



Module	Assessment Type
Distributed and Cloud Systems Programming	Individual Report

Workshop 2

Student Id : 2049867 (NP03A190017)
Student Name : Roshan Parajuli
Section : L5CG3
Module Leader : Rupak Koirala
Lecturer /Tutor : Saroj Sharma
Submitted on : 2020-03-30

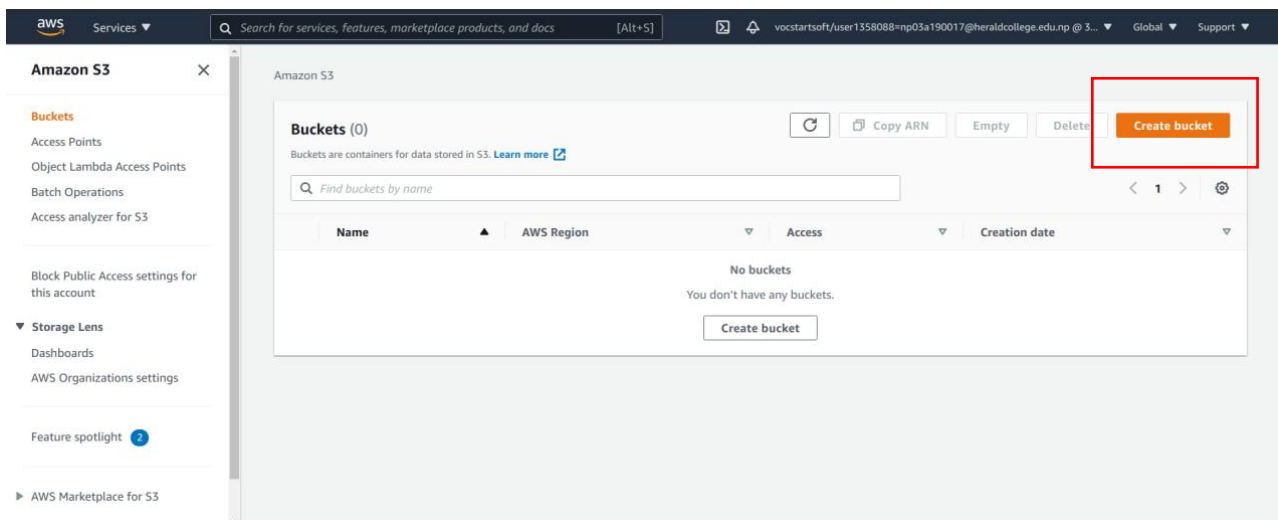
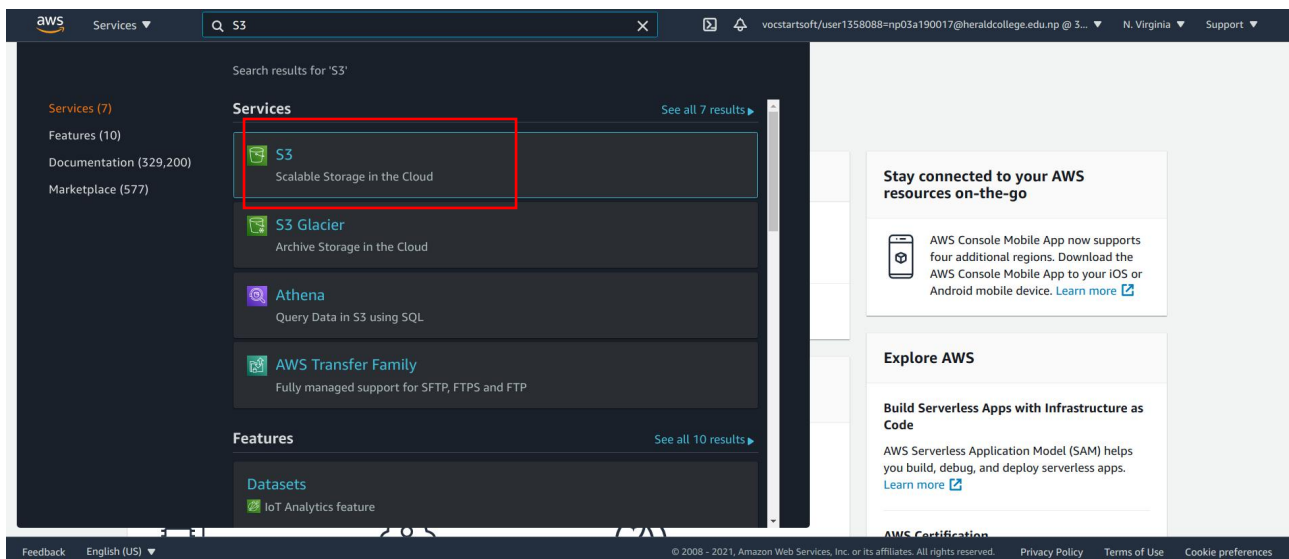
Table of Contents

Task 1: Static Web Hosting.....	3
Uploading content.....	4
Adding a Bucket Policy to Allow Public Reads.....	5
Enabling website hosting and testing the implementation.....	6
Screenshot of the S3 bucket.....	7
Task 2: User Management.....	8
Creating an Amazon Cognito User Pool.....	8
Testing the implementation.....	10

