

# Digital Zoo Project - Week 1

Abdellah Sabhi  
Jalal Mammadov  
Hazal



# Project Overview

- **Main Task:**

Developing a **web-based zoo management system** that allows users to:

- Register and manage animals, species, habitats, feeding schedules, and tickets.
- Implement authentication for user roles (admin, zookeeper, visitor).
- Track animal details and feeding schedules dynamically.

- **Week 1 Objective:**

Set up the **backend** using Django & Django REST Framework (DRF).

Develop API endpoints for **animals, species, habitats, feeding schedules, and tickets**.

Implement **user authentication** (registration & login).

Connect the **frontend (React.js)** with the backend.

# Features:

## User Interaction:

- **Admin Users:** Can add, update, and delete animals, species, and habitats.
- **Zookeepers:** Can manage feeding schedules.
- **Visitors:** Can book tickets and view animals.

## Animal & Habitat Management:

- Users can register **animals and habitats** in the system.
- Animals belong to **specific species and habitats**.

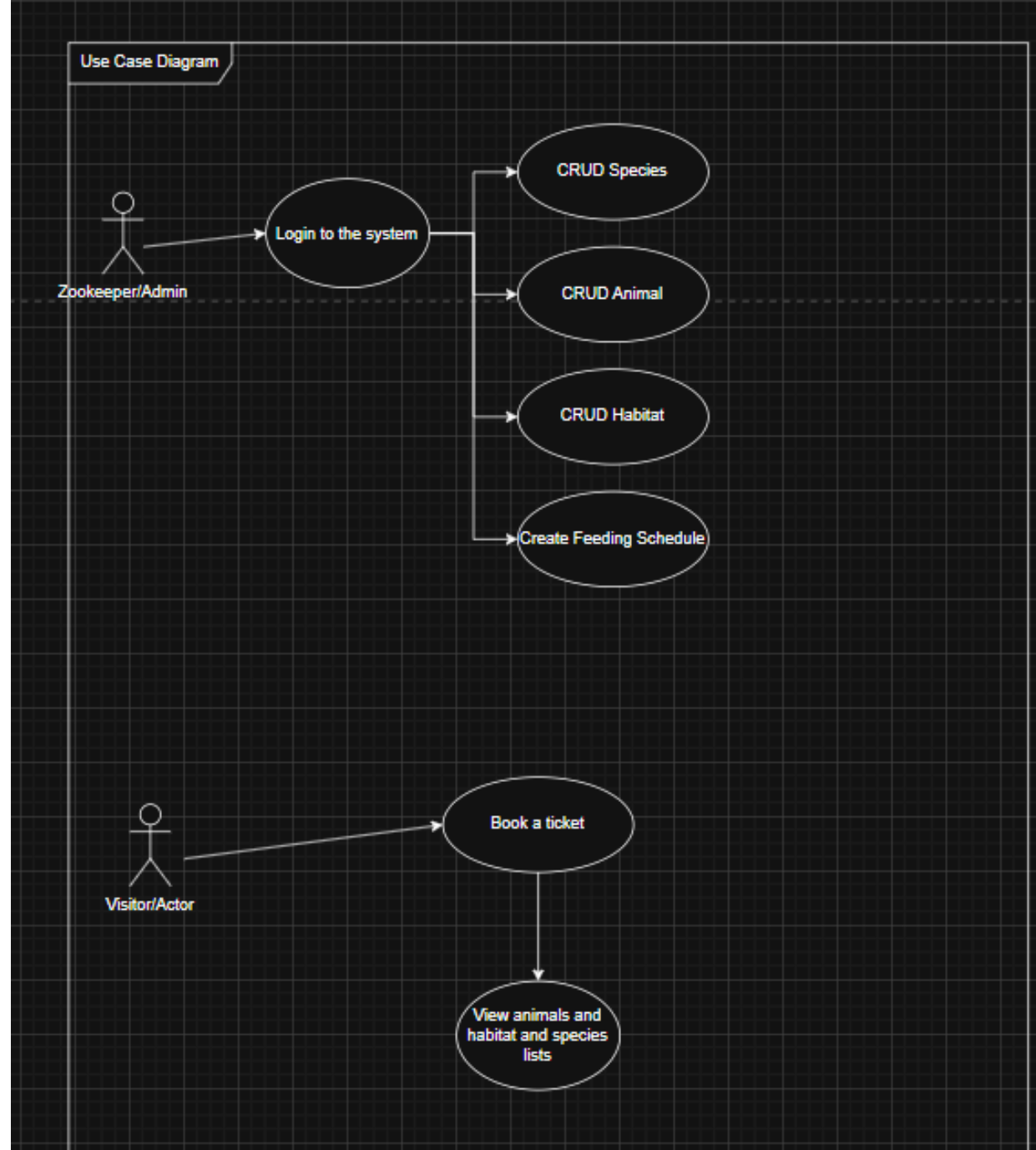
## Feeding Schedule Management:

- Feeding schedules are **assigned to animals**.
- System **records feeding logs**.

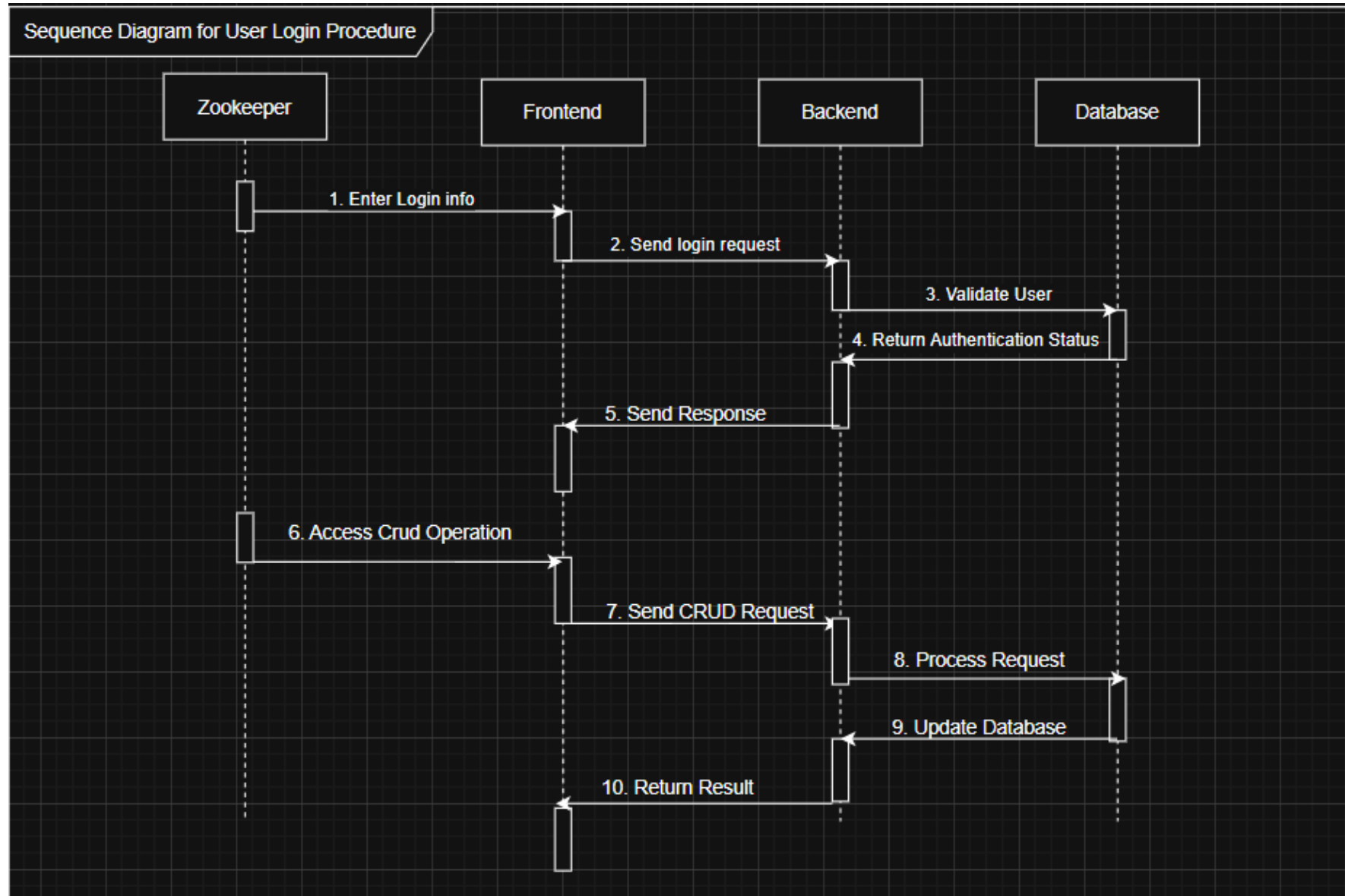
## Ticketing System:

- Visitors can **book tickets** for zoo visits.
- Admin can **manage ticket availability**.

