

Perancangan Aplikasi Aspirasi FIKSpresif Mahasiswa Berbasis Android pada Fakultas Ilmu Komputer UPN “Veteran” Jakarta

Ridho Dwi Darma Putra ¹, Jenny Aulia Az Zahra ², Raihan Shafa ³, Lisken Ratna Kinasih ⁴, Radinal Setyadinsa ^{5*}

Fakultas Ilmu Komputer

Universitas Pembangunan Nasional “ Veteran” Jakarta

Jalan RS. Fatmawati Raya, Pd. Labu, Kec. Cilandak, Kota Jakarta Selatan, Daerah Khusus Ibukota Jakarta 12450

e-mail: 12310501049@mahasiswa.upnvj.ac.id ¹; 22310501057@mahasiswa.upnvj.ac.id ²;
32310501069@mahasiswa.upnvj.ac.id ³; 42310501080@mahasiswa.upnvj.ac.id ⁴;
radinalsetyadinsa@upnvj.ac.id ^{5*}

Abstrak. Penyampaian aspirasi mahasiswa di lingkungan kampus seringkali masih dilakukan secara manual dan kurang efektif, sehingga menyulitkan proses komunikasi antara mahasiswa dan pihak fakultas. Untuk menjawab kebutuhan akan media yang lebih praktis dan mudah diakses, di rancanglah aplikasi FIKSpresif berbasis Android yang memungkinkan mahasiswa menyampaikan aspirasi secara langsung melalui ponsel. Aplikasi ini menyediakan fitur pengiriman aspirasi, pilihan untuk anonim, serta pengelolaan akun dan riwayat aspirasi. Dengan memanfaatkan teknologi mobile, aplikasi ini bertujuan untuk menciptakan lingkungan komunikasi dua arah yang lebih terbuka dan responsif di Fakultas Ilmu Komputer UPN “Veteran” Jakarta. Proses perancangan dilakukan melalui tahapan analisis kebutuhan, desain sistem, hingga implementasi tampilan antarmuka.

Kata Kunci: Android, Aplikasi, Aspirasi, Mobile.

1 Pendahuluan

Aspirasi mahasiswa mencakup berbagai kebutuhan dan permintaan yang disusun dalam bentuk ide-ide kreatif guna mengusulkan perubahan pada suatu hal. Aspirasi mahasiswa dianggap lebih efektif jika tidak hanya disampaikan secara lisan, tetapi juga secara tertulis, disertai argumen ilmiah yang memperhatikan norma dan prinsip keilmuan. Aspirasi memiliki nilai penting sebagai alat untuk mengevaluasi dan mendeteksi kelemahan dalam sistem kualitas perguruan tinggi, sehingga dapat diperbaiki. Oleh karena itu, aspirasi mahasiswa diperlukan untuk meningkatkan standar kualitas kampus [1].

Berdasarkan hasil observasi lapangan dan wawancara dengan Badan Eksekutif Mahasiswa (BEM) Fakultas Ilmu Komputer UPN “Veteran” Jakarta, mekanisme penyampaian aspirasi mahasiswa saat ini masih mengandalkan kotak saran fisik yang diletakkan di area fakultas dan penyampaian langsung kepada pihak birokrasi. Hal ini menyebabkan beberapa kendala, seperti aspirasi yang disampaikan tidak terdokumentasi dengan baik, keterlambatan tindak lanjut, serta kurangnya transparansi proses penyampaian dan penanganan aspirasi. Selain itu, belum ada platform digital terintegrasi yang memfasilitasi komunikasi dua arah antara mahasiswa dan pihak fakultas secara real-time. Oleh karena itu, dibutuhkan solusi digital yang dapat diakses kapan saja dan di mana saja untuk mempercepat, mempermudah, dan memantau tindak lanjut aspirasi secara transparan. Metode pengumpulan data untuk perancangan sistem ini dilakukan melalui wawancara dengan pengurus BEM Fakultas Ilmu Komputer dan observasi langsung terhadap alur penyampaian aspirasi yang berlaku. Data yang diperoleh digunakan untuk mengidentifikasi kebutuhan sistem, merancang fitur utama aplikasi, dan memvalidasi desain solusi yang diajukan. Fakultas Ilmu Komputer UPN “Veteran” Jakarta memerlukan media modern dan terintegrasi untuk mendukung

budaya komunikasi dua arah antara mahasiswa dan fakultas. Ada keyakinan bahwa aplikasi aspirasi mahasiswa berbasis Android dapat berfungsi sebagai alat yang berguna untuk menerima, mengawasi, dan melanjutkan tujuan siswa.

