Fauna Finder

Bikram Karki (Student Number: 7693230)
Birendra Chaulagain (Student Number: 7735492)
Prabhat Mandal (Student Number: 7650450)
Resha Adhikari (Student Number: 7077051)
Tsering Yangchen Lama (Student Number: 7595499)

October 13, 2023

Abstract

Tourism in biodiverse regions like Australia often faces a dual challenge: tourists inadvertently harming local wildlife due to a lack of awareness, and missing out on the rich experience of understanding native fauna. Current solutions, such as generic wildlife guides or tours, might not provide real-time, specific information or foster interactive learning, thereby leading to unintentional harassment of endangered species. Addressing these gaps, this project proposes an innovative mobile application aimed at enhancing the wildlife tourism experience in Australia. The application will utilize an image classification model, allowing users to capture pictures of fauna for immediate recognition. Additionally, a domain-trained Large Language Model (LLM) will power a conversational AI agent, enabling tourists to inquire and learn about the identified species in real-time. Beyond species identification, the application will offer insights into habitat, conservation status, and other relevant information. An analytic dashboard will monitor application performance and user engagement, driving iterative improvements. With its multifaceted approach, this project not only aims to enrich the tourist experience but also aspires to promote conservation awareness, reduce unintentional wildlife harassment and disturbances, and foster community engagement in ecotourism. Through this initiative, the project team seeks to meld machine learning, computer vision, and software development, producing a sustainable and impactful solution for both tourists and the environment.

Contents

1 Introduction **3**
- 1.1 Aims . . . 3
- 1.2 Related Work . . . 4

2 Methodology **7**
- 2.1 Image classification . . . 7
- 2.2 Large Language Model(LLM) . . . 8
- 2.3 System Architecture: . . . 8
  - 2.3.1 Mobile App Development: . . . 8
  - 2.3.2 Backend . . . 9
  - 2.3.3 Web- based Analytic Dashboard: . . . 9
- 2.4 Proposed Solution and Technology Stack . . . 10
  - 2.4.1 Mobile App Development: . . . 10
  - 2.4.2 Image Classification: . . . 10
  - 2.4.3 Conversational AI: . . . 10
  - 2.4.4 Backend: . . . 10
  - 2.4.5 Web-based Dashboard: . . . 10
  - 2.4.6 Deployment, Collaboration, and Development Tools: . . . 10
  - 2.4.7 Project Management Tools: . . . 10
- 2.5 Project Challenges . . . 11
  - 2.5.1 Data Acquisition . . . 11
  - 2.5.2 Image Classification Challenges . . . 11
  - 2.5.3 Large Language Models (LLMs) Challenges . . . 11
  - 2.5.4 Integration of System Components . . . 11
  - 2.5.5 User Engagement and Adoption . . . 12
- 2.6 Project Team . . . 12
- 2.7 Plan and Time Line . . . 13
  - 2.7.1 Initiation Phase (8/2/2023 - 9/15/2023) . . . 13
  - 2.7.2 Planning Phase (9/15/2023 - 10/13/2023) . . . 13
  - 2.7.3 Development Phase (1/3/2024 - 5/27/2024) . . . 14
  - 2.7.4 Testing Phase (3/15/2024 - 5/28/2024) . . . 15
  - 2.7.5 Deployment Phase (4/4/2024 - 5/28/2024) . . . 15

3 Outcomes **18**

1 Introduction

Tourism provides significant economic benefits to local communities but it can also threaten the ecosystem if the travelers are unaware about local wildlife. It is not only the matter of threat to the ecosystem but there also is the matter of intrigue in the local wildlife wherever we go. Wildlife is a very interesting aspect in tourism and garners quite some interest from tourists who often have many unanswered questions regarding the species, behaviours, conservation needs of species etc., when they encounter wildlife that is new to them. Answer to these questions not only allows tourists to learn more about the local wildlife but may also help them to not take actions that brings unwanted results and causes accidental harassment of the animals. Many local regions lack comprehensive wildlife identification resources tailored to meet the needs of ecotourists.

The recent advancement in computer vision and conversational AI presents new opportunities that can bridge the gap and enhance the wildlife experience. Image classification models can identify images with high accuracy. The Language model can provide conversational access to the expert knowledge about the wildlife. We aim to develop a project entailing development and deployment of classification model along with LLM on cloud and development of a mobile application for the end users that combines a aforementioned technologies to enrich ecotourism and promote conservation of species.

To be able to train the classification model that is built having a robust data-set is important. Obtaining a data-set of animal and bird images suited to our goal of identifying Australian wildlife poses an initial challenge. Publicly available data-sets are limited in regional specificity and availability of images as we need quite a few images to train our model. However, this challenge also presents an opportunity for us to create a custom data-set tailored to our needs.

Our application will simply allow users to take a picture of an unfamiliar animal or bird in Sydney region. An image classifier will identify the species and our application will gives the detailed information about the species to the user. If user wants to have more information they can ask the conversational AI agent questions to learn more. The AI agent provides information about the habitat, diet, status, threats, and unique traits for hundreds of species.

1.1 Aims

The main objective of this project is to develop an application that allows the tourists to easily identify the local wildlife species and get a detailed information about identified species through the chat interface of language model. In the application, we aim to develop an image classification model that can accurately identify the animal/bird species from pictures taken by users in the Sydney region and provide the detailed information to the users' questions about the local species through the conversational interface. We aim to provide information such as natural habitat of species, natural diet of the species, conservation status, specific traits of species to promote appreciation, understanding, and conservation of local wildlife among visitors through species identification and interactive learning. With this main aim in mind we will also have an entailed secondary aim of procuring a robust data-set of images of various species of animals and birds in Sydney region. It will be targeted that very concise and relevant information will be provided from the Conversational AI agent.

