```
demperm/
│   └── server/
│       └── vote/
│           ├── app/
│           │   ├── settings.py
│           │   ├── urls.py
│           │   ├── wsgi.py
│           │   ├── security_config.py
│           │   ├── neo4j_config.py
│           │   └── main.py
│           │
│           ├── api/
│           │   ├── vote_controller.py
│           │   ├── domain_controller.py
│           │   ├── graph_controller.py
│           │   └── health_controller.py
│           │
│           ├── core/
│           │   ├── models/
│           │   │   ├── domain.py
│           │   │   ├── delegation.py
│           │   │   ├── score.py
│           │   │   ├── leaderboard.py
│           │   │   └── graph_view.py
│           │   ├── dto/
│           │   │   ├── vote_update_request_dto.py
│           │   │   ├── vote_update_response_dto.py
│           │   │   ├── my_votes_response_dto.py
│           │   │   ├── leaders_response_dto.py
│           │   │   ├── graph_response_dto.py
│           │   │   └── health_response_dto.py
│           │   ├── services/
│           │   │   ├── vote_service.py
│           │   │   ├── graph_service.py
│           │   │   ├── leaderboard_service.py
│           │   │   ├── stability_service.py
│           │   │   └── batch_service.py
│           │   ├── mappers/
│           │   │   ├── vote_mapper.py
│           │   │   ├── leaders_mapper.py
│           │   │   └── graph_mapper.py
│           │   └── rules/
│           │       ├── domains.py
│           │       ├── stability_rules.py
│           │       └── visibility_rules.py
│           │
│           ├── db/
│           │   ├── graph_entities/
```

```
│           │   │   ├── domain_node.py
│           │   │   ├── user_node.py
│           │   │   ├── delegation_edge.py
│           │   │   └── score_snapshot_node.py
│           │   ├── repository/
│           │   │   ├── domain_repository.py
│           │   │   ├── vote_repository.py
│           │   │   ├── graph_repository.py
│           │   │   └── snapshot_repository.py
│           │   ├── neo4j/
│           │   │   ├── init.cypher
│           │   │   └── example_data.cypher
│           │   └── migrations-notes.md
│           │
│           ├── tests/
│           │   ├── test_vote_controller.py
│           │   ├── test_domain_controller.py
│           │   ├── test_graph_controller.py
│           │   └── test_security_config.py
│           │
│           ├── Dockerfile
│           ├── manage.py
│           ├── requirements.txt
│           ├── pyproject.toml
│           ├── .env.example
│           └── README.md
│
├── .gitignore
├── .gitattributes
├── README.md
└── LICENSE
```

Au vue de la nature du projet, nous proposons une architecture en micro-services.

Pour le CICD, nous proposons d'utiliser des GithubActions.

Pour la base de données, nous proposons du Neo4J étant donné que nous sommes sur de la modélisation Graph.

Nous utiliserons également des API REST, avec Swagger pour la documentation.