Aplikasi digital berbasis Android menawarkan solusi inovatif untuk membuat penyampaian aspirasi lebih mudah, cepat, dan transparan seiring dengan pesatnya perkembangan teknologi informasi. Aplikasi Aspirasi Mahasiswa berbasis Android memungkinkan mahasiswa secara langsung mengirimkan masukan, keluhan, dan ide konstruktif melalui perangkat seluler mereka. Selain itu, terdapat fitur anonim yang dapat menjaga privasi mahasiswa. Privasi mahasiswa terjaga karena adanya pemilihan privasi pada form, sehingga mahasiswa dapat memilih untuk dianonimkan atau tidak nama mahasiswa yang mengirim aspirasi [2].

2 Metodologi Penelitian

2.1 Android Studio

Android Studio adalah sebuah *Integrated Development Environment* (IDE) yang digunakan untuk membuat aplikasi. Aplikasi IDE membantu pengembang membangun, menguji, dan mempublikasikan aplikasi. Android Studio dirilis oleh Google pada tanggal 16 Mei 2013 dan tersedia secara gratis di bawah lisensi Apache 2.0. Aplikasi ini berfungsi sebagai pengganti Eclipse, aplikasi pengembangan Android sebelumnya. Berbagai fitur Android Studio seperti debugger dan emulator memudahkan proses pembuatan aplikasi [3]. Dengan menyediakan berbagai fitur dan alat yang diperlukan, Android Studio sangat membantu pengembang dalam membuat aplikasi Android yang inovatif dan bermanfaat [10].

2.2 XAMPP

Software web server Apache bernama XAMPP memiliki database server mysql dan dukungan untuk programing PHP [4]. XAMPP adalah kumpulan berbagai program yang mendukung banyak sistem operasi dan merupakan perangkat lunak open source. Fungsi utama XAMPP adalah sebagai server yang berdiri sendiri (localhost), yang memiliki beberapa program seperti Apache HTTP Server, database MySQL, dan penerjemah bahasa yang menggunakan PHP dan Perl [5]. Software ini mudah digunakan, gratis, dan sangat membantu dalam pengembangan aplikasi web karena menyediakan semua kebutuhan utama dalam satu paket instalasi. Dengan dukungan instalasi yang sederhana, XAMPP telah menjadi favorit pengembang karena memungkinkan mereka membuat dan menguji aplikasi secara lokal sebelum dikirim ke server produksi.

2.3 Java

JAVA adalah bahasa pemrograman berorientasi objek tingkat tinggi, berbasis kelas, yang dirancang untuk meminimalkan ketergantungan implementasi. Ini adalah bahasa pemrograman tujuan umum yang memungkinkan programmer menulis sekali dan berjalan di mana saja. Kode Java yang dikompilasi dapat berjalan di semua platform yang mendukung Java tanpa perlu dikompilasi ulang. James Gosling awalnya mengembangkan Java di *Sun Microsystems*; itu dirilis sebagai komponen inti dari platform *Java Sun Microsystems* pada Mei 1995. Kompiler Java, mesin virtual, dan perpustakaan kelas awalnya dirilis di bawah lisensi hak milik Sun. Sintaks Java mirip dengan C dan C++, tetapi keduanya memiliki fitur yang lebih rendah [6].

2.4 Aspirasi

Aspirasi dilakukan dengan tujuan untuk meraih keberhasilan di masa depan. Biasanya, aspirasi ini berisi keluhan dan harapan [1]. Dengan kata lain, aspirasi merupakan dorongan kuat untuk meraih tujuan serta mengubah kondisi menjadi lebih baik.

2.5 Mobile

Sistem atau perangkat lunak yang dapat digunakan pada perangkat berbasis mobile disebut aplikasi berbasis mobile. Perangkat lunak yang dikembangkan dirancang untuk memenuhi kebutuhan pengguna. Selanjutnya, kebutuhan ini akan menjadi alasan untuk membangun sistem. Pengembang akan mengimplementasikan ke dalam source code sesuai dengan kebutuhan yang sudah ada [7] [9]. Sebuah aplikasi dapat dibuat dari source code [7].

