Final Document for

**AWS-Powered Image Classification**

**R Group Consultancy Ltd.**

**Prepared by:** Rajeswari Tadepalli

**Version:** RT 1.1

**Document Id:** RT-2024-SP-002

**Date:** 2024 December

**TABLE OF CONTENTS**

[REVISIONS AND DISTRIBUTION](#_TOC_250045) 5

[PROJECT AUTHORIZATIONS 6](#_TOC_250044)

[PROJECT SPONSOR APPROVAL OF SCOPE STATEMENT 7](#_TOC_250043)

1. [PROJECT CHARTER 8](#_TOC_250042)
   1. [PROJECT AUTHORIZATION 8](#_TOC_250041)
   2. [EXECUTIVE SUMMARY 8](#_TOC_250040)
2. [BUSINESS OBJECTIVES AND EXPECTED BENEFITS 9](#_TOC_250039)
   1. [BUSINESS NEEDS, OPPORTUNITIES, OBJECTIVES 9](#_TOC_250038)
      1. [BUSINESS NEED: 9](#_TOC_250037)
      2. [BUSINESS OBJECTIVES: 10](#_TOC_250036)
      3. [MAGNITUDE OF THE NEED/OPPORTUNITY: 11](#_TOC_250035)
      4. [CONSEQUENCES OF NOT ADDRESSING THE NEED/OPPORTUNITY: 9](#_TOC_250034)
      5. [BUSINESS VISION: 9](#_TOC_250033)
   2. [PROPOSED SOLUTION 13](#_TOC_250032)

[2.1.0 SOLUTION DESCRIPTION: 13](#_TOC_250031)

* 1. PROJECT DELIVERABLES 17
  2. [Anticipated Benefits 19](#_TOC_250030)

1. PROJECT DESCRIPTION 22
   1. Functionality Inclusions AWS Powered Image Classification 24
   2. Functionality Exclusion 25
   3. [Completion and Acceptance Criteria 26](#_TOC_250029)
   4. [RISK ASSESSMENT 27](#_TOC_250028)
   5. [CONSTRAINTS 28](#_TOC_250027)
   6. [DEPENDENCY LINKAGES 30](#_TOC_250026)
   7. [IMPACTS 31](#_TOC_250025)
   8. [MEASURES OF PROJECT SUCCESS 33](#_TOC_250024)
   9. [ASSUMPTIONS 36](#_TOC_250023)
   10. [CRITICAL SUCCESS FACTORS 38](#_TOC_250022)
2. [FUNCTIONAL REQUIREMENTS 4](#_TOC_250021)1
   1. [BUSINESS FUNCTIONAL REQUIREMENTS BY GROUP 4](#_TOC_250020)1
   2. [System Functional Requirements 43](#_TOC_250019)
3. [PROJECT APPROACH 45](#_TOC_250018)
   1. [Planned Approach 45](#_TOC_250017)
4. [PROJECT COST AND RESOURCE ESTIMATES 48](#_TOC_250016)
   1. [MAJOR MILESTONES - ESTIMATED SCHEDULE 48](#_TOC_250015)
   2. [RESOURCE REQUIREMENTS – TEAM AND SUPPORT RESOURCES 49](#_TOC_250014)
   3. [ESTIMATED COSTS 5](#_TOC_250013)0
5. [PROJECT CONTROLS 53](#_TOC_250012)
   1. [CHECKPOINT FUNDING SCHEDULE 53](#_TOC_250011)
   2. [WEEKLY/MONTHLY STATUS AND STEERING COMMITTEE MEETINGS 53](#_TOC_250010)
   3. [WEEKLY/MONTHLY STATUS REPORTS 53](#_TOC_250009)
   4. [RISK MANAGEMENT 53](#_TOC_250008)
   5. [ISSUE MANAGEMENT 55](#_TOC_250007)
   6. [CHANGE MANAGEMENT 56](#_TOC_250006)
   7. [COMMUNICATION MANAGEMENT 56](#_TOC_250005)
6. [ADDITIONAL PROJECT DETAILS 58](#_TOC_250004)
7. [PROJECT DICTIONARY 60](#_TOC_250003)
8. [PROJECT ROLES AND STAKEHOLDERS 6](#_TOC_250002)1
   1. [PROJECT ROLES 6](#_TOC_250001)1
   2. [PROJECT STAKEHOLDERS 63](#_TOC_250000)
9. Project Plan / Gantt Chart 64

11.1 Project Plan 64

11.2 Gantt Chart 65

12. Data Flow Diagram with Use Cases 66

12.1 Data Flow Diagram 66

12.1.1 Context Diagram 66

12.1.2 DFD Level 0 Diagram 66

12.1.3 DFD Level 1 Diagram 67

12.2 User Stories 67

13. Network Diagram and Security Measures 69

13.1 Network Diagram 69

13.2 Security Measures 71

14. ER Diagram 72

15. Screenshots 74

16. Budget 81

### REVISIONS AND DISTRIBUTION

|  |  |  |
| --- | --- | --- |
| **REVISION ID** | **REVISION DATE** | **REVISED BY** |
| RT1.1 | December 7th, 2024 | Rajeswari Tadepalli |
|  |  |  |
|  |  |  |

The Project Scope Management Plan from time to time may require updates. ALL amendments to this plan shall be informed to the change control board by use of the change request form and approved by the project change control board prior to distribution. Only revised parts of the plan was distributed along with the approval and shall be accompanied by instructions on how to implement the changes.

The initial page numbering system (to be added upon initial approval) was a normal continuous numbering displayed in the lower right corner of each page. In the event that pages have to be added, characters shall be added to the number. In case entire pages are deleted, the corresponding page shall be replaced by a blank page stating, “page removed”.

Each added/changed page shall have the revision number and date of approval displayed on the bottom of the page.

### 

### PROJECT AUTHORIZATIONS

The Scope Statement was approved by:

* **Project Manager:** Rajeswari Tadepalli
* **Project Sponsor:** Prof. Mark Anthony Green

Project Changes was approved by:

* **Project Manager:** Rajeswari Tadepalli
* **Project Sponsor:** Prof. Mark Anthony Green

Project deliverables was approved/accepted by:

* **Project Manager:** Rajeswari Tadepalli
* **Project Sponsor:** Prof. Mark Anthony Green
* **Key Stakeholders:** Rajeswari Tadepalli

Specific task responsibilities of project resources was defined in the Project/Work Plan.

### 

### PROJECT SPONSOR APPROVAL OF SCOPE STATEMENT

The purpose of this document is to provide a vehicle for documenting the planning efforts for the project. It is used to reach a satisfactory level of mutual agreement between the project manager, project sponsors, and all impacted users on the objectives and scope of the project before significant resources are committed and expenses incurred.

Signatures below indicate that the information contained in this document has been reviewed and agreed upon.

|  |  |  |  |
| --- | --- | --- | --- |
| **REVISION**  **ID** | **REVIEWER NAME** | **REVIEWER ROLE** | **APPROVED DATE** |
| RT-001 | Prof. Mark Green | Project Sponsor | Sep 31, 2023 |
|  |  |  |  |
|  |  |  |  |

### PROJECT CHARTER

### PROJECT AUTHORIZATION

The AWS Powered Image Classification has received authorization and commitmentfrom Prof.Mark Anthony Green, the project sponsor, marking the official commencement of the project.

### EXECUTIVE SUMMARY

The goal of this project is to create an image classification system by utilizing AWS's advanced machine learning capabilities and scalable infrastructure. Robots can now read and evaluate visual data thanks to computer vision, which depends on picture classification in the digital age. Developing an adaptable application that scales dynamically in response to demand and guaranteeing cost-effectiveness by utilizing AWS cloud resources are the main goals. The project builds models for efficient visual data analysis using SageMaker and Amazon Rekognition services. The end product is a client-accessible RESTful web service that offers a reliable method for classifying images across a range of industries.

# **BUSINESS OBJECTIVES AND EXPECTED BENEFITS**

* The Image Classification System project uses AWS's scalable infrastructure and machine learning capabilities to meet the urgent need for enhanced visual data processing. The program intends to address the accessibility gap in visual data interpretation and analysis tools for developers and enterprises. The project aims to establish the firm as a leader in the computer vision sector by emphasizing scalability, cost-effectiveness, and client accessibility while providing an easy-to-use picture classification system via a RESTful web service.
* The difficulty of classifying images and the dearth of readily available information for companies looking to use this technology are the issues that have been discovered. The project's goals include using AWS technologies like SageMaker and Amazon Rekognition to create a flexible application that scales dynamically based on demand. As a result, computer vision applications are expected to advance and there was a competitive edge in the market, enhanced operational efficiency, and cost savings through resource optimization.

## **BUSINESS NEED, OPPORTUNITIES, OBJECTIVES**

### BUSINESS NEED:

### The necessity for the AWS-Powered Image Classification project in the current digital environment stems from the critical function that image classification plays in computer vision, especially when it comes to the interpretation and evaluation of large amounts of visual data. With industries depending more and more on automation and data-driven insights, a reliable picture categorization system is becoming more and more important. The need for a sophisticated and scalable solution is highlighted by the difficulty of creating models that can recognize and analyze visual data.

### This project uses AWS's superior machine learning capabilities and scalable infrastructure to solve the Business Need. The need arises from the incapacity of current technologies to effectively classify images into different groups for uses like robotic interpretation. By investigating AWS technologies like Amazon Rekognition and SageMaker, which provide a complete picture classification solution, the goal is to close this gap. The goal of the business need is to give enterprises an adaptable application that uses AWS's Infrastructure as a Service (IaaS) paradigm to grow dynamically on demand while maintaining cost effectiveness. The project finally satisfies the growing needs of sectors dependent on computer vision technology by providing a sophisticated picture categorization system that is available via a RESTful web API.

### BUSINESS OBJECTIVES:

* + - 1. **Flexibility and Scalability:** Create a very flexible picture classification system that can grow and shrink on demand. This was enable the company to effectively manage a range of workloads, guaranteeing the best use of its resources and adaptability to shifting demands.
      2. **Cost-Effectiveness:** Continue to be cost-effective by utilizing Amazon Web Services' (AWS) cloud capabilities. This entails minimizing wasteful spending, matching costs to real demand, and optimizing the use of computation, storage, and communications resources.
      3. **Client Accessibility:** Construct a client-facing application that can be accessed via a RESTful web service. This improves the user experience and makes it simple to incorporate the image classification system into the apps and processes of customers.

