

BOT PENYIMPAN DATA PENGUMPULAN TUGAS PESERTA ELEARNING BERBASIS TELEGRAM [ER-BOT PDPT]

¹ Rizky Parluka, ² Devan Cakra Mudra Wijaya, ³ Arista Pratama

¹² Program Studi Informatika, Fakultas Ilmu Komputer

³ Program Studi Sistem Informasi, Fakultas Ilmu Komputer

¹²³ Universitas Pembangunan Nasional "Veteran", Jawa Timur

Email: rizkyparlika.if@upnjatim.ac.id

Abstrak. Pada masa pandemi COVID-19, kegiatan rekap data tugas secara manual dirasa tidak efektif, sebab dapat berdampak pada psikologis maupun fisik seseorang karena biasanya jumlah rekap data yang dilakukan secara manual berlebihan, sedangkan pengumpulan tugas secara online di berbagai lembaga pendidikan yang ada saat ini terbelah tidak merata, karena keterbatasan sarana dan prasana dalam kegiatan belajar mengajar yang terjadi pada daerah tertentu terutama daerah tertinggal, terkadang juga terkendala faktor lain yaitu data tertimbun oleh aktifitas chat yang terjadi pada suatu grup kelas, sehingga dosen atau guru ingin merekap tugas menjadi kewalahan. Oleh karena itu, untuk mengatasi hal tersebut perlunya diciptakan sebuah alat untuk mempermudah pekerjaan guru atau dosen ketika merekap data tugas peserta didik. Sebenarnya hal tersebut dapat diatasi dengan berbagai macam metode, misalnya menggunakan media bot telegram. Kami menciptakan bot telegram berbasis python yang berfungsi sebagai alat bantu untuk merekap data tugas dan memberikan notifikasi kepada pengirim secara otomatis, serta dapat melakukan pemberitahuan akses kepada admin.

Kata Kunci: ER-BOT PDPT, elearning, rekap data.

Kemajuan teknologi dan informasi di dunia semakin berkembang [1]. Arus informasi akhir-akhir ini sangatlah padat yang salah satunya masyarakat dihebohkan dengan berita penyebaran virus COVID-19 (*Coronavirus Disease 2019*) yang telah menyebar di berbagai negara dan menimbulkan korban jiwa [2], [3]. Berbagai inovasi telah dilakukan oleh swadaya masyarakat maupun pemerintah dalam upaya memutus rantai penyebaran COVID-19 yaitu dengan cara melakukan adaptasi perubahan kegiatan yang dikenal dengan sebutan WFH (*Work From Home*) dan LFH (*Learn From Home*) secara online dengan memanfaatkan teknologi yang ada pada smartphone.

Salah satu fitur yang paling berguna di perangkat seluler adalah messenger. Ini menjadi alat utama untuk berkomunikasi satu sama lain, untuk berbagi informasi dan berinteraksi [4]. Sebagian besar masyarakat ingin mendapat informasi yang mudah, cepat dan akurat [5]. Oleh karena itu, aplikasi Telegram ini dipilih karena aplikasi tidak berbayar, ringan, dan multiplatform [6]. Fitur unggulan Telegram salah satunya yaitu *bot* sebagai layanan untuk mengirim pesan secara *realtime* yang dapat berjalan pada *platform desktop, mobile, dan web* serta dapat berjalan otomatis sesuai dengan perintah atau *request* dengan metode *long-polling* dan *Webhook*. Telegram memiliki fasilitas yang sangat membantu *developer* untuk membangun aplikasi pihak ketiga melalui

support *API* yang diberikannya [7]. Aplikasi instant messaging Telegram saat ini populer digunakan oleh berbagai kalangan, karena mempunyai fitur-fitur yang sangat canggih dalam hal keamanan [8].

