Final Year Project

[ICE AGE]

Project Advisor:

[Dr Nadeem Majeed]

Name:

Qurban Hanif **[BSEF20M534]**

Usman Ghias **[BSEF20M540]**

Kainat Bashir **[BITF20A503]**

Alisha Sultan **[BITF20A539]**

Usman Zahid **[BITF20A545]**

Submission Date:

[September 28, 2023]

Project Proposal ICE AGE

TABLE OF CONTENTS

Catalog

1.1 Project Title

ICE AGE

1.2 Project Overview Statement

In a world where climate change looms as an urgent global crisis, our project, Deep learning-based framework for monitoring of debris-covered glacier from remotely sense images emerges as a pivotal endeavor. Escalating pace of glacier melting brings forth dire consequences, including the rise in sea levels and heightened risk of Glacier Lake Outburst Floods (GLOF). In an era where environmental preservation stands as a top priority, our project seeks to revolutionize the monitoring and classification of debris-covered glaciers, offering both precision and accessibility through our website application. Focused on debris-covered glaciers, we employ advanced convolutional neural networks (CNNs) to enhance monitoring precision. This dual approach enhances classification accuracy and ensures robust feature extraction.

By generating missing data classes with generative adversarial networks (GANs) and optimizing our architecture, we pave the way for large-scale glacier mapping using freely available dataset. Additionally, our project extends to the development of a user-friendly web application, offering accessibility for on-the-go glacier information access. Furthermore, we commit to assessing our deep learning-based approach's performance by conducting a comprehensive comparative analysis with traditional machine learning methods, including Support Vector Machine (SVM), Random Forest (RF). In the ongoing battle against climate change, our project assumes a vital role, serving as a potent tool for comprehending glacier dynamics, promoting responsible environmental stewardship, and ensuring that critical information is readily available to researchers and stakeholders worldwide.

Project Title: Automated Deep Learning-based Framework for Monitoring of Debris-Covered Glaciers from Remotely Sensed Images

Group Leader: Muhammad Usman Ghias (bsef20m540@pucit.edu.pk)

Project Members:

Name	Roll No.	Email Address	Signature
Usman Ghias	BSEF20M540	bsef20m540@pucit.edu.pk	*Usman Ghias*
Qurban Hanif	BSEF20M534	bsef20m534@pucit.edu.pk	*Qurban*
Kainat Bashir	BSITF20A503	Bitf20m503@pucit.edu.pk	*Kainat*
Alisha Sultan	BSITF20A539	Bitf20a539@pucit.edu.pk	*Alisha*
Usman Zahid	BSITF20A545	Bitf20a545@pucit.edu.pk	*Usman Zahid*

Project Goal: Primary goal of Auto Pilot is to create web-based application for show case our deep learning-based framework for monitoring of debris covered glaciers. This platform will empower researchers to classify their images in a professional, online environment, bridging the gap between academic exploration and real-world applications in glacier research. Our model will be able to identify glacier debris automatically by just uploading image which is done manually till now in space and science. Presenting

This project will focus on several key strategies:

- Develop a versatile and adaptable deep learning framework capable of seamlessly integrating new functionalities and adjustments.
- Engineer a system with inherent ease of maintenance and scalability, achieved through modular code and reusability.
- Prioritize codebase testability to streamline debugging efforts and elevate overall system quality.
- Simplify database interactions and enhance codebase decoupling through the implementation of Object-Relational Mapping (ORM).
- Create **consistent**, easy-to-use, and reliable interfaces for interacting with the system for internal and external components and users.
- Properly working APIs for Web Application

Objectives:

