Perancangan Bot Telegram Untuk Pengamanan Kunci Optical Distribution Cabinet (ODC) Divisi *Maintenance*/Optima PT.Telkom Tasikmalaya

Design Of Telegram Bot For Security Optical Distribution Cabinet (ODC) Maintenance/Optima Division PT. Telkom Witel Tasikmalaya

1st Muhammad Katamin
Fakultas Ilmu Terapan
Universitas Telkom
Bandung, Indonesia
fahdyusran@student.telkomuniversity.ac.id

2nd Suci Aulia
Fakultas Ilmu Terapan
Universitas Telkom
Bandung, Indonesia
suciaulia@telkomuniversity.ac.id

3rd Suliyono
Fakultas Ilmu Terapan
Universitas Telkom
Bandung, Indonesia
suliyono2000@tass.telkomuniversity.ac.id

Abstrak—Seiring perkembangannya zaman dan berjalanya waktu Industry 4.0. PT. Telekomunikasi Witel Tasikmalaya terus berkembang salah satunya dalam Pengamanan Kunci ODC (Optical Distribution Cabinet). Semakin banyaknya Teknisi dalam Pengamanan Kunci di ODC proses Pengamanan Kunci ODC pun jadi semakin sulit seperti tidak bisa melihat siapa terakhir kali memakai kunci pengaman dan keamanan pada ODC cukup rumit. bot telegram atau robot telegram merupakan sebuah akun telegram khusus yang didesain dapat merespon pesan secara otomatis, yang tidak memerlukan nomor telepon tambahan. Google Sheets merupakan pengolahan data, mengurutkan data, serta memanipulasi data dalam bentuk tabel. Pada proyek akhir ini merancang dan merealisasikan sebuah aplikasi yaitu bot Telegram Pengamanan Kunci ODC, dengan tujuan untuk mempermudahkan perusahaan dalam menggakses ODC dan dapat meningkatkan kemajuan terhadap Telkom Witel Tasikmalaya. Hasil dari Proyek Akhir ini adalah berfungsinya bot telegram dan memiliki tingkat keberhasilan 100%, semua fitur dapat berjalan sesuai dengan hasil yang diharapkan. Hasil pengujian performa dengan mengukur respon time dari bot didapatkan hasil rata-rata respon time keseluruhan sebanyak 1,1 detik, dan hasil pengujian subjektivitas didapatkan hasil 82,93% – 92,1% responden menilai aplikasi ini Sangat Baik.

Kata kunci— pengamanan kunci ODC, telegram messenger, bot telegram

I. PENDAHULUAN

Seiring perkembangannya zaman dan waktu, industri 4.0. PT. Telekom Indonesia Witel Tasikmalaya terus berkembang salah satunya dalam pengamanan kunci Optical Distribution Cabinet (ODC). ODC merupakan suatu perangkat pasif yang diinstal di luar STO, bisa di luar ruangan (outdoor), dan bisa juga di dalam ruangan (indoor) [1]. Semakin banyaknya Teknisi dalam pengamanan kunci di ODC dan proses pengamanan kunci ODC pun jadi semakin sulit dan tidak bisa melihat siapa terakhir kali memakai kunci pengaman pada ODC cukup sangat rumit.

Semakin banyaknya kebutuhan Teknisi yang terus meningkat dalam pengambilan kunci ODC yang sangat sulit maka dari Admin dan Manager membuat bot telegram untuk memudahkan teknis dalam pengambilan kunci dan pengamanan kunci ODC dimana kunci ODC sekarang masih mengunakan kunci manual atau bisa disebut kunci gembok biasa. Bot telegram atau robot telegram merupakan sebuah akun telegram khusus yang didesain dan dapat merespon pesan secara otomatis, yang tidak memerlukan nomor telepon tambahan. Google sheets merupakan pengolahan data, mengurutkan data, serta memanipulasi data dalam bentuk table.

Berdasarkan permasalahan tersebut, maka Proyek Akhir ini merancang suatu aplikasi yaitu bot Telegram Pengamanan Kunci ODC. Penelitian ini dilakukan untuk mempermuadah perusahaan dalam mengatasi Pengamanan Kunci ODC. Dan pada proyek akhir ini diharapkan dapat mempermudahkan perusahaan dalam melaksanakan maintenance atau gangguan pada ODC PT. Telkom Indonessia Witel Tasikmalaya.

II DASAR TEORI

A. Telkom Indonesia Witel Tasikmalaya

PT Telkom Indonesia (Persero) Tbk (Telkom) adalah Badan Usaha Milik Negara (BUMN) yang bergerak di bidang jasa layanan teknologi informasi dan komunikasi (TIK) dan jaringan telekomunikasi di Indonesia [2]. Dalam upaya bertransformasi menjadi *digital telecommunication company*, TelkomGroup mengimplementasikan strategi bisnis dan operasional perusahaan yang berorientasi kepada pelanggan (*customer-oriented*). Transformasi tersebut akan membuat organisasi TelkomGroup menjadi lebih lean (ramping) dan agile (lincah) dalam beradaptasi dengan perubahan industri telekomunikasi yang berlangsung sangat cepat. Organisasi yang baru juga diharapkan dapat

meningkatkan efisiensi dan efektivitas dalam menciptakan customer experience yang berkualitas.

