

## **PENGEMBANGAN TELEGRAM BOT SEBAGAI SOLUSI PENGOLAHAN DATA KOLAM PADA BUDI DAYA UDANG UNTUK JALA TECH**

**Mohammad RA Firdaus<sup>1</sup>, Chanifah I Ratnasari\*<sup>2</sup>**

<sup>1,2</sup> Program Studi Informatika, Fakultas Teknologi Industri, Universitas Islam Indonesia, Indonesia  
Email: <sup>1</sup>[mohammad.firdaus@students.uii.ac.id](mailto:mohammad.firdaus@students.uii.ac.id), <sup>2</sup>[chanifah.indah@uii.ac.id](mailto:chanifah.indah@uii.ac.id)

(Naskah masuk: 01 Juli 2022, diterima untuk diterbitkan: 05 Juli 2022)

### **Abstrak**

Jala Tech merupakan salah satu perusahaan penyedia layanan budi daya udang digital dengan bantuan teknologi. Jala Tech menawarkan layanan seperti kemudahan dalam manajemen tambak udang, pengelolaan data kolam udang yang dapat direpresentasikan menggunakan grafik serta sistem pendukung keputusan untuk pengguna. Pengolahan data kolam dapat dilakukan menggunakan aplikasi Jala berbasis *web* dan *mobile*. Namun, pada kondisi di mana petambak tidak dapat menggunakan aplikasi Jala dikarenakan keterbatasan gawai atau jaringan internet, maka petambak udang akan kesulitan dalam mengolah data kolam pada budi daya udang mereka. Maka dari itu, tujuan dari penelitian ini adalah untuk mengembangkan bot Telegram yang dapat menunjang pengolahan data kolam udang. Telegram dipilih karena popularitas dan mudah untuk digunakan melalui *smartphone* dan komputer, serta tidak memerlukan kapasitas internet yang besar untuk penggunaannya dibandingkan dengan aplikasi Jala. Bot Telegram merupakan layanan *chatbot* disediakan oleh Telegram yang memungkinkan para pengguna dan pengembang untuk memanfaatkan dan menciptakan layanan *chatbot* sesuai yang diinginkan. Bot Telegram Jala memuat fitur utama seperti penyambungan akun Jala, pencatatan data, dan pemeriksaan hasil pencatatan data. Konsep *state management* juga diterapkan pada bot Telegram Jala yang berguna untuk mengatur aksi pengguna agar sesuai dengan prosedur pada fitur yang diakses. Bot Telegram Jala dikembangkan dengan bahasa pemrograman JavaScript, dan *Redis* sebagai *database* untuk menyimpan *state* pengguna. Metode pengembangan *scrum* juga diterapkan pada penelitian ini karena siklus pengembangannya yang cepat dan adaptif. Dibangunnya bot Telegram pada penelitian ini diharapkan menjadi solusi yang selalu tersedia bagi petambak yang memiliki kendala dalam menggunakan aplikasi Jala, tanpa mengkhawatirkan siklus budi daya udang menjadi terhambat.

**Kata kunci:** *budi daya udang, jala tech, sistem informasi, telegram, telegram bot*

## **DEVELOPMENT OF TELEGRAM BOT AS A SOLUTION FOR POND DATA PROCESSING IN SHRIMP FARMING FOR JALA TECH**

### **Abstract**

*Jala Tech is one of the technology providers of digital shrimp farming services. Jala Tech offers easy shrimp farm management, shrimp pond data processing that can be represented using graphs and decision support systems for users. Processing of pond data can be done using Jala web and mobile applications. However, if farmers are unable to access the Jala apps due to limited devices or internet networks, shrimp farmers will struggle to process pond data on their shrimp farming. Therefore, this study aims to develop a Telegram bot that can support shrimp pond data processing. Telegram was chosen because it is popular and easy to access through smartphones and computers. Unlike Jala apps, it does not require a lot of internet capacity to use. Telegram bot is a chatbot service provided by Telegram that allows users and developers to utilize and create chatbot services as desired. Jala's Telegram bot contains main features such as connecting to Jala account, recording data, and checking the recorded data. State management is also used to organize user actions in accordance with the guidelines on the features accessed. Jala's Telegram bot is developed with JavaScript programming language, and Redis as a database to store user state. Scrum development method is also applied to this study because of its fast and adaptive development cycle. The Telegram bot built in this study is expected to be a solution that always available for farmers who have problems accessing Jala apps without worrying about disrupting the shrimp farming cycle.*

**Keywords:** *information system, shrimp farming, jala tech, telegram, telegram bot*

## 1. PENDAHULUAN

Jala Tech merupakan salah satu perusahaan yang bergerak pada bidang akuakultur, khususnya pada sektor budi daya udang yang berkomitmen untuk membantu petambak udang dalam meningkatkan produktivitas dan menciptakan budi daya udang yang berkelanjutan. Permasalahan kompleks dalam budi daya udang yang menyebabkan kegagalan diharapkan dapat diminimalisir dengan menggunakan bantuan teknologi yang mendukung petambak mengambil langkah pencegahan. Jala Tech mengembangkan produk alat ukur untuk kolam udang berbasis *Internet of Things* (IoT) bernama Jala Baruno. Jala Baruno merupakan alat ukur kualitas air pintar pada kolam udang yang terhubung dengan internet, yang mana data hasil pengukuran kolam udang dapat disimpan secara *real-time* ke *database* Jala. Data hasil pengukuran tersebut dapat dipantau dan dikontrol melalui aplikasi Jala yang berbasis *web* dan *mobile*, sehingga pengukuran dan pencatatan kualitas air pada kolam udang dapat dilakukan dengan efektif dan efisien melalui *smartphone*, laptop, atau komputer.

