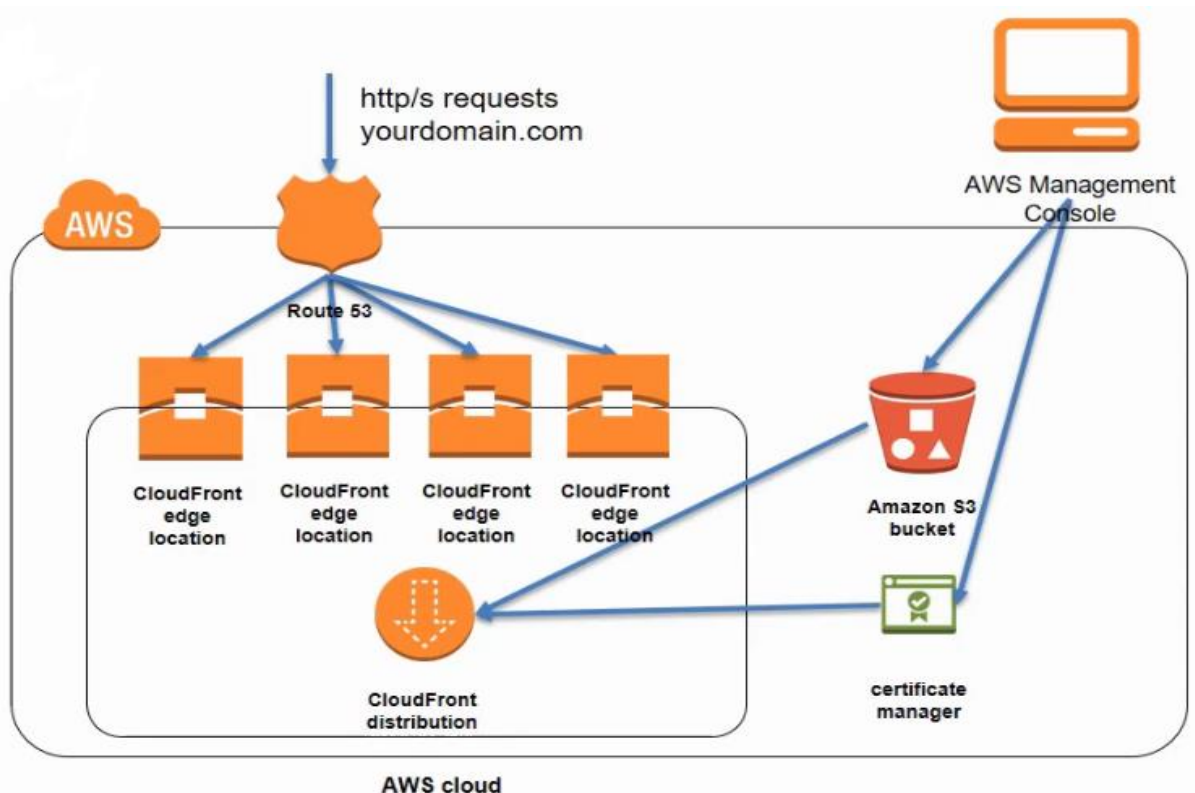


Static bulletproof Website Hosting Using AWS S3 Service

What is Amazon S3?

- Amazon S3 is an object storage service that offers industry-leading scalability, data availability, security, and performance.
- Store and protect any amount of data for a range of use cases, such as data lakes, websites, cloud-native applications, backups, archive, machine learning, and analytics.
- Amazon S3 is designed for 99.999999999% (11 9's) of durability, and stores data for millions of customers all around the world.

Architecture of this project



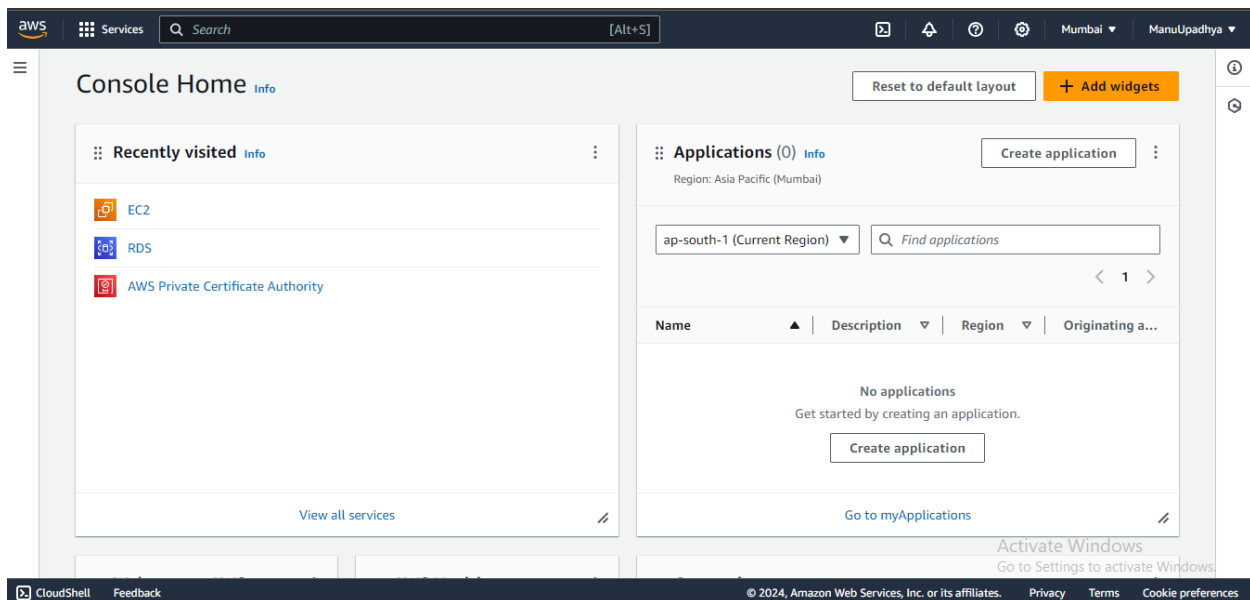
- Network Requests or Network Traffic (HTTP requests to access Our Static Website) to our Domain Name will come to Amazon Web Services(AWS) through **AWS Route 53** Service.
- **AWS Route 53** distributes requests to **AWS CloudFront edge location** close to our end user.

- **CloudFront distribution** will have a copy of our Static Website. **CloudFront distribution** will copy the website from **AWS S3 bucket**, where the original static website is hosted.
- **CloudFront distribution** Updates the copy (of hosted Static website) it contains in regular time interval (example once in a day) from **AWS S3 bucket**. and distributes the updated copy across **AWS CloudFront edge location**.
- We will also be going to have HTTPS enabled, so we're going to have SSL encryption on traffic coming to and from our website. For this we're going to use create SSL certificate using **AWS certificate Manager (ACM)** service.

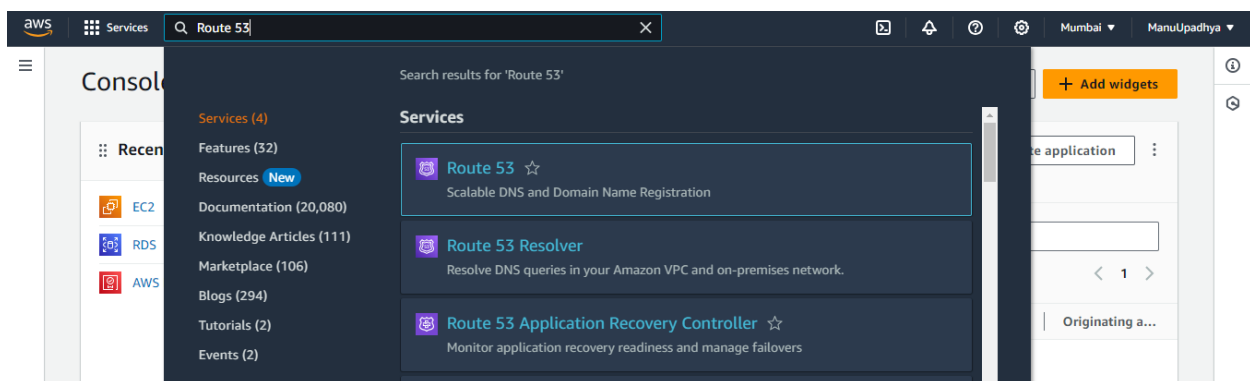
Steps of Hosting the static Bullet Proof Website

1. Purchasing Domain Names With AWS Route 53

Log in to Your AWS Management Console.



Navigate to Services then Search for **Route 53**.



Now Click on **Route 53** to navigate to **Route 53 management console**.

The screenshot shows the Amazon Route 53 console landing page. The header includes the AWS logo, a 'Services' menu, a search bar, and user information 'ManuUpadhya'. The main content area has a dark blue background with the text 'Amazon Route 53' and 'A reliable way to route users to internet applications'. Below this, it states 'Amazon Route 53 is a highly available and scalable cloud Domain Name System (DNS) web service.' To the right, there's a 'Get started with Route 53' section with a 'Get started' button. Below that is a 'Pricing (US)' section with a 'View pricing' link. Further down is a 'More resources' section with a 'Documentation' link. At the bottom, there's a 'How it works' section with a diagram showing a user's device connected to a cloud icon labeled 'Amazon Route 53' and then to a globe icon. A 'Copy link' button is next to the diagram. The footer includes 'CloudShell', 'Feedback', and copyright information for 2024 Amazon Web Services, Inc. or its affiliates, along with links for 'Privacy', 'Terms', and 'Cookie preferences'.

Click on **“Get Started”** Then click on **“Register Domain”**.

The screenshot shows the 'Register domains' page in the Amazon Route 53 console. The breadcrumb navigation at the top reads 'Route 53 > Registered domains > Register domains'. The page title is 'Register domains' with an 'Info' link. Below the title, it says 'Pricing for domain names varies by top-level domain (TLD). For more information, view [price with different TLDs](#)'. The main content area is divided into two columns. The left column has a 'Search for domain' section with a text input field labeled 'Enter a domain name' and a 'Search' button. The right column has a 'Selected domains (0/5)' section with a text input field labeled 'Search for domains and make a selection' and a 'Proceed to checkout' button. The footer includes an 'Activate Windows' watermark and a link to 'Go to Settings to activate Windows'.

Now Search for the domain name which we want to give to our website, consider for example “manojupadhy.com”

The screenshot shows the AWS 'Register domains' page. The breadcrumb trail is 'Route 53 > Registered domains > Register domains'. The page title is 'Register domains' with an 'Info' link. A sub-header states: 'Pricing for domain names varies by top-level domain (TLD). For more information, view [price with different TLDs](#).' The 'Search for domain' section contains a search bar with 'manojupadhy.com' and a 'Search' button. The 'Search result' section displays a table with one row: 'manojupadhy.com' with a green 'Exact match' badge, a price of '13.00 USD', and a 'Select' button. The 'Suggested available domains (10)' section is partially visible. On the right, the 'Selected domains (0/5)' section is empty, with a 'Proceed to checkout' button at the bottom. The footer includes 'CloudShell', 'Feedback', '© 2024, Amazon Web Services, Inc. or its affiliates.', 'Privacy', 'Terms', and 'Cookie preferences'.

If we get the exact match, it is good or we must make some tweaks so that we will have domain name available, proceed to checkout and buy the domain name.