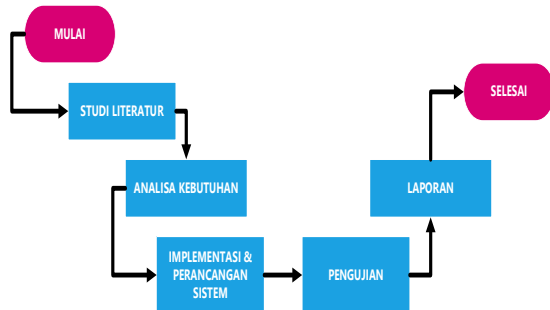
Dalam suatu lembaga pendidikan tentunya masih memiliki kendala yang mana rekap data masih dilakukan secara manual, sehingga dikhawatirkan akan terjadi *human error* ketika proses rekap data berlangsung secara berlebihan. Menurut kami, hal tersebut dapat diselesaikan dengan menggunakan bot telegram dengan bahasa yang populer saat ini yaitu python.

Python adalah bahasa pemrograman berorientasi objek tingkat tinggi yang kuat [9]. *Python* juga memiliki *library* yang lengkap sehingga memungkinkan *programmer* untuk membuat aplikasi yang mutakhir dengan menggunakan *source code* yang tampak sederhana [10].

Penelitian ini dibuat agar tenaga pendidik dapat terbantu pada saat proses merekap data tugas online milik peserta didiknya. Bot telegram yang kami ciptakan menggunakan bahasa pemrograman *python* yang berfungsi sebagai alat bantu untuk merekap data tugas dan memberikan notifikasi kepada pengirim secara otomatis, serta pemberitahuan akses bot kepada admin.

I. Metodologi

Metode penelitian yang digunakan dalam penelitian ini yaitu meliputi alur pelaksanaan pada proses pembuatan *bot* telegram *python* sebagai berikut :



Gambar 1. Diagram alir *bot* telegram *python*

Berikut penjelasan dari masing-masing proses yang terjadi pada saat membuat *bot* telegram *python* tersebut.

1. Studi literatur

Tahapan ini mengatur berbagai hal terkait pendalaman sumber referensi, seperti artikel, makalah, atau laporan yang bersangkutan terhadap permasalahan, mengingat sumber yang akan dimanfaatkan sebagai acuan terhadap pembuatan *bot*.

2. Analisa kebutuhan

Proses analisa kebutuhan sangat penting karena perlunya data dan perencanaan terkait aplikasi yang akan dibuat, contohnya mengutamakan kenyamanan pengguna (*user-friendly*).

3. Implementasi dan Perancangan sistem

Pada tahap ini yang perlu dilakukan adalah melakukan desain fitur pada menu *bot* yakni menentukan teks perintah apa saja yang tersedia pada *bot*.

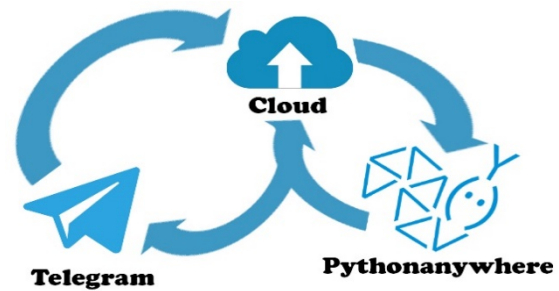
4. Pengujian

Pada tahap ini yang perlu dilakukan adalah pengecekan data secara menyeluruh hingga tidak ditemui adanya masalah dan dinyatakan telah lolos uji *software* (*valid*).

5. Laporan

Laporan yang dibuat harus sesuai dengan data penelitian. Biasanya berisi tata cara pembuatan, penggunaan, hasil, dan sebagainya.

Adapun konsep *bot* yang kami bangun yaitu dapat dilihat pada gambar 2 berikut ini :



Gambar 2. Konsep *bot* telegram *python*

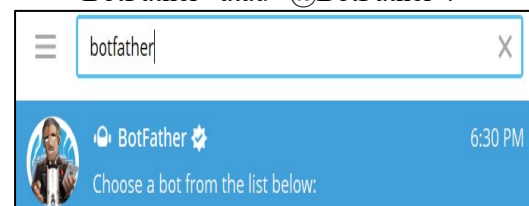
II. Hasil dan Pembahasan

Telegram *bot* adalah sistem cerdas yang diprogram untuk menjalankan serangkaian instruksi yang diberikan oleh pengguna.

