

**TUGAS BESAR 3**

**PENERAPAN STRING MATCHING DAN REGULAR EXPRESSION**  
**DALAM PEMBANGUNAN DEADLINE REMINDER ASSISTANT**

**IF2211**

**STRATEGI ALGORITMA**

**SEMESTER 2**

**Kelompok 48 234bot**

**Oleh**

**Arsa Daris Gintara                      13519037**

**Dwianditya Hanif Raharjanto        13519046**

**Fabian Savero Diaz Pranoto          13519140**



**PROGRAM STUDI TEKNIK INFORMATIKA**

**SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA**

**INSTITUT TEKNOLOGI BANDUNG**

**2020/2021**

# Daftar Isi

<b>Daftar Isi</b>	<b>2</b>
<b>BAB I</b>	
<b>Deskripsi Tugas</b>	<b>4</b>
<b>BAB II</b>	
<b>Landasan Teori</b>	<b>5</b>
2.1 Dasar Teori	5
2.1.1 Algoritma KMP	5
2.2 Chatbot	5
<b>BAB III</b>	
<b>Analisis dan Pemecahan Masalah</b>	<b>6</b>
3.1 Langkah Penyelesaian Masalah Tiap Fitur	6
3.1.1 Fitur Menambahkan Task Baru	6
3.1.2 Fitur Melihat Daftar Task yang Harus Dikerjakan	6
3.1.3 Fitur Menampilkan Deadline dari Suatu Task	7
3.1.4 Fitur Mengupdate Deadline Tugas	7
3.1.5 Fitur Menandai Task Sudah Dikerjakan	7
3.1.6 Menampilkan Opsi Help	7
3.2 Fitur Fungsional dan Arsitektur Chatbot yang Dibangun	7
<b>BAB IV</b>	
<b>Implementasi dan Pengujian</b>	<b>9</b>
4.1 Spesifikasi Teknis Program	9
4.2 Tata Cara Penggunaan Program	10
4.3 Hasil Pengujian	11
4.4 Analisis Hasil Pengujian	20
5.1 Kesimpulan	21
5.2 Saran	21
<b>Daftar Pustaka</b>	<b>22</b>

## **BAB I**

### **Deskripsi Tugas**

Dalam tugas besar ini, Kami membangun sebuah *chatbot* sederhana yang berfungsi untuk membantu mengingat berbagai deadline, tanggal penting, dan task-task tertentu kepada pengguna yang menggunakannya. *Chatbot* sederhana ini memanfaatkan algoritma String Matching dan Regular Expression untuk menyerupai layaknya Google Assistant yang akan menjawab segala pertanyaan mengenai informasi akan deadline maupun task-task yang ada.

Fitur yang tersedia dalam *Chatbot* kami ini antara lain adalah menambahkan task baru, melihat task yang tersedia, menampilkan deadline suatu task, menandai task sudah terlaksana, memperbaharui task, menampilkan opsi help mengenai fasilitas apa saja yang bisa dilakukan oleh assistant, pendefinisian suatu list kata penting untuk menentukan apakah kata tersebut merupakan suatu task atau tidak, menampilkan pesan error ketika assistant tidak paham masukan pengguna, dan (Bonus) assistant merekomendasikan suatu kata jika terdapat kesalahan kata pada perintah dari pengguna.

## **BAB II**

### **Landasan Teori**

#### **2.1 Dasar Teori**

##### **2.1.1 Algoritma KMP**

Algoritma Knuth-Morris-Pratt atau lebih dikenal KMP merupakan algoritma untuk mencari suatu pola pada suatu teks dari kiri ke kanan. Terkesan mirip dengan algoritma pencarian brute force tetapi, mencari suatu pola nya lebih mangkus daripada algoritma brute force.

Semisal terjadi ketidaksesuaian antara teks dan pola P pada P[indeks] seperti contoh teks  $T[i] \neq P[j]$  cara untuk menghindari pola perbandingan yang boros adalah dengan cara prefiks terbesar dari  $P[0..i-1]$  yang menjadi sufiks dari  $P[1..j-1]$ .

Algoritma Knuth-Morris-Pratt memiliki fungsi bantuan *Border-Function* yang bertujuan untuk mempreproses suatu pola untuk mencari prefiks dari suatu pola yang sesuai dengan pola itu sendiri.

##### **2.1.2 Algoritma BM**

Algoritma Boyer-Moore merupakan algoritma untuk mencari pola pada suatu teks dengan dua teknik.

Teknik yang pertama adalah teknik *looking-glass* yang berarti mencari suatu pola P pada teks T dengan cara mencarinya secara terbalik berdasarkan pola P, dengan awalan di akhiran P.

Teknik yang kedua adalah teknik *character-jump* ketika terjadi ketidaksesuaian pada  $T[i] \neq x$  maka karakter pada pola  $P[j]$  tidak sama dengan  $T[i]$ .

Algoritma Boyer-Moore memiliki fungsi bantuan yang biasa disebut dengan fungsi *Last Occurrence* yang bertujuan untuk mempreproses pola P dan alfabet A untuk membangun suatu penanda kemunculan terakhir pada fungsi  $L()$ .

##### **2.1.3 Algoritma Regex**

Regular Expression atau Regex ini merupakan suatu metode untuk mencari suatu pola dalam sebuah string. Regex juga menggambarkan string dan jumlah string dalam bentuk logis umum untuk mencari, mengganti, memanipulasi, atau memproses lebih lanjut dalam dokumen, kode, ataupun database.

#### **2.2 Chatbot**

Chatbot yang kami rancang merupakan chatbot berbasis web. Kami menggunakan bahasa pemrograman PHP serta menggunakan Framework Laravel dalam merancang web chatbot ini.

## **BAB III**

### **Analisis dan Pemecahan Masalah**

#### **3.1 Langkah Penyelesaian Masalah Tiap Fitur**

##### **3.1.1 Fitur Menambahkan Task Baru**

Jadi untuk fitur menambahkan sebuah task baru ini kami pertama-tama mendefinisikan Regular Expression untuk semua komponen berikut yaitu tanggal, kode mata kuliah, jenis tugas, dan topik tugas karena Chatbot dapat mengidentifikasi pengguna ingin menambahkan suatu fitur dengan cara memeriksa apakah seluruh komponen yang disebutkan ada dalam perintah pengguna.

