

# Team 22: Alpha Demo Day

---

By Bowen Pickrell, Jaret Smith, Franklin Lassandro, Christian Hartman, Aiman Zoqi,  
Saumya Rajasekaran, & Manuel Salazar.

# WeGo Transportation as a Service (TaaS) Platform – Stakeholders Strategic Framework

v1.2 2/8/24

## Strategic Goals

### Where am I and where am I going?

#### ➤ Mission

- identify and scale a suite of innovative transportation services.

• **Alliances:** cloud services, industry experts, autonomous vehicle/airborne drone suppliers, emerging technologies

• **Target Markets:** specialized, differentiated, innovative payload delivery services (where the payload is not human beings). WeGo's short term interest is autonomous vehicles navigating public roads... however, WeGo wants to explore several airborne drone services.

• **Agile-project approach:** creative, self-managed teams to trail blaze and systematically demonstrate stakeholder value.

**Business Dev. Steering Committee**

## SWE Best Practices & Tools

### How will my team get there?

- Incremental Stakeholder Value
  - Project Roadmap → Tailored Agile-Scrum
- Adopt Proven SWE Best Practices
  - Platform Canvas frames TaaS innovation
  - Focus on design: User Centered Design optimizes user experience, Sequence Diagram offers a blueprint for incremental system buildup
  - Strategic, maintainable software over tactical programming
  - Separation of concerns: multi-cloud system and OO applications
  - Self-managed scrum teams demonstrate stakeholder value at the end of sprints.
  - Demonstrations, checkpoints, KPI's & feedback drive continuous improvement
- Architecture Council Decisions
  - Technology: React, Multithread Vehicle Sim, REST API's
    - Demand Stack: DO Nginx/SQL/Py
    - Supply Stack: DO Nginx/MongoDB/Py
  - Tools: BitBucket, Postman, Slack, Google Drive Doc Repository, Trello, VS Code

**Architecture Council, Product Owner & PMO**

## Solution

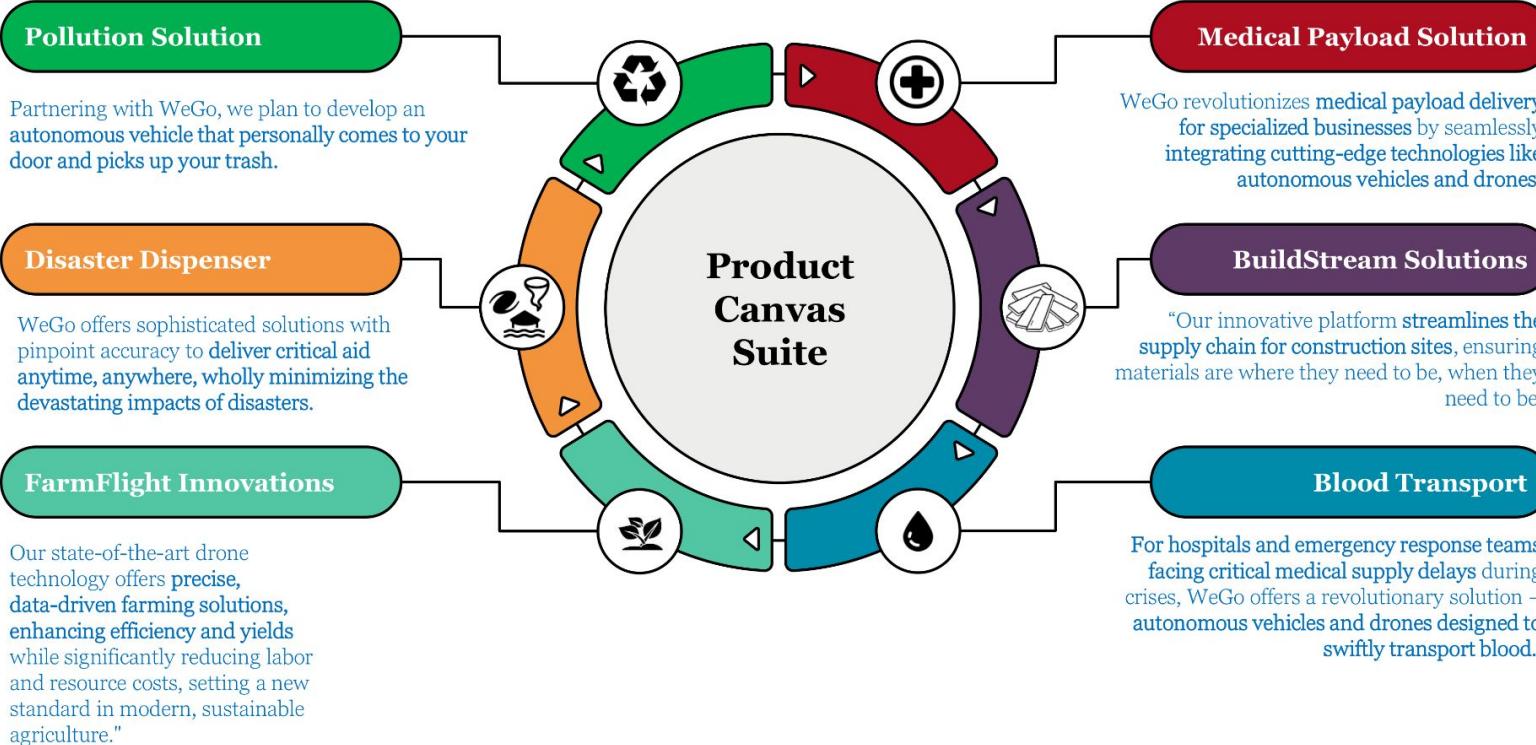
*Design, Build, Test and Demonstrate the Platform.*

*Solution implemented using Google Maps platform, CI/CD, Flask Libraries, Figma, Draw.io, Bootstrap, React + Vite,*

### TA Scorecard Metrics:

- TA Scorecard 1: 7
- TA Scorecard 2: 21
- TA Scorecard 3: 40

**SWE Scrum Team**



**Sprint 0**  
**Foundation**  
**1/15/24 (2w)**

- 1. Student Journeys & Course Value
- 2. Self Managed Team
- 3. WeGo Project Concept
- 4. Product Canvas
- 5. Solution Architecture and Design
- 6. User Centered Design
- 7. SWE Best Practices

**Sprint 1**  
**MVP Release v1.n**  
**1/29/24 (2w)**

- 1. Team Evolution
- 2. Demo Day Obj. + Agenda
  - Product Canvas Suite
  - System Req/Design
  - UI Extensibility & Value
  - OO Design/Build/Test
  - Operational SW
    - \* Cloud Srvcs
    - \* FE/BE/DB integration
      - \* Map Services
      - \* SW Vehicle Sim
- v1.x
- SWE Best Practices

**Sprint 2**  
**MVP Demo Day v2.n**  
**2/12/24 (1w)**

- Refine Stakeholder References
- Plan Demo
  - \* Risk mgmt.
  - \* 2/20 Dry run
- Sprint 3 plan
- **2/22 Demo Day (30 minutes)**
- **Nontechnical Stakeholders**

**Sprint 3**  
**Multi-cloud v3.n**  
**2/19/24 (2w)**

- Team Evolution
- Evolve System Req., Design, Build, Test
- Sys Arch
- Sequence Diag.
- REST API(s)
- Fleet Manager
- Evolve. OO Applications
- Design/Build/ Test
- Common Srvcs.
- Plug In(s)
- Black Box Test

**Sprint 4**  
**PoC Integration**  
**3/4/24 (3w)**

- Spring Break
- NoSQL Supply Cloud
- Vehicle Req. Design, Build, White+Gray Box & Simple Int Loop
- FM Ops Panel v4.0
- VSIM Multithread
- TA Scorecard Actions
- Integration Buildout/Test
- Plug-ins Int. Value
- Nonfunctional Req.