Bot Telegram dapat melakukan apa saja sesuai perintah yang ada. *Bot* telegram bisa digunakan untuk melakukan pencarian, sebagai penghubung, pengingat, pengajar, pengolah, pengintegrasian, dan lain-lain.

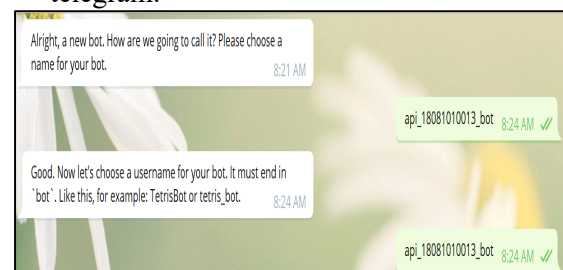
Membuat *bot* dengan BotFather.

1. Pilih menu pencarian dan ketikkan “BotFather” atau “@BotFather”.



Gambar 3. Buat *bot*

2. Lalu klik *START* untuk memulai interaksi dengan @BotFather.
3. Kita akan diberikan berbagai opsi menu. Karena tujuan kita membuat *bot* maka cukup kirim: /newbot.
4. Selanjutnya, buat nama dan *username bot* telegram.



Gambar 4. Nama & *username*

- Kemudian, ingat-ingat *API* dari *bot* yang telah dibuat karena ini penting untuk pengelolaan *bot* nantinya.

```
Use this token to access the HTTP API:
1386369772:AAFqTsKt2AHyeXNGoP7GiHJzJDXUUMzDdM
Keep your token secure and store it safely, it can be used by
anyone to control your bot.
```

Gambar 5. *API token bot*

Membuat bot dengan BotFather.

Dengan *free hosting Pythonanywhere* kita dapat membuat sebuah *Bot* Telegram yang interaktif.

- Buka *pythonanywhere* dan pastikan anda telah login, jika belum memiliki akun *pythonanywhere* silahkan mendaftar di [daftar-pythonanywhere](#).
- Lalu buat direktori, di dalam direktori tersebut anda dapat membuat *new file* atau *upload*.
- Install library python telegram bot dengan cara mengetikkan *command* pada *bash pythonanywhere* : \$ pip3.6 install --user pytelegrambotapi

```
$ pip3.6 install --user pytelegrambotapi
```

Gambar 6. *Library python telegram*

- Setelah *file* ada lalu langkah selanjutnya melakukan pengkodean pada kode sumber. Berikut contoh kode sederhananya :

Library python telegram bot

Merupakan *library* yang memuat fungsi telegram, *inkeyboard* maupun *custom keyboard*, dan waktu serta lokasi.

```
#!/usr/bin/python3.6
import telebot
from telebot import types
import datetime
import pytz .....(1)
```

Fungsi waktu

adalah fungsi yang digunakan mengatur waktu berdasarkan lokasi.

```
#Waktu
d = datetime.datetime.now()
tz = pytz.timezone("Asia/Jakarta")
d = tz.localize(d)
date = d.strftime("%a, %d-%m-%Y")
timestamp = date .....(2)
```

Fungsi API telegram

adalah fungsi yang mengatur konfigurasi *API* telegram *bot* dengan *pythonanywhere*.

```
#Api Telegram
api =
'1386369772:AAFqTsKt2AHyeXNGoP7GiHJzJDXUUMzDdM'
bot = telebot.TeleBot(api) .....(3)
```

Fungsi remove webhook

Merupakan fungsi untuk menghapus *webhook* yang ada sehingga memastikan *hosting* tetap hanya diatur menggunakan *pythonanywhere*.

```
# unset webhook
bot.remove_webhook() .....(4)
```