2.6 Tahapan Penelitian

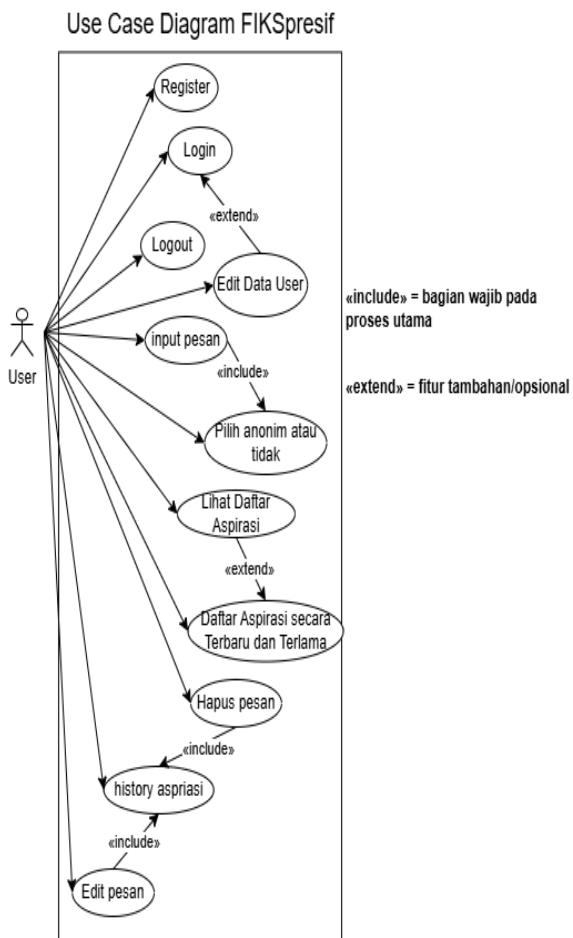
Tahapan penelitian ini digambarkan dalam bentuk diagram alir pada Gambar 1. Diagram ini menjelaskan langkah-langkah yang dilakukan mulai dari identifikasi masalah, pengumpulan data, analisis kebutuhan, desain sistem, implementasi, pengujian, hingga evaluasi.



Gambar. 1. Diagram Alir

2.7 Use Case Diagram

Aplikasi aspirasi berbasis android yang dirancang oleh penulis dapat diakses secara fleksibel melalui smartphone, sehingga memudahkan pengguna untuk mengaksesnya kapan saja dan di mana saja. Rancangan *use case* diagram untuk aplikasi aspirasi dapat dilihat pada Gambar 2.



Gambar. 2. Use Case Diagram

Gambar 2 menggambarkan fungsi utama yang dapat dilakukan oleh user seperti registrasi, Login, logout, input aspirasi, melihat daftar aspirasi dan dapat melakukan Edit serta hapus aspirasi yang telah dikirim. User juga dapat menentukan apakah aspirasi yang dikirim bersifat anonymous atau tidak.

2.8 Activity Diagram

Untuk menggambarkan alur kerja aplikasi penulis membuat activity diagram untuk setiap proses pada aplikasi aspirasi ini mulai dari Login, Register, input aspirasi, Edit dan hapus aspirasi, detail aspirasi, home page, logout dan Edit account. Activity diagram untuk setiap proses dapat dilihat pada tabel 1.

Tabel 1. Activity Diagram

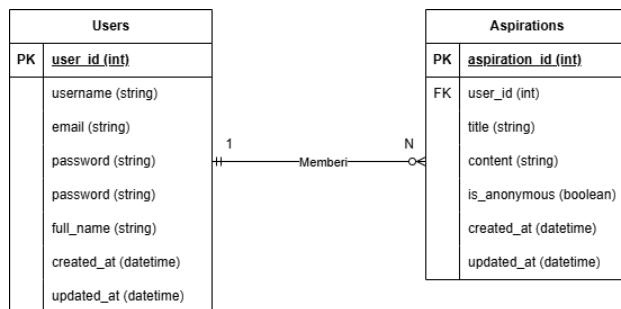
Proses	Penjelasan	Diagram
Login dan Register	<p>Activity diagram ini menjelaskan proses Login dan registrasi pengguna. Pengguna memulai dengan membuka halaman Login, mengisi username dan password, lalu klik tombol Login. Sistem memvalidasi input, mencocokkan data dengan database, dan jika sesuai, pengguna diarahkan ke halaman utama.</p> <p>Jika data tidak valid atau akun tidak ditemukan, sistem menampilkan pesan kesalahan. Pengguna dapat memilih untuk mendaftar dengan mengisi formulir. Setelah validasi berhasil, data disimpan ke database, dan sistem menampilkan pesan "Registrasi berhasil, silakan Login".</p>	<pre> graph TD Start(()) --> UserStart((User)) UserStart --> UserOpen[Menjalankan user 'Buka halaman login'] UserOpen --> UserInput[Input username dan password] UserInput --> UserLogin[Click tombol login] UserLogin --> SystemCheck[Menampilkan error 'Data tidak valid' / Valid] SystemCheck --> Decision1{Jadi data kosong?} Decision1 --> UserLogout[Menjalankan user 'Logout' / Berhasil] Decision1 --> UserRegister[Menjalankan user 'Buka halaman daftar' / Valid] UserRegister --> UserInputForm[Input data form pengajuan aspirasi] UserInputForm --> SystemCheckForm[Menampilkan error 'Data tidak valid' / Valid] SystemCheckForm --> Decision2{Validasi data kosong atau format salah?} Decision2 --> UserLogoutForm[Menjalankan user 'Logout' / Berhasil] Decision2 --> UserRegisterForm[Menjalankan user 'Buka halaman daftar' / Valid] UserRegisterForm --> UserInputForm UserLogout --> End(()) UserLogoutForm --> End UserRegister --> DatabaseSave[Simpan data user baru] DatabaseSave --> UserSuccess[Menampilkan halaman utama] UserSuccess --> End </pre>
Input Aspirasi	<p>Pengguna memulai dengan membuka halaman utama, kemudian mengklik ikon "Formulir Pengajuan Aspirasi" untuk menampilkan formulir. Setelah itu, pengguna mengisi judul, isi aspirasi, dan memilih apakah identitas ditampilkan atau anonim. Ketika tombol "Kirim Aspirasi" ditekan, sistem memvalidasi data yang dimasukkan. Jika terdapat data kosong atau format yang salah, sistem menampilkan pesan kesalahan "Data tidak valid".</p> <p>Jika data dinyatakan valid, sistem menyusun data aspirasi sesuai pilihan pengguna (anonim atau tidak), kemudian menyimpannya ke dalam database. Setelah berhasil disimpan, sistem menampilkan pesan bahwa aspirasi telah berhasil dikirim.</p>	<pre> graph TD Start(()) --> UserStart((User)) UserStart --> UserOpen[Menjalankan user 'Buka halaman utama'] UserOpen --> UserIcon[Click icon 'Formulir Pengajuan Aspirasi'] UserIcon --> SystemForm[Menampilkan form pengajuan] SystemForm --> UserInput[Input Judul, Isi, atau Status Anonim] UserInput --> SystemCheck[Menampilkan error 'Data tidak valid' / Valid] SystemCheck --> Decision1{Validasi data kosong atau format salah?} Decision1 --> UserLogout[Menjalankan user 'Logout' / Berhasil] Decision1 --> UserInputForm[Input data form pengajuan aspirasi] UserInputForm --> SystemCheckForm[Menampilkan error 'Data tidak valid' / Valid] SystemCheckForm --> Decision2{Validasi data kosong atau format salah?} Decision2 --> UserLogoutForm[Menjalankan user 'Logout' / Berhasil] Decision2 --> UserInputForm UserInputForm --> SystemSave[Simpan data aspirasi ke database] SystemSave --> UserSuccess[Menampilkan pesan 'Aspirasi berhasil dikirim' / Berhasil] UserSuccess --> End(()) </pre>