Sr.#	
1	**Improve code maintainability and reusability**: Utilize object-oriented design patterns to create modular and reusable code that is easily maintainable, making it easier to evolve the system over time.
2	**Develop a robust and extensible object-oriented architecture:** Apply the principles of object-oriented analysis and design to create a flexible and scalable system that can accommodate new features and changes in the future.
3	**Enhance code testability:** Implement techniques such as dependency injection and mock objects to increase the ease of testing the codebase and reducing the effort required for bug fixing.
4	**Responsive Design:** People prefer to visit from mobile application instead from web application, so our motive is to make website responsive for debris-covered glaciers.
5	**Deep Learning Framework:** A lot of deep learning models are available online; we will prefer to use open source and freely available resource for our deep learning frameworks.
6	**Implement Object-Relational Mapping (ORM):** Use an ORM framework like Hibernate or **Entity** Framework, to ease the interaction with the relational database and make the codebase more decoupled from the database
7	**Improve scalability and performance:** Design the system to handle increased usage and take measures to improve the performance and response times from our AI model.

Project Success criteria:

The success of our project will be determined by several critical factors. First, achieving a detection accuracy rate of over **90%** through deep learning models will be a primary indicator. And other measurable criteria such as a **flexible**, **extensible** architecture of web application; **maintainable**, understandable code-base integration; **efficient**, **effective** testing of deep learning model, web app; successful implementation of ORM and **DDD** (Domain Driven Design); Improved scalability and performance, and user-friendly interfaces for uploading images. Finally, optimizing efficiency, reducing research time, and demonstrating cost-effectiveness in project execution and environmental impact mitigation will round out the success criteria.

Assumptions

- Internet connectivity
- Compatible device and browser or mobile operating system
- Access to reliable upload satellite imagery and glacier raster data.
- User adoption and engagement.
- Basic understanding of how to use the internet and navigate a website app.
- Updated version of the Application/Website
- Have little bit background knowledge of GIS.
- Must be able to differentiate colors.

Risks and Obstacles

Risks and obstacles when accessing the ICE AGE **application** or **website**:

- Copyright and licensing issues with data sources.
- Technical difficulties
- Compatibility issues
- Network connectivity
- Data privacy concerns.
- Technical complexities in Algorithm understanding.
- Confusing or difficult to understand raster and vector data.
- Unreliable or slow customer service.

Organization Address: Punjab University College of Information Technology (**PUCIT**), University of The Punjab, Allama Iqbal Campus, Shahrah-e-Quaid-e-Azam (The Mall), 54000, **Lahore**, Pakistan

Type of project: □ *Research* □ *Development*

Target End users:

1. Climate Researchers and Scientists
2. Environmental Organizations
3. Government Agencies
4. Glacier Monitoring Agencies
5. Educational Institutions

Development Technology: □ Object Oriented Approach □ Optical Remote Sensing Appr.