**Sprint 5**  
**PoC Demo Day 3/25/24 (1w)**

- TA Scorecard #2
- UAT
- Update S6, S7, S8 backlogs
- Vehicles Status API
- Team Evolution
- Integration Buildout/Test
- 4/11 3pm: TA Demo Day
- Easter Break

**Sprint 6**  
**Multi-cloud v6.n**  
**4/1/24 (2w)**

- Continuous improvement
- Incr. Design, Build & Test
- Payment Services
- V Maint.
- Multiple Fleets
- FM Ops Panel v6.0
- Adv DO Srvcs.
- 4/11 3pm: TA Reunion

**Sprint 7**  
**Alpha Release**  
**4/15/24 (2w)**

- System Tests: e.g., Ethical Hack, Load Balancing...
- TaaS Plug-in timeline
- **Feedback TaaS**
- Publish Alpha Release
- TA Scorecard #3
- UAT
- **Demo Day Value Targets + Agenda**
- 4/23, 4/25 (Zoom): SWE Insights
- Sprint 7 demo video: T21 only

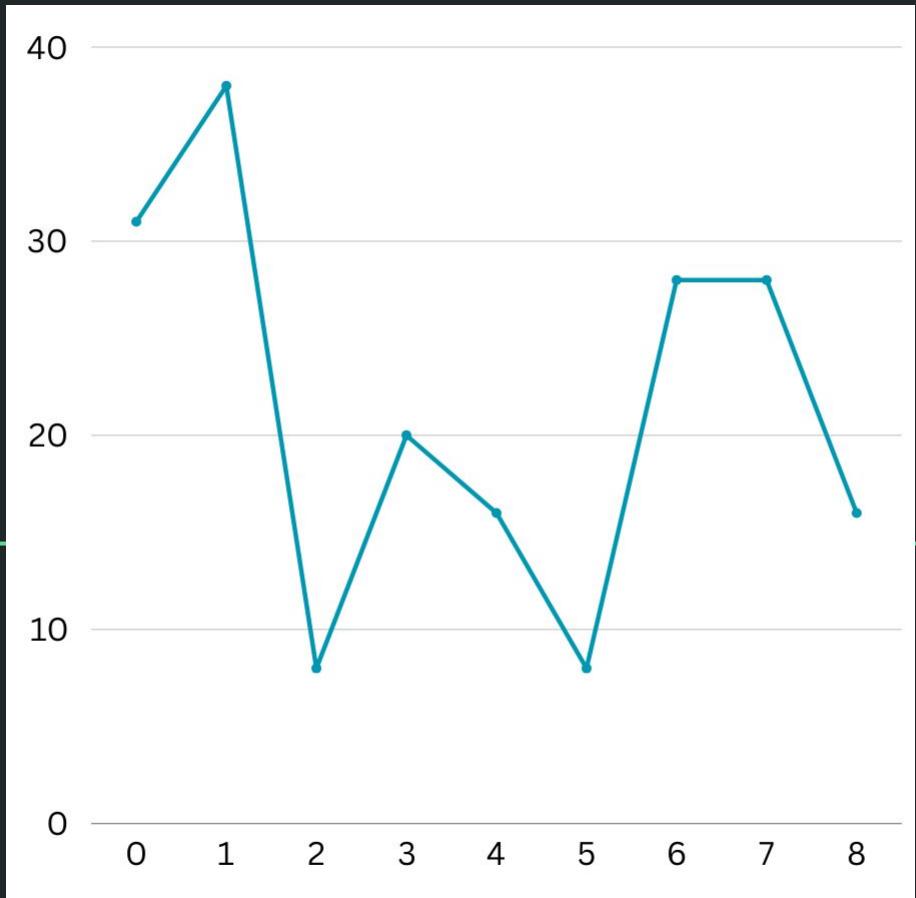
**Sprint 8**  
**Alpha Demos + UAT**  
**4/29/24 (1+ wk)**

- Th 5/2 (Zoom): Demo Dry Run
- Th 5/2 (Zoom), Class window 11am to 12:15pm:  
**T11 Demo + UAT**
- Th 5/2 (Zoom), Class window 12:30pm to 1:45pm  
**T21 Demo + UAT**
- Section 1 Exam window → **Mon 5/6** 11am – 12:30 Zoom:  
**T12 Demo + UAT**
- Section 2 Exam window → **Wed 5/8** 11am – 12:30 Zoom:  
**T22 Demo + UAT**

