

Rancang Bangun Aplikasi Mobile IniAtauItu sebagai Media Tanya Jawab Berbasis Komunitas

Albert Sanada Yapri¹, Stephanus Eko Wahyudi², Nehemia Sugianto³

Abstrak— Dalam menentukan pilihan terkait *lifestyle* dan aktivitas sehari-hari, umumnya seseorang dapat mengalami kesulitan dikarenakan oleh rasa bimbang, misalnya dari dua buah pilihan baju yang ada, manakah yang lebih bagus dipakai. Pertanyaan-pertanyaan demikian bersifat subjektif, dimana informasi dapat diperoleh dengan lebih maksimal apabila menggunakan *social search* (misalnya dengan media sosial) ketimbang *search engine* yang menyediakan data dan fakta. Akan tetapi, pada media sosial terdapat beberapa keterbatasan pula dan tidak fokus pada topik *lifestyle*. Tugas akhir ini berupaya untuk memberikan solusi berupa aplikasi *mobile* IniAtauItu untuk membantu pengguna menentukan pilihan terkait *lifestyle* dengan menerima *feedback* dari orang lain berupa preferensi atau komentar. Aplikasi dibuat menggunakan arsitektur klien-server, dimana aplikasi berbasis Android bertindak sebagai klien, dan Firebase digunakan untuk mendukung sisi server. Hasil akhir menyimpulkan bahwa aplikasi IniAtauItu dapat dimanfaatkan sebagai salah satu alternatif untuk menanyakan pilihan-pilihan terkait *lifestyle*, serta dapat membantu meningkatkan rasa yakin dan lebih mempercepat pengguna untuk menentukan pilihan akhir setelah menerima sejumlah *feedback* dari komunitas.

Kata Kunci: Pilihan, *Lifestyle*, Pertanyaan Subjektif, Aplikasi Mobile, Android.

Abstract— In making choices about lifestyle and daily activities, most people face difficulties due to hesitancy, for instance: from two choices of different shirts, which one is better to wear? These type of questions are subjective, and it has been proven that it is more suitable to do the social search (by using social media for example) to gain subjective information, rather than using search engine, which mostly dominated by objective facts. However, there are still several drawbacks in social media, and the topic is not focused on lifestyle. Therefore, this final assignment aims to provide a solution, which is a mobile app called IniAtauItu to overcome those drawbacks as well as to help users choose after receiving

preferences and comments from community. Client-server architecture is used in developing this app, with Firebase utilized to handle the backend part. The result concludes that IniAtauItu app can be used as an alternative to ask question and choices about lifestyle, and to help increasing user's confidence in making choices after receiving a number of feedbacks from the community.

Keywords: Choice, Lifestyle, Subjective Questions, Mobile Application, Android.

I. PENDAHULUAN

Dalam menentukan keputusan, baik itu keputusan besar maupun kecil, seringkali manusia mengalami kesulitan dikarenakan oleh adanya ketidakpastian [1]. Ketidakpastian tersebut dapat menimbulkan berbagai prasangka maupun reaksi emosional yang dapat menyebabkan ketidaktentuan dalam memutuskan sesuatu. Ketidaktentuan yang diakibatkan oleh ketidakpastian tersebut dapat mendatangkan penyesalan pula bagi si pembuat keputusan [2]. Maka, kegiatan menentukan keputusan terkadang dipersepsikan sebagai sesuatu yang sulit, secara negatif sarat akan emosi, dan menyusahkan [3].

Hal tersebut dapat terlihat pada saat kita ingin memutuskan sesuatu, seringkali kita dihadapkan dengan beberapa pilihan atau opsi yang membuat kita bimbang. Saat sebuah opsi lebih baik dibandingkan opsi lainnya dari segala segi, tentunya tidak akan muncul konflik dan kegiatan memutuskan menjadi mudah. Akan tetapi, ketika masing-masing opsi memiliki keuntungan dan kerugian yang signifikan, hal tersebut memunculkan konflik yang kemudian mendorong seseorang untuk menunda mengambil keputusan [4]. Dari survei yang telah diberikan kepada 60 responden, 83% juga setuju bahwa mereka mengalami kebingungan saat harus memilih salah satu dari beberapa opsi yang mereka miliki sehingga sulit untuk memutuskan, misalnya antara lain, dari dua pilihan baju yang ada, manakah yang lebih anggun untuk dipakai ke pesta, kemudian dari beberapa pilihan kado yang ada manakah yang lebih cocok untuk diberikan kepada teman, atau, dari dua atau tiga buah *selfie* yang ada manakah yang akan mendapat lebih banyak *like* ketika diupload ke media sosial, serta masih banyak lainnya.

Apabila kita mengamati contoh-contoh hal yang membuat para responden bimbang yang disebutkan sebelumnya, dapat terlihat bahwa pertanyaan-pertanyaan tersebut lebih membutuhkan jawaban yang berupa opini-opini atau pendapat orang lain, atau yang disebut dengan informasi subjektif. Telah dibuktikan bahwa untuk memperoleh informasi subjektif para pengguna internet sebagian besar lebih tertarik untuk melakukan *social search*

¹ Mahasiswa, Program Studi Teknik Informatika Fakultas Industri Kreatif Universitas Ciputra, UC Town, Citraland, Surabaya 60219 INDONESIA (telp: 031-745 1699; fax: 031-745 1698; e-mail: asanada@student.ciputra.ac.id)

² Dosen, Program Studi Teknik Informatika Fakultas Industri Kreatif Universitas Ciputra, UC Town, Citraland, Surabaya 60219 INDONESIA (telp: 031-745 1699; fax: 031-745 1698; e-mail: swahyudi@ciputra.ac.id)

³ Dosen, Program Studi Teknik Informatika Fakultas Industri Kreatif Universitas Ciputra, UC Town, Citraland, Surabaya 60219 INDONESIA (telp: 031-745 1699; fax: 031-745 1698; e-mail: nsugianto@ciputra.ac.id)

ketimbang menggunakan *search engine* seperti Google. Alasannya adalah karena pertanyaan-pertanyaan yang sifatnya subjektif tidak memiliki jawaban yang pasti (*no definite answer*), dan jawaban yang dibutuhkan lebih mengarah pada pandangan-pandangan yang diberikan oleh banyak orang, dan hasil yang diperoleh dipercaya lebih maksimal dengan melakukan *social search* (sebagai contoh, memanfaatkan media sosial untuk bertanya) dibandingkan mesin pencari yang pada umumnya lebih menyediakan data atau fakta [5].

Namun, terdapat beberapa kekurangan yang dapat ditemui dengan bertanya melalui media sosial, antara lain identitas asli dari penanya secara terbuka dapat diketahui oleh seluruh pihak yang membaca pertanyaan, lalu jumlah penjawab pertanyaan yang ada lebih sedikit karena hanya mencakup para *direct contact* dari si penanya, dan juga panjang pesan yang diperbolehkan pada sosial media masih terbatas, serta masih banyak lainnya [5].

