

Pengembangan Aplikasi *Mobile* HabiTroops

Kevin Pramudya^{#1}, Doro Edi^{*2}

*#Program Studi SI Teknik Informatika, Universitas Kristen Maranatha
Jl. Prof. Drg. Surya Sumantri No.65, Sukawarna, Bandung, Indonesia*

¹1972053@maranatha.ac.id

²doro.edi@it.maranatha.edu

Abstract — In this research, the development of a mobile application named HabiTroops was conducted with the aim of assisting in achieving regular habits. This application was developed using the Kotlin programming language and designed to easily and systematically manage habits. HabiTroops provides a convenient and practical solution for users to digitally manage habits. The main objective of this research is to develop this application in order to provide effective support in improving better habits. It is expected that the HabiTroops application can serve as a useful tool for users to achieve desired habit goals more easily and efficiently.

Keywords— Development, HabiTroops, Improving Habits, Kotlin Programming Language, Mobile application.

I. PENDAHULUAN

Banyaknya aktivitas berdampak besar pada produktivitas yang dijalani. Semakin banyak aktivitas maka semakin banyak juga waktu yang harus diatur untuk melakukan aktivitas secara produktif. Untuk melakukan aktivitas yang teratur perlu membentuk kebiasaan. Kebiasaan adalah aktivitas yang dilakukan secara berulang, teratur dan terlatih sehingga membentuk satu pola yang bergerak secara otomatis. Kebiasaan merupakan bagian yang sangat penting dalam kehidupan. Kebiasaan yang baik dapat membantu dalam mencapai tujuan dan menjalani hidup dengan lebih baik, sementara kebiasaan yang buruk dapat menghambat kinerja dan kesejahteraan. Namun, membentuk dan mengubah kebiasaan tidak selalu mudah, terutama jika tidak memiliki cara untuk memantau atau mengelola kebiasaan secara efektif. Kebiasaan juga perlu dibentuk menggunakan dorongan motivasi dan tujuan secara tertulis.

Dengan adanya teknologi, tujuan untuk mencapai kebiasaan yang teratur dapat dibuat, dilihat, dan dipantau menggunakan aplikasi. Untuk memantau kebiasaan tersebut maka dalam penelitian ini akan dikembangkan sebuah aplikasi *mobile* dengan operating system android yang akan diberi nama HabiTroops. HabiTroops adalah aplikasi *mobile* yang dikembangkan menggunakan bahasa pemrograman kotlin untuk membantu para pengguna mengelola kebiasaan dengan lebih mudah dan terarah.

Aplikasi HabiTroops memberi solusi yang mudah dan nyaman untuk mengelola kebiasaan pengguna secara digital. Oleh karena itu, dalam penelitian ini bertujuan untuk mengembangkan aplikasi ini yang diharapkan dapat membantu pengguna dalam meningkatkan kebiasaan yang lebih baik dengan lebih mudah.

II. KAJIAN TEORI

A. Definisi Kebiasaan

Kebiasaan adalah perilaku yang ditekankan dalam situasi atau keadaan tertentu. Kebiasaan memiliki kekuatan untuk mengendalikan perilaku seseorang. Kebiasaan juga bisa kuat atau lemah tergantung motivasi yang dikaitkan dengan maksud dan tujuan dari kegiatan yang menjadi kebiasaan [1]

B. Jenis Kebiasaan Manusia

Ada dua jenis kebiasaan yang berdampak positif dan sebaliknya berdampak negatif. Kebiasaan negatif lebih dikenal dengan sebutan kebiasaan buruk, yang dicap buruk karena menyebabkan seseorang mengalami kemunduran baik secara fisik maupun mental. Kebiasaan baik dapat membantu seseorang mencapai tujuannya dan menjalani kehidupan yang lebih baik [2].

C. Bahasa Pemrograman

Bahasa pemrograman adalah media komunikasi antara manusia dan komputer. Saat ini pada perkembangan teknologi, bahasa pemrograman telah menjadi sesuatu yang dianggap terbaik dan dijadikan sebagai unggulan tersendiri [3].

Bahasa pemrograman juga merupakan perintah atau instruksi yang dipahami komputer untuk melakukan tugas tertentu, salah satunya adalah perintah untuk menyimpan dan memanipulasi data. Setiap bahasa memiliki kelebihan dan kekurangan yang berbeda [4].

D. Bahasa Pemrograman Kotlin

Kotlin merupakan bahasa pemrograman yang termasuk bahasa pemrograman tingkat tinggi (*high-level programming language*) moderen berbasis *Java Virtual Machine* (JVM) [5].

Kotlin dapat digunakan untuk membangun aplikasi untuk *platform* seperti Android, iOS, *web*, dan *desktop*. Bahasa Kotlin adalah sebuah evolusi bahasa Java yang dulu populer. Bahasa Kotlin memiliki fitur linguistik lebih modern dari bahasa Java [6].

E. Perangkat Mobile

Seiring dengan tingginya tingkat mobilitas, jumlah *mobile terminal device* atau perangkat *mobile* juga meningkat dalam beberapa tahun terakhir. Salah satu perangkat *mobile* tercepat adalah *smartphone* berbasis Android yang banyak dimiliki [7].

Smartphone digunakan untuk media pembelajaran, juga sebagai media pertukaran informasi. Salah satu sistem operasi yang dapat digunakan pada *smartphone* adalah sistem operasi Android. Android adalah sistem operasi seluler berbasis Linux dan *open source* [8].

F. Aplikasi Mobile

Aplikasi *mobile* adalah aplikasi yang memungkinkan perpindahan informasi melalui perangkat. Dengan penggunaan aplikasi *mobile*, pengguna dapat dengan mudah melakukan berbagai aktivitas seperti mendapatkan informasi, berjualan, belajar, kerja kantor secara daring, *browsing*, dan aktivitas lainnya [9].

Aplikasi *mobile* dapat dipasang pada perangkat seluler seperti *smartphone* atau *tablet* dan dapat menjalankan berbagai fungsi atau memberikan layanan yang bermanfaat. Aplikasi seluler dapat mencakup berbagai kategori produktivitas, media sosial, hiburan, belanja, dan lainnya. Aplikasi *mobile* biasanya tersedia di toko aplikasi seperti *Google Play Store* atau *Apple App Store* tempat pengguna dapat mencari, mengunduh, dan membeli aplikasi yang diinginkan [10].