Platform: □ Web Application

Suggested Project Supervisor: Dr. Nadeem Majeed

Approved By: Dr. Nadeem Majeed

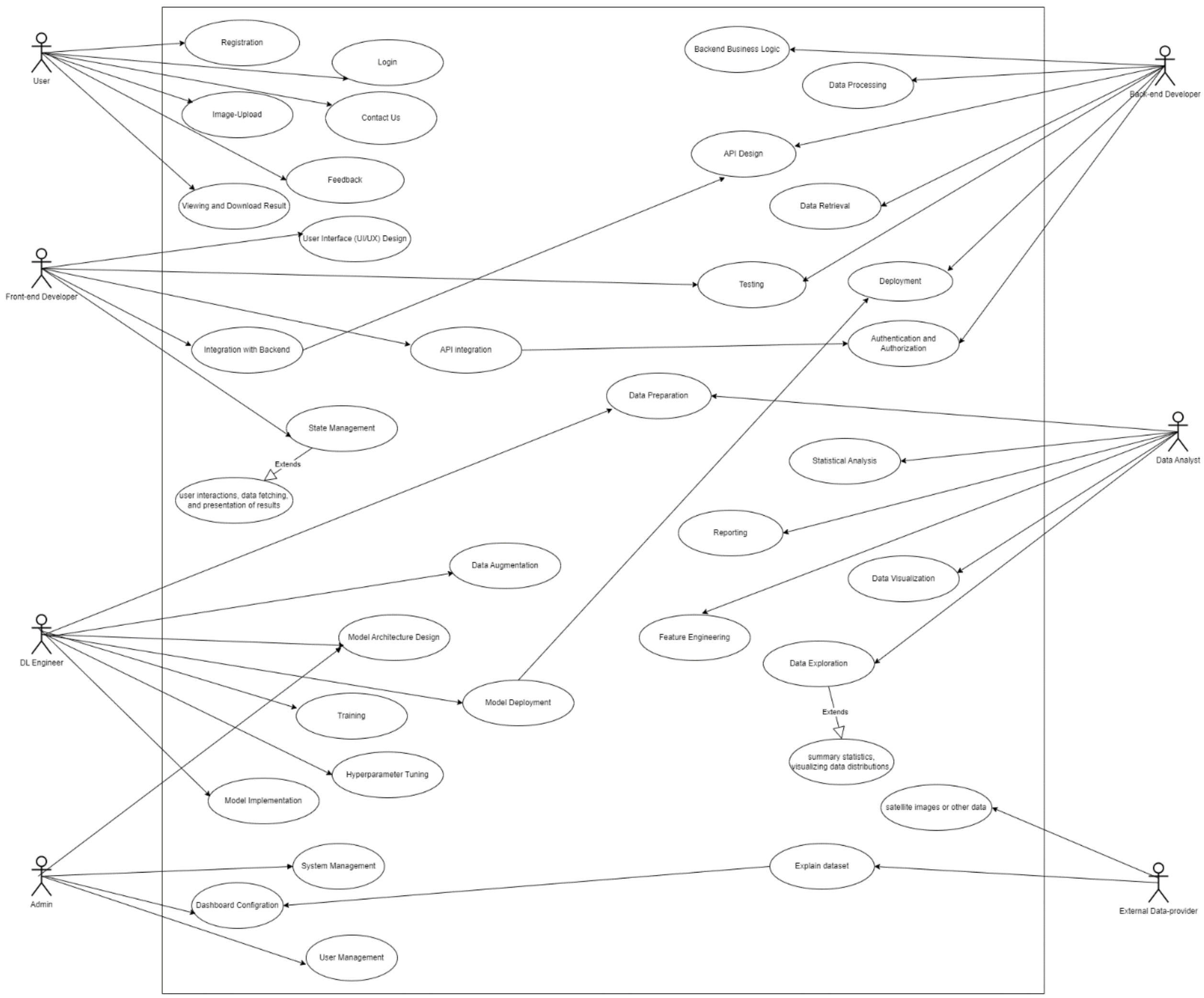
Date: September 26, 2023

1.3 High-level system components

- Authentication Module
- Deep Learning Module
- Uploading Raster
- Feedback

❖ Contact Us
❖ Profile
❖ Dashboard

1.4 High Level Use Case Diagram



1.5 List of optional functional units

- Chatbot Integration
- Feedback module
- Social Media Plugin
- Mobile Application

1.6 Exclusions

There will be no exclusions in our project.

