Autonomous Inventory Tracking Drone
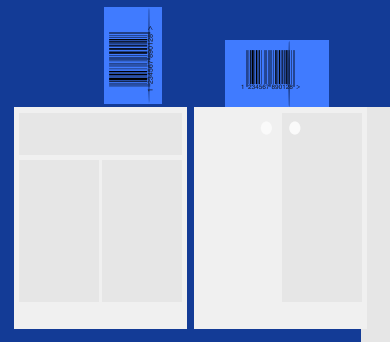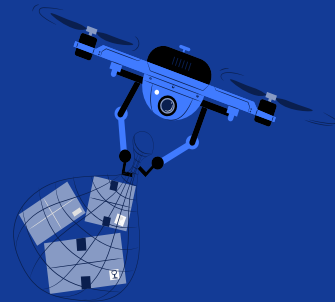
Paul ¦ Shiv ¦ Sai ¦ Steven ¦ Ashan ¦

OVERVIEW

01 **Problem Statement**
Project Overview and Objectives

02 **Background and Research**
Comparison to Existing Ideas | Use Cases

03 **Product Overview**
Initial Concepts to Final Product

04 **Engineering Design and Analysis**
Design and Architecture Choices | Flow Diagram

05 Design and Coding
Translate requirements & data models into solution
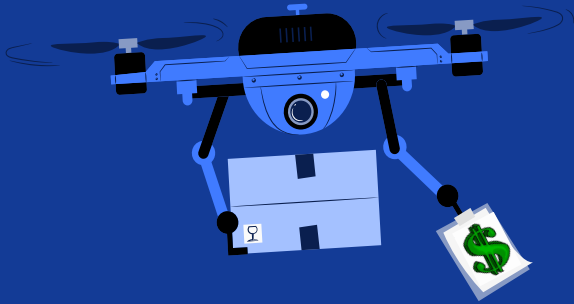
06 Demo
Video Demonstration

07 Product Testing Approach
Integration Testing | Acceptance Testing | Verification and Validation

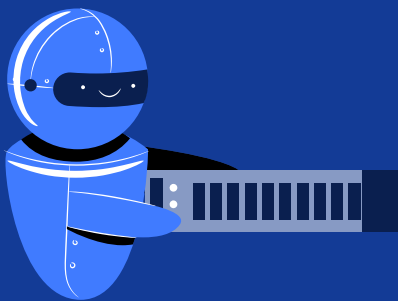## 08 Lessons Learned
Successes | Improvements

PROJECT OVERVIEW AND OBJECTIVES

➢ Designing a safe and reliable solution for inventory tracking using a drone
➢ Save companies time and money
➢ Improved accuracy with minimal errors
➢ Objective is to make our solution the new norm for inventory tracking
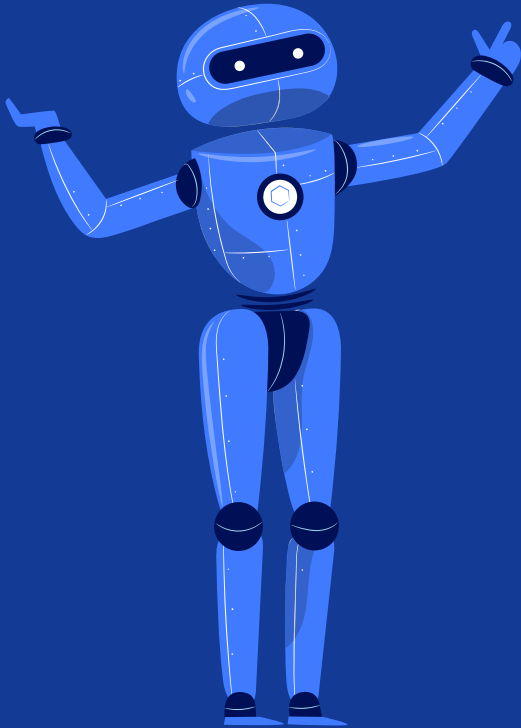
02 Background and Research

COMPARISON TO EXISTING SOLUTIONS

	Current Solutions	Pros	Cons
PINC (RFiD)	1 PINC™ [2]	Accurate and fast solution	Expensive Prone to interference
Flytbase (barcode)	2 flytbase [3]	Ability to scan barcodes off the boxes	Expensive Slower solution compared with RFID tag

OUR SOLUTION

Autonomous Inventory Tracking Drone (AID)

$300

Pros:
- ➤ Adaptability to warehouses
- ➤ No change to infrastructure or existing labelling system (qr code, barcode)
- ➤ Minimum supervision is required
- ➤ Cost and time efficient

