CS 492- Senior Project 2

CleaverWall

Ali Emre Aydoğmuş - 21901358
Arda Barış Örtlek - 21903472
Onur Korkmaz - 21802925
Selahattin Cem Öztürk - 21802856
Yekta Seçkin Satır - 21903227

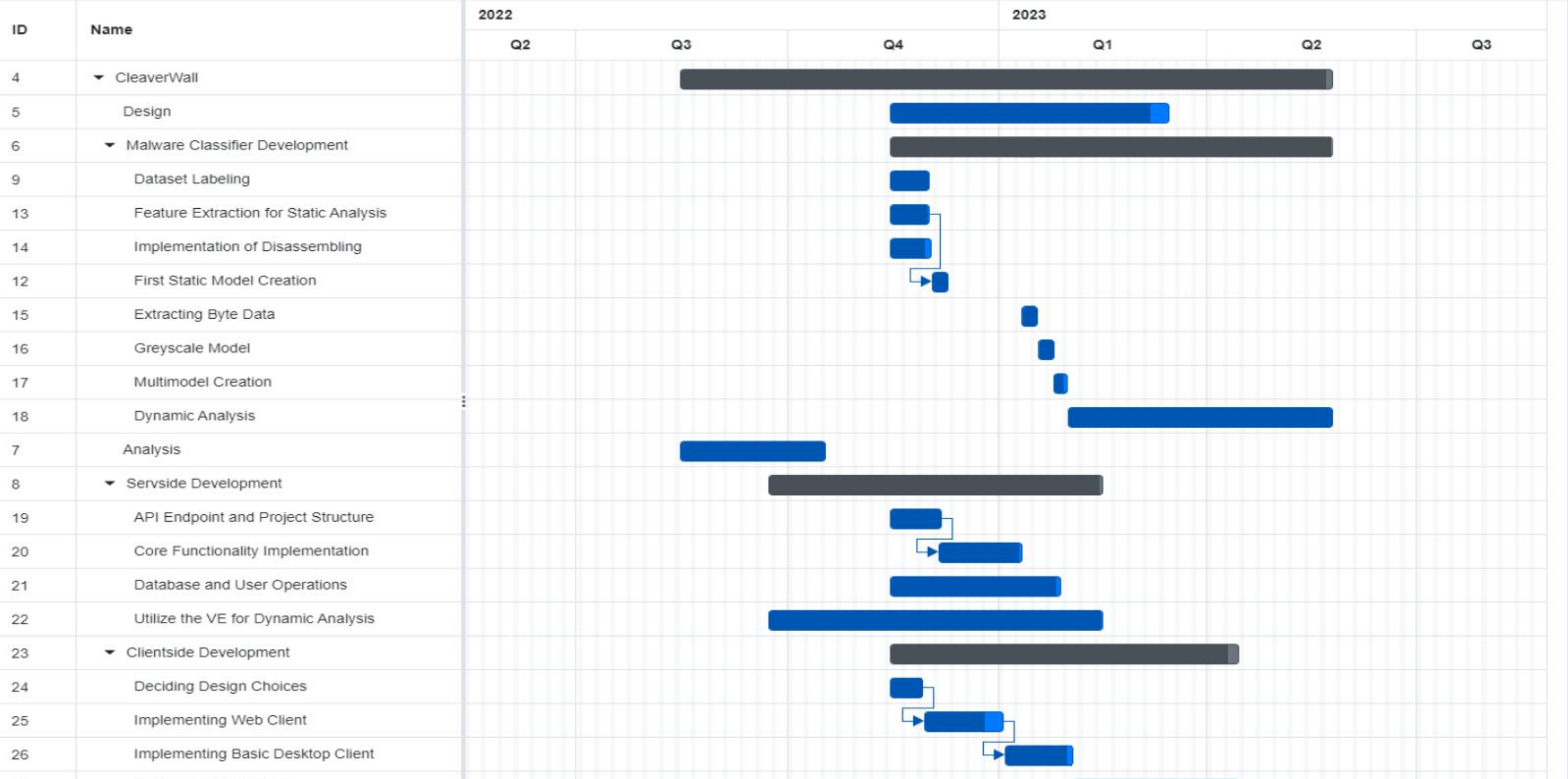
What is CleaverWall?

- Machine Learning based Anti-Malware system.
- Web and desktop applications.
- Static and dynamic analysis.
- Multiple malware classifiers.
- Focused on portable executable files.
- Our aim is to produce a light-weight application without lowering accuracy.

Why CleaverWall?

