

LAPORAN TUGAS BESAR

IF2211

STRATEGI ALGORITMA

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



DIBUAT OLEH:

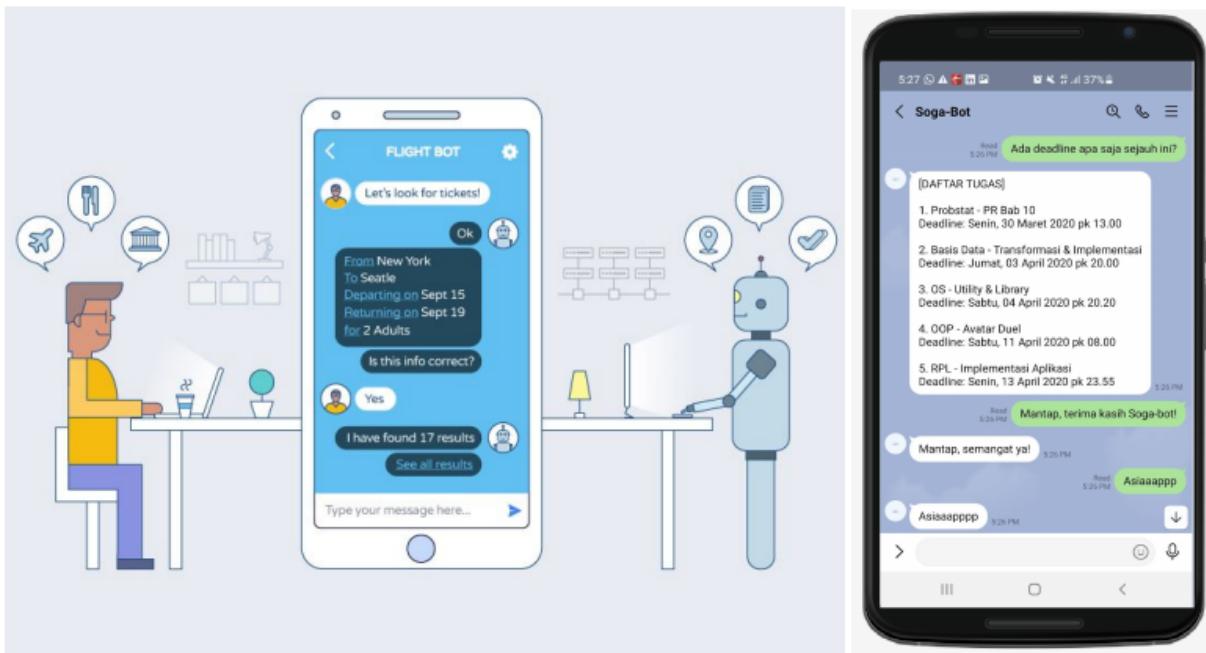
JOEL TRIWIRA / 13519073

HIZKIA RADITYA PRATAMA ROOSADI / 13519087

ANDRIANATA PUTRA TJANDRA / 13519059

Bab 1 Deskripsi Tugas

Bukan sesuatu yang janggal lagi jika semakin hari tugas-tugas di Teknik Informatika Semester 4 semakin bertambah banyak. Hal ini tentunya berakibat pada bertambahnya kegiatan dan pekerjaan yang harus dilakukan mahasiswa. Tak jarang pula ada tugas yang terlupakan karena mahasiswa sulit untuk mengingat semua tugas dan deadline tersebut. Oleh karena itu, mahasiswa Teknik Informatika berniat untuk membuat suatu Google Assistant sederhana berupa Deadline Reminder Assistant, atau dalam bahasa Indonesia adalah Asisten Pengingat Deadline.



Gambar 1. Ilustrasi Chatbot dan Asisten Pengingat Deadline Sumber :

<https://id.pinterest.com/pin/824299538024636729/> dan dokumentasi pribadi

Di era digital ini, kita tentu sudah pernah mendengar teknologi atau aplikasi seperti Chatbot, LINE Bot, atau Google Assistant. Ketiganya merupakan agen cerdas yang meniru kemampuan manusia untuk melakukan percakapan dengan user. Kehadiran Chatbot ini tentu membantu kehidupan manusia, khususnya dalam membantu menyajikan informasi yang diperlukan user dan menjawab berbagai pertanyaan yang sering ditanyakan oleh user. Secara spesifik dalam konteks Asisten Pengingat Deadline ini, Chatbot tersebut akan menjawab pertanyaan-pertanyaan mahasiswa yang sering ditanyakan seperti deadline seminggu ke depan,

deadline di bulan ini, dan task-task penting lainnya yang perlu dilakukan. Chatbot ini akan sangat membantu user agar tidak lagi melewatkkan deadline tugas.

