PacilMate
Week 7 - Deliverable

Kelompok B14

Aditya Pratama	1706039490
Farah Nazihah	1906350761
Hocky Yudhiono	1906285604
Muhammad Urwatil Wutsqo	1906351101
Wiena Amanda	1806186591

https://gitlab.com/pacilmate
https://www.youtube.com/watch?v=MSDkZ8EgjQA

Latar Belakang

Pada masa pandemi ini, **semakin ramai penggunaan** berbagai aplikasi **chat**. Tidak jarang kita ingin melakukan **janjian atau pertemuan** dengan orang-orang. Namun terkadang kita lupa dalam membuat agenda tersebut di kalender masing-masing.

Setelah berjanjian, alangkah baiknya **bila ada cara instan** yang dapat langsung diimplementasikan di tempat. Misalnya dalam agenda organisasi, kita dapat mengatur sebuah agenda yang dapat diikuti **semua orang secara langsung dari kotak chat**.

Ide Aplikasi

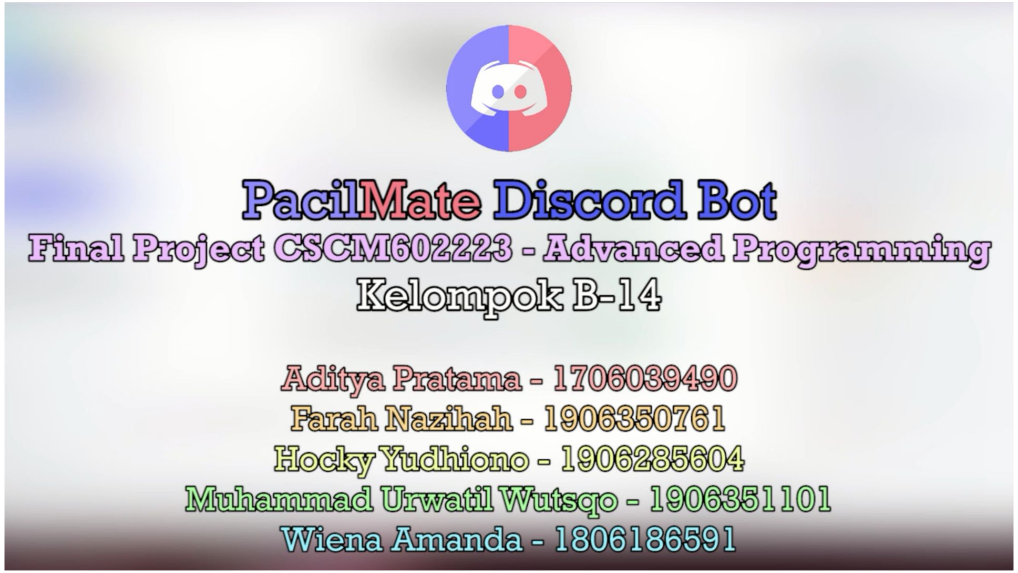
PacilMate ialah Calender + Utility Bot yang lebih relate ke Mahasiswa Fasilkom. Bot ini dalam bentuk paling sederhana akan berfungsi sebagai kalender yang semua orang dapat gunakan fitur-fitur dasarnya seperti **tambah event, hapus event, dan subscribe event**, dan secara global dapat difungsikan sebagai kumpulan event, menjadi kalender. Kemudian juga ada grader yang bisa digunakan untuk mengadakan kuis secara global!

Tujuan dan Manfaat

Memberikan **kemudahan dan utilitas lebih** bagi para mahasiswa dalam beraktivitas melalui aplikasi Discord, salah satu platform chat yang sedang populer karena kemudahan adanya kanal suara dan kanal streaming serta pembagian kanal *chat* yang mudah.

Link Video Demo Aplikasi

https://www.youtube.com/watch?v=MSDkZ8EgjQA



PacilMate Discord Bot

Final Project CSCM602223 - Advanced Programming

Kelompok B-14

Aditya Pratama - 1706039490
Farah Nazihah - 1906350761
Hocky Yudhiono - 1906285604
Muhammad Urwatil Wutsqo - 1906351101
Wiena Amanda - 1806186591

Tahapan Realisasi

- Menggunakan berbagai design pattern yang sesuai
- Menggunakan Java Spring Framework
- Menggunakan JDA sebagai Wrapper Discord, framework yang memudahkan pengembangan Discord Chat Bot
- Mengimplementasikan topik-topik yang seturut dengan pengembangan aplikasi ini.

Cara Kerja Aplikasi

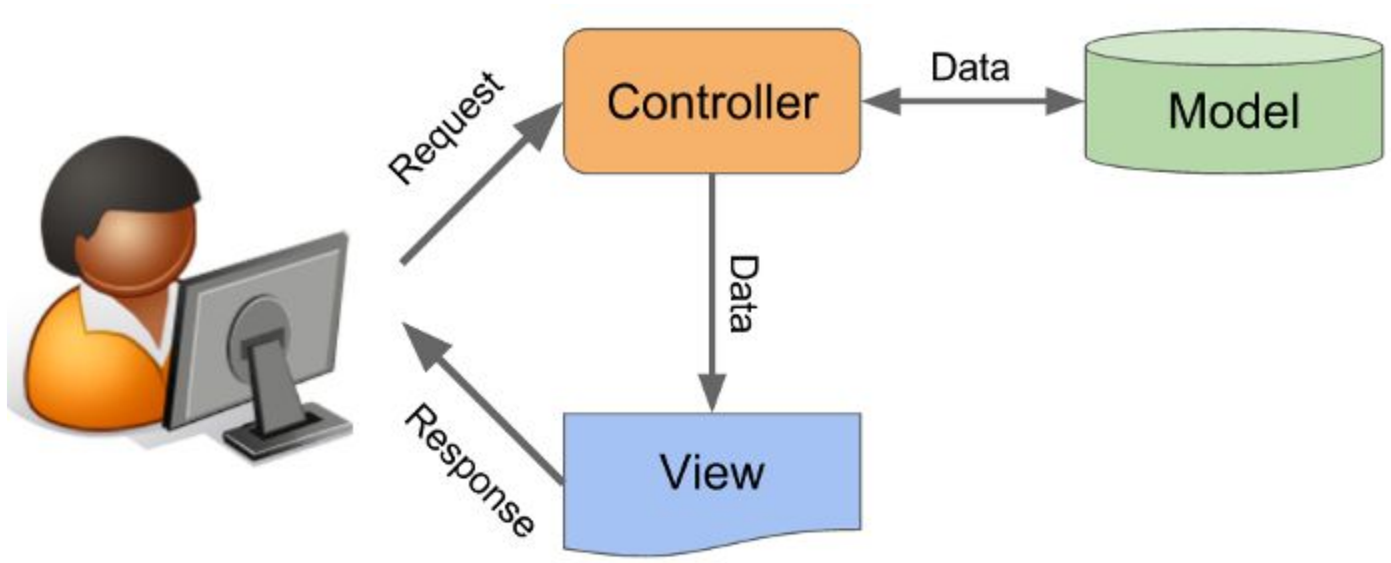
Pada chatbot, akan ada **controller** dalam bentuk **listener**.

Event yang ada akan diberikan secara event driven. Chatbot akan memiliki beberapa **Listener,** yang masing-masing akan dikategorikan commandnya apa saja.

Setelah itu, listener akan memanggil service sesuai dengan kategorinya masing-masing.

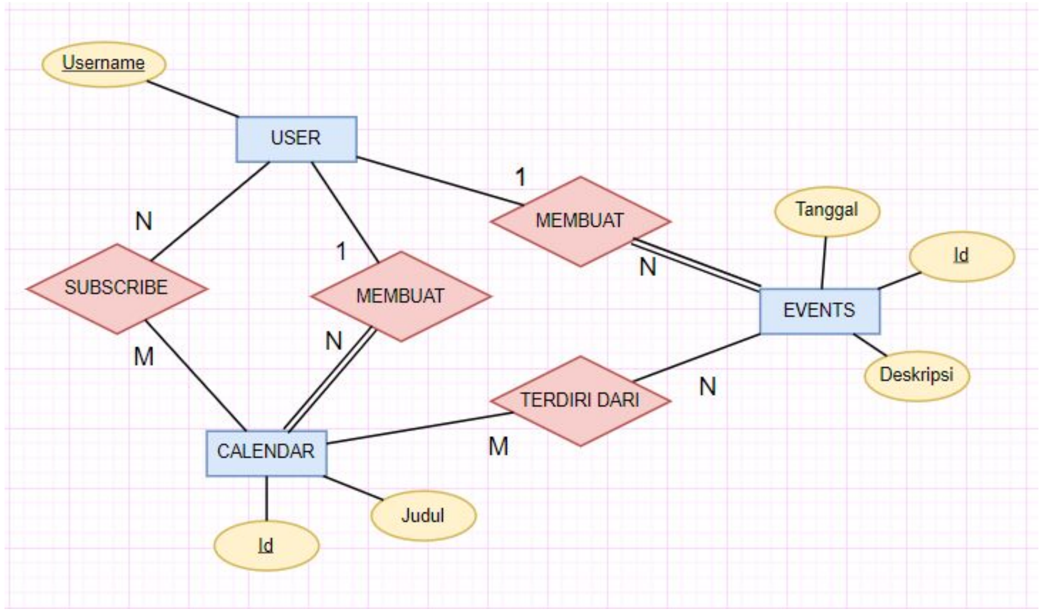
Akan digunakan arsitektur **MVC**. Dengan view berupa chat dengan fitur-fitur markup language, **Markdown**, yang pada dasarnya serupa dengan Web App.

Cara Kerja Aplikasi



Cara Kerja Aplikasi

Akan ada model **Events**, **Calendar**, dan **User**. Dengan **ERD Basis Data** kira-kira sebagai berikut.



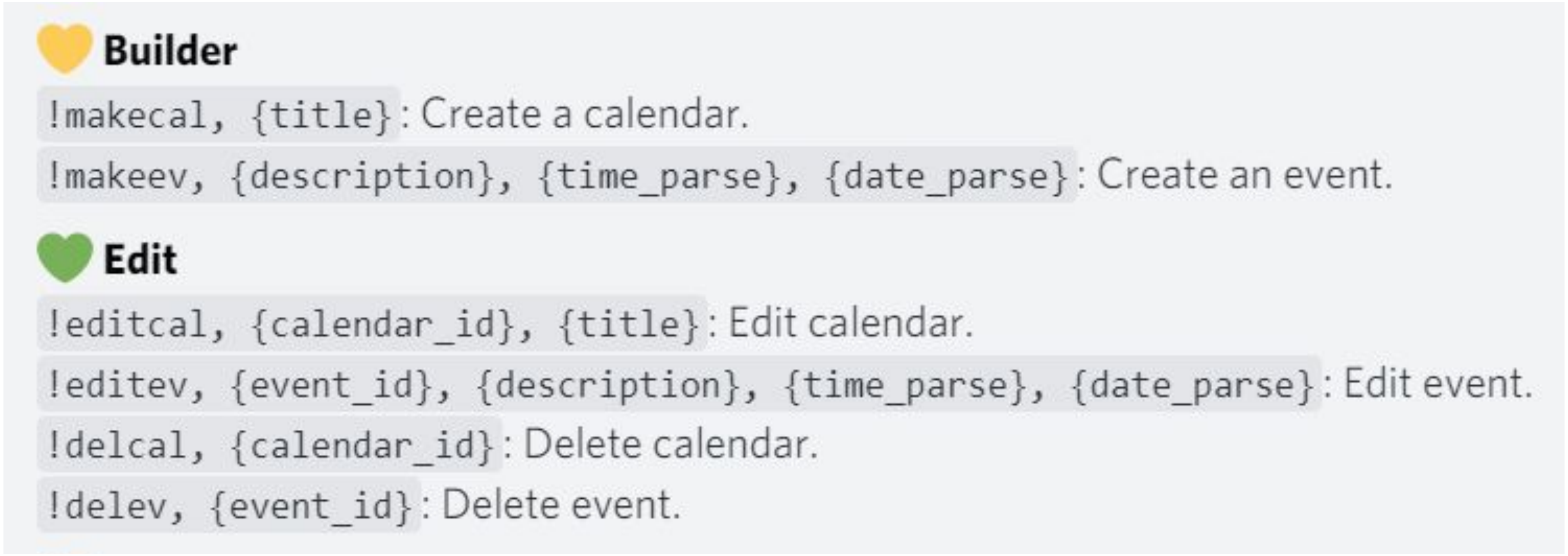
Cara Kerja Aplikasi

Akan ada beberapa class view lain untuk membuat struktur dari chat embed dari kalender dan events-events yang ada. Endpoint listener juga akan ditambahkan operasi **CRUD** lainnya. 😁

Pembagian Fokus Pengembangan Fitur

Fitur Lookup Global - Farah Nazihah
Fitur Calendar Lookup - Wiena Amanda
Fitur Builder dan Judge - Hocky Yudhiono
Fitur Manage Calendar dan Notification - Muhammad Urwatil Wutsqo
Fitur Manage Events dan Specific Lookup - Aditya Pratama

Fitur Builder dan Edit



💛 **Builder**

`!makecal, {title}`: Create a calendar.

`!makeev, {description}, {time_parse}, {date_parse}`: Create an event.

💚 **Edit**

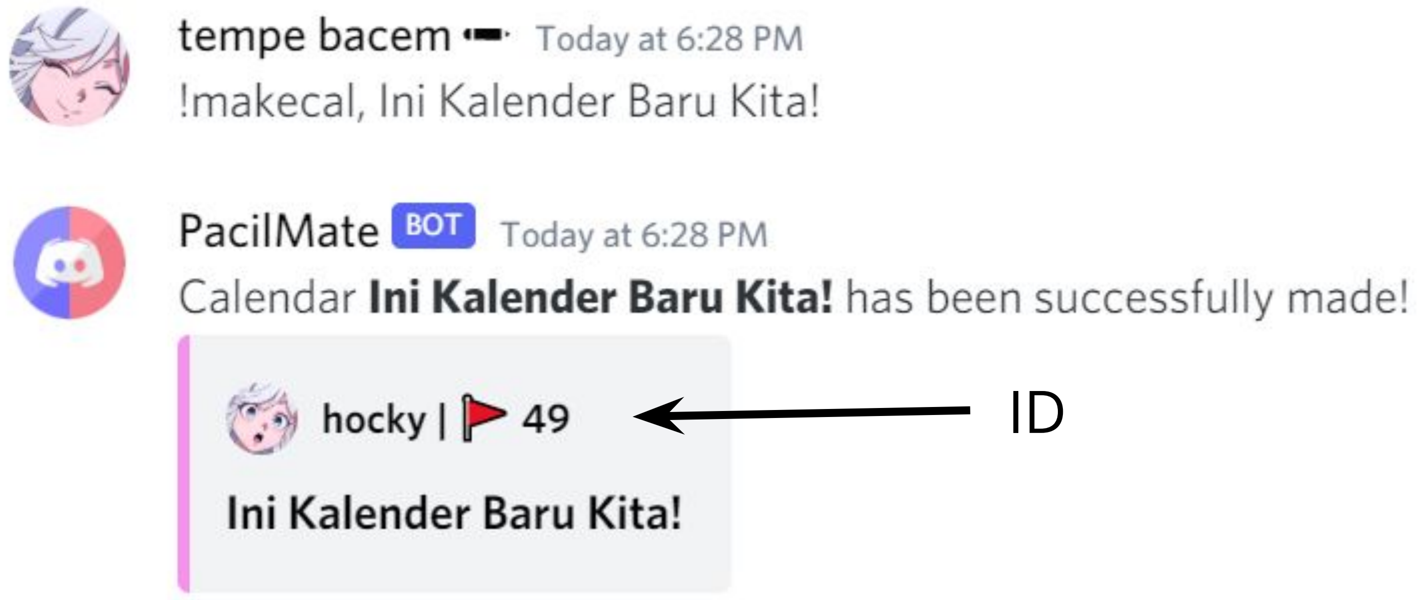
`!editcal, {calendar_id}, {title}`: Edit calendar.

`!editev, {event_id}, {description}, {time_parse}, {date_parse}`: Edit event.

`!delcal, {calendar_id}`: Delete calendar.

`!delev, {event_id}`: Delete event.

Fitur Builder dan Edit



tempe bacem Today at 6:28 PM

!makecal, Ini Kalender Baru Kita!

