**Pengembangan Aplikasi Daily Quest: Aplikasi Untuk Menangani Kemalasan Pada Anak Menggunakan Platform Android**

**Aditya Putra Pratama1, Agi Putra Kharisma2, Randy Cahya Wihandika3**

Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya

Email: 1adit.slash@student.ub.ac.id, 2agi@ub.ac.id, 3rendicahya@ub.ac.id

**Abstrak**

Kemalasan pada anak seringkali timbul akibat tidak adanya motivasi dalam melakukan sesuatu. Dalam studi psikologi menyatakan bahwa kemalasan pada anak timbul akibat tidak adanya kebiasaan dalam belajar yang teratur. Untuk meningkatkan motivasi orang tua perlu memberikan sebuah hadiah agar anak memiliki kemauan dalam mengerjakan sesuatu. Hal ini dilakukan untuk mulai membiasakan anak dalam mengerjakan sesuatu. Pemberian *daily activity* merupakan alat bantu untuk membiasakan anak melakukan kegiatan sehari-hari. Penelitian ini menggunakan metode pengembangan perangkat lunak *Waterfall*. Metode ini dipilih karena kebutuhan fungsional pada sistem sudah didefinisikan sejak awal dan tidak memiliki perubahan. Pengujian yang dilakukan pada aplikasi Daily Quest adalah pengujian validasi, usabilitas, dan kompatibilitas. Pada pengujian validasi menghasilkan tingkat keberhasilan 100%. Pengujian usabilitas menghasilkan *success rate* sebesar 82,8% pada sistem orang tua dan 91,6% pada sistem anak. Pengujian dengan metode *system usability scale* (SUS) menghasilkan nilai rata-rata sebesar 66 pada sistem orang tua, golongkan ke dalam kategori *grade* C yaitu *marginal* yakni berada diantara dapat diterima dan tidak. Sedangkan pada sistem anak menghasilkan nilai rata-rata sebesar 73, digolongkan ke dalam kategori *grade* B- yaitu *acceptable* yakni dapat diterima. Pengujian kompatibilitas sistem, aplikasi Daily Quest hanya dapat berjalan dengan baik pada versi android lollipop 5.0 dengan tingkatan API minimal 21.

**Kata kunci**: *reinforcement, android, aplikasi perangkat bergerak, firebase, parental controll*

**Abstract**

*Laziness in children often arises due to lack of motivation in doing something. In psychology, studies state that laziness in children arises due to the absence of habits in regular learning. To increase motivation, parents need to give a gift so that children have the will to do something. This is done to start getting children used to doing things. The provision of daily activity is a tool to get children used to doing daily activities. This research uses the Waterfall software development method. The method is chosen because the functional requirements of the system have been defined from the beginning and have no change. Tests conducted on the Daily Quest application are validation, usability, and compatibility testing. In validation testing results in a 100% success rate. Usability testing results in a success rate of 82.8% for the parent system and 91.6% for the child system. The system usability scale (SUS) method produces an average value of 66 in the parent system, classifying it into the C grade category, which is marginal. Whereas in the child system produces an average value of 73, classified into the category grade B- that is acceptable. In system compatibility testing, the Daily Quest application can only run well on Android Lollipop 5.0 (API level 21 or greater).*

**Keywords**: *reinforcement, android, mobile application, firebase, parental controll*

# PENDAHULUAN

Gawai merupakan salah satu teknologi yang paling banyak digunakan untuk saat ini. Lembaga riset digital marketing Emarketer memperkirakan pengguna ponsel pintar yang ada di Indonesia memiliki pengguna aktif lebih dari 100 juta orang pada tahun 2018 (Rahmayani, 2015). Fenomena tersebut membuat Indonesia menjadi salah satu negara terbesar ke empat dalam pengunaan ponsel pintar (Rahmayani, 2015). Aktivitas pengguna ponsel pintar ini hampir dilakukan setiap hari dan setiap saat. Banyaknya orang tua yang memanfaatkan ponsel pintar sebagai pendamping atau pengasuh anaknya. Dengan memanfaatkan berbagai fitur yang tersedia pada ponsel pintar sehingga orang tua dapat dengan bebas dan leluasa menjalani aktifitas kesehariannya (Chusna, 2017).

Pengaruh psikologi bagi anak yang paling sering kita temui adalah kemalasan. Kemalasan pada anak dalam melakukan suatu hal, seringkali timbul akibat tidak adanya motivasi dalam melakukan hal tersebut. Anak pun lebih memilih bermain gawai dari pada melakukan aktivitas lain. Tanpa disadari, anak pun menjadi terisolasi dari kehidupan sosialnya, fenomena ini disebabkan karena ketergantungan si anak pada permainan yang ada pada gawainya (Turangan, 2016). Salah satu yang sering kita temui adalah kemalasan dalam belajar pada anak. Dalam studi psikologi menyatakan bahwa kemalasan pada anak timbul akibat tidak adanya kebiasaan dalam belajar yang teratur. Orang tua perlu memberikan hadiah pada anak untuk meningkatkan kemauan atau minat pada anak (Damanik, 2016). Hadiah yang diberikan bertujuan untuk meningkatkan motivasi si anak sehingga dapat mendorong si anak untuk melakukan sesuatu (Ani & Aeni, 2011).