Proses	Penjelasan	Diagram
Edit dan Hapus Aspirasi	<p>Pengguna membuka halaman utama dan memilih "History Aspirasi". Sistem menampilkan daftar aspirasi dari database. Setelah memilih salah satu aspirasi, pengguna bisa mengedit atau menghapusnya.</p> <p>Jika mengedit, pengguna mengubah data lalu menyimpannya. Sistem memvalidasi dan menyimpan perubahan jika valid. Jika menghapus, sistem meminta konfirmasi, lalu menghapus data dari database dan menampilkan pesan sukses.</p> <p>Proses selesai setelah data diperbarui atau dihapus.</p>	<pre> graph TD Start((Start)) --> User[User] User --> Home[Home] Home --> History[History Aspirasi] History --> ListAspirasi[Daftar aspirasi] ListAspirasi --> PilihAspirasi[Pilih aspirasi] PilihAspirasi --> EditOrDelete{Edit atau Hapus} EditOrDelete --> Edit[Edit] Edit --> ValidasiEdit[Validasi edit] ValidasiEdit --> SimpanEdit[Simpan edit] SimpanEdit --> PesanEdit[Data berhasil diperbarui] EditOrDelete --> Delete{Hapus} Delete --> KonfirmasiDelete{Konfirmasi hapus} KonfirmasiDelete --> Hapus[Delete] Hapus --> PesanHapus[Data berhasil dihapus] PesanEdit --> Selesai[Selesai] PesanHapus --> Selesai </pre>
Detail Aspirasi dan Urutkan Daftar Aspirasi (Home Page)	<p>Pengguna membuka halaman utama, lalu sistem menampilkan daftar aspirasi dengan urutan terbaru. Pengguna dapat memilih salah satu aspirasi untuk melihat detailnya, dan sistem akan menampilkan informasi lengkap seperti nama pengguna (atau anonim), judul, isi, dan tanggal dibuat.</p> <p>Selain itu, pengguna juga bisa mengurutkan daftar aspirasi berdasarkan waktu (terbaru atau terlama). Sistem kemudian mengambil data sesuai urutan yang dipilih dan menampilkannya kembali di layar. Jika tidak ada aksi yang dipilih, maka proses selesai.</p>	<pre> graph TD Start((Start)) --> User[User] User --> Home[Home] Home --> ListAspirasi[Daftar aspirasi] ListAspirasi --> PilihAspirasi[Pilih aspirasi] PilihAspirasi --> Detail{Detail} Detail --> PesanDetail[Data berhasil diperbarui] Detail --> Urutkan{Urutkan} Urutkan --> PesanUrutkan[Data berhasil diperbarui] PesanDetail --> Selesai[Selesai] PesanUrutkan --> Selesai </pre>
Logout dan Edit Account	<p>Pengguna membuka halaman utama dan memilih "Account", di mana sistem menampilkan dua pilihan: Edit Profil dan Logout. Jika pengguna memilih Edit Profil, sistem mengambil data pengguna dari database dan menampilkannya dalam form Edit. Pengguna kemudian bisa mengubah data dirinya (seperti nama atau email) dan menekan tombol Edit Profil. Sistem akan memvalidasi data yang diinput. Jika data valid, maka sistem mengirim pembaruan ke database dan menampilkan pesan "Data profil berhasil diperbarui". Jika tidak valid, sistem menampilkan pesan error.</p>	<pre> graph TD Start((Start)) --> User[User] User --> Home[Home] Home --> Account[Account] Account --> EditOrLogout{Edit atau Logout} EditOrLogout --> Edit[Edit Profil] Edit --> ValidasiEdit[Validasi edit] ValidasiEdit --> SimpanEdit[Simpan edit] SimpanEdit --> PesanEdit[Data berhasil diperbarui] Edit --> Logout{Logout} Logout --> PesanLogout[Data berhasil diperbarui] PesanEdit --> Selesai[Selesai] PesanLogout --> Selesai </pre>