### MAGNITUDE OF THE NEED/OPPORTUNITY:

### The project AWS-Powered Image Classification offers significant commercial prospects in the context of the growing market for sophisticated computer vision solutions. The chance is in tackling the pressing requirement of effectively interpreting and analyzing visual data in the digital age. Businesses are realizing more and more how revolutionary picture classification may be, particularly in the context of robots and automated systems.

### This project is positioned to benefit from the expanding market for advanced image categorization systems. Offering a full solution is made possible by utilizing AWS's scalable infrastructure and state-of-the-art machine learning capabilities. In line with the current trend of scalable and affordable cloud solutions, the business opportunity extends to enterprises looking for a flexible application that scales dynamically based on demand.

### Using AWS services such as Amazon Rekognition and SageMaker, the project opens opportunities for innovation and differentiation. The business opportunity is to provide customers with a state-of-the-art solution on top of the image classification system, accessible via a RESTful web service. This places the project at the forefront of technological development in computer vision and provides a competitive advantage and potential market leadership.

### The project and success in realizing these business opportunities can lead to increased market share, industry recognition and the organization becoming a leading provider of image classification solutions, contributing to rapid growth and success. the emerging field of computer vision.

## **PROPOSED SOLUTION**

### SOLUTION DESCRIPTION:

* The proposed solution for AWS-Based Image Classification project's uses essential AWS services and technologies in a two-tier design with a Web-tier and an App-tier. Through the Flask framework, front-end and back-end communication is made easier with the help of an aesthetically pleasing static HTML and CSS web page on the Web-tier. The Web-tier and App-tier EC2 instances can seamlessly communicate with each other thanks to the integration of Boto3 libraries.
* A pre-trained ResNet model utilizing ImageNet labels from the Keras library is used in the App-tier instances on EC2, which are set up with an AMI of t2.small instance type for maximum cost-effectiveness. Installing the required libraries allows image classification tasks to be supported. The Web-tier must be manually started, the controller must be launched to manage SQS queue information and instance initialization, and an upload button must be used to submit dynamic images.
* Scalability is guaranteed since the number of instances that are spawned upon image submission adjusts to the number of input images. Classification results are sent via SQS messaging to the output S3 bucket when images are uploaded to the designated input S3 bucket. The output is shown as image names and labels that come from both SQS and S3 on the front-end webpage in a key-value pair format. Using AWS resources to their fullest potential, this all-inclusive solution offers users a scalable, affordable, and intuitive image classification process.
* Scalability is ensured during image submission as the number of instances produced adjusts based on the number of input images. Images are uploaded to the designated input S3 bucket, and via SQS communications, the categorization results are sent to the output S3 bucket. Outcomes are presented in a key-value pair format on the front-end homepage, displaying image names and labels that come from SQS and S3. This complete solution offers consumers an affordable, scalable, and user-friendly experience by making the best use of AWS resources for accurate image classification.

### Benefits of the Project:

### User- Friendly Interface: The Flask framework, in conjunction with HTML and CSS to create an aesthetically beautiful static web page, offers consumers an easy-to-use interface. This promotes broader adoption and improves the user experience overall.

### Integration with AWS Services: Using Simple Queue Service (SQS) for messaging and Amazon S3 for storage guarantees effective data transport and component coordination. By optimizing the utilization of AWS services, this integration raises the project's total functionality.

### Seamless Data Flow: Communication between the Web-tier and App-tier instances is made easier by integration with Boto3 libraries and SQS messaging. This guarantees smooth data transfer throughout the submission, categorization, and delivery of results of images.

### Advanced Image Classification: ImageNet labels applied to a pre-trained ResNet model improve the precision and performance of picture classification tasks. As a result, the classification results are strong and trustworthy, satisfying the needs of the businesses that depend on computer vision applications.

### Cost-Effectiveness: Cost-effectiveness is enhanced by using t2.small instances and utilizing AWS services. The project makes an economically viable solution for enterprises by maximizing resource use and guaranteeing that charges correspond with actual demand.

### Scalability: Smooth scaling is made possible by the project's two-tier architecture and dynamic instance initiation. With its ability to adjust to changing workloads, the system effectively manages rising demand for picture classification without sacrificing efficiency.

**Criteria for Success:** The success of the project was measured against key criteria identified by key stakeholders:

1. **Accuracy of Image Classification:** The project's accuracy in terms of picture categorization findings was used to gauge its success. Based on the input photos, the system ought to classify images reliably and accurately.
2. **User Satisfaction:** Analyze the web interface's usability and intuitiveness to determine how satisfied users are. Good user reviews and a smooth submission procedure help make the project successful.
3. **Integration with AWS Services:** Examine the smooth interaction with AWS services, such as communications via SQS and storage via Amazon S3. Effective data transport and communication between system components are guaranteed by successful integration.
4. **Comprehensive Results Display:** Determine whether the front-end webpage's result display is successful. Users should have a clear and comprehensive view of categorization results thanks to the key-value pair format that displays image names and labels from both SQS and S3.
5. **Reliability of System Initiation:** Analyze the system beginning process's dependability. The controller should automatically acquire SQS queue information and launch new instances as needed, however the Web-tier should be started manually.

In summary, the goal of the AWS-Based Image Classification project is to provide an efficient solution for visual data analysis that is scalable, affordable, and easy to use. With the help of AWS services, the project incorporates a ResNet model that has already been trained, guaranteeing precise image categorization, easy resource management, and the possibility of becoming a market leader in the rapidly evolving field of computer vision.

* 1. **PROJECT DELIVERABLES**

### Deliverables Included:

1. **Static Web Page Design:** An attractive static HTML and CSS web page that functions as the user interface for uploading images and displaying results.
2. **Web-tier Communication Framework:** Front-end and back-end communication implementation made possible by the Flask framework to guarantee a flawless user experience.
3. **Integration with Boto3 Libraries:** The Web-tier and App-tier EC2 instances may communicate easily thanks to the integration of Boto3 libraries, which facilitates effective data transfer.
4. **App-tier Instances on EC2:** deployment of App-tier instances on EC2 with an AMI of t2, set up with an ImageNet label-based pre-trained ResNet model. Tiny instance type to minimize costs.
5. **Image Submission Feature:** The web interface has an upload button that facilitates the implementation of an image submission function that lets users submit photographs for classification.
6. **Input and Output S3 Buckets:** Configuration of designated S3 buckets for input and output to facilitate the effective storage and transfer of images and categorization outcomes.
7. Result Display on Webpage: Classification results are shown in a key-value pair format on the front-end homepage, with image names and labels taken from SQS and S3.

### Deliverables Excluded:

* + 1. **Advanced Front-end Features**: sophisticated front-end elements like dynamic visualizations and image previews are left out. The project's goal is simplicity, with an emphasis on the features that are necessary for submitting images and displaying results.
    2. **Automated Deployment Tools:** excluding frameworks or tools for automated deployment. The project uses a controller script for dynamic resource management and assumes manual Web-tier initiation.
    3. **User Authentication and Authorization:** Features for user identification and permission are left out to concentrate the project's efforts on its essential functions. This project does not encompass advanced security mechanisms for user access.

The project maintains clearly defined included and excluded deliverables, allowing for the recognition of areas outside the immediate objectives while keeping focus on critical functionality.

### ANTICIPATED BENEFITS

### To justify resource allocation and match efforts with strategic goals, project managers must anticipate benefits. More operational efficiency, cost savings through AWS utilization, and market leadership through creative solutions are all anticipated benefits in the context of the AWS-Based Image Classification project. These benefits all support long-term business stability and growth. When these advantages are realized, the company is positioned as a progressive leader in the rapidly changing field of computer vision technology and can be certain of a return on investment.

1. **Increased Operational Efficiency:** The project's scalable and dynamic picture classification technology is anticipated to greatly increase operating efficiency. The optimization of AWS services and the automation of resource management lead to more efficient operations and lower expenses from manual interventions.
2. **Cost Savings through AWS Utilization:** It is expected that using the Infrastructure as a Service (IaaS) paradigm and AWS services was save a significant amount of money. By minimizing needless charges and maximizing budget allocation, the pay-as-you-go model guarantees that the business only incurs costs proportionate to actual resource consumption.
3. **Market Leadership and Competitive Advantage:** The company is now positioned as a market leader in the computer vision industry thanks to the successful deployment of an advanced image categorization system using AWS. Offering innovative solutions supported by scalable infrastructure can give you a competitive edge, drawing in new business and expanding your market share.
4. **Revenue Growth and Business Expansion:** By providing a variety of sectors with the developed picture classification system as a service, the initiative seeks to diversify sources of income. A higher rate of system adoption may result in higher revenue, more prospects for new ventures, and encouragement of company expansion into untapped markets.
5. **Improved Customer Satisfaction:** Customer satisfaction is increased when an approachable picture classification system is made available via a RESTful web service. Positive word-of-mouth recommendations and repeat business may result from the successful integration and quick analysis of visual data, which can enhance client relationships.
6. **Long-term Cost-Effectiveness:** Long-term cost-effectiveness is ensured by the application's flexibility to dynamically expand, and contract based on demand and by utilizing AWS's affordable cloud resources. The company may adjust to shifting requirements without having to pay extra for overhead.
7. **Innovation and Technological Advancement:** The project promotes creativity by utilizing cutting-edge AWS services like Amazon Rekognition and SageMaker, as well as sophisticated machine learning capabilities like the pre-trained ResNet model. This puts the company in a leading technological position in the computer vision industry.
8. **Financial Well-being and Business Stability:** Through addressing the business demand for effective visual data analysis, the project hopes to improve the organization's stability and financial health. Decision-making processes can be positively impacted by efficient picture categorization systems, which can result in more strategic and well-informed corporate judgments.
9. **Positive Brand Perception and Industry Recognition:** Delivering an image classification solution powered by AWS successfully helps build a favorable perception of the company. The organization's reputation is further enhanced by industry recognition and acknowledgment for its cutting-edge solutions, which draws in clients looking for trustworthy and innovative services.
10. **PROJECT DESCRIPTION:**

**Project Scope Statement:**

### The systemic problem of effective visual data processing in the digital age is addressed by the AWS-Based Image Classification project. Acknowledging computer vision's disruptive potential, the system aims to leverage AWS's strong infrastructure and machine learning capabilities to produce a dynamic and scalable picture classification solution. The project's System Scope Statement summarizes the deliverables, aims, and objectives that are essential to creating a novel image categorization system, thereby capturing the systemic perspective.

