

This repository Search

Pull requests Issues Gist



Branch: master ▾ Azra_MeetingBot / Milestone 1 / DESIGN.md

Find file Copy path

gverma Updated graphic links

422581d 3 days ago

1 contributor

SE_Project_Milestone_1

Problem Statement

Meetings are a frequently used method for gathering different perspectives and discussing possible solutions. In the domain of engineering, teamwork is an elegant way to tackle the challenges by visualizing the problem through different perspectives and developing an effective solution. This involves scheduling and organizing team meetings which incur monetary expenses in some cases, and concur of the team members about the location, start time, and the duration of the meeting. Though it seems a trivial task, it may cost resources whose overhead exceeds the merit of meeting itself. For example, in a team of 5 people, only one person may disagree at the start time of the meeting. But having a significant and equal role in the project, the members have to reschedule the meeting. Now, supposing everyone is busy with their work, they take more time searching for another available time and location. When they find one, two people are not comfortable with the location! The process continues and in the interim they waste their time and needlessly procrastinate the meeting, which in turn hinders the product development. Even when they are able to concert, it may not be on the optimal terms which further decreases their productivity and efficiency of the meeting.

In engineering, agile methodologies often require frequent meetings: daily scrums, sprint planning, backlog grooming, sprint close out, demos, as well as ad-hoc planning, designing, and training meetings.

