

IMPLEMENTASI KONSEP NO PROGRAMMING DALAM MEMBANGUN PERANGKAT LUNAK EMAIL BERBASIS ANDROID WIREFRAME MENGGUNAKAN JIM PROTOTYPER

FIQIH ISMAWAN

Program Studi Informatika

Fakultas Teknik dan Ilmu Komputer

Universitas Indraprasta PGRI

Jl. Nangka No. 58 C, Tanjung Barat, Jagakarsa, Jakarta Selatan 12530

Email: vQ.ismaone@gmail.com

Abstrak. Kebutuhan sistem dalam meningkatkan pelayanan publik yang baik pada pengguna, dilakukan dengan membuat berbagai macam inovasi khususnya pada sistem *email*. *E-mail* sendiri digunakan untuk menunjang kinerja dan birokrasi antara berbagai pihak dengan sesama pengguna. Salah satu bentuk penerapan sistem dalam pelayanan publik yaitu dengan adanya sistem *e-mail* yang berbasis *website* ataupun *mobile android*. Berbeda dengan sistem aplikasi lainnya, sistem *e-mail* biasanya dibentuk berdasarkan ketentuan data akun yang dimiliki oleh setiap pengguna yang terdaftar oleh sistem *email*. Dalam pengembangan aplikasi *e-mail* berbasis *website* atau *mobile android* sendiri, seringkali para *develepor* menggunakan model *prototype* dan *wireframe* dalam merancang sistem yang akan dibangun. Banyak aplikasi *prototype* yang dapat digunakan para *developer* dalam merancang sistem *e-mail* khususnya berbasis *mobile android*. Dari berbagai macam aplikasi *prototype* tersebut diantaranya adalah aplikasi *Justinmind Prototyper*. Dengan banyaknya pilihan aplikasi *software* yang dapat digunakan para *developer* dalam pengembangan aplikasi membuat mereka harus memilih akan menggunakan aplikasi *prototype* mana yang sesuai dengan sistem yang akan dibangun. *Prototype* sendiri ada yang berbentuk *low-fidelity* maupun *high-fidelity*, perbedaan dari keduanya yaitu terdapat pada tingkat kerincian dan interaksi yang dihasilkan. *High-fidelity* sendiri menggambarkan sistem yang lebih rinci dengan adanya interaksi yang dapat disimulasikan. Penelitian ini bertujuan membantu para *developer* atau siapa saja yang akan menggunakan aplikasi *prototype* dalam merancang suatu sistem khususnya *e-mail* berbasis *mobile android*, khususnya aplikasi *prototype Justinmind Prototyper* yang banyak digunakan dalam merancang desain suatu sistem. Metode yang akan digunakan adalah metode berbasis *wireframing* dan *prototyping* dalam mengimplementasikan dan membangun aplikasi sistem dengan bantuan *software Justinmind Prototyper*. Hasil dari penelitian ini berupa aplikasi email berbasis android yang dibuat menggunakan *software Justinmind Prototyper*.

Kata Kunci: *tanpa koding, wireframe, prototyping, android, email sistem,*

Abstract. *System requirements in improving good public services for users are carried out by making various kinds of innovations, especially in the email system. E-mail itself is used to support performance and bureaucracy between various parties with fellow users. One form of system implementation in public services is the existence of an e-mail system based on a website or mobile android. Unlike other application systems, e-mail systems are usually formed based on the provisions of the account data held by each user registered by the email system. In developing e-mail application based on website or mobile android itself, often the developers use prototype and wireframe models in designing the system to be built. Many prototype applications that can be used by developers in designing e-mail systems especially based on mobile android. Of the various prototype applications including the Justinmind Prototyper application. With so many choices of software applications that can be used by developers in developing applications, they have to choose which prototype application will suit the system to be built. Prototype itself is in the form of low-fidelity or high-fidelity, the difference between the two is that there is a level of detail and interaction that is*

produced. High-fidelity itself describes a more detailed system with interactions that can be simulated. This study aims to help developers or anyone who will use a prototype application in designing a system especially e-mail based on mobile android, especially the Justinmind Prototyper prototype application that is widely used in designing a system design. The method that will be used is a wireframing-based method and prototyping in implementing and building system applications with the help of Justinmind Prototyper software. The results of this study in the form of an android-based email application created using Justinmind Prototyper software.

Keywords: *no coding, wireframe, prototyping, android, system emai*

PENDAHULUAN

Berkembangnya dunia pemrograman saat ini yang semakin hari semakin cepat, instan dan praktis dengan mengandalkan beberapa konsep model dan metode yang beragam, maka kebutuhan sistem semakin menuntut seorang *developer* sistem untuk dapat bekerja cepat dalam mendesain, membangun prototipe, dan merancang seluruh kebutuhan sistem yang ada. Beragam jenis alat pendukung ditawarkan dalam dalam mengembangkan sistem baik berupa perangkat lunak dan pendukung lainnya, semua itu dibangun sesuai kebutuhan sistem yang ada.