Dari permasalahan yang dipaparkan di atas, maka dikembangkan aplikasi “Daily Quest” yang diharapkan dapat memudahkan orang tua dalam mendidik anak agar dapat menangani kemalasan pada anak. Karena pada dasarnya pemberian *Daily Activity* merupakan alat bantu untuk membiasakan anak melakukan kegiatan sehari-hari (Ani & Aeni, 2011). Dibantu akses internet, orang tua dapat memberikan daftar kegiatan apa saja yang perlu dilakukan oleh anak beserta hadiah yang akan diberikan jika si anak dapat menyelesaikan tugasnya. Dengan memanfaatkan sensor gps, dan kamera pada platform Android dapat menjadi media pelaporan atas setiap kegiatan anak. Dipilihnya platform android disebabkan android kini mendominasi pasar ponsel pintar Indonesia. Di Indonesia, android menguasai lebih dari separuh pasar pasar piranti cerdas. Android sejauh ini telah menguasai lebih dari 90 persen pasar yang ada di Indonesia maupun 75 yang ada di dunia. Dari sekitar 5 juta aplikasi yang berdiri diatas piranti cerdas, 40 persen di antaranya merupakan aplikasi yang berdiri disistem operasi android (Permana, 2019).

Dalam membangun sebuah perangkat lunak, dibutuhkan sebuah metode yang dapat membantu untuk merencanakan, mengelola dan mengontrol setiap proses pengerjaannya. Metode pengembangan perangkat lunak pada aplikasi ini dikembangkan oleh peneliti dengan menggunakan metode *Waterfall*. Metode *Waterfall* merupakan salah satu dari sekian banyak *Software Development Life Cycle* (SDLC) yang memiliki lima fase yang tersusun secara urut, yakni bisnis analisis, perancangan, implementasi, pengujian dan pemeliharaan (Bassil, 2012). Metode *Waterfall* cocok digunakan apabila segala kebutuhan yang diperlukan pada sistem sudah didefinisikan diawal dan tidak memiliki perubahan serta pengerjaannya dapat dilakukan secara berurutan (Pressman, 2010).

Dari permasalahan yang telah diuraikan sebelumnya, maka didapatkan rumusan masalah yakni bagaimana hasil analisis kebutuhan fungsional dan non fungsional dalam merancang aplikasi Daily Quest?, bagaimana hasil dari rancangan aplikasi Daily Quest?, bagaimana hasil implementasi dari aplikasi Daily Quest?, bagaimana hasil uji validasi, usabilitas, dan kompatibilitas pada aplikasi Daily Quest?.

# LANDASAN KEPUSTAKAAN

## 2.1. Penelitian Terkait

Terdapat beberapa penelitian sebelumnya yang relevan dengan penelitian, sehingga dapat membantu dalam pengerjaan penelitian ini.

Penelitian pertama ialah penelitian Agile vs *Waterfall*: A Comparative Analysis yang melakukan perbandingan antara metode pengembangan perangkat lunak Agile dan *Waterfall* sebagai landasan untuk pemilihan metode yang akan digunakan dalam pengembangan sebuah perangkat lunak. Pada penelitian tersebut disimpulkan bahwa dalam pengembangan perangkat lunak yang berbeda maka perlu ditangani dengan cara yang berbeda. Jika suatu proyek pengembangan perangkat lunak memiliki kebutuhan fungsional yang tidak berubah maka dapat menggunakan metode *Waterfall*, dan jika kebutuhan fungsional dapat berubah pada saat proses pengerjaan maka Agile merupakan pilihan metode yang tepat (Kannan, Smita, & Verma, 2014).

Penelitian kedua adalah penelitian Aplikasi Home Security dengan menggunakan Internet of Things (IoT) dan Firebase. Pada penelitian tersebut, pengembang menjelaskan kelebihan dari penggunaan Cloud Message sebagai perantara dalam mengirimkan notifikasi yang disebabkan oleh perubahan data yang ada pada Realtime-Database. Penelitian ini menyatakan bahwa Cloud Function sangat mudah untuk dimanipulasi dan disesuaikan dengan kebutuhan (Sarkar, Gayen, & Bilgaiyan, 2018).

Penelitian terakhir adalah penelitian Aplikasi Mobile Tracer Study yang menggunakan Platform Android merupakan aplikasi untuk menghubungan mahasiswa yang baru saja lulus dengan alumni lainnya. Penelitian tersebut menggunakan platform firebase sebagai pendukung dalam pengembangan aplikasi (Almasyhur, 2018).

