

## NASKAH JURNAL

### Aplikasi Pencarian Tempat Kerja Praktek Menggunakan Firebase Bagi Mahasiswa Fasilkom Universitas Mercu Buana

Muhammad Mahendra Budiarto<sup>1</sup>, Sri Dianing Asri, ST, M.Kom<sup>2</sup>, Jurusan Teknik Informatika,  
Fakultas Ilmu Komputer, Universitas Mercu Buana<sup>1</sup> Jl. Raya Kranggan No 6 Jatisampurna, Bekasi,  
17443

E-mail : [41515210020@student.mercubuana.ac.id](mailto:41515210020@student.mercubuana.ac.id)<sup>1</sup>, [dianing.asri@mercubuana.ac.id](mailto:dianing.asri@mercubuana.ac.id)<sup>2</sup>

#### Abstrak

*Kerja Praktek adalah kegiatan pendidikan, pelatihan dan pembelajaran yang dilaksanakan didunia usaha atau dunia industri yang relevan dengan kompetensi (kemampuan) mahasiswa sesuai bidangnya sedangkan Android adalah sebuah sistem operasi berbasis Linux yang dikembangkan oleh Google. Peningkatan pesat penggunaan smartphone khususnya dengan platform android dan kebutuhan akan informasi lowongan kerja praktek yang menjadi kebutuhan mahasiswa Universitas Mercu Buana maka dari dua hal tersebut mendorong penulis untuk membuat sebuah aplikasi pencarian lowongan kerja praktek yang dapat memberikan akses secara cepat dan akurat kepada para mahasiswa untuk mencari kerja praktek, aplikasi dibangun dengan menggunakan firebase API yang ditawarkan oleh google untuk memudahkan pembuatan aplikasi android developer, Firebase merupakan sebuah penyedia layanan berupa database realtime dan backend yang dapat digunakan pada berbagai platform, dengan menggunakan menggunakan smartphone android para pencari kerja praktek akan dimudahkan dalam melihat detail dari pekerjaan hanya dengan melalui smartphone dan juga mendaftar ke kerja praktek tersebut hanya dengan mengirim CV melalui satu sentuhan di layar smartphone-nya.*

**Kata Kunci:** *Android, Kerja Praktek, Firebase*

#### Pendahuluan

Kerja praktek merupakan salah satu mata kuliah dan merupakan program dari jurusan Teknik Informatika Universitas Mercu Buana yang harus diikuti oleh setiap mahasiswa sebagai syarat untuk kelulusannya. Kerja praktek adalah kegiatan pendidikan, pelatihan dan pembelajaran yang dilaksanakan didunia usaha atau dunia industri yang relevan dengan kompetensi (kemampuan) siswa maupun mahasiswa sesuai bidangnya [1]. Teknologi sangat penting bagi Mahasiswa Teknik Informatika, maka dengan adanya kerja praktek ini diharapkan dapat melatih, mengembangkan dan menerapkan ilmu pengetahuan yang diperoleh selama kuliah bisa memperoleh ilmu

yang tidak didapat di bangku kuliah sekaligus memperoleh pengalaman yang nantinya bisa menjadi bekal untuk terjun ke dunia kerja setelah lulus kuliah dan mempunyai kualitas yang mampu bersaing.

Pada era globalisasi saat ini banyak perusahaan atau instansi menawarkan lowongan pekerjaan kepada masyarakat didalam negeri maupun luar negeri [2]. Oleh sebab itu maka diperlukan sebuah terobosan untuk mempermudah mahasiswa untuk mencari kerja praktek dalam mengatasi persoalan - persoalan dalam pencarian lowongan pekerjaan seperti yang telah disebutkan. Dengan menggunakan suatu aplikasi yang dapat menampilkan informasi

pekerjaan dan juga dapat memasukkan lamaran yang langsung dikirim ke penyedia lowongan yang kemudian akan langsung diproses oleh penyedia lowongan.

Caranya adalah penyedia lowongan bisa bekerja sama dengan kampus Universitas Mercu Buana dan menyampaikan pada *admin* untuk membuka lowongan kerja praktek bagi mahasiswa dan mengisi form data tentang deskripsi pekerjaannya. *Admin* sendiri yang menambahkan data lowongan berdasarkan form data yang diisi penyedia lowongan dan kemudian di posting di Aplikasi Bursa Kerja Praktek. Mengenai lowongan kerja praktek tersebut tidaklah sembarangan, *admin* harus memastikan bahwa data tersebut benar, dengan melihat langsung ke lapangan, mengecek pekerjaan atau menyamakan dengan data yang sudah ada.