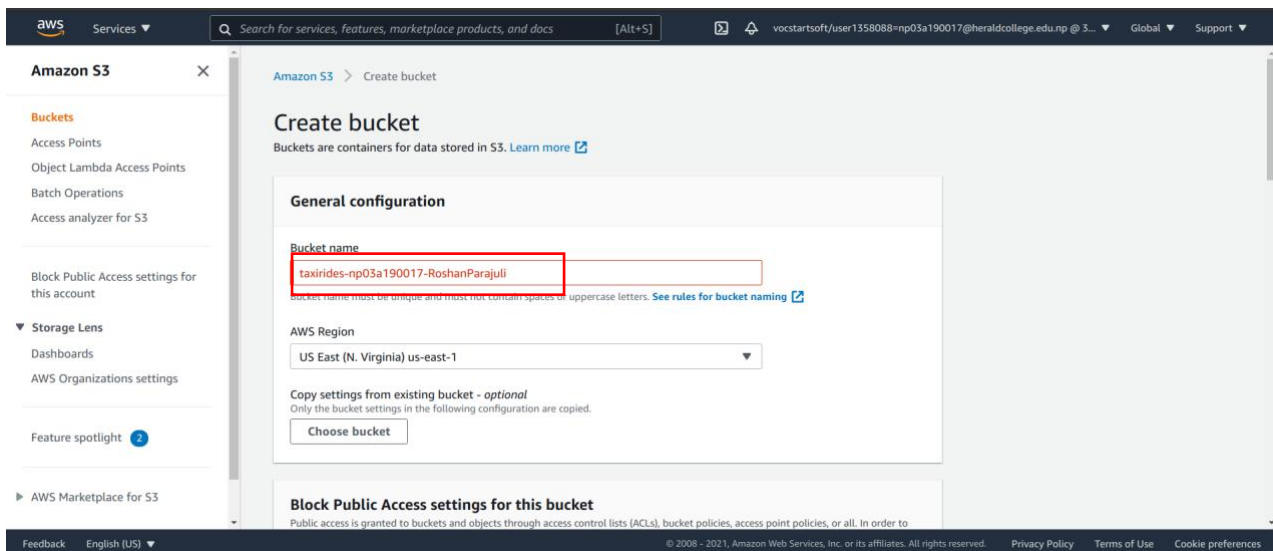
Task 1: Static Web Hosting

Creating an **S3 bucket** for hosting the static resources and pages for the web application. The dynamic functionalities are to be added later using Javascript to call remote RESTful APIs built with **AWS Lambda** and **Amazon API Gateway**.

First of all, S3 service is selected from the services section in AWS console.

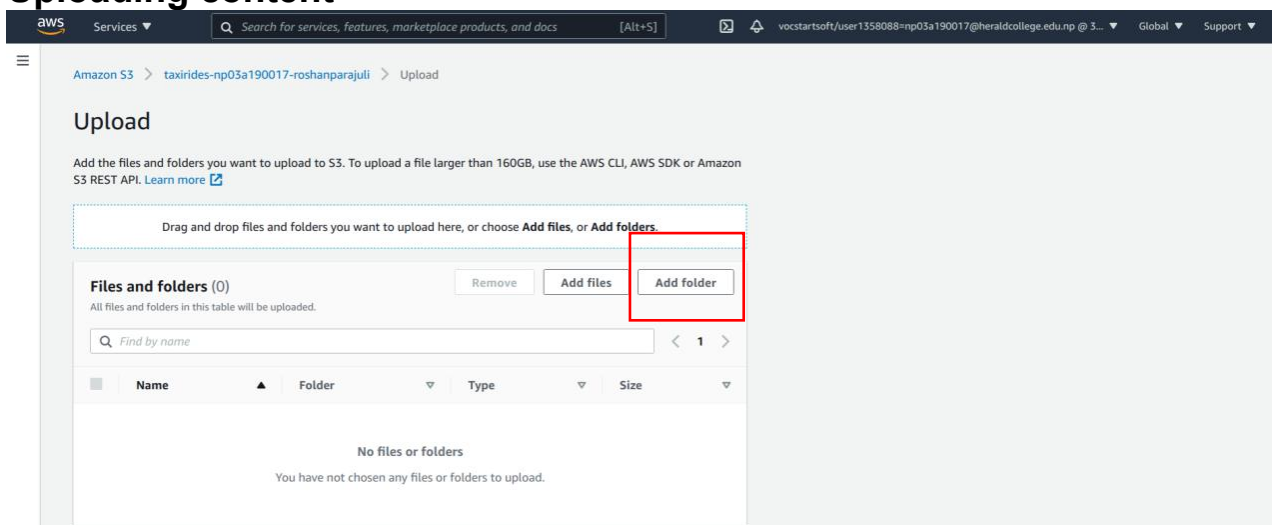


This is the page for the creation of an S3 bucket. A bucket with universally unique name is to be created for uploading the contents of a taxirides app and hosting the site.