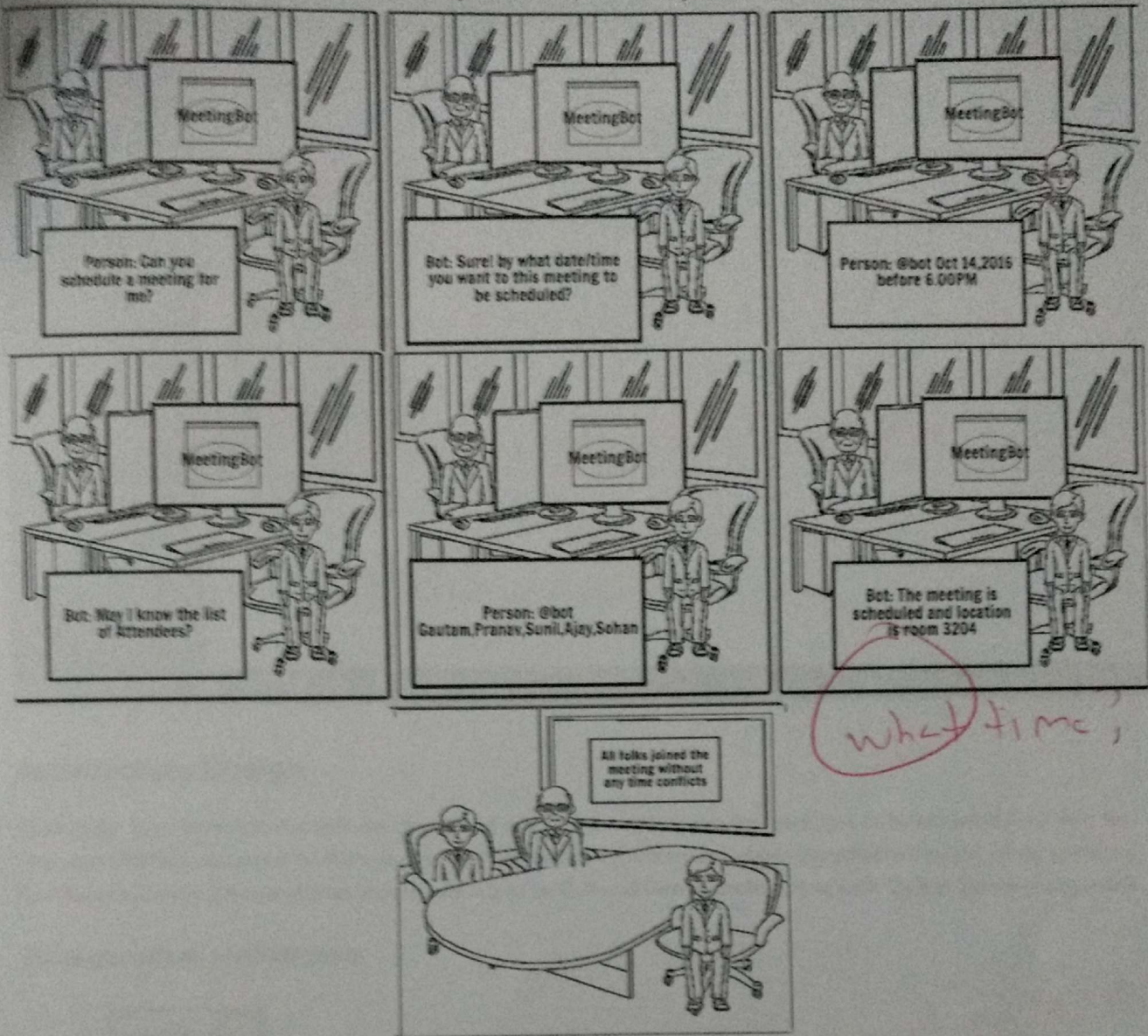
Bot Description

The meeting bot is a Slack based bot that can simplify burdensome administrative tasks such as scheduling and minimizing the cost of resources required for team meeting(s). The meeting bot is a good alternative to manually scheduling a meeting because it entails the consensus from the entire team on date, time, and location which might become a menial task. The meeting bot accesses the schedule of all team members through Google calendar and selects an optimal meeting time and location that is also comfortable for all the members. The users can interact with the bot using slack from application, or browser. The main user (team lead) would need to provide the bot with the member names, the date, and the expected duration of the meeting. Once the meeting bot finds a suitable time and location, it ask for confirmation from the organizer. If the organizer responses positively or does not response for 10 minutes, the bot would go ahead to schedule the meeting and notifies the team members. The meeting bot also responses to the events when any user tries to edit his schedule when a meeting is supposed to happen. The meeting bot asks the main user how to proceed further in such a case.