G. Android Studio

Android Studio adalah *Integrated Development Environment* (IDE) pemrograman Android resmi Google yang dikembangkan oleh IntelliJ. Android Studio juga memungkinkan pengembang untuk menggunakan bahasa pemrograman Kotlin atau Java untuk membuat aplikasi Android [11].

H. Firebase

Firebase adalah API yang disediakan oleh Google sebagai penyimpanan data dan aplikasi penargetan Android, iOS atau *Web*. *Database real-time* adalah salah satunya fasilitas yang menyimpan dan mengambil informasi dari *database* Informasi tentangnya dengan sangat cepat, tetapi Firebase tidak hanya *database real-time*, itu jauh lebih banyak tingkatannya. Firebase ini memiliki banyak fitur seperti otentikasi, basis data, Penyimpanan, *hosting*, notifikasi, dan lainnya [12].

I. Room Database

Room Database adalah *library* yang disediakan oleh Android Jetpack untuk menyederhanakan dan mempermudah penggunaan *database SQLite* dalam aplikasi Android. Room berfungsi sebagai penyimpanan objek akses data [17].

J. Gamification

Gamification adalah cara yang menjanjikan untuk meningkatkan keterlibatan pengguna. Akibatnya, pengembang aplikasi seluler semakin banyak mengintegrasikan gamifikasi ke dalam aplikasi untuk meningkatkan pengalaman pengguna [14].

Tren modern mengarah pada pengenalan berbagai teknologi inovatif dalam proses pendidikan, termasuk permainan [15].

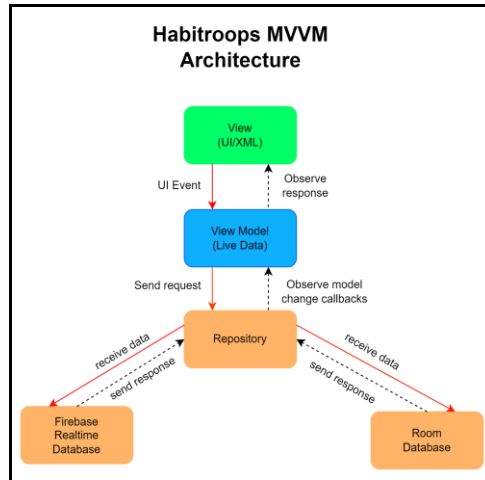
K. MVVM

Sebuah pola desain atau paradigma arsitektur baru yang disebut *Model-View-View-Model* (MVVM) berfungsi sebagai penghalang antara logika antarmuka pengguna dan bisnis, dan digunakan untuk mengatur konsentrasi pengembangan aplikasi Android. Google selalu menjadi pendukung MVVM sejak diperkenalkan pada konferensi Google I/O 2018.

MVVM sering digunakan dalam pengembangan aplikasi *desktop* dan *mobile*, terutama ketika menggunakan kerangka pengembangan *Windows Presentation Foundation* (WPF) dan *Silverlight*. Google menyarankan penggunaannya untuk membuat aplikasi Android karena reputasinya yang fleksibel, mudah dipelihara, dan mudah diuji [16].

III. ANALISIS DAN RANCANGAN

A. Diagram Arsitektur MVVM



Gambar 1. Diagram MVVM HabiTroops

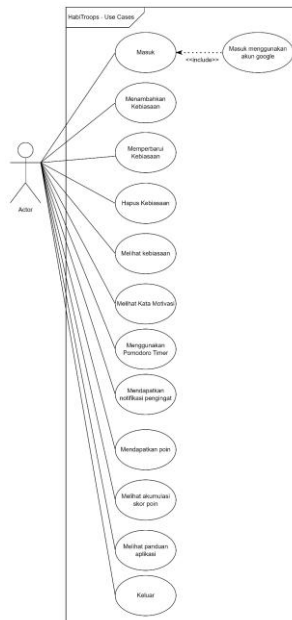
Pada Gambar 1 *ViewModel* layer ditempatkan di antara *View* dan *Repository*. *layer* ini mengekspos data ke *View* melalui objek *LiveData* yang dapat diobservasi, dan juga menerima peristiwa dan data dari *View*. Tugas *ViewModel* adalah berinteraksi dengan *Repository* untuk mengambil atau memperbarui data.

Pada *Repository* layer bertanggung jawab untuk mengabstraksi sumber data yang digunakan oleh aplikasi. Dalam hal ini, lapisan ini berinteraksi dengan *Firebase Realtime Database* dan *Room Database* untuk mengambil atau memperbarui data. Kemudian lapisan *Repository* mengembalikan data ke *ViewModel*.

Firebase Realtime Database adalah salah satu sumber data yang digunakan oleh *Repository*. *Database* ini menyimpan informasi pengguna, termasuk nama, *avatar*, dan skor. Ketika pengguna mengklik tombol konfirmasi, skor diperbarui di *database*.

Room Database adalah sumber data lain yang digunakan oleh *Repository*. *Database* ini menyimpan objek *Habit* yang dibuat oleh pengguna. *Room Database* menyediakan *cache offline* untuk data aplikasi. Ini digunakan untuk mempertahankan entitas *Habit*.