Kebutuhan sistem dalam meningkatkan pelayanan publik yang baik pada pengguna, dilakukan dengan membuat berbagai macam inovasi khususnya pada sistem *email*. *E-mail* sendiri digunakan untuk menunjang kinerja dan birokrasi antara berbagai pihak dengan sesama pengguna. Salah satu bentuk penerapan sistem dalam pelayanan publik yaitu dengan adanya sistem *e-mail* yang berbasis *website* ataupun *mobile android*. Berbeda dengan sistem aplikasi lainnya, sistem *e-mail* biasanya dibentuk berdasarkan ketentuan data akun yang dimiliki oleh setiap pengguna yang terdaftar oleh sistem *email*. Dalam pengembangan aplikasi *e-mail* berbasis *website* atau *mobile android* sendiri, seringkali para *developepor* menggunakan model *prototype* dan *wireframe* dalam merancang sistem yang akan dibangun. Banyak aplikasi *prototype* yang dapat digunakan para *developer* dalam merancang sistem *e-mail* khususnya berbasis *mobile android*. Dari berbagai macam aplikasi *prototype* tersebut diantaranya adalah aplikasi *Justinmind Prototyper*.

Dengan banyaknya pilihan aplikasi *software* yang dapat digunakan para *developer* dalam pengembangan aplikasi membuat mereka harus memilih akan menggunakan aplikasi *prototype* mana yang sesuai dengan sistem yang akan dibangun (Wisesa, 2015). Penelitian ini merujuk pada sebuah buku Kupas Tuntas Istilah {No Programming Diterapkan pada *JIM Prototyper* (Fiqih, 2016). Selain itu penelitian ini merujuk pada penelitian dengan objek *e-government mobile* dan web, perangkat lunak *Justinmind* dan *Mockplus* (Albaar, 2016), parameter komparasi menggunakan *functionality*, *usability* dan *efficiency* dari *International Standard Organization* (ISO/IEC, 2000). Perbedaan penelitian ini adalah objek penelitian lain adalah terfokus pada pengembangan aplikasi perangkat lunak dengan mengimplementasikan konsep *no programming* yang dikenalkan oleh *justinmind prototyper* dalam membangun perangkat lunak email berbasis android *wireframe* dan *prototyping* yang *powerful*, *user friendly* dan *user expereince*. Dengan mengenalkan konsep ini diharapkan membantu para *developer* sistem atau siapa saja yang akan menggunakan aplikasi *prototype* dalam merancang suatu sistem *wireframing* atau *prototyping*. Penelitian ini penting dilakukan karena bertujuan membantu para *developer* atau siapa saja yang akan menggunakan aplikasi *prototype* dalam merancang suatu sistem khususnya *e-mail* berbasis *mobile android*, khususnya aplikasi *prototype Justinmind Prototyper* yang banyak digunakan dalam merancang desain suatu sistem.

Pengertian *Prototyping*

Prototyping adalah suatu metode yang memperkenankan *stakeholder* berinteraksi untuk membayangkan sebuah produk, bertujuan agar mendapat beberapa pengalaman realistik dan dapat melakukan eksplorasi untuk membayangkan penggunaan dari sebuah produk. *Fidelity* adalah

ukuran untuk membedakan tingkat interaksi, tampilan visual dan detail dari suatu *prototype* (Walker et al, 2002). *Prototyping* dibagi menjadi dua sifat yaitu *low-fidelity prototyping* dan *high-fidelity prototyping*. Ciri - ciri *low-fidelity prototyping* adalah *prototyping* hasil rancangan tidak mempresentasikan produk final. Ada beberapa cara untuk menerapkan *low-fidelity prototyping* dengan menggunakan *metode storyboarding, sketching, prototyping with index card dan wizard of Oz*. *High-fidelity prototyping* adalah produk yang diharapkan menjadi produk final dan lebih mempresentasikan produk seperti produk akhir (Preece, 2002).

Pengertian *Justinmind Prototyper*

JIM (Justinmind) Prototyper adalah suatu *tool* yang mengadopsi konsep cepat dan instan dalam membuat prototipe, serta memudahkan Anda untuk membuat sebuah simulasi secara interaktif dan akurat dari aplikasi yang Anda kembangkan sebelumnya (Fiqih, 2016:2).

Melalui *JIM Prototyper*, dapat dengan mudah membuat aplikasi apapun untuk perancangan prototipe yang dibuat. Selain itu, dengan *JIM Prototyper* juga dapat mengeksport prototipe tersebut ke dalam bentuk format HTML untuk menunjukkan bahwa prototipe tersebut dapat diakses secara *online*. Serta secara otomatis, prototipe dapat menghasilkan semua file dokumentasi dalam bentuk Open Office atau dalam bentuk *Microsoft Word*. Semua itu dapat dilakukan tanpa membutuhkan pengetahuan *coding* pemrograman sedikitpun.

