



IMPLEMENTASI REACT JS DAN FIREBASE UNTUK PENGEMBANGAN APLIKASI PERENCANAAN KEUANGAN PEGAWAI DI NURUL FIKRI ACADEMY

Nelan¹, Nasrul², Reza Maulana³

^{1,2,3}Teknik Informatika, Sekolah Tinggi Teknologi Terpadu Nurul Fikri
Jakarta Selatan, DKI Jakarta, Indonesia 12640

nela21189ti@student.nurulfikri.ac.id, nasrul@nurulfikri.ac.id, rezamaulana@nurulfikri.ac.id

Abstract

Effective financial planning is an important aspect in supporting employee welfare. Within Nurul Fikri Academy, there is no integrated digital system that enables employees to conduct financial planning independently. This research aims to develop a web-based financial planning application designed to record, monitor, and analyze employee financial conditions. The application is built using the React JS framework for an interactive user interface and Firebase as the backend, providing authentication and real-time data storage. The development method used is Extreme Programming (XP), through the stages of planning, designing, coding, and testing. The main features of the application include financial calculators (emergency funds, pensions, and investments), user discussion forums, and financial education articles. Testing was conducted using Blackbox Testing and User Acceptance Testing (UAT) methods. The test results show that all features run well across various devices, with a UAT success rate of 90%, exceeding the minimum threshold of 80%. This application is expected to be an effective solution for improving employees' literacy and financial management skills in a structured, efficient manner.

Keywords: Employee, Extreme Programming, Financial Planning, Firebase, React JS

Abstrak

Perencanaan keuangan yang efektif merupakan aspek penting dalam mendukung kesejahteraan pegawai. Di lingkungan Nurul Fikri Academy, belum tersedia sistem digital terintegrasi yang dapat membantu pegawai dalam melakukan perencanaan keuangan secara mandiri. Penelitian ini bertujuan mengembangkan aplikasi perencanaan keuangan berbasis web yang dirancang untuk mencatat, memantau, dan menganalisis kondisi keuangan pegawai. Aplikasi dibangun menggunakan *framework* React JS untuk antarmuka pengguna yang interaktif, serta Firebase sebagai *backend* yang menyediakan autentikasi dan penyimpanan data secara *real-time*. Metode pengembangan yang digunakan adalah *Extreme Programming* (XP), melalui tahapan perencanaan, perancangan, pengkodean, dan pengujian. Fitur utama aplikasi mencakup kalkulator keuangan (dana darurat, pensiun, dan investasi), forum diskusi antar pengguna, serta artikel edukasi keuangan. Pengujian dilakukan dengan metode *black box testing* dan *User Acceptance Testing* (UAT). Hasil pengujian menunjukkan bahwa seluruh fitur dapat berjalan baik di berbagai perangkat, dengan tingkat keberhasilan UAT mencapai 90%, melebihi ambang batas minimal 80%. Aplikasi ini diharapkan dapat menjadi solusi yang efektif dalam meningkatkan literasi dan keterampilan pengelolaan keuangan pegawai secara terstruktur dan efisien.

Kata kunci: *Extreme Programming*, Firebase, Pegawai, Perencanaan Keuangan, React JS

1. PENDAHULUAN

Otoritas Jasa Keuangan (OJK) merupakan lembaga pemerintah yang bertanggung jawab atas pengawasan dan pengaturan sektor jasa keuangan di Indonesia. OJK memiliki visi untuk menjadi lembaga pengawasan terpercaya yang mendukung pertumbuhan industri jasa keuangan secara berkelanjutan dan berkontribusi terhadap kesejahteraan masyarakat. Salah satu misinya adalah

mendorong terciptanya sistem keuangan yang transparan, akuntabel, dan mampu melindungi kepentingan konsumen [1]. Namun, berdasarkan survei OJK pada tahun 2022, tingkat literasi keuangan masyarakat Indonesia hanya mencapai 49,68% [2]. Capaian tersebut mencerminkan bahwa sebagian besar masyarakat masih belum memahami prinsip dasar pengelolaan keuangan pribadi, yang berdampak pada kurangnya dana darurat, perencanaan

pensiun yang lemah, dan keputusan investasi yang kurang optimal.