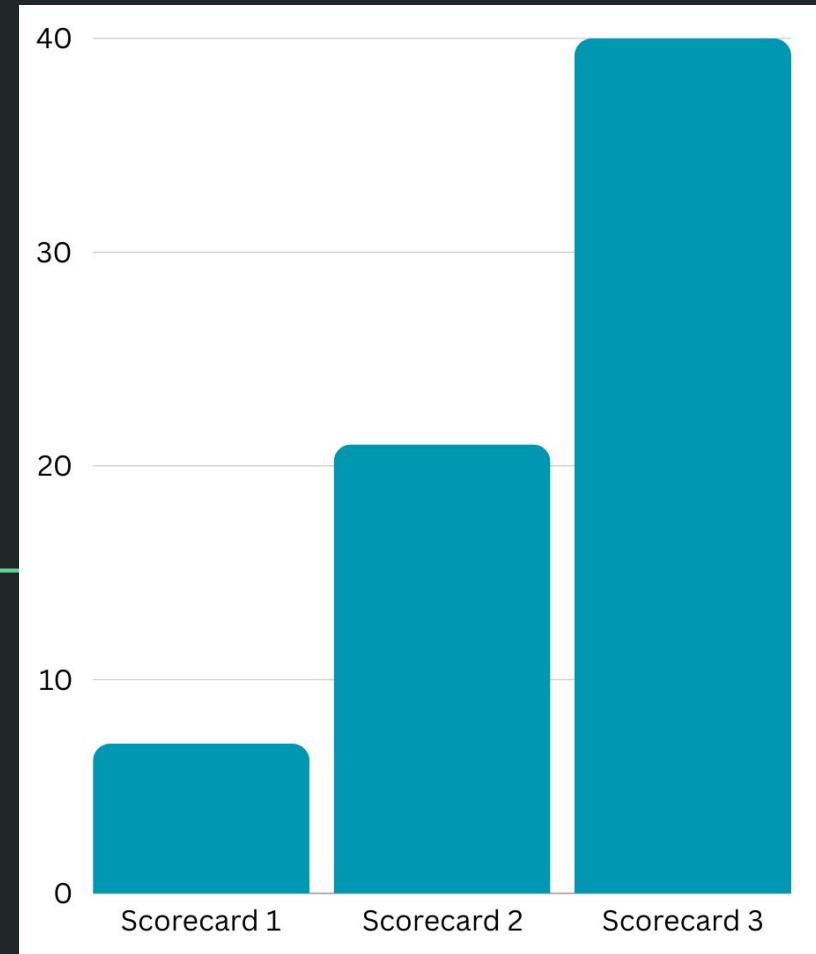
V7.2 4/16/2024

## Transportation as a Service (TaaS) Project Roadmap

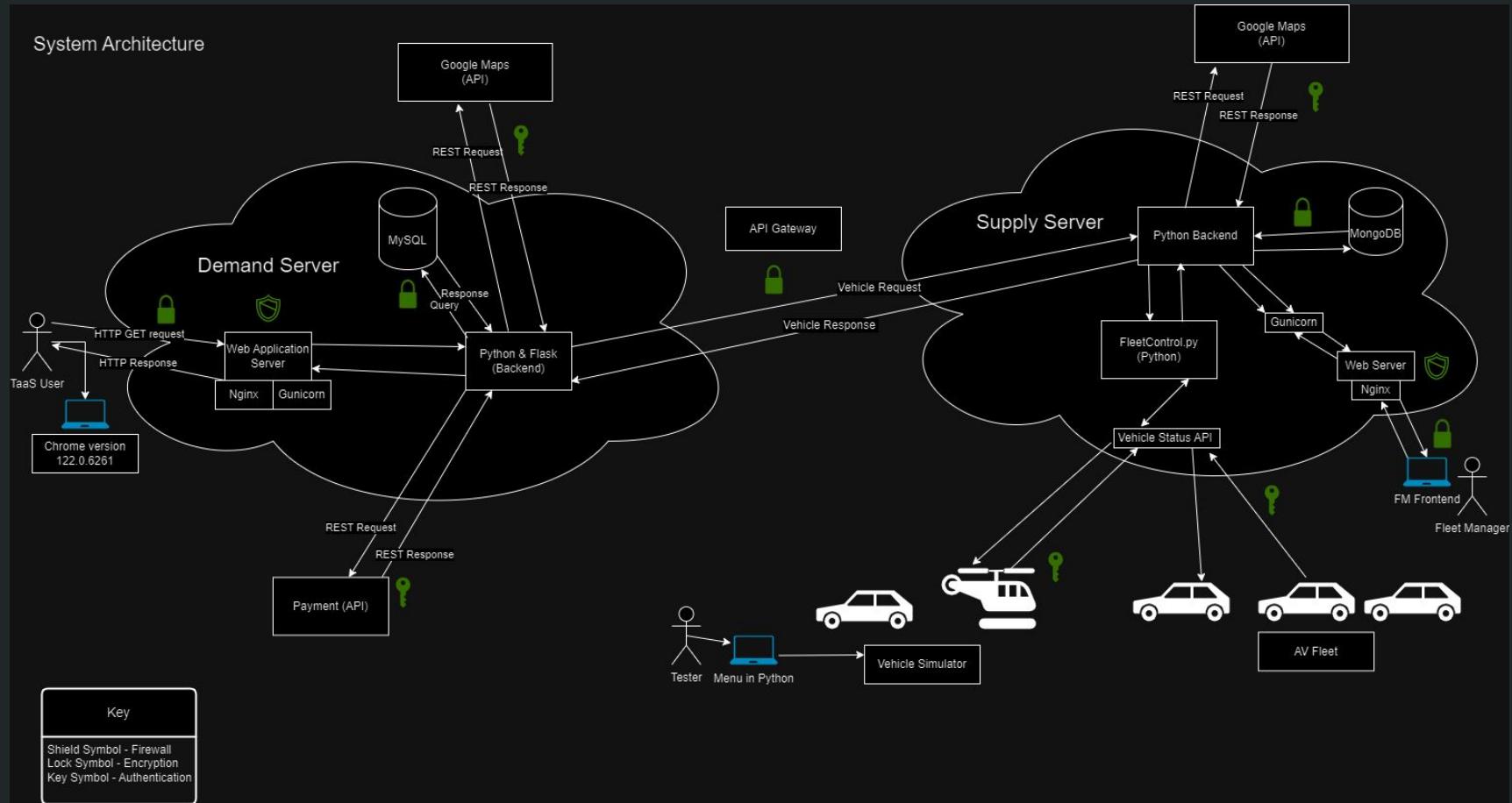
# Sprint Cards KPI

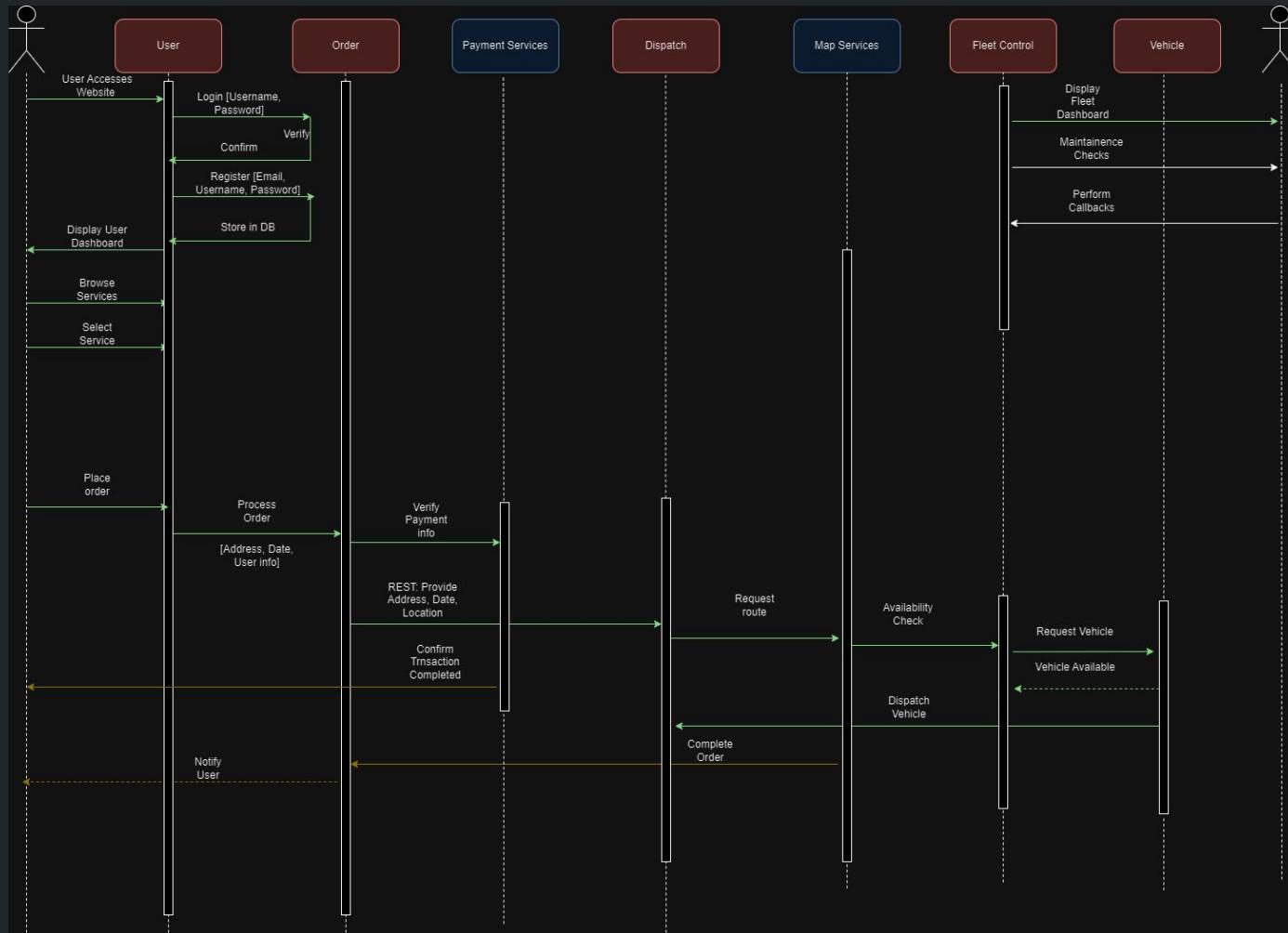


# TA Scorecard KPI



## System Architecture





## Try it Out!

Visit  
<https://team-22.seuswe.rocks>

Click the 'Sign Up' button and create an account

Click the 'Get Started' button to access the services

Fill out the Order requirements and click 'Place Order'