## 2.2. *Reinforcement*

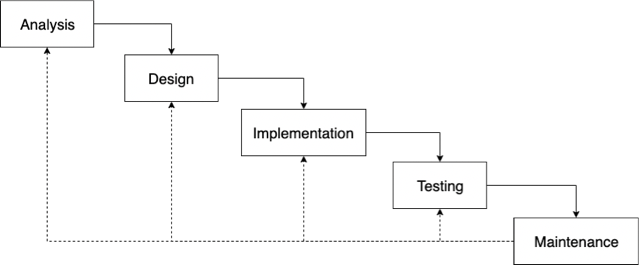
Reinforcement merupakan sebuah istilah untuk memperkuat suatu perilaku. Tidak semua perilaku yang diberikan hadiah dapat menyenangkan bagi orang tersebut. Pada dasarnya hal yang dapat memperkuat sesuatu merupakan hal yang sangat dibutuhkan sebagai makhluk hidup, seperti makanan, minuman, seks, dan perawatan dari orang tua yang dapat membuat keberlangsungan hidup (Feist & Feist, 2008).

Makanan merupakan reinforcement (Penguat) untuk manusia dan hewan karena dapat menghilangkan kondisi kekurangan. Dalam konteks ini uang temasuk ke dalam Conditioned reinforcement. Conditioned reinforcers juga disebut sebagai Secondary reinforcers yang secara tidak langsung dapat memberikan rangsangan karena memiliki hubungan dengan Primary reinforcers seperti makanan, minuman, seks dan kenyamanan secara fisik, sehingga dapat dijadikan sebagai pengganti Primary reinforcers. Terkadang uang juga dikatakan sebagai Generalized reinforcers karena berhubungan lebih dari satu Primary reinforcers (Feist & Feist, 2008).

Dalam pemberian hadiah terdapat teknik penjadwalan yang dapat digunakan, salah satunya ialah *variable-ratio.* Teknik *Variable-ratio* ini merupakan metode penjadwalan untuk memberikan hadiah secara berkelanjutan dimulai dengan nilai yang rendah hingga lebih tinggi. Sebaiknya pemberian hadiah dilakukan secara bertahap agar tidak melemahkan perilaku yang telah dilatih sebelumnya (Feist & Feist, 2008).

## 2.3. *Waterfall*

*Waterfall* merupakan salah satu dari siklus pengembangan perangkat lunak (*Software Development Life Cycle*) yang merupakan metodologi yang cepat untuk merancang dan membangun perangkat lunak. *Waterfall* bekerja secara berurutan dimana *waterfall* memiliki lima fase yang harus dilalui untuk mengembangkan suatu perangkat lunak (Bassil, 2012). Fase-fase tersebut menggambarkan kemajuan dalam setiap proses pengembangannya yang digambarkan mirip seperti air terjun yang digambarkan pada Gambar 1.



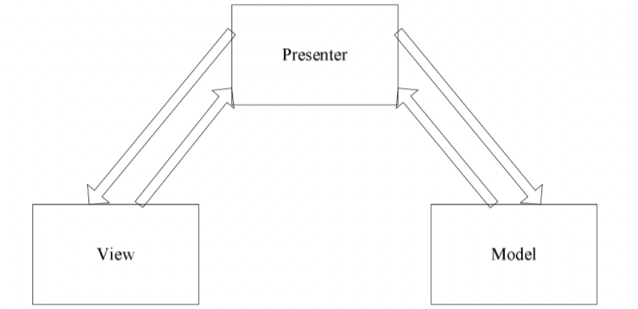
Gambar 1 Siklus Pengembangan Waterfall

## 2.4. *Best Practice*

Dalam membangun aplikasi perangkat bergerak, diperlukan sebuah arsitektur perangkat lunak untuk memudahkan pengembang dalam membangun aplikasi tersebut.

## 2.4.1. Model View Presenter *(MVP)*

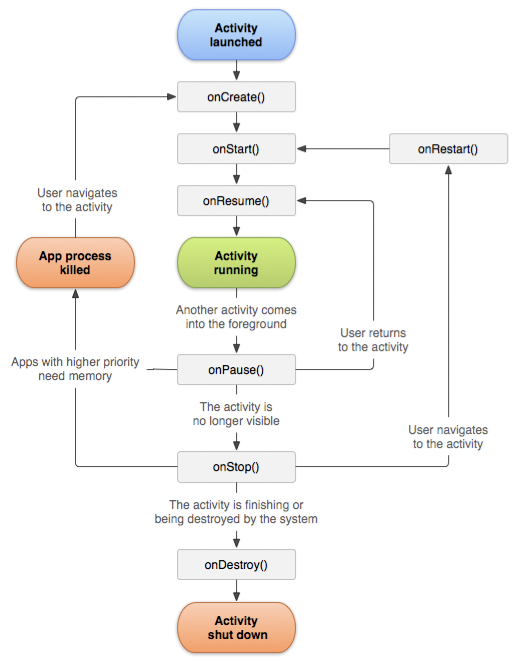
Masalah yang sering timbul untuk para pengembang agar aplikasi dapat dengan mudah dikembangkan dan dipelihara, maka aplikasi yang dibangun harus memiliki lapisan yang terpisah. Model View Presenter (MVP) merupakan sebuah arsitektur yang digunakan dalam mengembangkan perangkat lunak, sesuai dengan namanya MVP dibagi menjadi tiga bagian, di antaranya ialah Model, View dan Presenter. Model berperan sebagai representasi dari bisnis objek, View berfungsi menangani semua UI komponen dan meneruskan data yang akan tampil dari Presenter, dan Presenter berisikan seluruh logika yang ada pada aplikasi. Arsitektur ini terbilang mudah digunakan untuk para pengembang (Ojeda-Guerra, 2015).