Dengan demikian, terpikirkan sebuah ide untuk membuat sebuah aplikasi *mobile* bernama IniAtauTu, dimana dalam aplikasi ini memberikan kesempatan bagi penggunaannya untuk menanyakan cakupan pertanyaan-pertanyaan subjektif mengenai *lifestyle* dan kegiatan sehari-hari yang lebih membutuhkan jawaban berupa opini/pendapat/pandangan dari orang lain, dimana opini/pendapat tersebut berupa pilihan mana yang lebih diminati para penjawab dalam komunitas sehingga dapat membantu dan meyakinkan penanya untuk memutuskan pilihan akhirnya. Mengapa *mobile*? Karena melihat tingkat ketenaran *smartphone* yang begitu pesat di Indonesia, dimana pengguna *smartphone* di Indonesia pada tahun 2015 melonjak tinggi hingga mencapai 71,6 juta berdasarkan riset yang dilakukan APJII (Asosiasi Penyelenggara Jasa Internet Indonesia) [6].

Harapannya adalah dengan aplikasi ini setiap pengguna memiliki sebuah bantuan tambahan untuk mendiskusikan dengan individu lainnya mengenai pertanyaan-pertanyaan terkait *lifestyle* dan *daily activities* serta pilihan-pilihan apa yang membuat bimbang.

II. LANDASAN TEORI

Berikut adalah teori – teori yang digunakan sebagai referensi dalam pembuatan penelitian ini:

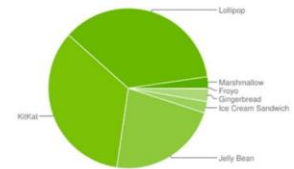
A. Mobile Application Development

Mobile application development atau yang disebut juga dengan pengembangan aplikasi *mobile* pada dasarnya mirip dengan pengembangan aplikasi Web, tetapi perbedaan utamanya adalah pada *mobile development*, aplikasi-aplikasi *mobile / apps* dibuat secara spesifik dengan memanfaatkan fitur-fitur unik yang ditawarkan dari perangkat *mobile*. Sebagai contoh, sebuah aplikasi *game* dibuat dengan memanfaatkan fitur *accelerometer* yang ada pada iPhone.

B. Android

Android adalah sebuah sistem operasi (*operating system*) berbasis Linux yang bersifat *open-source*, yang dirancang untuk perangkat mobile seperti *smartphone* dan komputer *tablet*. Android menawarkan pendekatan yang bersifat kompak untuk pengembangan aplikasi pada perangkat-perangkat *mobile*, yang artinya aplikasi yang dikembangkan setiap *developer* diharuskan untuk diperuntukkan hanya untuk Android dan aplikasi tersebut dapat dijalankan pada berbagai perangkat berbeda yang didukung oleh Android.

Version	Codename	API	Distribution
2.2	Froyo	8	0.1%
2.3.3 - 2.3.7	Gingerbread	10	2.6%
4.0.3 - 4.0.4	Ice Cream Sandwich	15	2.3%
4.1.x	Jelly Bean	16	8.1%
4.2.x		17	11.0%
4.3		18	3.2%
4.4	KitKat	19	34.3%
5.0	Lollipop	21	16.9%
5.1		22	19.2%
6.0	Marshmallow	23	2.3%



Gambar 1. Distribusi versi sistem operasi android

Dalam menentukan versi Android minimum dari aplikasi yang akan dibuat, dapat dilihat pada data statistik tentang distribusi semua versi android dari halaman dari Google Developer. Dari data tersebut dapat terlihat bahwa sistem operasi versi 4.1 (Jelly Bean) masih memiliki jumlah pengguna yang cukup tinggi sehingga *support* aplikasi dimulai dari versi ini.

C. Android Studio

Android Studio merupakan sebuah IDE (*Integrated Development Environment*) yang resmi diluncurkan Google sebagai perangkat yang membantu para developer dalam membangun aplikasi berbasis Android. IDE ini bersifat gratis untuk diunduh dan digunakan, dimana didalamnya terdapat sebuah *environment* yang kaya untuk pengembangan aplikasi dengan beragam *template* yang dapat dimanfaatkan oleh para developer baru sebagai landasan dalam mengembangkan aplikasi Android [7].

D. Aplikasi Hybrid vs. Native

Dalam mengembangkan aplikasi *hybrid*, proses yang dilaksanakan lebih cepat dibandingkan aplikasi *native*, serta dapat dijalankan pada berbagai platform. Akan tetapi masalah utamanya adalah aplikasi *hybrid* masih bergantung pada *browser native*, yang artinya aplikasi *hybrid* tidaklah secepat aplikasi *native*. Keuntungan utama dari aplikasi *native* adalah performa kerjanya, dimana pengguna yang memakainya dapat merasakan performa terbaik dari aplikasi, mulai dari animasi yang cepat, serta akses penuh pada *hardware* yang ada pada *smartphone* tersebut, dukungan *multi-touch*, serta API yang terbaru, walaupun dalam

pengembangannya tidak secepat dan semudah aplikasi *hybrid*.

E. Arsitektur *Client-Server*

Arsitektur dari sebuah aplikasi seringkali dimodelkan untuk mengilustrasikan layout secara keseluruhan dari *software* (kode aplikasi, platform, dll) serta *hardware* (*client*, *server*, perangkat *network*, dll). Walaupun terdapat banyak kemungkinan kombinasi antara *software* dan *hardware*, arsitektur dari aplikasi pada umumnya dimodelkan dalam bentuk arsitektur klien-server (*client-server architecture*), dimana secara garis besar satu atau lebih perangkat melakukan request informasi dari sebuah *server*, dan selanjutnya, *server* akan memberikan informasi yang di-request sebagai respon [8].

F. *Firestore Database*

Firestore Database merupakan *cloud database* yang disediakan oleh Google *Firestore*, dimana pada database ini data tersimpan dalam bentuk JSON dan tersinkronisasi secara *realtime* dengan setiap *client* yang terhubung. Pada *Firestore Database*, data terus berlanjut secara lokal, dan bahkan ketika *offline*, *event-event realtime* tetap dijalankan, sehingga memberikan pengalaman responsif bagi para penggunanya. Ketika perangkat kembali *online*, maka *Firestore Database* akan langsung mensinkronisasikan perubahan data lokal yang telah terjadi sebelumnya dan meng-update data yang ada pada *database* secara otomatis [9].