Dalam tugas besar ini, penulis diminta untuk membangun sebuah chatbot sederhana yang berfungsi untuk membantu mengingat berbagai deadline, tanggal penting, dan task-task tertentu kepada user yang menggunakannya. Dengan memanfaatkan algoritma String Matching dan Regular Expression, penulis dapat membangun sebuah chatbot interaktif sederhana layaknya Google Assistant yang akan menjawab segala pertanyaan terkait informasi deadline tugas-tugas yang ada

Bab 2 Landasan Teori

String matching merupakan proses pencocokan kata berdasar urutan karakter yang menyusun kata tersebut. Proses ini dapat diimplementasikan dengan beberapa cara. Salah satu cara yang naif adalah menggunakan metode brute-force yaitu mencocokan karakter string satu per satu dengan kalimat yang ada sampai ditemukan susunan yang sesuai. Cara ini tidak efisien karena perlu dilakukan pencocokan yang berulang. Oleh karena itu, beberapa metode pencocokan string ditemukan untuk meningkatkan efisiensi proses string matching. Tiga diantara banyak metode yang diajarkan pada perkuliahan IF2211 Strategi Algoritma adalah Knuth-Morris-Pratt Algorithm (KMP), Boyer-Moore Algorithm (BM), dan string matching menggunakan regular expression (regex). Dalam tulisan ini, ketiga metode tersebut akan dijelaskan dan algoritma yang paling cocok akan diimplementasikan untuk membuat bot asisten.

Algoritma KMP adalah algoritma pencocokan string yang merupakan algoritma Brute Force yang lebih cerdas. Algoritma Brute Force hanya bergeser ke kanan sebanyak satu karakter untuk setiap perbandingan, sedangkan KMP akan bergeser ke kanan sesuai dengan prefiks bagian string perbandingan yang sudah cocok dan suffixnya. Pola yang dicari digeser ke kanan sampai prefiks pada pola bertetangga dengan suffix pada bagian string yang dicari.

Algoritma BM adalah algoritma yang juga melakukan pergeseran ke kanan untuk setiap bagian string yang dibandingkan. Namun, tidak seperti KMP, patokan pergeseran BM adalah berdasarkan karakter pertama pada bagian string yang baru saja dibandingkan yang sama dengan pola yang dicari. Jika ada pada sebelah kanan bagian string, maka akan digeser sampai kedua karakter tepat bertetangga.

Regex merupakan urutan karakter yang dapat menspesifikasi pola pencarian. Regex pertama kali dikenalkan ke mahasiswa pada kelas Teori Bahasa Formal dan Otomata. Pada penggunaannya, regex dapat digunakan untuk menspesifikasi pola string yang akan dicari pada sebuah kalimat. Banyak bahasa pemrograman yang sudah memiliki modul untuk memproses regex dan akan digunakan juga dalam implementasi pembuatan bot asisten.

Chatbot adalah suatu perangkat lunak yang dapat digunakan oleh pengguna untuk berinteraksi dengan sebuah bot. Bot ini seperti manusia pada umumnya yang dapat merespon

sesuai dengan masukan pengguna namun jauh lebih terbatas dibanding manusia. Contoh chatbot yang terkenal adalah Simsimi.

Bab 3 Analisis Pemecahan Masalah

1. Menambah task baru

Pada pesan yang diketik oleh pengguna akan dicari kata kunci yang sesuai, yaitu “add” atau “tambah”. Jika ada kata yang sesuai, maka program akan menambahkan sebuah deadline.