Dalam implementasi akan menggunakan *firebase* sebagai database *realtime* dan *backend*. *Backend* sendiri adalah sebuah bagian dalam *code* aplikasi yang berhubungan langsung dengan isi *database*. dengan *firebase*, pengembangan aplikasi ini nantinya tidak perlu membuat *backend* sendiri melainkan memakai *API* yang telah disediakan oleh *Firebase*. Sehingga pengembang aplikasi dapat di persingkat. *Firebase* dikembangkan dengan menggunakan database *MongoDB* sehingga *firebase* menggunakan tipe database *NoSQL*. Karena memakai tipe database *NoSQL* maka struktur *database* dari *firebase* bersifat fleksibel dan cepat sehingga cocok untuk digunakan pada aplikasi berbasis *mobile* seperti yang akan peneliti kembangkan [3].

Tujuan dalam penelitian ini adalah untuk membuat aplikasi pencarian kerja praktek bagi mahasiswa Universitas Mercu Buana sehingga memudahkan dalam mengajukan kerja praktek ke perusahaan atau institusi. Manfaatnya adalah untuk membantu perusahaan yg sedang membutuhkan tenaga kerja magang yang bisa bekerja sama dengan Universitas Mercu Buana untuk membuka lowongan kerja praktek bagi mahasiswa yang membutuhkan nya.

## Landasan Pustaka

### Android

*Android* adalah sebuah sistem operasi berbasis *Linux* yang dikembangkan oleh *Google*. *Android* dikembangkan secara khusus untuk perangkat *mobile* yang memiliki perangkat *input* berupa layar sentuh. Pada awalnya, *android* dikembangkan khusus pada perangkat berbasis arsitektur *ARMv7* saja, namun setelah beberapa lama sebuah proyek bernama *Android-x86* memungkinkan *android* untuk mendukung perangkat berbasis arsitektur *x86*. *Android* dikembangkan secara *open source* sehingga para pengembang bisa mendistribusikan dan memodifikasi ulang secara bebas [4].

### Firebase

*Firebase* merupakan sebuah layanan infrastruktur *backend-as-a-service (BaaS)* yang diakuisisi oleh *Google* pada Oktober 2014 silam [5]. *Firebase* menawarkan kemudahan kepada para pengembang perangkat lunak dalam membangun aplikasi yang lebih baik serta mengembangkan bisnis yang sukses melalui seluruh fitur komplementernya. Saat ini, *Firebase* sudah memiliki fitur *Analytics*, *Cloud Messaging*, *Authentication*, *Realtime Database*, *Storage*, *Hosting*, *Test Lab*, *Crash Reporting*, *Notifications*, *Remote Config*, *App Indexing*, *Dynamic Links*, *Invites*, *AdWords*, dan *AdMob*. Seluruh fitur tersebut dikemas dalam sebuah *SDK Firebase* tunggal sehingga dengan kemudahan yang ditawarkan para pengembang perangkat lunak dapat fokus untuk memecahkan masalah *customer* melalui perangkat lunak yang dibuatnya dan tidak menghabiskan banyak waktu dalam membangun infrastruktur yang kompleks.

### Firebase Real Time Database

*Real Time Database* adalah sebuah *NoSQL database* yang disediakan oleh *firebase*. *NoSQL database* adalah *database* yang tidak menggunakan sistem relasi layaknya pada *database* tradisional (*MySQL* dll.). Metode penyimpanan data di dalam *NoSQL* menggunakan objek yang menggunakan format *JSON (JavaScript Object Notation)*. *firebase* memungkinkan kita untuk menggunakan *NoSQL database* yang di *share* kepada semua *user*, dan ketika terjadi perubahan data pada

*database* tersebut, *user* akan segera mendapatkan *update* data secara *real time*. Tetapi bukan berarti *database* ini tidak mempunyai unsur keamanan, karena kita bisa mengatur hak akses yang berbeda untuk setiap *user*. Salah satu fitur yang menarik adalah aplikasi bisa menyimpan data secara lokal ketika tidak ada koneksi *internet*, kemudian melakukan *sync* data segera setelah mendapatkan kembali koneksi *internet* [6].

