

User-Service

- The User Service creates new users. After creating a user, it sends a message with the user's ID, email, and notification settings to a Kafka topic called "user-topic."
- When a user updates their email or notification preferences, it also sends an update message to the same "user-topic."
- The service helps users recover their accounts by generating a one-time password (OTP) and sending an event about this to another Kafka topic called "password-reset-topic."
- It also managed the user session to ensure that user can login in one device at a time

Task-Service

- After creating a task, Subtasks, bugs, it generates an event and sends it to the **calendar-topic** to sync the task deadlines within the Task Service.
- When task details change—such as status updates or other important modifications—it creates new events with the Event-Type (ASSIGNED, UNASSIGNED, UPDATED, STATUS_CHANGED, PRIORITY_CHANGED) and sends them to both the **task-topic** and the **calendar-topic** to sync the latest details and trigger notifications to users.
- Additionally, when a user is assigned or unassigned to a task, the Task Service sends an event to the **task-topic** to notify other services of this change.

Notification-Service

- The Notification Service consumes different events from various topics such as "user-topic", "task-topic", and "password-reset-topic".
- When it receives an event from the **user-topic**, it updates the user details if the user already exists; otherwise, it creates a new user with the given details.
- When a message arrives on the **password-reset-topic**, it sends an OTP to the user's email.
- When a message arrives on the **task-topic**, it routes the event to the appropriate method based on the event type (Created, Updated, Completed, Assigned, or Unassigned) and sends either an in-app or email notification based on the user's preference.

Project-Service

- The Project Service is responsible for creating projects.
- It also updates project details and manages important client information.

Activity-Service (Calendar Service)

- The Calendar Service listens to the "calendar-topic"; when messages arrive, it either creates new Calendar Data or updates the details of existing tasks.
- It also manages GitHub commits by linking the commits or branches to specific tasks or modules.

GIT-HUB Configuration

The user enters your service's webhook URL into their GitHub repository's webhook settings to enable event notifications. When a GitHub webhook event with commit details is received, your service extracts the commit message and branch name from the payload. It then parses the commit message or branch name to find task or subtask IDs (e.g., TASK-123). If an ID is found, the service looks up the corresponding task or subtask in the database and links the commit to it. If no ID is found, the commit is linked only to the project or left unlinked. Finally, the commit information and its associations are saved for use in notifications, reports, or status updates.