

PROFESSIONAL TRAINING REPORT
at
Sathyabama Institute of Science and
Technology (Deemed to be University)

Submitted in partial fulfillment of the requirements for
the award of Bachelor of Technology in Information
Technology

By

Pallavi Kumari
REG. NO. 39120078



DEPARTMENT OF INFORMATION TECHNOLOGY

SCHOOL OF COMPUTING

SATHYABAMA INSTITUTE OF SCIENCE AND
TECHNOLOGY

JEPPIAAR NAGAR, RAJIV GANDHI SALAI,
CHENNAI – 600119, TAMILNADU

DECLARATION

I, **PALLAVI KUMARI** hereby declare that the project report entitled STATIC Website Hosting **Using Amazon Web Service S3** done by me under the guidance of **Dr. S. Gowri** is submitted in partial fulfillment of the requirements for the award of Bachelor of Technology Degree in Information Technology.

DATE:

PLACE:

**SIGNATURE OF THE CANDIDATE
PALLAVI KUMARI**

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management of SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T. Sasikala M.E., Ph.D, Dean**, School of Computing, **Dr. R. Subhashini, M.E., Ph.D. Head of the Department of Information Technology** for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Dr. S. Gowri**, for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of INFORMATION TECHNOLOGY** who were helpful in many ways for the completion of the project.

TRAINING CERTIFICATE



Elite

NPTEL Online Certification

(Funded by the Ministry of HRD, Govt. of India)



This certificate is awarded to

PALLAVI KUMARI

for successfully completing the course



Cloud Computing

with a consolidated score of **76** %

Online Assignments	23.96/25	Proctored Exam	51.82/75
--------------------	----------	----------------	----------

Total number of candidates certified in this course: **3189**

Prof. G P Raja Sekhar
Dean, Continuing Education
IIT Kharagpur

Aug-Oct 2021
(8 week course)

Prof. Debjani Chakraborty
Coordinator, NPTEL
IIT Kharagpur



Indian Institute of Technology Kharagpur



Roll No:NPTEL21CS62S14160154

To validate and check scores: <https://nptel.ac.in/noc>

ABSTRACT

EXAM HUNT Website is an approach towards those students who just miss the opportunity due to lack of information about various examinations conducted in all over the India.

This website will provide about almost every competitive/Scholarship/Olympiad examination conducted by state and central government in India on a single platform.

Project website is going to be static and deployed in cloud which is the platform provided by Amazon via the Amazon Web Service S3 service.

At the current time Cloud Computing is the solution of many of the problems faced by user and organization, be it in the form of highly flexible provisioning or near perfect service uptime guarantee

I have selected Cloud to host my website because of cloud is on the rise, year after year. After all, it is a simple solution that drives improved collaboration, productivity, security, and in turn, revenue.

LIST OF FIGURES

FIGURE NO.	FIGURE NAME	PAGE NO.
<i>Fig 1.1:</i>	<i>Create a website</i>	<i>4</i>
<i>Fig 1.2:</i>	<i>Make Website responsive and adaptive</i>	<i>4</i>
<i>Fig 1.3:</i>	<i>AWS S3 console</i>	<i>5</i>
<i>Fig 1.4:</i>	<i>Creating bucket in AWS S3 console</i>	<i>5</i>
<i>Fig 1.5:</i>	<i>Creating bucket in AWS S3 console</i>	<i>6</i>
<i>Fig 1.6:</i>	<i>Configuring public access</i>	<i>7</i>
<i>Fig 1.7:</i>	<i>Uploading files to bucket</i>	<i>8</i>
<i>Fig 1.8:</i>	<i>Enabling static hosting</i>	<i>9</i>
<i>Fig 1.9:</i>	<i>Properties section of packages</i>	<i>10</i>
<i>Fig 2.1:</i>	<i>Enabling bucket public access</i>	<i>10</i>
<i>Fig 2.2:</i>	<i>Frontpage(index) of hosted website</i>	<i>11</i>
<i>Fig 2.3:</i>	<i>Frontpage(index) of hosted website</i>	<i>12</i>

Table Of Content

CHAPTER	TITLE	PAGE NO
	ABSTRACT	i
	LIST OF ABBREVIATION	ii
01	INTRODUCTION	1
02	AIM AND SCOPE OF PRESENT INVESTIGATION	2
	2.1 AIM	2
	2.2 SCOPE	2
03	MATERIALS AND METHODS USED	3
	3.1 Working of AWS (S3)	3
04	STEPS TO HOST A WEBSITE	4
	Step 1	5
	Step 2	6
05	SYSTEM DESIGN	7
	5.1 FLOWCHART DIAGRAMS	8-10
06	RESULT AND DISCUSSION	11
07	CONCLUSION	13
08	References	14

CHAPTER 1: INTRODUCTION

We can define computing to mean any goal-oriented activity requiring, benefiting from, or creating computers.

Thus, computing includes designing and building hardware and software systems for a wide range of purposes, processing, structuring and managing various kinds of information, doing scientific studies using computers, making computer systems behave intelligently, creating and using communications and entertainment media; finding and gathering information relevant to any particular purpose, and so on. The list is virtually endless, and the possibilities are vast.

So going with the flow and considering the need/our useCase (that is to host a website to be accessible to the public) here we are going to talk about the usage of AWS S3 for the same.

STaaS lets you treat storage as OpEx. You sign a service level agreement (SLA) with your STaaS provider and pay for storage and data transfer rates (e.g., cost per gigabyte). Best of all, this whole process is automated, allowing you to scale your storage needs up and down as demand requires while maintaining performance and availability 24/7.