Apart from the benefits of the application to the users of it, the team aims to sharpen skills in machine learning and computer vision by building and optimizing the image classifier and its deployment in cloud server. With guidance of our supervisor, we look forward to expanding our knowledge on different framework, model and deployment of tools in our project.

1.2 Related Work

In recent advancements in Convolutional Neural Networks (ConvNets), the conventional methodologies of model scaling have been re-evaluated and restructured. Tan and Le, in their pioneering work [1], have highlighted the prevalent practice of developing ConvNets with a fixed resource allocation and subsequently scaling them for enhanced accuracy when resources are augmented. Through a systematic exploration of model scaling, they accentuated the importance of achieving a balance in network depth, width, and resolution, suggesting that this balance can significantly elevate performance. Introducing a novel ”compound coefficient” for scaling, the authors have demonstrated its efficacy on renowned models such as MobileNets and ResNet, with tangible improvements in performance .

Building upon their foundational research, Tan and Le utilized neural architecture search (NAS) to conceive a new baseline network. The culmination of this endeavor was the development of ”EfficientNets” [1], a suite of models that have set new benchmarks in terms of accuracy and efficiency in the realm of ConvNets. Among these, the EfficientNet-B7 model stands out, achieving state-of-the-art results, thus making a significant mark in the field of deep learning and model scaling.

Similarly, He et al. introduced a paradigm shift in the training of deep neural networks through their groundbreaking work on deep residual learning [2]. Recognizing the challenges inherent in training deeper networks, the authors presented the residual learning framework. This framework reformulates the network layers to learn residual functions relative to their inputs, thereby simplifying the optimization process. A standout concept introduced in the paper is the use of ”shortcut connections” in feedforward neural networks. These connections, as posited by the authors, facilitate the optimization of residual mapping, making it more manageable than the traditional unreferenced mapping. In their experiments, the residual networks, also known as ResNets, not only showed ease of optimization but also reaped the benefits of increased depth, marking a significant stride in the realm of deep learning [2].

In addition to the advancements in ConvNets and deep residual learning, another significant breakthrough in the realm of deep learning has been made by Howard et al. with their introduction of MobileNets [3]. Designed specifically for mobile and embedded vision applications, MobileNets come as a response to the ever-growing demand for computational efficiency on constrained platforms. The authors present a streamlined architecture which harnesses the power of depth-wise separable convolutions, allowing for the construction of lightweight yet efficacious deep neural networks. These convolutions effectively decouple the interaction between output channels and kernel size, leading to substantial computational savings. Accompanying this architectural innovation, the paper introduces two pivotal global hyperparameters. These parameters, by offering a balance between latency and accuracy, provide model builders with the flexibility to tailor models to specific application constraints, thus underlining the adaptability and applicability of MobileNets across diverse platforms [3].

Liu et al. proposed a real-time classification system for marine animal images based on MobileNetV2 and transfer learning [4]. The system uses an underwater robot equipped with an embedded device to collect marine animal images, which are then used to train a MobileNetV2 model based on convolutional neural network (CNN) using transfer learning. The trained model is then downloaded to the embedded device for real-time classification of marine animal images. The proposed solution aims to improve the accuracy of marine animal image classification, which can be useful in monitoring the growth of marine animals and water conditions in marine aquaculture and other related fields.

The animal classification system proposed by Shalika et al. [5] offers several benefits for animal researchers and wildlife photographers. The system can automatically detect and recognize wild animals, which can save time and effort for animal researchers and wildlife photographers. The system uses a high-end autofocus webcam to capture images of animals, which can result in high-quality images with more details and clarity. The system uses machine learning techniques to classify animals, which can improve the accuracy of animal recognition and classification. where as project only focuses on three kinds of animals, which may not be sufficient for some animal researchers and wildlife photographers who need to classify a wider range of animals. Overall, the animal classification system can help animal researchers and wildlife photographers overcome challenges in their work and improve the efficiency and accuracy of animal recognition and classification.

In a related work, Cardoso et al. have proposed solution to detect illegal pangolin trafficking on e-commerce platform via deep learning models [6]. It urges that not only website but also social medias have been sites for both illegal and legal wildlife trades. They used publicly available online images of pangolin commonly traded parts to identify pangolin species using predefined deep learning classification and object detection models. The challenges present here is the model limitation to clearly identify the images which are barely visible, or contain dark colours and emphasized shadows.

Mugambi et al. present an annotated camera trap data-set [7], DSAIL-Porini, consisting of images of wildlife species captured in a conservancy in Nyeri, Kenya. The data-set is intended to provide a resource for researchers and conservationists interested in studying and protecting the diverse wildlife of Kenya. The DSAIL-Porini dataset was manually annotated with the following fields: Filename, Species, and Count. The Filename field contains the exact date and time an image was taken and saved. The Species field identifies the species present in each image, and the Count field indicates the number of individuals per species in a particular image. The order of labeling in the Species and Count fields follows the order of the species from foreground to background and left to right [7]. The DSAIL-Porini dataset consists of 8524 images from four camera traps deployed in the Dedan Kimathi University Wildlife Conservancy in Kenya. The dataset includes six different species of mammals found in the conservancy, namely: impalas, Burchell's zebras, common warthogs, bushbucks, defassa waterbuck, and the Sykes' monkey.

The number of images containing each species collected in [7] are as follows (see Table 1):

The solution proposed by Battu et al [6], provides a way to accurately identify and classify animal species from camera-trap photos captured in noisy, densely populated areas. This is a significant problem in animal identification research, as there haven't been many effective methods introduced, especially in the area of predator species. The authors' method offers a reliable learning strategy for dealing with noisy labels and separating the training data into groups with various properties using k-means clustering [8]. Identifying and classifying animals from camera-trap photos in noisy, densely populated areas can be challenging due to several factors. For example, the images may have varying dimensions, locations, and lighting conditions, making it difficult to extract useful features. Addition-

Table 1: Number of Images by Species

Species	Number of Images
Bushbuck	313
Impala	5,649
Sykes' monkey	52
Defassa waterbuck	606
Common warthog	1,565
Burchell's zebra	598

ally, the presence of multiple species in the same image can lead to confusion and misclassification. Finally, noisy labels can also be a challenge, as they can lead to incorrect training of the deep neural networks used for classification.

