

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/339616775>

Implementasi Pola Arsitektur Model-View-ViewModel (MVVM) pada Sistem Informasi Akademik Universitas Darussalam Gontor Berbasis Mobile

Article · November 2019

CITATIONS

0

READS

1,732

3 authors:



Muhammad Syakir Arif

University of Darussalam Gontor

5 PUBLICATIONS 2 CITATIONS

[SEE PROFILE](#)



Aziz Musthafa

University of Darussalam Gontor

17 PUBLICATIONS 10 CITATIONS

[SEE PROFILE](#)



Dihin Muriyatmoko

University of Darussalam Gontor

40 PUBLICATIONS 29 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



sistem pakar [View project](#)



Implementation of Model-View-ViewModel (MVVM) Architecture Pattern in the Development of SIAKAD UNIDA Gontor Mobile Application [View project](#)

Implementasi Pola Arsitektur Model-View-ViewModel (MVVM) pada Sistem Informasi Akademik Universitas Darussalam Gontor Berbasis Mobile

Muhammad Syakir Arif¹, Aziz Musthafa², Dihin Muriyatmoko³

¹²³Program Studi Teknik Informatika, Fakultas Sains dan Teknologi, Universitas of Darussalam Gontor,

Jl. Raya Siman KM 5 Ponorogo 63471, Indonesia. Tel. +62-352-483762.

Email: muhammad.syakir@unida.gontor.ac.id¹, aziz@unida.gontor.ac.id², dihin@unida.gontor.ac.id³

Abstrak. Arsitektur perangkat lunak sangat cocok untuk mengatasi kompleksitas dan perubahan perangkat lunak yang kian meningkat. Model-View-ViewModel (MVVM) adalah salah satu dari pola arsitektur yang mengusung pemisahan kepentingan, yang memisahkan antara grafis antarmuka dan proses logika bisnis (*back-end logic*). Universitas Darussalam Gontor (UNIDA Gontor) memiliki sebuah Sistem Informasi Akademik (SIKAD) sebagai salah satu penunjang dalam manajemen administrasi data perkuliahan. Saat ini SIKAD UNIDA Gontor dirasa belum optimal, dikarenakan beberapa fitur didalamnya belum dapat diakses secara ramah pengguna (*user friendly*) terutama jika menggunakan perangkat ponsel. Maka dari itu, penelitian ini bertujuan untuk menerapkan metode MVVM pada SIKAD UNIDA Gontor dalam meningkatkan *user friendly*, khusus untuk pengguna ponsel Android. Penelitian ini dibuat dengan langkah – langkah pendekatan *waterfall*, dikembangkan menggunakan bahasa pemrograman Kotlin dan memanfaatkan Android Jetpack. Hasil uji aplikasi menggunakan metode Black Box menyatakan bahwa aplikasi berfungsi dengan lancar dan tidak terjadi *error*. Sedangkan hasil uji coba dengan penyebaran kuesioner kepada para pengguna (mahasiswa, dosen dan staf BAAK) menunjukkan bahwa aplikasi ini telah berjalan dengan baik dan memuaskan. Dari semua hasil uji coba menunjukkan bahwa penerapan MVVM pada SIKAD UNIDA Gontor berbasis mobile Android telah berhasil sesuai skenario dan siap diterapkan pada tahun ajaran 2019/2020 semester genap (<https://play.google.com/store/apps/details?id=com.amoled.sidago>). Pengembangan selanjutnya bisa diarahkan untuk penambahan fitur lain yang belum ada. Sebagai tambahan juga bisa dikembangkan untuk pengguna ponsel selain bersistem operasi Android, misalnya iOS, Tizen, hongmeng dan lain sebagainya.

Kata Kunci: Android Jetpack, Kotlin, MVVM, Sistem Informasi Akademik, UNIDA Gontor

Running title: Implementasi MVVM dalam Aplikasi SIKAD

PENDAHULUAN

Ponsel pintar merupakan media yang sangat canggih dalam akses informasi dan layanan data, hal ini memungkinkan semua bidang kehidupan manusia dapat semakin ringan dikerjakan dengan bantuan telepon pintar (Nuari, 2014). Oleh karenanya, menurut data statistik yang dilansir oleh Katadata menyatakan bahwa pada tahun 2019, pengguna ponsel pintar di Indonesia mencapai 92 juta pengguna (Katadata, 2016).

Android, sebagai salah satu sistem operasi ponsel pintar yang bersifat *open source* selalu meraih pencapaian yang semakin besar di setiap tahun. Menurut CEO Google, Sundar Pichai pada acara konferensi Google I/O 2017, ia menyatakan bahwa saat itu Android telah mencapai 2 miliar pengguna aktif bulanan di seluruh dunia (Popper, 2017). Sementara di Indonesia, hasil statistik menyatakan bahwa saham untuk sistem operasi *mobile* Android juga semakin meningkat setiap bulan, tercatat mencapai sebesar 88,37% di bulan Desember 2017 (“Market share of mobile operating systems in Indonesia from January 2012 to December 2017,” 2018).

Sistem perangkat lunak menjadi sangat rumit dan canggih untuk memenuhi tuntutan bisnis yang lebih baru. Arsitektur perangkat lunak (*software architecture*) sangat cocok untuk mengatasi kompleksitas dan perubahan perangkat lunak yang kian meningkat (Raj, Raman, & Subramanian, 2017). Pola arsitektur (*architecture pattern*) adalah pola yang terkenal sebagai solusi untuk menyelesaikan permasalahan arsitektur perangkat lunak. Pola arsitektur sebuah perangkat lunak adalah ‘organisasi’ kode secara keseluruhan (Tiari, 2015).

Model-View-ViewModel (MVVM) adalah salah satu dari pola arsitektur (*architecture pattern*) yang mengusung pemisahan kepentingan, yang mana hal tersebut memungkinkan untuk memisahkan antara grafis antarmuka (*graphical user interface*) dari proses logika bisnis atau logika halaman belakang (*back-end logic*) (Wikipedia contributors, n.d.) (Saleh, 2017).

