A Cloud Based Drone Service Platform for Smart Agriculture

281 - Team 12 - San Jose State University
Fall 2022 - Drone Project

Vamshidhar Reddy Parupally *Student ID: 016001427*
Ravindar Reddy Siddenki *Student ID : 016002753*
Lakshmi Ananya Mandava *Student ID : 016315481*
Anurag Gajam *Student ID : 016003377*

***Abstract*—The main objective of this project is to develop a cloud-based drone system platform. This application will provide a comprehensive high-level overview of the necessary components that are required for real-time autonomous drone rental platform, mainly, Cloud deployment, administration, integration of mission planner, database storage, Graphical User Interface (GUI) for selecting and reserving a required drone, tracking the UAV operations and user service management will be included in this prototype. Our application will comprise of the following components**

I. Introduction

The term "Drone" which is typically used to describe any unmanned aircraft, also known as unmanned aerial vehicles (UAVs), are capable of performing a diverse range of duties, from military and agricultural operations to package delivery. These UAVs range in size from the size of your hand's palm to that of an airplane. In order to maintain their altitude and provide the hovering capabilities required for photography or video motion, drones use air pressure sensors. Drones can fly precisely when barometric pressure sensors are used in conjunction with an accelerometer and gyroscope and they range from level 0 which is no autonomy to 5 which is full autonomy. Agricultural drones lower the demand for human labor and other input resources, and assist the farmer in increasing the productivity as well as reducing the expenses.

These drones are also one of the most promising technologies for enabling sustainable resource use and meeting global demand for quantity and quality in smart agriculture. Farmers with both large and small-scale enterprises already use digital tools like UAVs, sensors, and robotics to cut inputs and maximize harvests. When digital solutions are included on the labels of plant protection goods, the farmer's flexibility will grow. Furthermore, Agricultural drones can be used in various aspects including planting, monitoring of the crop, spraying, irrigation, and health assessment of the crop.

A. *Purpose*

The main purpose of the project is to provide farmers with UAVs to assist with their farming with the help of an accessible user-interface and a remote pilot assitance with the help of cloud mission planner.

B. *Objective*

The main objective is to develop a cloud-based drone system platform. This application will provide a comprehensive high-level overview of the necessary components that are required for real-time autonomous drone rental platform, mainly, Cloud deployment, administration, integration of mission planner, database storage, Graphical User Interface (GUI) for selecting and reserving a required drone, tracking the UAV operations and user service management will be included in this prototype. Our application will comprise of the following components.

- A map-based interface that shows status of the connected drones, locations, connectivity including statistics of drones, services, and tasks.
- An interface to manage booking and user payments.
- Storing and updating the information of connected UAVs and service records in the Database
- Integration of the mission planner.
- An interface dedicated to the farmer to select and request a drone service, raise maintenance request, track the drones
- An interface dedicated to drone pilot to view upcoming bookings, update their availability

- Farmers can track the drone and view its service reports
- An interface to display the statistics of all the drones and service records
- Admin can track the connected UAVs and monitor the status.

C. Market Analysis

According to a market analysis of UAVs, the market for these aircraft is expected to reach a size of 13.1 in 2021. This market's size is anticipated to increase to 40.7 billion USD by the end of 2026, per certain research and projections.

The market for agricultural drones is being significantly fueled by the rising global population and rising food consumption. In addition, agricultural drones may aid farmers in field surveillance, crop productivity enhancement, and crop health and infection management, all of which have boosted demand for agricultural drones on the world market. According to UN projections, there are currently 7.7 billion people on earth, and by the year 2050, that number is expected to increase to 9.6 billion.

Production and availability of food are predicted to double from the current rate. According to the Agriculture Outlook 2019 study from the OECD, agricultural output will rise by 15% over the next ten years. Improvements in farming practices, notably in precision farming and agricultural production per acre, are under pressure due to these factors and the global shortage of agricultural lands. The majority of farmers use drones and soil investigations in the field to understand the productivity pattern. Unmanned aerial vehicles are used to produce 3D maps in real-time and at a reasonable cost. These are employed to create seed planting strategies for organizing various data types. Drones are utilized to address pest infestations, productivity challenges in agriculture, and irrigation problems.

For instance, in 2019 a new firm called DroneSeed Company received FAA permission to utilize unmanned aerial vehicles to transport water, tree seeds, pesticides, fertilizers, and other supplies for reforestation projects in the United States. Unmanned aerial vehicles (UAVs) with advanced technology are well-equipped with a variety of sensors, multispectral capabilities, and RGB cameras to take numerous photographs. These technical developments are anticipated to fuel market expansion throughout the course of the forecast period. The global market is fragmented because there are large rivals operating in the region. Drone Deploy, DJI, GoPro, Precision Hawk, and AeroVironment Inc., some of the well-known companies that account for a significant proportion, are currently leading the market.