B. Use Case Diagram



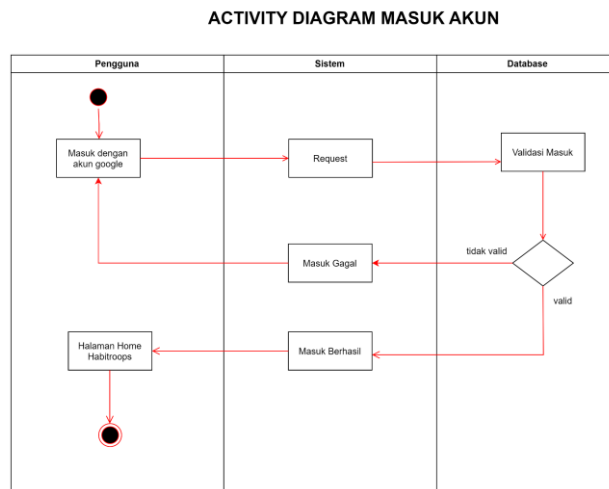
Gambar 2. Use Case Diagram HabiTroops

Pada gambar 2 terdapat fitur yang digunakan pengguna dalam aplikasi HabiTroops. Actor dalam Use Case Diagram ini adalah Pengguna, yang dapat melakukan beberapa tindakan seperti masuk menggunakan akun Google, menambahkan kebiasaan, memperbarui kebiasaan, menghapus kebiasaan, melihat kebiasaan, melihat kata motivasi, menggunakan Pomodoro Timer, mendapatkan notifikasi pengingat, mendapatkan poin, melihat akumulasi skor poin, dan keluar dari aplikasi. Setiap fitur memiliki use case-nya sendiri, seperti "Menambahkan Kebiasaan", "Melihat Kata Motivasi", dan sebagainya. Dalam beberapa kasus, seperti "Melihat Kebiasaan", pengguna dapat melihat data tanpa perlu melakukan aksi lebih lanjut. Beberapa use case seperti "Menggunakan Pomodoro Timer" dan "Mendapatkan Notifikasi Pengingat" memiliki interaksi langsung dengan sistem. Pengguna juga dapat melihat panduan aplikasi.

C. Activity Diagram

Activity diagram atau diagram aktivitas adalah diagram yang menggambarkan konsep aliran informasi/kontrol, kegiatan yang terstruktur dan terencana dengan baik dalam suatu sistem [13]. Diagram ini menunjukkan bagaimana aktivitas-aktivitas terkait saling terhubung dan bergantung satu sama lain. Fitur pada aplikasi HabiTroops tercakup pada activity diagram yang menggambarkan alur kerja dari suatu sistem secara visual

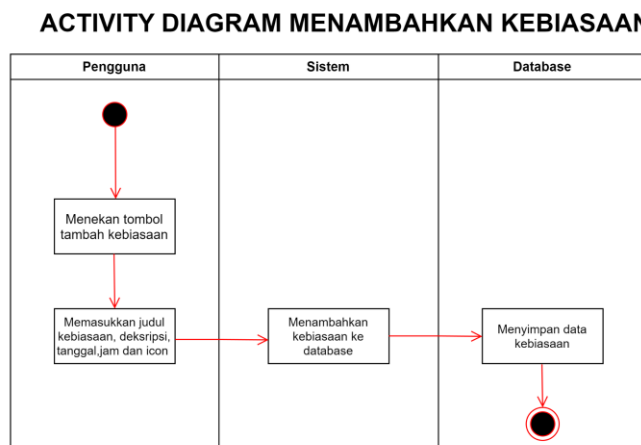
1) Activity Diagram Masuk



Gambar 3. Activity Diagram Login

Pada gambar 3 terdapat Activity dimulai dari Start. Pengguna memilih opsi Masuk pada halaman awal aplikasi. Aplikasi menampilkan layar Masuk dan pengguna memilih opsi Google Sign-In. Pengguna memasukkan kredensial Google-nya dan mengizinkan akses aplikasi. Jika autentikasi berhasil, aplikasi menampilkan halaman Home. Jika autentikasi tidak berhasil, aplikasi akan meminta untuk login kembali. Activity berakhir pada Home.

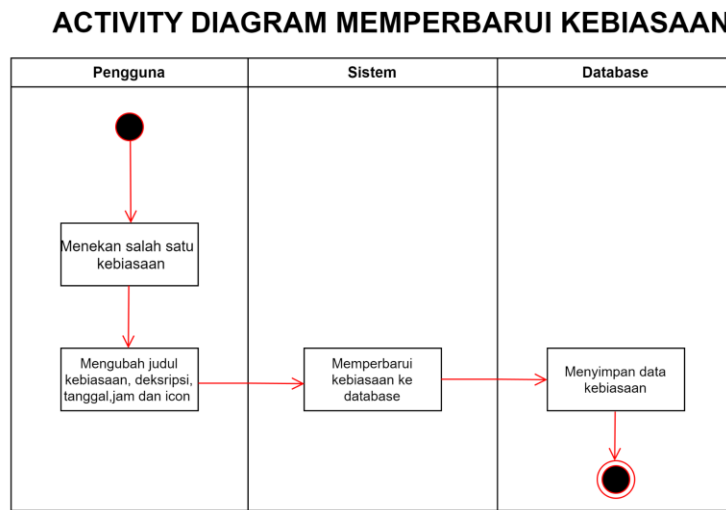
2) Activity Diagram Menambahkan Kebiasaan



Gambar 4. Activity Diagram Menambahkan Kebiasaan

Pada gambar 4 terdapat alur yang dimulai dari pengguna menekan tombol tambah kebiasaan. Pengguna memasukkan judul kebiasaan, deskripsi kebiasaan, tanggal dan jam dan memilih *icon logo* sebagai penanda. Lalu akan diarahkan ke sistem untuk menambahkan kebiasaan ke *database*, *database* akan menyimpan data kebiasaan.

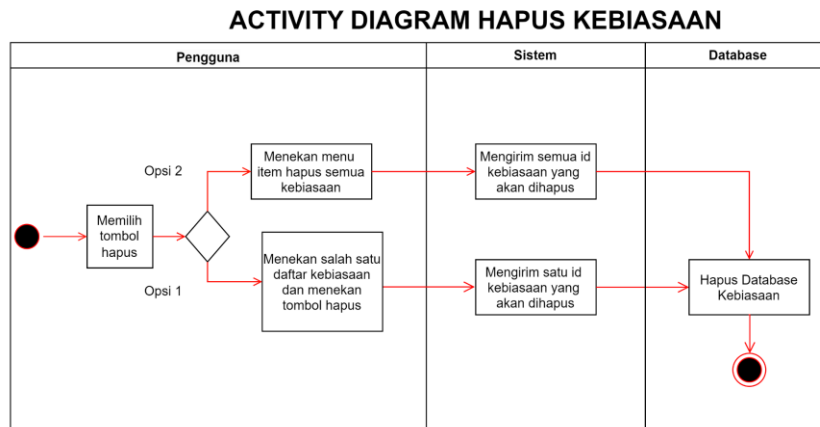
3) Activity Diagram Memperbarui Kebiasaan



Gambar 5. Activity Diagram Memperbarui Kebiasaan

Pada gambar 5 terdapat alur yang dimulai dari pengguna menekan tombol salah satu kebiasaan yang ingin di perbarui. Pengguna mengubah judul kebiasaan, deskripsi kebiasaan, tanggal, jam dan memilih *icon logo* sebagai penanda. Lalu akan diarahkan ke sistem untuk memperbarui kebiasaan ke *database*, *database* akan menyimpan data kebiasaan terbaru.