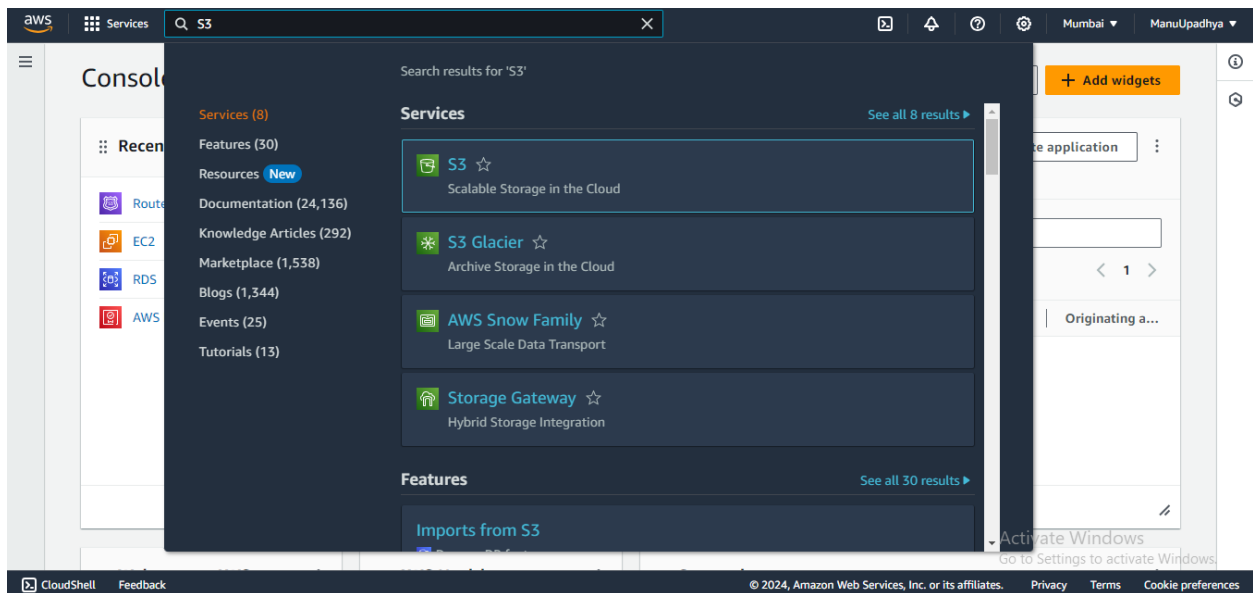
This screenshot shows the same AWS 'Register domains' page, but now the domain 'manojupadhy.com' is selected. The 'Search result' table shows 'manojupadhy.com' with a green 'Exact match' badge, a price of '13.00 USD', and a 'Selected' status. The 'Selected domains (1/5)' section on the right now lists 'manojupadhy.com' with a price of '13.00 USD' and a 'Remove' button. Below this, the 'Subtotal' is '13.00 USD' with a note: 'The domain registration fee displayed is for 1 year. You can change duration on the next page.' An orange 'Proceed to checkout' button is at the bottom of the right panel. The rest of the page layout remains the same.

After successful purchase of domain name, the purchased domain name is visible under “Registered Domain names” tab.

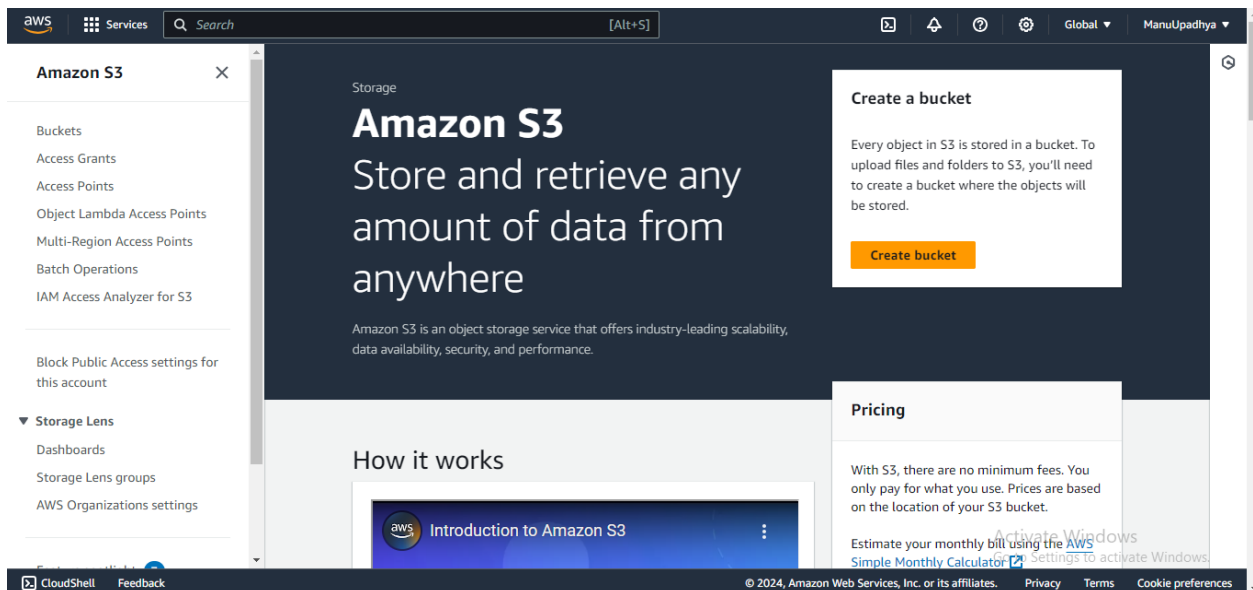
[I am not going to buy domain name here].

2. Creating an S3 Bucket and Hosting our Website

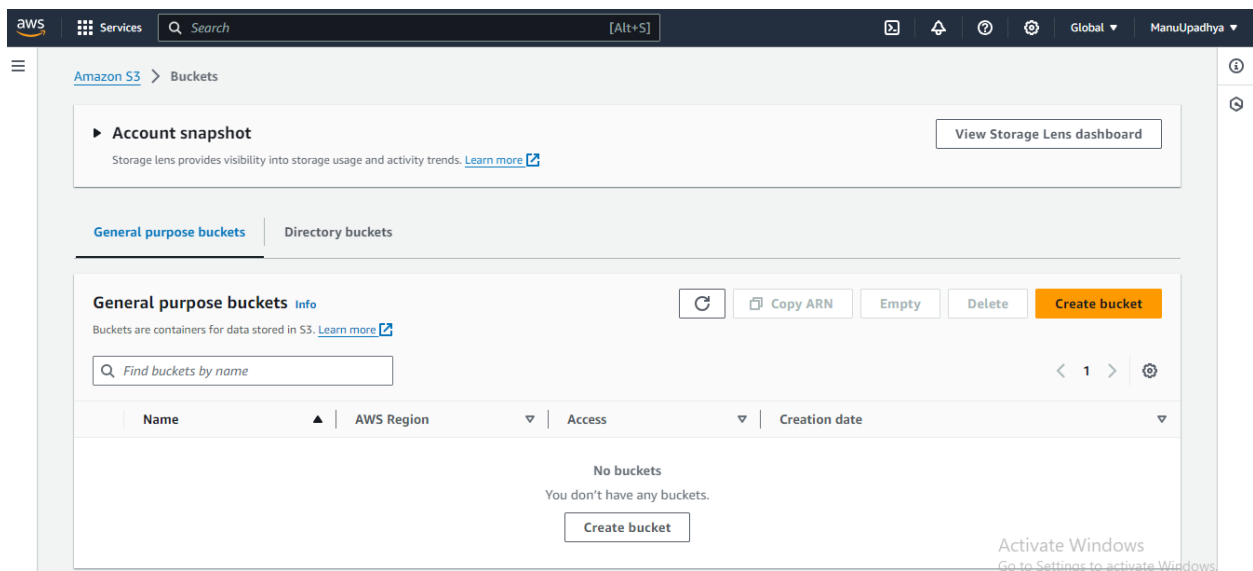
Navigate to “Services” then to “AWS S3 Service”.



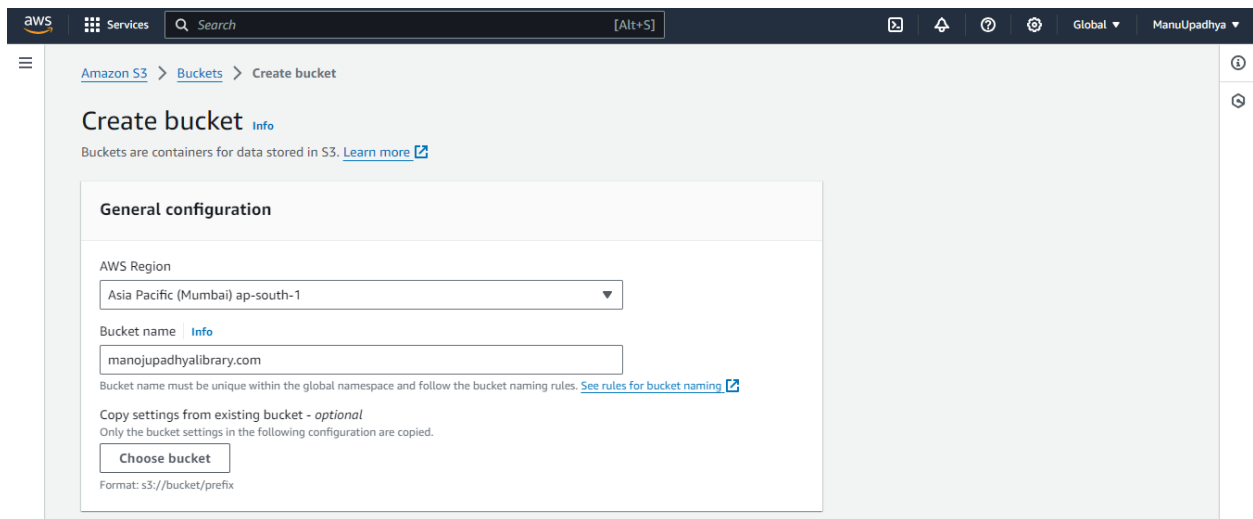
Click on “S3” to navigate to S3 management console.



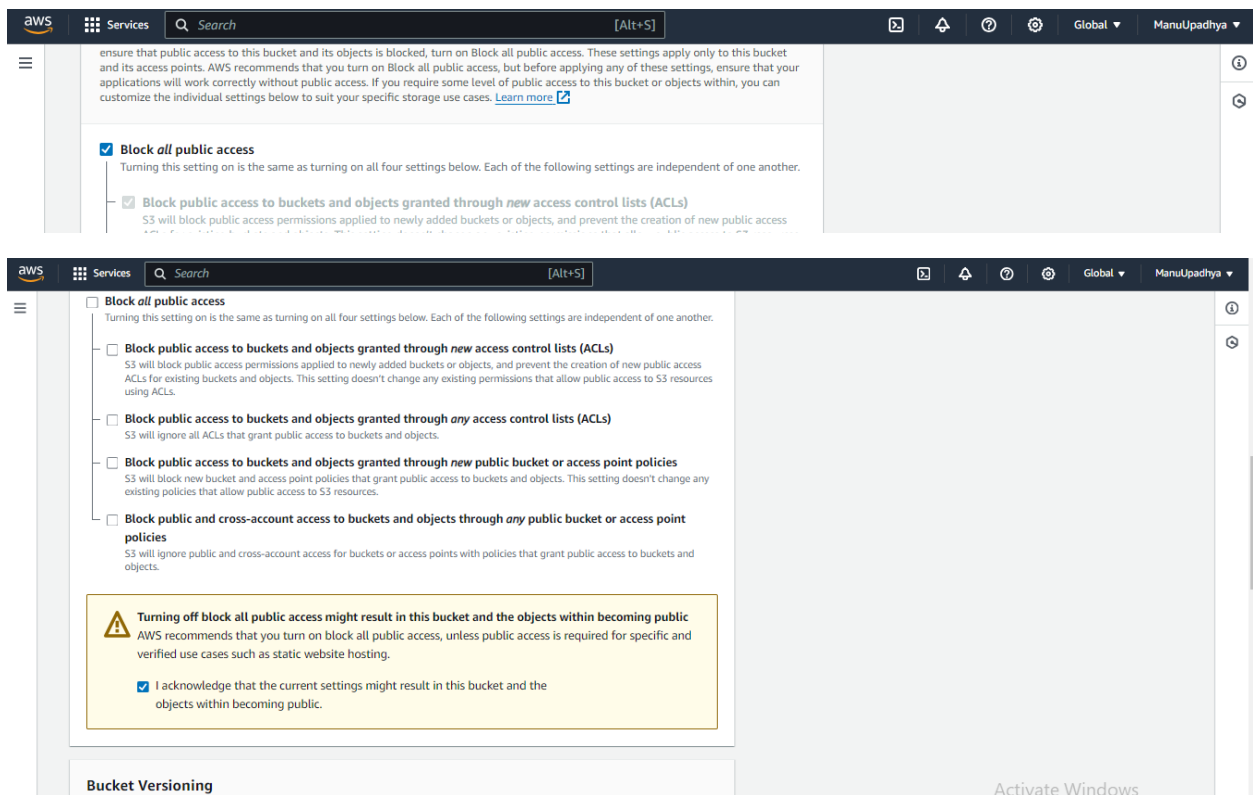
Click on “buckets” to view all previously created buckets.