Variabel global untuk ID Notif

Merupakan variabel *message_id* yang dapat diakses oleh semua fungsi sehingga tidak menuliskan variable secara berulang-ulang.

```
#Private Chat Notif ID
message_id = '1000064401' .....(5)
```

Variabel global untuk Pesan Notif

Merupakan variabel *alert_msg* yang dapat diakses oleh semua fungsi sehingga tidak menuliskan variable secara berulang-ulang.

```
#Message (Alert)
alert_msg = '⚠ <b>Ada yang mengakses bot
pertemuan 6</b> !\n\nID = {}\n Nama =
{}\n Username = {}\n Akses Menu =
{}\n Pada =
```

Fungsi commands start

Merupakan fungsi yang dapat menyambut kedatangan seseorang ketika adanya perintah *start*

```
#Welcome
@bot.message_handler(commands=['start'])
def action_start(message):
    chat_id = message.chat.id
    name = message.from_user.first_name
    last_name = message.from_user.last_name
    if (message.from_user.last_name, 'last_name')
is not None:
    name += " {}".format(last_name)
    username = message.from_user.username
    id_telegram = message.from_user.id
    menu = '/start'
```

```
markdown_html = '<b><a
href="https://t.me/pgjrkhdpn/">Pengajar
Kehidupan</a></b>'

msg = "👋 Selamat datang di <b>pertemuan
6</b>\n\n👋 Hai <b>{}</b>...<b>{}</b> Username =
<b>{}</b>\n👤 ID = <b>{}</b>
    Buatan : {}
    ""
bot.send_message(chat_id,msg.format(name,us
ername,id_telegram,markdown_html),parse_mo
de='HTML')
bot.send_message(message_id>alert_msg.format
(id_telegram,name,username,menu,timestamp),
parse_mode='HTML')
.....(7)
```

Fungsi commands help

Merupakan fungsi yang dapat memberikan informasi terkait semua *commands* yang tersedia di *bot*.

```
@bot.message_handler(commands=['help'])
def action_help(message):
    chat_id = message.chat.id
    name = message.from_user.first_name
    last_name = message.from_user.last_name
    if (message.from_user.last_name, 'last_name')
is not None:
    name += " {}".format(last_name)
    username = message.from_user.username
    id_telegram = message.from_user.id
    menu = '/help'

    msg = "📄 <b>Menu P-6 yang tersedia
:</b>\n\n👇👇👇👇👇👇👇👇👇👇👇👇
/start = mulai
/testbot = status
/about = tentang
/date = waktu
    ""
bot.send_message(chat_id,msg,parse_mode='H
TML')
bot.send_message(message_id>alert_msg.format
(id_telegram,name,username,menu,timestamp),
parse_mode='HTML')
.....(8)
```

Fungsi commands testbot

Merupakan fungsi yang dapat memberikan tanggapan *bot* aktif.

```
#TestBot
@bot.message_handler(commands=['testbot'])
def action_testbot(message):
    chat_id = message.chat.id
    name = message.from_user.first_name
    last_name = message.from_user.last_name
    if (message.from_user.last_name, 'last_name')
is not None:
    name += " {}".format(last_name)
    username = message.from_user.username
    id_telegram = message.from_user.id
    menu = '/testbot'

    msg = '✅ Status bot :<n---> <b>Saat ini bot
aktif...</b>'

bot.send_message(chat_id,msg,parse_mode='H
TML')

bot.send_message(message_id>alert_msg.format
(id_telegram,name,username,menu,timestamp),
parse_mode='HTML') .....(9)
```