### **Firestore Authentication**

Sebagian besar aplikasi tentu membutuhkan autentikasi untuk mengetahui identitas pengguna. Mengetahui identitas pengguna memungkinkan aplikasi untuk secara aman menyimpan data pengguna di *cloud* dan memberikan pengalaman pribadi yang sama di semua perangkat pengguna. *Firestore* menyediakan layanan *backend*, *SDK* yang mudah digunakan dan pustaka siap pakai untuk mengautentikasi pengguna ke aplikasi. Saat ini *Firestore* mendukung autentikasi menggunakan sandi, sosial media populer seperti *Google*, *Facebook*, *Twitter* dan lainnya. Ujicoba autentikasi yang dilakukan pada penelitian ini menggunakan email dan kata sandi.

#### a. Registrasi

Ketika pengguna pertama sekali mengakses aplikasi, maka akan ditampilkan formulir registrasi sederhana dengan meminta pengguna tersebut memasukkan email dan kata sandi. Secara otomatis setelah proses registrasi selesai, maka pengguna tersebut masuk ke aplikasi.

#### b. Login

Pengguna yang telah berhasil melakukan registrasi, maka ia bisa *login* ke aplikasi menggunakan *email* dan kata sandi tersebut.

#### c. Logout

*Logout* atau keluar dari aplikasi.

### **Firestore Storage**

*Firestore Storage* digunakan untuk menyimpan data seperti gambar, *audio* dan *video*. Sebagian aplikasi pasti menggunakan file gambar, *audio* atau *video*. Adanya *Firestore Storage* sangat memudahkan proses unggahan dan unduhan untuk aplikasi. Beberapa

kelebihan utama dari *Firestore Storage* adalah sebagai berikut:

#### a. Strong

*Firestore Storage* dapat melakukan unggahan dan unduhan file di semua kualitas jaringan *internet*. Ia dapat berhenti atau melanjutkan, sehingga menghemat waktu dan *bandwidth* pengguna.

#### b. Secure

*Firestore Storage* terintegrasi dengan *Firestore Authentication* untuk menyediakan autentikasi yang mudah dan intuitif untuk pengembang. Model keamanan dapat diatur berdasarkan nama file, ukuran, tipe konten dan metadata lainnya.

#### c. Scalable

*Firestore Storage* didukung oleh *Google Cloud Storage* hingga skala *petabyte*. Ini sangat penting, ketika aplikasi yang dari prototipe kemudian berkembang viral dengan jutaan pengguna [7].

### **Java**

Aplikasi *Android* dikembangkan dengan *Java*, bahasa pemrograman yang paling banyak digunakan di dunia. *Java* adalah bahasa pemrograman yang paling banyak digunakan merupakan pilihan yang logis untuk *platform Android*, karena itu sumber nya kuat, bebas, dan terbuka. *Java* digunakan untuk mengembangkan aplikasi perusahaan skala besar, untuk meningkatkan fungsi dari *web server*. Untuk menyediakan aplikasi untuk perangkat konsumen dan untuk banyak tujuan lain nya. *Java* memungkinkan anda untuk mengembangkan aplikasi yang akan berjalan pada berbagai perangkat tanpa kode *platform* khusus. Programmer *Java* dapat dengan cepat masuk ke dalam pengembangan *Android*, menggunakan *API Android (Application Programming Interface)* dan lain-lain yang tersedia dari pihak ketiga [8].

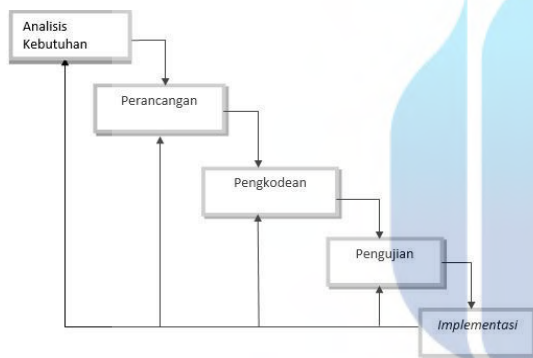
### **NoSQL**

*NoSQL* adalah suatu cara penyimpanan data (*data store*), dimana cara menyimpan dan mengambil kembali datanya dapat dilakukan dengan cepat, seperti basis data relasional pada umumnya, tetapi tidak berdasarkan relasi

matematika antar-tabel seperti pada relasional [9].