##### **3.1.2 Fitur Melihat Daftar Task yang Harus Dikerjakan**

Untuk fitur kali ini kami membagi beberapa perintah yang dapat diterima oleh Chatbot. Pertama adalah perintah untuk memberikan informasi seluruh deadline yang dimiliki oleh pengguna. Cara chatbot mengidentifikasi pengguna memberi perintah tersebut adalah dengan cara melakukan string matching inputan pengguna dengan kata “deadline” serta kata “sejauh”. Jadi ketika pengguna menginput suatu pesan yang mengandung “deadline” dan “sejauh” chatbot akan memberikan informasi mengenai seluruh deadline yang dimiliki oleh pengguna.

Kemudian perintah untuk menanyakan deadline berdasarkan periode waktu yang dibagi menjadi periode tertentu (dari tanggal sekian sampai sekian), N minggu ke depan, N hari ke depan, dan hari ini. Untuk pengidentifikasian perintah berdasarkan waktu tertentu chatbot akan mengecek apakah inputan memiliki kata “deadline” dan “antara” dengan string matching, kemudian mengecek apakah terdapat Regular Expression tanggal pada inputan pengguna. Ketika memenuhi seluruh syarat maka chatbot akan menampilkan deadline antar tanggal sekian sampai sekian.

Setelah itu untuk menanyakan deadline N minggu dan N hari ke depan chatbot mengidentifikasi inputan pengguna dengan mencari kata “deadline” “depan” “minggu” atau “hari” dengan string matching. Ketika kata-kata tersebut terdapat pada inputan pengguna maka chatbot akan menampilkan deadline untuk N minggu atau N hari ke depan.

Yang terakhir adalah menanyakan deadline pada hari ini. Chatbot mengidentifikasi kata “deadline” dan “hari ini” menggunakan string matching dan akan menampilkan deadline yang tersedia pada hari ini.

Selain fitur di atas, fitur yang berikutnya adalah menanyakan deadline sesuai dengan fitur di atas tetapi dengan jenis tugas yang spesifik. Jadi hanya tinggal menambahkan pengecekan untuk masing-masing fitur apakah ada Regular Expression yang memenuhi pada jenis tugas yang di input oleh pengguna.

##### **3.1.3 Fitur Menampilkan Deadline dari Suatu Task**

Fitur ini sebenarnya hampir mirip dengan fitur sebelumnya. Namun, yang berbeda disini adalah suatu deadline akan ditampilkan berdasarkan suatu jenis tugas tertentu. Cara chatbot mengenali perintah pada fitur ini adalah dengan cara mengecek apakah terdapat kata “deadline” dan “kapan” menggunakan

string matching kemudian mengecek menggunakan Regular Expression yang telah dibuat untuk mengecek kode mata kuliah serta jenis tugas yang diinputkan oleh pengguna. Jadi yang akan diberikan oleh chatbot ketika pengguna memerintahkan chatbot pada fitur ini adalah tanggal deadline dan topik dari jenis tugas dan mata kuliah yang diminta oleh pengguna.

### **3.1.4 Fitur Mengupdate Deadline Tugas**

Fitur ini dapat diidentifikasi perintah dengan cara mengecek apakah inputan pengguna terdapat kata “diundur” atau “dimajukan” menggunakan string matching serta mengecek apakah terdapat Regular Expression berupa tanggal dan membuat Regular Expression baru untuk mendeteksi id dari tugas tersebut. Ketika syarat-syarat tersebut terpenuhi maka data dalam database akan terupdate sesuai dengan yang diinginkan oleh pengguna.

### **3.1.5 Fitur Menandai Task Sudah Dikerjakan**

Fitur ini dapat diidentifikasi ketika inputan terdapat kata “selesai” yang dicek menggunakan string matching dan Regular Expression berupa id dari suatu tugas. Ketika memenuhi syarat di atas maka chatbot akan menghapus suatu tugas tertentu dari database yang ada.

### **3.1.6 Menampilkan Opsi Help**

Fitur ini akan berjalan ketika inputan pengguna terdapat kata “help” yang dicocokkan menggunakan string matching. Fitur ini akan menampilkan seluruh fitur-fitur yang tersedia yang dapat dilakukan oleh chatbot serta seluruh list kata penting yang tersedia pada chatbot.

## **3.2 Fitur Fungsional dan Arsitektur Chatbot yang Dibangun**

Fitur yang kami rancang untuk chatbot 234bot ini adalah :

1. Fitur Register
2. Fitur Login
3. Fitur menambahkan task
4. Fitur melihat task yang tersedia
5. Fitur menampilkan deadline untuk task tertentu
6. Fitur mengupdate deadline dari suatu task tertentu
7. Fitur menandai task selesai dikerjakan
8. Fitur menampilkan opsi help

Arsitektur chatbot yang kami rancang berupa terdapat menu login/register terlebih dahulu. Kemudian setelah berhasil login akan langsung terhubung ke laman chatbot. Laman chatbot yang kami rancang berupa wallchat, bubble teks dari chatbot dan pengguna.



## BAB IV

### Implementasi dan Pengujian

#### 4.1 Spesifikasi Teknis Program

##### 1. Database users

Menampung data dari user yang menggunakan chatbot.

Atribut dari database users adalah sebagai berikut

Users =

(id,name,email,email\_verified\_at,password,two\_factor\_secret,two\_factor\_recovery\_codes,remember\_token,created\_at,updated\_at).

##### 2. Database tasks

Menampung data task user yang menggunakan chatbot.

Atribut dari database tasks adalah sebagai berikut

Tasks =(id,user\_id,deadline,mata\_kuliah,jenis\_task,topik,created\_at,updated\_at).

##### 3. Class TaskController extends Controller

###### a. Static function KMPSearch(\$pat,\$txt)

Berfungsi untuk mencari pencocokan string dengan algoritma Knuth-Morris-Pratt

###### b. Static function computeLPSArray(\$pat,\$M,&\$lps)

Berfungsi untuk menjadi fungsi bantuan *Border-Function* yang bertujuan untuk memproses suatu pola untuk mencari prefiks dari suatu pola yang sesuai dengan pola itu sendiri.

###### c. Function Decide Task

Berfungsi untuk menentukan task apa yang akan dipanggil sesuai dengan inputan pengguna.

###### d. Function Delete Task

Berfungsi untuk menghapus task. Fungsi ini digunakan ketika task sudah selesai dikerjakan.

###### e. Function Update Task

Berfungsi untuk mengupdate task. Fungsi ini digunakan ketika ingin mengubah deadline suatu task.

###### f. Function Add Task

Berfungsi untuk menambahkan suatu task.

###### g. Function showDITask

Berfungsi untuk menampung data-data yang akan ditampilkan. Data-data yang ditampilkan adalah atribut deadline dan topik pada database tasks berdasarkan user\_id pengguna, jenis\_task yang diinginkan dan mata\_kuliah tertentu.