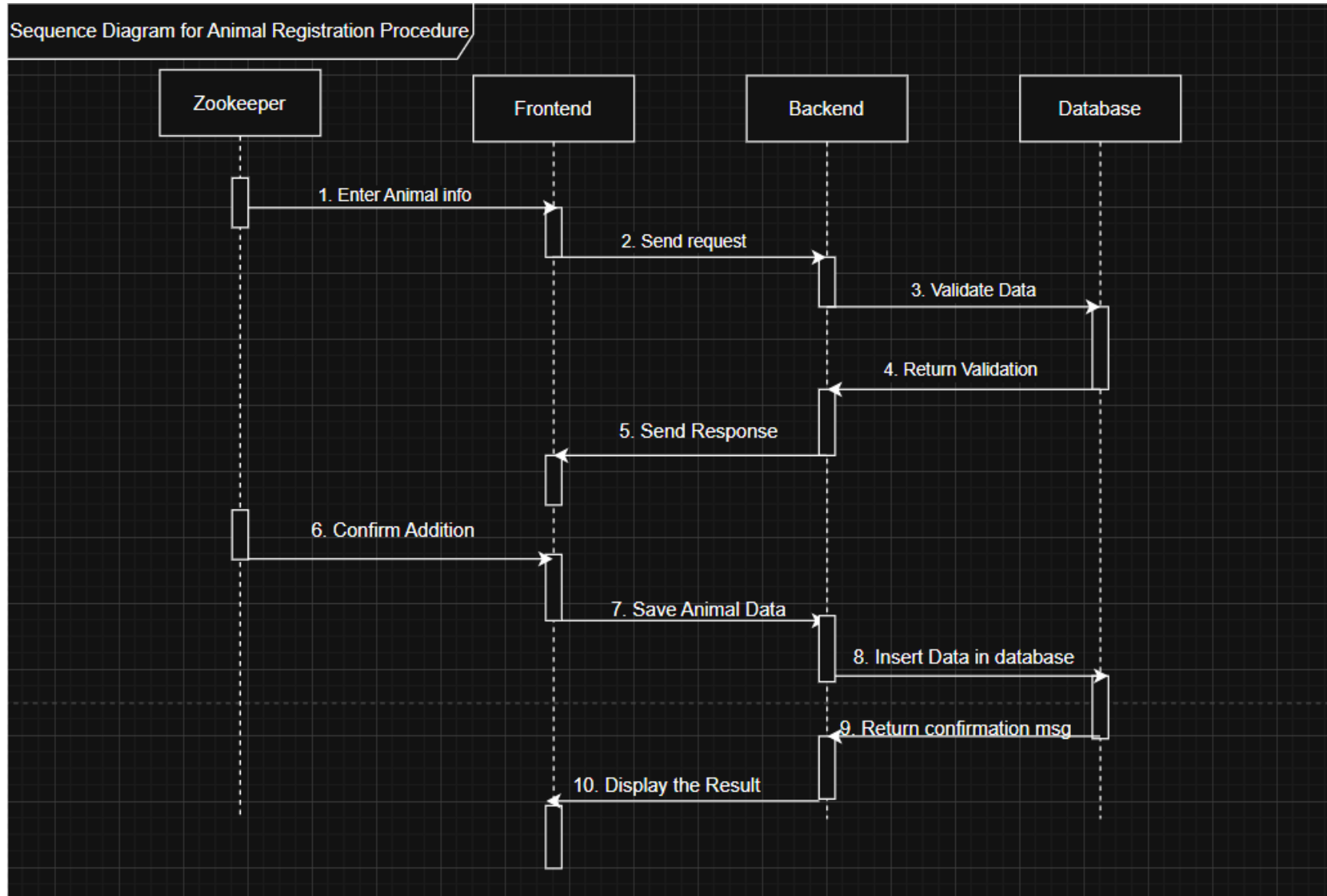
# Use case Diagram:



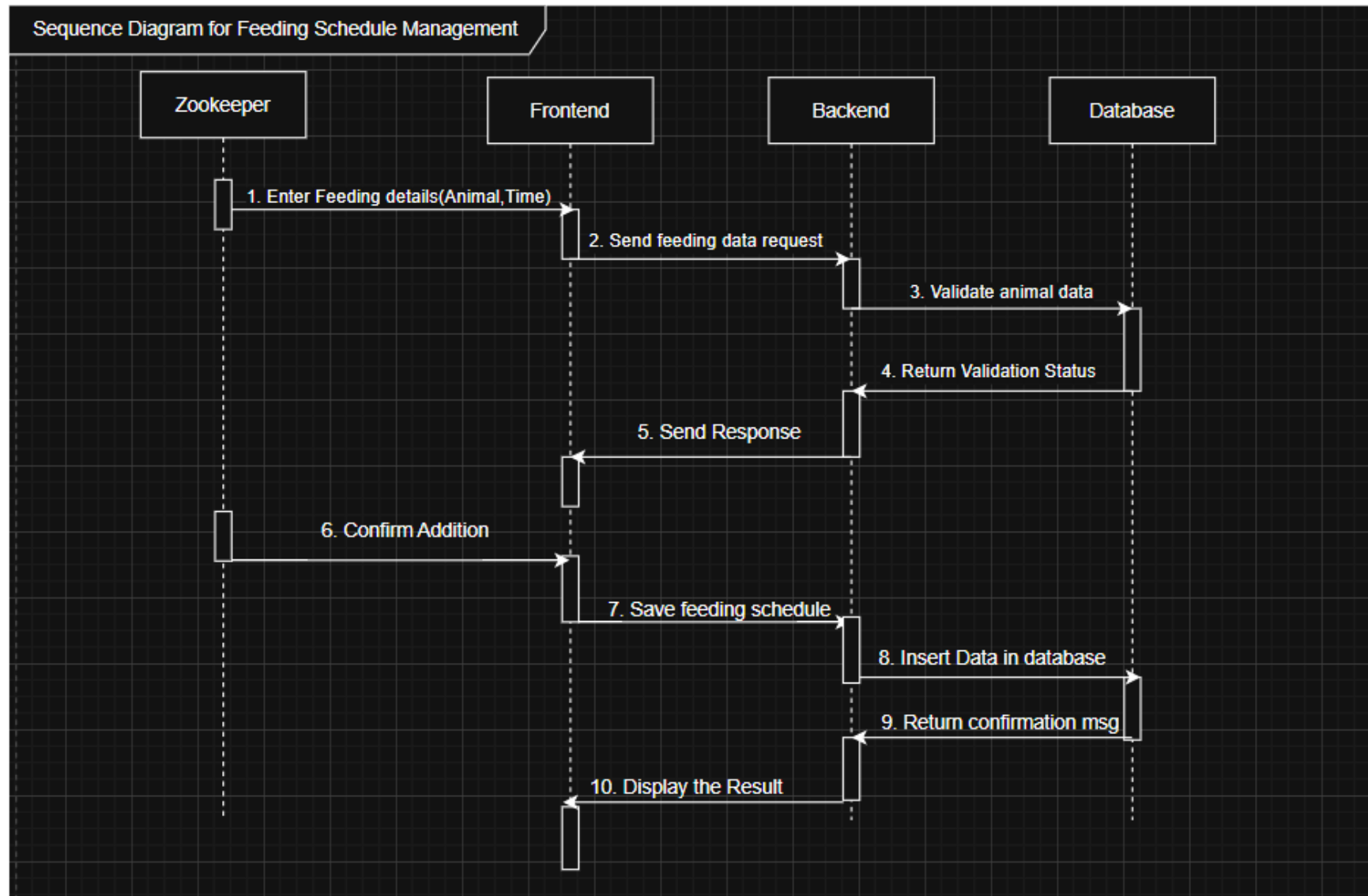
# Sequence diagram



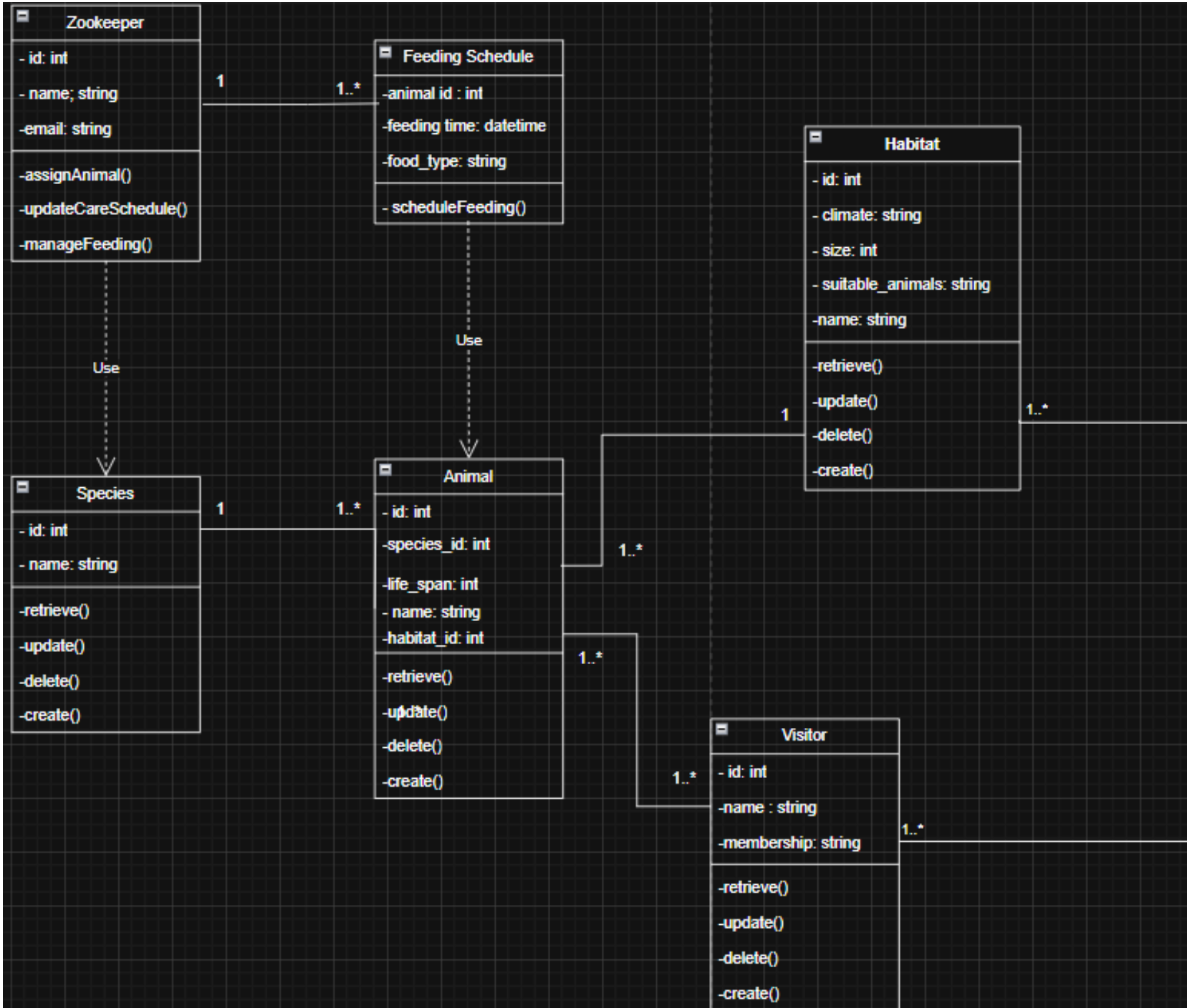
# Sequence diagram



# Sequence diagram



# Class diagram





# Challenges

- **Authentication Issues:** Backend login API not functioning properly.
- **Frontend Blank Pages:** Components disappearing after loading (token and authentication problem).

# Next Steps

- Fix authentication and authorization issues.
  - Ensure the frontend correctly fetches data from the backend.
  - Add error handling for missing or undefined states.
  - Implement data validation & user role permissions.
  - Improve UI/UX for better interaction.
- 