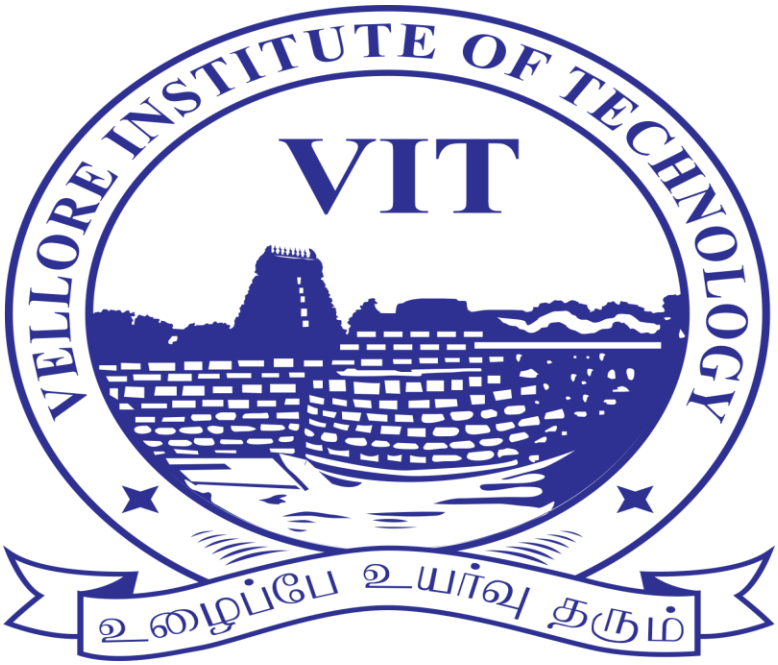


BCSE355L

Project name: *Sportech Connect*

Review 1

Team Members:

Ayush Kumar Acharya (23BPS1078)

Arogya Sharma (23BPS1179)

Project Introduction:

SportTech Connect is designed to be a vibrant and easy-to-use platform that puts the community at its core. Users can effortlessly **share their passion** by uploading photos and videos, whether it's a cool tech gadget or a game-winning highlight. You have complete control over your content and can easily **delete** anything you no longer want to share.

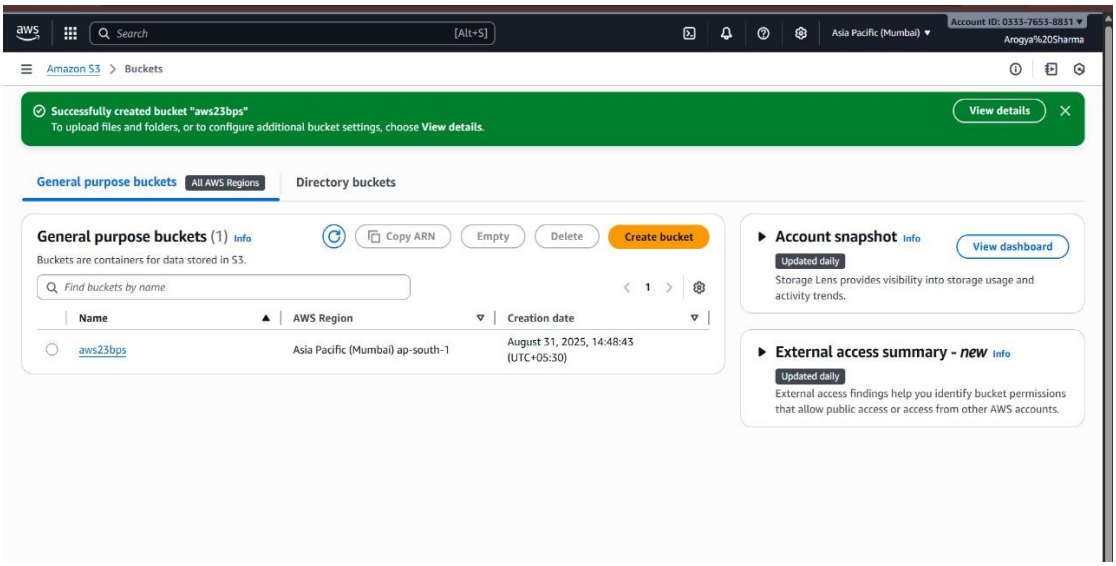
The platform makes it simple to **engage and connect** with others. You can show support with a single tap to **like** a post or join the conversation by leaving a **comment**. We also make it easy to **share** your favorite content both within the app and on other social media platforms like Twitter, so you can bring your passions to a wider audience.

