**🧱 Milestone 1: Requirements Specification**

**🎯 Objective:**

To establish and document the **foundational libraries, frameworks, and environment configurations** for both the frontend and backend. This ensures **reproducibility, maintainability**, and an efficient **project onboarding** experience for all contributors.

**⚙️ Backend – FastAPI**

**📁 Key Libraries and Dependencies:**

txt

CopyEdit

fastapi

uvicorn

pydantic

python-dotenv

sentence-transformers

pinecone-client

httpx

requests

pandas

scikit-learn

**🔧 Backend Environment Setup:**

1. Create virtual environment:

bash

CopyEdit

python -m venv venv

source venv/bin/activate # On Windows: venv\Scripts\activate

1. Install dependencies:

bash

CopyEdit

pip install -r requirements.txt

1. .env file structure:

env

CopyEdit

WATSONX\_API\_KEY=your\_watsonx\_api\_key

PROJECT\_ID=your\_project\_id

MODEL\_ID=your\_model\_id

PINECONE\_API\_KEY=your\_pinecone\_key

PINECONE\_ENV=your\_pinecone\_environment

1. config.py uses Pydantic for secure env loading:

python

CopyEdit

from pydantic import BaseSettings

class Settings(BaseSettings):

watsonx\_api\_key: str

project\_id: str

model\_id: str

pinecone\_api\_key: str

pinecone\_env: str

class Config:

env\_file = ".env"

**🌐 Frontend – Streamlit**

**📁 Key Libraries and Dependencies:**

txt

CopyEdit

streamlit

streamlit-option-menu

requests

pandas

**🔧 Frontend Environment Setup:**

1. Create and activate a virtual environment (optional but recommended).
2. Install dependencies:

bash

CopyEdit

pip install -r requirements.txt

1. Suggested structure:

CopyEdit

frontend/

├── smart\_dashboard.py

├── components/

│ ├── summary\_card.py

│ ├── chat\_assistant.py

│ ├── eco\_tips.py

│ └── feedback\_form.py

**✅ Deliverables for Milestone 1**

| **Task** | **Description** | **Status** |
| --- | --- | --- |
| 📦 Package List | Backend and frontend requirements.txt defined | ✅ Done |
| 🔐 .env Template | Created with Watsonx + Pinecone keys | ✅ Done |
| 🗂️ Config Management | config.py securely loads env variables | ✅ Done |
| 📁 Folder Structure | Modular folders for API, services, components | ✅ Done |
| 📚 ReadMe Guide | Environment setup instructions documented | ✅ Done |

**📌 Outcome:**

By the end of Milestone 1, the development team has a **ready-to-use environment setup** for both backend and frontend. This enables immediate module development and ensures consistent setup across different machines.

**requirements.txt**

txt

CopyEdit

# === Frontend ===

streamlit==1.33.0

requests==2.31.0

# === Backend & API ===

fastapi==0.110.0

uvicorn==0.29.0

# === Configuration Management ===

python-dotenv==1.0.1

pydantic-settings==2.2.1

# === Embedding & Vector Search ===

sentence-transformers==2.6.1

pinecone-client==3.0.0

# === ML Models & Forecasting ===

scikit-learn==1.4.1.post1

pandas==2.2.2

matplotlib==3.8.4

**🔧 How to Use**

1. Save the above to a file named requirements.txt in your root directory.
2. Create and activate a virtual environment (recommended).
3. Run:

bash

CopyEdit

pip install -r requirements.txt

**✅ To install all dependencies:**

**Open your terminal and run:**

bash

CopyEdit

pip install -r requirements.txt

**📝 Checklist After Installation**

* ✅ No errors in the terminal
* ✅ All packages listed in requirements.txt installed successfully
* ✅ You can now run your app using:
  + streamlit run smart\_dashboard.py *(for frontend)*
  + uvicorn server:app --reload *(for backend if app is in server.py)*

**⚠️ If you face errors, try:**

* Upgrading pip:

bash

CopyEdit

pip install --upgrade pip

* Rechecking the Python version (python --version should ideally be 3.9 or 3.10)
* Creating a fresh virtual environment:

bash

CopyEdit

python -m venv venv

source venv/bin/activate # On Windows: venv\Scripts\activate

pip install -r requirements.txt