ProductIQ- AI PRODUCT FEEDBACK

DAY-1 REPORT (17-12-2025)

1. Work Completed on Day-1

1.1 Frontend Development (React)

On Day-1, the frontend structure of the application was designed using **React** with a focus on user experience and extensibility.

The following frontend components and pages were planned and implemented using mock data:

- Landing page displaying the project title and entry point
- Login and signup pages for user access (UI only)
- Home dashboard with interactive cards showing feedback and feature statistics
- Feedback listing page
- Features priority page
- Roadmap visualization page
- Navigation bar with profile access

The frontend follows a **dark theme** with basic animations for improved user interaction. The application is structured to allow future integration with backend APIs and AI services without major changes.

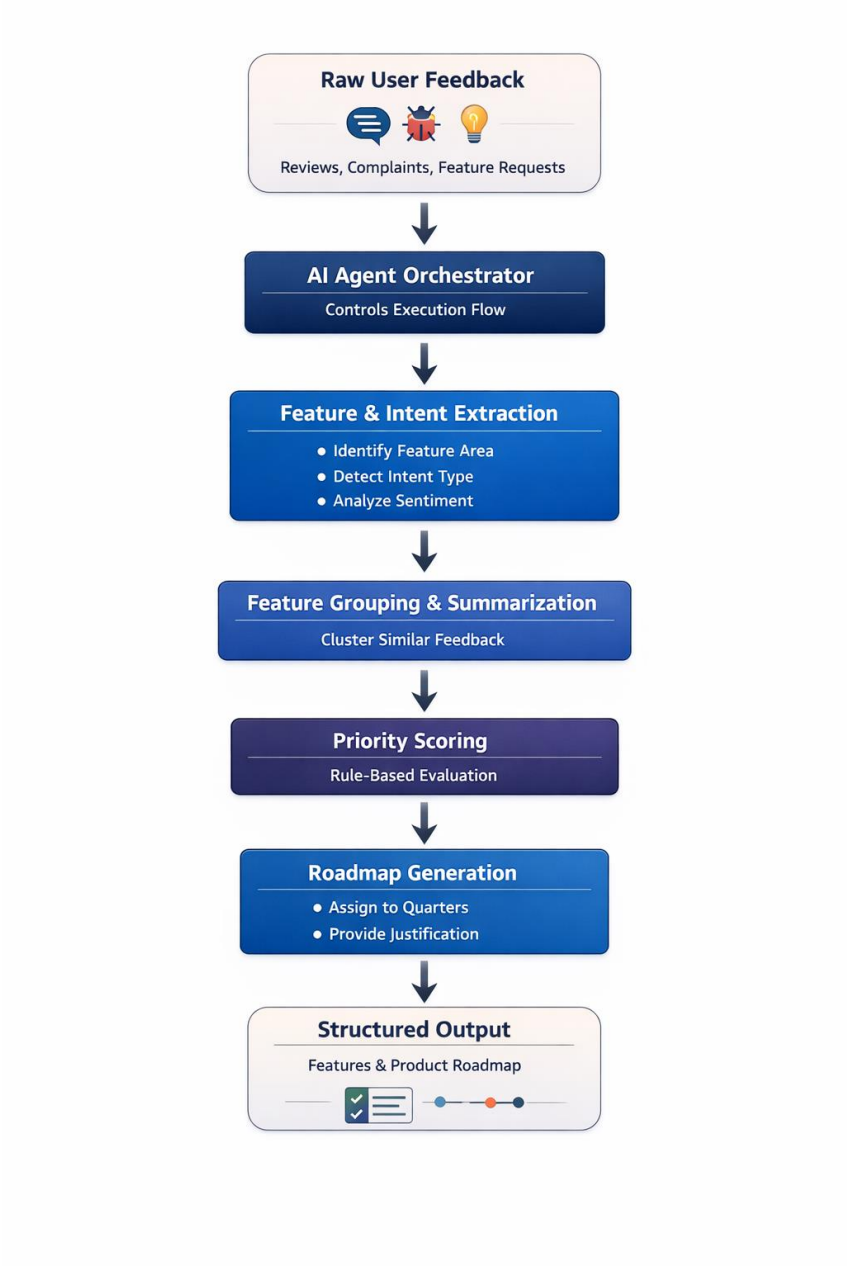
1.2 AI Engine Design and Initial Setup

The AI engine was designed as a **pipeline-based agent** responsible for transforming unstructured user feedback into structured product insights.

The AI agent design includes the following stages:

1. Feedback ingestion and preprocessing
2. Intent and feature extraction using a Large Language Model
3. Grouping of similar feedback into product features
4. Priority scoring based on defined logic
5. Roadmap generation with AI-generated justification

The AI engine was planned independently of the frontend to ensure modularity and maintainability.



2. Architecture Overview

The project follows a layered architecture:

- **Frontend:** React (UI and user interaction)
- **Backend:** Django with REST APIs (planned)
- **AI Layer:** Python-based AI engine for analysis and decision support
- **Database:** Relational database for structured data storage (planned)

This separation ensures scalability and clean integration between components.

3. Tools and Technologies Used (Day-1)

- React (Frontend UI)
- JavaScript (Frontend logic)
- Python (AI engine planning and setup)
- Git (version control)

4. Current Status

- Frontend UI structure created using mock data
- Core AI engine responsibilities and pipeline defined
- Project architecture finalized
- Backend and database integration planned for next phase

5. Next Steps (Day-2 Plan)

- Implement Django backend models and REST APIs
- Integrate AI engine with backend services
- Connect frontend components with backend APIs
- Test AI outputs with real data

6. Conclusion

Day-1 focused on establishing a strong foundation by designing the frontend interface and defining the AI engine architecture. This approach ensures that subsequent development phases can focus on integration and optimization without architectural changes.