sdmay22-36
Artificial Intelligence Volt-Var Optimization(AI-VVO)

Jaden Alamsya
Demetrius Christou
Evan Dinnon
William Dulaney
Megan Phinney
Rachel Owens
Derrick Vang

Client/Advisor: Dr. Gelli Ravikumar

Project Vision

Goal:

- To design and develop a software tool including AI-based Volt-VAR optimization (VVO) for ensuring the voltage profiles on the feeders are within the prescribed threshold bands, particularly in the case of high penetration of Distributed Energy Resources (DERS) integrated into the distribution grid

Benefits:

- The AI algorithms and software tool will provide an interactive tool for utility operators to enhance the reliability and voltage stability for distribution grids, especially those with a large number of DERs.
- The AI-VVO tool monitors the real-time data from sensors of DERs and the distribution grid SCADA system, providing intelligent decisions such as setpoint controls to control the DERs and VAR devices.

History and Evolution of Project

sdmay21-24

- Machine Learning Algorithm: Q Learning with Pandas
- Text-based output
- No database backend

- No support for Distributed Energy Resources

sdmay22-36

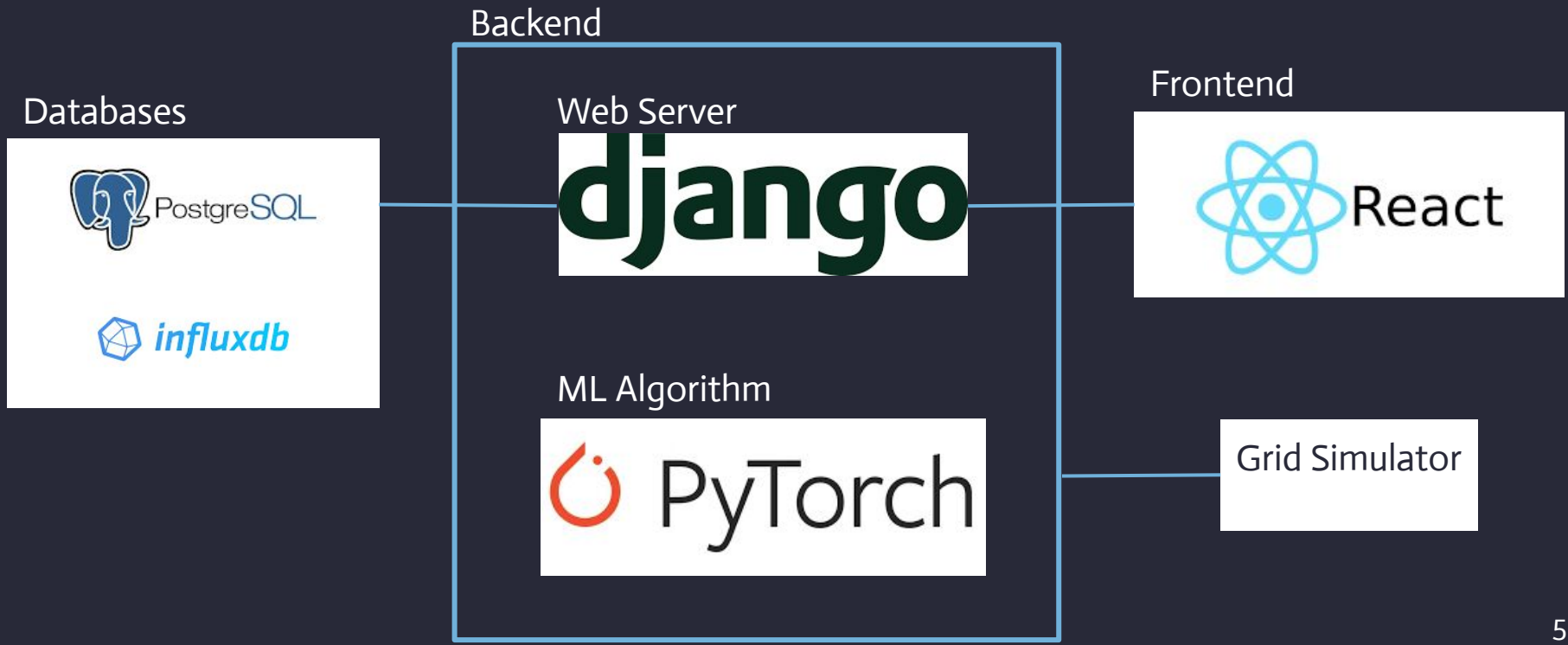
- Machine Learning Algorithm: Deep Q Learning with Pytorch
- Map-based grid display
- PostgreSQL and InfluxDB for backend
- Support for Distributed Energy Resources