PacilMate BOT Today at 6:28 PM

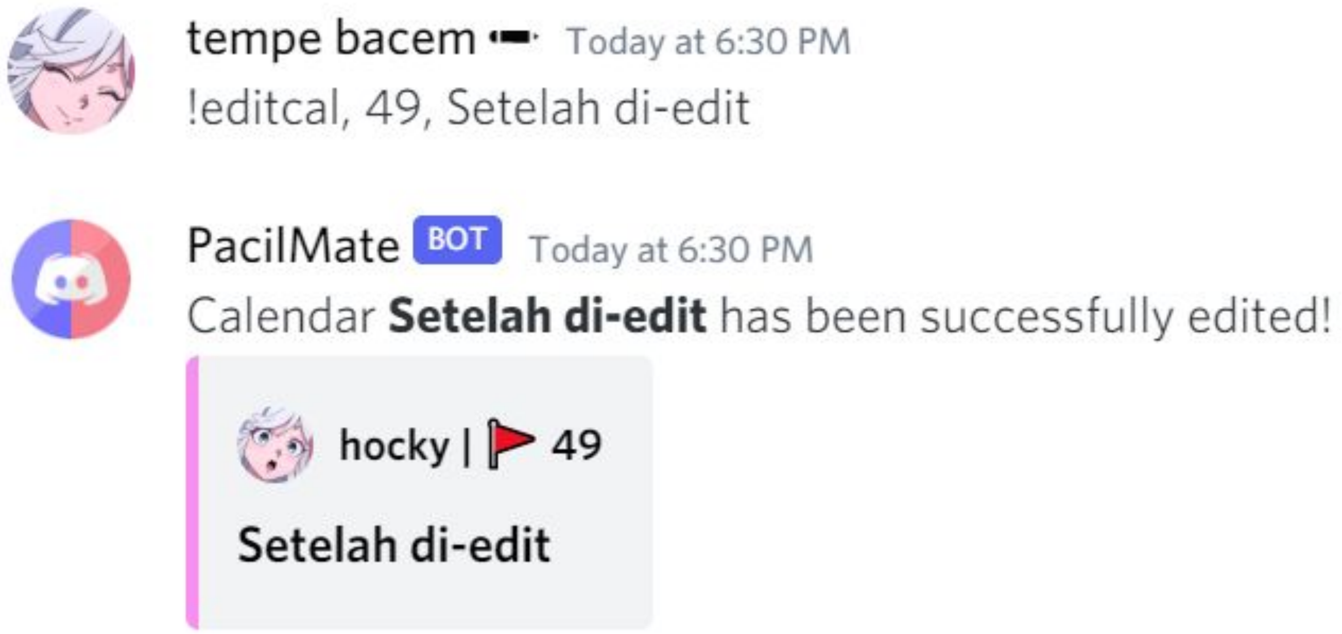
Calendar **Ini Kalender Baru Kita!** has been successfully made!

hocky | 🚩 49

Ini Kalender Baru Kita!

ID

Fitur Builder dan Edit


tempe bacem Today at 6:30 PM

!editcal, 49, Setelah di-edit

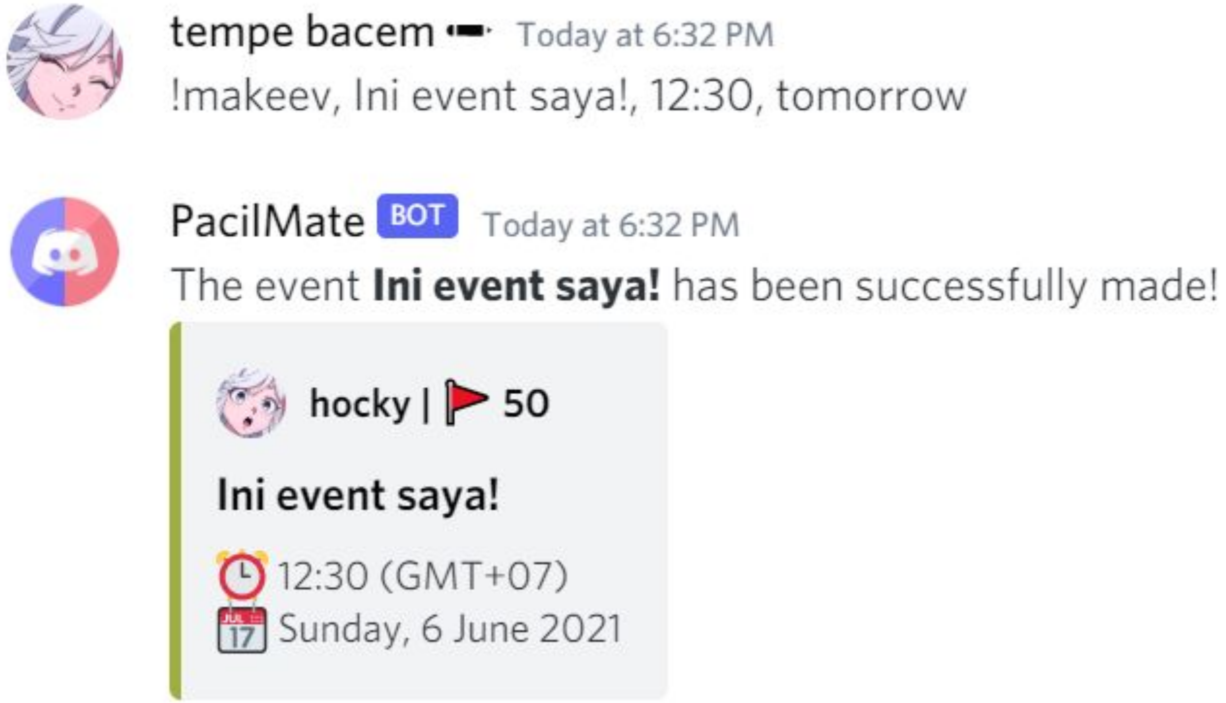
PacilMate BOT Today at 6:30 PM

Calendar **Setelah di-edit** has been successfully edited!

hocky | 🚩 49

Setelah di-edit

Fitur Builder dan Edit



tempe bacem Today at 6:32 PM

!makeev, Ini event saya!, 12:30, tomorrow

PacilMate BOT Today at 6:32 PM

The event **Ini event saya!** has been successfully made!

hocky | 🚩 50

Ini event saya!

⏰ 12:30 (GMT+07)
📅 Sunday, 6 June 2021

Fitur Builder dan Edit

date_parse bisa berupa:

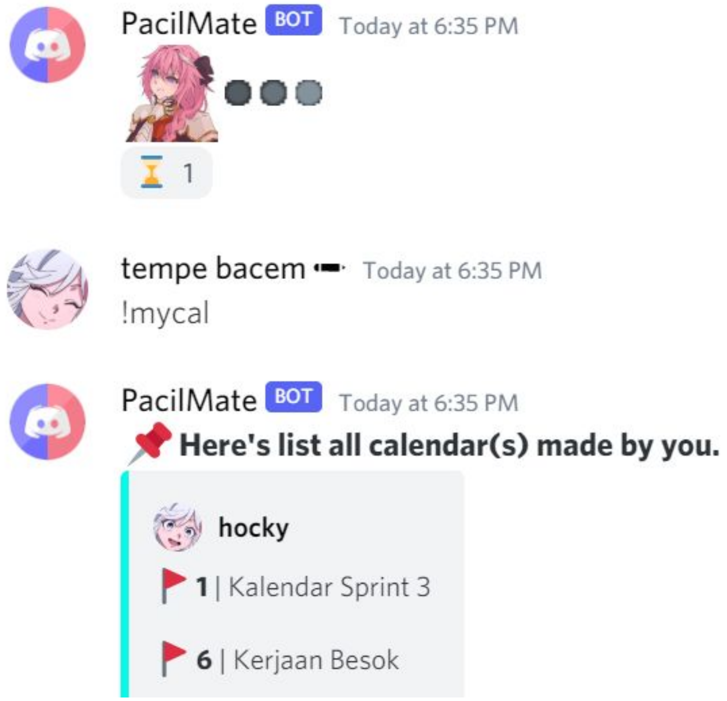
- today
- tomorrow
- next week
- next month
- next year
- dd-MM-yyyy

time_parse bisa berupa:

- **{jam}**h
- **{menit}**m
- hh:mm

Fitur Judge

Semua fitur Judge loading secara asynchronous, bisa melakukan proses lain selama menunggu koneksi dengan **PacilJudge**


PacilMate BOT Today at 6:35 PM

⌛ 1

tempe bacem Today at 6:35 PM
!mycal

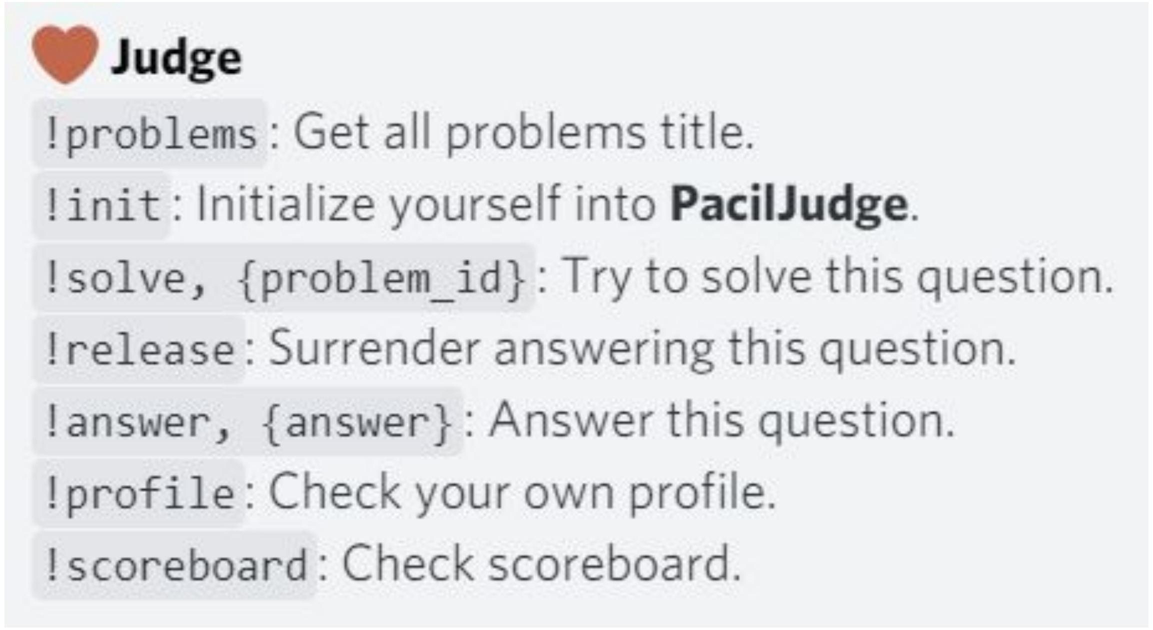
PacilMate BOT Today at 6:35 PM
📌 **Here's list all calendar(s) made by you.**

hocky

🚩 **1** | Kalendar Sprint 3

🚩 **6** | Kerjaan Besok

Fitur Judge


❤️ **Judge**

`!problems` : Get all problems title.

`!init` : Initialize yourself into **PacilJudge**.

`!solve, {problem_id}` : Try to solve this question.

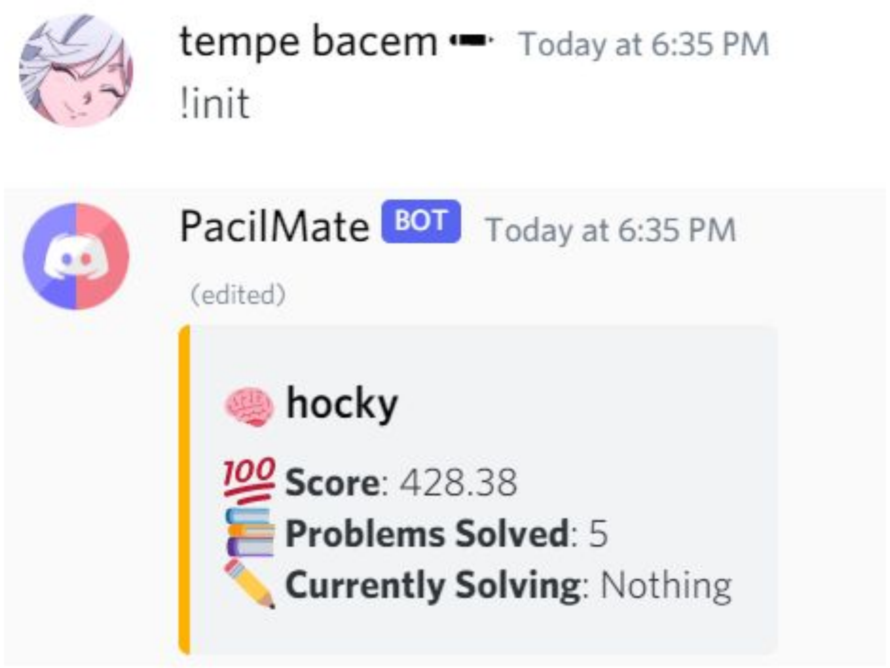
`!release` : Surrender answering this question.

`!answer, {answer}` : Answer this question.

`!profile` : Check your own profile.

`!scoreboard` : Check scoreboard.

Fitur Judge



tempe bacem Today at 6:35 PM

!init

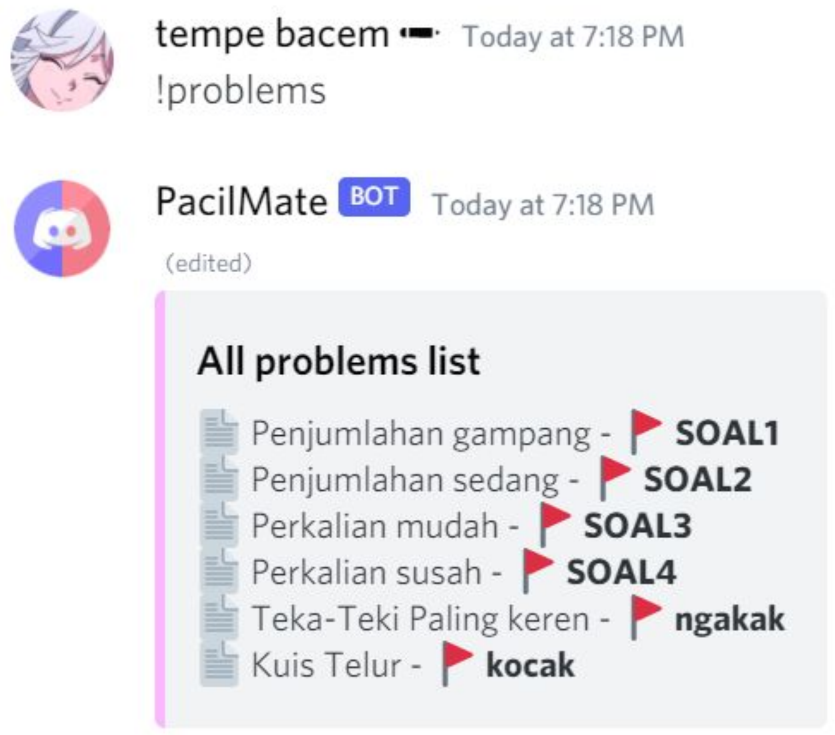
PacilMate BOT Today at 6:35 PM

(edited)

🧠 hocky

💯 **Score**: 428.38
📚 **Problems Solved**: 5
✏️ **Currently Solving**: Nothing

Fitur Judge



tempe bacem Today at 7:18 PM

!problems

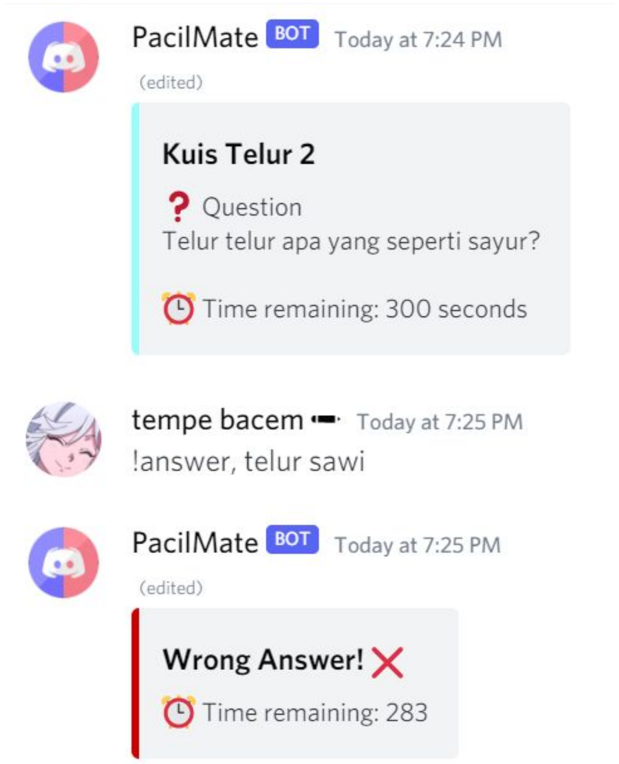
PacilMate BOT Today at 7:18 PM

(edited)

All problems list

📄 Penjumlahan gampang - 🚩 **SOAL1**
📄 Penjumlahan sedang - 🚩 **SOAL2**
📄 Perkalian mudah - 🚩 **SOAL3**
📄 Perkalian susah - 🚩 **SOAL4**
📄 Teka-Teki Paling keren - 🚩 **ngakak**
📄 Kuis Telur - 🚩 **kocak**

Fitur Judge


PacilMate BOT Today at 7:24 PM
(edited)

Kuis Telur 2

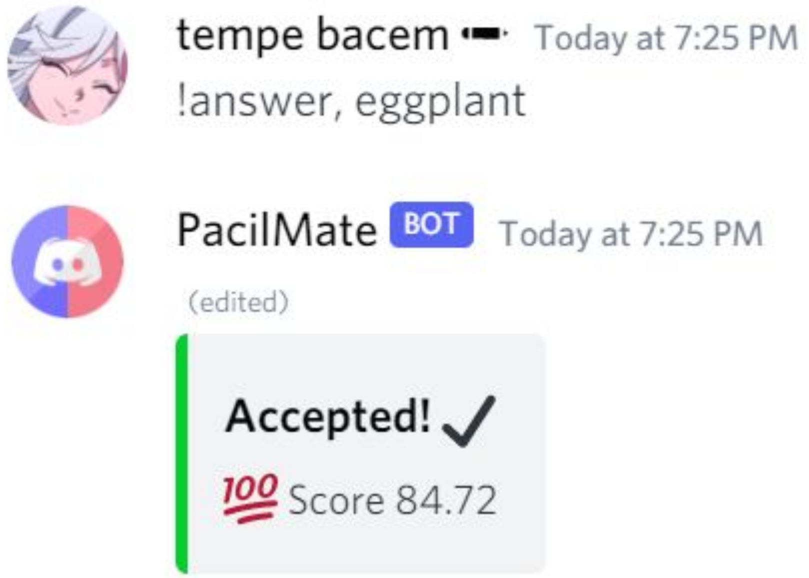
❓ Question
Telur telur apa yang seperti sayur?

