

Pengembangan Antarmuka Pengguna (*Frontend*) pada Aplikasi Monitoring Makanan Kucing

1st Fasya Hanifah Putti
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia

fasyahanifah@student.telkomuniversity.ac.id

2nd Purba Daru Kusuma
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia

purbodaru@telkomuniversity.ac.id

3rd Rifqi Muhammad Fikri
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia

rifqimff@telkomuniversity.ac.id

Abstrak — Dalam era digital saat ini, pemantauan dan pengelolaan makanan kucing peliharaan secara efisien menjadi kebutuhan penting bagi pemilik kucing peliharaan. *Project* ini bertujuan untuk mengembangkan aplikasi monitoring kucing peliharaan berbasis *Internet of Things* (IoT) yang dinamakan "IFT *PetFeeder*". Menggunakan *framework* React Native, aplikasi ini memungkinkan pengembangan lintas platform dengan satu basis kode untuk iOS dan Android. Visual Studio Code digunakan sebagai lingkungan pengembangan terintegrasi dan JavaScript dipilih sebagai bahasa pemrograman utama karena fleksibilitas dan kompatibilitasnya. Aplikasi IFT *PetFeeder* menawarkan berbagai fitur, termasuk monitoring isi penampungan makanan, penjadwalan pemberian makan, pemberian makan secara *realtime*, dan terdapat kamera untuk memonitoring aktivitas kucing. Hasil *User Acceptance Testing* (UAT) menunjukkan bahwa aplikasi ini layak digunakan dengan rata-rata persentase kelayakan di atas 80% dengan hasil 82,4% dari pengujian langsung dan 87,5% dari pengujian virtual. Hasil ini menegaskan bahwa aplikasi IFT *PetFeeder* efektif dan mudah digunakan, serta mampu memenuhi kebutuhan dan ekspektasi pengguna dalam berbagai kondisi.

Kata kunci— Kucing, IoT (*Internet of Things*), IFT *Pet Feeder*, *Framework*, Aplikasi, *User Acceptance Testing* (UAT).

I. PENDAHULUAN

Dalam sebuah proyek pengembangan aplikasi *mobile*. Aplikasi merupakan hal yang sangat penting dalam memberikan gambaran umum mengenai tujuan, metodologi, dan teknologi yang akan digunakan. Proyek ini akan mengembangkan aplikasi monitoring makanan kucing peliharaan berbasis *Internet of Things* (IoT) yang akan memberikan solusi bagi pemilik kucing dalam memantau dan mengelola pemberian makan hewan peliharaan secara efisien.

Aplikasi ini dirancang menggunakan *framework* React Native, React Native adalah sebuah *framework* yang mampu menghasilkan aplikasi *mobile hybrid* untuk lintas platform. React Native cenderung memiliki satu tempat untuk menulis kode dalam bahasa pemrograman JavaScript ES6 dan membangun aplikasi *mobile* untuk iOS dan Android secara

bersamaan dengan satu kode yang ditulis untuk kedua *platform native* tersebut [1].

Proses pengembangan dilakukan menggunakan Visual Studio Code sebagai lingkungan pengembangan terintegrasi, yang dikenal luas karena keandalannya dalam pengembangan aplikasi berbasis JavaScript. JavaScript dipilih sebagai bahasa pemrograman karena fleksibilitas dan kompatibilitasnya yang luas dalam pengembangan aplikasi modern. Dalam perancangan design User Interface menggunakan Figma, sebuah *tools* desain kolaboratif yang memungkinkan *developer* untuk membuat prototipe UI yang intuitif sebelum diimplementasikan ke dalam kode. UI adalah ilmu tentang tata letak grafis suatu web atau aplikasi. Cakupan UI adalah tombol yang akan diklik oleh pengguna, teks, gambar, *text entry fields*, dan semua item yang berinteraksi dengan pengguna [2].

Aplikasi monitoring makanan kucing peliharaan ini akan terintegrasi dengan perangkat IoT. Aplikasi akan menampilkan informasi yang diintegrasikan dengan perangkat seperti informasi mengenai isi penampungan, informasi mengenai jumlah porsi yang dikeluarkan, dan hasil monitoring dengan kamera. Aplikasi juga digunakan untuk mengatur penjadwalan makanan, mengeluarkan makanan secara *realtime*, dan menampilkan informasi tentang riwayat pengeluaran makanan. Aplikasi ini dinamakan "IFT *PET FEEDER*". Melalui penggunaan teknologi canggih dan design yang *user-friendly*, aplikasi ini diharapkan menjadi solusi inovatif yang membantu pemilik kucing dalam memantau pemberian makanan kucing peliharaan mereka.

