**Milestone 2: Environment Initialization & API Key Setup**

**🎯 Objective:**

To configure, validate, and securely manage external service credentials—specifically **IBM Watsonx** and **Pinecone**—for proper backend access and modular scalability.

**🧩 Step 1: Create .env File**

Create a .env file in your project root directory and store your sensitive API keys:

env

CopyEdit

# IBM Watsonx Credentials

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

PROJECT\_ID=your\_project\_id

MODEL\_ID=your\_model\_id

# Pinecone Credentials

PINECONE\_API\_KEY=your\_pinecone\_api\_key

PINECONE\_ENV=your\_pinecone\_environment

PINECONE\_INDEX=smartcity-policies

🔒 **Note**: Never upload .env to GitHub. Add it to .gitignore.

**⚙️ Step 2: Load Environment Variables with config.py**

Use pydantic-settings to manage environment variables securely and elegantly.

**📄 config.py (inside core/ or utils/)**

python

CopyEdit

from pydantic\_settings import BaseSettings

class Settings(BaseSettings):

watsonx\_api\_key: str

project\_id: str

model\_id: str

pinecone\_api\_key: str

pinecone\_env: str

pinecone\_index: str = "smartcity-policies"

class Config:

env\_file = ".env"

settings = Settings()

**🌐 Step 3: Use Settings in Your Code**

Example usage in a FastAPI route or service:

python

CopyEdit

from core.config import settings

import pinecone

pinecone.init(api\_key=settings.pinecone\_api\_key, environment=settings.pinecone\_env)

pinecone\_index = pinecone.Index(settings.pinecone\_index)

For Watsonx:

python

CopyEdit

headers = {

"Authorization": f"Bearer {settings.watsonx\_api\_key}",

"Content-Type": "application/json"

}

**✅ Deliverables for Milestone 2**

| **Task** | **Description** | **Status** |
| --- | --- | --- |
| 🔑 .env File Created | API keys securely stored | ✅ Done |
| ⚙️ config.py Module | Dynamic config loading | ✅ Done |
| 🔄 Integration Test | Validated Pinecone & Watsonx access | ✅ Done |
| 🔐 Security | .env added to .gitignore | ✅ Done |

**📌 Outcome:**

* Secrets are loaded **securely and modularly**.
* External services like **IBM Watsonx** and **Pinecone** are now **fully configured**.
* Developers can change keys/environment without touching application logic.

**Activity 2.1: Generate API Keys**

**🔷 Step 1: Generate IBM Watsonx Granite API Credentials**

**📍 Source:** [**IBM Cloud Dashboard**](https://cloud.ibm.com)

**✅ Follow these steps:**

1. **Log in** to your IBM Cloud account.
2. Go to **Watsonx** > **Granite Foundation Models**.
3. Create or open a **Watsonx project**.
4. Under **Service Credentials**, click **"Create new credential"**.
5. Copy the following values:
   * **WATSONX\_API\_KEY**
   * **PROJECT\_ID**
   * **MODEL\_ID** (for example, "granite-13b-chat")

🔐 Save these in your .env:

env

CopyEdit

WATSONX\_API\_KEY=your\_generated\_key

PROJECT\_ID=your\_project\_id

MODEL\_ID=granite-13b-chat

**🔷 Step 2: Generate Pinecone API Key & Environment**

**📍 Source:** [**Pinecone Console**](https://app.pinecone.io)

**✅ Follow these steps:**

1. Log in to your [Pinecone account](https://app.pinecone.io).
2. Create a new **index** (e.g., smartcity-policies) with dimension **384**.
3. Navigate to **API Keys** in the sidebar.
4. Click **Create API Key** or use the default.
5. Copy:
   * **PINECONE\_API\_KEY**
   * **PINECONE\_ENVIRONMENT** (e.g., gcp-starter, us-west1-gcp)

🔐 Save these in your .env:

env

CopyEdit

PINECONE\_API\_KEY=your\_pinecone\_key

PINECONE\_ENV=your\_environment

PINECONE\_INDEX=smartcity-policies

**🛡️ Best Practices**

* ✅ Add .env to .gitignore to prevent leaks:

txt

CopyEdit

.env

* ✅ Create a .env.example file for reference (without real keys):

env

CopyEdit

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

PROJECT\_ID=your\_project\_id

MODEL\_ID=granite-13b-chat

PINECONE\_API\_KEY=your\_pinecone\_key\_here

PINECONE\_ENV=your\_pinecone\_env\_here

PINECONE\_INDEX=smartcity-policies

**✅ Outcome:**

By the end of Activity 2.1, you will have:

* Valid **Watsonx Granite API** credentials
* A functioning **Pinecone index**
* Your .env file securely configured and ready for backend use

**Activity 2.2: Define .env File**

**📝 Current .env Content (Looks Great):**

env

CopyEdit

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

WATSONX\_PROJECT\_ID=your\_project\_id

WATSONX\_URL=https://your-region.ml.cloud.ibm.com

WATSONX\_MODEL\_ID=ibm/granite-13b-instruct-v2

PINECONE\_API\_KEY=your\_pinecone\_key

PINECONE\_ENV=your\_pinecone\_env

INDEX\_NAME=smartcity-policies

Your keys and values are correctly defined with:

* Watsonx’s API key, project ID, model ID, and region URL
* Pinecone’s API key, environment, and index name

**🛡️ Next Step: Secure the .env**

**✅ Add to .gitignore (if not already):**

txt

CopyEdit

.env

**🧪 Optional: Verify Load with config.py**

You can quickly test loading the environment like this:

python

CopyEdit

# config.py

from pydantic\_settings import BaseSettings

class Settings(BaseSettings):

watsonx\_api\_key: str

watsonx\_project\_id: str

watsonx\_url: str

watsonx\_model\_id: str

pinecone\_api\_key: str

pinecone\_env: str

index\_name: str

class Config:

env\_file = ".env"

settings = Settings()

print(settings.watsonx\_project\_id)

🧠 This ensures everything is loading from .env correctly.