PT. Telekomunikasi Indonesia, Tbk mempunyai kantor yang tersebar di seluruh Indonesia. Salah satunya ialah PT Telekomunikasi Indonesia Witel Tasikmalaya yang akan menjadi objek perusahaan yang akan di teliti oleh penulis. PT Telekomunikasi Indonesia Witel Tasikmalaya terbentuk pada 6 juli 1965, yang membawahi kantor Kandatel Garut, Singaparna, Ciamis, Banjar, dan Tasikmalaya.

GAMBAR 1
TELKOM INDONESIA

B. Bot Telegram

Bot telegram merupakan akun telegram khusus yang didesain dapat menerima pesan secara otomatis [5]. Akun bot telegram ini berjalan sesuai dengan kode dan function yang berjalan di sebuah server. Terdapat dua metode setting server bot telegram, yaitu dengan menggunakan metode long-polling dan webhook. Metode long-polling artinya server bot telegram bisa diakses menggunakan laptop sendiri sebagai servernya. Pada metode ini, server akan mengecek aktivitas bot secara periodik. Jika ada pesan yang masuk, maka server akan mengeksekusi berdasarkan pesan request yang dikirim pengguna. Jika tidak ada maka kondisi server adalah idle. Apabila menggunakan metode webhook, server bot telegram ini berada pada hosting menggunakan HTTPS. Artinya, perancangan bot telegram dengan metode ini memakai server yang dapat diakses oleh user lain. Pada Proyek Akhir ini metode yang dipilih adalah metode webhook. Adapun ilustrasi metode webhook ditampilkan pada Gambar 2

GAMBAR 2
ILUSTRASI METODE WEBHOOK

Pada Gambar 2 diilustrasikan bahwa server yang digunakan berada pada hosting menggunakan HTTPS sehingga server dan bot terhubung secara realtime tanpa adanya request. Dengan begitu metode webhook ini dapat membalas pesan dengan lebih cepat. bot telegram ini dirancang menggunakan Telegram bot Application Programming Interface (API) yang memberikan akses ke developer untuk membuat bot sesuai yang diinginkan. Sehingga pada bot telegram terdapat input-proses-output [6]. Proses interaksi yang dilakukan oleh pengguna dan bot telegram akan berbentuk Javascript Object Notation (JSON) [7]. JSON (JavaScript Object Notation) adalah format pertukaran data yang ringan, mudah dibaca dan ditulis oleh manusia, serta mudah diterjemahkan dan dibuat (generate) oleh komputer [8]. Hampir semua bahasa pemrograman dapat digunakan untuk merancang suatu bot telegram.

C. Telegram Bot API

Telegram bot API merupakan antarmuka berbasis *Hypertext Transfer Protocol* (HTTP) yang dibuat untuk pengembang yang ingin membangun bot untuk Telegram. API (Application Programming Interface) adalah seperangkat antar muka yang memungkinkan sebuah perangkat lunak berhubungan dengan perangkat lunak lainnya. API mendefinisikan jenis Request yang dapat dibuat, bagaimana membuat Request tersebut, bagaimana format data yang harus digunakan, serta pendefinisian lain-lainnya [9]. Adapun beberapa jenis metode pada Telegram bot API, diantaranya: *sendMessage*, *forwardMessage*, *sendPhoto*, *sendAudio*, *sendDocument*, *callback_data*, *callback_query*, *inline_keyboard* dan lain-lain [10].

D. *Webhook*

Webhook merupakan rancangan API *(Application Programming Interface*) yang sering digunakan khalayak banyak dan sangat berguna bagi aplikasi karena menyediakan rancangan dari beberapa aplikasi lain dengan informasi secara langsung atau r*eal-time* [11].

Webhook yang digunakan pada bot telegram merupakan fitur yang efisien terutama untuk *developer* pembuat bot karena hanya membutuhkan satu instruksi yaitu setWebhook untuk melakukan pengaturan awal dalam perancangan bot telegram. Perintah ini dapat dijalankan menggunakan *Client* URL (cURL) atau diketikan langsung pada URL *browser* [12]. Adapun contoh penulisan perintah konfigurasi *webhook* ditampilkan pada Gambar 3

```
curl -H "Content-Type: application/json"
-X POST -d
'{"url":"https://www.example.com/my-
secret-
webhook.php"}'https://api.telegram.org/b
otYOUR_BOT_TOKEN/setWebhook
```

GAMBAR 3
KONFIGURASI *WEBHOOK*

Pada **Error! Reference source not found.** 3 merupakan perintah untuk mengirimkan permintaan JSON ke Telegram API. Dengan menggunakan metode *webhook* ini artinya *server* bot telegram berada pada *hosting* menggunakan HTTPS sehingga bot telegram akan lebih cepat memberikan respon balasan [12].