Proses	Penjelasan	Diagram
	<p>Jika pengguna memilih Logout, sistem akan menampilkan konfirmasi. Jika pengguna menyetujui, maka sistem menghapus sesi Login, mengarahkan pengguna ke halaman Login, dan menampilkan pesan "Anda berhasil logout". Proses pun selesai.</p>	

2.9 Class Diagram

Gambar 3 menunjukkan diagram kelas aplikasi Aspirasi berbasis Android. Diagram kelas membantu menjelaskan struktur data dan hubungan antara tabel dalam sistem database.

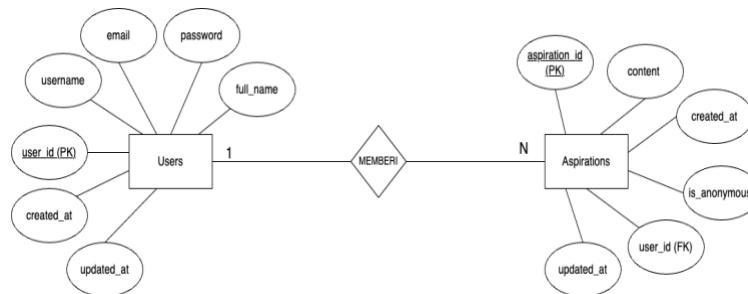


Gambar. 3. Class Diagram

Diagram kelas pada gambar 3 terdiri dari dua kelas utama: Users dan Aspirations. Kelas Users menyimpan data pengguna seperti identitas, nama, email, dan password. Kelas Aspirations menyimpan identitas, isi pesan, waktu pengiriman, dan status anonim. Dengan menggunakan asosiasi "memberi" dengan kardinalitas satu-ke-banyak, satu pengguna dapat mengirim banyak aspirasi, sedangkan satu aspirasi hanya dikirim oleh satu pengguna dalam relasi antar kelas.

2.10 Entity Relationship Diagram

Entity Relationship Diagram (ERD) adalah model yang digunakan untuk menunjukkan hubungan antar data sistem. ERD sistem aplikasi aplikasi berbasis android dapat dilihat pada Gambar 4.

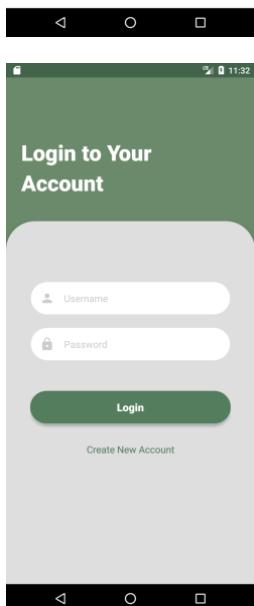


Gambar. 4. Entity Relationship Diagram

3 Hasil dan Pembahasan

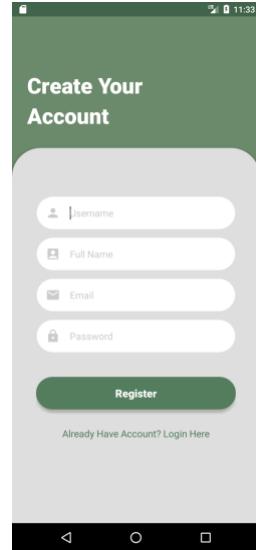
Hasil dari perancangan yang telah dilakukan penulis mengimplementasikannya pada aplikasi aspirasi FIKSpresif. Implementasi bertujuan untuk menggambarkan tampilan fungsi untuk masing-masing halaman pada FIKSpresif. Pengimplementasian yang telah dilakukan dapat dilihat pada tabel 2.