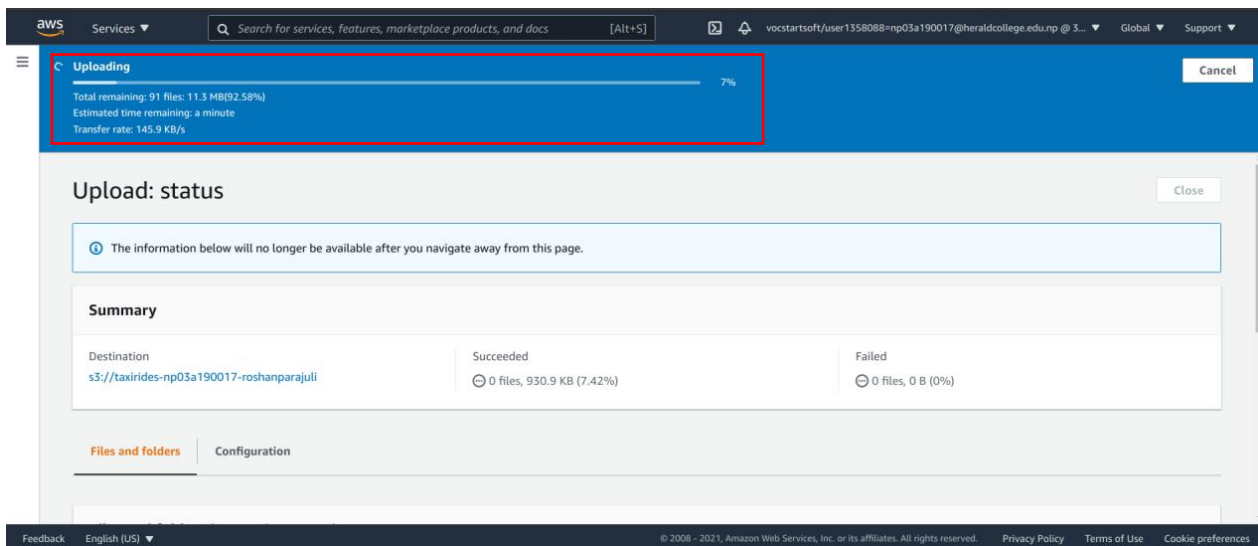


The region is selected to be US East. **AWS Regions** are large and widely dispersed into separate geographic locations with AWS Servers. After creating the bucket in a selected region, the files are to be uploaded.

Uploading content

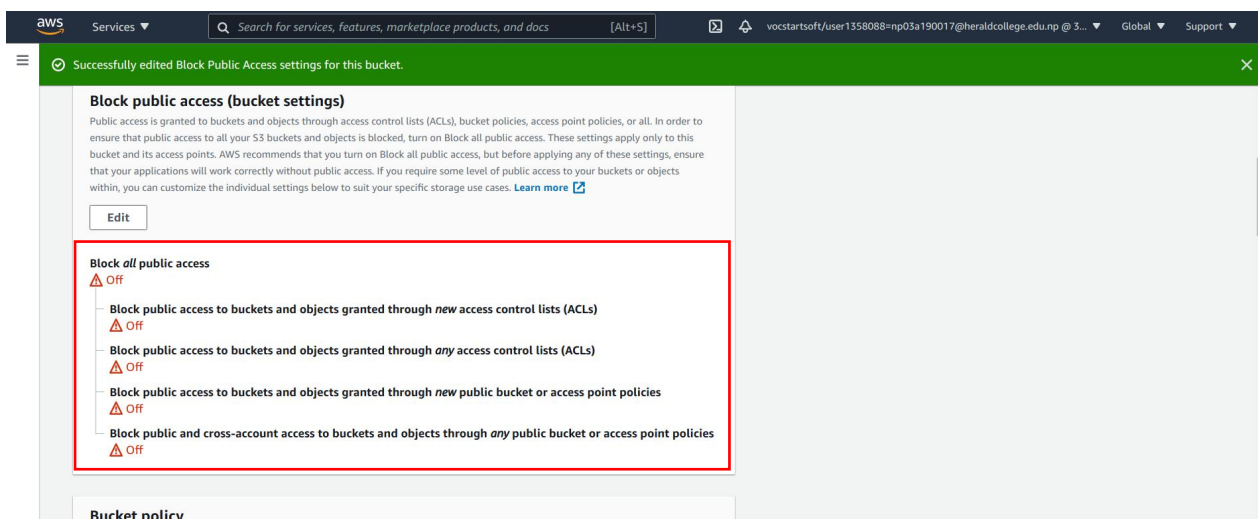


After clicking on the bucket's name, We can choose the "Upload" option and choose the required files for hosting the site.

