


Sistem Monitoring Kerja Kerukunan Mahasiswa Pinrang Universitas Muslim Indonesia (KMP-UMI) Menggunakan *Push Notification*

Muh Asrai Taufik^{a,1,*}, Dolly Indra^{a,2}, Mardiyah Hasnawi^{a,3}

^aUniversitas Muslim Indonesia, Jalan Urip Sumoharjo, Makassar, 90231, Makassar
¹ azraytaufiq1997@gmail.com; ² dolly.indra@umi.ac.id; ³ mardiyah.hasnawi@umi.ac.id
*corresponding author

INFORMASI ARTIKEL	ABSTRAK
Diterima : 18 – 09 – 2021 Direvisi : 28 – 10 – 2021 Diterbitkan : 30 – 11 – 2021	Penyampaian informasi kerja kepada Badan Pengawas (MPO) Organisasi masih dilakukan secara konvensional dan belum adanya aplikasi yang dapat memberikan informasi kerja kepada MPO secara cepat. Penelitian ini bertujuan Untuk merancang dan membangun sistem yang menggunakan <i>push notification</i> yang dapat membantu pengurus KMP-UMI untuk mengetahui informasi program kerja secara <i>real time</i> . Penelitian ini menghasilkan sebuah Sistem Monitoring Kerja yang menggunakan layanan <i>push notification</i> yang memungkinkan badan pengawas (MPO) untuk mengetahui informasi program kerja secara <i>real time</i> . Sistem atau aplikasi ini dapat menjalankan fungsinya dengan baik dan efisien. Berdasarkan hasil pengujian beta menunjukkan form-form yang terdapat pada aplikasi berjalan dengan semestinya dan setiap validasi yang terdapat pada aplikasi menunjukkan sesuai perancangan aplikasi dengan total persentase pengujian betha yang dicapai sebesar 53,2%.
Kata Kunci: <i>Push Notification</i> <i>Real Time</i> Sistem Monitoring Kerja Informasi Program Kerja Pengujian Beta	

This is an open access article under the [CC-BY-SA](#) license



I. Pendahuluan

Organisasi Kerukunan Mahasiswa Pinrang melakukan monitoring dan pengukuran secara terus-menerus terhadap kinerja pengurus untuk memastikan ketercapaian tujuan yang telah ditetapkan. Informasi program kerja dan luaran kegiatannya diperoleh oleh badan pengawas (MPO) secara konvensional. Selain itu, Teknologi Informasi belum diterapkan dalam organisasi tersebut sehingga tindak lanjut atas kegiatan tersebut tidak dapat dilakukan secara cepat.

Push notification memperbolehkan sebuah aplikasi untuk memberitahukan aplikasi lainnya bila ada peristiwa baru yang terjadi secara *real time* tanpa harus membuka aplikasi tersebut [1]. Dengan layanan ini, di harapkan mampu memudahkan badan pengawas (MPO) untuk menerima informasi program kerja secara *real time* sehingga penyampaian informasi kerja dari pengurus bisa lebih cepat dari sebelumnya. *Push notification* juga dapat diterapkan dalam sistem monitoring pintu air berbasis *android* menggunakan *firebase cloud messaging* yang memudahkan pengguna untuk mendapatkan informasi kondisi pintu air secara *real time* [2]. Implementasi *push notification* juga dapat diterapkan pada informasi perkuliahan dan kegiatan mahasiswa berbasis *android* yang dapat membantu mahasiswa untuk mendapatkan informasi secara *real time* seperti, pemberitahuan jadwal kuliah dan jadwal ujian [3][4].

Berdasarkan permasalahan di atas diperlukan sebuah Sistem Monitoring Kerja Kerukunan Mahasiswa Pinrang Universitas Muslim Indonesia (Kmp-Umi) menggunakan *Push Notification*. Dengan memanfaatkan layanan *push notification* maka Badan Pengawas KMP dapat mengetahui informasi program kerja yang telah dilakukan oleh pengurus KMP-UMI secara *real time* sehingga memperoleh informasi kerja secara efisien dan efektif.