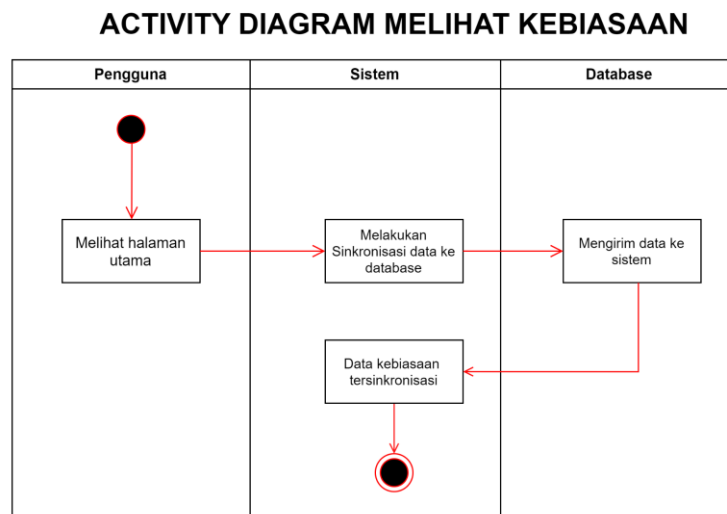
4) Activity Diagram Hapus Kebiasaan



Gambar 6. Activity Diagram Hapus Kebiasaan

Gambar 6 terdapat alur yang dimulai dari pengguna memilih tombol hapus. Jika pengguna memilih opsi 1, pengguna harus menekan salah satu daftar kebiasaan. Setelah menekan tombol daftar kebiasaan pengguna harus menekan tombol berlogo hapus untuk melakukan aksi hapus kebiasaan. saat pengguna memilih untuk menghapus kebiasaan maka pada sistem akan mengirim satu *id* kebiasaan yang akan dihapus, pada *database* akan dilakukan penghapusan data pada *id* yang dikirim oleh sistem. Jika pengguna memilih opsi 2, pengguna harus menekan *menu item* yang memiliki tombol hapus semua kebiasaan. saat pengguna memilih untuk menghapus semua kebiasaan maka pada sistem akan mengirim semua *id* kebiasaan yang akan dihapus, pada *database* akan dilakukan penghapusan data pada *id* yang dikirim oleh sistem.

5) Activity Diagram Melihat Kebiasaan

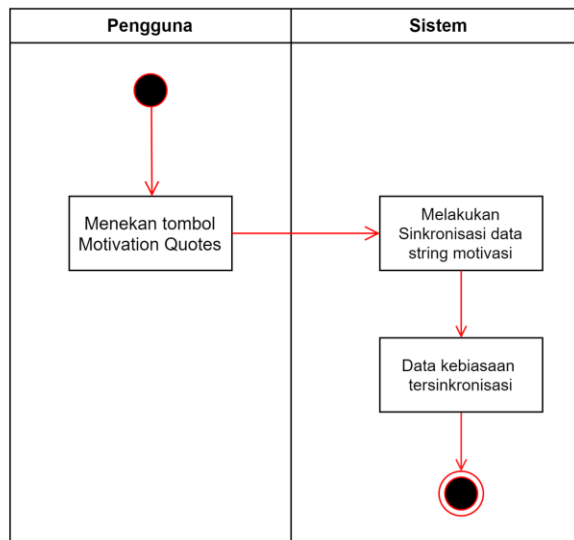


Gambar 7. Activity Diagram Melihat Kebiasaan

Pada gambar 7 dimulai dari pengguna melihat halaman utama, pada sistem akan melakukan sinkronisasi data ke *database* kumpulan kebiasaan yang sudah dibuat, lalu pada *database* mengirim data kebiasaan ke sistem. Pada sistem data tersinkronisasi dan dapat dilihat oleh pengguna.

6) Melihat Kata Motivasi

ACTIVITY DIAGRAM MELIHAT KATA MOTIVASI

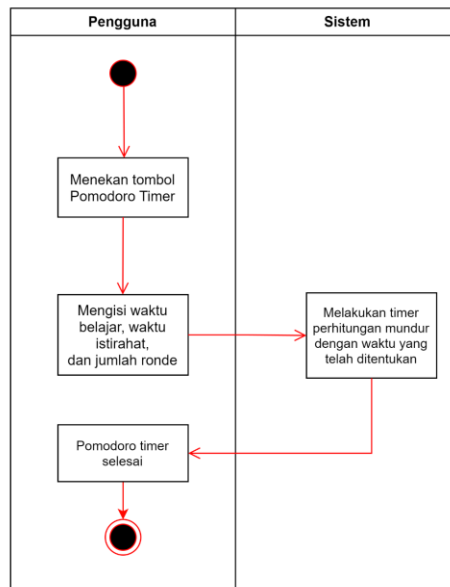


Gambar 8. Activity Diagram Melihat Kata Motivasi

Gambar 8 terdapat alur yang dimulai dari pengguna menekan tombol *Motivation Quotes*. Sistem akan melakukan sinkronisasi data kata motivasi yang bertipe *string*. Lalu data kebiasaan pada sistem tersinkronisasi dan dapat dilihat oleh pengguna.

7) Activity Diagram Menggunakan Pomodoro Timer

ACTIVITY DIAGRAM MENGGUNAKAN FITUR POMODORO TIMER

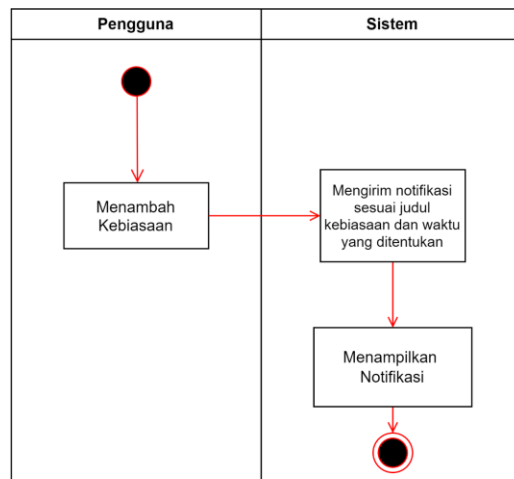


Gambar 9. Activity Diagram Menggunakan Fitur Pomodoro Timer

Gambar 9 terdapat alur yang dimulai dari pengguna menekan tombol *pomodoro timer*, pengguna harus mengisi waktu belajar, waktu istirahat dan jumlah ronde yang ditentukan. Lalu sistem akan melakukan *timer* penghitungan mundur dengan waktu yang telah ditentukan.