Universitas Darussalam Gontor (UNIDA Gontor) adalah salah satu Perguruan Tinggi di Indonesia yang berlokasi di Kabupaten Ponorogo, Jawa Timur. Sebagai salah satu penunjang dalam manajemen administrasi data perkuliahan, UNIDA Gontor memiliki sebuah Sistem Akademik (SIKAD) yang dikelola oleh tim Pusat Pelayanan Teknologi Informasi dan Komunikasi (PPTIK). Saat ini SIKAD belum dapat diakses secara ramah pengguna (*user friendly*) jika

menggunakan perangkat ponsel (*mobile*). Berdasarkan kendala tersebut, hingga saat ini penggunaan SIAKAD dinilai belum optimal.

Penelitian ini bertujuan untuk mengimplementasikan pola arsitektur MVVM dalam rancang bangun aplikasi SIAKAD UNIDA Gontor. Implementasi dilakukan dengan menggunakan bahasa pemrograman Kotlin serta memanfaatkan Android Jetpack sebagai Komponen Arsitektur Android (Android Architecture Components). Aplikasi tersebut diharapkan akan memberikan pelayanan SIAKAD yang aman, cepat dan mudah sehingga produk dari penelitian ini selaras dengan Maqashid Syari’ah.

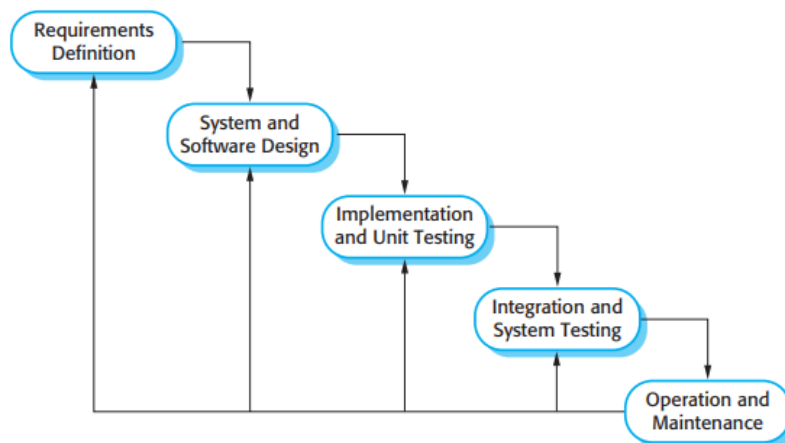
BAHAN DAN METODE

Bahan

Lingkungan pengembangan terpadu (*Integrated development environtment*) yang dipakai adalah Android Studio IDE, dengan bahasa pemrograman Kotlin untuk menangani logika *back-end* dan XML untuk tampilan *layout front-end*. Kemudian sumber data yang dipakai adalah bersumber dari *Application Programming Interface* (API) milik UNIDA Gontor, sebagai sumber utama pengambilan data. Kemudian untuk sumber data lokal menggunakan Room, yaitu sebuah komponen arsitektur Android untuk basis data aplikasi Android yang berbasis SQLite. Sumber data lokal berfungsi sebagai penampung sementara data-data yang telah diambil dari API. Implementasi pola arsitektur MVVM akan dilakukan dengan memanfaatkan Android Jetpack yang memiliki Komponen Arsitektur Android yaitu: Data Binding, LiveData, ViewModel, Room dan Navigation.

Metode

Peneliti menggunakan metode perancangan Air Terjun (*Waterfall*), yang terdiri dari lima tahap sebagaimana dijelaskan sebagai berikut (Sommerville, 2011):



Gambar 1. Tahapan model Air Terjun (Waterfall)

Requirements Definition

Pengguna aplikasi terdiri dari kategori Mahasiswa, Dosen dan Umum. Oleh karena itu, setiap kategori memiliki kebutuhan sistem masing-masing. Kebutuhan setiap kategori pengguna dijabarkan dalam beberapa tabel berikut ini:

Tabel 1. Kebutuhan Sistem kategori pengguna Umum

No	Kebutuhan Sistem
1	Pengguna dapat melihat berita kampus
2	Pengguna dapat melihat kampus: profil, fasilitas, dll

Tabel 2. Kebutuhan sistem kategori pengguna Mahasiswa

No	Kebutuhan Sistem
1	Mahasiswa dapat melakukan <i>login</i> ke dalam aplikasi
2	Mahasiswa dapat membuat Kartu Rencana Studi (KRS)
3	Mahasiswa dapat melakukan konfirmasi pembayaran
4	Mahasiswa dapat melihat daftar mata kuliah

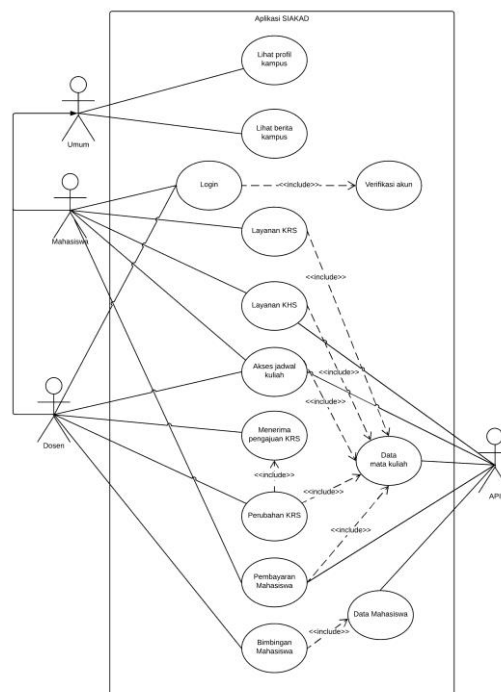
- 5 Mahasiswa dapat melihat jadwal mata kuliah
- 6 Mahasiswa dapat melihat Kartu Hasil Studi (KHS)
- 7 Mahasiswa dapat melihat tagihan administrasi

