

Pengembangan Aplikasi WISTA dengan Pendekatan Model Driven Development

Roy Deddy Hasiholan Lumbantobing¹, Christin Situmorang^{2*}, Samuel Silalahi³, Marthin Satria Pasaribu⁴

* Corresponding author:

¹ Program Studi D4 Teknik Informatika Fakultas Teknologi Informatika dan Elektro, Institut Teknologi Del, Jln.Sisimangaraja Sitoluama, Laguboti, Tobasa 22381 INDONESIA (email: roy.deddy@del.ac.id)

² Program Studi D4 Teknik Informatika Fakultas Teknologi Informatika dan Elektro, Institut Teknologi Del, Jln.Sisimangaraja Sitoluama, Laguboti, Tobasa 22381 INDONESIA (email: christinseventina@gmail.com)

³ Program Studi D4 Teknik Informatika Fakultas Teknologi Informatika dan Elektro, Institut Teknologi Del, Jln.Sisimangaraja Sitoluama, Laguboti, Tobasa 22381 INDONESIA (email: if413009@students.del.ac.id)

⁴ Program Studi D4 Teknik Informatika Fakultas Teknologi Informatika dan Elektro, Institut Teknologi Del, Jln.Sisimangaraja Sitoluama, Laguboti, Tobasa 22381 INDONESIA (email: if413004@students.del.ac.id)

Sejarah penerimaan

Diterima pertama kali:
02/01/2021

Diterima setelah perbaikan:
18/03/2021

Tanggal penerbitan:
06/04/2021

Copyright © 2021 IT Del Press

Abstract— Smartphone is a communication tool that increasingly in demand by all circles in the world. On the smartphone there are various applications built by the developers. Applications can also meet a variety of human needs such as the needs in worship. At this time, the worship applications contained in the playstore can display all the biblical content and lyric of the song desired by the user of the application. But sometimes when a user needs a worship app to see the order of worship for a condition.

To assist the user in finding the order of worship so this project built an application that aims to produce an application that can display the order of worship in accordance with the order of worship used in a church. The purpose of this application is also to reduce the use of paper in a church that can cost very much every week. The development of this application also uses the method of driven development model (MDD) that can help developers like code engineering. MDD is also implementing Software Development Life Cycle which is very critical to the application development process consisting of 4 (four) stages, namely inception, elaboration, construction and transition. This Final Project has also resulted in web and mobile worship companion application where the web is used by fillers in each church while mobile is used by android application user.

Keywords—Smartphone; Model Driven Development; MDD;Worship Applications;

Intisari— *Smartphone* merupakan alat komunikasi yang semakin diminati oleh semua kalangan di dunia. Pada *smartphone* terdapat berbagai aplikasi yang dibangun oleh para developer. Aplikasi yang dibangun juga dapat memenuhi berbagai macam kebutuhan manusia seperti kebutuhan dalam beribadah. Pada saat ini, aplikasi ibadah yang terdapat pada *playstore* dapat menampilkan semua isi alkitab dan lirik nyanyian yang diinginkan oleh pengguna aplikasi atau *user*. Tetapi ada kalanya *user* memerlukan aplikasi ibadah untuk melihat tata ibadah untuk suatu kondisi.

Untuk membantu *user* dalam menemukan tata ibadah maka pada Tugas Akhir ini dibangun sebuah aplikasi yang bertujuan untuk menghasilkan sebuah aplikasi yang dapat menampilkan tata ibadah sesuai dengan tata ibadah yang digunakan pada sebuah gereja. Tujuan dibangunnya aplikasi ini juga adalah untuk mengurangi penggunaan kertas pada sebuah gereja yang dapat mengeluarkan biaya yang sangat banyak setiap minggunya. Pembangunan aplikasi ini juga menggunakan metode *model driven development* (MDD) yang dapat membantu developer seperti code engineering. MDD juga menerapkan *Software Development Life Cycle* yang sangat kritis terhadap proses pengembangan aplikasi yang terdiri dari 4 (empat) tahap, yaitu *inception*, *elaboration*, *construction* dan *transition*.

Pengerjaan Tugas Akhir ini juga telah menghasilkan aplikasi pendamping ibadah berbasis *web* dan *mobile* dimana *web* digunakan oleh pengisi konten pada masing-masing gereja sedangkan *mobile* digunakan oleh pengguna aplikasi *android*.

Kata Kunci— Smartphone; Model Driven Development; MDD;Worship Applications;

I. PENDAHULUAN

Perkembangan teknologi yang semakin pesat saat ini telah membuat banyak perubahan dalam gaya hidup masyarakat. Salah satu perubahannya adalah banyaknya penggunaan *smartphone*, terutama berbasis Android untuk kehidupan sehari-hari. Para pengguna *smartphone* berasal dari berbagai kalangan dan umur. Di *Google I/O 2014*, *Google* memberitakan bahwa terdapat lebih dari 1 miliar pengguna aktif bulanan Android, meningkat dari 583 juta pada bulan Juni 2013[1]. Di Indonesia, pada akhir tahun 2015 diperkirakan sekitar 55 juta pengguna *smartphone* dengan total penetrasi mencapai 37,1%[2]. Di Benua Asia total pertumbuhan pengguna *smartphone* yang paling besar adalah China, peringkat kedua adalah India, dan diikuti oleh Indonesia pada peringkat ketiga[3]. Aplikasi Android saat ini di *playstore* sudah mencapai 2,2 juta aplikasi, dimana menduduki peringkat pertama dalam urutan *app store*[4]. Beberapa kategori diantaranya adalah *Android Wear, Art and Design, Auto & Vehicles, Beauty, Books and Reference, Business, Comic, Communication, Dating, Education, Entertainment, Events, Finance, dan Food and Drink*[5]. Salah satu contoh aplikasi yang menarik pada kategori *books and reference* adalah Alkitab dan Kidung Jemaat. Hal ini disebabkan karena pada saat ini masyarakat khususnya yang beragama Kristen Protestan sudah sering menggunakan *smartphone* untuk membuka Alkitab dan Kidung Pujian.

