**Product Requirements Document: AI-Powered Presentation Generator**

**1. Introduction**

This document outlines the product requirements for an AI-powered presentation generation platform. The system will leverage a multi-agent architecture to intelligently create compelling and visually appealing presentations based on user inputs. The platform will streamline the presentation creation process, from initial concept to final output, by automating content research, data analysis, visual design, and layout.

**2. Guiding Principles**

The following principles will guide the development of this platform:

**Technology Stack (Starting Point):**

* The output of each agent will be standardized using Base Model in Pydantic library making communication seamless between agents.
* Agents will be built using Pydantic AI, Agent Module which provides multi LLM access through a single window.
* Logging will be maintained using Pyrantic Log Fire.
* Agent orchestration and chaining will be implemented using LangGraph
* Pre-built capabilities and tools available on the MCP server will be leveraged where possible.
* **FastAPI:** For high-performance communication between the frontend and the Director agents, and potentially between the agents themselves. FastAPI is extremely fast, includes automatic API documentation, and has native integration with Pydantic, which simplifies development.
* JSON as the primary communication protocol between agents, with the slide's visual content embedded as an HTML string within the JSON payload.

**Development Focus:**

* Significant attention will be given to the quality and clarity of prompts for each agent.
* A clear structure will be established for how agents utilize short-term (contextual) and long-term memory.

**3. High Level Agent Workflow**

The platform will utilize a series of specialized agents, orchestrated by a central "Director" agent, to manage the presentation creation process. The workflow is designed to be iterative and collaborative, allowing for user feedback at key stages.

**UX Architect**

(layout, html, slide themes)

Director (Inbound)

**Researcher**

(content research, internal + external)

User/ Frontend

**Visual Designer**

(images, gifs, videos, icons / vectors)

**Data Analyst**

(synthetic data, charts, python code)

Director (Outbound)

**UX Analyst**

(diagrams, diagram template)

**Workflow Stages:**

1. User Input & Clarification: The Director (Inbound) agent will interact with the user to gather and clarify project requirements.
2. Straw Man Creation: The Director (Inbound) will generate a "straw man" or outline of the presentation for user approval.
3. Task Delegation: Once the straw man is approved, the Director (Inbound) will delegate tasks to specialized "worker" agents.
4. Content Generation & Design: The worker agents (Researcher, Data Analyst, Visual Designer, UX Architect, UX Analyst) will generate the content, visuals, and layout for each slide.
5. Assembly & Quality Assurance: The Director (Outbound) will assemble the outputs from the worker agents, ensure quality and consistency, and present the final slides to the user.
6. Iteration & Refinement: The user can provide feedback on the slides generated, which the Director (Inbound) will use to initiate further revisions.

**4. Roles and Responsibilities of Agents**

**4.1. User/Frontend**

* The user interacts with the system through a frontend interface.
* The frontend is responsible for rendering the output provided by the Director agents and capturing user input.
  1. **Director (Inbound):** The Director (Inbound) agent is the primary point of contact for the user and the orchestrator of the presentation creation process.