Gambar 2 MVP Arsitektur

## 2.4.2. Android

Sistem operasi android merupakan pengembangan dari Linux kernel 2.6 yang berjalan pada smartphone. Android dikembangkan oleh Google yang kemudian Open Handset Alliance manjadi pengembang berikutnya. Android dapat berjalan diberbagai perangkat smartphone sehingga Android digunakan oleh banyak perusahaan smartphone sebagai sistem operasinya (Kirthika & Prabhu, 2015). Berikut adalah gambaran dari *lifecycle* android yang dapat dilihat pada Gambar 3.



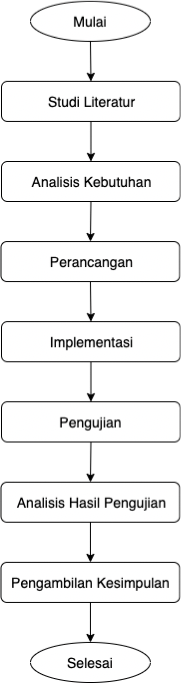
Gambar 3 Android Activity Lifecycle

# 2.4.3. Firebase

Firebase merupakan sebuah mobile platform yang dibangun oleh Google untuk membantu developer dalam mengembangkan aplikasi dengan kualitas tinggi dan sangat mudah digunakan. Firebase memiliki banyak fitur didalamnya seperti realtime-database, cloud Messaging, authentication, dan cloud storage. Digunakannya firebase karena selain mudah untuk di implementasikan, firebase juga dapat membesar secara otomatis jika memang dibutuhkan oleh aplikasi. Fiture authentication yang dimiliki oleh firebase dapat membantu dalam mengamankan verifikasi dan validasi. Ditambah lagi fitur ini dapat memungkinkan developer untuk mengintegrasikan aplikasi dengan Google, Facebook, Twitter, dll. Sedangkan fitur cloud messaging, realtime-database, dan cloud storage dapat membantu developer dalam menangani push notification, database yang berjalan secara realtime dan penyimpanan data (Ramadhan, Tolle, & Fanani, 2018).

# METODOLOGI

Metodologi penelitian dibutuhkan dalam mengembangkan perangkat lunak agar pengembangan dapat terstruktur dengan baik. Alur dari pengembangan pada penelitian ini digambarkan dengan diagram alir penelitian yang ada pada Gambar 4.



Gambar 4 Digram Alir Penelitian

# ANALISIS KEBUTUHAN

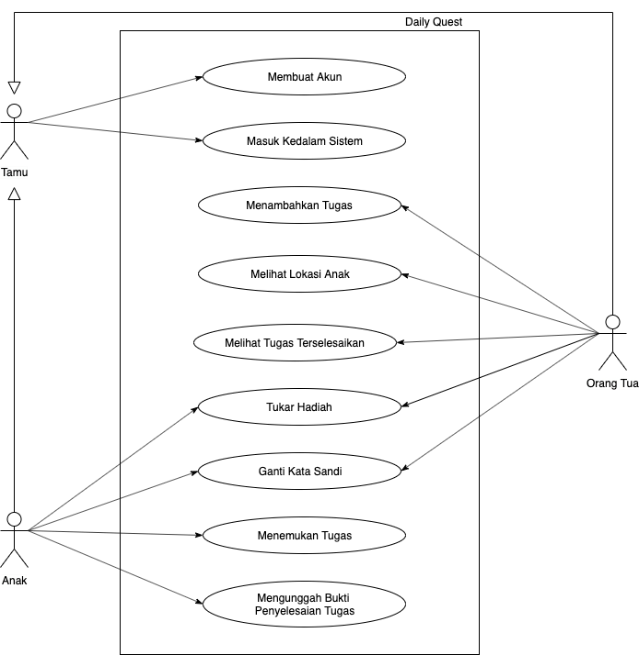
Pada tahapan ini segala kebutuhan dalam sistem akan dijelaskan secara umum dan lengkap. Penentuan lingkungan sistem yang akan dibuat yang kemudian menentukan calon pengguna yang akan berperan sebagai aktor pada aplikasi ini. Setelah itu fungsionalitas akan ditetapkan berdasarkan Studi Literatur. Setelah fungsional didapatkan maka akan digambarkan kedalam *use case* diagram.