Tabel 3. Kebutuhan Sistem kategori pengguna Dosen

No	Kebutuhan Sistem
1	Dosen dapat melakukan <i>login</i> ke dalam aplikasi
2	Dosen Pembimbing Akademik (PA) dapat melihat daftar nama mahasiswa bimbingan
3	Dosen PA dapat menerima laporan KRS mahasiswa
4	Dosen PA dapat menyetujui laporan KRS mahasiswa
5	Dosen PA dapat memberikan izin perubahan KRS mahasiswa
6	Dosen dapat melihat jadwal kuliah
7	Dosen dapat melihat nilai mahasiswa bimbingan

System and Software Design

Berikut adalah Use Case Diagram untuk fitur-fitur dalam aplikasi SIAKAD UNIDA Gontor:



Gambar 2. Use Case Diagram aplikasi SIAKAD UNIDA Gontor

Berdasarkan Gambar 3, dapat diketahui bahwa pengguna Mahasiswa dan Dosen adalah merupakan turunan dari pengguna Umum, yang mana semua pengguna Umum dapat mengakses beberapa fitur umum seperti menu berita dan profil kampus. Kemudian untuk fitur yang dapat diakses setiap kategori pengguna telah ditampilkan dalam gambar diagram diatas, sesuai dengan kategori masing-masing.

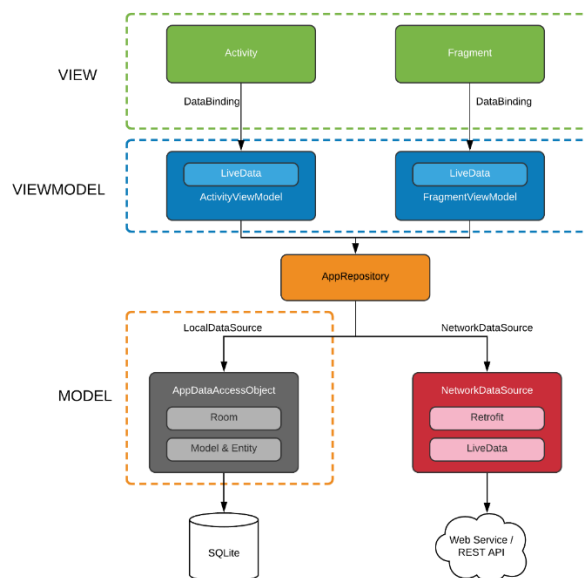
Selanjutnya penjelasan dari setiap use case yang ditampilkan dalam Gambar 3 adalah sebagai berikut:

Tabel 4. Identifikasi Use Case

No	Nama Use Case	Deskripsi	Aktor
1	Lihat profil kampus	Aktor dapat melihat informasi profil kampus UNIDA Gontor	Umum, Mahasiswa, Dosen
2	Lihat berita kampus	Aktor dapat melihat berita kampus UNIDA Gontor	Umum, Mahasiswa, Dosen
3	Login	Aktor dapat masuk ke dalam akun SIAKAD dengan menggunakan akun yang ditentukan	Mahasiswa, Dosen

4	Verifikasi akun	Proses verifikasi akun agar aktor dapat masuk ke dalam akun SIAKAD	Mahasiswa, Dosen
5	Layanan KRS	Aktor dapat mengakses fitur layanan KRS	Mahasiswa, API
6	Layanan KHS	Aktor dapat mengakses fitur layanan KHS	Mahasiswa, API
7	Daftar mata kuliah	Proses pengambilan data dan daftar mata kuliah	API
8	Akses jadwal kuliah	Proses pengambilan data jadwal perkuliahan	Mahasiswa, Dosen, API
9	Menerima pengajuan KRS	Dosen dapat menerima daftar pengajuan KRS dari mahasiswa bimbingannya.	Mahasiswa, Dosen
10	Perubahan KRS	Dosen dan Staf dapat memberikan izin perubahan data KRS	Mahasiswa, Dosen
11	Pembayaran Mahasiswa	Aktor dapat melihat daftar tagihan pembayaran/administrasi	Mahasiswa
12	Bimbingan Mahasiswa	Aktor dapat memantau mahasiswa bimbingannya	Dosen

Dikarenakan penelitian ini akan menerapkan pola arsitektur Model-View-ViewModel, maka berikut adalah rancangan arsitektur yang akan diterapkan dalam aplikasi SIAKAD UNIDA Gontor:



Gambar 3. Rancangan arsitektur aplikasi mengimplementasikan MVVM

Pada Gambar 3 diatas dijelaskan bahwa kategori View mencakup Activity dan Fragment. Kemudian untuk kategori ViewModel, mencakup kelas ActivityViewModel dan FragmentViewModel. Kedua kelas tersebut mengimplementasikan (menurunkan/inherit) pustaka LiveData yang berfungsi untuk menangani penyajian data yang akan ditampilkan. Komponen View dan ViewModel dihubungkan oleh pustaka DataBinding.

Selanjutnya terdapat kelas AppRepository yang berfungsi menangani koneksi kepada database yang dipakai aplikasi, baik bersumber dari local data source maupun network data source.

Selanjutnya untuk kategori Model, mencakup kelas-kelas yang akan menangani manajemen data dalam local data source yang terdiri dari pustaka Room dan kelas Model (banyak kelas Model disesuaikan dengan data yang akan diolah). Local data source menggunakan penyimpanan SQLite.

Selanjutnya terdapat kelas NetworkDataSource yang menangani pengambilan atau *fetching* data dari API atau dari basis data lokal.

Implementation and Unit Testing

Untuk tahap Implementation, dilakukan implementasi pola arsitektur MVVM saat Penulis memprogram aplikasi SIAKAD UNIDA Gontor. Kemudian untuk Unit Testing, Penulis melakukan uji coba dengan menggunakan metode Black Box.

Integration and System Testing

Untuk tahap Integration and System Testing, Penulis melakukan uji coba aplikasi bersama perwakilan responden dari Staf BAAK, Dosen dan Mahasiswa. Hasil uji coba tersebut dievaluasi dengan memanfaatkan data kuesioner dari para responden.