Tabel 2. Implementasi User Interface

Tampilan	Penjelasan	UI
<i>Splash Screen</i>	<i>Splash screen</i> merupakan tampilan awal saat pengguna membuka aplikasi yang tampil selama 5 detik.	
<i>Login</i>	<i>Login</i> merupakan tampilan setelah <i>splash screen</i> dimana user diminta untuk menginput <i>username</i> dan <i>password</i> akun untuk masuk ke aplikasi.	

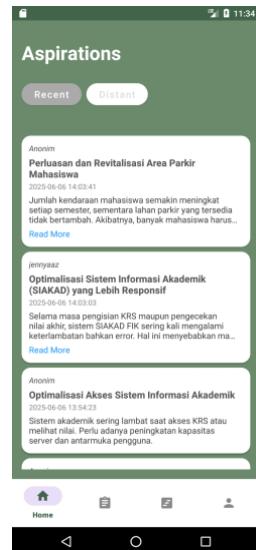
Register

Register merupakan tampilan yang digunakan untuk membuat akun bagi *user* yang belum memiliki akun.



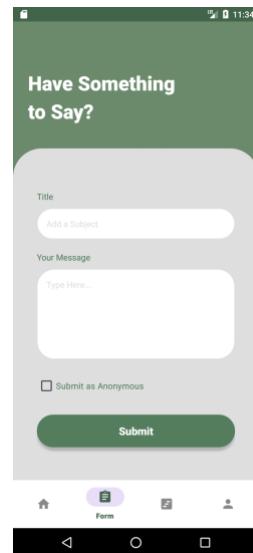
Home Page

Home page merupakan tampilan yang menampilkan informasi seluruh aspirasi yang telah masuk. Pada tampilan *home page* juga *user* dapat memfilter aspirasi berdasarkan waktu terbaru dan terlama.



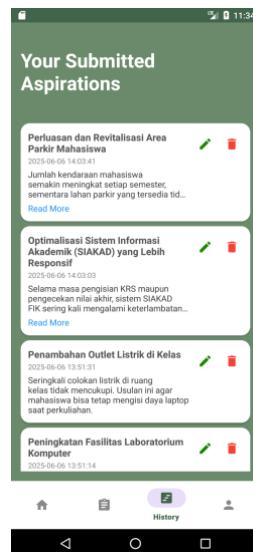
Input Aspirasi

Input aspirasi merupakan tampilan form yang digunakan *user* untuk mengirimkan aspirasi, *user* akan diminta mengisi judul dan isi aspirasi yang ingin disampaikan. *User* juga dapat memilih apakah aspirasi akan dikirim secara anonymous atau tidak.



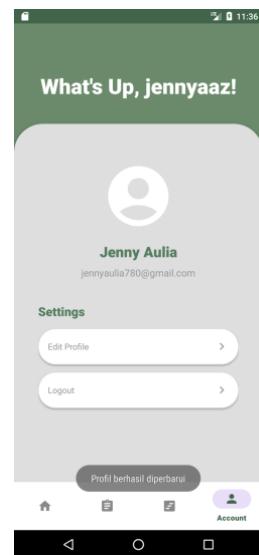
History Aspirasi

History aspirasi merupakan tampilan dimana semua aspirasi yang telah dikirim user untuk akun tersebut ditampilkan



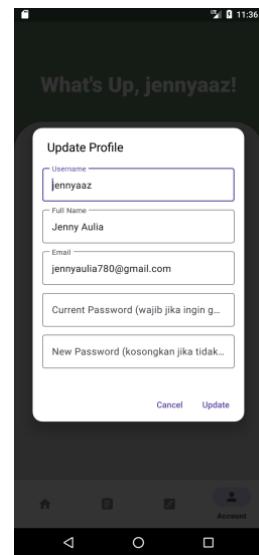
Account

Account merupakan tampilan detail akun.



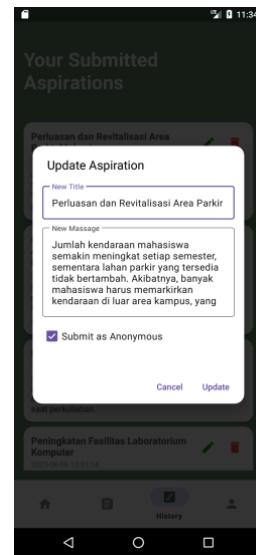
Edit *Account*

Edit *account* merupakan tampilan form yang digunakan untuk jika user ingin mengubah *username* ataupun *password*.

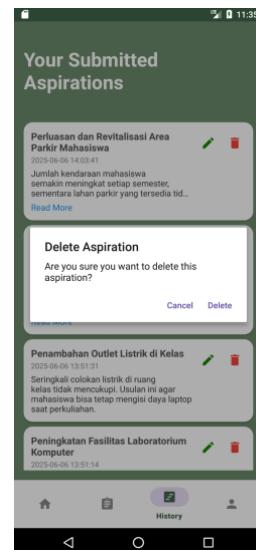


Edit Aspirasi

Edit aspirasi merupakan tampilan form yang digunakan jika user ingin mengubah aspirasi yang telah dikirim.