Di lingkungan Nurul Fikri Academy, sebagai lembaga pendidikan dan pelatihan teknologi informasi [3], permasalahan serupa juga ditemukan. Sebagian besar pegawai tidak memiliki latar belakang manajemen keuangan, sehingga mengalami kesulitan dalam menyusun perencanaan finansial yang baik. Tekanan gaya hidup dan rendahnya kesadaran terhadap pentingnya perencanaan jangka panjang turut memperburuk kondisi ini.

Menanggapi permasalahan tersebut, penelitian ini mengembangkan sebuah aplikasi perencanaan keuangan berbasis web untuk meningkatkan literasi dan kemampuan pengelolaan keuangan pegawai di Nurul Fikri Academy. Aplikasi ini dibangun menggunakan *framework* React JS untuk antarmuka pengguna dan Firebase sebagai *backend*, dengan pendekatan pengembangan perangkat lunak *Extreme Programming* (XP). Fitur utama aplikasi mencakup kalkulator dana darurat, investasi, dan pensiun, forum diskusi, serta artikel edukatif yang dikategorikan. Dengan adanya aplikasi ini, diharapkan pegawai dapat lebih mudah dan efektif dalam mengelola keuangan secara terstruktur dan berkelanjutan.

React JS adalah pustaka JavaScript sumber terbuka (*Open Source*) yang diperuntukkan dalam membuat antarmuka pengguna (*User Interface*) untuk halaman dalam suatu aplikasi. React JS pertama kali diperkenalkan sebagai pustaka JavaScript yang dirancang dan dikaji lebih lanjut oleh Jordan Walke kurang lebih tahun 2013. React JS digunakan untuk pengembangan *frontend* dan berfungsi mengatur lapisan visual aplikasi versi *desktop* dan *mobile* [4]. React JS bertujuan untuk menyediakan kemudahan, kejelasan, dan kemampuan untuk mengatasi skalabilitas [5].

Penggunaan React JS dalam pengembangan aplikasi perencanaan keuangan berbasis *web* bertujuan untuk menciptakan antarmuka yang menarik dan responsif, serta memastikan skalabilitas aplikasi yang optimal. Dengan kemampuan React JS dalam membangun komponen yang *reusable*, pengembang dapat meningkatkan efisiensi dalam pengembangan dan pemeliharaan aplikasi, sehingga aplikasi dapat dengan mudah disesuaikan seiring dengan kebutuhan pengguna yang terus menerus berkembang.

Firebase merupakan sebuah *service* yang diberikan oleh Google yang digunakan para pengembang perangkat lunak agar dapat mempermudah dalam proses pengerjaannya. Firebase, juga disebut sebagai *Backend as a Service* (BaaS), merupakan sebuah *service* yang dikembangkan oleh Google dengan tujuan meningkatkan efektivitas dalam proses pengembangan perangkat lunak [6]. Firebase memungkinkan pengembang untuk fokus mengembangkan perangkat lunak mereka tanpa terlalu mengkhawatirkan masalah *backend*.

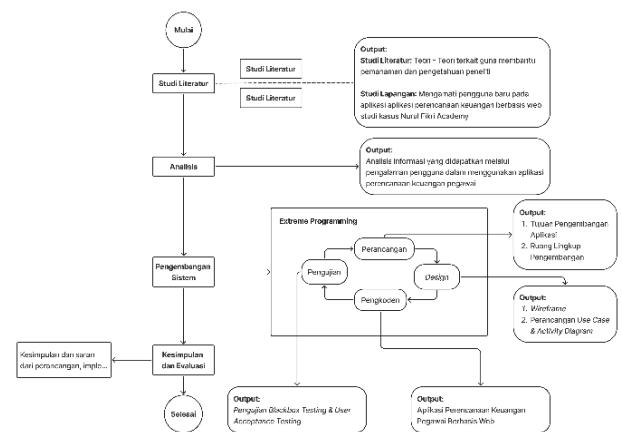
Sebuah metode pengembangan *software* untuk visualisasi, merancang dan mendokumentasikan sistem yang menggunakan konsep pemrograman berorientasi objek yang saling berintegrasi dengan pemrograman yang dijadikan sebagai dasar pengambilan keputusan dari tahap analisis hingga implementasi sistem merupakan definisi dari *Unified Model Language* (UML) [7]. Salah satu jenis UML yang merepresentasikan kasus yang terjadi secara garis besar ialah *use case diagram*. Salah satu jenis UML yang memvisualisasikan alur aktivitas dalam suatu proses kerja ialah *activity diagram*.