Sesuai dengan eksplorasi yang telah dilakukan pada aplikasi-aplikasi tersebut terdapat aplikasi eKatholik yang merupakan aplikasi yang memiliki tata ibadah yaitu kalender liturgi yang dibuat setiap minggunya untuk umat katolik sedangkan untuk umat Kristen Protestan sejauh ini belum ada aplikasi yang dapat membuat tata ibadah setiap minggunya. Hal ini yang membuat penulis membangun aplikasi pendamping ibadah yang memiliki fitur berbeda dari aplikasi lain. Aplikasi yang dibangun dapat menyesuaikan sesuai dengan kebutuhan masyarakat gereja. Aplikasi WISTA dapat menyimpan data tata ibadah untuk gereja HKBP, GKPI dan HKI. Pembangunan aplikasi ini juga menerapkan pembangunan aplikasi berbasis MDD (*Model Driven Development*). MDD merupakan sebuah ide yang mampu mentransformasikan model ke dalam sebuah sistem computer. MDD digunakan untuk mendukung pengembangan sistem dengan sebuah pemodelan. Model dapat dibentuk dengan berbagai jenis pemodelan, salah satunya adalah UML[5]. UML merupakan penerapan model untuk pengembangan *software* tradisional.

II. STUDY LITERATUR

2a. *Software Development Life Cycle*

Menurut Dennis, Wixom, dan Tegarden (2010) SDLC merupakan pengembangan sistem informasi yang terdiri dari beberapa tahapan, yaitu: 1) perencanaan, 2) analisis, 3) perancangan, dan 4) implementasi[6].

- a. Perencanaan, merupakan proses untuk mengerti mengapa sebuah sistem informasi dibangun dan menentukan bagaimana tim dapat membangun sistem.

- b. Analisis, merupakan proses yang akan menjawab pertanyaan tentang siapa yang akan menggunakan sistem, apa saja yang dapat dilakukan sistem, dan dimana dan kapan saja sistem tersebut dapat digunakan.
- c. Desain, merupakan proses bagaimana sistem akan berjalan pada perangkat keras (*hardware*), perangkat lunak (*software*) dan sarana jaringan (*network*), tampilan *user interface*, laporan yang digunakan, spesifik program, database, dan file-file yang dibutuhkan.
- d. Implementasi, merupakan tahapan akhir dari SDLC dimana selama proses ini sistem yang telah direncanakan, dianalisis dan didesain akan dibangun menjadi sebuah sistem yang nyata.

2b. *Object Oriented Analysis and Design (OOAD)*

Desain adalah langkah pertama dalam tahap pengembangan untuk setiap produk atau sistem. Desain didefinisikan sebagai proses dan strategi yang digunakan untuk mengatur kompleksitas. Proses desain perangkat lunak adalah sebuah proses yang iterative atau proses yang berulang dimana kebutuhan-kebutuhan akan perangkat lunak diubah ke dalam sebuah *blueprint* atau gambaran untuk membangun perangkat lunak[7].

Menurut Srivastava dan Sabharwal (2006), *Object Oriented Analysis (OOA)* adalah sebuah proses perubahan masalah yang *real* ke dalam sebuah model menggunakan objek dan kelas sebagai model dari sistem yang akan dibangun[8]. Sedangkan *Object Oriented Design (OOD)* adalah proses mengubah masalah dari OOA ke dalam sebuah model berdasarkan objek[8]. Dari OOA dan OOD dapat dihasilkan sebuah pendekatan yang disebut dengan *Object Oriented Analysis and Design (OOAD)* yang merupakan sebuah metodologi yang terdiri dari proses-proses, teknik-teknik (seperti menentukan rumus, model dan notasi), dan mungkin saja memiliki *tools* (contoh: CASE). Untuk teknologi yang dipakai pada OOAD adalah teknologi OO dan bahasa pemrograman seperti Java dan C[9].

2c. *Model Driven Development (MDD)*

MDD merupakan gagasan untuk dapat membangun sebuah model dari sistem yang akan dibangun[10]. MDD merupakan sebuah metodologi pengembangan yang bertujuan untuk menunjukkan langkah-langkah pengembangan secara otomatis dengan menggunakan model pengembangan dimana masing-masing model dipetakan berdasarkan masalahnya masing-masing. Penggunaan MDD tidak terlepas dari pendekatan OOAD[11]. Oleh karena itu, *developer* membutuhkan orang-orang yang mampu untuk menganalisa atau memodelkan setiap detail dari implementasi sistem yang dibangun menggunakan sebuah bahasa pemrograman, dan memungkinkan untuk menganalisis model fungsi seperti apa yang dibutuhkan dan bagaimana arsitektur yang dimiliki sistem secara keseluruhan[11]. Maka, MDD bertujuan untuk mengotomatiskan tugas pemrograman yang kompleks[11]. Hal ini yang mendasari MDD dapat meningkatkan

