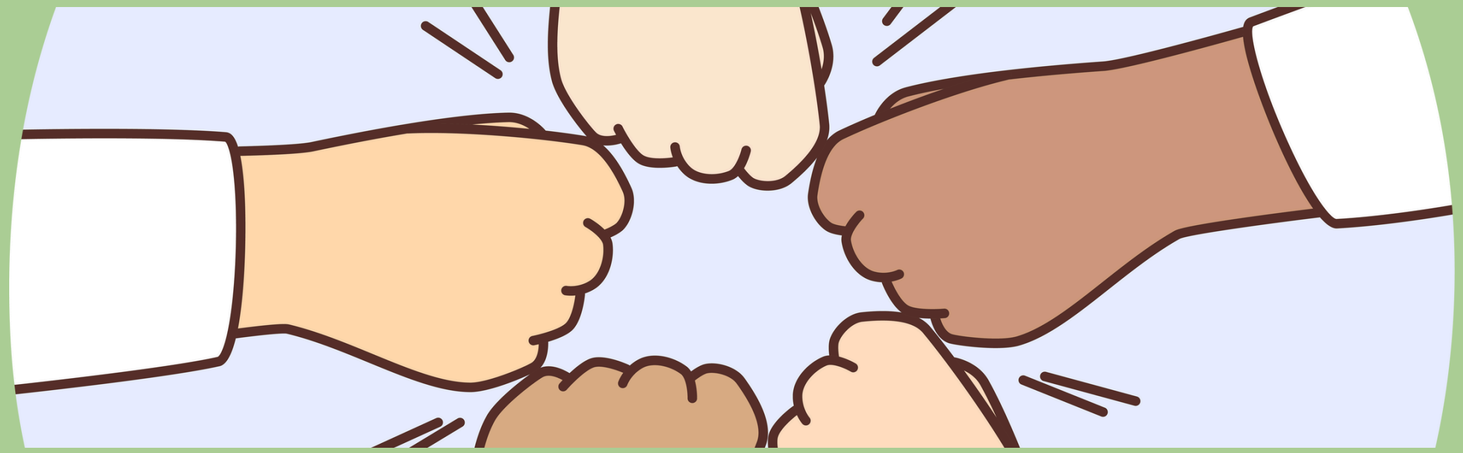
WEATHER BASED IRRIGATION SYSTEM

Team:- Code Wears

Team leader: Man Lakhani

Team Member: Harsh Kathrotiya

Team Member: Miraj Shekhda



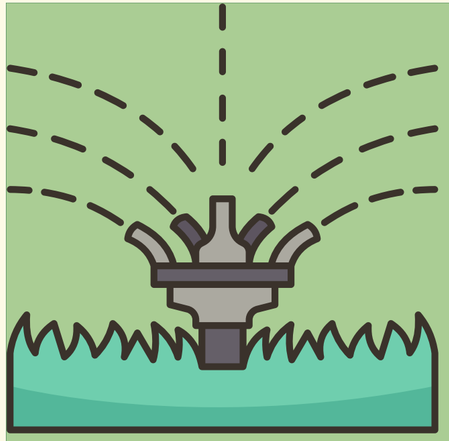
Issues & Threats

Current challenges

Water scarcity and inefficient irrigation are critical challenges in agriculture.

Farmers face difficulty optimizing water usage, causing waste and lower crop yields.

A solution is needed that integrates real-time weather data and soil conditions.



Solution

- Developed a smart irrigation scheduler to optimize water usage.
- Analyzes real-time weather data and soil characteristics.
- Calculates ideal irrigation schedules and sends notifications to farmers.

Proposed System

1)User Interface:

- Farmers can input crop type, soil type, and location.
- Provides a user-friendly interface for accessing irrigation schedules and notifications.

2)Weather Data Integration:

- Connects to weather APIs to fetch real-time data, including temperature, humidity, wind speed, and precipitation.
- Analyzes weather conditions to determine the need for irrigation.

3)Soil and Crop Data Management:

- Stores information on various soil types, their water retention capacities, and specific crop water requirements.
- Associates soil data with different crop types to ensure accurate irrigation calculations.