Cons:
- ➤ Slower to process all products
- ➤ High battery consumption, less fly time
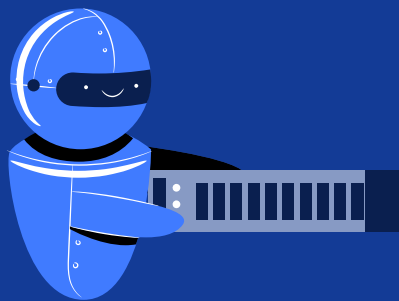- ➤ Job cuts
- ➤ System Security

USE CASES

➤ Drone can scan
  ○ Barcodes
  ○ QR Codes
➤ Maintaining an up-to-date database of the inventory
➤ Safe usage with minimal errors
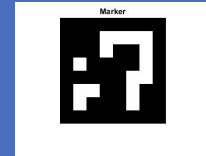
03 Product Overview

Technical Specification

➤ Python

➤ Tello SDK
(https://github.com/dji-sdk/Tello-Python)

➤ OpenCV
  ○ ArUco Markers
  ○ Pyzbar

➤ Lib h264 Decoder

➤ WiFi and TCP/IP
(192.168.10.1)

[1]

Marker

[5]

FLOW DIAGRAM

INVENTORY   DRONE   WIFI   PYTHON PROGRAM   FIREBASE
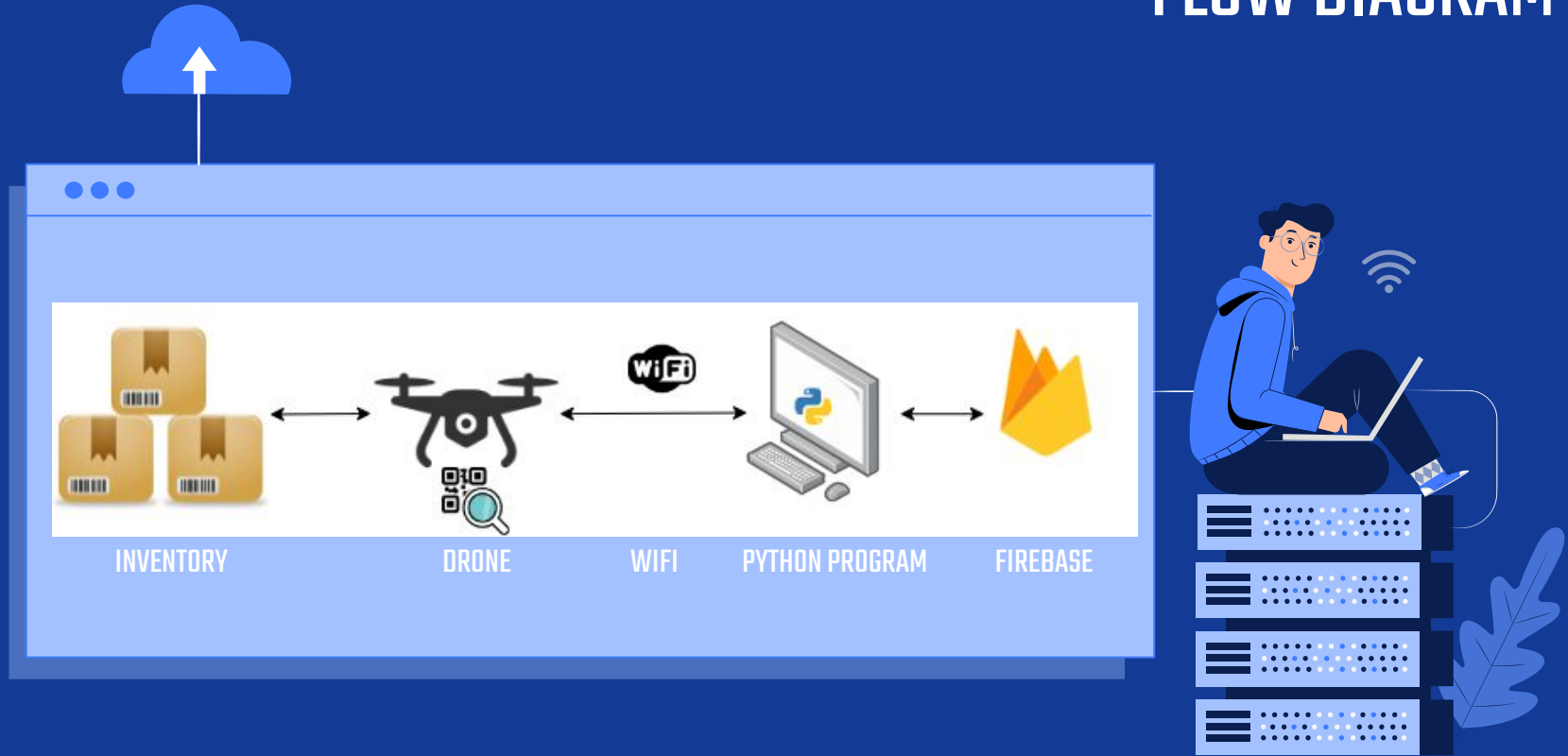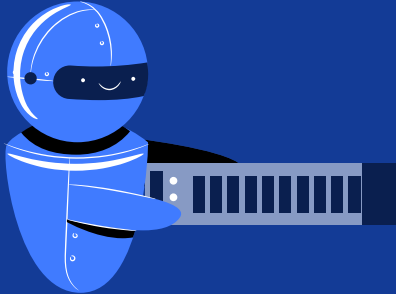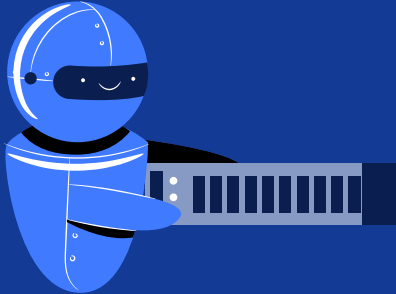
04 Engineering Design and Analysis