Firestore milik *Firestore* ini merupakan basis data yang bersifat non-relasional atau yang disebut dengan NoSQL, dimana *database* ini merupakan jenis database yang tidak menggunakan sistem tabel dalam implementasinya serta tidak menyimpan data secara lokal pada perangkat melainkan menyimpannya pada awan (*cloud*). Selain itu, *Firestore Database* juga memiliki optimisasi dan fungsionalitas yang berbeda apabila dibandingkan dengan basis data relasional.

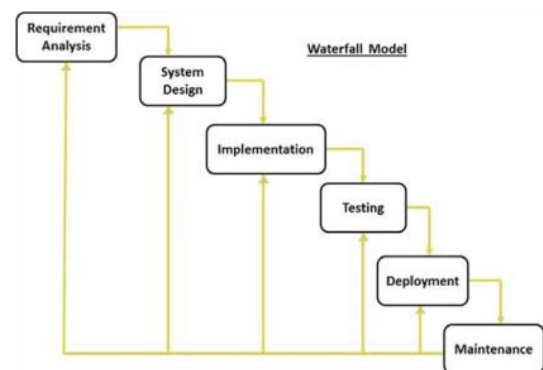
G. SDLC

SDLC atau *Software Development Life Cycle* secara khusus merupakan serangkaian langkah atau tahap-tahap yang mencerminkan sebuah model untuk pengembangan serta pengaturan *lifecycle* sebuah aplikasi atau *software*. Tujuan utama dari proses SDLC adalah untuk membantu menghasilkan sebuah produk yang bersifat hemat biaya, efektif serta berkualitas tinggi. Sesaat setelah sebuah aplikasi telah berhasil dibuat, pemetaan dari SDLC selanjutnya akan membantu proses *deployment* dari *software* tersebut dengan baik [10].

Salah satu versi yang umumnya secara luas digunakan dari SDLC adalah model Waterfall, yang terdiri dari tahap-tahap sebagai berikut:

1. *Requirement Gathering and Analysis*. Seluruh kebutuhan yang mungkin digunakan di dalam sistem yang akan di-develop digambarkan pada tahap ini lalu kemudian didokumentasikan.

2. *System Design*. Tahap ini membantu menentukan kebutuhan *hardware* dan sistem mana saja yang bakal dipakai serta membantu menggambarkan arsitektur sistem secara keseluruhan.
3. *Implementation / Coding*. Dengan input yang diperoleh dari tahap sebelumnya, pengembangan sistem kemudian mulai dikerjakan. Tahap ini merupakan tahap yang memakan waktu paling lama dari SDLC.
4. *Testing*. Selanjutnya adalah melakukan percobaan terhadap kebutuhan-kebutuhan yang ada untuk memastikan bahwa produk yang ada dapat mengatasi kebutuhan-kebutuhan yang ingin ditangani.
5. *Deployment*. Setelah berhasil melakukan testing, maka produk yang ada disebar / di-deploy pada *customer* untuk digunakan. Segera setelah diberikan, maka para *customer* pertama-tama akan melakukan *beta testing*. Jika ternyata dibutuhkan perubahan atau apabila terdapat *bugs*, maka akan diperbaiki.
6. *Maintenance*. Sesaat setelah pengguna mulai memakai sistem yang telah dikembangkan, maka masalah-masalah aktual pun akan muncul dan butuh untuk ditangani dari waktu ke waktu.



Gambar 2. Bagan Model Waterfall

III. ANALISIS DAN DESAIN

Bab ini berisi penjelasan mengenai analisa kebutuhan sistem, serta desain sistem dari aplikasi yang dibuat.

A. Analisis Sistem

Sebelum perancangan sistem dilakukan, penulis perlu melakukan analisis terhadap permasalahan, serta kebutuhan sistem yang dirancang dan dibangun. Analisis permasalahan dilakukan untuk menentukan fitur utama dari aplikasi yang akan dirancang dan dibangun, sementara analisis kebutuhan sistem dilakukan untuk mengetahui struktur dan subsistem yang mendukung sistem utama.

- a) Perbandingan Sarana-Sarana Online Sebagai Media Tanya Jawab yang Telah Ada

Pada poin-poin berikut ini telah dirangkum beberapa sarana *online* yang dapat dimanfaatkan oleh para pengguna internet sebagai media tanya jawab yang dapat membantu mereka menanyakan pertanyaan-pertanyaan subjektif mengenai *lifestyle*, beserta kelebihan dan kekurangannya masing-masing.

1. **Instagram.** Yang membuat Instagram digunakan untuk membantu menanyakan preferensi adalah fitur postingan yang telah dilengkapi dengan Carousel, dimana pengguna dapat meng-*upload* dua atau lebih gambar sekaligus. Kelebihannya antara lain *suitable* digunakan untuk memperoleh informasi subjektif, pertanyaan yang ditanyakan sebagian besar mengenai *lifestyle* maupun aktivitas sehari-hari, dapat disertai dengan gambar untuk lebih memperjelas maksud pertanyaan, serta pertanyaan dapat ditanyakan secara *private* ke pihak tertentu saja melalui *direct message*. Namun, kekurangannya antara lain terdapat kesulitan melihat jawaban mana yang paling dominan/diminati apabila pengguna menanyakan pertanyaan yang bersifat *closed question*, cakupan penjawab pertanyaan menjadi lebih sempit apabila pengguna meng-set akun mereka ke mode *private*, serta identitas asli penanya diketahui secara terbuka.
2. **TwitterPolls.** Yang membuat Twitter dipakai untuk menanyakan preferensi adalah fitur TwitterPolls, dimana para *tweeter* dapat memposting postingan berupa *open poll* yang dapat ditujukan pada semua orang di komunitas untuk membantu memberikan preferensi mereka. Kelebihannya antara lain *suitable* digunakan untuk memperoleh informasi subjektif serta nyaman digunakan untuk menanyakan pertanyaan yang bersifat *closed question*. Kekurangannya antara lain tidak dapat disertai dengan gambar untuk lebih memperjelas maksud pertanyaan, cakupan penjawab pertanyaan menjadi lebih sempit apabila pengguna meng-set akun mereka ke mode *private*, serta umumnya identitas asli penanya diketahui secara terbuka.
3. **Quora.** Yang membuat Quora digunakan untuk menanyakan preferensi adalah adanya fitur Ask Question, dimana pengguna dapat menanyakan pertanyaan beserta dengan pilihan-pilihan yang sedang dipertimbangkannya. Kelebihannya antara lain *suitable* digunakan apabila ingin memperoleh informasi subjektif, cakupan penjawab pertanyaan potensial luas karena meliputi seluruh pengguna dalam komunitas serta dapat mengajukan pertanyaan secara *anonymous*. Namun, kekurangannya antara lain tidak dapat disertai dengan gambar untuk lebih memperjelas maksud pertanyaan, pertanyaan tidak dapat ditanyakan secara *private*, serta tidak spesifik/fokus untuk pertanyaan-pertanyaan mengenai *lifestyle* dan aktivitas sehari-hari.
4. **Yahoo!Answers.** Yang membuat Yahoo!Answers digunakan untuk menanyakan preferensi adalah adanya Ask a Question, dimana pengguna dapat

menanyakan pertanyaan beserta dengan pilihan-pilihan yang sedang dipertimbangkannya pada komunitas yang ada. Kelebihannya antara lain cakupan penjawab pertanyaan potensial yang luas karena meliputi seluruh pengguna lain dalam komunitas, adanya sistem poin tiap kali menjawab pertanyaan, sehingga mendorong pengguna giat menjawab pertanyaan, serta dapat menanyakan pertanyaan secara *anonymous*. Namun kekurangannya antara lain kesulitan untuk melihat jawaban mana yang lebih diminati, tidak dapat disertai dengan gambar untuk lebih memperjelas maksud pertanyaan, serta pertanyaan tidak dapat ditanyakan secara *private*.