Finally, SportTech Connect helps you **build your community**. You can **create and join groups** dedicated to specific sports teams or areas of technology, finding a dedicated space to discuss your interests. The app will even learn what you love, tailoring your personal feed to show you the most relevant content, helping you stay connected and discover new things every day.

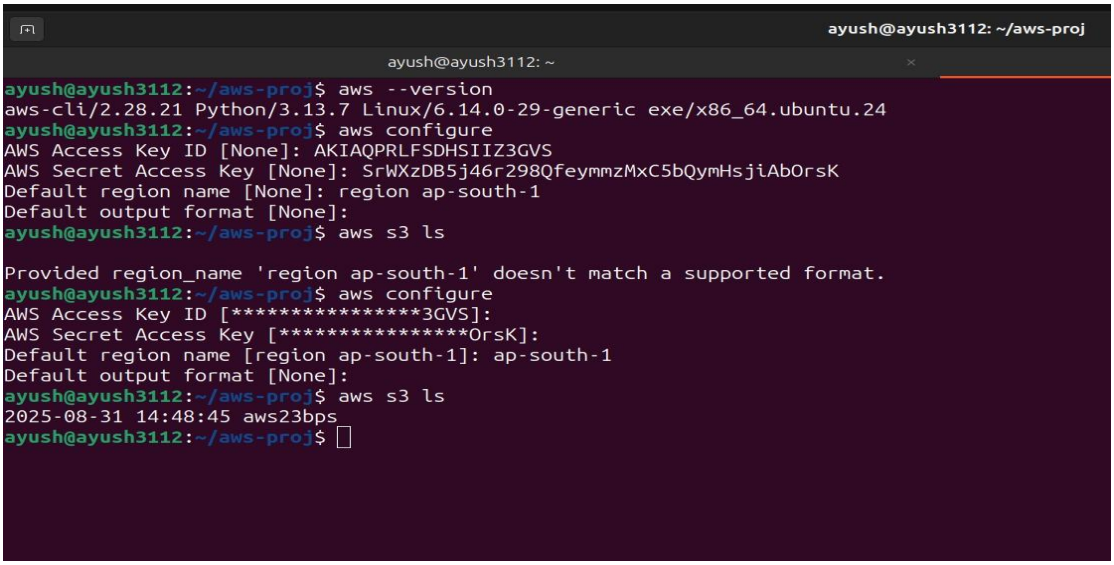
Deliverables:

1. Validate the S3 bucket configuration.

a. S3 Bucket Setup:



b. Setting up S3 Bucket gateway from our Linux Machine:



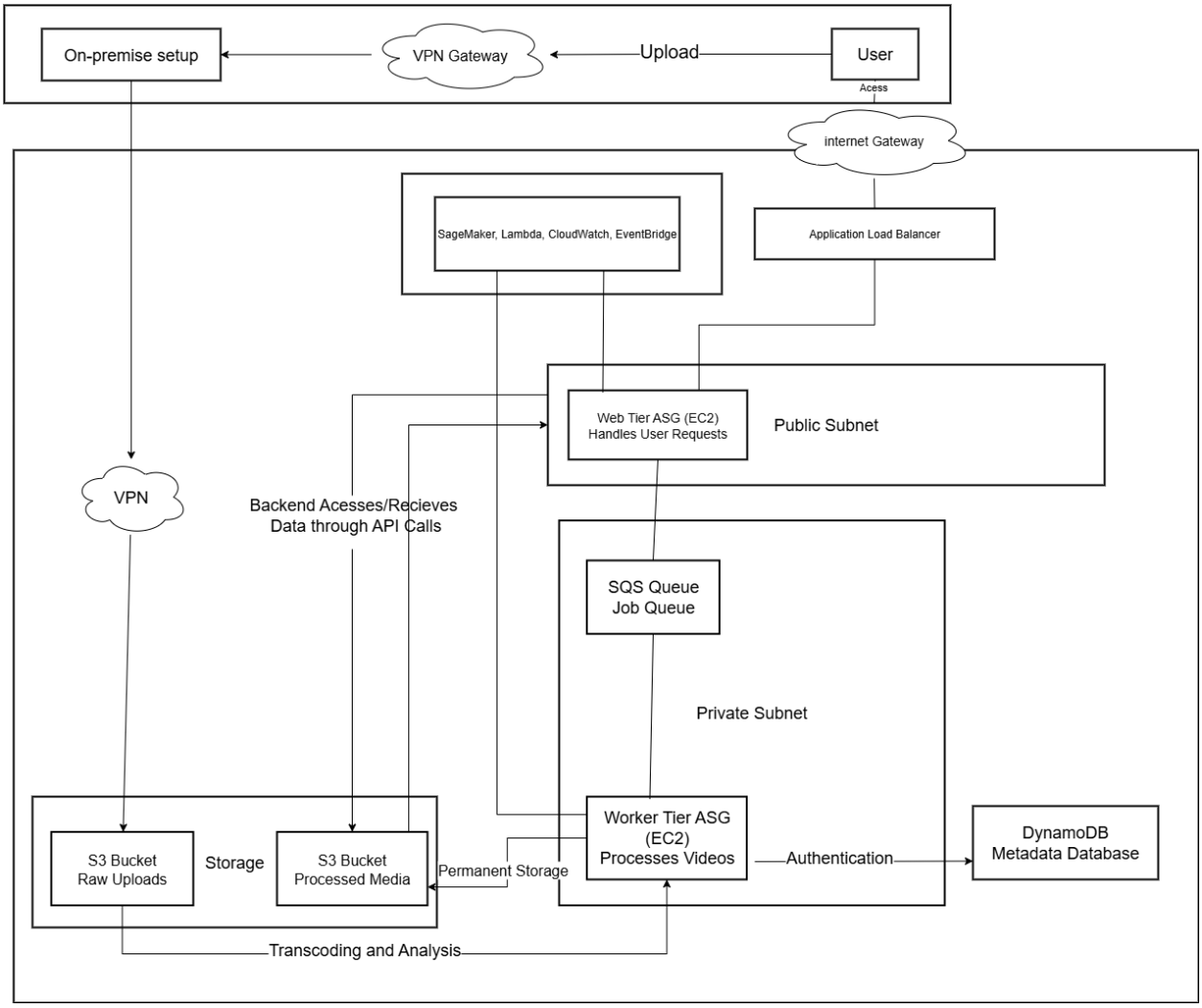
Hybrid Architecture (On-prem + Cloud):

Our website is designed using a hybrid architecture that combines both on-premises and cloud-based components to achieve scalability, efficiency, and reliability. When a user interacts with the platform through the frontend, the request is sent to the backend, which is built using Django or Flask and deployed on out own local on premise Linux machine which is not running on VM ware rather than on our own local machine , exposing its IP through a VPN. This backend layer is responsible for handling authentication, processing requests, validating uploads, and coordinating how the data flows through the system.

The data that users upload is first stored temporarily in an on-premises environment, which acts as a local buffer. This ensures that the system continues to function smoothly even if there are short interruptions in cloud connectivity. Once validated, the backend securely pushes the data to AWS S3, which serves as the main cloud storage solution. By doing this, we maintain the advantage of having quick local data handling while also benefiting from the scalability and durability of cloud infrastructure.

This hybrid setup allows the application to function in a flexible and cost-effective way. The local server ensures fast response times and reliability, while the cloud storage guarantees that large amounts of data can be stored and accessed globally without limitations. In essence, the hybrid architecture enables our platform to provide a smooth user experience while ensuring that the underlying system is resilient, scalable, and optimized for future growth.

Network Architecture:



Checklist of AWS resources:

1. EC2:

Provides the powerful virtual servers needed to process and transcode high-definition sports clips.