Adapun beberapa karakteristik *JIM Prototyper* yang di rujuk oleh sebuah buku kupas tuntas istilah *{No Programming}* (Fiqih, 2016:2) sebagai berikut:

1. *No Programming*, di mana *justinmind prototyper* merupakan sebuah *tool* yang intuitif. Kebutuhan pada *tool* ini, semua dipermudah dan disajikan dalam kemasan yang praktis. Ketika ingin membuat prototipe pada *tool* ini, dapat lakukan hanya dengan menyeret dan meletakkan komponen-komponen yang dibutuhkan atau disebut teknik *Drag & Drop*. Kemudian dapat melakukan sebuah interaksi dari palet-palet yang ada ke area kerja yang diinginkan. Semua itu dapat diimplementasikan dengan istilah *NO PROGRAMMING*.
2. Simulasi secara instan. Hanya dengan satu klik pada tombol "*Simulate*", maka sudah dapat melihat hasil prototipe aplikasi yang sudah dibuat dalam satu waktu.
3. Terdokumentasi secara otomatis. Jika ingin prototipe yang dibuat dilengkapi dengan semua *file* dokumen yang dibutuhkan, maka *JIM Prototyper* dapat menghasilkan *file* dokumentasi *prototype*.
4. Ekspor dalam format HTML. File prototipe dapat di ekspor ke dalam bentuk HTML. Tujuan ekspor ini agar para pengguna yang menggunakan prototipe aplikasi dapat memberikan *feedback* (umpan balik) ketika menggunakan prototipe aplikasi tersebut, serta mereka dapat memberikan pendapat dan pandangan positif yang sifatnya membangun.

METODE

Penelitian ini di bangun oleh sebuah konsep *No Programming* dengan metode *prototyping* dalam mengimplementasikan dan membangun aplikasi perangkat lunak email berbasis android menggunakan bantuan *software Justinmind Prototyper*.

Paradigma *prototyping* diawali dengan komunikasi antara pengembang dan pengguna yang bertemu dan mendefinisikan sasaran-sasaran menyeluruh dari perangkat lunak yang akan dibangun, mengidentifikasi kebutuhan apa saja yang diinginkan. Iterasi *prototyping* direncanakan secara cepat, demikian juga pemodelan dalam bentuk rancangan segera dibuat. Perancangan yang cepat berfokus pada penggambaran aspek-aspek perangkat lunak yang akan dilihat oleh pengguna, seperti tampilan *interface* antara *user* dengan sistem, atau format tampilan *output*. Rancangan yang cepat ini akan membawa ke arah pembuatan program (konstruksi) dari *prototype*. *Prototype* diserahkan dan dievaluasi oleh pengguna. *Feedback* dari

pengguna digunakan untuk memperbaiki kriteria kebutuhan perangkat lunak (Pressman, 2005). Berdasarkan tingkat ketelitiannya, metode *Prototyping* dibagi menjadi dua karakteristik yaitu *low fidelity Prototyping* dan *high fidelity Prototyping*. *Low Fidelity Prototyping* yaitu metode *Prototyping* dengan tingkat ketelitian yang rendah, perancangan lebih fokus kepada konsep dan *layout* layar, menggunakan perangkat konvensional dalam pembuatan prototipe seperti *storyboard* dan sketsa. Sedangkan *High Fidelity Prototyping* yaitu metode *Prototyping* dengan ketelitian tinggi dan hasil *Prototyping* lebih mengacu kepada gambaran akhir sistem. Pada *Prototyping* jenis ini, prototipe yang dihasilkan memungkinkan kita dapat berinteraksi penuh dengan desain antarmuka aplikasi.

HASIL DAN PEMBAHASAN

Solusi dari permasalahan adalah dengan mengimplementasikan konsep *No Programming* berbasis *wireframing* dan *prototyping* dalam membuat aplikasi *email device* berbasis android. Berdasarkan metode penelitian yang digunakan adapun tahapan proses yang dilakukan yakni:

Communication

Tujuan umum dari kebutuhan sistem aplikasi *email* berbasis android ini adalah memberikan solusi bagi para *developer* sistem dalam membangun sebuah sistem sesuai kebutuhan sistem yang pengguna inginkan dengan membuat berbagai macam inovasi khususnya pada sistem *email*.

Adapun kebutuhan sistem aplikasi *email* berbasis android pada penelitian ini yakni membuat *splash screen*, membuat *login screen*, membuat *inbox screen*, dan simulasi prototipe sistem aplikasi *email* berbasis *mobile* android.