⏰ Time remaining: 300 seconds

tempe bacem Today at 7:25 PM
!answer, telur sawi

PacilMate BOT Today at 7:25 PM
(edited)

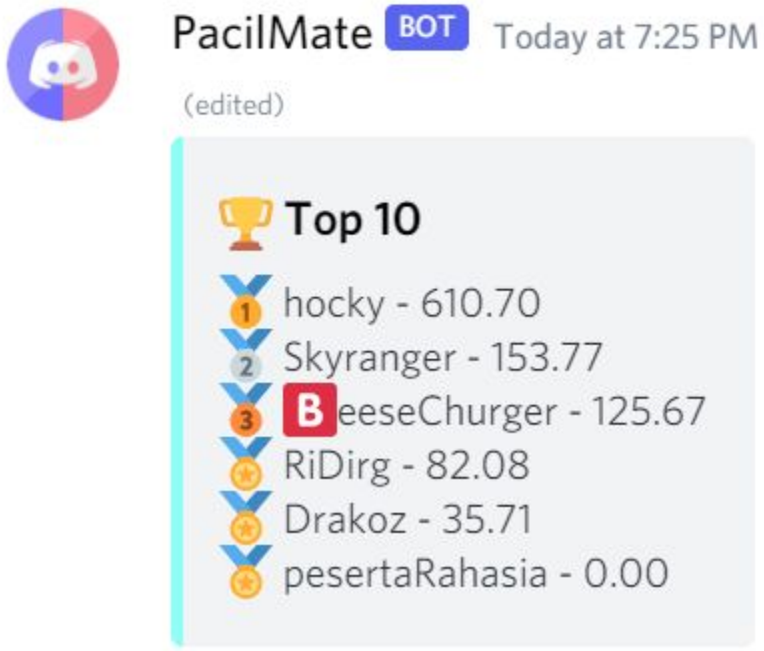
Wrong Answer! ❌
⏰ Time remaining: 283


tempe bacem Today at 7:25 PM
!answer, eggplant

PacilMate BOT Today at 7:25 PM
(edited)

Accepted! ✔
💯 Score 84.72

Fitur Judge

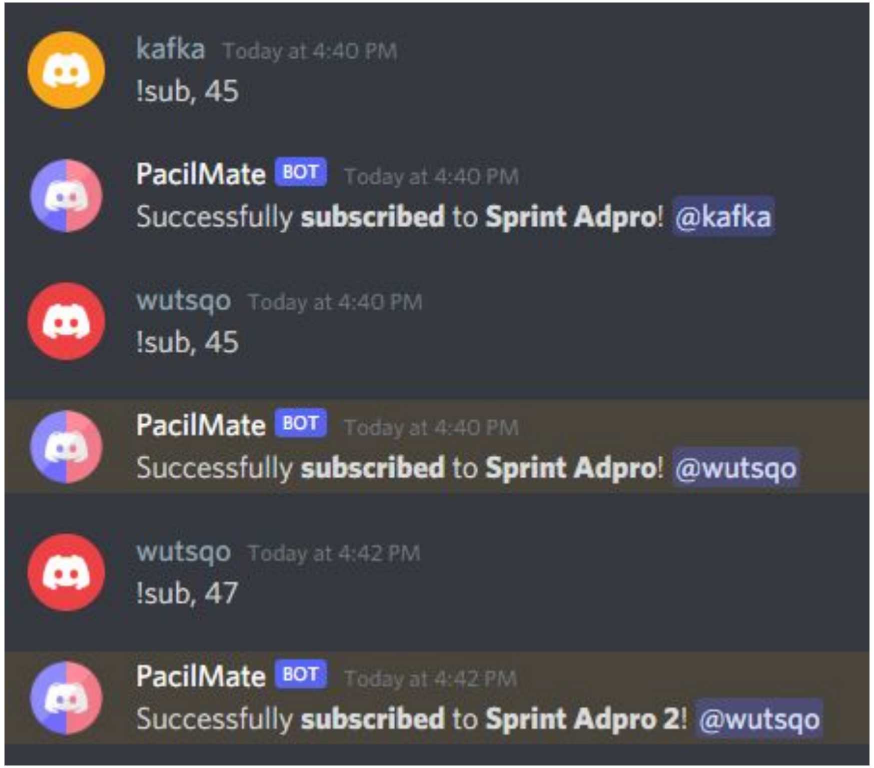


PacilMate BOT Today at 7:25 PM
(edited)

🏆 **Top 10**

🥇 hocky - 610.70
🥈 Skyranger - 153.77
🥉 🅱eeseChurger - 125.67
🏅 RiDirg - 82.08
🏅 Drakoz - 35.71
🏅 pesertaRahasia - 0.00

Fitur Manage Calendar



kafka Today at 4:40 PM
!sub, 45

PacilMate BOT Today at 4:40 PM
Successfully **subscribed** to **Sprint Adpro**! @kafka

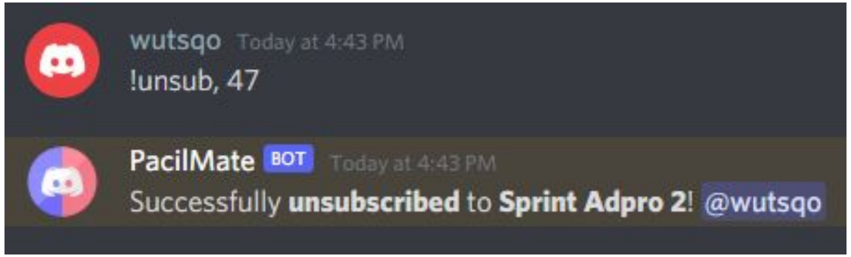
wutsqo Today at 4:40 PM
!sub, 45

PacilMate BOT Today at 4:40 PM
Successfully **subscribed** to **Sprint Adpro**! @wutsqo

wutsqo Today at 4:42 PM
!sub, 47

PacilMate BOT Today at 4:42 PM
Successfully **subscribed** to **Sprint Adpro 2**! @wutsqo

Setiap user dapat mensubscribe satu atau beberapa calendar. User yang sudah mensubscribe sebuah calendar juga dapat membatalkan subscriptionnya.

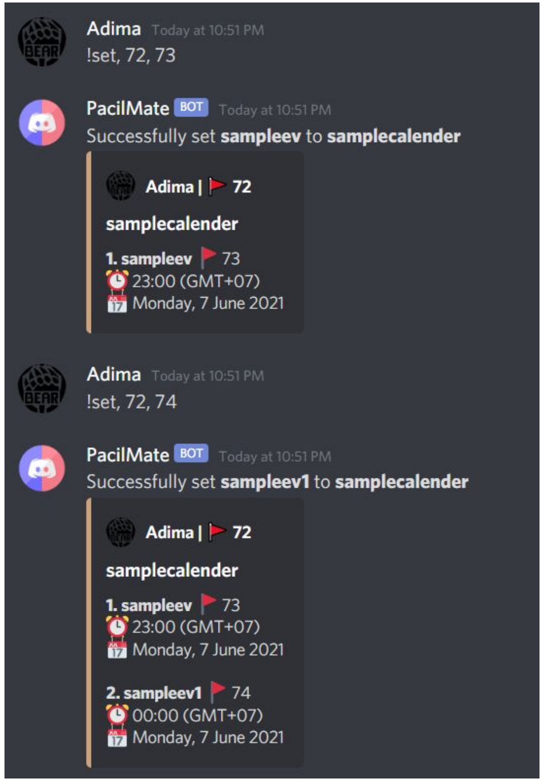


wutsqo Today at 4:43 PM
!unsub, 47

PacilMate BOT Today at 4:43 PM
Successfully **unsubscribed** to **Sprint Adpro 2**! @wutsqo

Fitur Manage Event

Setiap user dapat mengatur event yang diinginkan ke dalam kalendernya sendiri dan mengatur event yang tidak diinginkan dalam kalendernya.



Adima Today at 10:51 PM
!set, 72, 73

PacilMate BOT Today at 10:51 PM
Successfully set **sampleev** to **samplecalender**

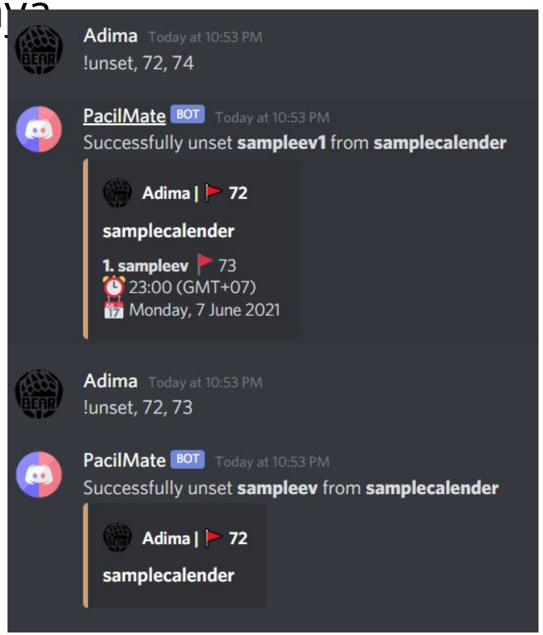
Adima | 🚩 72
samplecalender
1. sampleev 🚩 73
⏰ 23:00 (GMT+07)
📅 Monday, 7 June 2021

Adima Today at 10:51 PM
!set, 72, 74

PacilMate BOT Today at 10:51 PM
Successfully set **sampleev1** to **samplecalender**

Adima | 🚩 72
samplecalender
1. sampleev 🚩 73
⏰ 23:00 (GMT+07)
📅 Monday, 7 June 2021

2. sampleev1 🚩 74
⏰ 00:00 (GMT+07)
📅 Monday, 7 June 2021



Adima Today at 10:53 PM
!unset, 72, 74

PacilMate BOT Today at 10:53 PM
Successfully unset **sampleev1** from **samplecalender**

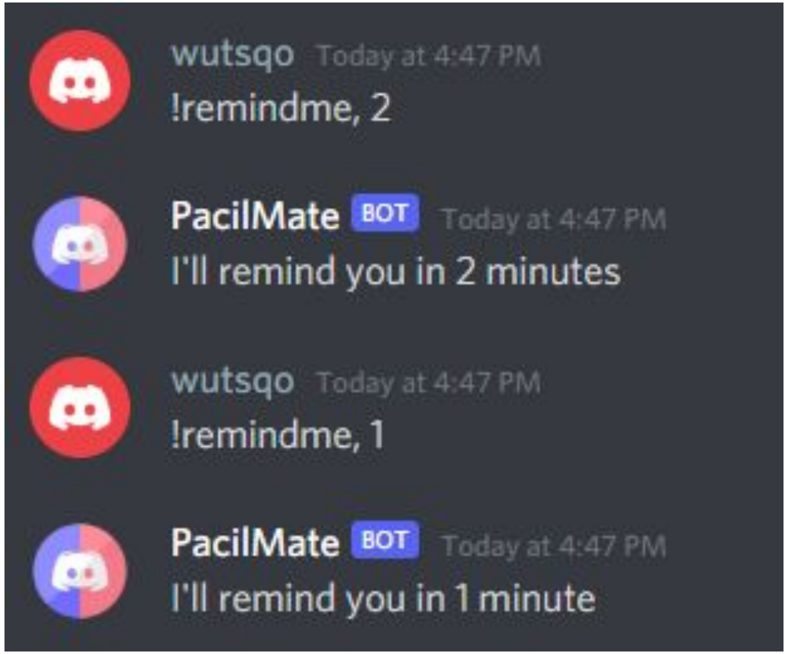
Adima | 🚩 72
samplecalender
1. sampleev 🚩 73
⏰ 23:00 (GMT+07)
📅 Monday, 7 June 2021

Adima Today at 10:53 PM
!unset, 72, 73

PacilMate BOT Today at 10:53 PM
Successfully unset **sampleev** from **samplecalender**

Adima | 🚩 72
samplecalender

Fitur Notification #1: Remind Me

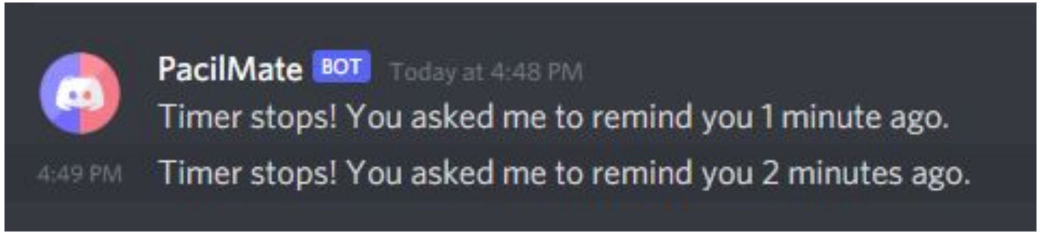

wutsqo Today at 4:47 PM
!remindme, 2

PacilMate BOT Today at 4:47 PM
I'll remind you in 2 minutes

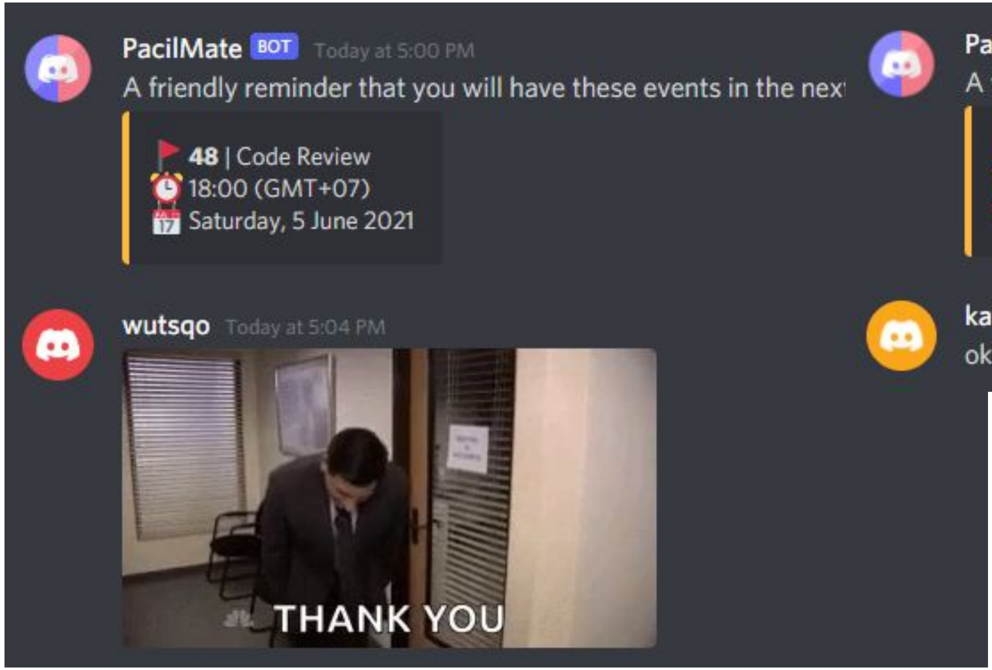
wutsqo Today at 4:47 PM
!remindme, 1

PacilMate BOT Today at 4:47 PM
I'll remind you in 1 minute

Fitur ini berfungsi layaknya countdown timer. User dapat mengatur berapa lama timer akan berjalan. Lalu bot akan mengirim notifikasi melalui direct message.

