GridAI Final Presentation

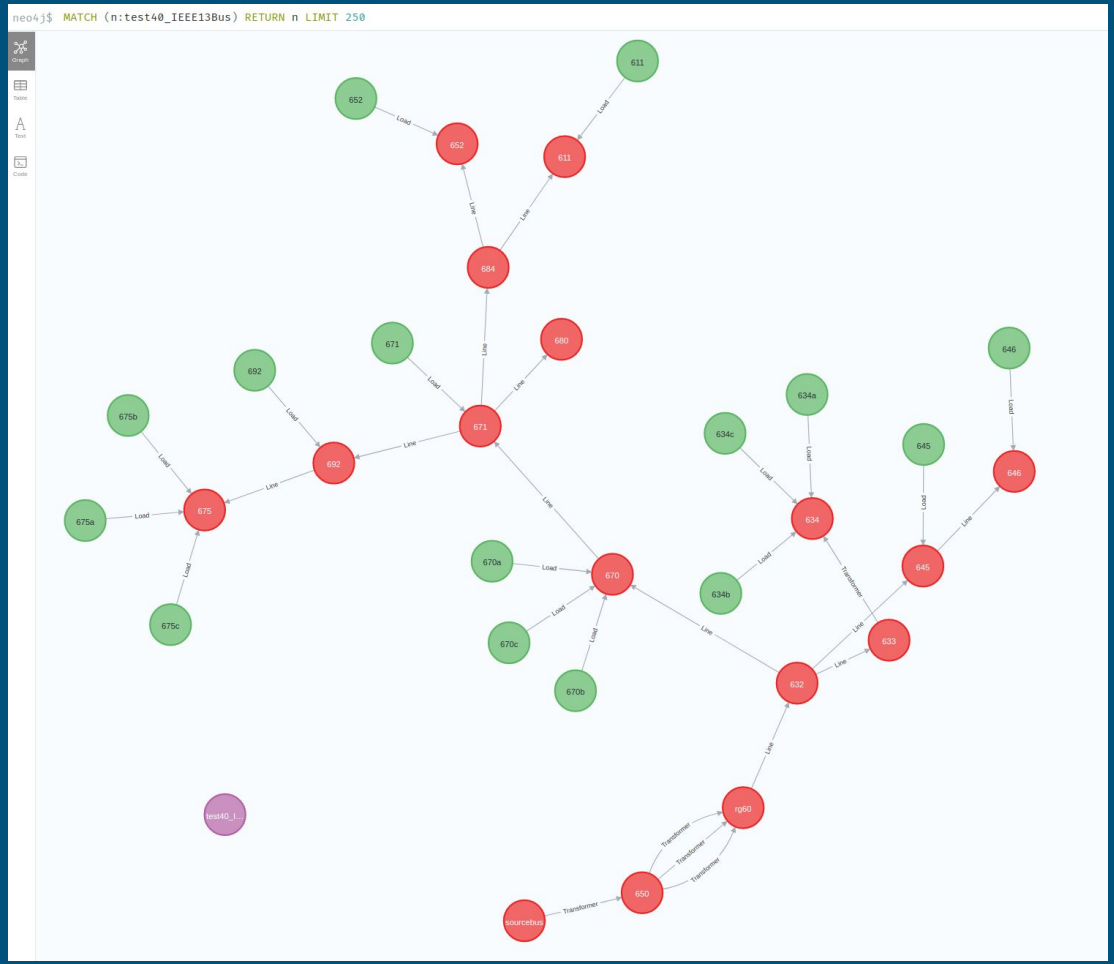
Team: Ryan Herren, Rolf Anderson, Tanay Parikh, Joshua Clinton, Elvis Kimara
Advisor: Ravikumar Gelli
Team: sdmay23-38

Introduction

- Problem Statement:
  - Power Grid is getting more complex by the day, and utility providers and grid operators lack solid visualization and analysis tools.
- Solution:
  - GridAI, a cloud-based system that provides power grid visualization and analysis, allowing for real-time updates.
- Intended Users:
  - Grid Operators
  - Utility companies
  - Organizations at any level of government

Grid Representation

- Static data with geographical information
  - Graph
  - Neo4j
- Time-series data
  - InfluxDB
- Components:
  - Bus
  - Load
  - Capacitor
  - Line