Program akan mencari kode kuliah, tipe tugas, topik tugas, dan tanggal deadline pada pesan yang diberikan oleh pengguna. Kode kuliah sudah didefinisikan secara eksplisit pada sebuah array sehingga mencari kode kuliah pada pesan hanya perlu mencari yang sama. Tipe tugas juga telah didefinisikan sebelumnya dan dicari yang cocok. Topik tugas didapat dengan mengambil kata pertama setelah kode kuliah dan sebelum kata ‘pada’. Tanggal didapat dengan mengambil substring setelah kata ‘pada’ sampai integer terakhir, yaitu tahunnya.

Jika semua informasi yang diperlukan sudah didapat, deadline akan bertambah sebanyak 1 dengan ID otomatis diberikan sesuai dengan indeksnya. Jika ada informasi yang kurang atau kesalahan format, bot mengeluarkan pesan gagal.

2. Melihat daftar task

Bagian program ini akan terbagi menjadi 3 kasus, namun sebelumnya harus ada kata kunci “**apa saja**” untuk memastikan pengguna ingin melihat deadline:

- Semua task: Program akan mencari kata kunci “**sejauh ini**”, jika ditemukan, berarti program akan menampilkan semua task yang ada
- Antara 2 Tanggal: Jika ada dua tanggal pada sebuah pesan, dapat dipastikan pengguna ingin mencari deadline apa saja di antara kedua tanggal itu. Maka program akan menampilkan semua deadline antara rentang waktu itu.
- N Minggu ke depan: Program akan mencari kata kunci “**N minggu ke depan**” (N adalah integer). Jika ditemukan, program akan menampilkan semua deadline untuk N minggu ke depan.
- N Hari ke depan: Program akan mencari kata kunci “**N hari ke depan**” (N adalah integer). Jika ditemukan, program akan menampilkan semua deadline untuk N hari ke depan.
- Hari ini: Program akan mencari kata “**hari ini**” dan kemudian menampilkan deadline pada hari ini.

3. Melihat deadline dari task tertentu

Program akan mencari kata kunci “**kapan**”. Jika ditemukan, program akan lanjut dengan mencari tipe deadline pada pesan, misalnya tugas, lalu mencari pada deadline yang ada manakah yang cocok dengan permintaan pengguna.

4. Mengubah deadline dari task tertentu

Saat pengguna mengirim pesan kepada bot, diperiksa apakah pesan yang dikirim mengandung kata kunci 'maju', 'undur', atau 'ganti'. Jika ada kata yang cocok, maka bisa diketahui bahwa pengguna ingin mengundur atau memajukan deadline.

Proses dimulai dengan mengambil data yang diperlukan pesan pengguna, yaitu tanggal tugas tersebut dimajukan atau dimundurkan dan id tugas yang dituju. Karena id adalah indeks dari array deadline ditambah satu, maka dicari indeksnya dan mengganti tanggal dari deadline yang sudah ada menjadi tanggal masukan pengguna.

Jika ID atau tanggal tidak diberikan pada pesan, akan ditampilkan untuk memasukkan ID atau tanggal. Jika ID tidak ada, maka akan ditampilkan pesan "ID tidak ada"

5. Menyelesaikan sebuah deadline

Proses dimulai dengan mencari kata kunci "selesai" atau "done". Jika ada kata kunci tersebut, maka diketahui pengguna sudah menyelesaikan sebuah deadline.

Pertama program akan mencari data yang diperlukan, yaitu id yang sudah selesai pada pesan pengguna. Deadline yang sudah ada dan sesuai dengan ID yang dimasukkan akan dihapus pada array deadline yang ada.

Jika ID tidak ada pada pesan, akan ditampilkan pesan untuk meminta ID. Jika ID yang diberikan tidak valid, akan ditampilkan pesan ID tidak ada.

6. Menampilkan daftar command

Kata kunci untuk menampilkan daftar command yang diterima adalah ‘help’. Jika ditemukan, maka bot akan mengeluarkan pesan yang berisi kata kunci yang dapat diterima bot.

7. Menampilkan pesan error

Pesan error akan ditampilkan jika pada masukan pengguna tidak terdapat kata kunci yang telah dijabarkan pada poin sebelumnya.

Bab 4 Implementasi dan Pengujian

Secara umum, program akan terlebih dahulu mengambil deadline yang sudah ada dari file deadline.txt. File ini berisi sebuah integer yang menyatakan jumlah deadline yang sudah ada dan diikuti dengan deadline sebanyak integer tersebut. Tiap deadline dipisahkan dengan sebuah newline.

