sddec20-10

Warehouse Inventory Automation

Team Members:

- Amir Hamza (Team Leader)
- Jack Creighton (Backend Developer)
- Jacob Ramsey-Smith (Simulation Developer)
- Stamatios Morellas (Frontend Developer)

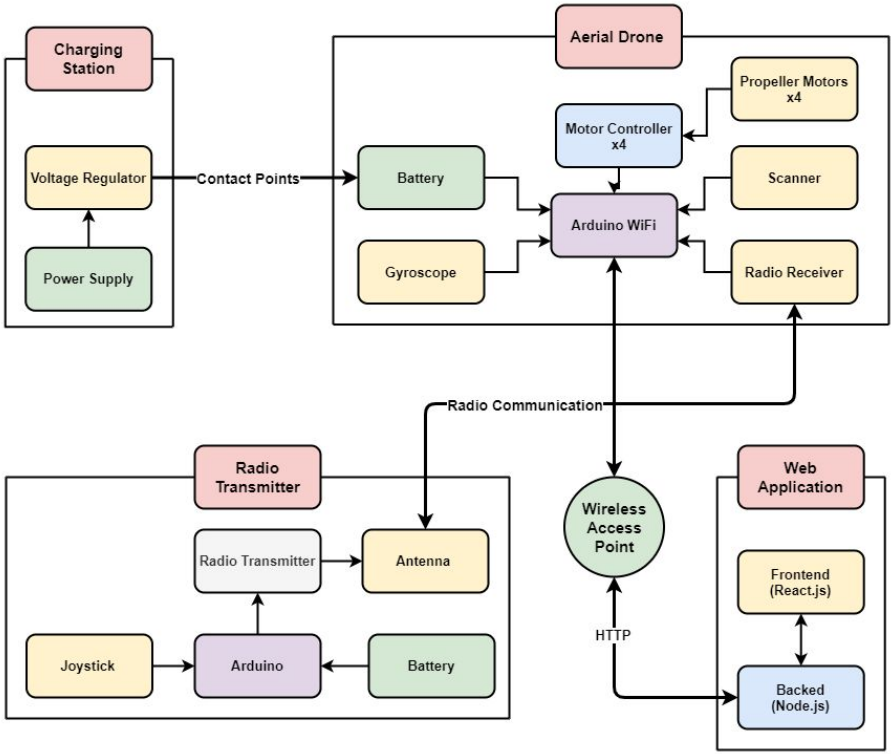
Advisor:

- Dr. Diane Rover

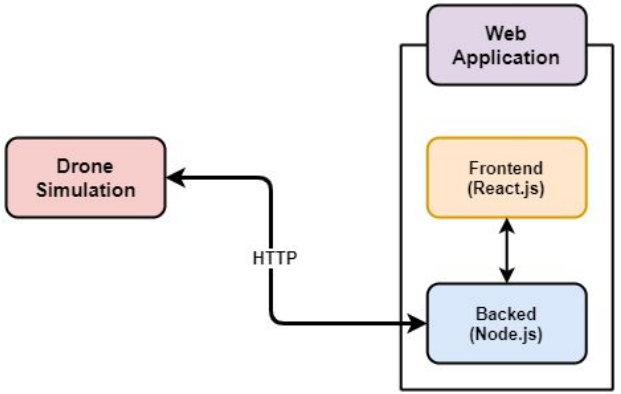
Email:
hr24612@iastate.edu

Problem Statement

Original Project Plan



Revised Project Plan



Non-Functional Requirements

- The drone should have intuitive controls
- Project components communicate quickly and reliably
- The safety of people working in the warehouse must be considered
- A user-friendly, consistent interface for the web application

Functional Requirements

- The drone must be able to scan inventory at all heights of the warehouse
- The drone must be capable of scanning QR codes off of the inventory
- The gathered data from the drone must be transmitted to the server
- Data in the database must be visible to users