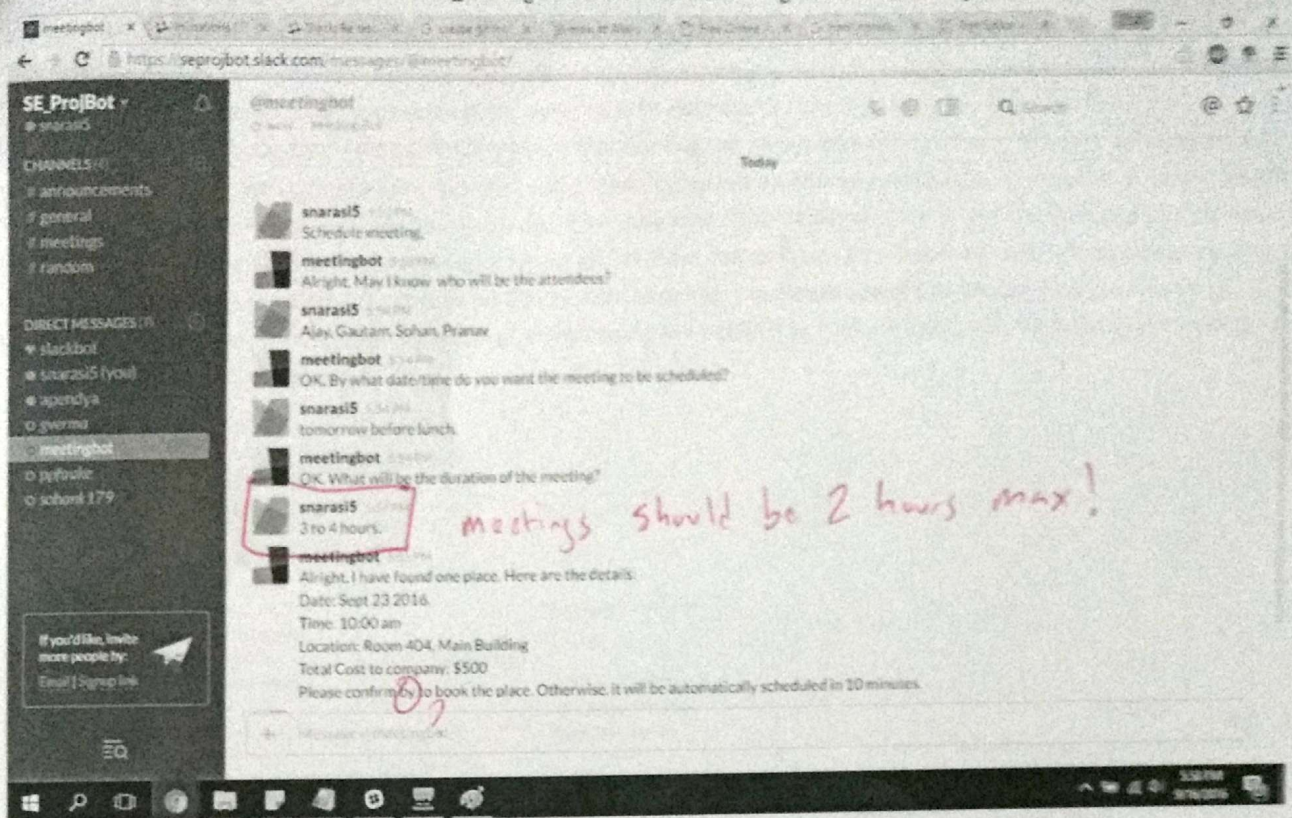
Design Sketches

Storyboard

missing description / motivation of bad meeting time.



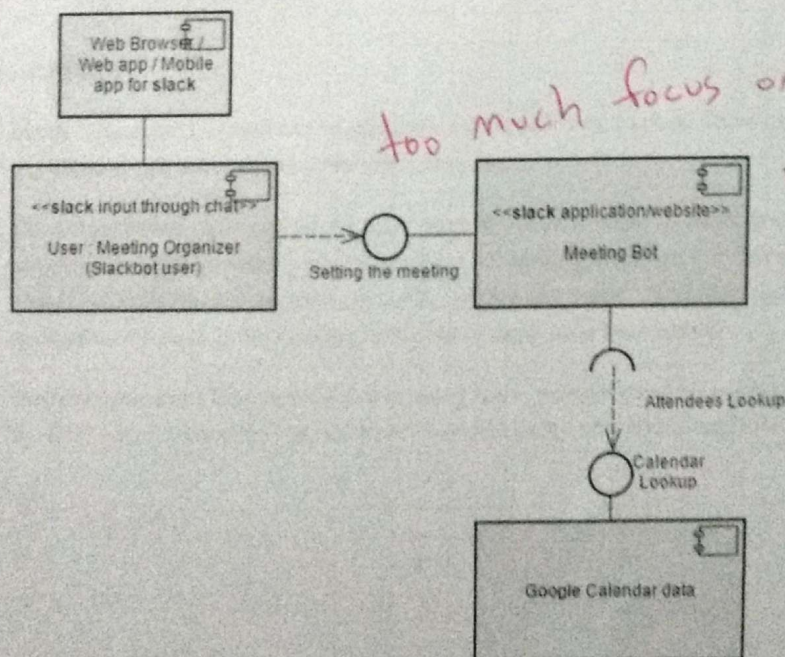
Wireframe



Architecture Design

Meeting bot must be connected with the calendar of every person with whom the meeting is to be scheduled and with the main user (Meeting organizer) for the purpose of getting details of meeting and delivering suitable meeting timing solutions. For this functionality, the best-suited architecture would be **Call and Return architecture with Object Oriented approach**.