8) Activity Diagram Mendapatkan Notifikasi Peningat

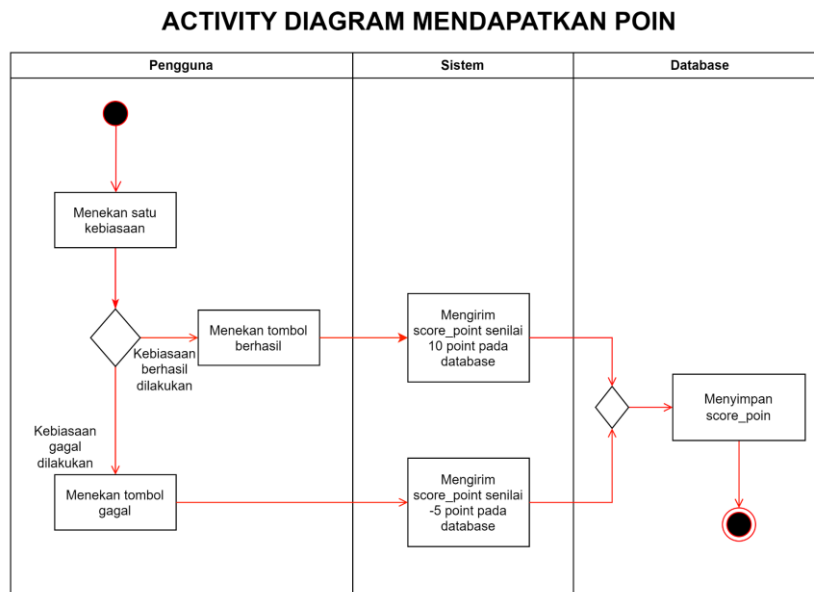
ACTIVITY DIAGRAM MENDAPATKAN NOTIFIKASI PENGINGAT



Gambar 10. Activity Diagram Mendapatkan Notifikasi Peningat

Gambar 10 terdapat alur yang dimulai saat pengguna menambah kebiasaan, sistem akan mengirim notifikasi sesuai judul kebiasaan dan waktu yang ditentukan. Lalu sistem akan menampilkan notifikasi sesuai tenggat waktu yang ditentukan.

9) Activity Diagram Mendapatkan Poin

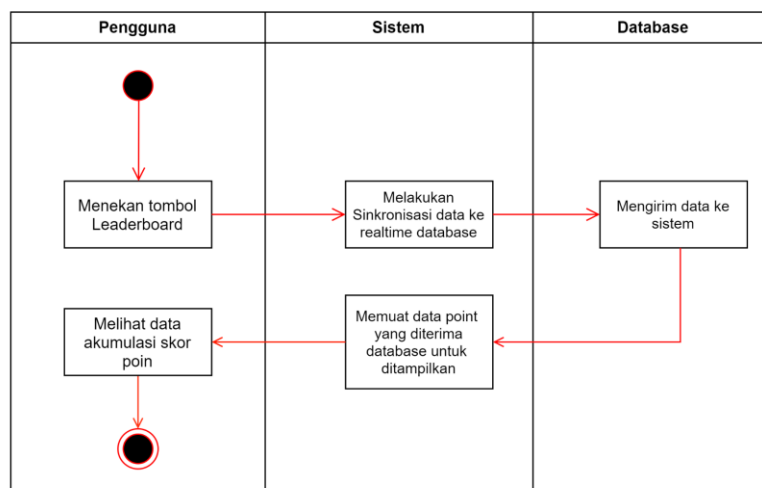


Gambar 11. Activity Diagram Mendapatkan Poin

Gambar 11 terdapat alur yang dimulai saat pengguna menekan salah satu kebiasaan. Terdapat 2 validasi yaitu jika kebiasaan berhasil dilakukan, pengguna harus menekan tombol berhasil. Sistem akan mengirim *score_point* senilai 10 poin ke *realtime database* lalu *database* akan menyimpan *score_point*. Sedangkan jika kebiasaan gagal dilakukan, pengguna harus menekan tombol gagal, sistem akan mengirim *point* senilai -5 poin pada *realtime database*, lalu *database* menyimpan *score_point*.

10) Activity Diagram Melihat Akumulasi Skor Poin

ACTIVITY DIAGRAM MELIHAT AKUMULASI SKOR POIN

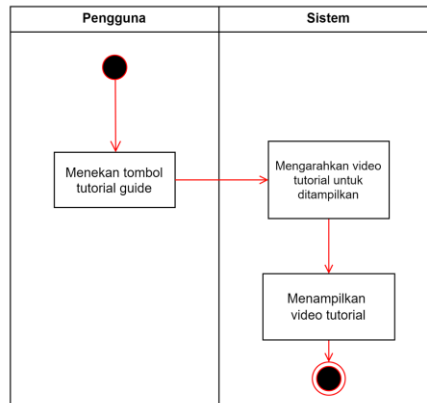


Gambar 12. Activity Diagram Melihat Akumulasi Skor Poin

Gambar 12 terdapat alur yang dimulai saat pengguna menekan tombol *leaderboard*, sistem akan melakukan sinkronisasi data ke *realtime database* lalu *database* akan mengirim data akumulasi skor poin yang berisikan *userdata* ke sistem. Pada sistem akan memuat data *point* yang diterima dari *database* untuk ditampilkan. Pengguna dapat melihat akumulasi skor *point*.

11) Activity Diagram Melihat Panduan Aplikasi

ACTIVITY DIAGRAM MELIHAT PANDUAN APLIKASI



Gambar 13. Activity Diagram Melihat Akumulasi Skor Poin

Gambar 13 terdapat alur yang dimulai saat pengguna menekan tombol *tutorial guide*, sistem akan mengarahkan *video tutorial* untuk ditampilkan lalu menampilkan *video tutorial*.

