

1 Project Charta

1.1 Problem Definition

New players coming to Pokemon Go are going to struggle with the highly complex combat system. There is much to consider such as type weaknesses, charged moves, fast moves, pokemon stats, and more.

Another problem that new players will have is identifying the many different pokemon and learning their names. Given that there are over 800 pokemon in pokemon go, it'll be no easy task.

1.2 Situation Assessment

We are developing a **Pokémon battle assistant** that recommends the best Pokemon choices for players during battles. Additionally, we are creating an **image recognition model** that identifies Pokémon based on images and outputs their names.

We will use Python libraries such as **pandas**, **selenium**, and **csv** for data processing and automation. Our data source and battle simulations will rely entirely on pvpoke.com. ChatGPT will assist us with coding and other support tasks. As for models we will use the following models **Linear Regression**, **Random Forest**, **Gradient Boosing**, and **Categorical Boost (CatBoost)**.

1.2.1 Constraints

- The project must be completed within **4 weeks**, requiring careful planning and time management.
- The team size is limited to four members
- Complete dependency on pvpoke.com for battle data and simulations; potential risk if the site changes or becomes unavailable.
- The image recognition model's accuracy depends on the quality and quantity of training data.

1.2.2 Risks

- Potential changes in pvpoke.com's site structure or access policies could disrupt data gathering and simulations.
- Insufficient training data or poor labeling could reduce image recognition model performance.
- Integration challenges between the battle assistant and the image recognition model or difficulties with Python libraries.
- The tight 4-week timeline may limit thorough testing, optimization, and documentation.
- Risk of scope creep if project goals are not clearly defined and managed.

1.3 Project Goals and Success Criteria

For our battle assistant to be successful it needs to boost the winrate for new players overall while utilizing multiple aspects for pokemon fights.

For our image recognition to be successful it should be able to recognize a pokemon and output the correct name with a high accuracy rate.

1.4 Data Mining Goals

1.4.1 Battle Outcome Prediction (Classification)

- Input Data: Battle simulation dataset (`data_acquisition/vectorized_data/battle_data_numeric.csv`)
- Task: Binary classification to predict which Pokémon wins (left vs. right)
- Target Variable: **winner** (1 = left wins, 0 = right wins)
- Metrics and Targets:
 - Accuracy 0.79
 - F1-score 0.78
- Goal: Provide reliable win-probabilities to inform user decisions in real-time.

1.4.2 Image Recognition (Classification)

- Input Data: Pokémon image dataset (`data_acquisition/image_dataset/final_pokemon_dataset`)
- Task: Multi-class classification to identify Pokémon species (727 classes)
- Target Variable: **pokemon_name**
- Metrics and Targets:
 - Top-1 Accuracy 0.90

– Precision & Recall 0.88

- Goal: Enable seamless user input via image upload with high confidence.

1.5 Project Plan

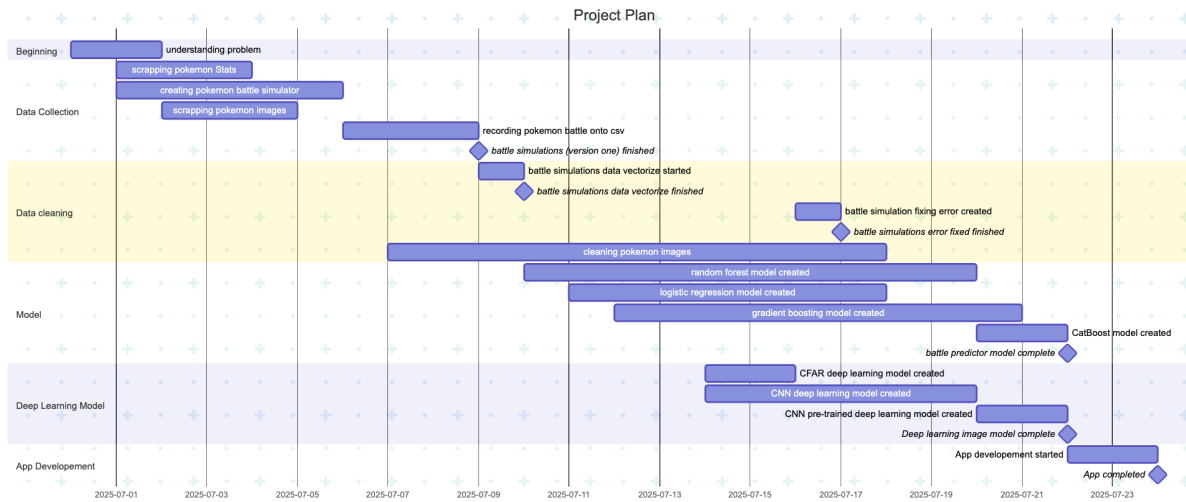


Figure 1.1: Project Plan

1.6 Roles and Contact Details

1.6.1 Caleb Boerma

Roles:

- Battle Simulator Architect
- Gradient Boosting & CatBoost Model Developer
- Web App Contributor

Tasks:

- Designed and implemented the 1v1 battle simulation engine
- Developed and tuned Gradient Boosting and CatBoost models for battle prediction

- Assisted with building and integrating the Streamlit web interface

Contact Info:

- Email: Boermaca@mail.gvsu.edu
- Github: <https://github.com/Calebb2202>

1.6.2 Diego De Jong

Roles:

- Battle Simulator Contributor
- Linear Regression Model Developer
- Web App Engineer

Tasks:

- Assisted in creating the battle simulation framework
- Implemented the Linear Regression baseline model for battle outcome prediction
- Built and deployed the final Streamlit web app application

Contact Info:

- Email: dejondie@mail.gvsu.edu
- Github: <https://github.com/Dasnapplel>

1.6.3 Dylan Baumann

Roles:

- Image Pipeline Engineer
- Deep Learning Model Developer

Tasks:

- Scraped and cleaned image data for the deep learning pipeline
- Developed and trained CFAR, CNN, and transfer-learning models for Pokémon classification

Contact Info:

- Email: baumadyl@students.zhaw.ch
- Github: <https://github.com/baumadyl01>

1.6.4 Max Vangsted**Roles:**

- Data Acquisition Specialist
- Random Forest Model Developer
- Deep Learning Model Collaborator

Tasks:

- Scraped initial Pokémon stats, moves, and types; performed vectorization
- Implemented the Random Forest classifier for battle prediction
- Contributed to the deep learning image classification project

Contact Info:

- Email: vangsmax@students.zhaw.ch
- Github: <https://github.com/vangmax>