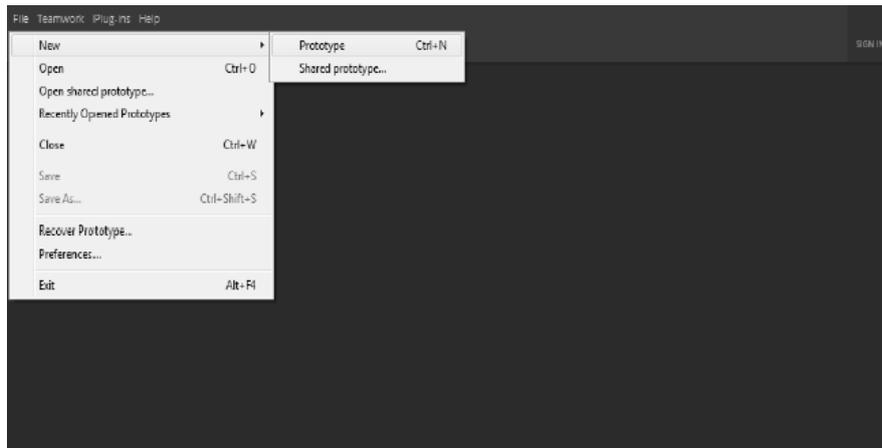
Quick Plan

Software yang digunakan adalah *justinmind prototyper* sebagai alat pendukung pembuatan aplikasi perangkat lunak *email* berbasis *mobile* android. Pada tool JIM Prototyper, *developer* dapat membuat berbagai prototipe aplikasi program yang dibutuhkan, sesuai dengan kebutuhan sistem. Semua tersedia di panel pada kanvas editor. Dalam panel-panel tersebut kaya akan *widget*, *data masters*, *masters*, *screens*, *template*, *variables*, *outline*, *navigator*, *change view*, *event type*, *event toolbar*, dan *interactions* yang dapat dikembangkan selanjutnya. Manfaatkan kemampuan serta konsep yang terstruktur, agar prototipe yang dibuat semakin menarik dan menjadi daya tarik pengguna lainnya. Berikut jenis-jenis aplikasi yang dapat dibuat serta dikembangkan dengan tool *Justinmind Prototyper*: *Aplikasi Web Apps*, *Aplikasi Mobile iPhone 4*, *Aplikasi Mobile iPhone 5*, *Aplikasi Mobile iPhone 6*, *Aplikasi Mobile iPhone 6+*, *Aplikasi iPad*, *Aplikasi Mobile Android*, *Aplikasi Android Tablet*, *Aplikasi Costum Tablet*, *Aplikasi Google Glass*, *Aplikasi Apple Watch (Versi 6.8)*

Modelling Quick Design

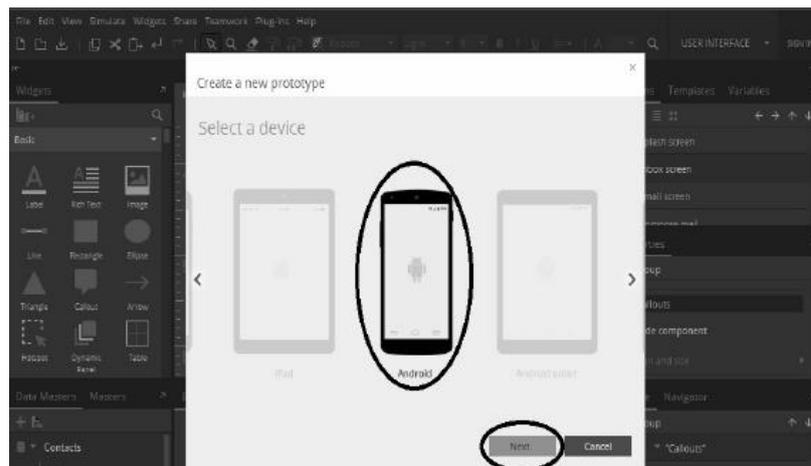
Ada beberapa tahapan untuk mengimplementasikan konsep *no programming* dalam membuat aplikasi *email* berbasis *mobile* android pada *justinmind prototyper* sebagai berikut:

Pada tahapan ini *developer* dapat membuat *prototype* baru dengan cukup mengarahkan *pointer mouse* ke menu *File > New > Prototype* atau dengan kombinasi tombol pada *keyboard* dengan menekan tombol *Ctrl+N*.