### **Metodologi Waterfall**

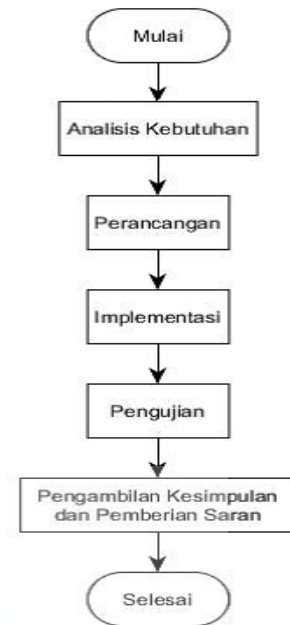
*Metodologi* pengembangan aplikasi yang digunakan adalah *waterfall*. *Metodologi waterfall* sudah digunakan secara luas untuk pengembangan sistem (aplikasi) perangkat lunak dan merupakan model klasik dari rekayasa perangkat lunak yang banyak digunakan dalam proyek – proyek pemerintah dan di perusahaan – perusahaan besar. *Metodologi* ini sangat terstruktur, tetapi cenderung bersifat linier dan tidak fleksibel. *Metodologi waterfall* memerlukan pendekatan yang sistematis dan sekuensial di dalam pengembangan sistem perangkat lunaknya [10].



Gambar 1 Metodologi Waterfall

### **Metode Penelitian**

Aplikasi Bursa Kerja praktek Bagi mahasiswa mercu buana pada sistem operasi *Android* merupakan sebuah penelitian dengan jenis penelitian implementatif pengembangan keminatan *mobile*. Penelitian ini memiliki cakupan pendekatan rancang bangun aplikasi atau sistem baru. Proses pelaksanaan penelitian ini terdiri dari tahap analisis kebutuhan, perancangan, implementasi, pengujian dan pengambilan kesimpulan serta pemberian saran. Aliran proses pelaksanaan penelitian tersebut dapat dilihat dalam Gambar 2.



Gambar 2 Tahapan Metode Penelitian

### **Analisis Kebutuhan**

Analisis kebutuhan merupakan tahap awal dalam pengembangan metode penelitian yang harus dilakukan dalam membangun suatu aplikasi. Tahap ini dilakukan untuk menganalisis dan mendefinisikan kebutuhan yang harus dipenuhi oleh aplikasi yang akan dibangun. Berdasarkan analisis yang dilakukan, maka penulis membagi menjadi beberapa kebutuhan antara lain:

#### 1. Kebutuhan *Admin*

- a) lowongan kerja praktek dari perusahaan.
- b) menginput data lowongan.
- c) membuka lowongan yang sudah penuh.
- d) Memprint data report lowomgan.

#### 2. Kebutuhan *User*

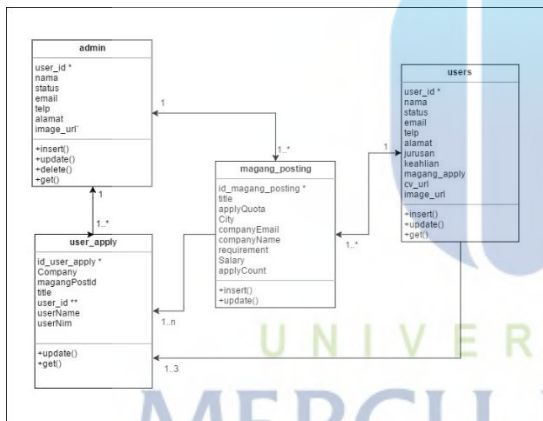
- a) Menginput data *user*.
- b) Mengupload CV.
- c) Dapat menampilkan lowongan Kerja Praktek.
- d) Dapat memilih lowongan Kerja Praktek.

- e) Dapat mengirim lowongan Kerja Praktek.

### Perancangan

Yang selanjutnya dilakukan setelah analisis kebutuhan yaitu melakukan perancangan sistem. Tahap ini dilakukan setelah semua kebutuhan sistem terpenuhi. Perancangan sistem dilakukan untuk memenuhi kebutuhan para pengguna sistem dan untuk memberikan gambaran yang jelas serta rancang bangun yang lengkap kepada programmer. Tahapan ini sangat penting dilakukan agar proses pembuatan aplikasi bisa ter-arah dan aplikasi yang dihasilkan berjalan dengan baik. Perancangan sistem ini terdiri dari *class diagram*, *use case diagram*, dan *activity diagram* aplikasi.