Requirements

- Functional:
  - Map-based power grid display for DER-integrated distribution grid
  - Real-time AI-VVO tool for generating setpoint controls for DERs and Var devices in the grid
  - Interactive AI-VVO tool visualization by plots and tabular data
  - AI-VVO tool capable of archiving power grid data from the simulations

- Non-Functional:
  - Aesthetically pleasing UI
  - Ease of use for novice and experts

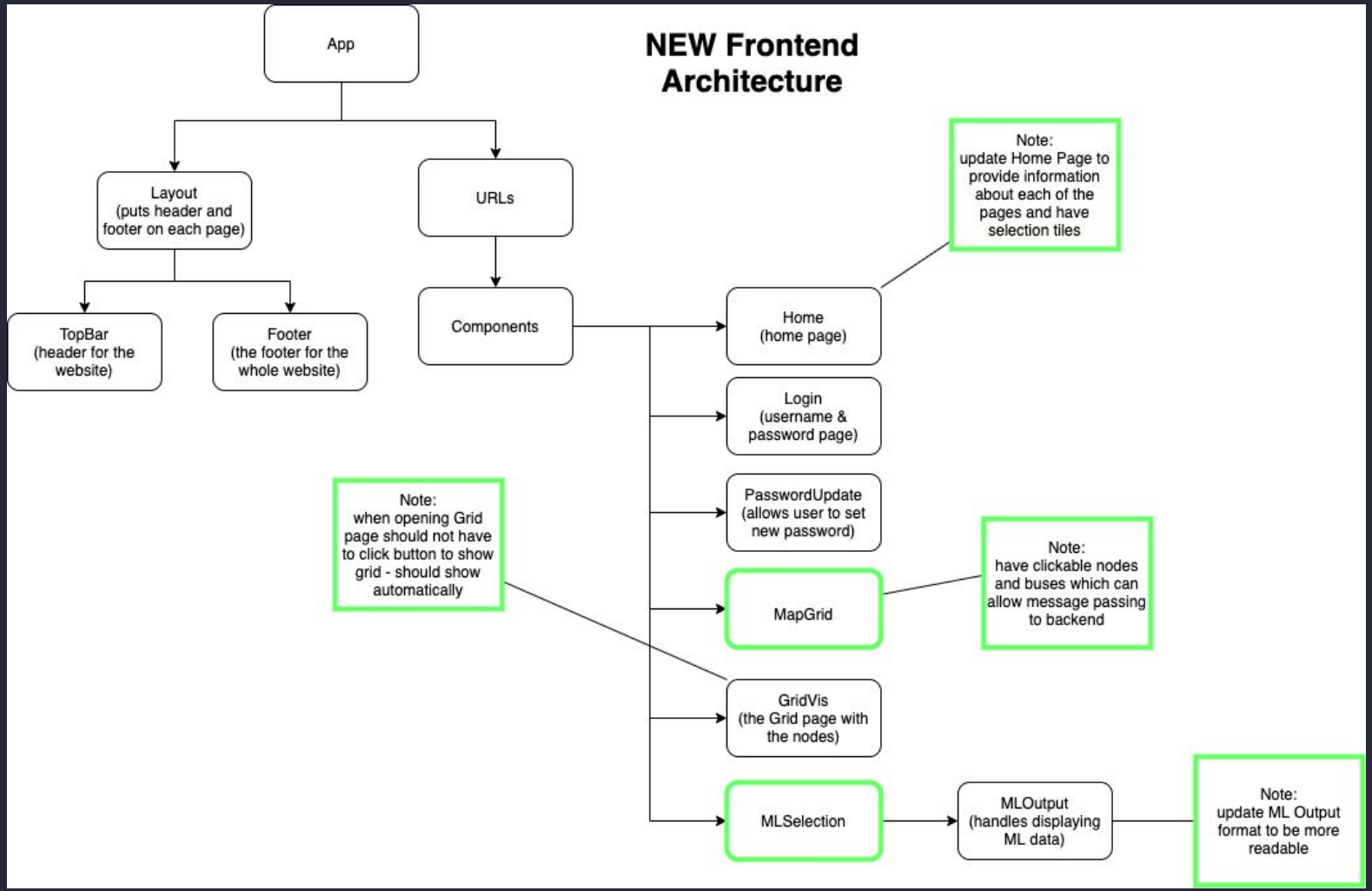