1. Engages the user with customized questions to clarify requirements such as presentation length, audience type, topic, number of slides, and desired balance between verbose and visual content.
2. Confirms its understanding of the requirements with the user before proceeding. This can be an auto-accept or require explicit user approval.
3. Before building the straw man, the director confirms with the user its understanding and what it is planning to do. It might be auto-accept or the user can stop to make some changes. Generally, that is the confirmation after asking questions the director will do, and then it starts the process of building the straw man.
4. **Builds the straw man including:**
   1. What is the theme that is going to be used across the presentation?
   2. The title of each slide
   3. Narrative and hook in each slide
   4. Description of each slide and high level of how it would be structured
   5. Potential Images / Analysis / Videos / Diagrams to be added to the slide.
   6. How it should be structured for the audience for impact
   7. What is this type of slide which includes:
      * Is it a visual heavy slide?
      * Is it an analysis heavy slide?
      * Is it a diagrammatic slide?
      * Is it a plain text slide?, etc.
      * If it is a plain text slide, how might be best positioned? Or would it be a major text with some image / analysis
      * Is it a sub-section / breaker slide, etc.
   8. the straw man is provided in HTML format embedded in JSON to be rendered by the front-end.
   9. Iteratively refines the straw man based on user feedback, with the ability to edit specific slides while retaining others.
   10. Once confirmed by user, this would also be passed on to further sub-worker agents.
5. Task Delegation:
   1. Provides clear and detailed instructions to each worker agent. This includes specifying user-provided inputs versus director-generated inputs, with user inputs being treated as sacrosanct.
   2. Can request work for slides one by one to maintain focus.
   3. Maintains an iteration number for tracking changes.
6. Director intake provides tasks clearly for each of the next set of agents to do in additional to the simple HTML. This includes, say,
   1. for example, the researcher: What content is needed per slide, what references, what the narrative is, what is the structure of the content needed (title, sub-topic, classification etc.). Approximate words. These could be more than the actual number of words needed. Could be 3 acts or so because the director outbound would take a very strategy call on the content needed based on the layout architect's inputs
   2. to the artist, it would provide content on image needed or video needed or GIF needed and necessary details to make it into the right context. It would also share the theme and what are the requirements of the image
   3. to the analyst, it would provide the context of what the analysis is needed, what the analysis would prove out in the context of the slide and how it will add value, etc
   4. to the layout architect, it would provide clear instructions on the initial thought process on the slide, what elements need to be there. While director may provide multiple options like an image or chart for specific slide, the layout architect has to be clear and layout architects makes a very strategic choice which will be used as a final HTML that we are using to build that specific page
   5. to the analyst, building diagrams would be clear instructions in terms of what kind of diagram it is (process diagram, hierarchy diagram, architecture diagram, etc.) and using that context on what needs to be built. If it is such a slide, the input would have also gone to the researcher that the content that the researcher is preparing would be sent to the diagrams analyst who would actually convert that into a diagram. So, specifics of that would also go to the researcher.
7. While internally interacting, the director has to make it explicit as to what is provided by the user as an input vs. what's an input from the director. A user input has to be sacrosanct. A director input could have some variations because the director is looking at the others to validate that. So, user input has to be validated by the director outbound to make sure that it meets all the quality criteria
8. Director inbounds also communicates with Director Outbound to ensure it is prepared when the outputs come to it from the worker agents
9. One Director (Outbound) shares output with user and any further interactions would again be with the Director (Inbound). This case the input might on specific slides or specific aspects of slides like a chart or such. In this case the Director calls specific agents with specific instructions to change.
10. The director can ask for input for slides one by one giving worker agents focus. Rather than trying to get all at once.
11. the iteration number might also need to be maintained.
12. Capabilities:
    1. Knowledge of best practice frameworks for Decks
    2. Search user-uploaded documents
    3. Themes to use. Type of slides to use
    4. Build straw man
    5. Ask questions and clarify
    6. Clearly provide instructions

