

## Pengembangan Aplikasi Android SIMIPA Modul Dosen Menggunakan Metode Scrum

<sup>1</sup>Fergie Nando Dwi Putra, <sup>2</sup>Ardiansyah, <sup>3</sup>Dwi Sakethi dan <sup>4</sup>Anie Rose Irawati

<sup>1,2,3,4</sup>Program Studi Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Lampung, Jl. Prof. Dr. Ir. Sumantri Brojonegoro, No. 1, Gedong Meneng, Kecamatan Rajabasa, Kota Bandar Lampung, Lampung, Indonesia

e-mail: <sup>1</sup>[fergienando09@gmail.com](mailto:fergienando09@gmail.com), <sup>2</sup>[ardiansyah@fmipa.unila.ac.id](mailto:ardiansyah@fmipa.unila.ac.id), <sup>3</sup>[dwijim@fmipa.unila.ac.id](mailto:dwijim@fmipa.unila.ac.id),  
<sup>4</sup>[anie\\_rose@gmail.com](mailto:anie_rose@gmail.com)

---

**Abstract** — *The use of mobile-based technology is very useful for all fields, especially education. The rapid development of mobile technology encourages the University to be able to take full advantage of mobile technology. The Faculty of Mathematics and Natural Sciences, University of Lampung currently only has a web-based information system and does not yet have a mobile-based application that can be used to help the learning process, so that problems often occur during the learning process. Problems that often occur are forgetting class schedules, forgetting to attend student seminars, forgetting if there is a guidance schedule, not knowing the progress of guidance student studies, etc., with the development of the SIMIPA application it is hoped that it can help overcome problems that often occur in the learning process. This development focuses on the lecturer module with features such as schedule, agenda, academic calendar, endorsements, and notifications. The method used in the development of SIMIPA is the scrum method. The Scrum method has a product backlog that is divided into eight sprints and must be worked out to meet all application requirements. The Scrum method has three roles at the time of its work, namely the product owner, the Scrum master, and the development team. The scrum method has 4 stages starting from sprint planning, daily scrum, sprint review, and sprint retrospective. The result of this research is the successful development of the SIMIPA application for the Android-based lecturer module. For testing application functionality, the black box testing method is used. The results of the black box testing on the lecturer module SIMIPA application show all functionality in the application is running and as previously expected.*

**Keywords:** *Black box testing; Lecturers; Mobile technology; Notifications; Scrum.*

---

### 1. PENDAHULUAN

Proses pembelajaran merupakan proses kegiatan yang dijalani oleh peserta didik untuk mencapai tujuan pendidikan. Proses pembelajaran biasanya dilakukan oleh pendidik profesional misalnya dosen. Dosen adalah pendidik profesional dan ilmuwan dengan tugas utama untuk mentransformasikan, mengembangkan, dan menyebarluaskan ilmu pengetahuan kepada masyarakat terutama mahasiswa. Pemanfaatan TIK terutama penerapan penggunaan sistem informasi pada bidang akademik di perguruan tinggi saat ini menjadi suatu kebutuhan wajib supaya dapat meningkatkan daya saing perguruan tinggi [1].

Kemajuan teknologi berbasis *mobile* yang semakin pesat mendorong masyarakat untuk menggunakan teknologi *smartphone* agar dapat mengikuti perkembangan zaman. Sesuai dengan perkembangan teknologi saat ini, maka pengembangan aplikasi telah mengarah kepada penggunaan teknologi *smartphone*, di mana semua informasi dapat diakses pada *smartphone* hanya dengan bantuan akses internet.

Penggunaan teknologi *smartphone* pada tingkat universitas akan memudahkan penggunaannya terutama dosen untuk mendapatkan informasi-informasi yang dibutuhkan dengan cepat dan juga dapat menjadi solusi terhadap masalah-masalah yang terjadi pada proses pembelajaran. Beberapa permasalahan yang biasanya terjadi yaitu, lupa jadwal, tidak mengetahui kemajuan studi mahasiswa, tidak mengetahui kalender akademik dan lainnya.

