

LAPORAN HASIL TUGAS BESAR 3
Penerapan String Matching dan Regular Expression dalam
Pembangunan Deadline Reminder Assistant “Deadline Chatbot”

DIAJUKAN UNTUK MEMENUHI TUGAS MATA KULIAH
IF2211 - Strategi Algoritma
SEMESTER II TAHUN 2020/2021



Disusun oleh :

Kelompok :Deadline-Tubes

Muhammad Fahmi Alamsyah	13519077
Widya Anugrah Putra	13519105
Muhammad Jafar Gundari	1351919

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

DAFTAR ISI

DAFTAR ISI	2
BAB I	3
1.1. Deskripsi Tugas	3
1.2. Spesifikasi Tugas	3
BAB II	5
2.1. Dasar Teori	5
2.2. Apa itu chatbot	6
BAB III	7
3.1. Langkah-Langkah Pemecahan Masalah	7
3.2. Fitur Fungsional dan Arsitektur Chatbot	9
BAB IV	10
4.1. Spesifikasi teknis program	10
4.2. Tata Cara Penggunaan Program	25
4.3. Hasil Pengujian	26
4.4 Analisis Hasil Pengujian	31
BAB V	32
5.1. Kesimpulan	32
5.2. Saran	32
5.3 Refleksi dan Komentar	32
DAFTAR PUSTAKA	33

BAB I

DESKRIPSI MASALAH

1.1 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.

1.2 Spesifikasi Tugas

1. Aplikasi yang dibuat berbasis web (wajib) dan anda dapat menggunakan salah satu kakas website: PHP, Flask, Django, JavaScript.
2. Aplikasi (backend) harus menggunakan algoritma pencocokan string KMP, Boyer-Moore, dan Regex dengan menggunakan bahasa yang menunjang regular expression: Java, Javascript, PHP, Python.
3. Penyimpanan data-data dan pengetahuan yang diperlukan oleh Chatbot bisa didefinisikan melalui 2 cara (pilih salah satu), yaitu:
 - a. Membuat suatu database sederhana (penerapan Basis Data dalam Strategi Algoritma). Implementasi skema database (relasi, atribut) dibeaskan. Skema basis data tidak perlu dinormalisasi.
 - b. Menyimpannya dalam bentuk struktur data sendiri, pengambilan data dilakukan dengan menggunakan mekanisme load / save dari suatu file .txt. Struktur penyimpanan data dibeaskan.
4. Data-data yang diperlukan dan akan disimpan dalam suatu chatbot adalah sebagai berikut.
 - a. List kata-kata penting
 - b. Daftar task yang tercatat oleh Chatbot
 - c. Data-data pendukung lainnya (kreativitas kelompok)

5. Pencocokan string dapat anda implementasikan sesuai kriteria berikut.
- a. Deteksi perintah (contoh: “Apa saja deadline yang ada sejauh ini?”) tidak dilakukan secara exact matching (input dibebaskan ke user --bukan programmer-- selama mengandung kata kunci tertentu), anda dapat memanfaatkan regular expression dan string matching untuk mencari kata kunci dan melakukan pencocokan.
 - b. Rekomendasi kata: pencocokan exact matching (KMP, Boyer-Moore) dimanfaatkan untuk menentukan tingkat kemiripan suatu kata di perintah. Anda dapat mengembangkan algoritma yang telah diajarkan untuk menentukan kemiripan string.
 - c. Pengekstrakan nilai-nilai berjenis numerik dan tanggal dilakukan dengan memanfaatkan Regular Expression.

BAB II

LANDASAN TEORI

2.1 Dasar Teori

a. *Knuth-Morris-Pratt Algorithm*

Algoritma KMP adalah algoritma pencocokan string yang hampir sama dengan algoritma Brute Force, yaitu mencari pola dalam string secara terurut dari kiri ke kanan, tetapi memiliki algoritma pergeseran yang lebih baik ketika terjadi *mismatch*. Ketika terjadi *mismatch* pada index ke- j dari pola, algoritma ini hanya akan menggeser sejauh panjang string dikurangi panjang prefix terpanjang pola[0.. j -1] yang juga suffix dari pola[1.. j -1]. Pergeseran ini dihitung menggunakan border function, yang intinya juga sama seperti algoritma KMP sendiri.

b. *Boyer-Moore Algorithm*

Berbeda dengan algoritma Brute Force dan KMP, algoritma Boyer-Moore ini menggunakan 2 jenis teknik yang disatukan. Teknik yang pertama adalah teknik *looking-glass*, yaitu teknik yang mencari pola dalam teks dengan mengiterasi pola dari belakang ke depan. Teknik kedua adalah teknik lompatan yang dibagi menjadi 3 kasus:

- Jika terjadi *mismatch* pada indeks i dari teks dan indeks j dari pola dan karakter teks yang mengakibatkan *mismatch* misalnya x dan ada last occurrence dari x pada pola yang berada di kiri indeks j , maka geser pola sehingga last occurrence dari x pada pola sejajar dengan x pada teks.
- Jika terjadi *mismatch* lagi, tetapi last occurrence berada di sebelah kanan indeks j , maka geser pola ke kanan sejauh 1 blok saja
- Jika terjadi *mismatch*, tetapi pola tidak memiliki karakter pada teks yang mengakibatkan *mismatch*, geser pola sehingga indeks ke-0 pola sejajar dengan indeks ke $i+1$ teks.

c. *Regex*

Regular Expression adalah sebuah teknik string matching yang dikembangkan berdasarkan teori bahasa formal. Regex memiliki Finite State

Machine Model yang memodelkan mesin abstrak yang mampu menerima input, yaitu teks yang akan diuji oleh pola mesinnya, dan mengeluarkan 2 jenis output, ditolak atau diterima. FSM yang dimodelkan memiliki beberapa states, dengan beberapa di antaranya berupa accepting states, jika teks mengakibatkan mesin berubah ke salah satu accepting state, maka mesin menerima inputan. Sebaliknya mesin akan menolak inputannya.

2.2 Apa itu chatbot

Chatbot adalah sebuah program komputer berbasis AI (Artificial Intelligence), alias robot virtual yang dapat mensimulasikan percakapan layaknya manusia. Teknologi ini juga dikenal sebagai asisten digital yang dapat memahami serta memproses permintaan pengguna, dan memberikan jawaban yang relevan dengan cepat. Chatbot ini dapat diaplikasikan pada website, aplikasi desktop maupun android/ios. Beberapa perusahaan besar seperti facebook, line, telegram atau discord sudah banyak memilikinya Deadline reminder bot adalah sebuah chatbot yang memiliki fungsi sebagai pengingat untuk user terhadap *deadline* atau tenggat waktu tugas tertentu yang sudah didaftarkan oleh user sebelumnya. Bot ini cukup pintar dengan dapat memahami kata-kata dari user secara natural.

BAB III

ANALISIS PEMECAHAN MASALAH

3.1 Langkah-Langkah Pemecahan Masalah

Pertama-tama kita membagi pesan user ke beberapa kasus seperti yang telah disediakan oleh spesifikasi tugas, kasus-kasus tersebut adalah:

1. Menambah task baru
2. Melihat daftar task
3. Menampilkan deadline suatu task
4. Memperbarui task
5. Menandai task sudah selesai
6. Menampilkan apa saja yang bisa dilakukan chat bot
7. Memberikan pesan error jika ada kata yang tidak dikenali
8. Memberikan sugesti kepada typo terhadap kata-kata yang dianggap

penting (Bonus)

Setelah dibagi menjadi beberapa kasus, kami mengidentifikasi kombinasi-kombinasi kata kunci yang unik untuk tiap kasus, sehingga jika pesan user sesuai dengan kombinasi kata kunci pada suatu kasus, maka bot akan mengenali fitur mana yang diminta oleh user. Kata kunci tugas yang disediakan oleh bot ini antara lain : tubes, tucil, pr, praktikum, ujian, tugas, dan kuis. Dalam menentukan apakah suatu masukan kata cocok dengan kata kunci menggunakan algoritma booye moore dan regular expression. Sedangkan untuk pengecekan typo digunakan algoritma levenshtein.

a. Menambah task baru

Kombinasi kata kunci unik yang didapatkan untuk kasus ini adalah terdapat kata kunci task/tugas yang dikenali bot, lalu memiliki kode perkuliahan yang valid (di ITB, kode perkuliahannya diawali dengan 2 huruf dan diakhiri 4 digit angka), dan memiliki tanggal valid (belum dilewati hari ini) yang menjadi deadline tugas bersangkutan. Tanggal harus didahului kata “pada” untuk memisahkan topik tugas dengan tanggal. Jika semua kombinasi kata kunci ini terpenuhi, maka bot akan mengenali pesan user sebagai permintaan menambah task.