CHAPTER 2: AIM AND SCOPE OF THE PRESENT INVESTIGATION

As we know Storage as a service (STaaS) is a data storage business model where a provider rents storage resources to a customer through a subscription. STaaS saves you money through operating expenditure (OpEx) agility—you only pay for the storage you need, when you need it. Here using the same benefits in mind, we are going to use the service provided by Amazon via its AWS S3, which have promising SLA that assure service uptime up to 99.9999%.

Even after going through all this, one might ask as to “Why Use Storage as a Service?”. Buying new storage capacity can be an expensive capital expenditure (CapEx), especially if you aren’t sure how much capacity you’ll need in the future. You can try to predict the growth of your business and purchase with the future in mind, but it can tie up financial resources that might have more impact elsewhere in your business.

Fortunately, there’s no shortage of major tech companies with large data centers that are willing to sell their excess capacity. For these businesses, storage is just another service that’s part of their expansive product offerings, and they’re more than happy to absorb the expenses of managing, upgrading, and maintaining large-scale storage area networks (SANs). Amazon Web Services (AWS), Microsoft Azure, Google Cloud, and Oracle Cloud are all examples of major cloud storage providers with STaaS subscription options.

CHAPTER 3: MATERIALS AND METHODS USED

Amazon Web Services S3 is the premier platform for hosting static files. AWS makes it easy to upload full websites - plain HTML/CSS/JS and React/Vue/Angular projects alike - and serve them to the public, similar to how Apache, NGINX, and other web server work.

WORKING OF AWS S3

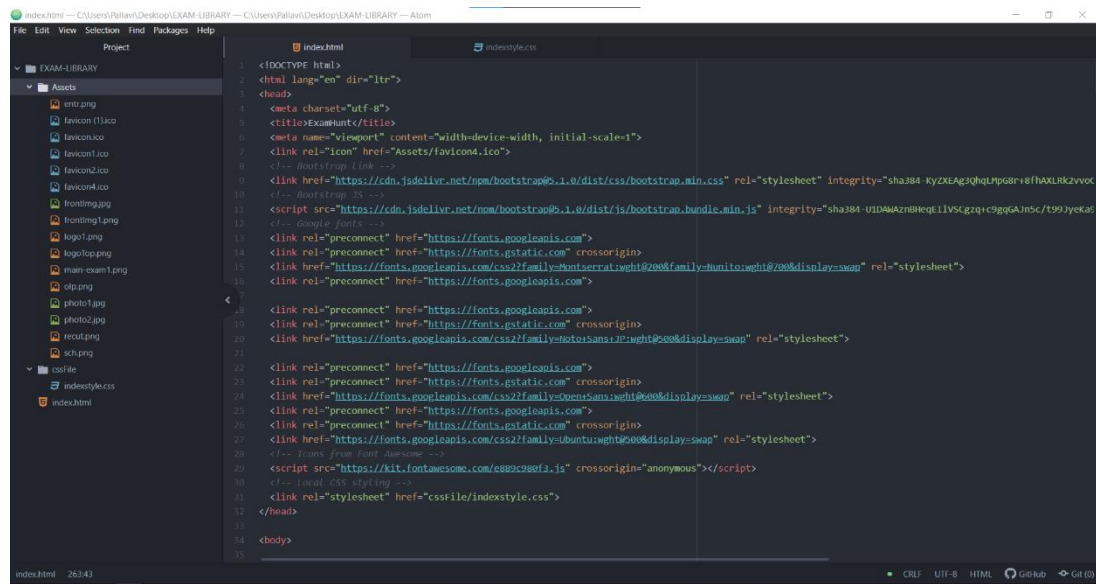
S3 works by allowing users to define buckets, collections of files that can be accessed at the same URI. Each bucket has its own set of access rules that determine who is able to access and upload files to it, as well as other properties like encryption and version control. S3 also allows you to replicate buckets across data centers, which adds redundancy and reduces access times for users in distant regions.

S3 buckets are most commonly used to store static content, such as images, videos, and documents. Many sites use S3 to store user-uploaded content, and most web server frameworks have plugins that enable easy access to S3 for that purpose. We can also use S3 to serve static sites, or sites that do not rely on a web server to dynamically generate HTML files.