Aplikasi Jala memungkinkan para petambak udang untuk memproses segala hal dan kebutuhan dalam siklus budi daya udang mereka secara daring dan fleksibel. Aplikasi Jala memuat fungsi yang dapat menunjang pengelolaan tambak budi daya udang digital seperti pemantauan dan pencatatan data kualitas air pada kolam udang, pengelolaan keuangan tambak, pengelolaan stok pakan dan bahan kimia tambak, serta masih banyak lagi fungsi terkait pengelolaan tambak udang. Dengan hadirnya aplikasi Jala, diharapkan aplikasi ini dapat menunjang para petambak dalam pemeliharaan tambak budi daya udang mereka secara digital dengan semestinya. Namun, terdapat beberapa petambak yang memiliki keterbatasan gawai dan jaringan internet untuk menggunakan aplikasi Jala. Hal ini menjadi kendala bagi para petambak yang hendak menggunakan layanan aplikasi Jala. Oleh karena itu, diperlukan solusi lain yang dapat membantu dalam pencatatan dan pemantauan data tambak yang memungkinkan dalam kondisi minim sinyal ataupun gawai dengan spesifikasi rendah.

Telegram merupakan aplikasi perpesanan instan yang bersifat *open source* dan memungkinkan para penggunanya untuk bertukar pesan baik dalam bentuk perpesanan pribadi maupun grup. Telegram juga menyediakan bot API yang mendukung para pengembang untuk mengembangkan bot pada aplikasi telegram yang dapat diintegrasikan dengan layanan aplikasi pihak ketiga. Telegram dipilih karena aplikasi ini cukup populer dengan fitur *chatbot*nya, juga dapat digunakan dari berbagai perangkat dengan mudah seperti *smartphone* dan komputer [1]. Selain itu, Telegram merupakan salah satu aplikasi perpesanan instan yang cukup ringan

dan bersahabat untuk digunakan dari berbagai macam tipe *smartphone*. Pengembangan bot pada Telegram telah diimplementasikan dalam berbagai bidang seperti integrasi bot dengan IoT [2].

Berdasarkan [3], penelitian sebelumnya yang memiliki topik serupa dengan penelitian ini, bot Telegram diintegrasikan dengan IoT sebagai platform untuk memantau kualitas air pada kolam udang, selanjutnya *Early Warning System* (EWS) dimanfaatkan untuk memprediksi kualitas air pada kolam udang berdasarkan waktu tertentu. Kemudian pada penelitian [4], yang juga mengembangkan sebuah *chatbot* namun tidak menggunakan layanan bot dari Telegram. Pada penelitian tersebut *chatbot* dikembangkan dengan tujuan untuk membantu menjawab pertanyaan-pertanyaan mahasiswa di Institut Bisnis dan Informatika Kosgoro dengan sistem bot berbasis *Frequently Asked Questions* (FAQ) dan menerapkan model *Long Short Term Memory* (LSTM).

Pada penelitian ini dibangun bot Telegram dengan metode *scrum*, yang mana metodologi pengembangan sistem ini memiliki fleksibilitas dan kelincahan yang tinggi. Tujuan dari penelitian ini adalah membantu petambak dalam pemantauan dan pencatatan data budi daya udang secara daring, yang memungkinkan pemantauan dan pencatatan dilakukan dalam kondisi minim sinyal atau spesifikasi gawai yang kurang mumpuni. Bot Telegram dikembangkan supaya petambak dapat mencatat hasil pengukuran data kualitas air pada kolam udang yang telah diukur menggunakan Jala Baruno. Dengan adanya bot Telegram ini, apabila petambak dalam kondisi yang tidak memungkinkan untuk menggunakan aplikasi Jala, seperti pada kondisi minim sinyal atau gawai yang digunakan kurang mumpuni, maka bot ini dapat menjembatani permasalahan tersebut. Selain itu, bot Telegram dikembangkan agar dapat mencatat data stok pakan dan mencatat data *anco* (wadah pakan) untuk mengetahui nafsu makan udang. Hasil pencatatan data-data tersebut disimpan ke *database* Jala yang nantinya juga dapat diolah melalui aplikasi Jala. Bot Telegram pada penelitian ini juga dapat menampilkan hasil pencatatan suatu data pada hari terkini supaya para petambak dapat memeriksa rekapan hasil pencatatannya pada hari tersebut.

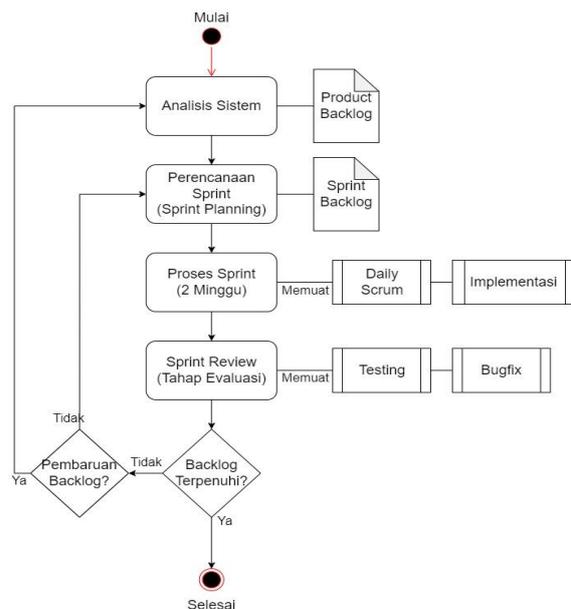
Dalam segi keilmuan, penelitian ini diharapkan dapat menjadi referensi bagi penelitian sejenis, bahwa bot pada Telegram dapat dikembangkan menjadi sebuah sistem aplikasi berskala kecil yang memiliki fungsional inti seperti aplikasi Jala. Dalam segi industri, pengembangan bot Telegram pada penelitian ini diharapkan dapat menjadi solusi alternatif yang dapat diandalkan bagi Jala Tech.