Software Engineering Process

- ➤ Requirements Gathering
  - ○ Identifying problem and conceptual solutions
- ➤ System Design
  - ○ Hardware
  - ○ Algorithms
- ➤ Implementation
- ➤ Testing
  - ○ Integration testing
  - ○ Acceptance testing
- ➤ Verification and Validation
  - ○ Performed through acceptance testing
  - ○ Fixing any bugs

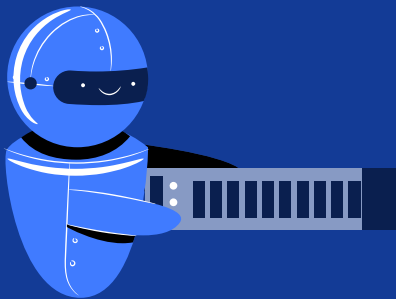05 Design and Coding

Design and Development Phase

➢ End result of design and coding:
  - Data and architectural design
    - Preliminary design specification
  - Procedural design
    - Prototype
    - Detail design specification
  - Coding
    - Program source code

06 Demo

Demo
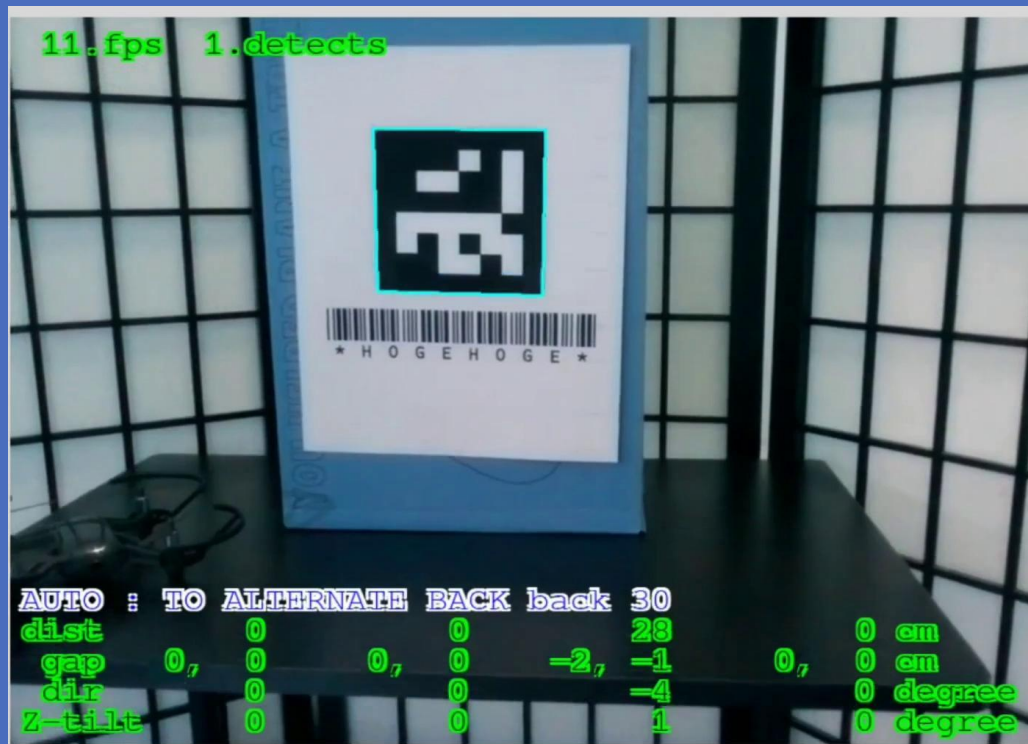
07 Product Testing Approach

Integration Testing

Sequential Tests that allow the flow of logic to be analyzed

- ➤ Flight and Movement Test
  - ○ Validates if the entire system functions coherently

