

CSCI 5411

Advance Cloud Architecting

Fall 2024

Assignment 1

**Banner ID:** B00984406

**Name:** Jems Shaileshkumar Patel

# 1. Aim of the Task

The aim of this project is to develop a user-friendly photo gallery application named **DalDrive24**. This web application allows users to upload, view, and download their images seamlessly. It provides a centralized platform for managing personal images, enabling users to capture and cherish their memories effectively. The application interacts with an AWS Lambda function and S3 for handling image uploads and retrievals, ensuring a scalable and efficient backend. It also uses a AWS Dynamo DB for storing the user information as well as photos.

## Key Features:

* User Authentication
* Image Upload Functionality using AWS S3.
* Store all the details in the DynamoDB.
* Display of the uploaded image for the user.
* Download Option for each Displayed Image.

This application offers a seamless authentication experience, allowing users to sign up and log in effortlessly. User details are securely stored in DynamoDB, ensuring efficient data management. As a serverless application, it leverages AWS Lambda for backend processing, enabling scalability and reduced infrastructure management.

The application utilizes Amazon S3 for image storage. Upon logging in, users have the option to upload images, which triggers a Lambda function responsible for uploading the images to S3 and storing the corresponding links in DynamoDB. This process enhances efficiency and organization of user data.

Once users access the dashboard, they can view all the images they have uploaded. Additionally, there is a convenient option for downloading images, allowing users to easily retrieve their cherished memories.

# 2. Thought Process of Developing the Application

The development of this application involved several key phases:

**a. Planning:**

* **Requirement Analysis:** Identifying the core functionalities that the application should provide, including user authentication, image upload, and retrieval.
* **Technology Stack:** Deciding to use React for the frontend, AWS S3 for image storage, and AWS Lambda for backend processing. This combination allows for a serverless architecture, minimizing infrastructure management.

**b. Design Considerations:**

* **User Interface**: The UI was designed to be simple and intuitive. The navigation bar allows easy access to upload and manage images. A welcoming homepage sets the tone for the user experience.
* **Responsiveness**: Utilizing Tailwind CSS ensured that the application is responsive and accessible across different devices and screen sizes.

**c. Implementation:**

* **Frontend Development**: React was used to create a component-based architecture. Key components include `Navbar`, `Dashboard`, and `HomePage`, each focusing on specific functionalities.
* **Backend Integration**: AWS Lambda functions were developed to handle image uploads and retrievals. The API Gateway facilitates secure communication between the frontend and the backend.

**d. Challenges Faced:**

* **CORS Issues**: Encountered Cross-Origin Resource Sharing (CORS) issues when fetching images from AWS S3. This was resolved by configuring CORS settings in the S3 bucket.

# 3. Deployed URL of the Application

Deployed Link: <https://imaginative-madeleine-446957.netlify.app/>

# 4. Output Screenshots

A screenshot of a computer

Description automatically generated

Figure 1Home page of the DalDrive24

A screenshot of a login form

Description automatically generated

Figure 2 Sign Up Page

A screenshot of a computer

Description automatically generated

Figure 3 Login Page

A computer screen shot of a computer

Description automatically generated

Figure 4 Uploading an Image

A screenshot of a computer

Description automatically generated

Figure 5 Dashboard of the DalDrive24

A screenshot of a person

Description automatically generated

Figure 6 Option for the Downloading

# 5. References

1. Amazon Web Services, “What is AWS Lambda?” [Online]. Available: https://aws.amazon.com/lambda/. [Accessed: Date].

2. React Documentation, “Getting Started with React,” [Online]. Available: https://reactjs.org/docs/getting-started.html. [Accessed: Date].

3. Tailwind CSS, “Installation,” [Online]. Available: https://tailwindcss.com/docs/installation. [Accessed: Date].