However, the lack of significant entry barriers is expected to encourage more domestic competitors to enter the global market. To their international clientele, other major providers like Trimble Navigation Ltd. and 3D Robotics provide complete hardware and software solutions for drones. These companies are also involved in the development of original goods and innovations. The three different types of drones—rotary wing, fixed wing, and hybrid—can also be used to categorize the market study. The rotary-wing sector now holds the largest market share and is expected to maintain its dominant position. Rotary wing drones are frequently employed for a number of farming applications because of their great structural benefits.

Multi-rotor aircraft can VTOL (vertical takeoff and landing) to carry large payloads without needing additional airspace. In addition, multi-rotor drones have a longer flight time, which makes them perfect for aerial photography, image capturing, and precision spraying. Since hybrid drones have significant structural advantages and are well-suited for a range of farming applications, the hybrid drone market is predicted to grow at the highest CAGR. They combine the benefits of fixed-wing and multirotor UAVs. These elements are expected to boost demand for farm UAVs during the course of the projected period.

II. Relevant Work

A market analysis of UAVs predicts that in 2021, the market for these aircraft will grow to a size of 13.1. According to some studies and projections, the size of this market is expected to increase to 40.7 billion USD by the end of 2026.

Drones, also known as unmanned aerial vehicles (UAVs), are gaining popularity recently and ushering in a new era of mobile computing. Due to the support of extremely cutting-edge technologies, net-

works of drones may soon be employed to provide civilian drone services and become omnipresent.

A recent paper on Drone services, has proposed on how to use drone for fly-in, fly-out computing infrastructure and drones as a service while highlighting the problems with data management and system design that these notions raise.

A. Micro Drones

Microdrones is a company that utilises the drone services to provide research and analysis of the terrain and other geographical components on the earth to facilitate easier and faster access to new discoveries and make analysis easier and faster.

Building a model to optimize the monitoring and detection of areas that could be susceptible to flooding, warn of impending flooding, and find appropriate solutions helped RSS-Hydro recently measure potential flooding areas in the community south of Luxembourg.

This serves as a motivation for our project to make it useful for the drone agricultural services of the farmer and survey the farmland.

III. Cloud Infrastructure Components

*1) **Amazon Route 53:**:* An extremely scalable Domain Name System (DNS) service is Route 53. By transforming the names of our application into numeric IP addresses, it is very accessible and designed to enable designers and organizations a very practical and reliable way to direct end users to Internet apps.

*2) **Elastic Load Balances:**:* Our incoming traffic is automatically split up across numerous targets, including EC2 instances, containers, and IP addresses, in at least one Availability Zone, thanks to elastic load balancing. It keeps an eye on the well being of the targets it has chosen and only sends traffic to the reliable ones.

*3) **Auto Scaling:**:* Auto Scaling keeps track of your apps and adjusts their availability as needed to provide constant, reliable services at the affordable cost. AWS Auto Scalability makes it quick and easy to build up application scaling for a very long time and across multiple administrations.

*4) **AWS EC2:**:* Scalable computing capabilities is provided via the Amazon Elastic Compute Cloud (AWS Cloud) by Amazon Web Services. As part of the EC2 setup, creating an Amazon Machine Image (AMI) is required. This image consists of working programs, frameworks, and designs. As long as an AMI is registered with EC2 and mounted to the Amazon Simple Storage Service (S3), customers can dispatch virtual machines as needed.

*5) **AWS RDS:**:* Amazon Web Services(AWS) provides managed SQL database assistance under the name Relational Database Service (RDS) . To store and organize the data, Amazon RDS supports a number of data set motors. Additionally, it helps with tasks related to the management of social information bases, such as information mobility, reinforcement, recovery, and repairing.

*6) **AWS CloudWatch:**:* With the aid of Cloud Watch, a surveillance and reliability service, you can keep an eye on your apps, respond to changes in system performance, and make the most use of your resources. The primary operational and surveillance data types that CloudWatch captures are logs, metrics, and events. In order to establish alarms for the side-by-side presentation of metrics and logs and to detect odd behavior in the settings, it is used. It helps in the search for new insights to keep our application's effectiveness.

*7) **AWS S3:**:* Through a web-based administration interface, the S 3 service provides object storage. It makes use of the scalable storage platform that Amazon employs to power its global network of online shops. Because it provides for the storage of any type of object, Amazon S3 is the greatest choice for tasks like hybrids cloud storage, backup and restoration recovery plans, data preservation, and large volumes of data for analytics.