CHAPTER 4: STEPS TO HOST A WEBSITE

STEP 1: Create a website

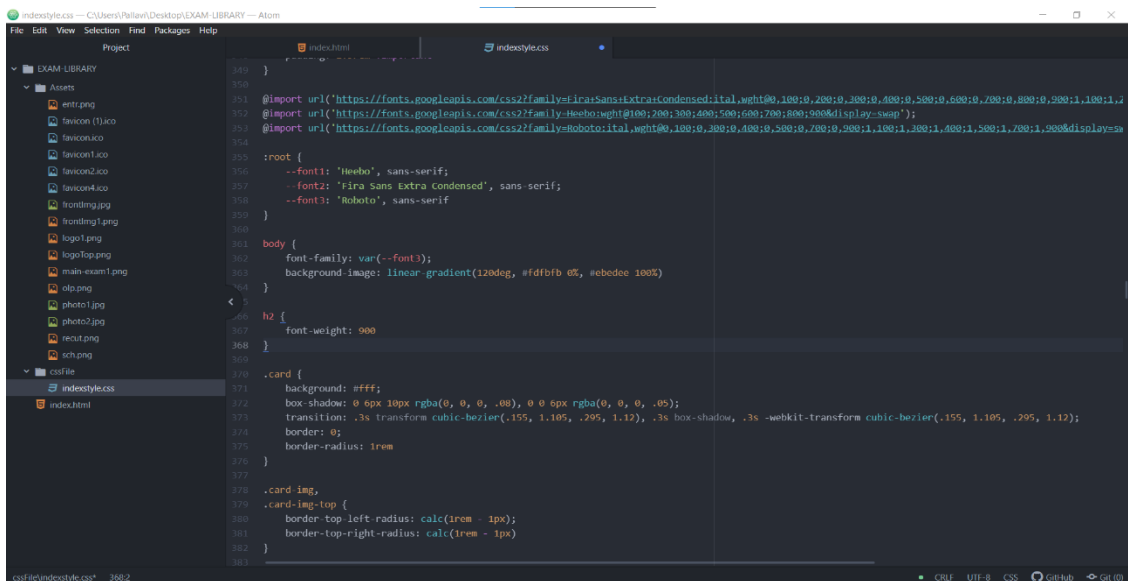
✳ The first step is to create a website using HTML, CSS, Bootstrap, JavaScript and other assets.



```
1 <!DOCTYPE html>
2 <html lang="en" dir="ltr">
3 <head>
4   <meta charset="utf-8">
5   <title>Exam Hunt</title>
6   <meta name="viewport" content="width=device-width, initial-scale=1">
7   <link rel="icon" href="Assets/favicon.ico">
8   <!-- Bootstrap Link -->
9   <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.0/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-KyZKEAg3qHUP68+8fHxLHK2vvoC"
10   <!-- Bootstrap JS -->
11   <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.0/dist/js/bootstrap.bundle.min.js" integrity="sha384-UdDAZnBHeq11VSCgzi+C9q4Jn5c/199jyetas"
12   <!-- Google Fonts -->
13   <link rel="preconnect" href="https://fonts.googleapis.com">
14   <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
15   <link href="https://fonts.googleapis.com/css2?family=Montserrat:wght@200&family=Ubuntu:wght@200&display=swap" rel="stylesheet">
16   <link rel="preconnect" href="https://fonts.gstatic.com">
17   <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
18   <link href="https://fonts.googleapis.com/css2?family=Roboto:wght@400&family=Ubuntu:wght@400&display=swap" rel="stylesheet">
19   <link rel="preconnect" href="https://fonts.gstatic.com">
20   <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
21   <link href="https://fonts.googleapis.com/css2?family=Roboto:wght@400&family=Ubuntu:wght@400&display=swap" rel="stylesheet">
22   <link rel="preconnect" href="https://fonts.gstatic.com">
23   <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
24   <link href="https://fonts.googleapis.com/css2?family=Roboto:wght@400&family=Ubuntu:wght@400&display=swap" rel="stylesheet">
25   <link rel="preconnect" href="https://fonts.gstatic.com">
26   <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
27   <link href="https://fonts.googleapis.com/css2?family=Roboto:wght@400&family=Ubuntu:wght@400&display=swap" rel="stylesheet">
28   <!-- Icons from Font Awesome -->
29   <script src="https://kit.fontawesome.com/c889c98f3.js" crossorigin="anonymous"></script>
30   <!-- Local CSS styling -->
31   <link rel="stylesheet" href="cssfile/indexstyle.css">
32 </head>
33 <body>
```

Fig 1.1: Code for Web Page (Atom Text Editor)

✳ Make Website responsive and adaptive



```
349 }
350
351 @import url('https://fonts.googleapis.com/css2?family=Heebo:wght@100;200;300;400;500;600;700;800;900;1.100;1.2
352 @import url('https://fonts.googleapis.com/css2?family=Heebo:wght@100;200;300;400;500;600;700;800;900&display=swap');
353 @import url('https://fonts.googleapis.com/css2?family=Roboto:ital,wght@0;100;200;300;400;500;600;700;800;900;1.100;1.200;1.300;1.400;1.500;1.700;1.900&display=sw
354
355 :root {
356   --font1: 'Heebo', sans-serif;
357   --font2: 'Pira Sans Extra Condensed', sans-serif;
358   --font3: 'Roboto', sans-serif;
359 }
360
361 body {
362   font-family: var(--font1);
363   background-image: linear-gradient(120deg, #fdfbfb 0%, #ebedee 100%);
364 }
365
366 h2 {
367   font-weight: 900;
368 }
369
370 .card {
371   background: #fff;
372   box-shadow: 0 6px 10px rgba(0, 0, 0, .08), 0 0 6px rgba(0, 0, 0, .05);
373   transition: .3s transform cubic-bezier(.155, 1.105, .295, 1.12), .3s box-shadow, .3s -webkit-transform cubic-bezier(.155, 1.105, .295, 1.12);
374   border: 0;
375   border-radius: 1rem;
376 }
377
378 .card-img,
379 .card-img-top {
380   border-top-left-radius: calc(1rem - 1px);
381   border-top-right-radius: calc(1rem - 1px);
382 }
383
```

Fig 1.2: CSS Code to make website Adaptive

STEP 2: Create an S3 bucket

When deploying website on cloud the first step is to create and name your bucket.

✳ In the AWS console, navigate to the S3 dashboard.