II. Metode

A. *Push Notification*

Push notification memungkinkan sistem untuk dapat memberikan notifikasi kepada pengguna walaupun tidak sedang mengakses aplikasi berbasis tersebut [4]. Layanan *Push Notification* umumnya banyak diterapkan pada aplikasi *mobile* seperti *Android* dan *IOS*. Pemakai Sistem Operasi *Android* terbesar berdasarkan

statcounter periode 2012 sampai dengan 2016 di Indonesia [5]. *Push notification* dibagi menjadi dua pendekatan yaitu notifikasi lokal dan notifikasi secara jarak jauh (*remote*). Notifikasi jarak jauh ditangani oleh server jarak jauh. Aplikasi klien harus terdaftar di server dengan kunci unik. Server kemudian melepaskan pesan terhadap kunci unik untuk mengirimkan pesan ke aplikasi klien melalui protokol klien / server yang disepakati dan klien menampilkan pesan yang diterima. Saat *push notification* tiba, ia dapat mentransmisikan pesan pendek dan pemberitahuan, menetapkan badge pada ikon aplikasi atau memutar suara peringatan untuk menarik perhatian pengguna [6].

Tahapan penggunaan *push notification framework* pertama kali perlu mendaftar *push notification provider* atau *service provider*. Aplikasi akan memberitahu sistem operasi untuk meminta registrasi dengan tujuan untuk menerima notifikasi. *Provider* kemudian mengirimkan kembali token yang akan dikirimkan ke aplikasi. Aplikasi kemudian mengirimkan token ke perangkat mobile pengguna atau *third party server* untuk pengaturan penjadwalan notifikasi atau pengaturan pengguna [7-8].

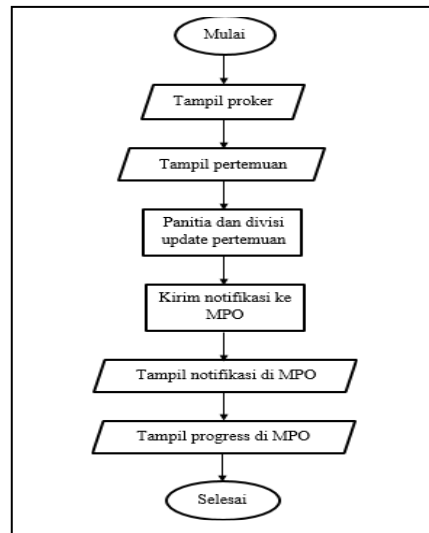
B. Android

Android adalah sistem operasi untuk telepon seluler yang berbasis Linux. *Android* menyediakan platform terbuka bagi para pengembang buat menciptakan aplikasi mereka sendiri untuk digunakan oleh bermacam peranti bergerak [9-10].

III. Hasil dan Pembahasan

A. Analisis dan Perancangan Sistem

Berdasarkan analisis permasalahan maka diperoleh sistem usulan yang akan memudahkan pengguna untuk mengetahui program kerja yang telah dikerjakan oleh pengurus KMP-UMI. Sistem usulan sebagai berikut:



Gambar 1. Sistem Usulan Monitoring Kerja

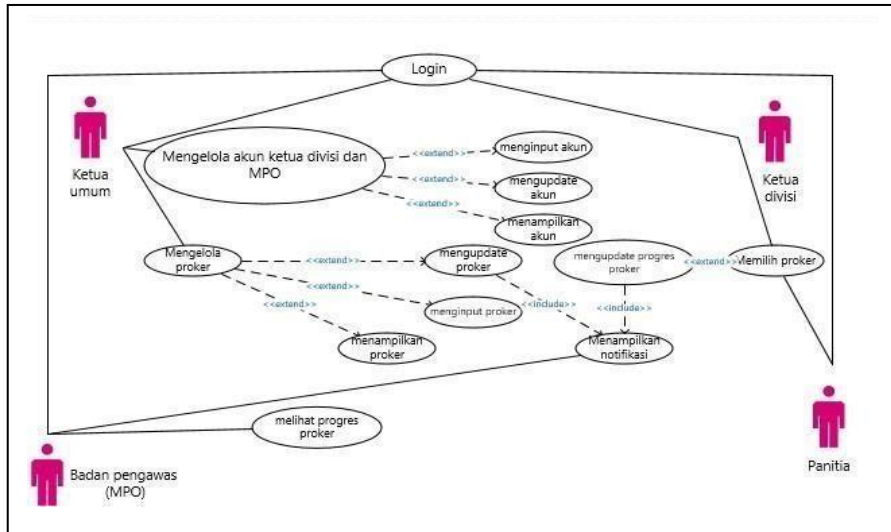
Pada gambar 1. Alur sistem monitoring usulan dengan tahapan sebagai berikut:

- Sistem menampilkan program kerja yang akan dilakukan oleh pengurus selama satu periode.
- Sistem menampilkan pertemuan dari program kerja yang akan dilakukan oleh pengurus selama satu periode.
- Pengurus melakukan pembaruan pertemuan ketika pertemuan tersebut telah selesai dijalankan.
- Sistem mengirimkan notifikasi kepada MPO ketika panitia dan divisi telah melakukan pembaruan pertemuan melalui layanan *push notification*.
- Sistem menampilkan notifikasi kepada MPO.
- Sistem menampilkan *progress/* kemajuan pelaksanaan program kerja di MPO.

Perancangan sistem dilakukan dengan membuat arsitektur aplikasi menggunakan *Unified Model Language* (UML) yaitu *Use Case Diagram*, *Activity diagram* dan *Class diagram*. *Use Case Diagram* menggambarkan hubungan interaksi antara sistem dan aktor yaitu Ketua Umum, Badan Pengawas, Ketua Divisi dan Panitia:

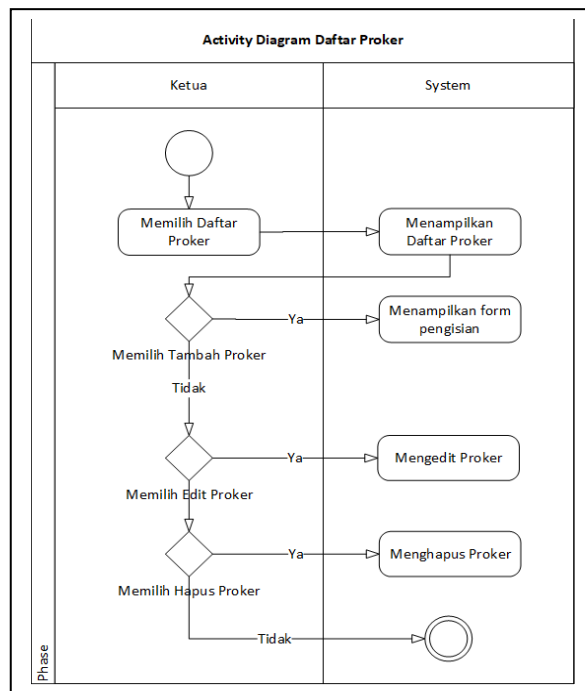
- Ketua umum melakukan login untuk masuk ke aplikasi.
- Ketua umum mengelola akun ketua divisi dan badan pengawas (MPO).
- Ketua umum melakukan *input*, *update* dan melihat akun dari ketua divisi dan badan pengawas (MPO).
- Ketua umum melakukan *input*, *update* dan melihat program kerja dari organisasi KMP – UMI.
- Ketua divisi melakukan login untuk masuk di aplikasi.
- Ketua divisi melakukan *update* kemajuan kerja sesuai dengan kemajuan kerja yang telah ditentukan.