Operation and Maintenance

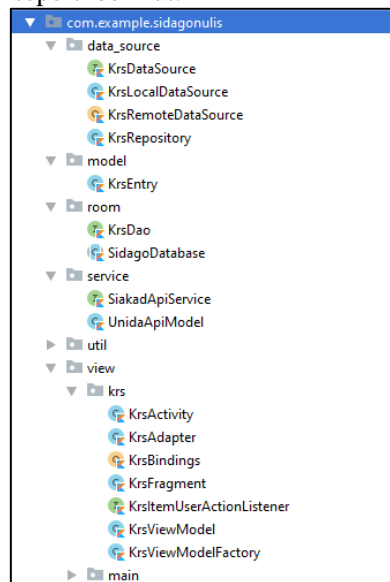
Pada tahap ini dilakukan perbaikan aplikasi berdasarkan timbal balik dan evaluasi dari pengguna atau pembimbing, kemudian memperbaruinya. Evaluasi pengguna dapat diambil berdasarkan hasil uji coba serta penilaian yang diberikan di Google Play Store (jika aplikasi telah dipublikasikan).

HASIL DAN KESIMPULAN

Pada bab ini membahas metode penelitian untuk tahap Implementation and Unit Testing (tahap ketiga Waterfall), Integration and System Testing (tahap keempat Waterfall), serta Operation and Maintenance (tahap kelima Waterfall). Mengenai hasil dan pembahasan, penulis akan berfokus hanya membahas hasil penerapan MVVM pada fitur KRS saja, dikarenakan untuk mempersingkat pembahasan. Seluruh fitur dalam aplikasi telah menerapkan MVVM yang konsepnya sama seperti fitur KRS.

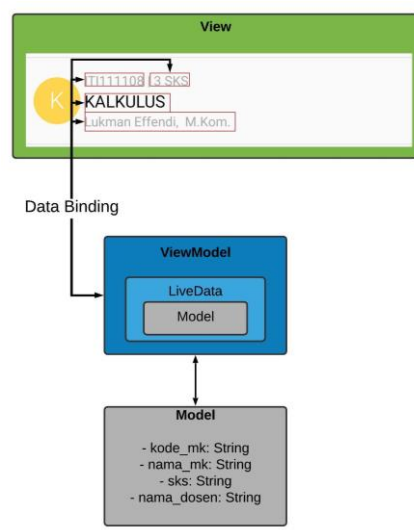
Implementation and Unit Testing

Demi kerapian struktur *package* serta mempermudah dalam peletakkan kelas-kelas, penulis melakukan pengelompokkan kelas-kelas dalam proyek seperti berikut:



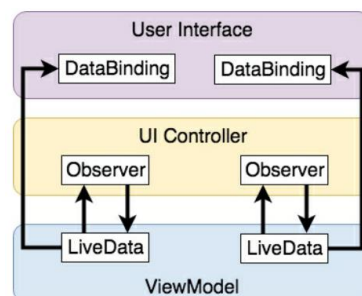
Gambar 4. Hirarki kelas dalam project untuk fitur KRS

Pada Gambar 4 menampilkan kelas-kelas yang dipakai untuk menangani fitur KRS dengan pola arsitektur MVVM. Sementara itu, berikut adalah ilustrasi menampilkan Model KRS ke View melalui perantara ViewModel:



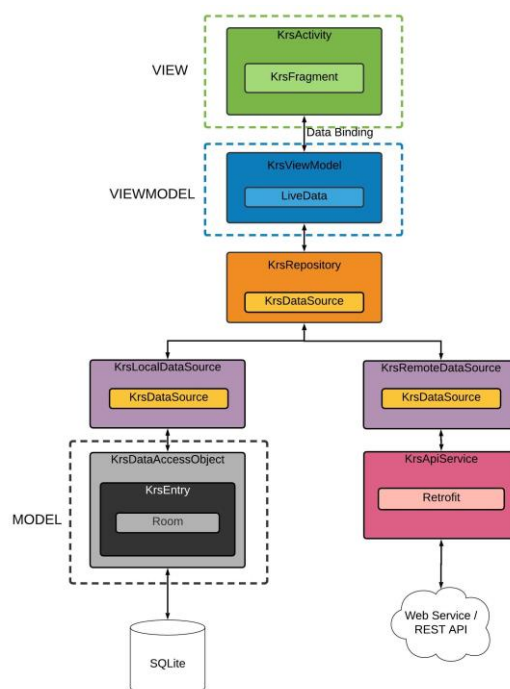
Gambar 5. Ilustrasi menampilkan Model KRS ke View melalui ViewModel

Dalam Gambar 5 ditampilkan bahwa Model KRS yang memuat data `kode_mk`, `nama_mk`, `sks`, dan `nama_dosen` ditampilkan kepada View dengan perantara LiveData dalam ViewModel. Penggunaan Data Binding sangat berperan dalam proses ini. Data Binding dapat secara langsung menanggapi perubahan data yang terjadi di LiveData dalam ViewModel, kemudian menampilkannya dalam User Interface (View). Berikut adalah ilustrasinya:



Gambar 6. Ilustrasi hubungan antara LiveData dalam ViewModel dengan Data Binding

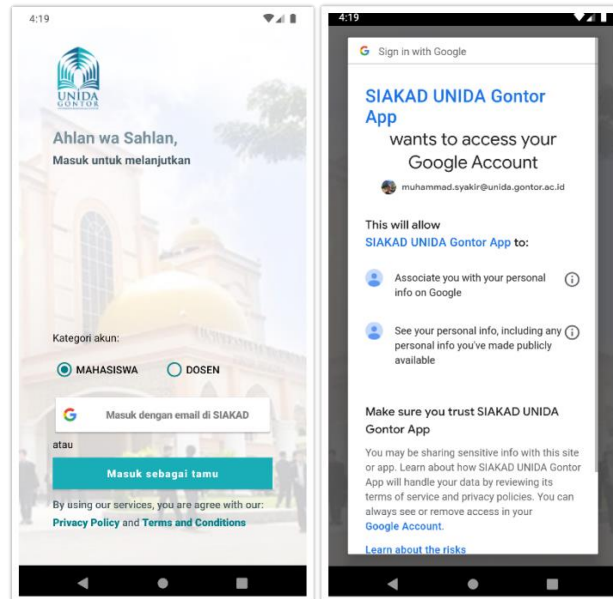
Kemudian, untuk hasil implementasi pola arsitektur MVVM disajikan ke dalam diagram ilustrasi berikut ini:



Gambar 7. Ilustrasi arsitektur MVVM yang berhasil diterapkan, salah satunya dalam fitur KRS mahasiswa

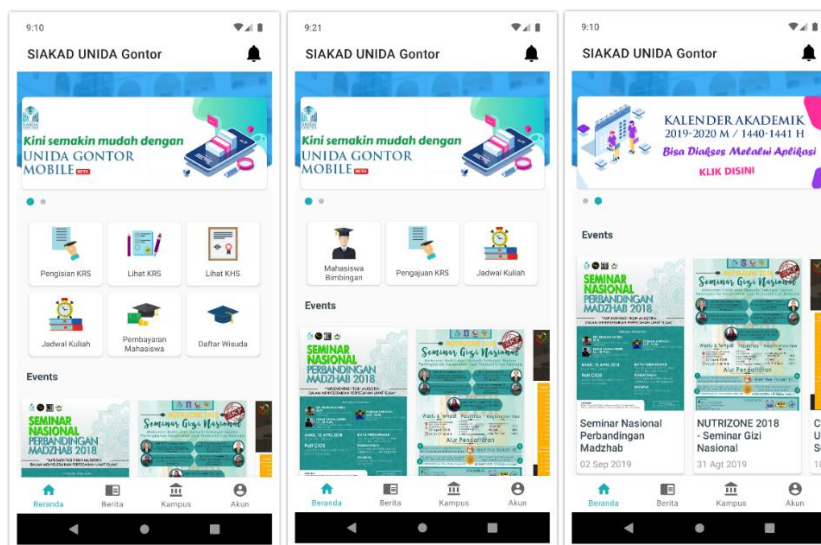
Dari Gambar 7 dapat dijelaskan bahwa untuk menerapkan MVVM dalam fitur KRS, dibutuhkan hirarki antar kelas-kelas seperti tergambar. Untuk komponen View terdiri dari `KrsActivity` dan `KrsFragment`. Lalu untuk komponen ViewModel, diberi nama `KrsViewModel`. Selanjutnya, Data Binding berperan dalam penyaluran data antara komponen View dan ViewModel. Selanjutnya kelas Repository bertugas mengelola dua sumber basis data dalam aplikasi, yaitu `LocalDataSource` dan `RemoteDataSource`. `RemoteDataSource` memiliki tugas untuk mengambil data KRS dari API. Sementara itu, `LocalDataSource` bertugas untuk menyimpan data dari API ke SQLite Database dan mengirimkan data tersebut jika sewaktu-waktu di minta oleh kelas Repository. Hal ini akan berguna ketika Pengguna menggunakan aplikasi, tapi tidak terkoneksi dengan internet, maka data KRS yang ditampilkan berasal dari basis data yang disimpan secara lokal.

Kemudian untuk hasil Unit Testing, penulis menggunakan metode Black Box dan menyatakan bahwa seluruh fitur aplikasi berfungsi baik dan tidak terjadi *error* atau *force close*. Uji coba Unit Testing dilakukan menggunakan versi Android terbaru, yaitu Android 10 (Beta). Berikut adalah beberapa contoh tampilan aplikasi SIAKAD UNIDA Gontor:



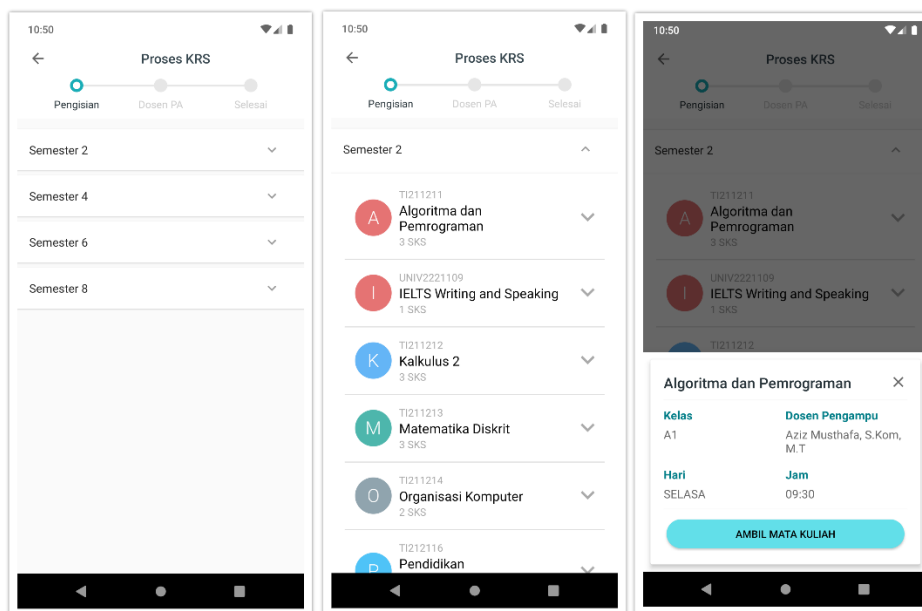
Gambar 8. Halaman masuk menggunakan *email* UNIDA Gontor

Sesuai yang tertera pada Gambar 8, pengguna akan memilih kategori akun terlebih dahulu. Kemudian setelah itu, menekan tombol *Masuk dengan email UNIDA Gontor* lalu pengguna akan diminta mengisi data email beserta kata sandi. Proses masuk akun UNIDA Gontor tidak ditangani oleh aplikasi SIAKAD UNIDA Gontor, melainkan ditangani oleh Google Services. Layanan autentikasi akun Google dapat terintegrasi dengan aplikasi SIAKAD UNIDA Gontor dengan baik.