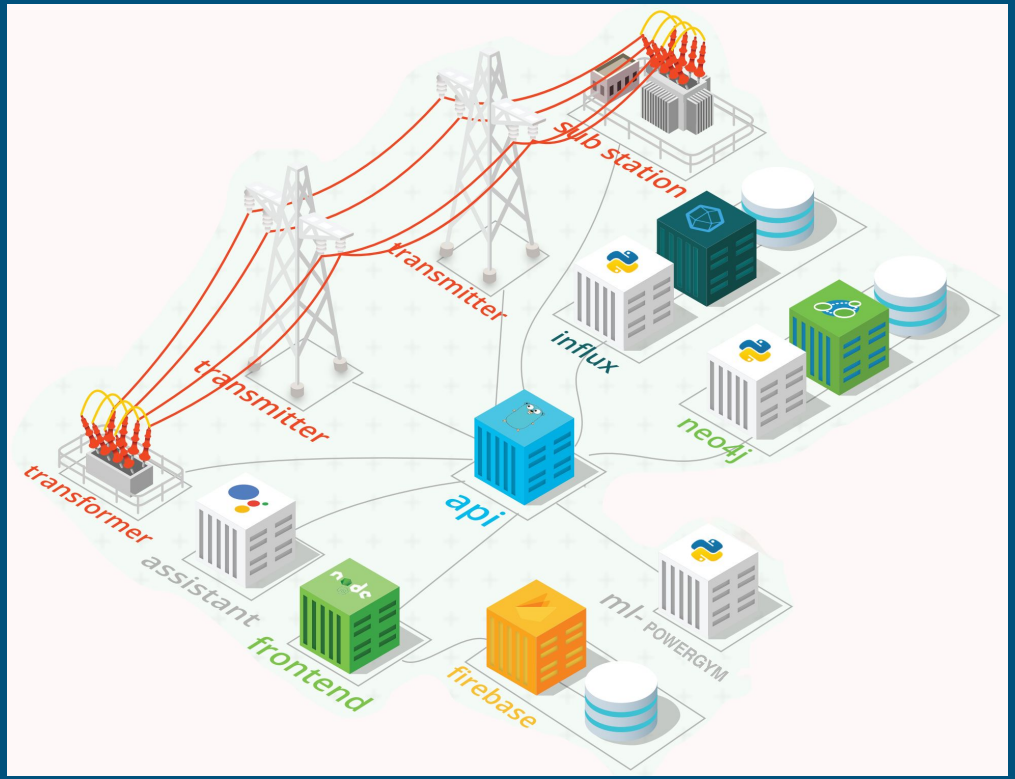


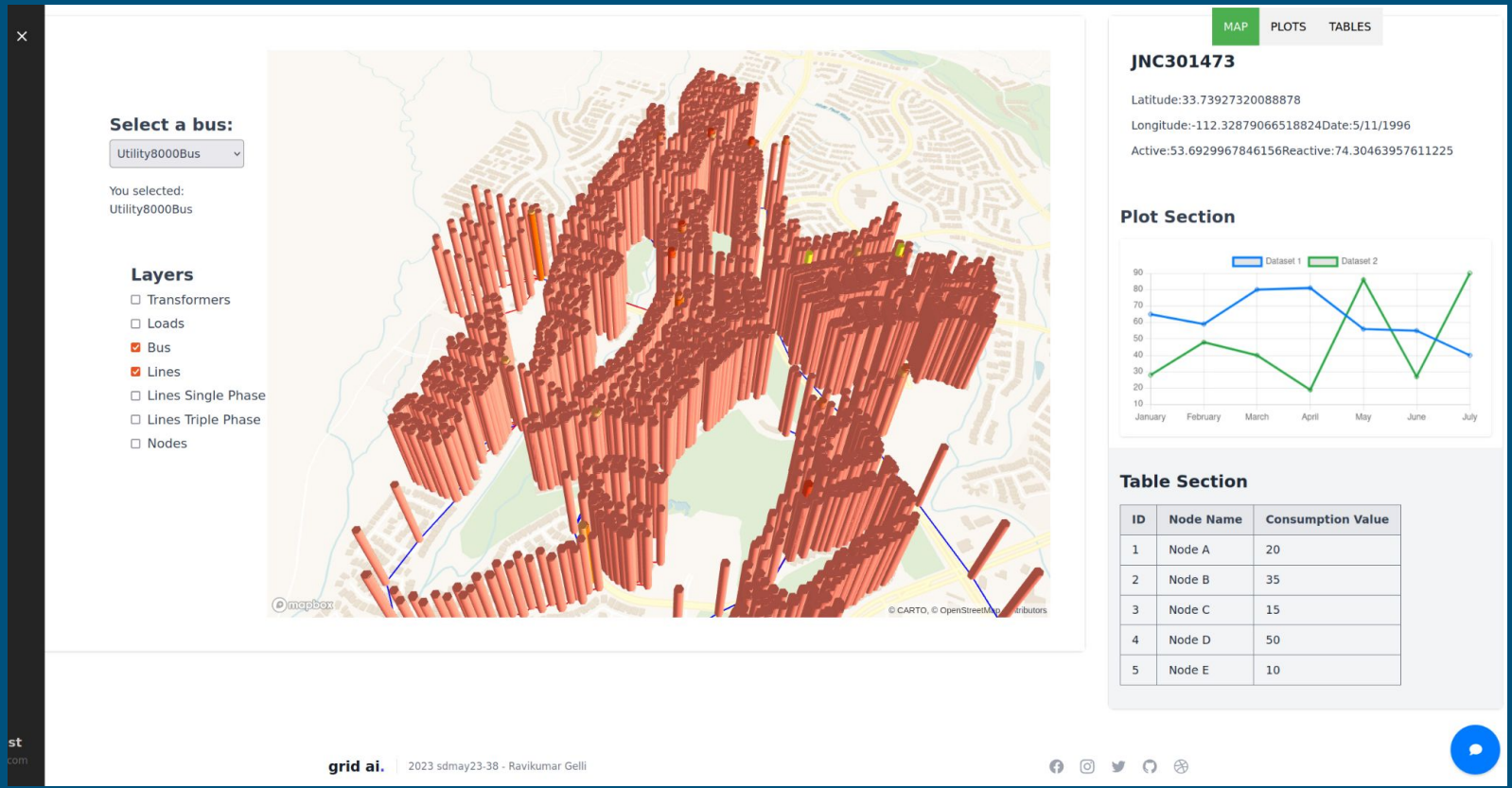
Key Contributions

- Rolf:
  - Filled out API routes for all service endpoints
  - Managed DevOps platform (CI/CD, GCP, Testing)
  - Assisted others
- Josh:
  - Designed and implemented Frontend
  - Created MapBox to display node data
  - Integrated ML Data to the Frontend
- Elvis:
  - Designed and implemented Assistant
  - Assisted with Frontend
- Tanay:
  - Designed and implemented Database
  - Uploaded simulated data to Influx
  - Building Neo4j Busses to component relationships
  - Neo4J/Influx queries to get specific data at busses or lines
- Ryan:
  - Designed and implemented RL system
  - PowerGym for RL environment
  - Stable-baselines3 for RL algorithm