We know that there can be a problem of fine grained bird species classification. With many different species of birds or even animals looking very similar the classification models struggles to give accurate results in such case as opposed to more coarser classification problem such as being able to tell apart a seagull from raven. With aim to produce robust classification task for fine-grained bird species, Ge et al. have brought forth a novel idea which is based upon hierarchical subset learning [9]. Hereby they have grouped together subsets of similar species into classes forming a similarity tree. Now upon doing so a local classifier with power to distinguish between these similar species is made to learn from each subset. The authors mention that employing their method they were able to see significant rise in accuracy on a challenging data set. In essence they are proposing a hierarchical classification system. While this method may give better results it remains to be explored how the results will be diminished with input images that are of good quality.

Navigating the landscape of NLP, Radford et al. present a paradigm-shifting perspective on the capabilities of unsupervised language models [10]. Their groundbreaking work underscores the potential of such models, particularly GPT-2, in mastering a spectrum of NLP tasks devoid of explicit task-specific supervision. Drawing from the vastness of the WebText dataset, the authors showcased that when the model is furnished with a document and its related queries, it can generate answers that not only achieve competitive scores on benchmarks like the CoQA dataset but also rival extensively trained baseline systems. This feat is achieved without the crutch of a plethora of training examples, emphasizing the inherent adaptability and learning capabilities of the model. A salient point of their research lies in the model's capacity, with indications that augmenting this capacity can lead to log-linear performance enhancements across diverse tasks. The introduction of GPT-2, a behemoth with 1.5 billion parameters, stands testament to their research's depth and the model's prowess [10].

Moving farther into the domain of conversational AI, Bocklisch et al. highlight the capabilities of the Rasa platform, including the tools Rasa NLU and Rasa Core [11]. These tools, which are based on the notion of making machine learning accessible for dialogue management and language interpretation, are intended at developers who do not have a specific specialization in the field. The study emphasizes the importance of developing conversational software that may be easily integrated into daily activities without requiring extensive initial training data. This is consistent with a broader industry trend in which conversational systems, exemplified by Apple's Siri and Amazon's Alexa, have grown essential to human-computer connection [11].

Recently, the team from Meta AI unveiled LLaMA, a suite of models characterized by parameters ranging from a mere 7B to an impressive 65B [12]. A noteworthy aspect of this work is the extensive training on trillions of tokens, solidifying the assertion that top-tier models can be crafted utilizing exclusively public datasets. The authors avoid using proprietary datasets, emphasizing the approach's accessibility and replicability. LLaMA-13B stands out, outperforming the massive GPT-3 model [13] with 175B parameters over various benchmarks. This trend continues with the LLaMA-65B, which competes admirably against top models like the Chinchilla-70B [14] and the PaLM-540B [15]. LLaMA, which is based on the transformational architecture of transformers, pushes the limits of what foundational language models can achieve, particularly when trained on large, publically available corpora [12]

2 Methodology

Our process first starts with the development of image classification model. For this we also need to perform image acquisition task as readily available comprehensive data set does not seem to be available. After that we will train the classification model to be able to recognise images. End users will use mobile application. With this application the users shall be able to click the pictures of the wild animals and birds that they encounter. The application will send over the image to the cloud server where the classification model shall be deployed. The result of the classification shall be sent back to the end user mobile application. Next the application shall provide the user with the prompts to ask questions regarding the result. If the user proceeds to ask any question then the application will respond to the question. For the conversational text generation we will be relying on Large Language Model which too will be deployed on cloud server.

2.1 Image classification

- **Data Acquisition:** As mentioned earlier, Obtaining a comprehensive dataset of images for Australian Wildlife species is crucial for training and evaluation of our model. To construct our own dataset, we will first compile the target list of Australian animals and birds to collect images for, database like Atlas of Living Australia can be leveraged to generate list of species found.

  With our target list in hand, we will write a script to crawl online images sources like Google Images, Wikimedia, etc. and download the creative common licensed photos of the listed species. This will generate the gathering of the candidate images for the dataset, we will manually review each of the images and remove the ones that are not useful for us. The images that can be used in the dataset will be annotated by mapping each picture to the corresponding species name from our target list. Applying labels to the images will produce a tailored dataset for our use.
- **Model Planning:** We plan to do the experiment with Mobilenet, EfficientNet(B0 - B7), and ResNet models of the convolutional neural network for image classification.
  - **Mobilenet:** MobileNet is efficient model for mobile and embedded vision applications, and it is based on the streamlined architecture that uses depth wise separable convolutions to build the light weight deep neural networks. And, from our research we have found this to be one of the suitable option for our project as it provides a good balance of accuracy and latency [3].

- **EfficientNet:** EfficientNet leverages a compound scaling method that uniformly scales the network depth, width, and resolution to yield improved performance with better parameters compared with other conventional CNN's [1]. From the research, we have found that EfficientNet-B0 - EfficientNet-B7 provide a strong baseline that can be scaled up as per the need and are well suited for deployment on our app. Efficiency and accuracy of the EfficientNet model make it promising candidate for our application goal, and it is worth exploring further through prototyping and comparative analysis during the model selection.
- **ResNet:** ResNet model is another promising candidate for our Wildlife classifier. These models are highly effective for image classification because of their deep architecture, improvement on their gradient flow through the skip connections. Pre-trained ResNet models are also widely available, so we can leverage transfer learning rather than training huge models from scratch. The extensive capabilities and adaptation through fine tuning make it well suited for our classification model.

