

Tugas Besar 3 IF2211 Strategi Algoritma

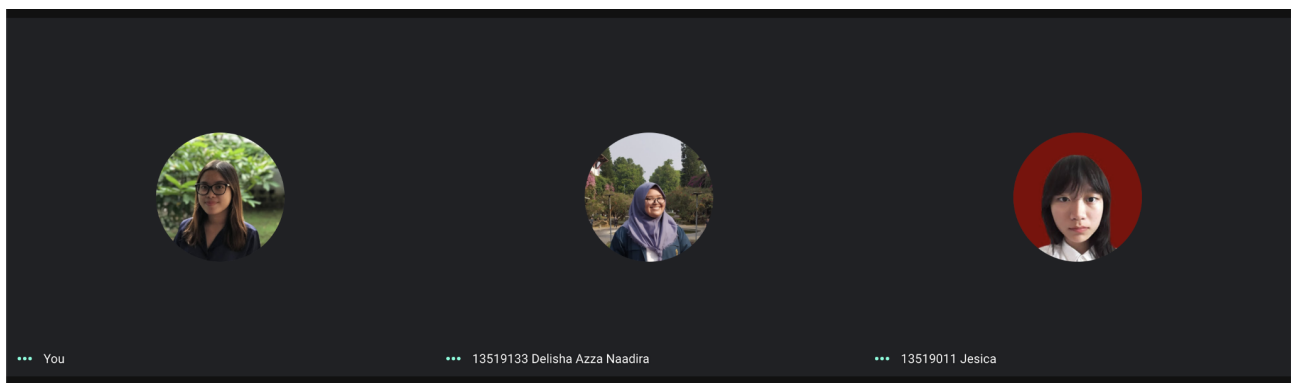
*Penerapan String Matching dan Regular Expression dalam Pembangunan Deadline
Reminder Assistant*

Kelompok 13519011

13519011 Jesica

13519040 Shafira Naya Aprisadiani

13519133 Delisha Azza Naadira



**PROGRAM STUDI SARJANA TEKNIK INFORMATIKA
SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA
INSTITUT TEKNOLOGI BANDUNG
2021**

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.

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 melewatkan deadline tugas. Deskripsi tugas: Dalam tugas besar ini, Anda akan 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, Anda dapat membangun sebuah chatbot interaktif sederhana layaknya Google Assistant yang akan menjawab segala pertanyaan Anda terkait informasi deadline tugas-tugas yang ada.

Deadline Reminder Assistant. akan dibangun dengan sistem Question and Answer dimana pengembang diharapkan sudah menyediakan kumpulan formula tertentu untuk melakukan pendeteksian setiap perbedaan command atau perintah pada aplikasi Chatbot.

2. Landasan Teori

2.1. Deskripsi singkat algoritma KMP, BM, dan Regex

KMP

Algoritma KMP melakukan pencocokan antara *pattern* dengan teks dari ke kiri ke kanan seperti algoritma Brute Force tapi pergeseran *pattern* lebih baik daripada algoritma Brute Force. Algoritma ini akan melakukan pencocokan dari kiri ke kanan, karakter per karakter antara *pattern* dengan teks yang bersesuaian sampai ditemukan ketidakcocokan antara karakter di *pattern* dengan karakter di teks atau sampai semua karakter antara *pattern* dengan teks cocok. Algoritma ini kemudian akan melakukan pergeseran *pattern* sesuai dengan tabel yang tercatat. Pencatatan table dilakukan ketika membandingkan teks dengan *pattern*. Kemudian diulangi langkah pergeseran sampai *pattern* berada di ujung teks.

Boyer-Moore

Algoritma *pattern matching* Boyer-Moore didasarkan oleh dua teknik. Teknik yang pertama adalah teknik ‘*looking glass*’, yaitu mencari *pattern* pada teks dengan bergerak dari akhir *pattern*. Teknik kedua adalah teknik ‘*character jump*’. Jika karakter yang sedang dibandingkan tidak sama, ada tiga kasus. Kasus pertama, jika karakter pada teks terdapat pada *pattern*, *pattern* digeser sehingga karakter tersebut sejajar. Kasus kedua, jika karakter pada teks yang dibandingkan terdapat pada *pattern*, namun tidak bisa digeser ke kanan hingga *last occurrence*, maka *pattern* digeser sebanyak satu karakter. Kasus terakhir, jika kedua kasus tersebut tidak terpenuhi, *pattern* digeser sebanyak banyak karakter pada *pattern* hingga karakter yang sedang dibandingkan.