###### h. Function getDI

Berfungsi untuk menampung data-data yang akan ditampilkan. Data-data yang ditampilkan adalah seluruh atribut tasks berdasarkan user\_id pengguna atau jika pengguna ingin mendapatkan data yang lebih spesifik berdasarkan jenis\_task.

###### i. Function getDIBetweenDates



Berfungsi untuk menampung data-data yang akan ditampilkan. Data-data yang ditampilkan adalah seluruh atribut tasks berdasarkan user\_id pengguna, tanggal mulai dan tanggal akhir. Jika pengguna ingin mengambil data yang lebih spesifik berdasarkan jenis\_task juga bisa.

## **4.2 Tata Cara Penggunaan Program**

Clone repository dan masukkan command sebagai berikut:

```
cd src/234bot
```

```
composer install
```

Salin file .env dan masukkan kredensial database sql beserta generate app key dan melakukan migrasi database melalui command di bawah:

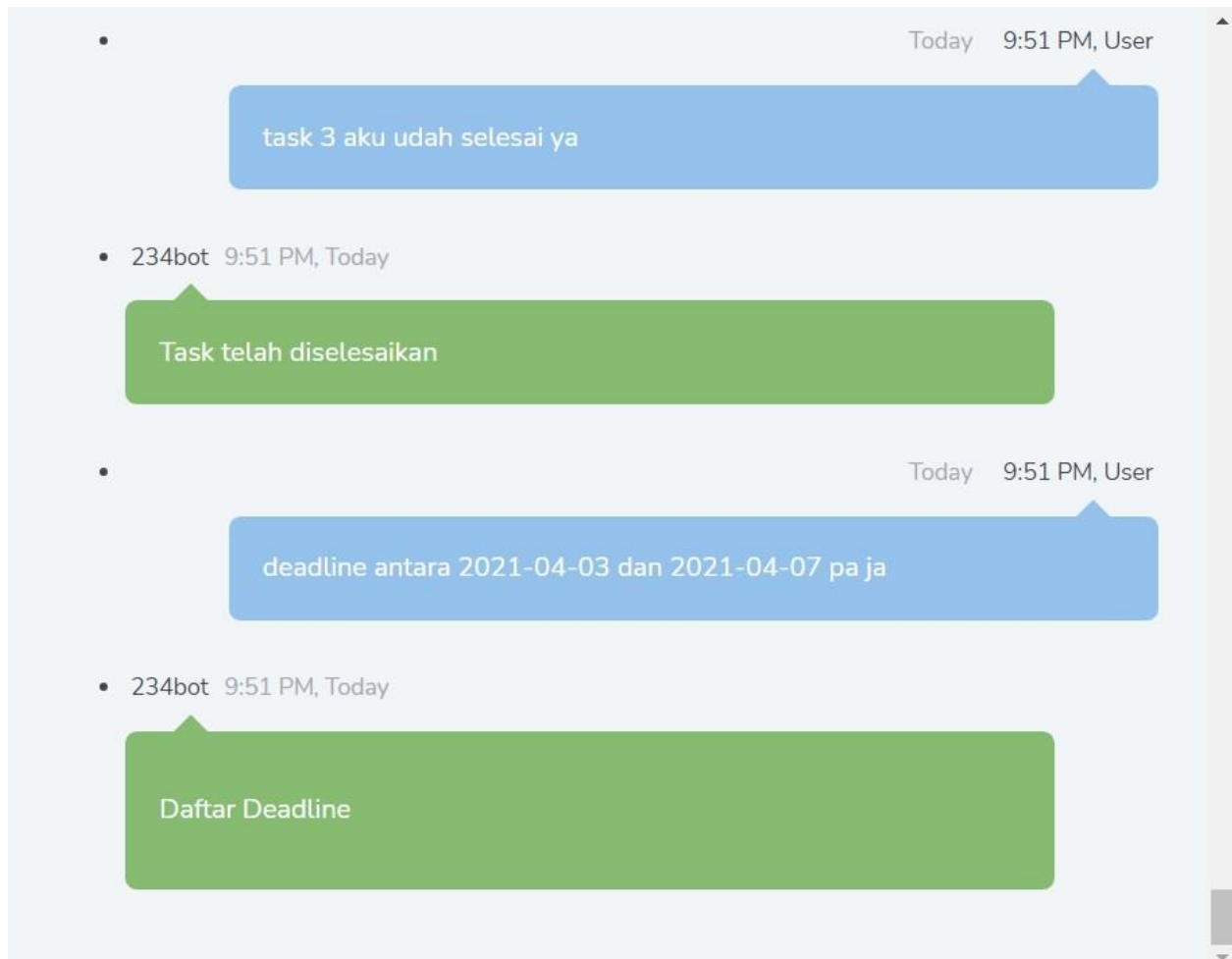
```
php artisan key:generate
```

```
php artisan migrate
```