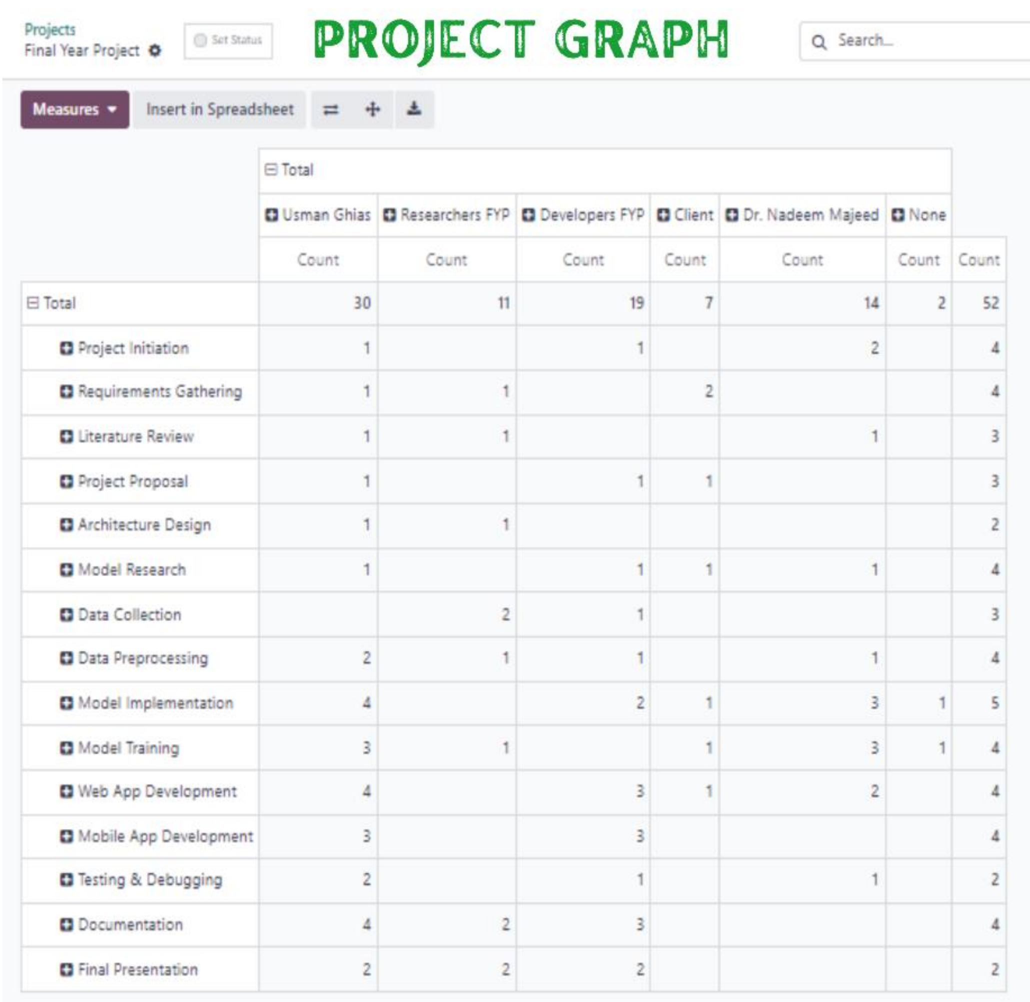
1.7 Application Architecture

We will follow 3-tier Architecture.