Berdasarkan hasil analisis kebutuhan sistem dilakukan identifikasi aktor yang dapat dilihat pada Tabel 1.

Tabel 1 Identifikasi Aktor

|  |  |
| --- | --- |
| **Aktor** | **Deskripsi** |
| Tamu | Tamu merupakan aktor umum yang menggunakan aplikasi ini. |
| Anak | Anak merupakan aktor yang telah teridentifikasi sebagai anak dalam proses autentikasi. |
| Orang tua | Orang tua merupakan aktor yang telah teridentifikasi sebagai orang tua dalam proses autentikasi. |

Didapatkan pula *use case* diagram dari masing-masing aplikasi. *Use case* diagram aplikasi sukarelawan dapat dilihat pada Gambar 5.



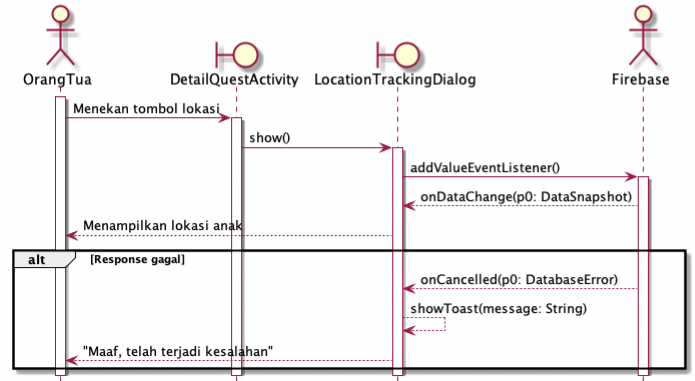
Gambar 5 Use Case Diagram Aplikasi Daily Quest

# PERANCANGAN

Pada tahap perancangan aplikasi Daily Quest terdiri dari perancangan arsitektur sistem, *sequence diagram, class diagram,* perancangan tabel data, perancangan algoritma dan perancangan antar muka.

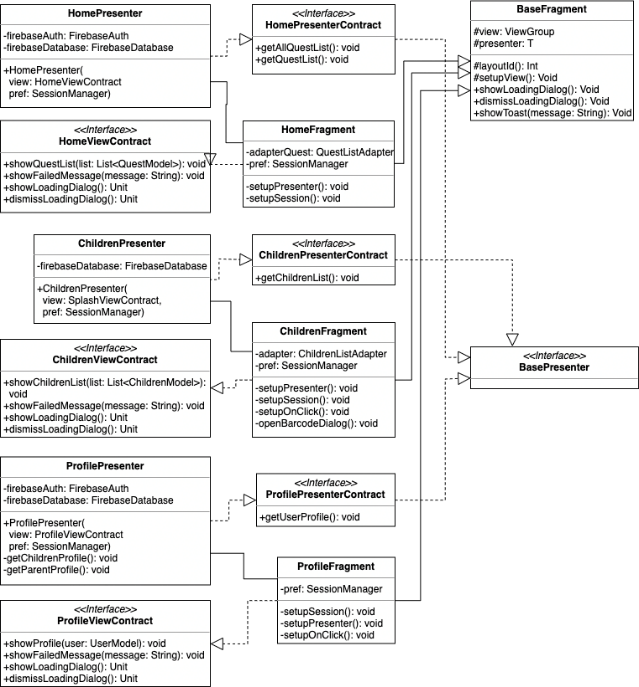
Perancangan arsitektur sistem memberikan gambaran terhadap apa saja yang digunakan dalam aplikasi, baik layanan yang digunakan maupun desain arsitektur dari kode program. Layanan yang digunakan pada aplikasi Daily Quest ialah firebase dan google Maps API. Dengan memanfaatkan GPS untuk mendapatkan posisi dari pengguna yang berupa *latitude* dan *longitude.* Untuk arsitektur kode program digunakan design arsitektur *Model View Presenter (MVP).*

*Sequence diagram* menjelaskan interaksi antar class pada sistem secara berurutan sehingga dapat mencapai setiap tujuan dari kebutuhan fungsional. Contoh dalam perancangan sistem dengan *sequence diagram* digambarkan pada Gambar 6.



Gambar 6 Sequence Diagram Melihat Lokasi Anak

Adapun hubungan antar kelas yang digambarkan dengan *class diagram*. *Class diagram* memberikan gambaran terkait hubungan dari kelas satu dengan kelas lainnya sehingga dapat dijadikan pedoman dalam pengimplementasian sebuah kelas. Contoh dari *class diagram* pada aplikasi Daily Quest digambarkan pada Gambar 7.



