specified by the ACL. The access control method used before IAM was IAM was ACLs.

When an item is uploaded to your S3 bucket by another AWS account, that account (the object writer) automatically owns the object, has access to it, and may allow additional users access to it using ACLs. ACLs may be deactivated, and you automatically own each and every item in your bucket by using Object Ownership to override this default behavior. As a result, policies like as IAM policies, S3 bucket policies, virtual private cloud (VPC) endpoint policies, and AWS Organizations service control policies are used to manage access to your data (SCPs).

**Regions**

The geographical AWS Region in which Amazon S3 stores the buckets you create is your choice. To improve latency, reduce costs, or satisfy regulatory standards, you could select a Region. Unless you specifically move or copy an object to another AWS Region, it never leaves that Region. For instance, no items are ever removed from the Europe (Ireland) Region.

**1.2** **What are the services offered by the Existing Systems?**

1. Travel advisory- Advising clients
2. Travel arrangements, e.g. visas and passports
3. Keeping clients up to date with any changes and risks etc
4. Bookings
5. Arranging flights, insurance and
6. Hotel and transport booking

**1.3 Problem Definition**

Web application is a Web site where user input (navigation through the site and data entry) affects the state of the business: beyond, of course, access logs and hit counters. In essence, a Web application uses a Web site as the front end to a business application,

A web application (web app) is any application software that runs in a web browser and is created in a browser supported programming language (such as the combination of JavaScript, HTML and CSS) and relies on a web browser to render the application. The main problem is that when multiple users need to use any type of web applications such as Blogger to watch video or to share post in the same time , they will face difficulties because the high loading on main server. To solve this problem we host this web application on Amazon Cloud to provide special server to this web application

CHAPTER 2

**REQUIREMENT SPECIFICATION**

**2.1 INTRODUCTION:**

To be used efficiently, all computer software needs certain hardware components or the other software resources to be present on a computer. These pre-requisites are known as(computer) system requirements and are often used as a suggestion as against an absolute rule. Most software defines two sets of system requirements: minimum and recommended. With increasing demand for higher processing power and resources in newer versions of software, system requirements tend to increase over time. Industry analysts suggest that this trend plays an even bigger part in driving upgrades to existing computer systems than technological advancements.

The travelling system is all about the use of proper information about a location and how to organize that data in a working format which is efficient and compatible to the user. Better knowledge helps the traveler to have some ease in the journey. With their trip being taken care of by the package manager, the traveler can enjoy the trip to its full extent without any care of staying, eating, etc. They can also share their reviews about how the travelling experience was and if the feedback is bad, improvements can be made in that matter.

Also, the website that provides all these services is itself too much difficult to manage with all the traffic and hosting the website on a web server. But this can be done easily with Amazon S3(Simple Storage Service) service provided by AWS.

**2.2 HARDWARE REQUIREMENTS:**

The most common set of requirements defined by any package or software application is that the physical computer resources, also referred to as hardware. A hardware requirements list is usually in the midst of a hardware compatibility list (HCL), especially just in case of operating systems. An HCL lists tested, compatibility and sometimes incompatible hardware devices for a selected software or application. the subsequent sub-sections discuss the varied aspects of hardware requirements.

**HARDWARE REQUIREMENTS FOR PRESENT PROJECT**:

* Processor: Any (e.g., Intel Core i3 / Intel Core i5)
* RAM: 2GB or above
* Hard Disk Drive: 5 MB or more

**2.3 SOFTWARE REQUIREMENTS:**

Software Requirements house defining software resource requirements and pre-requisites that require to be installed on a computer to produce optimal functioning of an application. These requirements or pre-requisites are generally not included within the software installation package and want to be installed separately before the software is installed.

**SOFTWARE REQUIREMENTS FOR PRESENT PROJECT:**

OPERATING SYSTEM : Windows 7/ XP/8

FRONT END : HTML, CSS, JavaScript query

DATABASE : MySQL

IDE : VS CODE

Web Browser : Any latest browser

**CHAPTER 3**

**ANALYSIS**

**3.1 EXISTING SYSTEM:**

* Online travel agents can charge commission on every sale. This can range between 10-20 per cent of the gross cost.
* There may be restrictive terms and conditions imposed by OTAs such as guest cancellation and automatic room reselling policies.
* You may need to find a way of managing room availability across a range of OTAs, your own website, front desk and telephone sales. This can be time consuming and labour intensive. There are software options to help you manage this.
* Using OTAs does not reduce the need to have your own website with booking engine.
* You may still need to invest in a balanced multi-channel marketing strategy.

Although online travel agents can help you fill rooms, it is important to **maximise your most profitable revenues** through sales on your own website. You should continually work on your search engine optimisation tactics and other digital marketing techniques. Focus on customer retention tactics. Directly target your existing customers (including those gained through OTAs) through email and direct marketing.

For our project we did some research and found out major limitation of a tour and travel packages

* They are not having control on the package
* There are strict schedules and guidelines one needs to follow and if one desires to alter the package it cannot be done
* One needs to rightfully select a travel and tour package with enough time on hand and with the right company

**3.2 PROPOSED SYSTEM:**

The package tour industry is often high risk, high breakeven, high-quality product, and competitively priced. As a result, thorough tour planning and market research are necessary for tour management. However, before a tour is created, the tour manager should consider a few elements that are essential to the planning process. The satisfaction of the visitor is significantly impacted by these variables. The primary elements are:

The aim of proposed system is to develop a system of improved facilities. The provides proper security and reduces the manual work. It also provides a better GUI for the user

* Enter the destination and no. of guests and the date of arrival and leaving.
* User gets all the details via email. Such as Cab driver’s number, Hotel’s details, a detailed map of area .
* User can also choose a package for the respective locations, and then follow the plan and can enjoy the whole trip without any worry.
* Arranging Concierges / guides etc as needed
* Deals with complaints or refunds.
* Collecting and processing payments.

By offering as many specialized ideas as possible, we hope to assist tourists in finding destinations and journeys that best suit their tastes. Users won't need to compromise on their preferences or considerations thanks to this website, which also eliminates the time-consuming job of conducting online research.

**3.3 FEASIBILITY STUDY**

The feasibility of the project is analyzed during this phase and business proposal is put forth with a really general plan for the project and a few cost estimates. During system analysis the feasibility study of the proposed system is to be disbursed. this is often to confirm that the proposed system isn't a burden to the corporate. For feasibility analysis, some understanding of the foremost requirements for the system is important.

Three key considerations involved in the feasibility analysis are:

**3.3.1 Economic Feasibility**

This study is distributed to test the economic impact will wear the system will wear the organization. the quantity of fund that the corporate can pour into the research and development of the system is proscribed. The expenditures must be justified. Thus, the developed system yet within the budget and this was achieved because most of the technologies used are freely available. Only the customized products should be purchased.

**3.3.2 Technical Feasibility**

This study is dole out to test the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. this may cause high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes for the implementing this technique.

**3.3.3 Operational Feasibility**

The aspect of study is to test the amount of acceptance of the system by the user. This includes the method of coaching the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. the extent of acceptance by the users solely depends on the methods that are employed to coach the user about the system and to create him conversant in it. His level of confidence must be raised so he's also ready to make some constructive criticism, which is welcomed, as he's the ultimate user of the system.

**3.4 SOFTWARE SPECIFICATION**

**CASCADING STYLE SHEETS** (**CSS**):

It is a [style sheet language](http://en.wikipedia.org/wiki/Style_sheet_language) used for describing the [look and formatting](http://en.wikipedia.org/wiki/Presentation_semantics) of a document written in a [markup language](http://en.wikipedia.org/wiki/Markup_language). While most often used to style [web pages](http://en.wikipedia.org/wiki/Web_page) and [interfaces](http://en.wikipedia.org/wiki/Interface_(computing)) written in [HTML](http://en.wikipedia.org/wiki/HTML) and [XHTML](http://en.wikipedia.org/wiki/XHTML), the language can be applied to any kind of [XML](http://en.wikipedia.org/wiki/XML) document, including [plain XML](http://en.wikipedia.org/wiki/Plain_Old_XML), [SVG](http://en.wikipedia.org/wiki/Scalable_Vector_Graphics) and [XUL](http://en.wikipedia.org/wiki/XUL). CSS is a cornerstone specification of [the web](http://en.wikipedia.org/wiki/The_web) and almost all web pages use CSS style sheets to describe their presentation.

CSS is designed primarily to enable the separation of document content from document presentation, including elements such as the [layout](http://en.wikipedia.org/wiki/Page_layout), [colors](http://en.wikipedia.org/wiki/Color), and [fonts](http://en.wikipedia.org/wiki/Typeface).[[1]](http://en.wikipedia.org/wiki/Cascading_Style_Sheets#cite_note-1) This separation can improve content [accessibility](http://en.wikipedia.org/wiki/Accessibility), provide more flexibility and control in the specification of presentation characteristics, enable multiple pages to share formatting, and reduce complexity and repetition in the structural content .

CSS can also allow the same markup page to be presented in different styles for different rendering methods, such as on-screen, in print, by voice (when read out by a speech-based browser or [screen reader](http://en.wikipedia.org/wiki/Screen_reader)) and on [Braille-based](http://en.wikipedia.org/wiki/Braille_display), tactile devices. It can also be used to allow the web page to display differently depending on the screen size or device on which it is being viewed. While the author of a document typically links that document to a CSS file, readers can use a different style sheet, perhaps one on their own computer, to override the one the author has specified. However, if the author or the reader did not link the document to a specific style sheet the default style of the browser will be applied.

**MySql:**

MySQL is developed by Dr. Richard Hipp and is a database engine written in the C programming language. It is not a standalone app; rather, it is a library that software developers embed in their apps. As such, it belongs to the family of embedded databases. MySQL is a database system used on the web it runs on a server. MySQL is ideal for both small and large applications. It is very fast, reliable, and easy to use. It supports standard SQL. MySQL can be compiled on a number of platforms.

The data in MySQL is stored in tables. A table is a collection of related data, and it consists of columns and rows. Databases are useful when storing information categorically.

**FEATURES OF MySQL:**

Internals and portability:

* Written in C and C++.
* Tested with a broad range of different compilers.
* Works on many different platforms.
* Tested with Purify (a commercial memory leakage detector) as well as with Val grind, a GPL tool.
* Uses multi-layered server design with independent modules.