Regex

Regular Expression adalah notasi yang mendeskripsikan kumpulan karakter dari sebuah *string*. Pada saat sebuah himpunan *string* (teks) mengandung sebuah *pattern string*, maka dapat dibilang terdapat kecocokan antara *string* dan *pattern* tersebut. Terdapat berbagai tanda baca yang merupakan notasi dari Regular Expression yang dapat mempermudah dalam pencocokan *pattern* dengan teks. Proses pencocokan yang dilakukan oleh Regular Expression juga bukanlah *exact matching* sehingga dapat memberikan hasil penelusuran yang lebih luas.

2.2. Penjelasan singkat mengenai Chatbot

Chatbot adalah salah satu bentuk dari artificial intelligence (AI) yang digunakan pada aplikasi pengiriman pesan. Chatbot merupakan program yang dapat berinteraksi dengan pelanggan layaknya manusia. Chatbot banyak digunakan untuk urusan customer service, request routing, atau information gathering. Chatbot yang advance menggunakan proses klasifikasi kata yang advance atau menerapkan natural language processing. Namun, chatbot sederhana dapat dibuat dengan meng-scan keyword general dan membuat respons chatbot berdasarkan library atau database.

Program yang dibuat pada Tugas Besar kali ini adalah chatbot sederhana dengan memanfaatkan ekstraksi informasi menggunakan algoritma string matching yaitu KMP, Boyer-Moore, dan Regular Expression. Data atau pengetahuan dari chatbot disimpan pada suatu file .txt. Lalu dibuat fungsi-fungsi untuk mengekstrak informasi, mendeteksi fitur, dan mengembalikan respons dari chatbot.

3. Analisis Pemecahan Masalah

3.1. Langkah penyelesaian masalah setiap fitur

1) Fitur 1 - Menambahkan task baru

Pada fitur ini, dilakukan pencarian tanggal, kode mata kuliah, jenis tugas dan topik tugas. Apabila keempat komponen tersebut ditemukan, maka data-data tersebut diolah dan kemudian dimasukan ke dalam *database*. Pertama, dilakukan pengecekan pada tiap-tiap komponen. Apabila data yang ingin dimasukan ternyata sudah terdapat di *database* maka akan dikeluarkan pesan error, sedangkan jika data sukses ditambahkan maka akan dikeluarkan pesan bahwa tugas berhasil ditambahkan.

2) Fitur 2 - Melihat daftar task yang harus dikerjakan

Pada fitur ini, pertama-tama dilakukan pencarian terhadap info-info yang mungkin terdapat di dalam pesan yang diinput *user*. Kemudian dengan menggunakan pencocokan string Boyer-Moore, dilakukan pencarian kata 'kapan' untuk memastikan tidak terdapat kata tersebut untuk menggunakan fitur ini, dilakukan juga pencarian kata 'deadline' untuk memastikan terdapat kata tersebut di dalam pesan untuk menggunakan fitur ini. Semua info yang dicari sebelumnya akan diolah dan dilakukan pencarian data tugas di *database* yang sesuai dengan info yang dimasukan oleh *user*. Jika data tugas ditemukan maka akan ditampilkan data tugas tersebut, jika tidak ditemukan maka akan ditampilkan pesan error.

3) Fitur 3 - Menampilkan deadline dari suatu task tertentu

Sama seperti fitur sebelumnya, akan di-*extract* info-info dari pesan yang dimasukan *user*. Kemudian dengan algoritma pencocokan string Boyer-Moore, dilakukan pencarian kata 'kapan' dan kata 'deadline' untuk memastikan terdapat kata tersebut dalam pesan agar fitur ini dapat digunakan. Kemudian akan dilakukan pencarian data tugas sesuai dengan info-info yang telah diperoleh sebelumnya. Jika data tugas ditemukan maka data tersebut akan ditampilkan ke layar. Jika tidak ditemukan maka akan ditampilkan pesan error.

4) Fitur 4 - Memperbaharui task tertentu

Pada fitur ini, dibuat fungsi untuk mendeteksi keyword untuk opsi update, task ID, dan tanggal pada pesan yang dimasukkan pengguna. Akan dicek juga apabila task ID terdapat pada database. Apabila task ID valid, akan dilakukan pembaharuan tanggal deadline sesuai dengan tanggal yang diinput oleh pengguna. Jika tidak, akan ditampilkan pesan error.