Jalankan aplikasi melalui command

```
php artisan serve
```

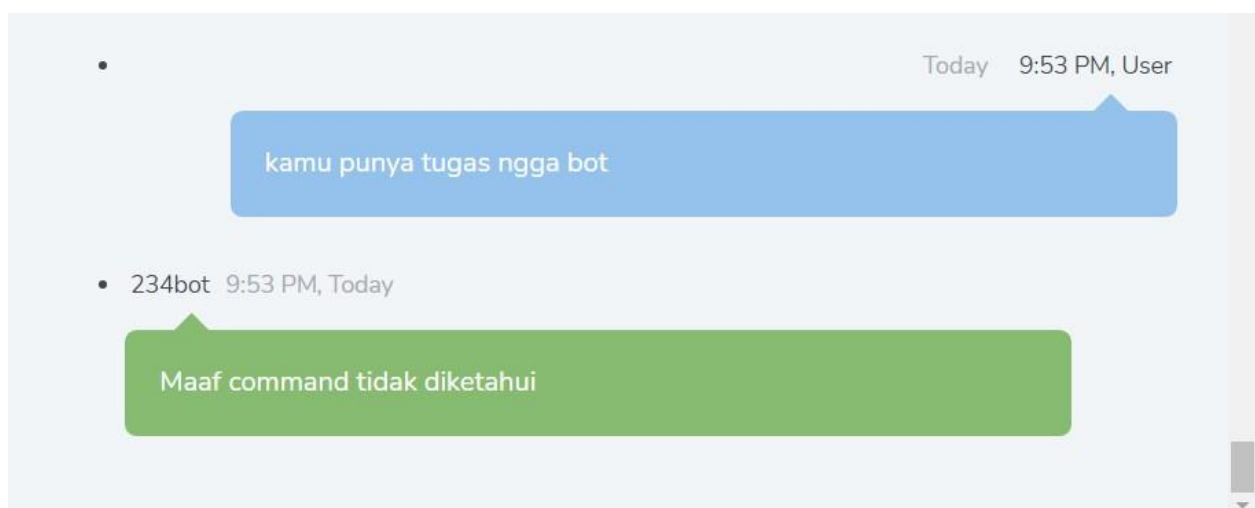
### 4.3 Hasil Pengujian



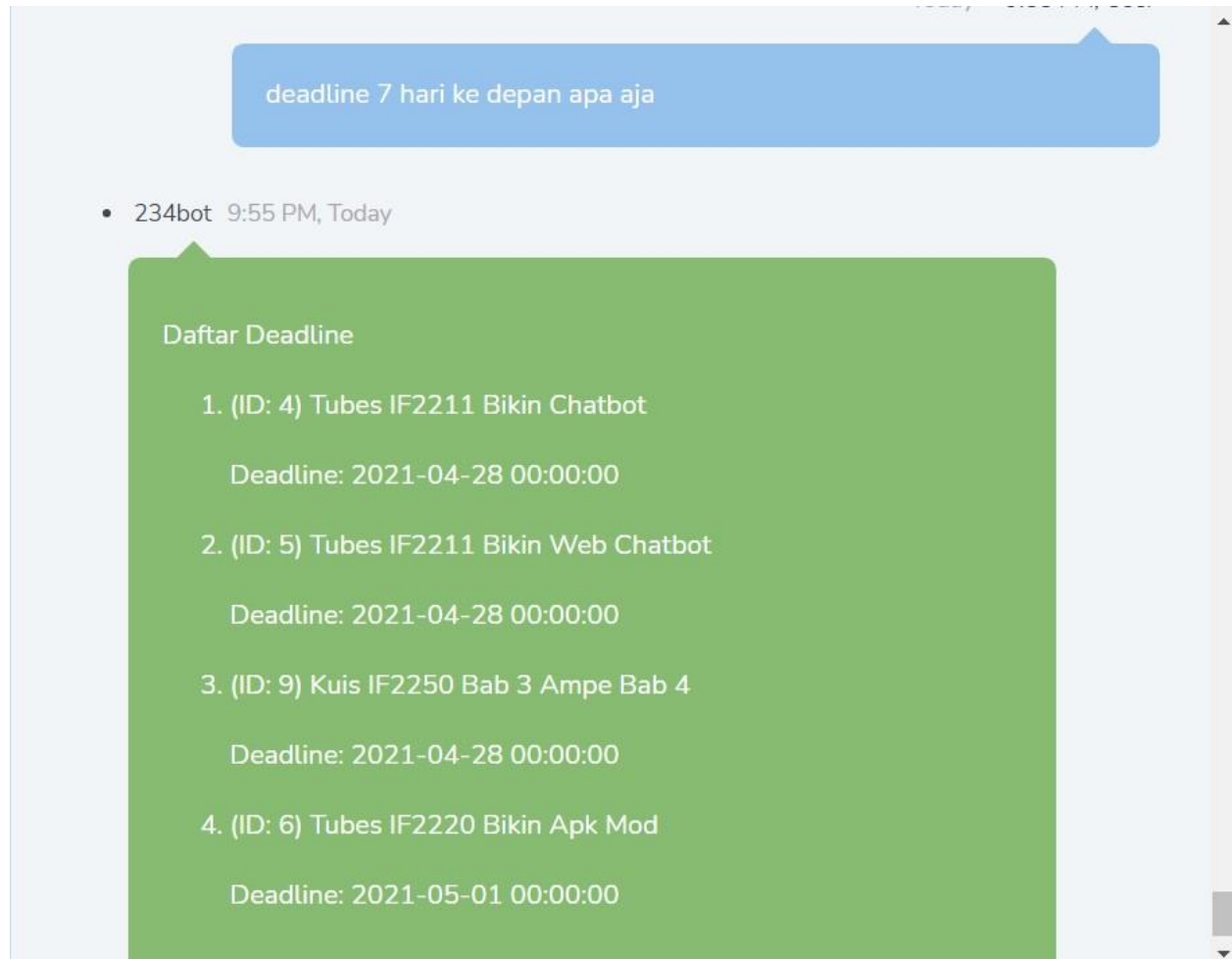
Gambar4.3.1 Hasil Pengujian Task Diselesaikan