Perangkat lunak Sistem Informasi MIPA (SIMIPA) modul dosen menggunakan metode *scrum* dalam pengembangannya. *Scrum* adalah sebuah metode kerangka kerja untuk menyelesaikan permasalahan yang kompleks, dan adaptif untuk menghasilkan produk yang bernilai tinggi dengan kreativitas dan produktivitas kerja pada waktu yang bersamaan [2]. Metode ini cocok digunakan untuk pengembangan aplikasi SIMIPA yang dibagi ke dalam beberapa tim. Pengembangan aplikasi menjadi lebih terkontrol dan terorganisir serta memiliki pembagian tugas yang jelas.

Penelitian sebelumnya mengenai Rancang Bangun Aplikasi Pengingat Aktivitas Perkuliahan Berbasis Android [3], secara umum aplikasi yang dikembangkan memiliki fungsi untuk mengingat aktivitas perkuliahan mahasiswa. Aplikasi ini dapat membantu aktivitas mahasiswa dalam mengingat aktivitas perkuliahan, namun belum dibuatkan modul dosen pada aplikasi ini.

Berdasarkan uraian tersebut, penelitian yang dilaksanakan adalah mengembangkan perangkat lunak SIMIPA modul dosen dengan metodologi *scrum* berbasis Android. Fitur-fitur yang dikembangkan pada aplikasi ini meliputi fitur jadwal, fitur agenda hari ini, fitur kalender akademik, dan fitur rekapitulasi yang dapat mendukung kegiatan akademik dosen pada Fakultas MIPA Universitas Lampung.

## 2. METODOLOGI PENELITIAN

Pengembangan perangkat lunak SIMIPA modul dosen memiliki beberapa tahapan yang dikerjakan, yaitu perumusan masalah, studi literatur, pengembangan sistem, dan pelaporan.

### 2.1. Perumusan Masalah

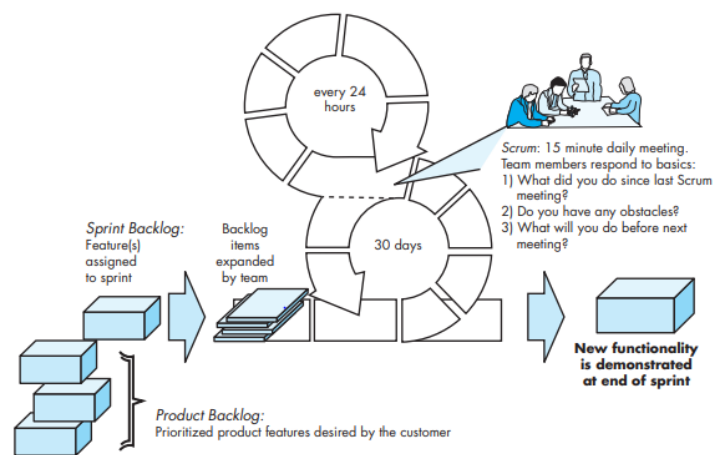
Pada Tahapan ini berfungsi untuk mengidentifikasi masalah yang harus diselesaikan dalam penelitian melalui pengumpulan data. Pada tahap ini juga ditentukannya tujuan, manfaat, dan batasan masalah agar dalam pengembangan perangkat lunak SIMIPA lebih terarah dan tidak keluar dari ruang lingkup penelitian serta batasan penelitian.

### 2.2. Studi Literatur

Tahapan ini dilakukan untuk mengumpulkan teori-teori dan hasil-hasil penelitian terlebih dahulu yang dijadikan sebagai landasan penelitian. Tahapan ini dilakukan dengan mengumpulkan informasi-informasi mengenai tinjauan studi dan tinjauan pustaka pada penelitian.