Components in architecture



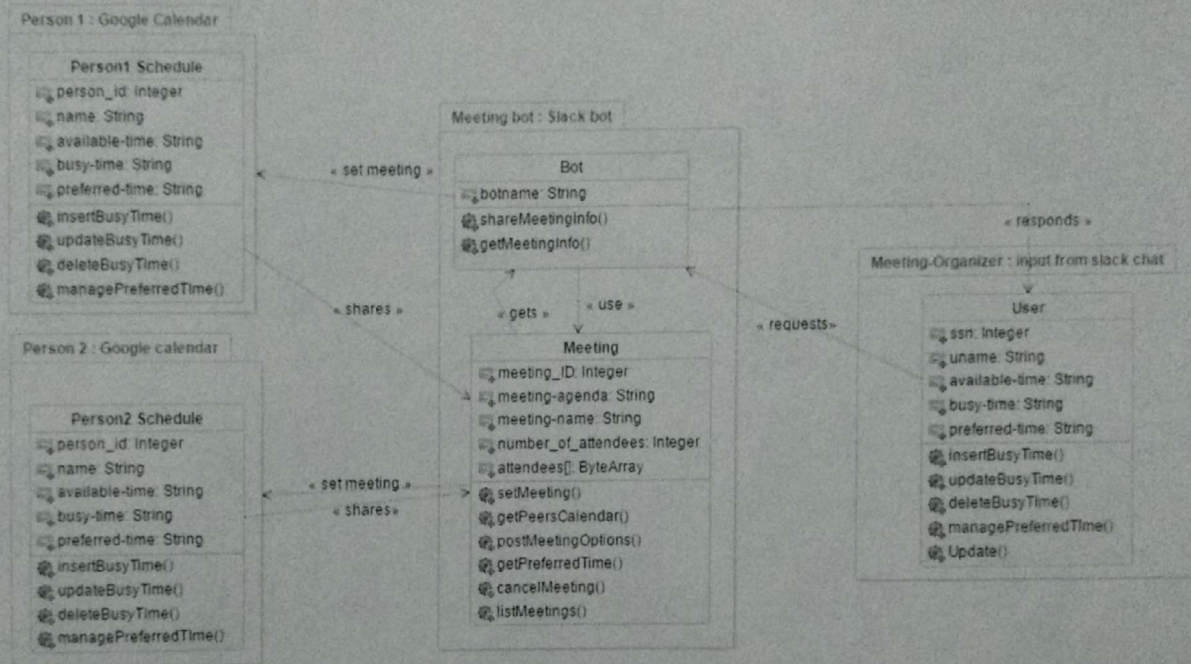
too much focus on formal notation without telling or thinking about design

This project involves the usage of following components:

- **Slack** - users will chat through slack;

- **Meeting Bot and Google Calendar** - the bot will get information from the calendars of team members. The main user will ask the bot to schedule a meeting which would include the name (id) of attendees, and the data/time by which the meeting is to be organized. The meeting bot will chat with the main user for any further assistance related to organizing/scheduling the meeting. It will look-up the calendars of all the attendees. These calendars will basically be the Google calendars synchronized with slack. The meeting bot will search for available (idle) time slots of attendees and would find a time when all of them are available. It will also take into consideration any specific instruction by the main user, try to find an optimistic time and location for all the team members, and minimize the cost of organizing meeting to the company. The meeting bot will then notify the main user with the details about the meeting and ask him for an approval to set-up the meeting. So, the platform for using the meeting bot will be slack, and the bot will use third-party services from google calendar.

Class Diagram



Constraints

1. **Slack** : Slack will be used as the platform to chat with meeting bot. There will be a channel for specific slack team and the meeting bot will manage scheduling meeting.
2. **Main User** : Main User can be the team-lead or team-manager or meeting-coordinator of the particular team. Main user will interact with the meeting bot on slack for scheduling meetings with team members by saying something like "schedule meeting with person1, person2, ... within one week ". The main user will get reply from bot with the meeting details like the date, time, location, and cost of organizing the meeting.
3. **Google Calendar** : Google calendar of every team member must be accessible and need to be synchronized with the meeting bot. The meeting bot can look for empty slots in their schedule and set-up the meeting accordingly.

Actions

- motivate ideal time in problem statement
- Architecture: too little high level info, too much low level info
- Patterns?