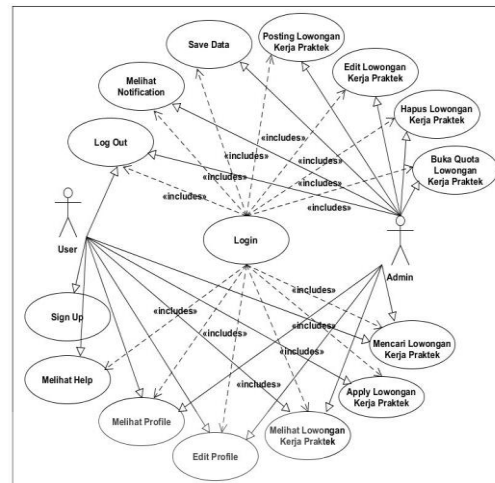
### Class Diagram



Gambar 3 Class Diagram

*Class diagram* pada gambar 3 proses sistem di atas, menjelaskan tentang kelas yang terdapat pada sistem berdasarkan proses yang ada di dalam sistem. *Class admin* dapat menambahkan, edit, dan hapus di *magang\_posting* serta menyimpan data report *user\_apply*. *Class user* dapat *apply* lowongan di *magang\_posting* setelah *apply* data akan masuk ke *user\_apply*. *Class magang\_posting* dapat di update, edit, dan hapus oleh *admin*, dan bisa di *apply* oleh *user* kemudian data akan di simpan ke *user\_apply*. *Class user\_apply* dapat menyimpan data *user* yang sudah meng-*apply* ke *magang\_posting* serta menyimpan data report lowongan yang terkirim untuk *admin*.

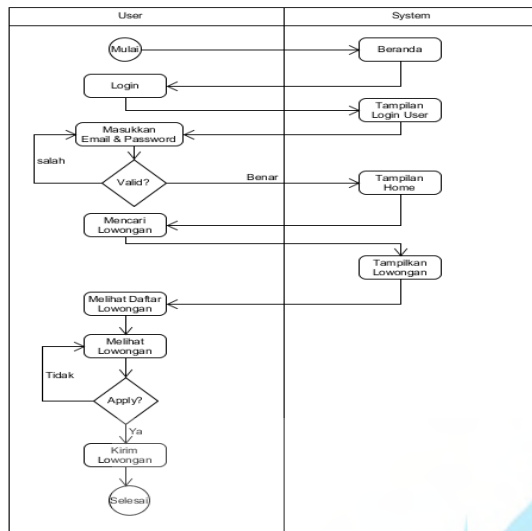
### Use Case Diagram



Gambar 4 Use Case Diagram

*Use Case Diagram* Pada gambar 4 merupakan diagram yang digunakan untuk memodelkan perangkat lunak berdasarkan kebutuhan fungsional perangkat lunak itu sendiri. Setiap kebutuhan fungsional yang dimaksud dimodelkan berdasarkan sudut pandang aktornya. Pada diagram *use case* yang ditunjukkan dalam Gambar terdapat 2 aktor yaitu *user* yang merupakan mahasiswa yang ingin mencari kerja praktek dan *Admin* yang merupakan operator yang mengupdate lowongan kerja praktek serta terdapat *use case* yang masing-masing mewakili setiap kebutuhan fungsional perangkat lunak yang telah dirancang sebelumnya.

## Activity Diagram



Gambar 5. Activity Diagram

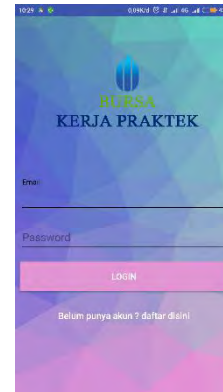
Activity diagram user pada gambar 5 yang menggambarkan proses terjadinya pencarian dalam sistem. Pengolahan data ini dimulai dari login di halaman login, jika sudah terdaftar maka akan masuk ke dalam tampilan home. Selanjutnya user mencari lowongan yang diinginkan, maka akan tampil daftar lowongan yang tersedia. Lalu user dapat melihat detail lowongan yang dipilih, jika YA, user akan mengirim CV yang sudah di upload ke email perusahaan. Jika sudah terkirim user tinggal menunggu kabar dari perusahaan untuk panggilan kerja praktek.