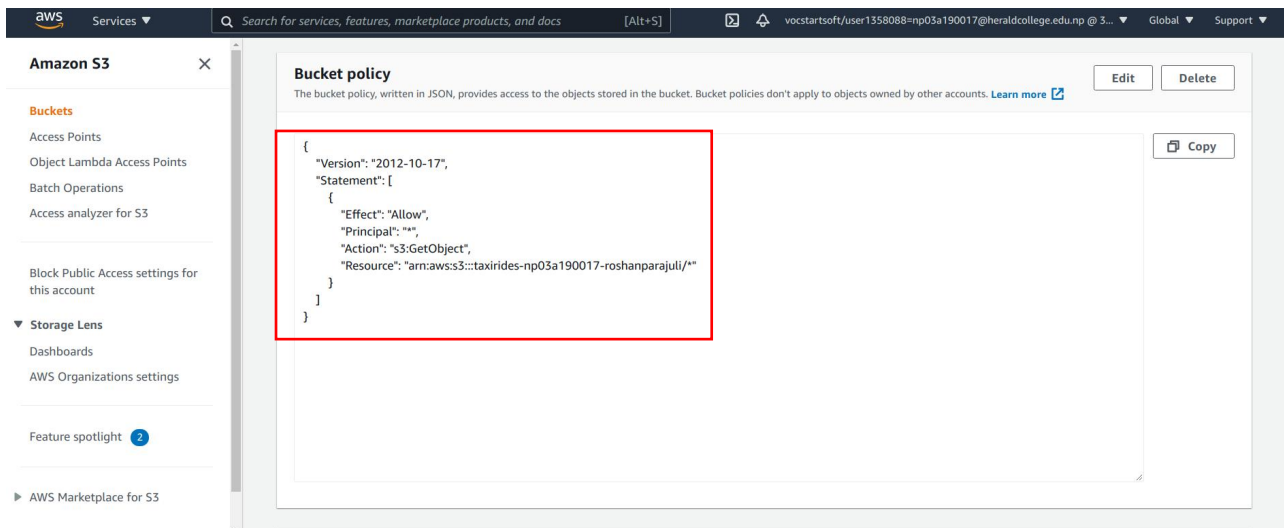


The files are being uploaded into the bucket and the progress is shown in the blue ribbon. After the upload is complete, the contents of the website directory listed in the S3 console.

Adding a Bucket Policy to Allow Public Reads

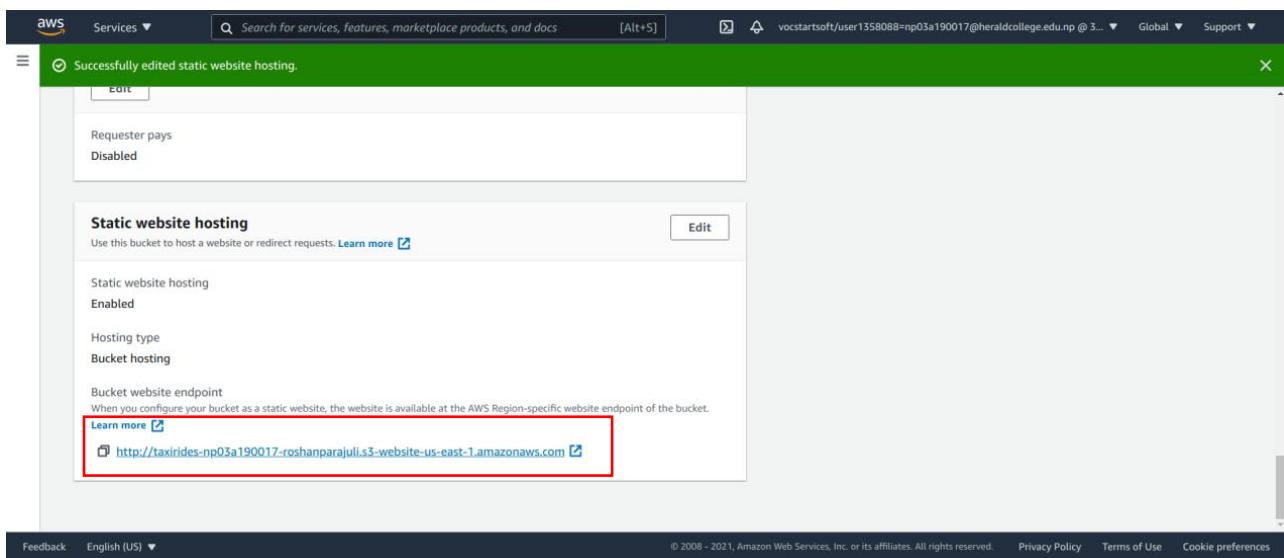


Since the website needs to be public, the public access is turned on the S3 bucket by adding a bucket policy and turning on all the related options. This enables the anonymous users to view the site. If the public access was not turned on, the bucket would only be accessible by authenticated users with access to the AWS account of the person who created the bucket.

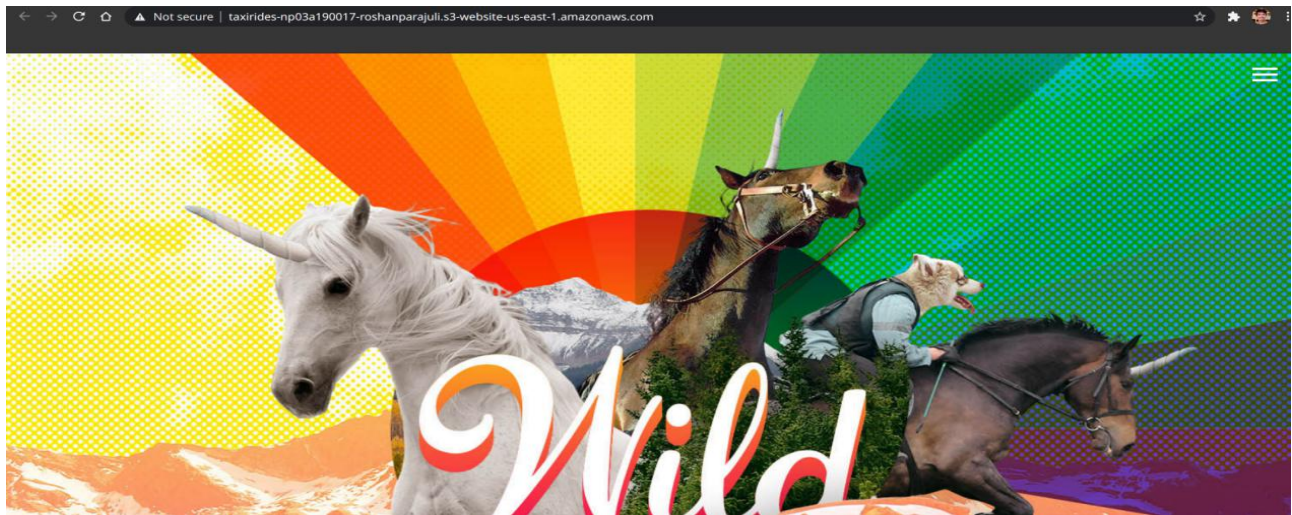


Bucket policy is a JSON document that specifies what principles are allowed to execute various actions against the objects in the bucket.