## 2. METODE PENELITIAN

Dalam penelitian ini, metode yang digunakan dalam pengembangan perangkat lunak untuk

menunjang siklus pengembangan atau *Software Development Life Cycle* yaitu *Scrum*. *Scrum*, metodologi yang cukup populer, seperti metodologi pengembangan perangkat lunak lainnya, berfokus pada peningkatan proses perangkat lunak, kualitas, kinerja, nilai bisnis, kegunaan, dan efisiensi sekaligus mengurangi biaya, risiko, dan ketidakpastian [5]. Tidak seperti pada beberapa pendekatan konvensional, *scrum* juga bertujuan untuk meningkatkan aspek-aspek seperti kelincihan, kepercayaan, motivasi, tanggung jawab, dan transparansi [6].

Secara garis besar, urutan tahapan yang dilakukan dalam penelitian adalah sebagai berikut: (1) identifikasi permasalahan pada aplikasi Jala, (2) merumuskan gagasan dan ide untuk dijadikan solusi, (3) analisis kebutuhan-kebutuhan sistem, (4) merencanakan dan membagi tugas untuk dikerjakan dalam satu *sprint*, (5) inialisasi *sprint* dan implementasi pengembangan bot Telegram, dan (6) pengujian, *review*, dan evaluasi *sprint*. Digunakannya metode *scrum* karena pengembangan bot Telegram ini dilakukan dalam waktu yang cukup singkat dan tidak memiliki bobot pengembangan proyek yang begitu tinggi dan kompleks. *Scrum* merupakan metode yang tepat untuk digunakan karena dapat mengantisipasi perubahan fungsional sistem yang mungkin timbul selama pengembangan bot Telegram [7]. Jika dibandingkan dengan metode *waterfall*, *scrum* memberikan fleksibilitas yang tinggi dan kecepatan dalam pengembangannya sehingga siklus pengembangan dapat berjalan lebih mulus dan semestinya [6]. Metodologi penelitian dengan menggunakan metode *scrum* pada penelitian ini dapat dilihat pada Gambar 1.



Gambar 1. Diagram Alur Metode Scrum

Berdasarkan Gambar 1, proses siklus pengembangan bot Telegram mencakup tahapan berikut:

a. Analisis Sistem

Pada tahapan analisis sistem, dilakukan perumusan dan identifikasi kebutuhan fungsional maupun non fungsional sistem. Kebutuhan fungsional sistem merupakan suatu aksi dalam sebuah proses bisnis yang dikemukakan untuk dijadikan sebuah fitur pada perangkat lunak yang menjadi dasar fungsional sistem untuk memproses atau mengolah suatu data yang dibutuhkan [8]. Kebutuhan non fungsional merupakan fitur-fitur yang dapat menunjang aksi prosedural pada kebutuhan fungsional [8].

Pada tahap ini, juga sekaligus dibuat desain arsitektur bot Telegram Jala, diagram alur pencatatan data, pemeriksaan data bot Telegram, serta mendefinisikan *state object* yang dibutuhkan untuk memanipulasi *state* menggunakan *Redis* dan API Jala yang dibutuhkan untuk mengintegrasikan data dari *database* Jala dengan Telegram. Kemudian semua hasil identifikasi tersebut dirangkum dan dijadikan sebuah *product backlog*, yang mencakup semua kebutuhan fungsional sistem yang menjadi fitur pada sebuah produk yang akan dikembangkan nantinya [9].

b. Sprint Planning

Pada tahap ini, dilakukan kegiatan perencanaan yang mengonversi *Product Backlog* menjadi satu tugas untuk dikerjakan dalam interval waktu *sprint* yang telah ditentukan. Hasil dari tahapan ini adalah terciptanya *Sprint Backlog* yang memuat perencanaan berisi tugas-tugas yang perlu diselesaikan oleh para pengembang [10]. *Sprint Backlog* dapat diperbarui apabila terdapat pembaruan pada *Product Backlog* dan telah dirancang pada tahap *Sprint Planning*. Pada penelitian ini, *Sprint Planning* diterapkan dalam rangka memecah beberapa entitas pada *Product Backlog* guna dipetakan menjadi beberapa tugas untuk diselesaikan dalam satu *sprint*. Satu *sprint* dalam siklus *scrum* di Jala dijalankan dalam kurun waktu dua minggu [10]. Pada minggu pertama dimanfaatkan untuk tahap pengembangan, sedangkan pada minggu kedua dan mendekati waktu berakhirnya suatu *sprint* dimanfaatkan untuk *review* dan perbaikan.

c. Sprint Process

*Sprint Process* umumnya merupakan tahapan implementasi dalam *scrum*, yang dilakukan berdasarkan *Sprint Backlog* yang sudah dirancang pada tahap *Sprint Planning*. Pada penelitian ini, *sprint process* memuat tiga hal, yaitu *daily scrum*, implementasi, dan *testing*. *Daily scrum* adalah kegiatan untuk menyampaikan progres dari tugas-tugas yang telah dikerjakan pada hari sebelumnya dan juga menyampaikan rencana aktivitas pada hari terkini [10]. Selanjutnya, pengujian secara fungsional dilakukan dalam satu *sprint*, tepat pada

minggu terakhir. Pengujian ini bertujuan untuk menguji dan memvalidasi salah satu fungsionalitas pada bot Telegram yang telah diselesaikan berdasarkan kesesuaian *Product Backlog*.

d. *Sprint Review*

*Sprint Review* merupakan kegiatan untuk mengevaluasi hasil implementasi dari tugas-tugas sesuai *Sprint Backlog*. Selain itu, pada tahap ini disampaikan progres implementasi kebutuhan fungsional sesuai *Product Backlog* terkait kebutuhan fungsional yang telah diselesaikan dan yang belum diselesaikan. Jika terdapat kebutuhan yang belum diselesaikan, maka *Scrum Master* akan menginisialisasi sebuah *sprint* baru untuk mengcover semua kebutuhan yang perlu diselesaikan. Di dalam *Sprint Review* juga dilakukan *Sprint Retrospective* yang membahas analisis pembaruan kebutuhan fungsional pada *Product Backlog* untuk diimplementasikan pada *sprint* yang baru [10].

