

## 🦋 PDF Learning Assistant (Task-4)

### Project Goal

Develop a **PDF Learning Assistant** that **summarizes PDF documents** and **generates quizzes**, helping users learn efficiently. The project leverages **AI agents** via the **Gemini CLI** and **Context7 MCP server**.

### 🔧 Project Features

#### 1. AI-Driven Summarization

- Uses a **Summarizer Agent** (gemini-2.5-flash) to create **concise and accurate summaries** of uploaded PDFs.
- Offers **multiple display styles** for the summary.



```
tips for getting started:
. Ask questions, edit files, or run commands.
. Be specific for the best results.
. /help for more information.

/mcp

onfigured MCP servers:

  context7 - Ready (2 tools)
Tools:
- get-library-docs
- resolve-library-id

Using: 2 GEMINI.md files | 1 MCP server

> /m

mcp          Manage configured Model Context Protocol (MCP) servers
memory       Commands for interacting with memory
model        Opens a dialog to configure the model
compress     Compresses the context by replacing it with a summary
theme        Change the theme
vim          Toggle vim mode on/off
terminal-setup  Configure terminal keybindings for multiline input (VS Code, Cursor, Windsurf)
C:\Quiz Generator Agent                               no sandbox (see /docs)
```

#### 2. Quiz Generation

- ❖ Uses a **Quiz Generator Agent** (gemini-2.5-flash) to create quizzes based on PDF content.
- ❖ Generates **15-20 questions** of various types:
  - **MCQ** – Multiple Choice Questions (4 options)
  - **T/F** – True/False
  - **F/B** – Fill-in-the-Blank

### 3. [Interactive Quiz Interface](#)

- ❖ Users can **answer questions directly** in the Streamlit app.
- ❖ Provides **real-time scoring** and **instant feedback** for correct and incorrect answers.



## ⚙️ [Technical Implementation](#)

### 1. [Connecting to MCP Server](#)

- ❖ Sign in to **Context7**, create an **API key**, and configure the **Gemini CLI**.
- ❖ `.gemini/settings.json` contains server commands and API key configuration.

```
{
  "mcpServers": {
    "context7": {
      "command": "npx",
      "args": [
        "-y",
        "@upstash/context7-mcp"
      ],
      "env": {
        "CONTEXT7_API_KEY": "ctx7sk-a534d1fa-c5b8-492b-8ad6-"
      }
    }
  }
}
```

### 2. [Streamlit App](#)

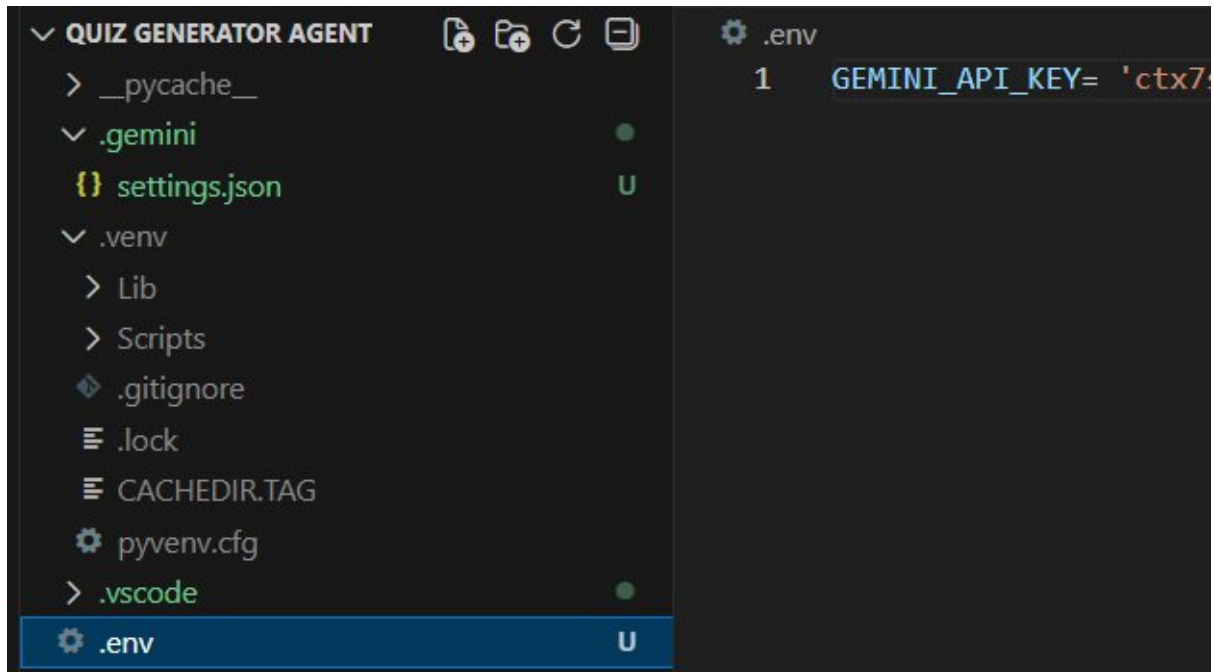
- ❖ **PDF upload and text extraction** via pypdf.
- ❖ **Session state management** to store summary, quiz, answers, and errors.

### 3. [Error Handling](#)

- ❖ Handles errors in **PDF text extraction**, **JSON parsing**, and **agent execution**.

#### 4. Environment Setup

- ❖ .env file stores GEMINI\_API\_KEY and optional GEMINI\_API\_BASE.
- ❖ Dependencies: streamlit, pypdf, python-dotenv, openai.



#### User Workflow

1. Upload a PDF file.
2. Extract text automatically.
3. Summarizer Agent generates a **summary**.
4. Quiz Creator Agent generates a **quiz**.
5. User answers the quiz and receives **instant scoring with correct answers highlighted**.

#### Tech Stack

- ❖ **Backend / Agents:** Gemini CLI, Context7 MCP Server
- ❖ **Frontend / UI:** Streamlit
- ❖ **PDF Processing:** pypdf
- ❖ **Environment Management:** python-dotenv
- ❖ **AI Integration:** openai-agents SDK

## Outcome

A **fully interactive PDF Learning Assistant** that:

- ❖ Simplifies learning by **summarizing large documents**.
- ❖ Reinforces knowledge through **automatically generated quizzes**.
- ❖ Provides **real-time feedback and scoring** to enhance learning efficiency.