Enabling website hosting and testing the implementation



On the properties tab, at the end of the page, the domain was noted for the site deployed.



The site is now successfully deployed.

Screenshot of the S3 bucket

taxirides-np03a190017-roshanparajuli

Publicly accessible

Objects Properties Permissions Metrics Management Access Points

Objects (16)

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 Inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

List versions Delete Actions Create folder Upload

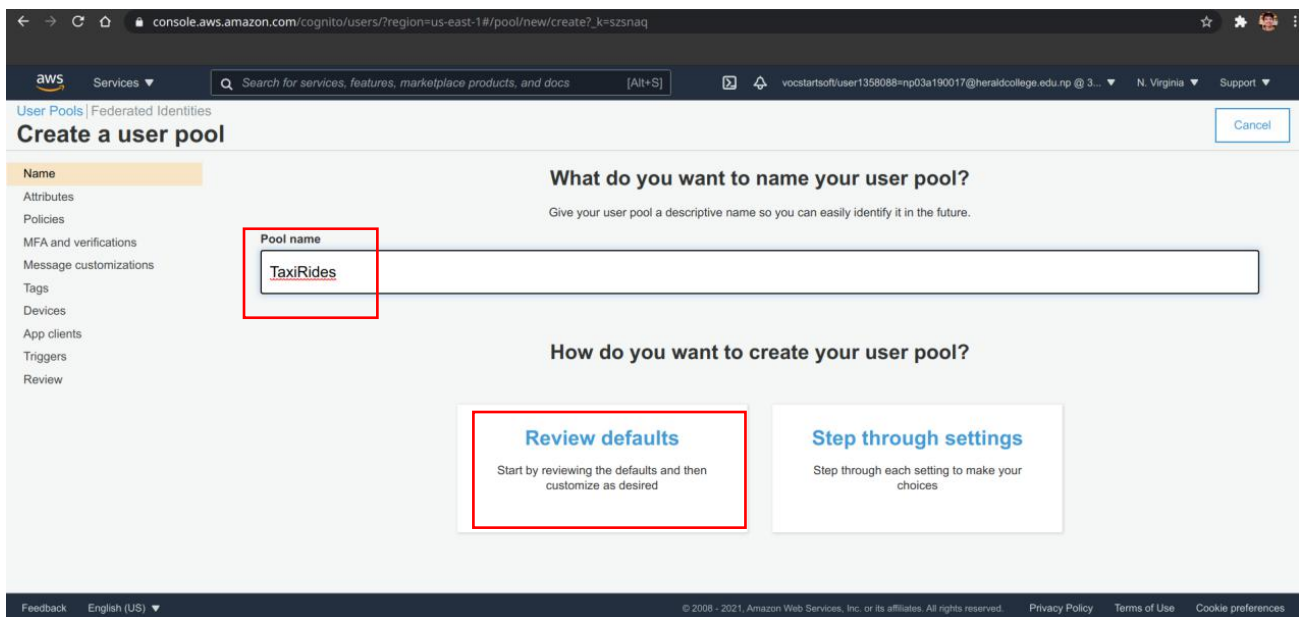
Find objects by prefix

	Name	Type	Last modified	Size	Storage class
<input type="checkbox"/>	apply.html	html	March 24, 2021, 11:10:08 (UTC+05:45)	2.9 KB	Standard
<input type="checkbox"/>	config.js	js	March 24, 2021, 15:52:39 (UTC+05:45)	360.0 B	Standard
<input type="checkbox"/>	css/	Folder	-	-	-
<input type="checkbox"/>	faq.html	html	March 24, 2021, 11:10:09 (UTC+05:45)	6.4 KB	Standard
<input type="checkbox"/>	favicon.ico	ico	March 24, 2021, 11:10:10 (UTC+05:45)	1.4 KB	Standard
<input type="checkbox"/>	fonts/	Folder	-	-	-
<input type="checkbox"/>	images/	Folder	-	-	-
<input type="checkbox"/>	index.html	html	March 24, 2021, 11:10:11 (UTC+05:45)	6.6 KB	Standard
<input type="checkbox"/>	investors.html	html	March 24, 2021, 11:10:12 (UTC+05:45)	5.9 KB	Standard
<input type="checkbox"/>	js/	Folder	-	-	-
<input type="checkbox"/>	register.html	html	March 24, 2021, 11:10:14 (UTC+05:45)	2.7 KB	Standard
<input type="checkbox"/>	ride.html	html	March 24, 2021, 11:10:15 (UTC+05:45)	4.2 KB	Standard
<input type="checkbox"/>	robots.txt	txt	March 24, 2021, 11:10:16 (UTC+05:45)	43.0 B	Standard
<input type="checkbox"/>	signin.html	html	March 24, 2021, 11:10:17 (UTC+05:45)	2.5 KB	Standard
<input type="checkbox"/>	unicorns.html	html	March 24, 2021, 11:10:18 (UTC+05:45)	4.9 KB	Standard
<input type="checkbox"/>	verify.html	html	March 24, 2021, 11:10:20 (UTC+05:45)	2.5 KB	Standard