II. KAJIAN TEORI

Pada pengembangan Aplikasi monitoring menggunakan beberapa *tools* yaitu Figma, React Native, Java Script, Android Studio, dan Visual Studio Code. Masing – masing memiliki keunggulannya tersendiri untuk mendukung proses pengembangan secara keseluruhan.

A. Figma

Figma adalah aplikasi desain berbasis cloud dan alat *prototyping* untuk proyek digital. Figma dibuat untuk dapat membantu para penggunanya agar bisa berkolaborasi dalam

proyek dan bekerja dalam bentuk tim sekaligus dimana saja [3]. Figma dapat digunakan untuk membuat aplikasi, web, serta berbagai komponen *user interface* yang dapat digabungkan ke dalam project lainnya.

B. React Native

React Native adalah kerangka kerja pengembangan aplikasi seluler yang populer dibuat. React Native memungkinkan pengembangan lintas platform, yang dapat digunakan untuk menghasilkan aplikasi untuk platform iOS dan Android. *Mobile Framework* ini menggunakan JavaScript, yang merupakan bahasa pemrograman yang populer dan umum digunakan, bersifat *open source* yang memungkinkan pengembang untuk berkontribusi pada pengembangan dan mengakses sumber daya secara bebas.

C. Java Script

JavaScript adalah bahasa pemrograman ringan yang umumnya digunakan oleh *web developer* untuk menciptakan interaksi yang lebih dinamis saat membuat halaman web, aplikasi, server, atau game. *Web developer* biasanya menggunakan JavaScript bersama HTML dan CSS karena ketiganya bisa bekerja sama tanpa masalah [4].

D. Android Studio IDE

Android Studio adalah *Integrated Development Environment* (IDE) resmi untuk pengembangan aplikasi Android. IDE ini dikembangkan oleh Google dan berbasis pada IntelliJ IDEA, sebuah IDE populer untuk pengembangan aplikasi Java. Android Studio menawarkan berbagai fitur yang memudahkan dan mempercepat proses pengembangan aplikasi Android [5].

E. Visual Studio Code

Visual Studio Code (VS Code) adalah *editor source code* buatan Microsoft yang beroperasi pada komputer desktop dan kompatibel dengan sistem operasi Windows, macOS, dan Linux. VS Code cocok untuk JavaScript, TypeScript, dan Node.js. Selain itu, editor ini memiliki ekosistem extension yang luas yang dapat mendukung berbagai bahasa pemrograman dan runtime lainnya, termasuk C++, C#, Java, Python, PHP, Go, dan .NET, karena serbaguna dan memiliki *extension* yang beragam, Visual Studio Code menjadi pilihan populer di kalangan *developer* untuk beragam tugas pemrograman [6].