Click the 'Log In' button and sign in with your new credentials

Select your desired service and payload

Check back in with us for live confirmation from the Fleet Manager Dashboard

# Vehicle Simulator & More

---

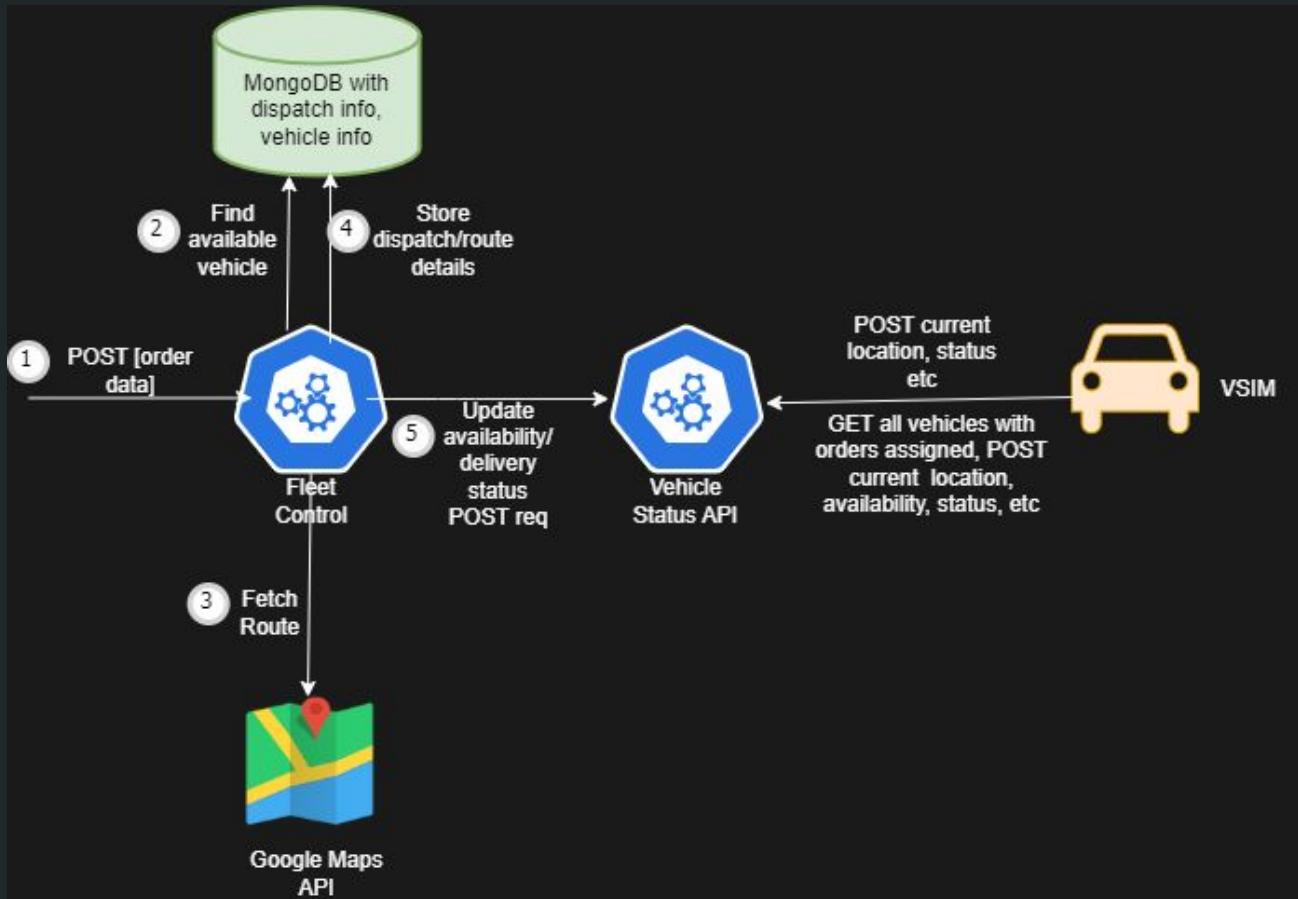
# Fleet Management System: Order to Dispatch

The fleet management system coordinates vehicle assignment, route planning, and real-time tracking to ensure efficient dispatch operations.

Utilizing:

- REST APIs for real-time vehicle management - Fleet Control and Vehicle Status
- Google Maps API - fetching and plotting routes
- Vehicle Simulator (VSIM) - simulates real-time vehicle movement

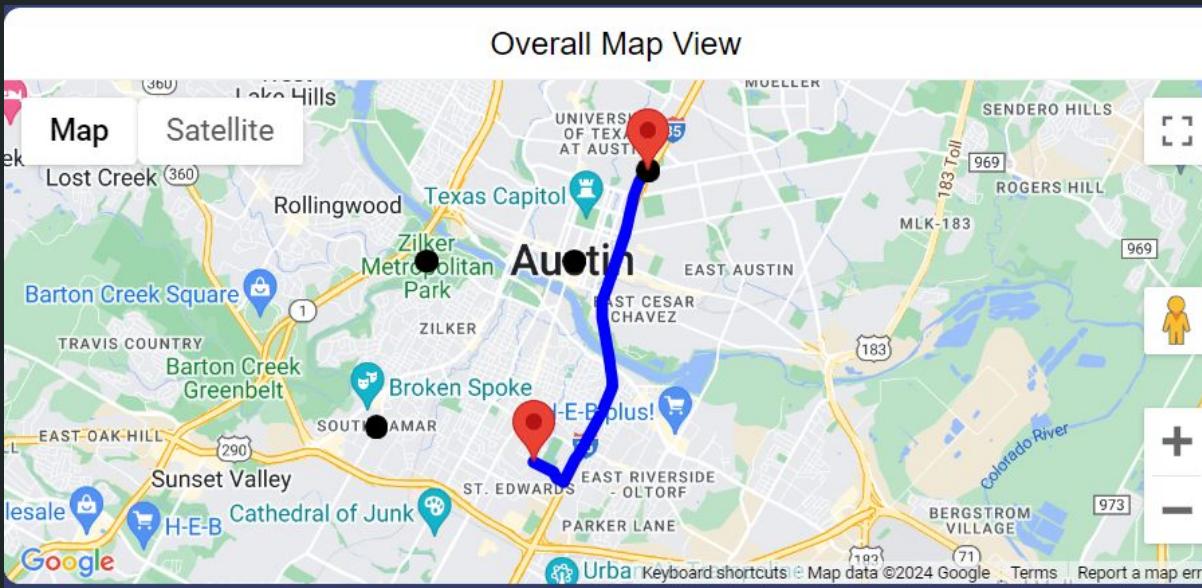
# High Level Overview



# Map Services

## Integration of Google Maps API

- Dynamic Mapping
- Fetch and Plot Routes
- Real-Time Visualization of Vehicle Location(s)