### 3. PERANCANGAN SISTEM

Sesuai dengan permasalahan pada aplikasi Jala yang telah diungkapkan dalam bab pendahuluan, bot Telegram Jala harus memuat fungsionalitas dalam melakukan pengolahan data kolam pada tambak udang. Cakupan dalam proses pengolahan data kolam udang adalah:

- Pencatatan data kualitas air
- Pencatatan data kimia
- Pencatatan data plankton
- Pencatatan data mikrobiologi

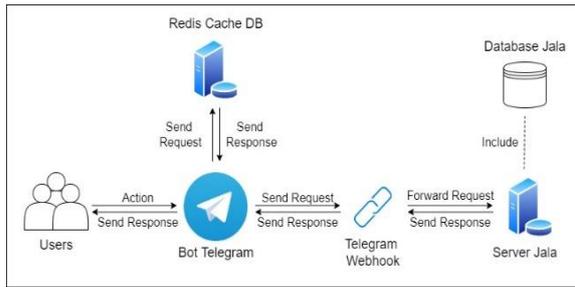
- Pencatatan data pakan
- Pencatatan data *anco* (wadah pakan)
- Pemeriksaan pemasukan data pada hari terkini

Berdasarkan hal tersebut, dapat diidentifikasi kebutuhan fungsional pada bot Telegram Jala yang dijabarkan pada Tabel 1.

Gambaran arsitektur bot Telegram dapat dilihat pada Gambar 2. Dimulai dari pengguna yang melakukan pemasukan atau aksi dalam menggunakan bot, yang mana setiap aksi pengguna akan diterapkan *state management* yang berguna untuk mengatur perilaku bot dalam mengirimkan respon kepada pengguna. *State management* juga sebagai sanitasi aksi yang mana pengguna harus melakukan aksi sesuai prosedur pada fungsi yang ditetapkan. Selain itu, *state management* juga berguna untuk menyimpan data yang bersifat sementara (*caching*) ketika pengguna melakukan pencatatan data kolam udang. *State management* dan *caching* data diintegrasikan dengan *Redis*, karena *Redis* dapat mendukung sistem *caching* berbasis objek sehingga ketika bot hendak mendapatkan data dari *Redis*, data tersebut dapat dengan mudah diolah seketika hanya dengan melakukan *parsing* ke dalam bentuk json [11]. Cuplikan dari kode program *state management* ditunjukkan pada Gambar 3.

Tabel 1. Kebutuhan Fungsional Bot Telegram Jala

Fitur	Deskripsi	Fitur	Deskripsi
Sambungkan akun	Menyambungkan akun Telegram pengguna dengan akun yang terdaftar pada aplikasi Jala.	Pemasukan data <i>anco</i>	Melakukan pencatatan data <i>anco</i> ( <i>feeding tray</i> ). <i>Anco</i> merupakan alat untuk mengukur tingkat konsumsi pakan pada udang. Pencatatan <i>anco</i> dilakukan saat ada pakan yang telah diberikan pada sebuah kolam.
Putuskan akun	Memutuskan akun Telegram pengguna dengan akun yang terdaftar pada aplikasi Jala.	Cek data kualitas air	Memeriksa data kualitas air pada kolam yang telah dicatat pada hari terkini.
Pemasukan data kualitas air	Melakukan pencatatan data pada kolam dengan jenis data kualitas air yang memuat parameter seperti pH air, suhu, salinitas, oksidasi, dan lain-lain.	Cek data kimia	Memeriksa data kimia pada kolam yang telah dicatat pada hari terkini.
Pemasukan data kimia	Melakukan pencatatan data pada kolam dengan jenis data kimia yang memuat parameter seperti <i>amonia</i> (NH <sub>3</sub> ), kalsium, magnesium, dan lain-lain.	Cek data plankton	Memeriksa data plankton pada kolam yang telah dicatat pada hari terkini
Pemasukan data plankton	Melakukan pencatatan data pada kolam dengan jenis data plankton yang memuat parameter seperti <i>algae</i> , <i>floc</i> , <i>protozoa</i> , dan lain-lain.	Cek data mikrobiologi	Memeriksa data mikrobiologi pada kolam yang telah dicatat pada hari terkini.
Pemasukan data mikrobiologi	Melakukan pencatatan data pada kolam dengan jenis data mikrobiologi yang memuat parameter seperti vibrio, intensitas cahaya, total bakteri, dan lain-lain.	Cek data pakan	Memeriksa data pakan pada kolam yang telah dicatat pada hari terkini.
Pemasukan data pakan	Melakukan pencatatan data pakan pada kolam, seperti pencatatan total kuantitas pakan yang diberikan, status puasa pada kolam, dan keterangan tambahan.	Cek data <i>anco</i>	Memeriksa data <i>anco</i> pada kolam yang telah dicatat pada hari terkini.



Gambar 2. Gambaran Arsitektur Bot Telegram

```

let context = await
redisClientGet(message.chat.id).catch(
() => null);
context = {
  context: '/start-menu',
  state: '',
  isConnected: false,
  data: {},
};
await redisClientSet(message.chat.id,
JSON.stringify(context));

```

Gambar 3. Baris Kode State Management

Bot Telegram menyimpan *state* pengguna dan mengirimkan *request* ke *Redis*, lalu bot Telegram juga mengirimkan *request* yang diproses melalui Telegram *webhook*. *Webhook* merupakan suatu metode yang memungkinkan bot Telegram atau *server* Telegram untuk berintegrasi secara langsung dengan suatu aplikasi. Umumnya *webhook* digunakan untuk bertukar data secara langsung dengan aplikasi terkait serta melakukan validasi data yang berhubungan dengan aplikasi tersebut [12]. Pada penelitian kali ini, *webhook* digunakan karena bot Telegram Jala memerlukan pertukaran data dengan cepat dan dapat menerima banyak *request* dalam satu waktu [13]. Setelah *request* diterima oleh *webhook*, *webhook* akan meneruskan *request* tersebut ke *server* Jala dan menerima respon dari *server* Jala sesuai *request* terkait, yang kemudian diteruskan lagi kepada bot Telegram. Terakhir bot Telegram akan mengolah respon tersebut untuk diteruskan kembali kepada pengguna.