- g) Panitia melakukan login untuk masuk ke aplikasi.
- h) Panitia melakukan *update* progress kerja sesuai *progress* kerja yang telah ditentukan.
- i) Badan pengawas (MPO) melakukan login untuk masuk ke aplikasi.
- j) Badan pengawas (MPO) melihat *update* proker yang telah di lakukan oleh ketua divisi dan panitia sekaligus menerima notifikasi informasi kerja secara *real time*.



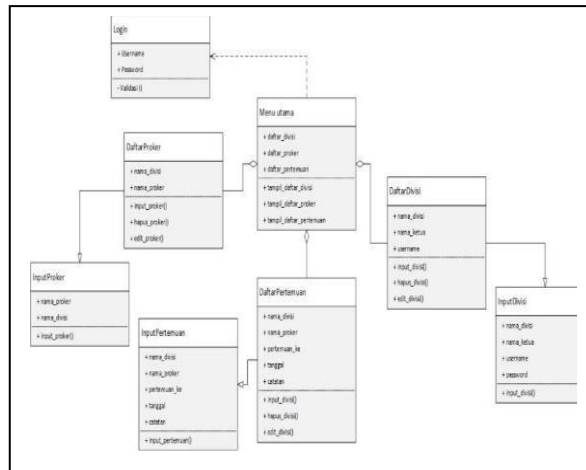
Gambar 2. Use Case Diagram Sistem Monitoring Kerja

Kelola data proker ditunjukkan dalam *Activity Diagram*. Ketua umum atau admin dapat mengelola data proker, dimana pengelolaan data proker berupa tambah proker, edit proker dan hapus proker. Data proker terdiri dari nama divisi, dan nama proker.



Gambar 3. Activity Diagram Kelola Data Proker

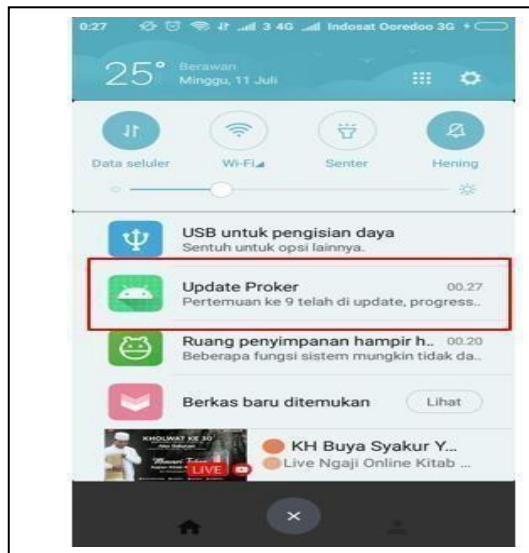
Class Diagram merupakan diagram yang menggambarkan struktur dan deskripsi kelas serta hubungan atau relasi dengan kelas lain pada aplikasi sistem monitoring.



Gambar 4. *Class Diagram* Sistem Monitoring Kerja

B. Implementasi *Push Notification*

Layanan *Push Notification* berupa pesan pemberitahuan secara singkat. Implementasi *Push Notification* pada sistem monitoring kerja, informasi program kerja (Proker) dan pelaksanaannya. Implementasi Push Notification sebagai berikut:

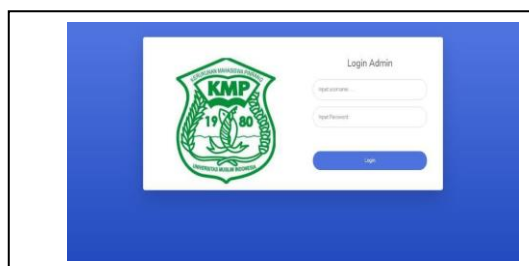


Gambar 5. Implementasi *Push Notification*

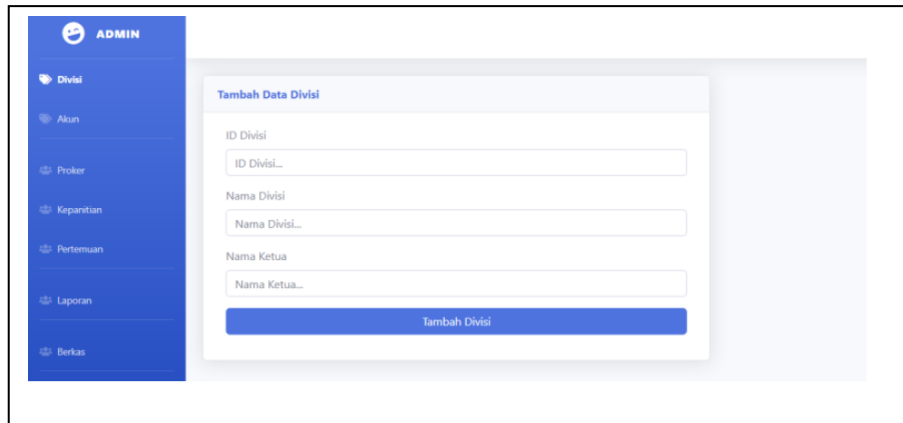
Pada gambar 5. menunjukkan notifikasi pesan terbaru yang diterima oleh Ketua Divisi atar program kerja yang telah terlaksana.

C. Implementasi Sistem

Sistem monitoring kerja terdiri dari dua bentuk aplikasi. Pada sisi klien dibangun aplikasi berbasis mobile sedangkan aplikasi berbasis web untuk pengelolaan data. Adapun hasil implementasi aplikasi berbasis web diperoleh beberapa fitur antara lain: login, kelola akun, divisi, kelola data proker, kelola pertemuan, kelola kegiatan (kepanitian), laporan dan berkas. Adapun hasil implementasi sebagai berikut :

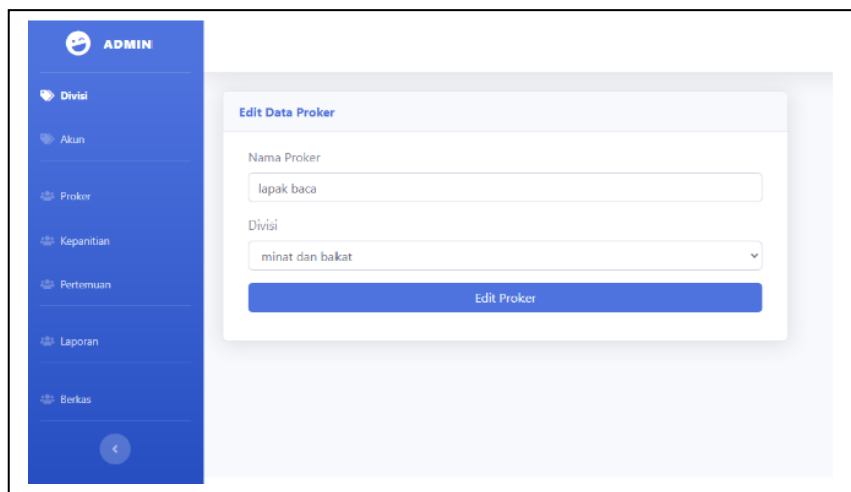


Gambar 6. Form Login Admin

The image shows a web interface for an administrator. On the left is a blue sidebar menu with the title 'ADMIN' and a list of items: 'Divisi', 'Akun', 'Proker', 'Kegiatan', 'Pertemuan', 'Laporan', and 'Berkas'. The main content area is titled 'Tambah Data Divisi' and contains three text input fields: 'ID Divisi', 'Nama Divisi', and 'Nama Ketua'. Below these fields is a blue button labeled 'Tambah Divisi'.

