

# Research Paper Platform

A comprehensive platform for analyzing, summarizing, and generating research papers. This application allows users to upload research papers, chat with them using AI, get summaries, and even generate new research papers on any topic.

## Table of Contents

1. [Features](#)
2. [Prerequisites](#)
3. [Installation](#)
  - [Setting Up the Backend](#)
    - [Using venv](#)
    - [Using Conda](#)
  - [Setting Up the Frontend](#)
4. [Running the Application](#)
5. [Using the Application](#)
6. [Troubleshooting](#)
7. [FAQ](#)

## Features

- **Upload and Analyze Papers:** Upload PDF, DOCX, TXT, or MD files for analysis
- **AI-Powered Chat:** Ask questions about your research paper and get intelligent answers
- **Paper Summarization:** Get comprehensive summaries of your papers
- **Paper Generation:** Generate new research papers on any topic
- **Download Options:** Save summaries and generated papers

## Prerequisites

Before you begin, ensure you have the following installed:

- **Python 3.10+:** For the backend application
  - Option 1: Install from [python.org](#)
  - Option 2: Install via [Anaconda](#) or [Miniconda](#)
- **Node.js 18.0+:** For the frontend application
  - Download from [nodejs.org](#)
- **Google API Key:** For AI capabilities (Gemini API)
  - Get from [Google AI Studio](#)

## Installation

Clone this repository to your local machine:

```
git clone https://github.com/yourusername/research-paper-platform.git
cd research-paper-platform
```

### Setting Up the Backend

You can choose between using Python's virtual environment (venv) or Conda for managing dependencies.

#### Using venv

1. **Create a virtual environment:**

For Windows:

```
python -m venv venv
venv\Scripts\activate
```

For macOS/Linux:

```
python3 -m venv venv
source venv/bin/activate
```

2. **Install backend dependencies:**

```
pip install -r requirements.txt
```

#### Using Conda

1. **Create a new Conda environment:**

```
conda create -n research-paper python=3.10
conda activate research-paper
```

2. **Install pip inside the Conda environment:**

```
conda install pip
```

3. **Install backend dependencies:**

```
pip install -r requirements.txt
```

If any package installation fails, you can try installing through conda:

```
conda install -c conda-forge package-name
```

4. **Set up your Google API Key:**

For Windows:

```
set GOOGLE_API_KEY=your_google_api_key_here
```

For macOS/Linux:

```
export GOOGLE_API_KEY=your_google_api_key_here
```

For Conda environments (add to environment):

```
conda env config vars set GOOGLE_API_KEY=your_google_api_key_here
conda activate research-paper # Reactivate to apply changes
```

Alternatively, you can update the default key in `app.py` (line 36).

## Setting Up the Frontend

1. **Navigate to the frontend directory:**

```
cd frontend
```

2. **Install frontend dependencies:**

```
npm install
```

## Running the Application

You'll need to run both the backend and frontend separately.

### Running the Backend

1. **From the root directory, start the backend server:**

Make sure your virtual environment (venv or conda) is activated, then:

```
uvicorn app:app --reload
```

The backend will start on <http://localhost:8000>

2. **Check if the backend is running correctly:**

Open your browser and go to <http://localhost:8000/docs> to see the API documentation.

### Running the Frontend

1. **In a new terminal, navigate to the frontend directory:**

```
cd frontend
```

2. **Start the frontend development server:**

```
npm run dev
```

The frontend will start on <http://localhost:5173>

### 3. Access the complete application:

Open your browser and go to <http://localhost:5173>

## Using the Application

---

### 1. Uploading a Research Paper

1. Click on "Get Started" from the homepage or navigate to the Research page
2. Click the "Upload Paper" button
3. Select a research paper file (PDF, DOCX, TXT, or MD)
4. Wait for the upload and processing to complete

### 2. Chatting with Your Paper

1. After uploading a paper, go to the "Chat with Paper" tab
2. Type your question in the text box
3. Press Enter or click "Send"
4. View the AI-generated response

### 3. Viewing and Downloading Summaries

1. Click on the "View Summary" tab
2. Review the automatically generated summary
3. Click "Download Summary" to save it as a DOCX file

### 4. Generating a New Paper

1. Click on the "Generate New Paper" tab
2. Enter a title for your paper
3. Select the paper type (Research, Survey, Methodology, or Case Study)
4. Add any additional details or requirements
5. Click "Generate Paper"
6. Once generated, download in your preferred format

## Troubleshooting

---

### Common Issues and Solutions

#### 1. Backend Won't Start

- Ensure Python 3.10+ is installed ( `python --version` or `conda list python` )
- Verify all dependencies are installed ( `pip list` or `conda list` )
- Check if the port 8000 is already in use
- For Conda: Make sure you're in the correct environment ( `conda env list` )

#### 2. Frontend Won't Start

- Ensure Node.js 18.0+ is installed ( `node --version` )
- Verify all dependencies are installed ( `npm list` )
- Check if the port 5173 is already in use

#### 3. Paper Upload Fails

- Ensure the file is in a supported format (PDF, DOCX, TXT, or MD)
- Check if the file size is too large (max 10MB)
- Verify your internet connection

#### 4. API Key Issues

- Ensure your Google API Key is correctly set (check with `echo $GOOGLE_API_KEY` or `echo %GOOGLE_API_KEY%` )
- For Conda: Verify with `conda env config vars list`
- Verify the API key has access to Gemini API

#### 5. Generated Papers Look Incomplete

- Try providing more detailed instructions
- Check if the paper generation is still in progress

#### 6. Conda Environment Issues

- If you see "command not found" errors, ensure the package is installed in your environment
- Try running `conda update conda` to ensure conda is up to date
- For library conflicts, create a fresh environment and install packages one by one

## FAQ

---

**Q: Do I need a Google API Key?**

A: Yes, the application uses Google's Gemini API for AI capabilities. You can get a free API key from Google AI Studio.

**Q: What types of papers can I upload?**

A: The system supports PDF, DOCX, TXT, and MD file formats.

**Q: Is there a limit to how many papers I can upload?**

A: The system processes one paper at a time, but you can upload as many as you need.

**Q: How accurate are the paper summaries?**

A: The summaries are AI-generated and provide a good overview of the paper's content, but they should be reviewed for accuracy.

**Q: Can I customize the generated papers?**

A: Yes, you can provide specific details, requirements, and choose the paper type before generation.

**Q: Are my papers stored securely?**

A: Papers are processed temporarily and not permanently stored on the server.

**Q: Which environment is recommended - venv or Conda?**

A: Both work well. Use venv for simplicity if you only need Python dependencies. Use Conda if you require additional non-Python packages or prefer environment isolation.

**Q: Can I use a different port for the backend or frontend?**

A: Yes, for the backend use `uvicorn app:app --reload --port 8080`. For the frontend, modify the `vite.config.js` file.

---

## License

This project is licensed under the MIT License - see the LICENSE file for details.

## Acknowledgments

---

- Built with FastAPI, React, and Google Gemini API
- Utilizes Tailwind CSS for styling
- Special thanks to all contributors