- React JS (Front End)
- Django Rest (Back End)
- MongoDB

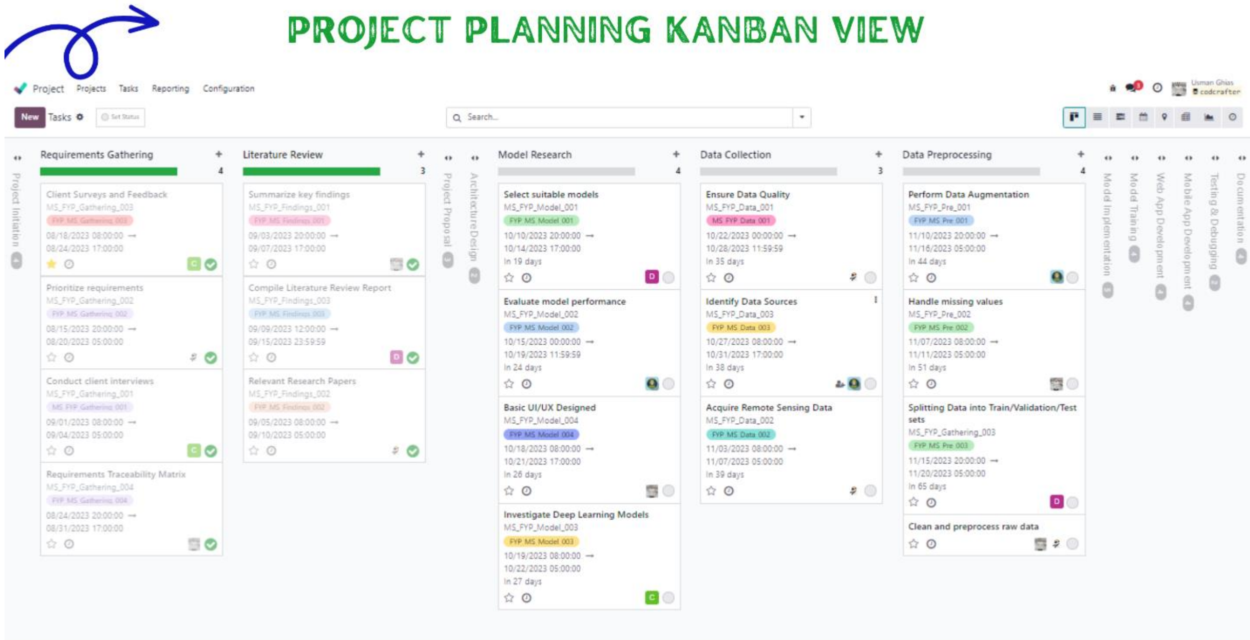
1.9 Gantt chart

Timeline

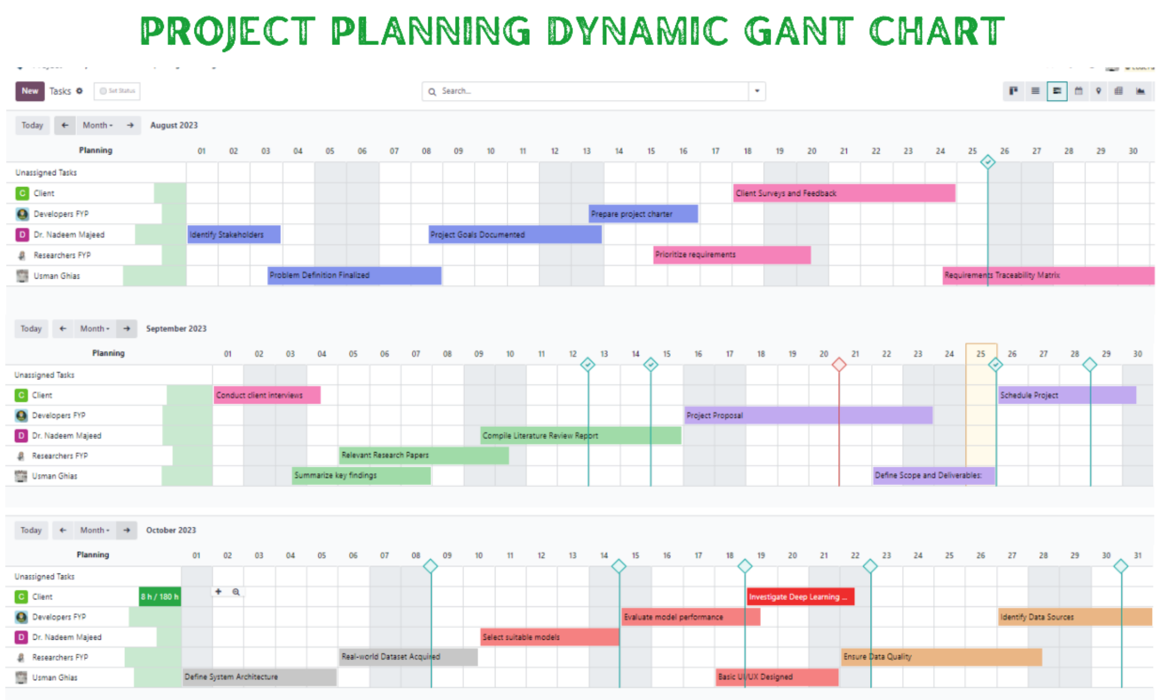


	Total						
	Usman Ghias	Researchers FYP	Developers FYP	Client	Dr. Nadeem Majeed	None	
	Count	Count	Count	Count	Count	Count	Count
Total	30	11	19	7	14	2	52
Project Initiation	1		1		2		4
Requirements Gathering	1	1		2			4
Literature Review	1	1			1		3
Project Proposal	1		1	1			3
Architecture Design	1	1					2
Model Research	1		1	1	1		4
Data Collection		2	1				3
Data Preprocessing	2	1	1		1		4
Model Implementation	4		2	1	3	1	5
Model Training	3	1		1	3	1	4
Web App Development	4		3	1	2		4
Mobile App Development	3		3				4
Testing & Debugging	2		1		1		2
Documentation	4	2	3				4
Final Presentation	2	2	2				2

Project Kanban View



Gant View



1.10 Hardware and Software Specification

Hardware: (Minimum)

- Intel Core i5-6500 or with 2.5 GHz
- 8 GB of DDR4 RAM
- 256 GB SSD
- RTX 1080 Ti Graphical Processing Unit (GPU)

Hardware: (Recommended)

- Intel Core i7-9700 processor with 3.00 GHz or more
- 16 GB of DDR4 RAM, and an NVIDIA GeForce GTX
- RTX 3070 Ti Nvidia graphical processing unit (GPU)

We will be using the recommended hardware for our training our model on deep learning frameworks that require significant GPU acceleration, you might consider adding one or more powerful GPUs, such as NVIDIA RTX series or NVIDIA A100 GPUs, to increase model training speed.

Operating System:

- Windows
- MacOS
- Linux

1.10.1 Network Connectivity:

Ensure you have a stable and high-speed internet connection, especially if your project involves downloading large geospatial datasets or working with cloud-based resources.

1.11 Tools and technologies used with reasoning

Project Tools

- **Visual Studio Code:** A popular code editor for developing frontend and backend applications, including React Native, Python, and Django.