Conceptual Design Diagram



System Design: Frontend

- Tasks:
  - Add algorithm selection to main page
  - Add map-based power grid display
  - Implement data passing and configuration options through power grid nodes on map display
  - Add voltage information of nodes to grid display page
  - Increase readability for future teams

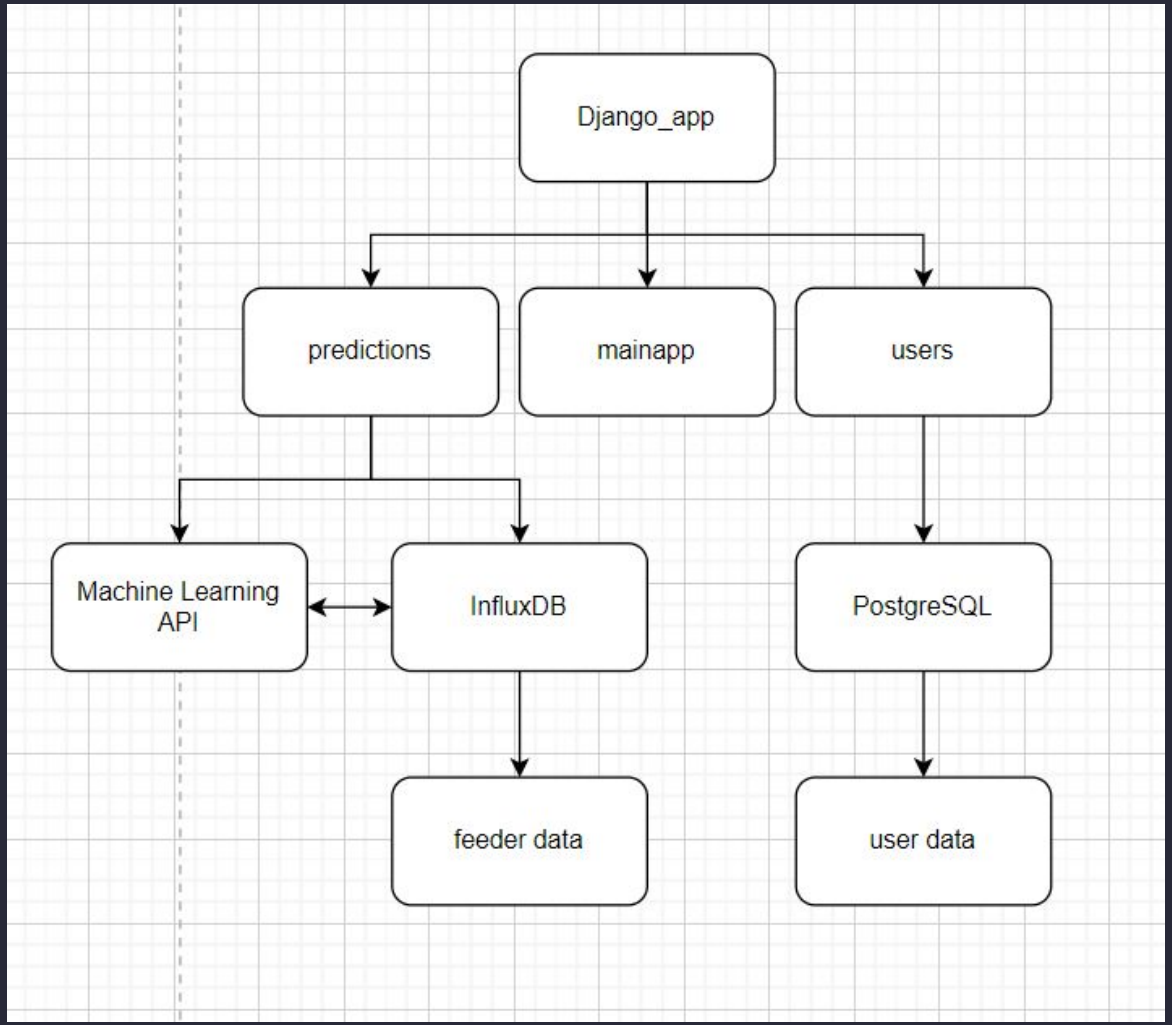
- Technology and Frameworks
  - ReactJS
  - React-Leaflet
  - Axios