Hasil pembacaan file akan disimpan pada sebuah array bernama deadline. Array akan berubah sesuai dengan masukan pengguna dan setiap perubahan yang ada akan dimasukkan kembali ke dalam deadline.txt

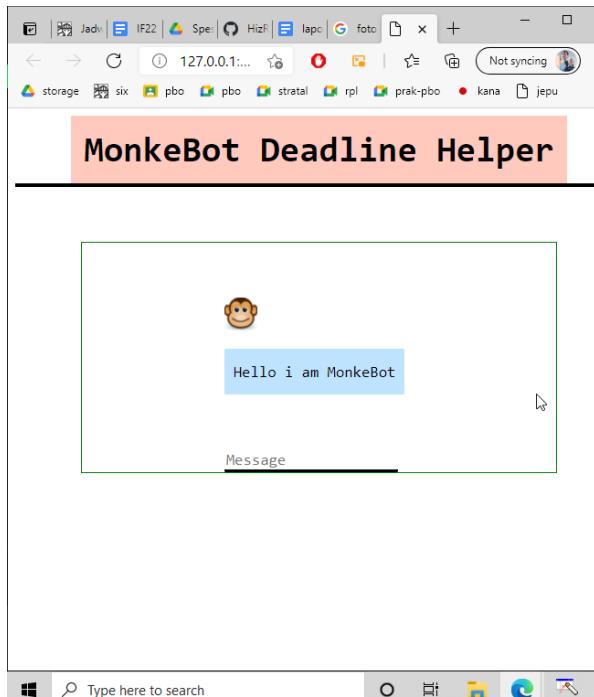
Fungsi utama pada program terdapat pada fungsi get_bot_response(). Fungsi ini akan menerima input dari pengguna kemudian mengolahnya sesuai dengan kasus yang ditangani dan mengembalikan respon berdasarkan masukan pengguna. Hasil yang dikembalikan berupa sebuah string yang akan ditampilkan pada layar.

Ada beberapa fungsi untuk mencari informasi dari masukan pengguna yang diawali dengan ‘find’ seperti findTanggal(). Ada fungsi findKodeKuliah() untuk mencari kode kuliah (IF2210), findTipeTugas() untuk mencari tipe tugas (Ujian/Praktikum/dll.), findTopikTugas() untuk mencari topik dari tipe tugas (bebas), dan findTanggal() untuk mencari tanggal.

Tanggal dapat dituliskan dengan berbagai format seperti DD/MM/YY, DD/MM/YYYY, DD-MM-YYYY, DD-MM-YY, atau DD <nama bulan> YY. Kasus ini ditangani pada fungsi convertStringToDate() untuk mengubah semua format yang ada menjadi tipe data datetime.

Program memiliki tampilan depan layaknya sebuah aplikasi mengirim pesan. Pengguna dapat mengetik sebuah teks pada kolom dan menekan enter untuk mengirimkannya pada bot.

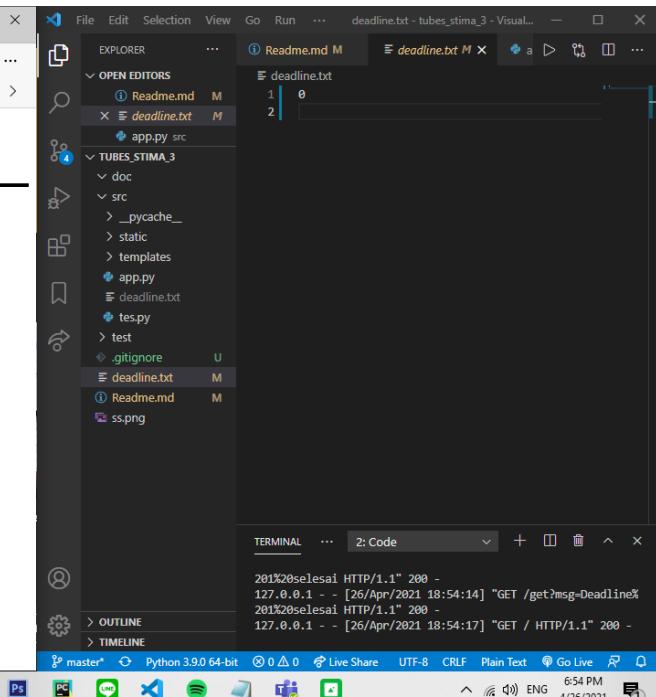
Dengan mengirim pesan ke bot tersebut, pengguna dapat menambahkan deadline dari sebuah task, bisa berupa ujian, tubes, tucil, kuis, atau praktikum, yang dimilikinya untuk dipantau. Deadline ini dapat juga diubah jika seandainya task yang ada diundur atau dimajukan deadlinenya. Setiap saat pengguna dapat memeriksa apa saja deadline yang ada untuk suatu rentang waktu atau bisa juga deadline untuk suatu task tertentu. Jika sudah selesai, pengguna dapat menghapusnya dengan memberitahu bot bahwa tasknya sudah selesai.