#### Security:

* A privilege and password system that is very flexible and secure, and that enables host-based verification.
* Password security by encryption of all password traffic when you connect to a server.

#### Scalability and Limits:

* Support for large databases. We use MySQL Server with databases that contain 50 million records. We also know of users who use MySQL Server with 200,000 tables and about 5,000,000,000 rows.
* Support for up to 64 indexes per table (32 before MySQL 4.1.2). Each index may consist of 1 to 16 columns or parts of columns. The maximum index width is 767 bytes for **InnoDB** tables, or 1000 for **MyISAM**; before MySQL 4.1.2, the limit is 500 bytes. An index may use a prefix of a column for [**CHAR**](https://dev.mysql.com/doc/refman/5.0/en/char.html), [**VARCHAR**](https://dev.mysql.com/doc/refman/5.0/en/char.html), [**BLOB**](https://dev.mysql.com/doc/refman/5.0/en/blob.html), or [**TEXT**](https://dev.mysql.com/doc/refman/5.0/en/blob.html) column types.

#### CONNECTIVITY:

Clients can connect to MySQL Server using several protocols:

* Clients can connect using TCP/IP sockets on any platform.
* On Windows systems in the NT family (NT, 2000, XP, 2003, or Vista), clients can connect using named pipes if the server is started with the [--enable-named-pipe](https://dev.mysql.com/doc/refman/5.0/en/server-options.html#option_mysqld_enable-named-pipe) option. In MySQL 4.1 and higher, Windows servers also support shared-memory connections if started with the [--shared-memory](https://dev.mysql.com/doc/refman/5.0/en/server-options.html#option_mysqld_shared-memory) option. Clients can connect through shared memory by using the --protocol=memory option.
* On UNIX systems, clients can connect using Unix domain socket files.

#### LOCALIZATION:

* The server can provide error messages to clients in many languages.
* All data is saved in the chosen character set.

**CLIENTS AND TOOLS**:

* MySQL includes several client and utility programs. These include both command-line programs such as [**MySQLdump**](https://dev.mysql.com/doc/refman/5.0/en/mysqldump.html) and **[MySQLadmin](https://dev.mysql.com/doc/refman/5.0/en/mysqladmin.html" \o "4.5.2 mysqladmin — Client for Administering a MySQL Server)**, and graphical programs such as [MySQL Workbench](http://dev.mysql.com/doc/refman/5.1/en/workbench.html).
* MySQL Server has built-in support for SQL statements to check, optimize, and repair tables. These statements are available from the command line through the [**MySQL check**](https://dev.mysql.com/doc/refman/5.0/en/mysqlcheck.html) client. MySQL also includes [**mismatch**](https://dev.mysql.com/doc/refman/5.0/en/myisamchk.html), a very fast command-line utility for performing these operations on **MyISAM** tables.
* MySQL programs can be invoked with the --help or -? option to obtain online assistance.

**WHY TO USE MySQL:**

* Leading open-source RDBMS
* Ease of use – No frills
* Fast
* Robust
* Security
* Multiple OS support
* Free
* Technical support
* Support large database– up to 50 million rows, file size limits up to 8 million TB

**JAVASCRIPT:**

JavaScript (js) is a light-weight object-oriented programming language which is used by several websites for scripting the webpages. It is an interpreted, full-fledged programming language that enables dynamic interactivity on websites when applied to an HTML document. It was introduced in the year 1995 for adding programs to the webpages in the Netscape Navigator browser. Since then, it has been adopted by all other graphical web browsers. With JavaScript, users can build modern web applications to interact directly without reloading the page every time. The traditional website uses js to provide several forms of interactivity and simplicity.

Although, JavaScript has no connectivity with Java programming language. The name was suggested and provided in the times when Java was gaining popularity in the market. In addition to web browsers, databases such as CouchDB and MongoDB uses JavaScript as their scripting and query language.

**FEATURES OF JAVASCRIPT**

There are following features of JavaScript:

* All popular web browsers support JavaScript as they provide built-in execution environments.
* JavaScript follows the syntax and structure of the C programming language. Thus, it is a structured programming language.
* JavaScript is a weakly typed language, where certain types are implicitly cast (depending on the operation).
* JavaScript is an object-oriented programming language that uses prototypes rather than using classes for inheritance.
* It is a light-weighted and interpreted language.
* It is a case-sensitive language.
* JavaScript is supportable in several operating systems including, Windows, macOS, etc.
* It provides good control to the users over the web browsers.

## Application of JavaScript

JavaScript is used to create interactive websites. It is mainly used for:

* Client-side validation,
* Dynamic drop-down menus,
* Displaying date and time,
* Displaying pop-up windows and dialog boxes (like an alert dialog box, confirm dialog box and prompt dialog box),
* Displaying clocks etc.

**WHY TO LEARN JAVASCIPT?**

JavaScript is the most popular and hence the most loved language around the globe. Apart from this, there are abundant reasons to learn it. Below are a listing of few important points:

* No need of compilers: Since JavaScript is an interpreted language, therefore it does not need any compiler for compilations.
* Used both Client and Server-side: Earlier JavaScript was used to build client-side applications only, but with the evolution of its frameworks namely Node.js and Express.js, it is now widely used for building server-side applications too.
* Helps to build a complete solution: As we saw, JavaScript is widely used in both client and server-side applications, therefore it helps us to build an end-to-end solution to a given problem.
* Used everywhere: JavaScript is so loved because it can be used anywhere. It can be used to develop websites, games or mobile apps, etc.
* Huge community support: JavaScript has a huge community of users and mentors who love this language and take it’s legacy forward.

**CHAPTER 4**

**DESIGN**

4.1 SYSTEM DESIGN:

4.1.1 INTRODUCTION TO AWS:

Amazon Simple Storage Service (Amazon S3) is an object storage service that offers industry-leading scalability, data availability, security, and performance. Customers of all sizes and industries can use Amazon S3 to store and protect any amount of data for a range of use cases, such as data lakes, websites, mobile applications, backup and restore, archive, enterprise applications, IoT devices, and big data analytics. Amazon S3 provides management features so that you can optimize, organize, and configure access to your data to meet your specific business, organizational, and compliance requirements.

**Features of Amazon S3:**

**Storage classes**

Amazon S3 offers a range of storage classes designed for different use cases. For example, you can store mission-critical production data in S3 Standard for frequent access, save costs by storing infrequently accessed data in S3 Standard-IA or S3 One Zone-IA, and archive data at the lowest costs in S3 Glacier Instant Retrieval, S3 Glacier Flexible Retrieval, and S3 Glacier Deep Archive.

You can store data with changing or unknown access patterns in S3 Intelligent-Tiering, which optimizes storage costs by automatically moving your data between four access tiers when your access patterns change. These four access tiers include two low-latency access tiers optimized for frequent and infrequent access, and two opt-in archive access tiers designed for asynchronous access for rarely accessed data.

For more information, see Using Amazon S3 storage classes. For more information about S3 Glacier Flexible Retrieval, see the *Amazon S3 Glacier Developer Guide*.

**Storage management**

Amazon S3 has storage management features that you can use to manage costs, meet regulatory requirements, reduce latency, and save multiple distinct copies of your data for compliance requirements.

**Access management**

Amazon S3 provides features for auditing and managing access to your buckets and objects. By default, S3 buckets and the objects in them are private. You have access only to the S3 resources that you create. To grant granular resource permissions that support your specific use case or to audit the permissions of your Amazon S3 resources, you can use the following features

**Data processing**

To transform data and trigger workflows to automate a variety of other processing activities at scale, you can use the following features.

**Storage logging and monitoring**

Amazon S3 provides logging and monitoring tools that you can use to monitor and control how your Amazon S3 resources are being used.

**Analytics and insights**

Amazon S3 offers features to help you gain visibility into your storage usage, which empowers you to better understand, analyze, and optimize your storage at scale.

**Strong consistency**

Amazon S3 provides strong read-after-write consistency for PUT and DELETE requests of objects in your Amazon S3 bucket in all AWS Regions. This behavior applies to both writes of new objects as well as PUT requests that overwrite existing objects and DELETE requests.

4.2 How Amazon S3 works

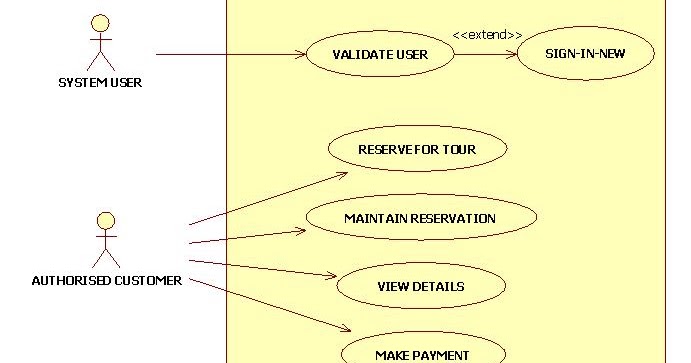
Amazon S3 is an object storage service that stores data as objects within buckets. An *object* is a file and any metadata that describes the file. A *bucket* is a container for objects.

To store your data in Amazon S3, you first create a bucket and specify a bucket name and AWS Region. Then, you upload your data to that bucket as objects in Amazon S3. Each object has a *key* (or *key name*), which is the unique identifier for the object within the bucket.

S3 provides features that you can configure to support your specific use case. For example, you can use S3 Versioning to keep multiple versions of an object in the same bucket, which allows you to restore objects that are accidentally deleted or overwritten.