D. Firebase Realtime Database

Pengguna	
userID	(String)
first_name	(String)
last_name	(String)
last_updated_date	(Date)
picture_avatar	(String)
score_point	(Integer)

Gambar 14. Database Firebase

Pada gambar 14 terdapat *database* yang digunakan menggunakan *Firebase Realtime Database*. Pada *database* ini terdapat "userID": tipe data *String*, digunakan sebagai *primary key* untuk mengidentifikasi user.

E. Room Database

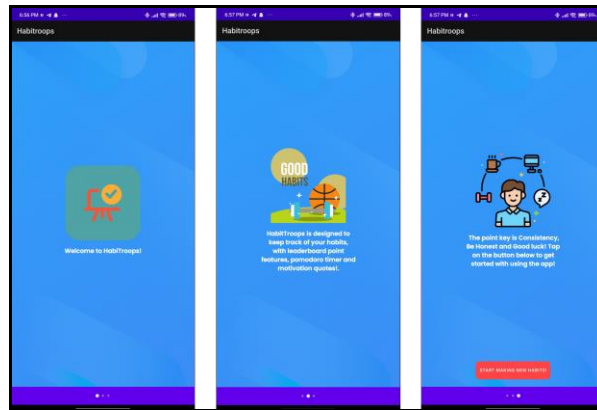
Habit	
habitId	(Integer)
habit_title	(String)
habit_description	(String)
habit_startTime	(String)
imageId	(Integer)

Gambar 15. Database Firebase

Dengan menggunakan *Room Database*, data-data ini dapat disimpan secara persisten di perangkat pengguna dan dapat diakses kapan saja dalam aplikasi. Pada gambar 15, *Room Database* menyediakan antarmuka yang mudah digunakan untuk melakukan operasi seperti penambahan, penghapusan, dan pembaruan data pada tabel *habit*

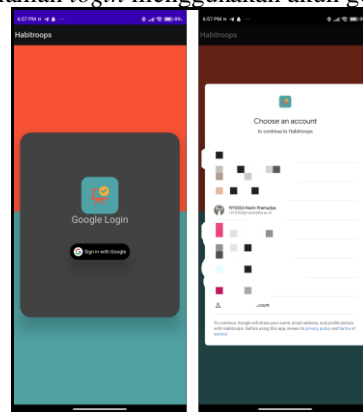
IV. IMPLEMENTASI

A. Halaman Intro dan Sign in



Gambar 16. Halaman Intro

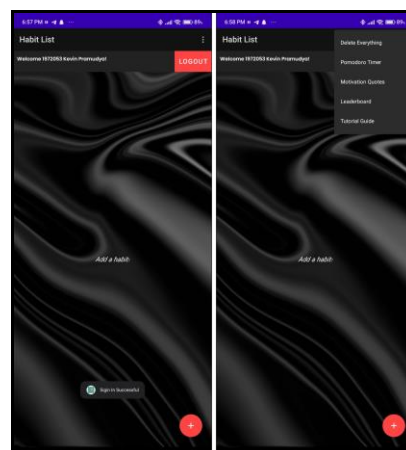
Pada gambar 16 terdapat tampilan awal saat aplikasi di buka. Tampilan *intro* awal berisikan *logo* HabiTroops juga terdapat *slider* yang dapat di geser berisi instruksi dasar penggunaan aplikasi HabiTroops. Pada saat tombol *start making new habits* ditekan akan mengarah kepada halaman *login* menggunakan akun google.



Gambar 17. Halaman Intro

Pada gambar 17 terdapat halaman *login* yang muncul saat pengguna menekan tombol *start making new habits*. Sistem meminta pengguna untuk melakukan autentikasi menggunakan akun google.

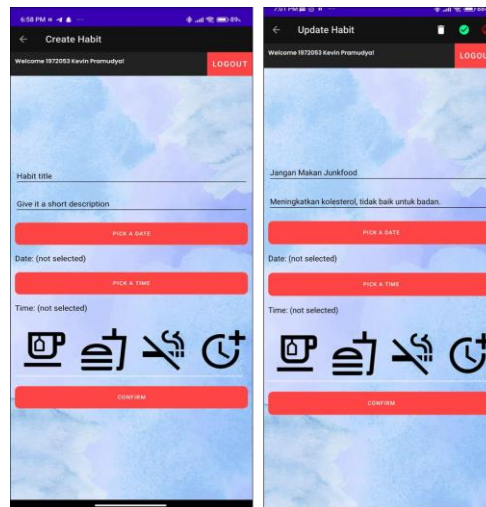
B. Halaman Utama Aplikasi



Gambar 18. Halaman Utama Aplikasi

Pada gambar 18 terdapat halaman utama aplikasi HabiTroops. Halaman ini berisi daftar kumpulan kebiasaan yang sudah dibuat. Juga terdapat tombol berbentuk lingkaran untuk menambah kebiasaan. Terdapat tombol *logout* untuk keluar dari akun google.

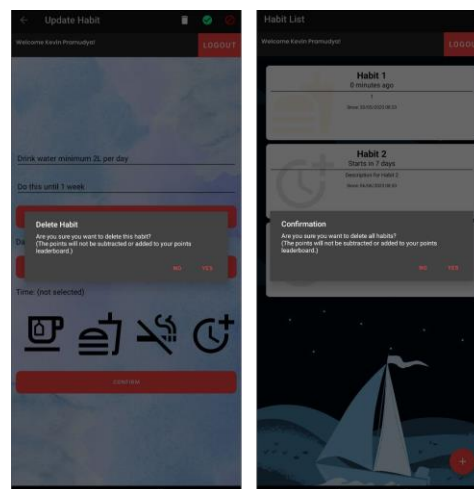
C. Halaman Tambah dan Perbarui Kebiasaan



Gambar 19. Halaman Tambah dan Perbarui Kebiasaan

Pada gambar 19 terdapat fitur untuk menambah dan menghapus kebiasaan. Isi dari *habit title* dapat berupa kalimat positif maupun negatif, tergantung pada preferensi pengguna. Pengguna mungkin lebih memilih menggunakan *habit title* yang memuat kalimat positif untuk memperkuat motivasi dan semangat.

D. Halaman Hapus Kebiasaan



Gambar 20. Halaman Hapus Kebiasaan

Pada gambar 20 terdapat 2 tombol untuk menghapus kebiasaan yaitu pada saat menekan salah satu kebiasaan juga menekan tombol *delete everything*. Terdapat teks peringatan poin tidak akan dikurang ataupun ditambah pada *leaderboard*. Pengguna dapat menggunakan fitur ini untuk menghapus data kebiasaan jika terdapat data kebiasaan yang tidak sesuai dalam kebiasaan yang dibuat pengguna.