b) Masalah yang Dihadapi

Saat ini, perkembangan dunia *online* yang pesat telah membuat kegiatan bertanya dan menjawab pertanyaan dengan skala yang besar menjadi lebih mudah. Dengan memanfaatkan mesin pencari (*search engine*) seperti Google, kita dapat mengajukan pertanyaan yang kita inginkan dalam bentuk *query* dan serangkaian hasil pencarian akan muncul. Selain *search engine*, terdapat pula situs Q&A seperti Quora, yang memungkinkan penggunaannya untuk mengajukan pertanyaan yang diinginkan, kemudian jawabannya dapat diperoleh dari seluruh pengguna di dalam komunitas.

Tidak hanya situs Q&A saja, para pengguna internet juga telah memanfaatkan akun *social media* mereka, seperti Facebook dan Instagram, sebagai sarana bertanya dengan beberapa alasan, antara lain: lebih mudah mendapatkan jawaban langsung dari *followers/friends* yang mereka miliki, kemudian lebih yakin dengan jawaban yang diberikan, sebab berasal dari sumber-sumber yang telah mereka kenal/ketahui. Selain itu, pertanyaan-pertanyaan yang ditanyakan pada media sosial seperti Facebook sebagian besar bersifat subjektif, dimana jawaban yang diharapkan penanya adalah opini / masukan dari individu lain dalam komunitas ketimbang fakta-fakta yang sifatnya objektif [5].

Dari survei yang dilakukan dengan menggunakan kuisioner online terhadap 60 responden di Surabaya, terlihat bahwa sebagian besar responden paling sering mengalami dilema ketika harus menentukan jawaban dengan memilih satu dari dua buah pilihan terkait masalah yang dihadapi. Sebagai contoh: dari dua pilihan baju yang ada, manakah yang lebih anggun untuk dipakai ke pesta; dari dua buah *selfie* yang ada manakah yang akan mendapat lebih banyak *like* ketika di-*upload*). Dari hasil ini, apabila dikaitkan dengan kegiatan tanya jawab memanfaatkan media sosial atau situs Q&A, ditemukan beberapa masalah, antara lain:

1. Penanya harus melihat satu per satu jawaban yang ada (umumnya dengan meng-*scroll* dari atas ke bawah) untuk mengetahui pilihan mana yang paling disetujui, dan kewalahan apabila jawaban yang ada jumlahnya banyak.

2. Jawaban dan alasan pendukung tergabung menjadi satu dalam bentuk *comment*, sehingga lebih membuat kewalahan untuk mengetahui pilihan mana yang paling disetujui dan jawaban yang diharapkan bisa menjadi tidak *straightforward*, tambah lagi apabila *comment* tertulis dalam kalimat-kalimat yang panjang.
3. Terkadang jumlah penjawab dari sebuah pertanyaan terhitung sedikit, salah satu alasannya adalah karena malas/tidak termotivasi untuk menuliskan jawaban, apalagi jika pihak penanya menanyakan banyak pertanyaan sekaligus.
4. Seringkali ada pertanyaan-pertanyaan subjektif terkait *lifestyle* dan *everyday life* yang ingin ditanyakan kepada pihak tertentu saja, karena penanya merasa lebih percaya dengan jawaban dari orang tertentu saja atau tidak nyaman apabila pertanyaan diketahui secara luas.

c) Solusi yang Ditawarkan

Dari masalah yang muncul, terdapat solusi yang ditawarkan untuk membantu menyelesaikan masalah tersebut dan untuk penjelasan selengkapnya disajikan dalam tabel berikut ini:

TABEL I
SOLUSI YANG DITAWARKAN BERUPA FITUR

No.	Masalah	Solusi
1	Penanya harus melihat satu per satu jawaban yang ada (umumnya dengan meng- <i>scroll</i> dari atas ke bawah) untuk mengetahui pilihan mana yang paling disetujui, dan kewalahan apabila jawaban yang ada jumlahnya banyak.	- Fitur melihat dan menjawab pertanyaan pada aplikasi IniAtauItu, - Fitur mengajukan pertanyaan pada aplikasi IniAtauItu, disertai dengan dua buah pilihan yang disertai dengan gambar dan <i>caption</i> .
2	Jawaban dan alasan pendukung tergabung menjadi satu dalam bentuk <i>comment</i> .	Fitur memberikan comment secara terpisah untuk tiap postingan pertanyaan yang ada.
3	Terkadang jumlah penjawab dari sebuah pertanyaan terhitung sedikit, salah satu alasannya adalah karena malas/tidak termotivasi untuk menuliskan jawaban.	Adanya pemrolehan poin untuk beberapa interaksi pada aplikasi, misalnya dengan membantu menjawab pertanyaan dari pengguna lain.
4	Seringkali ada pertanyaan-pertanyaan subjektif terkait <i>lifestyle</i> dan <i>everyday life</i> yang ingin ditanyakan kepada pihak tertentu saja.	- Fitur mengirim <i>direct question</i> . - Fitur melihat dan menjawab <i>direct question</i> ,

Dari tabel diatas dapat disimpulkan bahwa fitur-fitur utama yang ditawarkan dalam aplikasi ini adalah sebagai berikut :

1. Fitur melihat dan menjawab pertanyaan, dengan cara langsung memilih salah satu yang paling diminati dari dua buah pilihan yang ada untuk membantu penanya menentukan pilihan akhir. Pilihan yang diberikan bersifat *anonymous*, dimana penanya tidak akan mengetahui siapa saja yang memberikan jawaban pada pertanyaannya.
2. Fitur mengajukan pertanyaan pada aplikasi IniAtauItu, dimana penanya menyertakan dua buah pilihan yang disertai dengan gambar dan *caption* untuk memberitahu *top 2 choices* yang sedang dibicarakan.
3. Fitur memberikan *comment* secara terpisah untuk tiap postingan pertanyaan yang ada. Dapat digunakan untuk memberikan komentar terkait dengan pilihan pada postingan pertanyaan.
4. Fitur mengirim *direct question*, dimana penanya dapat mengajukan pertanyaan secara *private* kepada orang tertentu yang diinginkan saja beserta pilihan yang sedang dibicarakan.
5. Fitur melihat dan menjawab *direct question*, dimana penerima pertanyaan dapat melihat dan juga membantu menjawab pertanyaan yang dikirim dari pengguna lain.
6. Adanya sistem poin untuk beberapa interaksi pada aplikasi, misalnya dengan membantu menjawab pertanyaan dari pengguna lain.

d) Analisis Kebutuhan Aplikasi

Aplikasi yang dirancang dan dibuat pada tugas akhir ini memiliki kebutuhan spesifik agar proses dapat berjalan dengan lancar. Kebutuhan aplikasi dapat berbeda-beda sesuai dengan sistem yang diperlukan.