# Demand Side DevOps

---

# Making Sure Backend Is Stable

```
msala1203@ubuntu-s-1vcpu-2gb-sfo3-01:~$ sudo systemctl status nginx
● nginx.service - A high performance web server and a reverse proxy server
   Loaded: loaded (/lib/systemd/system/nginx.service; enabled; preset: enabled)
   Active: active (running) since Tue 2024-04-23 15:32:57 UTC; 2 weeks 0 days ago
     Docs: man:nginx(8)
 Main PID: 200865 (nginx)
    Tasks: 2 (limit: 2308)
   Memory: 5.1M
      CPU: 9.591s
     CGroup: /system.slice/nginx.service
             ├─200865 "nginx: master process /usr/sbin/nginx -g daemon on; master_process on;"
             └─200866 "nginx: worker process"
Apr 23 15:32:57 ubuntu-s-1vcpu-2gb-sfo3-01 systemd[1]: Starting nginx.service - A high performance web server and a reverse proxy server...
Apr 23 15:32:57 ubuntu-s-1vcpu-2gb-sfo3-01 systemd[1]: Started nginx.service - A high performance web server and a reverse proxy server.
msala1203@ubuntu-s-1vcpu-2gb-sfo3-01:~$ |
```

```
msala1203@ubuntu-s-1vcpu-2gb-sfo3-01:~$ sudo systemctl status demandcloud.service
● demandcloud.service - Gunicorn instance to serve demandcloud
   Loaded: loaded (/etc/systemd/system/demandcloud.service; enabled; preset: enabled)
   Active: active (running) since Thu 2024-04-25 00:28:52 UTC; 1 week 5 days ago
 Main PID: 222241 (gunicorn)
    Tasks: 4 (limit: 2308)
   Memory: 130.9M
      CPU: 4min 36.256s
     CGroup: /system.slice/demandcloud.service
             ├─222241 /home/team22/repos/demand_backend/env/bin/python3 /home/team22/repos/demand_backend/env/bin/gunic
             ├─222242 /home/team22/repos/demand_backend/env/bin/python3 /home/team22/repos/demand_backend/env/bin/gunic
             ├─222243 /home/team22/repos/demand_backend/env/bin/python3 /home/team22/repos/demand_backend/env/bin/gunic
             └─222244 /home/team22/repos/demand_backend/env/bin/python3 /home/team22/repos/demand_backend/env/bin/gunic
```

# Nginx (Host)

- Nginx is the person that hosts and lets people in.



# Flask (Chief)

- Flask is the person who cooks the food and takes requests



# Gunicorn (Waiter)

- Gunicorn is the middle man that makes sure the host and chief see eye to eye.

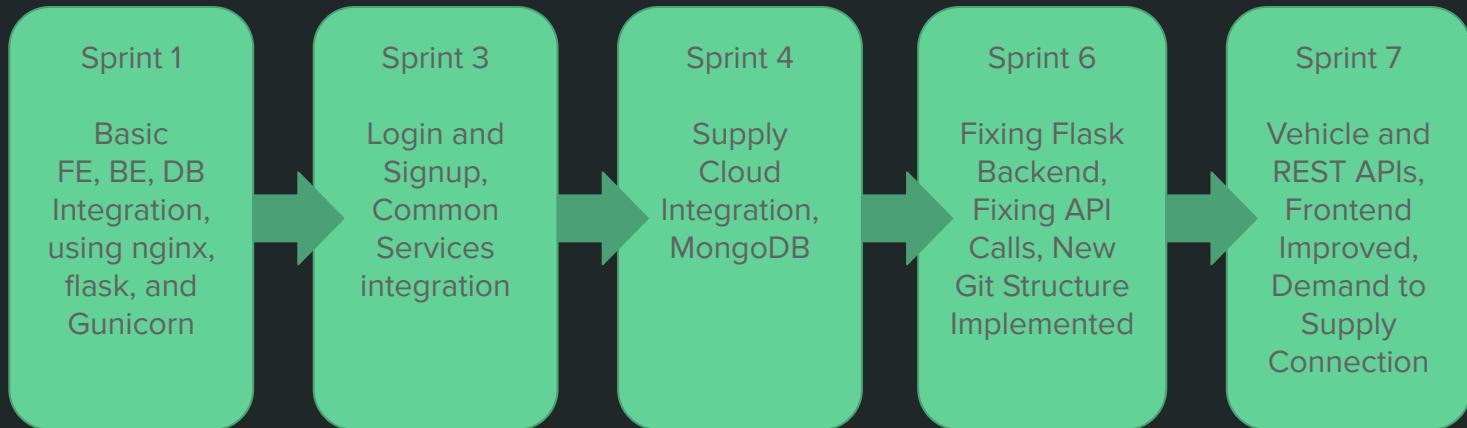


# Why does it bring value

- With all three working we are able to keep the guest happy and serve them swiftly and efficiently.



# DevOps Major Milestones Roadmap



- Spent all sprints fixing bugs
  - HTTP Errors etc.

# Integration, Fixing, & Refactoring

---

# Integration

- Integrated the frontend and backend on supply
- Made supply backend retrieve orders from MongoDB
- Made demand communicate with supply meaning...
  - This enables placement of orders on demand, which are then displayed on the FM Dashboard (in supply)
  - Additionally, when orders are placed, they go to a mySQL database and a mongoDB database collection



# Integration

- Integrated the frontend and backend on supply
- Made supply backend retrieve orders from MongoDB
- Made demand communicate with supply meaning...
  - This enables placement of orders on demand, which are then displayed on the FM Dashboard (in supply)
  - Additionally, when orders are placed, they go to a mySQL database and a mongoDB database collection

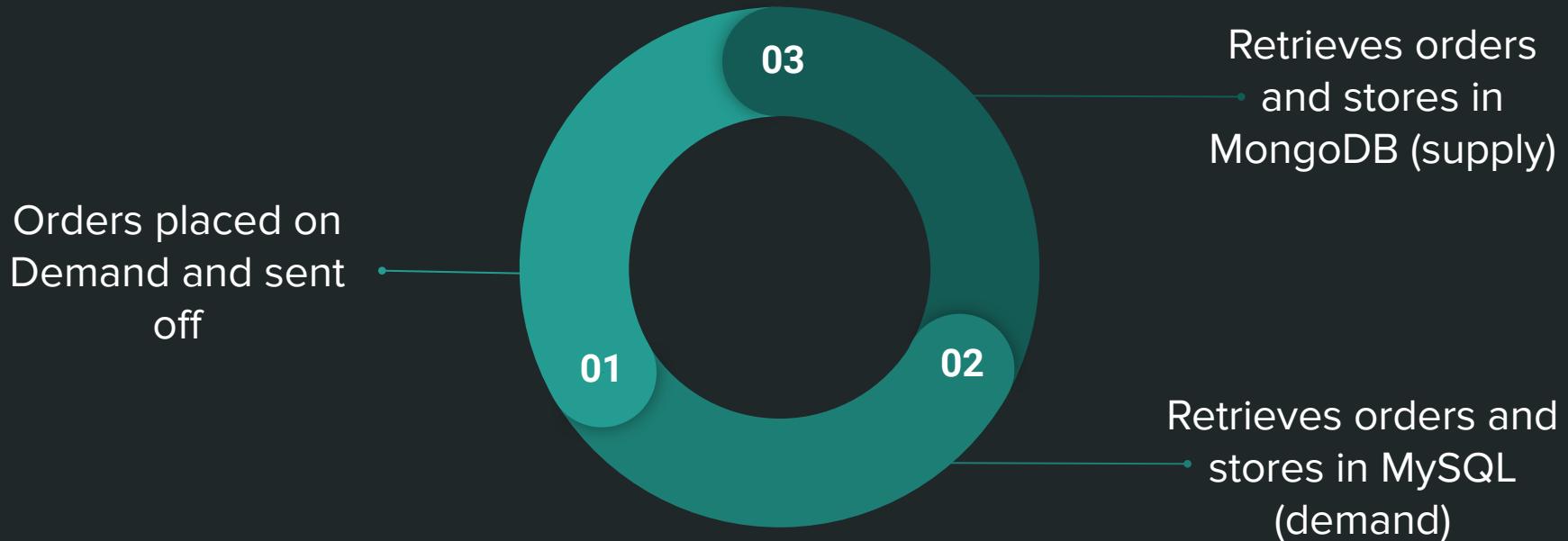


# Integration

- Integrated the frontend and backend on supply
- Made supply backend retrieve orders from MongoDB
- Made demand communicate with supply meaning...
  - This enables placement of orders on demand, which are then displayed on the FM Dashboard (in supply)
  - Additionally, when orders are placed, they go to a mySQL database and a mongoDB database collection