### 2.3. Pengembangan Sistem

Tahapan pengembangan sistem pada penelitian ini menggunakan metode *scrum*. Metode *scrum* mampu mengatasi masalah yang kompleks. Metode ini memprioritaskan tugas-tugas individu, lalu mendelegasikan tugas-tugas tersebut kepada anggota tim yang dianggap sesuai untuk menyelesaikan tugas tersebut. Metode ini juga banyak digunakan oleh beberapa praktisi teknologi informasi karena dapat mengetahui secara cepat dan tepat resiko dan kualitas perangkat lunak yang dikembangkan [4]. Proses pengembangan menggunakan metode *scrum* terdapat lima tahapan pengembangan, yaitu *product backlog*, *sprint planning*, *daily scrum*, *sprint review*, dan *sprint retrospective* [5]. Pada metode ini terdapat 3 peran yang diperlukan, yaitu *scrum master*, *product owner*, dan *development team*. *Scrum master* adalah pemimpin yang melayani *development team*, serta bertanggung jawab untuk mengenalkan dan menyokong penggunaan *scrum* agar membantu tim untuk memahami teori, praktik-praktik, aturan dan tata nilai *scrum*. *Product owner* pada metode *scrum* sebagai pihak yang menentukan fitur yang dikembangkan pada penelitian. *Development team* memiliki tugas untuk menganalisis, implementasi, perancangan, serta pengujian terhadap fitur yang telah ditentukan oleh *product owner*. Tahapan-tahapan yang dikerjakan di dalam metode *scrum* dapat dilihat pada Gambar 1 [6].



Gambar 1. Tahapan metode *scrum* [6].

### 2.3.1 Product Backlog

Tahapan pertama pada metode *scrum* adalah menyusun *product backlog*. *Product backlog* adalah fitur diperlukan untuk memenuhi visi pemilik produk (owner) untuk pengembangan produk yang sedang berlangsung, *product backlog* meliputi fitur baru, perubahan pada fitur yang ada, cacat yang membutuhkan perbaikan, peningkatan teknis, dan sebagainya [7]. *Product backlog* yang dikerjakan pada penelitian ini dapat dilihat pada Tabel 1.

Tabel 1. *Product backlog* SIMIPA modul dosen.

No	Product Backlog
1	Fitur home
2	Fitur intro dan splashscreen
3	Fitur Login
4	Fitur jadwal
5	Fitur agenda hari ini
6	Fitur kalender akademik
7	Fitur validasi
8	Fitur rekapitulasi

### 2.3.2 Sprint Planning

Tahapan kedua adalah *sprint planning*, yang berfungsi untuk mendiskusikan dan menjelaskan *product backlog* kepada *development team* serta menentukan *product backlog* yang dikerjakan.

### 2.3.3 Daily Scrum

Tahapan ketiga adalah *daily scrum*, yang digunakan untuk melaporkan dan menginspeksi perkembangan pekerjaan *development team* menuju *sprint goal* dan perkembangan menyelesaikan pekerjaan di *sprint backlog*.

#### 2.3.4 *Sprint Review*

Tahapan keempat adalah *sprint review*, yang berfungsi untuk mempresentasikan hasil pengerjaan pada *sprint backlog* yang berguna untuk mendapatkan masukan serta saran dari *scrum master*, *development team*, serta *stakeholder* yang diundang oleh *product owner*. Pada tahap ini selain mempresentasikan hasil pengerjaan akan dilaksanakan pengujian terhadap fitur yang telah dikembangkan menggunakan *black box testing*. *Black box testing* adalah pengujian yang dilakukan dengan menguji fungsionalitas sistem berdasarkan *input* dan *output* yang dihasilkan [8].

#### 2.3.5 *Sprint Retrospective*

Tahapan kelima adalah *sprint retrospective*, yang berfungsi untuk menyelesaikan saran dan masukan yang diberikan pada tahapan *sprint review*, serta *scrum master* mengedukasi tim untuk selalu bekerja di bawah batas waktu yang telah ditentukan.

### 2.4 Tahapan Pelaporan

Pada tahapan ini penulisan laporan dilakukan untuk mendokumentasikan seluruh kegiatan dari tahapan awal hingga tahapan akhir. Penulisan laporan ini menjelaskan bagaimana penelitian ini dilakukan saat mengembangkan perangkat lunak SIMIPA modul dosen berbasis Android.