NEW Frontend Architecture
App
Layout (puts header and footer on each page)
URLs
TopBar (header for the website)
Footer (the footer for the whole website)
Components
Home (home page)
Login (username & password page)
PasswordUpdate (allows user to set new password)
MapGrid
GridVis (the Grid page with the nodes)
MLSelection
MLOutput (handles displaying ML data)
Note: update Home Page to provide information about each of the pages and have selection tiles
Note: when opening Grid page should not have to click button to show grid - should show automatically
Note: have clickable nodes and buses which can allow message passing to backend
Note: update ML Output format to be more readable

System Design: Backend

- Tasks:
  - Fix PostgreSQL connections
  - Integrate InfluxDB
  - Increase readability for future teams

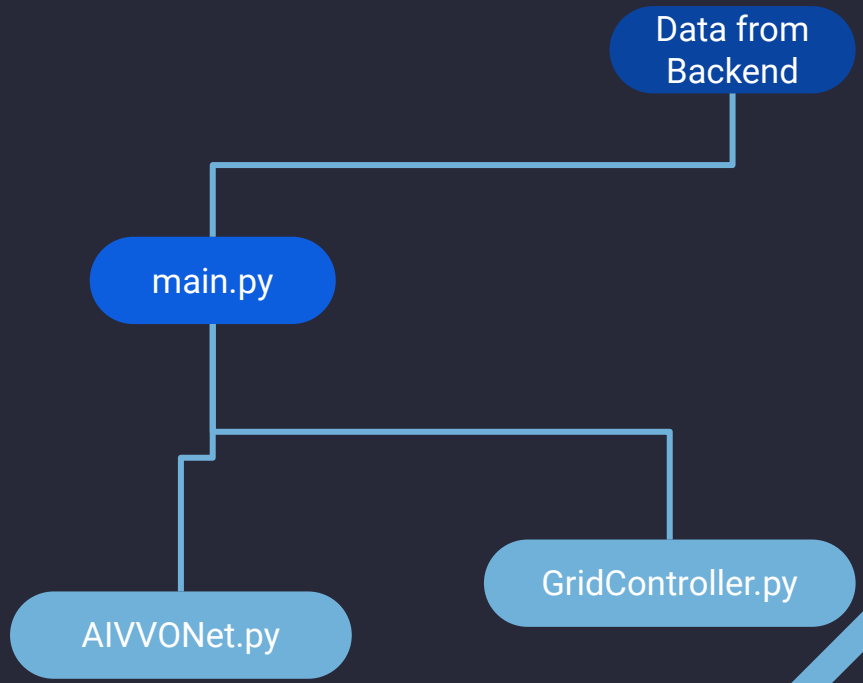
- Technology and Frameworks
  - Django
  - PostgreSQL
  - InfluxDB