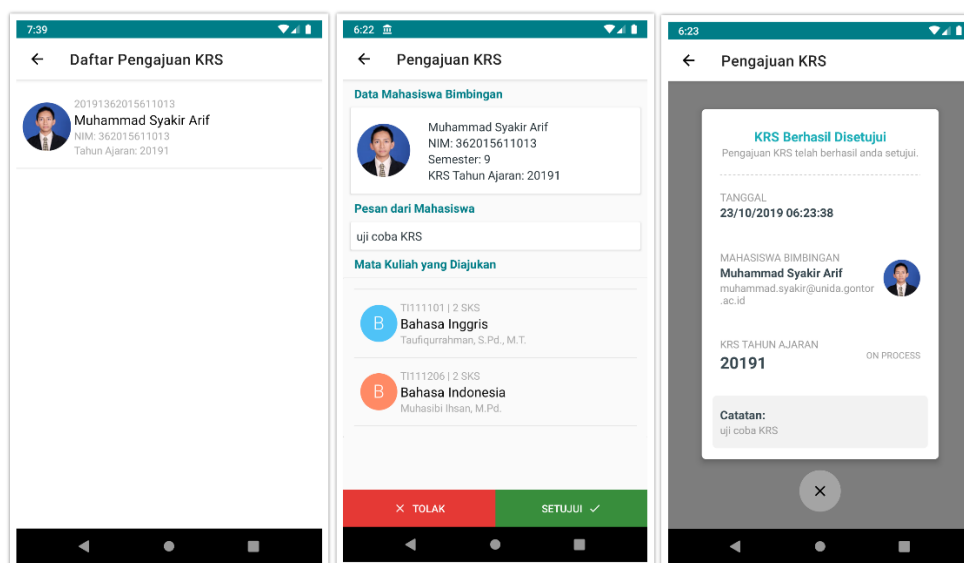
Gambar 9. Perbandingan tampilan antarmuka Beranda setiap kategori Mahasiswa, Dosen dan Umum

Dalam Gambar 9, ditampilkan bahwa fitur Beranda untuk kategori mahasiswa memuat menu Pengajuan KRS, Kartu Rencana Studi, Kartu Hasil Studi, Jadwal Kuliah, Daftar Wisuda serta Pembayaran Mahasiswa. Sementara Beranda untuk kategori Dosen, memuat menu Jadwal Kuliah, Absensi Mata Kuliah, Nilai Mahasiswa serta Bimbingan Mahasiswa. Terakhir, tidak terdapat menu apapun untuk kategori Umum.



Gambar 10. Proses Mengisi KRS oleh Mahasiswa

Dalam Gambar 10 ditampilkan proses pengisian KRS oleh Mahasiswa. Gambar sebelah kiri menampilkan menu *dropdown* tiap semester. Kemudian gambar tengah menampilkan daftar mata kuliah pada semester terkait. Kemudian pada gambar sebelah kanan menampilkan detail mata kuliah yang dipilih.



Gambar 10. Tampilan antarmuka proses Konfirmasi Pengajuan KRS oleh Dosen

Pada Gambar 10, ditampilkan proses Konfirmasi Pengajuan KRS oleh Dosen Pembimbing Mahasiswa terkait. Pada gambar sebelah kiri, Dosen melihat daftar pengajuan KRS dari mahasiswa bimbingannya. Kemudian pada gambar tengah ditampilkan detail KRS dari mahasiswa tersebut, untuk kemudian disetujui atau ditolak pengajuannya. Kemudian pada gambar sebelah kiri menampilkan keterangan KRS berhasil disetujui atau ditolak.

Integration and System Testing

Untuk tahap ini, Penulis melakukan uji coba aplikasi kepada pengguna. Pengguna tersebut adalah para perwakilan dari Staf BAAK, Dosen dan Mahasiswa UNIDA Gontor yang bersedia untuk menjadi responden penelitian.

Kategori Kepuasan Hasil Uji Coba Aplikasi

Sebagai tolak ukur kepuasan pengguna setelah dilakukan uji coba, maka dibuatlah kategori kepuasan hasil uji coba aplikasi sebagai berikut:

Tabel 5. Kategori Presentase Kepuasan Uji Coba

NO	Persentase	Kategori
1	$75,01 < x \leq 100$	Sangat Puas
2	$58,34 < x \leq 75,01$	Puas
3	$41,66 < x \leq 58,34$	Normal
4	$24,99 < x \leq 41,66$	Tidak Puas
5	$0 < x \leq 24,99$	Sangat Tidak Puas

Uji Coba Aplikasi oleh Staf BAAK

Pengujian ini bertujuan untuk memastikan bahwa integrasi sistem dalam aplikasi SIAKAD UNIDA Gontor sesuai dengan *Standard Operational Prochedure* (SOP) yang ditetapkan oleh Biro Administrasi Akademik dan Kemahasiswaan (BAAK). Pengujian ini dilakukan oleh Wakil Kepala BAAK UNIDA Gontor yang bernama Al-Ustadz Samsirin, M.Pd.I. Untuk fitur yang diujicobakan adalah Fitur KRS dan Fitur KHS. Dari uji coba tersebut, didapatkan hasil kuesioner sebagai berikut:

Tabel 6. Hasil Kuesioner Uji Coba dari Wakil Kepala BAAK

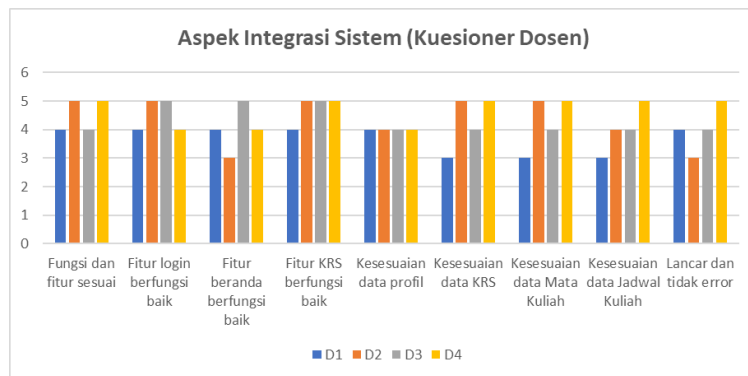
NO	Indikator	Pertanyaan	Nilai
1	Aspek Integrasi Sistem	Fungsi dan fitur aplikasi sesuai dengan tujuan aplikasi	5
2		Alur proses pengajuan KRS dalam aplikasi sudah sesuai dengan SOP yang ditetapkan BAAK UNIDA Gontor	4
3		Alur proses lihat KHS dalam aplikasi sudah sesuai dengan SOP yang ditetapkan BAAK UNIDA Gontor	4
4		Data Mata Kuliah yang disajikan sesuai dengan data dalam SIAKAD	5
5		Data Jadwal Kuliah yang disajikan sesuai dengan data dalam SIAKAD	5
6		Aplikasi berjalan lancar dan tidak terjadi <i>error</i>	4
7	Aspek Interaksi Antarmuka	Tampilan grafis antarmuka aplikasi mudah untuk dikenali	3
8		Tampilan grafis antarmuka aplikasi mudah untuk diingat	3
9		Tulisan dalam aplikasi mudah dibaca	3
10		Aplikasi mudah dioperasikan/digunakan	4
11		Tampilan warna aplikasi nyaman dilihat dan tidak membosankan	3
12		Mudahnya mengakses layanan SIAKAD melalui aplikasi	4
Rata-Rata			3,9166667