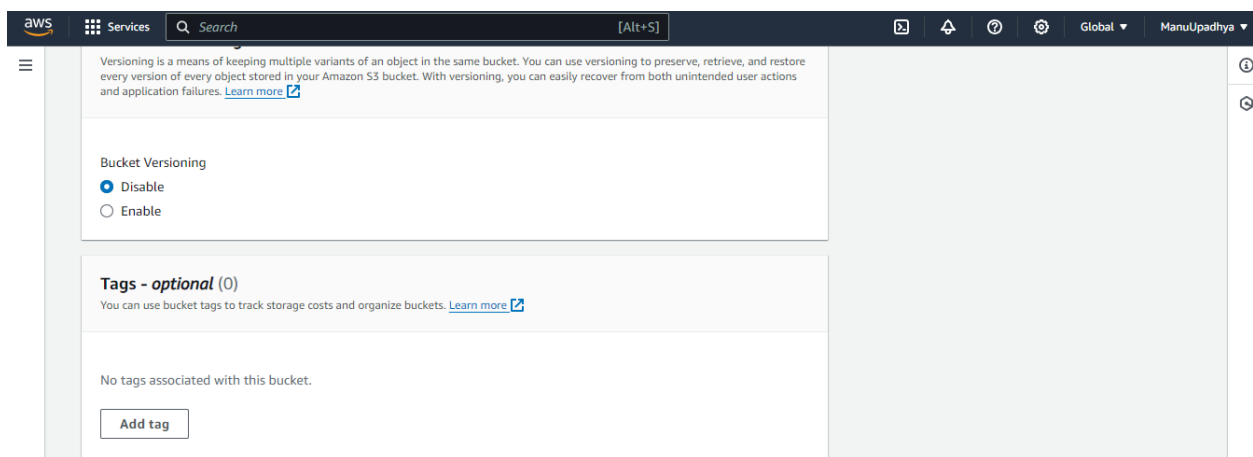
Now we need to create a new bucket to host our Website. Now Click on “Create Bucket”, Give our DNS website name (that we purchased) as bucket name. Select a Region (doesn’t matter as S3 is a global Service)



Disable the “Block all Public access” Option so that public should be able to access our websites.

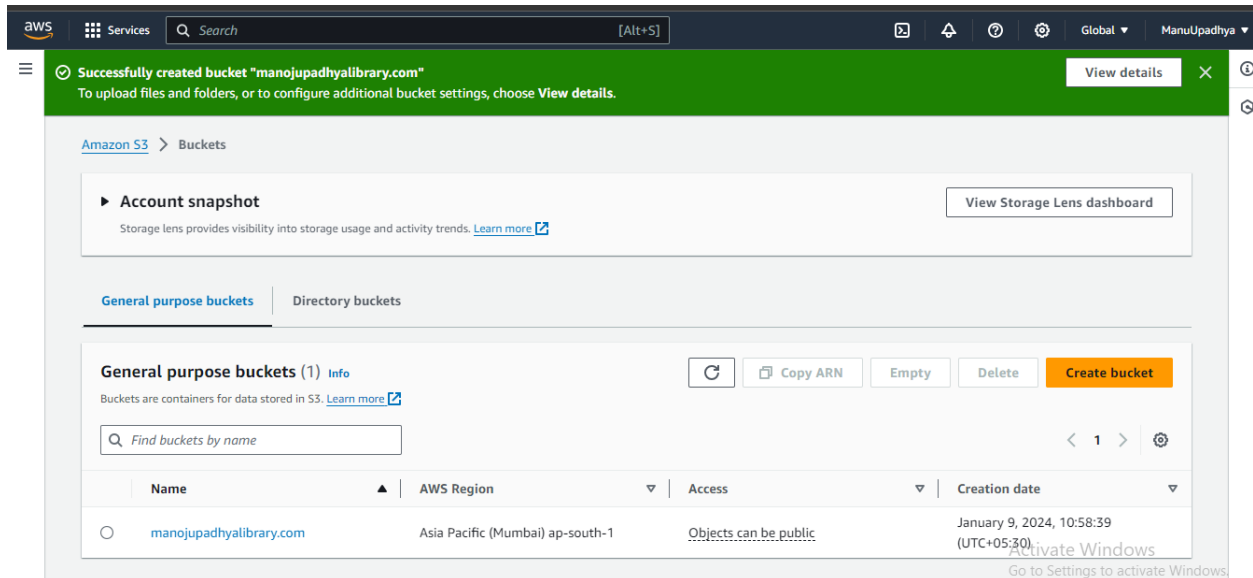


Make **Bucket Versioning** as Disabled, it will be enabled by default and keep **tags** as it is.

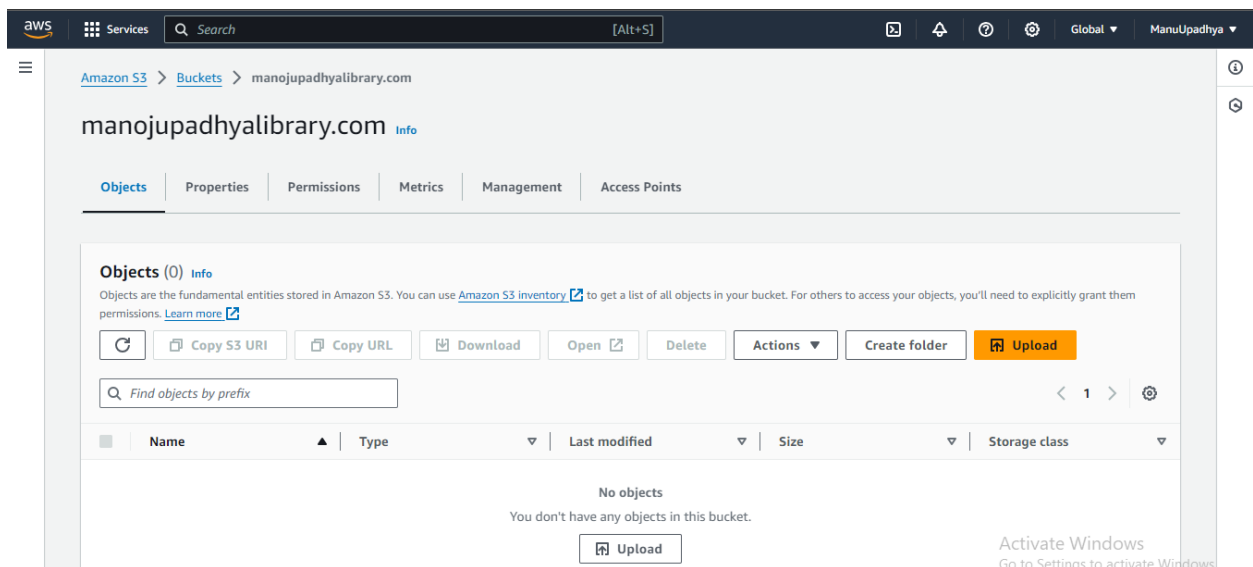


Now Click on “Create Bucket”.

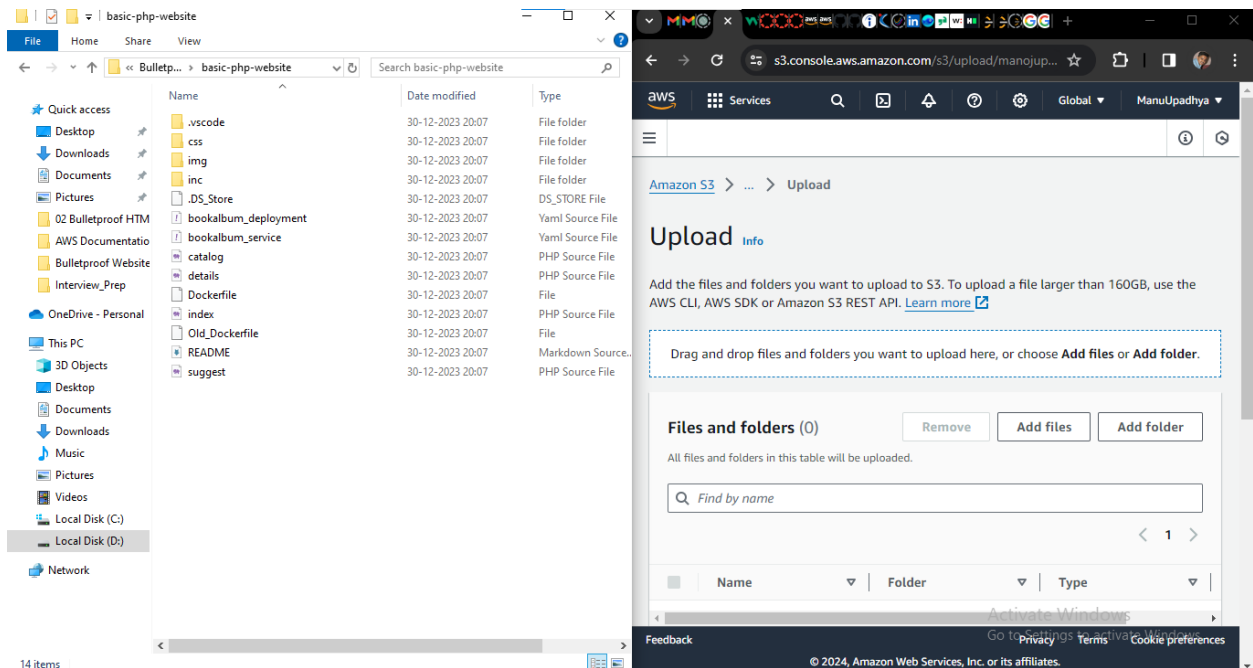
A new S3 bucket creation completed successfully. It will automatically navigate to “buckets” page where we can see our newly created bucket.



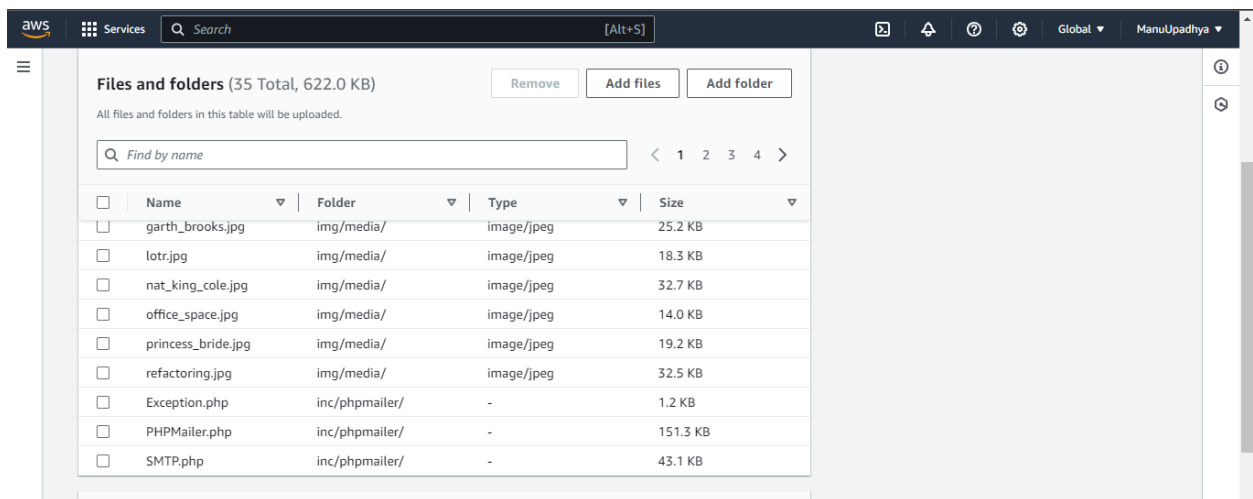
Now open the created bucket to upload our website.