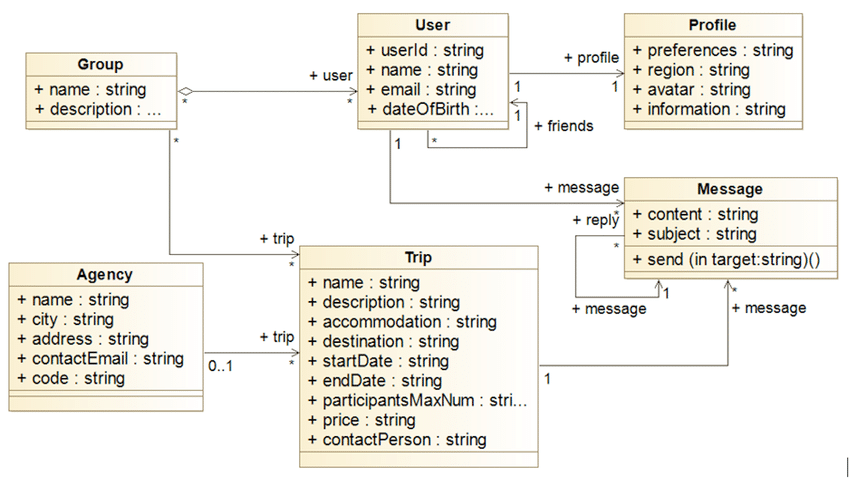
Buckets and the objects in them are private and can be accessed only if you explicitly grant access permissions. You can use bucket policies, AWS Identity and Access Management (IAM) policies, access control lists (ACLs), and S3 Access Points to manage access.

**Use case diagram of our project:**

****

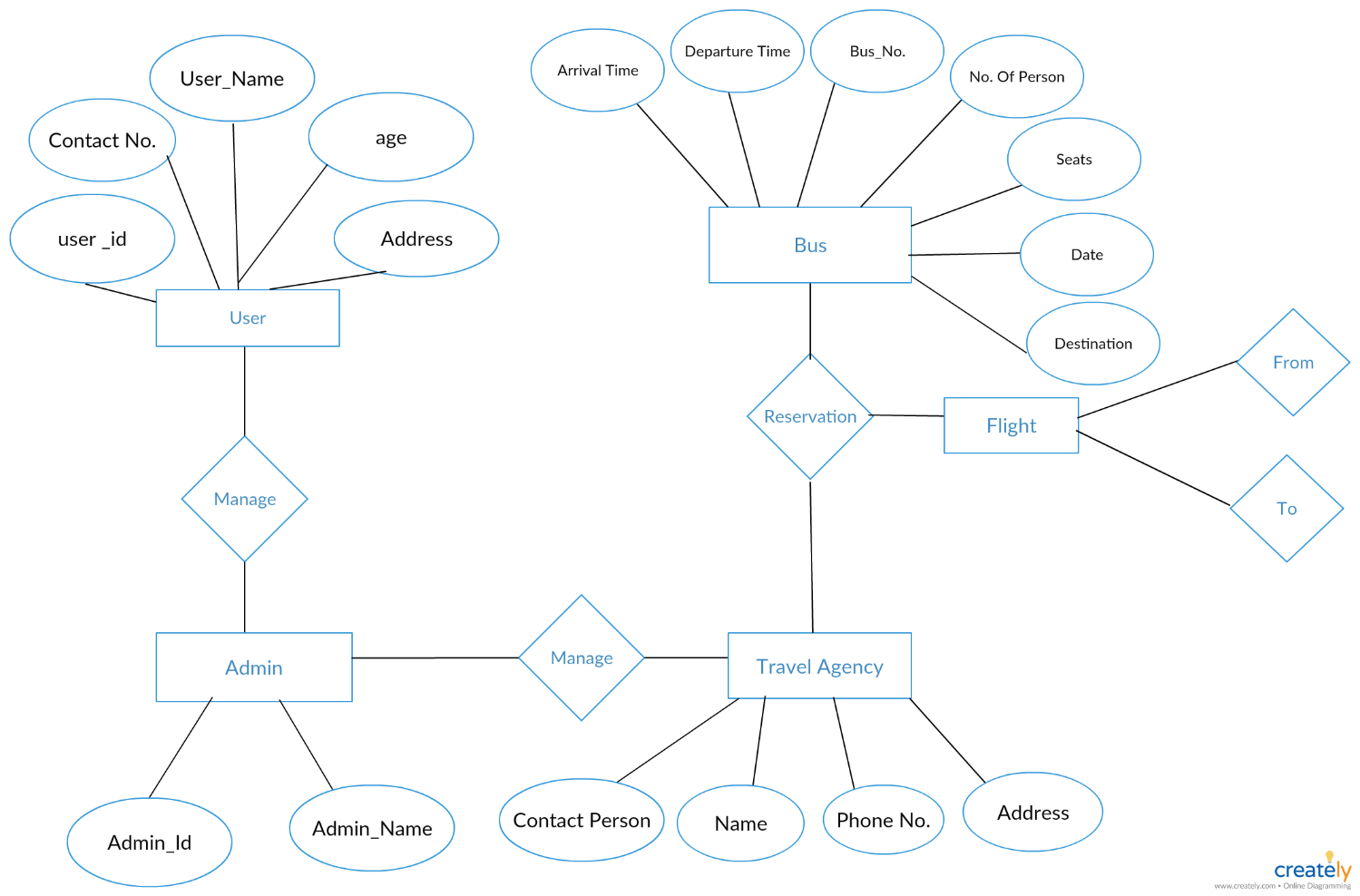
**Class Diagram**:

A Class is a category or group of things that has similar attributes and common behavior. A Rectangle is the icon that represents the class it is divided into three areas. The upper most area contains the name, the middle; area contains the attributes and the lowest areas show the operations. Class diagrams provides the representation that developers work from. Class diagrams help on the analysis side, too.



**ER Diagram:**

Entity Relationship Diagram (ER Diagram or ERD) could be a pictorial or visual representation of classifying groups or entities of common interest and defining the connection between these groups. Hence, a structure is formed with various symbols of various shapes and sizes in order that it may be used as a model to depict the interior structure & relationship.

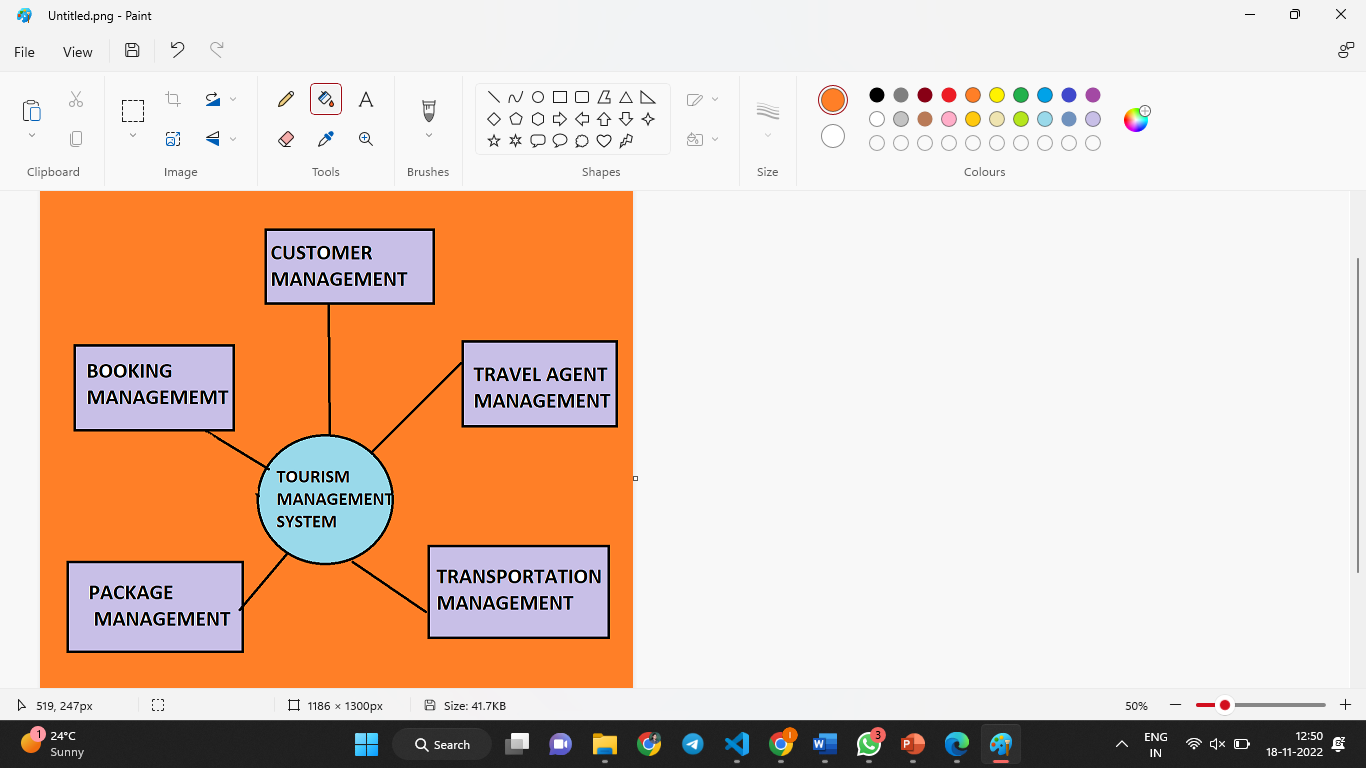
****

**DFD DIAGRAM:**

Data flow chart (DFD) could be a traditional visual representation of the data flows within a system. A neat and clear DFD can depict the proper amount of the system requirement graphically. It will be manual, automated, or a mix of both.

It shows how data enters and leaves the system, what changes the knowledge, and where data is stored.

The objective of a DFD is to indicate the scope and bounds of a system as a full. it's going to be used as a communication tool between a system analyst and someone who plays a component within the order that acts as a start line for redesigning a system. The DFD is additionally called as a knowledge flow graph or bubble chart.

