P2 Documentation

Group 8720: Ocean Chen, He Zhao, Jennifer Xi, Do-Oh Moon

Project Structure

Account/Contacts App

User Registration:

- Upon registration, users have the option to specify their availabilities, which will be applicable to all events and calendars.
- Each availability comes with a rank based on the user's preference.

Contacts Management:

- Each user maintains a list of contacts, allowing them to easily interact with other users within the system.

Scheduling App

Event and Meeting Calendar:

- Users have the capability to create both event and meeting calendars, enabling them to organize their schedules effectively.
- An event is associated with a meeting calendar, and vice versa.
- Each calendar contains a collection of meetings, facilitating the organization of various events and appointments.
- Each meeting is characterized by an organizer and one or more participants, facilitating seamless collaboration and communication.

Calendar Management:

- Each event/meeting calendar includes a start and end date, providing a clear timeframe for scheduling activities.

Meeting Management:

- Meetings are defined by their start and end times, as well as the date on which they occur.

Suggested Meeting Calendars:

- Organizers have the ability to generate suggested meeting calendars, which aim to find optimal meeting times based on participants' availabilities.
- The suggested meeting calendar employs bipartite matching algorithms to determine feasible meeting slots that accommodate the availability constraints of all participants.
- In cases where no suggested meeting calendar meets all availability constraints, the system returns maximal matchings and lists users who couldn't be matched separately.

Meeting Finalization:

- Organizers can review and finalize meetings based on the suggested calendars generated by the system.
- Once a calendar is finalized, unique meeting IDs are assigned to each meeting, along with corresponding start and end times.
- Meeting links are then sent to participants and the organizer, facilitating seamless coordination and attendance management.

Accounts app

Views

- **register**
  - Description: For user registration.
  - Method: POST
  - Payload: User registration data including username, email, and password.
- **login**
  - Description: For user authentication.
  - Method: POST
  - Payload: User credentials including username and password.
- **logout**
  - Description: For logging out users.
  - Method: POST
- **profile**
  - Description: For displaying and editing user profile information.
  - Method: GET (to retrieve user profile), POST (to update user profile)
  - Payload: User profile data including username, email, first name, and last name.
- **change_password**
  - Description: For users to change their password.
  - Method: POST
  - Payload: Old password and new password.

Contacts app

Models

- **UserContact**
  - Description: Represents a user's contacts.
  - Fields:
    - user: ForeignKey to the User who added the contact.
    - contact: ForeignKey to the User who is the contact.
    - added_on: DateTimeField indicating when the contact was added.
  - Constraints:
    - Users cannot have themselves as contacts.
- **Availability**
  - Description: Indicates the availability of a user for meetings.
  - Fields:
    - user: ForeignKey to the User whose availability is being represented.
    - start_time: TimeField indicating the start time of availability.
    - end_time: TimeField indicating the end time of availability.
    - day: DateField representing the day of availability.
    - rank: IntegerField representing the preference rank of the availability.
    - available: BooleanField indicating whether the user is available during this time.

Views

- **ListContactsView**

  - Description: Displays a user's list of contacts, allowing them to see who they have added and potentially their availability status for quick scheduling reference.
  - Method: GET
- **AddContactView**
  - Description: Enables a user to add a new contact by providing the contact's email address.
  - Method: POST
  - Payload: Contact's email address.
- **RemoveContactView**
  - Description: Allows users to remove a contact from their list, potentially with confirmation to prevent accidental deletions.
  - Method: DELETE
- **ContactDetailView**
  - Description: Offers detailed information about a specific contact, including their availability if shared, and any upcoming meetings scheduled with them.
  - Method: GET
- **AddContactAvailability**
  - Description: Users should be able to update their availability preferences through an interactive week calendar.
  - Method: POST
  - Payload: Availability preferences.
- **UpdateContactAvailability**
  - Description: Update a contact's availability preferences.
  - Method: PUT
  - Payload: Updated availability preferences.
- **UpdateContactPreferencesView**
  - Description: For contacts that are registered users, this view would allow them to update their preferences for meeting times directly in the app, rather than through email interaction.
  - Method: PUT
  - Payload: Updated preferences.
- **ViewContactInvitationsView**
  - Description: Enables users to see pending meeting invitations from other users, allowing them to accept, decline, or suggest alternative times based on their availability.
  - Method: GET
- **SendReminderView**
  - Description: A view for sending reminders to contacts who haven't responded to a meeting invitation, ensuring that users can follow up and finalize schedules efficiently.
  - Method: POST

Scheduling app

Models

- **Calendar**
  - Description: Represents a calendar for scheduling meetings.
  - Fields:
    - name: CharField representing the name of the calendar.