### Hasil Dan Pembahasan

#### Implementasi Antarmuka

Pada bagian ini diuraikan hasil implementasi antarmuka pada aplikasi Bursa Kerja Praktek :

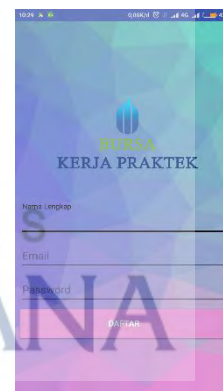
#### 1. Tampilan login



Gambar 6. Tampilan Login

Pada gambar 6 ini adalah tampilan pertama yaitu login, dimana pemilik akun harus login terlebih dahulu untuk masuk ke dalam aplikasi admin atau user. Jika user belum mempunyai akun bisa registrasi terlebih dahulu.

#### 2. Tampilan Registrasi



Gambar 7. Tampilan Registrasi

Pada gambar 7 ini adalah tampilan untuk registrasi akun untuk user jika belum mempunyai akun untuk login.

### 3. Tampilan *Home*



Gambar 8. Tampilan *Home*

Pada gambar 8 ini merupakan tampilan utama *user* dan *admin*, pada tampilan ini *user* dan *admin* dapat melihat lowongan kerja praktek serta memilih lowongan kerja praktek. Untuk *user* bisa mencari lowongan yang diinginkan. Sedangkan *admin* bisa *update* lowongan baru.

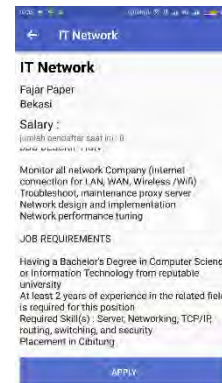
### 4. Tampilan Pencarian Lowongan



Gambar 9. Tampilan Lowongan

Pada gambar 9 ini merupakan tampilan saat *user* mencari lowongan yang diinginkan dan akan muncul lowongan yang sesuai di cari.

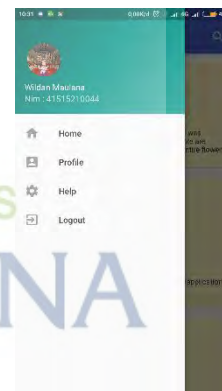
### 5. Tampilan Info Lowongan



Gambar 10. Tampilan Info Lowongan

Pada gambar 10 ini merupakan info lowongan yang berisi tentang *job*, *company*, lokasi, *salary*, dan *job requirement* yang bisa dilihat oleh *user* dan *admin*. *User* bisa *apply* untuk mengirim lowongan ke *job* tersebut, sedangkan *admin* bisa edit dan hapus lowongan tersebut.

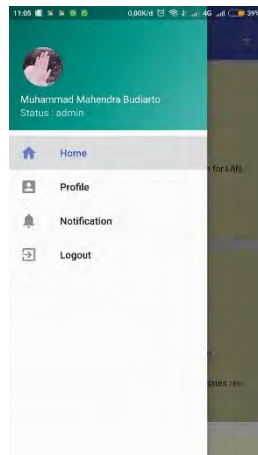
### 6. Tampilan *Menu User*



Gambar 11. Tampilan *Menu User*

Pada gambar 11 ini di jelaskan bagaimana tampilan tombol *menu* yang berisi tombol yang memiliki fungsi berbeda meliputi, tombol *home* untuk kembali ke halaman utama *home*, tombol *profile* yaitu tombol yang berfungsi untuk memperlihatkan *profile user*, tombol *help* yaitu tombol yang berfungsi untuk membantu *user* dalam menggunakan aplikasi, tombol *logout* yaitu untuk mengeluarkan akun *user* dari aplikasi.

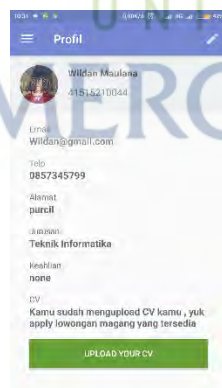
## 7. Tampilan Menu Admin



Gambar 12. Tampilan Menu Admin

Pada gambar 12 ini di jelaskan bagaimana tampilan tombol *menu* yang berisi tombol yang memiliki fungsi tak jauh berbeda meliputi, tombol *home* untuk kembali ke halaman utama *home*, tombol *profile* yaitu tombol yang berfungsi untuk memperlihatkan *profile admin*, tombol *notification* yaitu tombol yang berfungsi untuk memperlihatkan data report *user* yang sudah mengirim lowongan, tombol *logout* yaitu untuk mengeluarkan akun *admin* dari aplikasi.