Fungsi commands about

Merupakan fungsi yang dapat memberikan informasi seputar *bot* (keterangan singkat terkait *bot* dan juga *creator*).

```
#About
@bot.message_handler(commands=['about'])
def action_about(message):
    chat_id = message.chat.id
    name = message.from_user.first_name
    last_name = message.from_user.last_name
    if (message.from_user.last_name, 'last_name')
is not None:
        name += " {}".format(last_name)
    username = message.from_user.username
    id_telegram = message.from_user.id
    menu = '/about'
    msg = '👤 <b>INFORMASI TENTANG BOT
GRUP PERTEMUAN KE - 6</b>\n\n👤
<b>Nama bot :</b> \n-->
@api_18081010013_bot\n\n👤 <b>Dibuat oleh
:</b> \n--> Tim Rister 2020 \n\n👤 <b>Tujuan
:</b> \n--> Sebagai pembaca dan penyimpan
data-data tugas dalam ER-BOT \n\n👤
<b>Bahasa pemrograman :</b> \n--> Python'
markup = types.InlineKeyboardMarkup()
inkeyboard =
telebot.types.InlineKeyboardButton
markup.add(inkeyboard(text='🌐 Web
profile',url='bit.ly/tanyarizky))
markup.add(inkeyboard(text='👤📖
Scholar',url='bit.ly/gsrizky'))
```

```
bot.send_message(chat_id,msg,reply_markup=
markup,parse_mode='HTML')
bot.send_message(message_id>alert_msg.format
(id_telegram,name,username,menu,timestamp),
parse_mode='HTML') .....(10)
```

Fungsi commands date

Merupakan fungsi yang dapat memberikan informasi mengenai tanggal.

```
#Date
@bot.message_handler(commands=['date'])
def action_date(message):
    chat_id = message.chat.id
    name = message.from_user.first_name
    last_name = message.from_user.last_name

    if (message.from_user.last_name, 'last_name')
is not None:
        name += " {}".format(last_name)
    username = message.from_user.username
    id_telegram = message.from_user.id
    menu = '/date'

    msg = '📅 Current Date : \n--> <b>{}</b>'

    bot.send_message(chat_id,msg.format(timestamp),parse_mode='HTML')
    bot.send_message(message_id>alert_msg.format
(id_telegram,name,username,menu,timestamp),
parse_mode='HTML')
.....(11)
```

Fungsi deteksi dokumen

Merupakan fungsi yang berhubungan dengan dokumen yang dikirim oleh seseorang.

```
#Document New
@bot.message_handler(content_types=['document'])
def kirim_file(message):
    chat_id = message.chat.id
    name = message.from_user.first_name
    last_name = message.from_user.last_name
    if (message.from_user.last_name, 'last_name') is
not None:
        name += " {}".format(last_name)
    pdf = message.document
    file_id = pdf.file_id
    pdf_name = pdf.file_name
    file = bot.get_file(file_id)
    file_loc = file.file_path
    download = bot.download_file(file_loc)
    with open (pdf_name,"wb") as f:
        fl = f.write(download)
```

```

if (fl!=0) is True :
    msg = 'Terima kasih <b>{}</b>, anda telah
    mengirimkan tugas pada <b>pertemuan ke
    6</b>.\n\n■ Nama file = <b>{}</b>\n\n🕒
    Pada = <b>{}</b>'

    bot.send_message(chat_id,msg.format(name,
    pdf_name, timestamp),parse_mode='HTML')

elif (fl!=0) is False:
    upgmsg = 'File yang anda kirim telah
    diperbarui.\n\n■ Nama file =
    <b>{}</b>\n\n🕒 Pada = <b>{}</b>'

    bot.send_message(chat_id,upgmsg.format(pdf_
    name, timestamp),parse_mode='HTML')
    .....(12)
    
```

Fungsi new member group

Merupakan fungsi yang dapat menyapa *member* baru di dalam suatu grup.

```

#Welcome new member group
@bot.message_handler(func=lambda m: True,
content_types=["new_chat_members"])
def on_user_joins(new_chat_members):
    name =
    new_chat_members.from_user.first_name
    last_name =
    new_chat_members.from_user.last_name
    if (new_chat_members.from_user.last_name,
    'last_name') is not None:
        name += " {}".format(last_name)

    pesan = '👋 Selamat datang <b>{}</b> di
    <b>Pertemuan 6</b>. Semoga anda merasa
    nyaman dan selalu berbahagia' .....(13)
    
```