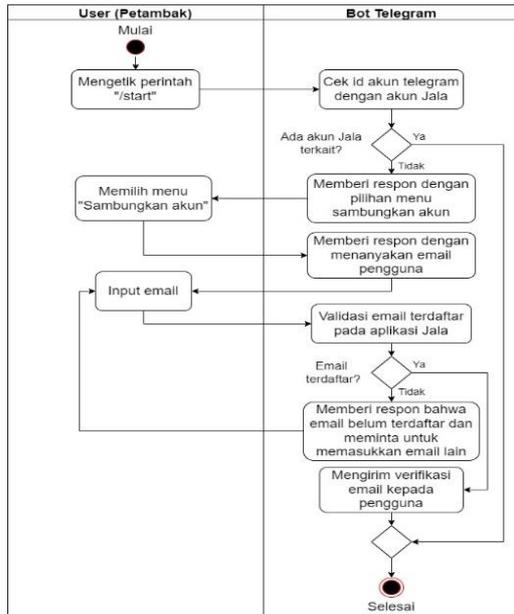
#### 4. HASIL DAN PEMBAHASAN

Sesuai visi dan misi Jala Tech, Jala ingin mewujudkan budi daya tambak udang yang dapat diproses secara digital. Untuk mewujudkan proses budi daya udang secara digital, maka dibutuhkan sebuah gawai berupa *smartphone* atau komputer/laptop dan jaringan serta kecepatan internet yang baik. Akan tetapi sesuai permasalahan yang telah dikemukakan sebelumnya, diperlukan sebuah solusi alternatif bagi para petambak udang (dan tim bantuan lapangan Jala) yang memiliki kendala dari segi gawai dan jaringan internet. Maka dari itu, dikembangkan sebuah bot Telegram yang mampu melakukan pencatatan serta pemantauan data yang berkaitan dengan kolam dan tambak udang.

Bot Telegram Jala bertindak sebagai aplikasi berbasis *middleware* yang memproses setiap masukan yang diterima dari pengguna dan meneruskan masukan tersebut yang menjadi *request* ke *server* Jala melalui *webhook*. Bot Telegram menggunakan API yang dapat diidentifikasi oleh *token* yang dihasilkan oleh Bot Father untuk berkomunikasi dengan *server* Telegram. Bot Father merupakan bot yang sudah disediakan oleh Telegram untuk memberikan kebebasan kepada para pengembang yang hendak mengembangkan dan mendaftarkan suatu bot Telegram [14]. Bot Telegram dikembangkan dengan menggunakan bahasa pemrograman JavaScript. JavaScript merupakan bahasa pemrograman berbasis *object-oriented* yang ringan dan dinamis. Bahasa ini umum digunakan dalam pengembangan perangkat lunak berbasis web yang sangat populer di kalangan pengembang. Namun, JavaScript juga cukup populer untuk digunakan dalam pengembangan aplikasi berbasis lainnya, seperti aplikasi berbasis *mobile*, *desktop*, hingga pengembangan *game* berbasis web. JavaScript mengedepankan pemrograman yang imperatif, *object-oriented programming* yang dinamis, serta gaya pemrograman secara fungsional (*functional programming*). Oleh karena itu, JavaScript merupakan salah satu bahasa pemrograman yang mudah dipelajari, namun juga menjadi bahasa yang sangat kuat dalam suatu pengembangan perangkat lunak [15], [16].

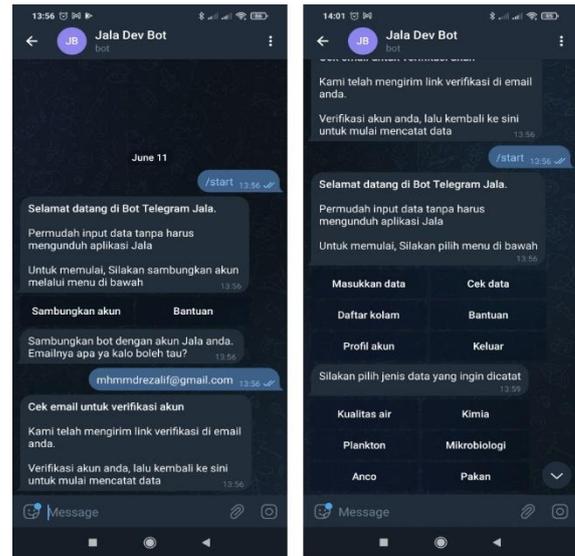
#### 4.1 Prosedur Penyambungan Akun Telegram dengan Akun JALA

Sebelum memproses pengolahan data kolam, pengguna perlu menyambungkan akun Telegramnya dengan akun Jala yang sudah terdaftar, dan memiliki tambak dan kolam di dalam akun Jala tersebut. Pengguna juga dapat memutuskan koneksi akun Telegramnya dengan akun Jala jika pengguna ingin mengganti akun Jala. Untuk memulai penggunaan bot Telegram, pengguna dapat mengetik perintah “/start” untuk memulai mengoperasikan bot. Setiap kali perintah tersebut dipanggil, bot Telegram Jala memeriksa akun Telegram pengguna dengan akun Jala yang terhubung. Jika belum ada akun Jala yang terhubung, maka bot akan memberikan respon kepada pengguna untuk menghubungkan akun Telegram pengguna dengan akun Jala terlebih dahulu. Diagram alur penyambungan akun Jala dapat dilihat pada Gambar 4.