5) Fitur 5 - Menandai bahwa suatu task sudah selesai dikerjakan

Pada fitur ini, akan diperiksa apakah terdapat task ID yang valid pada pesan yang dimasukan oleh pengguna. Apabila ditemukan data tugas yang bersesuaian dengan ID tersebut maka tugas tersebut akan ditandai sebagai tugas yang telah selesai dikerjakan. Sedangkan jika ID tidak ditemukan maka akan ditampilkan pesan error.

6) Fitur 6 - Menampilkan opsi help yang difasilitasi oleh assistant

Fitur ini akan menerima input pengguna dan melakukan string matching dengan algoritma Boyer-Moore untuk keyword-keyword untuk opsi help. Jika ditemukan, akan ditampilkan semua list fitur-fitur bot dan daftar kata penting.

7) Fitur 7 - Mendefinisikan list kata penting terkait apakah itu merupakan suatu task atau tidak

Fitur ini digabung dengan fitur 6 dimana ketika user memasukan pesan untuk melihat daftar command yang ada maka akan ditampilkan juga list kata penting yang ada.

8) Fitur 8 - Menampilkan pesan error jika pesan tidak dikenali

Fitur ini dibuat sebagai kondisi else dari semua fitur yang ada. Jika bot tidak dapat mengenali pesan sebagai fitur 1 sampai 7, fitur 8 akan dieksekusi.

9) Fitur 9 - Bonus

Fitur ini dibuat berdasarkan Levenshtein Distance. Levenshtein Distance dihitung dengan cara membuat matriks lalu mengisi setiap cell pada matriks dengan suatu rumus. Kemudian, return value dari Levenshtein Distance adalah cell pada pojok kanan bawah.

Dari nilai Levenshtein Distance, dicari nilai similarity dengan rumus:

$$\text{similarity}(a,b) = 1 - \text{levenshtein_distance}(a,b)/\max(\text{len}(a),\text{len}(b))$$

Lalu, dibuat fungsi `word_recommendation` yang mereturn string kata yang direkomendasikan apabila ada typo. Jika tidak, fungsi ini akan mereturn string kosong.

Pada fungsi utama `get_bot_response(userMessage)`, fungsi `word_recommendation` dipanggil sebagai kondisi alternatif dari fitur 8. Kemudian bot akan menyarankan kalimat dengan me-replace kata-kata yang typo dengan kata-kata yang sebenarnya.

3.2. Fitur fungsional dan arsitektur Chatbot yang dibangun

Fitur fungsional dari Chatbot yang dibangun adalah:

- 1) Fitur 1 - Menambahkan task baru
- 2) Fitur 2 - Melihat daftar task yang harus dikerjakan
- 3) Fitur 3 - Menampilkan deadline dari suatu task tertentu
- 4) Fitur 4 - Memperbaharui task tertentu
- 5) Fitur 5 - Menandai bahwa suatu task sudah selesai dikerjakan
- 6) Fitur 6 - Menampilkan opsi help yang difasilitasi oleh assistant
- 7) Fitur 7 - Mendefinisikan list kata penting terkait apakah itu merupakan suatu task atau tidak
- 8) Fitur 8 - Menampilkan pesan error jika pesan tidak dikenali
- 9) Fitur 9 - Memberikan rekomendasi kata apabila terdapat typo (BONUS)

Chatbot yang dibangun berbasis web dengan kaskas website yang digunakan adalah Flask (Python), backend bahasa Python, dan front-end nya menggunakan HTML, CSS, Javascript, dan jQuery. Data mengenai tugas-tugas disimpan dalam basis data menggunakan kaskas SQLite3. Pengetahuan tentang keyword-keyword disimpan dalam beberapa file .txt.

4. Implementasi dan Pengujian

4.1. Spesifikasi teknis program (struktur data, fungsi, prosedur yang dibangun)

1) Struktur data

Terdapat 2 struktur data yang digunakan dalam pengoperasian program ini. Pertama digunakan *database* yang menggunakan *sqlite* untuk menyimpan berbagai data tugas yang dimasukan oleh *user*. Kemudian disimpan juga berbagai kata penting/*keyword* dalam file berformat txt yang nanti akan di-*load* oleh program saat program dijalankan.

2) Fungsi dan Prosedur

Terdapat banyak fungsi pada program ini, yang dibagi menjadi empat bagian, yaitu:

a. Fungsi/prosedur untuk ekstraksi informasi

- find_date

Fungsi ini menerima input teks dan mengidentifikasi apabila terdapat tanggal pada teks tersebut. Tanggal-tanggal yang ditemukan akan dimasukkan ke dalam suatu list yang kemudian akan dikembalikan oleh fungsi.