Gambar 7 Class Diagram Fragment Dengan Base

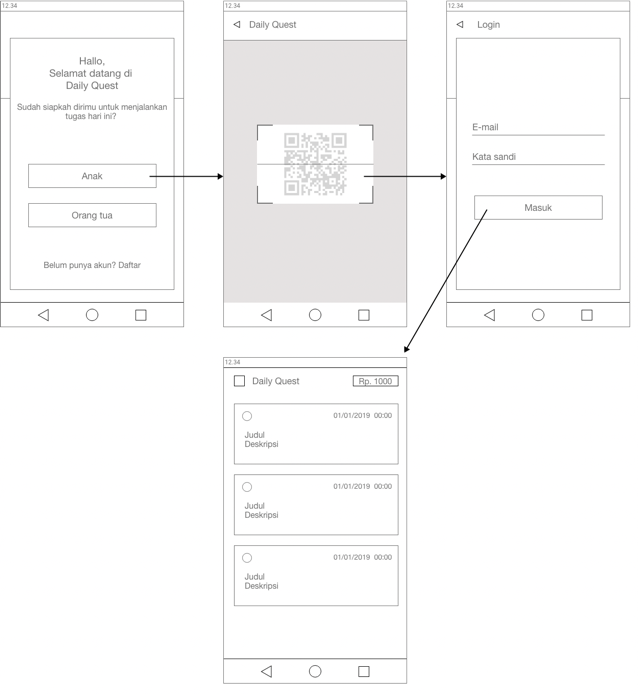
Untuk Perancangan tabel data dapat dilihat pada Tabel 2.

Tabel 2 Rancangan Data Quest

|  |  |  |
| --- | --- | --- |
| **No** | **Atribut** | **Dekripsi** |
| 1 | createAt | Tanggal dan waktu pembuatan tugas |
| 2 | description | Deskripsi dari tugas |
| 3 | hideReward | *Flag* untuk menentukan apakah hadiah akan dimunculkan atau tidak |
| 4 | endTime | Tanggal dan waktu penyelesaian tugas |
| 5 | opened | *Flag* untuk menentukan tugas telah dibuka atau belum |
| 6 | startTime | Tanggal dan waktu mulai tugas |
| 7 | status | Status tugas |
| 8 | title | Judul tugas |
| 9 | image | Uri dari bukti foto yang diunggah |

Selain itu dilakukan perancangan algoritme untuk memberikan gambaran terhadap alur dalam menyelesaikan tugas. Perancangan algoritme dilakukan dengan menuliskan *pseudocode* agar lebih mudah dipahami saat pengimplementasian kode program.

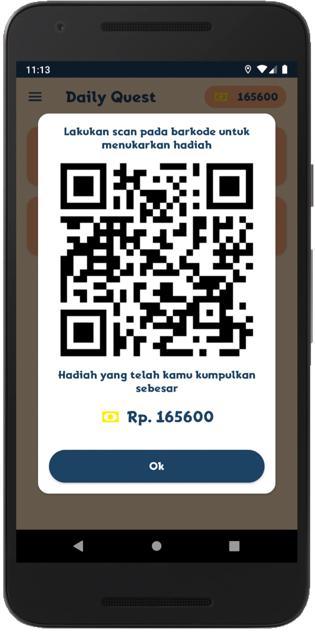
Terakhir, dilakukan perancangan antar muka dengan menggunakan *wireframe* dan *screenflow* yang dapat dilihat pada Gambar 8.



Gambar 8 Alur Antar Muka Masuk ke dalam Sistem

# IMPLEMENTASI

Pada tahap ini, perancangan yang sebelumnya telah dibuat akan menjadi dasar acuan untuk melakukan setiap pengerjaan implementasi perangkat lunak. Implementasi dilakukan berdasarkan *use case* dan sequence diagram yang telah tentukan sebelumnya. Setelah itu pengimplementasian algoritma ke dalam kode program dimana pengimplementasiannya menggunakan bahasa pemrograman kotlin dan antar muka akan diimplementasikan menggunakan bahasa pemrograman xml. Untuk implementasi basis data, otentikasi dan notifikasi akan memanfaatkan layanan yang diberikan firebase yakni realtime-database, authentication, dan cloud messaging. Hasil dari implementasi aplikasi Daily Quest dapat dilihat pada Gambar 9.

Gambar 9 Implementasi Aplikasi Daily Quest

# PENGUJIAN

Pengujian dilakukan untuk memastikan aplikasi yang dibangun dapat berjalan dengan semestinya dan fungsional yang telah ditentukan pada tahap analisis kebutuhan telah sesuai. Pengujian yang dilakukan adalah pengujian fungsional, pengujian usabilitas, dan pengujian kompatibilitas*.*

Hasil yang didapatkan setelah dilakukan pengujian fungsional yakni aplikasi Daily Quest dapat menjalankan seluruh kasus uji fungsionalitas dengan baik dan tanpa adanya kendala sesuai kebutuhan fungsional yang telah ditetapkan. Aplikasi Daily Quest dapat menyelesaikan kasus uji yang dilakukan sesuai prosedur dan hasil yang didapat sesuai dengan hasil yang diharapkan sehingga dapat dikatakan hasil dari kasus uji adalah valid. Maka didapatkan kesimpulan yaitu aplikasi Daily Quest memiliki tingkat keberhasilan sebesar 100% pada seluruh fungsionalitas yang ditetapkan.