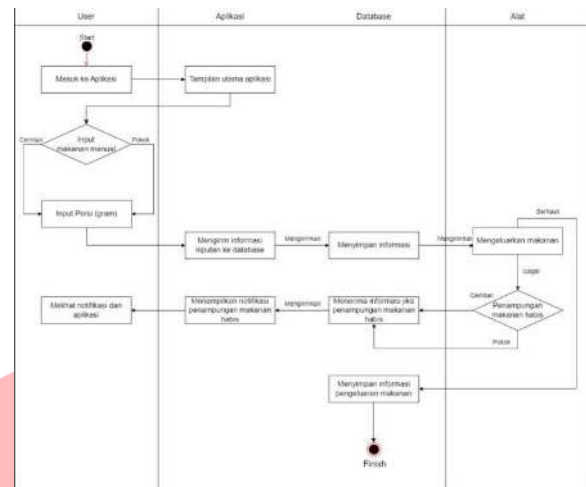
III. METODE

A. Activity Diagram

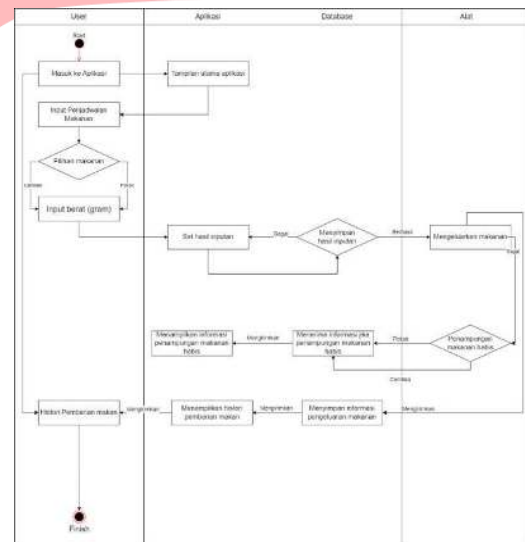
Diagram aktivitas dibawah ini menggambarkan proses pemberian makanan manual pada *pet feeder* secara rinci, mulai dari pengguna membuka aplikasi pet feeder hingga alat *feeder* mengeluarkan makanan. Diagram ini memberikan gambaran yang komprehensif mengenai bagaimana pengguna akan berinteraksi dengan alat, termasuk langkah-langkah yang perlu dilakukan, informasi yang perlu dimasukkan, dan hasil yang diharapkan.

Selain proses pemberian makanan manual, diagram aktivitas ini juga menggambarkan proses penjadwalan makanan otomatis dan monitoring hewan peliharaan. Proses ini dimulai dengan pengguna memasukkan informasi tentang jadwal pemberian makanan yang diinginkan. Informasi ini kemudian disimpan di database. Pada saat waktu yang telah ditentukan, alat *feeder* akan mengambil informasi tentang jadwal pemberian makanan dari database. Alat *feeder* kemudian akan mengeluarkan makanan sesuai dengan jadwal yang telah ditentukan. Proses penjadwalan makanan otomatis

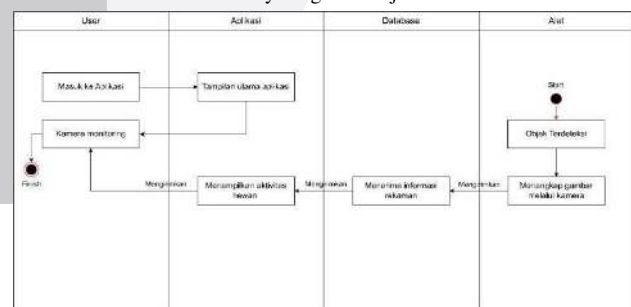
ini dapat membantu pengguna untuk menghemat waktu dan tenaga. Dengan adanya monitoring menggunakan kamera pengguna dapat memastikan kesehatan hewan peliharaan mereka dengan melihat hewan peliharaan mereka mendapatkan asupan makanan sesuai jadwal.



GAMBAR 1.
Activity Diagram Feeder Manual



GAMBAR 2.
Activity Diagram Penjadwalan Feeder



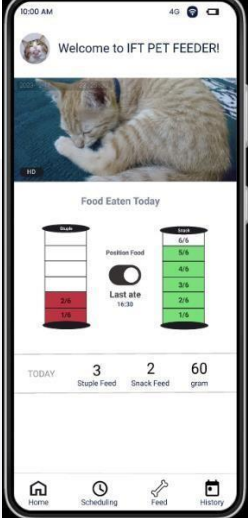
GAMBAR 3.
Activity Diagram Monitoring Feeder

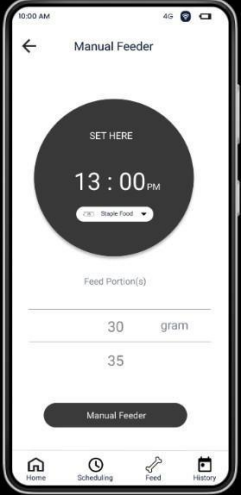
B. Design Figma Aplikasi

Aplikasi IFT dibawah ini dirancang sebagai solusi inovatif dan efektif yang bertujuan mengatasi permasalahan dalam manajemen pemberian makan hewan peliharaan, sebagaimana telah diidentifikasi dalam batasan dan

spesifikasi yang telah ditentukan. Desain aplikasi IFT melibatkan tahapan analisis solusi terpilih dan aktivitas sistem untuk memastikan pemenuhan kebutuhan pengguna dan optimalisasi fungsionalitas. Berikut ini adalah rinciannya:

TABEL. 1
Aplikasi Intelligent Food Timer

Aplikasi Intelligent Food Timer	
Splash Screen: <i>Splash screen</i> menampilkan logo dan nama aplikasi yang muncul selama 3 detik sebelum pengguna masuk ke Tab <i>Home</i> aplikasi. Logo "IFT" merupakan singkatan dari "Intelligent Food Timer" yang melambangkan bahwa aplikasi ini dapat membantu pemilik hewan peliharaan untuk mengatur jadwal pemberian makan secara otomatis.	
Tab Home: Tab <i>Home</i> menampilkan informasi sebagai berikut: - Kamera yang digunakan untuk memonitoring aktivitas kucing. - Persediaan makanan di penyimpanan: Informasi ini dapat membantu pengguna untuk memastikan jika makanan cukup untuk dimakan atau perlu diisi ulang. - Terakhir kucing makan: Informasi ini dapat membantu pengguna untuk melihat kebiasaan makan hewan peliharaan mereka. - Jumlah makanan pokok dan camilan yang sudah dikeluarkan: Informasi ini dapat membantu pengguna untuk melihat jumlah makanan yang dikonsumsi oleh hewan peliharaan mereka. - Tombol <i>Home</i>, <i>Scheduling</i>, <i>Feed</i>, <i>History</i>: Tombol-tombol ini memungkinkan pengguna untuk mengakses fitur-fitur lain dari aplikasi.	

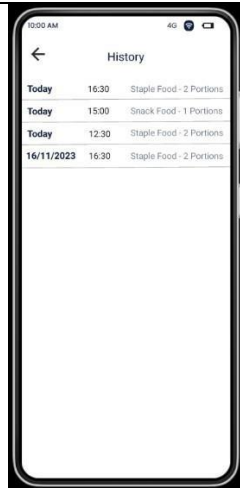
Tab Feed: Tab <i>Manual Feeder</i> menampilkan informasi sebagai berikut: - Waktu: tampilan waktu makan yang akan dikeluarkan. - Makanan: tampilan pemilihan jenis makanan yang akan dikeluarkan (<i>Staple/Snack</i>). - Porsi: berapa banyak porsi makanan yang akan dikeluarkan (Gram). - Tombol <i>Manual Feeder</i>: tombol yang berfungsi untuk mengeluarkan makanan secara manual.	
Tab Scheduling: Tab <i>Scheduling</i> menampilkan informasi jadwal pemberian makan kucing sebagai berikut: - Waktu pemberian makan: tampilan waktu saat memberi makan kucing. - Pemilihan makanan: jenis makanan yang akan diberikan (<i>Staple/Snack</i>). - Jumlah porsi: berapa banyak porsi makanan yang akan diberikan (Gram). - Hari: menunjukkan hari apa saja makanan akan diberikan. Tab <i>Scheduling</i> ini berfungsi untuk mengatur jadwal pemberian makan hewan peliharaan. Dengan jadwal pemberian makan yang teratur, hal ini dapat membantu menjaga kesehatan hewan peliharaan.	

Tab History:

Tab History menampilkan informasi riwayat pemberian makanan sebagai berikut:

- Waktu: menunjukkan kapan makanan diberikan.
- Makanan: jenis makanan yang diberikan.
- Porsi: berapa banyak porsi makanan yang diberikan.
- Status: menunjukkan apakah pemberian makan berhasil atau tidak.

Tab History ini berguna untuk melacak kebiasaan makan hewan peliharaan dan memastikan bahwa mereka makan dengan teratur. Informasi ini juga dapat berguna jika ingin mengetahui kapan dan berapa banyak makanan yang diberikan kepada hewan peliharaan saat pemilik hewan tidak ada di rumah.



