Cloud Computing Final Project - [Group Name]

Project Title

[Project Title]

Group Members

- [Student 1 Name]
- [Student 2 Name]
- [Student 3 Name]
- [Student 4 Name]
- [Student 5 Name]

Project Overview

This repository contains the final project for **CS1660/2060**. The objective is to design and deploy a cloud-based application using **Google Cloud Platform (GCP)**, integrating at least **five distinct GCP services** such as **Cloud Storage**, **Compute Engine**, **Cloud Run**, **BigQuery**, and **Artifact Registry**. You are encouraged to explore services beyond those discussed in class, using any programming language or framework.

Each team will select a project idea, build the application, and deploy it on GCP.

Grading Rubric

1. Integration of GCP Services (10 points)

- Use at least **five different GCP services** effectively.
- Points awarded based on the relevance and creativity of service integration.

2. Authentication Layer (5 points)

- Implement a secure login system with persistent storage.
- Users should be able to log in, modify their data, and retrieve the same data on subsequent logins.

3. Deployment Automation (5 points)

- Automate deployment using **GitHub Actions** or equivalent CI/CD platforms.
- Automation may include deploying containers to **Artifact Registry** or deploying the entire stack automatically.

4. Documentation, Recorded Demo, and Architecture Diagram (10 points)

- Provide a **recorded demo** (maximum 5 minutes) showcasing the application’s functionality.
- Clearly explain the problem being solved and the solution approach.
- Include an **architecture diagram** in this README.
- Provide comprehensive documentation covering setup, usage, and technical details.

Suggested Project Ideas

1. QR Code-Based Class Attendance System

- Create an attendance tracking system using QR codes.
- Use GCP services for scalability and real-time tracking.
- Store attendance records securely using **Cloud Storage** or **Firestore**.

2. Image Processing Web Service

- Develop a platform for users to upload and process images (e.g., grayscale conversion).
- Use:
  - **Cloud Storage** for image uploads.
  - **Cloud Functions** for processing images.
  - Provide URLs for downloading processed images.

3. Innovative Language Model (LLM) Application

- Integrate a **Language Model (LLM)** for text generation, smart responses, or content summarization.
- Utilize GCP's AI/ML services for language model integration.

Submission Requirements

- Submit all code via **GitHub repositories**.
- Each repository must include:
  - This **README** with an architecture diagram and all supporting documentation.
  - Detailed documentation covering the project's setup, execution, and technical overview.
- Submit a **recorded demo** (5-minute maximum) showcasing and explaining the project's functionality.
- Official submission must be done through **Canvas** with links to the GitHub repositories and all supporting materials.