1. Mobile App Client

Aplikasi mobile IniAtauItu di-install pada smartphone pengguna dengan spesifikasi *hardware* dan *software* sebagai berikut:

- 1) *Operating system* Android
- 2) Versi Android minimal 4.1 (Jelly Bean)
- 3) Akses internet yang cukup dan memadai.

2. Backend

Data-data pada aplikasi ini juga perlu disimpan dan diakses, dimana Firebase dimanfaatkan untuk keperluan ini. Firebase dipilih karena *developer* tidak perlu memikirkan lagi kode server yang akan digunakan ketika membuat aplikasi, serta data akan tersimpan dalam basis data yang bersifat *realtime*, membuat proses *developing* aplikasi menjadi lebih mudah dan cepat [11].

B. Desain Sistem

Pada bagian ini akan dijelaskan mengenai desain dari sistem yang dibuat. Desain dibuat berdasarkan hasil analisa yang telah dijelaskan pada bagian sebelumnya.

Desain sistem terdiri dari 4 bagian, yaitu desain arsitektur, desain *database*, desain proses, dan desain *user interface*.

a) Desain Arsitektur

Aplikasi ini akan menggunakan desain arsitektur *client-server*. Arsitektur ini memungkinkan banyak pengguna untuk dapat mengakses data yang terdapat di *server* secara bersamaan. Berikut pembagian aktivitas dari kedua bagian :

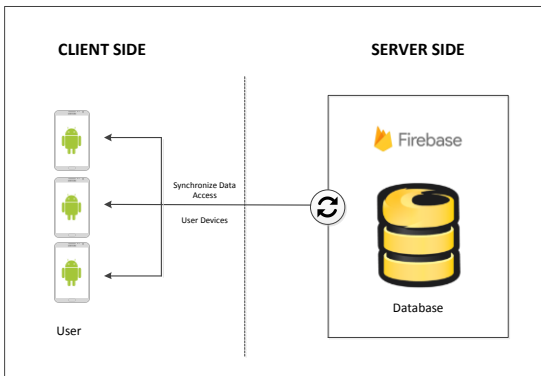
1. Client

Dari sisi *client*, pengguna dapat melakukan kegiatan melihat dan menggunakan aplikasi yang ada pada *smartphone*. Aplikasi *mobile* tersebut dapat digunakan untuk berkomunikasi langsung dengan *server* (Firebase), untuk menampilkan dan menyimpan data yang bersangkutan dengan pengguna.

2. Server

Dari sisi *server*, terdapat Firebase yang telah meng-*handle database* serta *backend*, sehingga *client* nantinya dapat mengakses data dari *database* dengan mudah. *Database* dari Firebase ini bersifat *realtime*, artinya, apabila ada perubahan data, hasilnya akan langsung terlihat pada perangkat *client* yang bersangkutan, sehingga *client* tidak perlu mengirim *request* lagi.

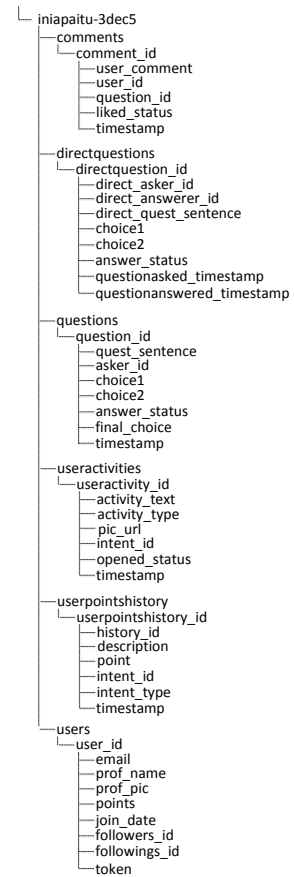
Desain arsitektur yang akan diterapkan pada aplikasi ini dapat dilihat pada gambar berikut ini :



Gambar 3. Desain Arsitektur

b) Desain Database

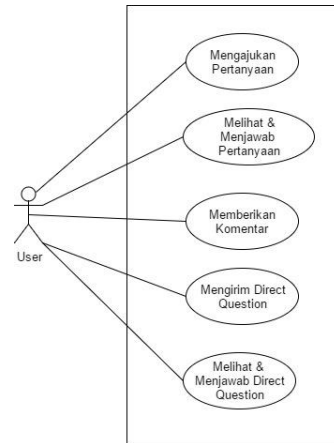
Data dari aplikasi ini akan disimpan pada *Realtime Database* milik Firebase sebagai objek JSON. Objek JSON ini tidak seperti basis data SQL, sehingga tidak ada tabel atau rekaman. Bentuk dari gabungan objek JSON ini dapat disebut sebagai *JSON Tree*. Berikut rancangan *database* dalam bentuk *tree*.



Gambar 4. Desain Database

c) Use Case Diagram

Berikut merupakan diagram *use case* dari aplikasi IniAtauItu yang akan menjelaskan secara garis besar fitur-fitur utama yang ada di dalam aplikasi IniAtauItu:



Gambar 5. Use Case Diagram

IV. IMPLEMENTASI

Pada bab ini akan dijelaskan mengenai implementasi dari desain yang telah dibuat pada bab sebelumnya, implementasi akan dibagi menjadi tiga bagian, yaitu implementasi fitur, implementasi basis data, serta implementasi *user interface* / tampilan.