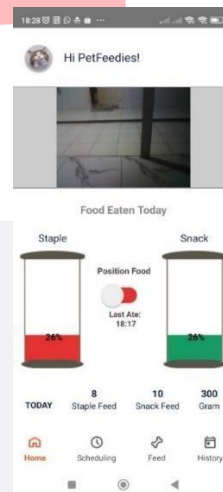
Time	Food	Portions
Today 16:30	Staple Food	2 Portions
Today 15:00	Snack Food	1 Portions
Today 12:30	Staple Food	2 Portions
16/11/2023 16:30	Staple Food	2 Portions



GAMBAR 5.
Tampilan Loginscreen

3. Homescreen

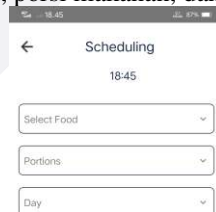
Pada halaman *homescreen* terdapat beberapa fitur seperti kamera untuk monitoring, informasi mengenai isi penampungan makanan, informasi position food, informasi last ate, informasi keseluruhan total makanan yang dikeluarkan.



GAMBAR 6.
Tampilan Homescreen

4. Scheduling Screen

Pada halaman *scheduling*, pengguna dapat mengatur penjadwalan makanan. Pengguna terlebih dahulu memilih jam, jenis makanan, porsi makanan, dan hari.



GAMBAR 7.
Tampilan Scheduling screen

IV. HASIL DAN PEMBAHASAN

A. Implementasi Design Aplikasi

1. Splash Screen

Splash screen akan menampilkan logo dan nama aplikasi yang muncul, sebelum masuk ke Tab Home aplikasi. Logo "IFT" merupakan singkatan dari "Intelegent Food Timer."



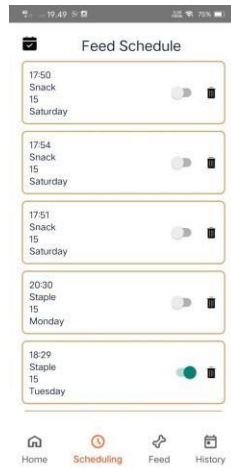
GAMBAR 4.
Tampilan Splashscreen

2. Login Screen

Pada halaman login terdapat kolom login dimana pengguna harus memasukkan kode token yang sudah diberikan, jika pengguna lupa dengan kode token tersebut, pengguna dapat menekan tulisan "Forgot token? Contact call center".

5. Feed Schedule

Pengguna dapat melihat riwayat penjadwalan makanan yang telah diatur pada tab *scheduling*. Pada *feed schedule screen* pengguna dapat melihat informasi mengenai jam penjadwalan, pilihan jenis makanan yang diatur, porsi makanan yang dikeluarkan, dan hari yang telah ditentukan. Pengguna juga dapat mengaktifkan ataupun menonaktifkan *schedule* dan juga menghapus *schedule*.

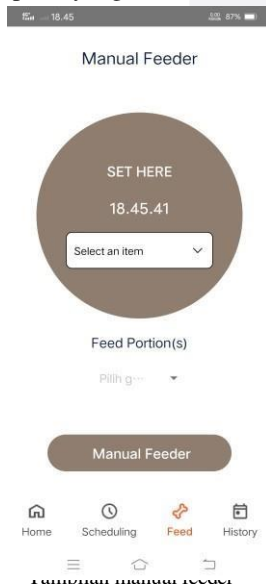


GAMBAR 8.

Tampilan Scheduling screen

6. Manual Feeder

Pada tampilan *manual feeder*, pengguna dapat memberi makan kucing peliharaan secara *realtime* tanpa harus mengatur penjadwalan melalui aplikasi. Pengaturan makanan pada *manual feeder* sama halnya dengan pengaturan makanan pada *scheduling*, pengguna harus memilih jenis makanan dan porsi yang akan dikeluarkan terlebih dahulu.



1. History screen

Pada halaman history pengguna dapat melihat riwayat pemberian makanan. Pengguna dapat melihat informasi tentang tanggal makanan dikeluarkan, jam, jenis makanan, dan porsi.

Riwayat pemberian makan ini juga dapat dihapus oleh pengguna.



GAMBAR 10.

Tampilan History screen

B. Pengujian UAT Aplikasi

Pengujian UAT (*User Acceptance Test*) dilakukan dengan mengajukan pertanyaan kepada pengguna publik dengan total responden sebanyak 67 responden, dimana 11 responden secara langsung mencoba aplikasi, dan 56 responden mengikuti tes secara virtual dengan menonton video yang disajikan.

Hasil *User Acceptance Testing* dinilai dengan 5 kategori, yaitu SS (sangat setuju), S (setuju), RR (ragu-ragu), TS (tidak setuju), dan STS (sangat tidak setuju). 5 kategori tersebut memiliki bobot nilainya masing-masing.