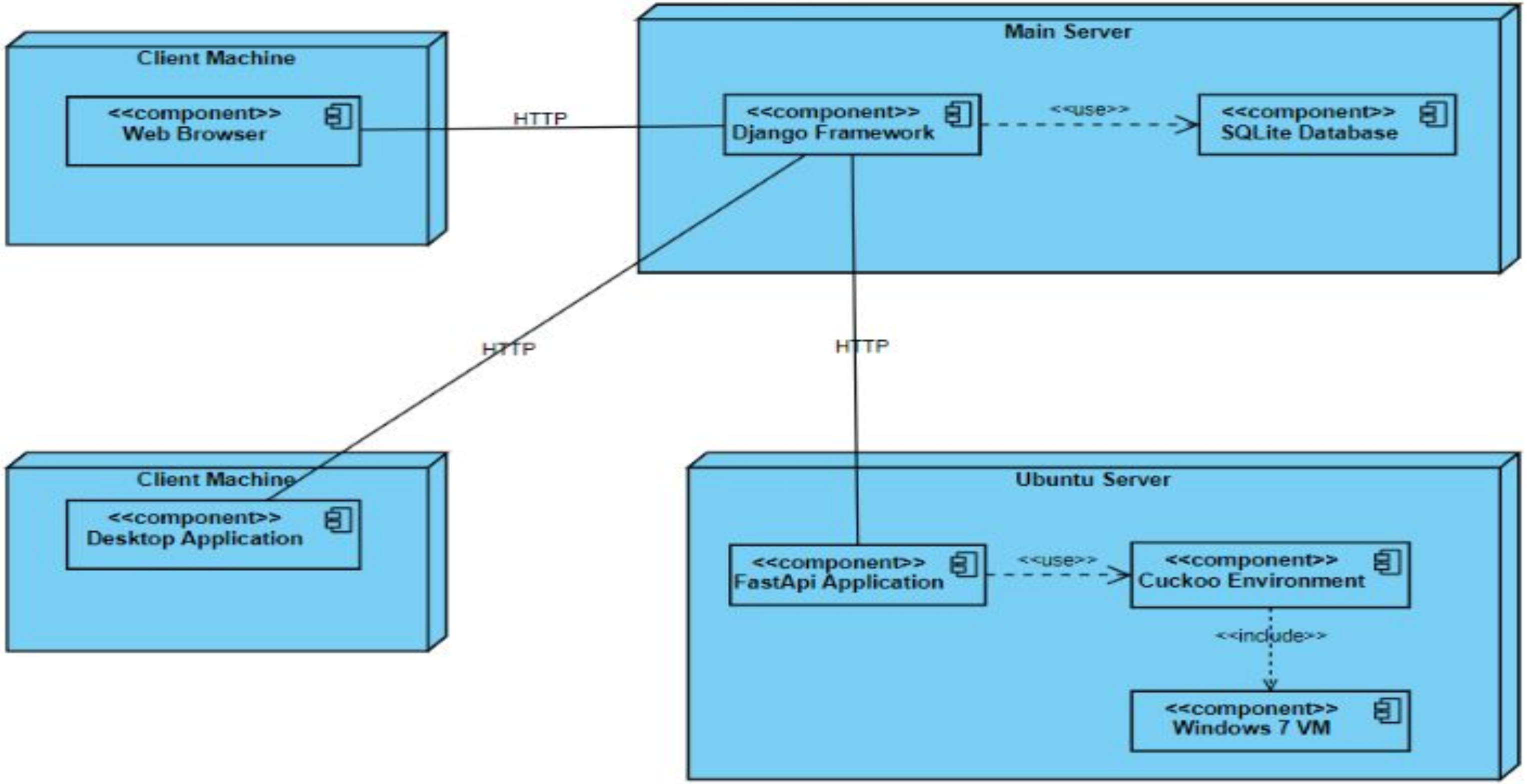
- Our competitors have problems:
  - High CPU and Memory usage
  - Datasets that need constant updates
  - High false-positive rates. Reliability is expensive!!
  - Open-source apps have high response times.
- Practical implementation of academic approaches

Development Process



ID	Name
4	CleaverWall
5	Design
6	Malware Classifier Development
9	Dataset Labeling
13	Feature Extraction for Static Analysis
14	Implementation of Disassembling
12	First Static Model Creation
15	Extracting Byte Data
16	Greyscale Model
17	Multimodel Creation
18	Dynamic Analysis
7	Analysis
8	Servside Development
19	API Endpoint and Project Structure
20	Core Functionality Implementation
21	Database and User Operations
22	Utilize the VE for Dynamic Analysis
23	Clientside Development
24	Deciding Design Choices
25	Implementing Web Client
26	Implementing Basic Desktop Client

System Design



Model Performances

- The first static model: 0.9835 validation accuracy and 0.04 false positive rate.
- The second static model: 0.9344 validation accuracy and 0.11 false positive rate.
- Although the performance metrics of the second model is worse than the first model, its operation elapsed time is less than the first model's elapsed time for large sized executables. Therefore, users can decide between the more accurate response and the quicker response for their needs.
- The dynamic model: 0.99 validation accuracy and there is not any false positive among 6586 benign executables in the validation set.