2.2 Large Language Model(LLM)

As we plan to make the application and user experience more interactive with the feature of being able to make a conversation on the basis of the results obtained from the image classification task, we will by relying on Large Language Model(LLM). There are various pretrained LLMs and convertational AI's such as GPT-2 [10], Rasa [11], and LLaMA [12]. There may be different versions of pretrained models available for the same LLM. Some of these state of the art models are not available freely but there are also free pretrained models available. With these available models we shall fine tune the models with the corpus of domain specific documents and see its performance. We can also compare multiple models to see how well they seem to provide natural conversation. The chosen and prepared model will be deployed on the cloud server and shall be in communication with the end user mobile application through APIs.

2.3 System Architecture:

The system architecture for Fauna Finder wildlife recognition project consists of both front-end and back-end components, as well as a database to store and manage data. The architecture is designed to provide a seamless and engaging experience for tourists visiting Australia, allowing them to identify and learn about local wildlife while promoting environmental awareness.

Given technologies and frameworks play a crucial role in creating a robust and feature-rich application that enhances the tourism experience and promotes environmental awareness in Australia.

2.3.1 Mobile App Development:

- *Framework:* We are going to use the React Native framework, which allows us to develop a cross-platform mobile application with JavaScript , which enables us to create a single codebase that works on both Android and iOS devices. It helps to maximize efficiency and reducing development time [16]. React Native offers a wide range of UI components and libraries for a responsive and intuitive user interface.

- *State Management:* we will employ Redux as state management libraries to manage the state of our mobile application efficiently. This is crucial for maintaining the consistency of data and user interactions across different components of the app [17]. With Redux, we can ensure that the wildlife recognition results, and user interactions are synchronized seamlessly.
- *Navigation:* For seamless navigation within the application, we will utilize either React Navigation or React Native Navigation, depending on the specific requirements of the project. These navigation libraries enable us to implement stack-based navigation, tab navigation, and drawer navigation, ensuring that users can effortlessly move between various sections of the application [18]. This is particularly important for enabling tourists to explore different aspects of Australian wildlife.

2.3.2 Backend

- *Framework:* For the project's backend, we will employ the Node.js runtime environment in combination with the Express.js framework. This choice ensures that our backend is highly efficient, scalable, and capable of handling concurrent requests [19]. Express.js simplifies the creation of RESTful APIs and routes, making it an ideal choice for our wildlife recognition application.
- *Database:* To store and manage data related to recognized wildlife species, habitat, and user interactions, we will use the PostgreSQL database system. PostgreSQL provides robust data handling and supports complex queries, which are essential for our application's functionalities.
- *Containerization and Deployment:* We will containerize our backend application using Docker and Docker Compose, allowing for easy deployment and scaling. The application will be hosted on a DigitalOcean Droplet, ensuring reliable and scalable cloud-based hosting for our backend services. This setup guarantees high availability and performance for our mobile application users.

2.3.3 Web- based Analytic Dashboard:

- *Frontend Framework:* For the web-based dashboard, we will utilize the React framework. React provides a fast and interactive user interface, making it an excellent choice for displaying information about wildlife species and their conservation status. It also allows for the creation of reusable UI components.
- *UI Library:* To ensure a consistent and visually appealing design for our dashboard, we will integrate the Material-UI library. Material-UI offers a wide range of predesigned components and a responsive design system, making it easier to create an aesthetically pleasing and user-friendly dashboard [20].
- *Data Visualization:* For data visualization, we have the option to choose between D3.js and Chart.js, depending on the specific visualization requirements. D3.js is a powerful library for creating custom, interactive data visualizations [21], while Chart.js offers a simpler approach for generating various types of charts and graphs. We will use these libraries to present wildlife-related data to users in an engaging and informative manner

2.4 Proposed Solution and Technology Stack

2.4.1 Mobile App Development:

- *Framework:* React
- *State Management:* Redux
- *Navigation:* ReactNavigation

2.4.2 Image Classification:

- *Framework:* TensorFlow
- *Model:* MobileNet, ResNet, EfficientNet
- *Containerization and Deployment:* Docker, Docker Compose, and DigitalOcean Droplet

2.4.3 Conversational AI:

- *Framework:* LLAMA, GPT, BERT, RASA
- *Containerization and Deployment:* Docker, Docker Compose, and DigitalOcean Droplet

2.4.4 Backend:

- *Framework:* Node.js with Express.js
- *Database System:* PostgreSQL
- *Containerization and Deployment:* Docker, Docker Compose, and DigitalOcean Droplet

2.4.5 Web-based Dashboard:

- *Frontend Framework:* React
- *UI Library:* Material-UI
- *Data Visualization:* D3.js or Chart.js
- *Containerization and Deployment:* Docker, Docker Compose, and DigitalOcean Droplet

2.4.6 Deployment, Collaboration, and Development Tools:

- *Version Control:* Git
- *Collaboration:* GitHub
- *Containerization & Orchestration:* Docker & Docker Compose
- *Hosting & VM:* DigitalOcean Droplet

2.4.7 Project Management Tools:

- *Task Management and Project Timeline:* Jira
- *Document Management and collablration:* Microsoft Office 365 and Overleaf

2.5 Project Challenges

2.5.1 Data Acquisition

- **Data Diversity:** Acquiring a diverse set of images that cover the vast biodiversity in Australia can be challenging.
- **Data Quality:** Ensuring the images are of high quality, with good resolution and clarity, is critical for model accuracy. Poor quality images can hinder model training.
- **Data Privacy:** While collecting images from public sources or individuals, there could be privacy concerns or copyright issues.
- **Annotation Accuracy:** Properly labeling and annotating the images is crucial. Mislabeling can lead to model training inaccuracies.