Django_app
predictions
mainapp
users
Machine Learning API
InfluxDB
PostgreSQL
feeder data
user data

System Design: Machine Learning

- Tasks:
  - Functional decomposition, system architecture and software architecture
    - AIVVONet.py
    - GridController.py
    - main.py

- Technology and Frameworks
  - Pytorch
  - torch.nn.module



Design Complexity

- Continuation of last year's team
  - Considered trade-off of switching technologies

- Design Decisions
  - ReactJS for frontend
  - Map-based power grid display using react-leaflet libraries
  - PostgreSQL and InfluxDB for backend
  - Pytouch based Machine Learning algorithm.

Project Plan Tasks

- Frontend:
  - Algorithm selection
  - Map-based grid display
  - Display voltage information
  - User registration

- Backend:
  - Fix PostgreSQL connection
  - Integrate InfluxDB
  - Easy connection for stored data
  - Ability to store user info

Project Plan Tasks

- Machine Learning:
  - Higher test set accuracy with a new Machine Learning algorithm
  - Deep learning model that's not over/underfitting data
  - Segment data into:
    - Training set
    - Test set
    - Dev set

Schedule

	October	November	December	Spring 2022
Research				
Development				
Integration				
Testing				
Final Product Showcase				

- Each feature has specific requirements
- Development and Integrations include:
  - Frontend
  - Backend
  - Machine Learning

Risk and Mitigation Plan

- Risks
  - Learning curve with new technologies and frameworks
  - Virtual collaboration and development
  - Backend issues from previous team

- Mitigation Plan
  - Utilize Git
  - Agile methodology
  - Separate into different data into different databases
  - Regular meetings
  - Separate Discord channels for each team