Fungsi commands error handling

Merupakan fungsi pencocokan tulisan dengan *commands* yang tersedia, jika tulisan atau *commands* tidak dikenali maka *bot* akan menyarankan untuk mengakses fungsi *help*.

```

#Handle wrong commands
@bot.message_handler(func=lambda message:
True)
def echo_message(message):
    usermsg = message.text
    name = message.from_user.first_name
    last_name = message.from_user.last_name
    if (message.from_user.last_name, 'last_name')
    is not None:
        name += " {}".format(last_name)
    
```

```

msg = '👋 Hai <b>{}</b>\n\nPerintah
<b>{}</b> pada menu tidak
tersedia.\n\nSilahkan akses <b>/help</b> untuk
mengetahui pilihan menu yang tersedia.'

bot.reply_to(message,msg.format(name,userms
g),parse_mode='HTML')
    .....(14)
    
```

Fungsi polling

Merupakan fungsi yang meng-*compile* dan menampilkan status *compile bot* berhasil dijalankan. Tampilan sisi *Prompt* :

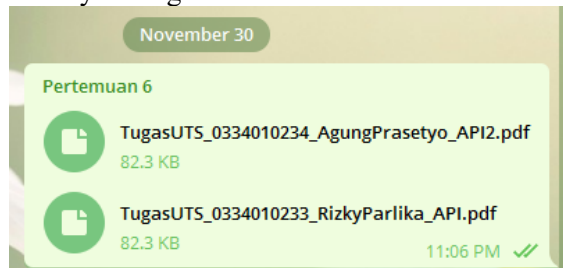


Gambar 7. *Polling*

Hasil implementasi *bot* telegram dengan pemrograman *python* telah selesai.

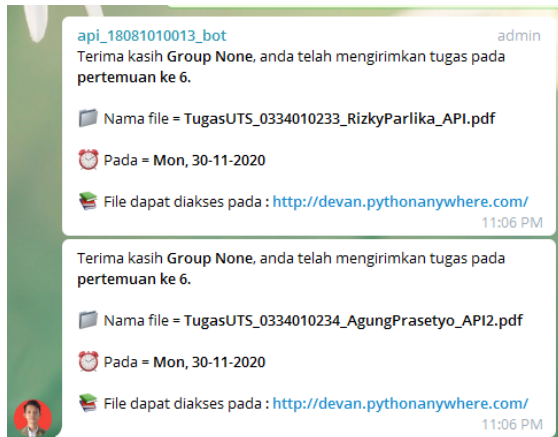
2.3 Uji Coba ER-BOT PDPT

Untuk menguji apakah *bot* telegram *ER-BOT PDPT* telah bekerja dengan baik atau tidak, maka masukkan langsung 2 *file* berekstensi *PDF* sebagai data uji, maka hasilnya sebagai berikut :



Gambar 8. Kirim *PDF*

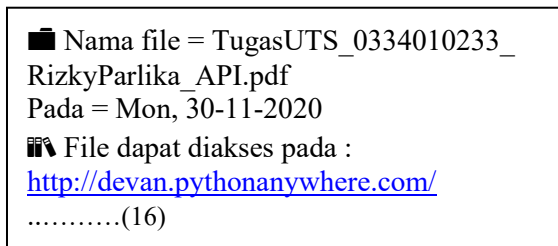
Dalam beberapa detik, *ER-BOT PDPT* akan membalas seperti berikut :