2.5.2 Image Classification Challenges

- **Diverse Species Variability:** The model might encounter species it hasn't been adequately trained on due to the vast biodiversity in Australia.
- **Visual Ambiguity:** Some closely-related species might have subtle differences that are hard to detect, leading to potential misclassifications.
- **Environmental Variables:** Lighting conditions, obstructions, or the angle of the photograph might affect classification accuracy.
- **Computational Capacity:** Training a high-performance image classification model requires significant computational resources. Limited capacity can impact the training time, the ability to fine-tune, or even the choice of models.

2.5.3 Large Language Models (LLMs) Challenges

- **Training and Fine-tuning:** LLMs, due to their size, require substantial computational resources for training and fine-tuning, which might pose challenges in terms of time and costs.
- **Domain Specificity:** Ensuring the LLM is adequately fine-tuned to provide accurate information about Australian fauna can be challenging.
- **Context Recognition:** The AI might struggle with understanding user context, leading to potential misinformation or generic responses.
- **Handling Complex Queries:** Users might pose questions that go beyond the AI's training, necessitating constant updates to the model.

2.5.4 Integration of System Components

- **Heterogeneous Platforms**: Integrating modules that might be developed on different platforms or in different programming languages can be complex.
- **Data Flow and Consistency**: Ensuring smooth data flow between the image classification model, AI agent (LLM), backend database, and mobile application while maintaining data consistency can be challenging.
- **Real-time Synchronization**: The analytics dashboard should reflect real-time data, requiring efficient synchronization mechanisms.

- **Scalability**: As the user base grows, ensuring that the integrated system can scale efficiently to accommodate increased demand will be crucial.
- **Error Handling and Recovery**: In an integrated system, an error in one component can propagate. Effective error handling and recovery mechanisms across components are essential.
- **Security and Privacy**: Integrating multiple components can introduce vulnerabilities, especially if data is transferred between components. Ensuring security and user privacy throughout the system is paramount.

2.5.5 User Engagement and Adoption

- **User Interface (UI) Challenges:** Designing an intuitive UI is critical. A non-intuitive design could deter users from fully utilizing the app's capabilities.
- **Awareness and Training:** Users might not be fully aware of all the application's features or how to use them effectively.

2.6 Project Team

- **Bikram Karki (Team Lead):** Experience in applying machine learning to build predictive models, developing mobile apps with React Native, and deploying scalable cloud services. Skilled in the end-to-end machine learning pipeline, big data analytics, Linux administration, Docker containerization, and project leadership across technical teams. Proficient in deploying and managing virtual machines, ensuring application availability, security, and seamless cross-platform experiences.

  As a team lead, Bikram Karki will look after the overall development and deployment of the project. He will apply his technical expertise for problem solving, code review, communication and adherence to deadlines. He will focus on image classification and conversational AI.
- **Birendra Chaulagain:** Foundational knowledge in data analytics, Machine learning, and computer vision algorithms. Intermediate knowledge in web-development, python, MySQL, javascript. Has experience in Project management and finance. Birendra Chaulagain will use his project management skills for project planning, team management, task management, quality assurance, communication, timeline management, reporting and closure and evaluation. He will also be working on image classification, conversational AI and documentation.
- **Prabhat Mandal:** Experienced in web-development, skilled in ASP.NET, PostgreSQL, MySQL, Bootstrap, javascript and basic web technologies. Has basic knowledge of React. Slight experience in native android development with Kotlin programming language. Skilled in design and implementation of database schema. Possessing knowledge in machine learning and computer vision algorithms along with python programming language. The primary focus of Prabhat will be data-set and data analytics life cycle processes. He will also be working on classification model and front end development.
- **Resha Adhikari:** Front-end developer with professional experience in Australia, specializing in ReactJS and React Native and is on the pursuit to become a full-stack developer by actively broadening skill in .NET. Has foundational knowledge

of data analytics, machine learning and computer vision, complemented by intermediate proficiency in Python. Resha's tasks will be primarily focused on mobile app development and web-based dashboard development. She will also work on image classification and conversational AI.

- **Tsering Yangchen Lama:** Experienced in full stack development including MongoDB, Express, React and JavaScript. Moderate knowledge in big data analytic life cycle. Familiar with machine learning algorithms and computer vision algorithms. Tsering will be focusing on backend and web-based analytic dashboard. She will also be working on data analytics lifecycle processes.

2.7 Plan and Time Line

The project encompasses multiple phases and milestones, meticulously organized to ensure efficient development and successful completion. The timeline presented below offers a structured overview of key project activities.

2.7.1 Initiation Phase (8/2/2023 - 9/15/2023)

During the Initiation Phase, the project was launched with a focus on assembling the project team, defining project scope and objectives, finding a project supervisor, gathering required resources, and initiating the first meeting with the project supervisor. Here are the key activities:

1. **Assemble Project Team (8/2/2023 - 8/8/2023):** Formation of the project team, involving the selection of skilled individuals to contribute to the project's success.
2. **Define Project Scope and Objectives (8/11/2023 - 8/18/2023):** In this phase, the project's scope and objectives were clearly defined, setting the foundation for subsequent project activities.
3. **Find Supervisor (8/21/2023 - 9/5/2023):** The identification and selection of a project supervisor, a crucial element in ensuring guidance and direction throughout the project.
4. **Gather Requirements and Resources (9/7/2023 - 9/13/2023):** This phase involved the comprehensive gathering of project requirements and essential resources necessary for the project's successful execution.
5. **Meet Up with Supervisor (9/15/2023):** An initial meeting with the project supervisor to discuss the project's progress and align on further steps.

The Initiation Phase represents the foundational steps taken to ensure that the project is set on the right path and equipped with the essential elements for its success.