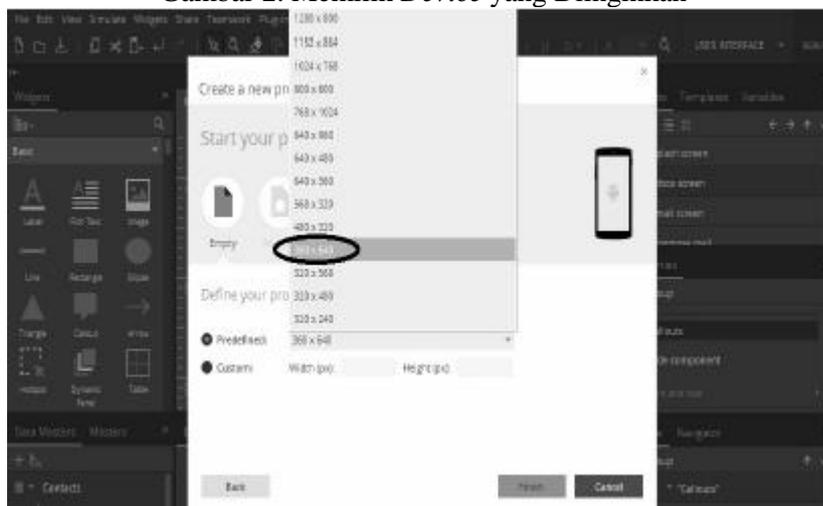


Gambar 1. Membuat Prototipe Baru Pada *JIM Prototyper*

JIM Prototyper menyediakan beberapa *device*, yaitu untuk *platform Website, Google Glass App, Android Phone, Android Custom, iPhone 4 5s 6, 6 Plus, iPad*, dan versi terbaru *Apple Watch*.



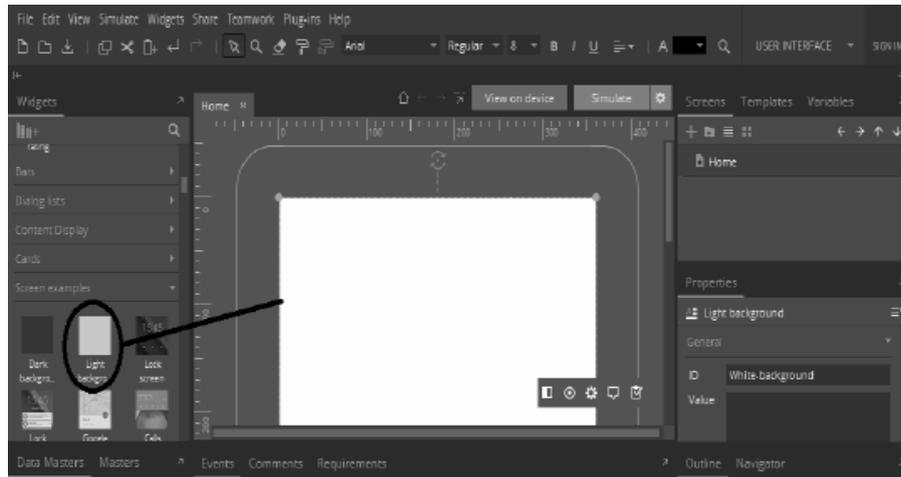
Gambar 2. Memilih *Device* yang Diinginkan



Gambar 3. Mendefinisikan Ukuran Kanvas

Construction of Prototype Membuat *Splash Screen*

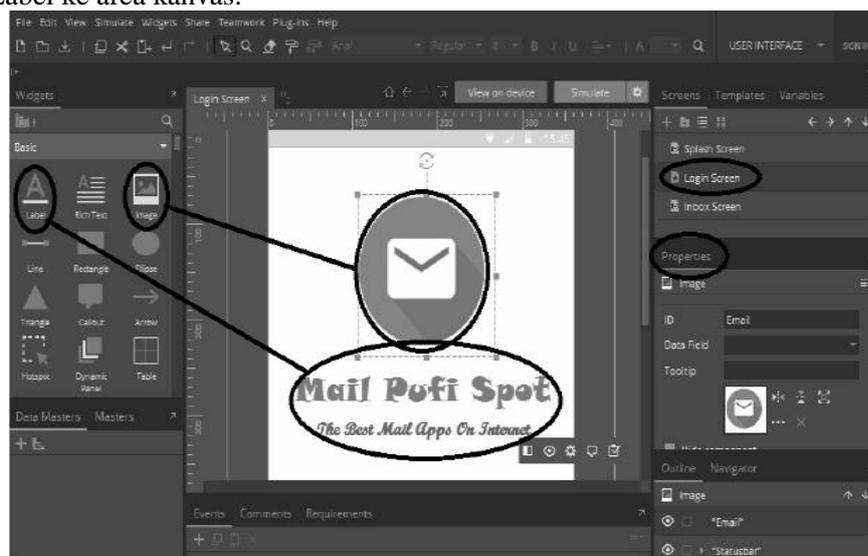
Dimulai dari *Splash Screen*, *developer* sistem dapat memanfaatkan *widget Light Background* sebagai dasar latar *background* prototipe. Cukup melakukan teknik *drag & drop* *widget* tersebut ke area kanvas yang tersedia.