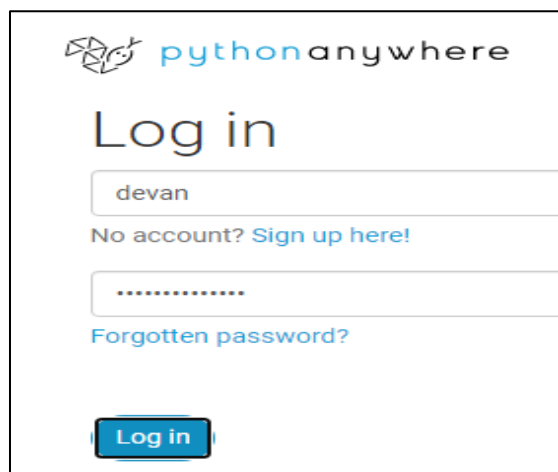
Gambar 9. Notifikasi kepada pengirim

Tampak kedua *file* dikenali oleh *bot* dan *bot* membalas salah satu *file* sebagai berikut:

Terima kasih *Group None*, anda telah mengirimkan tugas pada pertemuan ke 6.

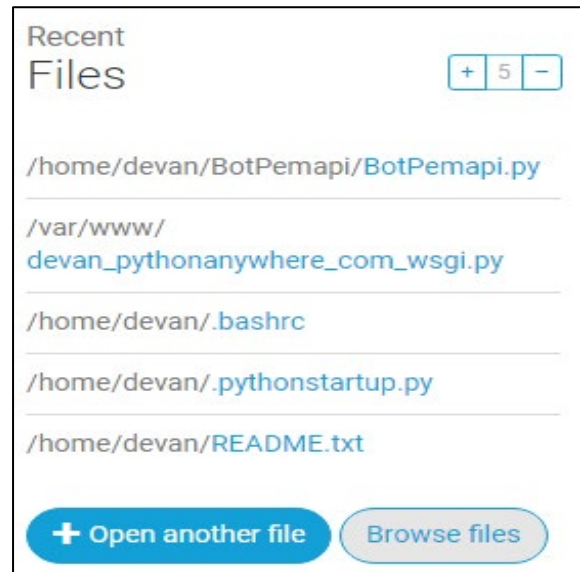


Dari keterangan *bot* diatas, selanjutnya kita cek apakah di *hosting* kita kedua *file* sudah masuk atau belum. Caranya kita *login* dulu pada *akun hosting* kita.



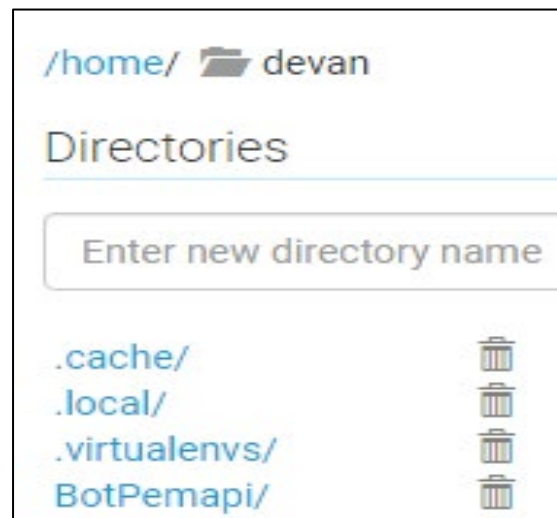
Gambar 10. Login

Selanjutnya kita pilih *Browse File*.



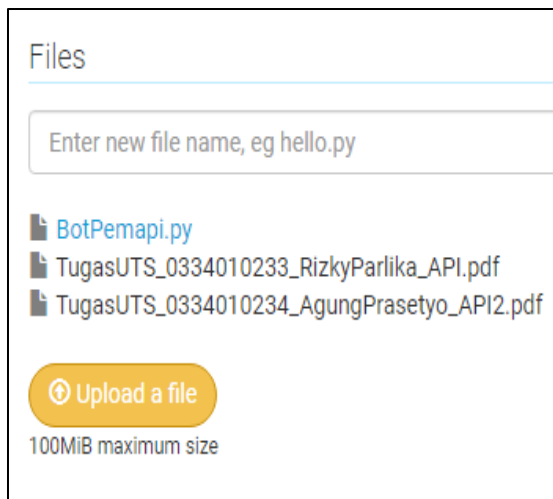
Gambar 11. Browse files

Selanjutnya buka di *folder* yang sudah disiapkan dalam contoh ini di *folder* BotPemapi.