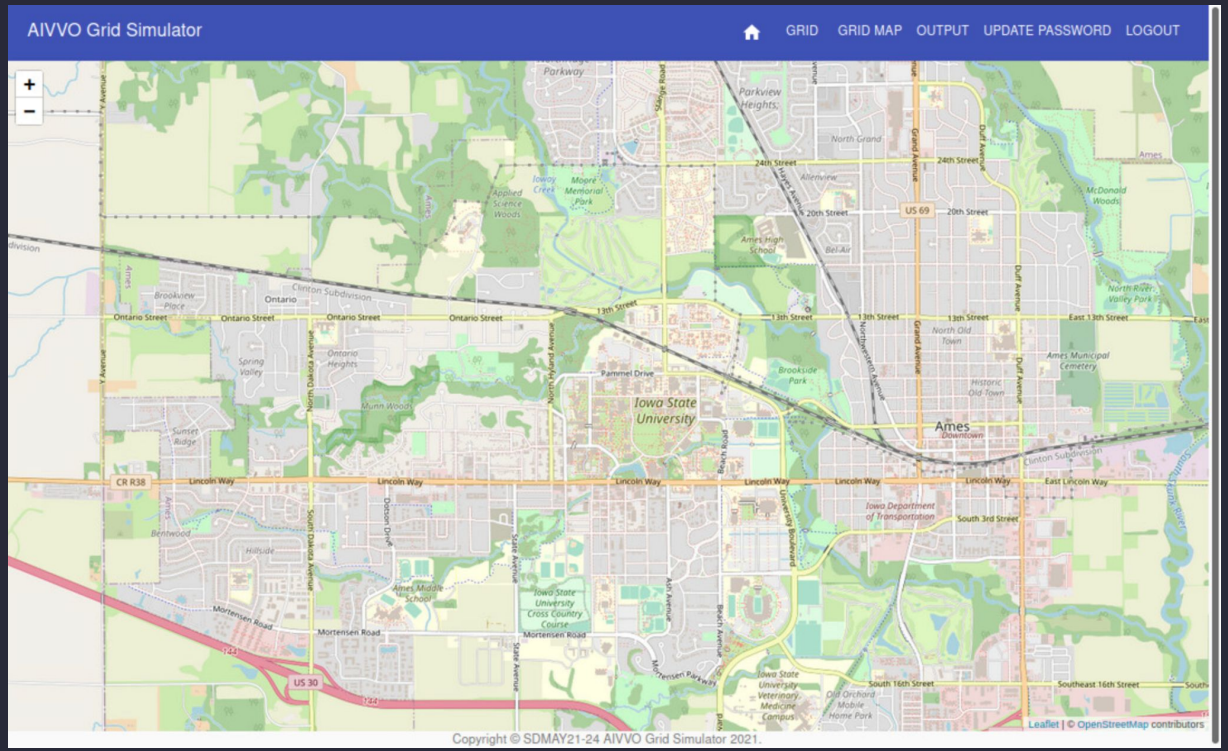
Test Plan

- Testable Units
  - Machine Learning API
  - Downloadable output
  - User registration
  - Machine Learning algorithms
- Tools
  - Frontend: Jest
  - Backend: Pytest
  - Machine Learning: Python asserts and Pytest
- Using Unit Tests
  - Looking and input/output
  - Run with incorrect input to test fail cases
  - Add new unit test for new features

Test Plan

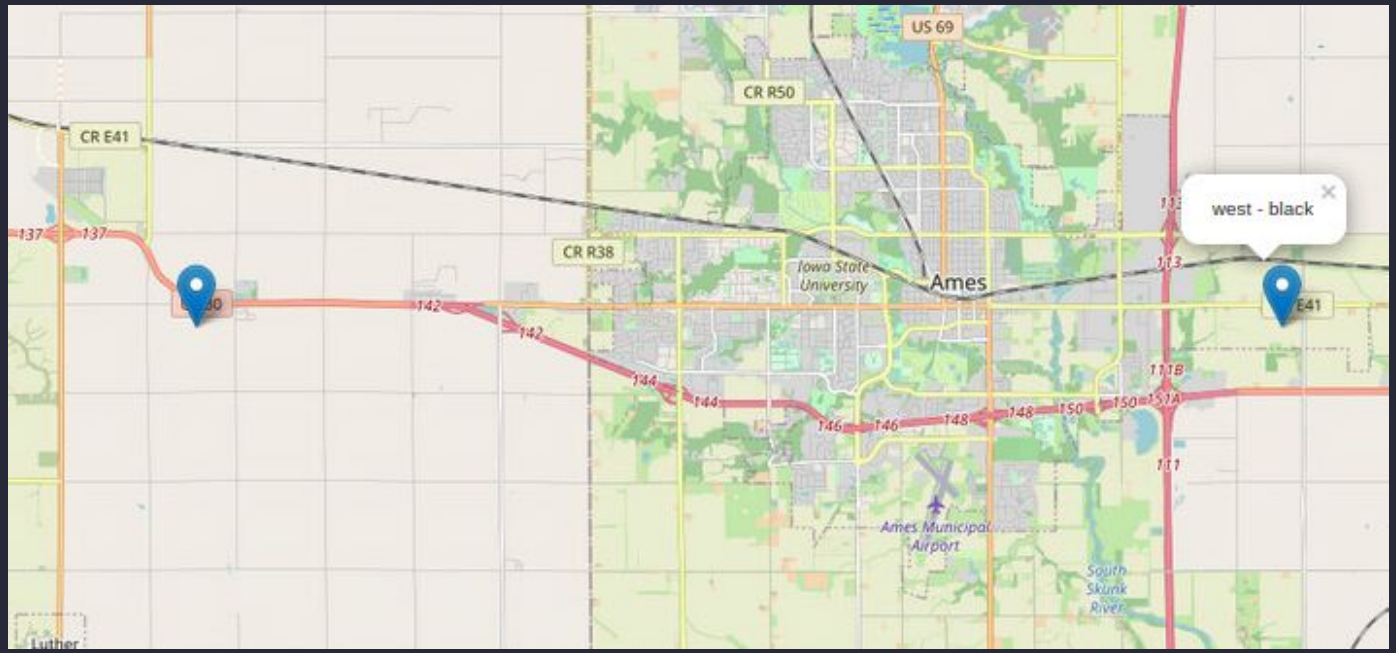
- Interface/Integration Testing
  - Refer to high level diagrams to indicate how components will interact
  - Incremental top-down approach
  - Test Docker communication
- System Level/Acceptance Testing
  - API communication
  - Test system response to test inputs
  - Communication tests
  - Combination of unit testing, prototypes and demos
  - Client is heavily involved in demos and give regular feedback