Click on Upload. Drag and drop the entire website folder to S3 bucket.



Uploaded website data in S3 bucket



Permissions:- as we set bucket to have public access here, we don't need to give permissions.

Properties:- Keep it default that is **“Standard”**

Storage class Info					
Amazon S3 offers a range of storage classes designed for different use cases. Learn more or see Amazon S3 pricing					
	Storage class	Designed for	Bucket type	Availability Zones	Minimum object size
<input type="radio"/>	S3 Express One Zone	Single-digit millisecond response times for the most frequently accessed data.	Directory	1	-
<input checked="" type="radio"/>	Standard	Frequently accessed data (more than once a month) with milliseconds access	General purpose	≥ 3	-
<input type="radio"/>	Intelligent-Tiering	Data with changing or unknown access patterns	General purpose	≥ 3	-
		Infrequently accessed data (once a month)	General		

Keep **Server-Side Encryption** same as default, keep **additional checksum** as off.

aws

Services

Search

[Alt+S]

Global

ManuUpadhy

Server-side encryption [Info](#)

Server-side encryption protects data at rest.

Server-side encryption

☒ Do not specify an encryption key
The bucket settings for default encryption are used to encrypt objects when storing them in Amazon S3.
 ☐ Specify an encryption key
The specified encryption key is used to encrypt objects before storing them in Amazon S3.

⚠

If your bucket policy requires objects to be encrypted with a specific encryption key, you must specify the same encryption key when you upload objects. Otherwise, uploads will fail.

Additional checksums [Info](#)

Checksum functions are used for additional data integrity verification of new objects. [Learn more](#)

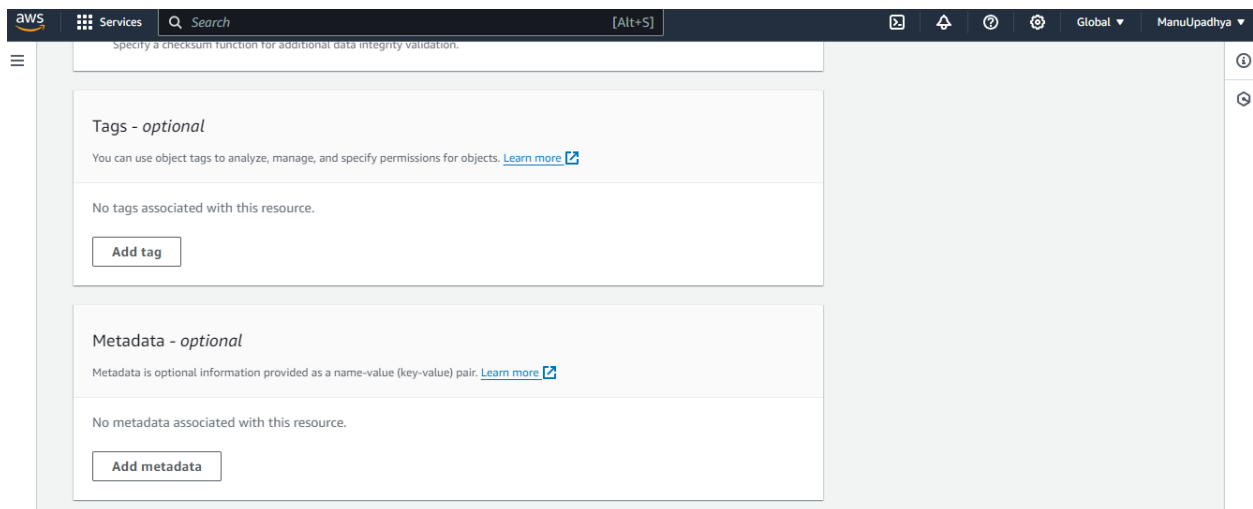
Additional checksums

☒ Off
Amazon S3 will use a combination of MD5 checksums and Etags to verify data integrity.

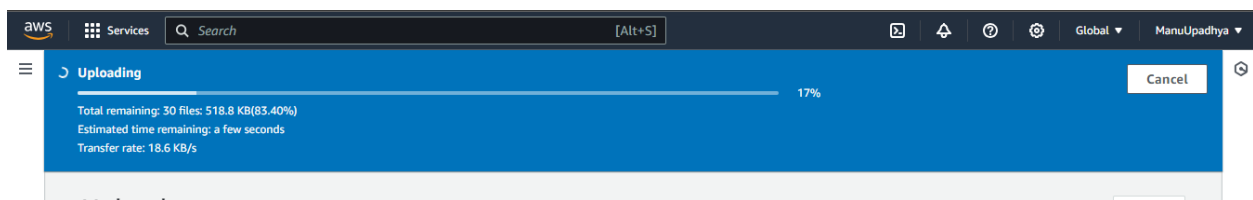
Activate Windows

Go to Settings to activate Windows

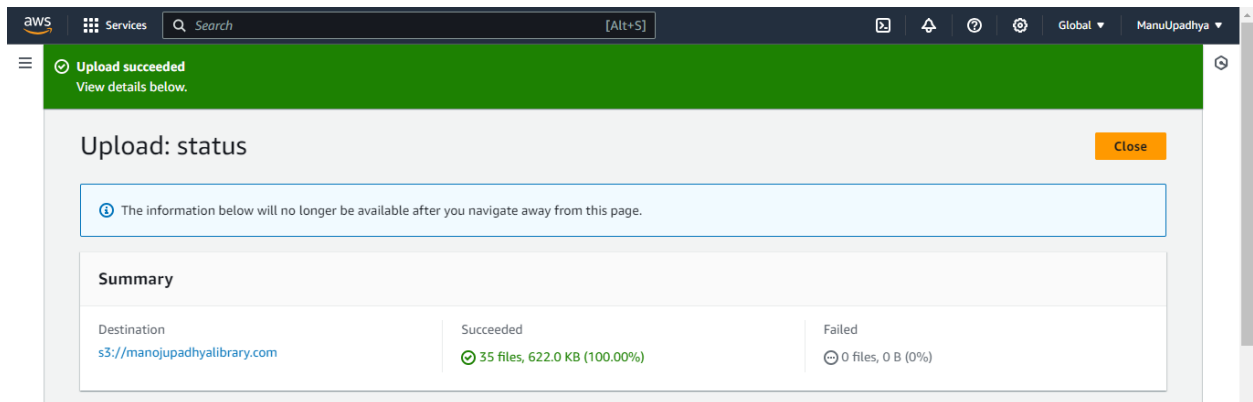
No “Tags” and “Meta data” are required.



Now click on **“Upload”** to Upload the website files as objects to S3 bucket.

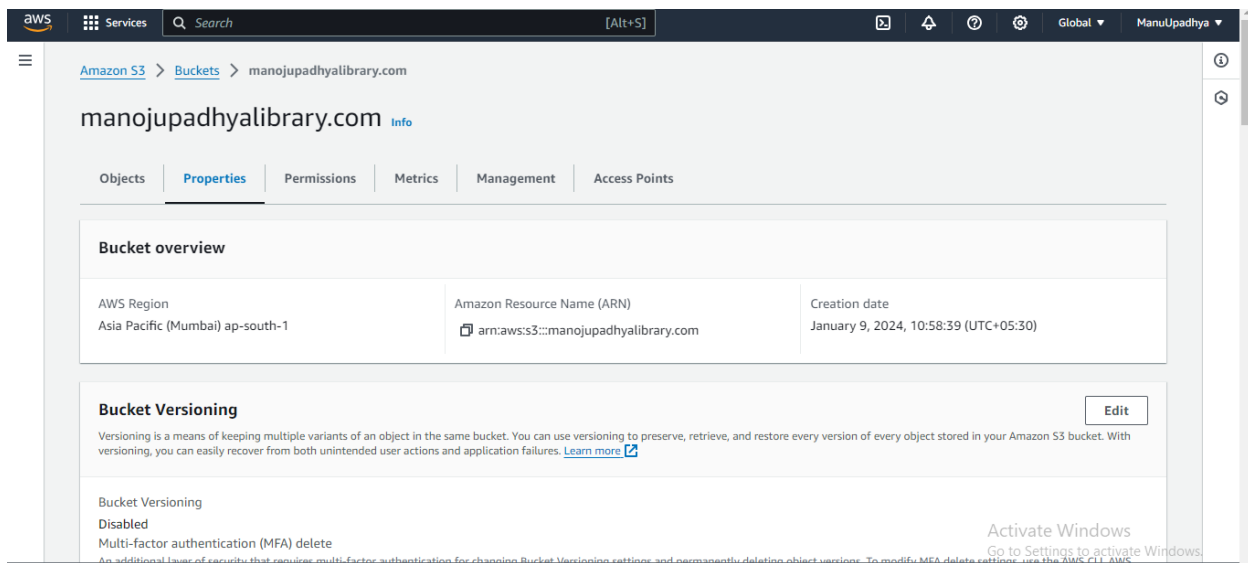


Once Uploading Completed Successfully. Make sure that bucket objects are public(Select all objects-> Actions -> make public using ACL).

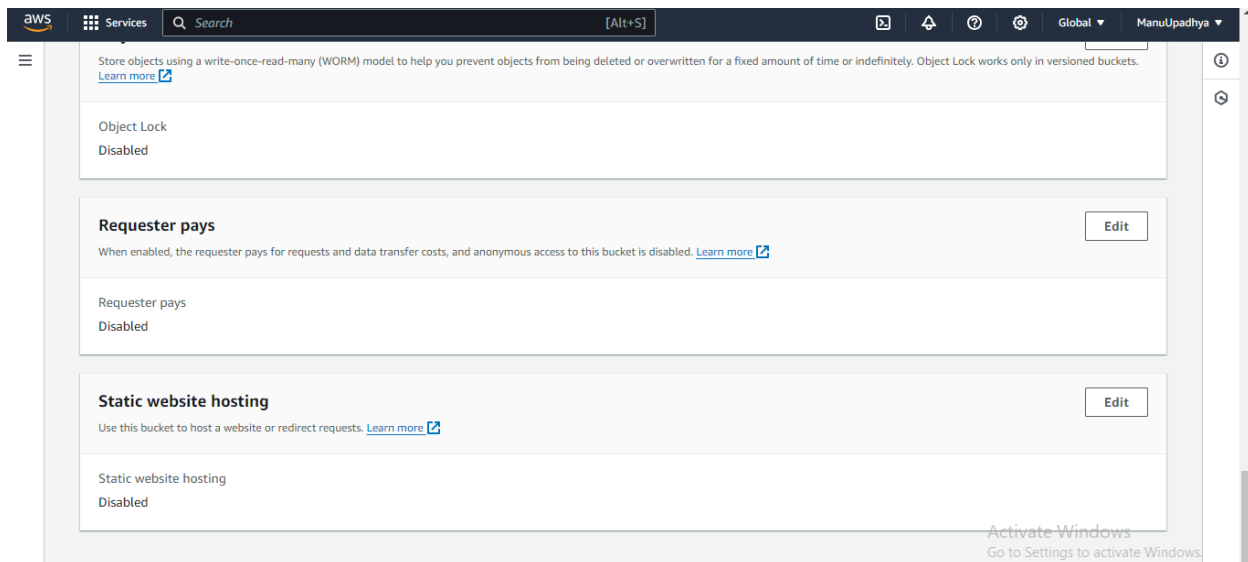


Next step is Hosting our S3 buckets as a website (enabling bucket objects to host as a website)