## 3. HASIL DAN PEMBAHASAN

Pengembangan perangkat lunak SIMIPA modul dosen pada penelitian ini menggunakan metode *scrum*. Pengembangan perangkat lunak SIMIPA berdasarkan pada *product backlog* yang telah disusun oleh *product owner*. Pengerjaan *product backlog* dibagi ke dalam beberapa *sprint backlog* yang dikerjakan. *Sprint backlog* adalah proses pemenuhan kebutuhan sesuai dengan yang diinginkan pada proses *product backlog* yang telah ditentukan [9].

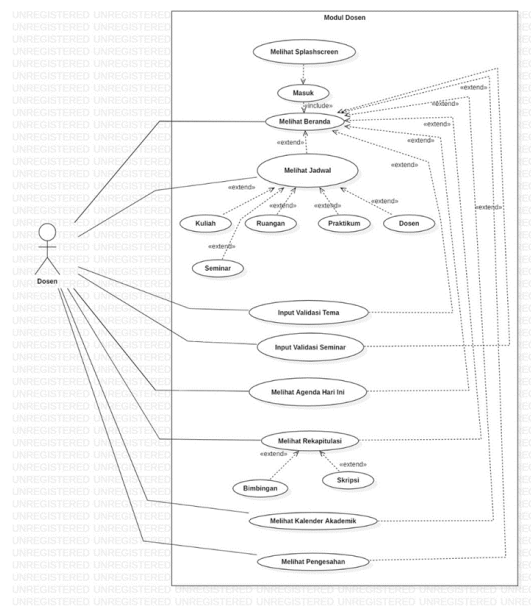
### 3.1 *Sprint ke-1*

#### 3.1.1 *Sprint Planning*

*Sprint planning* pertama memiliki tiga *product backlog* yang harus dikerjakan pada *sprint* pertama. *Sprint backlog* pertama yaitu *use case diagram*, *activity diagram*, dan *sequence diagram*.

#### 3.1.2 *Daily Scrum*

*Daily scrum* pertama mengerjakan pembuatan *diagram-diagram* yang berguna untuk mendefinisikan requirement, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek. Hasil dari pembuatan *use case diagram* dapat dilihat pada Gambar 2.



Gambar 2. Use case diagram modul dosen.

### 3.1.3 Sprint Review

*Sprint review* dilaksanakan oleh seluruh anggota tim *scrum*. *Sprint review* dimulai dengan mempresentasikan hasil yang telah dibuat pada *daily scrum* yang nantinya akan diberikan solusi, perbaikan, dan masukan yang akan diteruskan pada *sprint retrospective*.

### 3.1.4 Sprint Retrospective

*Sprint retrospective* pada *sprint* pertama mendapat saran dari *scrum master* untuk melengkapi beberapa diagram yang masih kurang dan menambahkan beberapa *use case* pada *use case diagram*.

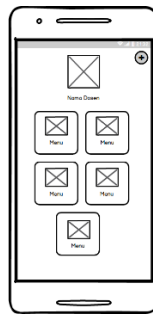
## 3.2 Sprint Ke-2

### 3.2.1 Sprint Planning

*Sprint planning* pada *sprint* kedua memiliki 6 *product backlog* yang dikerjakan dan diselesaikan. *Product backlog* pada *sprint* kedua yaitu UI *home*, UI *intro* dan *splashscreen*, UI *login*, UI *jadwal*, UI *agenda hari ini*, dan UI *pengesahan*.

### 3.2.2 Daily Scrum

*Daily scrum* pada *sprint* kedua berfokus pada pengerjaan dan penyelesaian tampilan pada perangkat lunak SIMIPA modul dosen. Contoh hasil dari pengerjaan *daily scrum* pada *sprint* kedua dapat dilihat pada Gambar 3.



Gambar 3. Rancangan tampilan *home*.