E. Google *Apps Script* (GAS)

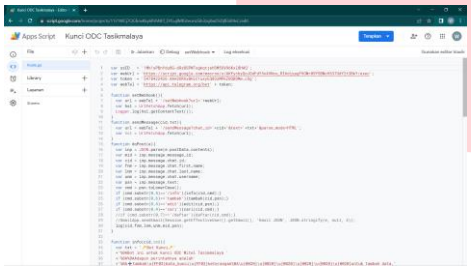
Google *apps script* adalah platform pengembangan aplikasi dan pengkodean yang dibangun di dalam google *apps* yang memungkinkan pengguna untuk menambahkan fungsionalitas ke spread sheet, gmail,

situs, dan layanan lainnya dari google. Google *apps script* dapat dibuat sebagai file mandiri pada drive, dokumen atau spread sheet yang dikenal sebagai wadah atau situs google [14].

Fungsi google *apps script* adalah sebagai berikut:

1. Menulis fungsi yang ditentukan pengguna untuk google sheet.
2. Menulis aplikasi tipe makro sederhana.
3. Mengembangkan aplikasi berbasis *spreadsheet*.
4. Mengintegrasikan produk dan layanan google lainnya.
5. Mengembangkan antarmuka pengguna grafis (GUIS) yang dapat dijalankan sebagai aplikasi *web*.
6. Berinteraksi dengan basis data relasional berbasis cloud melalui google JDBC *service*

Adapun tampilan penulisan *script* pada *google apps script* yang ditampilkan pada Gambar 4



GAMBAR 4
TAMPILAN GOOGLE *APP SCRIPT*

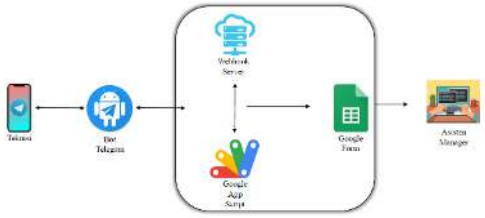
F. UDF

UDF adalah fungsi yang dibuat sendiri oleh seorang programmer untuk membuat program sesuai dengan kebutuhan. UDF ini dibuat karena programmer ingin membuat custom *function* yang tidak disediakan oleh google *spreadsheet*. UDF juga dapat digunakan di bahasa pemrograman *Hypertext Preprocessor* (PHP). Walaupun UDF dibuat secara custom dan tidak terdaftar di *function* pada google *spreadsheet* atau PHP, akan tetapi google *spreadsheet* dengan bahasa pemrogramannya yaitu google *apps script* dan PHP dapat memproses UDF ini dengan baik selagi cara penulisan kode-kode pada UDF tersebut telah memenuhi standarisasi penulisan kode dalam google apps script atau PHP [10].

III METODE

A. Blok Diagram Sistem

Berikut merupakan Blok Diagram Sistem dari perancangan sistem monitoring untuk laporan gangguan IndiHome dengan bot telegram yang ditampilkan pada Gambar 5



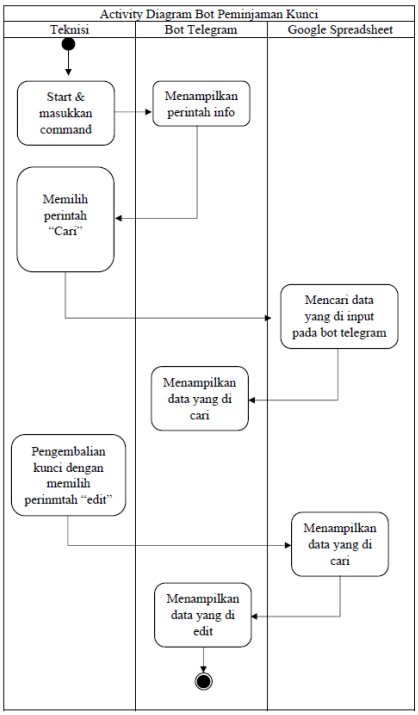
GAMBAR 5
BLOK DIAGRAM SISTEM

Pada Gambar 5 diilustrasikan bahwa saat teknisi mengakses bot telegram melalui aplikasi telegram yang dapat diakses melalui internet, teknisi dapat melakukan penginputan kunci ODC yang dibutuhkan dan cek kunci ODC yang dibutuhkan melalui bot telegram. Setelah data dimasukkan dan diterima oleh bot telegram, data akan otomatis tersimpan di Google *spreadsheet*. Google *spreadsheet* digunakan untuk mengumpulkan data yang masuk ke bot telegram dan digunakan untuk menentukan kunci ODC yang digunakan teknisi. Google *spreadsheet* ini diprogram menggunakan google *apps script* untuk menjalankan berbagai *function* tertentu untuk mendukung fitur pada bot telegram. *Server* yang digunakan untuk antarmuka bot ini di setting menggunakan metode *webhook connection*. Setelah data kunci ODC diterima asisten maneger melalui *google spreadsheet*, data akan di proses langsung. Kemudian, data di google *spreadsheet* divisualisasikan dalam bentuk tabel sebagai dashboard sehingga dapat memudahkan proses monitoring yang dilakukan oleh asisten menager PT. Telkom Witel Tasikmalaya.

