

“MEME MATCHING GAME”

Mini- Project

(Third Year/ Sem VI)

Submitted in fulfilment of the requirement of

University of Mumbai For the Degree of

Bachelor Of Engineering

(Computer Engineering)

By

1.	VEDANT SHIMPI	TE COMPS B-03	TU3F2223055
2.	ADITYA GUPTA	TE COMPS B-04	TU3F2223058
3.	PRANJAL BHATT	TE COMPS B-30	TU3F2223089

Under the Guidance of

Prof. Preeti Patil



Department of Computer Engineering

TERNA ENGINEERING COLLEGE

Plot no.12, Sector-22, Opp. Nerul Railway station,

Phase-11, Nerul (w), Navi Mumbai 400706

UNIVERSITY OF MUMBAI

2024-2025

Internal Approval Sheet



Terna Engineering College
NERUL, NAVI MUMBAI

CERTIFICATE

This is to certify that

VEDANT SHIMPI	TU3F2223055
ADITYA GUPTA	TU3F2223058
PRANJAL BHATT	TU3F2223089

Has satisfactorily completed the requirements of the Mini Project
(Third Year/Sem VI)
entitled

“MEME MATCHING GAME”

As prescribed by the University of Mumbai

Under the guidance of
Prof. Preeti Patil

Guide

Head of Department

Principal

Approval Sheet

Project Report Approval

This Mini Project Report entitled

“MEME MATCHING GAME”

by the following students is approved for the degree of Bachelor in
"Computer Engineering (Semester VI)".

Submitted by:

VEDANT SHIMPI	TU3F2223055
ADITYA GUPTA	TU3F2223058
PRANJAL BHATT	TU3F2223089

Examiners Name & Signature:

1. -----

2. -----

Date:

Place: NERUL

DECLARATION

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced The cartoon sources task-specific. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

VEDANT SHIMPI	TU3F2223055	
ADITYA GUPTA	TU3F2223058	
PRANJAL BHATT	TU3F2223089	

Date:

Place: NERUL

ACKNOWLEDGEMENT

We would like to express our sincere gratitude towards our guide and Mini Project Coordinator **Prof. Preeti Patil**, for their help, guidance and encouragement, they provided during the project development. This work would have not been possible without their valuable time, patience and motivation. We thank them for making our stint thoroughly pleasant and enriching. It was great learning and an honor being their student.

We are deeply thankful to **Dr. Kishor Sakure (H.O.D Computer Department)** and entire team in the Computer Department. They supported us with scientific guidance, advice and encouragement, they were always helpful and enthusiastic and this inspired us in our work.

We take the privilege to express our sincere thanks to **Dr. L. K. Ragha** our Principal for providing the encouragement and much support throughout our work.

VEDANT SHIMPI	TU3F2223055	
ADITYA GUPTA	TU3F2223058	
PRANJAL BHATT	TU3F2223089	

Date:

Place: NERUL

ABSTRACT

The Meme Matching Game is a web-based memory game developed using HTML, CSS, and JavaScript. It displays a dynamic grid of meme cards that users flip to find matching pairs, with matched pairs disappearing upon a correct match. Once all pairs are found, the grid resets with shuffled positions. The game integrates cloud services such as pipelines for continuous deployment and cloud storage buckets for hosting and managing meme assets. This project demonstrates core concepts of client-side development, cloud integration, and efficient asset management.

LIST OF FIGURES

Figure No.	Figure Name	Page No.
1	Architecture/Message Flow	13
2	Cloud Computing Model	17
3	Cloud Computing Deployment Model	17
4	Working of Amazon S3	20

LIST OF ABBREVIATIONS

Acronym	Abbreviation
AWS	Amazon Web Services
IaaS	Infrastructure as a Service
PaaS	Platform as a Service
SaaS	Software as a Service
NLU	Natural Language Understanding
ASR	Automatic Speech Recognition
SLU	Speech-Language Understanding
IVR	Interactive Voice Response
API	Application Programming Interface
NTTS	Neural Text-to-Speech
SNS	Simple Notification Service
IAM	Identity and Access Management

TABLE OF CONTENTS		
Sr.No	TITLE	Page No.
CHAPTER 1	INTRODUCTION	11
	1.1 Introduction to Amazon Web Services	11
	1.2 Introduction to S3 and Pipeline	12
	1.3. Features of S3	12
	1.4 The Need for S3 and Pipeline	13
	1.5 Architecture/ Message Flow	13
CHAPTER 2	PROBLEM STATEMENT	14
CHAPTER 3	METHODOLOGY	15
	3.1 Amazon Web Services	15
	3.2 How AWS Works	15
	3.3 Types of Cloud Computing	16
	3.4 Amazon S3	18
	3.5 Working of Amazon S3	20
	3.6 Applications of Amazon S3 and CI CD pipeline	21
	3.7 Key Features of Amazon S3 and Pipeline	22
	3.8 Amazon S3 and Pipeline - Use Cases	25
CHAPTER 4	CREATING MEME MATCHING GAME	27
	4.1 Steps of creating a meme matching game	27

CHAPTER 5	SNAPSHOTS	33
	5.1 Starting phase of the game	33
	5.2 Flipping of Cards	34
CHAPTER 6	CONCLUSION	35
	REFERENCES	36

CHAPTER 1

INTRODUCTION

1.1 Introduction to Amazon Web Services



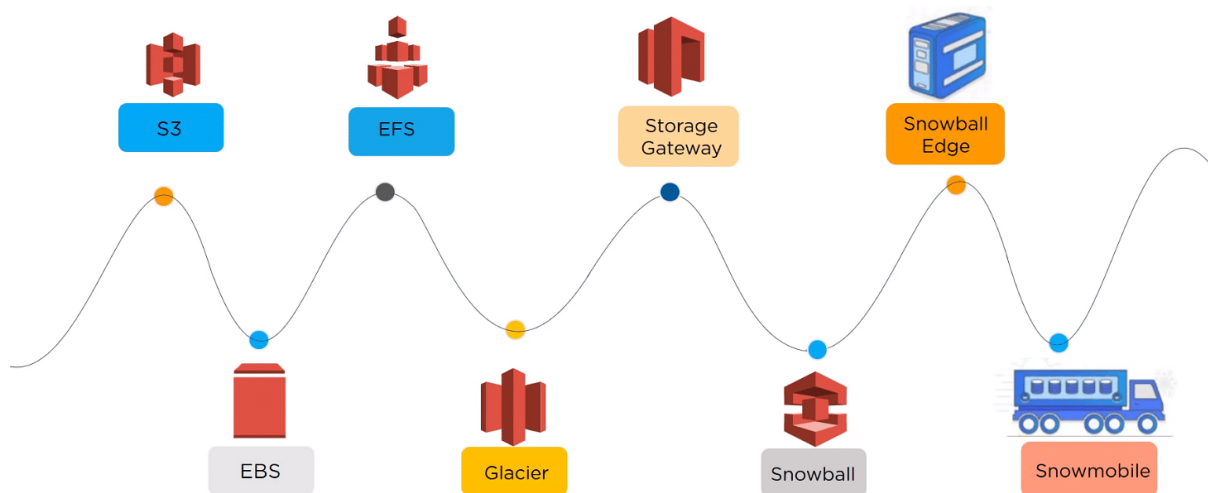
Digitalisation, the surge of mobile and internet-connected devices has revolutionised the way people interact with one another and communicate with businesses” (Eeuwen, M.V. (2017). Millennials are accepting and supporting new technology into the routine of their everyday life, this is becoming more and more prevalent as technology companies are streamlining Artificial Intelligence (AI) into the products they offer, such as; Google Assistant, Google Home and Amazon Alexa. The new and upcoming generation is expected to be critical and game-changing customers for businesses. “They demand effortless experiences, answers within seconds, not minutes and more intelligent self-service options” (Teller Vision, 2017). Most businesses and organisations are understanding the potential benefits of machine learning and artificial intelligence to have a positive change in how they perform business. Artificial intelligence has progressed to allow the development of more sophisticated chatbots. Organisations are focusing on specific areas of user engagement that take up a lot of time but can be replaced through the use of a chatbot. Chatbots can understand what the customer needs from a single text instead of the customer having to follow a process of multiple steps. Chatbots are used to automate customer service and reduce manual tedious tasks performed by employees so they can spend their time more productively on higher priority tasks.