2.7.2 Planning Phase (9/15/2023 - 10/13/2023)

The Planning Phase was dedicated to comprehensive planning and preparation for the upcoming Development Phase. Key activities during this phase included:

1. **Create a Detailed Project Plan (9/15/2023 - 9/20/2023):** Development of an intricate project plan, ensuring that all aspects of the project are meticulously outlined and structured.

2. **Define Project Milestones and Deliverables (9/21/2023 - 9/25/2023):** Specification of project milestones and deliverables, providing clear objectives to be achieved during the project's execution.
3. **Develop Strategic Plan for Software and Hardware (9/26/2023 - 9/27/2023):** The creation of a strategic plan to guide the development of both software and hardware components, ensuring alignment with project objectives.
4. **Identify and Secure Necessary Resources (9/29/2023 - 10/2/2023):** Ensuring that all required resources are identified, secured, and made ready for use during the upcoming phases of the project.
5. **Meeting with Supervisor (10/6/2023):** A pivotal meeting with the project supervisor, providing an opportunity to review the planning progress and align on future steps and strategies.

The Planning Phase served as the foundation for the subsequent Development Phase, ensuring that the project was well-prepared to proceed with its core activities. It emphasized meticulous planning, strategic alignment, and resource readiness to support the project's success.

2.7.3 Development Phase (1/3/2024 - 5/27/2024)

The Development Phase is a critical stage in the project's lifecycle, focusing on the creation, testing, and integration of various project components. This phase encompasses several intricate activities:

1. **Dataset Preparation (1/3/2024 - 1/10/2024):** During this stage, the team prepared the necessary datasets, ensuring data quality and suitability for the project's objectives. This involved data collection, cleansing, and transformation.
2. **Report Writing (1/11/2024 - 5/13/2024):** The Development Phase included an extensive report writing process, covering documentation of project progress, methodologies, findings, and results.
3. **Develop and Train Image Classification Model (1/11/2024 - 1/19/2024):** This activity involved the development and training of an image classification model, a fundamental component for the project's success.
4. **Develop and Train Conversational AI Agent (1/22/2024 - 1/30/2024):** The development and training of a conversational AI agent, integral for interacting with users, occurred during this period.
5. **Develop Backend (1/31/2024 - 2/27/2024):** Development of the project's backend system, supporting the integration of various components and functionalities.
6. **Integrate Image Classification Model and Conversational AI Agent with Backend (3/20/2024 - 3/26/2024):** The phase included the integration of the previously developed image classification model and conversational AI agent with the project's backend infrastructure.
7. **Develop Analytic Dashboard (3/26/2024 - 4/12/2024):** Creation of an analytic dashboard to provide users with insights and data visualization.
8. **Develop Mobile App (4/26/2024 - 5/24/2024):** The project team focused on the development of a mobile application, ensuring compatibility with multiple platforms.

9. **Meeting with Supervisor (5/27/2024):** Frequent meeting with the project supervisor for review, project alignment, and to ensure all components are on track for the upcoming Testing and Deployment Phases.

The Development Phase is a complex and dynamic stage, where various project components are developed, tested, and integrated to create a cohesive solution. It involves collaboration, technical expertise, and a rigorous development process to ensure the project's success.

2.7.4 Testing Phase (3/15/2024 - 5/28/2024)

The Testing Phase is a crucial stage where the project components are rigorously assessed and validated to ensure quality and functionality. This phase involves the following activities:

1. **Integration Testing (5/28/2024):** The project team conducted integration testing to ensure that all components work seamlessly together. This process involved checking data flows and interactions between different modules.
2. **Meeting with Supervisor (3/15/2024):** An initial meeting with the project supervisor for guidance and to outline the testing approach and objectives.
3. **Ensuring the Conversational AI Agent is Working Effectively (3/18/2024):** Specific attention was given to verifying the effectiveness of the conversational AI agent in interacting with users and providing meaningful responses.
4. **Perform User Acceptance Testing (3/19/2024):** User acceptance testing was conducted to evaluate the system from an end-user perspective, ensuring that it meets user requirements and expectations.
5. **Meeting with Supervisor (3/29/2024):** A follow-up meeting with the project supervisor to review the testing progress and address any concerns or challenges.
6. **Address Any Bugs or Issues (4/1/2024):** The project team dedicated this day to addressing and resolving any identified bugs or issues from previous testing phases.

2.7.5 Deployment Phase (4/4/2024 - 5/28/2024)

The Deployment Phase is a pivotal stage where the project is made accessible to users in a live environment. This phase comprises several key activities:

1. **Deploy the App to Selected Platforms (4/4/2024 - 4/10/2024):** The deployment process involves making the project available to users on the selected platforms, ensuring a smooth and error-free transition from development to production.
2. **Meeting with Supervisor (4/12/2024):** A meeting with the project supervisor to finalize deployment plans and ensure alignment with project objectives.
3. **Conduct Final Testing in a Live Environment (4/15/2024 - 4/23/2024):** Comprehensive testing was carried out in the live environment to identify and address any issues that may arise during actual usage.
4. **Meeting with Supervisor (4/26/2024):** A meeting to report on the progress of the project's deployment, ensuring that it aligns with expectations and goals.

5. **Monitor User Feedback and Make Necessary Improvements (4/29/2024):** Following deployment, the project team focused on monitoring user feedback and making any necessary improvements or updates.
6. **Meeting with Supervisor (5/3/2024):** A final meeting with the project supervisor to review the deployment phase and confirm the successful deployment of the project.
7. **Project Submission (5/28/2024):** The formal submission of the completed project, marking its availability for users and the project's official conclusion.

The Deployment Phase marks the transition from development to live use, ensuring that the project is operational, reliable, and ready for user interaction. During this phase, rigorous testing and close monitoring are essential to guarantee a successful deployment.

The project timeline and activities will be closely monitored and adapted to ensure successful project delivery. The Figure 1 shows the Gantt chart of the project with detailed breakdown and planning of tasks. The tasks will be further divided into manageable sub-tasks and will be assigned to team members. Each sub-task will have an assignee, a reporter and a reviewer to ensure accountability and quality control throughout the project's execution.