2. CloudWatch:
   Monitors SporTech's real-time traffic and server load to feed our predictive scaling model.

3. Lambda:
   Acts as the serverless brain that orchestrates our entire predictive auto-scaling strategy.

4. VPC:
   Creates a secure and isolated network environment to protect SporTech's backend infrastructure.

5. CloudFront:
   Delivers sports highlights to fans globally with low latency for a smooth viewing experience.

6. SQS:
   Manages the queue of incoming sports videos, ensuring every clip is processed reliably.

7. Auto Scaling Groups:
   Automatically adjusts the number of servers to handle surges during live game uploads.

8. Load Balancer:
   Distributes incoming fan traffic evenly across our web servers to prevent crashes.
9. RDS:
   Stores all essential SporTech data, including user profiles, video metadata, and comments.

10. S3 Bucket:

Provides scalable and durable storage for every sports highlight video uploaded to the platform.

11. Cognito:
Manages secure sign-up and login for all athletes, teams, and fans on SporTech.

12. Route 53:
Acts as the address for SporTech, connecting our domain name (e.g., https://www.google.com/search?q=sportech.com) to the application.

On Prem Simulation:

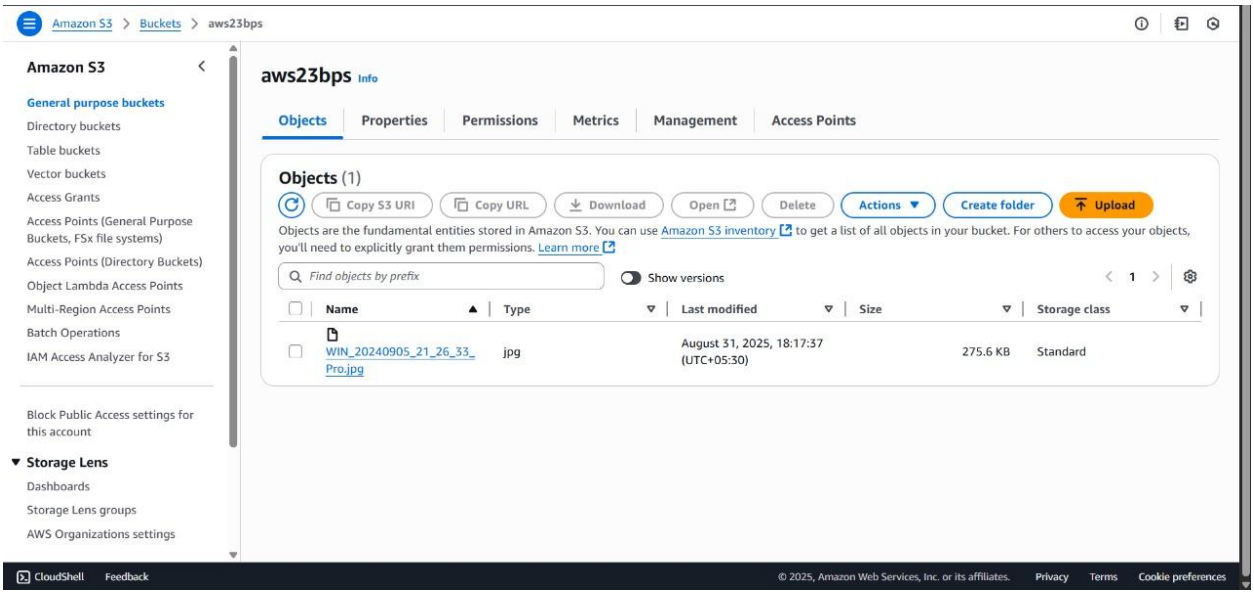
FrontEnd:

Backend Transferring Data to S3:

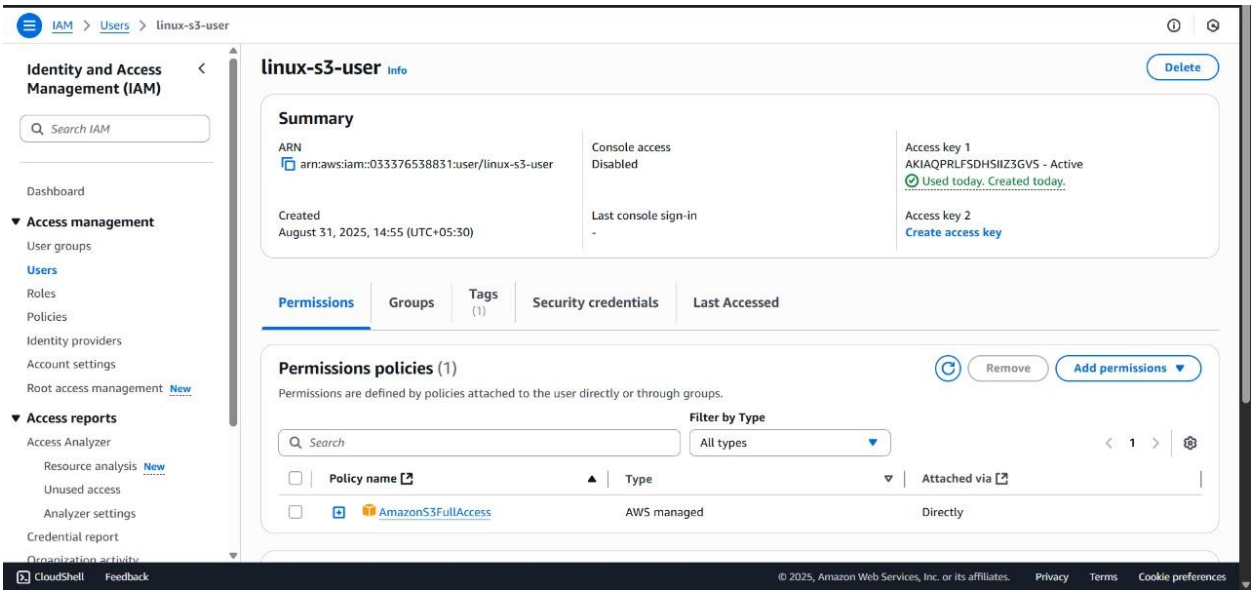
```
(aws-project-env) ayush@ayush3112:~/aws-proj$ python3 test.py
/home/ayush/aws-proj/test.py:3: DeprecationWarning: 'cgi' is deprecated and slated for removal in Python 3.13
  import cgi
Serving HTTP on 172.20.10.2 port 8000 (http://172.20.10.2:8000/) ...
172.20.10.4 - - [31/Aug/2025 18:17:35] "POST /upload HTTP/1.1" 200 -
172.20.10.4 - - [31/Aug/2025 18:17:37] "POST /upload HTTP/1.1" 200 -
```

The Django backend recieves data and send it to a temporary S3 bucket , where it stores it and then sends it to a ec2 instance for further analysis.