Pengujian usabilitas ditujukan untuk mengetahui kemudahan pengguna dalam menggunakan aplikasi Daily Quest. Pengujian ini menggunakan metode System Usability Scale (SUS) yang dilakukan kepada 10 orang responden yang terdiri dari 5 orang tua dan 5 orang anak. Pengujian dilakukan dengan memberikan kasus uji yang mengacu pada fungsionalitas aplikasi Daily Quest dan responden diberikan sedikit clue sebelum mencoba aplikasi Daily Quest.

Dari percobaan yang dilakukan oleh responden akan dibagi menjadi 3 bagian yakni berhasi, hampir berhasil, dan gagal. Berhasil merupakan kondisi dimana responden dapat menyelesaikan tugas dengan baik, sedangkan hampir berhasil adalah kondisi dimana pengguna dapat melakukan lebih dari 50% dari tugas yang diberikan, dan kondisi gagal adalah kondisi dimana pengguna tidak dapat menyelesaikan tugas. Untuk mendapatkan success rate maka jika berhasil akan dikalikan 1, jika hampir berhasil akan dikalikan 0.5, dan jika gagal akan dikalikan 0. Setelah dilakukan perkalian maka total nilai akan dibagi dengan jumlah kemungkinan, hasil akhir yang didapatkan dengan dikalikan dengan 100% (Nielsen, 2001). Untuk lebih jelasnya maka akan ditunjukan pada Persamaan (1).

|  |  |
| --- | --- |
|  | (1) |

*SR = Success* *rate*

*B* = Jumlah benar

*H* = Jumlah hampir berhasil

*T* = Tugas

*R* = Responden

Pada sistem orang tua menghasilkan success rate sebesar 82,8% yang didapatkan dari total nilai *B* yakni 27 yang dikalikan dengan 1, dan total nilai *H* yakni 4 yang dikalikan 0.5, yang selanjutnya dibagi dengan total pertanyaan berjumlah 7 pertanyaan yang dikalikan dengan banyaknya responden berjumlah 5 orang, setelah didapatkan hasil maka dikalikan dengan 100%.

Pada sistem anak menghasilkan success rate sebesar 91,6% yang didapatkan dari total nilai *B* yakni 25 yang dikalikan dengan 1, dan total nilai *H* yakni 5 yang dikalikan 0.5, yang selanjutnya dibagi dengan total pertanyaan berjumlah 6 pertanyaan yang dikalikan dengan banyaknya responden berjumlah 5 orang, setelah didapatkan hasil maka dikalikan dengan 100%.

Hasil yang didapatkan pada pengujian dengan metode system usability scale (SUS) pada pengguna orang tua menghasilkan nilai rata-rata 66 yang jika dimasukan pada skala konversi SUS dinyatakan memiliki grade C yaitu marginal yang berarti berada diantara dapat diterima atau tidak.

Hasil yang didapatkan pada pengujian dengan metode system usability scale (SUS) pada pengguna anak menghasilkan nilai rata-rata 73 yang jika dimasukan pada skala konversi SUS dinyatakan memiliki grade B- yaitu acceptable yang berarti dapat diterima.

# KESIMPULAN DAN SARAN

Dari hasil yang telah didapatkan berdasarkan tahap analisis kebutuhan, perancangan, implementasi, dan pengujian yang telah selesai dilakukan sebelumnya, maka didapatkan kesimpulan sebagai berikut:

Analisis kebutuhan pada aplikasi Daily Quest didapatkan berdasarkan studi pustaka metode *reinforcement* yang menghasilkan 9 kebutuhan fungsional yang dibagi menjadi 3 aktor yakni tamu sebagai pengguna aplikasi yang belum terautentikasi, orang tua sebagai pengguna yang terautentikasi sebagai orang tua, dan anak sebagai pengguna yang terautentikasi sebagai anak serta menghasilkan 2 kebutuhan non fungsional yakni dari segi usabilitas dan kompatibilitas.

Hasil rancangan pada aplikasi Daily Quest yakni berupa perancangan arsitektur sistem, *sequence diagram, class diagram*, perancangan tabel data, perancangan algoritma dalam bentuk *pseudocode* dan perancangan antarmuka yang digambarkan dalam bentuk *wireframe*.