1.2 Introduction to S3 and Pipeline

Amazon S3 (Simple Storage Service) is an object storage service offered by AWS that provides scalable, secure, and durable storage for any amount of data. In this project, S3 is used to host and serve static assets such as meme images, enabling fast and reliable access across the web.

CI/CD Pipeline refers to a set of automated processes that enable continuous integration and continuous deployment. In this project, a pipeline is configured to automatically build, test, and deploy the web application to ensure rapid development and reliable releases.

1.3 Features of Amazon Lex



1.4 The Need for S3 and CI CD Pipeline

- S3 is essential for hosting static assets (like meme images) used in the game. It ensures fast content delivery and easy management of media files without burdening the web server.
- Pipeline automates the build and deployment process, reducing manual errors and accelerating updates. It ensures that every code change is automatically tested and deployed, maintaining high code quality and reducing downtime.

1.5 Architecture/Message Flow

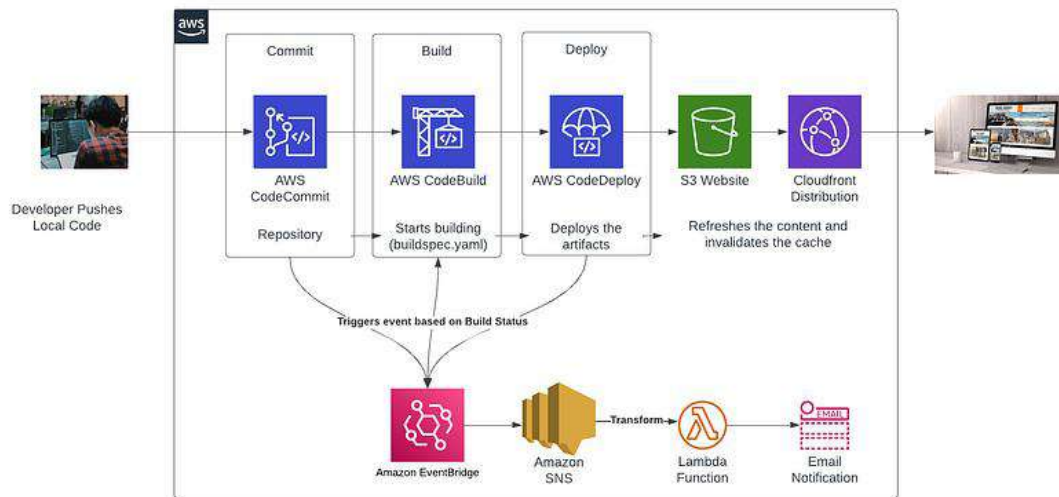


Figure 2: Architecture/Message Flow

CHAPTER 2

PROBLEM STATEMENT

In today's digital era, there is a growing demand for lightweight, engaging, and educational web-based games that not only entertain users but also enhance cognitive skills such as memory and pattern recognition. Traditional memory games often use generic icons or shapes, which may not appeal to a modern audience. There is a need for a more relatable and culturally relevant variation.

This project addresses that need by developing a meme-themed matching game that leverages popular internet culture to increase user engagement. Additionally, managing media assets and ensuring seamless deployment are challenges that are tackled using cloud-based storage (Amazon S3) and CI/CD pipelines to automate development and delivery workflows.

CHAPTER 3

METHODOLOGY

3.1 Amazon Web Services



AWS (Amazon Web Services) is a comprehensive, evolving cloud computing platform provided by Amazon that includes a mixture of infrastructure as a service (IaaS), platform as a service (PaaS) and packaged software as a service (SaaS) offerings. AWS services can offer organisation tools such as compute power, database storage and content delivery services.

AWS launched in 2006 from the internal infrastructure that Amazon.com built to handle its online retail operations. AWS was one of the first companies to introduce a pay-as-you-go cloud computing model that scales to provide users with compute, storage or throughput as needed.

AWS offers many different tools and solutions for enterprises and software developers that can be used in data centres in up to 190 countries. Groups such as government agencies, education institutions, nonprofits and private organisations can use AWS services.

3.2 How AWS Works?

- AWS is separated into different services; each can be configured in different ways based on the user's needs. Users should be able to see configuration options and individual server maps for an AWS service.
- More than 100 services comprise the Amazon Web Services portfolio, including those for compute, databases, infrastructure management, application development and security. These services, by category, include:
 - Compute

- Storage databases
- Data management
- Migration
- Hybrid cloud
- Networking
- Development tools
- Management
- Monitoring
- Security
- Governance
- Big data management
- Analytics
- Artificial intelligence (AI)
- Mobile development
- Messages and notification

3.3 Types of Cloud Computing

- Cloud computing is providing developers and IT departments with the ability to focus on what matters most and avoid undifferentiated work like procurement, maintenance, and capacity planning. As cloud computing has grown in popularity, several different models and deployment strategies have emerged to help meet the specific needs of different users. Each type of cloud service, and deployment method, provides you with different levels of control, flexibility, and management. Understanding the differences between Infrastructure as a Service, Platform as a Service, and Software as a Service, as well as what deployment strategies you can use, can help you decide what set of services is right for your needs.

Cloud Computing Models

There are three main models for cloud computing. Each model represents a different part of the cloud computing stack.



Infrastructure as a Service (IaaS)

Infrastructure as a Service, sometimes abbreviated as IaaS, contains the basic building blocks for cloud IT and typically provide access to networking features, computers (virtual or on dedicated hardware), and data storage space. Infrastructure as a Service provides you with the highest level of flexibility and management control over your IT resources and is most similar to existing IT resources that many IT departments and developers are familiar with today.



Platform as a Service (PaaS)

Platforms as a service remove the need for organizations to manage the underlying infrastructure (usually hardware and operating systems) and allow you to focus on the deployment and management of your applications. This helps you be more efficient as you don't need to worry about resource procurement, capacity planning, software maintenance, patching, or any of the other undifferentiated heavy lifting involved in running your application.

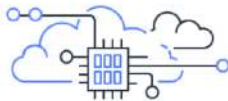


Software as a Service (SaaS)

Software as a Service provides you with a completed product that is run and managed by the service provider. In most cases, people referring to Software as a Service are referring to end-user applications. With a SaaS offering you do not have to think about how the service is maintained or how the underlying infrastructure is managed; you only need to think about how you will use that particular piece of software. A common example of a SaaS application is web-based email where you can send and receive email without having to manage feature additions to the email product or maintaining the servers and operating systems that the email program is running on.

Figure 3: Cloud Computing Model

Cloud Computing Deployment Models



Cloud

A cloud-based application is fully deployed in the cloud and all parts of the application run in the cloud. Applications in the cloud have either been created in the cloud or have been migrated from an existing infrastructure to take advantage of the [benefits of cloud computing](#). Cloud-based applications can be built on low-level infrastructure pieces or can use higher level services that provide abstraction from the management, architecting, and scaling requirements of core infrastructure.