produktivitas produk dan memberikan manfaat dengan acara mendasar, yaitu meningkatkan produktivitas produk jangka panjang dan jangka pendek. Meningkatkan produktivitas jangka panjang maksudnya adalah mengurangi tingkat dimana nilai artifak yang dihasilkan menjadi usang. Sedangkan meningkatkan produktivitas jangka pendek maksudnya adalah meningkatkan sebuah nilai artifak perangkat lunak yang utama dengan menunjukkan berapa banyak fungsi yang akan diberikan sistem. Dari kedua manfaat diatas maka dapat disimpulkan bahwa sistem atau aplikasi yang dihasilkan dapat berjalan dalam jangka waktu yang lama. Dengan kata lain, ada dua komsepi inti yang terkait dengan pengembangan MDD, yaitu abstraksi dan otomasi. Model aplikasi perangkat lunak didefinisikan pada tingkat abstraksi yang lebih tinggi dan kemudian diubah menjadi aplikasi dengan menggunakan transformasi otomatis atau interpretasi. Pendekatan pengembangan model yang tepat dapat memanfaatkan pencapaian model, dimana model secara otomatis berubah menjadi aplikasi perangkat lunak yang bekerja dengan menafsirkan dan menjalankan model yang telah dibuat.

Menurut Bran Selic (2008), otomasi sejauh ini merupakan sarana teknologi yang paling efektif untuk meningkatkan produktivitas dan kehandalan. Namun, upaya yang paling awal untuk menerapkan otomasi pada pemodelan perangkat lunak terbatas pada peran “bantuan daya”, seperti pembuatan diagram dan pembuatan kode perangkat lunak (*code program*). Hal ini seringkali tidak cukup substantive untuk membuat perbedaan yang signifikan terhadap produktivitas produk. Misalnya, setelah kode dimatikan, model ditinggalkan karena seperti semua dokumentasi perangkat lunak memerlukan sumber daya yang langka dan mahal. Ini sebabnya mengapa pemodelan MDD sangat diperlukan dan menjadi solusi berdasarkan teknik *roundtrip* yang disebut, yang secara otomatis mengubah model menjadi kode program yang jauh bermanfaat. Pada proses pengembangan sistem komputer dengan menggunakan metode MDD dibutuhkan banyak *tools* sesuai dengan tahapan pengembangannya. Pada tahapan analisis dan perancangan misalnya, dibutuhkan *Enterprise Architect* atau *visual diagram* yang dapat menggambarkan model-model dengan menggunakan notasi UML sedangkan pada tahapan implementasi (*code program*) yang dibutuhkan adalah seperti Yii Framework untuk pengembangan *web* atau *android studio* untuk pengembangan *mobile*.

2d. Firebase

Firestore merupakan layanan *DbaaS (Database as a Service)* dengan konsep *realtime* yang memungkinkan untuk membantu pengembangan aplikasi berkualitas tinggi. Produk utama dari *firebase* adalah suatu *database* yang menyediakan API untuk memungkinkan pengembang menyimpan dan mensinkronisasikan data lewat *multiple client*. Dengan adanya *firebase*, pesan yang terdapat pada aplikasi *chatting* dapat dengan mudah diterima oleh *user* hanya dengan hitungan millidetik. *Realtime firebase* memudahkan tim pengembang untuk membangun aplikasi yang kolaboratif dengan akses yang aman ke dalam basis data langsung dari *client-side code*. Jika kondisi aplikasi sedang *offline*, data akan bertahan secara

lokal dan pada saat *online*, *database* akan mensinkronisasikan perubahan data dengan *update remote*. Hingga saat ini, *firebase* menyediakan beberapa *fitur* diantaranya *Realtime Database*, *Firestore Analytics*, *Authentication*, *Cloud Messaging*, *Storage*, *Hosting*, *Remote Config*, *Test Lab* dan *Crash Reporting (Firestore, 2016)*[12].

III. ANALISIS

Pada penggunaan *smartphone* yang semakin meningkat, penulis telah melakukan survey terhadap penggunaan *smartphone*. Sampel dari survey tersebut adalah mahasiswa IT Del dan masyarakat umum. Hasil survey tersebut akan dilakukan analisis terhadap jumlah pemakaian *smartphone*. Survey dilakukan dengan 2 cara, yaitu:

1. Questioner, tahap ini dilakukan dengan menggunakan *google form* dengan pertanyaan sebagai berikut.

No	Pertanyaan
1.	Apakah anda memiliki <i>smartphone</i> ?
2.	Jika ya, pada saat anda membeli <i>smartphone</i> baru, aplikasi apa saja yang anda download terlebih dahulu? (Jika anda memilih aplikasi IBADAH pada pertanyaan ini, maka lanjut ke pertanyaan berikutnya. Jika tidak, berhenti di pertanyaan ini)
3.	Jika anda memilih aplikasi ibadah, fitur apa saja yang anda inginkan?
4.	Pada aplikasi ibadah yang anda miliki, apakah anda merasa bahwa jika menggunakannya selama ibadah anda kerepotan untuk mengganti alkitab dengan nyanyian seperti kidung jemaat ataupun kidung lainnya?
5.	Jika ya, maka kesulitan seperti apa yang anda alami?

Gambar 1 Daftar Pertanyaan untuk Survey menggunakan *Questioner*

Berdasarkan pertanyaan diatas, terdapat 53 mahasiswa IT Del dan 80 orang di luar IT Del yang menjadi responden dari *survey* yang dilakukan. Berikut data yang dihasilkan dari pertanyaan tersebut.

a. Mahasiswa

Untuk mahasiswa IT Del pertanyaan yang diberikan hanya diberikan pertanyaan 1 sampai dengan pertanyaan 3 karena *questioner* diberikan langsung tanpa menggunakan *google form*.