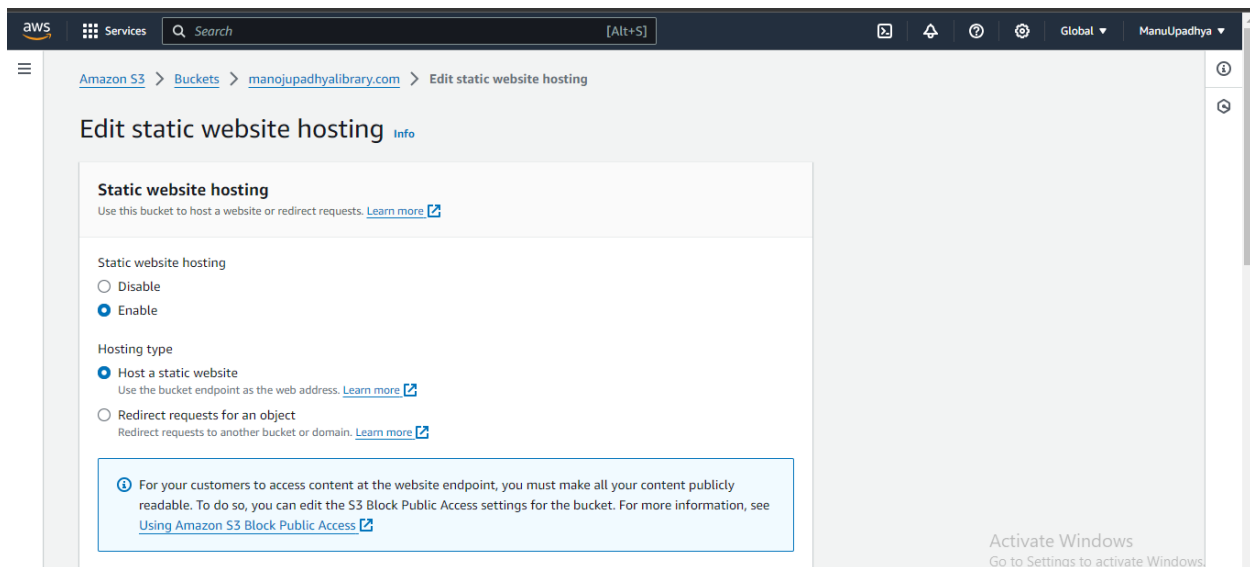
From inside the bucket navigate to properties.



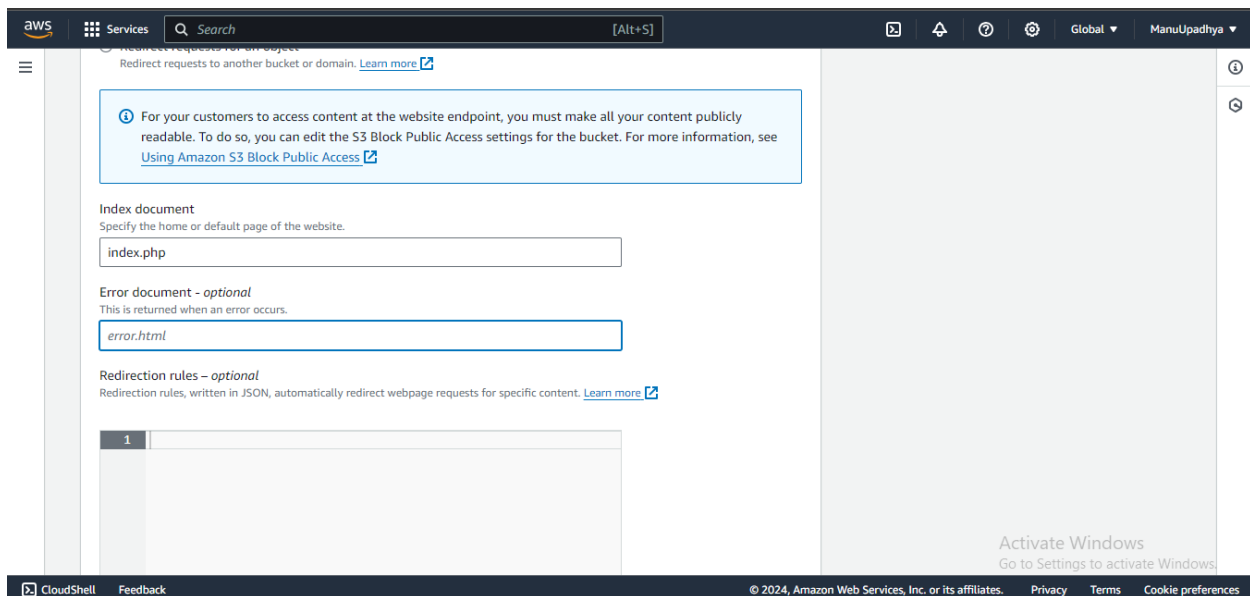
Navigate to “static Website hosting” .



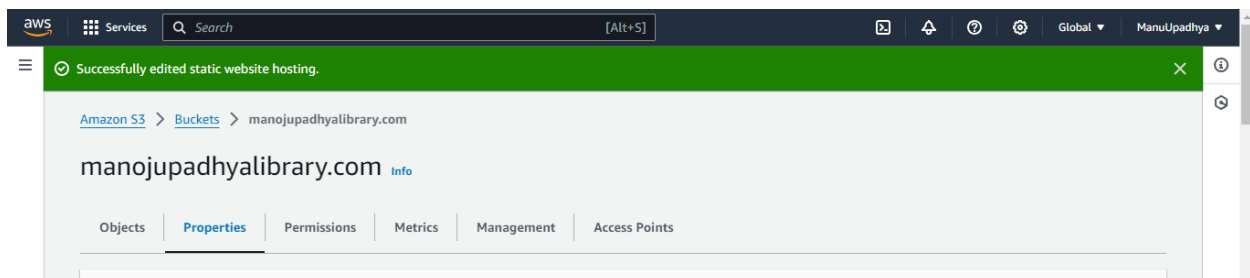
then click on “Edit” to enable the static website hosting option, by default it will be disabled.



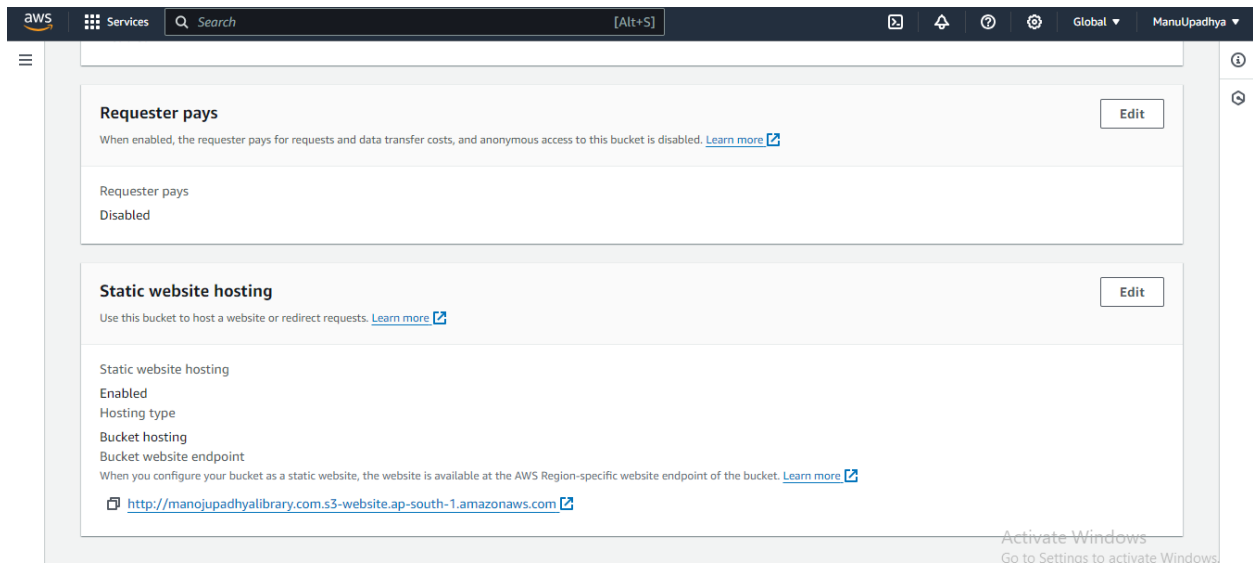
Insert index and error document name of our website.



Then click on “Save changes”. Successfully enabled the static website hosting.



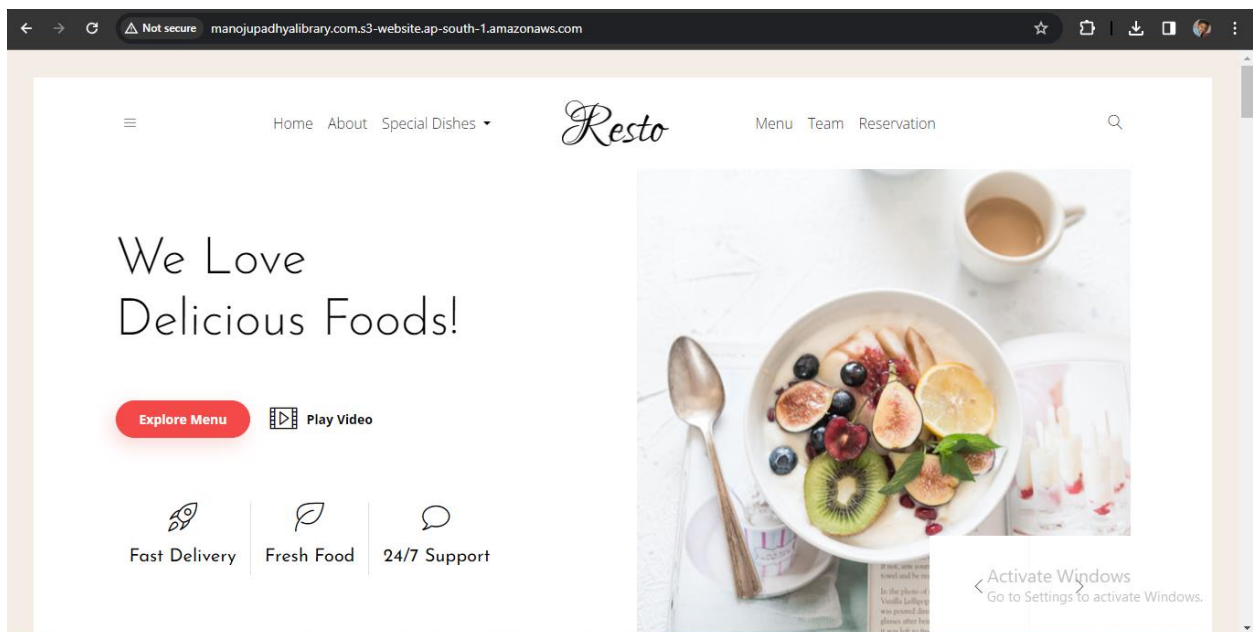
Now Navigate again to “Static website Hosting” option in Properties of Bucket to get the endpoint of our website.