Hybrid

A hybrid deployment is a way to connect infrastructure and applications between cloud-based resources and existing resources that are not located in the cloud. The most common method of hybrid deployment is between the cloud and existing on-premises infrastructure to extend, and grow, an organization's infrastructure into the cloud while connecting cloud resources to internal system. For more information on how AWS can help you with your hybrid deployment, please visit [our hybrid page](#).



On-premises

Deploying resources on-premises, using virtualization and resource management tools, is sometimes called "private cloud". On-premises deployment does not provide many of the benefits of cloud computing but is sometimes sought for its ability to provide [dedicated resources](#). In most cases this deployment model is the same as legacy IT infrastructure while using application management and virtualization technologies to try and increase resource utilization.

Figure 4: Cloud Computing Deployment Model

3.4 Amazon S3



Amazon S3 (Simple Storage Service)

Amazon S3 is a highly scalable, secure, and durable object storage service offered by AWS. It allows developers to store and retrieve any amount of data at any time, from anywhere on the web. Amazon S3 is designed for 99.999999999% (11 9's) durability and provides robust infrastructure to support mission-critical storage needs. It's widely used for storing static assets such as images, videos, backups, log files, and entire static websites.

With Amazon S3, developers and organizations can easily manage, store, and share large volumes of data, benefiting from built-in security features, access management, and extensive integrations across the AWS ecosystem.

Amazon S3 is accessible through the AWS Management Console, AWS CLI, SDKs, and RESTful APIs, enabling developers to automate workflows and integrate S3 storage with applications and services in the cloud or on-premises.

Some of the Benefits of Using Amazon S3 Include:

● Scalability and Performance

Amazon S3 automatically scales to handle large amounts of data and high request rates. You don't need to worry about provisioning storage or managing infrastructure — S3 grows with your storage needs.

● Durability and Availability

Amazon S3 is designed for 99.999999999% durability and 99.99% availability of objects over a given year. It stores data across multiple geographically separated facilities, ensuring

resilience and protection against data loss.

● **Cost-Effectiveness**

Amazon S3 offers flexible pricing models including a pay-as-you-go option and multiple storage classes (Standard, Intelligent-Tiering, Glacier, etc.) to optimize cost based on data access patterns. The S3 Free Tier makes it easy for developers to get started without upfront investment.

● **Security and Access Management**

Amazon S3 supports encryption at rest and in transit, fine-grained access control via bucket policies, IAM roles, and AWS KMS integration. This ensures that data remains secure and compliant with various security standards.

● **Static Website Hosting**

S3 allows developers to host static websites directly from a bucket. This makes it ideal for simple web apps and games, such as the Meme Matching Game, where HTML, CSS, JS, and media assets can be served directly from S3.

● **Seamless Integration with AWS Services**

Amazon S3 integrates natively with other AWS services such as AWS Lambda, Amazon CloudFront (CDN), Amazon CloudWatch (monitoring), AWS Glue (data processing), and Amazon Athena (querying data in S3). These integrations enable powerful workflows and serverless architectures.

● **Easy Data Management and Automation**

Using features like S3 Lifecycle Policies, versioning, and replication, users can automate data management tasks like archiving old files, backing up data across regions, and retaining previous versions of important objects.

● **Global Availability**

Amazon S3 is available in multiple AWS regions around the world, ensuring low-latency

access and redundancy. You can select the appropriate region for compliance, performance, and cost considerations.

3.5 Working of Amazon S3



Figure 5: Working of Amazon S3

Steps to Follow While Working with Amazon S3

1. **Create an S3 Bucket.**
2. **Upload Objects (Files)**
3. **Set Permissions and Access Control**
4. **Access and Use the S3 Assets**

3.6 Applications of Amazon S3 and Pipeline:

Applications of CI/CD Pipeline

1. **Automated Deployment**

Pipelines enable automatic deployment of applications after every code change, ensuring faster release cycles and minimal manual intervention.

2. **Continuous Integration and Testing**

Pipelines help in automatically building and testing code after every commit, improving code quality and detecting bugs early.

3. **Version Control Integration**

Easily integrates with platforms like GitHub or GitLab to trigger deployments based on specific branches or tags.

4. **Rollback Support**

Pipelines allow easy rollback to previous stable versions if a deployment fails or contains bugs.

5. **Multi-Environment Deployment**

CI/CD pipelines support staged deployments (e.g., dev, test, prod), allowing safe and structured application delivery.

6. **Developer Productivity**

Automating repetitive tasks reduces manual errors and lets developers focus more on writing code.

Applications of Amazon S3

1. **Static Website Hosting**

S3 is widely used to host static websites and frontend applications with HTML, CSS, and JavaScript directly from a bucket.

2. **Media Storage and Delivery**

Stores images, videos, and audio files with high durability, and delivers them globally using services like Amazon CloudFront.

3. **Backup and Archival**

Frequently used for backup of files, logs, and databases with automated lifecycle policies to move data to cheaper storage tiers (e.g., Glacier).

4. **Big Data and Analytics**

Acts as a data lake for storing large datasets, which can be processed and queried using services like Amazon Athena or AWS Glue.

5. **Application Asset Hosting**

Web apps use S3 to serve static resources like CSS, JS, JSON, and image files, reducing server load and improving scalability.

6. Disaster Recovery

S3's cross-region replication allows data to be backed up in multiple regions for disaster recovery and business continuity.

3.7 Key features of Amazon S3 and Pipeline

Amazon S3

Natural Storage and Retrieval

- **High-Performance Object Storage**

Amazon S3 provides scalable, high-performance object storage with virtually unlimited capacity. It allows storing any amount of data and is designed for 99.999999999% (11 9's) durability. This is achieved by automatically distributing data across multiple availability zones within AWS regions.

- **Data Durability and Availability**

Amazon S3 ensures that data is highly durable, and it is designed to survive the loss of two facilities without data loss. S3 automatically replicates data to different facilities within the same region, providing redundancy and minimizing the risk of data loss.

- **Multi-Region Replication**

S3 allows automatic replication of objects across different AWS regions using Cross-Region Replication (CRR). This ensures high availability and disaster recovery capabilities for your stored data globally.

- **Lifecycle Policies**

Amazon S3 provides lifecycle management capabilities to automatically transition data to different storage classes (e.g., Standard, Glacier) based on usage patterns. This helps optimize costs by automatically archiving or deleting data when no longer needed.

Builder Productivity

- **Easy File Management and Organization**

S3 allows developers to store, retrieve, and organize large amounts of unstructured data in an easy-to-manage way. Files can be organized into folders, and access permissions can be set at both the bucket and object levels.

- **One-Click Data Archiving and Retrieval**

Through integrations with AWS Glacier and Intelligent-Tiering, S3 supports automated archiving of older files while keeping them easily accessible when required. Developers can configure these settings without complex coding.

- **Versioning and Data Backup**

S3 versioning enables the storage of multiple versions of the same file, which is essential for tracking changes and backing up important files. It allows developers to roll back to previous versions of objects in case of accidental deletion or corruption.

- **Automatic File Encryption**

S3 integrates with AWS Key Management Service (KMS) and supports server-side encryption to ensure that data is stored securely, whether at rest or in transit.

Encryption can be enabled with just a few clicks, providing an additional layer of data security.

AWS Service Integrations

- **Integration with Amazon CloudFront**

Amazon S3 integrates seamlessly with Amazon CloudFront, AWS's Content Delivery Network (CDN), to serve static and dynamic content faster by caching and delivering data closer to end-users globally.