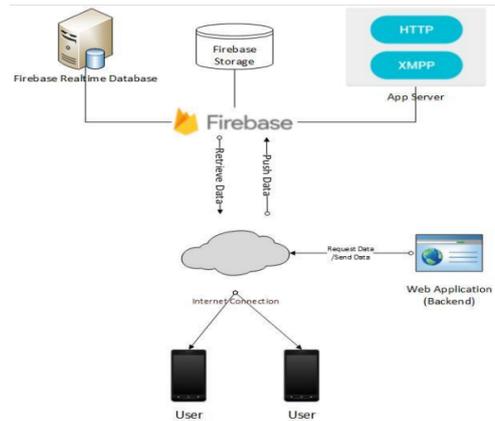
Dari pertanyaan yang terdapat pada *questioner* maka diperoleh hasil sebagai berikut.

- Pertanyaan 1, terdapat 49 mahasiswa yang memiliki *smartphone* dan 4 mahasiswa yang tidak memiliki *smartphone*
- Pertanyaan 2, terdapat 46 mahasiswa yang menjawab aplikasi rohani dan 8 aplikasi media sosial yang akan diunduh ketika pertama sekali membeli *smartphone*.
- Pertanyaan 3, terdapat 53 mahasiswa yang mengunduh alkitab, 42 nyanyian, dan 41 renungan harian. Hasil tersebut didapat dari responden diminta untuk menyebutkan aplikasi rohani tersebut secara spesifik.

b. Masyarakat

Jumlah responden masyarakat pada survey yang dilakukan adalah sebanyak 80 responden. Tetapi ada kalanya responden tersebut tidak menjawab salah satu pertanyaan. Maka pada hasil persentase dan rentang jumlah responden berbeda-beda.

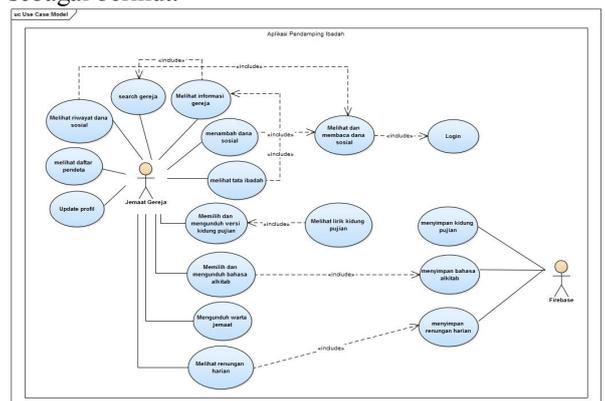
- Pertanyaan 1, terdapat 79 responden yang menjawab dan diperoleh bahwa 98,7% yang menjawab bahwa masyarakat memiliki *smartphone* dan sisanya tidak memiliki *smartphone*.
- Pertanyaan 2, terdapat 76 responden yang menjawab dimana 51 responden atau 67,1 % yang mengunduh media sosial ketika pertama sekali membeli *smartphone*, 48 responden atau 63,2% yang mengunduh aplikasi ibadah, 16 responden atau 21,1% yang akan mengunduh game, 27 responden atau 35,5% yang akan mengunduh aplikasi *chatting*, 6 responden atau 7,9% yang mengunduh aplikasi foto dan 1 responden atau sama dengan 1,3% akan mengunduh aplikasi yang lainnya.
- Pertanyaan 3, dijawab oleh 56 responden dimana 51 responden atau sama dengan 91,1% akan mengunduh aplikasi Alkitab, 39 responden atau 69,6% akan mengunduh nyanyian, 28 responden atau 50% menjawab renungan harian dan 3 responden ata 5,4% akan mengunduh aplikasi ibadah dengan fitur lainnya.
- Pertanyaan 4, dijawab oleh 58 responden dimana 77,6% responden menyatakan bahwa aplikasi ibadah yang dimiliki oleh responden memiliki tingkat kesukaran penggunaan pada saat ibadah tepatnya ketika akan mengganti alkitab dengan nyanyian. Sedangkan untuk 22,4% menyatakan bahwa tidak ada yang sukar pada penggunaan aplikasi yang dimilikinya.
- Pertanyaan 5, pertanyaan memberikan alasan untuk pertanyaan nomor 4. Adapun berbagai macam alasan yang diberikan oleh masyarakat dapat dilihat pada gambar dibawah ini.



Gambar 2 Rancangan Arsitektur Aplikasi

Dari hasil rancangan arsitektur aplikasi diatas jelas terlihat bahwa *user* dapat mengakses aplikasi WISTA melalui *smartphone*. Data yang diambil untuk aplikasi WISTA akan di-load menggunakan *firebase*. Dari sisi *back-end* atau *web application* berguna untuk meng-*input* data agar dapat diakses oleh *user* dan data disimpan menggunakan *realtime database*. Sehingga *firebase* harus menggunakan XAMPP dan HTTP sebagai *application server*.

1. Aplikasi WISTA yang dibangun terdiri dari dua jenis interface, yaitu web dan *android*. Aplikasi WISTA berbasis *web* dibangun menggunakan *Yii Framework* sedangkan aplikasi WISTA berbasis *android* dibangun menggunakan *android studio*.
2. Aplikasi WISTA memiliki beberapa fungsi yang telah diimplementasikan sesuai dengan *requirement user* ketika dilakukan *survey*. Fungsi-fungsi tersebut adalah sebagai berikut.