b. Menampilkan daftar task

Kombinasi kata kunci unik yang didapatkan untuk kasus ini adalah terdapat kata “deadline”, terdapat kosakata/kata yang bersifat periode waktu, terdapat kata perintah “tampilkan” atau “apa saja” atau “daftar”, dan kata kunci opsional yang berupa kata kunci tugas. Jika kata yang bersifat periode waktu ternyata “antara”,

maka tanggal yang dimasukkan harus berformat YYYY-MM-DD atau YYYY/MM/DD. Jika kosakata yang bersifat periode waktu ternyata periodik, seperti “hari ke depan” atau “minggu ke depan” maka kosakata tersebut wajib didahului besar periodenya. Jika semua kombinasi kata kunci ini terpenuhi, maka bot akan mengenali pesan user sebagai permintaan melihat daftar task.

c. Menampilkan deadline suatu task

Kombinasi kata kunci unik yang didapatkan untuk kasus ini adalah terdapat kata “deadline” atau “kapan”, terdapat kode perkuliahan yang valid (di ITB, kode perkuliahannya diawali dengan 2 huruf dan diakhiri 4 digit angka), dan mengandung kata kunci task/tugas yang dikenali bot. Jika semua kombinasi kata kunci ini terpenuhi, maka bot akan mengenali pesan user sebagai permintaan menampilkan deadline suatu task tertentu.

d. Memperbarui deadline suatu task

Kombinasi kata kunci unik untuk menggunakan fitur ini adalah mengandung salah satu kata 'diundur', 'jadwal ulang', 'perbarui', kemudian mengandung kata ‘deadline’, mengandung tanggal deadline baru yang berformat YYYY/MM/DD atau YYYY-MM-DD. Dan terakhir mengandung kata “task X” dengan X adalah nomor Id dari suatu task.

e. Menandai task sudah selesai

Kombinasi kata kunci unik yang didapatkan untuk kasus ini adalah terdapat kata “sudah selesai”, “beres”, “hapus”, lalu mengandung kata “task” yang diikuti dengan id dari task bersangkutan. Jika semua kombinasi kata kunci ini terpenuhi, maka bot akan mengenali pesan user sebagai permintaan penghapusan task. Jika task yang dicari tidak ditemukan, bot akan memunculkan pesan kesalahan.

f. Menampilkan apa saja yang bisa dilakukan chat bot

Kombinasi kata kunci unik yang didapatkan untuk kasus ini adalah terdapat kombinasi kata kunci “bisa”, dengan “lakukan” dan “apa” atau “ngapain”. Urutan kombinasi tidak dipedulikan, misal “Apa aja yang bisa dilakukan?” atau “Yang bisa kamu lakukan apa aja sih?” dan sebagainya. Jika semua kombinasi kata kunci ini terpenuhi, maka bot akan mengenali pesan user sebagai permintaan apa fitur-fitur yang dimiliki bot.

g. Menampilkan kalimat-kalimat yang tidak dikenali

Jika pesan tidak diterima oleh kasus-kasus lain, maka bot akan mengeluarkan pesan kesalahan bahwa bot tidak mengenali pesan tersebut.

h. Memberikan sugesti pada typo kata-kata yang dianggap penting

Sebelum pesan dianggap sebagai pesan yang tidak dikenali, bot mengecek apakah terjadi typo pada kata-kata yang menjadi kata kunci pada kasus-kasus di atas. Jika ternyata iya, maka bot akan memberikan rekomendasi.

3.2 Fitur Fungsional dan Arsitektur Chatbot

Arsitektur Chatbot :

```
Struktur folder / file aplikasi
assets
(berisi gambar pendukung)
Listdeadline
----- datadeadline.php
----- index.php
Readme
----- index.html
----- stylereadme.css
Src
----- booyermoore.php
----- database.php
----- function.php
Index.php
message.php
style.css
```

Chatbot memiliki 3 halaman. Yakni Index.php pada root folder yang berisi aplikasi chatbot langsung dapat melakukan percakapan. Index.php pada folder Listdeadline yang menampilkan halaman yang menampilkan deadline saat ini yang terdaftar. Terakhir index.html pada folder Readme yakni halaman yang berisi penjelasan tentang fitur program beserta kata kunci untuk menggunakannya.

Setiap fungsi/prosedur yang digunakan untuk mengolah kata masukan (pencocokan string) disimpan pada file function.php. Khusus untuk algoritma boyermoore disimpan pada file booyermoore.php.

Untuk menyimpan data deadline masukan dari pengguna, menggunakan database mysql. Inisialisasi database terdapat pada database.php. Database ini terdiri dari satu tabel chatbot (Id, Deadline, Subjects, Keyword, Topic)

Fitur-fitur berikut disimpan pada satu file function.php namun dimodularisasi.

Menggunakan regular expression dan string matching booyer moore. Lebih jelas pada implementasi bab 4.

- Menambah task baru
- Menampilkan daftar task
- Menampilkan deadline suatu task
- Memperbarui deadline suatu task
- Menandai task sudah selesai
- Menampilkan apa saja yang bisa dilakukan chat bot

- Menampilkan kalimat-kalimat yang tidak dikenali
- Memberikan sugesti pada typo kata-kata yang dianggap penting

BAB IV

IMPLEMENTASI DAN PENGUJIAN

4.1 Spesifikasi teknis program

4.1.1 Fitur tambah task

```
function isAddTask($input){
    if (preg_match("/hapus|hilangkan|sudah|hilang/i", $input)) return false;
    global $bulan;
    global $jenistask;
    $isThereTask = -1;
    $message = strtolower($input);
    foreach ($jenistask as $kata){
        if ($isThereTask == -1){
            $isThereTask = booyermoore($kata, $message);
        }
    }
    if ($isThereTask == -1) return false;
    if (! preg_match("/\b[a-z]{2}\d{4}\b/i", $input)) return false;
    if (! preg_match("/pada/", $message)) return false;
    $isdateusenamemonth = false;
    foreach ($bulan as $namabulan){
        if ($isdateusenamemonth == false){
            $pattern =
"/(pada\s+(\d{1,2})\s+(\d{1,2})\s+(\d{4}))|(pada\s+tanggal\s+(\d{1,2})\s+(\d{1,2})\s+(\d{4}))/i";
            $isdateusenamemonth = preg_match($pattern,$message,$date);
        }
    }
    if ($isdateusenamemonth){
        // echo "please ga nyampe sini";
        if (preg_match("/tanggal/", $message)){
            $day = (int) $date[5];
            $year = (int) $date[7];
            $month = $date[6];

        }else{
            $day = (int) $date[2];
            $year = (int) $date[4];
            $month = $date[3];
        }
    }else if
(preg_match("/(pada\s+(\d{1,2}))([\-\/\.\s]) (\d{2})\g{-2} (\d{4}))|(pada\s+tangg
al\s+(\d{1,2}))([\-\/\.\s]) (\d{2})\g{-2} (\d{4}))/i", $message)){
        // echo "please ga nyampe sini";

preg_match("/(pada\s+(\d{1,2}))([\-\/\.\s]) (\d{2})\g{-2} (\d{4}))|(pada\s+tangga
l\s+(\d{1,2}))([\-\/\.\s]) (\d{2})\g{-2} (\d{4}))/i", $message, $date);
        if (preg_match("/tanggal/", $message)){
            $day = (int) $date[7];
            $year = (int) $date[10];
            $month = $date[9];
        }else{
            $day = (int) $date[2];
```

```

        $year = (int) $date[5];
        $month = $date[4];
    }
    }else return false;
    // echo "helo";
    if ($isdateusenamemonth){
        $bulans = -1;
        for ($i = 0; $i < 12; $i++){
            if ($month == $bulan[$i]){
                $bulans = $i+1;
            }
        }
    }
    else $bulans = $month;
    return validateDate($day,$bulans, $year) && (date("Y-m-d") <=
date("Y-m-d",mktime(0,0,0,$bulans,$day,$year)));
}

```

Fungsi `isAddTask` ini mengecek apakah kombinasi kata kunci yang diperlukan untuk menambahkan task sudah valid atau tidak. Jika valid maka akan mereturn `true`, sebaliknya jika ada satu kata kunci yang salah maka akan mereturn `false`. String matching yang digunakan adalah Boyer-Moore dan Regex. Selain itu tanggal yang masuk juga dicek apakah valid atau tidak sekarang.

```

function addTask($input){
    $lowerinput = strtolower($input);
    $idxjenistask = -1;
    global $jenistask;
    global $bulan;
    foreach ($jenistask as $kata){
        if ($idxjenistask == -1){
            $idxjenistask = booyermooore($kata, $lowerinput);
        }
    }
    $jenistugas = "";
    while($input[$idxjenistask] != " "){
        $jenistugas .= $input[$idxjenistask];
        $idxjenistask++;
    }
    //aman sampai nyimpen jenis tugas
    preg_match("/^b[a-z]{2}\d{4}\b/i", $input,$match);
    $kodekuliah = $match[0];
    $isdateusenamemonth = false;
    foreach ($bulan as $namabulan){
        if ($isdateusenamemonth == false){