E. Halaman Daftar Kebiasaan



Gambar 21. Halaman Daftar Kebiasaan

Pada gambar 21 menampilkan tampilan kebiasaan yang telah dibuat. Setiap kolomnya mencakup deskripsi kebiasaan, waktu dimulainya kebiasaan, serta tanggal pembuatan kebiasaan. Terdapat juga *logo* yang menunjukkan kategori kebiasaan pada setiap kolomnya.

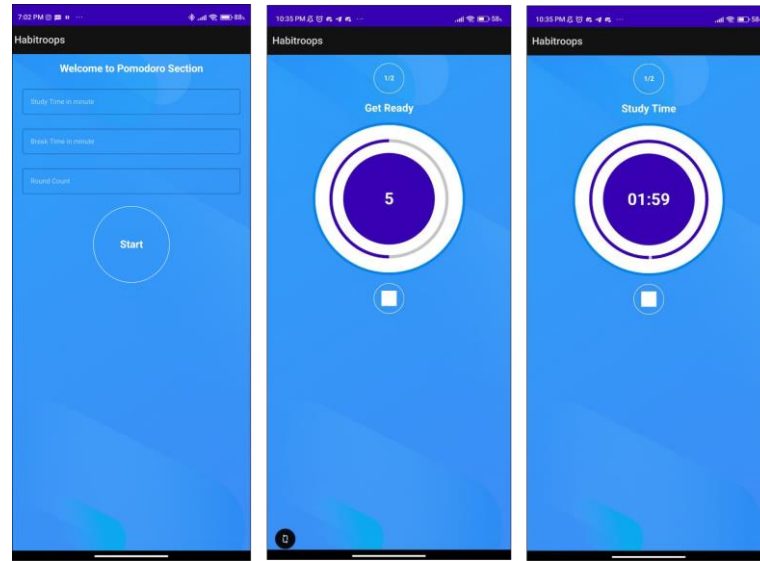
F. Halaman Motivation Quotes



Gambar 22. Halaman Motivation Quotes

Pada gambar 22 menampilkan tampilan laman motivasi yang dapat diperbarui ketika pengguna menekan tombol bulat hijau, yaitu tombol "Perbarui". Kata "Motivasi" berada di tengah-tengah. Fungsi fitur ini bertujuan untuk memberikan motivasi kepada pengguna dalam menggunakan aplikasi HabitTroops. Sumber konten motivasi yang ada dalam aplikasi ini berasal dari berbagai media *online*. Kata-kata motivasi ini kemudian dikurasi dan dipilih dengan cermat untuk disajikan kepada pengguna aplikasi. Dengan demikian, pengguna dapat memiliki koleksi konten motivasi yang beragam dan bervariasi untuk menginspirasi dan memotivasi pengguna dalam mencapai tujuan kebiasaan.

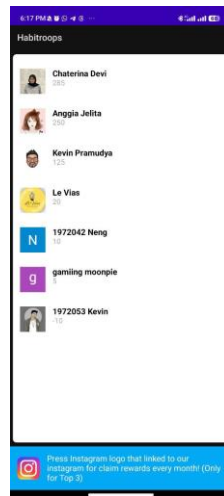
G. Halaman Pomodoro Timer



Gambar 23. Halaman Pomodoro Timer

Pada gambar 23 terdapat tampilan halaman *pomodoro timer*. Pengguna perlu mengisi kolom *Study Time in Minutes* untuk menentukan waktu belajar per menit, *Break Time in Minutes* untuk menentukan waktu jeda per menit, *Round Count* untuk menentukan jumlah pengulangan, dan tombol *Start* untuk memulai *pomodoro timer*. Setelah pengguna mengisi semua kolom, sistem aplikasi akan memulai menghitung mundur sesuai dengan waktu yang ditentukan oleh pengguna. *Pomodoro timer* akan mengeluarkan suara saat dimulai dan saat selesai.

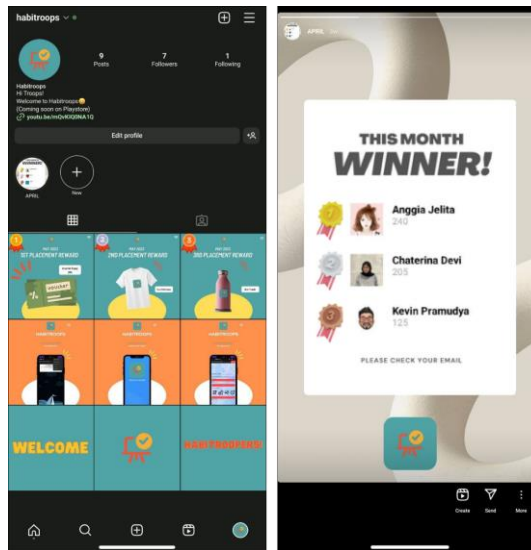
H. Halaman Leaderboard Gamification



Gambar 24. Halaman Leaderboard Gamification

Pada gambar 24 terdapat fitur *leaderboard* yang menampilkan urutan peringkat pengguna berdasarkan jumlah poin yang diraih. Terdapat kolom untuk nama pengguna, jumlah poin yang diperoleh, serta foto profil pengguna. Pengguna memiliki kesempatan untuk mendapatkan hadiah setiap bulannya dengan syarat mencapai posisi 3 besar pada akhir bulan, dan poin akan di *reset* setiap bulannya. Terdapat juga tombol *logo* Instagram yang mengarahkan pengguna ke halaman Instagram HabiTroops.

1. Halaman Instagram HabiTroops



Gambar 25. Halaman Instagram HabiTroops

Pada gambar 25 terdapat halaman Instagram HabiTroops yang berisi informasi pemenang setiap bulan dan hadiah yang didapatkan. Halaman ini menampilkan informasi terkait pemenang setiap bulan dan hadiah yang diperoleh oleh para pemenang. Pengguna dapat melihat daftar pemenang dan hadiah yang telah diberikan sebagai bentuk penghargaan atas pencapaian dalam mencapai kebiasaan yang berhasil. Aplikasi ini didanai melalui pendapatan yang diperoleh dari iklan yang ditampilkan di dalamnya, sehingga dapat disediakan secara gratis.