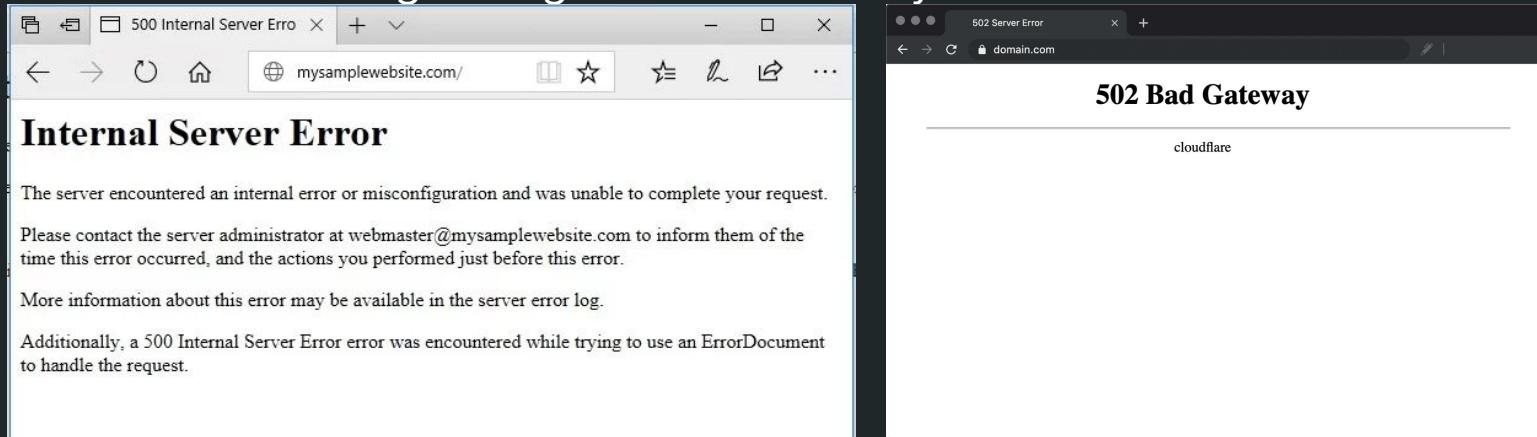


# Integration



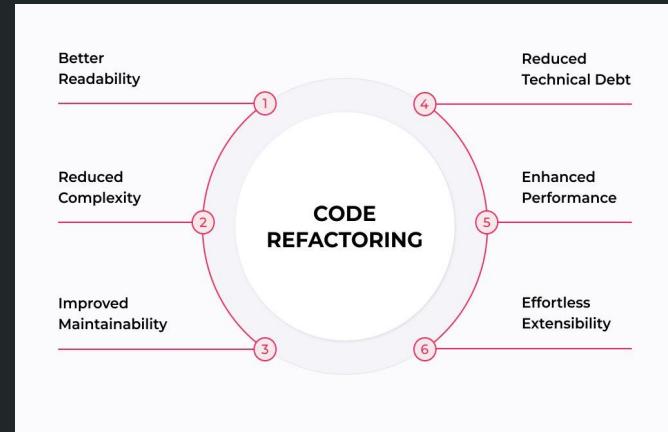
# Fixing

- During integration, several HTTP errors would pop up
  - HTTP 500, 502
  - HTTP 404, 405
- Mainly, the blueprints needed to be added to both sides (with routes)
  - Additionally, routes were inside classes on supply side
- And not calling the right variables in MySQL on demand side



# Refactoring

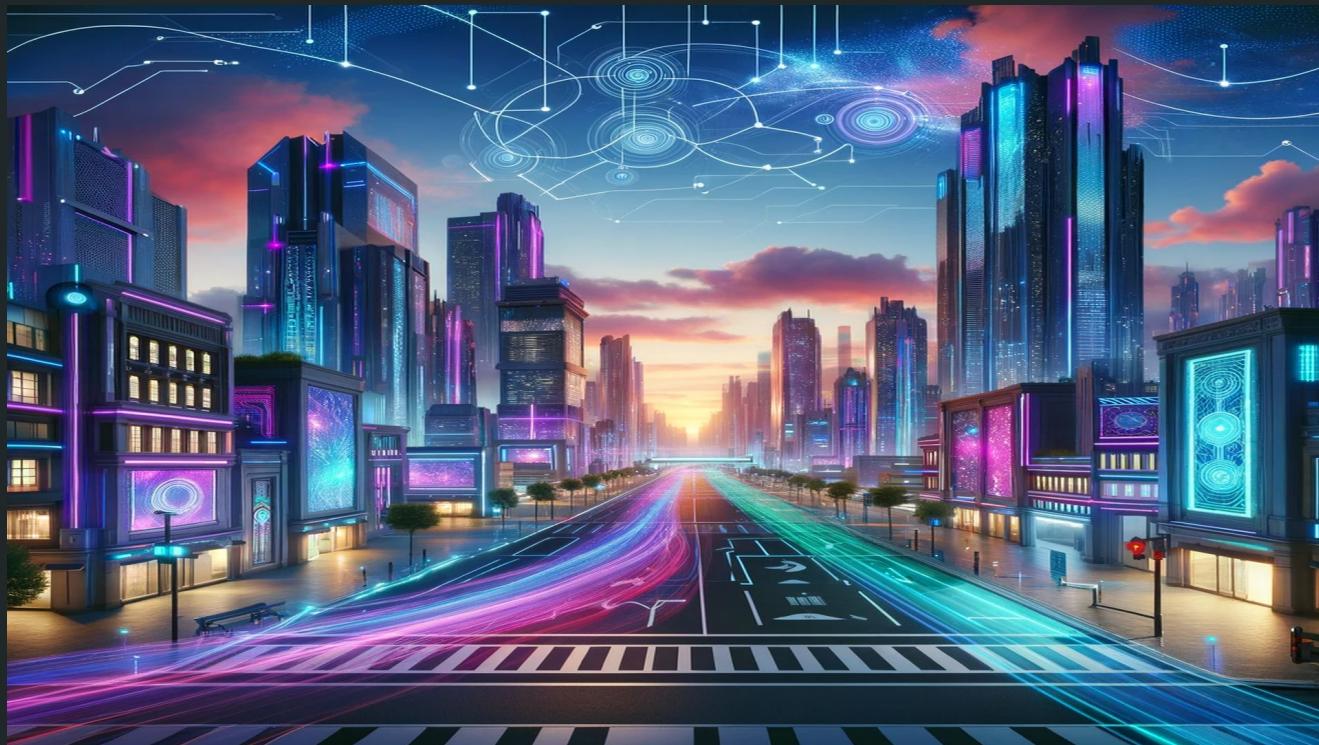
- Once we knew code worked, I started to refactor
  - To improve security,
  - To improve maintainability, and
  - Broke up supply backend into individual pieces (to match demand) w/ Jaret
- Now, our code is cleaner and easier to read
  - Code is using OO
  - Code was formatted and linted using VSCode extensions



# Additionally... How We Used AI

- For Debugging / Fixing
  - It helped us troubleshoot the various HTTP errors
  - Told us that we needed to register the blueprints
- For Making Unit Test Cases
  - It generated example Pytest (a Unit Test tool for Python) cases for us
- Helped with Code Refactoring
  - It gave us suggestions on how to make our code more maintainable and secure
- For Generating Images
  - It gave us an image of what our site's homepage background would've looked like