- convert_string_to_date

Fungsi ini digunakan untuk mengubah tanggal dalam format string, misalnya “28 april 2021”, menjadi format YYYY-MM-DD yang dapat di-*compute* oleh library *datetime*.

- convert_to_date

Fungsi ini digunakan untuk mengubah tanggal dalam format DD/MM/YYYY menjadi format YYYY-MM-DD.

- find_task_id

Fungsi ini menerima input teks dan mengembalikan id dari task.

- find_course_id

Fungsi ini menerima input teks dan mengembalikan kode matakuliah.

- find_duration

Fungsi ini menerima input teks dan mengembalikan durasi pada teks, misalnya “2 minggu”.

- convert_duration_to_days

Fungsi ini berguna untuk mengubah durasi dari semula bisa bulan, minggu, hari menjadi dalam satuan hari dan yang dikembalikan bertipe integer.

- build_last

Fungsi ini digunakan untuk membuat list untuk menunjukkan *last occurence* dari setiap huruf pada suatu string.

- boyer_moore

Fungsi ini merupakan algoritma string matching yang digunakan pada banyak fungsi lainnya, dengan mencari keberadaan string pattern pada teks, dan akan mengembalikan indeks mulainya kesamaan pattern jika ditemukan, atau -1 jika tidak.

- find_update_keyword

Fungsi ini digunakan untuk mencari apakah ada update keyword pada teks menggunakan algoritma Boyer Moore, jika ada akan mengembalikan indeks kemunculan pertamanya. Deadline keyword terdefinisi pada teks “update_keywords.txt”.

- find_done_keyword

Fungsi ini digunakan untuk mencari apakah ada done keyword pada teks menggunakan algoritma Boyer Moore, jika ada akan mengembalikan indeks kemunculan pertamanya. Deadline keyword terdefinisi pada teks “done.txt”.

- find_help_keyword

Fungsi ini digunakan untuk mencari apakah ada help keyword pada teks menggunakan algoritma Boyer Moore, jika ada akan mengembalikan indeks kemunculan pertamanya. Deadline keyword terdefinisi pada teks “help_keywords.txt”.

- find_deadline_keyword

Fungsi ini digunakan untuk mencari apakah ada deadline keyword pada teks menggunakan algoritma Boyer Moore, jika ada akan mengembalikan indeks kemunculan pertamanya. Deadline keyword terdefinisi pada teks “deadline.txt”.

- levenshtein_distance

Fungsi ini berguna untuk menghitung Levenshtein Distance yang kemudian akan terpakai untuk menghitung similarity.

- similarity

Fungsi ini berguna untuk menghitung similaritas antara dua kata. Fungsi ini akan terpakai untuk fungsi `word_recommendation`.

- `word_recommendation`

Fungsi ini mengembalikan rekomendasi kata yang bertipe string apabila typo. Jika tidak typo, akan mengembalikan string kosong.

b. Fungsi/prosedur untuk mengelola basis data

- `add_tugas`

Fungsi ini menerima input berupa objek tugas. Kemudian tugas tersebut akan dimasukan ke *database*. Jika berhasil maka tugas akan mengembalikan Id dari tugas yang baru dimasukan tersebut sedangkan jika gagal maka akan mengembalikan Id tidak valid yaitu -99.

- `showAllTugas`

Fungsi ini berfungsi untuk mengembalikan semua data tugas yang ada dalam bentuk *array of tuple*. Fungsi ini juga tidak memiliki parameter input apapun.

- `showTugasFrom`

Fungsi ini berfungsi untuk mengembalikan semua tugas antara suatu waktu sampai waktu tertentu dimana kedua waktu tersebut merupakan parameter input dari fungsi ini.

- `showTugasbyDate`

Fungsi ini menerima masukan berupa satu tanggal. Kemudian akan dikembalikan semua tugas yang memiliki *deadline* di tanggal tersebut.

- `showTugasbyJenis`

Fungsi ini menerima masukan berupa jenis tugas. Kemudian akan dikembalikan semua tugas yang memiliki jenis sesuai dengan masukan tersebut.

- `showTugasbyMatkul`

Fungsi ini menerima masukan berupa kode mata kuliah. Kemudian akan dikembalikan semua tugas yang memiliki kode mata kuliah sesuai dengan masukan tersebut.

- `showTugasbyId`

Fungsi ini menerima masukan berupa id dari sebuah tugas dan kemudian mengembalikan tugas dengan id yang bersesuaian.

- isIdExist

Fungsi ini menerima masukan berupa id dan mengecek apakah id tersebut terdapat di database atau tidak. Fungsi ini mengembalikan nilai *boolean* berupa nilai *True* atau *False*.