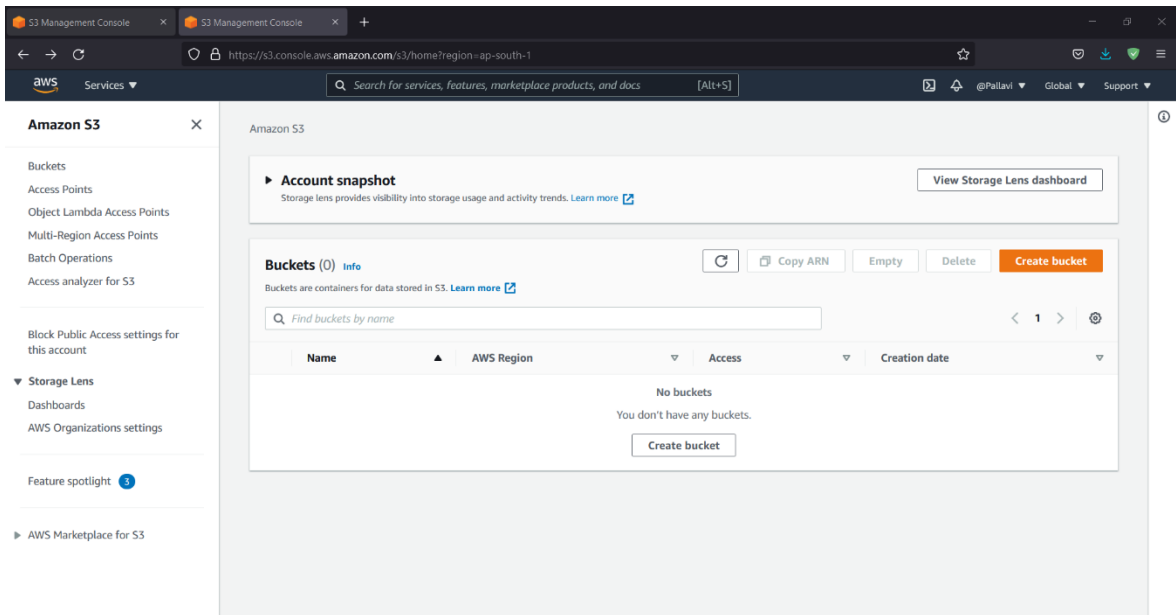


Fig 1.3: AWS S3 console

✳ In the Buckets tab, click Create bucket.

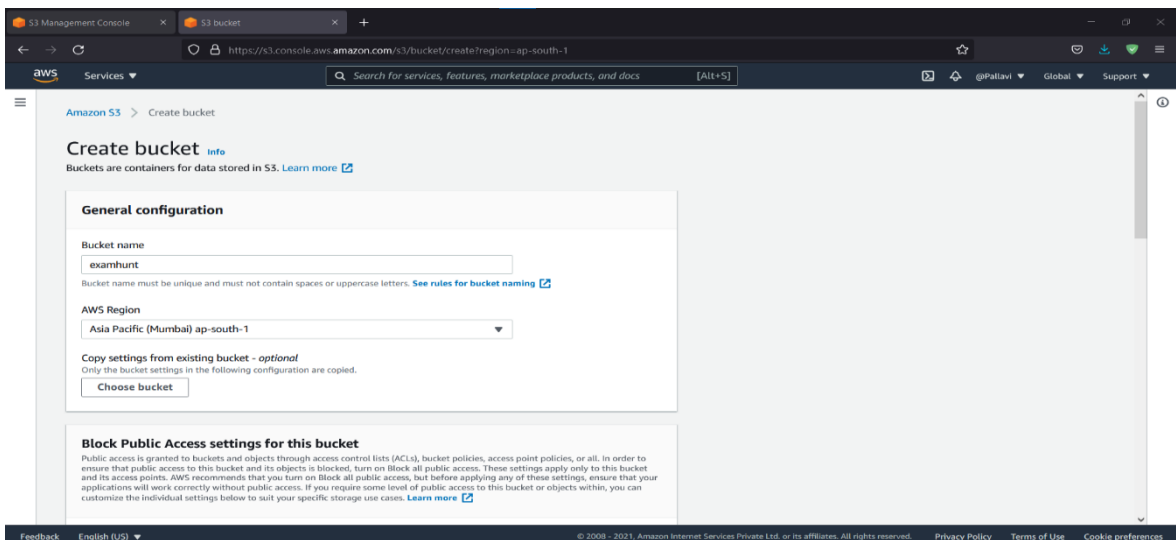


Fig 1.4: Creating bucket in AWS S3 console

✳ Give your bucket a name. This name must be unique across all S3 buckets in your partition (a grouping of regions: aws, aws-cn, or aws-us-gov). If you intend to attach a custom domain to your site later via Route 53, this name needs to be identical to the root domain of the site (e.g. examhunt.com).