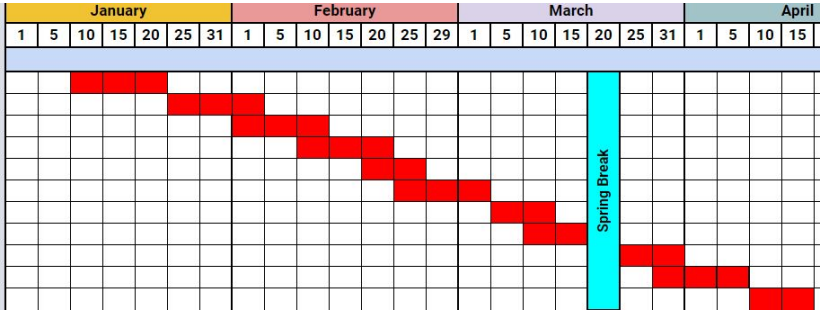
Operational Environment

- Our project simulates a warehouse environment using an application called Webots

Project Milestones and Schedule

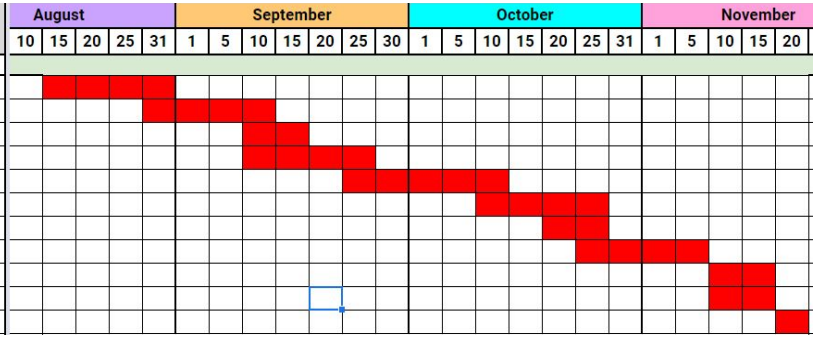
Spring 2020 Schedule

	#	Tasks	Start Date	End Date
Phase A: Project Planning	1	Preliminary Discussion	01/13	01/17
	2	Biweekly status report	01/27	02/02
	3	Lightning Talk	02/03	02/09
	4	Biweekly status report	02/09	02/16
	5	Design Document v1	02/16	02/23
	6	Biweekly status report	02/24	03/01
	7	Lightning Talk	03/02	03/08
	8	Biweekly status report	03/09	03/15
	9	Design Document v2	03/23	03/29
	10	Lightning Talk	03/30	04/05
	11	Biweekly status report	04/06	04/12



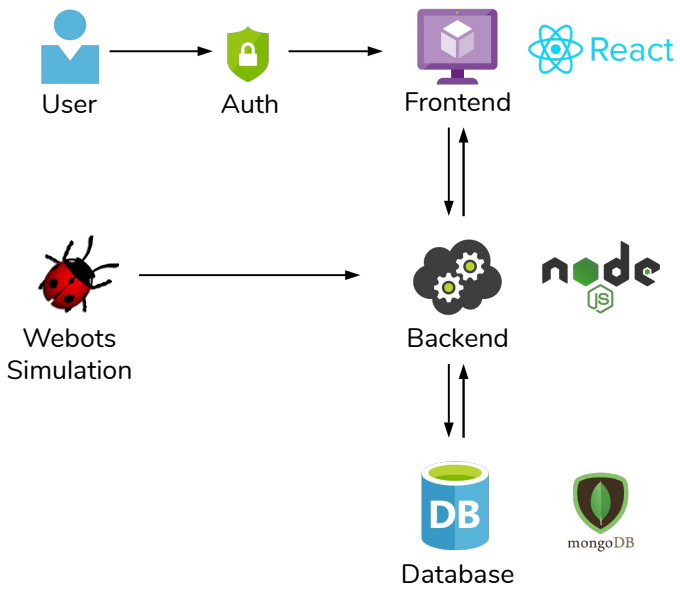
Fall 2020 Schedule

Phase B: Project Build	12	Biweekly status report	8/17	8/31
	13	Biweekly status report	8/31	9/14
	14	PIRM meeting	9/14	9/17
	15	Biweekly status report	9/14	9/28
	16	Biweekly status report	9/28	10/12
	17	Biweekly status report	10/12	10/26
	18	PIRM meeting	10/24	10/29
	19	Biweekly status report	10/26	11/9
	20	Final report	11/9	11/15
	21	Final poster	11/9	11/15
	22	Final presentation	11/16	11/19