**4.3 UX Architect:**

1. Starts on confirmation from the Director (Inbound)
2. initially sets up a consistent theme across slides, including title slides, content slides, and visual slides. These include main title: size, position, fonts, font colors, slide elements: themes, footers, etc. This is determined and stored. This will be the standard across all slides. This is built using director's and user's input, trying to match as close as possible following design principles. This one set is typically not changed unless asked by the director or user.
3. slide by slide: the slide structure + theme provided in Straw Man is converted into actual boxes placed in the layout, and the title, header, footer, etc. follow point #2. The content space is laid out in this phase following best practices on alignment, white spaces, structural balance, theme, etc. This could naturally differ between text heavy slides vs. text with supporting visual slides vs visual with supporting text slides vs title slides vs section header slides etc.
4. The slide is built as a HTML or YAML with placeholder text for title, subtitle, description, visual etc. This will be replaced by a Director Outbound when stitching together with the actual content and actual text and images and analysis and other things provided across.
5. Layout Architect sends slide-by-slide to Director(Outbound) to ensure that it can be easily assembled rather than wait for the entire content.
   1. **Researcher:**
6. The Researcher takes the input from the director + user to build content necessary for the appropriate slides. The content is built in the structure asked for by the director.
7. The Researcher can search internal documents and external data through search for building content.
8. The Researcher can summarize data and bring it into a structure that is necessary. The reference links to where the content is extracted from is kept and provided back to the director outbound so that reference links can also be published.
9. The content is shared with the director outbound slide by slide in the structure provided by the director inbound.
10. Capabilities:
    * + Internal document search
      + External search
      + Summarization
    1. **Data Analyst**: a very interesting profile and a differentiator with competitors.
11. Based on the slide where an analysis is asked for and its prominence in the slide bracket, that is, if the slide is only for showcasing the analysis vs. if it is supporting a text, larger text on the slide - basically the size and relevance of the charting space.; The analyst takes a call in what might be an analysis that can be shown to make an impact and the data needed for the same.
12. The analyst looks at existing data available, and if existing data can provide value, that is taken as priority #1 to build one of the potential analyses to support the slide.
13. If existing data is not available, analysts contemplate between external data and synthetic data that it can generate to build high-priority analysis for the slide.
14. The data is processed, transformed, and quality-checked. The chart or visualization is built. The chart for each slide is provided as an image with a link to the underlying data. An analysis of the chart including some text of insights along with the title of the chart is provided. We need to ensure the chart is the scales are shown, the x-axis, y-axis, or the various axes are shown, etc.
15. Tools:

* Ability to potentially run Python-based scripts, import a library like Skykit Learn or Matplotlib or Seaborn
* Ability to search external libraries for data
* Ability to create synthetic data using Python code, etc
* Knowledge of potential visualizations that can really make a difference in slides
  1. **Visual Designer:** builds an image, GIF, icon, vector, or video.

1. Builds a consistent theme to be used across all the images for a specific tech. This is a one-time activity, and this is a text content in terms of how what is the theme across all the different things it builds.
2. Image: Based on the t-shirt sizing of the image received along with the content that is necessary in the image. It builds the amount of details necessary in it to convey the story. It calls a diffusion model to build the image with the right context, content, theme, size, etc. to build the image
3. GIFs: same as above, with animation explained.
4. Icons: provided as vectors, small size, simpler models. Typically one color vectors.
5. Videos. Storyline is built and shared with a theme and context.
6. Output is provided slide by slide to the Director (Outbound).
   1. **UX analyst (Diagrams Guy):** The UX Analyst specializes in creating diagrams to visually represent complex concepts, which is a significant differentiator.
7. Selects the appropriate diagram type (e.g., process flow, hierarchy) based on the Director's instructions. What is the type of diagram that can best represent this construct? Is it a process flow or is it a hierarchical structure, etc.? Typically, this is provided by the director inbound.
8. Based on the type of diagram, an existing template or a framework is leveraged to build the appropriate structure.
9. Size of the content that can fill within the template is provided in terms of number of characters.
10. Content provided by the researcher is structured, simplified, and fit into the diagram to actually build the diagram. Therefore, the UX analyst is only able to work after the researcher has completed the research.
11. Color combination for the diagrams template would need to be aligned with the overall theme.
12. The UX analyst would need to have access to a collection of pre-built templates for easy processing.
    1. **Director (Outbound):**
13. **Slide Assembly:**

* Uses the templates from the UX Architect as the foundation for each slide.
* Summarizes and fits the content from the Researcher into the slide layout.
* Stitches together all elements of a slide, including text, visuals, and diagrams.