B. Activity Diagram Bot Peminjaman Kunci ODC

Activity diagram merupakan penjelasan aliran kerja sebuah sistem yang ada pada perangkat lunak dan bukan gambaran perilaku aktor. Berikut merupakan *Activity diagram* dari setiap fitur pada bot telegram.

Pada Gambar 6 dibawah akan dijelaskan tentang *activity* bot peminjaman kunci yang telah dibuat pada Proyek Akhir ini.

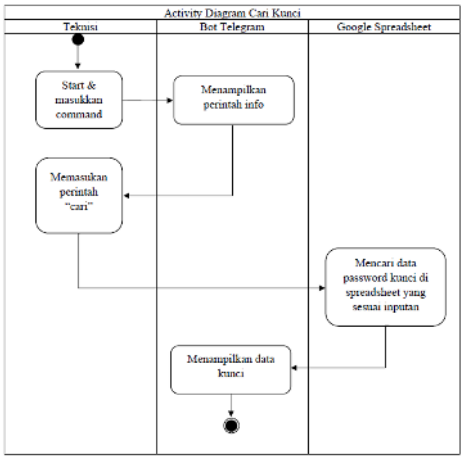


GAMBAR 6
ACTIVITY DIAGRAM BOT PEMINJAMAN KUNCI

Activity diagram proses bot peminjaman kunci ODC pada Gambar 6 menjelaskan mengenai activity diagram proses peminjaman bot kunci ODC. Langkah pertama teknisi memilih perintah "Perintah Cari", kemudian mencari data yang di input pada bot telegram. Setelah itu

proses input menampilkan data yang di cari,tambah, dan edit yang akan segera ditindak lanjuti.

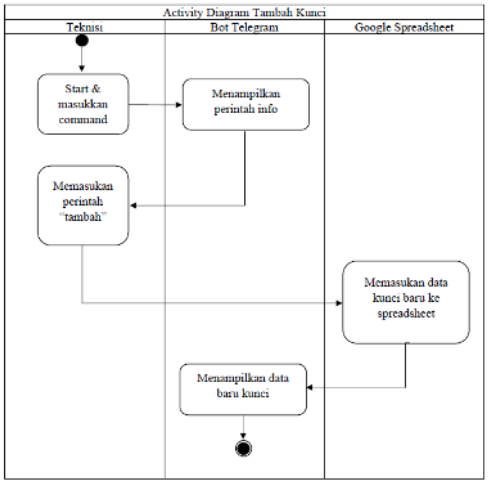
Pada Gambar 7 dibawah akan dijelaskan tentang *activity* bot peminjaman kunci yang telah dibuat pada Proyek Akhir ini.



GAMBAR 7
ACTIVITY DIAGRAM CARI KUNCI

Activity diagram cari bot peminjaman kunci ODC. Adapun *activity diagram* proses bot peminjaman kunci ODC pada Gambar 7 menjelaskan mengenai *activity diagram* proses cari kunci teknisi dapat melakukan cari kunci ODC sesuai dengan kebutuhan yang dipakai.

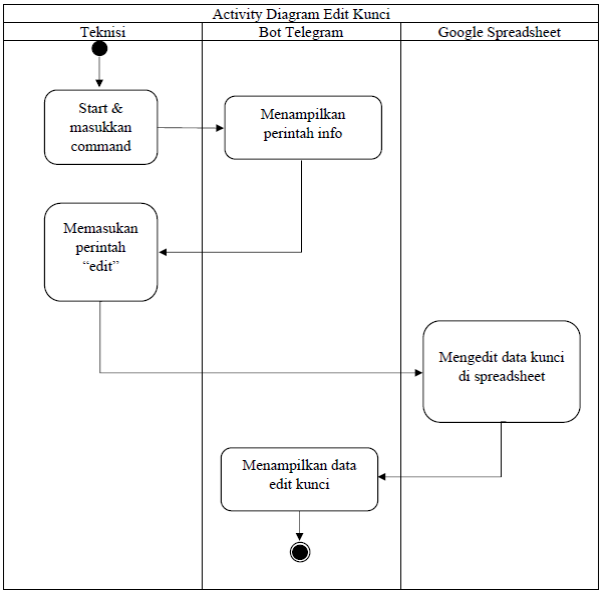
Pada Gambar 8 dibawah akan dijelaskan tentang *activity* bot tambah kunci yang telah dibuat pada Proyek Akhir ini.