### 3.2.3 *Sprint Review*

Tahapan yang dilakukan pada *sprint review* adalah melakukan presentasi dari hasil yang telah dikerjakan sebelumnya pada *daily scrum* yang berguna untuk mendapatkan *feedback*, masukan, dan solusi yang menjadi *retrospective* untuk *sprint* kedua

### 3.2.4. *Sprint Retrospective*

Pada *sprint* ke dua ini *scrum master* dan seluruh anggota tim secara keseluruhan telah menyetujui hasil pengerjaan sehingga pada pada *sprint* kedua tidak memiliki *retrospective* dan dapat melanjutkan untuk mengerjakan *sprint* ketiga

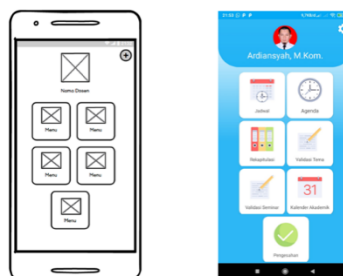
## 3.3 *Sprint Ke-3*

### 3.3.1 *Sprint Planning*

*Sprint planning* pada *sprint* ketiga memiliki satu *product backlog* yang harus dikerjakan dan diselesaikan. *Sprint backlog* pada *sprint* ketiga yaitu implementasi *mockup* ke XML.

### 3.3.2 *Daily Scrum*

*Daily scrum* pada *sprint* ketiga mengerjakan satu *product backlog* yaitu mengimplementasikan semua *mockup* yang telah dibuat ke dalam XML. Contoh hasil implementasi *mockup* ke dalam xml dapat dilihat pada Gambar 4.



Gambar 4. Hasil implementasi *mockup* ke XML.

### 3.3.3 *Sprint Review*

*Sprint review* dilaksanakan oleh seluruh anggota tim *scrum*. Tahapan awal yang dilakukan dalam pelaksanaan *sprint review* adalah mempresentasikan hasil yang telah dikerjakan pada *daily scrum*.

### 3.3.4 *Sprint Retrospective*

Pada *sprint* ketiga yang sudah diselesaikan hingga tahap *sprint review*, keseluruhan pengerjaan sudah disetujui oleh *scrum master* dan anggota tim pengembang sehingga pada *sprint* ketiga tidak memiliki *retrospective*.

## 3.4 *Sprint Ke-4*

### 3.4.1 *Sprint Planning*

*Sprint planning* pada *sprint* keempat memiliki tiga *product backlog* yang dikerjakan dan diselesaikan secara tim. *Sprint backlog* pada *sprint* keempat yaitu fitur *intro* dan *splashscreen*, fitur *login*, dan fitur *home*.

### 3.4.2 *Daily Scrum*

*Daily scrum* pada *sprint* keempat mengerjakan tiga *product backlog* yaitu pengembangan fungsi *intro* dan *splashscreen*, pengembangan fungsi *login*, dan pengembangan fungsi *home*. Pengembangan fungsi ini menggunakan bahasa pemrograman Java. Hasil dari pengerjaan *product backlog* pada *sprint* keempat dapat dilihat pada Gambar 5.



Gambar 5. Hasil pengembangan fungsi *sprint* ke-4.

### 3.4.3 *Sprint Review*

*Sprint review* dilakukan dengan tahapan awal yaitu mempresentasikan hasil dari pengerjaan *sprint* keempat. *Sprint review* pada *sprint* keempat dilakukan pengujian-pengujian terhadap pengembangan fungsi yang telah dikerjakan menggunakan metode *black box testing*.

### 3.4.4 *Sprint Retrospective*

*Sprint retrospective* dilaksanakan oleh seluruh anggota tim untuk berdiskusi membahas kendala dan solusi yang telah dilaksanakan pada *sprint* keempat. *Scrum master* dan *development team* secara keseluruhan telah menyetujui hasil pengerjaan pada *sprint* keempat.

## 3.5 *Sprint Ke-5*

### 3.5.1 *Sprint Planning*

*Sprint planning* kelima memiliki tiga *product backlog* yang harus diselesaikan. *sprint backlog* pada *sprint* kelima yaitu fitur agenda hari ini dan notifikasi, fitur jadwal kuliah, dan fitur jadwal praktikum.



### 3.5.2 Daily Scrum

Daily scrum pada *sprint* kelima mengerjakan tiga *sprint backlog* yaitu melakukan pengembangan fungsi agenda hari ini beserta notifikasi, jadwal kuliah dosen, dan jadwal praktikum dosen. Hasil dari pengerjaan *daily scrum* pada *sprint* kelima dapat dilihat pada Gambar 6.