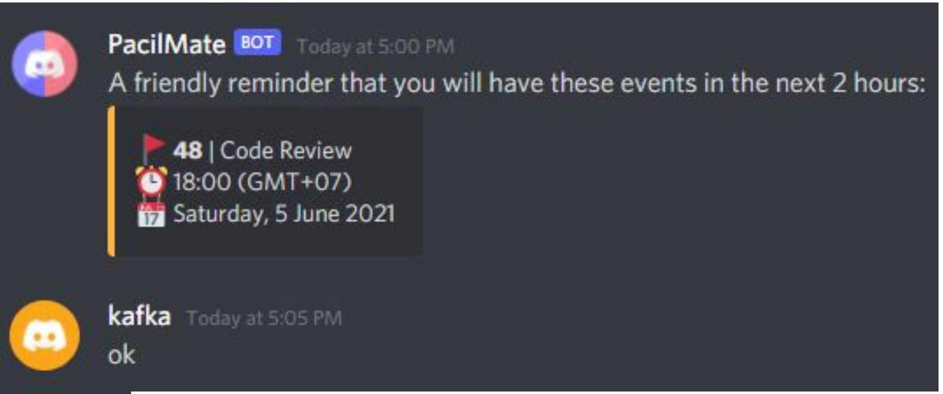

PacilMate BOT Today at 4:48 PM
Timer stops! You asked me to remind you 1 minute ago.
4:49 PM Timer stops! You asked me to remind you 2 minutes ago.

Fitur Notification #2: Event Reminder


PacilMate BOT Today at 5:00 PM
A friendly reminder that you will have these events in the next

48 | Code Review
18:00 (GMT+07)
Saturday, 5 June 2021

wutsqo Today at 5:04 PM

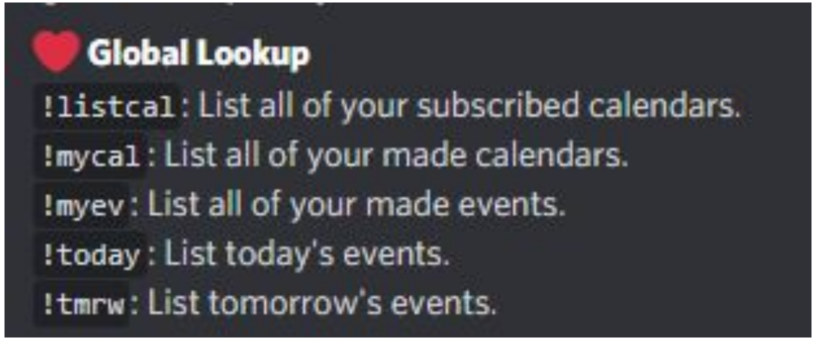

PacilMate BOT Today at 5:00 PM
A friendly reminder that you will have these events in the next 2 hours:

48 | Code Review
18:00 (GMT+07)
Saturday, 5 June 2021

kafka Today at 5:05 PM
ok

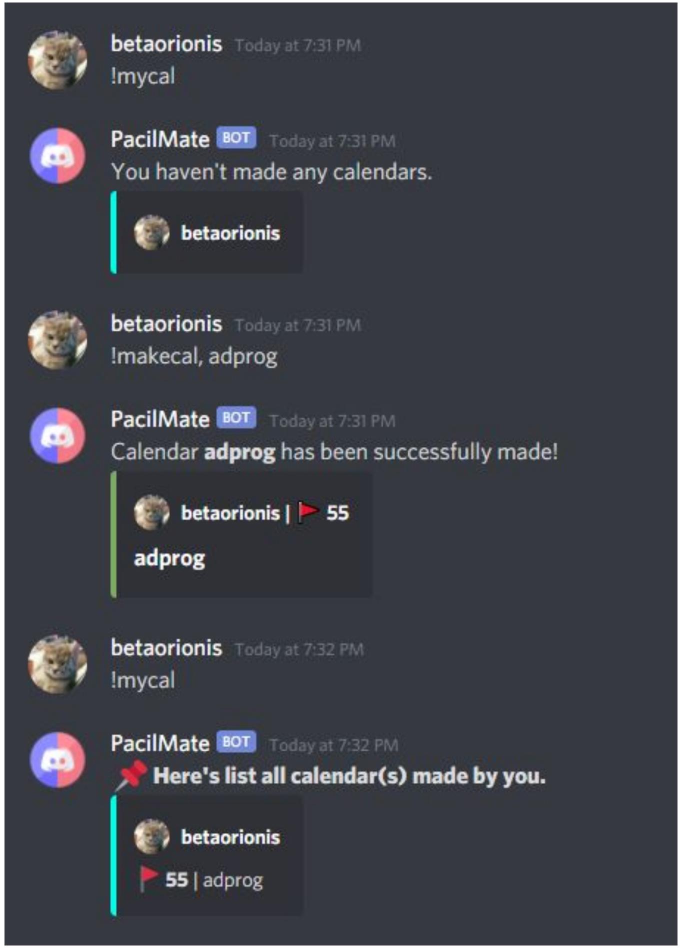
Fitur ini akan mengirim pesan reminder secara berkala melalui direct message kepada para subscriber dari sebuah kalender terkait event yang akan berlangsung.

Fitur lookup global #1



Global Lookup

`!listcal`: List all of your subscribed calendars.
`!mycal`: List all of your made calendars.
`!myev`: List all of your made events.
`!today`: List today's events.
`!tmrw`: List tomorrow's events.



betaorionis Today at 7:31 PM
!mycal

PacilMate BOT Today at 7:31 PM
You haven't made any calendars.

betaorionis

betaorionis Today at 7:31 PM
!makecal, adprog

PacilMate BOT Today at 7:31 PM
Calendar **adprog** has been successfully made!

betaorionis | 🚩 55

adprog

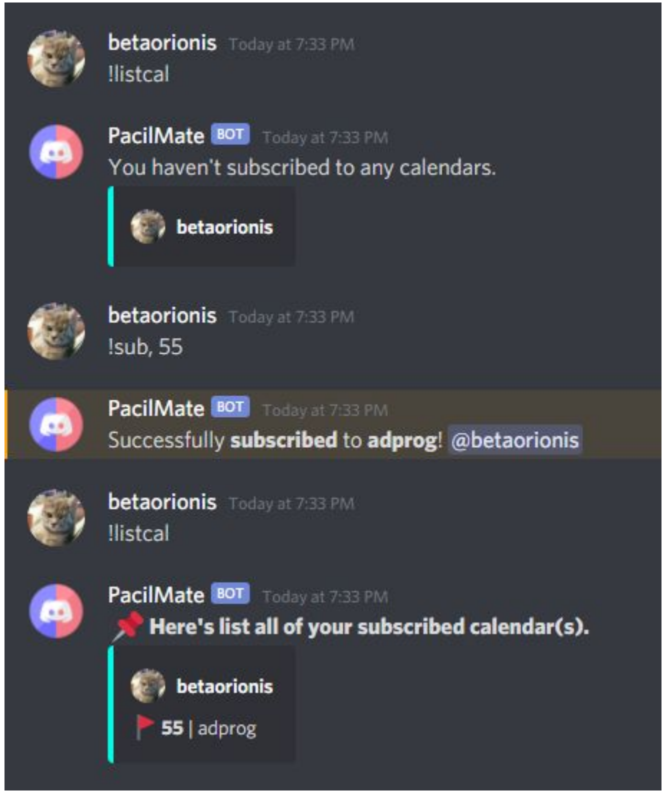
betaorionis Today at 7:32 PM
!mycal

PacilMate BOT Today at 7:32 PM
📌 **Here's list all calendar(s) made by you.**

betaorionis

🚩 **55** | adprog

Fitur lookup global #2



betaorionis Today at 7:33 PM
!listcal

PacilMate BOT Today at 7:33 PM
You haven't subscribed to any calendars.

betaorionis

betaorionis Today at 7:33 PM
!sub, 55

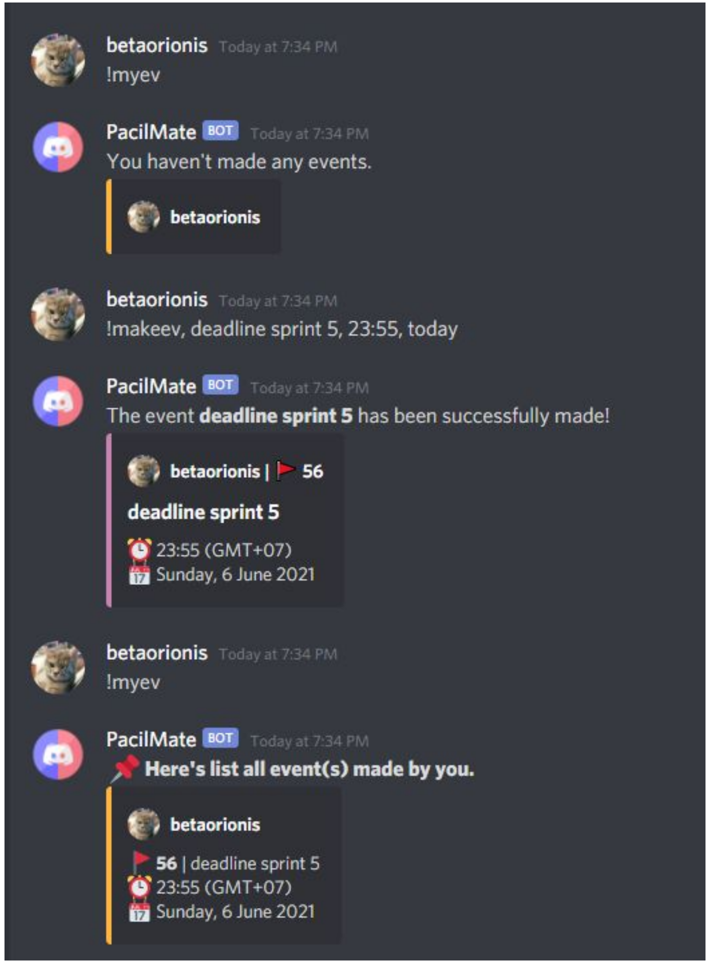
PacilMate BOT Today at 7:33 PM
Successfully **subscribed** to **adprog**! @betaorionis

betaorionis Today at 7:33 PM
!listcal

PacilMate BOT Today at 7:33 PM
📌 **Here's list all of your subscribed calendar(s).**

betaorionis

🚩 **55** | adprog



betaorionis Today at 7:34 PM
!myev

PacilMate BOT Today at 7:34 PM
You haven't made any events.

betaorionis

betaorionis Today at 7:34 PM
!makeev, deadline sprint 5, 23:55, today

PacilMate BOT Today at 7:34 PM
The event **deadline sprint 5** has been successfully made!

betaorionis | 🚩 56

deadline sprint 5

⏰ 23:55 (GMT+07)
📅 Sunday, 6 June 2021

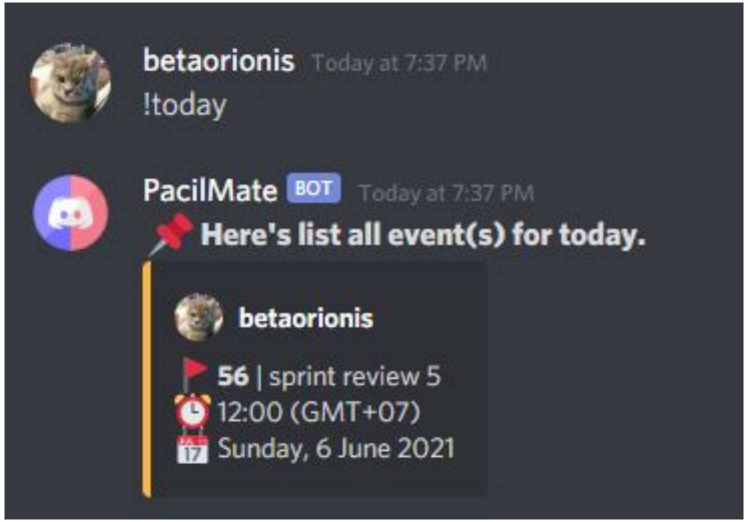
betaorionis Today at 7:34 PM
!myev

PacilMate BOT Today at 7:34 PM
📌 **Here's list all event(s) made by you.**

betaorionis

🚩 **56** | deadline sprint 5
⏰ 23:55 (GMT+07)
📅 Sunday, 6 June 2021

Fitur lookup global #3

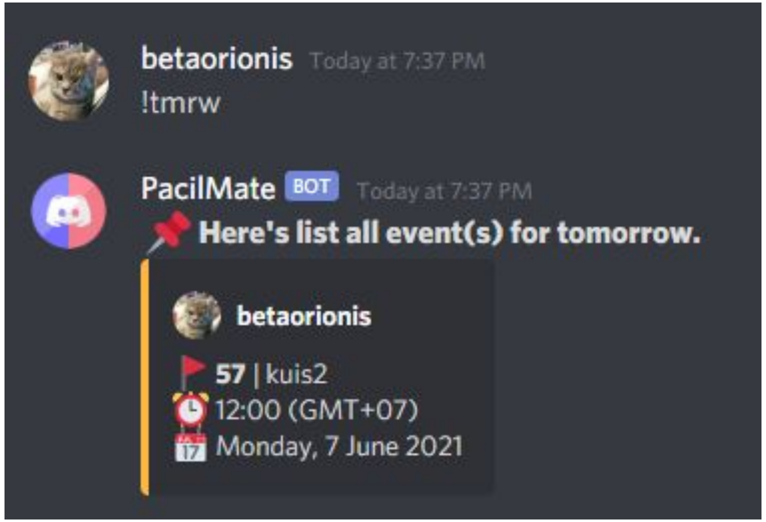


betaorionis Today at 7:37 PM
!today

PacilMate BOT Today at 7:37 PM
📌 **Here's list all event(s) for today.**

betaorionis

🚩 **56** | sprint review 5
⏰ 12:00 (GMT+07)
📅 Sunday, 6 June 2021



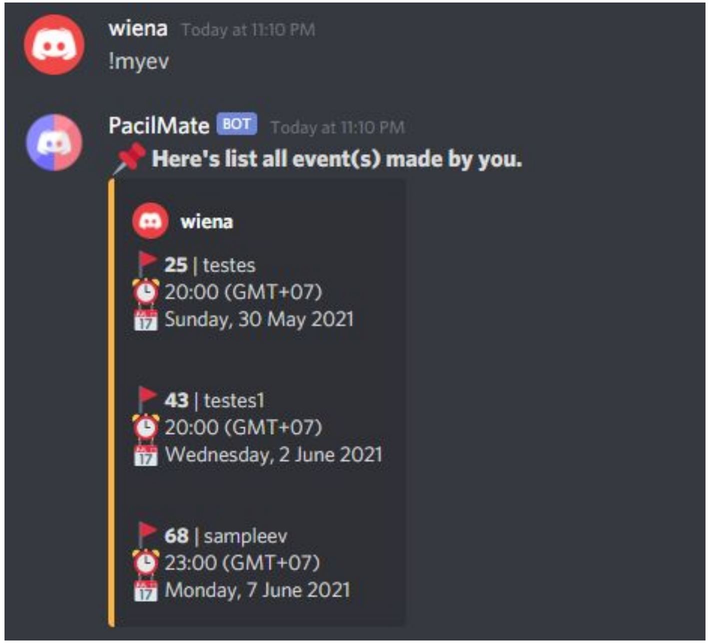
betaorionis Today at 7:37 PM
!tmrw

PacilMate BOT Today at 7:37 PM
📌 **Here's list all event(s) for tomorrow.**

betaorionis

🚩 **57** | kuis2
⏰ 12:00 (GMT+07)
📅 Monday, 7 June 2021

Fitur lookup calendar#1


wiena Today at 11:10 PM
!myev

PacilMate BOT Today at 11:10 PM
📌 **Here's list all event(s) made by you.**

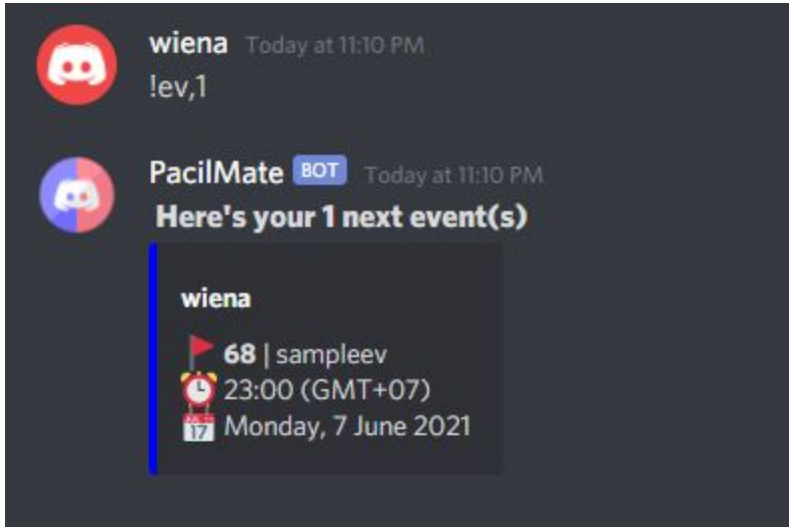
wiena

🚩 **25** | testes
⏰ 20:00 (GMT+07)
📅 Sunday, 30 May 2021

🚩 **43** | testes1
⏰ 20:00 (GMT+07)
📅 Wednesday, 2 June 2021

🚩 **68** | sampleev
⏰ 23:00 (GMT+07)
📅 Monday, 7 June 2021

Contoh event yang dimiliki


wiena Today at 11:10 PM
!ev,1

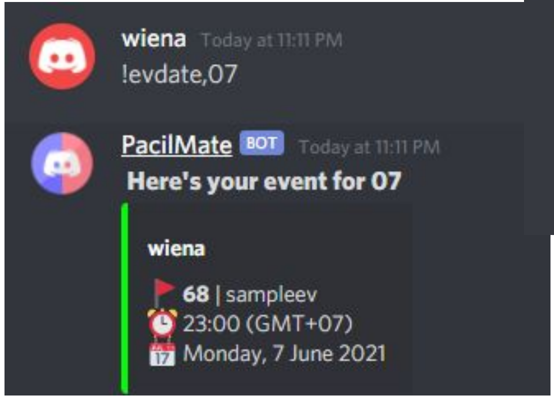
PacilMate BOT Today at 11:10 PM
Here's your 1 next event(s)