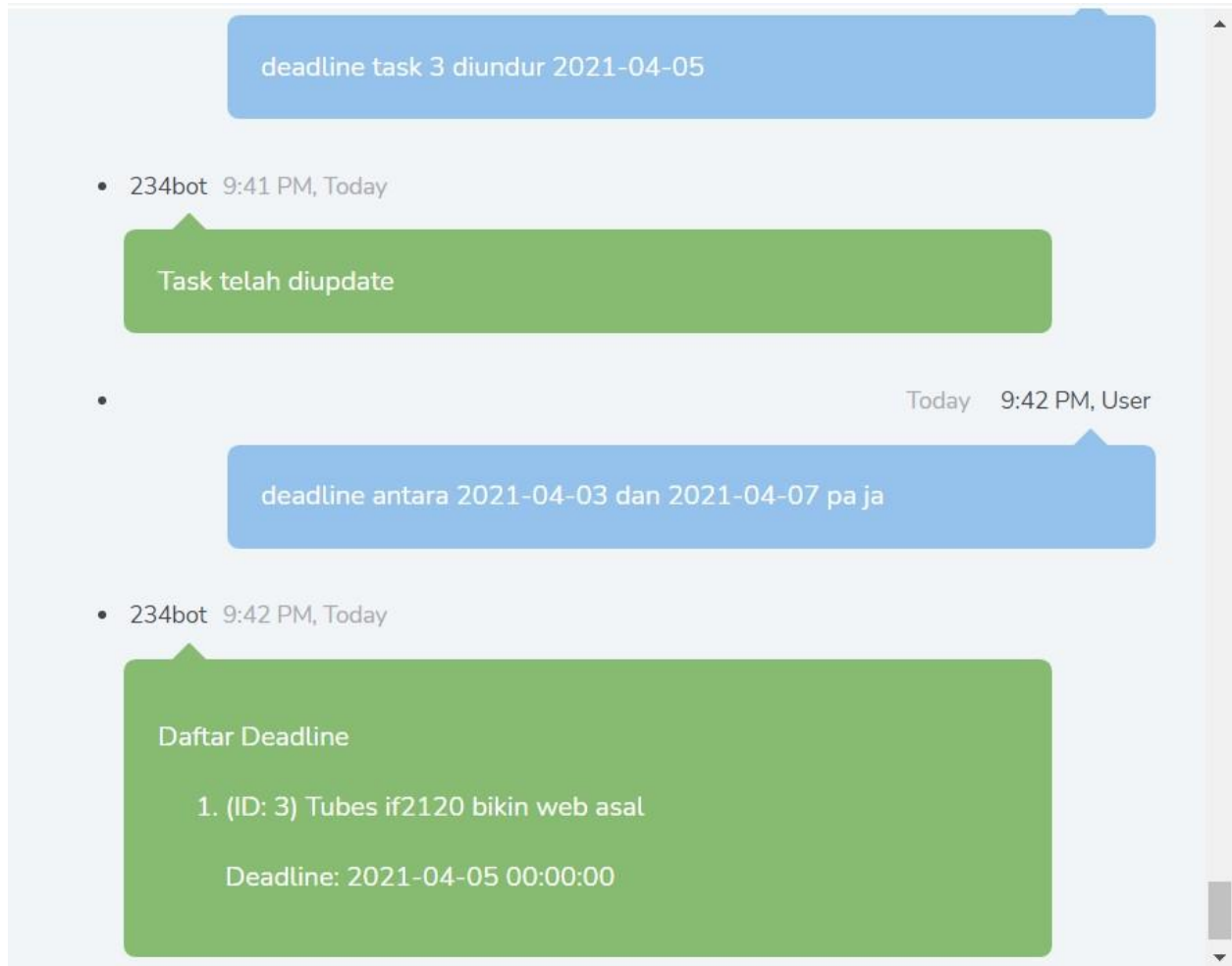
Gambar 4.3.2 Hasil Pengujian Fitur Help



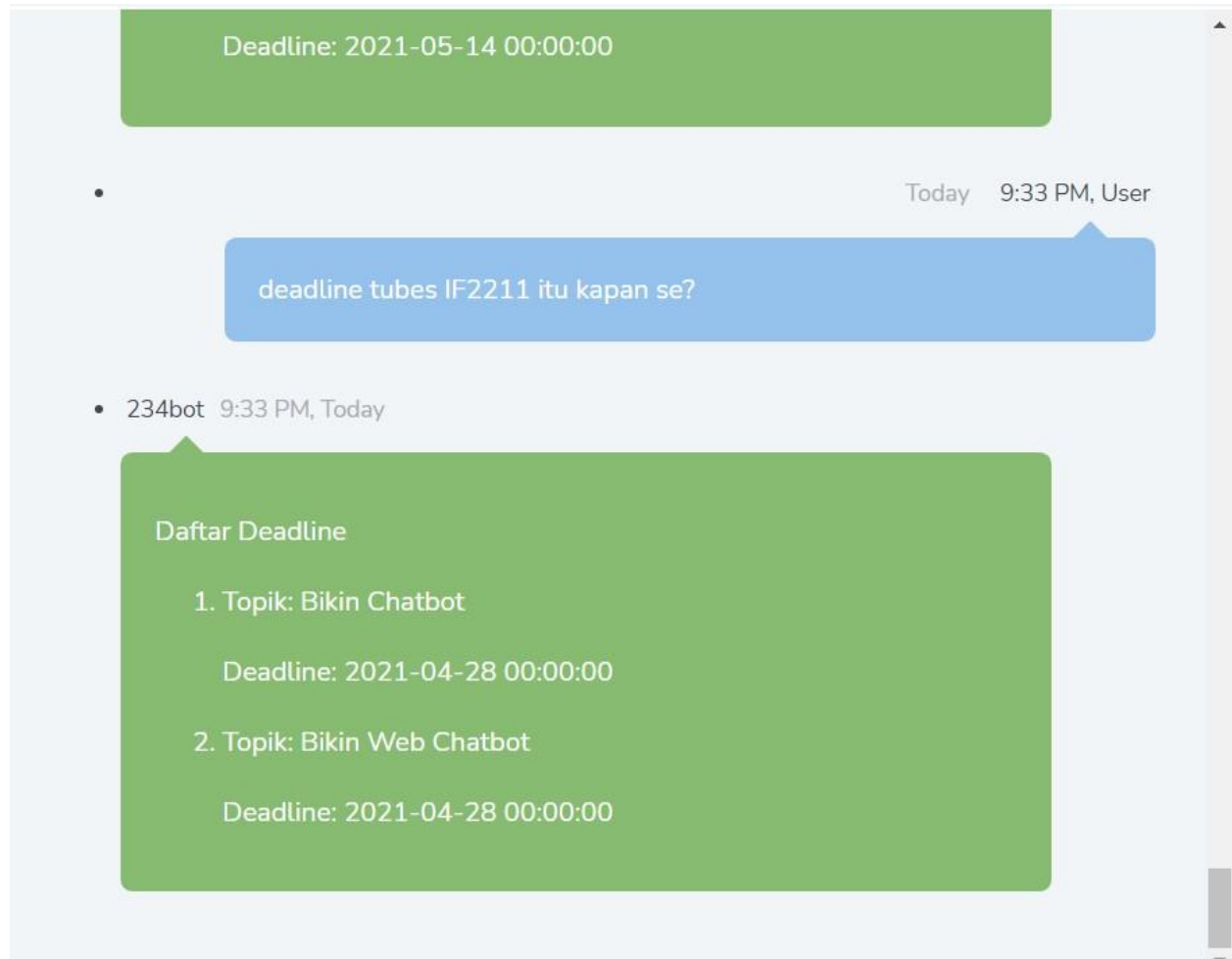
Gambar 4.3.3 Hasil pengujian command tidak diketahui