Project Design

Project Design Overview



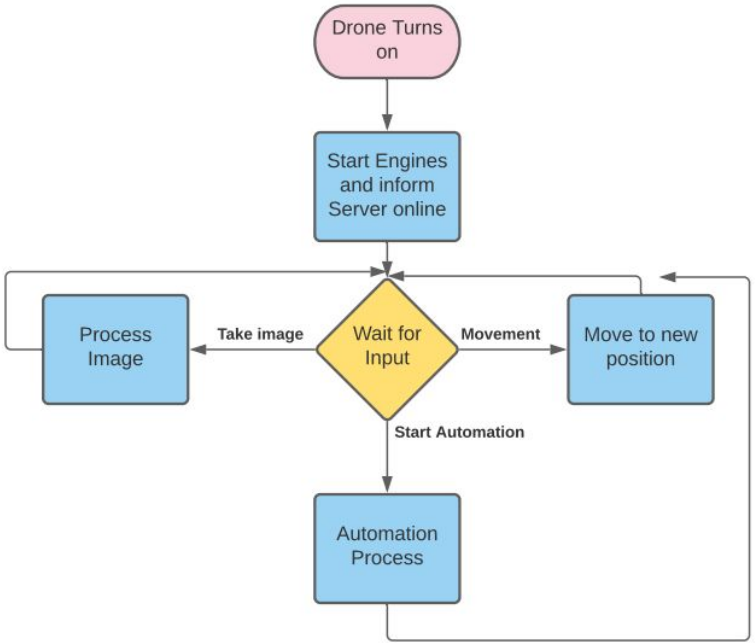
- Our project utilizes a full stack to allow for users to sign on to a web application and view data obtained from webots.
- Main components of system:
  - Webots
  - Client Application
  - Server Application
  - Database

Development Practices

- Agile
- Peer-review
- Model-View-Controller (MVC) architecture
- Side-by-side development (Pair Programming)
- Engineering Standards:
  - IEEE P1936.1: Standard for Drone Applications Framework
  - ISO 18004: Automatic identification and data capture techniques

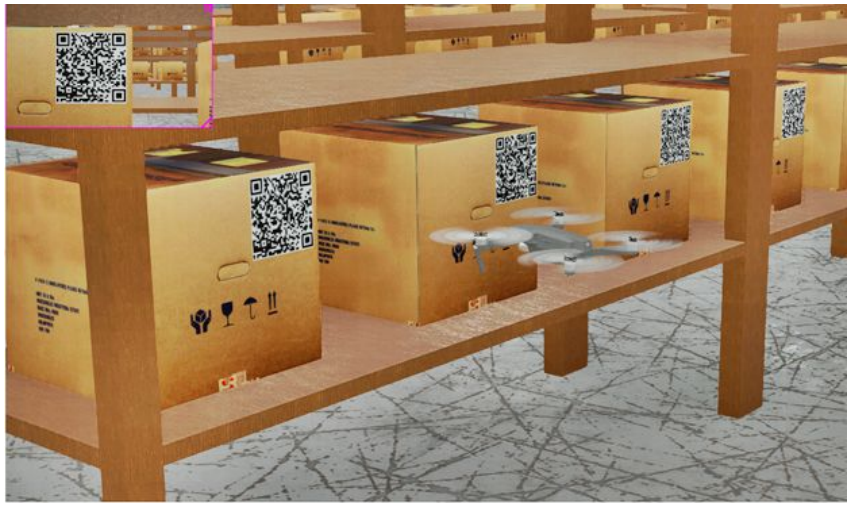
Webots Simulation Implementation & Demo

Webots Design Overview



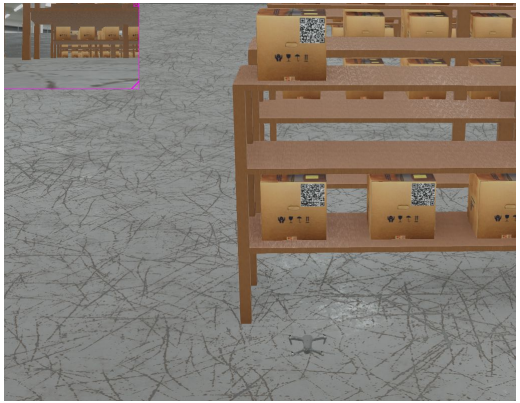
- The team set up a sample warehouse with inventory and QR codes
- A drone was used to fly around and scan the QR codes
- An autonomous program was created for the drone to assist in the efficiency in the scanning of inventory