2. METODE PENELITIAN

Penelitian ini menggunakan metode *Research and Development* (R&D) ialah sebuah pendekatan yang dilakukan secara sistematis guna mengembangkan produk baru atau menyempurnakan produk yang telah tersedia. Tujuan utama dari pendekatan ini untuk memastikan bahwa produk yang valid, dapat diukur, dan dipertanggungjawabkan secara ilmiah [8]. Adapun langkah - langkah dalam penelitian ini akan diuraikan meliputi tahapan penelitian, metode pengumpulan data, dan metode pengujian.

2.1 Tahapan Penelitian

Tahapan penelitian dalam studi ini dirancang secara sistematis dan terstruktur untuk memastikan bahwa seluruh proses pengembangan aplikasi dapat berjalan secara efektif dan sesuai dengan tujuan yang ditetapkan. Setiap tahapan dirancang untuk saling melengkapi, sehingga mendukung kelancaran proses dari awal perencanaan hingga tahap evaluasi akhir. Adapun tahapan-tahapan yang dilaksanakan dalam penelitian ini meliputi:



Gambar 1. Tahapan Penelitian

Pada Gambar 1 menampilkan beberapa tahapan penelitian, yang akan diuraikan sebagai berikut:

2.1.1 Studi Literatur

Sebagai bagian dari penelitian ini, studi literatur dilakukan untuk memahami teori-teori terkait yang digunakan guna membantu pemahaman dan pengetahuan peneliti. Teori

berfokus pada identifikasi solusi teknologi baru terkait pengembangan aplikasi perencanaan keuangan pegawai berbasis web menggunakan *framework* React JS dan Firebase. Selain itu juga dilakukan studi lapangan dimana observasi dilakukan melalui *website* aplikasi perencanaan keuangan pegawai berbasis *web menggunakan framework* React JS dan Firebase untuk mengidentifikasi kebutuhan pengguna di lingkungan Nurul Fikri Academy.

2.1.2 Analisis

Analisis proses pengembangan aplikasi perencanaan keuangan pegawai berbasis web diperoleh melalui umpan balik dari pengguna ketika menggunakan aplikasi tersebut. Umpan balik ini merupakan aspek krusial bagi pengembangan aplikasi perencanaan keuangan berbasis web menggunakan *framework* React JS dan Firebase agar dapat memenuhi kebutuhan pengguna di lingkungan Nurul Fikri Academy. Dengan mengumpulkan dan menganalisis respons dari pengguna, pengembang dapat mengidentifikasi berbagai aspek yang perlu ditingkatkan baik dari segi fitur, antarmuka pengguna (UI/UX), dan kinerja aplikasi. Selain itu, proses ini membantu mengoptimalkan pengalaman pengguna dan memastikan bahwa aplikasi berjalan secara efektif.

2.1.3 Pengembangan Sistem

Pengembangan sistem dilakukan dengan menggunakan metode *Extreme Programming (XP)*. *Extreme Programming* merupakan sebuah pendekatan pengembangan *software* berbasis *Agile* yang paling sering diterapkan, terutama pada kasus pengembangan aplikasi skala kecil. Metode XP cukup ringkas, namun dapat menerapkan macam-macam konsep *Agile* yang dianggap *breakthrough* dalam pengerjaan pengembangan perangkat lunak yang efektif dan efisien. Metode pemrograman ekstrem lebih cocok ketika persyaratan proyek ambigu dan mengalami perubahan secara dinamis dalam waktu yang singkat [9]. Secara spesifik tahapan pengembangan perangkat lunak menggunakan sebuah metodologi *Extreme Programming* sebagai berikut:

a. Perencanaan (*Planning*)

Fase perencanaan merupakan tahapan awal dalam tahapan pengembangan aplikasi dan melibatkan berbagai aktivitas perencanaan, seperti: mengidentifikasi masalah, menganalisis kebutuhan, dan rencana implementasi pengembangan aplikasi.

b. Perancangan (*Design*)

Fase perancangan dilaksanakan kegiatan pemodelan, di antaranya perancangan sistem, struktur arsitektur, serta basis data. Untuk pemodelan sistem dan arsitektur menggunakan diagram UML (*Unified Modeling Language*), ERD (*Entry Relationship Diagram*) digunakan untuk pemodelan basis data.

c. Pengkodean (*Coding*)

Fase pengkodean mengimplementasi model yang telah dibuat menggunakan bahasa pemrograman dan diterjemahkan ke dalam antarmuka pengguna (*User Interface*). Bahasa pemrograman yang digunakan ialah JavaScript dengan *library* React JS serta Firebase digunakan sebagai *database*.

d. Pengujian (*Testing*)

Setelah tahap pengkodean selesai, tahap selanjutnya adalah pengujian, fase pengujian mengidentifikasi kesalahan atau masalah apa pun yang terjadi sebelum proses *deployment*. Untuk fase pengujian yang digunakan ialah *black box testing*.