- **Integration with AWS Lambda**

S3 can trigger AWS Lambda functions to process data automatically when files are uploaded or modified. This enables serverless computing and automates workflows like image resizing, data transformation, or event handling.

- **Integration with Amazon Glacier**

S3's Glacier storage class offers a cost-effective solution for archiving long-term data. Integration with Glacier enables automatic or manual archiving of older, infrequently accessed data at a much lower cost.

- **Integration with Amazon Athena**

Amazon Athena is integrated with S3 for querying data directly from the storage without moving it. It simplifies analytics by providing an SQL interface to data stored in S3.

AWS Pipeline (CI/CD)

Automated Development and Deployment

- **Continuous Integration and Deployment (CI/CD)**

AWS Pipeline allows developers to automate the process of building, testing, and deploying applications, improving collaboration and speeding up delivery. It integrates with other AWS services to ensure that new code changes are automatically built, tested, and deployed.

- **Version Control and Code Repository Integration**

AWS Pipeline integrates with version control systems such as GitHub, GitLab, and AWS CodeCommit, allowing developers to trigger build and deployment actions based on code changes, tags, or branches.

- **Rollback and Versioning Support**

AWS Pipeline allows you to keep track of different versions of applications. In the event of deployment failure, rollback mechanisms make it easy to revert to a previous stable version, minimizing downtime and ensuring business continuity.

Builder Productivity

- **One-Click Deployment to Multiple Platforms**

AWS Pipeline makes it easy to deploy applications across multiple environments and platforms (e.g., EC2, Lambda, ECS, and on-premises servers) with minimal configuration. Developers can push code directly from a repository to the desired environment.

- **Automated Testing**

Pipelines automatically run tests as part of the CI/CD workflow to ensure code quality. Developers can define unit tests, integration tests, and other validation checks, which are triggered every time code changes are pushed to the repository.

- **Seamless Integration with Other AWS Services**

AWS Pipeline integrates natively with other AWS services such as AWS Lambda, EC2, Amazon S3, and AWS Elastic Beanstalk. This allows developers to build and deploy cloud-native applications that interact seamlessly with other AWS resources.

- **Easier Monitoring and Debugging**

Integration with AWS CloudWatch allows you to monitor the pipeline's activities and

automatically trigger notifications if something goes wrong. Logs and error messages are centralized, making troubleshooting and debugging easier for developers.

Cost-Efficiency and Scalability

- **Pay-As-You-Go Pricing**

AWS Pipeline follows a pay-as-you-go pricing model, which means that you only pay for the resources you use. There are no upfront costs or long-term commitments, making it an affordable solution for businesses of all sizes.

- **Scalable Automation**

AWS Pipeline automatically scales with your project requirements. Whether you are deploying a small web app or a complex enterprise system, the pipeline adapts to handle large numbers of deployments and infrastructure changes efficiently.

3.8 Amazon S3 - Use Case

1. Backup and Recovery

- **Data Backup:** Amazon S3 is widely used for backing up important data, including files, applications, and system configurations. The high durability and availability of S3 ensure that data is safely stored and can be easily retrieved during recovery.
- **Disaster Recovery:** S3's cross-region replication capabilities make it an ideal choice for disaster recovery. Organizations store critical backup data in multiple AWS regions, ensuring data availability in case of a regional failure.

2. Static Website Hosting

- **Hosting Static Content:** S3 is an excellent solution for hosting static websites, including HTML, CSS, JavaScript, images, and videos. By enabling S3 static website hosting, users can serve static assets with low latency and at high availability.
- **Cost-Effective Web Hosting:** Hosting a static website on S3 is cost-effective as users only pay for the storage and data transfer without worrying about server management.

3. Media Storage and Streaming

- **Media Files Storage:** S3 is often used for storing large media files, such as videos, images, and audio. Content delivery can be optimized through integration with Amazon CloudFront (CDN).

- **Video On Demand (VOD):** S3 enables storing high-quality video files, which can be streamed to users via CloudFront or other media delivery tools, making it ideal for building scalable streaming applications.

4. Big Data Storage and Analytics

- **Data Lake:** Organizations use S3 to create data lakes where they can store large datasets in their raw format. Tools like Amazon Athena, Amazon Redshift Spectrum, and AWS Glue can then be used to analyze and transform this data.
- **Big Data Processing:** S3 integrates seamlessly with big data frameworks like Apache Hadoop, Apache Spark, and AWS EMR (Elastic MapReduce), enabling scalable and distributed processing of large datasets.

5. Archiving and Long-Term Storage

- **Cold Storage:** With S3 Glacier and Glacier Deep Archive, businesses can archive large amounts of data that are rarely accessed but need to be retained for compliance or historical purposes. This provides a low-cost storage solution for long-term data retention.
- **Compliance and Legal Storage:** S3 is used for storing data that needs to be kept for extended periods for regulatory or legal reasons, including audit logs, contracts, and government records.

6. Application Hosting and File Sharing

- **Storing and Sharing Application Data:** S3 can be used as a storage backend for applications, where user data such as profiles, images, and documents can be uploaded, stored, and shared securely.
- **Content Delivery for Apps:** Many mobile and web applications use S3 to store assets like user-generated content, images, and media files, which are dynamically served to users across the globe.

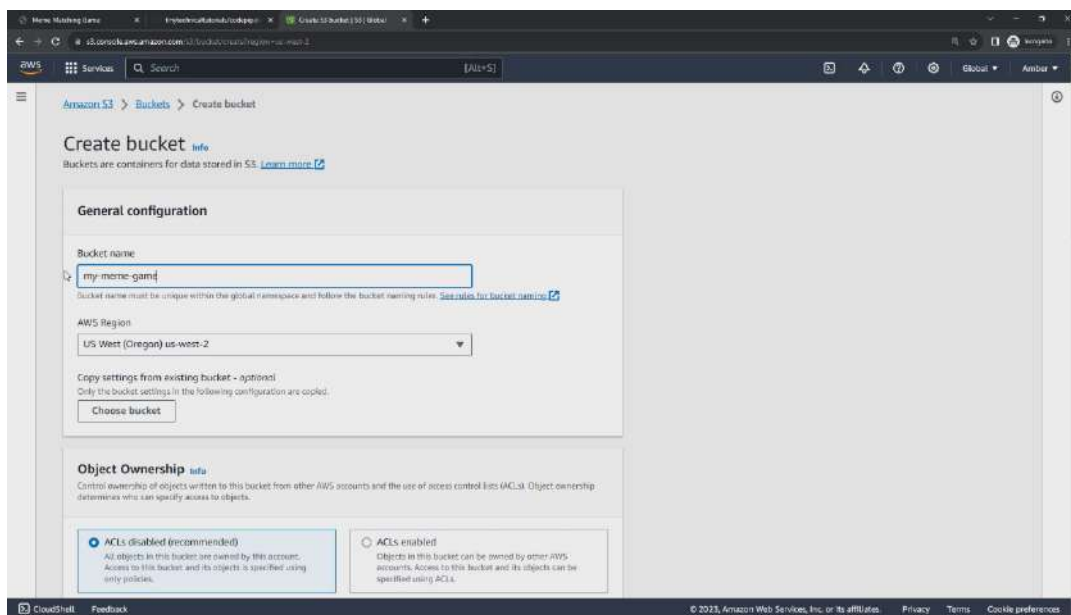
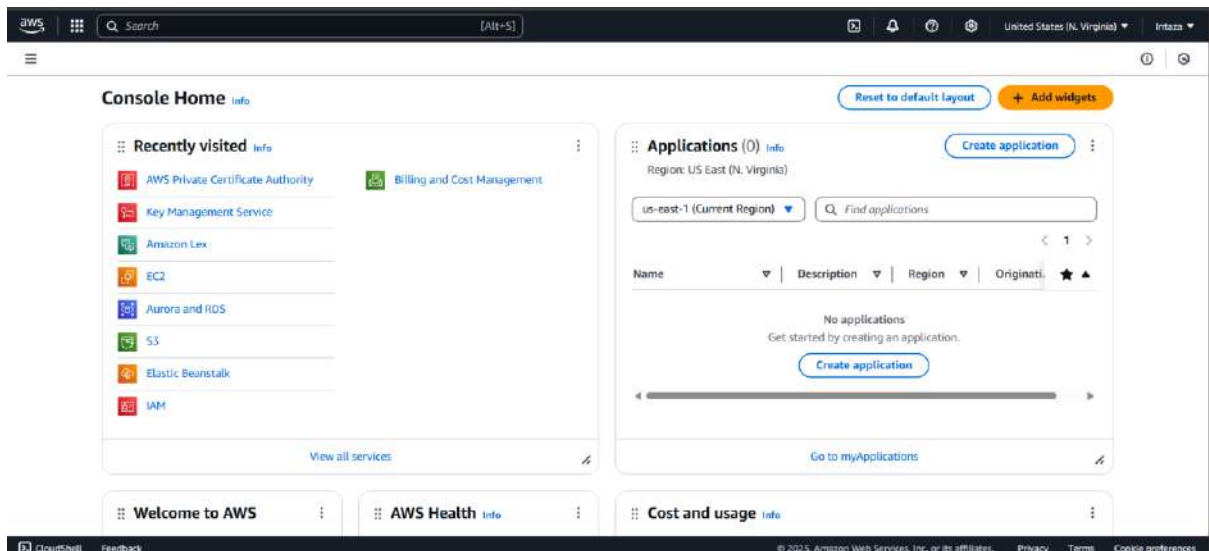
CHAPTER 4