- updateTanggal

Fungsi ini menerima masukan berupa id dan akan melakukan pembaruan terhadap tugas dengan id yang bersesuaian.

- tugasDone

Fungsi ini berfungsi untuk menandai suatu tugas apabila sudah selesai dikerjakan.

c. Fungsi/prosedur untuk mendeteksi fitur

Fitur ini parameter inputnya adalah user message (string), dan me-return dua tipe yaitu boolean, response (string) dalam bentuk tuple. Fitur dideteksi berdasarkan keyword-keyword yang diperoleh dari hasil ekstraksi informasi.

- isFitur1
- isFitur2
- isFitur3
- isFitur4
- isFitur5
- isFitur6
- get_bot_response

Fungsi ini merupakan *main function* yang akan digunakan pada aplikasi Flask.

Fungsi ini berisi kondisional mengenai pendeteksian fitur. Fungsi ini mereturn string berupa respons dari bot.

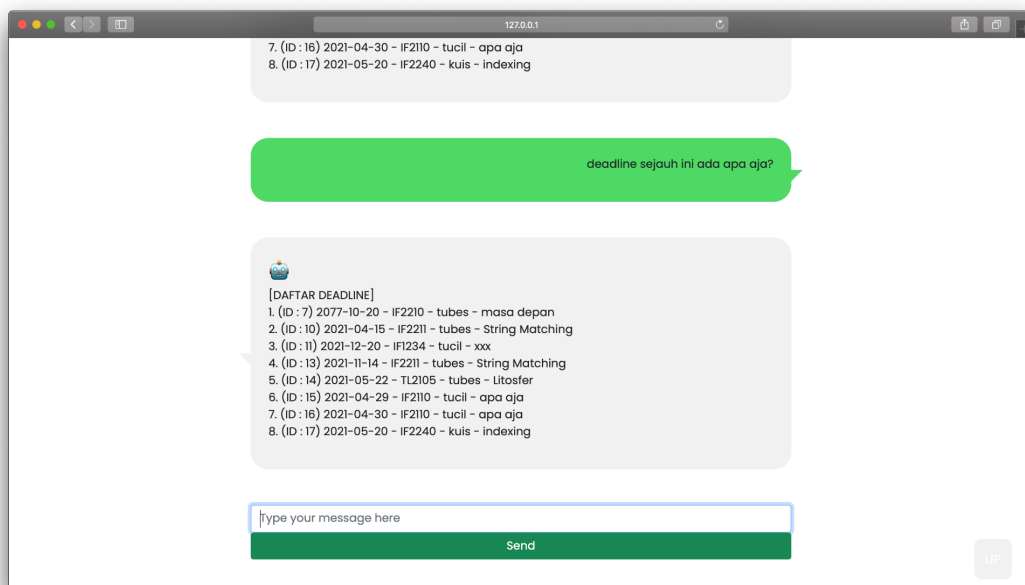
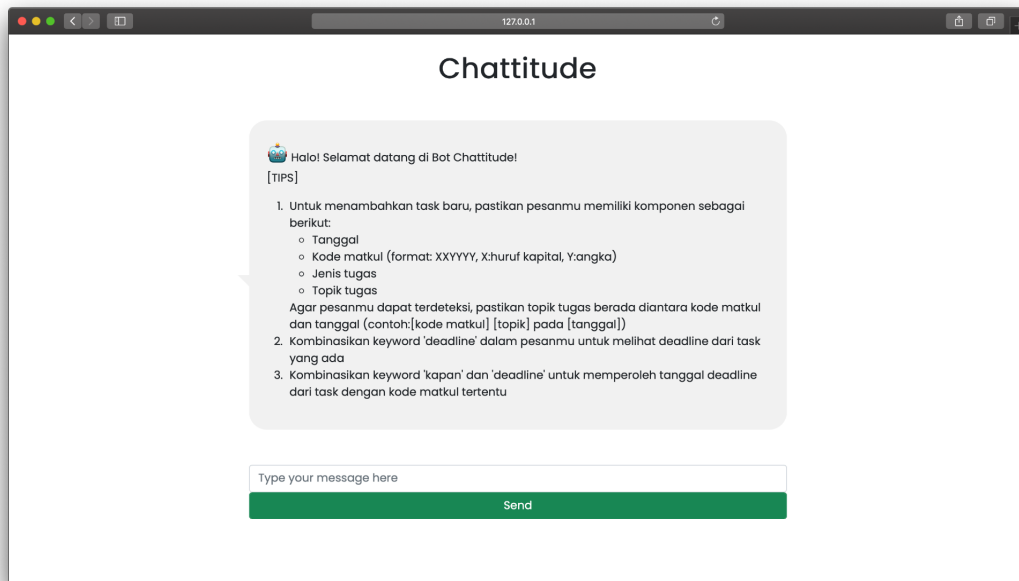
d. Fungsi untuk string formatting

Terdapat beberapa fungsi untuk string formatting, untuk membuat format output string task, me-replace kata-kata typo dengan kata-kata sebenarnya, dan lain-lain.