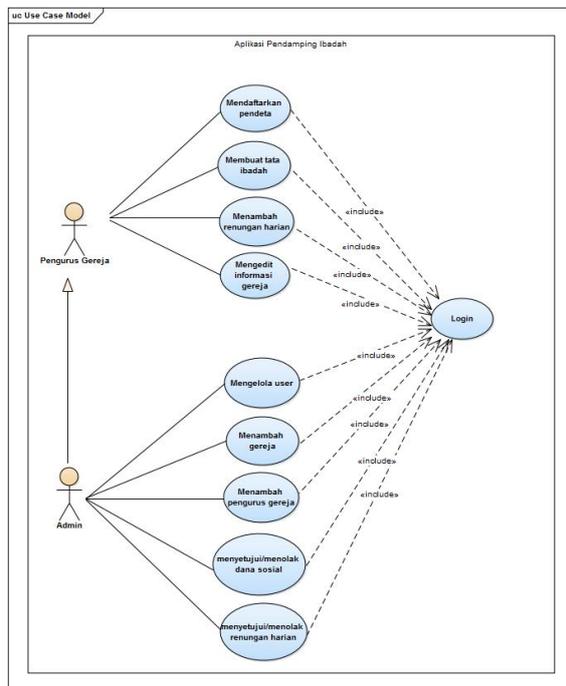


Gambar 3 Use Case Diagram Aplikasi WISTA versi *Mobile*

2. Mengunjungi gereja-gereja, survey ini dilakukan pada beberapa gereja yang menjadi target untuk penerapan aplikasi, yaitu HKBP, GKPI dan HKI.

IV. EKSPERIMEN

Pada percobaan yang dilakukan maka rancangan arsitektur aplikasi dihasilkan sebagai berikut.



Gambar 4 Use Case Diagram Aplikasi WISTA versi Web

3. Scrapping Data

Untuk mendapatkan data alkitab, kidung jemaat dan buku ende yang digunakan pada saat pembangunan aplikasi maka cara yang dicoba dan digunakan adalah melakukan *scrapping data*. Proses *scrapping data* dilakukan sebagai usaha untuk menggunakan *database* dari pihak lain seperti mengambil data dari halaman *website* lain. Untuk data masing-masing diambil dari halaman *website* sebagai berikut:

- Untuk scrapping data Alkitab diambil dari:
<http://www.sabda.org/alkitab/tb/?versi=tb&kitab=>
- Data kidung pujian diambil dari
<http://kidung.co/?title=KJ>
- Data buku ende diambil dari
<http://ende.sibirong.com/index.php?nomor=>

4. Code Engineering

Penggunaan MDD salah satu manfaat dan tujuannya adalah mempercepat pekerjaan tim pengembang dengan cara mengotomisasikan tugas pemrograman yang kompleks. Pada penerapan MDD, otomisasi *code program* tidak secara langsung menghasilkan *code program* yang sepenuhnya menghasilkan aplikasi yang utuh melainkan *code program* yang hanya menghasilkan kerangka *code program* dari desain yang dibuat yang sering disebut dengan *code skeletons*. Sedangkan untuk *logic* pada aplikasi yang akan dibangun hanya tergantung pada *requirement* yang diperoleh oleh tim *developer*. Pada penerapan

code engineering untuk aplikasi WISTA maka digunakan sebuah *tools Enterprise Architect (EAP)* version 12.1 dimana *tools* tersebut dapat melakukan otomisasi desain menjadi sebuah *code program*.

5. Penerapan MDD pada aplikasi WISTA

Penerapan MDD pada aplikasi WISTA menerapkan langkah-langkah pembangunan aplikasi sesuai dengan SDLC, yaitu *Inception*, *Elaboration*, *Construction*, dan *Transition*. Pada pelaksanaan pengembangan SDLC tersebut dilakukan beberapa kali iterasi untuk memastikan bahwa desain dan *code program* sesuai dan dapat diimplementasikan.

- *Inception*, pada proses ini beberapa kegiatan yang dilakukan untuk pencapaian rancangan desain pada pembangunan aplikasi WISTA. Rancangan dilakukan berdasarkan kebutuhan pengguna. Pada tahapan ini dilakukan *survey* kepada mahasiswa dan masyarakat khususnya yang menjadi target-target pengguna aplikasi WISTA seperti pengurus dan jemaat gereja.
- *Elaboration*, pada tahap ini dilakukan perancangan sistem sesuai dengan rencana yang telah dilakukan pada tahap *inception*. Perancangan yang dilakukan berupa desain *database*, fungsi-fungsi aplikasi, dan desain *requirement* pada aplikasi seperti e-r diagram, domain model, class diagram dan lain-lain.
- *Construction*, tahapan ini merupakan tahap untuk mengimplementasikan hasil desain yang telah dilakukan pada tahap *elaboration* sebelumnya.
- *Transition*, merupakan pengujian pada aplikasi yang dibangun sebelum aplikasi diserahkan kepada pelanggan atau *customer*. Pengujian aplikasi dapat dilakukan dengan 2 cara, yaitu pengujian secara *black box* dan *white box*. Pengujian secara *black box* merupakan pengujian yang dilakukan tanpa melihat bagian *code program* aplikasi melainkan hanya menguji apakah fungsi-fungsi yang dimiliki oleh aplikasi dapat berjalan dengan baik atau tidak. Sedangkan untuk pengujian secara *white box* merupakan sebuah pengujian yang tidak hanya memperhatikan fungsi melainkan memperhatikan *code program* yang biasanya diuji dengan menggunakan *test case*.

