Service 1: Data Update

- Name: DataUpdate
- API Endpoint: /api/v1/updatedata
- HTTP Command: POST
- Payload Schema:

  ```
  {
      "filename": "string",
  }
  ```

- Request-Response Codes:
  - 200 OK Update successful
  - 404 Bad Request: Server Unavailable

Function:

The `updatedata` function is designed to update the model embeddings of documents. This function takes new or modified documents and processes them to update their corresponding embeddings in the model. Embeddings are numerical representations of the documents that the model uses to understand and work with the text data. By updating these embeddings, the function ensures that the model remains accurate and up-to-date with the latest document information.

Service 2: Vectorization

- Name: Vectorization
- API Endpoint: /api/v1/vectorization
- HTTP Command: POST
- Payload Schema:

  ```
  {
      "filename": "string",
      "mothod": "string",
      "timestamp": "datatime"
  }
  ```

- Request-Response Codes:
  - 200 OK execute successful
  - 400 Bad Request: Timestamp behand

Function:

The `Vectorization` function is designed to convert text documents into vector representations. This function processes the text and transforms it into numerical vectors that can be used by machine learning models. These vector representations, also known as embeddings, capture the semantic meaning of the text, enabling the model to perform various tasks such as classification, clustering, and similarity analysis. By vectorizing the text, the function allows the model to effectively analyze and work with the textual data.

Service 3: Model Embedding

- Name: Model embedding
- API Endpoint: /api/v1/embedding
- HTTP Command: POST
- Payload Schema:

  ```
  {
      "filename": "string",
      "mothod": "string",
      "timestamp": "datatime"
  }
  ```

- Request-Response Codes:
  - 200 OK Model embedding successful
  - 400 Model embedding unsuccessfu

Function:

The `embedding` function is designed to perform model embedding on text data. This function processes the input text and generates embeddings, which are dense numerical representations of the text. These embeddings capture the semantic meaning and context of the text, allowing the model to understand and work with the data more effectively. The `embedding` function is crucial for tasks such as natural language processing, text classification, and similarity analysis, as it transforms textual information into a format that machine learning models can utilize.

Service 4: Summary Note

- Name: Summary Note
- API Endpoint: /api/v1/summary
- HTTP Command: POST
- Payload Schema:

```
{
    ["ID": "string",
    "username": "string",
    "timestamp": "datatime",
    "content": "string"],
    ["ID": "string",
    "username": "string",
    "timestamp": "datatime",
    "content": "string"]
}
```

- Request-Response Codes:
  - 200 OK Model summary note successful
  - 404 Server Unavailable

Function:

The `summary` function is designed to generate summaries of meeting content using a language model. This function takes the transcript or notes from a meeting and processes it to produce a concise and coherent summary. By distilling the key points and essential information, the `summary` function helps users quickly understand the main outcomes and important details of the meeting. This is particularly useful for keeping records, sharing updates, and ensuring that all team members are informed without needing to review the entire meeting content.

Service 5: Jira Update

- Name: Jira Update
- API Endpoint: /api/v1/jira
- HTTP Command: POST
- Payload Schema:

```
{
    "boardID": "string",
    "issueid": "string",
    "timestamp": "datatime",
    "assignto": "sting",
    "descripton": "string"
}
```

- Request-Response Codes:
  - 200 OK Jira update successful
  - 400 Update unsuccessful
  - 404 Server Unavailable

Function:

The `jira` function is designed to update Jira based on user requests. This function processes user inputs and performs the necessary actions to create, update, or manage Jira issues. By automating interactions with Jira, the `jira` function helps streamline project management tasks, ensuring that issue tracking and task management are efficiently maintained according to user specifications. This function is essential for maintaining accurate and up-to-date project records within the Jira platform.

Service 6: Slack Event Listener

- Name: Slack Event Listener
- API Endpoint: /api/v1/event
- HTTP Command: POST
- Payload Schema:

  ```
  {
      "eventtype": "string",
      "text": "string",
      "timestamp": "datatime",
      "channel": "sting",
      "user": "string"
  }
  ```

- Request-Response Codes:
  - 200 OK Message received
  - 404 Server Unavailable

Function:

The `event` function is designed to listen for Slack events. This function monitors and processes events from a Slack workspace, such as messages, reactions, user activities, and other interactions. By handling these events, the `event` function enables the system to respond to user actions, automate workflows, and integrate Slack activities with other applications or services. This function is crucial for maintaining real-time communication and interaction within a Slack environment.