GAMBAR 8
ACTIVITY DIAGRAM TAMBAH KUNCI

Activity diagram tambah kunci. Adapun *activity diagram* proses tambah kunci ditampilkan pada Gambar 3.6 menjelaskan mengenai *activity diagram* proses tambah kunci ODC dan memasukan data kunci ODC baru ke dalam *spreadsheet* dan fungsi tambah dapat di akses oleh admin.

Pada Gambar 9 dibawah akan dijelaskan tentang *activity* bot peminjaman kunci yang telah dibuat pada Proyek Akhir ini.

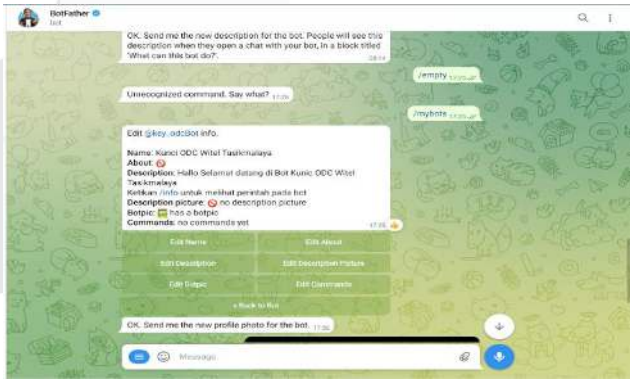


GAMBAR 9
ACTIVITY DIAGRAM EDIT KUNCI

Activity diagram edit kunci Adapun activity diagram proses edit kunci ditampilkan pada Gambar 9 menjelaskan mengenai *activity diagram* edit kunci ODC proses tersebut dapat dilakukan hanya oleh pihak admin, fungsi tersebut untuk mengedit kunci ODC dan data ODC

C. Menghubungkan Bot Dengan Google *Spreadsheet*

Bot telegram ini perlu dihubungkan dengan google *spreadsheet* agar dapat melakukan proses perancangan *function* bot menggunakan google *apps script*. Bot telegram dengan google *spreadsheet* dihubungkan dengan menggunakan token bot yang didapatkan dari Bot Father. Token ini digunakan untuk mengakses telegram bot API. Adapun cara mendapatkan token bot telegram seperti pada Gambar 10



GAMBAR 10
BOT FATHER

Pada Gambar 10 ditampilkan token bot telegram yang didapatkan dari bot *father*. Kemudian token bot telegram itu diketikkan di file google *apps script* seperti pada tampilan script dibawah.

```
var ssID   = '1VTWb9Qg47wtiy4rm7ZKM29RnfnFbnBW4d7pzBGGGBPE';
var webUrl = 'https://script.google.com/macros/s/AKfycbxVmYRGuEDrdMWp
fn3GiVeOGLoksOxTpWf5YHY3CH2u0DoMArLRivLrZXOEYjDkhNz/exec';
var token  = '5508490638:AAGRXjnJ66QmmJ-d2K9OG9ttzVJDTZ_4YSA';
var webTel = 'https://api.telegram.org/bot' + token;
```

Token bot ini dideklarasikan bersamaan dengan SSID google *spreadsheet* yang dipakai dan *link web* app dari *google apps script*.

D. Setting *Webhook* Sebagai *Server*

Bot telegram ini menggunakan server yang di setting dengan metode *webhook*. Setting metode *webhook* ini menggunakan *link hosting*, sehingga tidak diperlukan *server* lokal dan akan lebih hemat *resource*. *Server* ini dapat menampung *request* dari *user* secara bersamaan dan dapat merespon lebih cepat.

```
function setWebhook(){
  var url = webTel + '/setWebhook?url='+webUrl;
  var hsl = UrlFetchApp.fetch(url);
  Logger.log(hsl.getContentText());
      }
```

Pada script diatas merupakan *function setWebhook* yang dituliskan pada google *apps script*. Dengan menjalankan *function* tersebut maka di tampilan log akan muncul pesan "*Webhook was set*." yang artinya *webhook server* telah berhasil diatur.

E. Perancangan UDF

Pada proses perancangan UDF ini dilakukan di google *apps script.*

```
function sendMessage(cid,txt){
  var url = webTel + '/sendMessage?chat_id='+cid+'&text='+txt+'&parse_
mode=HTML';
  var hsl = UrlFetchApp.fetch(url);
      }
```

Script UDF *sendMessage* merupakan yang dibuat dengan menggunakan telegram bot API sendMessage. *Function* tersebut digunakan untuk mengirim pesan teks dari bot dengan format HTML.

```
function doPost(e){
  var inp = JSON.parse(e.postData.contents);
  var mid = inp.message.message_id;
  var cid = inp.message.chat.id;
  var fnm = inp.message.chat.first_name;
  var lnm = inp.message.chat.last_name;
  var unm = inp.message.chat.username;
  var psn = inp.message.text;
  var cmd = psn.toLowerCase();
  if (cmd.substr(0,5)=='/info'){info(cid,cmd);}
  if (cmd.substr(0,5)=='/form'){form(cid,cmd);}
  if (cmd.substr(0,6)=='tambah'){tambah(cid,psn);}
  if (cmd.substr(0,4)=='edit'){edit(cid,psn);}
  if (cmd.substr(0,4)=='cari'){cari(cid,cmd);}
  if (cmd.substr(0,5)=='hapus'){hapus(cid,cmd);}
        log(cid,fnm,lnm,unm,mid,psn);
          }
```

Script UDF doPost merupakan UDF yang digunakan untuk parsing bentuk *JSON* menjadi data.