Gambar 4. Drag & Drop Widget Light Background Splash Screen

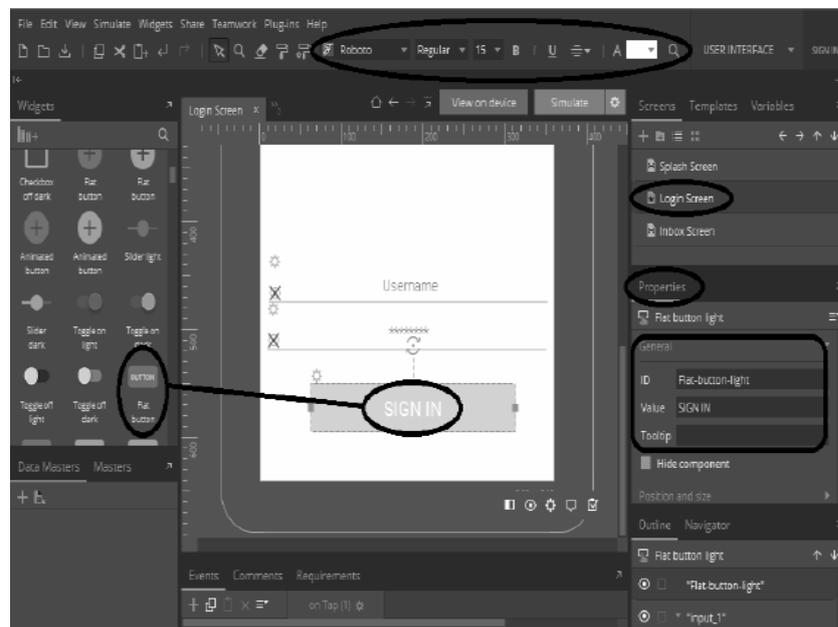
Membuat *Login Screen*

Login merupakan bagian dari sebuah tahapan proses pertama dalam sebuah aplikasi pada saat aplikasi meminta data pengguna. Biasanya berupa *username* dan *password* untuk proses validasi pencocokan kepemilikan akun pengguna. Implementasi *form login* saat ini sudah banyak diterapkan oleh beberapa aplikasi, baik *online* maupun *offline*. Selanjutnya, *developer* sistem dapat desain terlebih dahulu *screen* yang terdapat pada *Splash Screen* yang telah dibuat sebelumnya. Dengan melakukan teknik *drag & drop* sebuah *widget Image* dan *Label* ke area kanvas.



Gambar 5. Drag & Drop Widget Label dan Image

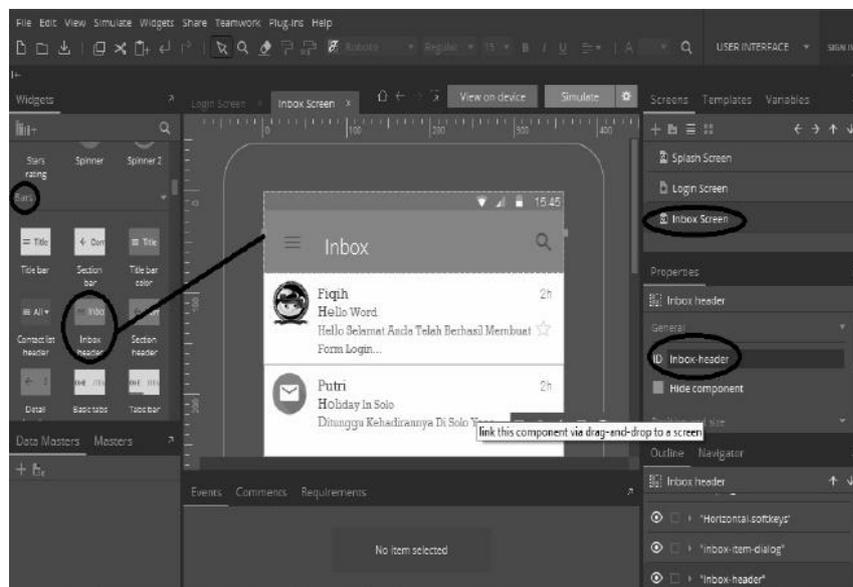
Kemudian *developer* sistem dapat melakukan teknik *drag widget Flat Button* pada bagian bawah *widget Input Username* dan *Input Password* yang telah dibuat, dan drop ke area kanvas