A. Cloud Infrastructure Diagram

Fig. 1. Cloud Infrastructure Diagram

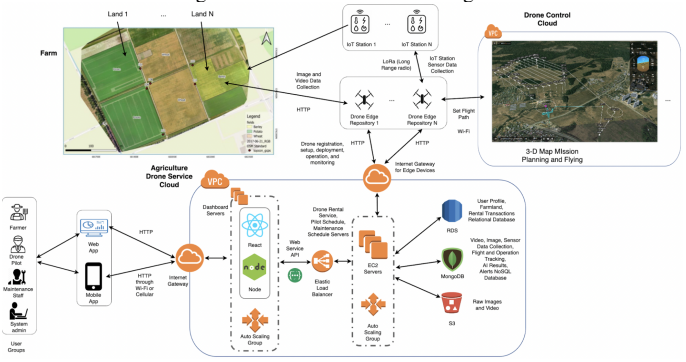
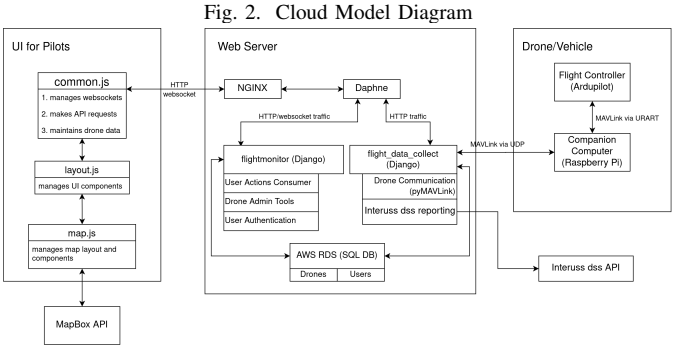


Fig. 2. Cloud Model Diagram



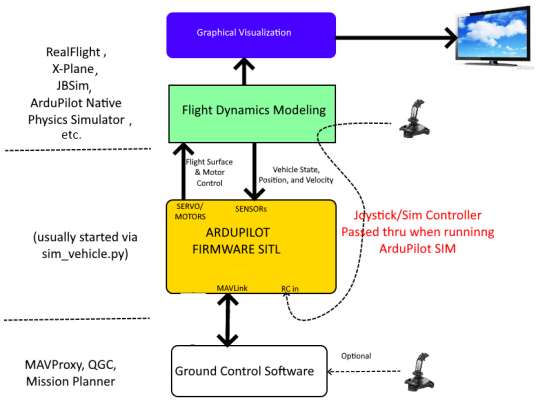
The application includes mainly four components those are UI for client and pilot, Web Server,Cloud Mission Planner and Drone Vehicle. We use map box api for the map related activities necessary for mission planner. UI contains set of dashboards and other services required for the user and pilot to use the application effectively. Web server involves Ng- nix, RDS and other monitoring and data col- lection services. This components also involves REST Api's for com- munication and data transfer.

Drone vehicle, here we use a simulated drone for the purpose of demonstration. This component in real world would be a drone and it sends the feed to the system which would store it in a No SQL database for future analysis.

IV. Simulator Design and Implementation

A. *Simulator Design and Implementation*

Fig. 3. Ardu Pilot Simulator Design



SITL, the aggregate name for ArduPilot's native firmware simulator and FDM program, is often utilized alongside its creator GCS's MAVProxy (all launched by sim vehicle.py). To execute SITL simulations, Mission Planner can also make use of FDM and ArduPilot's firmware simulator.

The simulator module of Ardu Pilot is interfaced to other FDMs or Graphical modules, such as Real Flight or X-Plane, that are frequently used as independent drone simulators in order to acquire higher quality images and/or more accurate physics models.

The simulators with the highest usage are:

- View available drones and services based on the preferences.
- The simulator that developers employ most frequently is called SITL (Software In The Loop). All SITL builds of ArduPilot have a straightforward simulator. The autotester makes use of it, and the simulators below were really constructed using SITL.
- To mimic using Mission Planner as the GCS, utilize Windows and Mission Planner.

Without requiring any specialized hardware, SITL enables you to easily run ArduPilot on your PC. It makes use of ArduPilot's portability and ability to function on a very wide range of systems. Your PC is merely one more platform on which ArduPilot can be created and used.

The flight dynamics model in the flight simulator is where the sensor data is sourced from when SITL is operating. ArduPilot can interact with several external simulators and includes a large selection of built-in vehicle simulators. This makes it possible to test ArduPilot on a very broad range of vehicle types. SITL, for instance, can simulate:

- the multirotor aircraft
- a fixed-wing vehicle
- tethered vehicles
- undersea automobiles
- lens gimbals
- antenna monitors

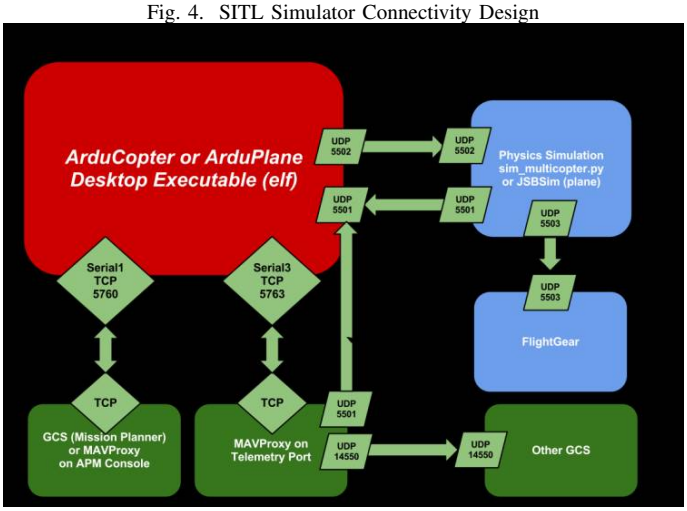
It is simple to add new simulated vehicle types or sensor types.