- ➤ Ability to capture images and videos
  - ○ Tests the logic and validates if the camera transmits and streams videos onto the host computer

Acceptance Testing

Acceptance Test Driven Development

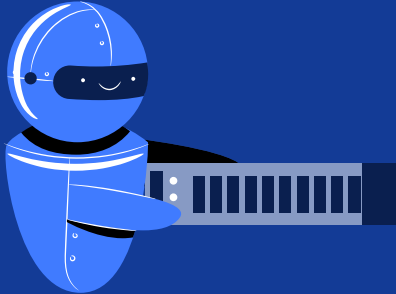➢ Agile development practice
➢ Acceptance tests
  ○ Connection, Flight and Movement Test
  ○ Image Capture and Videos Test
  ○ QR and barcode detection
  ○ Overall Acceptance Test

Test Writer:	Sai Pedamallu					
Test Case Name:		Overall Acceptance Test		TestID#:		ACCPT-Test-01
Description:		Test validates if the entire system functions coherently and how it performs during certain cases. The drone should successfully be able to establish connection with the companion computer and maintain the WiFi connection throughout the flight time. Furthermore, The drone should be able to fly and move based on the programmed path and objectives as defined by the user without crashing or failing during flight.		Type:		Black Box
Tester Information						
Name of Tester:		Sai Pedamallu, Paul Femi-Gege		Date:		
Hardware Version:		Version 1.0.1		Time:		
Setup:		Once the drone is switched on, and a connection is established with the companion computer, the drone will need to be kept in a well-lighted space (starting point). Next step is to place the barcodes in the room. Once the previous steps are completed, the Python script main.py will need to be executed. "Autopilot" will need to be clicked on the user interface to let the drone automatically navigate around the room and find the barcodes.				
Test	Action	Expected Result	Pass	Fail	N/A	Comments
1	Press power button	Drone powers up				
2	Establish connection with the companion computer over WiFi	Drone establishes a connection with the companion computer				
3	Run main.py Python script on the companion computer	Drone will enter SDK mode and then takeoff. Once it takesoff, Autopilot mode can be enabled from the UI. Drone will start moving around the room and scan barcodes.				
4	Drone returns to original position and lands	Drone lands on the landing pad.				
Overall Test Result:						

Verification and Validation

➢ Ensure that the system meets customer needs
  ○ Performed through acceptance testing
➢ Ensure that the system is well-engineered and error-free.
  ○ Performed through debugging and resolving issues and defects

08 Lessons Learned
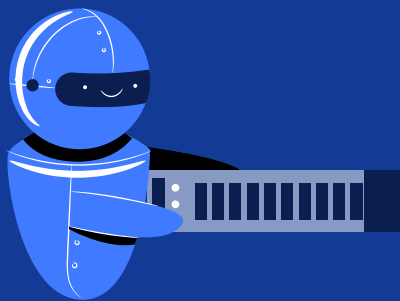
LESSONS LEARNED

What did we do well?

➤ Entire team was committed
➤ Worked well together

What could we improve?

➤ Further research into the parts and products we plan on using

Mistakes to avoid

➤ Wrong estimation of time
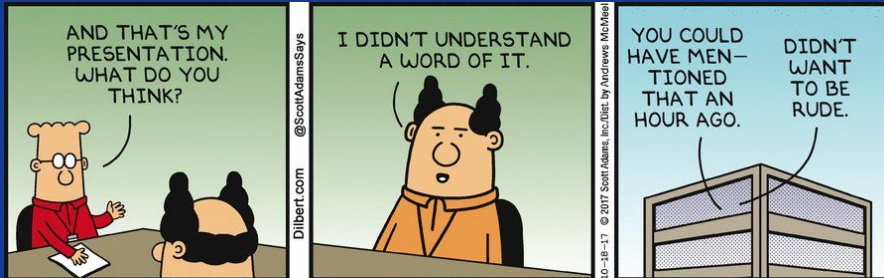➤ Not using a project management tool

What should we continue to do?

➤ Team meetings to ensure everyone understands objectives and deliverables

Thanks!

Does anyone have any questions?

REFERENCES

[1] https://www.thegoodguys.com.au/dji-tello-drone-djitello

[2] https://www.supplychain247.com/images/article/PINC-logo-600trans.png

[3] https://s3.amazonaws.com/i3.cleantech/uploads/company/company_logo/41/404741/Flyt-base-logo-1.png

[4] https://i.pinimg.com/originals/51/93/d7/5193d73736de98c89ed6fe2e6d9c86fd.gif

[5] https://user-images.githubusercontent.com/1608945/61642578-89882f00-ac77-11e9-91ae-1a15585aa97f.png