Prototype Implementation - Frontend



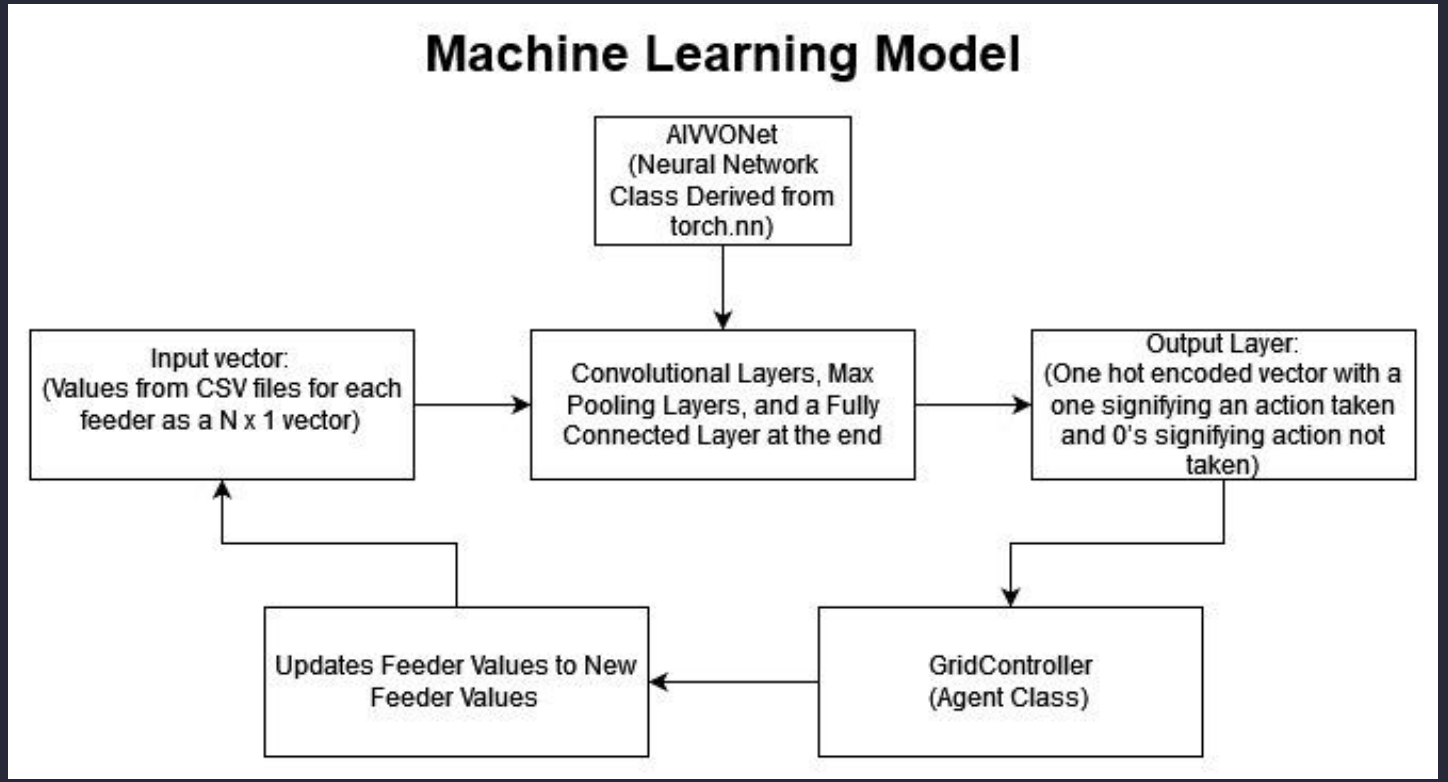
- Map-based grid display

Prototype Implementation - Frontend



- Custom markers and icons with pop ups

Prototype Implementation - ML



Prototype Implementation

- Video

Conclusion

- Started Implementation Stage
  - New Machine Learning algorithm with higher set accuracy
  - Upgrade database by utilizing PostgreSQL and InfluxDB
- Next Semester Plan
  - Finish implementation of requirements previously stated
- Individual Contribution
  - Jaden: route highlighting between icons
  - Demetrius: Implementation of InfluxDB and PostgreSQL
  - Evan: Communication between backend and ML loop
  - William: Writing class files from new ML algorithm
  - Megan: Created new Docker plan to connect all aspects
  - Rachel: Implement react-leaflet maps and pins
  - Derrick: Map display and custom icon

Acknowledgements

- This work is supported in part by the National Science Foundation under award number 2105269 (Physics-based Machine Learning Models and Algorithms for Attack-Resilient Smart Grid Infrastructure) and Iowa Energy Center Grant under award number 21-IEC-009 (Micro-DERMS: Distributed Energy Resource Management System (DERMS) for Real-time Monitoring & Control of Mobile Microgrids and DER Distribution Grid).

References

- H. Liu and W. Wu, "Two-Stage Deep Reinforcement Learning for Inverter-Based Volt-VAR Control in Active Distribution Networks," in IEEE Transactions on Smart Grid, vol. 12, no. 3, pp. 2037-2047, May 2021, doi: 10.1109/TSG.2020.3041620.