ArduPilot on SITL has the complete complement of C ++ development tools available to it, tools for dynamic analysis, static analysis, and interactive

debugging. This greatly simplifies the process of creating and testing new ArduPilot features.

B. *Simulation Connectivity Design*

Fig. 4. SITL Simulator Connectivity Design



SITL, the aggregate name for ArduPilot's native firmware simulator and FDM program, is often utilized alongside its creator GCS's MAVProxy (all launched by sim vehicle.py). To execute SITL simulations, Mission Planner can also make use of FDM and ArduPilot's firmware simulator.

The simulator module of Ardu Pilot is interfaced to other FDMs or Graphical modules, such as Real Flight or X-Plane, that are frequently used as independent drone simulators in order to acquire higher quality images and/or more accurate physics models.

V. Cloud Data Design and Implementation

A. *Cloud DB Design*

The project's database management system is essential to the successful completion of all data-related operations. In this case, the cloud platform is used to store user, drone, reservation data, and information about the administrators and employees who manage the end-to-end application's back-end infrastructure system.

By gathering the information produced by the simulated drone, these employees are in charge of tracking and monitoring the drone. Since the platform offers a number of software services that meet our needs for building our application together with extremely high availability and scalability, Amazon Web Services is the cloud provider utilized for our project.

TABLE I
Database Table

Database	Tables
Relational DBMS-MySQL	• Drone details • Booking records • maintain user data • Scheduling services
NoSQL Database-MongoDB	Spotting location inputs, Analyzing, Collecting and Storing Data.

A relational database management system (RDBMS) is a collection of data items with pre-described relationships between them that is used to store and retrieve information about public customers' offerings. The admin team can keep track of drone statistics in the non-relational database, and both the public user and the admin can operate drones by retrieving location information from the non-relational database. The admin may keep an eye on the drone data in the non-relational database. Drones can be tracked by both the general user and the administrator by obtaining location information from the non-relational database. Above all, we want to make sure that using our website provides a pleasant user experience.

The relational database stores user information like Farmer ID, Name of the Farmer, booking time, Booking ID, Start, and waypoint location.

Sensor statistics consist of drone place, acceleration, altitude perspective, and the sensor snapshots are saved within the non-relational database.

B. *Cloud DB Design*

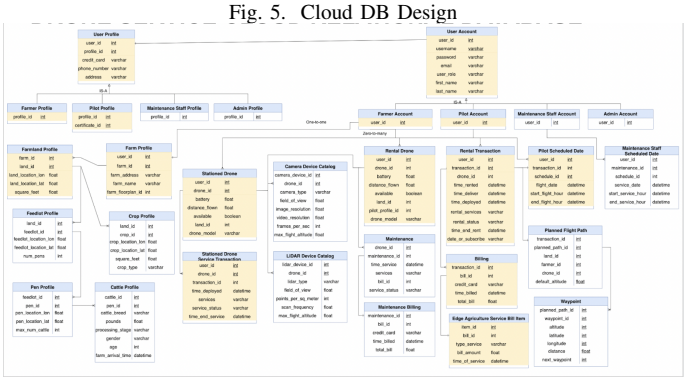
A relational database management system (RDBMS) is a collection of data items with pre-described relationships between them that is used to store and retrieve information about public customers' offerings. The admin team

can keep track of drone statistics in the non-relational database, and both the public user and the admin can operate drones by retrieving location information from the non-relational database. The admin may keep an eye on the drone data in the non-relational database. Drones can be tracked by both the general user and the administrator by obtaining location information from the non-relational database. Above all, we want to make sure that using our website provides a pleasant user experience.

The relational database stores user information like Farmer ID, Name of the Farmer, booking time, Booking ID, Start, and waypoint location.

Sensor statistics consist of drone place, acceleration, altitude perspective, and the sensor snapshots are saved within the non-relational database.

Fig. 5. Cloud DB Design



Tables as below :

User profile - To store information about the users involved in the application, we are storing user id, profile id, credit card , phone numbers and addresses of the users

Farmer profile - This table is to specifically store the farmers profile ids and inherits the attributes from the User Profile table.

Pilot profile - This is designed to specifically store the information related to the pilots and inherits the attributes from the User profile table.

User Account- This is used to store the information related to the Users (Farmers, pilots, Admin, Maintenance staff) such as username, password, email, user role, first and last names.

Farm profile - This table has been introduced to store the general information related to the Farmlands such as its address, name that farmer enters into the application. Additionally, every farmer is associated with a farm profile and vise-versa.

Farmland profile - store the information related to the Farmlands such as longitudes, latitudes and square feet of the farm

Rental Drone - Table that stores the details related to the drone that has been rented by the farmer. Helps in storing distance flown,drone model information,its availability, a column named drone id to identify a drone uniquely.