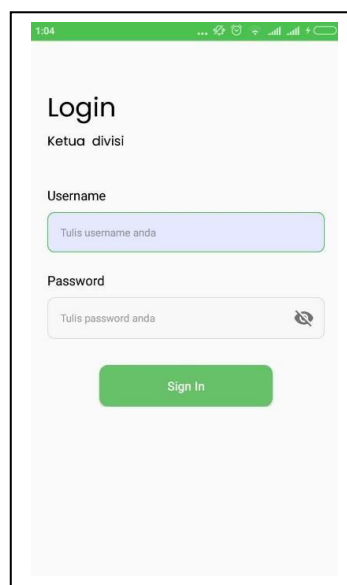
Gambar 7. Form Kelola Data Divisi

Kelola data divisi terdiri dari fitur tambah, edit dan hapus divisi. Sedangkan untuk kelola kegiatan sesuai proker sebagai berikut

The image shows the same administrator interface as Gambar 7. The sidebar menu is identical. The main content area is titled 'Edit Data Proker'. It features a text input field for 'Nama Proker' containing the text 'lapak baca', and a dropdown menu for 'Divisi' with 'minat dan bakat' selected. A blue button labeled 'Edit Proker' is positioned at the bottom of the form.

Gambar 8. Form Kelola Data Proker

Sedangkan hasil implementasi aplikasi berbasis android diperoleh beberapa fitur aplikasi untuk klien antar lain: login, dashboard: proker, pertemuan, kegiatan. Pengurus KMP dapat mengakses aplikasi *mobile* untuk memperbarui kegiatan yang sedang atau telah terlaksana.

The image shows a mobile application interface for a login screen. At the top, the status bar shows the time '1:04' and various icons. The screen has a light green header with the title 'Login' and the role 'Ketua divisi'. Below this, there are two input fields: 'Username' with the placeholder text 'Tulis username anda' and 'Password' with the placeholder text 'Tulis password anda' and a toggle icon for visibility. A green button labeled 'Sign In' is located at the bottom of the form.

Gambar 9. Login Pengurus Divisi KMP

Setelah berhasil login maka akan menampilkan dashboard. Dashboard tersebut digunakan untuk melihat persentase kegiatan dalam proker yang telah terlaksana. Masing ketua devisi memperbarui informasi kegiatan yang sedang atau telah terlaksana.



Gambar 10. Dashboard untuk Ketua Divisi

Badan Pengawas MPO dapat memantau hasil kegiatan yang telah dilaksanakan setiap divisi. Proker berisi berbagai kegiatan dan dapat dipantau melalui aplikasi tersebut. Misal Divisi minat dan bakat telah melaksanakan proker maka Pengurus Badan Mengawas MPO dapat melihat dashboard.



Gambar 11. Dashboard untuk Badan Pengawas MPO



Gambar 12. Dashboard untuk Badan Pengawas MPO

D. Pengujian Sistem

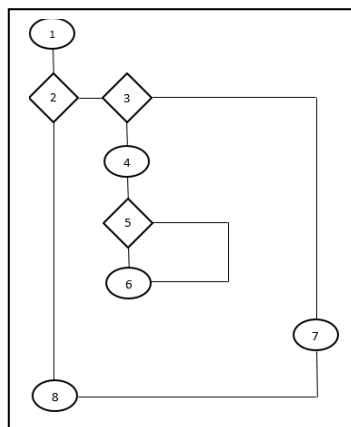
Pengujian aplikasi dilakukan dengan dua pendekatan yaitu whitebox testing dan blackbox testing. Adapun hasil pengujian *whitebox testing* sistem monitoring (*Android*) menggunakan *push notification* sebagai berikut:

Tabel 1. Pseudocode

```

- Begin
- Update pertemuan request
- If (Response.isSuccess) then
- if (Response.value == 1) then
- Get al dokumentasi()
- For each (Dokumentasi) begin
- Input dokumentasi request (id pertemuan, dokumentasi)
- end push notification()
- end for each
- Else progress.dismiss
- end if
- else progress.dismiss
- end if
- end
    
```

Berdasarkan pseudocode tersebut maka diperoleh *flow graph* sebagai berikut :