CREATING MEME MATCHING GAME

4.1 Steps of creating a meme game

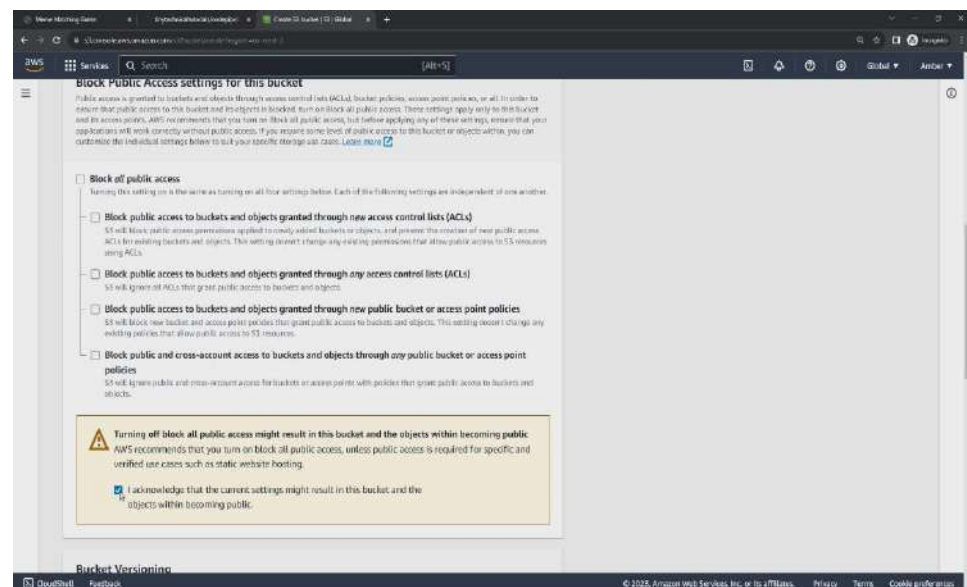
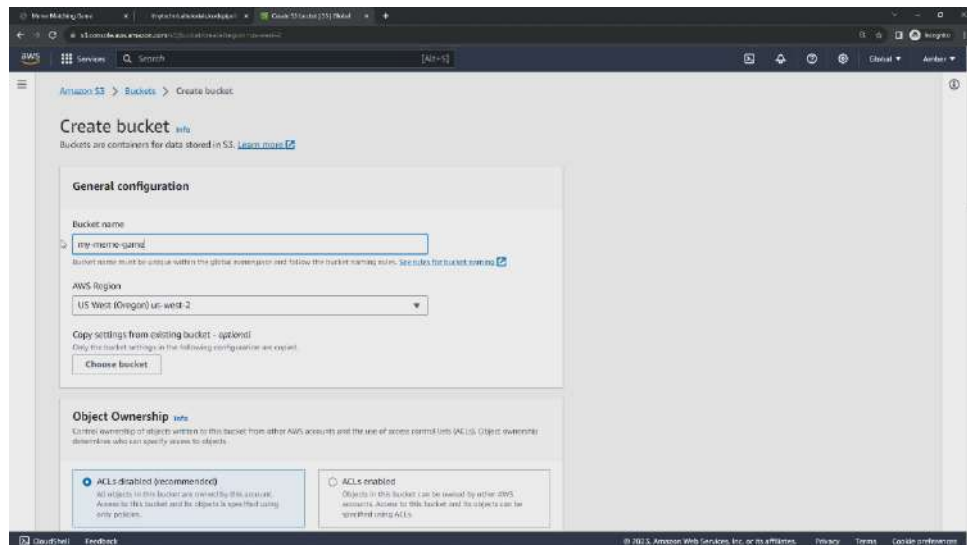
Step 1:

Open AWS Management Console and go to S3

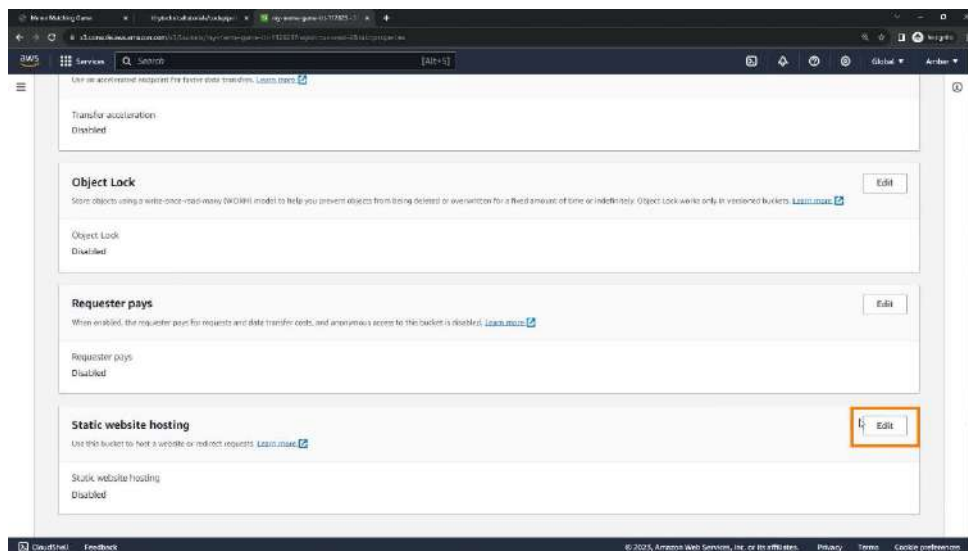
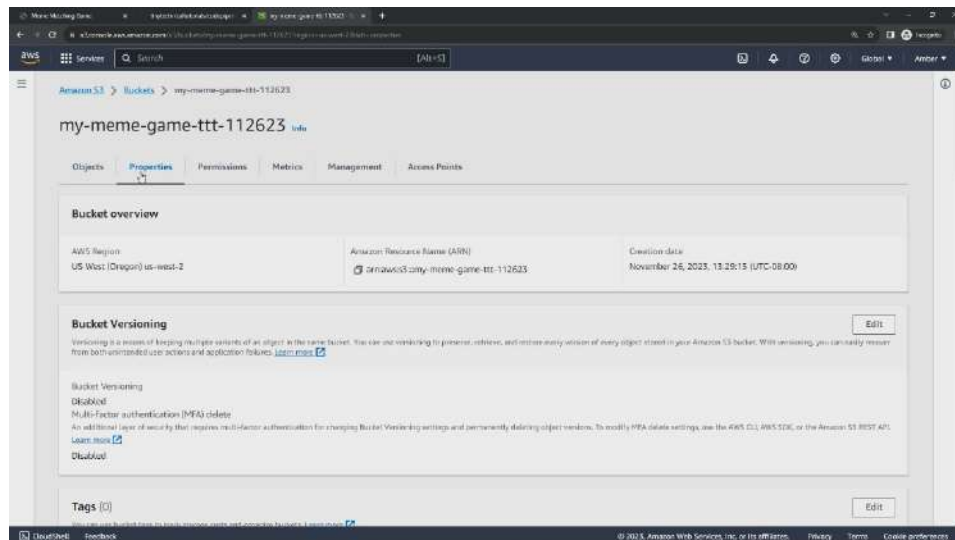


