



# **SWARMS FOR ADVERTISING**

\_\_\_\_\_ Aneesh Chodisetty (117359893) \_\_\_\_\_

Bhargav Kumar Soothram (117041088)

Joseph Pranadeer Reddy Katakam  
(117517958)

# AGENDA

- 01 Motivation
  - 02 Project Overview
  - 03 Implementation
  - 04 UML Diagrams
  - 05 Running Code
  - 06 Demonstration
  - 07 Challenges
  - 08 Future Work
- 



# MOTIVATION

- There is a recent trend in aerial shows which is the usage of swarm robots, swarm drones to be precise.
- The swarm forms different shapes in the sky, these shapes and the lights are very entertaining and grabs the crowd's attention easily.
- We can use this aspect of swarm robots for advertising.
- Imagine a swarm of drones that form shapes of your product in the sky during a concert or a parade.





# PROJECT OVERVIEW

- Using swarm robots to bring a new paradigm to advertising
- These autonomous robots perform a safe, collision-free navigation using controlled localization and formation by appropriately communicating with each other
- Our goal is to simulate this scenario with a swarm of ground robots, using ROS and Gazebo



# IMPLEMENTATION



## DEVELOPMENT METHODOLOGY

- Agile software development model was used.
- Designed in a Test-Driven Development fashion.
- Implemented using Pair programming technique.
- AIP backlog sheet and Sprint notes have been well documented.

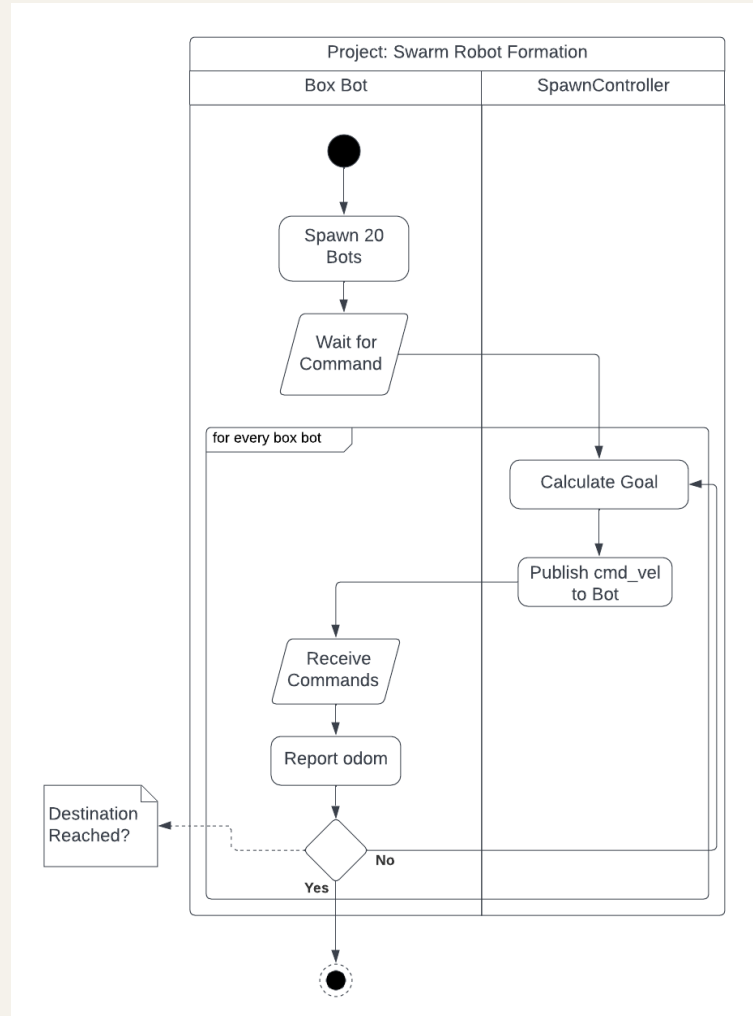
## SYSTEM ARCHITECTURE

- UML diagrams help lay the groundwork to get an overview of the software development.
- UML Class Diagram.
- UML Activity Diagram.

## SOFTWARE DEVELOPMENT

- The latest version of ROS2 has been used for designing the code.
- Gazebo has been used as the simulating environment.
- 20 'Box Bots' have been spawned in a empty Gazebo World.
- Different nodes communicate with each other to make the spawned bots form a desired geometric shape.