Penerapan MDD untuk aplikasi WISTA terdiri dari 3 kali iterasi yang dilakukan sesuai dengan kebutuhan masing-masing tahapan seperti:

- a. Iterasi 1, pada iterasi ini terjadi perubahan pada tahap *elaboration* seperti perubahan pada e-r diagram aplikasi. Perubahan yang terjadi adalah e-r diagram dibuat menjadi 2 versi sesuai dengan yang dibutuhkan, yaitu versi *web* dan *mobile*.

Pada e-r diagram yang dibuat pada versi *mobile* memiliki 15 tabel, sedangkan e-r diagram versi *web* terdiri dari 19 tabel.

- b. Iterasi 2, sama hal-nya dengan iterasi 1 bahwa tahapan yang dilakukan harus menggunakan sistem pengembangan MDD sesuai dengan SDLC yaitu *Inception*, *Elaboration*, *Construction* dan *Transition*. Tetapi pada iterasi 2 ini dilakukan implementasi untuk aplikasi WISTA sesuai dengan fungsi dan desain model yang telah dilakukan pada iterasi 1. Iterasi 2 terjadi karena pada iterasi 1 masih ada yang tidak sesuai dengan kebutuhan *user* baik dari segi desain model, *database* dan fungsinya. Yang diubah pada iterasi 2 adalah:

- *Use case scenario*, pada iterasi 1 sebelumnya, *use case scenario* dibuat secara manual. Maka pada iterasi 2 ini pembuatan *use case scenario* dibuat dengan menggunakan bantuan *tools* EAP.
- E-r diagram, perubahan dilakukan pada iterasi 2 dikarenakan adanya penambahan autentikasi pada aplikasi WISTA versi *web*.

- c. Iterasi 3, perubahan yang dilakukan pada iterasi ini adalah penambahan RBAC pada *database* untuk menangani hak akses pada masing-masing *user*. Untuk *use case diagram* terdapat perubahan pada fungsi yang dimiliki oleh jemaat gereja dan mengubah *mockup* aplikasi sesuai dengan desain aplikasi yang sebenarnya.

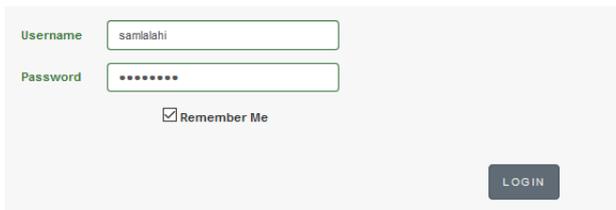
V. HASIL & PEMBAHASAN

Salah satu contoh hasil yang diperoleh untuk aplikasi WISTA versi *mobile* dan *web* adalah seperti berikut.

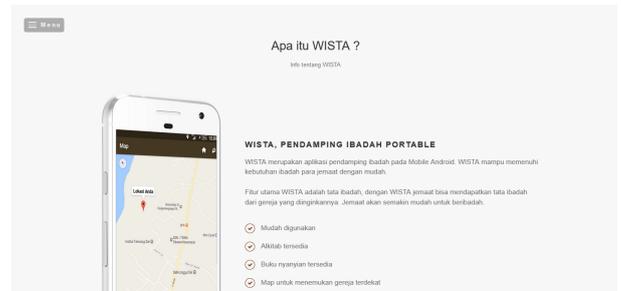
a. Versi *web*



Gambar 5 Tampilan Awal



Gambar 6 Tampilan Login



Gambar 7 Halaman Informasi WISTA

b. Versi *mobile*



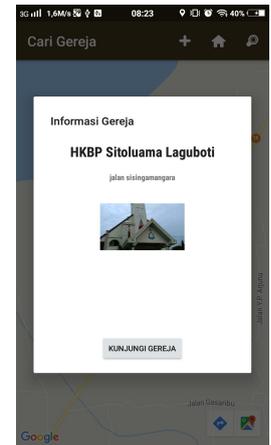
Gambar 8 Flashscreen



Gambar 9 Home



Gambar 10 Informasi Gereja



Gambar 11 Search Gereja

Proses pengembangan aplikasi WISTA telah diselesaikan dan menghasilkan sebuah aplikasi yang siap untuk diuji dan diserahkan kepada konsumen seperti pengurus dan jemaat gereja. Aplikasi WISTA telah dirilis dan diunggah di *PlayStore* pada tanggal 15 Mei 2017. Pada proses rilis, hal yang dilakukan tim pengembang sebelumnya adalah pengujian langsung kepada konsumen dengan pengujian secara *black box*.

Adapun proses pengujian yang dilakukan adalah sebagai berikut.