Gambar 12. Buka folder

Selanjutnya klik *folder* BotPemapi maka akan tampil kedua *file* yang telah di-*upload* melalui telegram.



Gambar 13. Data yang tersimpan

III. Kesimpulan

Kesimpulan yang dapat diambil dari makalah ini adalah sebagai berikut:

1. *Bot* Telegram *ER-BOT PDPT* adalah sebuah *bot* telegram yang dibangun menggunakan bahasa *python* untuk menyimpan data-data tugas peserta yang dimasukkan kedalam lingkungan *Elearning* melalui sebuah *group* telegram.
2. *ER-BOT PDPT* dijalankan pada *hosting pythonanywhere* yang memiliki dukungan terhadap bahasa pemrograman *python*.

Ucapan Terima Kasih

Kami mengucapkan terima kasih yang sebesar-besarnya kepada LPPM UPN "Veteran" Jawa Timur yang telah memberikan kesempatan dan amanah kepada kami untuk menyelesaikan kegiatan RISTER 2020.

Tak lupa kami berterimakasih kepada seluruh redaksi Jurnal Teknologi Informasi dan Komunikasi (SCAN) atas dimuatnya makalah kami ini.

IV. Daftar Pustaka

- [1] Q. J. Rizky Parlika, Dhany Satya Utama, Reza Achmad Gallanta and A. N. Nabilah. (2019). Studi Komparatif Implementasi Push Message Pada Media Sosial Secara Gratis. *e-NARODROID*.
- [2] R. Parlika, S. Ilham Pradika, A. Muhammad Hakim, and K. Rachman N.M. (2020). Bot Whatsapp Sebagai Pemberi Data Statistik Covid-19 Menggunakan Php, Flask, Dan Mysql. *J. Inform. dan Sist. Inf.*, vol. 1, no. 2 SE-Articles, pp. 282–293, Jul.

- [3] E. Supriatna. (2020). Wabah Corona Virus Disease (Covid 19) Dalam Pandangan Islam. *SALAM J. Sos. dan Budaya Syar-i*, vol. 7.
- [4] H. Setiaji and I. Paputungan. (2018). Design of Telegram Bots for Campus Information Sharing," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 325, p. 1.
- [5] T. A. Zuraiyah, D. K. Utami, and D. Herlambang. (2019). Implementasi Chatbot Pada Pendaftaran Mahasiswa Baru Menggunakan Recurrent Neural Network. *J. Ilm. Teknol. dan Rekayasa*.
- [6] R. Juniyantara Putra, N. Putra Sastra, and D. M. Wiharta. (2018). Pengembangan Komunikasi Multikanal Untuk Monitoring Infrastruktur Jaringan Berbasis Bot Telegram. *J. SPEKTRUM*; Vol 5 No 2 J. SPEKTRUM.
- [7] R. Parlika and A. Pratama. (2019). Aplikasi Mesin Penjawab Pesan Berbasis Bot Telegram, Php, Dan Mysql. *SCAN - J. Teknol. Inf. dan Komun.*
- [8] P. Febriyanti and S. Rusmin, "Febriyanti Panjaitan. (2019). Pemanfaatan Notifikasi Telegram Untuk Monitoring Jaringan. *J. SIMETRIS*, vol. 10, no. 2, pp. 725–732.
- [9] K. R. Srinath. (2017). Python – The Fastest Growing Programming Language. *Int. Res. J. Eng. Technol.*
- [10] R. Ananda, F. Imamah, Y. A. S, and A. Dores. (2018). Aplikasi Chatbot (Milki Bot) Yang Terintegrasi Dengan Web Cms Untuk Customer Service Pada Ukm Minsu. *J. Cendikia*, vol. 16, no. 2 Oktober *SE-Articles*, Nov.
- [11] pythonanywhere <https://www.pythonanywhere.com/> diakses pada tanggal 30 November 2020 pukul 11:28 PM