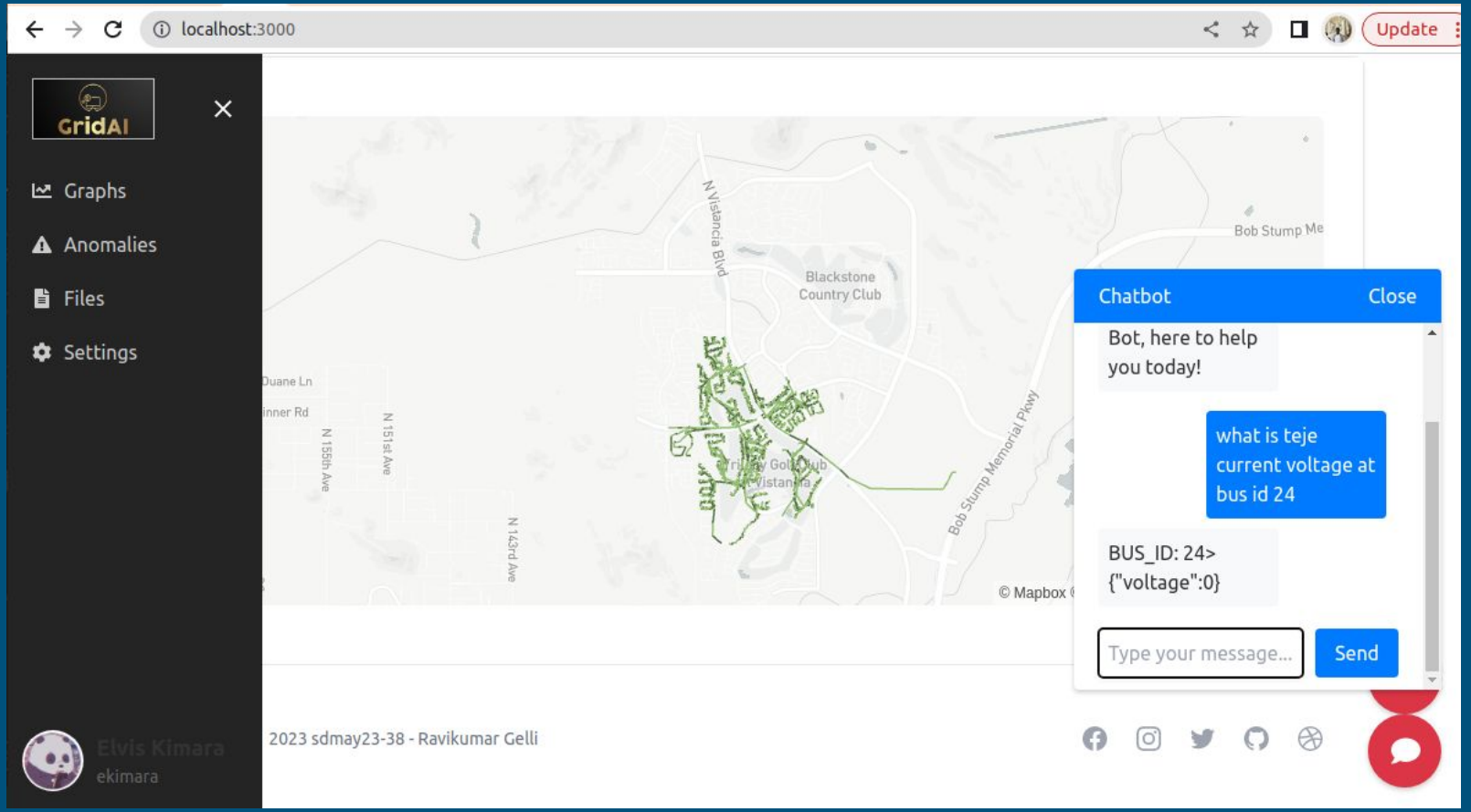
Implementation Architecture

1. Microservice Architecture
   a. Deployed in Docker containers to Cloud Run on GCP
2. Components:
   a. Frontend:
      i. Web app (Next.js/React)
      ii. Assistant (spaCy)
   b. Backend:
      i. API (Golang
      ii. Data Platform (Python)
      iii. ML (Python)