Rental Transaction - Table to store the information such as the time at which the drone was booked, delivered, deployed along with the type of services it offered and end date of the service.

Billing - This table is used to store the information about the billing such as transaction id, bill amount, billed time.

Pilot Scheduled date - This contains details of the drone flight such as start hour, end hour, date.

Edge Agriculture service bill item - Contains more details about the billing such as type of service, billing id, amount, time of the service.

Database implementation details:

```
CREATE  DATABASE dronedatabase;
/* Delete If exists */

Delete * from farmer;

/*create table farmer*/

CREATE TABLE FARMER(
ID int NOT NULL,
Email varchar(255),
Name varchar(255),
Password varchar(255),
Gender varchar(255),
dateOfBirth varchar(255),
Address varchar(255), state varchar(255),
streetNo varchar(255), zipcode varchar(255),
City varchar(255), unitNo varchar(255),
Driverlicense varchar(255),
Farmutility varchar(255),
Paymentmethod varchar(255),
Paymentdetails varchar(255),
phoneNumber varchar(255),
Role varchar(255),
```

```
 PRIMARY KEY (ID)
);


/* Delete If exists */

Delete * from pilot;

/*create table pilot*/

CREATE TABLE PILOT(
ID int NOT NULL,
Email varchar(255),
Name varchar(255),
Password varchar(255),
Gender varchar(255),
dateOfBirth varchar(255),
Licensedetails varchar(255),
Pilotlicense varchar(255),
Address varchar(255), state varchar(255),
 streetNo varchar(255), zipcode varchar(255),
City varchar(255), unitNo varchar(255),
phoneNumber varchar(255),
Role varchar(255),
 PRIMARY KEY (ID)
);



/*Delete if Exists*/

Delete * from DroneBookingDetails ;

/* DroneBookingDetails table creation */

CREATE TABLE DroneBookingDetails(
bookingId int NOT NULL,
droneBaseCost varchar(255),
serviceDuration varchar(255),
equipmentCost varchar(255),
shippingCost varchar(255),
pilotCharge varchar(255),
pilotName varchar(255),
pilotLicense varchar(255),
phoneNumber varchar(255), equipment varchar(255),
 Brand varchar(255),
fromDate Date, toDate Date,
 paymentMade varchar(255),
Status varchar(255), farmerEmail varchar(255),
farmLand varchar(255),
serviceType varchar(255), pilotEmail varchar(255),
 PRIMARY KEY (bookingId)
);
```

C. *NOSQL database*

NoSQL databases, commonly called ”never just SQL,” store data differently from relational databases. There are many different types of NoSQL databases, depending on the data model. The most common types are document, key-value, wide-column, and graph. They provide flexible schemas and can handle high user and data loads with ease.

Tables that would be implemented here are as described below

Camera Image - his table is to specifically store the image information and Farmer id to associate a particular image with the farmer

Video - This is designed to specifically store the information related to the video and like Camera

Trip - Stores the information related to the Drone flight such as timestamp and path id.

Actual Waypoint- Table that stores the details of the waypoints in order for the drone to navigate.

Data Collection Service Tracking - Table to store attributes such as longitude, latitude of the drone, boolean attributes to check if the video is recording or if the image has been taken.

D. *Scalability Design and Implementation*

To conquer the problem of scalability and handle the ever increasing load, we can choose one of the solution as Vertical Scaling i.e. using powerful computers with lots of RAM, CPUs etc. to handle the user load and work on the data. But again, there would be an upper limit to the performance you achieve using the hardware modifications.

And we have used the concepts and understanding of

- Network patterns
- Load balancers
- Parallel computing
- Event-driven architecture

We utilized the CAP theorem understanding to overcome the issue with the scalability and tolerance. The image describes the focus of the cap theorem.

E. *Load Balancing and Implementation*

We have tried to utitlize the software load balancing in our application and have used Weighted Scheduling Algorithm where the server is given work in accordance with the weight that has been given to the server. Different weights are assigned to