Gambar 6. Hasil pengembangan pada *sprint* ke-5

### 3.5.3 Sprint Review

Pada pelaksanaan *sprint review* dilakukan presentasi dan pengujian terhadap fungsi-fungsi yang telah dikembangkan menggunakan metode *black box testing*. Hasil pengujian *black box testing* pada *sprint* kelima dapat dilihat pada Tabel 2.

Tabel 2. Hasil pengujian *black box testing* *sprint* ke-5.

Kelas Uji	Hasil yang Diharapkan	Hasil Pengujian
Activity Jadwal (Kuliah dan Praktikum)	Aplikasi menampilkan jadwal (kuliah dan praktikum)	Sesuai yang diharapkan
Activity Agenda Hari Ini	Aplikasi menampilkan agenda hari ini	Sesuai yang diharapkan
Notifikasi	Aplikasi memberikan notifikasi agenda	Sesuai yang diharapkan

### 3.5.4 Sprint Retrospective

Pada *sprint* kelima, *scrum master* memberikan *sprint retrospective* perbaikan pada beberapa tampilan. *Retrospective* pada *sprint* ini dapat dilihat pada Tabel 3.

Tabel 3. *Sprint retrospective* *sprint* ke-5.

No	Retrospective
1	Penghapusan nama dosen dan judul seminar pada <i>cardview</i> agenda hari ini
2	Berikan perbedaan warna pada <i>cardview</i> agenda hari ini
3	Button notifikasi pada agenda hari ini apabila sudah di klik maka tombol tidak dapat di klik kembali



### 3.6 Sprint Ke-6

#### 3.6.1 Sprint Planning

Tahapan *sprint planning* pada *sprint* keenam didapatkan tiga *product backlog* yang harus dikerjakan. *Product backlog* pada *sprint* keenam yaitu fitur jadwal seminar dosen, fitur jadwal ruang, dan fitur jadwal dosen.

#### 3.6.2 Daily Scrum

*Daily scrum* pada *sprint* keenam mengerjakan tiga *product backlog* pada menu jadwal. Hasil dari pengembangan *daily scrum* dapat dilihat pada Gambar 7.



Gambar 7. Tampilan hasil pengembangan fungsi *sprint* ke-6

#### 3.6.3 Sprint Review

*Sprint review* pada *sprint* keenam diawali dengan melakukan presentasi hasil pengerjaan dan melakukan beberapa pengujian terhadap pengembangan fungsi-fungsi yang telah diselesaikan pada *sprint* ini dengan menggunakan metode *black box testing*. Hasil pengujian *black box testing* pada *sprint* keenam dapat dilihat pada Tabel 4.

Tabel 4. Hasil pengujian *black box testing* *sprint* ke-6.

Kelas Uji	Hasil yang Diharapkan	Hasil Pengujian
<i>Activity</i> Jadwal (Seminar, Ruang, dan Dosen)	Aplikasi menampilkan jadwal (seminar, ruang dan dosen)	Sesuai yang diharapkan

#### 3.6.4 Sprint Retrospective

*Sprint retrospective* yang dihasilkan pada *sprint* keenam adalah beberapa perbaikan pada tampilan *cardview* jadwal seminar dosen untuk menambahkan keterangan pembimbing dan memberikan nama mahasiswa pada *cardview*.

### 3.7 Sprint Ke-7

#### 3.7.1 Sprint Planning

*Sprint planning* ketujuh kembali dilaksanakan untuk membahas *product backlog* yang akan dikerjakan pada pelaksanaan *sprint* ketujuh. *Product backlog* yang dikerjakan pada *sprint* ketujuh terdapat tiga *product backlog*. *Product backlog* yang dikerjakan pada *sprint* ketujuh yaitu fitur publikasi, fitur penelitian dan fitur pengabdian.

### 3.7.2 Daily Scrum

*Daily scrum* pada *sprint* ketujuh yaitu mengembangkan fungsi pengesahan pada Fakultas MIPA Universitas Lampung. Pengembangan fungsi ini memiliki fitur yang berfungsi untuk menampilkan pengesahan penelitian, pengabdian, dan publikasi pada FMIPA Universitas Lampung sesuai dengan yang ditampilkan pada website FMIPA serta dapat menampilkan detail dari penelitian, pengabdian dan publikasi tersebut. Hasil pengembangan pada *sprint* ketujuh dapat dilihat pada Gambar 8.