```

```

    $pattern = "/(pada (\d{1,2}) (\".$namabulan.\") (\d{4}))[pada tanggal (\d{1,2})
(\".$namabulan.\") (\d{4})/i";
    $isdateusenamemonth = preg_match($pattern,$lowerinput,$date);
    }
}
if ($isdateusenamemonth){
    if (preg_match("/tanggal/", $lowerinput)){
        $day = (int) $date[5];
        $year = (int) $date[7];
        $month = $date[6];

    }else{
        $day = (int) $date[2];
        $year = (int) $date[4];
        $month = $date[3];
    }
}
}
preg_match("/(pada\s+(\d{1,2})([\-\/\.\s])(\d{2}))\g{-2}(\d{4}))(pada\s+tanggal\s+(\d{
1,2})([\-\/\.\s])(\d{2}))\g{-2}(\d{4})/i", $lowerinput, $date);
    if (preg_match("/tanggal/", $lowerinput)){
        $day = (int) $date[7];
        $year = (int) $date[10];
        $month = $date[9];
    }else{
        $day = (int) $date[2];
        $year = (int) $date[5];
        $month = $date[4];
    }
}
}
if ($isdateusenamemonth){
    $bulans = -1;
    for ($i = 0; $i < 12; $i++){
        if ($month == $bulan[$i]){
            $bulans = $i+1;
        }
    }
}
}
else $bulans = $month;
$d = mktime(0,0,0,$bulans,$day,$year);
$stopik = "";
$indexpada = booyermoore("pada", $lowerinput)-1;
$indexkode = booyermoore($kodekuliah,$input);
$iterate = $indexkode + 7;
if ($iterate <= $indexpada){
    while ($iterate != $indexpada){

```

```

    $topik .= $input[$iterate];
    $iterate++;
  }
}

global $tabelname;
$query = "INSERT INTO ".$tabelname."(Deadline,Subjects,Keyword,Topic)
VALUES(\"\".date("Y-m-d",$d)."\", \"\".$kodekuliah.\"\",
\"\".$jenistugas.\"\", \"\".$topik.\"\")";
return $query;
}

```

Fungsi `addTask` ini hampir mirip algoritmanya dengan fungsi `isAddTask`. Hanya saja fungsi ini mengambil semua nilai dari kolom-kolom yang dibutuhkan database. Lalu menghasilkan query yang digunakan untuk menambahkan record ke database. Fungsi ini hanya akan dipanggil jika `isAddTask` true.

4.1.2 Melihat daftar task

```

function isShowTask($input)
{
    $message = strtolower($input);
    $katapentingtampilkan = array('apa saja', 'tampilkan', 'daftar');
    global $timeperiod;

    $kata = 'deadline';
    if (!preg_match("/^b" . $kata . "\b/i", $message)) {
        return false;
    }
    $tampilkanfound = false;
    foreach ($katapentingtampilkan as $kata) {
        if (preg_match("/^b" . $kata . "\b/i", $message)) {
            $tampilkanfound = true;
            break;
        }
    }

    if ($tampilkanfound) {
        foreach ($timeperiod as $kata) {
            if (preg_match("/^b" . $kata . "\b/i", $message)) {
                return true;
            }
        }
    }
}

```

```

    }
  }
  return false;
}

```

Fungsi `isShowTask` berguna untuk mengecek apakah pesan user sesuai dengan kombinasi kata kunci untuk kasus melihat daftar task. Jika sudah sesuai maka fungsi akan mereturn `true`, dan sebaliknya `false`.

```

function getTimePeriodWord($input)
{
  $message = strtolower($input);
  global $timeperiod;
  foreach ($timeperiod as $kata) {
    if (preg_match("/^b" . $kata . "\b/i", $message)) {
      return $kata;
    }
  }
}

```

Fungsi `getTimePeriodWord` berguna untuk mengambil kata bersifat periode waktu yang digunakan di dalam pesan. Fungsi ini diambil jika pesan sudah lolos `isShowTask`

```

function getShowQuery($input, $kata)
{
  global $tabelname;
  $message = strtolower($input);
  $query = "";
  if ($kata == 'sejauh ini' || $kata == 'sampai saat ini') {
    $query = 'SELECT * FROM ' . $tabelname . ' WHERE Deadline >= \' ' .
    date("Y-m-d") . '\';
    // $query = 'SELECT * FROM chatbot WHERE Deadline >= \' ' . date("Y-m-d") . '\';
  } else if ($kata == 'minggu ke depan' || $kata == 'hari ke depan') {
    // get number N
    $index = booyermoore($kata, $message);

    // skip for whitespace
    $index--;
    while ($message[$index] == ' ') {
      $index--;
    }
  }
}

```

```

// go to the first number char
while ($message[$index] != ' ') {
    $index--;
}
$index++;
$num = "";
// take num
for ($i = $index; $i < strlen($message); $i++) {
    $num .= $message[$i];
    if ($message[$i] == ' ') {
        break;
    }
}

if (strlen($num) > 0 && (int)$num > 0) {
    $num = (int)$num; // convert num
    if ($kata == 'minggu ke depan') {
        $datetime = strtotime("+ $num Weeks");
    } else {
        // days
        $datetime = strtotime("+ $num Days");
    }
    $query = 'SELECT * FROM ' . $tablename . ' WHERE Deadline BETWEEN \' ' .
date("Y-m-d") . ' \' AND ' . ' ' . date("Y-m-d", $datetime) . ' ' . ' ';
}
} else if ($kata == 'hari ini') {
    $query = 'SELECT * FROM ' . $tablename . ' WHERE Deadline = \' ' . date("Y-m-d")
. ' ' . ' ';
} else if ($kata == 'antara') {
    $message = str_replace('/', '-', $message); // jika masukan format menggunakan /
convert menjadi -
    preg_match_all("(\\d{1,4}-\\d{1,2}-\\d{1,4})", $message, $arraydate);
    if (count($arraydate[0]) == 2) {
        $query = 'SELECT * FROM ' . $tablename . ' WHERE Deadline BETWEEN \' ' .
$arraydate[0][0] . ' \' AND ' . ' ' . $arraydate[0][1] . ' ' . ' ';
    }
} else if ($kata == 'semua') {
    $query = 'SELECT * FROM ' . $tablename;
    return $query;
}

// cek apakah mengandung spesifik task
// $jenistask = array('kuis', 'tubes', 'tucil', 'ujian', 'praktikum');
global $jenistask;
$hashtask = false;
foreach ($jenistask as $task) {

```



```

if (preg_match("/\b" . $task . "\b/i", $message)) {
    $hastask = true;
    $query .= ' AND Keyword = \'' . $task . '\'';
    break;
}
}
if (!$hastask) {
    $query .= ',';
}
return $query;
}

```

Fungsi `getShowQuery` berfungsi untuk mengambil data-data yang ada pada pesan user, lalu mengembalikan query yang sesuai untuk menampilkan datanya.

4.1.3 Menampilkan deadline suatu task

```

function isDeadline($input)
{
    $message = strtolower($input);
    $katauncideadline = array('kapan', 'deadline');
    $diundur = false;
    foreach ($katauncideadline as $task) {
        if (preg_match("/" . $task . "\b/i", $message)) {
            $diundur = true;
            break;
        }
    }

    $kata = 'deadline';
    // cek tanggal dan ada kata deadline
    if (!$diundur || !preg_match("/[a-z A-Z]{2}[\d]{4}/", $message) ||
        !preg_match('/kuis|tubes|tucil|tugas|ujian|praktikum/i', $message)) {
        return false;
    }
    return true;
}

```

Fungsi `isDeadline` hanya mengecek apakah pesan meminta bot untuk menampilkan deadline suatu task tertentu. Pembuatan querynya berada pada main program.