## Additionally... How We Used AI



Original Home Page Vision

# Frontend Development

---

# FrontEnd UI Design

## Color Palette

Primary Color: #3763f4  
Secondary Color: #dc3dcb  
Success Color: #19f5aa  
Info Color: #06a0b2  
Warning Color: #ffc107  
Danger Color: #dc3545  
Light Color: #f5f6f8  
Dark Color: #27262e  
Gray Palette:  
Gray: #4e5d78  
Gray Dark: #343a40  
Gray Light: #dee2e6  
Gray Extra Light: #f5f6f8

## Typography

Primary Font Family: Inter, Helvetica Neue, Helvetica, Arial, sans-serif  
Secondary Font Family: SFMono-Regular, Menlo, Monaco, Consolas, Liberation Mono, Courier New, monospace  
Font Sizes:  
Body: 1rem  
Heading: Varies  
Font Weight:  
Body: 400  
Heading: Varies

## Spacing

Padding:  
Input Elements: 0.6rem top and bottom, 1rem left and right  
Buttons: 0.6rem top and bottom, 2rem left and right  
Border Radius: 1rem (default), 2rem (for larger elements)  
Breakpoints  
Extra Small: 0px  
Small: 576px  
Medium: 768px  
Large: 992px  
Extra Large: 1200px  
Extra Extra Large: 1400px

## Other Guidelines

Box Shadow: Utilize box shadows for elevation effects.  
Form Validation Colors: Green for valid input, Red for invalid input.  
Link Decoration: Underlined links with hover effect.  
Selection: White text on primary color background with no text shadow.  
This guideline provides a basis for consistent styling throughout your UI components.  
Ensure to maintain visual consistency and usability across different screen sizes.

# FrontEnd Design

---

Before & After  
WireFrame

# Before

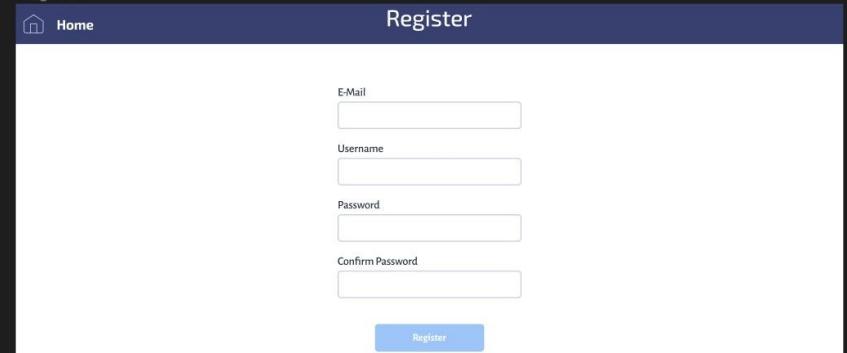
Homepage Layout



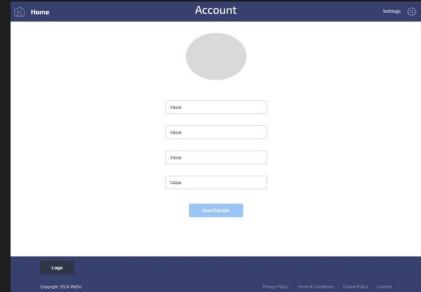
Services Button



Register



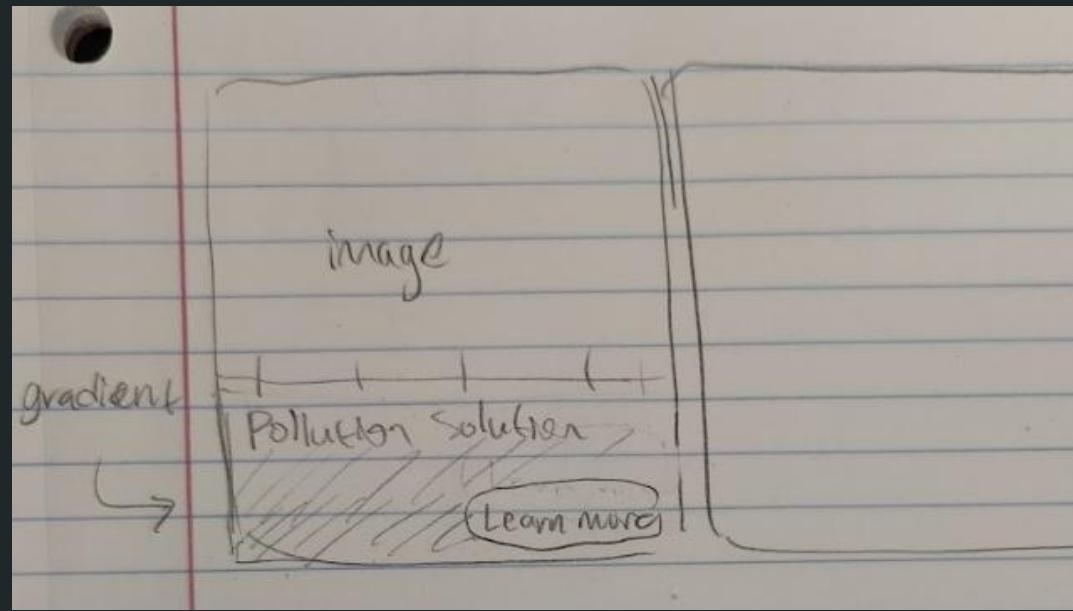
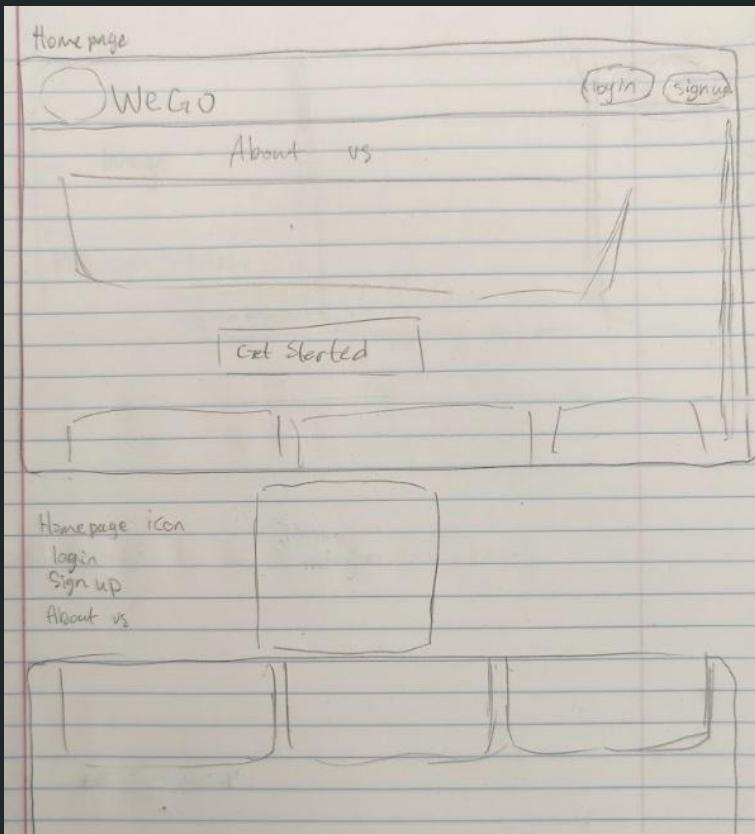
Account