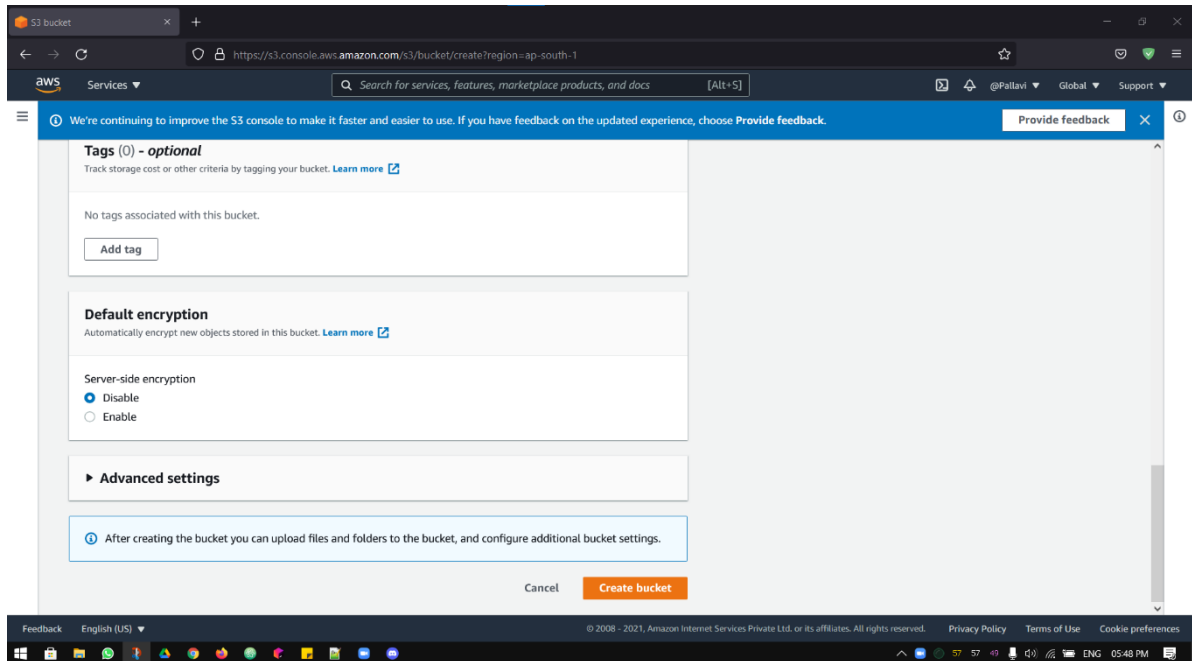


Fig 1.5: Creating bucket in AWS S3 console

Next, we'll discuss settings for restricting public access to your bucket. By default, S3 blocks all unauthorized access to your bucket to ensure the privacy of your data. However, this will make our static site inaccessible to regular users.

✳ Uncheck Block all public access and accept the warning that displays below. This gives us finer control over how public access can be granted.

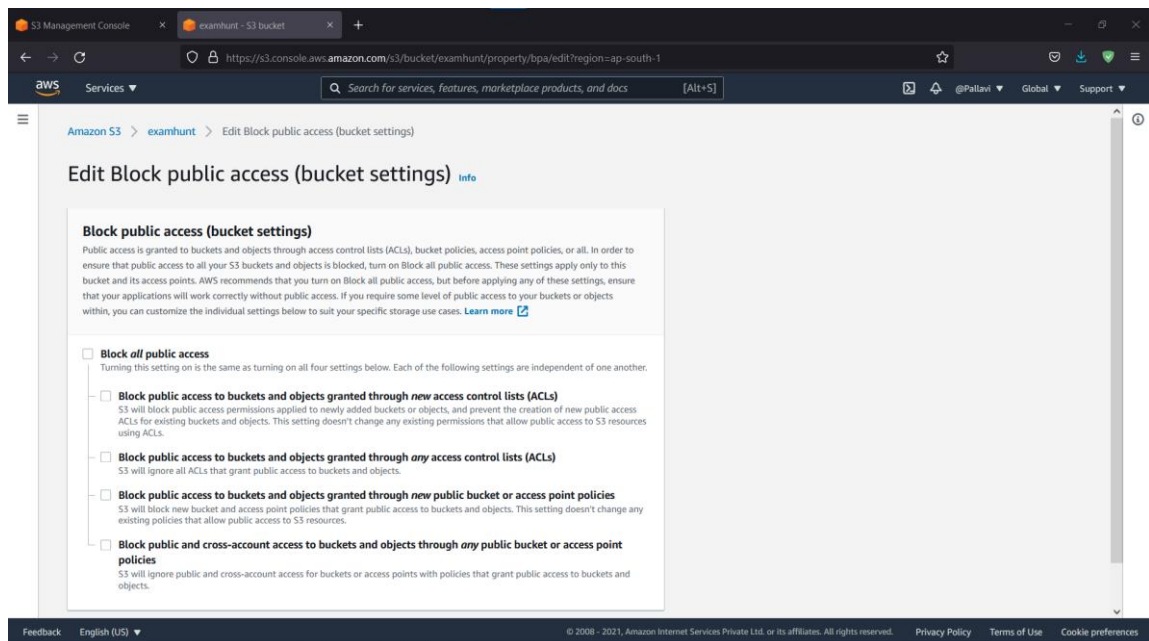


Fig 1.6: Configuring public access

- ✳ Check Block public access to buckets and objects granted through new access control lists (ACLs) and Block public access to buckets and objects granted through any access control lists (ACLs). These settings restrict the use of access control lists, which grant read/write permissions to specific AWS users. However, we will not be using this feature because we want to grant access to users not authenticated with AWS.
- ✳ Leave Block public access to buckets and objects granted through new public bucket or access point policies and Block public and cross-account access to buckets and objects through any public bucket or access point policies unchecked. We will be adding a new bucket policy later to grant public access to this bucket.
- ✳ Click the Create button at the bottom of the screen to create your new bucket.

STEP 3: Upload web packages to the Bucket

The next step is to upload the static site files that define the website into the S3 bucket.

❖ Steps for performing the upload to AWS S3:

✧ In the S3 dashboard, click on your new bucket to navigate to its overview page.

✧ In Objects, click Upload. In Files and Folders, upload your project files, then click Upload.

✧ Once everything is done uploading, click Exit to return to the bucket dashboard.