4.1.4 Memperbarui deadline suatu task

```

function isDelayTask($input)
{
    $message = strtolower($input);
    $katakuncidiundur = array('diundur', 'jadwal ulang', 'perbarui');
    $diundur = false;
    foreach ($katakuncidiundur as $task) {
        if (preg_match("/" . $task . "\b/i", $message)) {
            $diundur = true;
            break;
        }
    }

    $kata = 'deadline';
    // cek tanggal dan ada kata deadline
    $message = str_replace('/', '-', $message); // jika masukan format menggunakan /
    convert menjadi -
    if (!$diundur || !preg_match("/^b" . $kata . "\b/i", $message) ||
    !preg_match('(\\d{1,4}-\\d{1,2}-\\d{1,4})', $message)) {
        return false;
    }
    return true;
}

```

Fungsi isDelayTask berguna untuk mengecek apakah pesan user berisi kombinasi kata kunci yang sesuai dengan kasus memperbarui deadline suatu task.

```

function getDelayQuery($input)
{
    global $tabelname;
    $input = str_replace('/', '-', $input); // jika masukan format menggunakan / convert
    menjadi -
    // ambil tanggal
    preg_match('(\\d{1,4}-\\d{1,2}-\\d{1,4})', $input, $matches);
    $tanggal = $matches[0];
    $message = strtolower($input);
    $index = booyermoore('task', $message);

    // go to the first number char
    $index += 4;
    while ($message[$index] == ' ') {
        $index++;
    }
    $num = "";
}

```

```

// take num
for ($i = $index; $i < strlen($message); $i++) {
    $num .= $message[$i];
    if ($message[$i] == ' ') {
        break;
    }
}

$num = (int)$num;
$query = "";
if ($num > 0) {
    $tanggal = strtotime($tanggal);
    if (date("Y-m-d", $tanggal) > date("Y-m-d")) {
        // tanggal baru harus lebih besar dari hari ini
        $query = 'UPDATE ' . $tablename . ' SET Deadline = \' ' . date("Y-m-d", $tanggal) .
        '\ ' WHERE Id = ' . $num . ' ';
    } else {
        $query = 'Tanggal tidak valid';
    }
}
return $query;
}

```

Fungsi `getDelayQuery` mengembalikan query yang sesuai dengan pesan yang diterima. Fungsi ini akan dipanggil jika pesan yang diterima sudah memasuki kasus mengupdate deadline task.

4.1.5 Menandai task sudah selesai

```

function isdeletetask($input)
{
    $katapentingdelete = array('sudah selesai', 'beres', 'hapus');

    foreach ($katapentingdelete as $kata) {
        if (preg_match("/^b" . $kata . "\b/i", $input)) {
            return true;
        }
    }
    return false;
}

```

Fungsi `isdeletetask` mengecek apakah pesan user termasuk dalam kasus user menandai task sudah selesai atau tidak.

```

function getinputtaskid($input)
{
    $input = strtolower($input);
    $katapenting = array('task');
    foreach ($katapenting as $kata) {
        $index = booyermooore($kata, $input);
        if ($index != -1) {
            $panjangkata = strlen($kata);
            $index = $panjangkata + $index + 1;
            $num = "";

            // skip jika ada spasi lebih sebelum nomor id
            while ($input[$index] == ' ') {
                $index++;
            }

            for ($i = $index; $i < strlen($input); $i++) {
                $num .= $input[$i];
                if ($input[$i] == ' ') {
                    break;
                }
            }
            // jika tidak valid atau nomor id tidak ada, mengembalikan nilai 0
            return (int)$num;
        }
    }
    return -1; // tidak ditemukan
}

```

Fungsi `getinputtaskid` mengembalikan id task yang akan dihapus. Fungsi ini akan dipanggil jika pesan sudah dipastikan memasuki kasus menandai task sudah selesai.

4.1.6 Menampilkan apa saja yang bisa dilakukan chat bot

```

function isAskingForHelp($input){
    return preg_match("/(=?.*bisa)((=?.*lakukan)(=?.*apa)|(=?.*ngapain))\\w*/i",$input) ?
    true : false;
}

```

Fungsi `isAskingForHelp` mengecek apakah pesan masuk dalam kasus menampilkan apa saja yang bisa dilakukan chatbot. Jika return true, maka program utama akan menampilkan daftar fitur botnya.

4.1.7 Memberikan sugesti pada typo kata-kata yang dianggap penting

```

function lev ($str1, $str2){
    $size1 = strlen($str1);
    $size2 = strlen($str2);
    $matriks = array();
    for ($i = 0 ; $i < $size2+1; $i++){
        $dummy = array();
        for ($j = 0; $j < $size1+1; $j++){
            array_push($dummy,($i==0 || $j==0 ? max($i,$j) : 0));
        }
        array_push($matriks,$dummy);
    }
    // var_dump($matriks);
    for ($i = 1 ; $i <= $size2; $i++){
        for ($j = 1; $j <= $size1; $j++){
            if (min($i,$j) == 0){
                $matriks[$i][$j] = max($i,$j);
            }else{
                $matriks[$i][$j] = min(min($matriks[$i][$j-1]+1,$matriks[$i-1][$j]+1),
                $matriks[$i-1][$j-1] + ($str1[$j-1] == $str2[$i-1] ? 0 : 1));
            }
        }
    }
    // var_dump($matriks);
    return $matriks[$size2][$size1];
}

```

Fungsi lev mengembalikan Levenshtein Distance dari 2 string yang diterima.

```

function isTypo($str1,$str2){
    $lev = lev($str1,$str2);
    $maks = max(strlen($str1), strlen($str2));
    return $lev <= ($maks/4) && $lev > 0;
}

```

Fungsi isTypo memanfaatkan fungsi lev untuk menentukan apakah ada typo diantara kedua kata tersebut. Typo terjadi ketika kemiripan diatas 75%.

```

function isThereTypo($input){
    $data = preg_split('/ +/', $input);
    global $katakunciall;
}

```

```

for ($i = 0 ; $i < sizeof($data); $i++){
    for ($j = 0; $j < sizeof($katakunciall); $j++){
        if (isTypo($data[$i], $katakunciall[$j])){
            return true;
        }
    }
}
return false;
}

```

Fungsi isThereTypo mengecek apakah input mempunyai kata yang mirip dengan semua kata kunci tetapi ada typo sedikit. Memanfaatkan fungsi isTypo.

```

function replaceTypo($input){
    $lowerinput = strtolower($input);
    $data = preg_split('/ +/', $lowerinput);
    $dataasli = preg_split('/ +/', $input);
    $arraytypo = array();
    global $katakunciall;
    for ($i = 0 ; $i < sizeof($data); $i++){
        for ($j = 0; $j < sizeof($katakunciall); $j++){
            if (isTypo($data[$i], $katakunciall[$j])){
                $data[$i] = $katakunciall[$j];
                array_push($arraytypo, $i);
            }
        }
    }
    $ouput = "";
    $j = 0;
    for ($i = 0; $i < sizeof($data); $i++){
        if ($j < count($arraytypo) && $i == $arraytypo[$j]){
            $ouput .= "<i>". $data[$i]. "</i>";
            $j++;
        } else $ouput .= $dataasli[$i];
        $ouput .= " ";
    }
    return $ouput;
}

```

Fungsi replaceTypo menghapus semua typo dan digantikan dengan rekomendasi kata yang sesuai dengan kata kunci yang berkaitan.

4.1.8 Main program

```

<?php
include 'src/function.php';
// connecting to database
$conn = mysqli_connect("localhost", "root", "", "bot") or die("Database
Error");

```

```

// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}
// getting user message through ajax
$getMesg = mysqli_real_escape_string($conn, $_POST['text']);

// ===== MULAI MAIN PROGRAM
=====
if (isAddTask($getMesg)) {
    global $tabelname;
    $query = addTask($getMesg);
    $result = $conn->query($query);
    $query = "SELECT * FROM " . $tabelname . " WHERE Id = (SELECT max(Id) from "
    . $tabelname . ")";
    $result = $conn->query($query);
    $row = $result->fetch_assoc();
    echo "Task berhasil dicatat\n (ID: " . $row["Id"] . " ) " . $row["Deadline"]
    . " - " . $row['Keyword'] . " - " . ($row["Topic"] == "" ? "" : $row["Topic"]
    . " - ") . $row["Subjects"] . "<br><br>";
} else if (isAskingForHelp($getMesg)) {
    echo "[Fitur]\n<br><br>
    1. Menambahkan task baru\n<br>
    2. Menghapus task dengan id tertentu\n<br>
    3. Menampilkan task\n<br>
    4. Memperbarui task\n<br>
    5. Menampilkan deadline suatu task\n<br><br>
    \n<br>
    [Daftar kata penting]\n<br>
    1. kuis\n<br>
    2. tubes\n<br>
    3. tucil\n<br>
    4. ujian\n<br>
    5. praktikum\n<br>
    6. tugas\n<br>
    7. pr\n<br>
    \n<br><br>
    Lebih lengkapnya baca readme : ntar kasi link readme ke sini";
} else if (isdeletetask($getMesg)) {
    $nomorId = getinputtaskid($getMesg);
    if ($nomorId == 0) {
        echo "Nomor id ga bisa aku temuin, atau kamu cari nomor id 0 dimana itu
        tidak ada";
    } else if ($nomorId == -1) {
        echo 'Aku engga ngerti perintah kamu';
    } else {
        $deletequery = 'DELETE FROM chatbot WHERE Id = ' . $nomorId;
        if ($conn->query($deletequery) === TRUE) {
            if ($conn->affected_rows > 0) {
                echo "Id $nomorId berhasil dihapus :D, keren semangat terus ya!";
            } else {
                echo "Maaff, id yang kamu cari gaada tuh :(";
            }
        } else {
            echo "Terjadi kesalahan pada database";
        }
    }
}
} else if (isShowTask($getMesg)) {
    $kata = getTimePeriodWord($getMesg);
    $query = getShowQuery($getMesg, $kata);
    if ($query == ';') {
        echo 'Pesan tidak dikenali';
    }
}