Gambar 6. Drag & Drop Widget Flat Button Sign In Login Screen

Membuat Inbox Screen

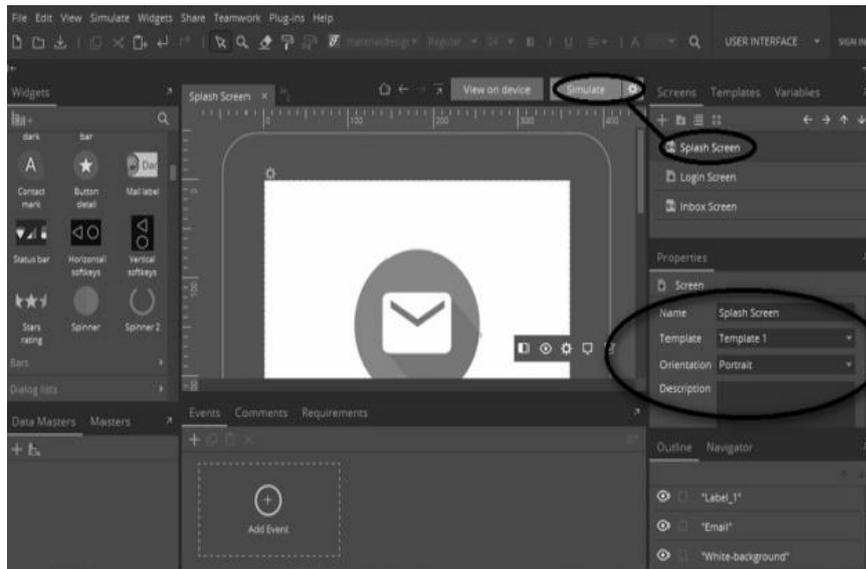
Developer sistem dapat melakukan teknik drag widget Light Background. Kemudian lakukan teknik drop ke area kanvas sebagai dasar latar background prototipe.



Gambar 7. Membuat Desain Prototype Inbox Screen

Simulasi Prototype Sistem Aplikasi Email Berbasis Mobile Android

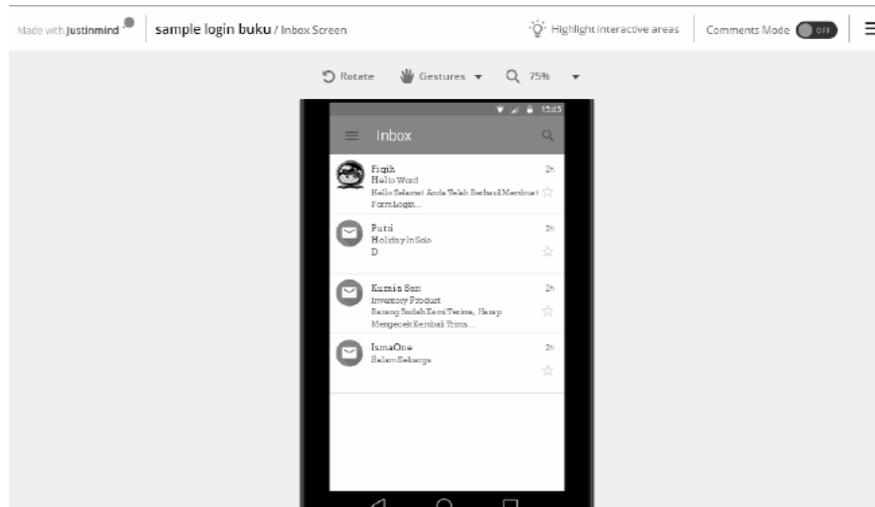
Tahapan terakhir, developer sistem dapat mensimulasikan prototipe yang telah dibangun. Seluruhnya dapat disimulasikan dalam satu template, satu area kanvas, satu screen tanpa emulator dan tool tambahan lainnya.



Gambar 8. Simulasi Prototipe *Login Email Device* Android



Gambar 13. Hasil Simulasi Prototipe *Login Email Device* Android



Gambar 14. Hasil Simulasi Prototipe *Inbox Email Device* Android

Hasil Pengujian Prototipe Perangkat Lunak

Untuk memastikan bahwa perangkat lunak yang dibuat memiliki standar minimal kualitas, maka salah satu metoda untuk pengukuran kualitas perangkat lunak secara kuantitatif adalah metoda SQA (*Software Quality Assurance*).

Tabel 1. Hasil *Metric of Software Quality Assurance (SQA)*

No	Metrik	Deskripsi	Bobot
1	<i>Auditability</i>	Memenuhi standar atau tidak	0.1
2	<i>Accuracy</i>	Keakuratan komputasi	0.15
3	<i>Completeness</i>	Kelengkapan	0.1
4	<i>Error Tolerance</i>	Toleransi terhadap kesalahan	0.1
5	<i>Execution</i>	Kinerja eksekusi	0.1
6	<i>Efficiency</i>	Kemudahan untuk dioperasikan	0.15
7	<i>Simplicity</i>	Kemudahan untuk difahami	0.15
8	<i>Training</i>	Kemudahan pembelajaran fasilitas Help	0.15

Tabel di atas merupakan 8 kriteria yang dapat digunakan untuk mengukur kualitas sebuah perangkat lunak secara kuantitatif. Dari 8 komponen tersebut akan dibuat 8

pertanyaan untuk angket yang akan disebarakan kepada 5 orang pengamat yang merupakan *user* yang diambil secara acak.