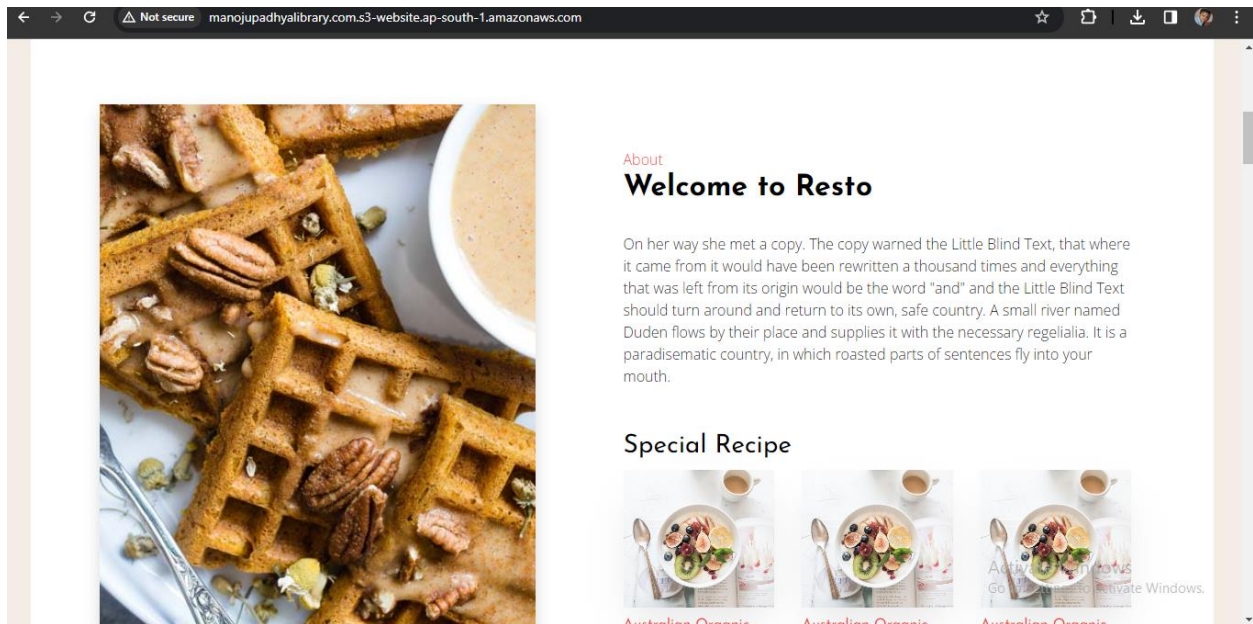


When we open the link/ endpoint we will see our hosted/static website

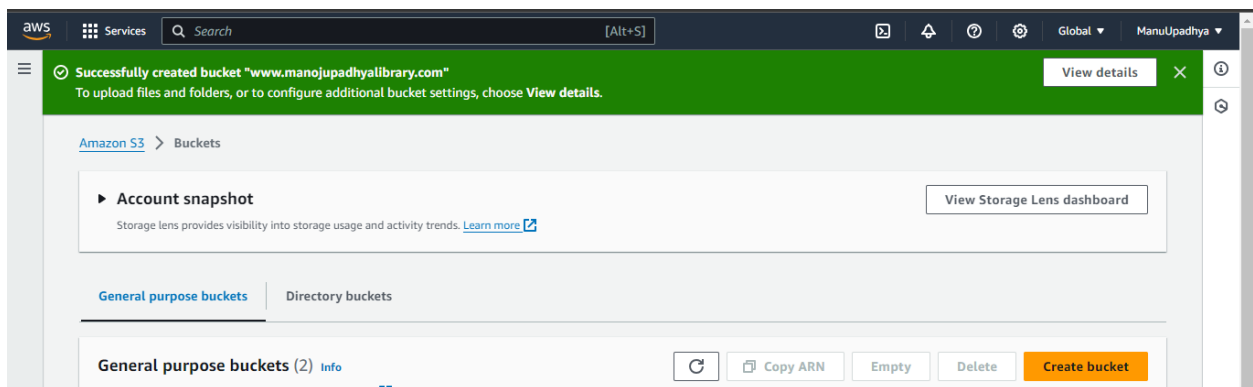
Link to our static website:-

<http://manojupadhyalibrary.com.s3-website.ap-south-1.amazonaws.com>

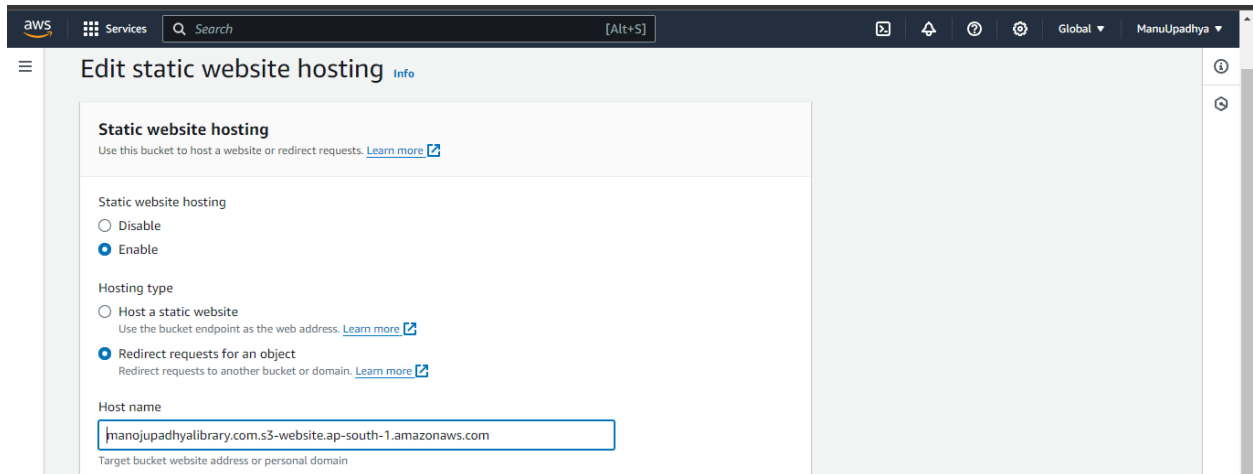




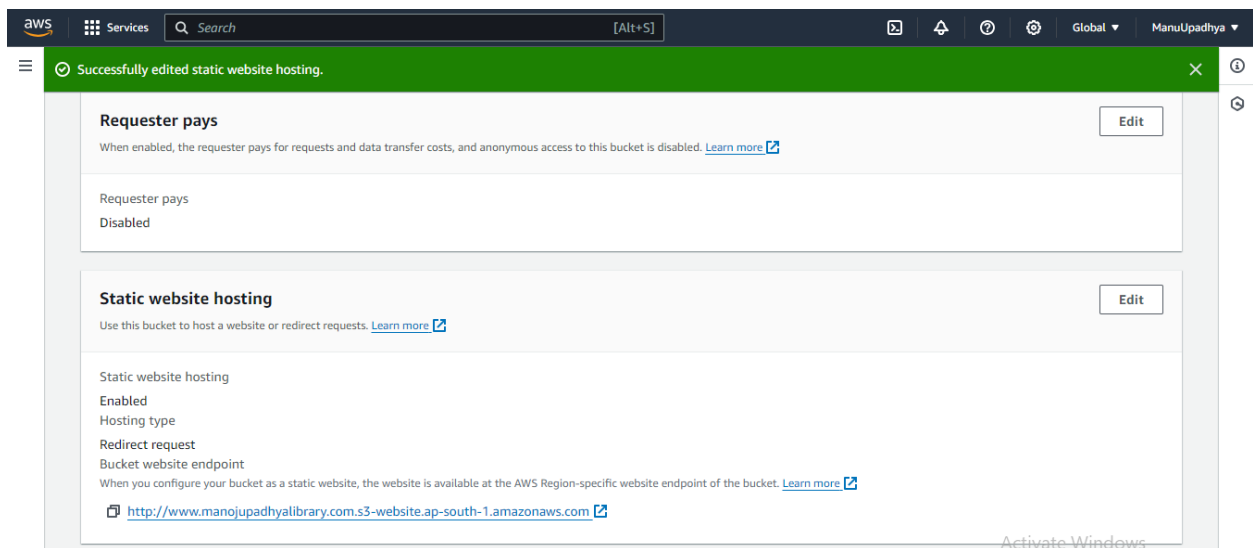
One problem here we can face is if we start our URL to site as `www. manojupadhyalibrary.com.s3-website.ap-south-1.amazonaws.com` It won't land on our website; it will throw 404 error because `www` is not identified by our bucket. So, to resolve this we create one more bucket **for `www. manojupadhyalibrary.com`** and will route the traffic coming to this bucket to the original bucket **`manojupadhyalibrary.com`**.



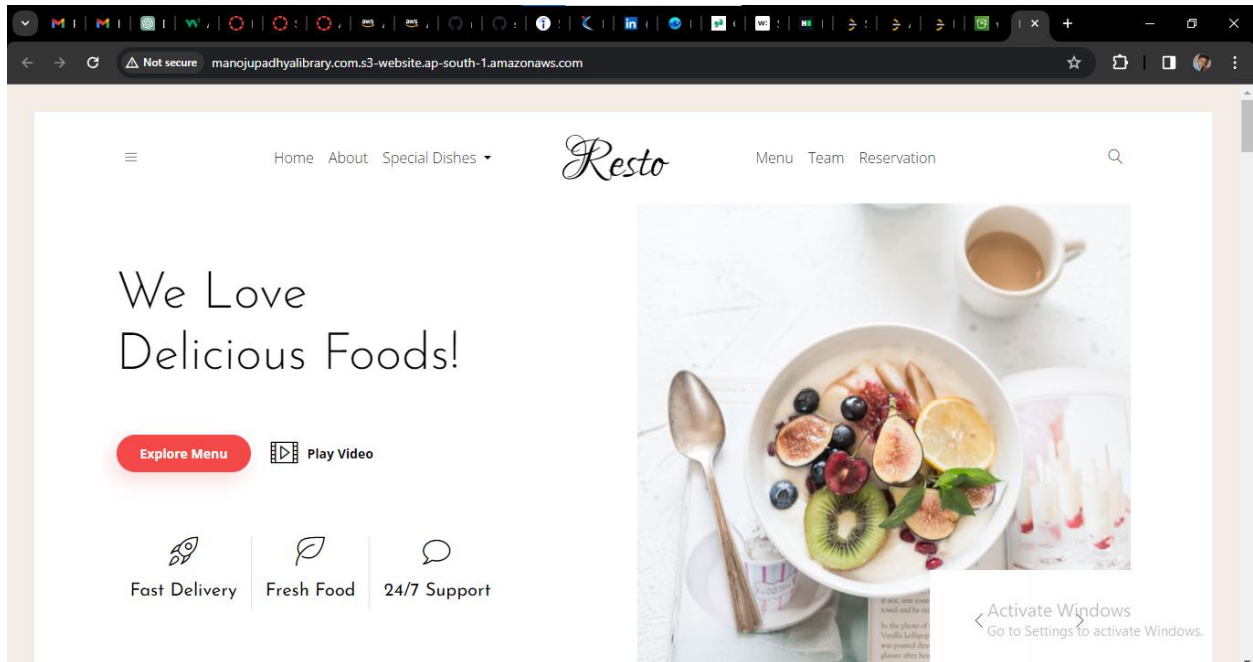
Now go inside the newly created bucket, navigate to properties then go to static website hosting. Now choose "Redirect request for an object" and enter the target bucket name and click save changes.



Now we will go again to the second bucket which is created for **www. manojupadhyalibrary.com**, Navigate to properties then to Static website Hosting.