### System Objectives:

The system's main objective is to increase operational efficiency by offering an image categorization solution that is adaptable, scalable, and reasonably priced. From a structural perspective, the goals include using AWS's Infrastructure as a Service (IaaS) paradigm to dynamically expand and contract the application based on demand. The system provides an easy-to-use interface that can be accessed via a RESTful web API, with the goal of streamlining the analysis of visual data. Specific objectives include:

1. **Enhance Operational Efficiency:** Implementing a scalable image classification system would increase operational efficiency by enabling the company to handle visual data effectively and dynamically adjust to changing workloads.
2. **Ensure Cost-Effective Resource Utilization:** To maximize resource utilization and ensure cost-effectiveness, use AWS's Infrastructure as a Service (IaaS) architecture to match costs to the real demand for to compute and storage resources.
3. **Facilitate Dynamic System Expansion and Contraction:** Provide an adaptable application that can adjust its size dynamically in response to demand, allowing for effective resource allocation and a reduction in idle instances during times of low demand.
4. **Streamline User Interaction:** By using a static webpage to create a simple and user-friendly interface, you can help users and the picture categorization system communicate more easily.
5. **Utilize Pre-Trained ResNet Model for Accurate Classification:** Use ImageNet labels to apply a pre-trained ResNet model, utilizing advanced machine learning capabilities to guarantee dependable and correct image classification.
6. **Optimize Resource Management with Controller Script:** Develop a controller script to handle SQS queue data and dynamically start new instances, maximizing resource management for effective system performance.
7. **Adapt Number of Instances Based on Image Count:** Optimize resource use and scalability by dynamically adjusting the number of App-tier instances based on the amount of input images during picture submission.
8. **Configure Input and Output S3 Buckets for Data Transfer:** To ensure that photos and classification results are stored and transferred across system components as efficiently as possible, configure specific input and output S3 buckets.
9. **Display Classification Results on Webpage:** Display the categorization results in a key-value pair format on the front end page, giving consumers a clear and thorough understanding of the names of the images and their descriptions.

**System Goals:** The overarching system goals are aligned with the business goals and include:

1. **Achieve Scalability:** Provide a system that can grow and shrink dynamically in response to demand, making it scalable enough to manage different image categorization workloads with ease.
2. **Optimize Resource Utilization:** To maximize resource usage and ensure cost-effectiveness while matching costs with real demand for computing and storage resources, make use of AWS's Infrastructure as a Service (IaaS) architecture.
3. **Enhance Operational Efficiency:** Improve operational effectiveness by shortening procedures, cutting down on manual interventions, and offering a responsive and adaptable image categorization solution.
4. **Provide User-Friendly Interface:** To improve user experience and enable smooth interaction with the picture categorization system, create a visually appealing static web page with an intuitive and user-friendly interface.
5. **Deliver Clear and Comprehensive Results:** Display the categorization results in a key-value pair format on the front-end page, giving consumers a clear and thorough understanding of the names of the images and their descriptions.

### Functionality Inclusions

Functionality Inclusion for AWS Powered Image Classification

1. **Visually Pleasing Web Page:** Using HTML and CSS to create a static webpage with an intuitive user experience.
2. **Flask-Based Communication Framework:** Front-end and back-end communication implemented using the Flask framework.
3. **Boto3 Libraries Integration:** In order to facilitate communication between Web-tier and App-tier EC2 instances, Boto3 libraries are included.
4. **App-tier Instances on EC2:** Deployment of App-tier instances using an ImageNet label-trained ResNet model on EC2.
5. **Controller Script:** Creation of a controller script to dynamically start new instances and manage SQS queue data.
6. **Image Submission Feature:** Adding an upload button to the web interface that allows users to submit dynamic images and adjusts the number of instances according to the number of images entered.
   1. **FUNCTIONALITY EXCLUSION:**

Functionality Exclusions for AWS-Based Image Classification:

1. **User-Authentication-and-Authorization:** Core functionalities are prioritized at this phase, with user login and authorization elements being implemented at a later iteration.
2. **Real-Time Image Processing:** Because of possible complexity and resource constraints, real-time image processing features are deferred to later versions.
3. **Advanced Front-End Features:** To Keep the present implementation simple, more complex front-end functionality like dynamic visualizations or picture previews are postponed to later versions.
4. **Automated Deployment Tools:** The present scope assumes human commencement for the Web-tier and does not include the implementation of automated deployment tools or frameworks.
5. **Custom-Model-Training:** For efficiency, the project uses a pretrained ResNet model in this iteration and defers training new models.
6. **Real-Time System Adaptation:** Future iterations was cover dynamic system growth and contraction, with real-time resource adaptation to variable demand not included in this one.

### COMPLETION AND ACCEPTANCE CRITERIA

Meeting these completion and acceptance criteria ensures the successful implementation and acceptance of the AWS based Image Classification.

* + 1. **Visually Pleasing Web Page:**
* **Deliverable Characteristics:** A static HTML and CSS web page with a visually appealing design.
* **Acceptance Criteria:** The website needs to be visually appealing, adhere to user-centric design guidelines, and work as planned.
  + 1. **Flask-Based Communication Framework:**
* **Deliverable Characteristics:** Flask framework implementation to enable smooth front-end and back-end communication.
* **Acceptance Criteria:** To provide effective and secure communication between the user interface and application logic, the Flask framework must be correctly integrated.
  + 1. **Boto3 Libraries Integration:**
* **Deliverable Characteristics:** Integration of Boto3 libraries to provide seamless communication between EC2 instances running on the Web and App tiers.
* **Acceptance Criteria:** To provide safe and effective data transmission across system components, Boto3 libraries must be used efficiently.
  + 1. **App-tier Instances on EC2:**
* **Deliverable Characteristics:** Deployment of App-tier VMs on EC2 equipped with ImageNet labels for pre-training a ResNet model.
* **Acceptance Criteria:** It is necessary to deploy and set up app-tier instances correctly using the pre-trained ResNet model that has been provided.
  + 1. **Controller Script:**
* **Deliverable Characteristics:** Development of a controller script that was dynamically start new instances and manage SQS queue information.
* **Acceptance Criteria:** To dynamically manage system resources based on SQS queue information, the controller script must work as planned.

## **RISK ASSESSMENT**

Risk Assessment for AWS-Based Image Classification Project:

### Infrastructure Scaling Challenges:

* **Risk Description**: Infrastructure that grows and contracts dynamically in response to demand may make it difficult to allocate resources optimally, which might result in poor performance or higher costs.
* **Mitigation Plan:** To dynamically modify resources, use automatic scaling strategies and proactive monitoring. Review and improve AWS resource settings often to keep up with changing demand.

### Security and Data Privacy Concerns:

* **Risk Description:** There are security and privacy issues while using cloud services, particularly when transferring and storing sensitive photographs. Trust might be damaged and user data compromised by a breach.
* **Mitigation Plan:** Use strong encryption techniques for both in-transit and at-rest data. Put in place frequent security assessments and access limits. To maintain a safe environment, follow AWS security best practices and compliance requirements.

### Dependency on Third-Party Services:

* **Risk Description:** Dependency on third-party services, such the Flask framework, AWS infrastructure, and Boto3 libraries, increases the possibility of service outages or modifications that impact the project's functioning.
* **Mitigation Plan:** Keep aware of upgrades and modifications to third-party services. Put in place fallback plans and version controls. To lessen the effect of a single service failure, diversify your dependents wherever you can.

## **CONSTRAINTS**

### Budgetary Constraints:

* **Description:** The project adheres to a predetermined budget, which restricts the amount of money that may be used for extra services, software, and infrastructure.
* **Implications:** The budgetary restriction may affect which AWS services are chosen, requiring careful allocation to maximize cost-effectiveness.

### Time Constraints:

* **Description:** The project has a time constraint and must be finished according to the planned timeline.
* **Implications:** Time restrictions may have an impact on the project's overall scope, testing rigor, and product implementation depth. Effective time management and work prioritization are essential.

### Resource Availability:

* **Description:** Restricted access to qualified workers for jobs involving development and maintenance.
* **Implications:** The size and experience of the project team may affect how quickly problems are solved, tests are conducted, and developments are made. Making the most use of the resources that are available is crucial.

### Technology and Service Dependencies:

* **Description:** The project depends on a number of technologies, including the Flask framework, the Boto3 libraries, and AWS services, all of which are subject to upgrades and modifications during the course of the project.
* **Implications:** Dependencies provide possible problems in the event that third-party services change. Plans for adaptation, version controls, and close monitoring are required.

### Regulatory Compliance:

* **Description:** Legal and regulatory requirements pertaining to data security and privacy must be followed by the project.
* **Implications:** Data handling procedures and system architecture may be impacted by compliance requirements. Ensuring compliance with pertinent rules is crucial.

## **3.6** **DEPENDENCY LINKAGES**

Dependency Linkages for AWS-Based Image Classification Project:

* + 1. **Dependency on AWS Service Updates:**
* **Description:** The project depends on updates and modifications to AWS services, particularly those offered by SageMaker and Amazon Rekognition.
* **Progress Monitoring:** Maintain up-to-date knowledge about impending changes, engage in pertinent forums, and regularly review AWS service release notes to make sure the project is in line with the most recent features and improvements.
  + 1. **External Library Dependencies (Boto3, Flask):**
* **Description:** The project depends on third-party libraries like Flask for front-end-back-end connection and Boto3 for AWS interaction.
* **Progress Monitoring:** Monitor community conversations, version compatibility, and library upgrades. Examine release notes on a regular basis to determine how library changes may affect the project.
  + 1. **Data Privacy and Regulatory Compliance:**
* **Description:** Monitor community conversations, version compatibility, and library upgrades. Examine release notes on a regular basis to determine how library changes may affect the project.
* **Progress** **Monitoring:** Keep an eye on revisions from legal departments, regulatory agencies, and pertinent industry standards. Make sure the project complies with the ever-changing regulations on data privacy and compliance.
  + 1. **Integration with Third-Party Systems:**
* **Description:** For some features, the project might need to integrate with third-party systems or APIs.
* **Progress** **Monitoring:** Create a clear line of contact with outside suppliers. Evaluate API documentation on a regular basis and keep the project architecture flexible to allow for updates or modifications from outside systems.
  + 1. **User Feedback and Iterative Development:**
* **Description:** Iterative development cycles and user input may be necessary for the project's ongoing progress.
* **Progress Monitoring:** To improve the system over time, set up systems for gathering user feedback, keep an eye on use trends, and design iterative development cycles based on user feedback.