Gambar 4. Alur Penyambungan Akun Jala

Untuk menghubungkan akun Jala dengan akun Telegram, pengguna dapat memasukkan email dari akun yang telah terdaftar pada aplikasi Jala. Kemudian bot memeriksa email tersebut dengan akun Jala yang tersedia sesuai email yang dimasukkan. Apabila akun Jala yang diminta sesuai dengan email yang tersedia, maka bot akan mengirimkan verifikasi Telegram melalui email. Ketika akun Telegram telah terverifikasi, pengguna dapat langsung menginisialisasi bot dengan mengetikkan perintah “/start” kembali untuk mendapatkan respon beserta pilihan menu untuk pengolahan data kolam udang. Pilihan menu dibuat dengan menggunakan fungsi *inline keyboard* yang disediakan oleh Telegram. Hal ini akan meminimalisir masukan berupa teks sehingga bot Telegram dapat dijalankan secara interaktif. *Inline keyboard* berfungsi untuk membuat ragam menu opsi dari setiap respon pesan bot dan hanya menggunakan *callback data* yang dapat diolah untuk memanipulasi sebuah pemanggilan fungsi berdasarkan *callback data* tersebut [17]. Tampilan bot penyambungan akun Jala ditunjukkan pada Gambar 5.

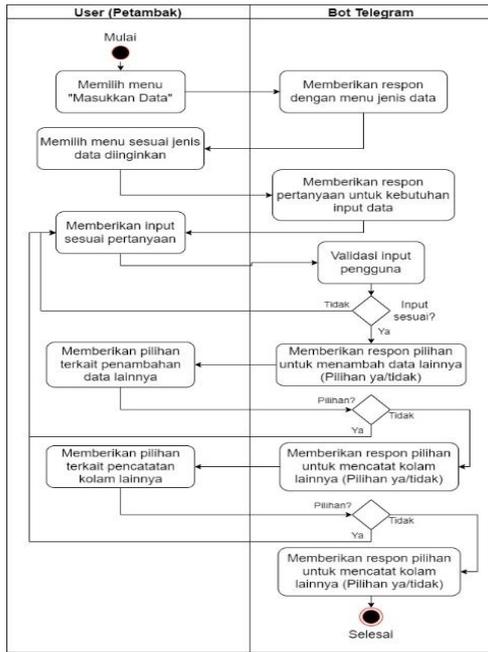


Gambar 5. Tampilan Bot Penyambungan Akun Jala

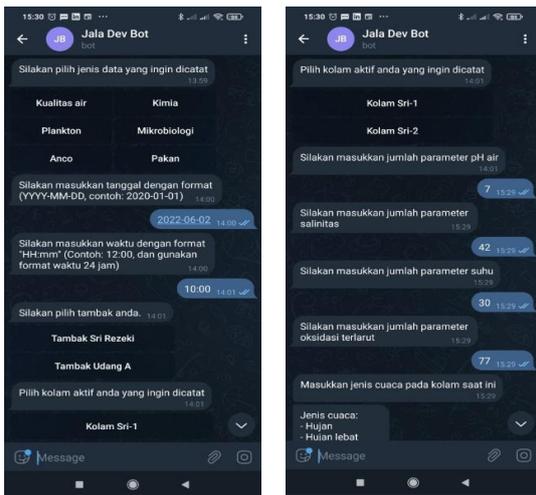
#### 4.2 Prosedur Pencatatan Data

Ketika akun Telegram pengguna sudah tersambung dengan akun Jala, pengguna dapat melakukan pencatatan maupun pemantauan data menggunakan bot Telegram. Akan tetapi harus dipastikan bahwa akun Jala yang dihubungkan memiliki profil tambak dan kolam yang dapat dibuat dengan menggunakan aplikasi Jala. Pengguna dapat menekan tombol “Masukkan data” untuk melakukan pencatatan data pada kolam udang, kemudian bot akan merespon dengan memberikan pesan serta pilihan menu untuk jenis data yang hendak dicatat. Pengguna dapat memilih kembali jenis data yang hendak dicatat dan memberikan masukan berdasarkan pertanyaan yang diberikan oleh bot Telegram terkait kebutuhan jenis data yang akan dimasukkan. Pencatatan data kolam dibuat dengan model percakapan supaya lebih interaktif dan tidak monoton. Gambaran alur pada prosedur pencatatan data dapat dilihat pada Gambar 6 dan tampilan bot pencatatan data dapat dilihat pada Gambar 7 dan Gambar 8. Secara umum, pertanyaan-pertanyaan terkait kebutuhan data sesuai jenis data untuk kolam antara lain:

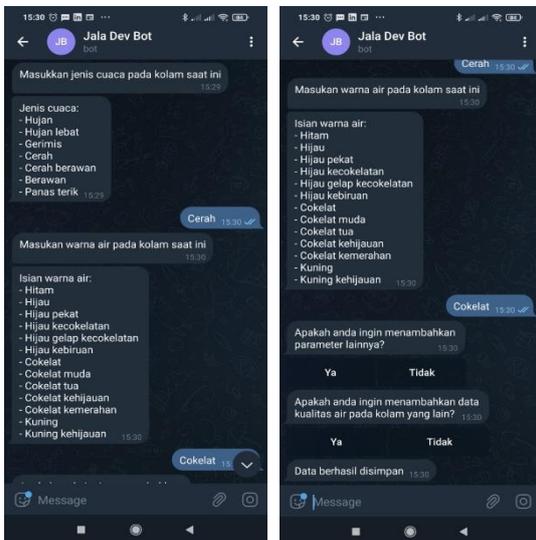
- Pertanyaan tanggal dan waktu pencatatan.
- Pilihan tambak dan kolam yang ingin dicatat.
- Pertanyaan masing-masing parameter dari setiap jenis data.
- Pertanyaan untuk penambahan data atau parameter tambahan.
- Pertanyaan untuk melakukan pencatatan pada kolam lainnya.