Task 2: User Management

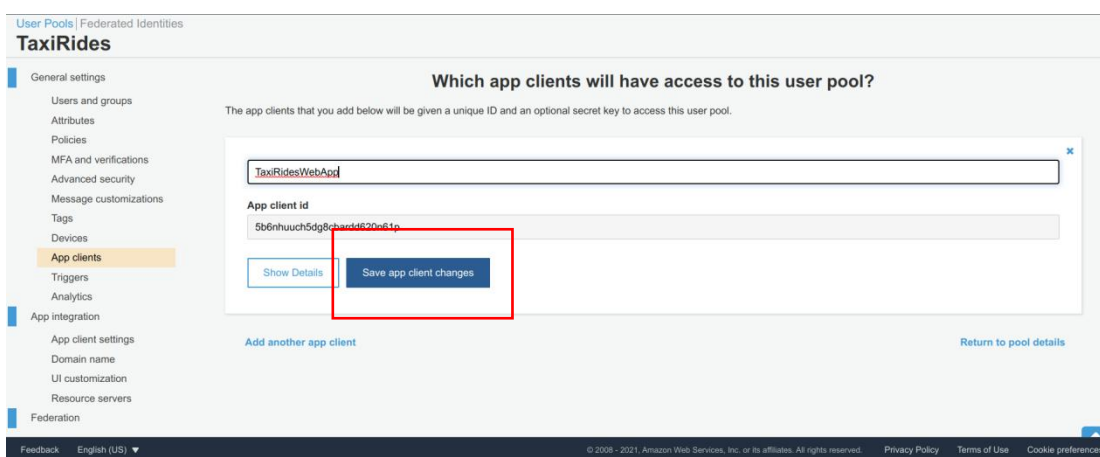
Creating an Amazon Cognito User Pool

Cognito user pool service is used to add the authentication (sign-up and sign-in functionality) to the web app. For that, a new user pool needs to be created at first.



The screenshot shows the AWS Management Console interface for creating a new Cognito user pool. The browser address bar shows the URL: `console.aws.amazon.com/cognito/users/?region=us-east-1#/pool/new/create?_k=szsnaq`. The console header includes the AWS logo, a search bar, and user information. The main heading is "Create a user pool". On the left, a sidebar lists various configuration steps: Name, Attributes, Policies, MFA and verifications, Message customizations, Tags, Devices, App clients, Triggers, and Review. The "Name" step is currently selected. The main content area asks "What do you want to name your user pool?" and provides a text input field where "TaxiRides" has been entered. Below this, it asks "How do you want to create your user pool?" and offers two buttons: "Review defaults" (which is highlighted with a red box) and "Step through settings".

Creating a user pool named TaxiRides and adding a client for the pool with the name TaxiRidesWebApp and the app client id was noted.

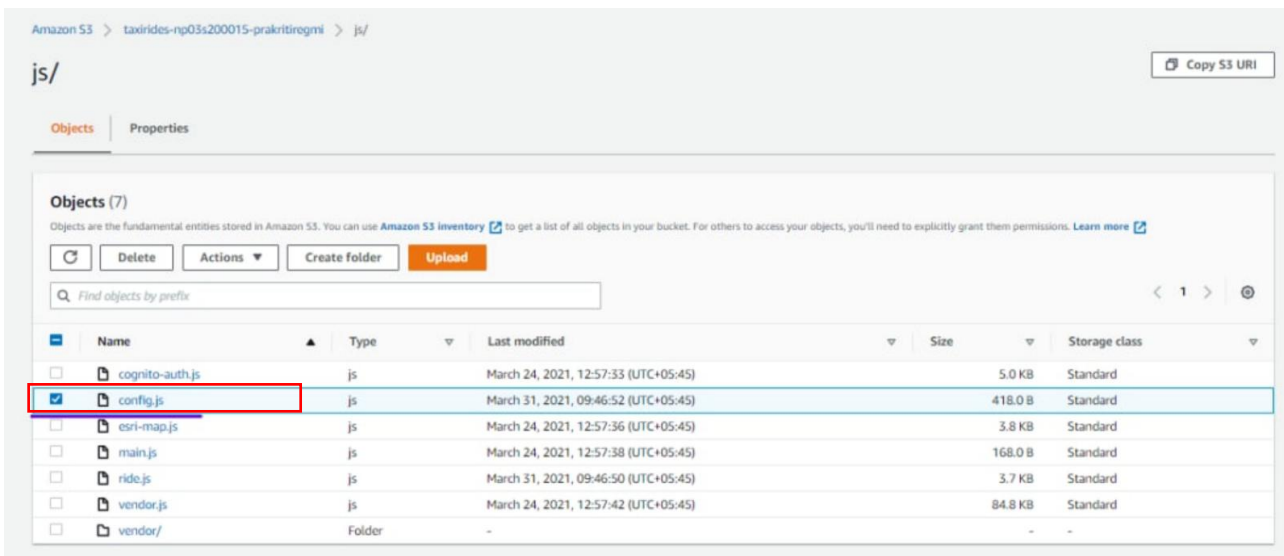


The screenshot shows the AWS Management Console interface for the "TaxiRides" user pool. The left sidebar lists various settings: General settings, Users and groups, Attributes, Policies, MFA and verifications, Advanced security, Message customizations, Tags, Devices, App clients (highlighted), Triggers, Analytics, App integration, App client settings, Domain name, UI customization, Resource servers, and Federation. The main content area is titled "Which app clients will have access to this user pool?" and includes a text input field for the app client name, which contains "TaxiRidesWebApp". Below this, the "App client id" is displayed as "5b6nhuuch5dg8chad4620e61p". A red box highlights the "App client id" field and the "Save app client changes" button. There are also links for "Show Details", "Add another app client", and "Return to pool details".