- N. Tomin, V. Kurbatsky, D. Panasetsky, D. Sidorov, A. Zhukov,Voltage/VAR Control and Optimization: AI approach**The work was carried out within the framework of the scientific projects of SB RAS III.17.4.2 "'Theory and methods of substantiating the development and control of intelligent electric power systems"' (No. AAAA-A17-117030310438-1) and III.17.3.1 "Theory and methods of contemporary mathematical programming in intelligent power systems" (No. AAAA-A17-117030310442-8),IFAC-PapersOnLine,Volume 51, Issue 28,2018,Pages 103-108,ISSN 2405-8963,https://doi.org/10.1016/j.ifacol.2018.11.685. (https://www.sciencedirect.com/science/article/pii/S2405896318334049).
- S. Wang, L. Du and Y. Li, "Decentralized Volt/Var Control of EV Charging Station Inverters for Voltage Regulation," 2020 IEEE Transportation Electrification Conference & Expo (ITEC), 2020, pp. 604-608, doi: 10.1109/ITEC48692.2020.9161713.
- Q. Li, Y. Zhang, T. Ji, X. Lin, and Z. Cai, "Volt/VAR control for power grids with connections of large-scale wind farms: A Review," IEEE Access, vol. 6, pp. 26675–26692, May 2018.
- W. Wang, N. Yu, J. Shi and Y. Gao, "Volt-VAR Control in Power Distribution Systems with Deep Reinforcement Learning," 2019 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm), 2019, pp. 1-7, doi: 10.1109/SmartGridComm.2019.8909741.
- Y. Zhang, X. Wang, J. Wang, Y. Zhang, "Deep Reinforcement Learning Based Volt-VAR Optimization in Smart Distribution Systems," IEEE Transactions on Smart Grid, pp. 1-11, 2020.

Thanks

Does anyone have any questions?

Jaden Alamsya, Demetrius Christou, Evan Dinnon, William Dulaney, Megan Phinney, Rachel Owens and Derrick Vang