- **Web GIS:** A generic term that encompasses various web-based Geographic Information System (GIS) tools and technologies, which you might use for web-based mapping and visualization.
- **Google Collab:** For collaborative coding, data analysis, and machine learning using Jupyter notebooks in the cloud.
- **Version Control:** Tools like Git and platforms like GitHub, GitLab or GitBucket can help manage code versions and collaborate with others.
- **Google Earth Engine:** For data acquisition, processing and analysis of remote sensing data related to glaciers.
- **Erdas Imagine:** A powerful remote sensing and image processing software for working with geospatial data.
- **Figma:** For creating frontend design of our web application.

- **ArcGIS**: A comprehensive GIS platform for geospatial data management, analysis, and visualization.
- **MongoDB:** MongoDB's flexible document-based model allows us to store data in JSON-like documents, making it easy to represent geospatial information such as coordinates, timestamps, and associated metadata, making it suitable for storing and managing geospatial data associated with debris-covered glaciers.

Technology

- **Responsive Web Design:** Ensuring that frontend is responsive to various screen sizes and devices. We will be using HTML, CSS, JavaScript, and Bootstrap for responsive design.
- **Django Rest Framework:** Essential component is instrumental in constructing a RESTful API, which serves as the backbone for data delivery and seamless interaction with the project's frontend.
- **TensorFlow/Pytorch:** Empower the development of complex neural network architectures, enabling the system to analyze remotely sensed imagery, classify debris-covered glaciers and provide valuable insights into environmental changes.
- **Rasterio:** Rasterio plays a pivotal role in geospatial data processing. It acts as a bridge between the project's deep learning models and the geospatial data sources, facilitating the extraction of meaningful information from remotely sensed imagery.