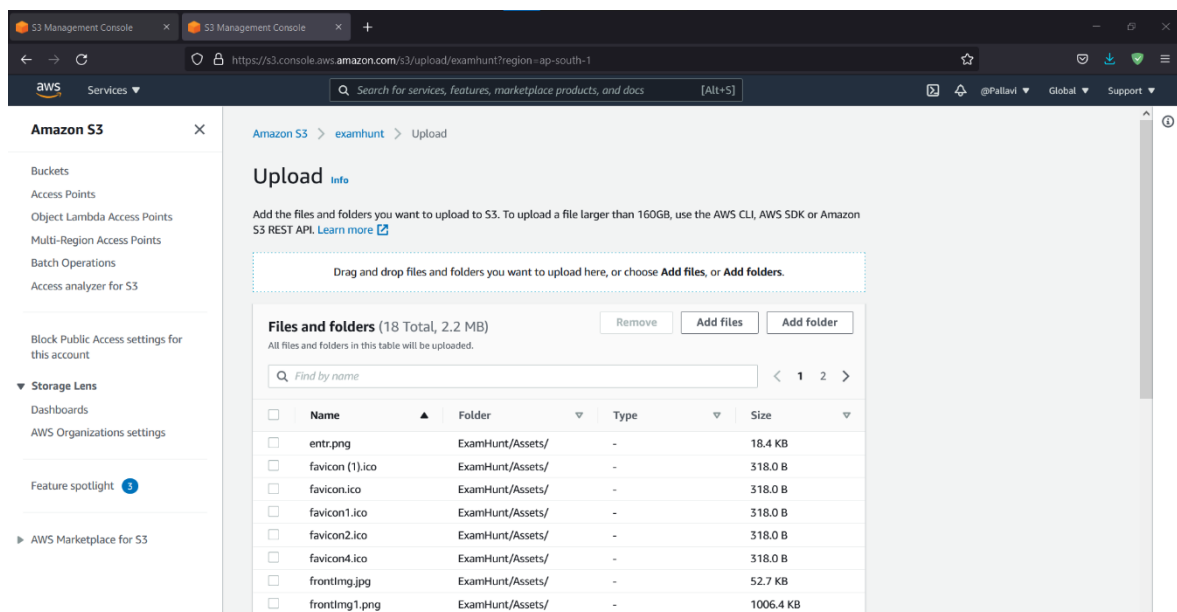


Fig 1.7: Uploading files to bucket

STEP 4: Configure the bucket for static website hosting

We will perform some additional configuration to configure the bucket specifically for static site hosting, as opposed to standard hosting for static files.

- ※ In your bucket's dashboard page, click on the Properties tab.
- ※ Scroll down to Static website hosting and click Edit.
- ※ Select Enable, then "Host a static website". Write index.html or your existing project's index page as your index document.

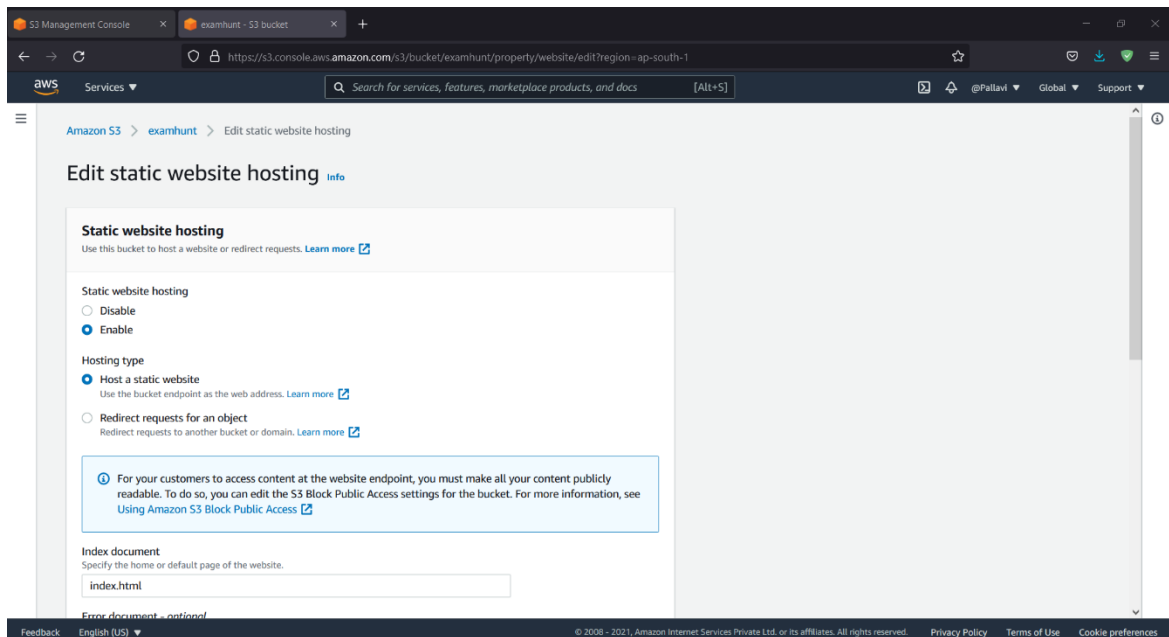


Fig 1.8: Enabling static hosting

- ※ If you are using an existing project, you can specify an existing error document as well. If not, write index.html as your error document for testing purposes.
- ※ Click Save changes to publish your website.

In the **Properties** tab again, we will see your site's URL listed in the **Static website hosting** section. Navigating here will return a **403 Forbidden** error, since we have not made our files fully public yet.