        - owner: ForeignKey to the User who owns the calendar.
        - participants: ManyToManyField to User representing the participants of the calendar.
        - events: ManyToManyField to Meetings representing the events/meetings associated with the calendar.
- **CalendarInvite**
    - Description: Represents an invitation to join a calendar.
    - Fields:
        - calendar: ForeignKey to the Calendar being invited to.
        - invitee: ForeignKey to the User being invited.
        - inviter: ForeignKey to the User who sent the invitation.
        - sent_at: DateTimeField indicating when the invitation was sent.
        - status: CharField representing the status of the invitation.
- **Meetings**
    - Description: Represents a meeting.
    - Fields:
        - name: CharField representing the name of the meeting.
        - start: DateTimeField representing the start time of the meeting.
        - end: DateTimeField representing the end time of the meeting.
        - owner: ForeignKey to the User who owns the meeting.
        - participants: ForeignKey to the User who is participating in the meeting.
        - calendar: ForeignKey to the Calendar associated with the meeting.
- **Suggested Schedule**
    - Description: Represents a suggested schedule for organizing meetings based on participants' availabilities.
    - Fields:
        - calendar: ForeignKey to the Calendar for which the suggested schedule is generated.
        - unmatched_users: ManyToManyField to User representing users who couldn't be matched in the suggested schedule.
        - proposed_meetings: ManyToManyField to SuggestedMeeting representing the proposed meetings included in the suggested schedule.
- **Suggested Meeting**
    - Description: Represents a suggested meeting within a suggested schedule.
    - Fields:
        - start: DateTimeField indicating the proposed start time of the meeting.
        - end: DateTimeField indicating the proposed end time of the meeting.
        - organizer: ForeignKey to the User who organized the suggested meeting.
        - participant: ForeignKey to the User who is invited to participate in the suggested meeting.

Views

Calendar Views

- **CalendarCreateView**
    - Description: Create a new calendar.
    - Method: POST

    - Permissions: User must be authenticated.
    - Payload:
        - name: String, max length 100, representing the name of the calendar.
- **CalendarListView**
    - Description: Retrieve a list of calendars owned by the current user.
    - Method: GET
    - Permissions: User must be authenticated.
- **CalendarEditView**
    - Description: Update an existing calendar.
    - Method: PUT
    - Permissions: User must be authenticated.
    - Payload:
        - name: String, max length 100, representing the new name of the calendar.
- **CalendarDeleteView**
    - Description: Delete an existing calendar.
    - Method: DELETE
    - Permissions: User must be authenticated.
- **InviteContactToCalendarView**
    - Description: Invite a contact to a calendar.
    - Method: POST
    - Permissions: User must be authenticated.
    - Payload:
        - calendar_id: ID of the calendar to invite the contact to.
        - invitee_id: ID of the contact to be invited.
- **AcceptCalendarInviteView**
    - Description: Accept a calendar invitation.
    - Method: POST
    - Permissions: User must be authenticated.
    - URL Parameter:
        - invitation_id: ID of the invitation to accept.

Meetings Views

- **MeetingCreateView**
    - Description: Create a new meeting.
    - Method: POST
    - Permissions: User must be authenticated.
    - Payload:
        - name: String, max length 100, representing the name of the meeting.
        - start: Datetime, indicating the start time of the meeting.
        - end: Datetime, indicating the end time of the meeting.
        - participants: List of user IDs, representing the participants of the meeting.
        - calendar: ID of the calendar associated with the meeting.
- **MeetingDetailView**
    - Description: Retrieve details of a specific meeting.
    - Method: GET
    - Permissions: User must be authenticated.
    - URL Parameter:
        - meeting_id: ID of the meeting to retrieve details.

- **MeetingEditView**
    - Description: Update an existing meeting.
    - Method: PUT
    - Permissions: User must be authenticated.
    - URL Parameter:
        - pk: ID of the meeting to be updated.
    - Payload:
        - name: String, max length 100, representing the new name of the meeting.
        - start: Datetime, indicating the new start time of the meeting.
        - end: Datetime, indicating the new end time of the meeting.
        - participants: List of user IDs, representing the new participants of the meeting.
        - calendar: ID of the new calendar associated with the meeting.
- **MeetingDeleteView**
    - Description: Delete an existing meeting.
    - Method: DELETE
    - Permissions: User must be authenticated.
    - URL Parameter:
        - pk: ID of the meeting to be deleted.

Suggest Schedule Views

- **GenerateSuggestedSchedulesView**
    - Description: Generate suggested schedules for organizing meetings based on participants' availabilities.
    - Method: GET
    - Permissions: User must be authenticated.
    - URL Parameter:
        - calendar_id: ID of the calendar for which suggested schedules are generated.
- **Helpers:**
    - **Find_maximal_matchings**
        - Description: For finalizing a meeting time and converting it into an event.
        - Description: Find up to 5 sets of maximal matchings for meetings based on weekly recurring availabilities.
        - Parameters:
            - calendar_id: ID of the calendar for which to generate matchings.
            - desired_meeting_duration: Desired duration of each meeting, in minutes.
    - **check_intersection**
        - Description: Check if there's an intersection between organizer's availabilities and a time slot.
        - Parameters:
            - organizer_availabilities: Organizer's weekly availabilities.
            - proposed_start: Proposed start time of the meeting.
            - proposed_end: Proposed end time of the meeting.
            - blocks_needed: Number of blocks needed for the meeting.