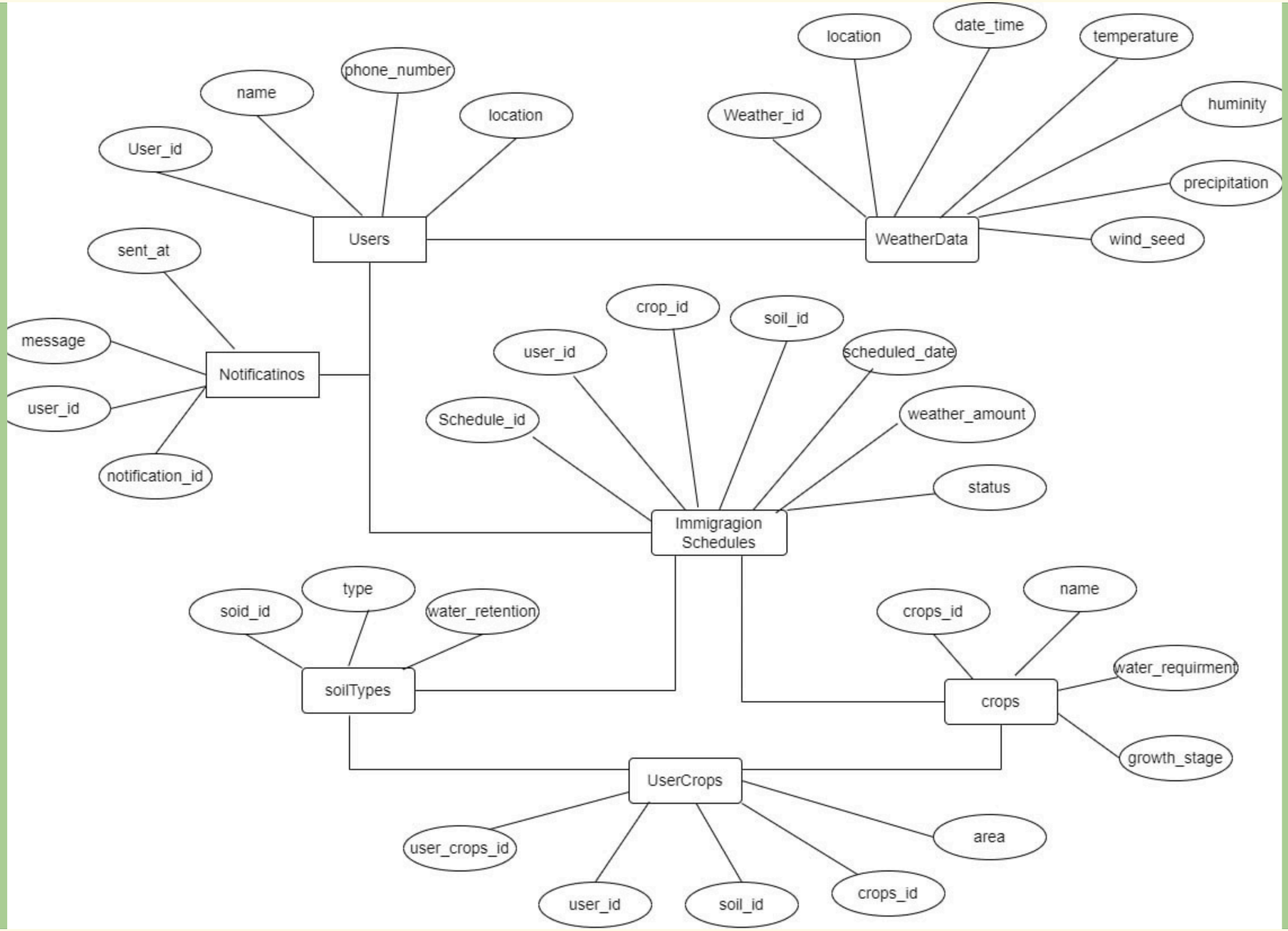
4)Irrigation Schedule Calculation:

- Combines weather, soil, and crop data to calculate the optimal irrigation schedule.
- Generates a schedule that specifies the amount of water needed and the best times for irrigation.

5)Notification System:

- Sends timely notifications to farmers, advising them when to irrigate based on the calculated schedule.
- Notifications are sent via SMS, email, or in-app alerts.

ERD


location
date_time
temperature
phone_number
name
huminity
location
Weather_id
User_id
precipitation
Users
WeatherData
wind_seed
sent_at
crop_id
soil_id
message
user_id
scheduled_date
Notificatinos
Schedule_id
weather_amount
user_id
status
notification_id
Immigragion
Schedules
type
name
soid_id
crops_id
water_retention
water_requirment
soilTypes
crops
growth_stage
UserCrops
area
user_crops_id
user_id
soil_id
crops_id

Technical Details And Implementation

- **Front-End:**

Overview of the main React components (e.g., IrrigationControl, App.js).
Explanation of state management for handling user inputs and displaying results.

- **Back-End:**

Overview of the Node.js server and the Express framework.
Description of the API routes and controllers, including how the weather data is fetched and processed.

- **Weather API:**

Mention the use of the OpenWeatherMap API to fetch real-time weather data.
Explain the logic used to determine whether to turn the irrigation system on or off.

Use Cases and Scenarios

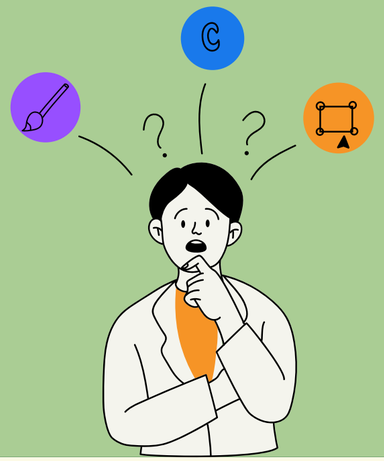
Real-Life Scenarios: Provide examples of how this system can be used in different agricultural settings.

Use Case 1: Farmer in a drought-prone area uses the system to conserve water by automatically turning off irrigation during rainy days.

Use Case 2: A gardener uses the system to maintain an optimal watering schedule for a small garden, ensuring plants receive the right amount of water.

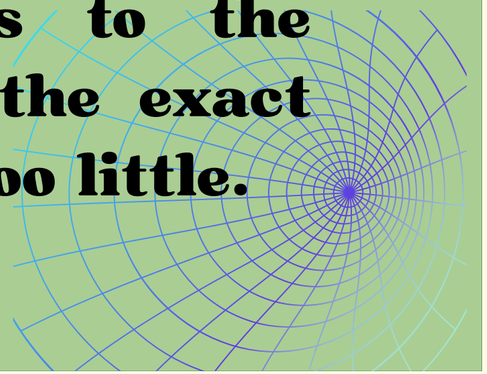
Expected Outcomes:

Water savings
Increased efficiency in water use
Enhanced crop growth and yields



Future Enhancements and Conclusion

- Integration with IoT devices (e.g., soil moisture sensors) for more precise irrigation control.
- Advanced weather prediction algorithms to improve accuracy.
- Mobile application development for easier access and control.
- the system could make real-time adjustments to the watering schedule, ensuring that crops receive the exact amount of water they need, neither too much nor too little.




Thank
you