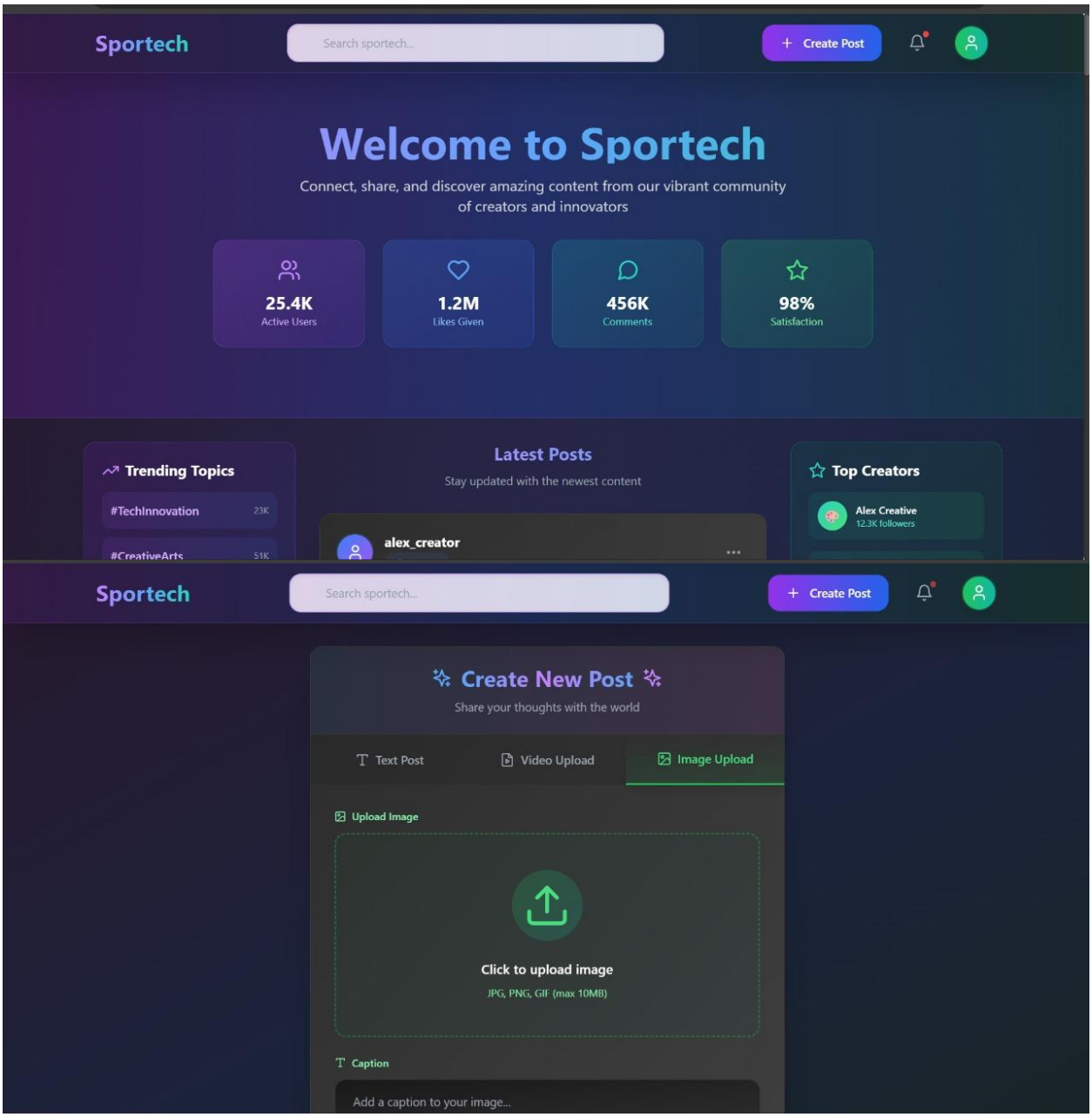
S3 Receiving Files:



IAM Roles Specifications:



Frontend :


Sportech
Search sportech...
Create Post
Welcome to Sportech
Connect, share, and discover amazing content from our vibrant community of creators and innovators
25.4K
Active Users
1.2M
Likes Given
456K
Comments
98%
Satisfaction
Trending Topics
#TechInnovation
23K
#CreativeArts
51K
Latest Posts
Stay updated with the newest content
alex_creator
Top Creators
Alex Creative
12.3K followers
Sportech
Search sportech...
Create Post
Create New Post
Share your thoughts with the world
Text Post
Video Upload
Image Upload
Upload Image
Click to upload image
JPG, PNG, GIF (max 10MB)
Caption
Add a caption to your image...

Timeline of Project:

Phase 1: Research and Planning

Dates: Aug 1 - Aug 20

- **Research:** We will define our core project idea and determine the best solutions and resources to use.
- **AWS Selection:** We will select the most suitable AWS services that will power our platform.
- **Architecture & Feasibility:** We will plan out the app's structure and tech stack to ensure it's a realistic and workable project.

Phase 2: Prototyping and Initial Setup

Dates: Aug 21 - Aug 31

- **AWS Setup:** We will begin setting up and configuring the AWS resources we selected.
- **Basic Website:** We will create a simple, working version of the website to test the foundational features.
- **User Setup:** We will establish the system for user accounts, including their different IAM roles and permissions.

Phase 3: Core Development and Deployment

Dates: Sep 2 - Sep 21

- **Resource Integration:** We will connect all the different AWS services so they work together seamlessly.
- **Photo/Video Handling:** We will build the key functionality for users to upload and manage their photos and videos.
- **Website Deployment:** We will launch the website, making it live and accessible to users.

Phase 4: Advanced Features and Optimization

Dates: Sep 15 - Sep 22

- **Performance Analysis:** We will analyze the performance of our auto-scaling to ensure the site can handle high traffic.
- **ML Optimization:** We will test the machine learning model to enhance the auto scaling functionality , and note how it is better than normal autoscaling.
- **Global Launch:** We will prepare the platform for a worldwide release to a global audience.