# ACTIVITY DIAGRAM



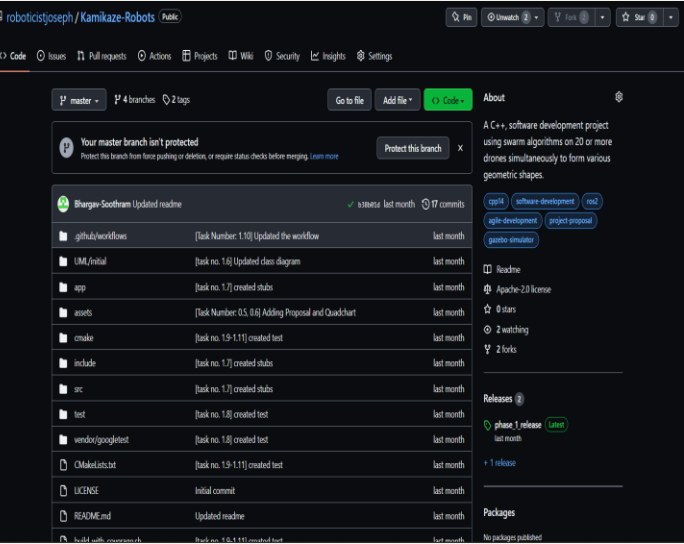
# SOFTWARE DEVELOPMENT PROCESS

## GITHUB

## AGILE ITERATIVE PROCESS

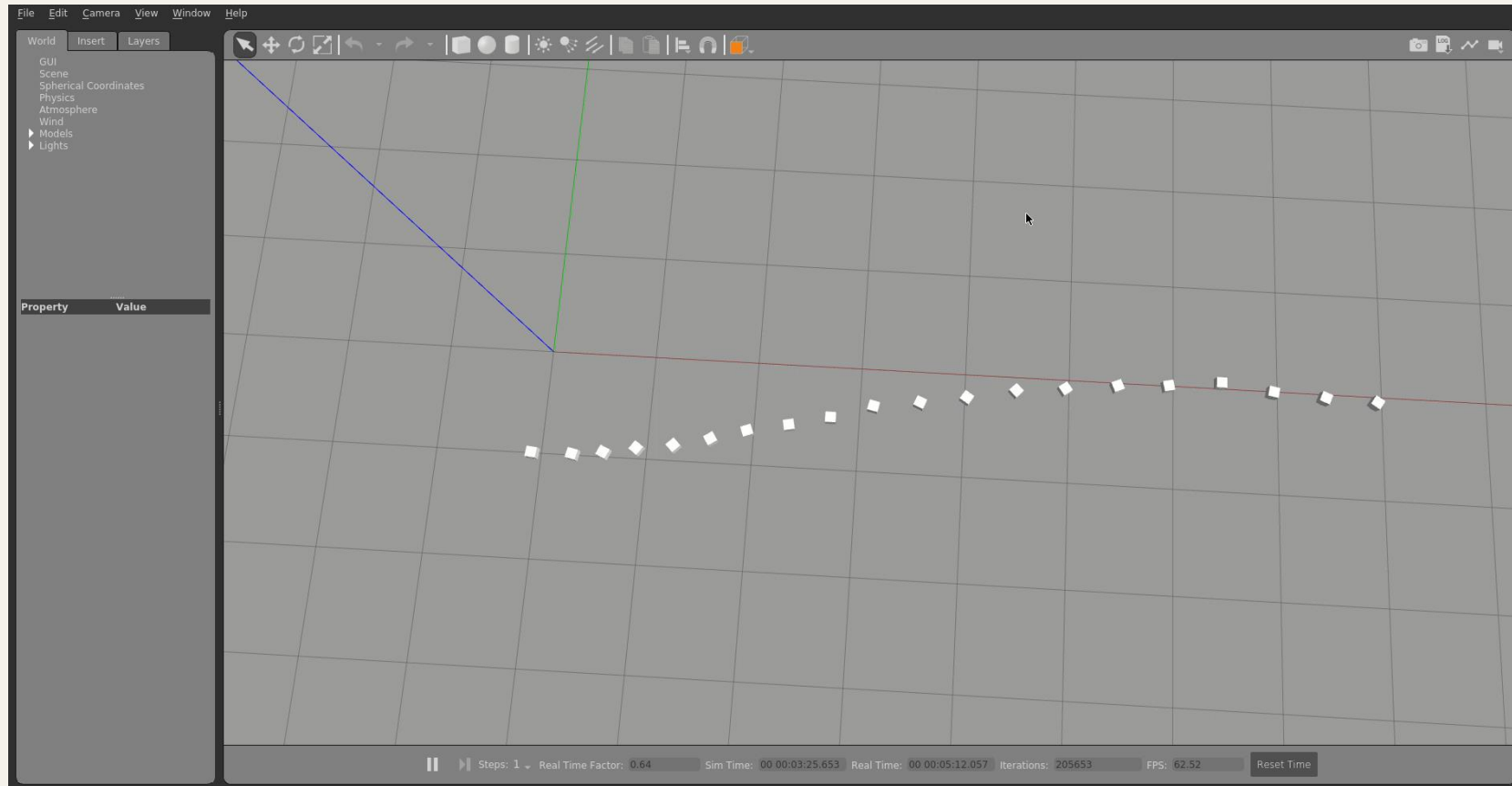
## AIP BACKLOG & OTHER DOCS

- We followed AIP for creating the project.
- Along with AIP we used pair programming methodology.

A screenshot of a spreadsheet titled 'Kamikaze Drones Product Backlog'. It contains a detailed list of tasks, their status, and their completion progress. The tasks are organized into columns: 'Task', 'Status', 'Estimated Time (minutes)', 'Actual Time taken for Iteration 1', and 'Time difference'. The tasks are color-coded: green for completed, yellow for in progress, and red for not started. The spreadsheet is titled '[Group 9 - Anesh Chodisetti, Bhargav Kumar Sothram, Joseph Katakam] Kamikaze Drones Product Backlog'.

# DEMONSTRATION

Link: [https://drive.google.com/file/d/1n2\\_pBgozSkXypZfkp2xUOTT\\_614eTAn/view?usp=share\\_link](https://drive.google.com/file/d/1n2_pBgozSkXypZfkp2xUOTT_614eTAn/view?usp=share_link)





# CHALLENGES

- No resources in ROS2 for swarm robots.
- Spawning multiple Robots.
- Coordinating each robot's location with respect to the other.
- Compatibility issue with ROS versions and packages.
- GitHub CI integration.

# FUTURE WORK

## Introduce Drone Swarms

Adapting the algorithm to drones

Spawning multiple drones.

## Use Boids Algorithm

Implement Boid's algorithm to simulate "flocking" behavior

## Move to 3D space

Adapt the algorithms to work in 3-dimensional space



Any Questions?

**THANK**

**YOU**