**3.7 IMPACTS**

## Impacts of AWS-Based Image Classification Project on the Organization:

## **1.** **Operational Efficiency Enhancement**

* **Impact:** By automating and simplifying the examination of visual data, the image classification system's installation improves operational efficiency. Decision-making processes become quicker as a result, and productivity increases.
* **Organizational Change Management:** Teams might have to adjust to new picture submission, system interaction, and result interpretation workflows and procedures.

1. **Resource Optimization and Cost Control:**

* **Impact:** The company may optimize resources based on demand by utilizing AWS's scalable infrastructure, which may result in cheaper operating expenses during slow periods.
* **Budgetary Considerations:** The dynamic nature of cloud resource use may cause variations in the organization's operational budget.

1. **Enhanced Data-Driven Decision Making:**

* **Impact:** With the help of the picture categorization system, the company may gain useful insights from visual data and make more educated, data-driven choices.
* **Retraining Needs:** Staff members may require retraining or upskilling in order to understand and use the insights produced by the system, depending on the extent of adoption.

1. **Improved Customer Experience and Satisfaction:**

* **Impact:** If the image categorization system is intended for use by customers, faster and more accurate picture analysis might result in better user experience and higher customer satisfaction.
* **Organizational Perception:** Enhancing the customer experience may have a favorable impact on the organization's market perception and brand image.

### Security and Compliance Adherence:

* **Impact:** It is essential to guarantee data privacy and adherence to legal requirements. The use of encryption and access controls demonstrates the organization's dedication to security.
* **Organizational Reputation:** Users and stakeholders was see the firm more favorably if security and compliance requirements are followed.

### Agility and Scalability Benefits:

* **Impact:** Organizational agility is facilitated by the project's usage of AWS services, which enables the system to dynamically expand resources in response to demand.
* **Operational Flexibility:** The company becomes more adaptable in handling tasks and effectively adapting to changing needs.

1. **Organizational Learning Curve:**

* **Impact:** The earliest phases of adoption may include a learning curve, dependent on the organization's expertise with AWS services and machine learning ideas.
* **Training and Education:** To make sure that staff members are competent in using and maintaining the AWS-based image categorization system, the company might need to spend money on training sessions.

## **MEASURES OF PROJECT SUCCESS**

Measures of Project Success for AWS-Based Image Classification Project:

1. **Accuracy and Precision of Image Classification:**

* **Metric:** Analyze how accurately and precisely the system can identify photos. This metric demonstrates how well the machine learning model performs in producing trustworthy outcomes.
* **Success Criteria:** Achieve a predefined degree of precision and accuracy in picture categorization while reducing false positives and false negatives.

### User Satisfaction and Ease of Use:

* **Metric:** Gather user opinions and gauge their level of satisfaction by conducting usability tests or surveys. Assess the user-interaction system's intuitiveness and simplicity of use.
* **Success Criteria:** Obtain a high user satisfaction rating, which shows that the system is easy to use and meets user needs.

### System Responsiveness and Throughput:

* **Metric:** Track the system's throughput and responsiveness, recording how long it takes to classify a picture and how many user requests it can process at once.
* **Success Criteria:** Reach a throughput and responsiveness level that either meets or surpasses certain performance criteria.

1. **Cost-Efficiency and Resource Utilization:**

* **Metric:** Analyze the project's cost-effectiveness by keeping an eye on cloud infrastructure expenses and resource consumption. Analyze how well operating costs and resource allocation are balanced.
* **Success Criteria:** Attain cost-effectiveness by making the best use of your resources and adhering to your budget.

### Adherence to Security and Compliance Standards:

* **Metric:** Verify and audit the system's compliance with security and compliance requirements on a regular basis. Analyze the performance of data privacy, access control, and encryption.
* **Success Criteria:** Attain excellent ratings in security evaluations and show complete adherence to pertinent regulatory requirements.
  + 1. **Scalability and Dynamic Resource Adaptation:**
* **Metric:** Analyze how well the system can adjust resource levels in response to demand. Calculate how well resources respond to changing workloads.
* **Success Criteria:** Adapt resources to demand and scale them successfully to maintain peak use performance and responsiveness.

### Timely Deployment and Iterative Development:

* **Metric:** Monitor the project's advancement in relation to the specified schedule to guarantee the prompt release of features and updates. Based on user feedback, assess the effectiveness of iterative development cycles.
* **Success Criteria:** Respect the project schedule by completing features and enhancements on time, using iterative development cycles.
  + 1. **Learning and Adaptation Rate:**
* **Metric:** Evaluate the organization's adoption and management of AWS services and machine learning principles in relation to their learning curve. Track how quickly employees become used to the new system.
* **Success Criteria:** Showcase a good learning curve where employees gain proficiency in using and maintaining the image classification system based on AWS.

## **3.9 ASSUMPTIONS**

Assumptions for AWS-Based Image Classification Project:

* + 1. **Continuous support from business sponsor and steering committee:**
* **Assumption**: Throughout the course of the project, it is expected that the Business Sponsor and the Steering Committee was offer continuing support, care, and direction.
* **Rationale:** For the purpose of getting the required permissions, addressing strategic concerns, and guaranteeing constant alignment with company goals, this assumption is essential.
  + 1. **Adequate Resource Availability:**
* **Assumption:** Sufficient resources was made available to staff the project efficiently, including qualified workers, financial allotments, and required technology.
* **Rationale:** This assumption guarantees that the required components are available, which is crucial for the project's success as it depends on having the appropriate knowledge and resources in place.
  + 1. **Stable Internet Connectivity:**
* **Assumption:** To facilitate data transfers, external dependencies, and smooth interactions with AWS services, the project requires a steady and dependable internet connection.
* **Rationale:** Maintaining constant system performance and making optimal use of cloud services require a steady internet connection.
  + 1. **Availability of Required Third Party Services:**
* **Assumption:** For the course of the project, third-party libraries, and services like Boto3 and Flask was remain accessible and supported.
* **Rationale:** To prevent disruptions in system functionality, dependencies on third-party services are expected to stay constant.
  + 1. **User Cooperation and Feedback Participation:**
* **Assumption:** End users was actively contribute during the system's development and deployment phases by offering comments and insights.
* **Rationale:** In order to improve the system based on actual usage and make sure it meets user needs and expectations, user involvement is essential.
  + 1. **Stability of AWS Service:**
* **Assumption:** For the duration of the experiment, AWS services—such as SageMaker and Amazon Rekognition—was continue to be reliable and continuously accessible.
* **Rationale:** It is believed that stability in AWS services was preserve the functionality and dependability of the image categorization system.
  + 1. **Adherence to Data Privacy and Compliance Standards:**
* **Assumption:** In order to ensure the ethical and legal use of visual data, the organization was continue to abide by data privacy and compliance regulations pertinent to the project.
* **Rationale:** Presumption of continuous compliance is essential to preserving the project's reputation and ethical standing.
  + 1. **Flexibility in System Evolution:**
* **Assumption:** The architecture of the system enables flexibility and adaptability to consider changing needs, advancements in technology, and modifications brought about by user input.
* **Rationale:** The flexibility of the system to change and adapt to new requirements without significant overhauls or disruptions is essential to the project's success.

## **CRITICAL SUCCESS FACTORS:**

Critical Success Factors for AWS-Based Image Classification Project:

* + 1. **Organizational Change Management Plan Acceptance:**
* **Factor:** The effective adoption and execution of the Organizational Change Management Plan is necessary to enable seamless transitions, address opposition, and guarantee organizational preparedness for the image categorization system.
* **Importance:** The degree to which the company successfully implements the new system, minimizing interruptions and optimizing user acceptance, depends on how effectively the change management strategy works.
  + 1. **Adequate System Infrastructure Environment:**
* **Factor:** Ensuring that the infrastructure environment of the system—which includes AWS services—is sufficient and dependable in order to support the image classification system's overall performance, responsiveness, and dynamic scalability.
* **Importance:** The infrastructure of the system is the foundation for its operation; any shortcomings can affect user experience, scalability, and performance.
  + 1. **Effective User Training and Education:**
* **Factor:** Ensuring that staff members are competent in utilizing the capabilities of the image classification system requires the successful execution of extensive user education and training programs.
* **Importance:** Users' ability to use the system efficiently was determine the project's success, therefore efficient training is essential for comprehending the system and maximizing its utilization.

### Stakeholder Communication and Engagement:

* **Factor:** Regular and open communication with all relevant parties, such as the steering committee, business sponsor, and end users, in order to gather input, resolve issues, and guarantee alignment with corporate objectives.
* **Importance:** Encouraging active involvement from stakeholders is essential to keeping the project in line with organizational goals and getting the support it needs over its entire life.
  + 1. **Adherence to Data Privacy and Compliance Standards:**
* **Factor:** Strict respect to compliance and data protection standards, guaranteeing that the picture categorization system functions morally, legally, and compliantly with laws.
* **Importance:** Adherence to data privacy rules is a vital success element since noncompliance can have negative legal and reputational effects.

### Scalability and Performance Optimization:

* **Factor:** The system's capacity to adjust resources dynamically in response to demand and to maximize efficiency in order to efficiently manage a range of workloads.
* **Importance:** Scalability guarantees that the system can adapt to shifts in usage patterns and maintain optimal performance during periods of high demand without adding needless expenses.

### 7. Effective Collaboration and Cross-Functional Coordination:

* **Factor:** Developers, IT, and operations teams are just a few of the cross-functional teams that successfully collaborated and coordinated throughout the project to guarantee a coherent and cohesive approach.
* **Importance:** A comprehensive approach to project management and execution is made possible by seamless cooperation, which also prevents silos and increases productivity.
  + 1. **Iterative Development Based on User Feedback:**
* **Factor:** The incorporation of a functional feedback loop guarantees that the system develops in accordance with user expectations and demands by permitting iterative development cycles based on user input.
* **Importance:** Iterative development guarantees that the system stays valuable and relevant over time, improves user happiness, and pinpoints opportunities for improvement.

1. **Functional Requirements**

The system's capabilities and aims are defined by the functional requirements, which work together to guarantee that the AWS-Based Image Classification Project meets its objectives and produces useful results for users and stakeholders.