wiena

🚩 **68** | sampleev
⏰ 23:00 (GMT+07)
📅 Monday, 7 June 2021

Fitur ini akan menampilkan event selanjutnya sesuai yg diinput. misal !ev,1 yang berarti menampilkan 1 event selanjutnya

Fitur lookup calendar#2

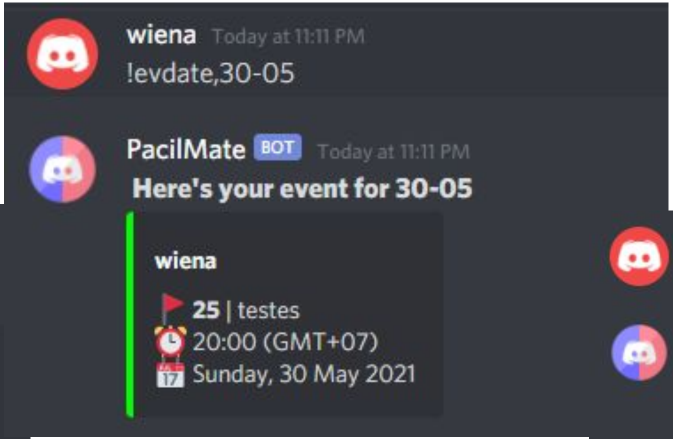


wiena Today at 11:11 PM
!evdate,07

PacilMate BOT Today at 11:11 PM
Here's your event for 07

wiena

🚩 **68** | sampleev
⏰ 23:00 (GMT+07)
📅 Monday, 7 June 2021

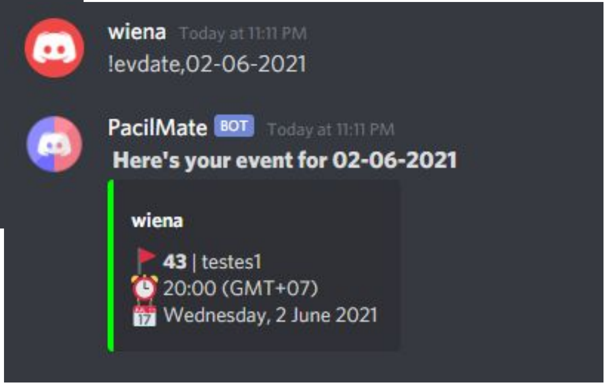


wiena Today at 11:11 PM
!evdate,30-05

PacilMate BOT Today at 11:11 PM
Here's your event for 30-05

wiena

🚩 **25** | testes
⏰ 20:00 (GMT+07)
📅 Sunday, 30 May 2021



wiena Today at 11:11 PM
!evdate,02-06-2021

PacilMate BOT Today at 11:11 PM
Here's your event for 02-06-2021

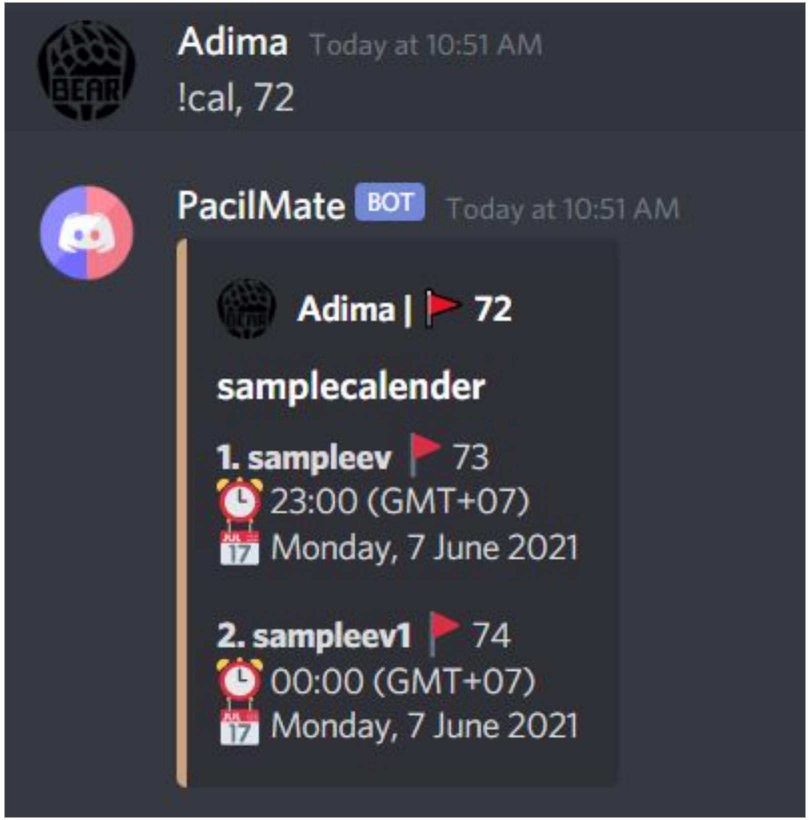
wiena

🚩 **43** | testes1
⏰ 20:00 (GMT+07)
📅 Wednesday, 2 June 2021

Fitur ini akan menampilkan event secara lebih spesifik. bisa berdasarkan tanggal, tanggal-bulan atau tanggal-bulan-tahun.

Fitur lookup calendar#3

Fitur lookup calendar dapat menampilkan seluruh event yang terdaftar pada salah satu calendar yang ada.



Adima Today at 10:51 AM
!cal, 72

PacilMate BOT Today at 10:51 AM

Adima | 🚩 72

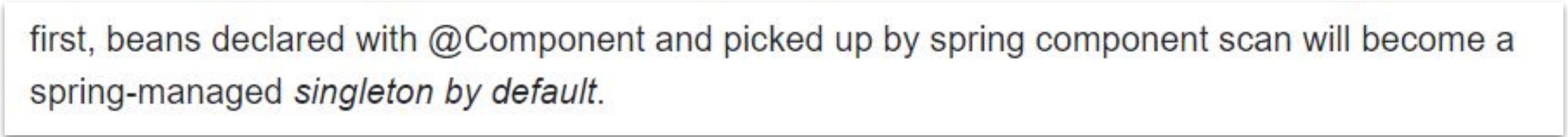
samplecalender

1. sampleev 🚩 73
⏰ 23:00 (GMT+07)
📅 Monday, 7 June 2021

2. sampleev1 🚩 74
⏰ 00:00 (GMT+07)
📅 Monday, 7 June 2021

Penerapan Design Pattern di Code

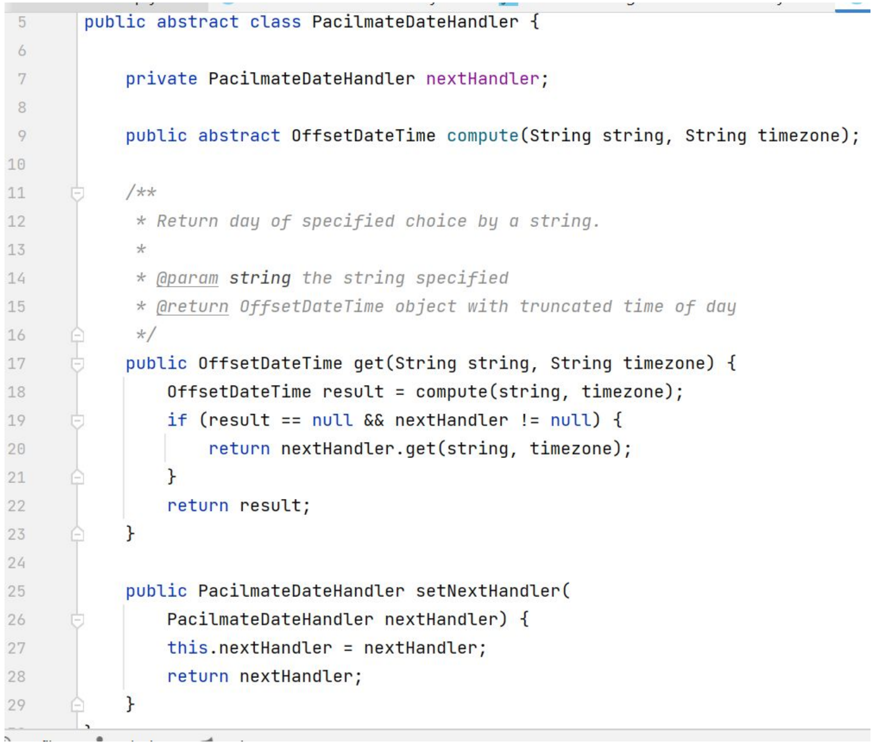
- **Singleton Pattern,** ternyata wrapper Discord Bot sangat cocok digunakan bersama Spring Boot, Autowired yang digunakan secara default diterapkan secara Singleton. 🦴



first, beans declared with @Component and picked up by spring component scan will become a spring-managed *singleton by default*.

- **Chain of Responsibility Pattern,** untuk datetime parser karena ada beberapa jenis. 🔗
- **Mediator Pattern,** sebelum masuk ke service, pesan dari controller mesti diparse terlebih dahulu, bisa dimanfaatkan sebuah service mediator. 🌉

Penerapan Design Pattern di Code



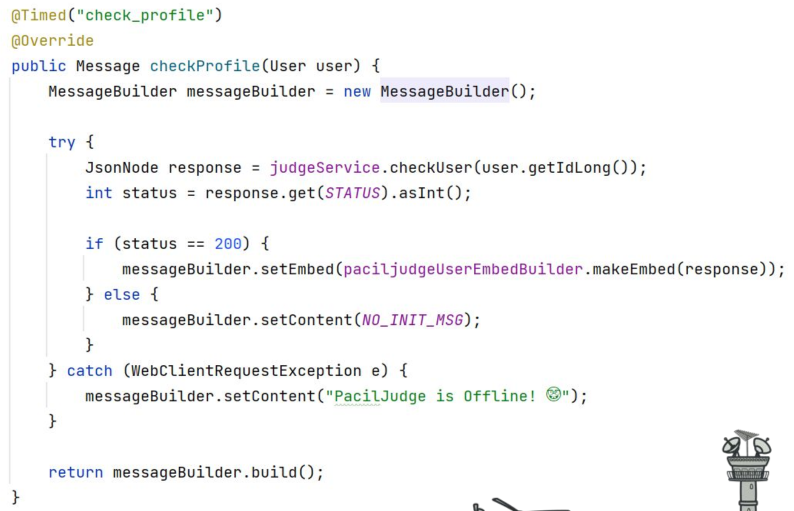
```java
public abstract class PacilmateDateHandler {

    private PacilmateDateHandler nextHandler;

    public abstract OffsetDateTime compute(String string, String timezone);

    /**
     * Return day of specified choice by a string.
     *
     * @param string the string specified
     * @return OffsetDateTime object with truncated time of day
     */
    public OffsetDateTime get(String string, String timezone) {
        OffsetDateTime result = compute(string, timezone);
        if (result == null && nextHandler != null) {
            return nextHandler.get(string, timezone);
        }
        return result;
    }

    public PacilmateDateHandler setNextHandler(
        PacilmateDateHandler nextHandler) {
        this.nextHandler = nextHandler;
        return nextHandler;
    }
```

Sebuah tanggal bisa diparse dengan banyak cara, misal:

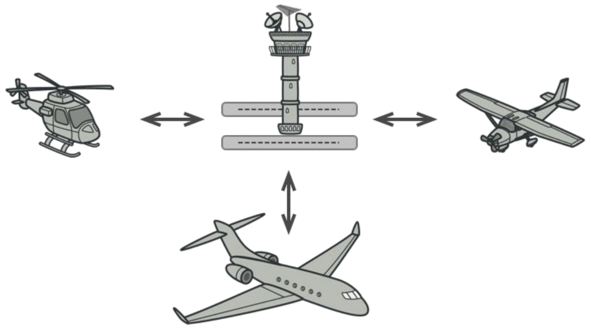
- 15-03-2021
- next year
- today
- tomorrow

Digunakan **chain of responsibility** 👍👍

Penerapan Design Pattern di Code



```java
@Timed("check_profile")
@Override
public Message checkProfile(User user) {
    MessageBuilder messageBuilder = new MessageBuilder();

    try {
        JsonNode response = judgeService.checkUser(user.getIdLong());
        int status = response.get(STATUS).asInt();

        if (status == 200) {
            messageBuilder.setEmbed(paciljudgeUserEmbedBuilder.makeEmbed(response));
        } else {
            messageBuilder.setContent(NO_INIT_MSG);
        }
    } catch (WebClientRequestException e) {
        messageBuilder.setContent("PacilJudge is Offline! 🤓");
    }

    return messageBuilder.build();
}
```



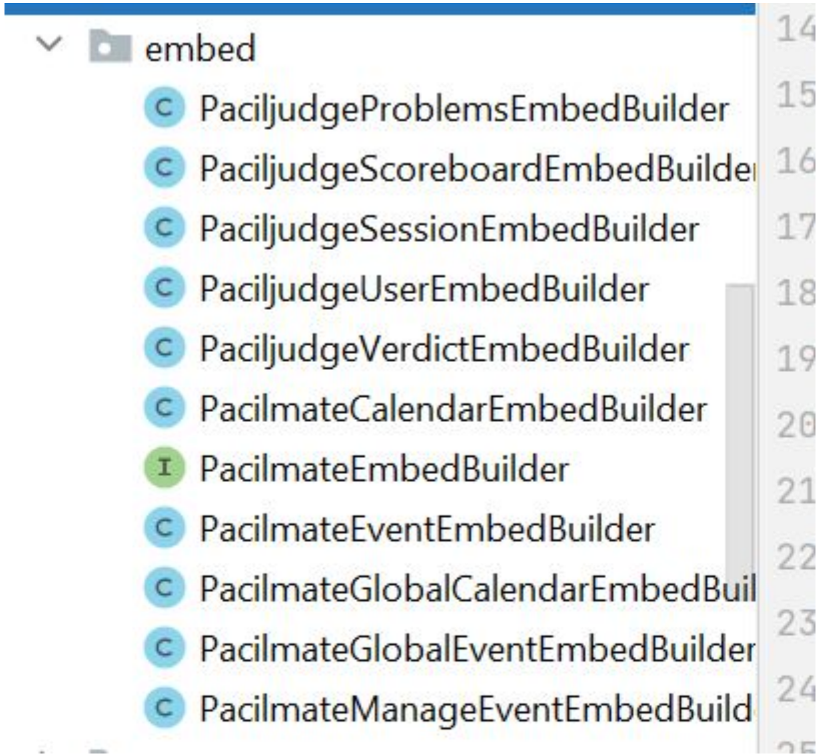
Sebuah Listener pada dasarnya akan dipanggil berulang-ulang kali oleh JDA. Sebelum memanggil service, **Mediator** akan dipanggil terlebih dahulu.