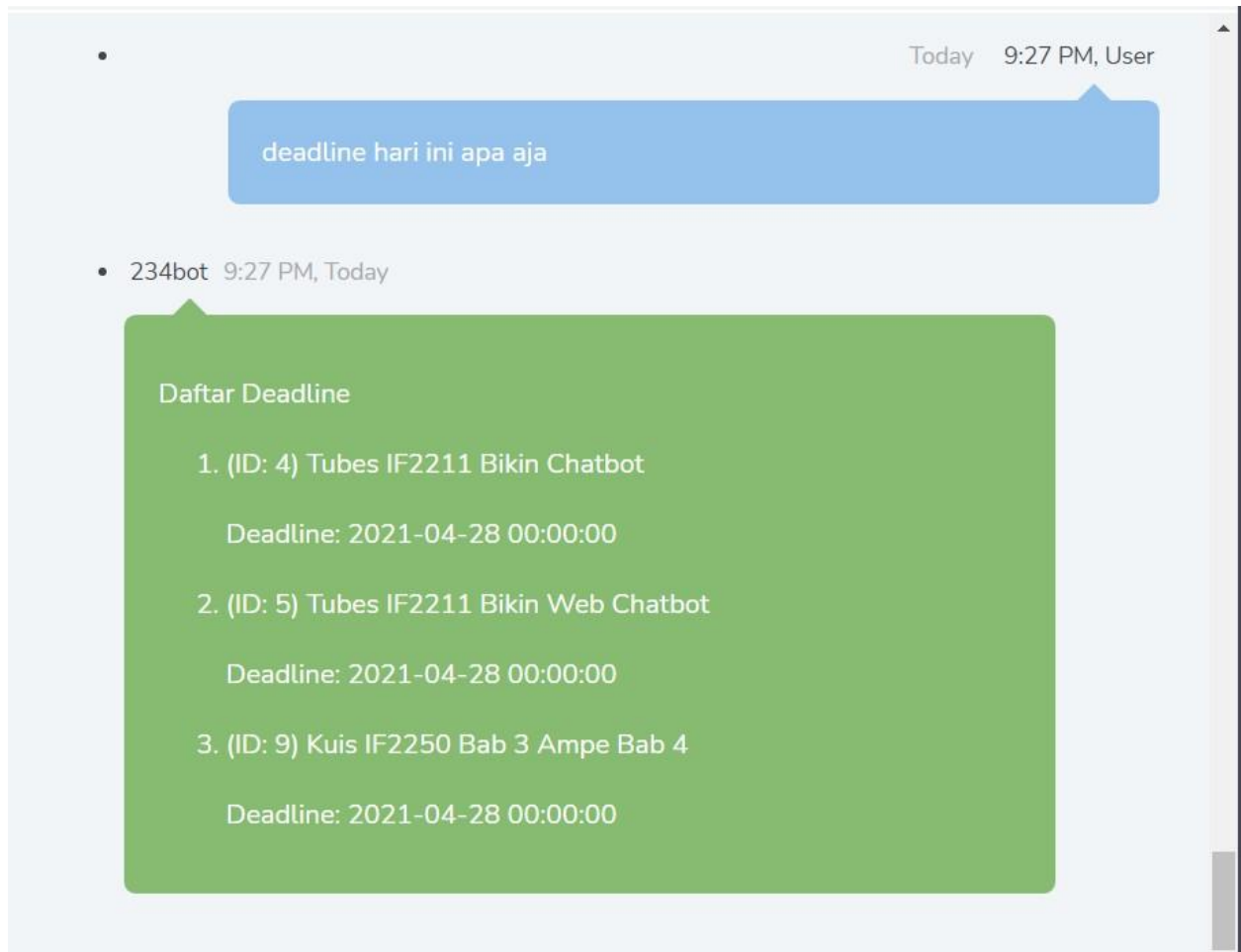
Gambar 4.3.4 Hasil Pengujian Fitur Deadline berdasarkan periode waktu



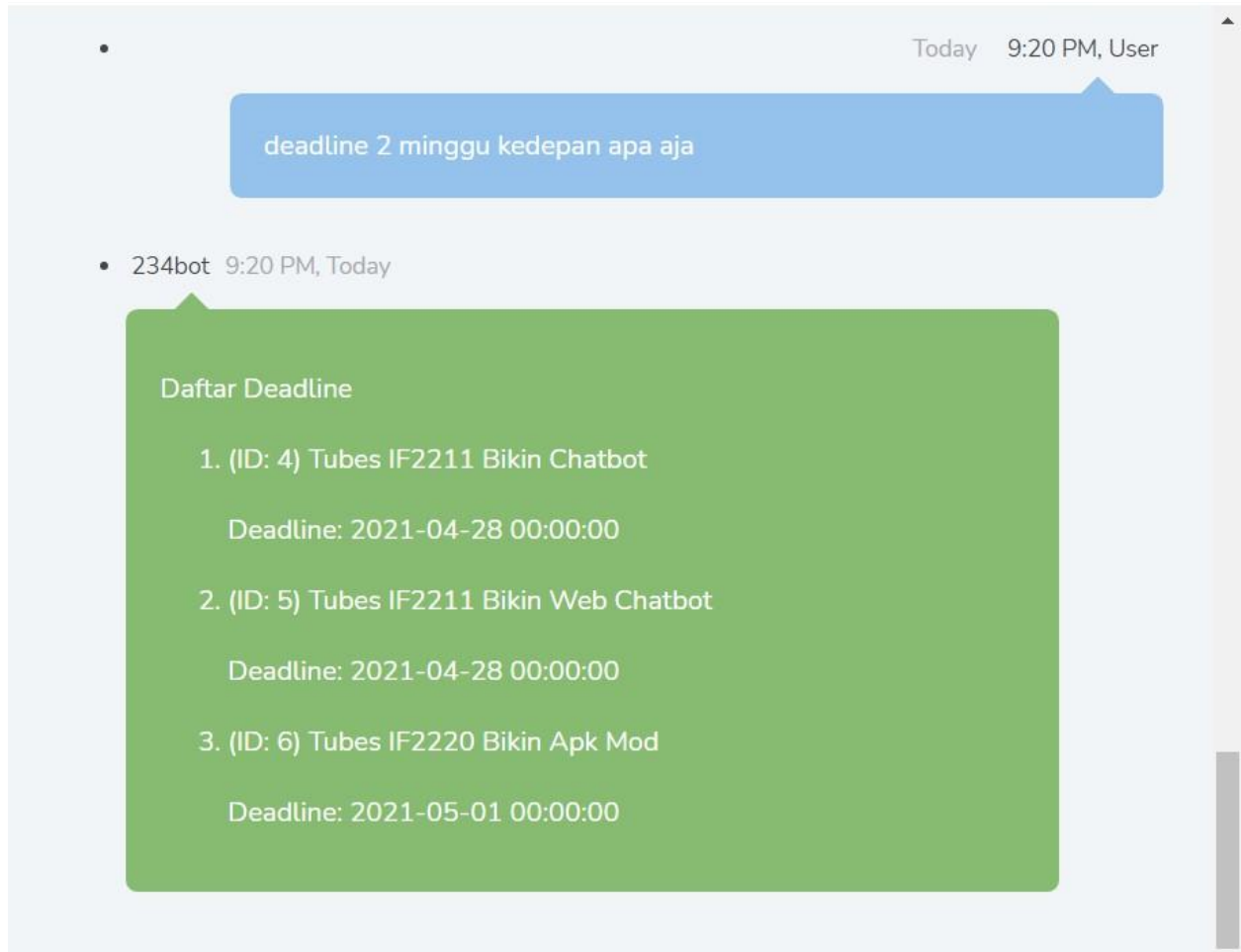
Gambar 4.3.5 Hasil pengujian fitur update deadline serta fitur menampilkan deadline berdasarkan periode waktu tertentu.