Step 2:

Here we have created a Bucket named my-meme-game

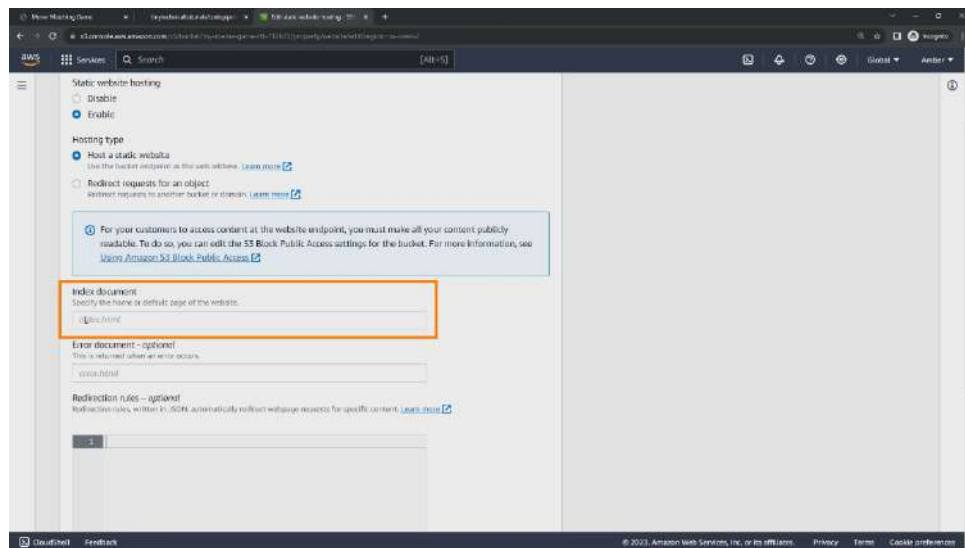


Step 3: after creating a bucket we have to enable the hosting from the properties section



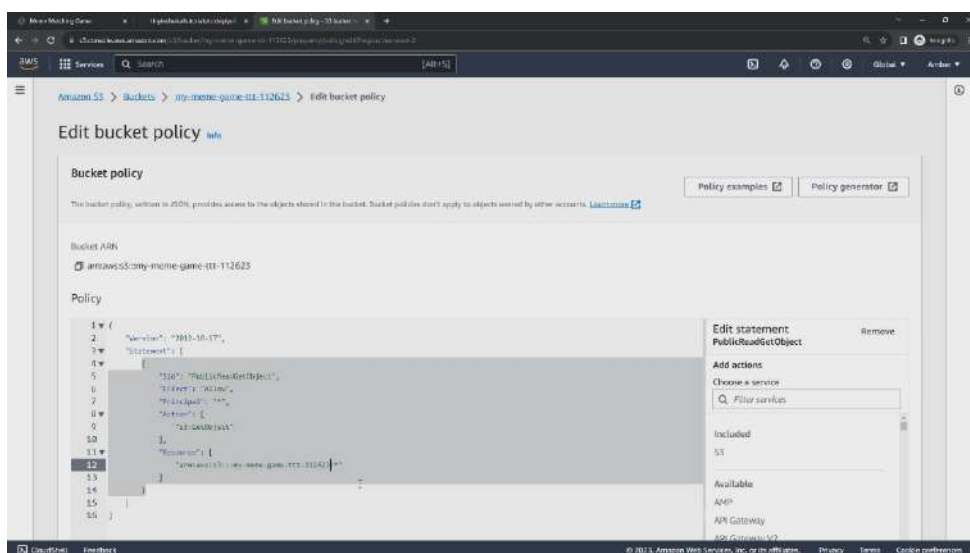
Step 4:

Now we have to edit in index.html



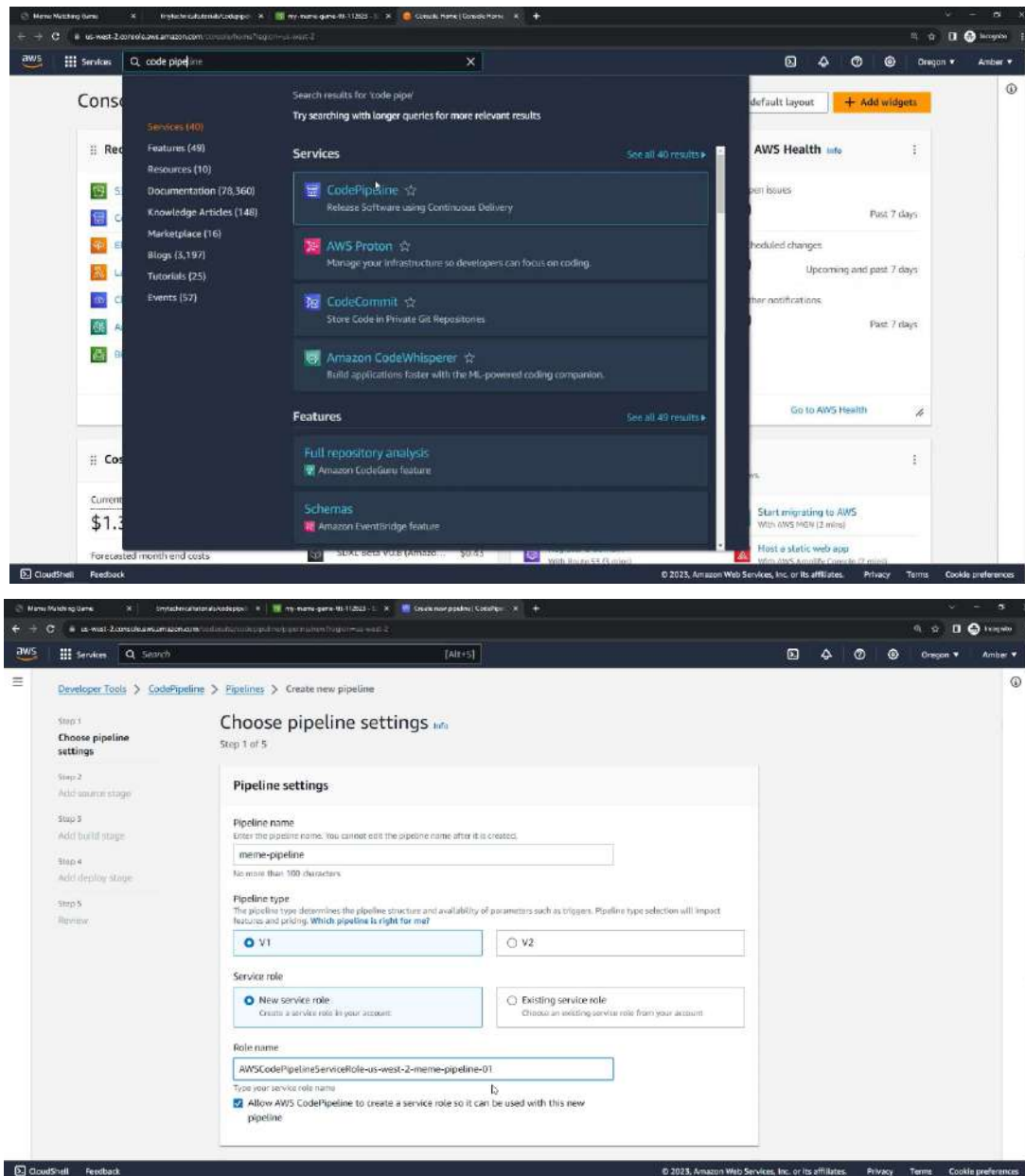
Step 5:

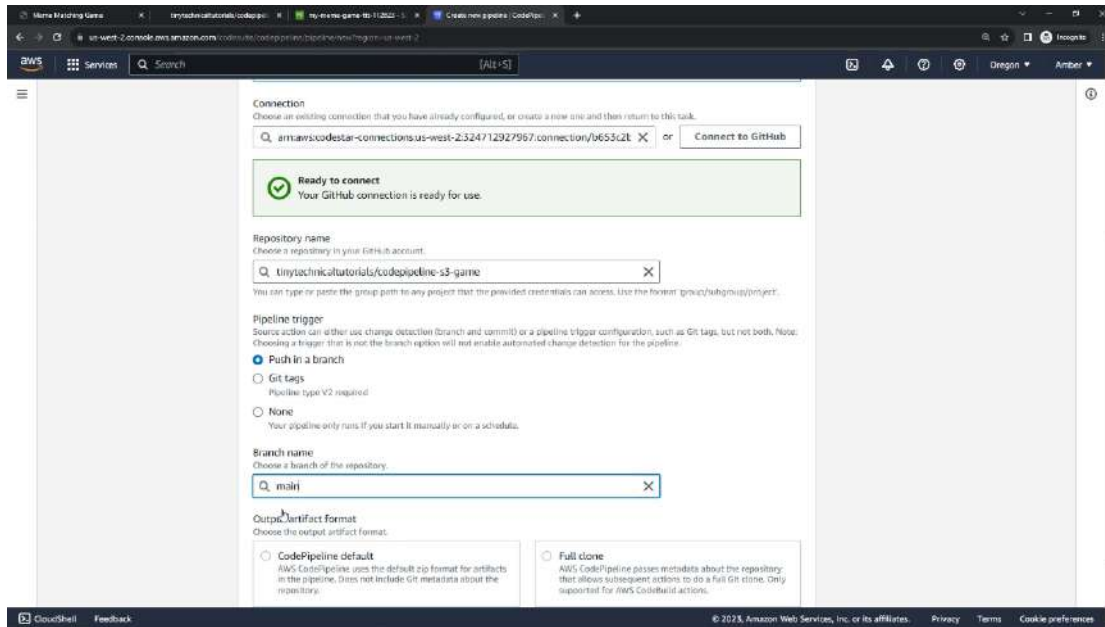
Now we have to go to properties section and there we have to edit the bucket policy for the link generation after that we have to make a change in the code changing the name to meme game pipeline in the code and save the change



Step 6:

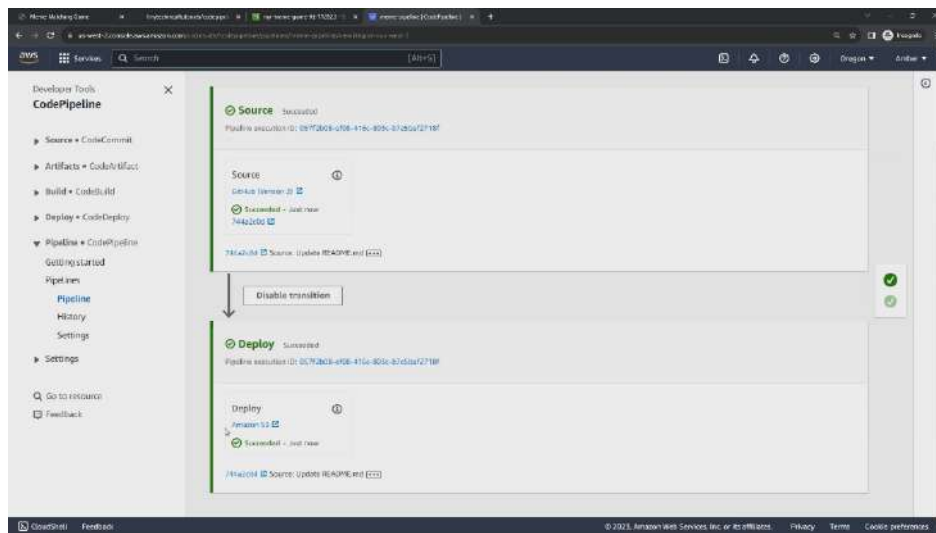
Now create a pipeline and connect it to your github to host the repository that is in github.





Step 7:

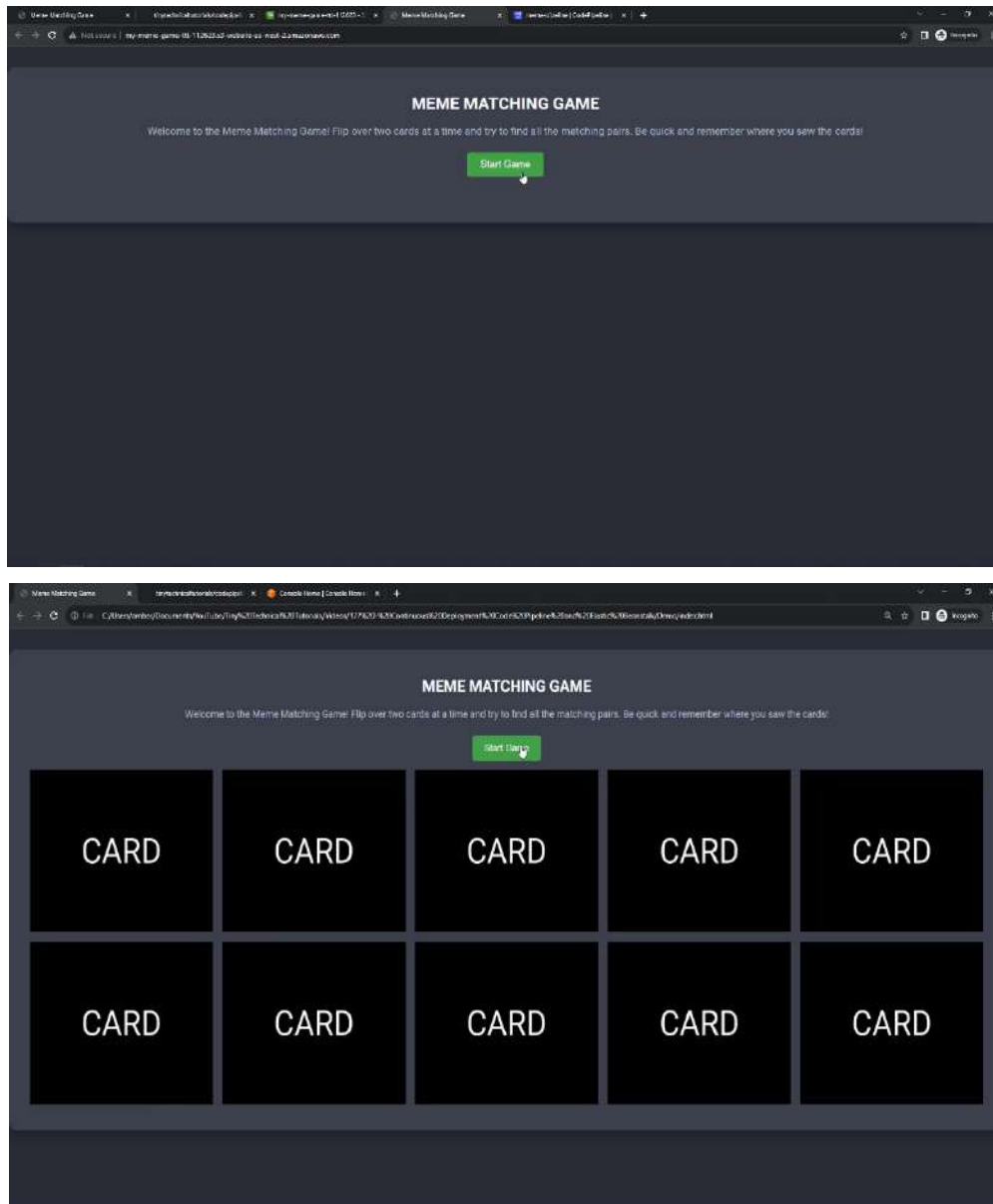
Now to deploy it let it connect from the github then open it and its done



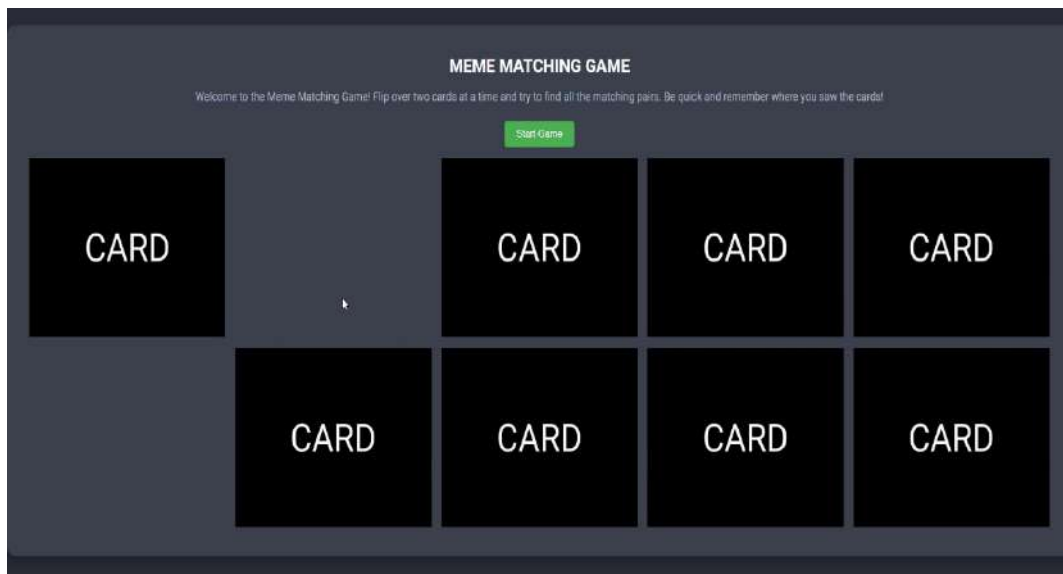
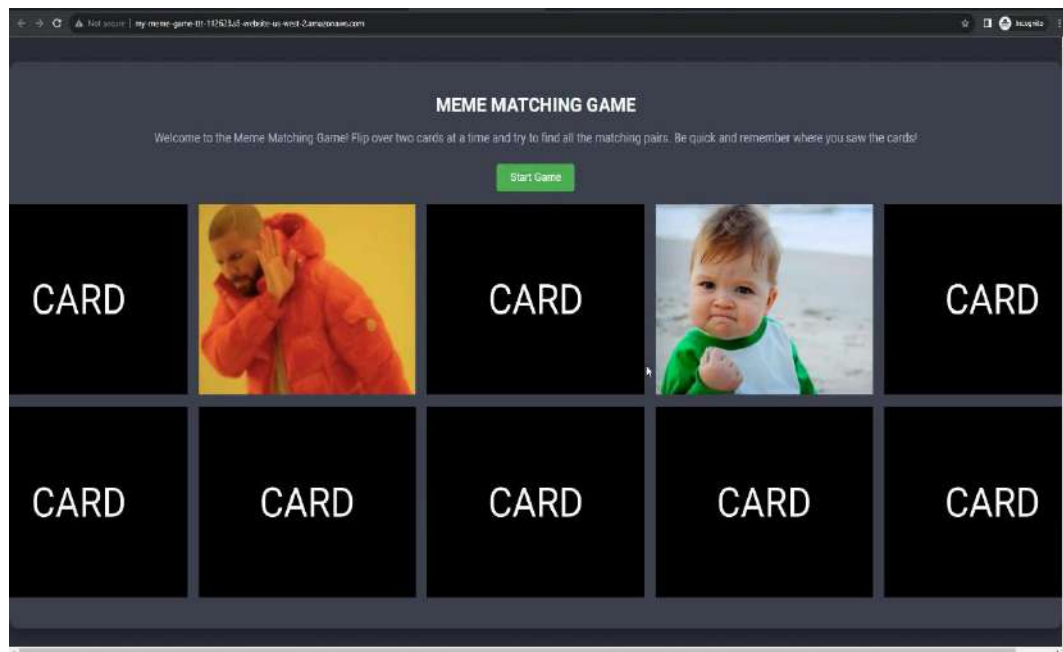
CHAPTER 5

SNAPSHOTS

5. 1 Starting phase of the game



5.2 Flipping of cards



CHAPTER 6

CONCLUSION

The meme matching game is a fun, interactive, and engaging application that incorporates cloud storage and serverless technology to enhance user experience. By utilizing AWS services like S3 and pipeline, the game efficiently handles data storage and deployment, ensuring smooth performance and scalability. With each session, players are challenged to match memes, creating an enjoyable and dynamic experience. The automatic shuffling of memes after a win ensures that the game remains fresh and engaging, keeping players coming back. The cloud-based architecture ensures that the game remains scalable, secure, and accessible across different devices. Overall, the project successfully demonstrates how cloud services and a well-designed user interface can come together to create an entertaining and responsive web-based application.

REFERENCES

1. <https://aws.amazon.com>
2. <https://docs.aws.amazon.com>
3. https://www.researchgate.net/publication/363788800_Computer_Vision_and_Internet_Meme_Genealogy_An_Evaluation_of_Image_Feature_Matching_as_a_Technique_f_or_Pattern_Detection
4. <https://peer.asee.org/the-meme-game-a-hands-on-activity-to-introduce-first-year-engineers-to-concepts-in-mathematical-computational-modeling.pdf>