The config.js file on the js folder was downloaded and the following configuration was changed.

```
1 window._config = {
2   cognito: {
3     userPoolId: 'us-east-1_lSvDn1TXb', // e.g. us-east-2_uXboG5pAb
4     userPoolClientId: '22r5r4bf4fvo8s0pr9frv571c5', // e.g. 25ddkmj4v6hfsfvrhpf7n4hv
5     region: 'us-east-1' // e.g. us-east-2
6   },
7   api: {
8     invokeUrl: '' // e.g. https://rc7nyt4tql.execute-api.us-west-2.amazonaws.com/prod,
9   }
10 };
11
```

The config file now has the configuration files for the cognito services. After editing the file, it was deleted in the server and uploaded again in the same directory.



Amazon S3 > taxirides-np03s200015-prakritirogmi > js/

js/ Copy S3 URI

Objects Properties

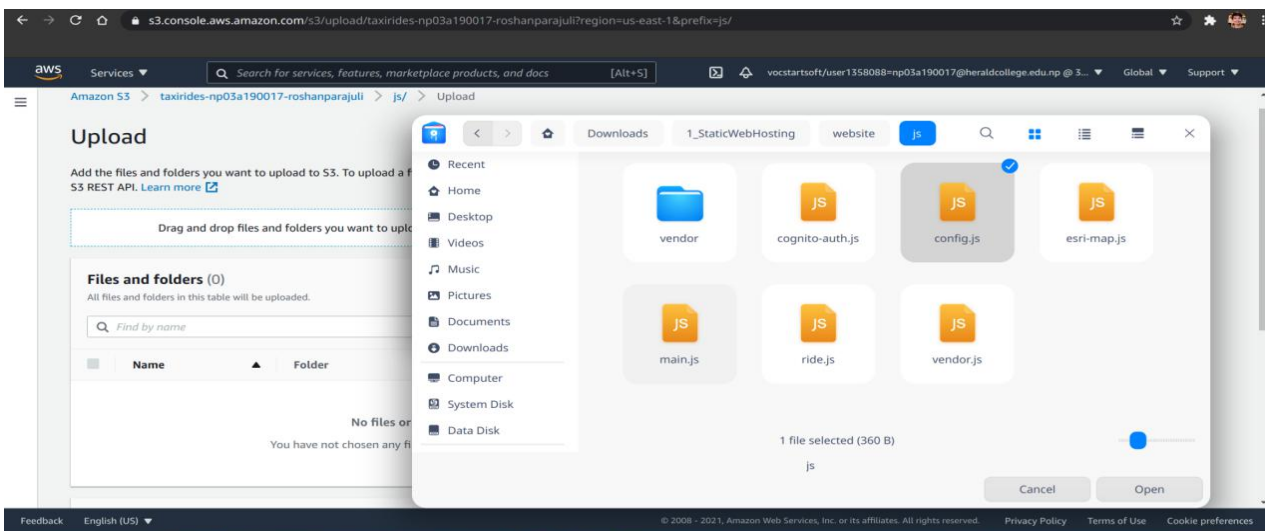
Objects (7)

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

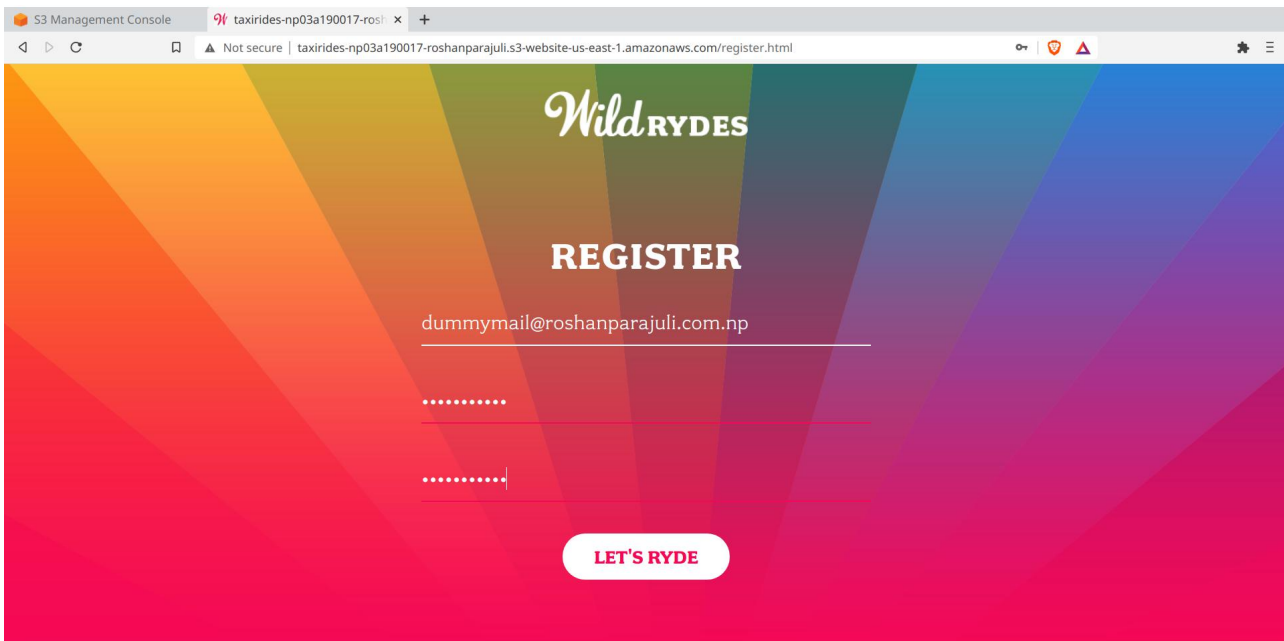
Refresh Delete Actions Create folder Upload