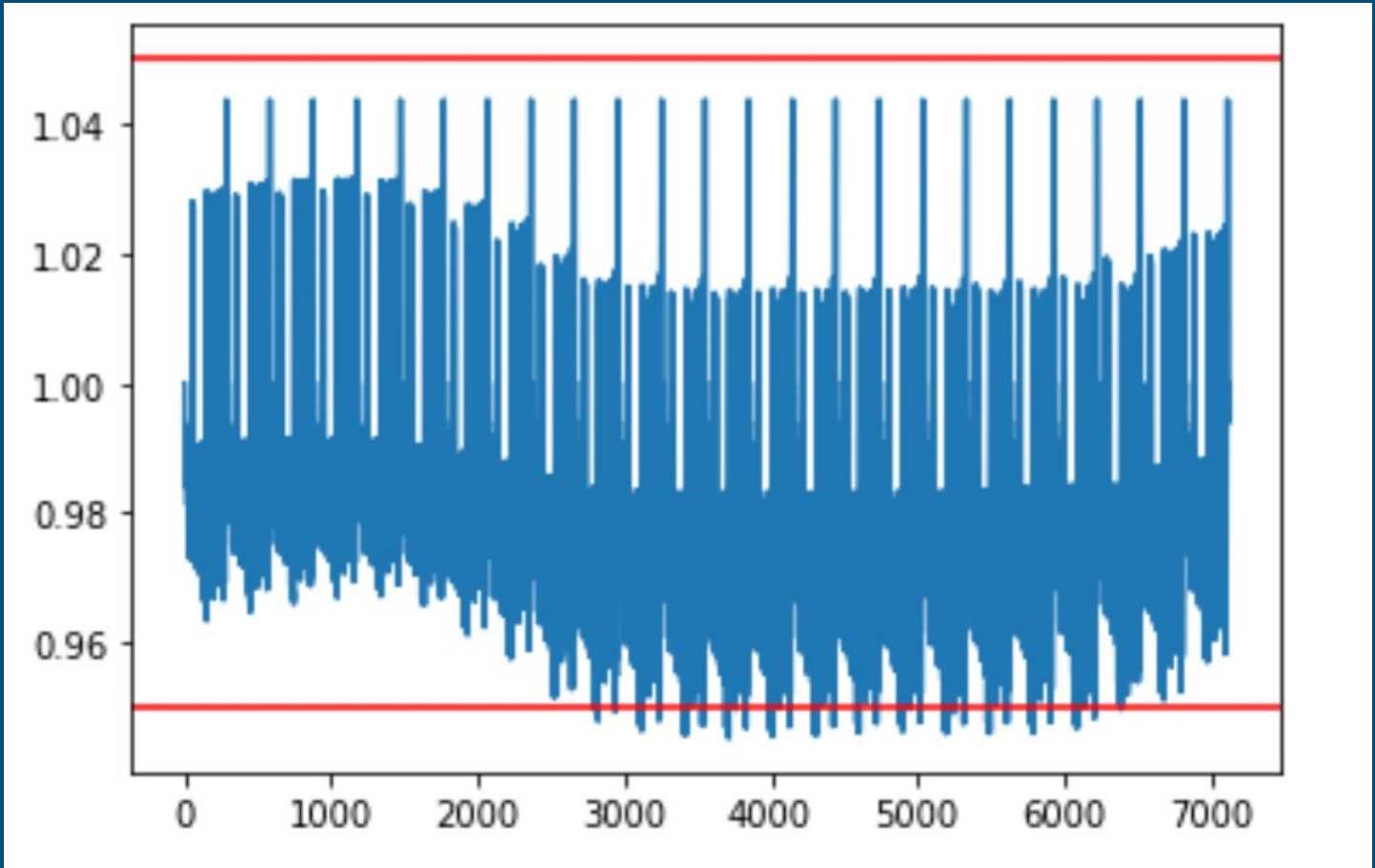



MAP
PLOTS
TABLES
Select a bus:
Utility8000Bus
You selected:
Utility8000Bus
Layers
Transformers
Loads
Bus
Lines
Lines Single Phase
Lines Triple Phase
Nodes
JNC301473
Latitude:33.73927320088878
Longitude:-112.32879066518824Date:5/11/1996
Active:53.6929967846156Reactive:74.30463957611225
Plot Section
Dataset 1
Dataset 2
90
80
70
60
50
40
30
20
10
January
February
March
April
May
June
July
Table Section
ID
Node Name
Consumption Value
1
Node A
20
2
Node B
35
3
Node C
15
4
Node D
50
5
Node E
10
grid ai.
2023 sdmay23-38 - Ravikumar Gelli

Frontend


localhost:3000
Update
GridAI
Graphs
Anomalies
Files
Settings
Elvis Kimara
ekimara
N Vistancia Blvd
Blackstone
Country Club
Bob Stump Me
Bob Stump Memorial Pkwy
N 151st Ave
N 155th Ave
N 143rd Ave
© Mapbox
Chatbot
Close
Bot, here to help you today!
what is teje current voltage at bus id 24
BUS_ID: 24> {"voltage":0}
Type your message...
Send
2023 sdmay23-38 - Ravikumar Gelli