Gambar 13. FlowGraph Fungsi Sistem Monitoring

Hasil Perhitungan perhitungan *cyclomatic complexity* di atas didapatkan 3 independen path yaitu Path 1 : 1,2,8 , Path 2 : 1,2,3,7,8, dan Path 3 : 1,2,3,4,5,6,5. Untuk melakukan uji coba basis *path* di atas menggunakan *graph matriks*. Graph matriks adalah matriks yang mempunyai ukuran yang sama dengan node pada flow graph. Berdasarkan hasil perhitungan *cyclomatic complexity* dan *graph matriks* diperoleh hasil yang sama yaitu 3 sehingga dapat disimpulkan bahwa pengujian alur logika pada sistem yang dibangun telah optimal di sebabkan semua jalur *statement* logika pada unit sesuai dengan nilai kompleksitas basis path dan graph matriks yang telah terstruktur sesuai unit tersebut.

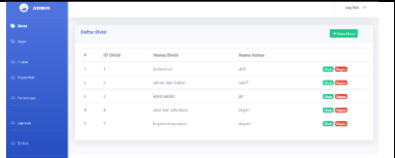
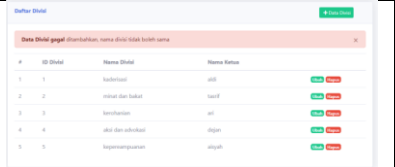
Pengujian penerimaan aplikasi dilakukan dengan dua tahap yaitu pengujian alpha dan pengujian beta. Beberapa Hasil pengujian alpha sebagai berikut :

Tabel 2. Pengujian Alpha Halaman Login Admin

Kasus dan Hasil Uji (Normal)	
Data masukan	Menginputkan Username dan password kemudian mengklik tombol login.
Yang diharapkan	Menampilkan halaman dashboard admin ketika mengklik tombol login
Kesimpulan	Hasil sesuai yang diharapkan
Gambar	
Kasus dan Hasil Uji (Tidak Normal)	

Data masukkan	Menginputkan Username dan password kemudian mengklik tombol login.
Yang diharapkan	Ketika username atau password salah maka tidak dapat masuk ke dashboard aplikasi dan menampilkan pesan kesalahan
Kesimpulan	Tida berhasil
Gambar	

Tabel 3. Pengujian Alpha Halaman Kelola Divisi

Kasus dan Hasil Uji (Normal)	
Data masukkan	Menginputkan Id divisi, nama ketua, dan nama divisi.
Yang diharapkan	Berhasil menambahkan data divisi yang baru
Kesimpulan	Hasil sesuai yang diharapkan
Gambar	
Kasus dan Hasil Uji (Tidak Normal)	
Data masukkan	Menginputkan Id divisi, nama ketua, dan nama divisi.
Yang diharapkan	Ketika ada data yang kosong atau data yang sama maka tidak bisa menambahkan data divisi yang baru
Kesimpulan	Tida berhasil
Gambar	

Berdasarkan seluruh hasil pengujian alpha terhadap seluruh fitur aplikasi ditarik kesimpulan bahwa aplikasi sudah berjalan dengan baik, akan tetapi tidak menutup kemungkinan dapat terjadi kesalahan suatu saat ketika aplikasi kembali digunakan. Maka untuk memastikan bahwa aplikasi tersebut dapat berjalan oleh pengguna maka dilakukan pengujian beta. Pengujian beta juga dikenal sebagai pengujian pengguna langsung di lokasi pengguna akhir. Pengguna akhir memvalidasi kegunaan, fungsi, kompatibilitas, dan uji reliabilitas dari aplikasi yang dibuat.

Berdasarkan pengujian beta yang dilakukan oleh pengguna akhir dengan menyebarkan kuesioner kepada 10 pengguna akhir menunjukkan bahwa persentase penerimaan aplikasi sebesar 53,2%. Penerimaan aplikasi ditinjau dari tampilan yang menarik, kemudahan penggunaan, efisiensi waktu penerimaan pesan terbaru, dan informatif.