Tabel 2. Hasil Evaluasi SQA

User	Skor Metrik								Skor
	1	2	3	4	5	6	7	8	
#1	80	90	80	80	85	90	90	100	88
#2	85	80	80	78	80	85	85	80	81.8
#3	80	90	85	95	80	80	90	95	87.25
#4	90	85	80	90	85	78	80	85	83.7
#5	85	75	88	90	85	83	95	90	86.25
Rata-Rata									85.4

Tabel di atas merupakan hasil angket yang dilakukan pada 5 orang pengamat yang berperan sebagai *user* dan diambil secara acak.

$$\text{Skor} = (\text{SkorAuditability} \times 0.1) + (\text{SkorAccuracy} \times 0.15) + (\text{SkorCompleteness} \times 0.1) + (\text{SkorErrorTolerance} \times 0.1) + (\text{SkorExecutionEfficiency} \times 0.1) + (\text{SkorOperability} \times 0.15) + (\text{SkorSimplicity} \times 0.15) + (\text{SkorTraining} \times 0.15)$$

Perhitungan pada skor *user* :

$$(80 \times 0.1) + (90 \times 0.15) + (80 \times 0.1) + (80 \times 0.1) + (85 \times 0.1) + (90 \times 0.15) + (90 \times 0.15) + (100 \times 0.15) \\ = 8 + 13.5 + 8 + 8 + 8.5 + 13.5 + 13.5 + 15 \\ = 88$$

Perhitungan rata-rata skor pada semua *user* :

$$= \text{total skor} / \text{jumlah user} \\ = 427 / 5 \\ = 85.4$$

Nilai optimal untuk sebuah perangkat lunak yang memenuhi standar kualitas berdasarkan uji SQA adalah 80. Sedangkan skor rata-rata hasil pengujian prototipe pada perangkat lunak ini adalah 85.4. Sehingga dapat disimpulkan bahwa kualitas perangkat lunak prototipe sistem aplikasi *email* berbasis *mobile* android ini cukup baik.

PENUTUP

Simpulan

Berdasarkan hasil penelitian ini dapat disimpulkan bahwa manfaat konsep *No Programming* menjadi solusi saat ini dalam menjawab kebutuhan sistem yang akan dibangun oleh para *developer* sistem. Konsep ini mengadopsi metode *prototyping* yang cepat, praktis, interaktif dan *powerfull* dalam mendesain dan mengembangkan sebuah sistem aplikasi berbasis *wireframe*. Dikarenakan konsep ini mengadopsi metode tersebut, maka aplikasi *Justinmind Prototyper* merupakan aplikasi yang sangat cocok untuk pengembangan aplikasi dan memiliki keunggulan kompetitif pada generasi *wireframe* seperti saat ini serta kaya akan *widget library*, simulasi dalam satu *workspace* dan lain-lain.

Saran

Saran untuk penelitian selanjutnya bahwa penelitian ini hanya mengeksplorasi sedikit tentang keunggulan *tool Justinmind Prototyper* yang mengadopsi konsep *no programming* dengan metode *prototyping*. Aplikasi yang dibuat masih dikatakan belum sempurna dan butuh eksplorasi lebih mendalam lagi terutama pada konsep dan ide-ide kreatif dalam membangun sebuah sistem aplikasi instan. Ini hanya sebagai pengenalan dan contoh kecil dari hasil eksplorasi peneliti dalam mengenalkan sebuah konsep *no programming* dalam *tool justinmind* yang pernah di tulis juga dalam sebuah bukunya yang berjudul *kupas tuntas*

istilah no programming (Fiqih, 2016). Sebagai jangka panjang bahwa dibutuhkan beberapa penyempurnaan, menambahkan fitur menu-menu baru untuk pengembangan aplikasi dan penambahan model dari sistem aplikasi ini agar lebih menarik.

DAFTAR PUSTAKA

- Ismawan, Fiqih. 2016. *Kupas Tuntas Istilah {No Programming Diterapkan pada JIM PROTOTYPER}*. Jakarta: Elexmedia Komputindo.
- Albaar, A. H. (2016) *Komparasi Perangkat Lunak High - Fidelity Prototyping : Justinmind dan Mockplus dalam Pengembangan Aplikasi Web dan Mobile E - government (Studi Kasus : LPSE)*. Universitas Islam Indonesia.
- Panhale, M. (2015). *Begining Hybrid Mobile Application Development*. Pune: Appress
- Preece, J., Rogers, Y., & Sharp, H. (2002). *Interaction Design: Beyond Human-Computer Interaction. Book* (1st ed.). A John Wiley & Sons.
- Pressman, B., *Software Engineering, McGraw-Hill, 2005*
- Wisesa, D. U. (2015) *Komparasi Perangkat Lunak High-Fidelity Prototyping : Axure dan Balsamiq dalam Pengembangan Aplikasi Web dan Mobile Commerce*. Universitas Islam Indonesia.
- Walker, M., Takayama, L., & Landay, J. A. (2002). High-Fidelity or Low-Fidelity, Paper or Computer? Choosing Attributes when Testing Web Prototypes. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 46(5), 661-665