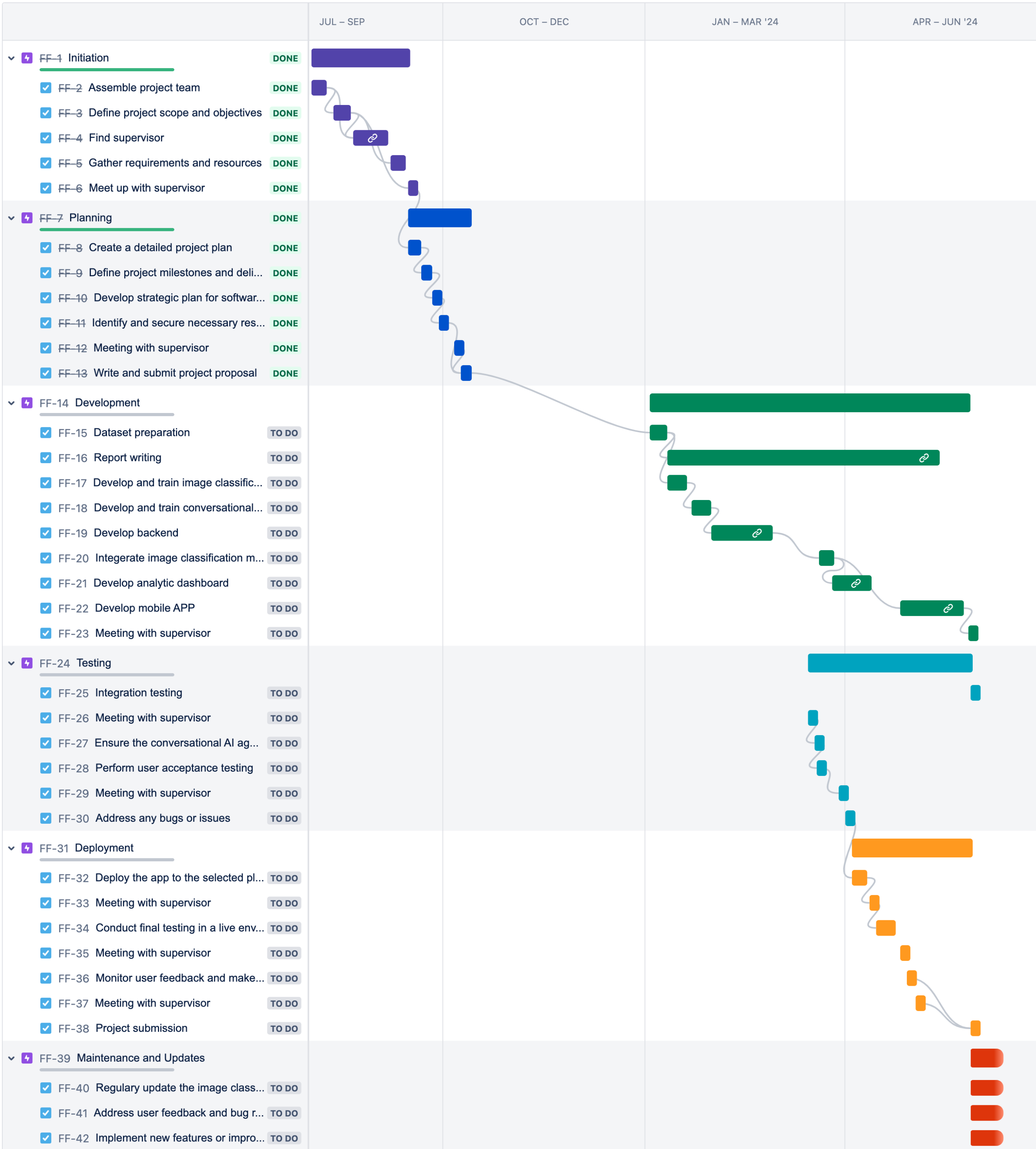

Figure 1: Project Timeline

3 Outcomes

The key outcome from this project are Image Dataset Of Faunas in Australia, Image Classification Model, Domain Trained LLM, Analytic Dashboard and Mobile application.

- **Dataset:** One of the key outcome of this Project is the creation of the dataset of list of images annotated with species label. This dataset will be used to train the classifiers. It will be a high-quality resource that can be used in the future research.
- **Dashboard:** A web based dashboard will be developed to monitor the usage and performance of the application. This dashboard will provide the data visualization and analytics for the team to track on user growth, accuracy of the species, AI agent queries, etc. It will be key in the improvements and feature additions of the app.
- **Classification Model:** An image classification model tailored to accurately identify the animals and bird species will be developed. The best performing classifier will be selected for the classification tasks and it enables real-time species recognition.
- **Conversational AI Agent:** A Conversational AI Agent powered by fine tuned large language model to provide response to user's questions about wildlife. It will be trained on domain specific dataset to provide accurate and specific conversational access to wildlife information.
- **Mobile Application:** An integrated mobile application that delivers the image recognition and conversational feature to the user is key outcome of this project. This application will demonstrate how AI can be applied in the field of tourism to enrich and promote ecotourism.