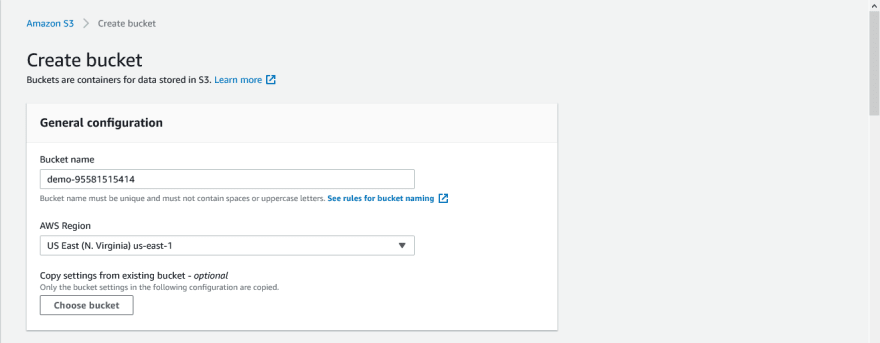


**Working Model**

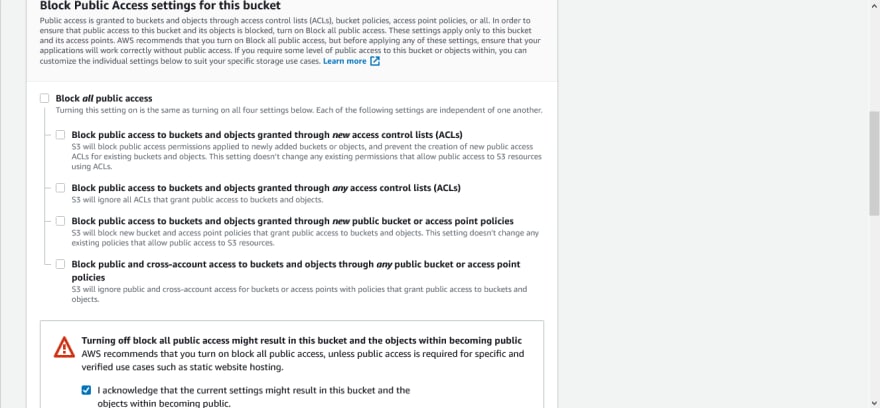
**HOSTING STAIC WEBSITE ON AWS**

**Step 1 — Create an S3 bucket**

Click on **Create bucket**. Give the bucket a unique name, the name you choose must be globally unique (for best practice, attach your AWS account ID to the name).

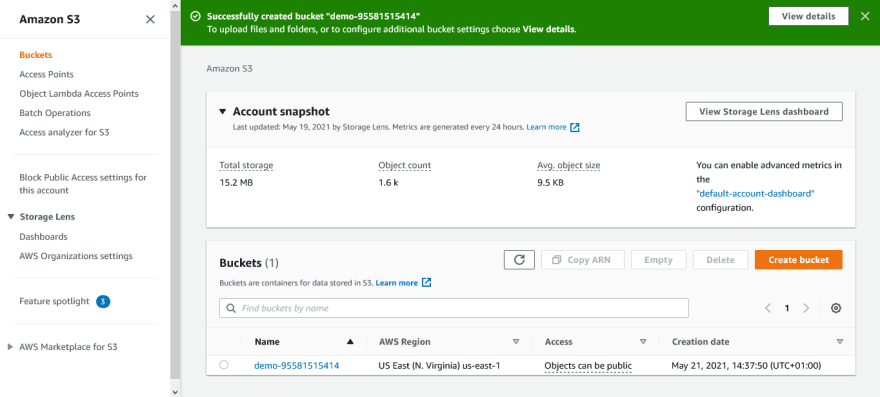
[](https://res.cloudinary.com/practicaldev/image/fetch/s--v0zCOAFZ--/c_limit%2Cf_auto%2Cfl_progressive%2Cq_auto%2Cw_880/https:/user-images.githubusercontent.com/33374159/119145517-9f1c3a80-ba41-11eb-9c6b-f4a17a052e82.png)

Uncheck the **Block all public access** checkbox and accept the acknowledgement.

[](https://res.cloudinary.com/practicaldev/image/fetch/s--6HPrELCo--/c_limit%2Cf_auto%2Cfl_progressive%2Cq_auto%2Cw_880/https:/user-images.githubusercontent.com/33374159/119145647-c115bd00-ba41-11eb-8c50-b27cfa04fbbb.png)

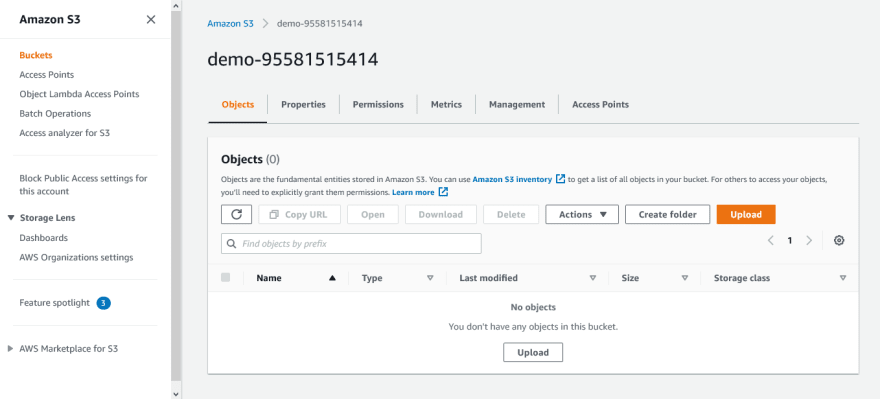
Click on **disable** for Bucket Versioning.

Then click on **Create bucket**.

[](https://res.cloudinary.com/practicaldev/image/fetch/s--gYTnSFey--/c_limit%2Cf_auto%2Cfl_progressive%2Cq_auto%2Cw_880/https:/user-images.githubusercontent.com/33374159/119146169-47ca9a00-ba42-11eb-9917-5c2fb5bb7802.png)

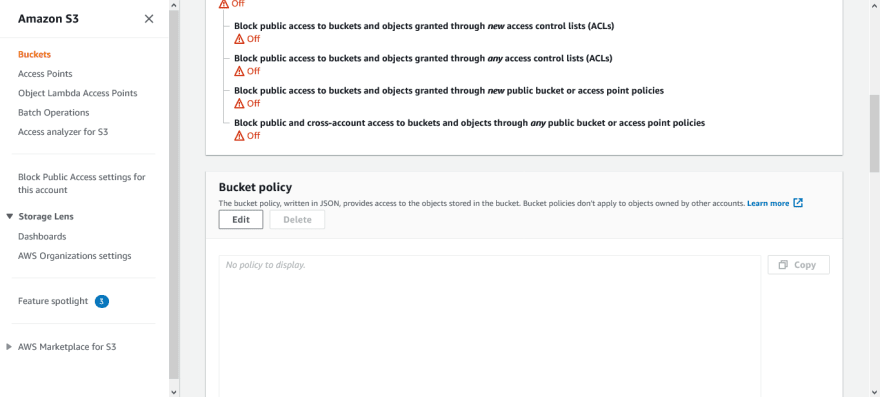
**Step 2 — Upload web files to S3 bucket**

After creating the bucket, you need to upload your website’s files and folders into it.

[](https://res.cloudinary.com/practicaldev/image/fetch/s--iW4-qjrh--/c_limit%2Cf_auto%2Cfl_progressive%2Cq_auto%2Cw_880/https:/user-images.githubusercontent.com/33374159/119146402-806a7380-ba42-11eb-903e-c24995bae41c.png)

**Step 3 — Secure S3 bucket through IAM policies**

From the **S3** dashboard, click on the **name** of the bucket, then click on **Permissions** tab. Scroll down to the **Bucket policy** section and click on its **Edit** button.

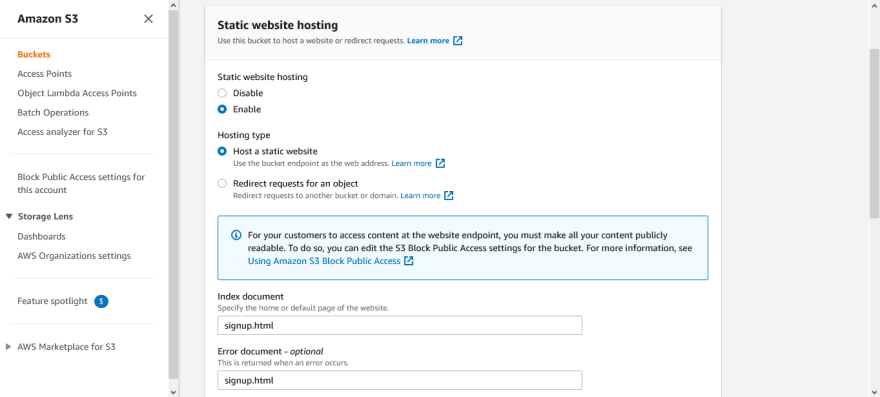
[](https://res.cloudinary.com/practicaldev/image/fetch/s--GjVVeV7M--/c_limit%2Cf_auto%2Cfl_progressive%2Cq_auto%2Cw_880/https:/user-images.githubusercontent.com/33374159/119147505-93ca0e80-ba43-11eb-8af4-25792cedf124.png)

**Step 4 — Configure S3 bucket**

click on the **Properties** tab.

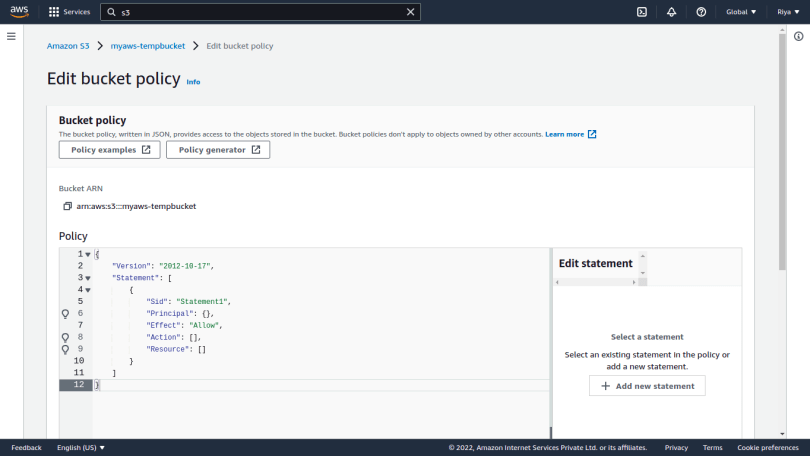
Scroll down to the **Static website hosting** section and click on its **Edit** button.

Select **Enable** for Static website hosting.

[](https://res.cloudinary.com/practicaldev/image/fetch/s--7YjwPae5--/c_limit%2Cf_auto%2Cfl_progressive%2Cq_auto%2Cw_880/https:/user-images.githubusercontent.com/33374159/119148824-cb858600-ba44-11eb-9556-73525506d83a.png)

**Step 4 —Bucket Policy**.

For hosting our website, we need to provide our own custom-defined policy.