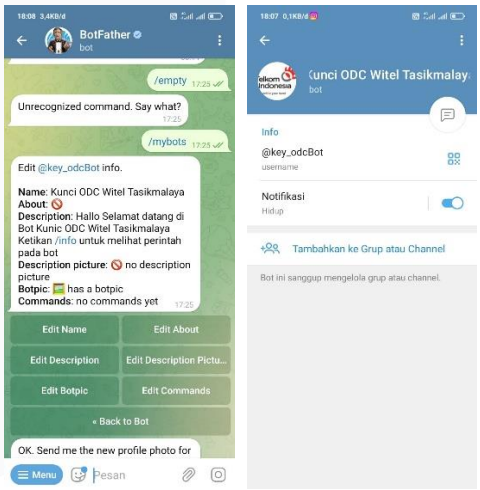
```
function bacadata(){
  var ws = SpreadsheetApp.openById(ssID);
  var ss = ws.getSheetByName('data');
  var lr = SpreadsheetApp.openById(ssID).getActiveSheet().getLastRow()
  var lc = SpreadsheetApp.openById(ssID).getActiveSheet().getLastColumn()
  var rg = 'data!A2:B'+lr;
  var rs = SpreadsheetApp.openById(ssID).getActiveSheet().getRange(rg).getV
alues();
  return rs;
}
```

1. *Script* UDF bacadata merupakan UDF yang dibuat membaca *spreadsheet* dari ID *spreadsheet* yang telah di tambahkan serta membaca *rows* & *range name* yang telah diatur.
2. *Script* UDF info merupakan UDF yang dibuat menampilkan perintah yang terdapat pada bot telegram.
3. *Script* UDF tambah merupakan UDF yang dibuat untuk menambah kunci dari sisi warehouse pada bot telegram.
4. *Script* UDF edit merupakan UDF yang dibuat mengedit data kunci pada *spreadsheet* yang terdapat pada bot telegram. Proses edit hanya dapat dilakukan oleh admin.
5. *Script* UDF cari merupakan UDF yang dibuat mencari data pada *spreadsheet*. Proses cari bisa digunakan oleh teknisi dan admin.

IV. HASIL DAN PEMBAHASAN

A. Hasil Perancangan Bot Telegram

Pada Gambar 11 merupakan hasil dari perancangan aplikasi lapor peminjaman kunci ODC. *Username* dari bot telegram adalah: @key_odcBot.



(a) (b)

GAMBAR 11
INFORMASI BOT

Pada Gambar 11 (a) menampilkan informasi bot telegram. Pada Gambar 11 (b) ditampilkan hasil tampilan awal saat memulai bot akan dijelaskan mengenai beberapa fitur pada bot telegram, dan saat mengetikkan *command*/info akan muncul pesan pembuka dan perintah *command*.

Adapun Gambar 11 ditampilkan fitur (tambah,edit,cari) bot telegram yang dapat diakses oleh Teknisi.

(a) (b)

GAMBAR 12
CEK KUNCI

Pada Gambar 12 (a) merupakan tampilan fitur cek kunci. Adapun Gambar 12 (b) tampilan fitur tambah bot telegram yang dapat diakses oleh admin.

B. Pengujian Non Fungsional

Pada tahap pengujian fungsionalitas bertujuan untuk mengetahui apakah semua fitur pada bot telegram dapat berjalan dengan baik dan sesuai yang diharapkan atau tidak. Spesifikasi pertama yang diuji pada bot telegram adalah proses pengajuan peminjaman kunci ODC, dan cek hasil peminjaman. Pada proses pengajuan peminjaman kunci ODC data yang dimasukan oleh teknisi harus sesuai dengan yang di perlukan. Dan pada cek peminjaman kunci ODC data yang dimasukan harus seusai dengan *database*.

Masing-masing skenario mewakili proses cek kunci dan edit kunci. Adapun hasil pengujian fungsionalitas spesifikasi pertama pada dapat disimpulkan pengujian pada spesifikasi pertama sudah sesuai dengan hasil yang diharapkan dan memiliki tingkat keberhasilan 100 pada Tabel 1.