Gambar 4.3.6 Fitur penampilan deadline berdasarkan jenis tugas dan matkul tertentu

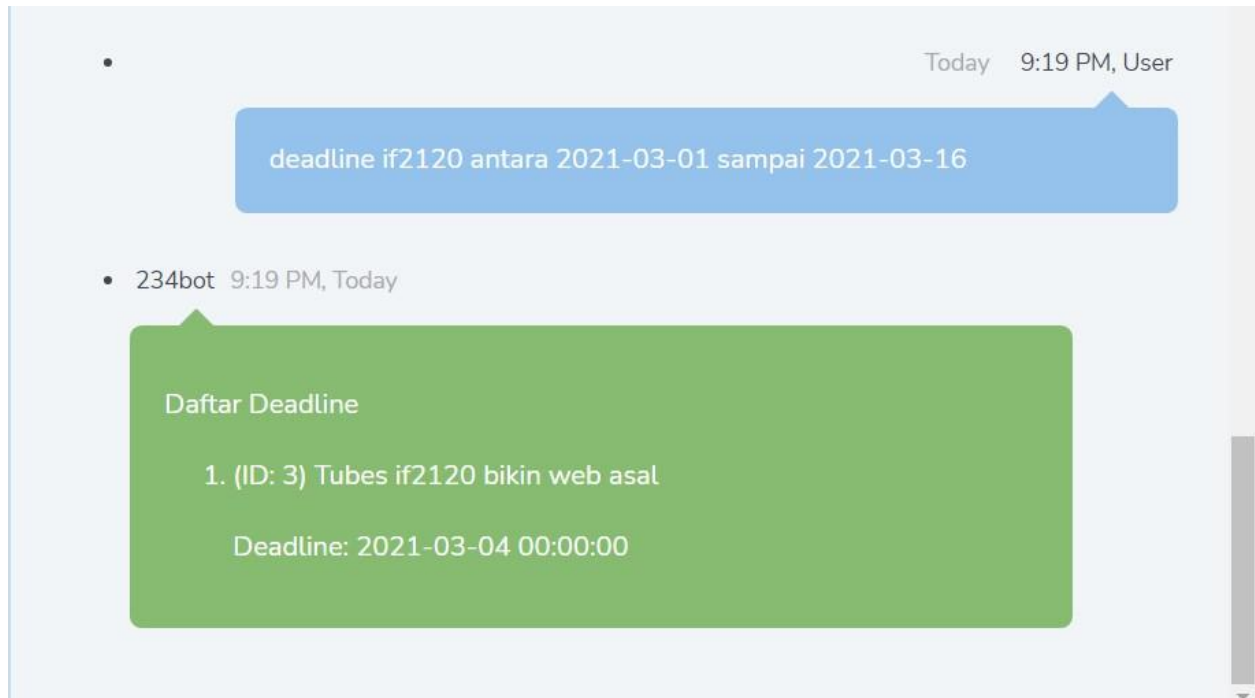


Gambar 4.3.7 Fitur penampilan deadline berdasarkan periode waktu tertentu

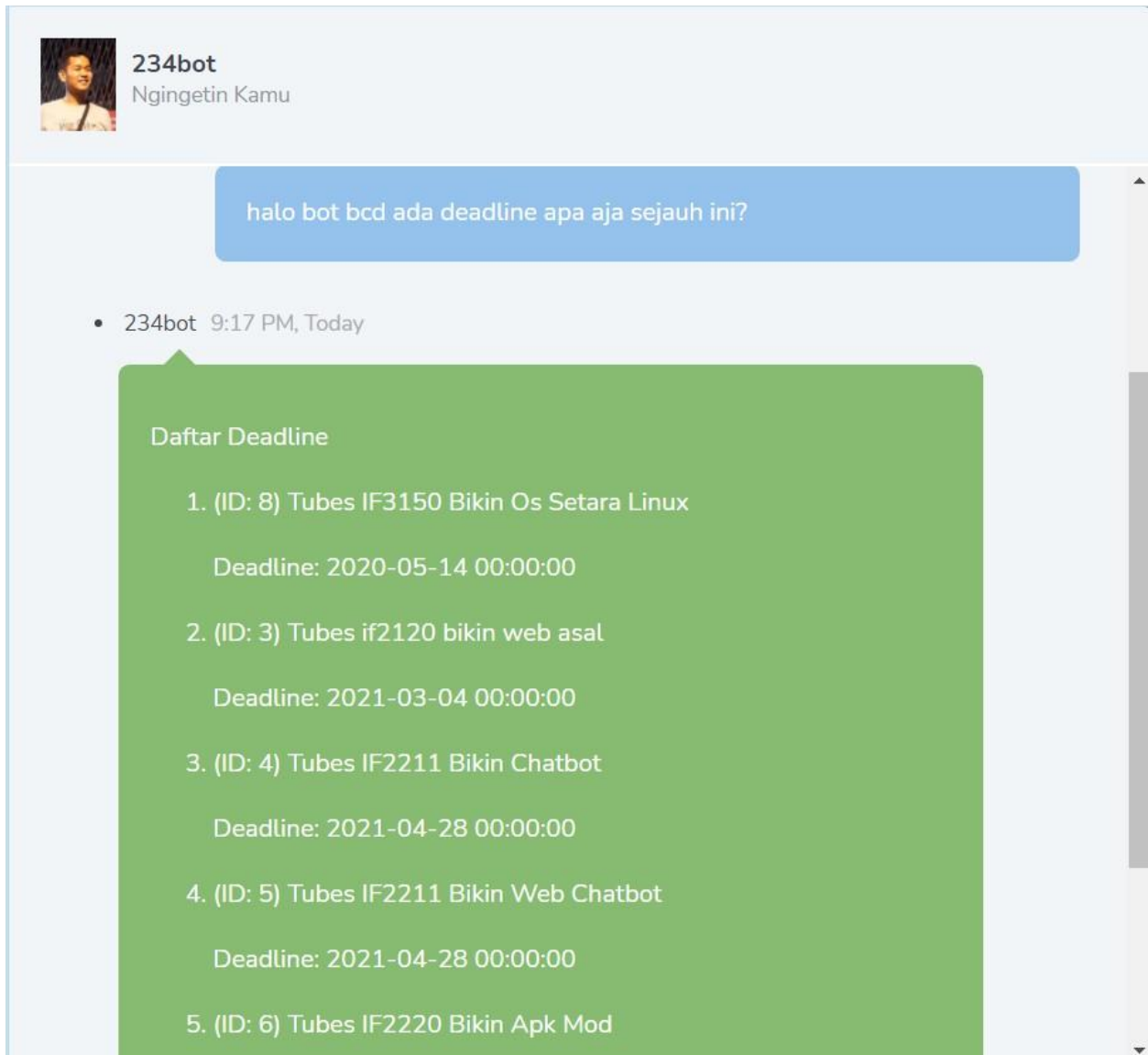


Gambar 4.3.8 Fitur penampilan deadline berdasarkan periode waktu tertentu

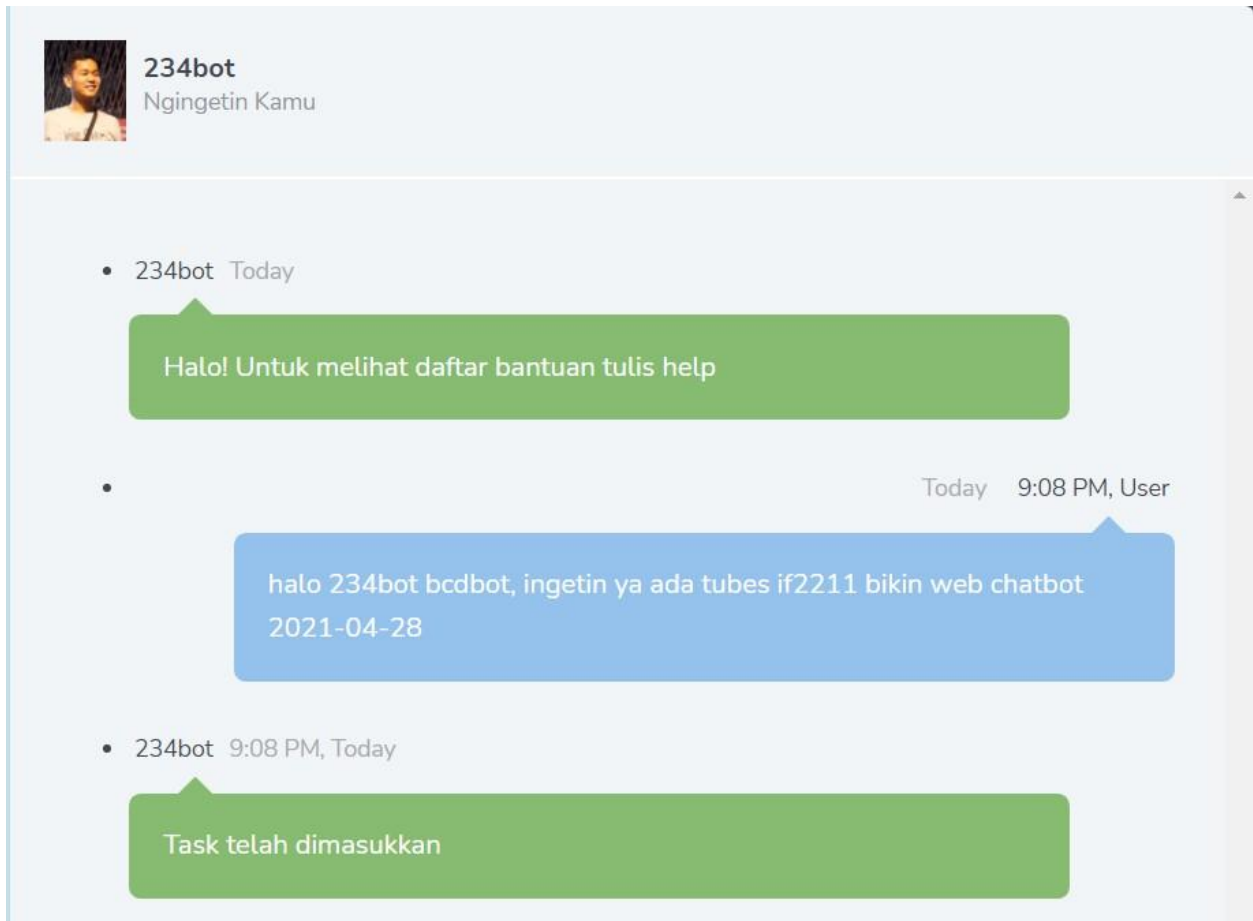




Gambar 4.3.9 Fitur Penampilan deadline berdasarkan periode waktu tertentu.



Gambar 4.3.10 Fitur menampilkan seluruh deadline yang tersedia.



Gambar 4.3.11 Fitur penambahan task

#### 4.4 Analisis Hasil Pengujian

Berdasarkan hasil yang kami dapatkan, dapat disimpulkan bahwa chatbot pengingat 234bot yang kami rancang sudah memenuhi spesifikasi tugas besar yang ada. Fitur-fitur yang ada juga sudah bekerja dengan baik.

Alasan kami menggunakan KMP string matching algorithm adalah karena time complexity dari algoritma tersebut adalah  $O(n)$ .

Kemudian untuk regular expression yang kami definisikan sudah dapat mencangkup seluruh kata penting yang ada. Hanya saja di bagian topik yang masih kurang sempurna untuk pendefinisian regular expression.

## **BAB V**

### **Kesimpulan dan Saran**

#### **5.1 Kesimpulan**

Dalam membuat suatu chatbot mengingat algoritma string matching dan regular expression merupakan algoritma yang tepat untuk menunjang bot tersebut. Karena dengan adanya algoritma string matching dan regular expression, bot yang dibuat dapat menerima perintah-perintah terkesan lebih interaktif daripada bot sederhana.

#### **5.2 Saran**

1. Sebaiknya batas waktu tugas besar tidak selisih 1 hari dengan tugas besar lain.
2. Beberapa regex dapat diperbagus untuk mendapatkan informasi yang paling tepat.
3. Tampilan UI dapat dibuat lebih bagus dan interaktif.

## Daftar Pustaka

Munir, Rinaldi. 2021. “[Pencocokan String \(String/Pattern Matching\)](https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Pencocokan-string-2021.pdf).”  
<https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Pencocokan-string-2021.pdf>  
(diakses pada 27 April 2021).

Munir, Rinaldi. 2021. “[String Matching dengan Regular Expression](https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2018-2019/String-Matching-dengan-Regex-2019.pdf)”  
<https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2018-2019/String-Matching-dengan-Regex-2019.pdf>  
(diakses pada 27 April 2021).