1. Pengujian dilakukan oleh tim pengembang sebelum disampaikan kepada mahasiswa, pengurus dan jemaat gereja.
2. Pengujian aplikasi dilakukan oleh mahasiswa IT Del dengan jumlah 30 orang mahasiswa. Kegiatan pengujian adalah sebagai berikut.
 - APK aplikasi WISTA diberikan dan diinstal pada *smartphone android* masing-masing mahasiswa.
 - Aplikasi digunakan ketika ibadah minggu dengan memperhatikan setiap fitur khususnya fitur tata ibadah.
 - Jika terdapat error ataupun bug, maka mahasiswa yang melakukan pengujian wajib melaporkan error tersebut kepada tim pengembang.
3. Pengujian aplikasi dilakukan oleh pengurus gereja dan jemaat gereja, terdapat 5 gereja yang menjadi target uji coba yang telah disebutkan sebelumnya pada bab VI. Pengujian dilakukan dengan tahapan yang sama dengan pengujian yang dilakukan oleh mahasiswa. Hal yang membedakan pengujian mahasiswa dan gereja adalah pengujian yang dilakukan oleh mahasiswa hanya sebatas pengujian aplikasi versi *mobile* sedangkan pengujian yang dilakukan oleh pengurus gereja adalah pengujian aplikasi versi *web* dan jemaat gereja menguji aplikasi versi *mobile*.
4. Pada proses pengujian dilakukan juga proses pengumpulan data berupa *questioner* untuk dapat memperoleh kesimpulan dari pembuatan aplikasi WISTA. Adapun hasil dari pengujian tersebut adalah sebagai berikut.
 - Responden penggunaan aplikasi WISTA terdiri dari mahasiswa, pelajar dan pekerja yang terdaftar di berbagai jenis gereja yang terdiri dari 75 responden
 - a. Pertanyaan 1, “Menurut anda, apa kesulitannya menggunakan tata ibadah yang biasa?”
Dari pertanyaan diatas, 75 responden menjawab pertanyaan tetapi ada beberapa jawaban yang paling sering muncul yang diberikan oleh responden, seperti berikut.
 - Terlalu banyak biaya yang dikeluarkan oleh gereja dan memakan waktu yang banyak juga untuk mempersiapkannya.
 - Jika kertas tata ibadah habis pada saat ibadah, maka tidak dapat dengan baik mengikuti ibadah.
 - Tulisan tidak jelas pada kertas tata ibadah
 - Jemaat mengalami kerepotan membolak balikkan kertas, membuka alkitab dan kidung pujian yang digunakan
 - b. Lirik atau teks untuk ayat alkitab maupun kidung pujian sering tidak dilampirkan pada kertas tata ibadah
 - c. Pertanyaan 2, “Menurut anda apakah fitur tata ibadah pada WISTA sangat berguna?”
Pada pertanyaan diatas banyak responden yang menyatakan bahwa aplikasi WISTA sangat berguna untuk digunakan pada saat ibadah, karena pada hasil *questioner* diperoleh 73 responden yang menyatakan sangat setuju dan 2 responden yang menyatakan biasa saja.
 - d. Pertanyaan 3, “Apakah anda memiliki saran untuk fitur tata ibadah pada aplikasi ini? Jika ada, tuliskan!”
Pada pertanyaan diatas terdapat hanya 62 responden yang menjawab pertanyaan ketiga, sedangkan untuk 13 responden lainnya tidak menjawab pertanyaan dengan mengosongkan kolom jawaban. Dari hasil persentase diatas diperoleh 3 (tiga) kesimpulan:
 - 22,4% menjawab dengan kata nihil (-) yang menyatakan bahwa saran tidak diberikan.
 - 7,9% menjawab dengan kata tidak ada yang menyatakan juga bahwa saran tidak diberikan.
 - 2,6% menjawab agar fitur tata ibadah juga dibuat untuk gereja-gereja lain seperti Khatolik dan Methodist.
 - Sedangkan untuk jawaban dengan persentase 1% menjawab dengan jawaban yang berbeda dari ketiga point diatas.
 - e. Pertanyaan 4, “Apakah design UI/UX aplikasi WISTA menarik?”
Pada pertanyaan diatas, diperoleh bahwa 75 responden menjawab pertanyaan diatas dengan 3 kategori jawaban, terdapat 66 responden menjawab bahwa desain UI/UX pada aplikasi WISTA menarik, 1 responden menjawab tidak dan 8 responden menjawab biasa saja.
 - f. Pertanyaan 5, “Apakah anda memerlukan manual book (petunjuk) untuk menggunakan aplikasi?”
Setelah menggunakan aplikasi WISTA, terdapat 63 responden yang menjawab tidak memerlukan manual *book* lagi untuk menggunakan aplikasi WISTA sedangkan 12 responden yang lain masih memerlukan

manual *book* untuk penggunaan aplikasi WISTA tersebut. Dapat disimpulkan bahwa aplikasi WISTA dapat digunakan tanpa harus adanya manual *book* sebagai petunjuk penggunaan aplikasi.

- f. Pertanyaan 6, "Fitur lain yang akan sering anda pakai selain tata ibadah?"

Untuk pertanyaan diatas, selain fitur tata ibadah yang disediakan pada aplikasi WISTA terdapat beberapa fitur yang akan sering digunakan oleh jemaat, seperti Alkitab, Buku Nyanyian, Renungan Harian, dan lain-lain. Pada pertanyaan diatas, jemaat boleh memilih lebih dari satu jenis fitur yang akan sering digunakan. Dari 75 responden yang menjawab *questioner* tersebut terdapat 53 responden yang menjawab akan menggunakan Alkitab, 47 responden menjawab buku nyanyian, 31 responden menjawab renungan harian, dan 10 responden menjawab fitur lain seperti PKJ (Pelengkap Kidung Jemaat), Mazmur, Puji Syukur dan Madah Bakti (biasanya digunakan oleh jemaat yang beragama katolik).