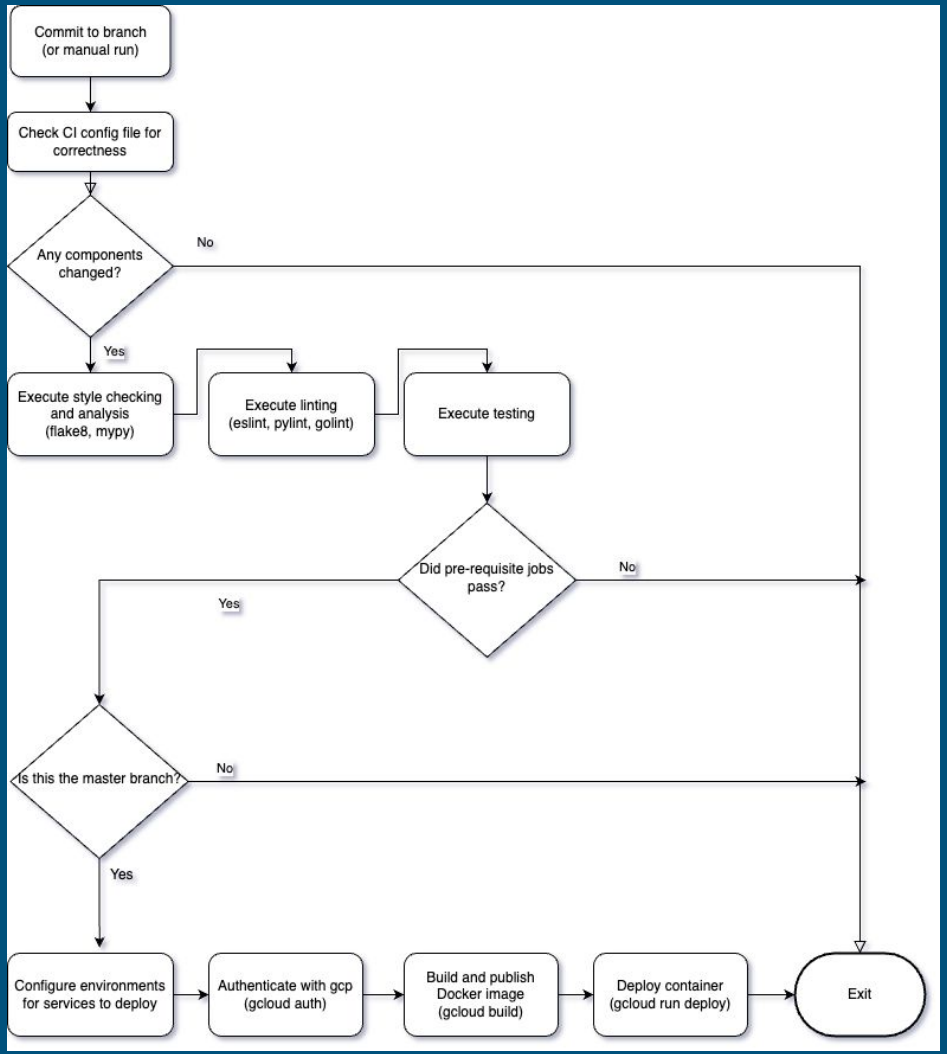
Assistant

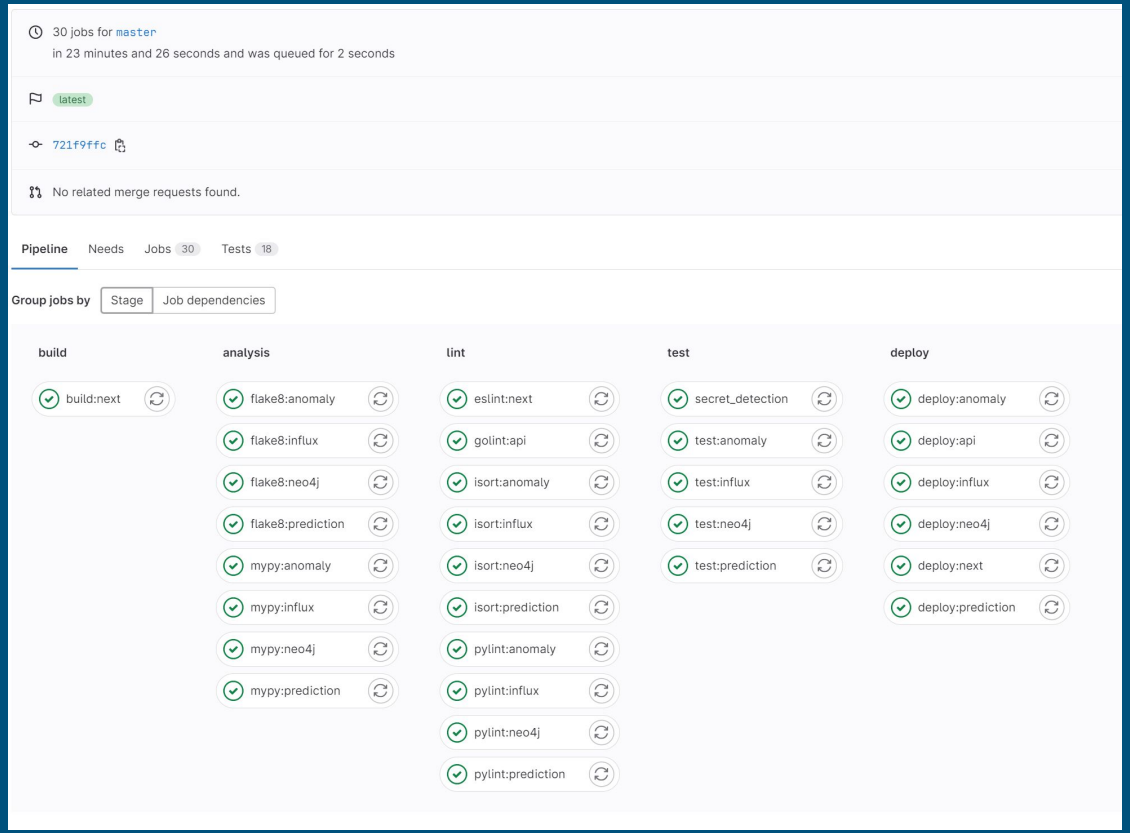


RL

CI/CD

- Gitlab CI/CD Pipelines
  - Only on changed components
  - Or when deploying
- CI Stages:
  - Build
  - Analysis (flake8, mypy)
  - Lint (eslint, pylint, golint)
  - Test (pytest)
  - Deploy
    - Publish and Deploy Docker containers
    - Gcloud interface
    - Prod only when merging into master
    - Deploy only if pre-requisite jobs succeed
    - Several dev deployments





CI/CD Pipeline Example

Work Accomplishments

- Fully redesigned Frontend to give UX similar to similar fully-featured platforms
  - Display map of grid, with components and edges
  - Ability to select and deselect data layers (e.g. to filter all transformers)
  - Authentication, user management with graph association
- Added a chatbot to give users specific data
- Switched ML system to use Reinforcement Learning (RL)
- Implemented cloud services, adding significant functionality
- Built a robust database system: simulated/Influx, firebase, graph/neo4j
  - Integrated Databases
  - Uploading Simulated data

Challenges and Solutions

- Power Grid, and ML concepts are complicated
  - Significant time devoted to understanding was required
- Last year's system included many inefficiencies
  - CI/CD was broken, many issues in components
- Google deprecated the Conversational Actions API
  - We were using this for our Assistant NLP
- GCP-wide issues with Cloud Run deployments in late April
  - Eventually this was resolved
- Following styling conventions could be difficult
  - Fortunately CI/CD caught them, forcing developers to fix it
- Upload functionality was all based upon reading web hosted .csv files
  - Redesign the architect of DB and integrating python dss
- We were an undersized group compared to most other senior design teams

Future Work

- Architecture and Platform Ops:
  - Move to using Kubernetes and GKE
    - Cloud Run doesn't give orchestration functionality
  - Manage Cloud resources with Terraform
- Continue to implement full RL system
  - Live data from API, more accurate models
- Implement functionality that would simulate time (Master dss file)
- Frontend
  - Improve Frontend UI design consistency
  - Add Frontend functionality as other services improve their functionality
- Complete Assistant system
  - Create a automated testing envt for future assistant deployments, analytics, reports
  - Expand assistant functionality, scalability, and training, and data collection
  - Adopt speech to text, more commands, and API calls
  - Do more requirements gathering and define a long term NLP goal, product, and technology

Conclusion

- This was, at times, a grueling project
  - Power System concepts are difficult to understand, even for Electrical Engineers
- While as a team we have made a lot of progress on the project and we are very close to completion, we learned a lot valuable lessons and skills that will help us our professional careers
- Build a foundation for the next team
- We wish we could’ve seen the project to completion
  - However, we are satisfied with the results