[](https://i0.wp.com/blog.knoldus.com/wp-content/uploads/2022/01/Screenshot-from-2022-01-09-19-56-33.png?ssl=1)

Click on the **Policy Generator** option. This page will let you define your own policy required to host the website.

Step 1, select policy type as *S3 Bucket Policy*.

In Step 2,

— Select Effect as *Allow*.

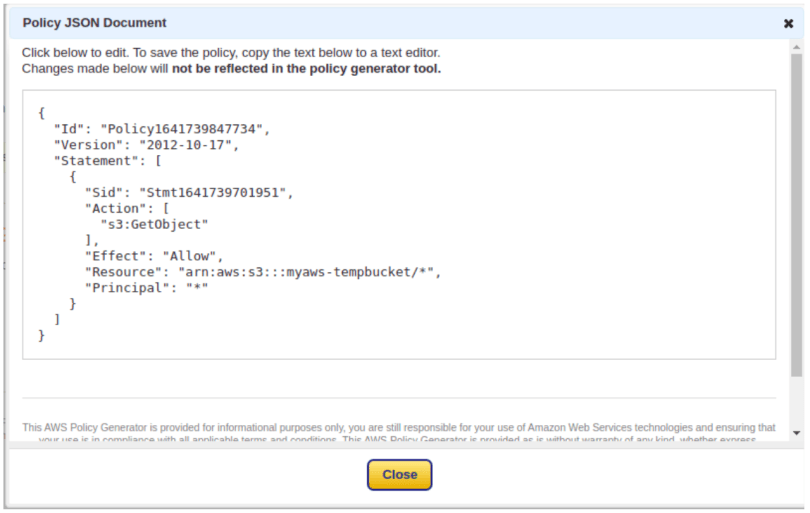
— Write \* in the Principal section.

— Under Actions, select *GetObject*.

— In the Amazon Resource Name (ARN), give the ARN of your bucket which you can find under the Properties section.

— Also, add a **/\*** in the end for the policy to be applied to every object in your bucket.

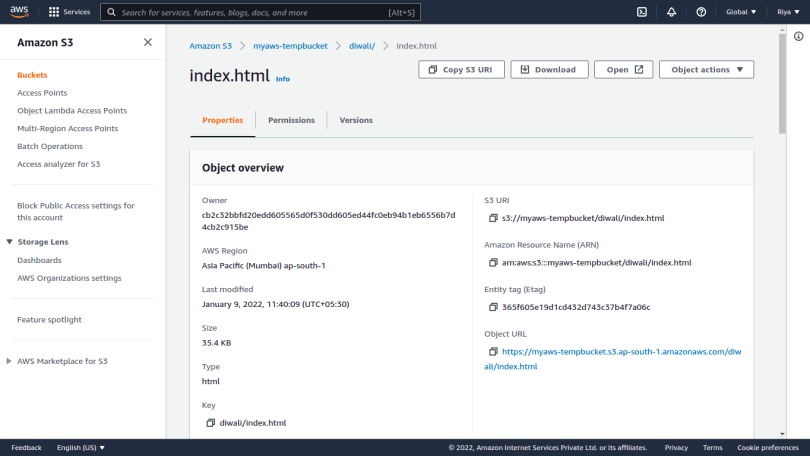
Finally in Step 3, click on Generate policy and you’ll see a page like this.

[](https://i0.wp.com/blog.knoldus.com/wp-content/uploads/2022/01/Screenshot-from-2022-01-10-19-19-45.png?ssl=1)

Now, copy this policy, paste it in the Edit Policy section overwrite the default one, and save your changes.

**Step 6 —** Get the link for your website

Just head over to your objects and select the home or default page of your website as specified earlier. You will see an object URL for your website as shown below.

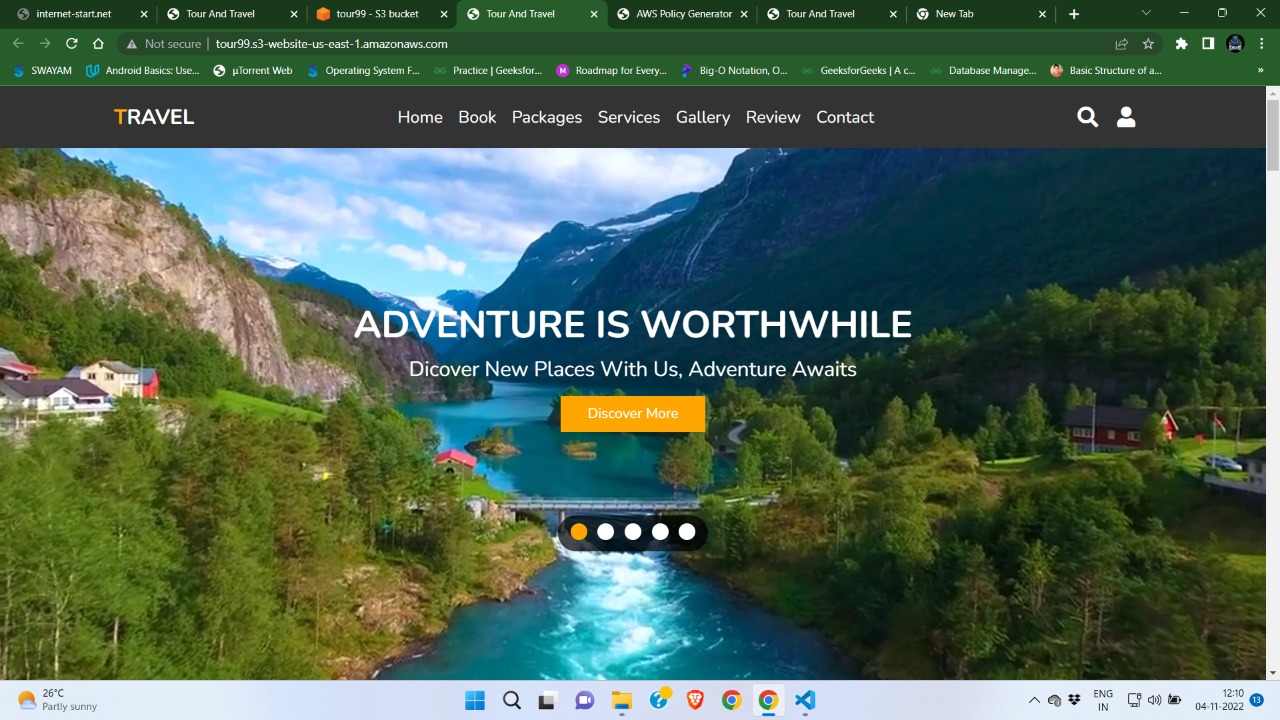
[](https://i0.wp.com/blog.knoldus.com/wp-content/uploads/2022/01/Screenshot-from-2022-01-09-20-28-25.png?ssl=1)