A. Implementasi Fitur

Pada bagian ini akan dijelaskan mengenai implementasi dari fitur-fitur yang terdapat pada aplikasi ini, baik fitur-fitur utama maupun fitur-fitur tambahan lainnya.

a) Mengajukan Pertanyaan

Dimulai dari pengguna membuka aplikasi IniAtauItu dan meng-klik tab ketiga untuk menuju halaman mengajukan pertanyaan. Pengguna kemudian mengetikkan pertanyaan yang diinginkan, dan meng-*upload* gambar baik dari galeri maupun langsung dari kamera, serta menambahkan *caption* untuk menjelaskan masing-masing pilihan secara ringkas. Setelah selesai, pengguna dapat menekan *button* Okay! untuk menyimpan pertanyaan.

b) Melihat dan Menjawab Pertanyaan

Pengguna dapat melihat pertanyaan-pertanyaan yang telah di-*upload* oleh pengguna lainnya dengan meng-klik tab pertama, untuk menuju halaman *Friends Feeds*, berisikan pertanyaan-pertanyaan dari pengguna itu sendiri serta para pengguna lain yang telah di-*follow* sebelumnya, serta tab kedua, untuk menuju halaman *Community Feeds*, yang menampilkan pertanyaan-pertanyaan dari seluruh komunitas aplikasi IniAtauItu secara luas.

c) Memberikan dan meng-like komentar

Pengguna juga dapat menambahkan komentar pada tiap postingan pertanyaan yang ada dengan meng-klik tombol *comment* yang ada pada postingan, pengguna dapat menuliskan kalimat komentar terkait pertanyaan/pilihan yang ada untuk kemudian di-*posting*.

d) Menentukan Pilihan Akhir Pertanyaan

Setelah pengguna mengajukan pertanyaan yang diinginkan, apabila merasa jumlah jawaban/*vote* yang didapatkan sudah cukup dan akhirnya telah menentukan salah satu dari kedua pilihan yang diinput sebelumnya, pengguna dapat menginformasikannya dengan menggunakan fitur Pilihan Saya pada postingan pertanyaan yang telah dibuat.

e) Melihat Profil Pengguna

Para pengguna dapat saling melihat profil mereka satu sama lain saat menggunakan aplikasi ini, dengan cara: mencarinya pada halaman *search*, atau bisa juga dengan langsung meng-klik *profile picture* yang tersedia pada setiap postingan pertanyaan. Setelah itu, pengguna akan langsung menuju ke halaman *profile* pengguna lainnya, dimana informasi-informasi dapat terlihat.

f) Meng-follow/unfollow Pengguna Lain

Ketika membuka halaman profil, para pengguna dapat berteman dengan para pengguna lainnya dengan meng-follow akun mereka, dengan cara menekan tombol + *follow* yang ada.

g) Filter Mengirim Direct Question

Selain mengajukan pertanyaan untuk kemudian dilihat oleh para pengguna lain secara luas, pengguna aplikasi juga dapat mengirimkan pertanyaan langsung

berupa *direct question* kepada para pengguna lain yang telah masuk ke daftar *following* sebelumnya. Sama halnya dengan mengajukan pertanyaan, pengguna dapat menekan tombol Pertanyaan Baru untuk menuliskan kalimat pertanyaan yang diinginkan, serta dua buah pilihan yang dapat dilengkapi dengan meng-*input caption* serta gambar, kemudian menekan tombol Okay!

h) Melihat dan Menjawab *Direct Question*

Pada halaman *Direct Question*, selain dapat mengirimkan pertanyaan langsung ke pengguna lain, pengguna juga dapat mengecek *direct question*, baik yang diterima dari pengguna lain (*inbox*) maupun yang dikirimkannya sendiri (*outbox*).

i) Melakukan Pencarian Akun Pengguna

Pengguna dapat meng-klik icon *search* yang ada pada halaman *Friends Feeds* dan kemudian mengetikkan nama akun pengguna yang ingin dicari pada kolom pencarian yang tersedia pada bagian atas.

j) Melihat Berita Aktivitas Akun

Pengguna dapat meng-klik tab ke-2 dari kanan untuk menuju ke halaman berita aktivitas akun. Pada halaman ini, pengguna dapat mengecek satu per satu aktivitas dari akun yang dimilikinya.

k) Melihat Histori Poin

Pengguna dapat meng-klik *progress* poin yang ada pada halaman profil mereka untuk mengecek darimana poin-poin yang ia peroleh berasal.

l) Membagikan Postingan ke Facebook

Setiap postingan pertanyaan yang ada pada *Friends Feeds* dan *Community Feeds* dapat dibagikan ke akun media sosial Facebook pengguna, dengan meng-klik tombol *share* yang ada pada tiap item postingan.

B. Implementasi Basis Data

Pada bagian ini akan dijelaskan mengenai implementasi basis data yang digunakan pada aplikasi IniAtauItu. Data pada aplikasi ini disimpan pada Realtime Database milik Google Firebase, dimana data akan disimpan kedalam bentuk *JSON tree*. Saat ini data yang disimpan didalam objek *JSON tree* mempunyai enam *child* utama yaitu, *users*, *questions*, *directquestions*, *comments*, *useractivities*, dan *userpointshistory*.

```

"JP2fu6BQbZ4ifKynaP2ZD71Mar2" : {
  "email" : "taylorswift@gmail.com",
  "followers_id" : {
    "-KkTr4ib1l5MxqG12uac" : "XntFDYs9FmdNf7F72mXOIP5wg1I3"
  },
  "followings_id" : {
    "-Kj3bHrVjCmDy9vsmk5" : "vHmRzHciacbv1jZ2Qx0e86pe2",
    "-KkeA5a3Ht_qeX5Ah12" : "XntFDYs9FmdNf7F72mXOIP5wg1I3",
    "-KkGCV9tdw1Mve1UjbnD" : "bhW3ztfF52fySKNi9fAmM1rZ0y1",
    "-KkGDBVHZ0Yq4Ehub9Q" : "KjymAD8KNksQ8Fgl0eKX9TGnNjF2"
  },
  "join_date" : "1493729069",
  "points" : 2,
  "prof_name" : "Taylor",
  "prof_pic" : "https://firebasestorage.googleapis.com/v0/b/iniapaitu-3dec5.appspot.com/o/search_prof_name : \"taylor\",
  "search_prof_name" : "taylor",
  "token" : {
    "-KkAK0wINVGP633hc1e" : "ey8hdA-jQt8:APA91bELF7wRYKAF0Sv61RC_9X1BU4iU6pdia0HCuvsgn3Ju",
    "user_id" : "JP2fu6BQbZ4ifKynaP2ZD71Mar2"
  }
}

```

Gambar 6. Contoh implementasi child users dalam JSON Object

C. Implementasi User Interface

Aplikasi *mobile* memiliki desain *user interface* yang sesuai dengan fitur-fitur yang ada pada aplikasi. Dalam proses implementasi, tampilan dibuat dengan menggunakan panduan desain *standard* milik Google yakni material desain. Berikut merupakan masing-masing halaman yang ada pada aplikasi IniAtauItu.

a) Halaman *Splash*

Saat pertama kali membuka app, halaman yang berisikan logo aplikasi IniAtauItu ini muncul selama proses *loading resource* aplikasi selesai. Lalu dilanjutkan ke halaman login.