Hapus Aspirasi

Hapus aspirasi merupakan tampilan yang digunakan jika user ingin menghapus aspirasi yang telah dikirim yang menampilkan alert untuk memverifikasi penghapusan aspirasi.



3.1 Pengujian dan Evaluasi

Pengujian sistem dilakukan dengan metode black-box testing untuk memastikan setiap fitur aplikasi berjalan sesuai dengan spesifikasi. Hasil pengujian menunjukkan seluruh fitur inti seperti pengiriman aspirasi, pengaturan anonim, pengEditan, penghapusan aspirasi, dan pengelolaan akun dapat berfungsi tanpa error. Selain itu, dilakukan evaluasi terhadap 30 responden mahasiswa FIK UPNVJ menggunakan kuesioner berbasis skala Likert (1-5) terkait kemudahan penggunaan, kecepatan penyampaian aspirasi, dan transparansi proses. Hasil evaluasi menunjukkan 86% responden menyatakan aplikasi mempermudah penyampaian aspirasi, 90% menyatakan waktu penyampaian lebih cepat dibanding metode manual, dan 84% menilai fitur anonim memberikan rasa aman.

Langkah-langkah metode penelitian adalah sebagai berikut:

1. Identifikasi masalah penyampaian aspirasi di Fakultas Ilmu Komputer UPNVJ.
2. Pengumpulan data melalui observasi dan wawancara.
3. Analisis kebutuhan sistem berdasarkan data yang diperoleh.

ISSN: 2655-139X (ONLINE)

ISSN: 0216-4221 (PRINT)

4. Perancangan arsitektur dan antarmuka aplikasi.
5. Implementasi sistem menggunakan Android Studio dan teknologi pendukung.
6. Pengujian fungsional dan evaluasi kepuasan pengguna.
7. Penyusunan laporan dan publikasi hasil penelitian.

Agile adalah metode iteratif yang menekankan fleksibilitas, keterlibatan pengguna, dan pengembangan yang berulang melalui sprint atau iterasi kecil. Metode Agile memiliki beberapa pendekatan populer, seperti Scrum dan Kanban [8]. Agile memungkinkan tim untuk menyesuaikan perubahan kebutuhan secara dinamis, serta memberikan kesempatan bagi pengguna untuk memberikan umpan balik secara berkala agar sistem yang dibangun tetap relevan dan sesuai tujuan. Kelebihan metode ini terletak pada kemampuannya beradaptasi terhadap perubahan, dukungan terhadap pengembangan bertahap, dan kerja tim yang kolaboratif. Namun demikian, metode ini juga memiliki kelemahan, diantaranya membutuhkan partisipasi aktif dari pihak pengguna dan kurang cocok diterapkan pada proyek yang memerlukan dokumentasi formal yang sangat rinci dan ketat.

Tabel 3 menunjukkan hasil evaluasi terhadap 30 responden mahasiswa FIK UPNVJ menggunakan kuesioner berbasis skala Likert (1-5) terkait kemudahan penggunaan, kecepatan penyampaian aspirasi, dan transparansi proses. Hasil evaluasi menunjukkan 86% responden menyatakan aplikasi mempermudah penyampaian aspirasi, 90% menyatakan waktu penyampaian lebih cepat dibanding metode manual, dan 84% menilai fitur anonim memberikan rasa aman.

Tabel 3. Hasil Evaluasi

Aspek Penilaian	Persentase Setuju/Sangat Setuju
Kemudahan penggunaan	86%
Kecepatan penyampaian aspirasi	90%
Transparansi proses	82%
Keamanan dan privasi	84%

Pengujian yang dilakukan menggunakan black box testing dimana Tabel 4 merupakan hasil dari pengujian menggunakan black box testing yang didapat adalah bahwa semua berhasil dalam aplikasi yang digunakan.

Tabel 4. Hasil Pengujian

No	Fitur Diuji	Hasil Uji	Keterangan
1	Login & Registrasi	Berhasil	Data valid diterima, error input ditolak
2	Pengiriman Aspirasi (Anonim / Terbuka)	Berhasil	Data tersimpan & tampil di riwayat
3	Edit & Hapus Aspirasi	Berhasil	Data diperbarui/dihapus sesuai permintaan
4	Urutkan Aspirasi	Berhasil	Urutan sesuai pilihan pengguna
5	Pengelolaan Akun	Berhasil	Data profil berhasil diperbarui
6	Logout	Berhasil	Sesi terhapus, kembali ke login