## 8. Tampilan Profile User

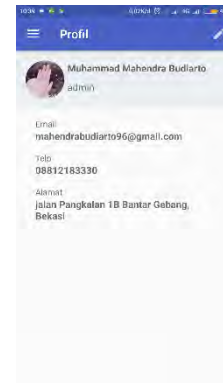


Gambar 13. Tampilan Profile User

Pada gambar 13 ini merupakan *profile user* yang harus di *input* berisi nama, nim, email, nomor telpon, alamat, jurusan, keahlian, dan CV yang harus di upload dari *user*. Dalam

*profile* juga bisa di edit untuk merubah data yang salah.

## 9. Tampilan Profile Admin



Gambar 14. Tampilan Profile Admin

Pada gambar 14 ini merupakan *profile admin* yang harus di *input* berisi nama, status, email, nomor telpon, dan alamat dari *admin*. Dalam *profile* juga bisa di edit untuk merubah data yang salah.

## 10. Tampilan Help



Gambar 15. Tampilan Help

Pada gambar 15 ini merupakan bantuan bagi *user* untuk menggunakan aplikasinya.



11. Tampilan *Notification*Gambar 16. Tampilan *Notification*

Pada gambar 16 ini merupakan notifikasi pada *admin* saat ada *user* yang mengirim apply lowongan dan bisa di unduh juga sebagai data report.

### Pengujian

Dalam penelitian ini pengujian yang dilakukan terhadap sistem yaitu pengujian secara fungsional. Metode yang digunakan dalam pengujian adalah pengujian *black-box* yang berfokus pada persyaratan fungsional dari sistem yang dibangun.

### Pengujian *Black-Box*

Pengujian *black-box* merupakan pengujian yang bersumber dari kebutuhan fungsional. Pengujian *black-box* berfungsi untuk menguji setiap fitur yang terdapat pada aplikasi apakah telah sesuai dengan kebutuhan fungsional atau tidak. Setiap fitur akan diuji sesuai dengan kebutuhan fungsionalnya menggunakan kasus uji yang berbeda-beda. Analisis terhadap hasil pengujian ini menghasilkan validitas sebesar 100% [11].

Tabel 1 Pengujian Black-Box

No.	Test Case	Hasil yang diharapkan	Hasil yang didapatkan	Ket
1.	Pengujian Masuk Aplikasi	Aplikasi akan menampilkan halaman <i>login</i>	Aplikasi menampilkan halaman <i>login</i>	Berhasil
2.	Pengujian Registrasi	Aplikasi akan menampilkan halaman registrasi <i>akun</i>	Aplikasi menampilkan halaman registrasi <i>akun</i>	Berhasil
3.	Pengujian <i>Login</i>	Aplikasi akan menampilkan halaman utama serta lowongan terbaru	Aplikasi menampilkan halaman utama serta lowongan terbaru	Berhasil
4.	Pengujian <i>Menu</i>	Aplikasi akan menampilkan <i>an menu</i>	Aplikasi menampilkan <i>an menu</i>	Berhasil
5.	Pengujian <i>Profile</i>	Aplikasi akan menampilkan info <i>profile</i> user/admin	Aplikasi menampilkan info <i>profile</i> user/admin	Berhasil
6.	Pengujian <i>Help</i>	Aplikasi akan menampilkan info <i>help</i>	Aplikasi menampilkan info <i>help</i>	Berhasil
7.	Pengujian <i>Notification</i>	Aplikasi akan menampilkan data lowongan yang sudah dikirim <i>user</i>	Aplikasi akan menampilkan data lowongan yang sudah dikirim <i>user</i>	Berhasil
8	Pengujian pencarian Lowongan	Aplikasi akan mencari hasil lowongan	Aplikasi menampilkan hasil pencarian lowongan	Berhasil