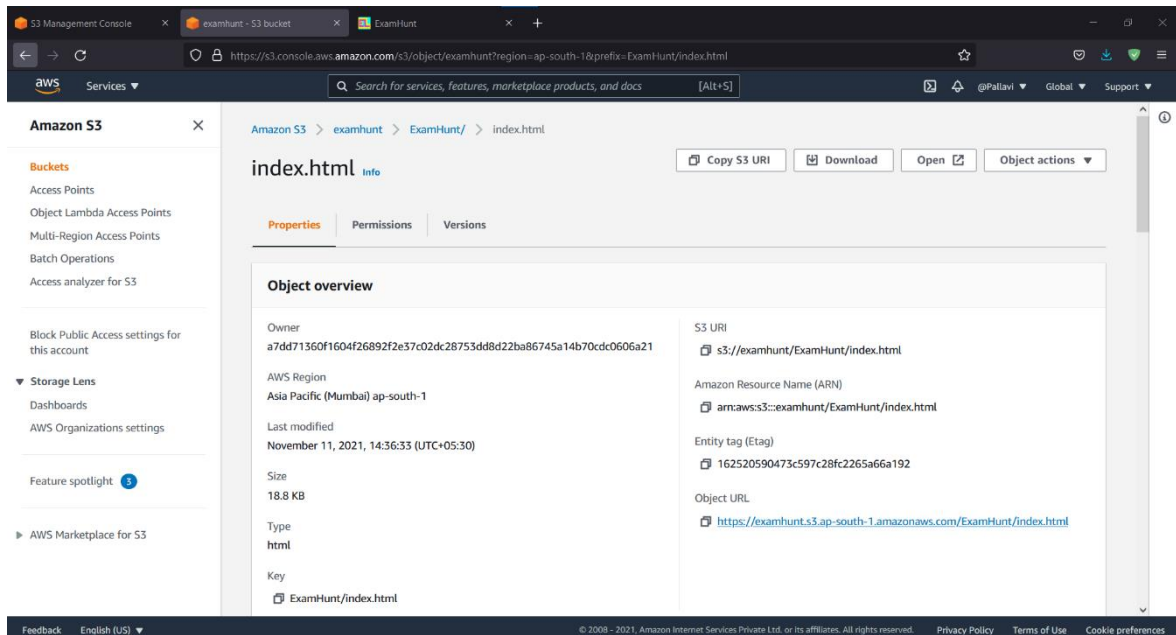


Fig 1.9: Properties section of packages

STEP 5: Configure the bucket for static website hosting

✳ We'll finalize the security configuration so that our static site is accessible to the public.

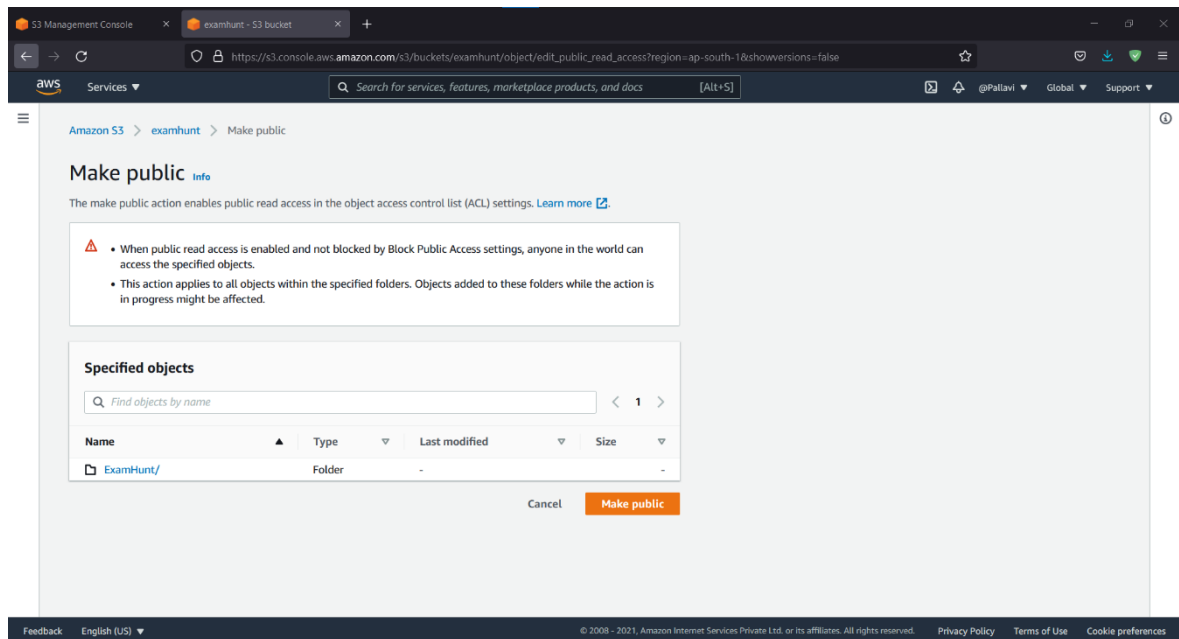


Fig 2.1: Enabling bucket public access

CHAPTER 5: RESULT AND DISCUSSION

After following through all the above process and instruction we finally will be able to access the live static website through the public link generated.