* 1. **Business Functional Requirements by Group**

## **Image Processing and Classification:**

* **Automated Image Labeling:** Using machine learning techniques, the system automatically groups submitted photographs into predetermined groupings. This procedure entails examining the visual content of photos in order to precisely tag pertinent information, making it easier to organize and retrieve visual data.
* **Bulk Image Processing:** The system allows users to upload many photos at once, which are processed in batches. This feature's goal is to efficiently manage large amounts of picture data, cutting down on waiting times for classification results and raising user satisfaction.
* **Image Data Management:** All uploaded photographs and their categorization outcomes are kept up to date in an organized repository by the system. This guarantees that data is maintained efficiently and is kept current. It also offers safe storage, simple access for inspection, and the flexibility to categorize or amend picture information as needed.
  + - 1. **User Interaction and Feedback:**
* **User Account Management:** In order to provide customized settings and preferences for picture categorization, users may establish and maintain their accounts. Security elements like login passwords and customizable categorization preferences are part of account management.
* **Real-Time Classification Feedback:** Users receive instant feedback on the categorization results after uploading photographs. Through this interaction, users may check if the system's classifications are accurate and can offer corrections or extra information to improve the system's accuracy and learning over time.
* **Classification History and Analytics:** In addition to analytics on the most common image types submitted and their classification outcomes, users may view their historical history of uploaded photos and classification results. This feature makes it easier for customers to make better decisions for future uploads by assisting them in understanding trends in their picture data and the system's performance.
  + - 1. **System Management and Scalability:**
* **Resource Scaling Based on Demand:** Based on the current workload, the system dynamically modifies its computing resources to ensure that picture classification jobs are processed efficiently without using excessive amounts of resources. During periods of high consumption, this feature keeps up excellent performance while supporting economical operations.
* **Security and Data Protection:** Makes certain that all user data is handled and kept securely, including uploaded photos and categorization results. Giving users peace of mind regarding the security and privacy of their data entails putting strong encryption, access restrictions, and compliance with data protection laws into practice.
* **System Monitoring and Maintenance:** Continuously monitoring of user activity, error logs, performance data, and system operational status. By taking a proactive stance, problems may be found and fixed quickly, keeping the system current, dependable, and operating at peak efficiency.
  1. **System Functional Requirements**

System Functional Requirements for AWS-Based Image Classification Project

1. **Web Services and User Interface:**

* **Web Service Accessibility:** For smooth operation, the system needs a web service that users may access. It would also be a good idea to investigate the creation of a user-friendly website and possible mobile applications. If applicable, integration with social media platforms has to be specified in order to take advantage of these avenues for system marketing and user interaction.

1. **Security and Privacy:**

* **Physical and System Security:** Strict security protocols, including both physical and system-level security, must be followed by the system. Restricted access to server rooms, data centers, and backup storage sites are among the physical security requirements. To protect sensitive information and guarantee that only authorized users may carry out particular tasks inside the system, system security measures like encryption, access restrictions, and role-based permissions must be put in place.

1. **Recovery:**

* **Business Continuity Planning:** The company must establish recovery time goals (RTOs) that specify how long vital functions may continue in the event of a severe system breakdown. To develop a staged recovery strategy, prioritized functions that must be accessible first should be determined. It's also important to have backup plans in place in case the system fails for a short while.

**4. System Availability and Reliability:**

* **Availability Targets:** Give consumers a clear understanding of the expected system availability by outlining the required access times for the application. Determine when different user groups was utilize the system at its peak to maximize resources during busy times. While taking user preferences into account and attempting to minimize interruptions, define permissible downtimes for maintenance or upgrades.

**5.General Performance:**

* **Response Time Expectations:** Describe how long you estimate the system to take to respond to user updates and requests. To guarantee the best possible user experience and system efficiency, define performance metrics. In order to determine system load and performance needs, predict the expected quantities of transactions on a monthly, daily, and hourly basis.

**6. Capacity:**

* **Transaction Volumes and Growth Trends:** Give an indication of the anticipated transaction volumes over a period of time, without becoming too detailed. Examine growth patterns to make sure the system is able to accommodate rising usage and requests for data processing. This entails taking into account elements like user interactions, classifications, and picture submissions.

**5. PROJECT APPROACH**

**5.1 Planned Approach**

To guarantee effective development, testing, and deployment, the AWS-Based Image Classification Project was implemented using a systematic methodology. The proposed approach includes **a few** important aspects:

### Phased Development:

* There was several phases to the project, with each concentrating on features and functions.
* A more controllable development process is made possible by phasing, which permits ongoing testing and improvement.

1. **Outsourcing and Collaboration:**

* Outsourcing some jobs, including visual design components or specialist machine learning knowledge, was taken into consideration.
* Working with other specialists might speed up some parts of the project and improve its overall quality.

### Resource Management:

* To guarantee timely growth, temporary resources may be employed during periods of high development.
* Variable workloads can be accommodated with a flexible staffing strategy, which was optimize resource allocation as needed.

### Testing Environments:

* To replicate diverse circumstances, multiple testing environments with varying picture collections and user loads was built up.
* Staging environments for validation and pre-production testing was included in the environments.

### User Feedback Integration:

* User feedback was regularly integrated into the development process through the use of an iterative approach.
* Feedback loops was help to ensure that the system is developed in a way that meets user expectations.

### Continuous Integration and Deployment (CI/CD):

* Processes related to testing, integration, and deployment was automated using CI/CD pipelines.
* Automation lowers the possibility of deployment problems by ensuring fast and dependable updates.

### Documentation and Training:

* Every stage was accompanied by thorough documentation that includes user manuals, APIs, and system architectural details.
* Administrators, support personnel, and end users was all get training.

### Performance Monitoring and Optimization:

* There was ongoing system performance monitoring, with an emphasis on resource usage and response times.
* We was use optimization techniques to deal with any performance bottlenecks found.

### Security Audits and Compliance Checks:

* There was frequent security audits to find and fix any possible weaknesses.
* Compliance with data privacy and security regulations shall be upheld and confirmed.

### Scaling Strategies:

* To adjust to shifting workloads, dynamic resource scaling strategies was put into practice.
* Based on use trends, proactive scaling choices was made easier with the help of monitoring tools.

1. **Collaboration with AWS Support:**

* The establishment of collaboration with AWS support services aims to harness knowledge and effectively resolve any issues or questions pertaining to AWS services.
* Consistent participation was guarantee conformity with optimal methodologies and developing technology.

1. **User Acceptance Testing (UAT):**

* To make sure the system satisfies user needs, UAT was carried out prior to every significant release.
* Testing procedures was guided by user acceptance criteria to guarantee user satisfaction.

1. **PROJECT COST AND RESOURCE ESTIMATES**

## **MAJOR MILESTONES - ESTIMATED SCHEDULE**

|  |  |  |
| --- | --- | --- |
| **Milestone** | **Date** | **Comments** |
| Project Kickoff and Scope Agreement | Jan 26th, 2024 | Declare the project's final scope, get stakeholder approval for its goals, and formally start the work. |
| AWS Service Activation and Testing | Feb 10th, 2024 | Activate selected AWS services and start initiating testing to verify correct operation. |
| User-Interactive System Development Begins | Mar 1st, 2024 | Start building the user-interactive system using features listed in the project scope. |
| System Development | Apr 12th, 2024 | Make the system available to end users in the production environment and ensure a smooth changeover. |
| **Development Phase Begins** | **Sept 7th, 2024** | Start of the semester; initiate project development activities. |
| **Data Storage and Management Setup** | **Sept 14th, 2024** | Finalize configuration of AWS S3 for image data storage. |
| **Image Data Collection and Preparation** | **Sept 21st, 2024** | Gather and preprocess images for the machine learning model. |
| **Model Development and Training** | **Oct 5th, 2024** | Develop and train the image classification model using SageMaker. |
| **System Integration and Testing** | **Oct 19th, 2024** | Integrate all system components and conduct testing for functionality. |
| **User Interface Development** | **Nov 2nd, 2024** | Create the user interface for image submission and results display. |
| **Final System Testing and User Training** | **Nov 16th, 2024** | Conduct final testing and provide training for end users. |
| **Project Closure and Review** | **Nov 30th, 2024** | Finalize documentation and review outcomes with stakeholders. |

## RESOURCE REQUIREMENTS – TEAM AND SUPPORT RESOURCES

The following personnel resources are required to complete this project:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ROLE OR NAME** | **DEPARTMENT** | **EST HOURS** | **RATE** | **EST TOTAL** |
| Project Manager | Project management | 80 hours | $80/hour | $6400 |
| Software Developers (Front-End + Back-End) | Development | 200 hours each | $70/hour | $28000 |
| UI/UX Designer | Design | 60 hours | $40/hour | $2400 |
| Quality Assurance Analyst | Testing | 80 hours | $60/hour | $4800 |
| Business Analyst | handles business perspectives of the system | 30 hours | $30/hour | $900 |
| Training and Support Specialist | Training stakeholders how the system works | 85 hours | $55/hour | $4675 |
| Technical Writer | Documenting all the project tasks | 40 hours | $50/hour | $2000 |
| **Subtotal** |  | 575 hours |  | $49175 |

## **ESTIMATED COSTS**

As with major milestones, costs are estimates during the creation of the Scoping Document… when the Final Scope Document is agreed to, these initial costs no longer considered estimates, but are then committed cost…costs may vary during the course of a project…but the budget is then managed outside of this document…

### HARDWARE EXPENSES

|  |  |
| --- | --- |
| **Item** | **Estimated Amount ($)** |
| 1 laptop | 1,000 |
| Networking equipment | 800 |
| AWS Cloud Services | 5000 |
| Ongoing costs | 500 |
| **Total** | **7300** |

**SOFTWARE EXPENSES**

|  |  |
| --- | --- |
| **Item** | **Estimated Amount ($)** |
| Licensed Windows Operating  systems for laptop | 600 |
| Web Services | 1,500 |
| Python for backend | 1,000 |
| Ongoing costs | 1,000 |
| **Total** | **4,100** |

### APPROVED PROJECT COST ESTIMATES

|  |  |  |  |
| --- | --- | --- | --- |
| **Department** | **Resource** | **Expense** | **Estimated Amount**  **($)** |
| Human Resource | Resource | Team and Support | 49,175 |
| IT/Procurement | Hardware | All hardware expenses | 7,300 |
|  |  |  |  |
|  | Software | All software Cost | 4,100 |
| Various | Marketing | Marketing | 10,000 |
|  | Others | Other ongoing cost | 3,000 |
| **Total** | | | **$73,575** |

Total Estimated Budget: **$73,575**

### Resources:

### Personnel resources, such as project management, development, testing, documentation, training, and support staff, account for the majority of the projected expenditure.