Policies/Information



# The Idea



# Now



**Log In** **Sign up**

## About Us

**WEGO INC.**

**The best solution for you and your Delivery's**

Welcome to WeGo! Our mission: revolutionize online transportation with innovative solutions. We specialize in payload delivery services, focusing on non-human cargo markets. While our short-term focus is on autonomous vehicles for public roads, we're also exploring airborne drone services. Join us as we shape the future of



**Login**

**Welcome back**



**Email**

**Password**

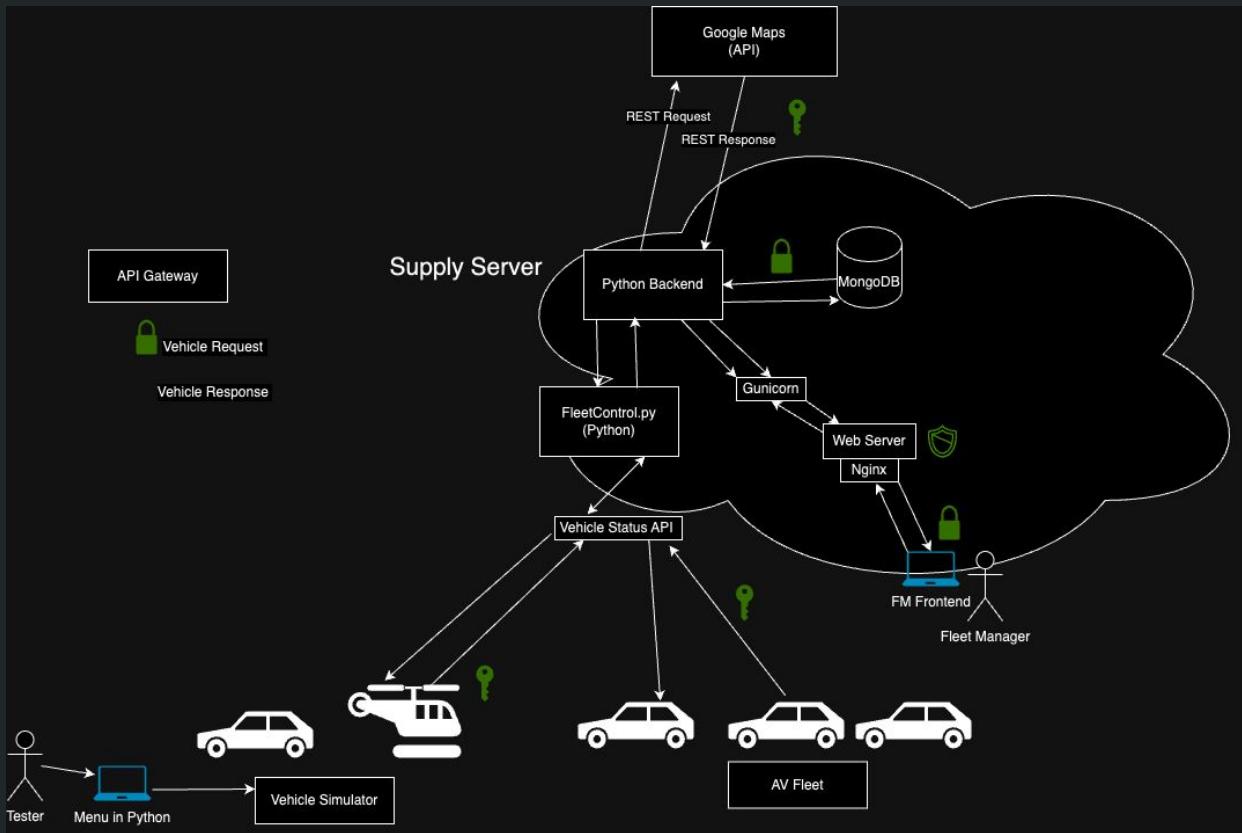
**Log in**

Forgot your password?

# Supply Side and Version Control System

---

# Supply Side Configuration



# Version Control System

## VCS Roadmap

### Sprint 6

- Bitbucket adoption
- CI/CD Pipeline
  - Demand
  - Supply
- Demand backend
  - Refactor procedural code to object oriented
- Web Stack configuration
- Unit Testing
- Knowledge share

### Sprint 7

- Modularity in backend
- Two additional Plugins
  - Farm Flight
  - Disaster Aid
- Front End Overhaul
- Cloud-to-cloud connection
- Vehicle Tracking
- VSIM Reiteration
- Solutions for 404, 405, 500, and 502 errors

### Future Plans

- Pull requests during deployment
  - Easier code reviews
- More unit testing
  - Include in pipeline deployment
- Design, Build and Test
- Web App evolution

# Team Evolution

Students pivot from traditional student learning-silo to discovery-based, small team learning



## *Team Evolution Lifecycle*

Forming	Storming	Norming	Performing
Team members still see themselves as individuals  Weekly expectations are clear	Members start seeing themselves as a team  Communication issues & internal conflicts occur	Team goals, working rules & plans drive collaboration and value  Members build relationships & knowledge share across the team.	The team is one agile, highly collaborative unit... demonstrating shared leadership & responsibilities  Highly systematic stakeholder value creation
Low levels of communication  Weekly expectations are unclear	Misunderstandings on responsibilities occur	Team members become dependent on each other  Collective work products are emerging	Continuous improvement mindset  Continuous learning mindset
Inclined to isolate in a silo	Struggling to apply critical thinking & systematic behavior	Evolving empathy & grasp on stakeholder interests  Evolving grasp on self-managed team, SWE Best Practices & ongoing performance improvement	Team members each demonstrate mastery of SWE Best Practices

# SWE BEST PRACTICES

Continuously Learning

Time Management

Flexibility

Communication

Commiting wisely

Empathy

Learning from mistakes



Organization

Planning Ahead

Positive Attitude



Teamwork



Maintaince

Documentation

Testing

# Fleet Manager Dashboard Roadmap

## Sprint 6

- Fleet Manager Dashboard design v3
- Initial Implementation of Frontend
- Deployed to cloud

## Sprint 7

- Map rendering
- Integration of Vehicle Status API Rendering
- Weather rendering SPIKE
- Weather rendering troubleshooting
- Vehicle Tracking Visualisation for Fleet Manager
- Modular CSS Implementation

## Future Plans

- Integration of Maintainability of Vehicles
- Call-back Functionality
- Live KPI Updates
- Vehicle Collision Alerts
- User Interface and Experience Enhancements
- Integration of AI for Decision Making

# FM Dashboard Vision

Hello Fleet Manager

Select period: from Jan 1, 2024 to Dec 31, 2024

Search... Filter

Total Profit \$250,000.00

Fuel Efficiency 84.2%

Average Delivery Time 00:00:24:37

Number of Completed Trips 29

Delivery Status

In Maintenance 30.3%

In Route 13.1%

Current Orders 28.6%

Troubled Vehicles 28%

Selected Vehicle

Overall Map View

Weather Updates

# VSIM Roadmap

## Sprint 1

- First iteration
- Menu function with start stop vehicle, navigate route functionality
- Multithreading
- Navigate predefined route
- Hard coded two vehicle objects
- Unit Testing

## Sprint 6-7

- Latest iteration
- Code overhaul
- Managing threads with locks
- More user-friendly interface
- Fetch real-time vehicle and route info via API calls
- Update status and location via API calls
- Error handling
- Updated test cases

## Future Plans

- Expansion of Sensor Integration
- Enhanced Testing and Validation Framework
- Dynamic Environment Interaction
- User Interface and Experience Enhancements
- Integration of AI for Decision Making

# Questions?

---

# Thank you!

---

Signing Off,  
Team 22