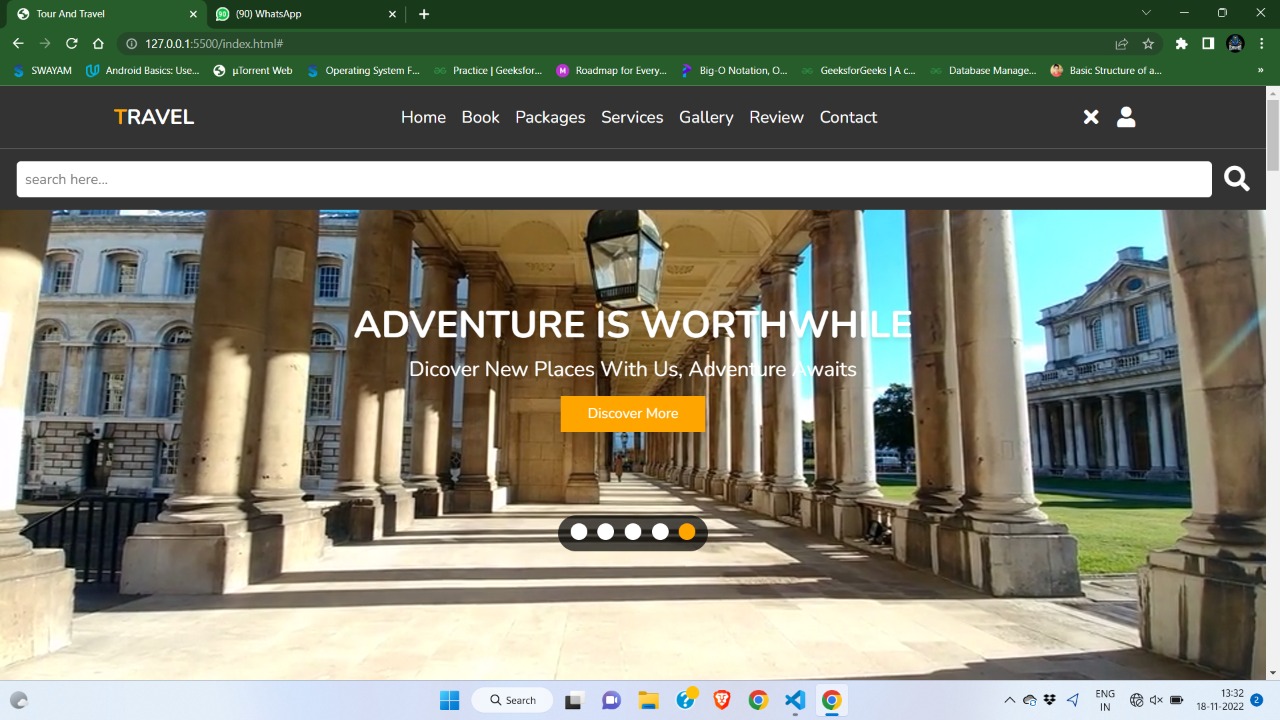
**CHAPTER 5**

**SAMPLE SCREENSHOTS**

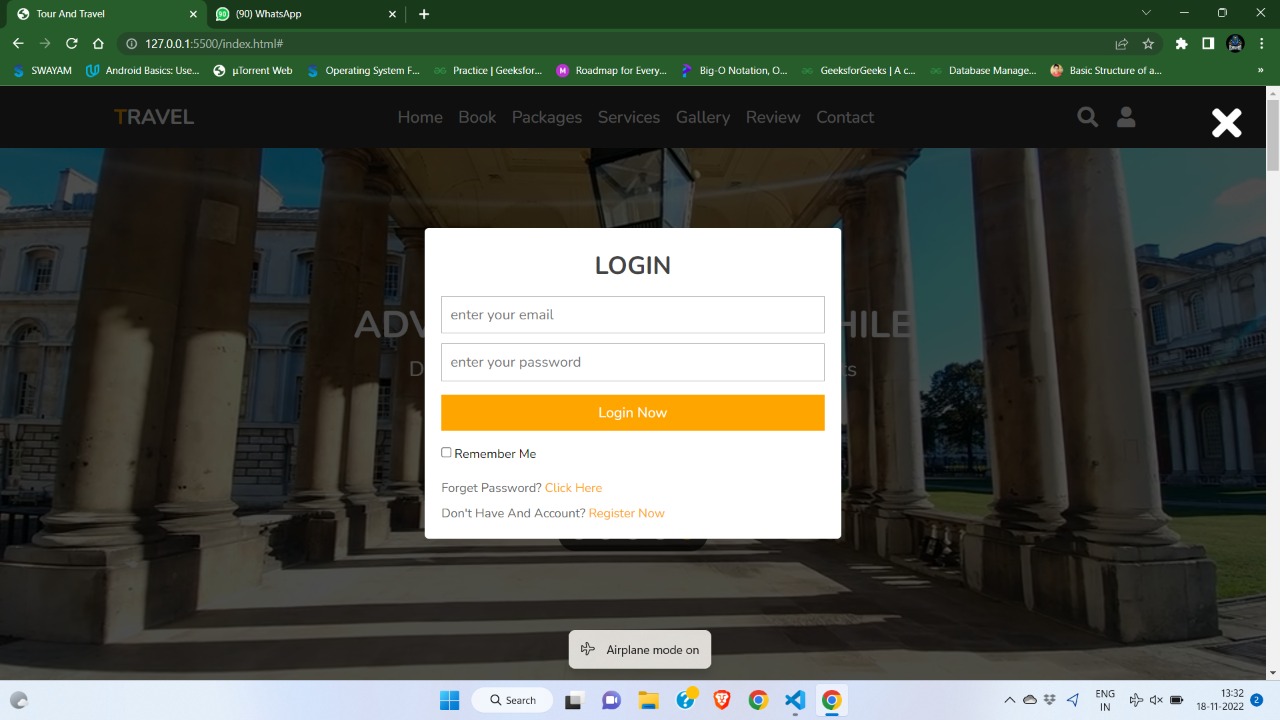
Fig. no. 1

**Home Page:-**

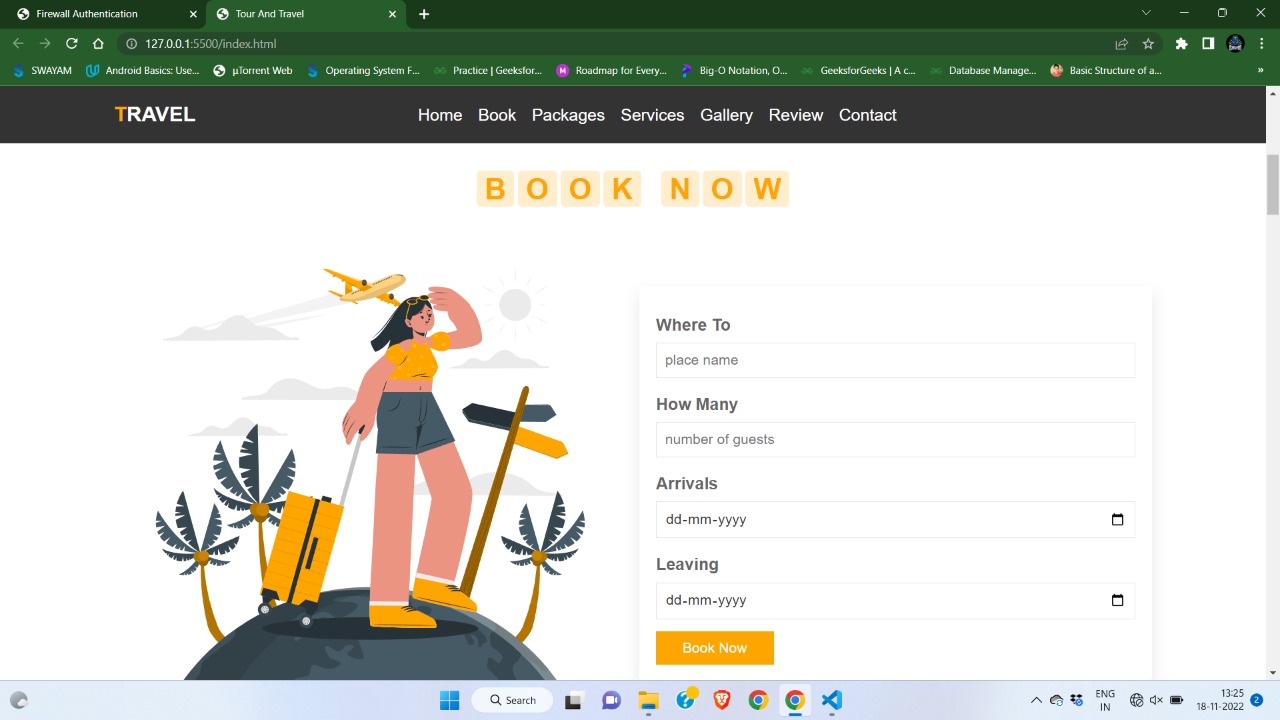
****



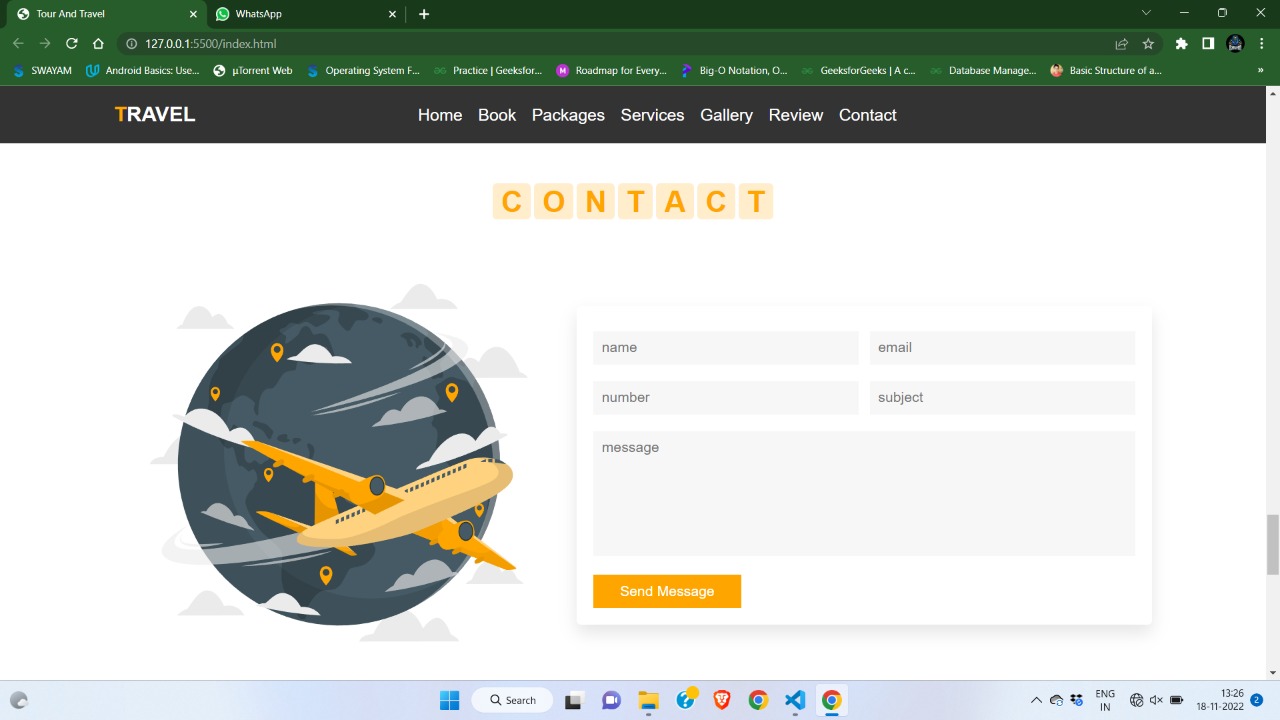
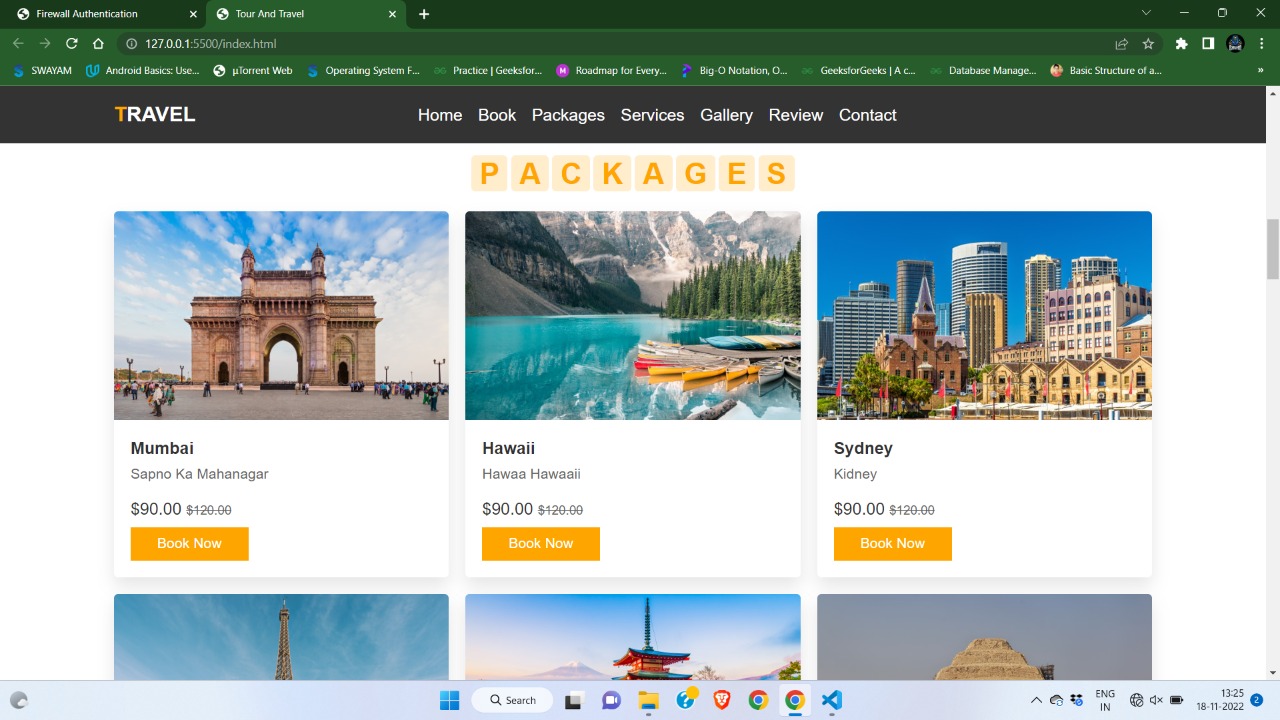
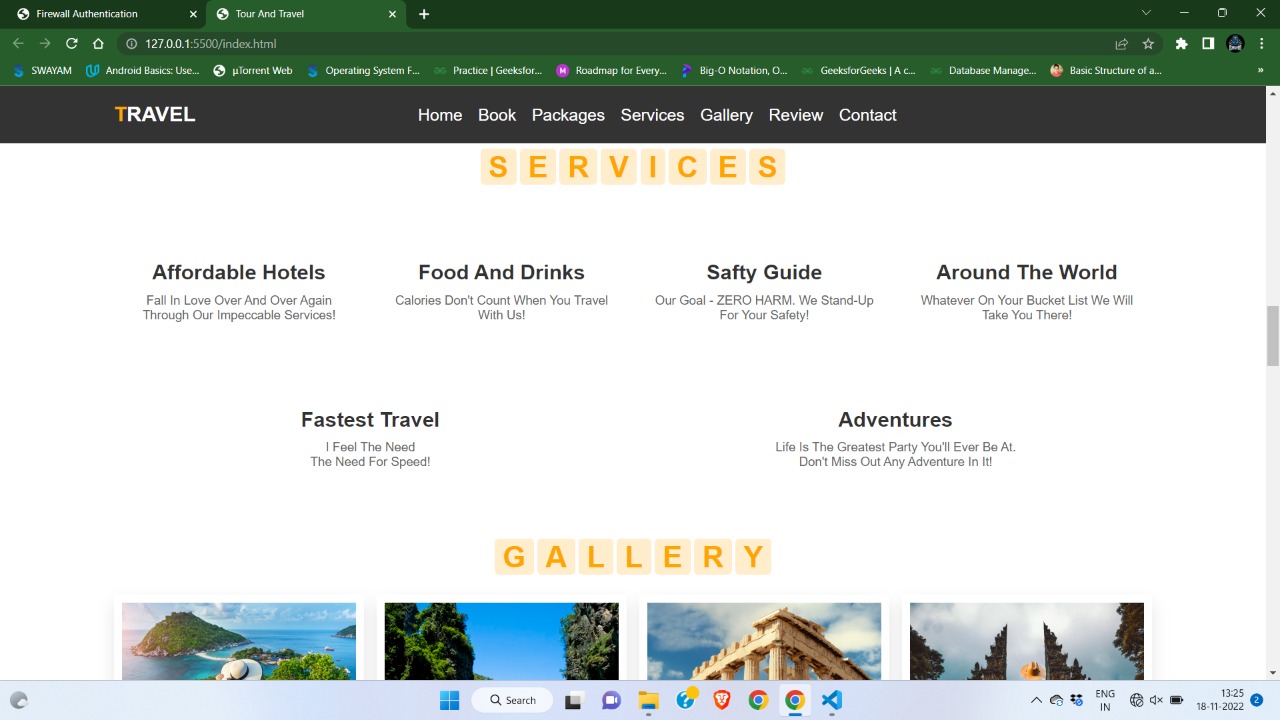
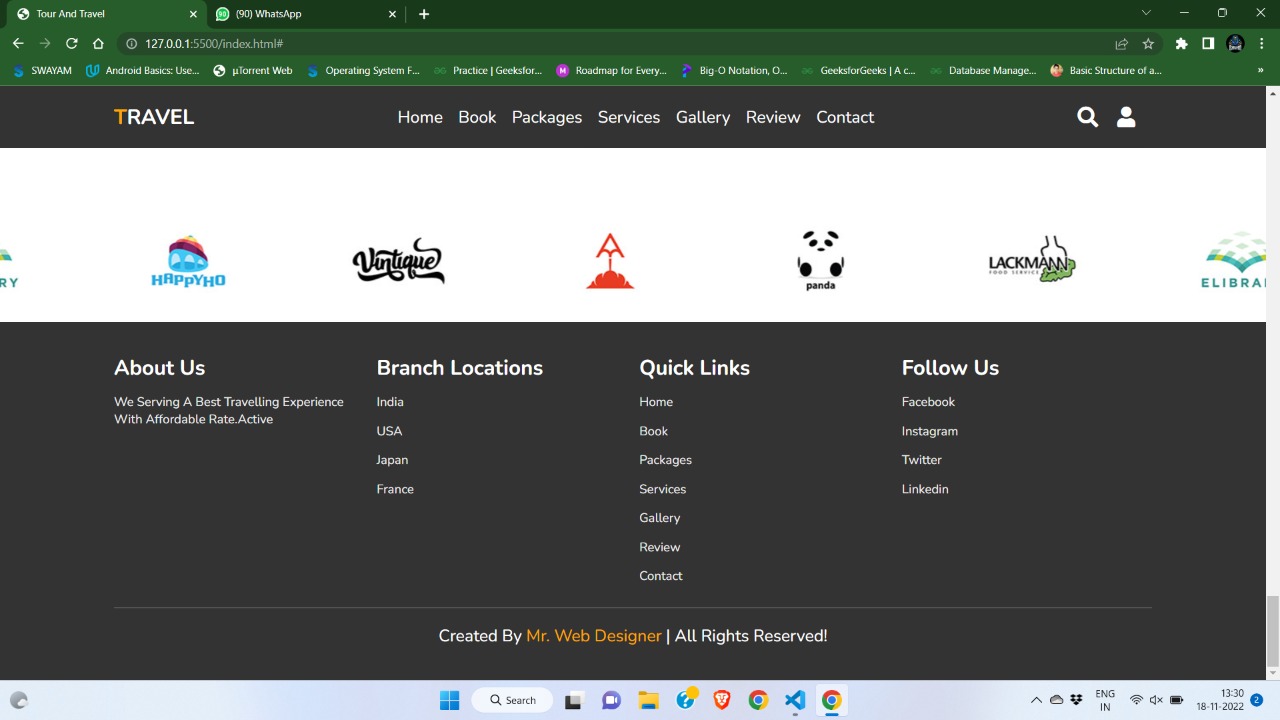