MonkeBot Deadline Helper

Hello i am MonkeBot

Message



File Edit Selection View Go Run ... deadline.txt + tubes_stima_3 - Visual... ...

EXPLORER OPEN EDITORS

Readme.md M deadline.txt M app.py src

TUBES_STIMA_3

doc

src

pycache

static

templates

app.py

deadline.txt

tes.py

test

.gitignore

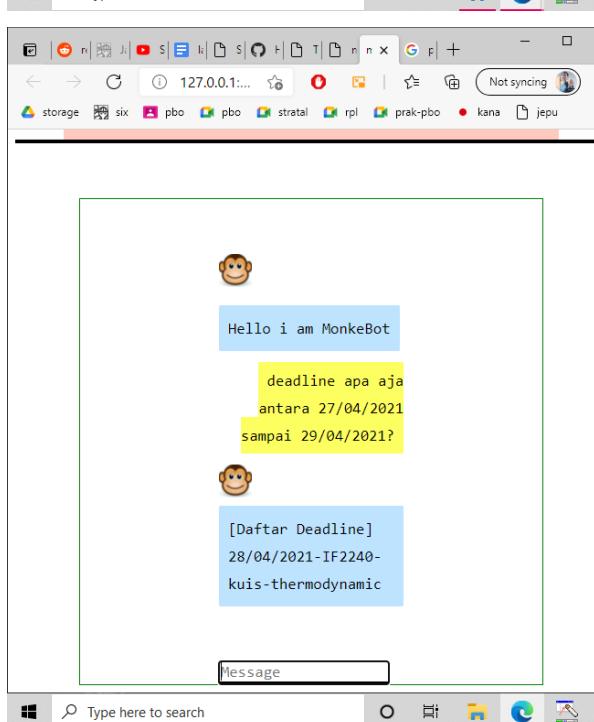
deadline.txt M

Readme.md M ss.png

TERMINAL 2: Code

```
201%20selesai%20HTTP/1.1" 200 -
127.0.0.1 - - [26/Apr/2021 18:54:14] "GET /get?msg=Deadline%
201%20selesai%20HTTP/1.1" 200 -
127.0.0.1 - - [26/Apr/2021 18:54:17] "GET / HTTP/1.1" 200 -
```