b) Halaman Login dan Register

Setelah dari halaman *splash*, pengguna akan melihat halaman *login*, dimana pada halaman ini user dapat melakukan aktivitas login ke aplikasi dengan menggunakan *email* dan *password* yang telah didaftarkan sebelumnya. Apabila pengguna belum mendaftar, maka dapat menekan tombol Sign Up pada bagian bawah halaman untuk menuju ke halaman *register*.

c) Halaman *Friends Feeds*

Pada halaman ini ditampilkan seluruh postingan pertanyaan baik milik pengguna yang sedang login maupun pengguna lain yang telah masuk dalam daftar *following*. Pengguna dapat melihat sambil membantu menjawab postingan-postingan pertanyaan yang ada dengan memilih salah satu dari dua buah pilihan (berupa gambar dan *caption*) yang ada.

d) Halaman *Community Feeds*

Halaman ini ditampilkan setelah *user* berpindah tab dengan meng-klik tab kedua pada aplikasi. Pada halaman ini ditampilkan seluruh postingan pertanyaan yang ada dalam komunitas IniAtauItu, dengan pengecualian untuk postingan pengguna yang sedang login dan teman-teman pengguna, karena telah ditampilkan sebelumnya pada halaman *Friends Feeds*.

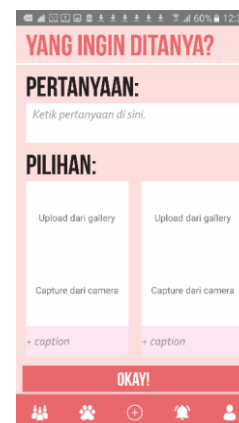


Gambar 7. Halaman *Community Feeds*

e) Halaman Memposting Pertanyaan

Halaman ini ditampilkan setelah pengguna berpindah tab dengan meng-klik tab ketiga pada aplikasi. Pada

halaman ini pengguna dapat menuliskan pertanyaan yang ingin ditanyakan ke komunitas dengan mengisi kalimat pertanyaan dan melengkapi dua buah pilihan terkait pertanyaan.



Gambar 8. Halaman Memposting Pertanyaan

f) Halaman Berita Aktivitas Akun

Halaman ini ditampilkan setelah pengguna berpindah tab dengan meng-klik tab keempat pada aplikasi. Yang ditampilkan pada halaman ini yaitu berita-berita mengenai aktivitas akun pada aplikasi IniAtauItu. Interaksi-interaksi seperti adanya follower baru, mendapat *like* untuk komentar yang dituliskan, dan mendapat *direct question* dari pengguna lain akan ditampilkan pada halaman ini dalam bentuk daftar.

g) Halaman Profil

Halaman ini ditampilkan setelah pengguna berpindah tab dengan meng-klik tab terakhir pada aplikasi. Disinilah pengguna dapat melihat informasi-informasi mengenai akun mereka sendiri, mulai dari nama, jumlah poin, *level*, *progress* poin, daftar *follower* dan *following*, serta postingan-postingan pertanyaan yang telah dibuat.

h) Halaman Detail Postingan

Pengguna dapat melihat detail postingan dengan mengklik salah satu dari daftar postingan yang ada pada halaman profil. Aktivitas menentukan pilihan akhir terkait pertanyaan yang diposting dapat dilakukan pengguna pada halaman profil mereka sendiri.

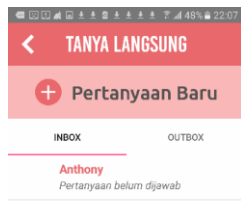
i) Halaman Dialog Pilihan Saya

Dengan mengklik *button* Pilihan Saya, maka dialog untuk menentukan menentukan pilihan final dari pertanyaan yang ada akan muncul. Untuk memilih tinggal mengklik salah satu pilihan yang ada kemudian klik tombol Selesai untuk menyimpan pilihan final pertanyaan.

j) Halaman *Direct Question*

Dengan mengklik logo yang ada di sisi kanan aplikasi pada halaman *Friends Feeds*, maka halaman mengenai *direct question* akan ditampilkan. Pada halaman ini, pengguna dapat melihat pertanyaan-pertanyaan langsung/*direct question* yang diterima dari pengguna lain pada tab *inbox* serta *direct question* yang dikirimkan oleh

pengguna itu sendiri pada tab *outbox*.



Gambar 9. Halaman Direct Question

k) Halaman Kirim Direct Question

Pada halaman Direct Question, pengguna akan menuju ke halaman untuk membuat dan mengirim *direct question* dengan meng-klik tombol Pertanyaan Baru. Pada halaman ini, pengguna dapat menuliskan kalimat pertanyaan yang diinginkan, memilih *recipient*, serta melengkapi dua buah pilihan terkait *direct question* yang akan dikirim.

l) Halaman Komentar

Untuk setiap postingan pertanyaan yang ada pada *Friends Feeds* dan *Community Feeds*, pengguna dapat menuju ke halaman komentar dengan meng-klik tombol berbentuk *comment* pada sisi kanan bawah postingan. Pada halaman ini pengguna dapat menuliskan komentar yang diinginkan.

m) Halaman Search

Halaman ini dimunculkan setelah pengguna mengklik *icon* berbentuk kaca pembesar pada sisi kiri atas halaman *Friends Feeds*. Pengguna dapat mencari dan melihat profil pengguna lain dalam komunitas *IniAtauItu* dengan mengetikkan nama pada kolom pencarian.

n) Halaman Histori Poin

Pada halaman ini akan diperlihatkan histori poin dari akun pengguna, sehingga pengguna dapat mengetahui darimana penambahan dan pengurangan poin berasal.

V. HASIL PENGUJIAN

Bab ini berisi penjelasan mengenai hasil pengujian aplikasi yang telah dilakukan penulis.

A. Pengujian Fitur

Pengujian fitur dilakukan untuk mengetahui apakah fitur yang dibuat pada aplikasi *IniAtauItu* dapat berjalan sesuai dengan yang diharapkan atau terdapat pesan *error* maupun kesalahan sistem atau *bug* yang muncul selama proses pengujian berlangsung. Pengujian dilakukan untuk tiap fitur yang ada pada aplikasi, mulai dari mengajukan pertanyaan, melihat dan menjawab pertanyaan,

memberikan dan meng-like komentar, menentukan pilihan akhir pertanyaan, melihat profil pengguna, meng-follow/unfollow pengguna lain, mengirim *direct question*, melihat dan menjawab *direct question*, melihat berita aktivitas pengguna, melihat histori poin, dan membagikan postingan ke Facebook. Berikut merupakan salah satu contoh pengujian fitur aplikasi yang dilakukan.