```

```

} else {
    $result = $conn->query($query);
    if (mysqli_num_rows($result) > 0) {
        // output data of each row
        echo "[Daftar Deadline]<br><br>";
        while ($row = $result->fetch_assoc()) {
            echo "(ID: " . $row["Id"] . ") " . $row["Deadline"] . " - " .
$row["Subjects"] . " - " . $row["Keyword"] . " - " . $row["Topic"] .
"<br><br>";
        }
    } else {
        echo "Tidak ada";
    }
}

} else if (isDelayTask($getMesg)) {
    $query = getDelayQuery($getMesg);
    if ($query == '') {
        echo 'Tambahkan id task pada pesan ya, contoh: (task X) X adalah id';
    } else if ($query == "Tanggal tidak valid") {
        echo "Tanggal yang kamu masukan tidak valid, tanggal baru harus lebih
besar dari hari ini";
    } else {
        $result = $conn->query($query);
        if ($result && $conn->affected_rows > 0) {
            echo 'Berhasil memperbarui task';
        } else {
            echo "Gagal memperbarui task, id tidak ada atau deadline yang kamu
masukan salah";
        }
    }
}

} else if (isDeadline($getMesg)) {
    preg_match("/[a-z A-Z]{2}[\d]{4}/", $getMesg, $matches2);
    preg_match("/kuis|tubes|tucil|tugas|ujian|praktikum|pr/i", $getMesg,
$matches3);
    // echo $matches2[0];
    // echo $matches3[0];
    $query = "SELECT * FROM chatbot WHERE Subjects = '$matches2[0]' AND Keyword
= '$matches3[0]'";
    if ($query == '') {
        echo 'Pesan tidak dikenali';
    } else {
        $result = $conn->query($query);
        if ($result->num_rows > 0) {
            // output data of each row
            while ($row = $result->fetch_assoc()) {
                echo $row["Deadline"];
            }
        } else {
            echo "Tidak ada";
        }
    }
}

} else if (strtolower($getMesg) == 'halo' || strtolower($getMesg) == 'hai') {
    echo "Halo halo, tanya dong aku bisa ngapain aja";
} else if (isThereTypo((strtolower($getMesg)))) {
    $recommend = replaceTypo($getMesg);
    echo "Mungkin maksud kamu <br> $recommend";
} else {
    echo "Pesan tidak dikenali";
}

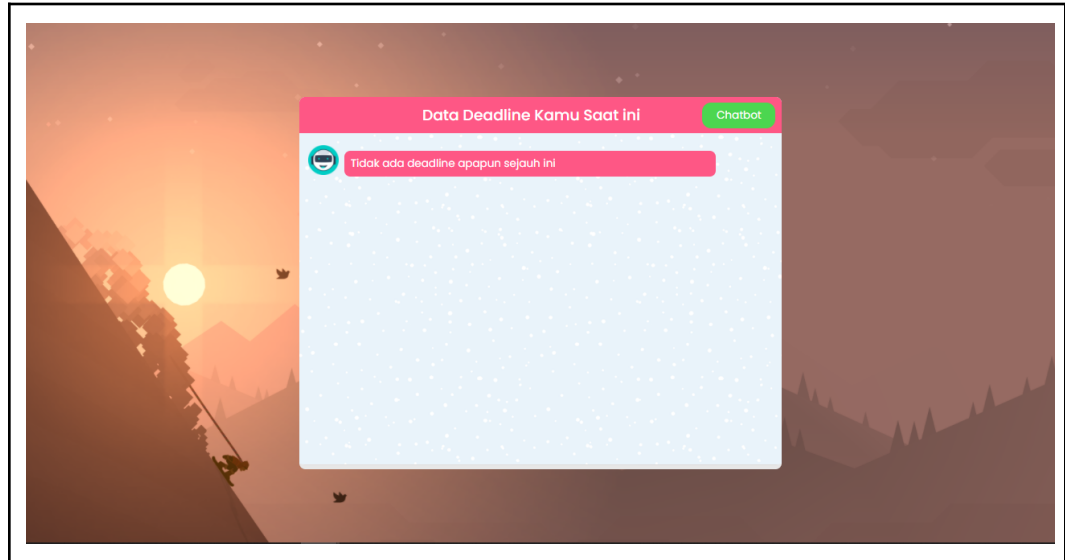
mysqli_close($conn);

```

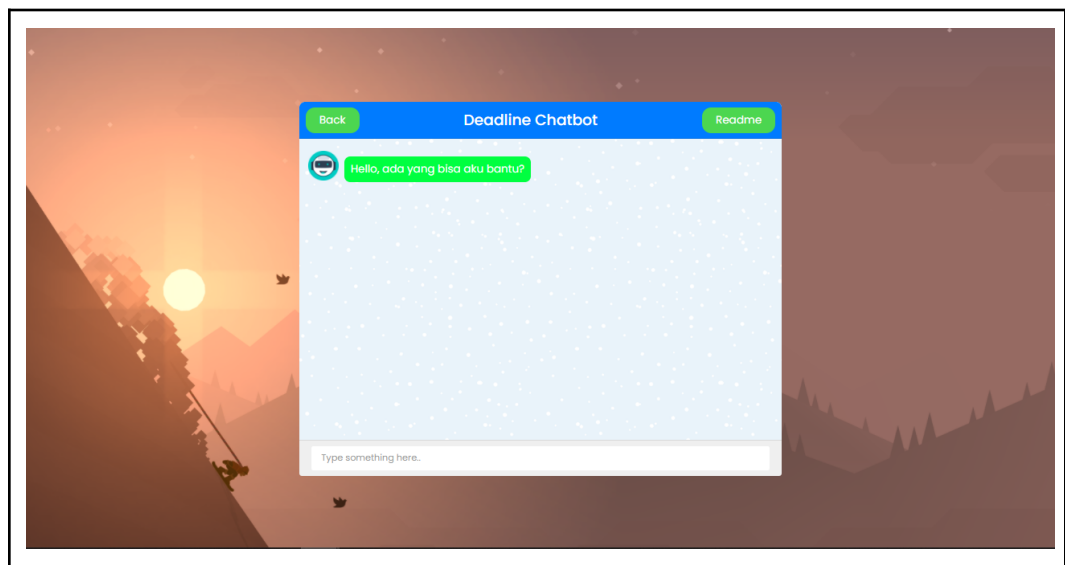

4.2 Tata Cara Penggunaan Program

1. Interface

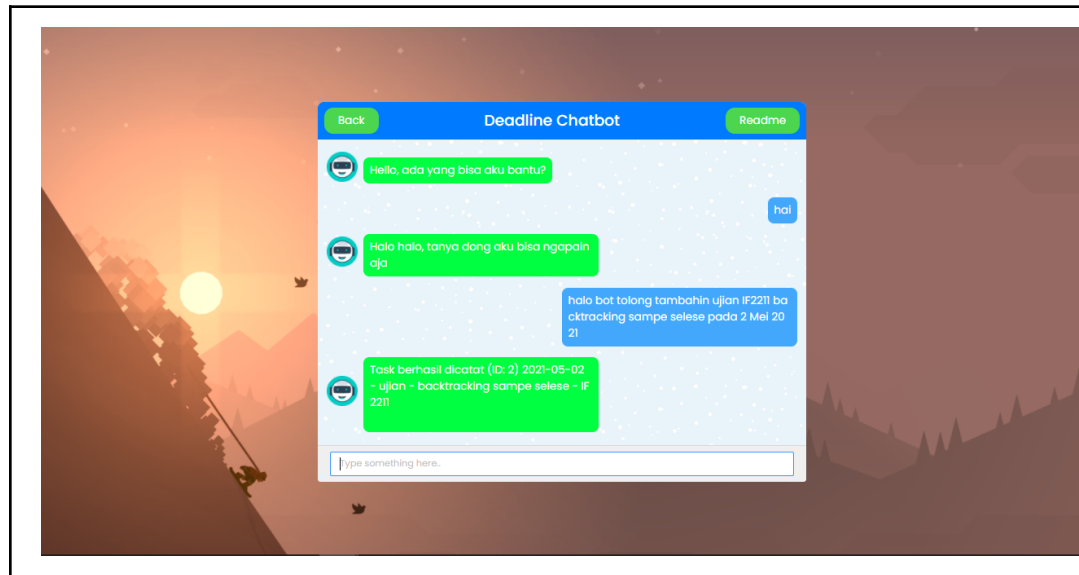
Halaman utama, pengguna melihat daftar deadline yang ada. Untuk mulai memasuki chatbot, dapat menekan tombol “chatbot” pada kanan atas.



