**Afya-Smart-ML Project**

**Objectives**:

Briefly describe the goals and what the project intends to achieve, including its impact on users or the industry.

**Scope:**

Define the boundaries of the project, and what will and won't be included.

**Team Member: Roles**

Names & Roles: List all project members along with their roles and responsibilities.

**Development Tools:**

List the software, frameworks, libraries, and tools used for development, specifying versions where applicable.

**Project Management Tools**:

Specify the tools used for tracking progress, tasks, and collaboration (e.g., Jira, Trello, Slack).

**System Architecture:**

Provide a high-level overview of the system's architecture, possibly with diagrams showing how different components interact.

(Debalina’s diagram would go here) along with concise bullet points explaining how different components interact and also mentioning how data is flowing through our system. Data flow normally has its separate section but since we only have PCP dummy data and some consults, we can briefly talk about it here only.)

**API Design**

Describe the structure of the APIs, including endpoints, request/response formats, and authentication methods.

**Machine Learning Models**

**Model Selection:**

Discuss the machine learning models chosen, including the rationale behind their selection.

( here we can discuss all the test trials with different models you guys have done and based on those results, why we decided to with GPT 3.5).

**ML Model Integration**: Explain how machine learning models are integrated and interact with the back-end.

(something along the lines of the routing part that Debalina discussed today, we can talk about the kind of data structure we are using to store and display the responses generated as we discussed + any other important backend logic utilised for manipulating the model generated responses)

**Model Evaluation:** Detail the methods used for evaluating model performance

(as mentioned under the Performance Testing subsection)

**Testing**

**Unit Testing**: Describe the approach to unit testing for both front-end and back-end components.

(just mention for what components are you writing unit tests for and what are you checking)

**Integration Testing:** Outline how different components are tested together

(possibly after today, we can add some tests for integration testing, once we have some output to display on the Front–End.

**Performance Testing:** Detail the performance metrics that will be measured and the tools used for testing.

(up for discussion if we need to include this, I surfed what are good evaluation metrics for NLP tasks and they are either (a) very simple like accuracy and precision and F1score or (b) very specific to the NLP task, so for example for Question Answering, we have exact matching, where we compare them with ground truth labels, we can decide this later).

**Deployment**

**Deployment Strategy:** Describe the strategy for deploying the application, including continuous integration and continuous deployment (CI/CD) pipelines if used.

**Environment Configuration**: List the configurations for different environments (development, staging, production).

**Security Measures**

**Data Protection:** Outline the measures in place to protect user data and privacy.

**Application Security:** Describe the security protocols and practices implemented to safeguard the application.

**Assumptions and Dependencies:**

List any assumptions made during the planning and development processes and any dependencies on external services or components.

(we can list some of the assumptions we made while designing the front-end, like assuming the PCP has access to some form of econsult notes when they interact with our application.

**Risks and Mitigations:**

Identify potential risks to the project and how they will be mitigated.

(something along the lines of ethical use of AI and the security measures taken to secure important medical data).