TABEL 1
SKENARIO CEK DAN EDIT

Skenario Pengujian	Data Masukan	Hasil yang diharapkan	Kesimpulan
Skenario 1	cek hasil peminjaman kunci yang di butuhkan	Data cek di tampilkan pada bot telegram	Sesuai
Skenario 2	Mengedit kunci yang sudah di pakai	Menampilkan data kuncil pada *database*	Sesuai

Kemudian, spesifikasi kedua yang diuji pada bot telegram adalah fitur untuk admin. Pada proses (tambah, edit, cari) data yang harus dimasukan harus sesuai dengan database kunci ODC. Detail bukti *screenshot expected result* atau hasil yang diharapkan pada pengujian fungsionalitas ditampilkan pada pada Tabel 2

TABEL 2
SKENARIO PENGUJIAN KE 2

Skenario Pengujian	Data Masukan	Hasil yang diharapkan	Kesimpulan
Tambah	Menambah data ODC dan kunci ke dalam google *spreadsheet*	Data masuk ke database / data sheet admin dan tampil di bot telegram	Sesuai
Edit	Mengedit data ODC dan kunci	Mengedit data ODC dan kunci pada database lalu di tampilkan pada bot telegram	Sesuia
Cari	Melihat kunci ODC yang di perlukan	Menampilkan data ODC dan kunci pada *database*	Sesuai

Berdasarkan Tabel 2 dapat di simpulkan pengujian pada spesifikasi kedua-keduanya sudah sesuai dengan hasil yang diharapkan dan memiliki tingkat keberhasilan 100%. Detail bukti *screenshot expected result* atau hasil yang diharapkan pada pengujian fungsionalitas ditampilkan pada pada Tabel 2.

C. Pengujian Performasi Sistem

Pengukuran *respon time* merupakan scenario untuk melakukan pengujian dari performa bot telegram. *Respon time* dihitung berdasarkan hasil *respon reply* dari chat bot Karena bot ini menggunakan *server* yang di setting menggunakan metode *webhook* maka tidak perlu dilakukan pengecekan secara periodic terhadap *request* yang masuk ke bot, namun apabila ada *request* yang masuk ke bot, *server* akan merespon secara *realtime.* Pesan dibalas secara *realtime* berdasarkan script yang berjalan di *server*. Adapun hasil pengukuran *respon time* ditampilkan pada Tabel 3

TABEL 3
RESPON TIME

No	Jenis Percobaan	*Respon Time* (detik) *Berdasarkan Balasan Dari Bot*										Rata-rata
		1	2	3	4	5	6	7	8	9	10	(detik)
1	Masukkan *command* /start	0,2	0,3	0,3	0,3	0,3	0,4	0,2	0,3	0,3	0,3	0,3
2	Klik "Info"	0,7	0,6	0,6	0,5	0,6	0,5	0,5	0,5	0,6	0,6	0,6
4	Data masuk ke Bot telegram dan sheet	2,2	1,5	2,7	1,9	1,9	2,3	1,4	2,8	1,7	1,3	2,0
5	Input perintah cari	1,1	1,2	1,5	1,5	1,2	1	0,8	1,3	1	0,9	1,2
6	Input perintah tambah	1,1	1	1,6	1	1,1	1,2	0,7	1,1	1,1	1,5	1,1
7	*Input perintah edit*	2,3	2,1	1,4	2,1	1,8	1,8	1,7	1,5	1,5	2,3	1,9

Berdasarkan Tabel 3 telah didapatkan nilai *respon time* dari tiap fitur pada bot telegram. Pengukuran *respon time* ini dilakukan dengan menggunakan koneksi data seluler. Hasil dari pengujian *respon time* didapatkan nilai rata-rata dari 10 kali percobaan pada setiap jenis percobaan sebesar 1.1 detik.

V KESIMPULAN

A. Kesimpulan

Berdasarkan hasil perancangan, pengujian dan analisa yang telah dilakukan maka dapat diambil beberapa kesimpulan sebagai berikut :

1. Bot telegram yang telah dirancang pada proyek akhir ini dapat menerima pengecekan kunci ODC Witel Tasikmalaya secara otomatis yang terhubung ke

google *spreadsheet*s berjalan dengan baik sebagaimana semestinya. Dibuktikan dengan pengujian fungsionalitas spesifikasi pertama yang memiliki tingkat keberhasilan 100%.

2. Bot telegram yang telah dirancang pada proyek akhir ini dapat menerima informasi kunci ODC oleh admin manager dan teknisi. Dibuktikan dengan pengujian fungsionalitas spesifikasi kedua yang memiliki tingkat keberhasilan 100%.
3. Bot telegram yang telah dirancang pada proyek akhir ini dapat merealisasikan aplikasi BOT Telegram berguna untuk memudahkan admin manager dan teknisi bagi perusahaan PT.Telkom Witel Tasikmalaya dalam kebutuhan kunci ODC. Dibuktikan dengan pengujian fungsionalitas spesifikasi ketiga yang memiliki keberhasilan 100%.
4. Bot telegram yang telah dirancang pada proyek akhir ini dapat menampilkan hasil kebutuhan kunci ODC yang di perlukan oleh teknisi dengan cara mengunakan bot Telegram yang ada di daerah PT. Telkom Witel Tasikmalaya

B. Saran

Berdasarkan hasil pembangunan Proyek Akhir ini, dapat disampaikan beberapa saran untuk pengembangan selanjutnya sebagai berikut :

1. Sebaiknya menggunakan *database* tambahan karena apabila semua *database* kunci ODC ada di google *spreadsheet* akan memperlambat proses eksekusi pada bot telegram.
2. Menambahkan fitur sesuai dengan kebutuhan yang diinginkan.