Catatan: Indikator nilai yang ditetapkan dalam kuesioner adalah yaitu 5 = Sangat Setuju, 4 = Setuju, 3 = Normal, 2 = Tidak Setuju, 1 = Sangat Tidak Setuju.

Rata-rata nilai yang didapatkan dari hasil uji coba diatas adalah sebesar 3,91 (normal). Persentase kepuasan dari hasil uji coba bersama Staf BAAK tersebut mendapatkan nilai 78,33% (Sangat Puas).

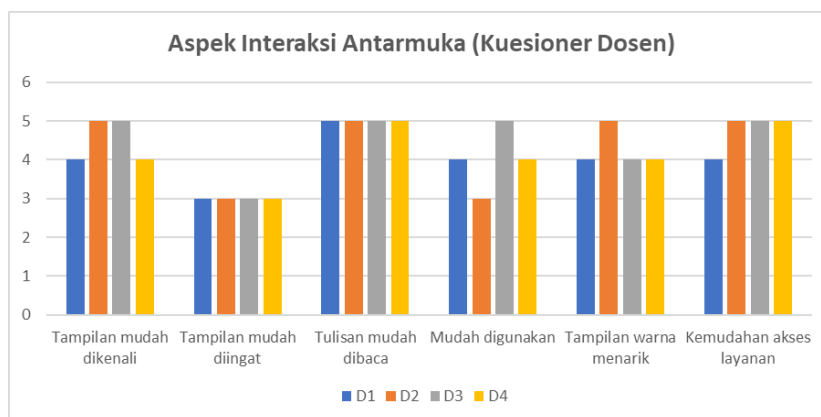
Uji Coba Aplikasi oleh Dosen

Pengujian dilakukan bersama perwakilan responden dari dosen Teknik Informatika UNIDA Gontor. Indikator nilai yang ditetapkan dalam kuesioner adalah yaitu 5 = Sangat Setuju, 4 = Setuju, 3 = Normal, 2 = Tidak Setuju, 1 = Sangat Tidak Setuju. Berikut adalah diagram penilaian dari uji coba tersebut:



Gambar 11. Hasil Kuesioner Dosen untuk Aspek Integrasi Sistem

Dalam Gambar 12 diatas, dapat diketahui bagaimana penilaian dari setiap poin-poin yang terdapat dalam Aspek Integrasi Sistem. Secara keseluruhan, penilaian untuk Aspek Integrasi Sistem ini mendapatkan nilai rata-rata 4,25 (setuju/baik).



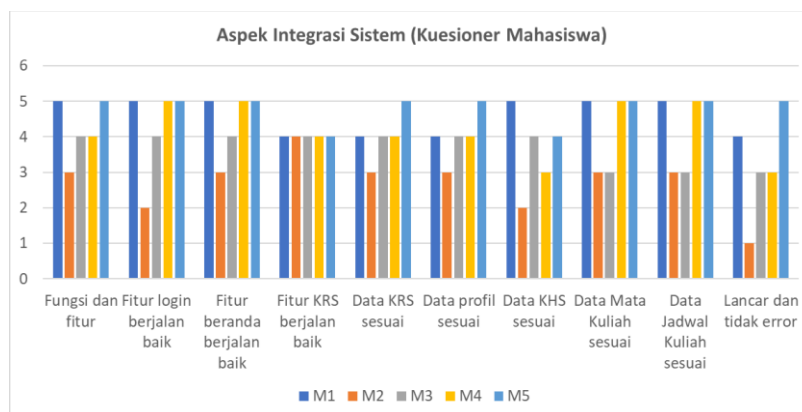
Gambar 12. Hasil Kuesioner Dosen untuk Aspek Interaksi Antarmuka

Dalam Gambar 13 diatas, dapat diketahui bagaimana penilaian dari setiap poin-poin yang terdapat dalam Aspek Interaksi Antarmuka. Secara keseluruhan, penilaian untuk Aspek Interaksi Antarmuka ini mendapatkan nilai rata-rata 4,25 (setuju/baik).

Secara keseluruhan, nilai rata-rata yang didapatkan dari uji coba aplikasi oleh Dosen adalah sebesar 4,25 (setuju/baik). Adapun presentase kepuasannya adalah 85% (Sangat Puas)

