**What happens in this?**

Take scheduled meeting data from outlook using scheduled tasks / service by running it early in the morning.Save the scheduled data of meetings pulled from the outlook calendar in a meeting database. Then the meeting will be conducted in the meeting application like zoom or webex or teams. After the meeting, the closed captions/transcripts will be collected and also the participants' attendance will be collected and also the meeting recording is collected.

Then comes the summarizing process. This uses the meeting data that is meeting transcripts to summarize the meeting. In some cases meeting transcripts won't be available. Then the tool will use the meeting recording. It collects the audio and then changes it into text using speech to text tools like Whisper by OpenAI Then this text will be used to summarize the meet. This summarizing will be done at the end of the day for all the meetings that happened during that day using some scheduled task / service again.

Sometimes there can be meetings that happen instantly and the meeting schedule may not be found on the calendar. For these meetings, using API of these meeting applications are used to find out what extra meetings happened on that day and they are verified and then those meetings data will be collected and stored in the same way explained above.

Now comes the dashboard part. All these meetings data Collected and Analyzed are shown to the Admin/Organizers (both are basically the same) and participants differently. Admins/Organizers can see all the meetings data no matter if that organizer has organized the meeting or not. But, as far as the participants are concerned, they can only see the data of those meetings in which they were a participant of.

It's like the participants can see their meetings activity. If they click on some of the meetings, they can see the invitees, attendance, agenda of meet, summary and some additional obvious information like time and platform of that meeting. On the other hand, the admins/organizers/Top level executives can see all the meetings data. If they click on a particular meeting, they can see some metrics of the participants. Along with this, they can see the overall meetings dashboard that has analytics info from all the meetings.

There is one more process called survey mechanism that sends emails to both organizers and participants. It is like taking the feedback from them and save that data in meeting database and this will also be shown to the organizers and participants with the other stuff. There will be pre meeting survey which will be sent to the organizer in which he can fill out information like Agenda of the meet, provide supporting documentations for the participants as well. After the meeting is concluded, a post survey will be sent to the participants in which they can provide feedbacks such as was the agenda of the meeting met, if they have any questions / queries etc etc.

Entities:

1. User

- UserID (PK)

- FirstName

- LastName

- Email

- Password

- Role (FK to Role)

- CreatedAt

- UpdatedAt

2. Role

- RoleID (PK)

- RoleName (e.g., Admin, Organizer, Participant)

3. Meeting

- MeetingID (PK)

- Title

- Description

- StartDateTime

- EndDateTime

- MeetingType (e.g., Scheduled, Instant)

- MeetingPlatform (e.g., Zoom, WebEx, Teams)

- MeetingURL

- RecordingURL

- Agenda

- CreatedAt

- UpdatedAt

4. MeetingAttendee

- MeetingAttendeeID (PK)

- MeetingID (FK to Meeting)

- UserID (FK to User)

- AttendanceStatus (e.g., Attended, Absent)

- JoinTime

- LeaveTime

5. MeetingTranscript

- TranscriptID (PK)

- MeetingID (FK to Meeting)

- TranscriptText

- Language

- CreatedAt

6. MeetingSummary

- SummaryID (PK)

- MeetingID (FK to Meeting)

- SummaryText

- CreatedAt

7. MeetingAnalytics

- AnalyticsID (PK)

- MeetingID (FK to Meeting)

- TotalAttendees

- AvgEngagementScore

- AvgAttendanceDuration

8. Survey

- SurveyID (PK)

- SurveyType (e.g., Pre-meeting, Post-meeting)

- Title

- Description

- CreatedAt

- UpdatedAt

9. SurveyQuestion

- QuestionID (PK)

- SurveyID (FK to Survey)

- QuestionText

- QuestionType (e.g., Multiple Choice, Open-ended)

10. SurveyResponse

- ResponseID (PK)

- SurveyID (FK to Survey)

- QuestionID (FK to SurveyQuestion)

- UserID (FK to User)

- ResponseText

- CreatedAt

11. MeetingDocument

- DocumentID (PK)

- MeetingID (FK to Meeting)

- DocumentName

- DocumentURL

- CreatedAt

Relationships:

1. User (1) - (M) MeetingAttendee

2. User (1) - (M) SurveyResponse

4. Role (1) - (M) User

5. Meeting (1) - (M) MeetingAttendee

6. Meeting (1) - (1) MeetingTranscript

7. Meeting (1) - (1) MeetingSummary

8. Meeting (1) - (1) MeetingAnalytics

9. Meeting (1) - (M) MeetingDocument

10. Survey (1) - (M) SurveyQuestion

11. Survey (1) - (M) SurveyResponse

12. SurveyQuestion (1) - (M) SurveyResponse

CHECK IT OUT BELOW

—----------------------------------------------------------------------------------------------------------------------------

Entities:

1. User

- UserID (PK)

- FirstName

- LastName

- Email

- Password

- Role (FK to Role)

- CreatedAt

- UpdatedAt

- LastLoginTimestamp (added here)

2. Role

- RoleID (PK)

- RoleName (e.g., Admin, Organizer, Participant)

3. Meeting

- MeetingID (PK)

- Title

- Description

- StartDateTime

- EndDateTime

- MeetingType (e.g., Scheduled, Instant)

- MeetingPlatform (e.g., Zoom, WebEx, Teams)

- MeetingURL

- RecordingURL

- Agenda

- CreatedAt

- UpdatedAt

4. MeetingAttendee

- MeetingAttendeeID (PK)

- MeetingID (FK to Meeting)

- UserID (FK to User)

- AttendanceStatus (e.g., Attended, Absent)

- JoinTime

- LeaveTime

5. MeetingTranscript

- TranscriptID (PK)

- MeetingID (FK to Meeting)

- TranscriptText

- Language

- CreatedAt

6. MeetingSummary

- SummaryID (PK)

- MeetingID (FK to Meeting)

- SummaryText

- CreatedAt

7. MeetingAnalytics

- AnalyticsID (PK)

- MeetingID (FK to Meeting)

- TotalAttendees

- AvgEngagementScore

- AvgAttendanceDuration

8. Survey

- SurveyID (PK)

- SurveyType (e.g., Pre-meeting, Post-meeting)

- Title

- Description

- CreatedAt

- UpdatedAt

9. SurveyQuestion

- QuestionID (PK)

- SurveyID (FK to Survey)

- QuestionText

- QuestionType (e.g., Multiple Choice, Open-ended)

10. SurveyResponse

- ResponseID (PK)

- SurveyID (FK to Survey)

- QuestionID (FK to SurveyQuestion)

- UserID (FK to User)

- ResponseText

- SelectedOption (if applicable)

- CreatedAt

11. MeetingDocument

- DocumentID (PK)

- MeetingID (FK to Meeting)

- DocumentName

- DocumentURL

- CreatedAt

Relationships:

1. User\_Attends\_Meeting (User (1) - (M) MeetingAttendee)

- A User can attend multiple Meetings

- A Meeting can have multiple Users attending

2. User\_Responds\_To\_Survey (User (1) - (M) SurveyResponse)

- A User can provide multiple Survey Responses

- A Survey Response belongs to a single User

3. Role\_Assigned\_To\_User (Role (1) - (M) User)

- A Role can be assigned to multiple Users

- A User is assigned a single Role

4. Meeting\_Has\_Attendees (Meeting (1) - (M) MeetingAttendee)

- A Meeting can have multiple Attendees

- An Attendee belongs to a single Meeting

5. Meeting\_Has\_Transcript (Meeting (1) - (1) MeetingTranscript)

- A Meeting has one Transcript

- A Transcript belongs to a single Meeting

6. Meeting\_Has\_Summary (Meeting (1) - (1) MeetingSummary)

- A Meeting has one Summary

- A Summary belongs to a single Meeting

7. Meeting\_Has\_Analytics (Meeting (1) - (1) MeetingAnalytics)

- A Meeting has one set of Analytics

- Analytics belong to a single Meeting

8. Meeting\_Has\_Documents (Meeting (1) - (M) MeetingDocument)

- A Meeting can have multiple Documents

- A Document belongs to a single Meeting

9. Survey\_Has\_Questions (Survey (1) - (M) SurveyQuestion)

- A Survey can have multiple Questions

- A Question belongs to a single Survey

10. Survey\_Has\_Responses (Survey (1) - (M) SurveyResponse)

- A Survey can have multiple Responses

- A Response belongs to a single Survey

11. Question\_Has\_Responses (SurveyQuestion (1) - (M) SurveyResponse)

- A Survey Question can have multiple Responses

- A Response belongs to a single Survey Question

The cardinality is represented using the (1) and (M) notation, where (1) denotes a one-to-one or one-to-many relationship, and (M) denotes a many-to-one or many-to-many relationship.