REFERENSI

[1] D. A. Arham And N. A. Syarif , Analisis Redaman Optical Distribution Cabinet (Odc) Menuju Optical Distribution Point (Odp) Menggunakan Metode Link Power Budget, Makassar: UNIVERSITAS MUHAMMADIYAH MAKASSAR, 2018.

[2] T. Indonesia, "Profil dan Riwayat Singkat The Telkom Hub," Telkom Indonesia, 1 Januari 2020. [Online]. Available: https://www.telkom.co.id/sites/about-telkom/id_ID/page/profil-dan-riwayat-singkat-22. [Accessed 2 Agustus 2020].

[3] S. P. Nova And D. M. Firdaus, M.Si, "Efektivitas Komunikasi Aplikasi Telegram Sebagai Media Informasi Pegawai Pt.Pos Indonesia (Persero) Kota Pekanbaru," *JOM FISIP,* vol. 5, no. 1, pp. 1-11, 2018.

[4] F. Fitriansyah and A. , "Penggunaan Telegram Sebagai Media Komunikasi Dalam Pembelajaran Online," *Cakrawala-Jurnal Humaniora,* vol. 20, no. 2, pp. 111-117, 2020.

[5] A. D. Kusuma, Penggunaan Telegram Bot Pada Telegram Messenger Dengan Metode Webhooks Untuk Sistem Peminjaman Infrastruktur Di UIN Maulana Malik Ibrahim Malang, Malang: UIN Maulana Malik Ibrahim, 2019.

[6] G. C. Lenardo, H. and Y. Irawan, "Pemanfaatan Bot Telegram Sebagai Media Informasi Akademik di STMIK Hang Tuah Pekanbaru," *JTIM : Jurnal Teknologi Informasi dan Multimedia,* vol. 1, no. 4, pp. 351-357, 2020.

[7] A. Cokrojoyo, J. Andjarwirawan and A. Noertjahyana, "Pembuatan Bot Telegram Untuk Mengambil Informasi dan Jadwal Film Menggunakan PHP," *Jurnal Informatika,* vol. 1, no. 1, pp. 1-4, 2020.

[8] L. N. Gunawan, J. Anjarwirawan and . A. Handojo, "Aplikasi Bot Telegram Untuk Media Informasi Perkuliahan Program Studi Informatika-Sistem Informasi Bisnis Universitas Kristen Petra," *Jurnal Informatika ,* vol. 1, no. 1, pp. 1-6, 2020.

[9] M. Reynaldi, S. . A. Khairi, N. . G. Hendarman and . F. I. Nugroho, "Sistem Informasi Berbasis Bot Telegram Sebagai Media Sosialisasi Keselamatan Berkendara," *Journal of Software Engineering, Information and Communication Technology,* vol. 1, no. 1, pp. 27-32, 2020.

[10] S. D. Nastiti, D. N. Ramadan and . R. Tulloh, "Sistem Monitoring Untuk Laporan Gangguan Indihome Dengan Bot Telegram," *e-Proceeding of Applied Science,* vol. 7, no. 5, pp. 1892-1906, 2021.

[11] K. S. Widi and M. S. Utomo, "Sistem Rekomendasi Makanan Khas Jawa Tengah Berbasis Aplikasi Telegram," *JIRE (Jurnal Informatika & Rekayasa Elektronika) ,* vol. 4, no. 2, pp. 181-188, 2021.

[12] T. Istiana, R. I. A, B. Dharmawan and B. Prakoso, "Pengembangan Sistem Diseminasi Prakiraan Cuaca Menggunakan Aplikasi Bot Telegram dengan Metode Webhook," *Elektron Jurnal Ilmiah,* vol. 12, no. 1, pp. 41-47, 2020.

[13] I. Handayani, H. Kusumahati and A. N. Badriah, "Pemanfaatan Google Spreadsheet Sebagai Media Pembuatan Dashboard pada Official Site iFacility di Perguruan Tinggi," *Jurnal Informatika,* vol. 7, no. 2, pp. 177-186, 2017.

[14] A. Rahman, Rancang Bangun Sistem Informasi Penjadwalan Menggunakan Google Apps Script Pada Politeknik Kampar, Pekanbaru: Universitas Islam Negeri Sultan Syarif Kasim Riau, 2020.