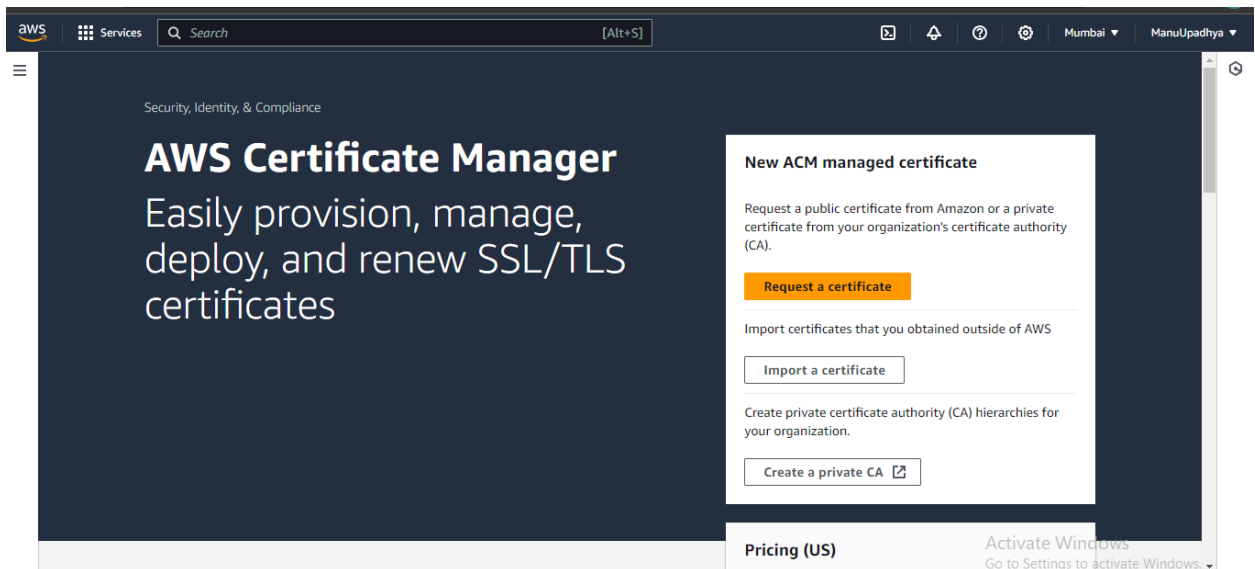


If we open the end point it will land to our static website.

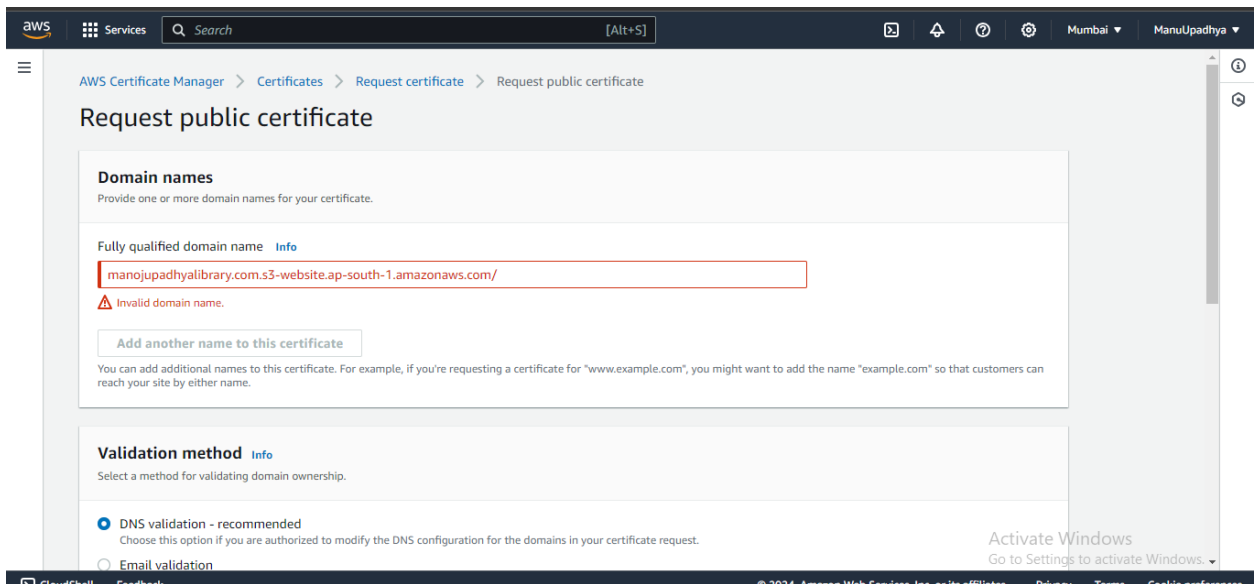


3. Creating SSL Certificate Using Certificate Manager

Navigate to Services->Certificate Manager Management Console



Click on Request a Certificate-> then enter our domain name



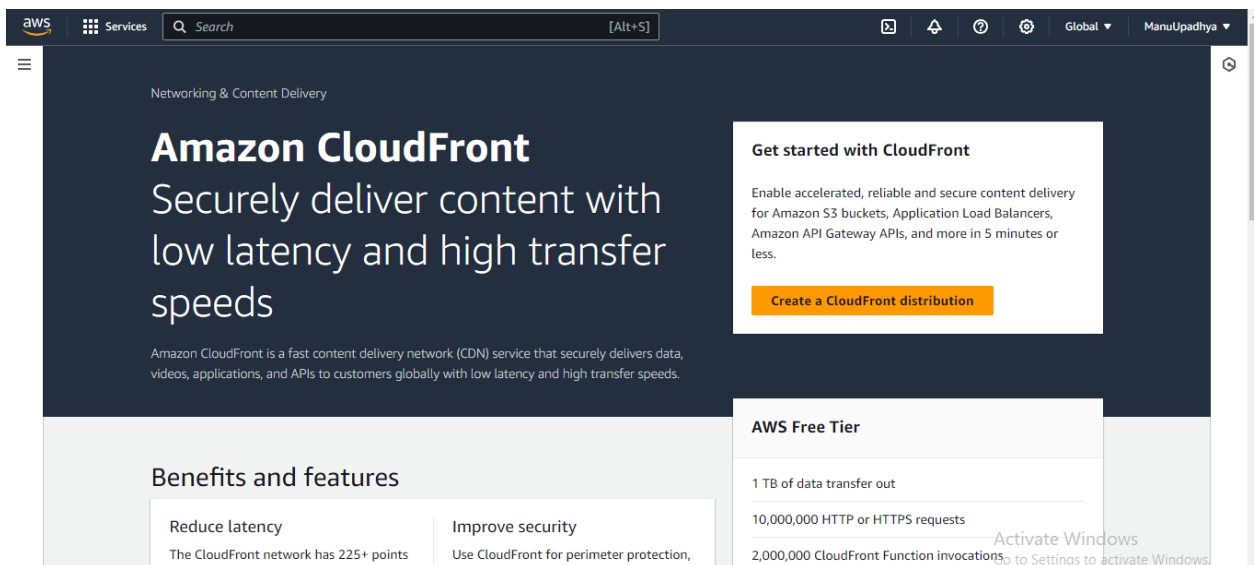
As I have not purchased the domain name it will throwing me error.

We can add our subdomains as well, if we have multiple subdomains, we can use wild cards, and enter our sub domain name as ***.domain name**.

Certificates need to be validated; link will be sent to our mail address. After approving the certificate, we will see the newly created certificate in certificate management console.

4. Creating CloudFront Distribution

Goto Services->CloudFront management console



Now create a distribution

The screenshot displays the AWS CloudFront 'Create distribution' console. The top navigation bar includes the AWS logo, 'Services', a search bar, and user information. The breadcrumb trail shows 'CloudFront > Distributions > Create'.

Create distribution

Origin

Origin domain
Choose an AWS origin, or enter your origin's domain name.

manojupadhyalibrary.com.s3.ap-south-1.amazonaws.com

Warning: This S3 bucket has static web hosting enabled. If you plan to use this distribution as a website, we recommend using the S3 website endpoint rather than the bucket endpoint.

[Use website endpoint](#)

Origin path - optional [Info](#)
Enter a URL path to append to the origin domain name for origin requests.

Enter the origin path

Name

Viewer protocol policy

- ☒ HTTP and HTTPS
- ☐ Redirect HTTP to HTTPS
- ☐ HTTPS only

Allowed HTTP methods

- ☒ GET, HEAD
- ☐ GET, HEAD, OPTIONS
- ☐ GET, HEAD, OPTIONS, PUT, POST, PATCH, DELETE

Restrict viewer access
If you restrict viewer access, viewers must use CloudFront signed URLs or signed cookies to access your content.

- ☒ No
- ☐ Yes

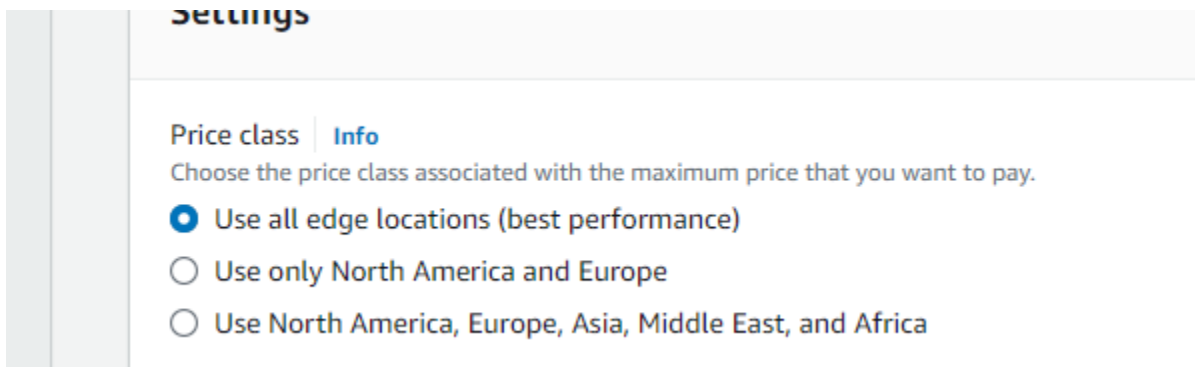
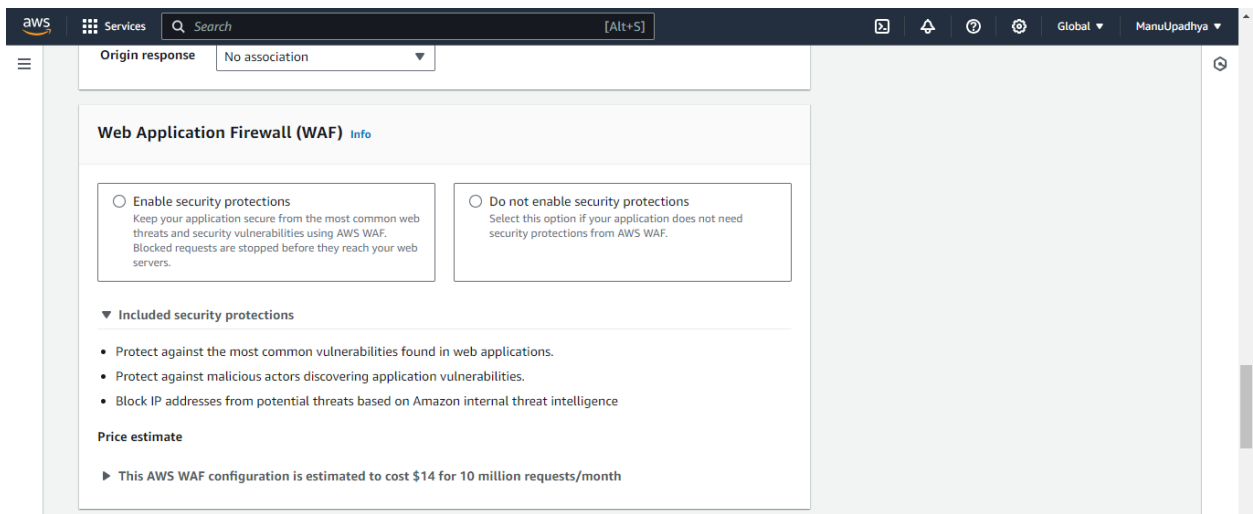
Cache key and origin requests
We recommend using a cache policy and origin request policy to control the cache key and origin requests.