### The overall cost of resources is influenced by the hourly rates for each position.

### Hardware:

### It makes use of AWS Cloud Services, such as load balancing for effective traffic distribution and EC2 instances for hosting applications.

### Hardware expenses are covered by the total AWS service charges.

### Software:

### Software costs are kept to a minimum since most development tools, libraries, and frameworks are open source.

### By utilizing widely used, reasonably priced development resources, costs are reduced to a minimum.

# PROJECT CONTROLS

## CHECKPOINT FUNDING SCHEDULE

Funding checkpoints facilitates efficient progress and decision-making by ensuring that sufficient resources are allotted at critical project phases.

* Project Initiation.
* Requirements Gathering and Analysis
* AWS infrastructure Setup
* UI System Development
* System Deployment Readiness
* Post-Deployment Monitoring and Support
  1. **WEEKLY/MONTHLY STATUS AND STEERING COMMITTEE MEETINGS**

The development process was involve defining the meetings and timetables. However, the project sponsor (Prof. Mark Anthony Green) was **take weekly** meetings on Saturdays to look at the project's progress.

### WEEKLY/MONTHLY STATUS REPORTS

Before the weekly meeting starts, status reports was written and sent in. Every Saturday, the project manager prepares and presents the reports to the project sponsor.

### 7.4 Risk Management

A better understanding of each recognized risk, including probability, severity, and associated mitigation strategy, is given by this well-organized list.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **RISK ID NUMBER** | **IDENTIFIED RISK** | **SEVERITY** | **PROBABILITY** | **MITIGATION PLAN** |
| RISK-001 | Insufficient AWS knowledge | HIGH | Medium | Provide the team with training on AWS and get advice from experts certified by AWS about crucial setups and performance optimizations. |
| RISK-002 | Security vulnerabilities | HIGH | MEDIUM | Develop strict security procedures, routine audits, and security best practices training for staff members. For data that is both in transit and at rest, use encryption. |
| RISK-003 | Changes in the project scope | MEDIUM | HIGH | Create a comprehensive change management procedure that evaluates the effects of scope changes on the budget and schedule and requires the project management committee's permission for any modifications. |
| RISK-004 | Budget overruns | MEDIUM | MEDIUM | Set up notifications for budget thresholds to prevent unexpected overruns and use AWS cost management tools to monitor and limit spending. |

### 

### Issue Management

Issues was addressed with the Project Manager and communicated in the project status report.

* **Issue Tracking and Documentation:** As problems appear, record all relevant information in a centralized issue tracking system, such as the issue's description, level of severity, and identification date. This ensures that nothing escapes attention and offers a historical account for later use.
* **Prioritization and Assignment:** Prioritize each problem as High, Medium, or Low based on how it could affect the project's scope, timing, and quality. Assign problems to the team member with the most experience and the least amount of work on your plate to guarantee a successful resolution.
* **Resolution and Monitoring:** Create and carry out action plans with the required resources and modifications for high-priority situations. Keep a careful eye on these resolutions' development to make sure they are headed toward conclusion and make any required adjustments to their plan.
* **Communication and Updates:** Report important issues and their developments to all stakeholders on a regular basis through meetings or project status reports. This openness promotes a cooperative atmosphere for solving issues and helps in managing expectations.

### CHANGE MANAGEMENT

The change control procedures to be followed was consistent with Project Management Procedures and consist of the following processes:

* Change Control Database Establishment by the project manager to thoroughly monitor every project-related change. Every alteration was have its own distinct identity for exact documentation.
* All Change Requests was go through an in-depth evaluation in order to determine possible choices and associated costs. This evaluation ensures that the implications of the recommended changes are properly acknowledged.
* Change Requests was reviewed and approved by the project sponsor.
* The effects of approved Change Requests on the scope and schedule of the project was reflected in updates to the project plan.
* The Change Control database was updated frequently to accurately represent Change Requests' current status. This provides interested parties with the most recent information on any approved or ongoing modifications.
  1. **Communication Management**

The following strategies have been established to promote effective communication within and about this project:

* The Project Manager was provide the Project Owner with a weekly status report on the project. The project manager was decide whether to call ad hoc meetings when new problems or change control items come up.
* The Project Manager was provide a written status report to the Project Owner each month. The monthly progress report was provided alongside the minutes of the project team meetings.
* The project owner was get emails on any important issues. The urgency of the service request was clear from the impact and time constraints of the problem notifications.
* The project team was hold weekly update/status meetings to assess activities that have been performed and set current work priorities. Minutes from each meeting was used to document the discussions and decisions made.

### ADDITIONAL PROJECT DETAILS

Here are additional project details for this project:

1. **Project Objectives:** The goal of this project is to create an image classification system that properly and effectively classifies photos into predetermined groups by utilizing AWS services. This system might have a lot of uses, such providing sophisticated AI-driven analytics in the healthcare, retail, and security industries, or helping with digital asset management.
2. **Key Technologies and Servies:**

* **Amazon S3:** To save and access any quantity of data at any moment. It was act as the foundation for the storage of unprocessed photos, processed data, and outputs from categorization.
* **AWS Lambda:** To ensure scalability and effective resource utilization, the classification code is executed in response to triggers (such as new picture uploads to S3).
* **Amazon Rekognition:** To utilize AI models that have already been trained and tuned for certain categorization tasks in order to analyze images.
* **AWS EC2:** To have more control over the setting in which models for image processing and classification are created and evaluated.
* **Amazon SageMaker:** To develop, improve, and implement large-scale machine learning models. It offers an integrated notebook interface for Jupyter that makes working with models easy.
* **AWS IAM:** To securely control access to AWS services and resources, making sure that only those with permission may use certain resources.

1. **Potential Challenges and Considerations:**

* **Data Privacy and Security:** Adopt strict security protocols to safeguard confidential information while adhering to data protection laws (e.g., GDPR, HIPAA).
* **Model Bias and Fairness:** Adopt strict security protocols to safeguard confidential information while adhering to data protection laws (e.g., GDPR, HIPAA).
* **Continuous Learning:** Include features that allow the system to gradually learn from fresh data, increasing accuracy and adjusting to different kinds of images.
* **Integration with Existing Systems:** Verify that the target industry sectors' current databases, CRM, or ERP systems are compatible and simple to integrate with.

1. **Supporting Materials:**

* AWS documentation for each service utilized (e.g., such as AWS Lambda, Amazon SageMaker, Amazon S3, and Amazon Rekognition).
* Studies and industry standards for image categorization methods and performance indicators.
* Case studies of comparable deployments in related sectors to comprehend the effect and application scenarios.

### PROJECT DICTIONARY

1. **AWS (Amazon Web Services):** 
   * A cloud computing platform that provides a range of services, including database storage and processing power.

### Image Classification:

* + A computer vision challenge in which an image from a preset list of categories is labeled by an algorithm.

### Amazon S3(Simple Storage Service):

* + A web service that allows users to store and retrieve any volume of data from any location at any time.

### AWS Lambda:

* + A serverless computing service that autonomously maintains the necessary computer resources and executes code in response to events.

### Amazon Rekognition:

* + A service that uses proven, highly scalable deep learning technology to make it simple to integrate picture and video analysis to apps.

### EC2(Elastic Compute Cloud):

* + A web service that makes web-scale computing simpler by offering resizable compute resources in the cloud.

# **PROJECT ROLES AND STAKEHOLDERS**

## **PROJECT ROLES**

The following role definitions are being applied to the resources assigned to this project:

|  |  |
| --- | --- |
| **Project Sponsor** | Provides executive team approval and sponsorship for the project. Has budget ownership for the project and is the major stakeholder and recipient for the project deliverables. |
| **Project Owner** | Provides policy definition to the Project team. Resolves all policy issues with the appropriate policy owners in order to provide a clear, decisive definition. Makes final decisions and resolves conflicts or issues regarding project expectations across organizational and functional areas. The project owner and the project manager have a direct link for all communication. The project manager was work directly with the project owner on all policy clarification. |
| **Project Manager** | Provides overall management to the project. Accountable for establishing a Project Charter, developing and managing the work plan, securing appropriate resources and delegating the work and insuring successful completion of the project. All project team members report to the project manager. Handles all project administrative duties, interfaces to project sponsors and owners and has overall accountability for the project. |
| **Steering Committee** | Provide assistance in resolving issues that arise beyond the project manager’s jurisdiction. Monitor project progress and provide necessary tools and support when milestones are in jeopardy. |
| **Stakeholder** | Key provider of requirements and recipient of project deliverable and associated benefits. Deliverable was directly enhance the stakeholders’ business processes and environment. Majority of stakeholders for this project was agency heads, CIO’s and project management representatives. |
| **Team Member** | Working project team member who analyzes, designs and ultimately improves or replaces the business processes. This includes collaborating with teams to develop high level process designs and models, understanding best practices for business processes and partnering with team members to identify appropriate opportunities, challenging the old rules of the |

## **PROJECT STAKEHOLDERS**

The following people have been identified as project stakeholders. Also listed is the stakeholder role.

|  |  |  |
| --- | --- | --- |
| **NAME** | **DEPARTMENT** | **ROLE** |
| Rajeswari Tadepalli | Department of Computer  Science | Student/Developer/Project  Manager/ |
| Prof. Mark Anthony Green | Department of Computer  Science | Instructor/Project Sponsor |
| Monroe College | Monroe College | Project Owner |