4.2. Penjelasan tata cara penggunaan program

1) Interface program

Interface program adalah seperti di bawah ini.



2) Fitur-fitur yang disediakan program

Tata cara penggunaan program adalah sebagai berikut:

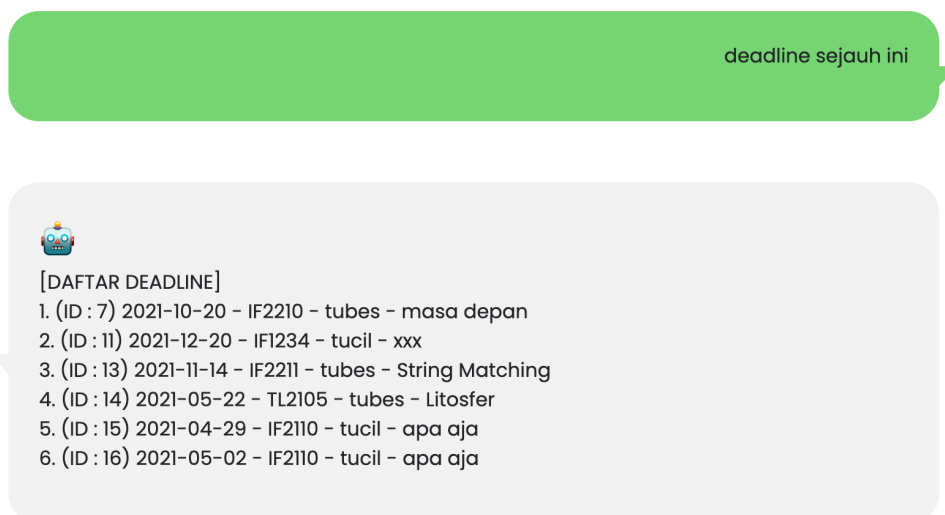
- Untuk menambahkan task baru, pastikan pesanmu memiliki komponen sebagai berikut: Tanggal (format: DD/MM/YYYY atau dalam bahasa Indonesia (contoh: “28 April 2021”, Kode matkul (format: XYYYYY, X:huruf, Y:angka), Jenis tugas, Topik tugas
- Agar pesanmu dapat terdeteksi, pastikan topik tugas berada diantara kode matkul dan tanggal (contoh:[kode matkul] [topik] pada [tanggal])

- Kombinasikan keyword 'deadline' dalam pesanmu untuk melihat deadline dari task yang ada
- Kombinasikan keyword 'kapan' dan 'deadline' untuk memperoleh tanggal deadline dari task dengan kode matkul tertentu

4.3. Hasil pengujian (screenshot antarmuka dan skenario yang memperlihatkan berbagai kasus yang mencakup seluruh fitur dalam Chatbot)

1) Fitur 1 - Menambahkan task baru

Kondisi awal:



Command untuk insert:



Setelah insert:

deadline sejauh ini



[DAFTAR DEADLINE]

1. (ID : 7) 2021-10-20 - IF2210 - tubes - masa depan
2. (ID : 11) 2021-12-20 - IF1234 - tucil - xxx
3. (ID : 13) 2021-11-14 - IF2211 - tubes - String Matching
4. (ID : 14) 2021-05-22 - TL2105 - tubes - Litosfer
5. (ID : 15) 2021-04-29 - IF2110 - tucil - apa aja
6. (ID : 16) 2021-05-02 - IF2110 - tucil - apa aja
7. (ID : 18) 2021-04-30 - IF2210 - tucil - design pattern

2) Fitur 2 - Melihat daftar task yang harus dikerjakan

deadline banyak bgt sih ada apa aja hiks T__T



[DAFTAR DEADLINE]

1. (ID : 7) 2077-10-20 - IF2210 - tubes - masa depan
2. (ID : 10) 2021-04-15 - IF2211 - tubes - String Matching
3. (ID : 11) 2021-12-20 - IF1234 - tucil - xxx
4. (ID : 13) 2021-11-14 - IF2211 - tubes - String Matching
5. (ID : 14) 2021-05-22 - TL2105 - tubes - Litosfer
6. (ID : 15) 2021-04-29 - IF2110 - tucil - apa aja
7. (ID : 16) 2021-04-30 - IF2110 - tucil - apa aja
8. (ID : 17) 2021-05-20 - IF2240 - kuis - indexing

3) Fitur 3 - Menampilkan deadline dari suatu task tertentu

deadline tubes IF2211 itu kapan deh?