Type here to search

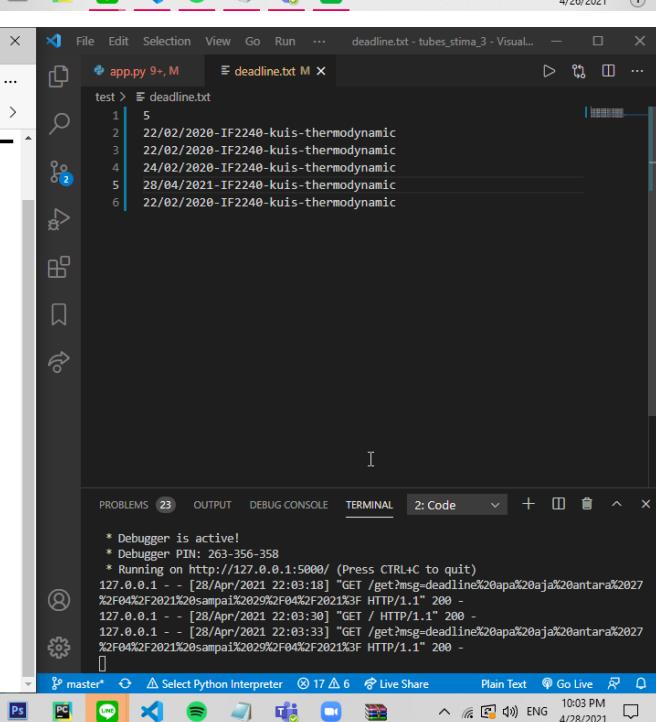


Hello i am MonkeBot

deadline apa aja
antara 27/04/2021
sampai 29/04/2021?

[Daftar Deadline]
28/04/2021-IF2240-
kuis-thermodynamic

Message



File Edit Selection View Go Run ... deadline.txt M app.py 9+ M

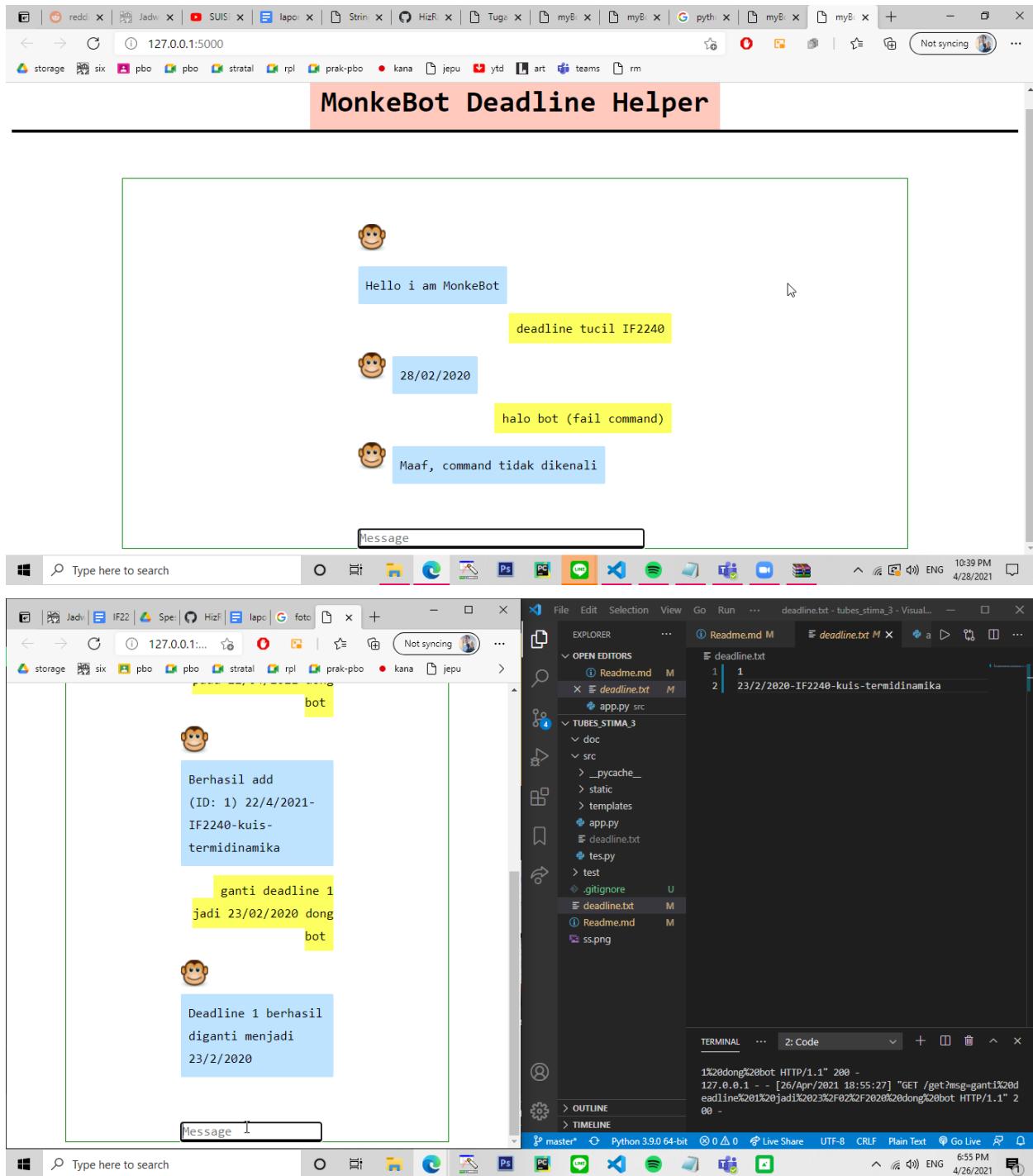
test > deadline.txt

```
1 5
2 22/02/2020-IF2240-kuis-thermodynamic
3 22/02/2020-IF2240-kuis-thermodynamic
4 24/02/2020-IF2240-kuis-thermodynamic
5 28/04/2021-IF2240-kuis-thermodynamic
6 22/02/2020-IF2240-kuis-thermodynamic
```