Gambar 6. Alur Pencatatan Data



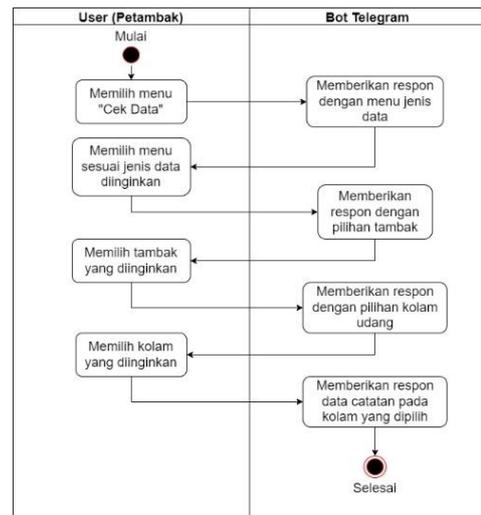
Gambar 7. Tampilan Bot Pencatatan Data (1)



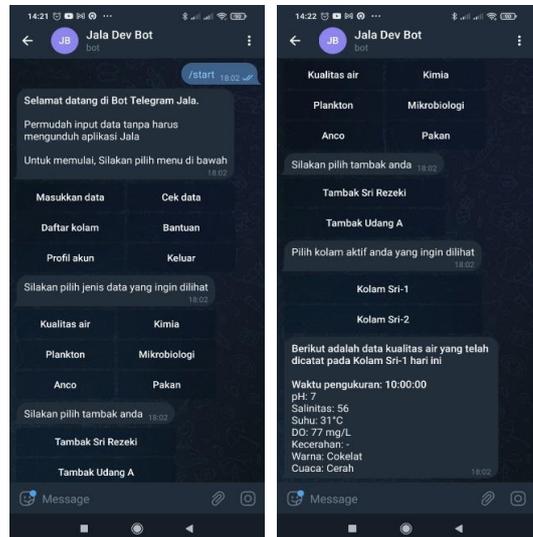
Gambar 8. Tampilan Bot Pencatatan Data (2)

### 4.3 Prosedur Pemeriksaan Data Pencatatan

Saat pengguna hendak memeriksa masukan data pada hari terkini, pengguna dapat mengetikkan perintah “/start” dan memilih menu “Cek Data”. Kemudian bot akan merespon dengan meminta pengguna untuk memilih tambak dari kolam yang hendak diperiksa. Setelah pengguna memilih tambak yang diinginkan, selanjutnya bot akan merespon kembali dengan meminta pengguna untuk memilih kolam yang diinginkan sesuai dari tambak yang dipilih sebelumnya. Setelah pengguna memilih kolam yang diinginkan, kemudian bot akan mengirimkan pesan yang memuat data kolam yang telah dicatat. Jika tidak ada data yang dicatat pada hari tersebut, maka bot akan mengirimkan pesan yang mengindikasikan bahwa tidak ada data yang dicatat pada hari tersebut. Apabila terdapat banyak data yang dicatat pada hari tersebut, maka data akan dikelompokkan sesuai dengan jam pencatatan data. Gambaran alur pada proses pemeriksaan pencatatan data dapat dilihat pada Gambar 9 dan tampilan botnya dapat dilihat pada Gambar 10.



Gambar 9. Alur Pemeriksaan Hasil Pencatatan Data



Gambar 10. Tampilan Bot Pemeriksaan Hasil Pencatatan Data

#### 4.4 Pengujian Bot Telegram JALA

Setelah bot Telegram Jala telah selesai diimplementasikan, selanjutnya adalah untuk menguji bot Telegram dengan menggunakan metode *blackbox testing*. Bot Telegram diuji untuk menyelaraskan kebutuhan fungsional yang diimplementasi dengan daftar kebutuhan fungsional dari *product backlog* [18]. *Blackbox testing* merupakan salah satu metode pengujian perangkat lunak yang bertujuan untuk melakukan validasi spesifikasi fungsional yang telah diimplementasikan ke suatu sistem perangkat lunak dengan spesifikasi kebutuhan fungsional sistem sesuai rancangan. *Blackbox testing* dilakukan dari sudut pandang pengguna untuk menemukan celah *error* dan kontradiksi dalam kebutuhan fungsional [19]. Tabel hasil pengujian dapat dilihat pada Tabel 2.

metode pengujian *blackbox* yang bertujuan untuk memvalidasi kebutuhan fungsional yang telah diimplementasikan dengan daftar kebutuhan fungsional yang terdapat pada *product backlog*. Bot Telegram Jala diharapkan dapat menjadi solusi yang diandalkan ketika terdapat kendala pada gawai dan jaringan internet, sehingga proses siklus budi daya udang dapat terus berjalan tanpa mengalami hambatan.

Tabel 2. Hasil Pengujian Bot Telegram Jala

Fitur	Test Case	Ekspektasi	Hasil
Sambungkan akun	Menyambungkan akun Telegram dengan email yang valid dan terdaftar pada akun Jala	Bot berhasil memeriksa email valid dan mengirimkan verifikasi	Success
Putuskan akun	Memutuskan akun Telegram dengan akun Jala	Bot berhasil memutuskan akun Telegram dengan akun Jala	Success
Pemasukan data	Memasukkan data dengan nilai yang valid dan format yang benar pada jenis data kualitas air	Bot berhasil menyimpan data kualitas air sesuai pemasukan data	Success
	Memasukkan data dengan nilai yang valid dan format yang benar pada jenis data kimia	Bot berhasil menyimpan data kimia sesuai pemasukan data	Success
	Memasukkan data dengan nilai yang valid dan format yang benar pada jenis data plankton	Bot berhasil menyimpan data plankton sesuai pemasukan data	Success
	Memasukkan data dengan nilai yang valid dan format yang benar pada jenis data mikrobiologi	Bot berhasil menyimpan data mikrobiologi sesuai pemasukan data	Success
	Memasukkan data dengan nilai yang valid dan format yang benar pada jenis data pakan	Bot berhasil menyimpan data pakan sesuai pemasukan data	Success
	Memasukkan data dengan nilai yang valid dan format yang benar pada jenis data anco	Bot berhasil menyimpan data anco sesuai pemasukan data	Success
Cek data	Melakukan pemeriksaan data setelah dilakukan pemasukan data	Bot berhasil menampilkan data sesuai hasil pencatatan yang telah dilakukan	Success