Hal ini memenuhi Single Responsibility Principle.

Penerapan Design Pattern di Code

- **Builder Pattern,** proses pembuatan message dan embed discord. 🏭
- **Iterator Pattern,** Listener ada banyak dan akan dilakukan iterasi untuk setiap listenernya. Proses ini sudah ada interface JDA-nya. ➰
- **Observer Pattern,** untuk mengabarkan notifikasi kepada setiap subscribernya. 🕶
- **Proxy Pattern,** sebuah Judge pada dasarnya akan memberikan banyak sekali informasi, butuh suatu proxy antara Judge dan Pengguna, melalui **PacilMate**.

Penerapan Design Pattern di Code

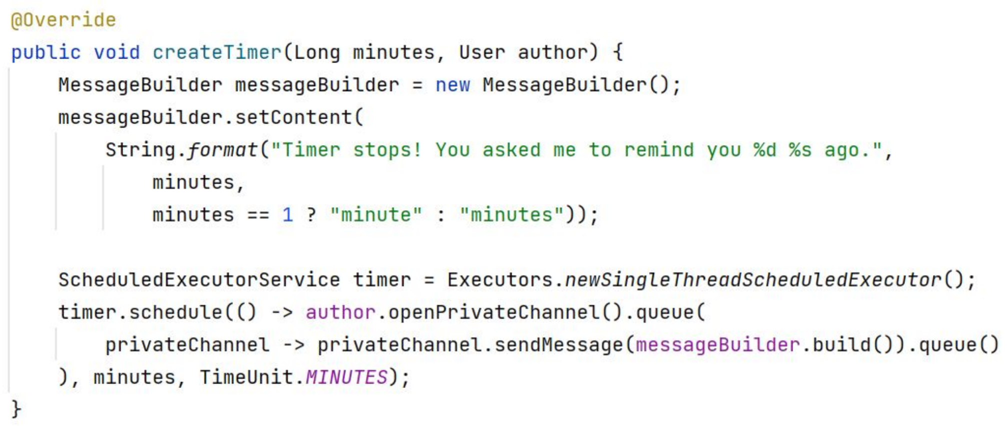


embed
- PaciljudgeProblemsEmbedBuilder
- PaciljudgeScoreboardEmbedBuilde
- PaciljudgeSessionEmbedBuilder
- PaciljudgeUserEmbedBuilder
- PaciljudgeVerdictEmbedBuilder
- PacilmateCalendarEmbedBuilder
- PacilmateEmbedBuilder
- PacilmateEventEmbedBuilder
- PacilmateGlobalCalendarEmbedBuil
- PacilmateGlobalEventEmbedBuilder
- PacilmateManageEventEmbedBuild

Di dalam message ada beberapa embed yang di dalamnya bisa diisi dengan berbagai informasi.

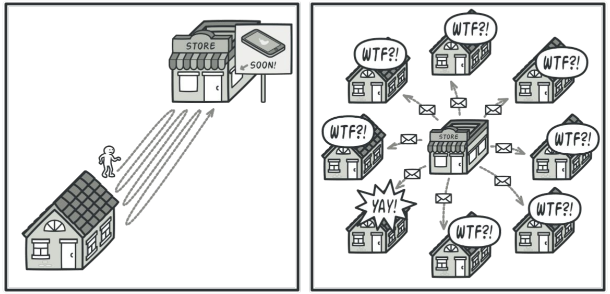
Disini cocok digunakan **Builder Pattern**. 🔨

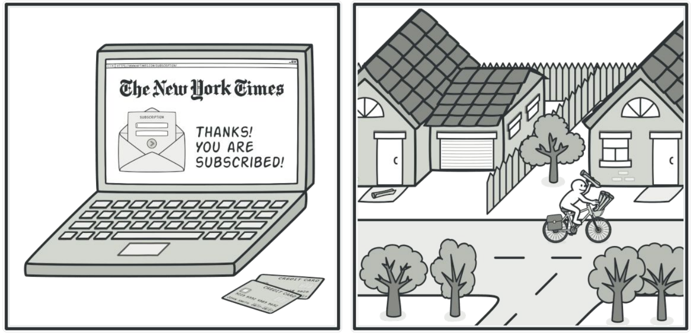
Penerapan Design Pattern di Code



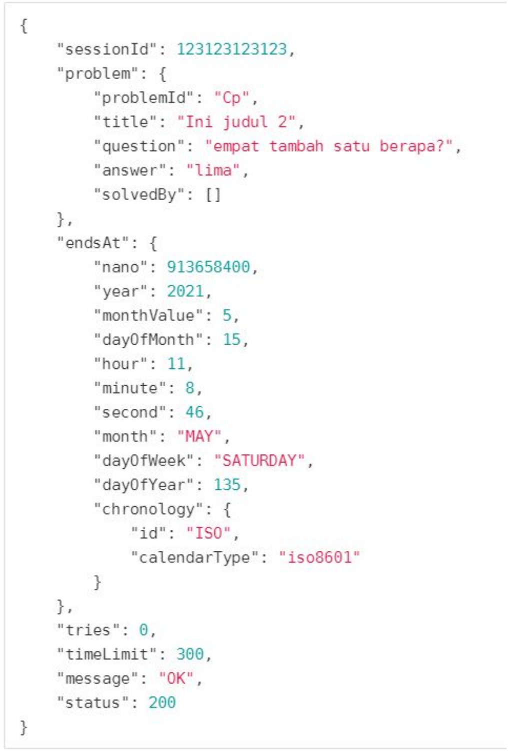
```java
@Override
public void createTimer(Long minutes, User author) {
    MessageBuilder messageBuilder = new MessageBuilder();
    messageBuilder.setContent(
        String.format("Timer stops! You asked me to remind you %d %s ago.",
            minutes,
            minutes == 1 ? "minute" : "minutes"));

    ScheduledExecutorService timer = Executors.newSingleThreadScheduledExecutor();
    timer.schedule(() -> author.openPrivateChannel().queue(
        privateChannel -> privateChannel.sendMessage(messageBuilder.build()).queue()
    ), minutes, TimeUnit.MINUTES);
}
```

Ada notifier yang mengirim pesan sesuai jadwal, disini sangat berguna **Observer Pattern**!



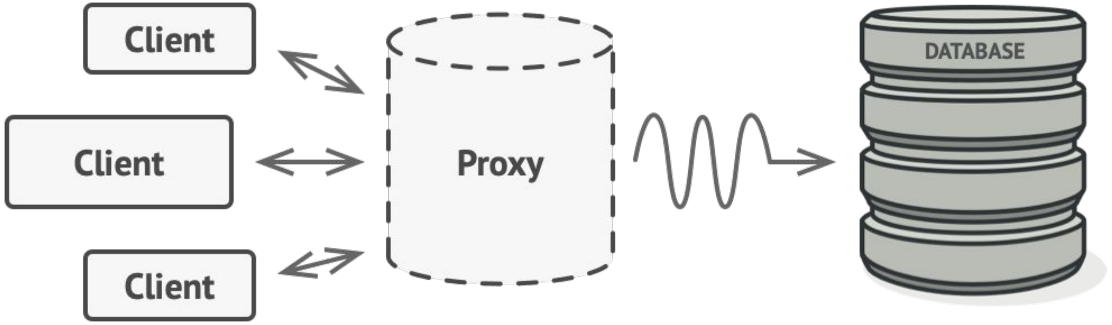


Penerapan Design Pattern di Code



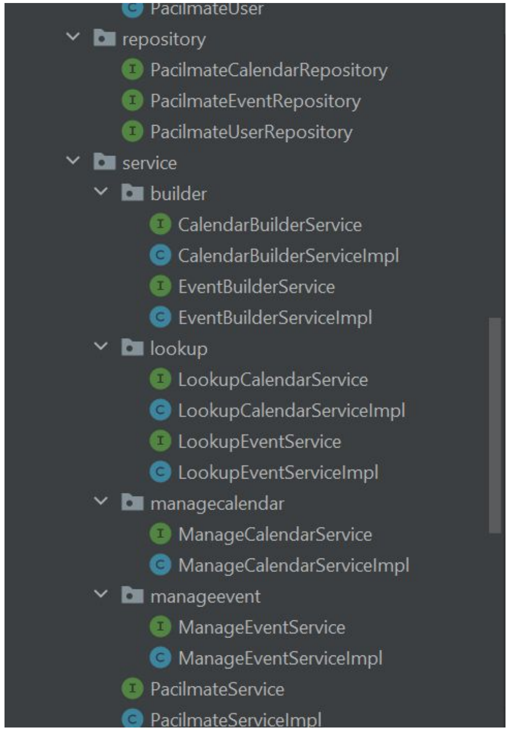
```
{
    "sessionId": 123123123123,
    "problem": {
        "problemId": "Cp",
        "title": "Ini judul 2",
        "question": "empat tambah satu berapa?",
        "answer": "lima",
        "solvedBy": []
    },
    "endsAt": {
        "nano": 913658400,
        "year": 2021,
        "monthValue": 5,
        "dayOfMonth": 15,
        "hour": 11,
        "minute": 8,
        "second": 46,
        "month": "MAY",
        "dayOfWeek": "SATURDAY",
        "dayOfYear": 135,
        "chronology": {
            "id": "ISO",
            "calendarType": "iso8601"
        }
    },
    "tries": 0,
    "timeLimit": 300,
    "message": "OK",
    "status": 200
}
```

PacilJudge merupakan Microservices, API nya exposes terlalu banyak informasi, juga bisa down. Disini **Proxy Pattern** pada PacilMate sangat bergunaa! 😁🌎



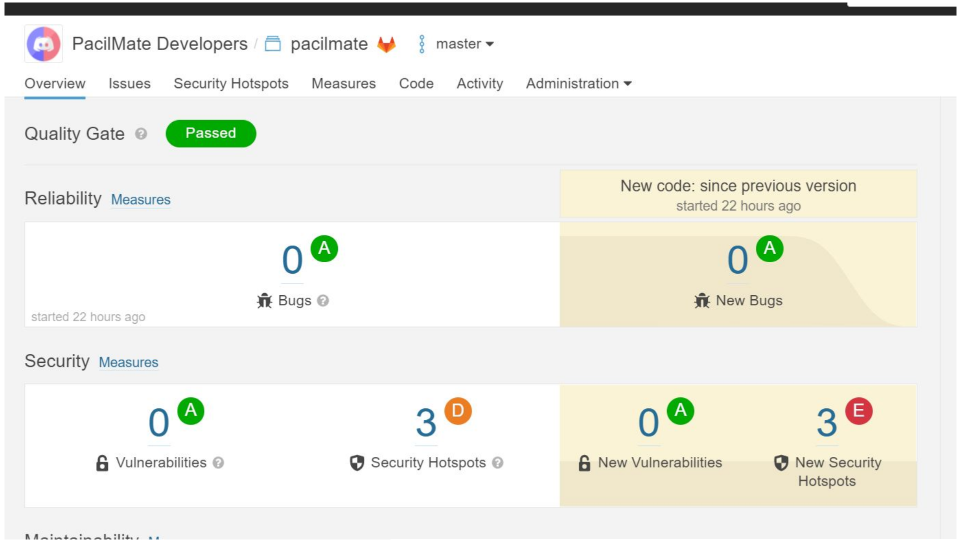
Penerapan Clean Code

Interface first, No Code Duplicate



```
    PacilmateUser
  ⌄ repository
      PacilmateCalendarRepository
      PacilmateEventRepository
      PacilmateUserRepository
  ⌄ service
    ⌄ builder
        CalendarBuilderService
        CalendarBuilderServiceImpl
        EventBuilderService
        EventBuilderServiceImpl
    ⌄ lookup
        LookupCalendarService
        LookupCalendarServiceImpl
        LookupEventService
        LookupEventServiceImpl
    ⌄ managecalendar
        ManageCalendarService
        ManageCalendarServiceImpl
    ⌄ manageevent
        ManageEventService
        ManageEventServiceImpl
      PacilmateService
      PacilmateServiceImpl
```

SonarQube! Detects code smell, and Progresses of our codes!



PacilMate Developers / pacilmate master

Overview Issues Security Hotspots Measures Code Activity Administration

Quality Gate Passed

Reliability Measures

New code: since previous version
started 22 hours ago

0 A Bugs	0 A New Bugs

started 22 hours ago

Security Measures

0 A Vulnerabilities	3 D Security Hotspots	0 A New Vulnerabilities	3 E New Security Hotspots

sonarqube

Penerapan Clean Code

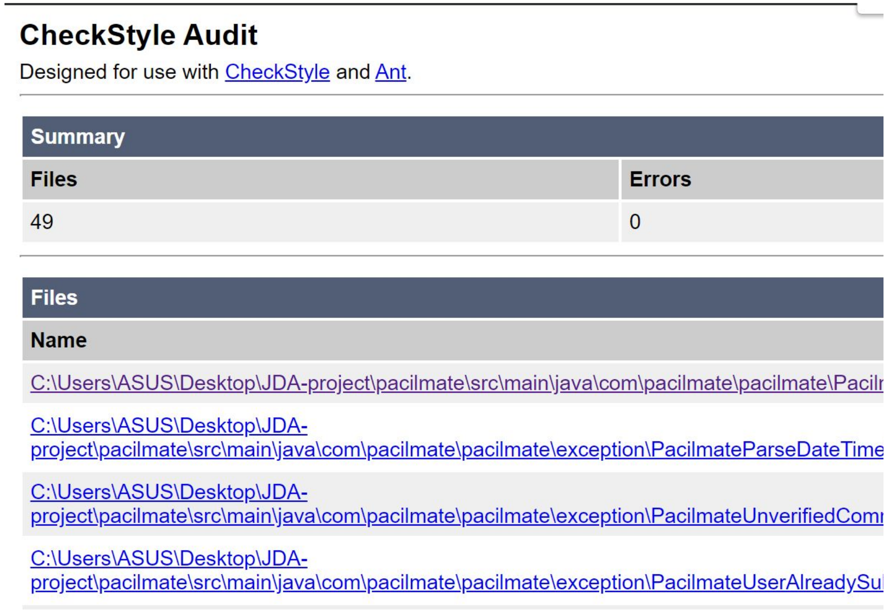
Checkstyle, Naming convention, Standardized Codes

(Camel case for methodsName and attributesName)

(Lower for packagename)

(Pascal case for ClassName) 😊

checkstyle



CheckStyle Audit

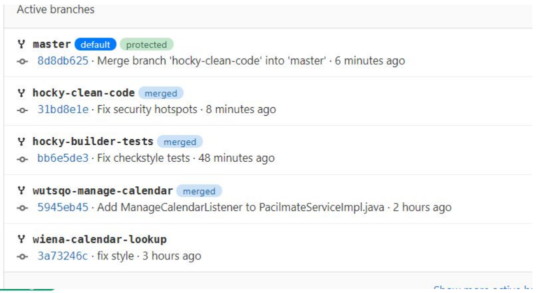
Designed for use with CheckStyle and Ant.

