DocuQuery: AI-Powered PDF Knowledge Assistant Using Google PALM

Planning logic is a critical part of the project documentation. It outlines the step-by-step approach the team will take to design, develop, and deliver **DocuQuery**. This section should include timelines, milestones, task breakdowns, and resource allocation. Below is a detailed template to help you structure the **Planning Logic** for the project.

1. Project Overview

- Provide a brief summary of the project goals and objectives.

- Highlight the key deliverables and outcomes.

Example: "The goal of DocuQuery is to develop an AI-powered PDF knowledge assistant that enables users to interact with PDF documents using natural language queries, summarization, and accessibility features. The project will be delivered in four phases, with key milestones for design, development, testing, and deployment."

2. Project Phases

- Break the project into logical phases.

- Define the objectives and deliverables for each phase.

Example:

Phase	Objectives	Deliverables
Phase 1: Research and Planning	**Understand user needs, define requirements, and finalize the technology stack.**	**Project plan, requirement specifications, and architecture design.**
Phase 2: Design and Prototyping	**Create wireframes, UI/UX designs, and a working prototype.**	**Wireframes, UI/UX designs, and a functional prototype.**
Phase 3: Development	**Build the core features and integrate AI capabilities.**	**Fully functional backend, front end, and AI integration.**
Phase 4: Testing and Deployment	**Test the system for bugs, performance, and usability. Deploy the solution.**	**Test reports, bug fixes, and a deployed application.**

3. Timeline and Milestones

- Create a timeline with key milestones and deadlines.

- Use a Gantt chart or table to visualize the timeline.

Example:

Milestone	Deadline	Description
Project Kickoff	**Week 1**	**Team alignment on goals, roles, and responsibilities.**
Requirement Specifications	**Week 2**	**Finalize user requirements and technical specifications.**
Architecture Design	**Week 3**	**Complete the solution architecture and technology stack.**
Prototype Development	**Week 4-5**	**Develop and test a working prototype.**
Core Feature Development	**Week 6-8**	**Build and integrate core features (search, summarization, Q&A).**
AI Integration (Google PALM)	**Week 9-10**	**Integrate Google PALM for natural language processing.**
Testing and Bug Fixing	**Week 11-12**	**Conduct system testing and fix bugs.**
Deployment and Launch	**Week 13**	**Deploy the application and make it available to users.**

4. Task Breakdown

- Break down each phase into specific tasks.

- Assign tasks to team members and define dependencies.

Example:

Task	Assigned To	Dependencies	Deadline
Conduct user research	**Member 1**	**None**	**Week 1**
Define technical requirements	**Member 2**	**User research completed**	**Week 2**
Design UI/UX wireframes	**Member 3**	**Requirements finalized**	**Week 3**
Develop backend architecture	**Member 4**	**Requirements finalized**	**Week 4**

Task	Assigned To	Dependencies	Deadline
Build PDF processing engine	Member 1	Backend architecture completed	Week 5
Integrate Google PALM API	Member 2	Backend and PDF engine completed	Week 9
Conduct system testing	Member 3	All features developed	Week 11
Deploy application	Member 4	Testing completed	Week 13

5. Resource Allocation

- List the resources (human, technical, and financial) required for each phase.

- Ensure resources are allocated efficiently.

Example:

Resource	Phase	Description
Team Members (4)	All phases	Developers, designers, testers, and project manager.
Google PALM API Access	Development	Access to Google PALM for natural language processing.
Cloud Infrastructure (AWS/GCP)	Development & Deployment	Hosting and scaling the application.
Design Tools (Figma, Adobe XD)	Design	Tools for creating wireframes and UI/UX designs.
Testing Tools (Jira, Selenium)	Testing	Tools for bug tracking and automated testing.

6. Risk Management

- Identify potential risks and mitigation strategies.

- Include technical, operational, and timeline risks.

Example:

Risk	Impact	Mitigation Strategy
Delays in AI integration	**Project timeline delay**	**Allocate extra time for testing and debugging.**
Budget overruns	**Financial strain**	**Monitor expenses closely and prioritize essential features.**
Technical challenges with PDF processing	**Feature delays**	**Use proven libraries (e.g., PyPDF2) and conduct early prototyping.**
User adoption issues	**Low engagement**	**Conduct user testing early and incorporate feedback into the design.**

7. Monitoring and Evaluation

- Define how progress will be tracked and evaluated.

- Include key performance indicators (KPIs) and review mechanisms.

Example:

- KPIs: Feature completion rate, bug resolution rate, user satisfaction score.

- Review Meetings: Weekly team meetings to track progress and address issues.

- Tools: Jira for task tracking, GitHub for version control, and Google Sheets for budget tracking.

Task Allocation for Planning Logic

To ensure equal participation, divide the tasks among your team members as follows:

NIKITA

- Writing the Project Overview and Task Breakdown sections.

- Focus on defining the project goals and breaking down tasks.

KESHAV GARG

- Create the Timeline and Milestones and Resource Allocation sections.

- Use tools like Gantt charts or tables to visualize the timeline.

PRATYKSHA

- Document the Risk Management and Monitoring and Evaluation sections.
- Identify risks and define strategies for tracking progress.

NILESHWAR

- Compile and edit the entire Planning Logic section.
- Ensure consistency in tone, style, and formatting.

Collaborative Tasks

- Review and Feedback: After completing their sections, each member should review and provide feedback on others' work.
- Final Compilation: Assign one member (e.g., Member 4) to compile the sections and ensure consistency.