Implementasi sistem dilakukan pada aplikasi mobile dengan sistem operasi android yang memanfaatkan layanan firebase, google maps APIs, dan kamera yang tersedia pada ponsel cerdas. Implementasi layanan yang digunakan dapat dilihat dalam kode program yang ditulis dengan menggunakan bahasa pemrograman kotlin.

Pengujian validasi yang dilakukan dengan mengujikan fungsionalitas pada sistem menghasilkan bahwa aplikasi Daily Quest memiliki tingkat keberhasilan 100% pada fungsionalitasnya. Pada pengujian usabilitas didapatkan *success rate* sebesar 82,8% pada sistem orang tua dan 91,6% pada sistem anak. Pengujian dengan metode system usability scale (SUS) menghasilkan nilai rata-rata sebesar 66 pada sistem orang tua yang berarti masuk ke dalam kategori *grade* C yaitu *marginal* yakni berada diantara dapat diterima dan tidak. Sedangkan pada sistem anak menghasilkan nilai rata-rata sebesar 73 yang berarti masuk ke dalam kategori *grade* B- yaitu *acceptable* yakni dapat diterima. Serta pada pengujian kompatibilitas sistem, aplikasi Daily Quest hanya dapat berjalan dengan baik pada versi android *lollipop* 5.0 dengan tingkatan API minimal 21.

# 9. DAFTAR PUSTAKA

Almasyhur, A. K., 2018. Pengembangan Aplikasi Mobile Tracer Study Menggunakan Platform Android. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer (J-PTIIK) Universitas Brawijaya*, *2*(11), 5402–5409.

Ani, O., & Aeni, N., 2011. Menanamkan Disiplin Pada Anak. *Pendidikan Agama Islam -Ta’lim*, *9*(1), 17–29.

Bassil, Y., 2012. A Simulation Model for the Waterfall Software Development Life Cycle. *International Journal of Engineering & Technology (IJET)*, *02*(05), 2049–3444.

Chusna, P. A., 2017. Pengaruh Media Gadget Pada Perkembangan Karakter Anak. *STIT Al-Muslihun*, *17*, 32–117.

Damanik, R. S., 2016. Seni Memberikan Reward kepada Anak. Kompasiana, [online] Tersedia di: <https://www.kompasiana.com/rahayusetiawatidamanik/585b0e196e7e616b068b4567/seni-memberikan-reward-kepada-anak> [Diakses 15 September 2019]

Feist, J., & Feist, G., 2008. *Theories of Personality*.

Kannan, V., Smita, J., & Verma, S., 2014. Agile vs waterfall : A Comparative Analysis. *International Journal of Science, Engineering and Technology Research (IJSETR)*, *3*(10), 2680–2686.

Kirthika, B., & Prabhu, S., 2015. Android Operating System : A Review. *International Journal of Trend in Research and Development*, *2*(5), 260–264.

Nielsen, J., 2001. Success Rate: The Simplest Usability Metric. Nielsen Norman Group, [online] Tersedia di: <https://www.nngroup.com/articles/success-rate-the-simplest-usability-metric> [Diakses 11 Desember 2019]

Ojeda-Guerra, C. N., 2015. A Simple Software Development Methodology Based on MVP for Android Applications in a Classroom Context. *2015 IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing*, 1429–1434.

Permana, K. S., 2019. Ketika Pengguna Internet dan Smartphone Terus Meningkat, Android Dominasi Pasar Indonesia dan Dunia. Jabar Tribun News, [online] Tersedia di: <http://jabar.tribunnews.com/2019/01/24/ketika-pengguna-internet-dan-smartphone-terus-meningkat-android-dominasi-pasar-indonesia-dan-dunia> [Diakses 16 September 2019]

Pressman, R., 2010. *Software Engineering: A Practitioner’s Approach* (7th ed.). New York, NY, USA: McGraw-Hill, Inc.

Rahmayani, I. 2015. Indonesia Raksasa Teknologi Digital Asia. KOMINFO, [online] Tersedia di: <https://www.kominfo.go.id/content/detail/6095/indonesia-raksasa-teknologi-digital-asia/0/sorotan\_media> [Diakses 14 September 2019]

Ramadhan, A. O., Tolle, H., & Fanani, L., 2018. Pembangunan Modul Penunjang Pembelajaran di Kelas Untuk Aplikasi Brawijaya Messenger Dengan Platform Firebase. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer (J-PTIIK) Universitas Brawijaya*, *2*(4), 1630–1637.

Sarkar, S., Gayen, S., & Bilgaiyan, S. 2018. Android Based Home Security Systems Using Internet of Things(IoT) and Firebase. *Proceedings of the International Conference on Inventive Research in Computing Applications, ICIRCA 2018*, (Icirca), 102–105.

Turangan, L. 2016. Efek Negatif Gadget Pada Anak. Kompas, [online] Tersedia di: <https://lifestyle.kompas.com/read/2016/02/17/081500023/Efek.Negatif.Gadget.pada.Anak> [Diakses 14 September 2019]