Summary	
Files	**Errors**
49	0

Files
Name
C:\Users\ASUS\Desktop\JDA-project\pacilmate\src\main\java\com\pacilmate\pacilmate\Pacil
C:\Users\ASUS\Desktop\JDA-project\pacilmate\src\main\java\com\pacilmate\pacilmate\exception\PacilmateParseDateTime
C:\Users\ASUS\Desktop\JDA-project\pacilmate\src\main\java\com\pacilmate\pacilmate\exception\PacilmateUnverifiedCom
C:\Users\ASUS\Desktop\JDA-project\pacilmate\src\main\java\com\pacilmate\pacilmate\exception\PacilmateUserAlreadySu

Penerapan Clean Code

Gitlab Branches, Issues, Cross Member Review, and Merge Requests



Active branches

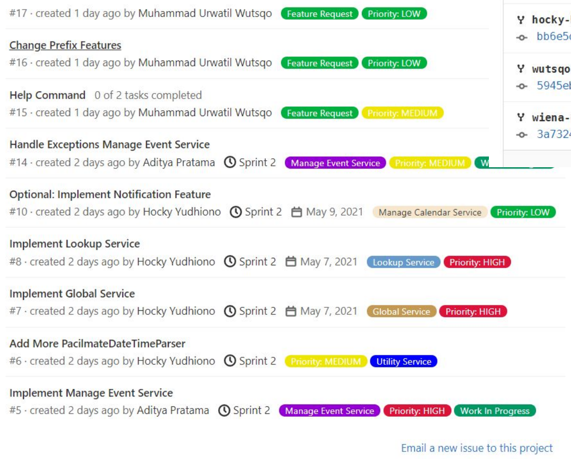
master default protected
8d8db625 · Merge branch 'hocky-clean-code' into 'master' · 6 minutes ago

hocky-clean-code merged
31bd8e1e · Fix security hotspots · 8 minutes ago

hocky-builder-tests merged
bb6e5de3 · Fix checkstyle tests · 48 minutes ago

wutsqo-manage-calendar merged
5945eb45 · Add ManageCalendarListener to PacilmateServiceImpl.java · 2 hours ago

wiena-calendar-lookup
3a73246c · fix style · 3 hours ago



#17 · created 1 day ago by Muhammad Urwatil Wutsqo Feature Request Priority: LOW

Change Prefix Features
#16 · created 1 day ago by Muhammad Urwatil Wutsqo Feature Request Priority: LOW

Help Command 0 of 2 tasks completed
#15 · created 1 day ago by Muhammad Urwatil Wutsqo Feature Request Priority: MEDIUM

Handle Exceptions Manage Event Service
#14 · created 2 days ago by Aditya Pratama Sprint 2 Manage Event Service Priority: MEDIUM W

Optional: Implement Notification Feature
#10 · created 2 days ago by Hocky Yudhiono Sprint 2 May 9, 2021 Manage Calendar Service Priority: LOW

Implement Lookup Service
#8 · created 2 days ago by Hocky Yudhiono Sprint 2 May 7, 2021 Lookup Service Priority: HIGH

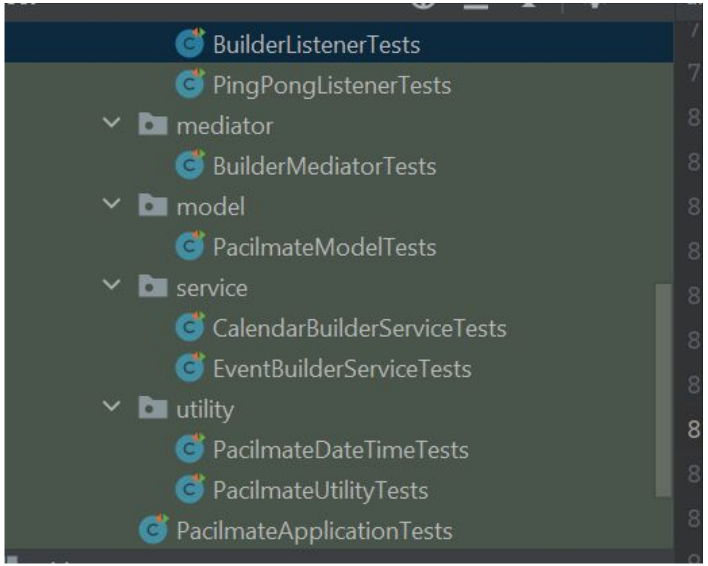
Implement Global Service
#7 · created 2 days ago by Hocky Yudhiono Sprint 2 May 7, 2021 Global Service Priority: HIGH

Add More PacilmateDateTimeParser
#6 · created 2 days ago by Hocky Yudhiono Sprint 2 Priority: MEDIUM Utility Service

Implement Manage Event Service
#5 · created 2 days ago by Aditya Pratama Sprint 2 Manage Event Service Priority: HIGH Work In Progress

Email a new issue to this project

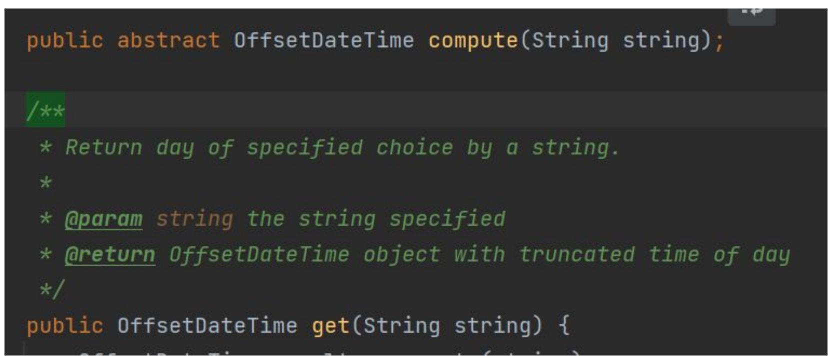
Semi-TDD, Interface → Live Tests → Implementation → Tests



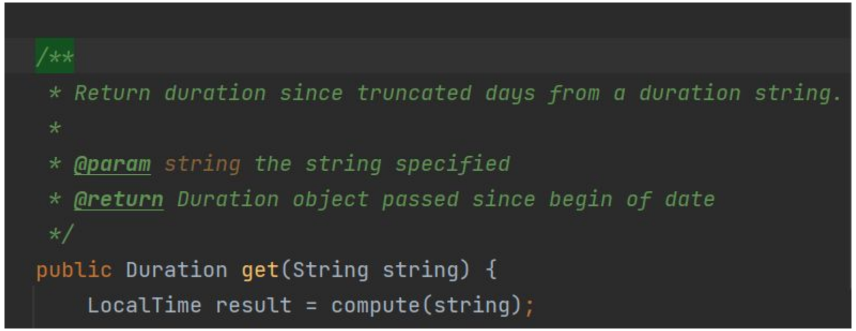
```
        BuilderListenerTests
        PingPongListenerTests
    mediator
        BuilderMediatorTests
    model
        PacilmateModelTests
    service
        CalendarBuilderServiceTests
        EventBuilderServiceTests
    utility
        PacilmateDateTimeTests
        PacilmateUtilityTests
    PacilmateApplicationTests
```

Penerapan Clean Code

Maximize Java Documentation For Public methods and Gitlab Wiki For Installing Guide



```java
public abstract OffsetDateTime compute(String string);

/**
 * Return day of specified choice by a string.
 *
 * @param string the string specified
 * @return OffsetDateTime object with truncated time of day
 */
public OffsetDateTime get(String string) {
```

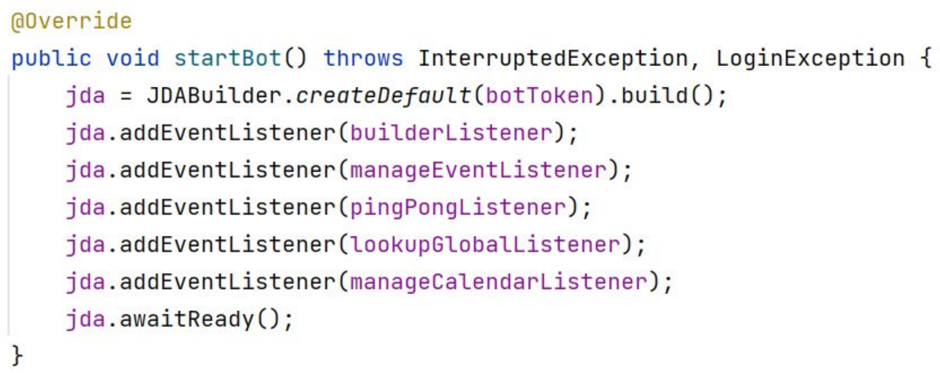


```java
/**
 * Return duration since truncated days from a duration string.
 *
 * @param string the string specified
 * @return Duration object passed since begin of date
 */
public Duration get(String string) {
    LocalTime result = compute(string);
```

Microservices?

Mengapa menggunakan microservices?

- Scalable 🐘
- Maintainable 🎾
- Testable 📄
- High Cohesion 👪
- Low Coupling 🏃‍♀️
- Centered Around Business Capability 💵

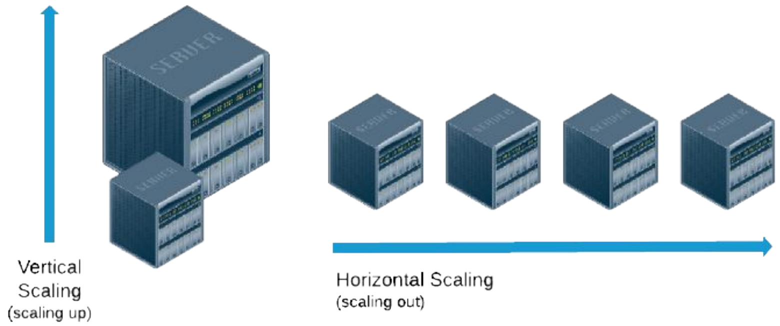


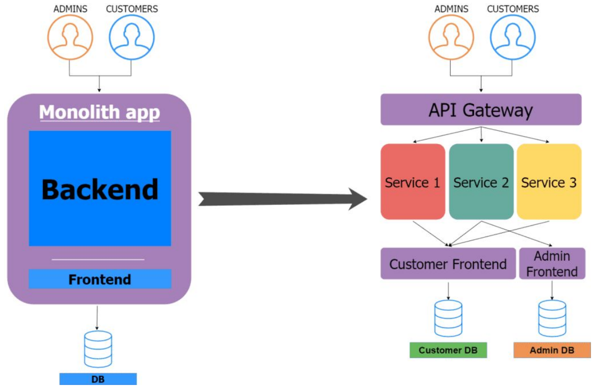
```
@Override
public void startBot() throws InterruptedException, LoginException {
    jda = JDABuilder.createDefault(botToken).build();
    jda.addEventListener(builderListener);
    jda.addEventListener(manageEventListener);
    jda.addEventListener(pingPongListener);
    jda.addEventListener(lookupGlobalListener);
    jda.addEventListener(manageCalendarListener);
    jda.awaitReady();
}
```

Microservices?

Mengapa tidak semua menggunakan microservices?

- Low Response Time **Needed** (We're building a chat bot **here**) 🕒, untuk aplikasi skala kecil seperti ini, malah **menambahkan** waiting time.
- Users are often separated, Horizontal Scaling is more based! 📏
- Chat bot separated by module, bisa deploy banyak instance dengan fitur yang bisa dibuat berbeda! 🤖
- Aplikasi chatbot kami cukup **monolithic**, 🧠 memisahkannya hanya **akan menambahkan workload**, bottlecap di komunikasi. Instead lakukan **refactor** *seoptimal* mungkin, setiap aplikasi di dalam aplikasi tersebut dipisahkan berdasarkan service, sehingga masih bisa scaled masing-masing. **Tapi..........**


Vertical Scaling (scaling up)
Horizontal Scaling (scaling out)
SERVER



Penerapan Microservices

This guy has separate DB

He can scale separately

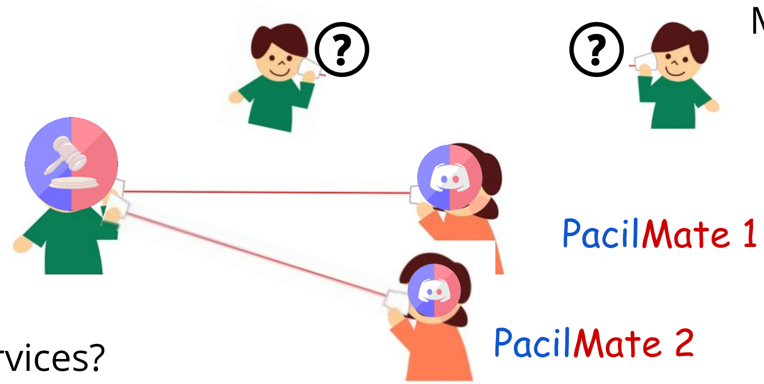
REST API Based

PacilJudge 👩‍⚖️

HE CAN TALK WITH 2 BOTS!

He is Independent 🏆

More microservices in the future(?)

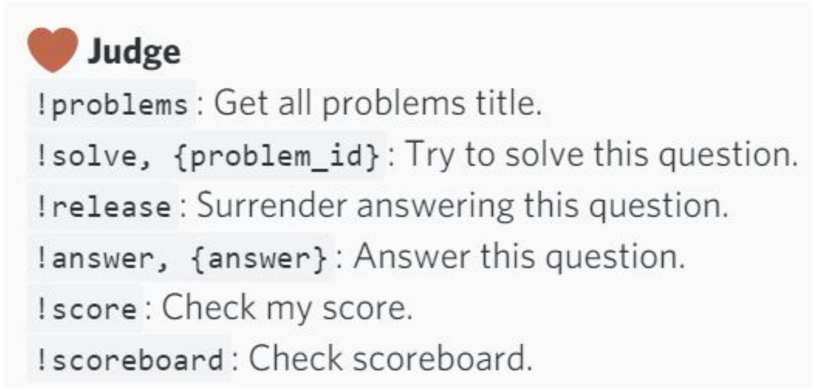


How they communicate?
Web client, asynchronously, statelessly!

No bottlecap!

Mengapa dibuat microservices?

- ⛳ **Independen**, tidak ada hubungannya dengan kalender (**aplikasi utama**).
- Menilai **bisa membutuhkan waktu** 🕒, waktu fitur utama untuk melayani request lain bisa terganggu 😡.
- Yang paling penting:
  Kalau user mengirim pesan yang berbahaya 😈, **pacilmate** tidak down 📉
  😭😭😭



❤️ **Judge**

`!problems` : Get all problems title.

`!solve, {problem_id}` : Try to solve this question.

`!release` : Surrender answering this question.

`!answer, {answer}` : Answer this question.

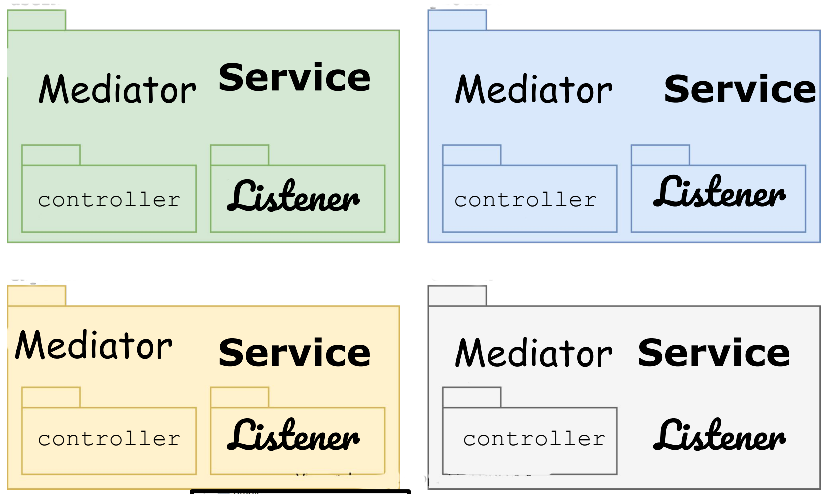
`!score` : Check my score.

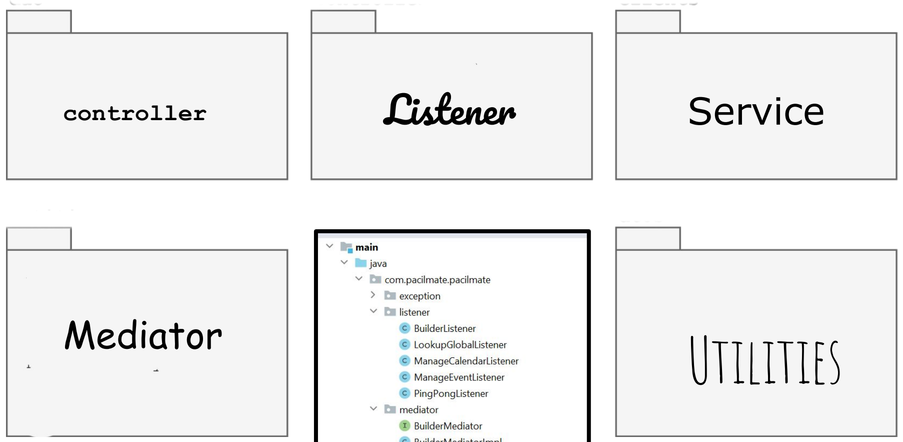
`!scoreboard` : Check scoreboard.

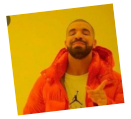
Penerapan Microservices - Pacilmate Inside 🕸️🕸️🕸️🕸️!

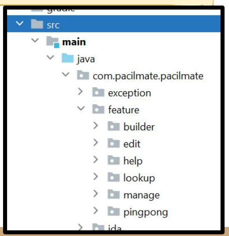
Package by Feature (Services) vs Package by Layer

Not really “microservices”, but it’s less **monolith** and can be scaled!