2021-04-15 - tubes - String Matching
2021-11-14 - tubes - String Matching

4) Fitur 4 - Memperbaharui task tertentu

Kondisi awal:

deadline sejauh ini



[DAFTAR DEADLINE]

1. (ID : 7) 2077-10-20 - IF2210 - tubes - masa depan
2. (ID : 11) 2021-12-20 - IF1234 - tucil - xxx
3. (ID : 13) 2021-11-14 - IF2211 - tubes - String Matching
4. (ID : 14) 2021-05-22 - TL2105 - tubes - Litosfer
5. (ID : 15) 2021-04-29 - IF2110 - tucil - apa aja
6. (ID : 16) 2021-05-02 - IF2110 - tucil - apa aja

Command untuk update:

task 7 diubah deadline nya jadi 20/10/2021



Sukses memperbaharui deadline tugas dengan ID:7
dari tanggal 2077-10-20 menjadi 2021-10-20

Setelah update:

deadline sejauh ini



[DAFTAR DEADLINE]

1. (ID : 7) 2021-10-20 - IF2210 - tubes - masa depan
2. (ID : 11) 2021-12-20 - IF1234 - tucil - xxx
3. (ID : 13) 2021-11-14 - IF2211 - tubes - String Matching
4. (ID : 14) 2021-05-22 - TL2105 - tubes - Litosfer
5. (ID : 15) 2021-04-29 - IF2110 - tucil - apa aja
6. (ID : 16) 2021-05-02 - IF2110 - tucil - apa aja

- 5) Fitur 5 - Menandai bahwa suatu task sudah selesai dikerjakan
- Kondisi awal:

deadline sejauh ini



[DAFTAR DEADLINE]

1. (ID : 7) 2077-10-20 - IF2210 - tubes - masa depan
2. (ID : 10) 2021-04-15 - IF2211 - tubes - String Matching
3. (ID : 11) 2021-12-20 - IF1234 - tucil - xxx
4. (ID : 13) 2021-11-14 - IF2211 - tubes - String Matching
5. (ID : 14) 2021-05-22 - TL2105 - tubes - Litosfer
6. (ID : 15) 2021-04-29 - IF2110 - tucil - apa aja
7. (ID : 16) 2021-04-30 - IF2110 - tucil - apa aja
8. (ID : 17) 2021-05-20 - IF2240 - kuis - indexing

Command untuk update:

task 10 udah kelar



Sukses menandai task 10 selesai

Setelah diupdate:

deadline sejauh ini



[DAFTAR DEADLINE]

1. (ID : 7) 2077-10-20 - IF2210 - tubes - masa depan
2. (ID : 11) 2021-12-20 - IF1234 - tucil - xxx
3. (ID : 13) 2021-11-14 - IF2211 - tubes - String Matching
4. (ID : 14) 2021-05-22 - TL2105 - tubes - Litosfer
5. (ID : 15) 2021-04-29 - IF2110 - tucil - apa aja
6. (ID : 16) 2021-04-30 - IF2110 - tucil - apa aja
7. (ID : 17) 2021-05-20 - IF2240 - kuis - indexing

6) Fitur 6 - Menampilkan opsi help yang difasilitasi oleh assistant

ada command apa aja sih?



[FITUR]

1. Menambahkan task baru
2. Melihat daftar task yang harus dikerjakan
3. Menampilkan deadline dari suatu task tertentu
4. Memperbaharui task tertentu
5. Menandai bahwa suatu task sudah selesai dikerjakan
6. Menampilkan opsi help yang difasilitasi oleh assistant
7. Mendefinisikan list kata penting terkait apakah itu merupakan suatu task atau tidak
8. Menampilkan pesan error jika pesan tidak dikenali
9. Memberikan rekomendasi kata apabila ada typo dari user

[DAFTAR KATA PENTING]

1. Kuis
2. Ujian
3. Tugil
4. Tubes
5. Praktikum

7) Fitur 7 - Mendefinisikan list kata penting terkait apakah itu merupakan suatu task atau tidak

ada kata penting apa aja sih di bot ini?