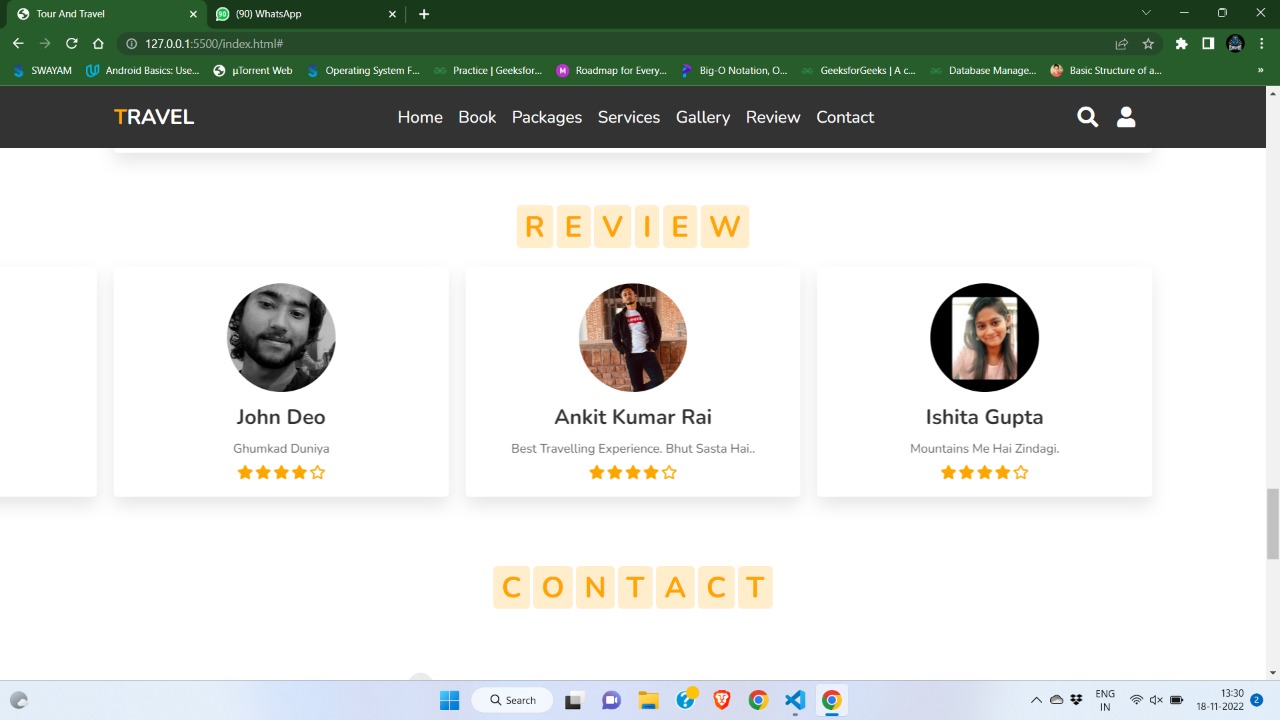
**LOGIN PAGE:**



**BOOKING PAGE:**

****

SEVICES

**** 

**CHAPTER 6**

**CONCLUSION**

We the blessings of "Goddess Sarasvati" We have accomplished our project and documentation. We are in the final session after a long period of thought, discussion, and implementation, and we are really pleased with the outcome.