- g. Pertanyaan 7, "Menurut anda, berapa nilai yang anda akan berikan untuk aplikasi ini? (Skala 1-5)"

Keterangan:

1. Skala 5 : Sangat Setuju
2. Skala 4 : Agak Setuju
3. Skala 3 : Netral
4. Skala 2 : Agak Tidak Setuju
5. Skala 1 : Sangat Tidak Setuju

Berdasarkan data yang diperoleh pada saat survey berdasarkan *questioner* yang direspon oleh responden pada sub bab 7.4 Hasil Pengujian Aplikasi berdasarkan *Questioner* diatas, dapat disimpulkan bahwa:

- a. Aplikasi WISTA yang dilengkapi fitur tata ibadah setidaknya dapat membantu pengguna untuk menangani masalah yang biasanya terjadi seperti yang telah dijelaskan pada bab 7.4 Hasil Pengujian Aplikasi berdasarkan *Questioner* pada pertanyaan nomor 1 dan 2.
- b. Aplikasi WISTA juga diharapkan dapat menjadi aplikasi yang memiliki fitur yang lebih lengkap lagi seperti bahasa Alkitab yang dapat terdiri dari beberapa bahasa, kidung pujian yang terdiri dari beberapa versi, dan lain-lain.

Aplikasi WISTA dinilai sebagai aplikasi yang baik dimana hal ini dibuktikan dengan respon para responden dengan nilai antara skala 1 sampai dengan skala 3.

Dan untuk penerapan MDD pada aplikasi WISTA telah dilakukan berdasarkan tahapan pengembangan aplikasi. Pengaruh yang diperoleh dari penerapan MDD tersebut adalah sebagai berikut.

1. MDD lebih cepat, hal ini disebabkan karena pada proses pengembangan aplikasi MDD mengizinkan dilakukannya perubahan desain secara otomatis menjadi sebuah kerangka *code* program yang dapat membantu para *developer*.
2. MDD dapat mengurangi kesalahan, hal ini disebabkan karena hal yang paling mendasar dilakukan sebelum proses pengembangan aplikasi adalah memastikan bahwa desain atau pemodelan aplikasi telah benar dan dapat dilanjutkan ke tahap implementasi.
3. MDD dapat meningkatkan kualitas, jika MDD dapat mengurangi kesalahan, maka dengan demikian kualitas aplikasi yang dibangun menggunakan MDD dapat meningkat dengan baik.
4. MDD dapat menghasilkan aplikasi yang sensitif terhadap perubahan, hal ini disebabkan karena pada proses pengembangan aplikasi dengan menggunakan MDD sering terjadi perubahan baik dari tahap *inception* sampai dengan tahap *construction*. Hal ini disebabkan karena adanya ketidaksesuaian desain dengan implementasi aplikasi. Oleh karena itu, seperti yang telah disebutkan sebelumnya, penerapan MDD harus dilakukan dengan cara yang sangat teliti dan dikerjakan oleh sumber daya manusia yang baik.

VI. KESIMPULAN

1. MDD telah diterapkan dalam pengembangan aplikasi WISTA yang berfokus pada pemodelan desain secara detail. Pemodelan yang detail didapatkan dari iterasi yang dilakukan sebanyak 3 kali. Setiap iterasi berguna untuk menyempurnakan pemodelan desain dari iterasi yang sebelumnya. Pada setiap iterasi yang dilakukan menghabiskan waktu sekitar 2 bulan.
2. Penerapan MDD menghabiskan waktu yang lama pada tahap pemodelan desain, namun pengimplementasian desain ke dalam *code program* lebih mudah dan cepat karena *code skeleton* telah otomatis di-*generate*.
3. Aplikasi WISTA telah mendapatkan hal yang positif dari pengguna aplikasi karena dilihat dari data *survey* yang telah dilakukan terdapat 40 dari 75 responden yang sangat menyetujui bahwa aplikasi ini sangat berguna.

REFERENSI

- [1] Techspot, "Google shows off new version of Android, announces 1 billion active monthly users," Techspot, 2014.
- [2] "Okezone.com," Okezone, 19 September 2015. [Online]. Available: <http://techno.okezone.com/read/2015/09/19/57/1217340/2015-pengguna-smartphone-di-indonesia-capai-55-juta>. [Accessed 13

- October 2016].
- [3] "Emarketer," 19 09 2015. [Online]. Available: <http://www.emarketer.com/Article/Asia-Pacific-Boasts-More-Than-1-Billion-Smartphone-Users/1012984>. [Accessed 13 October 2016].
 - [4] "Statista," 15 October 2015. [Online]. Available: <https://www.statista.com/statistics/276623/number-of-apps-available-in-leading-app-stores/>. [Accessed 13 October 2016].
 - [5] "google play store," google, [Online]. Available: <https://play.google.com/store/apps/category>. [Accessed 13 October 2016].
 - [6] A. Dennis, B. H. Wixom and R. M. Roth, *System Analysis and Design*, John Wiley & Sons, International Student Edition, 2010.
 - [7] A. Dennis, B. H. Wixom and R. M. Roth, *System Analysis and Design*, John Wiley & Sons, International Student Edition, 2010.
 - [8] S. K. Pandey, G. P. Singh and D. V. Kansal , "Study of Object Oriented Analysis and Design Approach," *Journal of Computer Science* , vol. 7, no. ISSN 1549-3636 , 2011.
 - [9] H. Berlioz, "Object-Oriented Analysis and Design," pp. 3 - 16.
 - [10] S. J. Mellor, A. N. Clark and T. Futagami, "Model-Driven Development," IEEE Computer Society, 2003, p. 14.
 - [11] C. Atkinson and T. Kuhne, "Model-Driven Development: A Metamodeling Foundation," in *Model-driven Development*, IEEE Computer Society, 2003, pp. 36-41.
 - [12] "Firebase," Google, [Online]. Available: <https://firebase.google.com/docs/>. [Accessed 20 01 2017].