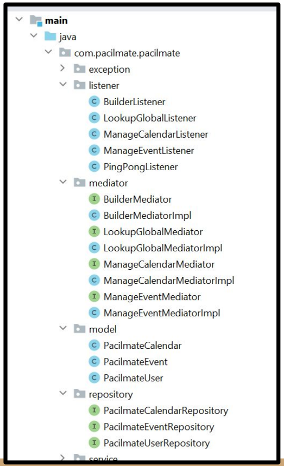






```
src
  main
    java
      com.pacilmate.pacilmate
        exception
        feature
          builder
          edit
          help
          lookup
          manage
          pingpong
        ida
```

✔✔✔

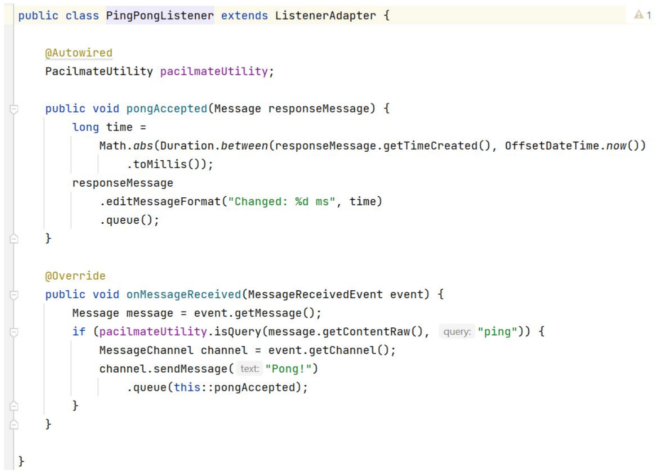


```
main
  java
    com.pacilmate.pacilmate
      exception
      listener
        BuilderListener
        LookupGlobalListener
        ManageCalendarListener
        ManageEventListener
        PingPongListener
      mediator
        BuilderMediator
        BuilderMediatorImpl
        LookupGlobalMediator
        LookupGlobalMediatorImpl
        ManageCalendarMediator
        ManageCalendarMediatorImpl
        ManageEventMediator
        ManageEventMediatorImpl
      model
        PacilmateCalendar
        PacilmateEvent
        PacilmateUser
      repository
        PacilmateCalendarRepository
        PacilmateEventRepository
        PacilmateUserRepository
      service
```

❌❌❌

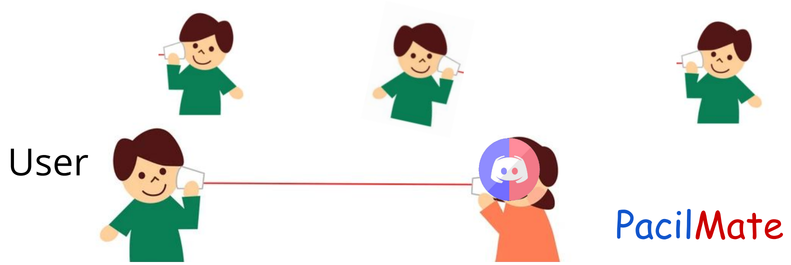
Penerapan Asynchronous Programming

Asynchronous programming bisa diterapkan dalam bentuk Async Spring! Chat Bot juga di dalamnya sudah banyak Rest yang sudah diintegrate oleh message queue JDA! Contoh: Fitur Ping!



```java
public class PingPongListener extends ListenerAdapter {

    @Autowired
    PacilmateUtility pacilmateUtility;

    public void pongAccepted(Message responseMessage) {
        long time =
            Math.abs(Duration.between(responseMessage.getTimeCreated(), OffsetDateTime.now())
                .toMillis());
        responseMessage
            .editMessageFormat("Changed: %d ms", time)
            .queue();
    }

    @Override
    public void onMessageReceived(MessageReceivedEvent event) {
        Message message = event.getMessage();
        if (pacilmateUtility.isQuery(message.getContentRaw(),  query: "ping")) {
            MessageChannel channel = event.getChannel();
            channel.sendMessage( text: "Pong!")
                .queue(this::pongAccepted);
        }
    }

}
```

PacilMate will Notify your events!



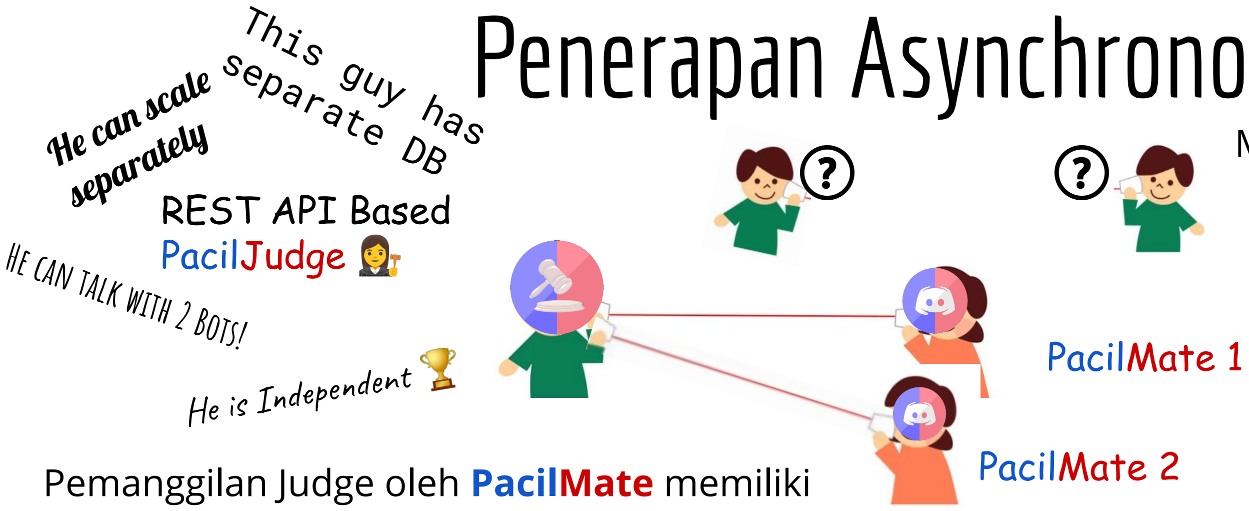
PacilMate will run a thread that will check the latest event coming, and will notify user that subscribed to this!

Notification?

Observer Pattern!

Penerapan Asynchronous Programming

More microservices in the future(?)



How they communicate?
Web client, asynchronously, statelessly!

Proxy Pattern to the rescue !

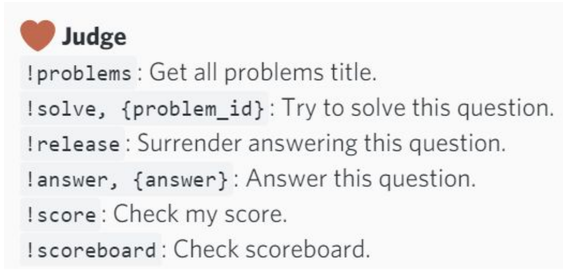
No bottlecap!

Pemanggilan Judge oleh **PacilMate** memiliki satu permasalahan, yaitu lama jika webclient tidak kunjung mendapatkan response.

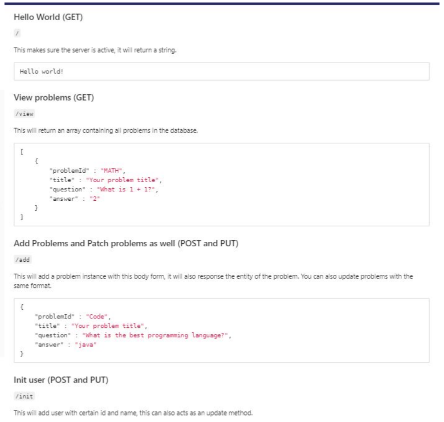
Bagaimana cara mempercepat hal ini?

Asynchronous Services

Mesti dilakukan pemanggilan secara asynchronous agar setiap fitur tidak terganggu oleh karena microservice ini!


Judge

`!problems` : Get all problems title.
`!solve, {problem_id}` : Try to solve this question.
`!release` : Surrender answering this question.
`!answer, {answer}` : Answer this question.
`!score` : Check my score.
`!scoreboard` : Check scoreboard.


Hello World (GET)

`/`

This makes sure the server is active, it will return a string.

```
Hello world!
```

View problems (GET)

`/view`

This will return an array containing all problems in the database.

```
[
    {
        "problemId" : "MATH",
        "title" : "Your problem title",
        "question" : "What is 1 + 1?",
        "answer" : "2"
    }
]
```

Add Problems and Patch problems as well (POST and PUT)

`/add`

This will add a problem instance with this body form, it will also response the entity of the problem. You can also update problems with the same format.

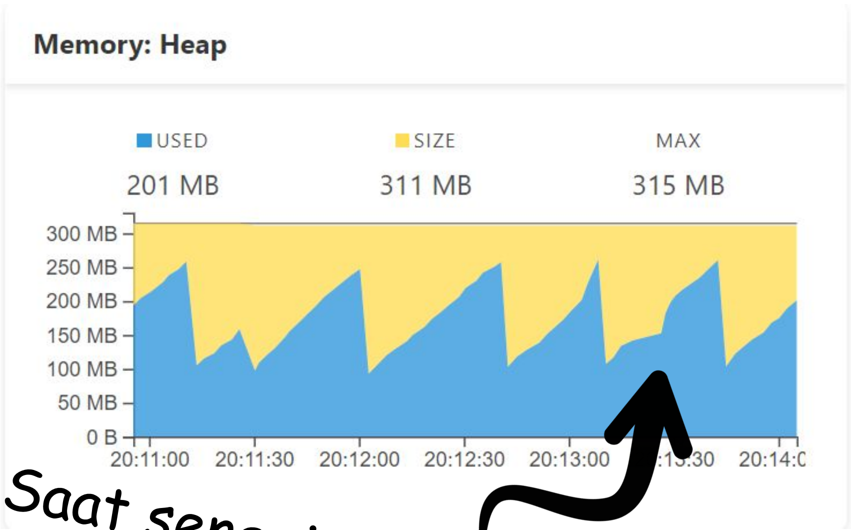
```
{
    "problemId" : "Code",
    "title" : "Your problem title",
    "question" : "What is the best programming language?",
    "answer" : "java"
}
```

Init user (POST and PUT)

`/init`

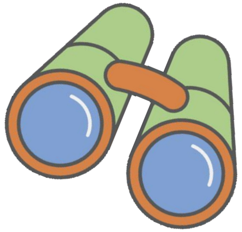
This will add user with certain id and name, this can also acts as an update method.

Penerapan Profiling

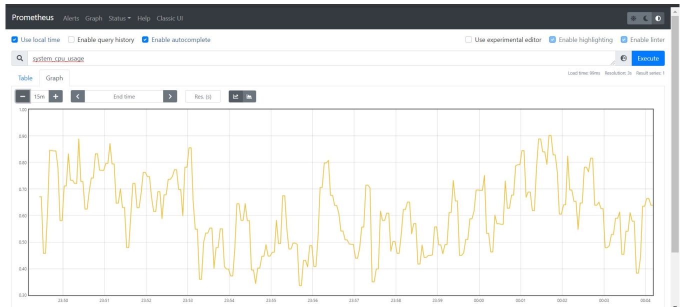


Saat sengaja dibuat spike request!

Spring Boot Actuators + Spring Boot Admin + Micrometer + Prometheus + Grafana

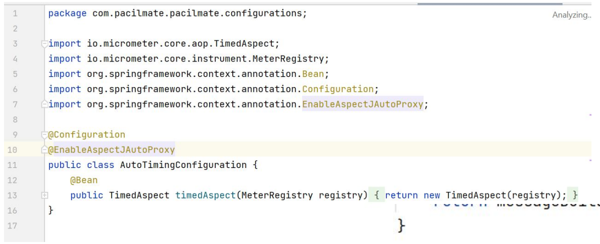


Buat memonitor memori terpakai, memori bebas, disk usage, threads, uptime, environment,

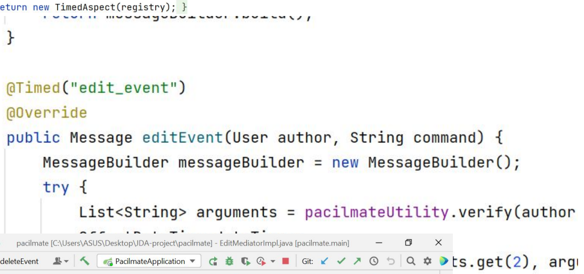


system_cpu_usage

Penerapan Profiling

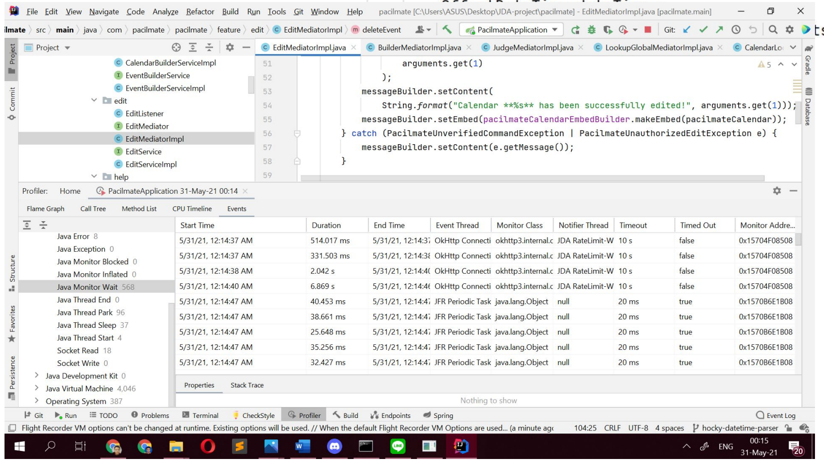


```java
package com.pacilmate.pacilmate.configurations;

import io.micrometer.core.aop.TimedAspect;
import io.micrometer.core.instrument.MeterRegistry;
import org.springframework.context.annotation.Bean;
import org.springframework.context.annotation.Configuration;
import org.springframework.context.annotation.EnableAspectJAutoProxy;

@Configuration
@EnableAspectJAutoProxy
public class AutoTimingConfiguration {
    @Bean
    public TimedAspect timedAspect(MeterRegistry registry) { return new TimedAspect(registry); }
}
```



```java
}

@Timed("edit_event")
@Override
public Message editEvent(User author, String command) {
    MessageBuilder messageBuilder = new MessageBuilder();
    try {
        List<String> arguments = pacilmateUtility.verify(author
```

```java
ts.get(2), arg
```



```java
                arguments.get(1)
            );
        messageBuilder.setContent(
            String.format("Calendar **%s** has been successfully edited!", arguments.get(1)));
        messageBuilder.setEmbed(pacilmateCalendarEmbedBuilder.makeEmbed(pacilmateCalendar));
    } catch (PacilmateUnverifiedCommandException | PacilmateUnauthorizedEditException e) {
        messageBuilder.setContent(e.getMessage());
    }
```

Start Time	Duration	End Time	Event Thread	Monitor Class	Notifier Thread	Timeout	Timed Out	Monitor Addre...
5/31/21, 12:14:37 AM	514.017 ms	5/31/21, 12:14:37	OkHttp Connecti	okhttp3.internal.c	JDA RateLimit-W	10 s	false	0x15704F08508
5/31/21, 12:14:37 AM	331.503 ms	5/31/21, 12:14:38	OkHttp Connecti	okhttp3.internal.c	JDA RateLimit-W	10 s	false	0x15704F08508
5/31/21, 12:14:38 AM	2.042 s	5/31/21, 12:14:40	OkHttp Connecti	okhttp3.internal.c	JDA RateLimit-W	10 s	false	0x15704F08508
5/31/21, 12:14:40 AM	6.869 s	5/31/21, 12:14:46	OkHttp Connecti	okhttp3.internal.c	JDA RateLimit-W	10 s	false	0x15704F08508
5/31/21, 12:14:47 AM	40.453 ms	5/31/21, 12:14:47	JFR Periodic Task	java.lang.Object	null	20 ms	true	0x1570B6E1B08
5/31/21, 12:14:47 AM	38.661 ms	5/31/21, 12:14:47	JFR Periodic Task	java.lang.Object	null	20 ms	true	0x1570B6E1B08
5/31/21, 12:14:47 AM	25.648 ms	5/31/21, 12:14:47	JFR Periodic Task	java.lang.Object	null	20 ms	true	0x1570B6E1B08
5/31/21, 12:14:47 AM	35.256 ms	5/31/21, 12:14:47	JFR Periodic Task	java.lang.Object	null	20 ms	true	0x1570B6E1B08
5/31/21, 12:14:47 AM	32.427 ms	5/31/21, 12:14:47	JFR Periodic Task	java.lang.Object	null	20 ms	true	0x1570B6E1B08

Flight Recorder VM options can't be changed at runtime. Existing options will be used. // When the default Flight Recorder VM Options are used... (a minute ago

Bisa mengukur **waktu** 🕒 berjalannya suatu query 📚 dengan annotation `@Timed`, menghitung ✋ berapa kali query tersebut dipanggil 📞 dan diakses.

Melalui **Prometheus**, dengan endpoint yang disediakan **actuators**! Di `localhost` bisa menggunakan **SLF4J** dan **Intellij Profiler**!

Terima Kasih