In addition to the outcome on application level, the development of this application has potential to generate wide ranging outcomes that further the conservation, education, tourism and research goals. Some of the outcome of the application on the user level are discussed below:

- **Knowledge Dissemination:** Tourists and local enthusiasts will have a user-friendly platform to quickly identify and learn about the wildlife around them. This promotes educational engagement with nature.
- **Conservation Awareness:** By highlighting endangered species and providing detailed information about their conservation status, the app will raise awareness and potentially drive conservation initiatives.
- **Enhanced Tourist Experience:** With real-time species identification and information retrieval, tourists can have richer, more informed experiences during their travels, deepening their connection with the local environment.
- **Reduction in Wildlife Harassment:** By educating tourists about local wildlife, inadvertent harassment or harmful interactions can be minimized, ensuring a safer coexistence between humans and wildlife.
- **Data Collection and Research:** The app can serve as a platform for citizen science, where users' submissions can be utilized for research purposes, aiding in biodiversity studies and conservation planning.
- **Community Engagement:** By allowing users to share their discoveries on social media and interact with the conversational AI, we foster a sense of community and shared learning.

These outcomes not only fulfill the immediate objectives of the ”Fauna Finder” app but also contribute to broader environmental, educational, and social goals. By focusing on tangible and realistic impacts, the project stands to make a lasting positive difference in the realm of wildlife appreciation and conservation.

Additionally, successful completion of this project will offer a practical opportunity to gain firsthand exposure to the fields of Machine Learning, Computer Vision, Data Analytics, and Software Development. The project team will have the chance to apply the insights acquired during their coursework. Moreover, the ongoing updates and maintenance of the project even after completion will serve as an important addition to the team members' portfolios, enabling highlight their skills and enhance the prospects for employment and career growth.

References

[1] M. Tan and Q. V. Le, “Efficientnet: Rethinking model scaling for convolutional neural networks,” 2020.

[2] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” 2015.

[3] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, “Mobilenets: Efficient convolutional neural networks for mobile vision applications,” 2017.

[4] X. Liu, Z. Jia, X. Hou, M. Fu, L. Ma, and Q. Sun, “Real-time classification of marine animal images based on mobilenetv2 and transfer learning,” Journal of Marine Science and Engineering, vol. 8, no. 11, p. 846, 2020.

[5] A. W. D. U. Shalika and L. Seneviratne, “Animal classification system based on image processing &; support vector machine,” Journal of Computer and Communications, vol. 04, no. 01, pp. 12–21, 2016.

[6] A. S. Cardoso, S. Bryukhova, F. Renna, L. Reino, C. Xu, Z. Xiao, R. Correia, E. Di Minin, J. Ribeiro, and A. S. Vaz, “Detecting wildlife trafficking in images from online platforms: A test case using deep learning with pangolin images,” Biological Conservation, vol. 279, 2023.

[7] L. Mugambi, J. N. Kabi, G. Kiarie, and C. W. Maina, “Dsail-porini: Annotated camera trap image data of wildlife species from a conservancy in kenya,” Data Brief, vol. 46, p. 108863, 2023, mugambi, Lorna Kabi, Jason N Kiarie, Gabriel Maina, Ciira Wa eng Netherlands 2023/01/11 Data Brief. 2022 Dec 27;46:108863. doi: 10.1016/j.dib.2022.108863. eCollection 2023 Feb. [Online]. Available: https://www.ncbi.nlm.nih.gov/pubmed/36624766

[8] T. Battu and D. S. Reddy Lakshmi, “Animal image identification and classification using deep neural networks techniques,” Measurement: Sensors, vol. 25, 2023.

[9] Z. Ge, C. McCool, C. Sanderson, A. Bewley, Z. Chen, and P. Corke, “Fine-grained bird species recognition via hierarchical subset learning,” in 2015 IEEE International Conference on Image Processing (ICIP). IEEE, 2015, pp. 561–565.

[10] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, “Language models are unsupervised multitask learners,” 2019. [Online]. Available: https://api.semanticscholar.org/CorpusID:160025533

[11] T. Bocklisch, J. Faulkner, N. Pawlowski, and A. Nichol, “Rasa: Open source language understanding and dialogue management,” 2017.

[12] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, A. Rodriguez, A. Joulin, E. Grave, and G. Lample, “Llama: Open and efficient foundation language models,” 2023.

[13] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei, “Language models are few-shot learners,” 2020.

[14] G. Delétang, A. Ruoss, P.-A. Duquenne, E. Catt, T. Genewein, C. Mattern, J. Grau-Moya, L. K. Wenliang, M. Aitchison, L. Orseau, M. Hutter, and J. Veness, “Language modeling is compression,” 2023.

[15] A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, P. Barham, H. W. Chung, C. Sutton, S. Gehrmann, P. Schuh, K. Shi, S. Tsvyashchenko, J. Maynez, A. Rao, P. Barnes, Y. Tay, N. Shazeer, V. Prabhakaran, E. Reif, N. Du, B. Hutchinson, R. Pope, J. Bradbury, J. Austin, M. Isard, G. Gur-Ari, P. Yin, T. Duke, A. Levskaya, S. Ghemawat, S. Dev, H. Michalewski, X. Garcia, V. Misra, K. Robinson, L. Fedus, D. Zhou, D. Ippolito, D. Luan, H. Lim, B. Zoph, A. Spiridonov, R. Sepassi, D. Dohan, S. Agrawal, M. Omernick, A. M. Dai, T. S. Pillai, M. Pellat, A. Lewkowycz, E. Moreira, R. Child, O. Polozov, K. Lee, Z. Zhou, X. Wang, B. Saeta, M. Diaz, O. Firat, M. Catasta, J. Wei, K. Meier-Hellstern, D. Eck, J. Dean, S. Petrov, and N. Fiedel, “Palm: Scaling language modeling with pathways,” 2022.

[16] React native documentation. [Online]. Available: https://reactnative.dev/docs/getting-started

[17] Redux documentation. [Online]. Available: https://redux.js.org/

[18] React navigation documentation. [Online]. Available: https://reactnavigation.org/docs/getting-started/

[19] Node.js documentation. [Online]. Available: https://nodejs.org/en/docs

[20] Introduction - material-ui. [Online]. Available: https://mui.com/x/introduction/

[21] D3.js documentation. [Online]. Available: https://d3js.org/getting-started