For our specific one it can be accessed through [HERE](#) 

Which when accessed will look like this

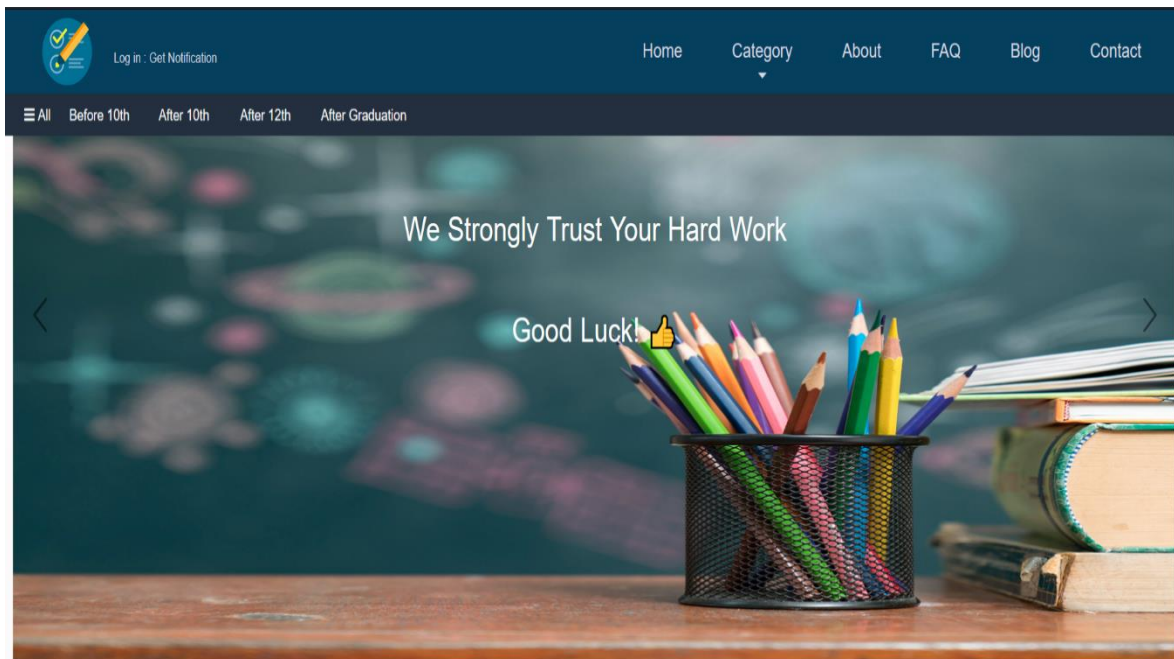


Fig 2.2: Frontpage(index) of hosted website

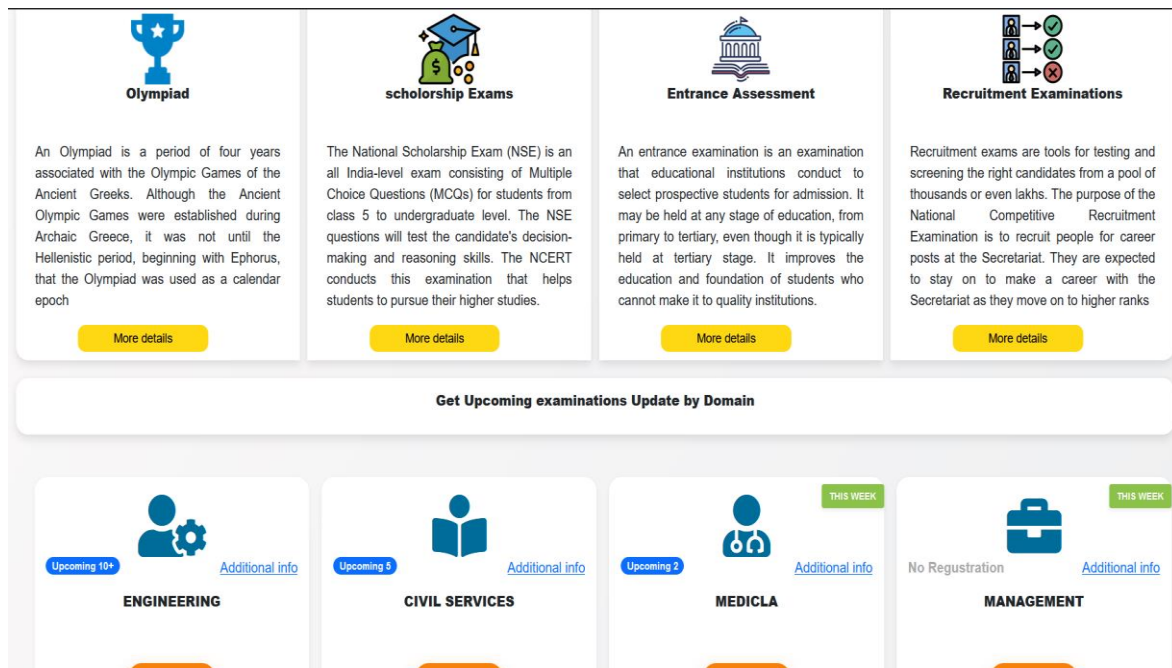


Fig 2.3: Frontpage(index) of hosted website

CHAPTER 6: SUMMARY AND CONCLUSION

There are numerous architectural and conceptual considerations when you are contemplating migrating your web application/ website to the AWS Cloud. The benefits of having a cost-effective, highly scalable, and fault-tolerant infrastructure that grows with your business far outstrips the efforts of migrating to the AWS Cloud.

References

Cloud service portal

- <https://aws.amazon.com/websites/>
- https://en.wikipedia.org/wiki/Cloud_computing
- <https://www.techtarget.com/searchcloudcomputing/definition/cloud-computing>
- <https://aws.amazon.com/what-is-cloud-computing/>