High level Architecture

System Architecture

Our system architecture is designed to facilitate complex information retrieval and analysis tasks. It's comprised of the following main components:

1. Query Interface: A user-friendly portal where users can input their queries. This can be web-based or integrated into existing company platforms.

2. Query Processor: A component that interprets user queries, identifies the query type, and directs it to the appropriate data sources.

3. Database Engine: Our core relational database that houses all the information. It's linked to various other data sources within the corporate ecosystem.

4. AI Analytics Engine: This engine takes processed queries and performs analytics by calling upon large language models. It's responsible for aggregating, summarizing, and interpreting data.

5. Dashboard and Reporting Tools: For users who want daily updates, this component generates dashboard modules based on real-time data otherwise it’s just a chat bot.

Data Models

Goals

- Attributes:
    - Summary
    - Latest Key Metrics Data, each with a context —linked to→ A detailed history of that metric with related description of events that has happened to it before —linked to→ Raw data
    - Summary of Latest Actions —linked to→ detail explanation of that part of the action that is related to that goal —linked to→ All the data for that action
    - Related People, each one their responsibility toward that goal —linked to→ Detail description of what that person did for the goal with history of the events and like to related entities —linked to→ Profile of the person
    - Sub-Goals—linked to→ How each one of the goals contributed to the goal—linked to→ the goal’s page

  - Parental goals—linked to→ The description of how this goal will contribute to that goal—linked to→ Each parental goal’s page

  - All the entities in a timely manner, each one with a descriptive title—linked to→ the part of that entity that is related to the goal with a description generated by AI—linked to→ The data of the entity

  - Risks —linked to→ Detail explanation

  - Dependencies —linked to→ Detail explanation

  - Progress —linked to→ Detail explanation

  - Prediction —linked to→ Detail explanation

Entities

- Attributes:

  - Type

  - Date Created

  - Summary—linked to→ the raw data

  - Creators—linked to→ the part of the data that has been created by him —linked to→ the user profile

  - Goals—linked to→ That part of the information that is related to that goal—linked to→ the goal’s page

People

- Attributes:

  - Name

  - Team—linked to→ what he did in the team and collaboration with people—linked to→ the team’s page

  - Goals —linked to→ what he did for that goal, what happened and what will happen —linked to→ the goal’s page

  - Entities in time order—linked to→ the part of that entity that he has created —linked to→ The entity’s page

  - Performance Metrics summary —linked to→ Each metrics explanation and history for the person —linked to→ the metric’s raw data

  - Summary of latest actions —linked to→ Detail explanation of actions

  - Responsibilities Summary —linked to→ Detail explanation

  - Leader —linked to→History of the relationship between the person and the leader—linked to→ the leader’s page

  - People who report to him —linked to→ The history of relationship —linked to→ each person’s profile

  - Summary of Dependencies —linked to→ Detail explanation

  - Achievements Summary —linked to→ Detail explanation

  - Timeline of goals in a summary —linked to→ Detail explanation

Teams

- Attributes:

  - Name

  - Positions —linked to→ Detail explanation

  - Summary of latest actions —linked to→ Detail explanation of actions

  - Responsibilities Summary —linked to→ Detail explanation

  - Performance Metrics summary —linked to→ Each metrics explanation and history for the person —linked to→ the metric’s raw data

  - People —linked to→ contribution of each person to the team—linked to→

  - Goals in order—linked to→ what they did for that goal, what happened and what will happen —linked to→ the goal’s page

  - Achievements —linked to→ Detail explanation

  - Timeline of goals in a summary —linked to→ Detail explanation

Each goal has a owner which is a person. Also each goal as an AI PM (artificial Intelligence product manager). Whenever new information comes in from assistants, Systems, Other PMs, or the owner, the PM will synthesize the info, find out what this info means, what actions it should take, and bring the actions to the owner. If the owner confirms, it does the actions.

Each goal also can have some subgoals. Goals (or subgoals with together) can be depnedant to each other. They can be parent/child. For example each goal is a parent for its subgoals.

Here are some events that can lead to the updates in a system designed in this architecture:

1. Updates to Key Metrics Data: As the project progresses, the latest key metrics data will be updated to reflect the current status of each goal. For example, if the goal is to create a personalized virtual health assistant, the key metrics data could include user engagement rates, retention rates, and user satisfaction scores.

   - Completion of user surveys and data analysis

   - Regular monitoring of user engagement and satisfaction

   - Implementation of tracking mechanisms for relevant metrics

2. Updates to Sub-Goal Progress: As each sub-goal is completed, the progress will be updated to reflect the current status. For example, if the sub-goal is to develop an MVP for the virtual health assistant with basic features and functionality, the progress could be updated to reflect the percentage of tasks completed and the estimated time to completion.

   - Milestone achievements and task completions

   - Regular project status updates and meetings

   - Feedback from team members and stakeholders on progress

3. Updates to Dependencies: As dependencies between sub-goals change, the system will need to be updated to reflect these changes. For example, if the backend team needs more time to develop a scalable backend architecture, this could impact the timeline for MVP development and feature selection.

   - Delays or changes in the timeline of dependent tasks

   - New requirements or constraints identified during development

   - Communication from teams regarding changes in their progress

4. Updates to Risks: As new risks emerge or existing risks change in severity, this information will need to be updated in the system. For example, if there is a new healthcare regulation that impacts data storage and retrieval mechanisms, this could pose a risk to the development of the virtual health assistant.

- Identification of new risks or potential issues
- Changes in regulations or compliance requirements
- Feedback from stakeholders or experts on potential risks

5. Updates to Related People: As team members change or new stakeholders become involved in the project, this information will need to be updated in the system. For example, if a new data science team manager is hired, this information will need to be updated in the system to ensure effective collaboration and communication.

- Hiring or onboarding of new team members or stakeholders
- Changes in team roles or responsibilities
- Communication of relevant contact information or updates

6. Updates to Entities: As new entities are created or existing entities change in functionality, this information will need to be updated in the system. For example, if a new tool is introduced for user feedback collection and analysis, this entity will need to be added to the system and integrated with other relevant entities.

- Introduction of new tools, software, or technologies
- Changes in functionality or integration requirements of existing entities
- Communication or documentation regarding the addition or modification of entities

These events serve as triggers that prompt the need for updates within the system, ensuring that the information remains accurate and up-to-date for effective project management and goal achievement.