Webots



- Scanning a QR code
  - The drone is equipped with a camera, and takes pictures of the QR codes on inventory
  - That picture is then scanned using Google's Zebra Crossing library
  - After the image is successfully processed, the data obtained from it is put into a JSON object and sent to the backend server in an HTTP request
- Automation process
  - The drone starts by scanning the QR code in front of it.
  - The drone then moves 5 boxes over, stopping at each on to ensure a successful scan of them
  - The drone then accounts for the gap in the shelves and repeats
  - The automation process can be cancelled and reset if needed.

Webots Challenges



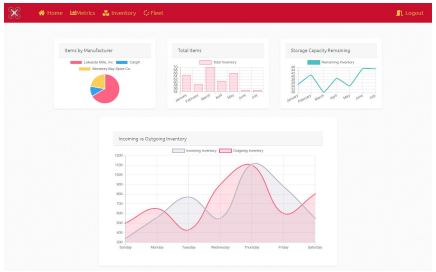
- Transitioning to a real environment
  - The maximum distance a successful scan was obtained: 1.7 meters
  - The maximum distance successful scans were consistently obtained: 1.4 meters
  - Drone was capable of successfully capturing image in non-ideal lighting

Webots Demonstration

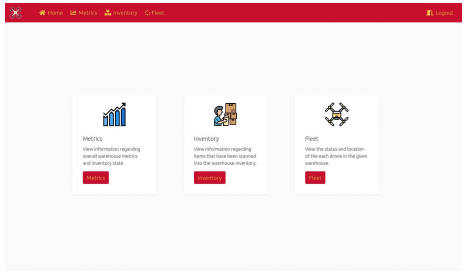
Web Application Implementation & Demo

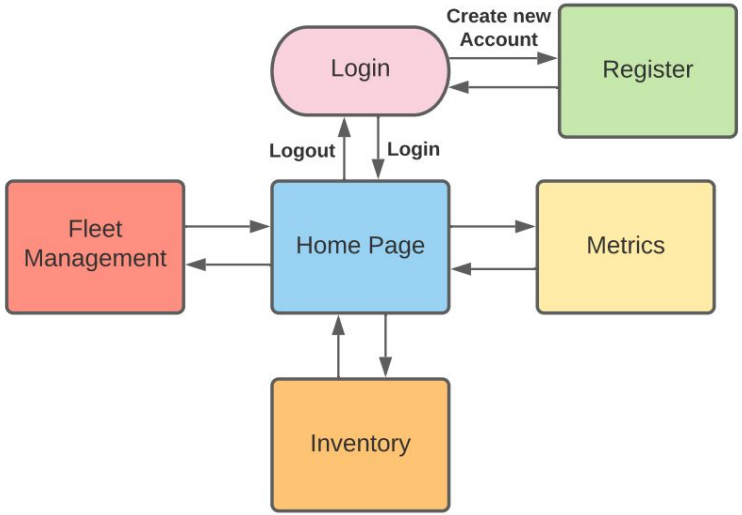
Website Design Overview

Metrics Page

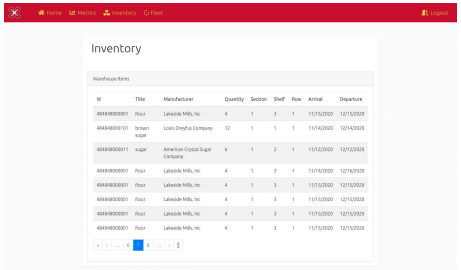


Home Page

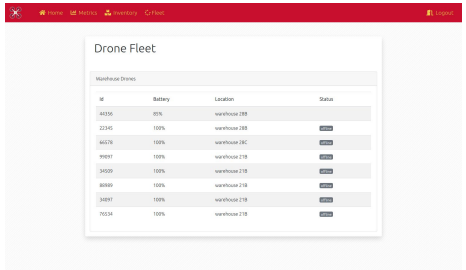




Inventory Page



Fleet Page



Technologies Used

Webots:

- Java

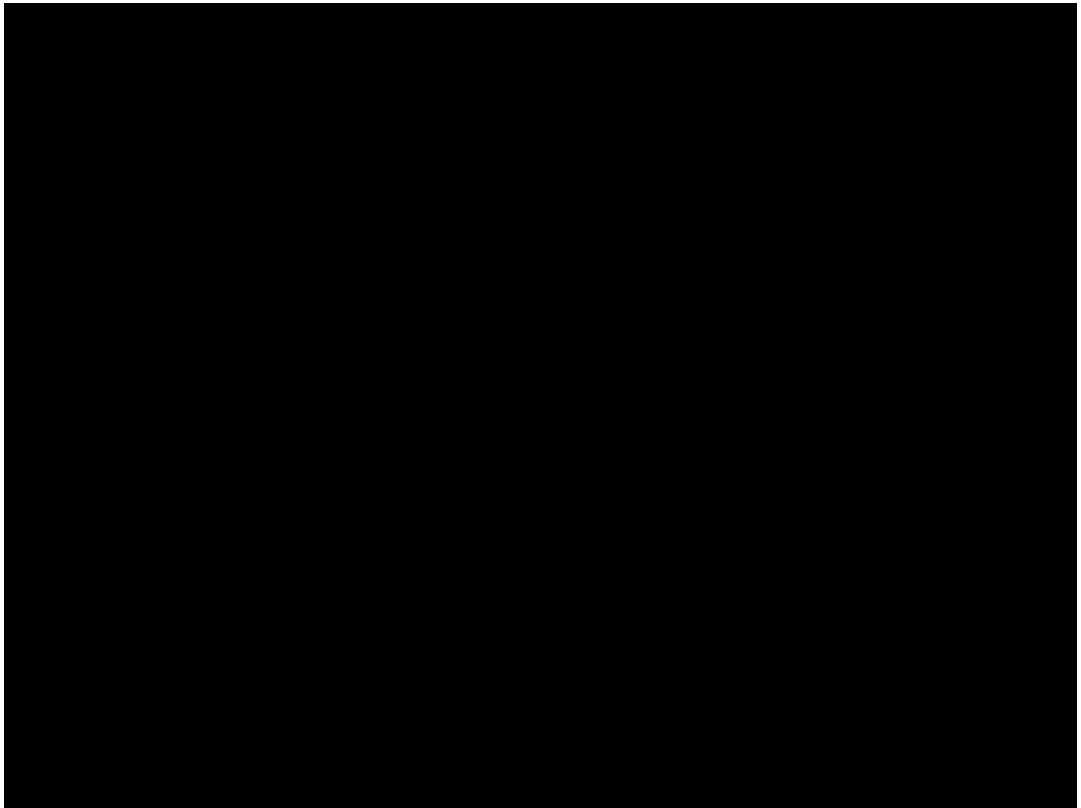
Web Application:

- MongoDB
- Express
- ReactJS
  - Core UI
- Node.js

Web Application

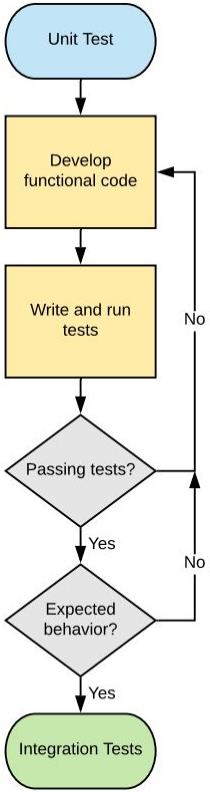
- Client Application
  - Adobe Xd - Mockups/Wireframing
  - Bootstrap - CSS library
  - CoreUI for React - UI Components Library
- Server Application
  - Express - Web framework for Node.js
  - Mongoose - MongoDB object modeling for Node.js
- Database
  - MongoDB - Document-based cloud database