PROBLEMS 23 OUTPUT DEBUG CONSOLE TERMINAL 2: Code

```
* Debugger is active!
* Debugger PID: 263-356-358
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
127.0.0.1 - - [28/Apr/2021 22:03:18] "GET /get?msg=Deadline%20apa%20aja%20antar%2027%2F04%2F2021%20sampa%2028/04/2F2021%3F HTTP/1.1" 200 -
127.0.0.1 - - [28/Apr/2021 22:03:30] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [28/Apr/2021 22:03:33] "GET /get?msg=Deadline%20apa%20aja%20antar%2027%2F04%2F2021%20sampa%2029/04/2F2021%3F HTTP/1.1" 200 -
```

Type here to search



Dari hasil yang didapat pada pengujian di atas, bot berhasil menjalankan semua fungsionalitas dengan benar. Bot dapat menambah deadline dari sebuah task dengan benar, menampilkan deadline yang sudah ada baik dalam rentang waktu tertentu atau sebuah task spesifik,

memodifikasi deadline yang sudah ada, menyelesaikan sebuah deadline, dan menampilkan pesan bantuan penggunaan.

Bab 5 Kesimpulan, Saran dan Refleksi

Chatbot yang dibuat pada tugas ini menggunakan library regular expression milik Python. Hal ini membantu dalam mempermudah implementasi. Penulis program hanya cukup memikirkan pola string dalam text dan dimasukan dalam bentuk regular expression agar dapat ditemukan. Metode regular expression juga tentunya akan lebih efisien dibandingkan dengan metode seperti brute-force karena tidak mengecek satu per satu urutan string dalam text. Penggunaan regular expression juga mempermudah hal-hal kecil seperti penyesuaian format tanggal. Pengguna dapat memasukan berbagai format tanggal, “24-01-2021” , “24 Januari 2021” , “24/01/2021” , dan lainnya. Regex membantu penemuan pola pola seperti ini.

Sebagai saran untuk pengembang selanjutnya atau orang yang ingin membuat chatbot dengan string matching, pola pencocokan string selain regex dapat digunakan seperti KMP atau BM algorithm. Untuk implementasi selanjutnya, sebaiknya kata-kata yang dapat diterima oleh chatbot / grammar chatbot juga sebaiknya semakin dikembangkan lebih lanjut agar chatbot dapat menerima variasi yang lebih banyak. Terakhir, diusahakan untuk pengembang lainnya agar tidak mengerjakan tugas ini mendekati deadline mengingat banyaknya fitur yang harus disesuaikan

Secara umum, kelompok pengembang saat ini sudah melakukan tugasnya dengan cukup baik walaupun masih ada beberapa hal yang harus diperbaiki. Pembuatan chatbot ini menambah wawasan penulis program mengenai chatbot dan string matching secara umum. Semoga dengan berakhirnya tugas ini, kemampuan penulis dalam pemrograman dapat semakin berkembang dan

mampu mengimplementasi program dengan menggunakan strategi algoritma agar lebih efisien dan efektif.

Daftar Pustaka

<https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Tugas-Besar-3-IF2211-Strategi-Algoritma-2021.pdf>

<https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Pencocokan-string-2021.pdf>

<https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2018-2019/String-Matching-dengan-Regex-2019.pdf>