9.	Pengujian Lowongan	Aplikasi akan menampilkan info lowongan	Aplikasi menampilkan info lowongan	Berhasil
10.	Pengujian Apply Lowongan	Tombol ini akan mengirim apply lowongan	Tombol telah mengirim apply lowongan	Berhasil
11.	Pengujian Update Lowongan	Tombol ini akan menampilkan halaman update lowongan	Tombol ini telah menampilkan halaman update lowongan	Berhasil
12.	Pengujian Hapus Lowongan	Tombol ini untuk menghapus data lowongan	Tombol ini telah menghapus data lowongan	Berhasil
13.	Pengujian Unduh Data Report Notification	Tombol ini untuk unduh data report lowongan terkirim	Tombol ini telah unduh data report lowongan terkirim	Berhasil
14.	Pengujian Logout	Tombol ini untuk mengeluarkan akun dari aplikasi	Tombol ini telah mengeluarkan akun dari aplikasi	Berhasil

### Kesimpulan dan Saran

Dengan adanya aplikasi dapat membantu proses pencarian kerja praktek bagi mahasiswa agar tidak perlu lagi mencari secara personal dan datang ke perusahaan. Aplikasi ini dapat menampilkan beberapa *list* lowongan kerja praktek yang di tampilkan pada *menu* awal. Untuk mencari lowongan mahasiswa harus mengisi *profile user* terlebih dahulu. Aplikasi ini dapat membantu para perusahaan untuk mencarikan mahasiswa kerja praktek atau magang.

Saran dari penulis sistem ini untuk penelitian kedepannya, sistem ini perlu dikembangkan dalam masalah jumlah kriteria dan perluasan wilayah yang meliputi seluruh Fakultas. Dengan adanya pengembangan ini diharapkan para pelamar dari mahasiswa dapat

menemukan kerja praktek dengan mudah untuk syarat kelulusan.

### Daftar Pustaka

- [1] A. A. Rahman, "SISTEM INFORMASI PRAKERIN DENGAN METODE PUSH NOTIFICATION PADA BAGIAN HUBUNGAN INDUSTRI ( HUBIN ) DI SMK MATHLA ' UL ANWAR," 2018.
- [2] B. A. Nugroho, "Informasi Lowongan Kerja Berbasis Android (Studi Kasus di Disnaker Kab.Nganjuk)," *J. Inform. dan Multimed.*, vol. 08, no. 01, pp. 10–16, 2016.
- [3] A. Rozaq, "Pembangunan Aplikasi Brawijaya Messenger dengan menggunakan Platform Firebase pada Universitas Brawijaya," vol. 2, no. 2, pp. 667–673, 2018.
- [4] S. B. Ibe, "Aplikasi Crowdsorce pada Sistem Pencarian Tutor Berbasis Web dan Android," *J. Infra Univ. Petra*, vol. 3, pp. 2–6, 2015.
- [5] L. A. Sandy, R. Januar, and R. Hariadi, "Rancang Bangun Aplikasi Chat pada Platform Android dengan Media Input berupa Canvas dan Shareable Canvas untuk Bekerja Dalam Satu Canvas secara Online," vol. 6, no. 2, 2017.
- [6] P. R. Sutanto, H. N. Palit, and A. Handojo, "Aplikasi Manajemen Pelayanan Gereja Berbasis Android," no. 031, pp. 3–7, 2016.
- [7] M. Ilhami, "Pengenalan Google Firebase Untuk Hybrid Mobile Apps Berbasis Cordova," vol. 3, no. 124, pp. 16–29, 2017.
- [8] N. Arya, "Pembuatan Aplikasi Kegiatan Warga Pada RT 06 RW 06 Kelurahan Airlangga Berbasis Android," 2018.
- [9] Monika Mindari, "PENERAPAN

DOCUMENT ORIENTED  
DATABASE (NOSQL) PADA  
SISTEM INFORMASI RUMAH  
SAKIT UMUM DAERAH (RSUD)  
PRABUMULIH Monika,” no. 12,  
2017.

- [10] Kurnia Muludi, “Rancang Bangun Aplikasi Pembelajaran Iqra Berbasis Android,” *J. Momentum*, vol. 17, no. 1, pp. 1–9, 2015.
- [11] E. Winardo Toepak, “Pembangunan Aplikasi Penyedia Informasi Lowongan Pekerjaan Menggunakan Youtube API Pada Smartphone Android,” *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 2, no. 1, pp. 168–174, 2018.



UNIVERSITAS  
MERCU BUANA