# **PROJECT PLAN / GANTT CHART**

* 1. **Project Plan**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Index** | **Tasks** | **Original Start Date** | **Revised Start Date** | **Actual Start Date** | **Original End Date** | **Revised End Date** | **Actual End Date** | **Duration** | **Day** | **%Done** | **Status** |
| **Project** | **Driver Drowsiness Detection** | **6-Jan-24** | **n/a** | **6-Jan-24** | **12-Apr-24** | **n/a** | 12-Apr-24 | **97** | **38** | **39%** | In Progress |
| **1** | **Project Selection and Initiation** | **6-Jan-24** | **n/a** | **6-Jan-24** | **20-Jan-24** | **n/a** | **20-Jan-24** | **14** | **14** | **100%** | Complete |
| 1.1 | Project team introduction | 6-Jan-24 | **n/a** | 6-Jan-24 | 13-Jan-24 | **n/a** | 13-Jan-24 | **7** | 7 | 100% | Complete |
| 1.2 | Project selection | 13-Jan-24 | n/a | 13-Jan-24 | 16-Jan-24 | n/a | 16-Jan-24 | 3 | 3 | 100% | Complete |
| 1.3 | Project description | 16-Jan-24 | n/a | 16-Jan-24 | 18-Jan-24 | n/a | 18-Jan-24 | 2 | 2 | 100% | Complete |
| 1.4 | Project Approval | 18-Jan-24 | n/a | 18-Jan-24 | 20-Jan-24 | n/a | 27-Jan-24 | 7 | 7 | 100% | Complete |
| **2** | **Scope and Charter Documentation** | **20-Jan-24** | **n/a** | **27-Jan-24** | **27-Jan-24** | **n/a** | 28-Jan-24 | **12** | **12** | **100%** | Complete |
| 2.1 | Project objectives definition | 20-Jan-24 | n/a | 27-Jan-24 | 27-Jan-24 | n/a | 29-Jan-24 | 2 | 2 | 100% | Complete |
| 2.2 | Requirements Identification | 22-Jan-24 | n/a | 29-Jan-24 | 29-Jan-24 | n/a | 31-Jan-24 | 2 | 2 | 100% | Complete |
| 2.3 | Project Cost Estimation | 24-Jan-24 | n/a | 31-Jan-24 | 31-Jan-24 | n/a | 2-Feb-24 | 2 | 2 | 100% | Complete |
| 2.4 | Project controls Identification | 26-Jan-24 | n/a | 2-Feb-24 | 2-Feb-24 | n/a | 4-Feb-24 | 1 | 1 | 100% | Complete |
| **3** | **Project Plan & budget** | **27-Jan-24** | **n/a** | **4-Jan-24** | **14-Feb-24** | **n/a** | **14-Feb-24** | **38** | **38** | **94%** | In Progress |
| 3.1 | Tasks Break down | 27-Jan-24 | n/a | 4-Feb-24 | 30-Jan-24 | n/a | 5-Feb-24 | 1 | 1 | 100% | Complete |
| 3.2 | Assign Resources | 30-Jan-24 | n/a | 5-Feb-24 | 1-Feb-24 | n/a | 6-Feb-24 | 1 | 1 | 100% | Complete |
| 3,3 | schedule tasks | 1-Feb-24 | n/a | 6-Feb-24 | 5-Feb-24 | n/a | 7-Feb-24 | 1 | 1 | 100% | Complete |
| 3.4 | draw plan/Gannt Chart | 5-Feb-24 | n/a | 10-Feb-24 | 10-Feb-24 | n/a | 13-Feb-24 | 3 | 3 | 100% | Complete |
| 3.5 | create budget | 10-Feb-24 | n/a | 10-Feb-24 | 14-Feb-24 | n/a | 14-Feb-24 | 3 | 3 | 45% | In Progress |
| **4** | **Design DFDs and user Stories** | **14-Feb-24** | **n/a** | **14-Feb-24** | **24-Feb-24** | **n/a** | **24-Feb-24** | **10** | **10** | 0% | Not Started |
| 4.1 | Identify system entities and | 14-Feb-24 | n/a | 14-Feb-24 | 16-Feb-24 | n/a | 16-Feb-24 | 2 | 0 | 0% | Not Started |
| 4.2 | Determine entity relationships | 16-Feb-24 | n/a | 16-Feb-24 | 18-Feb-24 | n/a | 18-Feb-24 | 2 | 0 | 0% | Not Started |
| 4.3 | Identify User stories | 18-Feb-24 | n/a | 18-Feb-24 | 20-Feb-24 | n/a | 20-Feb-24 | 2 | 0 | 0% | Not Started |
| 4.5 | Draw DFDs | 20-Feb-24 | n/a | 20-Feb-24 | 24-Feb-24 | n/a | 24-Feb-24 | 4 | 0 | 0% | Not Started |
| **5** | **Database Design** | **24-Feb-24** | **n/a** | **24-Feb-24** | **16-Mar-24** | **n/a** | **16-Mar-24** | **21** | **0** | **0%** | Not Started |
| 5.1 | Identify tables and columns | 24-Feb-24 | n/a | 24-Feb-24 | 29-Feb-24 | n/a | 29-Feb-24 | 5 | 0 | 0% | Not Started |
| 5.2 | Identify keys and foreign keys | 29-Feb-24 | n/a | 29-Feb-24 | 2-Mar-24 | n/a | 2-Mar-24 | 2 | 0 | 0% | Not Started |
| 5.3 | mid-project presentation | 2-Mar-24 | n/a | 2-Mar-24 | 9-Mar | n/a | 9-Mar | 7 | 0 | 0% | Not Started |
| 5.3 | Map table relationships | 9-Mar-24 | n/a | 9-Mar-24 | 12-Mar-24 | n/a | 12-Mar-24 | 3 | 0 | 0% | Not Started |
| 5.4 | Draw ERDs | 10-Mar-24 | n/a | 10-Mar-24 | 16-Mar-24 | n/a | 16-Mar-24 | 6 | 0 | 0% | Not Started |
| **6** | **Application Design** | **16-Mar-24** | **n/a** | **16-Mar-24** | **23-Mar-24** | **n/a** | **23-Mar-24** | **7** | **0** | **0%** | Not Started |
| 6.1 | System screen definition and | 16-Mar-24 | n/a | 16-Mar-24 | 18-Mar-24 | n/a | 18-Mar-24 | 2 | 0 | 0% | Not Started |
| 6.2 | UI & UX Design | 18-Mar-24 | n/a | 18-Mar-24 | 23-Mar-24 | n/a | 23-Mar-24 | 5 | 0 | 0% | Not Started |
| **7** | **Network Design** | **23-Mar-24** | **n/a** | **23-Mar-24** | **30-Mar-24** | **n/a** | **30-Mar-24** | **7** | **0** | **0%** | Not Started |
| 7.1 | Select Network Design | 23-Mar-24 | n/a | 23-Mar-24 | 25-Mar-24 | n/a | 25-Mar-24 | 2 | 0 | 0% | Not Started |
| 7.2 | Draw the Network Design | 24-Mar-24 | n/a | 24-Mar-24 | 28-Mar-24 | n/a | 28-Mar-24 | 4 | 0 | 0% | Not Started |
| 7.3 | Define Security requirements | 28-Mar-24 | n/a | 28-Mar-24 | 30-Mar-24 | n/a | 30-Mar-24 | 2 | 0 | 0% | Not Started |
| **8** | **Final Documentation &** | **30-Mar-24** | **n/a** | **30-Mar-24** | **12-Apr-24** | **n/a** | **12-Apr-24** | 13 | 0 | 0% | Not Started |
| 8.1 | Final interview & presentation | 30-Mar-24 | n/a | 30-Mar-24 | 6-Apr | n/a | 6-Apr | **7** | **0** | **0%** | Not Started |
| 8.3 | Final Documentation | 6-Apr-24 | n/a | 6-Apr-24 | 12-Apr-24 | n/a | 12-Apr-24 | 6 | 0 | 0% | Not Started |

* 1. **GANTT CHART**

**A graph of a project

Description automatically generated with medium confidence**

1. **DATA FLOW DIAGRAM WITH USE CASES**
   1. **DATA FLOW DIAGRAMS**

**12.1.1 CONTEXT DIAGRAM**

A diagram of a computer

Description automatically generated

**12.1.2 DFD LEVEL 0 DIAGRAM**

A diagram of a user interface

Description automatically generated

**12.1.3 DFD LEVEL 1 DIAGRAM**

A diagram of a computer system

Description automatically generated

* 1. **USER STORIES**
* As a user, I want to be able to upload images to the system through a user-friendly web interface, so that I can initiate the image classification process.
* As a user, I want to be able to upload multiple images simultaneously, so that I can efficiently initiate the classification process for a batch of images.
* As a client, I want the system to provide real-time feedback on the classification progress, so that I can monitor the status of image processing tasks.
* As a system administrator, I want to receive automated alerts when system resources approach their limits, so that I can proactively manage and optimize resource allocation.
* As a user, I want the system to provide an option for manual verification and correction of image classifications, so that I can ensure the accuracy of the results.
* As a client, I want the system to maintain a log of all image classification activities, so that I can review and audit the processing history for compliance and analysis purposes.
* As a data analyst, I want to be able to export classification results in a standardized format for further analysis and reporting, so that I can integrate the output into other data workflows.
* As a user, I want the system to automatically handle image pre-processing tasks such as resizing and normalization, so that I can streamline the input data preparation process.
* As a developer, I want the system to provide comprehensive documentation and APIs for seamless integration with external applications and services, so that I can build robust image classification workflows.
* As a client, I want the system to offer secure access controls and user authentication mechanisms, so that I can ensure data privacy and regulatory compliance in my image classification activities.

1. **NETWORK DIAGRAM AND SECURITY MEASURES**

**13.1 Network Diagram**

A screenshot of a video game

Description automatically generated

**Fig: Network Design for AWS powered Image Classification**

The diagram above shows the high-level network design for the AWS powered Image Classification. Here is the brief explanation about it:

**Client:** To upload photos for categorization, users interact with the program via a web or mobile client application.

**AWS route 53:** As a traffic controller, this service directs user requests to the right AWS infrastructure resources.

**Web Application Firewall (WAF):** This service assists in defending the application from fraudulent attacks and typical online vulnerabilities.

**Amazon API Gateway:** The client application can upload images via the public API endpoint that this service creates.

**Amazon CloudFront:** In its capacity as a content delivery network (CDN), this service caches frequently visited data and delivers material to users quickly, maybe from edge sites that are closer to them geographically. When users submit photographs, responsiveness is improved as a result.

**AWS Lambda:** This serverless computing service may be triggered by the API Gateway when it receives an image upload request. It might initiate the picture categorization process or carry out simple image pre-processing (resizing, format conversion).