4 Kesimpulan

Aplikasi FIKSpresif hadir sebagai sarana baru yang memudahkan mahasiswa Fakultas Ilmu Komputer UPN "Veteran" Jakarta dalam menyampaikan aspirasi secara praktis dan nyaman. Aplikasi ini menyediakan berbagai fitur seperti pengiriman aspirasi, sistem anonim, riwayat aspirasi, serta pengelolaan akun, yang

semuanya dapat diakses melalui perangkat Android. Dengan adanya platform digital ini, proses komunikasi dua arah antara mahasiswa dan pihak fakultas diharapkan menjadi lebih transparan, efektif, dan partisipatif. Dengan hadirnya aplikasi ini, diharapkan komunikasi antara mahasiswa dan pihak fakultas menjadi lebih terbuka dan berjalan dua arah. Selain itu, aplikasi ini juga mendorong mahasiswa untuk lebih aktif dalam menyampaikan pendapat dan ikut serta dalam membangun lingkungan kampus yang lebih baik. Aplikasi FIKSpresif juga mendukung peningkatan literasi digital serta budaya menyampaikan pendapat secara sehat di lingkungan kampus.

Referensi

- [1] P. Rizky Pangestu and A. Voutama, “PEMANFAATAN UML (UNIFIED MODELLING LANGUAGE) PADA SISTEM PENGELOLAAN ASPIRASI MAHASISWA BERBASIS WEBSITE,” *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 8, no. 6, pp. 11846–11851, Nov. 2024, doi: 10.36040/jati.v8i6.11732.
- [2] Moch M. N. M. M and S. Suhendri, “Rancang Bangun Sistem Informasi Aspirasi Mahasiswa Berbasis Web Dengan Framework Laravel (Studi Kasus : BPM Fakultas Teknik Universitas Majalengka),” *IKRA-ITH Informatika : Jurnal Komputer dan Informatika*, vol. 5, no. 2, pp. 85–92, Jan. 2021.
- [3] F. R. Chan, Harni Dusri, Muhammad Ramadani, Hanifah, and Liza Efriyanti, “Perancangan Aplikasi Pengelolaan Gudang Berbasis Android Menggunakan Android Studio,” *Journal of Informatics and Advanced Computing (JIAC)*, vol. 3, no. 2, pp. 103–107, Nov. 2022, doi: 10.35814/jiac.v3i2.4522.
- [4] I. P. Sari, A. Jannah, A. M. Meuraxa, A. Syahfitri, and R. Omar, “Perancangan Sistem Informasi Penginputan Database Mahasiswa Berbasis Web,” *Hello World Jurnal Ilmu Komputer*, vol. 1, no. 2, pp. 106–110, Jul. 2022, doi: 10.5621/helloworld.v1i2.57.
- [5] E. N. Hartiwati, “APLIKASI INVENTORI BARANG MENGGUNAKAN JAVA DENGAN PHPMYADMIN,” *Cross-border*, vol. 5, no. 1, pp. 601–610, Feb. 2022.
- [6] Mohamad Firdaus, “Perancangan aplikasi chat-room dengan prinsip threading melalui pemrograman dengan bahasa java,” *TEKNOSAINS : Jurnal Sains, Teknologi dan Informatika*, vol. 9, no. 2, pp. 121–135, Jul. 2022, doi: 10.37373/tekno.v9i2.242.
- [7] T. A. Nugrahani, D. Dianasari, and K. N. Amalia, “IMPLEMENTASI DESIGN THINKING PADA PERANCANGAN UI/UX APLIKASI PEMESANAN JAMU BERBASIS MOBILE,” *Jurnal Cakrawala Ilmiah*, vol. 3, no. 3, pp. 889–900, Nov. 2023.
- [8] L. Kismawan, “PERBANDINGAN METODE SDLC WATERFALL, AGILE, DAN SPIRAL DALAM PENGEMBANGAN SISTEM,” unknown. [Online]. Available: https://www.researchgate.net/publication/389438414_PERBANDINGAN_METODE_SDLC_WATERFALL_AGILE_DAN_SPIRAL_DALAM_PENGEMBANGAN_SISTEM
- [9] R. Dwi Cahyani, M. Ravtayya Attaturk Maklin, S. Ananta Karenina, S. Ashiila Rabbani, A. Intan Haifa Sumawidjaya, and R. Setyadinsa, “Perancangan Aplikasi Mobile ‘ProtectU’ untuk Pelaporan Kekerasan Seksual di Lingkungan Universitas Berbasis Android Studio,” *JURNAL INFORMATIK*, vol. 20, no. 1, pp. 16–27, 2024.
- [10] R. Setyadinsa, U. C. Pendit, and N. T. Hadi, “Food and Physical Activity Tracking Application with Simple Dietary Pattern Analysis,” *Sinkron*, vol. 9, no. 1, pp. 303–313, Jan. 2025, doi: 10.33395/sinkron.v9i1.14234.