1. **User-Facing Output:**

* Can provide a first draft of the content to the user before all visual elements are ready, with visuals being lazy-loaded. Output produced and shared slide by slide
* Shares the completed slides with the user/frontend in HTML format packaged in a JSON.
* Provides a summary of the work accomplished at the end of the process.

1. **Quality Validation:**

* Ensures that all outgoing slides are consistent with the user's requirements and the Director (Inbound)'s instructions.
* Performs quality checks and can send content back to the Researcher for revisions if necessary (a feature that may be implemented later).

1. **Communication Protocols**

* Clear and consistent communication protocols are essential for the smooth operation of the multi-agent system and for a seamless user experience.
  1. **Director(I) and Director(O) to the User / Frontend**: Any communication with the user would be common protocol regardless of where it goes from Director (Inbound) or Director (Outbound). Need to have a simple and common messaging schema by which the Director(Inbound) and the Director (Outbound) communicate with the user. If it is a common protocol, it makes it easy for the front-end to display to the user. This should include:
  + data to be displayed in the slide section of the interface which includes HTML/YAML template with links to image, videos, charts, diagrams, other links, etc.
  + data to be displayed on the chat section with the user. This could be:
    - Questions to the user
    - Summary of actions performed
    - Processing updates when the system is working
    - Any actions that the system wants a user to perform (like "accept changes" or "acknowledge that it can proceed", etc.)
  1. **User / Frontend to Director (Inbound):**

This could be answers to questions or general inputs from user at any point of time, including text.

* + There could be images, pdfs, ppts, docs or other modes and data the user may upload, including voice. These must have a method for taking in and processing.
  + They could be action callouts like accepting changes also.
  + There could be specific changes within a slide that the user might want to change, including the HTML tags for the same so that a specific slide could be referenced. This should include a common protocol by which we may let communication from the user on the slide and the elements within the slide be transferred to it.
  1. **Director (Intake) to All Sub-Agents and Director (Outbound):**
  + Director (Intake) interacts with the user and orchestrates the process. It is important for a director to therefore provide clear instructions to other agents in the line.
  + It provides a base HTML, YAML, or JSON layout the structure, title, description, hook, narrative per slide, and the type of slide (like a visual heavy or analysis heavy or text heavy or diagrammatic) including images, gifs, icons, analyses (i.e. charts) required by each slide.
  + Further it could provide separate instruction for each of the sub-agents in the JSON. E.g., (i) The image/ analysis needed specifically for the slides and context, etc. This makes it easer for each agent to look at the exact instruction for it + use the rest of it as context. (ii) Diagram needed could include type of presentation (like process or hierarchy or etc.). Diagram is unique and starts only after content is prepared by the researcher.
  + There is a unique ID for every content being built by each, and this could include a slide number, item number, etc.
  1. **Worker Agents to Director (Outbound) and UX Analyst (Diagrams):**
  + this could include a combination of referencing to the item like slide number, item number, HTML tag, type of item, etc.
  + and the work output. This could be text, link to an image, link to a chart with the data, link to the data of the chart, etc.

1. **Non-Functional Requirements**

* **Performance:** The system should be responsive, with minimal delays in agent communication and content generation. The lazy loading of images is one strategy to enhance perceived performance.
* **Scalability:** The architecture should be able to handle multiple users and presentation generation requests concurrently.
* **Reliability:** The system should be robust and handle errors gracefully, with clear error messaging to the user when a request cannot be fulfilled.
* **Security:** User-uploaded documents and personal information must be handled securely and with privacy in mind.

**7. Success Metrics**

The success of the AI-powered presentation generator will be measured by:

* **User Adoption:** The number of active users and presentations created.
* **User Satisfaction:** Positive user feedback, high ratings, and low churn rate.
* **Task Completion Rate:** The percentage of users who successfully generate a complete presentation.
* **Time to Completion:** The average time it takes for a user to create a presentation from start to finish.
* **Quality of Output:** User ratings on the quality, relevance, and visual appeal of the generated presentations.