Gambar 8. Tampilan hasil pengembangan fungsi *sprint* ke-7.

### 3.7.3 Sprint Review

*Sprint review* dilaksanakan dengan diawali *development team* melakukan presentasi hasil pengerjaan *product backlog* pada *sprint* ketujuh. *Sprint review* pada *sprint* ketujuh dilakukan pengujian terhadap fungsi-fungsi yang sudah diselesaikan dengan menggunakan metode *black box testing*. Hasil pengujian *black box testing* pada *sprint* ketujuh dapat dilihat pada Tabel 5.

Tabel 5. Hasil pengujian *black box testing* *sprint* ke-7.

Kelas Uji	Hasil yang Diharapkan	Hasil Pengujian
Activity Pengesahan (Pengabdian, Penelitian, dan Publikasi)	Aplikasi menampilkan <i>fragment</i> pengabdian, penelitian, dan publikasi	Sesuai yang diharapkan

### 3.7.4 Sprint Retrospective

*Sprint retrospective* pada *sprint* ketujuh menghasilkan beberapa *retrospective* untuk memperbaiki tampilan, diantaranya memperbaiki tampilan detail data dengan menambahkan *subtitle*, *app bar*, dan tampilan nama.

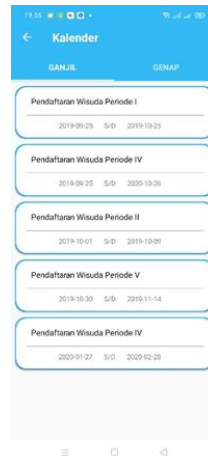
## 3.8 Sprint Ke-8

### 3.8.1 Sprint Planning

*Sprint planning* dilakukan untuk membahas *product backlog* yang dikerjakan pada *sprint* kedelapan. Hasil dari *sprint planning* pada *sprint* kedelapan adalah mengerjakan satu *product backlog*. *Product backlog* yang dikerjakan pada *sprint* kedelapan adalah pengembangan fungsi kalender akademik ganjil dan genap pada Universitas Lampung.

### 3.8.2 Daily Scrum

*Daily scrum* pada *sprint* kedelapan yaitu mengembangkan fungsi pada menu kalender akademik. Pengembangan pada fungsi kalender akademik berguna untuk menampilkan dan memberikan informasi jadwal akademik pada tahun yang sedang berjalan. Menu ini berisi daftar jadwal kalender akademik sesuai dengan tahun akademik yang sedang berjalan. Hasil pengembangan pada *sprint* ke-8 dapat dilihat pada Gambar 9.



Gambar 9. Tampilan hasil pengembangan fungsi *sprint* ke-8.

### 3.8.3 Sprint Review

*Sprint review* dilakukan untuk melaporkan hasil yang telah dikerjakan pada *sprint* kedelapan untuk mendapatkan masukan serta perbaikan yang berasal dari *scrum master*. *Sprint review* pada *sprint* kedelapan dilakukan pengujian fungsi-fungsi yang telah dikerjakan menggunakan metode *black box testing*. Hasil pengujian *black box testing* pada *sprint* kedelapan dapat dilihat pada Tabel 6.

Tabel 6. Hasil pengujian *black box testing* *sprint* ke-8.

Kelas Uji	Hasil yang Diharapkan	Hasil Pengujian
Activity Kalender Akademik	Aplikasi menampilkan kalender akademik (ganjil dan genap)	Sesuai yang diharapkan

### 3.8.4 Sprint Retrospective

*Sprint retrospective* pada *sprint* kedelapan mendapatkan masukan dari *scrum master* untuk menyesuaikan tampilan pada *cardview* agar tidak kaku dan dapat menyesuaikan pada layar *smartphone* yang berukuran kecil.