Fig. 6. CAP Design

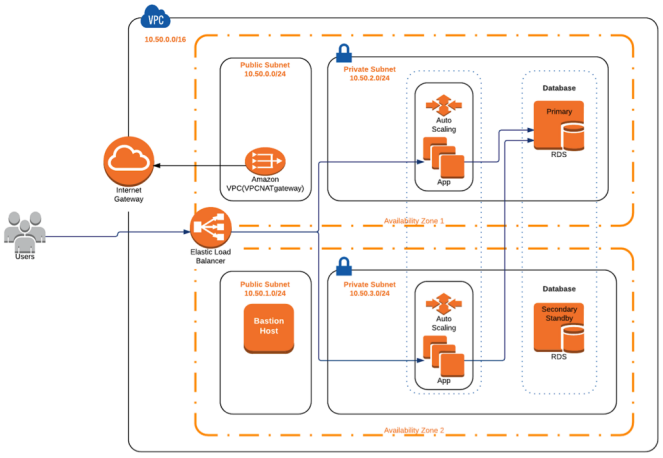


Fig. 7. Scalability Design

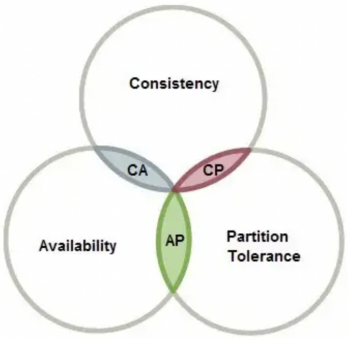
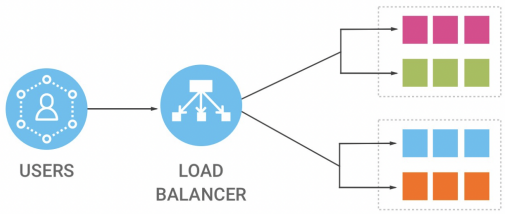


Fig. 8. Load balancing types

TCP/IP Model		OSI Model	
Application Layer	HTTP, HTTPS, SMTP, IMAP, FTP, DNS, NNTP	Application	'Layer 7' Load Balancing
		Presentation	
		Session	
Transport	UDP, TCP, SCTP	Transport	'Layer 4' Load Balancing
Internet		Network	
Network Access (Link)		Data Link	
		Physical	

Fig. 9. Load balancing



the group's various server types in order to disperse the load.

Below are some common load balancer and we have used nginx for our purpose in this application as aload balancet to handle the requests and route them to un-loaded free servers to prevent failure or timeout of the requsets.

- HAProxy
- NGINX
- mod athena
- LVS

VI. System GUI Design

The technology use for developing the front end application for the application is called React Js which is a popular open source library to get the UI work done with ease and lot of features that enables user interface which is very intuitive and easier for the user to understand and use the application.

We have multiple screen each has their significance in the application.

- Pilot Dashboard
- Farmer Dashboard
- Admin Dashboard

Pilot Dashboard has the details like Update the pilot schedule, view services, mark services complete.

Farmer Dashboard has the billing, booking, and other components that help farmer to manage his farm profile and to book the services for the farm and manage the tracking of the drones./