Find objects by prefix

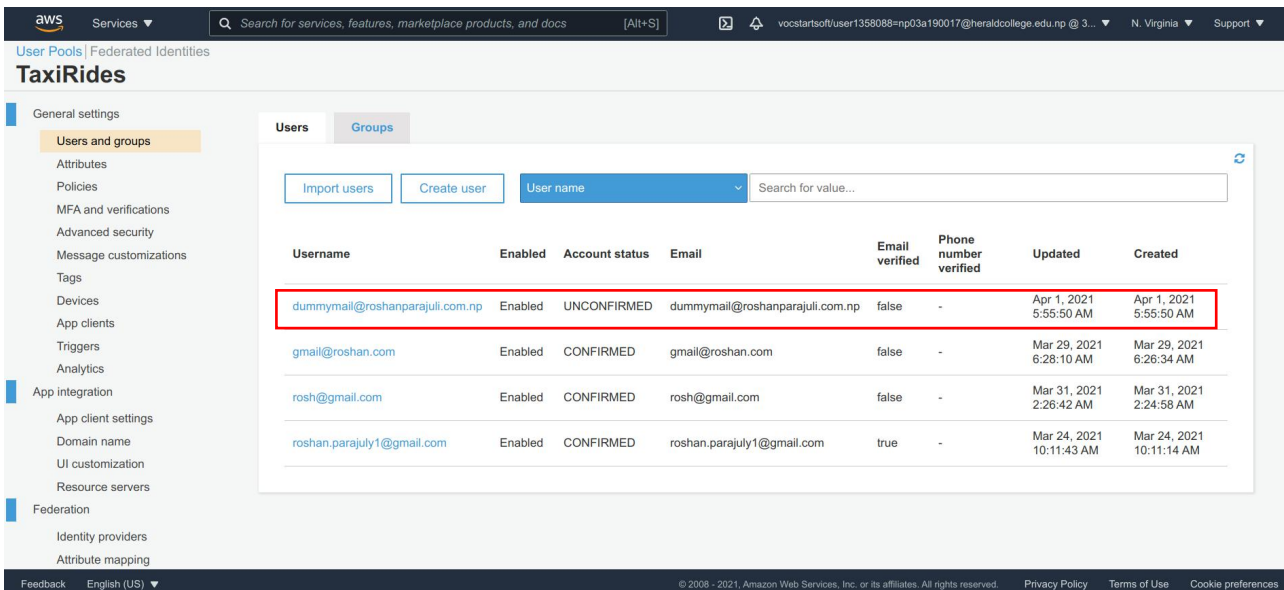
	Name	Type	Last modified	Size	Storage class
<input type="checkbox"/>	cognito-auth.js	js	March 24, 2021, 12:57:33 (UTC+05:45)	5.0 KB	Standard
<input checked="" type="checkbox"/>	config.js	js	March 31, 2021, 09:46:52 (UTC+05:45)	418.0 B	Standard
<input type="checkbox"/>	esri-map.js	js	March 24, 2021, 12:57:36 (UTC+05:45)	3.8 KB	Standard
<input type="checkbox"/>	main.js	js	March 24, 2021, 12:57:38 (UTC+05:45)	168.0 B	Standard
<input type="checkbox"/>	ride.js	js	March 31, 2021, 09:46:50 (UTC+05:45)	3.7 KB	Standard
<input type="checkbox"/>	vendor.js	js	March 24, 2021, 12:57:42 (UTC+05:45)	84.8 KB	Standard
<input type="checkbox"/>	vendor/	Folder	-	-	-



Testing the implementation



A dummy email is registered in the site for testing its implementation.



Username	Enabled	Account status	Email	Email verified	Phone number verified	Updated	Created
dummymail@roshanparajuli.com.np	Enabled	UNCONFIRMED	dummymail@roshanparajuli.com.np	false	-	Apr 1, 2021 5:55:50 AM	Apr 1, 2021 5:55:50 AM
gmail@roshan.com	Enabled	CONFIRMED	gmail@roshan.com	false	-	Mar 29, 2021 6:28:10 AM	Mar 29, 2021 6:28:34 AM
rosh@gmail.com	Enabled	CONFIRMED	rosh@gmail.com	false	-	Mar 31, 2021 2:26:42 AM	Mar 31, 2021 2:24:58 AM
roshan.parajuly1@gmail.com	Enabled	CONFIRMED	roshan.parajuly1@gmail.com	true	-	Mar 24, 2021 10:11:43 AM	Mar 24, 2021 10:11:14 AM

The dummy email registered can now be seen in the “Users and Group” tab in the **cognito** service. It can be clearly seen that the account status is **unconfirmed**.

ride.html page successfully authenticates the load but is not functional because of no API invocation yet.