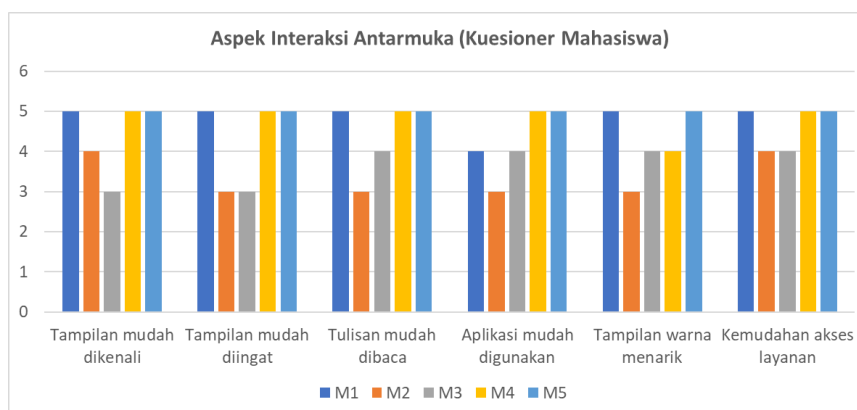
Uji Coba Aplikasi oleh Mahasiswa

Pengujian dilakukan bersama perwakilan responden dari mahasiswa Teknik Informatika UNIDA Gontor, sebanyak 5 mahasiswa. Indikator nilai yang ditetapkan dalam kuesioner adalah yaitu 5 = Sangat Setuju, 4 = Setuju, 3 = Normal, 2 = Tidak Setuju, 1 = Sangat Tidak Setuju. Berikut adalah diagram penilaian dari uji coba tersebut:



Gambar 13. Hasil Kuesioner Mahasiswa untuk Aspek Integrasi Sistem

Dalam Gambar 14 diatas, dapat diketahui bagaimana penilaian dari setiap poin-poin yang terdapat dalam Aspek Integrasi Sistem yang dilakukan oleh Mahasiswa. Secara keseluruhan, penilaian untuk Aspek Integrasi Sistem oleh Mahasiswa ini mendapatkan nilai rata-rata 4 (setuju/baik).



Gambar 14. Hasil Kuesioner Mahasiswa untuk Aspek Interaksi Antarmuka

Dalam Gambar 15 diatas, dapat diketahui bagaimana penilaian dari setiap poin-poin yang terdapat dalam Aspek Interaksi Antarmuka oleh Mahasiswa. Secara keseluruhan, penilaian untuk Aspek Interaksi Antarmuka oleh Mahasiswa ini mendapatkan nilai rata-rata 4,3 (setuju/baik).

Secara keseluruhan, rata-rata nilai yang didapatkan dari uji coba aplikasi oleh Mahasiswa adalah sebesar 4,125 (setuju/baik). Adapun nilai persentasenya mencapai 82,5% (Sangat Puas).

Operation and Maintenance

Perbaikan dalam Aspek Integrasi Sistem

Tabel 7. Perbaikan dalam Aspek Integrasi Sistem

No	Saran/Perbaikan	Status
1	Agar menonaktifkan <i>auto login</i> karena terdapat riwayat pengguna, saat masuk akun menggunakan surel	Selesai
2	Memperbaiki duplikasi data pada fitur KHS dan KRS	Selesai

Perbaikan dalam Aspek Interaksi Antarmuka

Table 8. Perbaikan dalam Aspek Interaksi Antarmuka

No	Saran/Perbaikan	Status
1	Tampilan User ID tidak perlu ditampilkan	Selesai
2	“NIY” agar diganti menjadi “Kode Dosen”	Selesai
3	EditText agar dibuat non-editable	Selesai
4	Untuk tampilan KRS masih terdapat duplikasi pada mata kuliah tertentu	Selesai
5	Berikan dialog konfirmasi saat Dosen menyetujui/menolak KRS	Selesai

Kesimpulan

Penerapan MVVM pada SIAKAD UNIDA Gontor untuk pengguna mobile Android dalam meningkatkan *user friendly* telah berhasil dengan baik. Hasil uji Black Box menunjukkan bahwa aplikasi ini berjalan lancar dan tanpa *error*. Hasil uji kuisisioner menunjukkan bahwa penerapan MVVM telah sesuai skenario dan mendapat respon positif dari mahasiswa, dosen dan staf BAAK UNIDA Gontor. Aplikasi *mobile* Android ini siap diterapkan mulai tahun ajaran 2019/2020 (<https://play.google.com/store/apps/details?id=com.amoled.sidago>) hingga seterusnya. Pengembangan selanjutnya bisa diarahkan untuk penambahan fitur lain yang belum ada. Sebagai tambahan juga bisa dikembangkan untuk pengguna ponsel selain bersistem operasi Android, misalnya iOS, Tizen, hongmeng dan lain sebagainya.

UCAPAN TERIMA KASIH

Ucapan terima kasih disampaikan kepada Universitas Darussalam Gontor Ponorogo yang telah mendanai penelitian ini.

REFERENSI

- Katadata. (2016). Pengguna Smartphone di Indonesia 2016-2019. Retrieved March 2, 2019, from <https://databoks.katadata.co.id/datapublish/2016/08/08/pengguna-smartphone-di-indonesia-2016-2019>
- Market share of mobile operating systems in Indonesia from January 2012 to December 2017. (2018). Retrieved December 8, 2018, from <https://www.statista.com/statistics/262205/market-share-held-by-mobile-operating-systems-in-indonesia/>
- Nuari, N. (2014). Perancangan Aplikasi Layanan Mobile Informasi Administrasi Akademik Berbasis Android Menggunakan Webservice (Studi Kasus Reg. B Universitas Tanjungpura). *Jurnal Sistem Dan Teknologi Informasi (JustIN)*, 1, 1–7.
- Popper, B. (2017). Google announces over 2 billion monthly active devices on Android. Retrieved December 8, 2018, from <https://www.theverge.com/2017/5/17/15654454/android-reaches-2-billion-monthly-active-users>
- Raj, P., Raman, A., & Subramanian, H. (2017). *Architectural Patterns*. Packt Publisher.
- Saleh, H. (2017). MVVM architecture, ViewModel and LiveData Part 1. Retrieved December 6, 2018, from <https://proandroiddev.com/mvvm-architecture-viewmodel-and-livedata-part-1-604f50cd1>
- Sommerville, I. (2011). *Software Engineering* (9th ed.). Massachusetts: Pearson.
- Tiari, P. K. (2015). What's the difference between design patterns and architectural patterns? Retrieved December 9, 2018, from Stack Overflow website: <https://stackoverflow.com/a/33757364>
- Wikipedia contributors. (n.d.). Model–view–viewmodel. Retrieved December 8, 2018, from <https://en.wikipedia.org/w/index.php?title=Model–view–viewmodel&oldid=871113657>