

Contribution: Enhancing the Research Assistant Project

Detailed Contribution Plan

Contribution: Enhancing the Research Assistant Project

1. Improved README.md

- Reorganized the structure to include Overview, Features, Installation, Usage, and Contributing sections.
- Added specific setup commands and usage examples.
- Included troubleshooting tips and architecture diagrams for clarity.

2. Added CONTRIBUTING.md

- Provided clear guidelines for forking the repository, coding standards, and submitting pull requests.
- Included instructions for raising issues and testing code.

3. Enhanced Code Comments and Docstrings

- Added module-level docstrings explaining functionality.
- Improved inline comments for complex logic in core files like main.py and components/chat.py.

4. Created USAGE.md

- Detailed usage examples with step-by-step instructions for creating projects, loading files, generating summaries, and using chat features.

5. Added Example Data

- Introduced a "samples" directory containing example PDFs and their generated summaries for testing.

6. Improved Logging and Error Handling

Contribution: Enhancing the Research Assistant Project

- Enhanced log messages in running_logs.log for better debugging.
- Added try-except blocks for robust error management.

7. Provided Example Pull Request

- Drafted a pull request template highlighting the above changes.

Sample Code Example:

```
# Example: Enhanced logging
```

```
import logging
```

```
logging.basicConfig(filename='logs/running_logs.log', level=logging.INFO)
```

```
def summarize_document(document_path):
```

```
    try:
```

```
        logging.info(f"Summarizing document: {document_path}")
```

```
        # Summarization logic
```

```
    except Exception as e:
```

```
        logging.error(f"Error summarizing document: {e}")
```

Contact the maintainers for further clarification or suggestions.