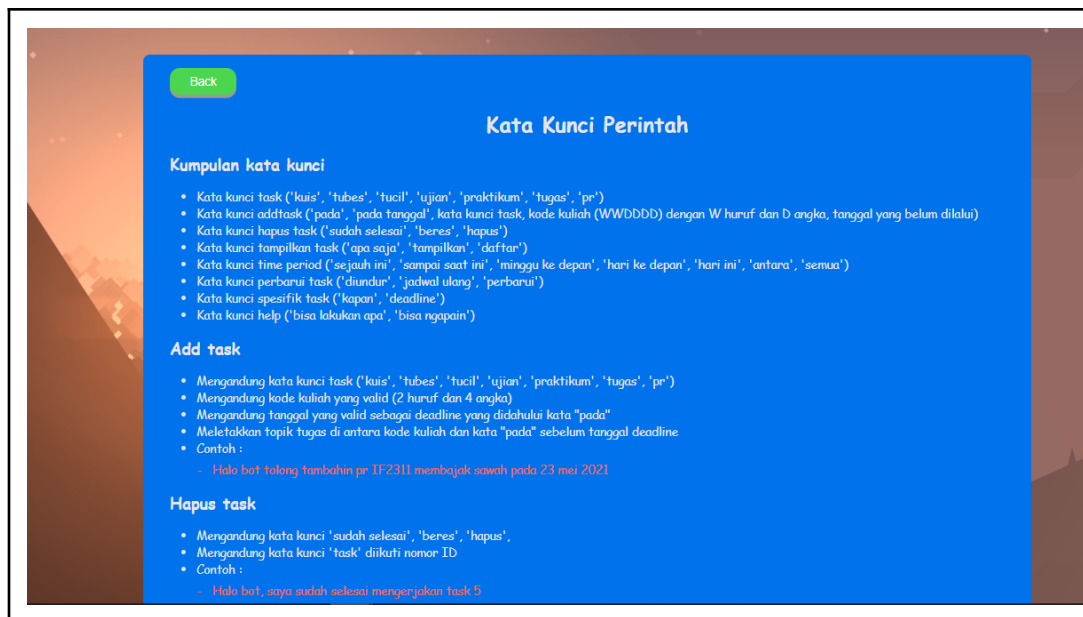
Halaman chatbot, bagian ini pengguna dapat bertanya kepada Deadline Chatbot pada message box dibawah. Kemudian terdapat tombol back untuk kembali ke halaman utama. Dan tombol readme untuk menuju halaman readme dimana berisi penjelasan kata kunci untuk tiap fitur.



Contoh tampilan bot menerima masukan



Halaman readme, terdapat tombol back untuk kembali ke chatbot

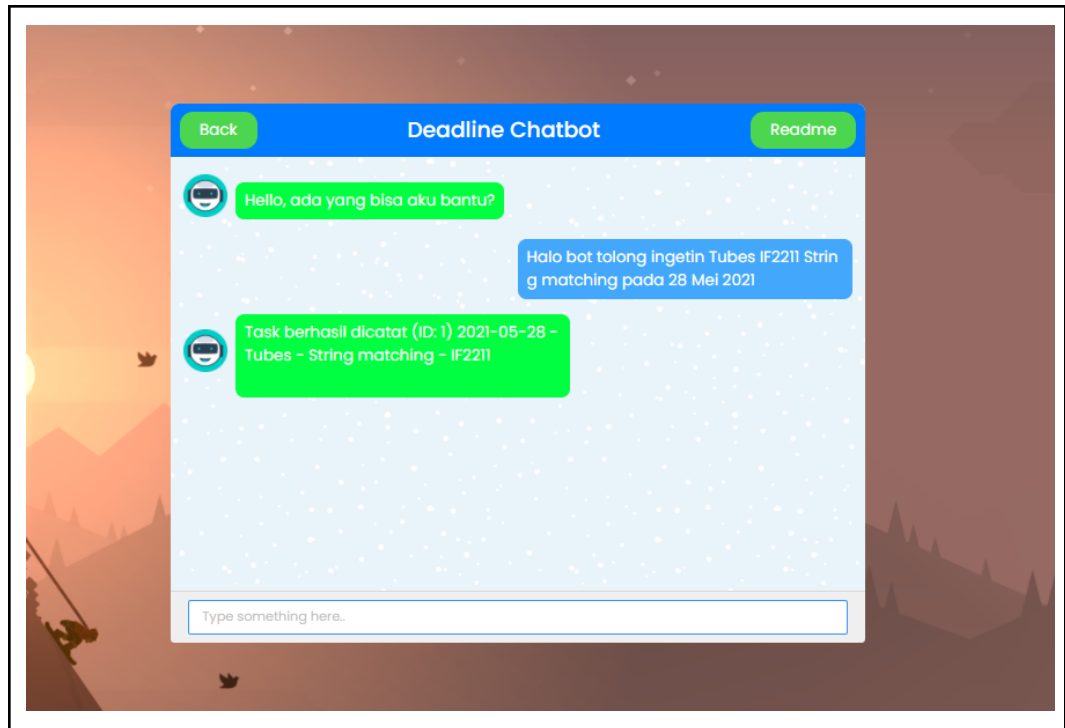


2. Fitur-fitur

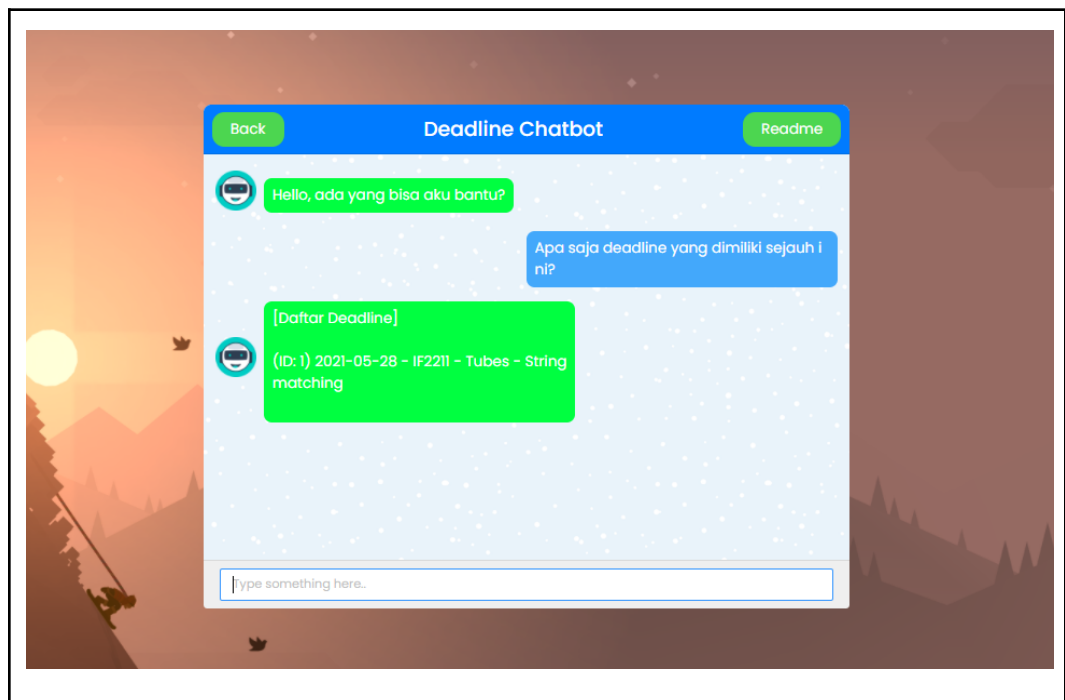
Sesuai dengan yang telah didefinisikan pada bab 3.1 mengenai kata kunci untuk tiap fitur. Asalkan masukan pengguna memenuhi syarat tersebut. Maka permintaan akan diproses oleh chatbot.