TABEL. 2
Bobot Hasil Nilai UAT

Jawaban	Bobot
SS = Sangat Setuju	5
S = Setuju	4
RR = Ragu-ragu	3
TS = Tidak Setuju	2
STS = Sangat Tidak Setuju	1

$$\text{Nilai Rata Rata} = \frac{\text{Jumlah Bobot Nilai Responden}}{\text{Total Responden}}$$

Setelah mendapatkan nilai rata-rata maka dibutuhkan perhitungan persentase dari masing-masing pertanyaan untuk mendapatkan hasil kualitas sistem untuk layak untuk digunakan bagi pengguna.

$$\text{Presentase} = \frac{\text{Nilai rata - rata}}{\text{Bobot Maximum}} \times 100\%$$

TABEL. 3
Bobot Nilai Hasil Persentase UAT

Presentase	Keterangan
0% - 20%	Sangat Tidak Setuju
21% - 40%	Tidak Setuju
41% - 60%	Ragu-Ragu
61% - 80%	Setuju
81% - 100%	Sangat Setuju

TABEL. 4
Hasil Pengujian UAT Secara Langsung (11 Responden)

PENGUJIAN USER ACCEPTANCE TEST APLIKASI DAN ALAT IFT PETFEEDER									
NO	PERTANYAAN	ST S	T S	RR	S	S S	Total Perhitungan	Nilai rata-rata	
Aplikasi									
1	Apakah menurut Anda tata letak dan desain produk menarik secara visual?	1	0	2	2	5	40	3.636363636	
2	Apakah informasi yang ada pada aplikasi mudah dipahami?	1	2	0	4	4	41	3.727272727	
3	Apakah aplikasi ini menyediakan semua fitur yang Anda butuhkan?	1	0	0	3	7	48	4.363636364	
4	Apakah Anda merasa bahwa aplikasi ini berfungsi dengan lancar tanpa kendala teknis?	0	0	0	4	7	51	4.636363636	
5	Apakah anda setuju aplikasi ini mudah digunakan?	0	1	0	4	6	48	4.363636364	

Analisis Pengujian : Pada tabel diatas merupakan hasil UAT yang sudah dikalikan dengan bobot pada tabel. Data pada tabel ini kemudian digunakan untuk menghitung nilai presentase. Berikut ini merupakan rumus nilai rata-rata

TABEL. 5
Hasil Pengujian UAT Secara Langsung (11 Responden)

PENGUJIAN USER ACCEPTANCE TEST APLIKASI DAN ALAT IFT PETFEEDER				
NO	PERTANYAAN	Presentase	Keterangan	
Aplikasi				
1	Apakah menurut Anda tata letak dan desain produk menarik secara visual?	72.72727273	Setuju	Presentase Rata-Rata : 82.4 Kesimpulan : Sangat Setuju
2	Apakah informasi yang ada pada aplikasi mudah dipahami?	74.54545455	Setuju	
3	Apakah aplikasi ini menyediakan semua fitur yang Anda butuhkan?	87.27272727	Sangat Setuju	
4	Apakah Anda merasa bahwa aplikasi ini berfungsi dengan lancar tanpa kendala teknis?	92.72727273	Sangat Setuju	
5	Apakah anda setuju aplikasi ini mudah digunakan?	87.27272727	Sangat Setuju	

TABEL. 6
Hasil Pengujian UAT Secara Virtual (57 Responden)

PENGUJIAN USER ACCEPTANCE TEST APLIKASI DAN ALAT IFT PETFEEDER									
NO	PERTANYAAN	S T S	T S	R R	S	SS	Total Perhitungan	Nilai rata-rata	
Aplikasi									
1	Apakah menurut Anda tata letak dan desain produk menarik secara visual?	0	3	1	29	23	240	4.285714286	
2	Apakah informasi yang ada pada aplikasi mudah dipahami?	0	1	1	28	27	252	4.5	
3	Apakah aplikasi ini menyediakan semua fitur yang Anda butuhkan?	0	0	7	31	19	240	4.285714286	
4	Apakah anda setuju aplikasi ini mudah digunakan?	0	0	2	29	26	252	4.5	

TABEL. 7
Hasil Pengujian UAT Secara Virtual (56 Responden)