- ☒ Cache policy and origin request policy (recommended)
- ☐ Legacy cache settings

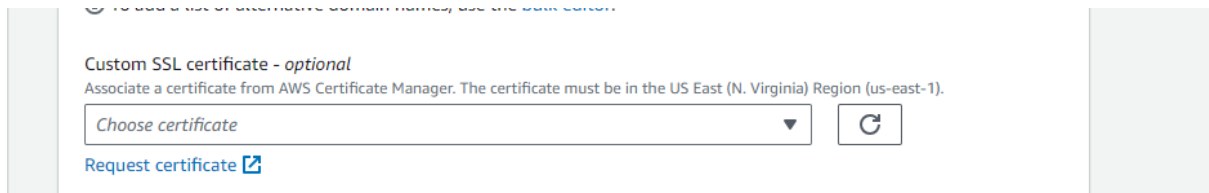
Cache policy
Choose an existing cache policy or create a new one.

CachingOptimized Recommended for S3 [Refresh](#)

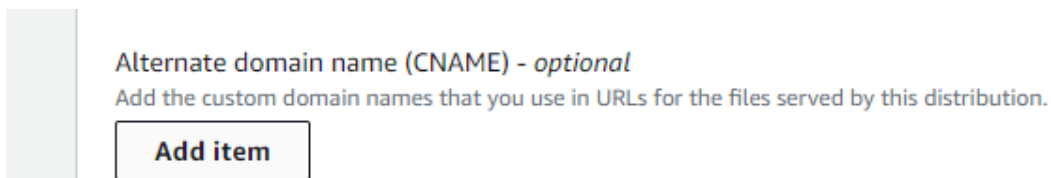
If we want a Web application firewall, we can add it from here for the purpose of security.



Here we are informing to use all edge location for best performance.

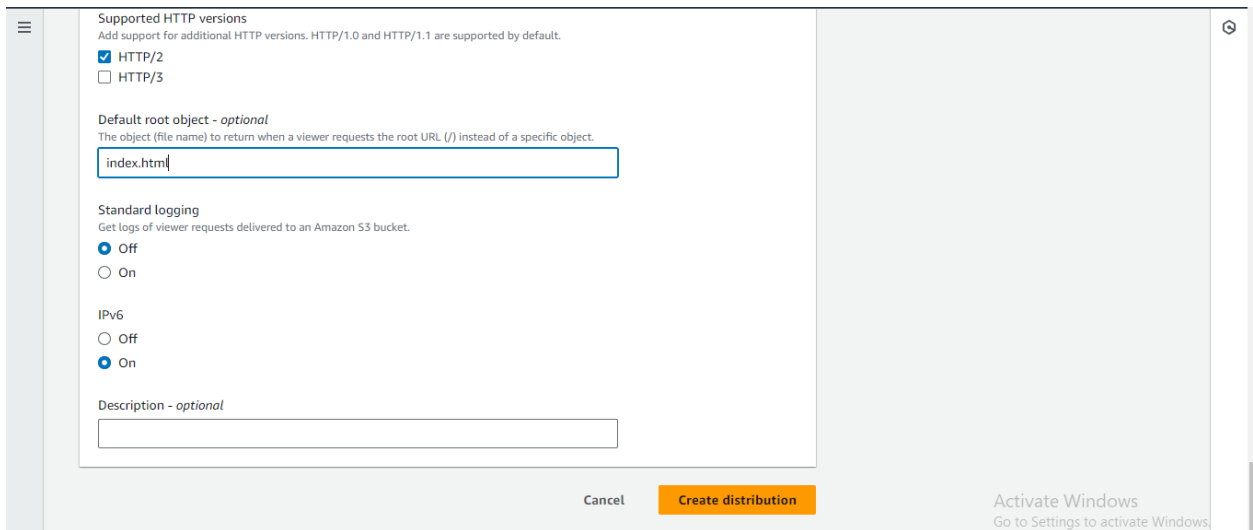


Here we will select our Created Custom SSL Certificate.



Here we will add domain name of our website.

Add root object that is index.html and Click on create distribution.



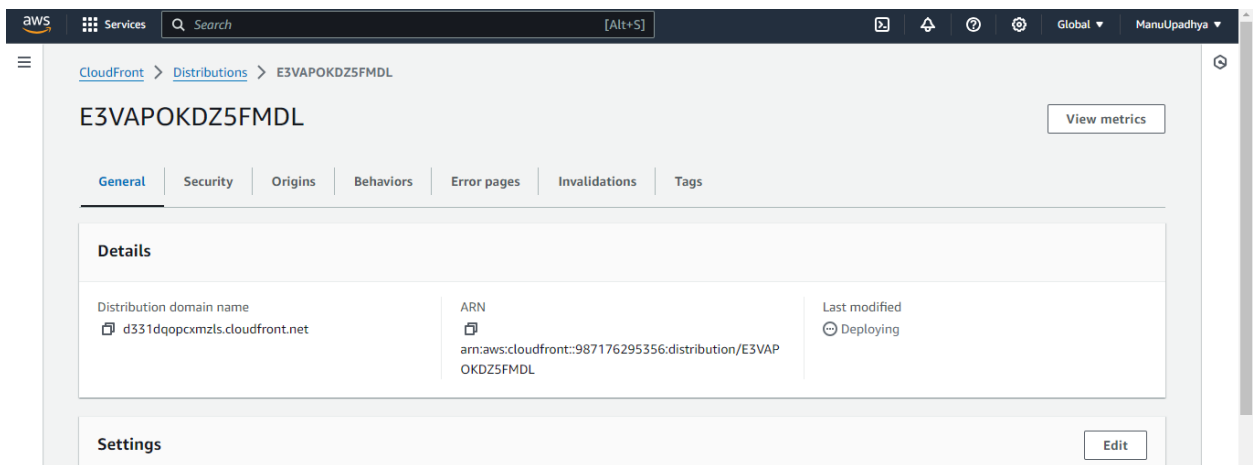
The screenshot shows the 'Create distribution' form in the AWS CloudFront console. The form is divided into several sections:

- Supported HTTP versions:** A section with a sub-header 'Add support for additional HTTP versions. HTTP/1.0 and HTTP/1.1 are supported by default.' It contains two radio buttons: 'HTTP/2' (selected) and 'HTTP/3' (unselected).
- Default root object - optional:** A section with a sub-header 'The object (file name) to return when a viewer requests the root URL (/) instead of a specific object.' It contains a text input field with the value 'index.html'.
- Standard logging:** A section with a sub-header 'Get logs of viewer requests delivered to an Amazon S3 bucket.' It contains two radio buttons: 'Off' (selected) and 'On' (unselected).
- IPv6:** A section with two radio buttons: 'Off' (unselected) and 'On' (selected).
- Description - optional:** A section with a text input field.

At the bottom of the form, there are two buttons: 'Cancel' and 'Create distribution'. In the bottom right corner, there is a watermark that says 'Activate Windows Go to Settings to activate Windows'.

Created Distribution can be found in CloudFront distribution management console.

Initially the status will be deploying after some time status will be changed to deployed.



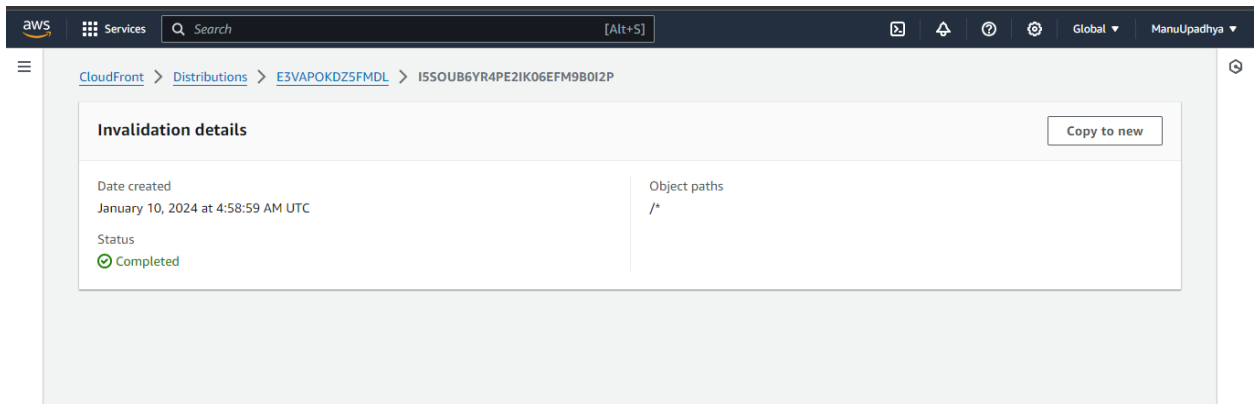
The screenshot shows the 'E3VAPOKDZ5FMDL' distribution page in the AWS CloudFront console. The page has a navigation bar at the top with 'CloudFront' and 'Distributions' links, and a search bar. The main content area is divided into several tabs: 'General', 'Security', 'Origins', 'Behaviors', 'Error pages', 'Invalidations', and 'Tags'. The 'General' tab is selected.

The 'Details' section shows the following information:

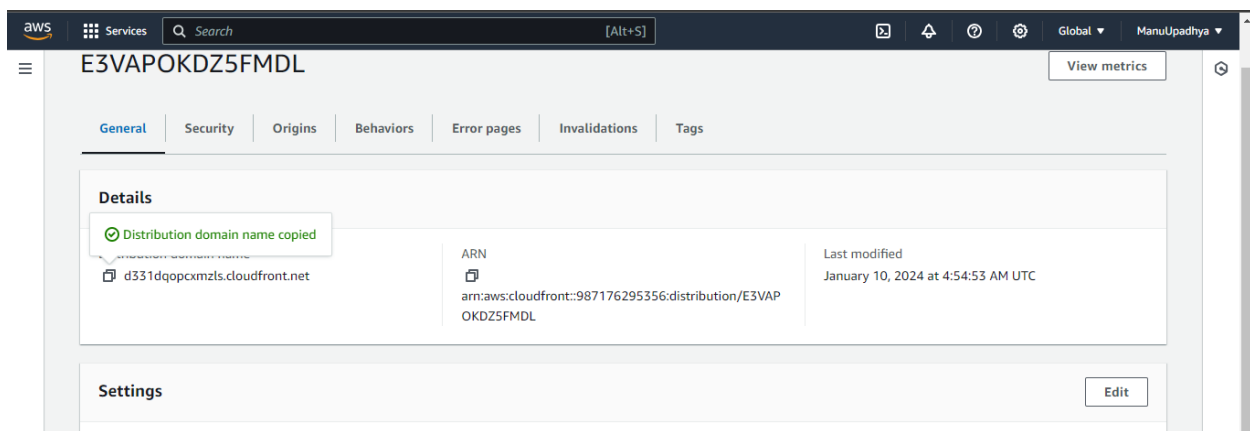
Property	Value	Last modified
Distribution domain name	d331dopcxmzls.cloudfront.net	Deploying
ARN	arn:aws:cloudfront::987176295356:distribution/E3VAPOKDZ5FMDL	

At the bottom of the page, there is a 'Settings' section with an 'Edit' button.

Navigate to invalidation tab in CloudFront Distribution. Create invalidation And add files which we need to invalidate. If we want to invalidate everything enter the wild card *



5. Routing traffic with AWS Route 53



This is the CloudFront Distribution Domain name, if we open this in web browser it will directly land on our website. We will use this as alias target name in Route 53 for previously created domain name.

After successful setup if user enters just domain in web browser, he will be directed to alia target name that is CloudFront Distribution Domain name, and website will be loaded in his web browser.

Not performed this step as I have not created Domain name. for my website. Still I can access the hosted website using the S3 bucket endpoint.