Fig. 10. GUI Design - farmer dashboard

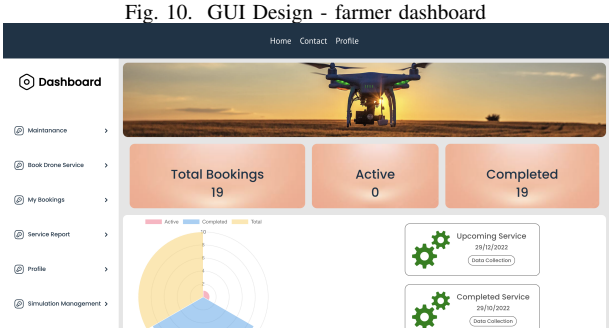


Fig. 11. GUI Design - service booking

Fig. 12. GUI Design - Booking details

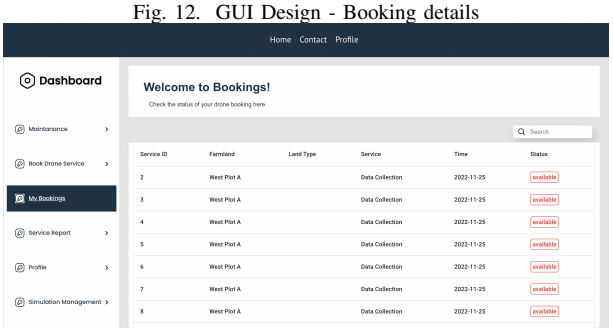


Fig. 13. GUI Design - Booking details

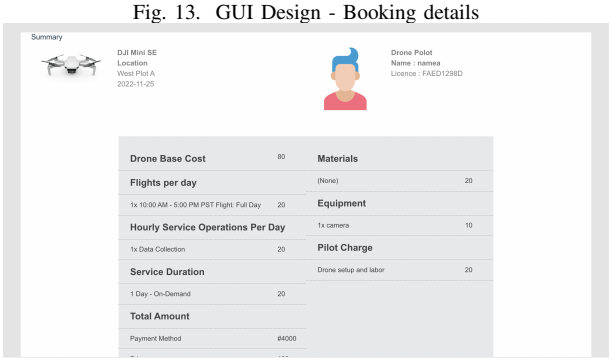


Fig. 14. GUI Design - Profile

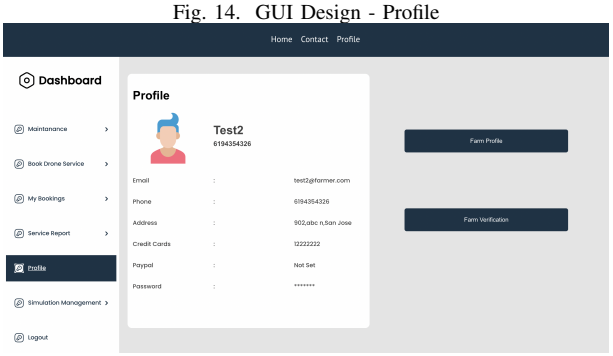


Fig. 15. GUI Design - Pilot dashboard

Fig. 16. GUI Design - Pilot schedule

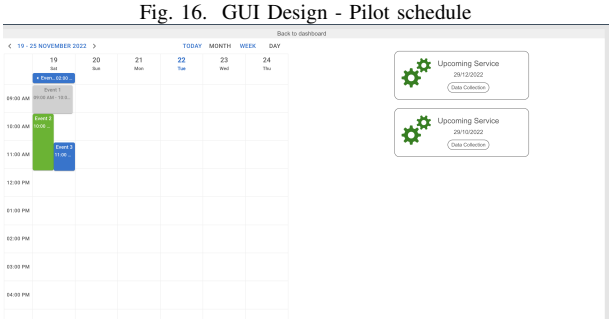


Fig. 17. GUI Design - Pilot service details

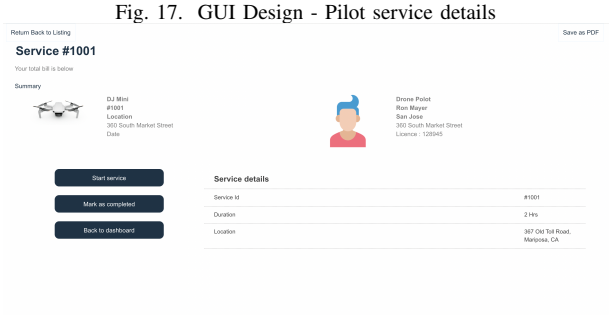


Fig. 18. GUI Design - Mission Planner

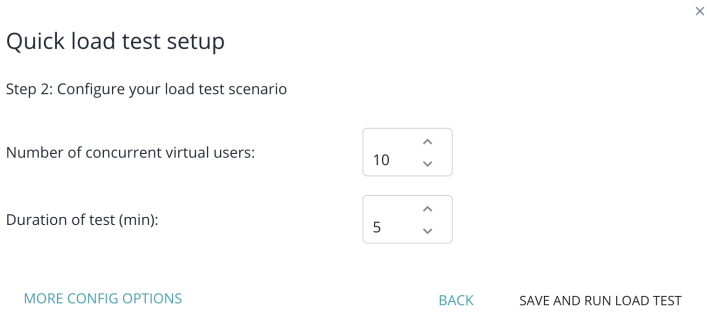
VII. System Application

A. *Scenario for booking*

- 1.Select the Login type as Farmer in the home page and enter the details of the login in the form.
- 2.Once the details are authenticated by the system in the database the API will respond with the details of success for the verification.
- 3.After successful login the user can see his dashboard with the details of the booking like

Fig. 19. GUI Design - Mission Planner view for farmer

Fig. 20. Load test - concurrent users



total, active, and completed bookings in as an overview.

- 4.The use has the options of the selecting the options from the sidebar.
- 5.The sidebar has options and user can select the option and click on book services to create a bookings in the system.
- 6.Once the use selects the book option, he will see the details and can select the filter to get the drone details and select the drone based on the availability.
- 7.Once the details are filled the user can proceed and go to booking.
- 8.Once the booking is made the use can view the booking status in the system booking tab and can view the service report once it is generated.

VIII. System Performance Evaluation

For the purpose of testing the application for load we have used the load ninja application with a concurrent users of 10 users at a time and checked the load on the application and the reports are as furnished below.

The comparisons were made with a pre-defined scenario that we have recorded using the load ninja feature that helps us record the steps to be replicated by the concurrent users in the actual test scenario.

This test tells us about how the system will perform in the actual scenario with the users accessing the application all at once and how the load would impact the application response time and lockouts.

The table lets us know the time taken for each step involved in the scenario with the API response time and the application load time as well as the errors that it encountered during the scenarios.

Fig. 21. Load test - concurrent users

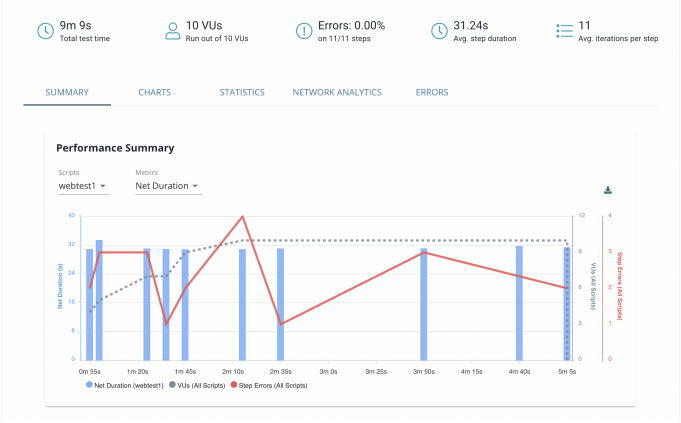
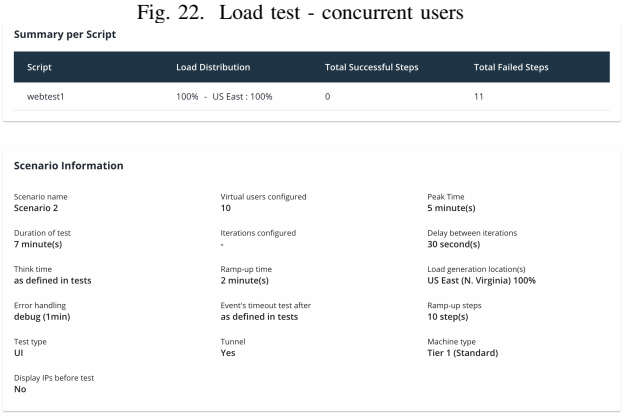