PENGUJIAN USER ACCEPTANCE TEST APLIKASI DAN ALAT IFT PETFEEDER				
NO	PERTANYAAN	Presentase	Keterangan	
Aplikasi				
1	Apakah menurut Anda tata letak dan desain produk menarik secara visual?	85.71428571	Sangat Setuju	Presentase Rata-Rata : 87.5 Kesimpulan : Sangat Setuju
2	Apakah informasi yang ada pada aplikasi mudah dipahami?	90	Sangat Setuju	
3	Apakah aplikasi ini menyediakan semua fitur yang Anda butuhkan?	85.71428571	Sangat Setuju	
4	Apakah anda setuju aplikasi ini mudah digunakan?	90	Sangat Setuju	

Dari hasil UAT yang dilakukan baik secara langsung maupun virtual aplikasi dan alat IFT *PetFeeder* untuk ini secara umum layak digunakan oleh pengguna dengan rata-rata presentase di atas 80%. Pengujian langsung menunjukkan hasil 82,4%, sementara pengujian virtual menunjukkan hasil yang lebih tinggi dengan 87,5%. Hal ini menunjukkan bahwa aplikasi ini tidak hanya memenuhi standar kelayakan, tetapi juga ekspektasi pengguna dalam berbagai kondisi. Keberhasilan ini mencerminkan efektivitas dan kemudahan penggunaan aplikasi IFT *PetFeeder*.

V. KESIMPULAN

Pengembangan aplikasi IFT *PetFeeder* berhasil mencapai tujuan utama yaitu menyediakan solusi inovatif dan efisien bagi pemilik kucing peliharaan untuk memantau dan mengelola pemberian makanan kucing peliharaan mereka dari jarak jauh. Penggunaan React Native sebagai framework pengembangan memungkinkan pembuatan aplikasi lintas platform dengan satu basis kode, yang meningkatkan efisiensi pengembangan. Hasil UAT yang tinggi menunjukkan bahwa aplikasi ini memenuhi standar kelayakan dan ekspektasi pengguna dalam berbagai kondisi. Keberhasilan ini mencerminkan efektivitas dan kemudahan penggunaan aplikasi IFT *PetFeeder*, serta potensinya untuk diterima secara luas oleh pengguna. Dengan fitur-fitur yang canggih dan desain yang *user-friendly*, IFT *PetFeeder* diharapkan dapat menjadi solusi bagi pemilik kucing dalam mengelola pemberian makanan secara efisien dan efektif.

REFERENSI

[1] V. Kaushik, K. Gupta, and D. Gupta, “React Native Application Development,” papers.ssrn.com, 2018. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3330011

[2] M. A. Muhyidin, M. A. Sulhan, and A. Sevtiana, “Perancangan UI/UX Aplikasi My CIC Layanan Informasi Akademik Mahasiswa Menggunakan Aplikasi Figma,” Jurnal Digit : *Digital of Information Technology*, vol. 10, no. 2, pp. 208–219, Dec. 2020, Available: <https://www.jurnaldigit.org/index.php/DIGIT/article/view/171/131>

[3] Rully Pramudita, Rita Wahyuni Arifin, Ari Nurul Alfian, Nadya Safitri, and Shilka Dina Anwariya, “Penggunaan Aplikasi Figma Dalam Membangun UI/UX yang Interaktif Pada Program Study Teknik Informatika STMIK TASIKMALAYA,” JURNAL BUANA PENGABDIAN, vol. 3, no. 1, pp. 149–154, Jul. 2021,

doi:<https://doi.org/10.36805/jurnalbuanapengabdian.v3i1.1542>.

[4] “Apa Itu JavaScript? Pemahaman Dasar untuk Pemula,” Hostinger Tutorial, Sep. 11, 2018. <https://www.hostinger.co.id/tutorial/apa-itu-javascript>

[5] “Android Studio: IDE Resmi untuk Pengembangan Aplikasi Android,” Institut Teknologi dan Bisnis PalComTech Palembang. <https://palcomtech.ac.id/android-studio-ide-resmi-untuk-pengembangan-aplikasi-android/>

[6]]“Apa itu Visual Studio Code? Pengertian dan contoh 2023 | RevoU,” revou.co. <https://revou.co/kosakata/visual-studio-code>