[FITUR]

1. Menambahkan task baru
2. Melihat daftar task yang harus dikerjakan
3. Menampilkan deadline dari suatu task tertentu
4. Memperbaharui task tertentu
5. Menandai bahwa suatu task sudah selesai dikerjakan
6. Menampilkan opsi help yang difasilitasi oleh assistant
7. Mendefinisikan list kata penting terkait apakah itu merupakan suatu task atau tidak
8. Menampilkan pesan error jika pesan tidak dikenali
9. Memberikan rekomendasi kata apabila ada typo dari user

[DAFTAR KATA PENTING]

1. Kuis
2. Ujian
3. Tupil
4. Tubes
5. Praktikum

8) Fitur 8 - Menampilkan pesan error jika pesan tidak dikenali

ajsdflkladsfhka ngasal ngasalll



Maaf, pesanmu ga bisa dikenali nih

[TIPS]

1. Untuk menambahkan task baru, pastikan pesanmu memiliki komponen sebagai berikut:
 - Tanggal (format: DD/MM/YYYY atau dalam bahasa Indonesia)
 - Kode matkul (format: XYYYYY, X:huruf, Y:angka)
 - Jenis tugas
 - Topik tugas

Agar pesanmu dapat terdeteksi, pastikan topik tugas berada diantara kode matkul dan tanggal (contoh:[kode matkul] [topik] pada [tanggal])
2. Kombinasikan keyword 'deadline' dalam pesanmu untuk melihat deadline dari task yang ada
3. Kombinasikan keyword 'kapan' dan 'deadline' untuk memperoleh tanggal deadline dari task dengan kode matkul tertentu

9) Fitur 9 - Memberikan rekomendasi kata apabila terdapat typo

deadlain tubez IF2210 kapan??



Mungkin maksudmu: "deadline tubes IF2210 kapan??"

deadline tubes IF2210 kapan?



2077-10-20 - tubes - masa depan

10) Fitur tambahan

Kami membuat fitur tambahan yaitu jika user memasukkan kalimat seperti sedang demot atau sedih, bot akan memberikan semangat.

sedih bgt kok tugas banyak banget sih mau nangis



Semangat yaaa pasti bisa kok :D

4.4. Analisis hasil pengujian

Dari hasil pengujian, bot dapat melaksanakan semua fitur dengan baik. Respons dari bot juga relatif cepat. Hal ini dikarenakan pengetahuan bot masih sangat sederhana dan data-data yang disimpan juga relatif sedikit sehingga bot lebih mudah untuk melakukan *load* terhadap data-data yang ada.

5. Kesimpulan dan Saran

5.1. Kesimpulan

Algoritma string matching seperti KMP, Boyer-Moore, dan Regular Expression sangat berguna dalam pembangunan chatbot. Algoritma string matching ini dapat mengekstraksi pesan dari pengguna, sehingga tidak diperlukan pesan yang sama persis namun dengan menggunakan keyword bot dapat menjalankan fitur yang diinginkan pengguna. Selain itu, pendeteksian typo dapat dilakukan menggunakan algoritma Levenshtein Distance yang mengukur kemiripan antara dua buah string.

5.2. Saran

Saran dalam pengerjaan tugas ini adalah lebih mengeksplorasi berbagai kemungkinan pesan yang akan dimasukan oleh *user* sehingga bot dapat memberikan respon yang lebih sesuai kepada berbagai pesan. Selain itu dapat juga dilakukan eksplorasi yang lebih dalam membangun tampilan web agar web dapat terlihat lebih menarik dan informatif. Selain itu, bot juga dapat dibuat lebih interaktif.

5.3. Refleksi

Melalui pengerjaan tugas ini, kami menjadi tahu bagaimana proses pembuatan chatbot yang banyak digunakan sekarang ini. Kami juga dapat mengimplementasikan materi dari mata kuliah Basis Data dalam perancangan *database* yang berfungsi untuk menyimpan informasi tugas. Selain itu, kami juga dapat lebih mengeksplorasi bagaimana membangun tampilan web dengan *html*. Kami juga memperoleh wawasan lebih tentang algoritma pencocokan *string* serta cara pengaplikasiannya dalam kehidupan nyata.

Daftar Pustaka

1. [Understanding the Levenshtein Distance Equation for Beginners | by Ethan Nam | Medium](#)
2. [Building a Chatbot in Python using Flask - Tutorial \(codinginfinite.com\)](#)
3. [Pencocokan String \(itb.ac.id\)](#)
4. [String Matching dengan Regular Expression \(itb.ac.id\)](#)