#### 5. KESIMPULAN

Bot Telegram Jala berhasil dikembangkan dengan tujuan sebagai solusi alternatif bagi para petambak udang untuk menunjang proses pengolahan data kolam budi daya udang seperti pencatatan data dan pemeriksaan hasil pencatatan data. Bot Telegram Jala dikembangkan dengan metode *scrum* dan bahasa pemrograman JavaScript, serta memanfaatkan *state management* dengan menggunakan *Redis*, juga *webhook* yang berfungsi untuk menjembatani komunikasi antara bot Telegram dengan *server* Jala. Setelah melalui tahap implementasi, bot Telegram Jala diuji dengan

## 6. DAFTAR PUSTAKA

- [1] T. Sutikno, L. Handayani, D. Stiawan, M. A. Riyadi and I. M. I. Subroto. 2016. "WhatsApp, viber and telegram: Which is the best for instant messaging?". *International Journal of Electrical and Computer Engineering*, vol.6(3), pp. 909–914, doi: 10.11591/ijece.v6i3.10271.
- [2] A.D Nobari, N. Reshadatmand and M. Neshati. 2017. "Analysis of telegram, an instant messaging service". *International Conference on Information and Knowledge Management, Proceedings*, vol.F131841, pp. 2035–2038, doi: 10.1145/3132847.3133132.
- [3] M. A. Wisnu., W. I. Wedashwara., and A. Mardiansyah., 2021. "Implementasi IoT Untuk Early Warning System(EWS) Pada Tambak Udang Vaname". *Journal of Computer Science and Informatics Engineering (J-Cosine)*, 5(1), pp.20-19.
- [4] A. Silvanie., and R. Subekti., 2022. "APLIKASI CHATBOT UNTUK FAQ AKADEMIK DI IBI-K57 DENGAN LSTM DAN PENYEMATAN KATA". *JIKO (Jurnal Informatika dan Komputer)*, 5(1), pp.19-27.
- [5] B. Grebić., and A. Stojanović., 2021. "Application of the Scrum Framework on Projects in IT Sector". *European Project Management Journal*, 11(2), pp.37-46.
- [6] P. Kokol., M. Kokol., And S. Zagoranski., 2020 "Software Development with Scrum: A Bibliometric Analysis and Profile Using bibliometrics for knowledge discovery View project Encyclopedia of Business and Professional Ethics View project". *Library Philosophy and Practice (e-journal)*, no. 4705.
- [7] S. Nova., A. Widodo., and B. Warsito., 2022. "Analisis Metode Agile pada Pengembangan Sistem Informasi Berbasis Website: Systematic Literature Review". *Techno.Com*, 21(1), pp.139-148.
- [8] M. Nugroho., A. Damayanti., M. Rifai., and S. Windarti., 2021. "PENGEMBANGAN APLIKASI QnA UNTUK PENDAFTARAN MAHASISWA BARU STMIK AKAKOM". *Journal of Information System Management (JOISM)*, 3(1), pp.18-23.
- [9] R. Riwanto., R. Wijaya., A. Prawoto., F. Andry., and T. Prayogo., "Development Point of Sales Using SCRUM Framework". *JOURNAL OF SYSTEMS INTEGRATION*, 10(1), doi: 10.20470/jsi.v10i1.359.
- [10] K. Bhavsar\*, D. Shah., and D. Gopalan., 2020. Scrum: An Agile Process Reengineering In Software Engineering. *International Journal of Innovative Technology and Exploring Engineering*, 9(3), pp.840-848.
- [11] D. Li., M. Dong., Y. Yuan., J. Chen., K. Ota, K. and Y. Tang., 2018. SEER-MCache: A Prefetchable Memory Object Caching System for IoT Real-Time Data Processing. *IEEE Internet of Things Journal*, 5(5), pp.3648-3660.
- [12] T. Davies., 2011. "Webhooks". Tersedia [<https://webhooks.pbworks.com/w/page/13385124/FrontPage>] diakses 10 Juni 2022.
- [13] H. Setiaji., and I. Paputungan., 2018. Design of Telegram Bots for Campus Information Sharing. *IOP Conference Series: Materials Science and Engineering*, 325, p.012005.
- [14] Telegram, 2020, "Bots: an introduction for developers". Tersedia [<https://core.telegram.org/bots>] diakses 10 Juni 2022.
- [15] K. Sun., and S. Ryu., 2018. Analysis of JavaScript Programs. *ACM Computing Surveys*, 50(4), pp.1-34.
- [16] MDN Web Docs, 2022, "JavaScript". Tersedia [<https://developer.mozilla.org/en-US/docs/Web/JavaScript>] diakses 10 Juni 2022.
- [17] M. Sinev., D. Neshko., A. Semenov., N. Karamysheva., and D. Trokoz., 2021. "Virtual assistant for remote submission of applications for enrollment in educational organizations". In: *Information Technologies and Intelligent Decision Making Systems II*. Moscow: CEUR Workshop Proceedings. ISSN 1613-0073. Tersedia [<http://ceur-ws.org/Vol-2922/paper010.pdf>] diakses 11 Juni 2022.
- [18] T. Hamilton., 2022, "What is BLACK Box Testing? Techniques, Example & Types," Tersedia [<https://www.guru99.com/black-box-testing.html>] diakses 11 Juni 2022.
- [19] T. Sandhika., 2018, "Pengujian Aplikasi dengan Metode Blackbox Testing Boundary Value Analysis (Studi Kasus: Kantor Digital Politeknik Negeri Lampung)". *Jurnal Informatika: Jurnal Pengembangan IT (JPIT)*, 3(1). DOI: <http://dx.doi.org/10.30591/jpit.v3i1.647>