

## Assignment - V2

Create and Send Appointment Reminders to the recipients

1. Integrate with google calendar to get notified on calendar events for a given user and create appointment reminders. Use below mentioned links for the reference to google calendar API

<https://developers.google.com/calendar/api/concepts/events-calendars>

<https://developers.google.com/calendar/api/guides/push>

OR

Integrate with Calendly API to get notified on calendar events for a given user and create appointment reminders. Use below mentioned links for the reference to Calendly API

<https://developer.calendly.com/api-docs/d7755e2f9e5fe-calendly-api>

<https://developer.calendly.com/api-docs/c1ddc06ce1f1b-create-webhook-subscription>

2. Create a Webhook to get notified on Calendar events from either google calendar or Calendly and create appointment reminders for the user whose calendar event has been received. If a cancel event is received then delete the corresponding reminder.
3. As per the appointment reminders created for an user, schedule an event (15 min prior to the event) to send message via SMS to the recipient. You can have a csv file or database to map the user with his phone number. Refer to below example

User Email	User Phone	User Country Code
abc@foo.com	9800112345	91
xyz@foo.com	4442124678	1

**Note (Optional):** It would be great to trigger real messages using SMSMagic APIs, which would allow you to send SMS messages.

**API DOC :** [https://api.sms-magic.com/doc/#api-Send\\_SMS-Send\\_SMS\\_\\_\\_post](https://api.sms-magic.com/doc/#api-Send_SMS-Send_SMS___post)

**Guidelines:**

- Any other assumptions should be clearly stated
- You can pick any language of your choice to complete this assignment and follow coding standards as applicable to the language
- You are expected to follow SOLID principles of OOD
- You could simply zip your code or follow any packaging methods you wish but DO INCLUDE a README file with instructions on how to execute or use your application
- Best practices link -  
<https://stackoverflow.blog/2020/03/02/best-practices-for-rest-api-design/>