# Project Documentation: Academix

## Introduction

Academix is a project designed to help university students study more efficiently. It offers a set of features aimed at enhancing the study experience. The main features of Academix include:

* **Study Guide Creation**: Academix allows users to create a personalized study guide by adding courses of interest.
* **Question Generation**: With the integration of the OpenAI API, Academix generates practice questions based on the selected study guide courses.
* **Study Guide Management**: Academix provides functionalities for adding and removing courses from the study guide.

### Installation

To get started with Academix, follow these steps:

#### Prerequisites

* Python 3.x installed on your system
* Access to the OpenAI API (API key required)

##### Step 1: Clone the Repository

1. Open a terminal or command prompt.
2. Navigate to the directory where you want to store the project.
3. Clone the Academix repository by running the following command:

git clone [https://github.com/berkeokur/Academix]

##### Step 2: Set Up the Environment

1. Install *virtualenv* package using *pip*:

pip install virtualenv

OR

pip3 install virtualenv

1. Create a virtual environment for Academix by running the following command in the project's root directory:

* Windows:

Python3 -m venv env

* Mac/Linux:

venv env

1. Activate the virtual environment:

* For Windows:

env\Scripts\activate

* For macOS/Linux:

source env/bin/activate

##### Step 3: Install Dependencies

1. Install the required Python packages by running the following command:

pip install -r requirements.txt

OR

pip3 install -r requirements.txt

##### Step 4: Set Up the .env File

1. In the project's root directory, create a file named .env.
2. Open the .env file in a text editor.
3. Add the following lines to the file:

DISCORD\_API\_KEY="MTExMDI4ODg1Njc0MzQyNDE1MA.G6uHwf.vBH5AGscNY4izFv3hs3F02XneLqNYlGpKwN-n8"

OPENAI\_API\_KEY="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"

GUILD= 1075771998942281738

ht="193.204.40.146"

user="gr1"

pwd="group1#23"

db="group1"

##### Step 5: Start Academix

1. In the terminal or command prompt, ensure that you are still in the project's root directory and the virtual environment is active.
2. Run the following file inside Visual Studio Code to start Academix:

main.py

## Usage

Academix provides a set of commands and features to help university students study more efficiently. Here's an overview of the main functionalities:

### Autocompletion

When entering course or topic names, Academix provides autocompletion suggestions to help you select the correct option. The autocompletion feature will automatically suggest available course and topic names as you type.

### Registering and Logging in

#### Register

The register command allows you to register your account to the database. You can use the command as follows:

* */access register <email> <password>*

#### Login

The login command allows you to use the functionalities of our application. You can use the command as follows:

* */access login <email> <password>*

### Course management

#### Creating a Study Guide

The add\_course command allows you to add a course to your study guide. You can use the command as follows:

* */study\_guide add\_course <course\_name>*

#### Adding Topics

The *add\_topic* command enables you to add a topic to an existing course in your study guide. Use the command as follows:

* */study\_guide add\_topic <course\_name> <topic\_name>*

#### Removing Courses and Topics

You can remove a course from your study guide using the *remove\_course* command:

* */study\_guide remove\_course <course\_name>*

To remove a topic from a course, use the *remove\_topic* command:

* */study\_guide remove\_topic <course\_name> <topic\_name>*

### Generating Questions

The *generate\_question* command generates a question based on the specified course, topic, and difficulty level. Here's how to use the command:

* */gpt generate\_question <course\_name> <topic\_name> <difficulty\_level>*

### Asking GPT

The *ask\_gpt* command allows you to ask a question to the GPT model. Temperature is the creativity of GPT and it takes a value between 0 and 1(default=0.5). Here's an example of how to use it:

* */gpt ask\_gpt <your\_question> <temperature>*