Web Application Demonstration



Testing

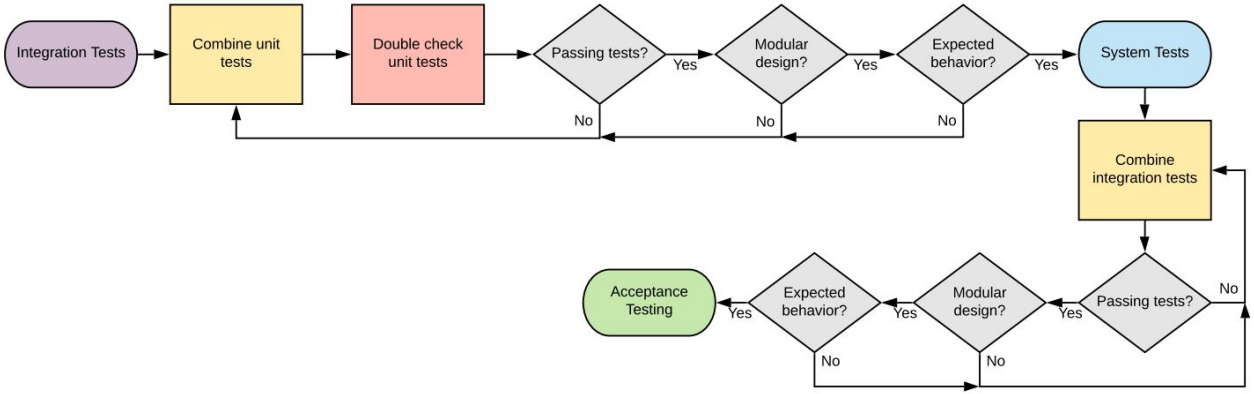
Unit Testing



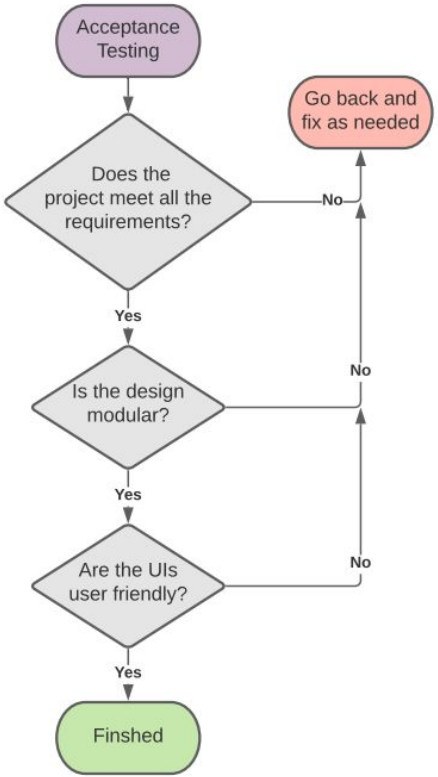
- Unit tests were performed throughout the project to ensure functionality of each component before proceeding to use the component in the full system
  - Querying and saving data to the database
  - Testing the Zebra Crossing library with sample pictures
  - Testing the camera's ability to capture and save images.

Integration and System Testing

- Integration testing was performed to check if the unit tests functioned together.
  - Testing that the QR code scanning works in the Webots environment
- System testing was performed to ensure that different modules in our project work together.
  - Putting data from Webots into the database and using that in the front end



Acceptance Testing



- Acceptance testing was performed to ensure that our project meets are requirements and is a modular design
  - Changing front end to be more intuitive to first time users
  - Cleaning code to be more readable and more modular in design

Conclusion

Conclusion

- Main Goal → Increase warehouse efficiency through automated inventory management
- Original Plan → Fully autonomous, physical drone
  - Not feasible due to time and remote work constraints
- New Plan → Transition select components to a digital environment
  - Full autonomy → Partial autonomy
  - Viable alternative
- Challenges
  - COVID-19
  - Learning new technologies
  - Time management

Questions?