4.3 Hasil Pengujian

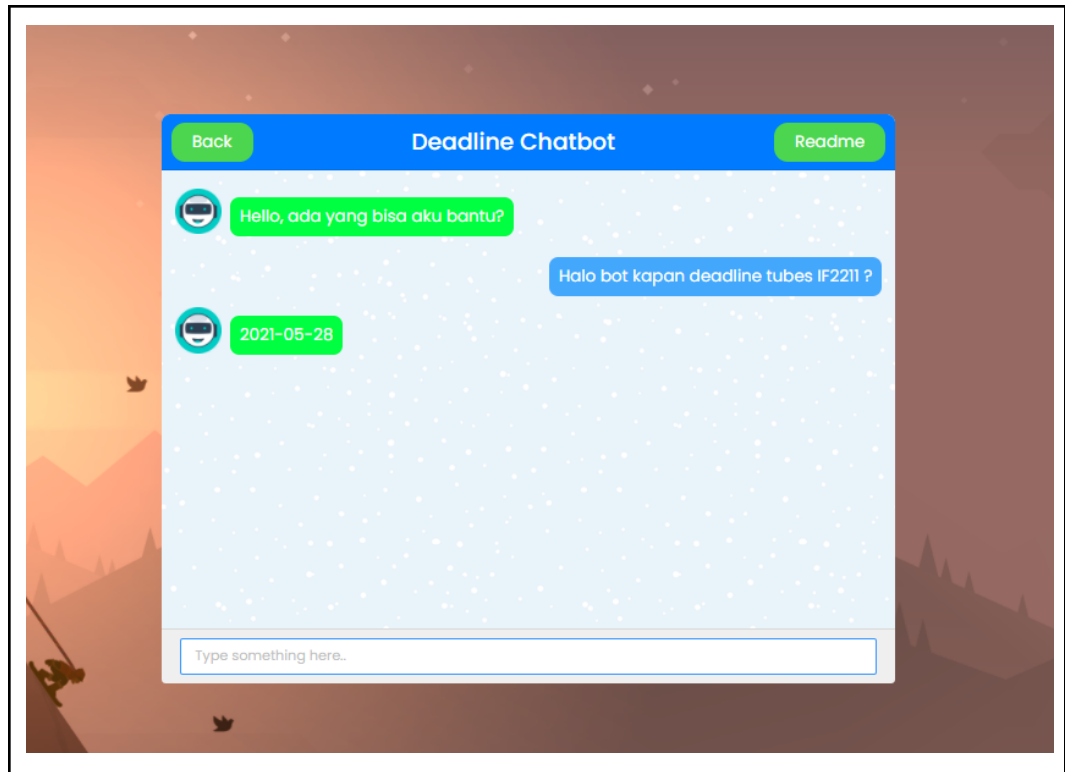
- **Pengujian Tambah Task**



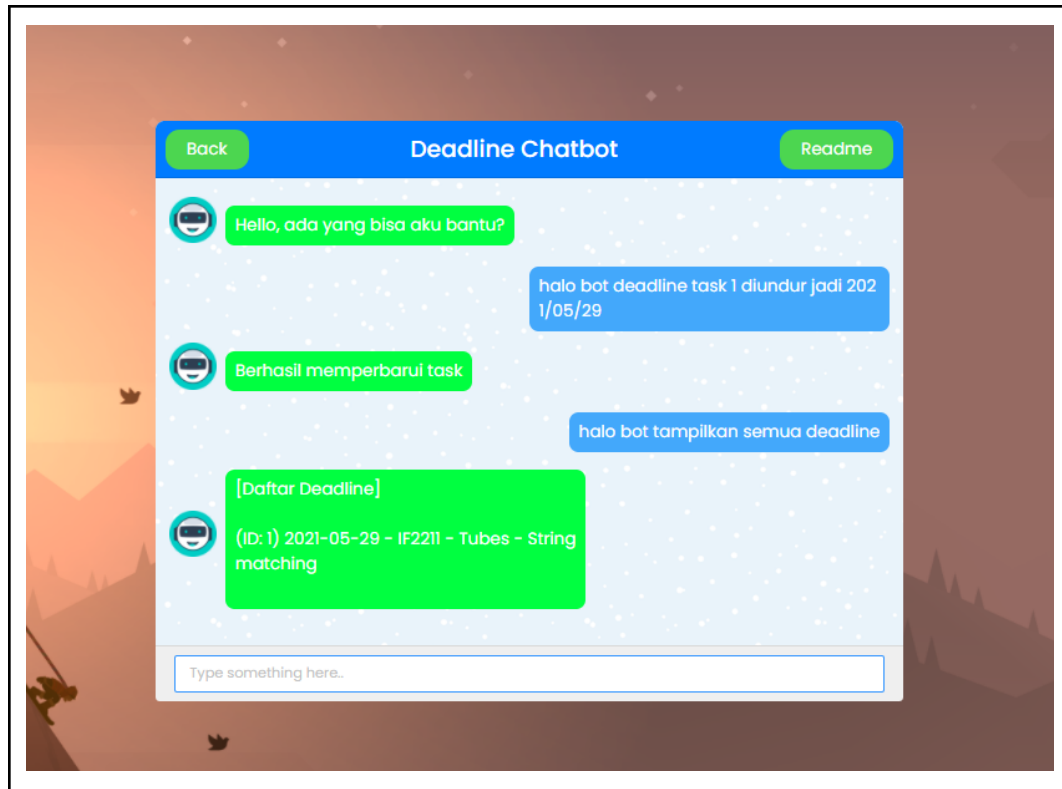
- **Melihat daftar task**



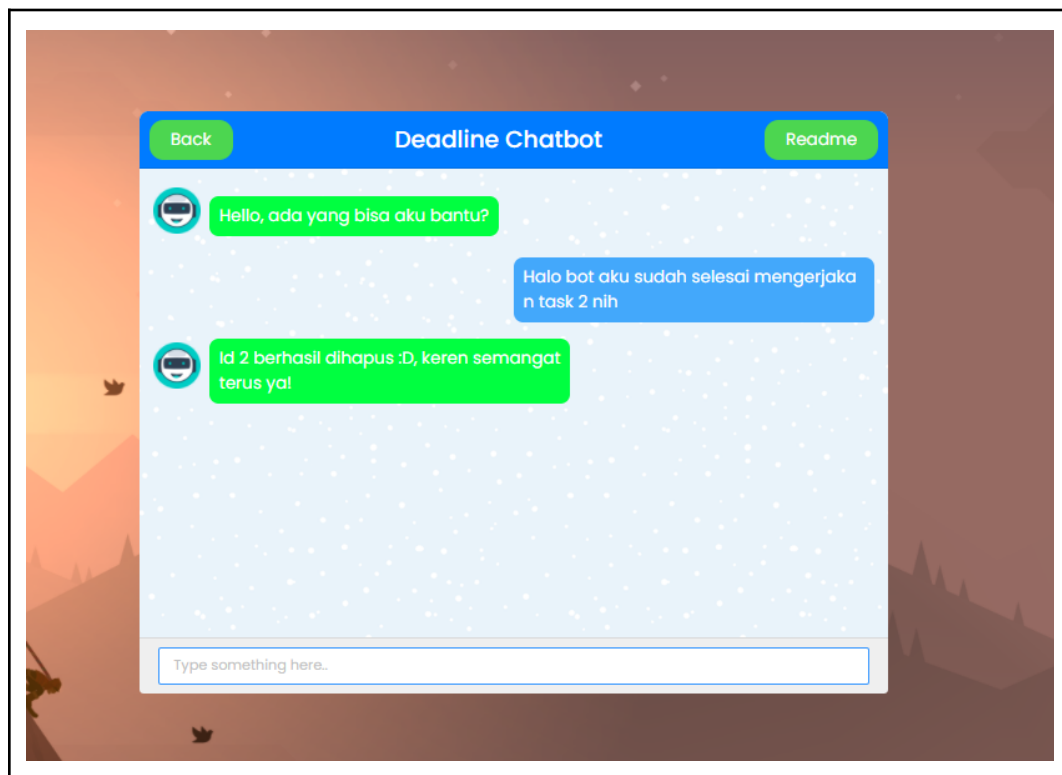
- **Menampilkan deadline suatu task**



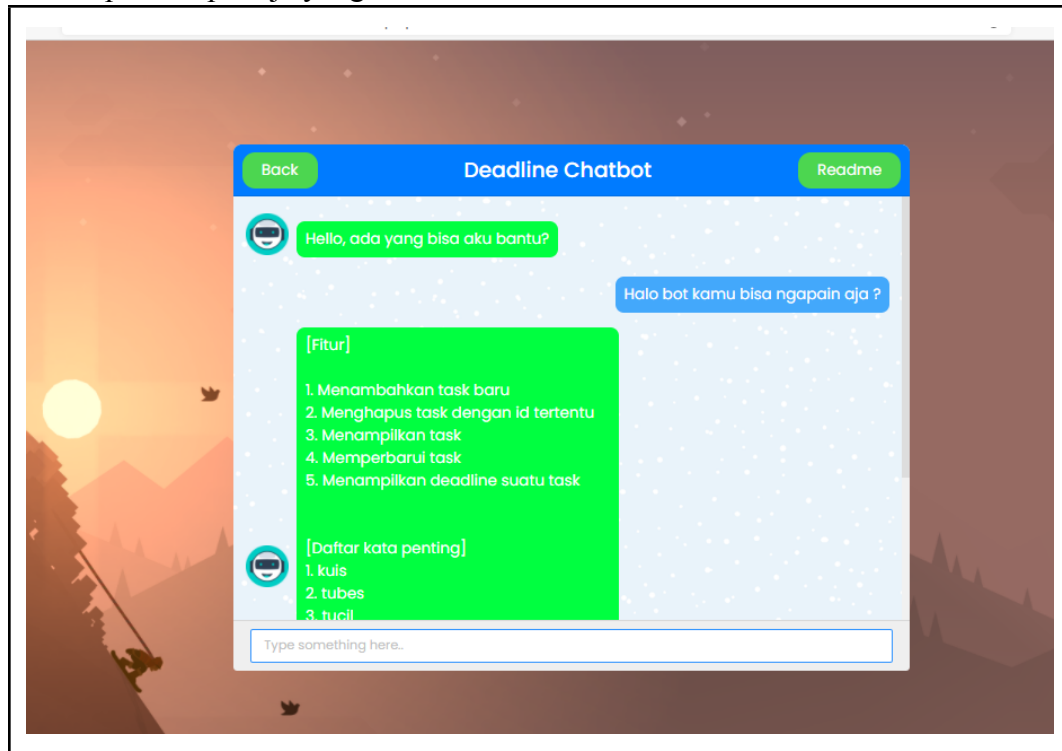
- Memperbarui deadline suatu task



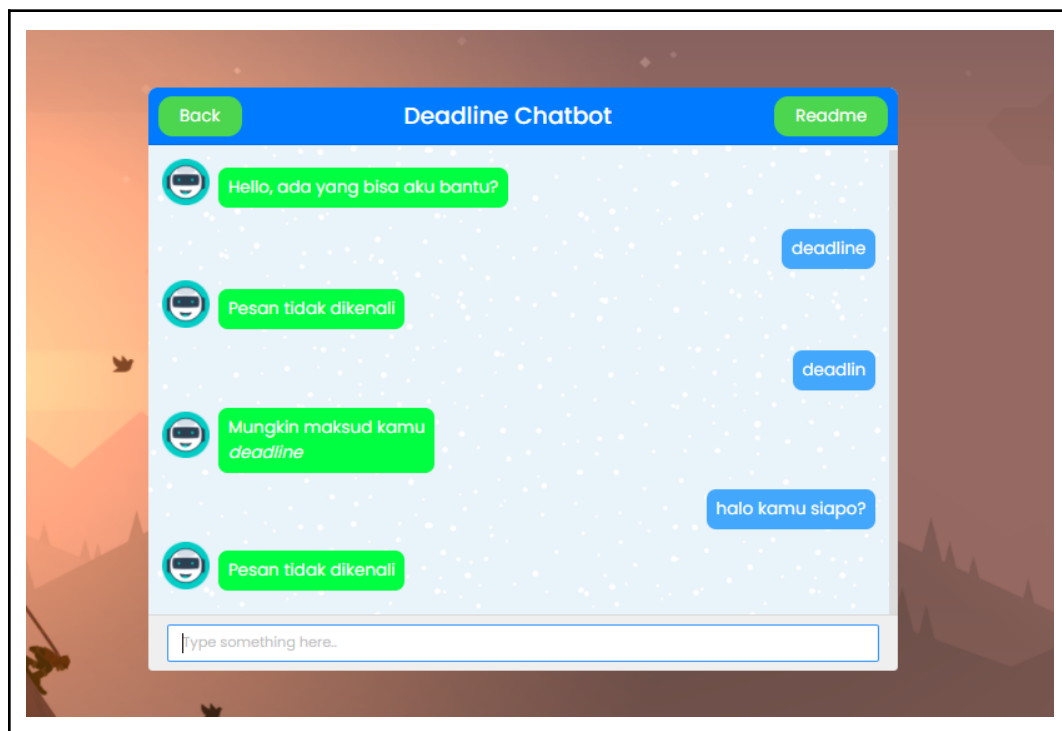
- Menandai task sudah selesai



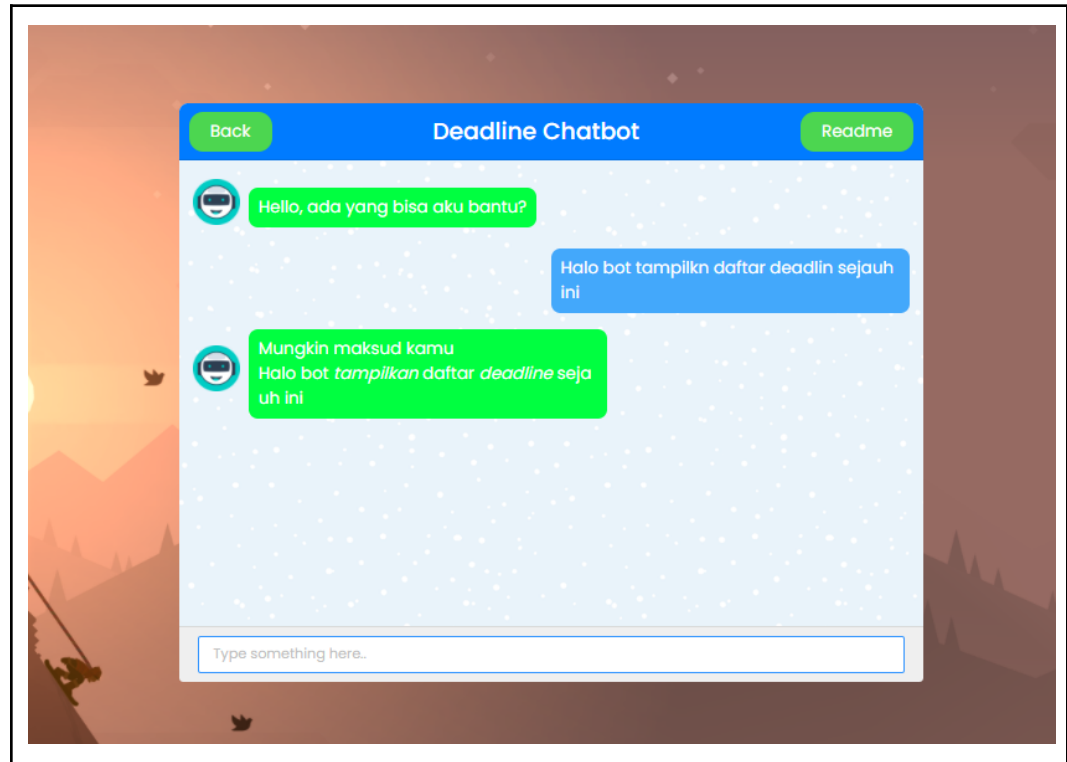
- Menampilkan apa saja yang bisa dilakukan chat bot



- Menampilkan kalimat-kalimat yang tidak dikenali



- Memberikan sugesti pada typo kata-kata yang dianggap penting



4.4 Analisis Hasil Pengujian

Berdasarkan hasil pengujian, *deadline chatbot* sudah dapat menerima kata kunci yang menjadi pembeda perintah. Hanya saja ke depannya untuk fitur penerima tanggal dapat lebih flexible tidak terpaku pada format YYYY/MM/DD. Dan untuk pesan kesalahan ketika error fitur dapat lebih variatif.

BAB V

KESIMPULAN DAN SARAN

5.1. Kesimpulan

Pembangunan Deadline Reminder Assistant “Deadline Chatbot” dapat dilakukan dengan menerapkan String Matching dan Regular Expression. String Matching dan Regular Expression diimplementasikan dengan menggunakan bahasa pemrograman php. String Matching dibentuk dengan algoritma booyermore dan Regular Expression dibentuk dengan fungsi preg_match() dalam php.

5.2. Saran

Dalam pengerjaan proyek ini lebih baik untuk membaca dokumentasi php, javascript, html, dan css terlebih dahulu agar lebih lancar. Selain itu, pada pengembangan selanjutnya proyek ini diharapkan mampu berjalan sebagai aplikasi android maupun IOS.

5.3. Refleksi dan Komentar

Proses pengerjaan tugas ini sudah sangat bagus. Semua anggota kelompok bekerja sesuai pembagian tugas yang telah diberikan, dan juga sesuai dengan waktu yang telah ditetapkan. Untuk tugas stima selanjutnya mohon deadlinenya dipersingkat dan juga dipersulit lagi tingkatannya.

DAFTAR PUSTAKA

- Munir, R. (2020). *Pencocokan String (String/Pattern Matching)*. Institut Teknologi Bandung. Diakses pada 25 Mei 2021 pukul 11.21 WIB melalui <https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Pencocokan-string-2021.pdf>.
- Munir, R. (2019). *String Matching dengan Regular Expression*. Institut Teknologi Bandung. Diakses pada 25 Mei 2021 pukul 11.33 WIB melalui <https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2018-2019/String-Matching-dengan-Regex-2019.pdf>