TABEL II
CONTOH PENGUJIAN PADA FITUR MENGAJUKAN PERTANYAAN PADA APLIKASI

Aktivitas	Hasil yang diharapkan	Hasil sebenarnya
Menekan tab ketiga (icon plus) pada halaman utama	Menampilkan halaman Memposting Pertanyaan dimana pengguna dapat mengisi data-data pertanyaan yang akan diposting.	Halaman Memposting Pertanyaan tampil lengkap beserta kolom untuk mengisi pertanyaan dan <i>caption</i> , serta dua buah kolom untuk mengupload gambar.
Menekan kolom meng-upload gambar dari galeri <i>smartphone</i>	Menampilkan galeri yang ada pada <i>smartphone</i> milik pengguna untuk memilih satu gambar yang akan di-upload.	Galeri <i>smartphone</i> tampil dimana foto-foto yang ada pada galeri dapat dipilih salah satu untuk di-upload.
Menekan kolom meng-upload gambar langsung dari kamera	Mengakses dan membuka kamera untuk meng-capture gambar untuk kemudian di-upload.	Kamera terbuka dan pengguna dapat melakukan <i>capture</i> gambar.
Menekan tombol Okay!	Dialog <i>loading</i> akan muncul, data pertanyaan akan disimpan pada <i>database</i> di <i>Firebase</i> , dan setelah selesai, dialog <i>loading</i> akan hilang.	Dialog <i>loading</i> tampil, data pertanyaan tersimpan dengan baik pada path ('/questions) di <i>Firebase</i> , dan dialog <i>loading</i> hilang.

VI. KESIMPULAN DAN SARAN

Pada bab ini terdapat penarikan kesimpulan terhadap hasil proses rancang bangun aplikasi *IniAtauItu*. Selain itu juga terdapat saran untuk pengembangan topik tugas akhir yang sejenis bagi pembaca yang mengerjakan tugas akhir.

a) Kesimpulan

Untuk merancang dan membangun aplikasi *IniAtauItu*, dibutuhkan sejumlah perangkat, mulai dari *smartphone* dengan sistem operasi *Android*, termasuk kamera yang ada pada *smartphone* tersebut,

hingga akun Firebase untuk keperluan *backend* dari aplikasi serta akses Internet yang memadai.

Arsitektur sistem sendiri bersifat klien-server, dimana aplikasi IniAtauItu akan bertindak sebagai pihak klien dan Firebase bertindak sebagai pihak *server*. Untuk implementasinya, IDE Android Studio dimanfaatkan sebagai tool *coding*, yang merupakan fokus utama dari penulis untuk mengembangkan aplikasi, serta fitur-fitur yang disediakan oleh Firebase, seperti *Realtime Database* untuk penyimpanan data, *Authentication* untuk aktivitas login, *Storage* untuk penyimpanan file, serta *Functions* untuk keperluan notifikasi.

Setelah aplikasi IniAtauItu berhasil di-*develop*, maka selanjutnya adalah melakukan uji coba kepada pengguna nyata untuk mengetahui kesalahan dan error yang dapat terjadi untuk kemudian diperbaiki kembali. Ketika semua fitur dari aplikasi telah berhasil dijalankan, maka aplikasi telah siap untuk dirilis ke Google PlayStore untuk digunakan oleh para *end-user* sebagai salah satu alternative untuk menanyakan preferensi dan pendapat dari orang lain dalam komunitas mengenai pilihan *lifestyle*.

b) Saran

Pada bagian ini akan dijabarkan mengenai saran dari penulis kepada pembaca, bagian ini akan dibagi menjadi dua, yaitu saran perbaikan dan saran perluasan sebagai berikut :

1. Saran Perbaikan

Pada bagian ini akan dijabarkan mengenai saran penulis terhadap perbaikan aplikasi ini kedepannya, berikut adalah saran yang diberikan :

- 1) Paket akun Firebase yang digunakan sebagai backend dapat di-*upgrade* ke paket yang berbayar untuk meningkatkan performa aplikasi apabila jumlah pengguna telah terhitung banyak.
- 2) Dengan di-upgradenya akun Firebase, maka pada fitur mengajukan pertanyaan pada aplikasi IniAtauItu selanjutnya dapat menggunakan *short video* sebagai pilihan pertanyaan.
- 3) Poin yang telah dikumpulkan pengguna bisa di-*redeem* dengan voucher atau barang lain yang dapat meningkatkan kepuasan pengguna.

2. Saran Perluasan / Penajaman

Pada bagian ini akan dijabarkan mengenai saran penulis terhadap pembaca yang ingin membuat aplikasi sejenis, berikut adalah saran yang diberikan :

- 1) Merancang sistem rekomendasi pada halaman Community Feeds.
- 2) Merancang sistem rekomendasi untuk menampilkan daftar akun-akun lain yang memiliki *interest* pada topik *lifestyle* yang sama/mendekati dengan pengguna aplikasi.

DAFTAR PUSTAKA

- [1] E. Shafir, "Reason-based choice," *Cognition*, vol. 49, no. 1-2, pp. 11-36, 1993.
- [2] M. Zeelenberg, "Anticipated regret, expected feedback and behavioral decision making," *Journal of Behavioral Decision Making*, vol. 12, no. 2, pp. 93-106, 1999.
- [3] M. Hanselmann and C. Tanner, "Taboos and Conflicts in Decision Making: Sacred Values, Decision Difficulty, and Emotions," *Judgment and Decision Making*, no. 3, pp. 51-63, 2008.
- [4] A. Tversky and E. Shafir, "Choice under Conflict: The Dynamics of Deferred Decision," *Psychological Science*, vol. 3, no. 6, pp. 358-361, 1992.
- [5] M. R. Morris et al, "What Do People Ask Their Social Networks, and Why? A Survey Study of Status Message Q&A Behavior," *Proceedings of the 28th International Conference on Human Factors in Computing Systems*, pp. 1739-1748, 2010.
- [6] P. K. Sari and Candiwan, "Measuring Information Security Awareness of Indonesian Smartphone Users," *TELKOMNIKA*, pp. 493-500, 2014.
- [7] Android Studio, "Mengenal Android Studio," May 2013. [Online]. Available: <https://developer.android.com/studio/intro/index.html>.
- [8] H. S. Oluwatosin, "Client-Server Model," *IOSR Journal of Computer Engineering*, pp. 67-71, 2014.
- [9] Firebase, "Firebase Realtime Database," Firebase, 6 July 2016. [Online]. Available: <https://firebase.google.com/docs/database/>. [Accessed 20 April 2017].
- [10] I. Sommerville, *Software engineering*. Boston: Pearson, 2011.
- [11] K. M. N. Kumar et al, "Implementing Smart Home Using Firebase," *International Journal of Research in Engineering and Applied Sciences*, pp. 193-198, 2015.