V. SIMPULAN DAN SARAN

Penelitian ini berhasil dalam pembuatan aplikasi *mobile* menggunakan Kotlin yang memungkinkan pengguna untuk melakukan aktivitas kebiasaan secara teratur. Aplikasi HabitTroops dirancang dengan antarmuka yang intuitif dan fitur-fitur yang memudahkan pengguna dalam mengatur dan melacak kebiasaan. Saran pengembangan meliputi penambahan fitur interaktif, integrasi dengan platform lain, personalisasi, fitur gamifikasi dengan hadiah beragam dan rencana prestasi, serta menyediakan kebiasaan default. Selain itu, saran pelaksanaan penelitian meliputi pengumpulan data lebih luas, pengujian lanjutan seperti *user testing*, dan evaluasi efektivitas fitur secara kuantitatif maupun kualitatif. Implementasi saran-saran tersebut diharapkan dapat meningkatkan pengalaman pengguna dan efektivitas aplikasi HabitTroops dalam membantu pengguna menjalankan kebiasaan dengan lebih baik.

DAFTAR PUSTAKA

- [1] N. Hidayat, "Implementasi Pendidikan Karakter Melalui Pembiasaan Di Pondok Pesantren Pabelan," Jurnal Pendidikan Sekolah Dasar Ahmad Dahlan, vol. 2, no. 1, pp. 95-106, 2015.
- [2] G. Monica Goeifanny et al., "PERANCANGAN BUKU PANDUAN PEMAHAMAN KEBIASAAN BURUK ANAK USIA 1-4 TAHUN." M. Wegmuller, J. P. von der Weid, P. Oberson, & N. Gisin, "High resolution fiber distributed measurements with coherent OFDR," *Prosiding ECOC'00*, 2000, paper 11.3.4, p. 109.
- [3] N. Gusti Ayu Putu Harry Saptarini et al., "AJARINCODE: APLIKASI PEMBELAJARAN BAHASA PEMROGRAMAN BERBASIS WEB."
- [4] D. Gusriyon, "Membuat Aplikasi Penyimpanan dan Pengolahan Data dengan VB.NET," Jurnal KomtekInfo, vol. 5, no. 1, pp. 2018, doi: 10.35134/komtekinfo.v5i1.10.M. Shell. (2002) IEEEtran homepage on CTAN. [Online]. Tersedia: <http://www.ctan.org/tex-archive/macros/latex/contrib/supported/IEEEtran/>
- [5] A. Febriandirza, "Perancangan Aplikasi Absensi Online Dengan Menggunakan Bahasa Pemrograman Kotlin," *Pseudocode*, vol. 7, no. 2, pp. 123-133, 2020. "PDCA12-70 data sheet," Opto Speed SA, Mezzovico, Switzerland.
- [6] R. P. Irsan, M. Lestari, and N. W. Parwati, "Intelligent Learning Online Education Berbasis Aplikasi Mobile," presented at Semnas Ristek (Seminar Nasional Riset dan Inovasi Teknologi), vol. 6, no. 1, Jan. 2022.
- [7] E. D. Listiono, A. Surahman, and S. Sintaro, "ENSIKLOPEDIA ISTILAH GEOGRAFI MENGGUNAKAN METODE SEQUENTIAL SEARCH BERBASIS ANDROID STUDI KASUS: SMA TELADAN WAY JEPARA LAMPUNG TIMUR," Jurnal Teknologi dan Sistem Informasi (JTSI), vol. 2, no. 1, p. 35, 2021, [Online]. Tersedia pada: <http://jim.teknokrat.ac.id/index.php/JTSI>.
- [8] H. Supriyono et al., "RANCANG BANGUN APLIKASI PEMBELAJARAN HADIS UNTUK PERANGKAT MOBILE BERBASIS ANDROID," 2014.

- [9] S. Surahman and E. B. Setiawan, "Aplikasi mobile driver online berbasis Android untuk perusahaan rental kendaraan," *Ultima InfoSys: Jurnal Ilmu Sistem Informasi*, vol. 8, no. 1, pp. 35-42, 2017.
- [10] A. Z. Mubarak, C. Carudin, and A. Voutama, "Perancangan User Interface/User Experience Pada Aplikasi Baby Spa Berbasis Mobile Untuk User Customer Dan Terapis Menggunakan Metode User Centered Design," *Jurnal Pendidikan dan Konseling (JPDK)*, vol. 4, no. 5, pp. 6368-6380, 2022.
- [11] S. Mulyati, "Kreativitas Matematis Siswa Pada Pembelajaran Discovery Learning Dengan Media Berbasis Android Studio," *PRISMA*, vol. 2, pp. 788-797, 2019, [Online]. Tersedia pada: <https://journal.unnes.ac.id/sju/index.php/prisma/>.
- [12] G. R. Payara, R. Tanone, "Penerapan Firebase Realtime Database Pada Prototype Aplikasi Pemesanan Makanan Berbasis Android," *Jurnal Teknik Informatika dan Sistem Informasi*, vol. 4, no. 3, pp. 397-406, 2018.
- [13] M. Nazir, S. Fajariani Putri, D. Malik, "Perancangan Aplikasi E-VOTING Menggunakan Diagram UML (Unified Modelling Language)," 2022.
- [14] P. Bitrián, I. Buil, S. Catalán, "Enhancing user engagement: The role of gamification in mobile apps," *Journal of Business Research*, vol. 132, pp. 170-185, 2021.
- [15] S. D. Putra, D. Aryani, H. D. Ariessanti, "Pemanfaatan Aplikasi Gamifikasi Wordwall Di Era Pandemi Covid-19 Untuk Meningkatkan Proses Pembelajaran Daring," *TERANG*, vol. 4, no. 1, pp. 83-90, 2021.
- [16] F. Maulana, R. Afyenni, A. Erianda, "Aplikasi Manajemen Laboratorium Menggunakan Metode MVVM Berbasis Android," *JITSI: Jurnal Ilmiah Teknologi Sistem Informasi*, vol. 3, no. 3, pp. 88-93, 2022.
- [17] R. B. D. Putra, E. S. Budi, A. R. Kadafi, "Perbandingan Antara SQLite, Room, dan RBDLiTe Dalam Pembuatan Basis Data pada Aplikasi Android," *JURIKOM (Jurnal Riset Komputer)*, vol. 7, no. 3, pp. 376-381, 2020.