IV. Kesimpulan dan saran

Aplikasi sistem monitoring kerja kerukunan mahasiswa pinrang universitas muslim Indonesia (KMP - UMI) yang diterapkan *push notification* memberikan kemudahan kepada pengurus dalam memperoleh informasi terbaru dengan tingkat penerimaan sebesar 53,2% dan berdasarkan pengujian *whilebox* menunjukkan alur logika pada sistem yang dibangun telah optimal.

Daftar Pustaka

- [1] J. G. Ariana, A. Pinandito, and M. T. Furqon, "Rancang Bangun Aplikasi Pengelolaan Bahan Baku Kafe Menggunakan Metode Analytical Hierarchy Process (AHP) dan Push Notification," *J-PTIHK, Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 2, no. 12, p. 7271-7277, 2018.

- [2] F. Fernando, A. Arini, and F. Fahrianto, "Push Notification Monitoring Sistem Pintu Air Berbasis Android Menggunakan Firebase Cloud Messaging," *Jurnal Teknik Informatika*, vol.13, no.1, pp. 79–88, 2020, <https://doi.org/10.15408/jti.v13i1.15884>
- [3] J. Setiawan, E. Kristianto, and F. Fredicia, "Implementasi Push Notification Pada Informasi Perkuliahan Dan Kegiatan Mahasiswa Berbasis Android," *Jurnal Teknik Dan Ilmu Komputer*, vol.4, no.14, pp. 211–219, 2015, [Online]. Available: <http://ejournal.ukrida.ac.id/ojs/index.php/TIK/article/view/1044>.
- [4] M. Muharir and N. Alamsyah, "Pemanfaatan Mobile Push Notification Dalam Penyampaian Informasi Perkuliahan Mahasiswa Pada Fakultas Teknologi Informasi Berbasis Android," *Technologia: Jurnal Ilmiah*, vol 11, no.2, pp.111-118, 2020, <https://doi.org/10.31602/tji.v11i2.2860>.
- [5] M. Siddik and A. Nasution, "Perancangan Aplikasi Push Notification Berbasis Android," *Jurteks*, vo.4, no.2, pp. 149–154, 2018, <https://doi.org/10.33330/jurteks.v4i2.56>.
- [6] L. Tommy, D. Wahyuningsih, and P. Romadiana, "Pengembangan Aplikasi Penerimaan Mahasiswa Baru Berbasis Android dengan Push Notification di STMIK Atma Luhur," *Jurnal Sisfokom (Sistem Informasi Dan Komputer)*, vol. 9, no. 1, pp. 108-121, 2020, <https://doi.org/10.32736/sisfokom.v9i1.813>.
- [7] V. Efendy, K. A. Nugraha, and D. Sebastian, "Implementasi Chat Room dan Push Notification pada e-Class Berbasis Mobile," *Jurnal Teknik Informatika Dan Sistem Informasi*, vol.5, no.2, pp. 267–282, 2019, <https://doi.org/10.28932/jutisi.v5i2.1763>
- [8] M. Imron, G. R. Sutikno, and I. N. Dazki, "Implementasi Push Notification Pada Sistem Peminjaman Sarana dan Prasarana Berbasis Website," *Jurnal Informatika*, vol.7, no. 2, pp. 174–182, 2020, <https://doi.org/10.31294/ji.v7i2.8694>.
- [9] N. A. Julita, H. Harlinda, and M. Hasnawi, "Rancang Bangun Sistem Pelaporan Kriminal Menggunakan Metode Location Based Services (LBS)," *Buletin Sistem Informasi dan Teknologi Islam*, vol.1, no.3, pp. 182-186, 2020, doi:<https://doi.org/10.33096/busiti.v1i3.591>.
- [10] M. A. Sudmar, T. Hasanuddin, and M. Hasnawi, "Rancang Bangun Aplikasi Pasar Online Di Kota Makassar Berbasis Mobile (Smart Market) Menggunakan Metode Location Based Service (LBS)," *Buletin Sistem Informasi dan Teknologi Islam*, vol.1, no.3, pp. 187-194, 2020, doi:<https://doi.org/10.33096/busiti.v1i3.605>