Fig. 22. Load test - concurrent users



The details show the load that the application can handle in the scenario where the user count is 10 and the average usage time is 5 minutes in the system.

Fig. 23. Load test - concurrent users

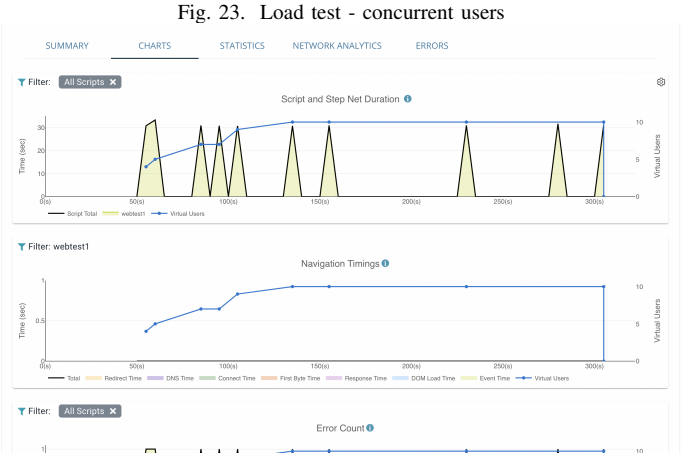
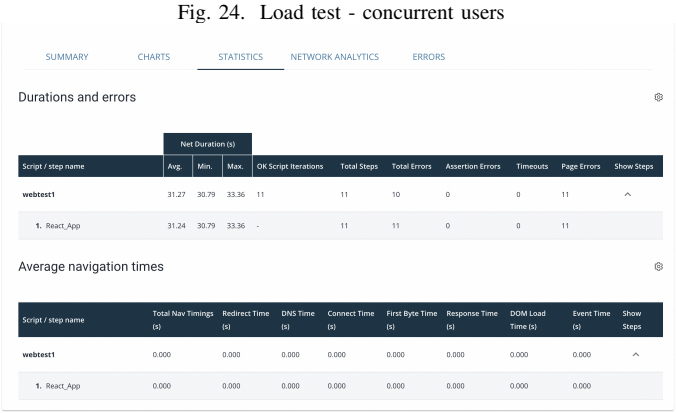


Fig. 24. Load test - concurrent users



IX. Contributions

A. *Work Contribution*

*1) **Vamshidhar Reddy Parupally** :* Developed the dashboards for farmer and pilot in the application with development of billing component and integration of mission planner to the application. Details for Vamshidhar Reddy Parupally activities are mentioned in **table below**

TABLE II
VAMSHIDHAR REDDY PARUPALLY'S CONTRIBUTION

Module	***Contribution briefing***
Mission Planner	Integrated the mission planner into the application
Farmer Dashboard	Developed the dashboard for farmer to book, pay and request for the services.
Pilot Dashboard	Developed the dashboard for pilot to view bookings, update schedule and completion of services.
Billing component	Developed billing component to generate the bill for the services along with the service report

*2) **Anurag Gajam**:* Developed the admin dashboard to view and monitor the drone status and booking details of all the users across the application and the activities are mentioned in **table**

TABLE III
ANURAG GAJAMS'S CONTRIBUTION

Module	***Contribution briefing***
Admin Dashboard	Dashboard with admin view of all drones
Drone management	Add and update new drones to the system

*3) **Ravindar Reddy Siddenki**:* Developed the schema for the entire database and populated with the data that is needed for the development. Developed RESTful API's for the application to add, update and for may other services in the application. **table**

TABLE IV
RAVINDAR'S CONTRIBUTION

Module	***Contribution briefing***
Databases	Creation of tables and establishing the relation with the tables
RESTful API's	Developed API to be consumed by the entire application with low latency.

*4) **Ananya Mandava**:* Developed the UAV catalogue for the Drone service fleet management so that the user can select the drone for specific purposes and then book the services. Worked on the booking component to enable booking of the services.

TABLE V
ANANYAS'S CONTRIBUTION

Module	***Contribution briefing***
UAV Catalog	Developed UAV catalog to fetch the details of available drone based on conditions
Booking component	Developed booking service to allow user to book a drone service.