By offering as many specialized ideas as possible, we hope to assist tourists in finding destinations and journeys that best suit their tastes. Users won't need to compromise on their preferences or considerations thanks to this website, which also eliminates the time-consuming job of conducting online research.

The choice of a handling agency makes the business run smoothly and profitably while also ensuring that tourists receive exceptional travel services. It is very important since the profitability of the travel industry rests much on how well the customers are treated while on the tour.

The advertised services and the visitors' actual experiences or sentiments line up favorably. As a result, while choosing a handling agency, the tour operator should consider the handling travel agency's expertise in the industry, the area of operation, reputation, credibility, competent personnel, credit facilities, and the competitive pricing.

You may benefit from running in AWS's security, scalability, cost, and agility with this AWS architecture. The top-notch infrastructure and security operations of AWS are advantageous for this design. The website is prepared for traffic surges thanks to Auto Scaling, so you are ready for product launches and viral websites. With AWS, there is no need to over-provision for peak capacity because you only pay for what you use. Additionally, because AWS services are scalable, you get greater agility. (Compare this to the conventional procedure, which might need weeks to provide servers, storage, or networking.) Because you don't have to manage the infrastructure, you have more time and money to invest in creating value that differentiates your company.

AWS upends conventional IT thinking and makes new "cloud-native" architectures possible. A contemporary static website may be built without using a single web server.

**CHAPTER 7**

**FUTURE SCOPE**

In a nutshell, it can be summarized that the future scope of the project circles around maintaining information regarding:

This type of software’s can be further extended –

* For generating reviews related to the tourist requirements.

• Can be used for generating reviews for the Online Videos provided on the software.

• Easy to find the nearby famous places, temples & monuments.

• Developer can be providing the update information of the places and also provide updates to the software for better serves.

• Provide offers for various places in budgets occasionally

The above-mentioned points are the enhancements which can be done to increase the applicability and usage of this project. Here we can maintain the records of our user. Also, as it can be seen that now-a-days the players are versatile, i.e., so there is a scope for introducing a method to maintain the Tour and travel system. Enhancements can be done to maintain all the Booking, Issues, Guides.

We have left all the options open so that if there is any other future requirement in the system by the user for the enhancement of the system then it is possible to implement them. In the last we would like to thanks all the persons involved in the development of the system directly or indirectly. We hope that the project will serve its purpose for which it is develop there by underlining success of process.

**CHAPTER 8**

**BIBLOGRAPHY**

1. Amazon Web Services in Action by Andreas Wittig and Michael Wittig
2. Programming Amazon Web Services: S3, EC2, SQS, FPS, and SimpleDB  by James Murthy
3. High Performance MySQL: Proven Strategies for Operating at Scale, Fourth Edition by Silvia Botros.
4. <https://docs.oracle.com/javase/8/javase-clienttechnologies.htm>
5. <https://docs.oracle.com/en-us/iaas/mysql-database/doc/getting-started.html>
6. <https://docs.oracle.com/en/java/>
7. <https://docs.aws.amazon.com/AmazonS3/latest/userguide/GetStartedWithS3.html>

**CHAPTER 9**

**REFERENCES**

[1]."How Cloud Computing Works", [Online]. Available: http://computer.howstuffworks.com/cloudcomputing/cloud-computing.htm

[2].Te-Shun C., 2013, "Security Threats on Cloud Computing Vulnerabilities", International Journal of Computer Science & Information Technology (IJCSIT) Vol 5, No 3.

[3]. Mosca, P., Zhang, Y., Xiao, Z. and Wang, Y. (2014) Cloud Security: Services, Risks, and a Case Study on Amazon Cloud Services. International Journal of Communications, Network and System Sciences, 7, 529-535.

[4]. Mutch, J., "How to Steal Data from the Cloud", 2010, [Online]. Available: <http://www.cloudbook.net/resources/stories/how-tosteal-data-from-the-cloud>

[5]. Archer J. (2010), “Top Threat to Cloud Computing”, Cloud Security Alliance. [Online]. Available: [https://cloudsecurityalliance.org/topthreats/csathreats.v 1.0.pdf](https://cloudsecurityalliance.org/topthreats/csathreats.v%201.0.pdf)

[6]. Kirchgaessner S. (2013), Cloud Storage Carries Potent Security Risk, [Online]. Available: http://www.ft.com/cms/s/0/4729ed7c-3722-11e3-9603- 00144feab7de.html

[7]. Simonite T (2012) “How to steal data from your neighbor in the cloud”. MIT Technology Review, 8 Nov 2012. [www.technologyreview.com/news/506976/howto-steal-data-from-your-neighbor-in-the-Cloud/](http://www.technologyreview.com/news/506976/howto-steal-data-from-your-neighbor-in-the-Cloud/)

[8]. Hanim. E. (2013), “Security threats and solutions in cloud computing,” in 2013 World Congress on Internet Security, WorldCIS 2013, 2013, pp. 139–143.

[9]. “AWS Market Share Reaches Five-Year High Despite Microsoft Growth Surge", (2015) [Online]. Available: <https://www.srgresearch.com/articles/aws-marketshare-reaches-five-year-high-despite-microsoft-growthsurge>

[10]. Sullivan B. (2016). “Amazon Web Services Public Cloud”, [Online]. Available: http://www.techweekeurope.co.uk/cloud/cloudmanagement/amazon-web-services-public-cloud185687 International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, Number 22 (2018) pp. 16077-16084 © Research India Publications. http://www.ripublication.com 16084

[11]. "Can AWS Microsoft reach 50 market share cloud", (2015), [Online]. Available: <http://marketrealist.com/2015/11/can-aws-microsoftreach-50-market-share-cloud/>

[12]. Cusumano M. 2010, “Cloud computing and SaaS as new computing platforms”. Communication. ACM 53, 4 (April 2010), 27-29. DOI: <https://doi.org/10.1145/1721654.1721667>

[13]. Ristenpart T., and Savage S., 2009, "Hey, you, get out of my cloud: exploring information leakage in third-party compute clouds", In Proceedings of the 16th ACM conference on Computer and communications security.

[14]. Haroon M. and Nick A., 2010, "Clobbering the cloud", defcon.org, Communication Inc, U.S.

[15]. Bleikertz S., Probst P. and Eriksson K 2010, "Security audits of multi-tier virtual infrastructures in public infrastructure clouds", In Proceedings of the 2010 ACM Workshop on Cloud Computing Security, p. 93

[16]. Balduzzi M., Zaddach J., Balzarotti D., Kirda E., 2012 "A Security Analysis of Amazon’s Elastic Compute Cloud Service", in Proceedings of the 27th Annual ACM Symposium on Applied Computing - SAC ’12, p. 1427.

[17]. Te Shun C., 2013, "Data Breach Trends & Stats, Symantec", In Proceedings of International Journal of Computer Science & Information Technology (IJCSIT) Vol 5, No 3.

[18]. "He has Delivered Her First Giant Data Breach" 2012, [Online]. Available at [http://www.infosecisland.com/blogview/19432-2012- Has-Delivered-Her-First-Giant-Data-Breach.html](http://www.infosecisland.com/blogview/19432-2012-%20Has-Delivered-Her-First-Giant-Data-Breach.html)

[19]. Al Awadhi E., Salah K., Marti T , 2013, "Assessing the Security of the Cloud Environment", in 2013 7th IEEE GCC Conference and Exhibition, GCC 2013, pp. 251– 256.

[20]. Chemerkin Y., 2013, “Limitations of Security Standards against Public Clouds”, Technical Co-Sponsored by IEEE Toronto Section i-Society 2013 Proceedings Contents, International Conference on Information Security.

[21]. Mosca P., Zhang Y., Xiao Z., Wang Y., 2014, “Cloud Security: Services, Risks, and a Case Study on Amazon Cloud Services”, International Journal of Communications, Network and System Sciences, Vol.7 No.12

[22]. Torkura K. and Meinel C., 2016, “Towards CloudAware Vulnerability Assessments ", In Proceedings of 11th International Conference on Signal Image Technology & Internet Based Systems, pp. 746–751.

[23]. Serrano N., Gallardo G., and Hernantes J., 2015, “Infrastructure as a Service and Cloud Technologies”, IEEE Softw., vol. 32, no. 2, pp. 30–36.

[24]. Narula S., Prachi M., Jain A., 2015, "Cloud Computing Security: AWS", In Proceedings of Fifth International Conference on Advanced Computing & Communication Technologies.

[25]. "Nessus", [Online] Available: <http://www.tenable.com/products/nessus/select-youroperating-system>

[26]. "ClamAV", [Online] Available: <https://www.clamav.net/downloads>

[27]. "How to secure your Amazon EC2", Amazon Inc. about Amazon web services, [Online] Available: <http://aws.amazon.com/articles/1233/>

[28]. "OpenVas", [Online] Available: <http://www.openvas.org/>

[29]. "Zeus Malware", [Online] Available: <https://en.wikipedia.org/wiki/Zeus>

[30]. Bracci F., Corradi A. and Foschini L., 2012, "Database security management for healthcare SaaS in the Amazon AWS Cloud," 2012 IEEE Symposium on Computers and Communications (ISCC), Cappadocia, pp. 000812- 000819.

[31]. Radford C.J., 2014, “Challenges and solutions protecting data within Amazon Web Services, Network Security”, Volume 2014, Issue 6, June 2014, Pages 5-8, ISSN 1353-4858, http://dx.doi.org/10.1016/S1353- 4858(14)70058-3.

**CHAPTER 10**

**APPENDIX**

**Appendix A: Project Reflection**

There is a proverb says” A team work leads to dream work”. In according to us a well-maintained group can accomplish their object point no matter how difficult it is. We were synchronized, completely we are in agreement and we knew our job well. This mentality of our group member helped a lot in doing this project.

When we first start our work, we were so cocksure and full of spirit but after some days we felt undone as we were struggling about from where to start. And when we find our purposes and got our starting point we were never looked back and keep moving forward. Whenever a fateful point came, we solved it together by group discussion.

Admittedly, the formation of a team is the most important. You need to enroll the right members who have interest in this field and a hard worker and then made a high-performance team. It was our privilege and good luck that we were to be put in the same group.

Finally, after so many difficulties and mass sleepless night, we did this project successfully