**Amazon S3 Bucket:** The submitted photos are kept safe in a secure bucket by this object storage service.

**Amazon SageMaker (or Rekognition)**

* **SageMaker:** SageMaker is probably the best option if the case calls for using your data to train a bespoke picture classification model. It offers the infrastructure and tools needed to train and use machine learning models.
* **Rekognition:** SageMaker wouldn't be needed in this scenario if the application uses Rekognition's pre-trained models for picture categorization. Rekognition is capable of immediately analyzing photos to identify faces, objects, and scenes.

**Amazon EC2:** When utilizing SageMaker for inference or training, EC2 instances offer sufficient processing capacity to execute the model.

**Amazon Simple Notification Services (SNS):** For example, this service might be used to notify users when there are problems in the picture categorization process or when human involvement is needed.

**Amazon CloudWatch:** This service keeps an eye on how well the program is working and provides information on faults, resource use, and general system functionality.

**13.2 SECURITY MEASURES**

**Security:** To defend against online threats, the design includes security features like WAF. To regulate access to AWS resources, IAM (Identity and Access Management) would probably be set up.

**User Lever Security:**

* **User Authentication:** Use multi-factor authentication (MFA) or other robust authentication techniques to confirm users' identities prior to allowing them to access the application. Implement authorization policies to limit access to features according to the roles and permissions of users.
* **User Education:** Educate users on safe password usage and how to avoid clicking on dubious links or files.

**Application-Level Security:**

* **AWS-WAF:** The current WAF integration aids in defending against frequent online vulnerabilities. Think about modifying WAF rules to handle certain risks that are pertinent to your application.
* **API Gateway Security**: Put access control measures in place for the API Gateway endpoint. To limit access to just approved users or apps, you can use AWS IAM rules or API keys.
* **Data Encryption:** To safeguard sensitive information from being intercepted, encrypt data both in transit (using HTTPS) and at rest (such as uploaded photos in S3).

**Database-Level security:**

* **AWS-IAM:** Configure IAM roles to provide least privilege access to the database if the design makes use of a relational database provider such as Amazon RDS. Access to and modification of database contents should be restricted to approved users or applications.
* **Data Encryption**: To provide an extra degree of security, think about encrypting data kept inside the database itself.
* **Database Auditing**: To monitor user activity and spot any shady access attempts, turn up database auditing.

1. **E-R DIAGRAM**

**Entities and their Attributes:**

1. **User**

* UserID (Primary Key)
* UserName
* Email

1. **Image**

* ImageID (Primary Key)
* UserID (Foreign Key links to User)
* UploadTImestamp
* ImageURL (Stored in S3)
* Status (e.g., pending, processed)

1. **Classification**

* ClassificationID (Primary Key)
* ImageID (Foreign Key links to Image)
* Label
* ConfidenceScore

1. **Instance**

* InstanceID (Primary Key)
* Status (e.g., active, idle, terminated)
* Type
* LaunchTime

1. **QueueMessage** (this represents messages in SQS for image processing tasks)

* MessageID (Primary Key)
* ImageID (Foreign Key links to Image)
* Status (e.g., pending, in\_progress, completed)

**Entity Relationship Diagram:**

A diagram of a system

Description automatically generated

**Relations:**

**User to Image:**

* One to Many
* A user can upload multiple images, but each image is uploaded by only one user.

**Image to Classification:**

* One to Many
* An Image can have multiple classifications, especially if re-classified or classified with models.

**Image to QueueMessage:**

* One to One
* Each image to be classified results in one message in the SQS queue, assuming a simple workflow where each image is processed once.

**Instance to QueueMessage:**

* One to Many
* An instance can process multiple queue messages, but each queue message is processed by one instance at a time.

1. **SCREENSHOTS**

**Component 1: Screen Map for Image Classification System**

**Menu:** "Upload Image," "View Gallery," and "Settings" are accessible via the menu.

**Upload Image:** Points either towards the "Classification Results" or back to the "Main Menu".

**View Result:** Reclassification, "Save to Gallery", "view Image details" and "Return to Main Menu" options are available in the view results.

**Settings:** "Return to Main Menu" option that allows managing accounts and changing settings.

**Image Details:** Displays categorization information along with a "Return to Gallery" option.

A screenshot of a computer

Description automatically generated

**Component 2: Screen Design (Web View)**

Screens to design are Upload Image, View Results and Settings screens.

1. **Upload Image Screen**  
   - whenever we click on the generate results, the page navigates to view results screen.

A computer screen shot of a computer

Description automatically generated

1. **View Results Screen**

A computer screen shot of a computer

Description automatically generated

Whenever we click on the expand button at the particular image, it gives the classification of that particular image which is shown below.

A computer screen shot of a computer screen

Description automatically generated

A computer screen shot of a computer

Description automatically generated

1. **Settings screen**

* Here we can manage our account and adjusts the image preferences.

A computer screen shot of a computer

Description automatically generated

**Component 2: Screen Design (Mobile View)**

Screens to design are Upload Image, View Results and Settings screens.

1. **Upload Image Screen**  
   - whenever we click on the generate results, the page navigates to view results screen.

A screen shot of a phone

Description automatically generated

1. **View Results Screen**

A screen shot of a cell phone

Description automatically generated

Whenever we click on the expand button at the particular image, it gives the classification of that particular image which is shown below.

A screen shot of a phone

Description automatically generated

1. **Settings screen**

* Here we can manage our account and adjust the image preferences.

A screen shot of a cell phone

Description automatically generated

1. **BUDGET**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Resource Estimates** | | | | | | |
| **Role** | **Comments** | **Weeks** | **Hours per week** | **Total Hours** | **Rate per hour** | **Total cost per role** |
| Project Manager | Manage and document project during design and development | 11 | 7 | 80 | $80/hour | $6400.00 |
| Software Developers | Develop backend and frontend during development phase of the project | 28 | 8 | 200 | $70/hour | $28000.00 |
| UI/UX Designer | Design screens and navigation of the application | 9 | 7 | 60 | $40/hour | $2400.00 |
| Quality Assurance Analyst | Tests the project | 11 | 7 | 80 | $60/hour | $4800.00 |
| Business Analyst | handle business perspectives of the system | 4 | 7 | 30 | $30/hour | $900.00 |
| Training and Support Specialist | Training stakeholders how the system works | 12 | 8 | 85 | $55/hour | $4675 |
| Technical Writer | Documenting all the project tasks | 6 | 6 | 40 | $50/hour | $2000 |
| **Subtotal** |  |  |  | 575 |  | **$**49175 |

**HARDWARE EXPENSES**

|  |  |
| --- | --- |
| **Hardware Estimates** | |
| **Item** | **Estimated Amount ($)** |
| 1 laptop | 1,000 |
| Networking equipment | 800 |
| AWS Cloud Services | 5000 |
| Ongoing costs | 500 |
| **Total** | **$7,300** |

**SOFTWARE EXPENSES**

|  |  |
| --- | --- |
| **Software Estimates** | |
| **Item** | **Estimated Amount ($)** |
| Windows Operating systems for laptop | $600.00 |
| MySQL Database Management System | $0.00 |
| Web Services | $1500.00 |
| Python for backend | $1000.00 |
| Ongoing costs | $1000.00 |
| **Total** | **$4100.00** |

**APPROVED PROJECT COST ESTIMATES**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Total Estimates** | | | | | |
|  | **Department** | **Resource** | **Expense** | **Estimated Amount** |  |
| **Development Cost** | Human Resource | Resource | Team and Support | **$49,175.00** | **$60,575.00** |
| IT/Procurement | Hardware | All hardware expenses | **$7,300.00** |
| Software | All software Cost | **$4,100.00** |
| **Operation Cost** |
| Human Resource | Marketing | Ongoing Training | **$10,000.00** | **$13,000.00** |
| Various | Others | All other ongoing cost | **$3,000.00** |
|  | **Total** | | |  | **$73,575** |

Total Estimated Budget: $73,575

1. **Hardware Expenses:**

Expenses: $7,300.00

Explanation: This includes the initial cost of purchasing hardware resources for the project, such as laptops, networking devices, and other necessary hardware. The price is calculated using vendor quotes and market research to guarantee that the required hardware is purchased within the allotted budget.

1. **Software Expenses:**

Expenses: $4,100.00

Explanation: Operating systems, development tools, and software licenses are among the crucial software tools needed for the project; these costs are included in the software expenses together with the licensing fees and procurement prices. Based on industry norms and software needs, these expenses are predicted.

1. **Marketing:**

Expenses: 10,000.00

Explanation: This amount is set aside for marketing initiatives meant to raise awareness of the project and connect with more people. The project expenses encompass advertising, branding, and promotional initiatives with the goal of raising awareness and generating interest among potential users and stakeholders.

1. **Other Ongoing Costs:**

Expenses: $ 3,000.00

Explanation: This category includes all the ongoing costs necessary to keep the project operating and maintained. It could cover maintenance, support service, and other extra charges that are necessary for the long-term viability and efficiency of the project.

**Cost Benefit Analysis (CBA):**

**Benefits:**

|  |  |
| --- | --- |
| **Item** | **Benefits ($)** |
| More Competitive Advantage | **16,000** |
| Wider Range of Customers | **33,000** |
| Inventory Management | **13,000** |
| Reduced Stockouts | **17,000** |
| **Total Benefits** | **79,000** |

These benefits are estimated based on the expected benefits and efficiency that the project was bring to the relevant departments and operations.

**Net Benefit = Total Estimated Benefits- Total Estimated Costs**

Net Benefit = 79,000- 73,575 **= $5,425**

**Return on Investment (ROI)**

ROI = (Net Benefit/ Total Costs) x 100%

ROI = ($5,425/ $73,575) x 100%

**ROI** ≈ **7.37 %**

**PAYBACK ANALYSIS**

Initial Investment= $73,575.00

Annual Net Benefit = $5,425.00

Calculate Payback Period:

Payback Period = Initial Investment/ Annual Cash Flow

Payback Period = $73,575/ $5,425 ≈ 13.56 years

**PRESENT VALUE CALCULATIONS**

**Determine Discount Rate:**

Hypothetical Discount Rate (Assuming 4%): 4%

**Present Value of Cash Flows:**

**Present Value (PV) = Cash Flow/ (1 + Discount Rate) ^n**

* PV Year 1 = $5,420 / (1 + 0.04) ^1 ≈ $5,211.5
* PV Year 2 = $5,420 / (1+0.04) ^2 ≈ $5,011.1
* PV Year 3 = $5,420/ (1+0.04) ^3 ≈ $4,818.36