## 4. KESIMPULAN

Penelitian ini berhasil mengembangkan perangkat lunak SIMIPA modul dosen menggunakan metode *scrum* sesuai dengan *sprint* dan *product backlog* yang telah ditentukan. *Sprint* yang harus dikerjakan pada penelitian ini berjumlah delapan *sprint*. *Sprint* pertama memiliki tiga *sprint backlog* yang harus dikerjakan, dan dapat diselesaikan dalam waktu 2 minggu tanpa hambatan. *Sprint* kedua memiliki enam *sprint backlog* yang harus dikerjakan, dan dapat diselesaikan dalam waktu 2 minggu. *Sprint* ketiga memiliki satu *sprint backlog* yang harus dikerjakan, dan dapat diselesaikan dalam waktu 1 minggu tanpa hambatan. *Sprint* keempat memiliki tiga

*sprint backlog* yang harus dikerjakan, dan dapat diselesaikan dalam waktu 3 minggu tanpa hambatan. *Sprint* kelima memiliki tiga *sprint backlog* yang harus dikerjakan, dan dapat diselesaikan dalam waktu 5 minggu dengan hambatan pemilihan fungsi notifikasi yang akan digunakan. Upaya yang digunakan untuk menyelesaikan hambatan yaitu dengan mendiskusikan kepada tim dan meminta saran, sehingga solusi yang didapatkan untuk fungsi notifikasi adalah *alarm manager*. *Sprint* keenam memiliki 3 *sprint backlog* yang harus dikerjakan, dan dapat diselesaikan dalam waktu 3 minggu tanpa hambatan. *Sprint* ketujuh memiliki 3 *sprint backlog* yang harus dikerjakan, dan dapat diselesaikan dalam waktu 1 minggu tanpa hambatan. *Sprint* kedelapan memiliki 1 *sprint backlog* yang harus dikerjakan, dan dapat diselesaikan dalam waktu 1 minggu tanpa hambatan. Pengujian perangkat lunak SIMIPA menggunakan metode *black box testing* pada *sprint* kelima hingga kedelapan menunjukkan keseluruhan fungsional pada aplikasi SIMIPA modul dosen berjalan dengan baik dan sesuai dengan *input* dan *output* yang diharapkan.

## DAFTAR PUSTAKA

- [1] D. Kurniadi & A. Mulyani, "Implementasi Pengembangan Student Information Terminal (S-IT) untuk Pelayanan Akademik Mahasiswa," *Jurnal Algoritma*, 2016.
- [2] K. Schwaber & J. Sutherland, *The Scrum Guide*. [Online], Tersedia: <https://www.scrumguides.org/scrum-guide-2017.html>, 2017.
- [3] B. Septian, G. B. Putra & F. Arkan, "Rancang Bangun Aplikasi Pengingat Aktivitas Perkuliahan Berbasis Android," *Prosiding SNRT (Seminar Nasional Riset Terapan)*, Politeknik Negeri Banjarmasin, 2016.
- [4] R. S. Pressman, *Software Engineering: A Practitioner's Approach, 8<sup>th</sup> Edition*. New York: McGraw-Hill, 2015.
- [5] M. Alqudah & R. Razali, "A review of scaling agile methods in large software development," *International Journal on Advanced Science, Engineering Information Technology*, Vol 6, No 6, 828-837, 2016.
- [6] L. Mutawali, B. K. Fathoni & H. Asyari, "Implementasi Scrum Dalam Pengembangan Sistem Informasi Jasa Desain Grafis," *Jurnal Manajemen informatika dan Sistem Informasi*, vol. III, no. 2, pp. 116-122, 2020.
- [7] B. Septian, I.K. Jayadi, M. Holil & I. Handriani, "Sistem Human Capital Management Menggunakan Metode Scrum," *Jurnal Sistem Informasi, Teknologi Informatika dan Komputer*, Universitas Muhammadiyah Jakarta, Vol 11, No 1, 2020.
- [8] I. Sommerville, *Software Engineering Tenth Edition (10th ed.; M. Hirsch, Ed.)*, London: Pearson, 2016.
- [9] H. Dafitri & M. Elsera, "Rancang Bangun Sistem Informasi Akademik Berbasis Web (studi kasus: SMA Swasta Harapan I Medan)," *Journal of Information Systems*, Vol 1, No 2, 2017.