2.1.4 Kesimpulan dan Evaluasi

Melalui kesimpulan dan evaluasi yang diperoleh melalui aplikasi perencanaan keuangan pegawai berbasis web yang dikembangkan, peneliti telah berhasil membantu pengguna di lingkungan Nurul Fikri Academy dalam meningkatkan literasi keuangan mereka. Fitur-fitur seperti kalkulator investasi, kalkulator dana darurat, dan kalkulator pensiun memberikan panduan yang jelas dan mudah diakses untuk perencanaan keuangan darurat dan jangka panjang, yang sangat dibutuhkan oleh pegawai yang belum memiliki kecakapan mendalam tentang pengelolaan keuangan yang memadai.

2.2 Metode Pengumpulan Data

Pada metode pengumpulan data peneliti menggunakan metode kualitatif. Metode kualitatif ialah pendekatan yang menggunakan data meliputi wawancara, observasi dokumen, serta pengalaman pribadi guna memperoleh pemahaman dan memberikan penjelasan fenomena sosial. Pada prosesnya, dalam mengumpulkan data melalui interaksi antar pegawai Nurul Fikri Academy adalah sebagai berikut:

a. Wawancara

Tahapan wawancara digunakan dalam pengembangan aplikasi perencanaan keuangan pegawai berbasis web di Nurul Fikri Academy dengan tujuan untuk mendapatkan informasi langsung dari para pegawai, menanyakan apa saja kebutuhan mereka, apa yang mereka inginkan dari aplikasi, dan masalah apa yang mereka hadapi dalam merencanakan keuangan. Dengan cara ini, peneliti bisa mendapatkan masukan yang jelas dan bermanfaat untuk memastikan aplikasi yang dikembangkan sesuai dengan kebutuhan mereka, serta mudah digunakan dan berfungsi dengan baik.

b. Observasi

Metode observasi digunakan saat proses pengembangan aplikasi perencanaan keuangan. Observasi dilakukan dengan tujuan mengidentifikasi kendala yang dialami oleh pengguna, mengevaluasi efektivitas aplikasi, dan

mengetahui bagaimana fitur - fitur dapat digunakan dalam kehidupan sehari - hari.

2.3 Metode Pengujian

Pada penelitian kali ini, metode pengujian yang akan dilakukan yakni dengan dua tahapan antara lain menggunakan *Blackbox Testing* dan *User Acceptance Testing (UAT)*. Uraian lengkapnya sebagai berikut:

a. Black Box Testing

Pengujian *black box* merupakan metode pengujian perangkat lunak yang digunakan dalam pengujian perangkat lunak. Pada proses pengujiannya melibatkan program yang dibuat dengan memasukkan *input* ke dalam *form*. Proses pengujian ini bertujuan untuk memastikan bahwa program beroperasi sesuai kebutuhan pengguna [10].

Pada prosesnya *black box testing* digunakan untuk mengevaluasi keluaran suatu aplikasi tanpa melihat proses atau fungsionalitas internalnya. Sebagai bagian dari aplikasi ini, pengujian ini dimaksudkan bahwa semua fitur berfungsi sesuai dengan spesifikasi yang telah ditentukan.

b. User Acceptance Testing (UAT)

Pengujian *User Acceptance Testing (UAT)* adalah tahap pengujian yang dilakukan oleh pengguna guna mengevaluasi perangkat lunak yang dikembangkan telah sesuai dengan kebutuhan serta harapan mereka. Dokumentasi yang dihasilkan dari pengujian ini berupa dokumen yang berfungsi sebagai bukti bahwa sistem telah diterima dan siap digunakan [11].

Pada prosesnya UAT dilakukan dengan melibatkan pengguna akhir dalam hal ini pengujian dilakukan oleh pegawai Nurul Fikri Academy guna mendapatkan umpan balik langsung dari pengguna dan Melakukan perbaikan serta penyesuaian untuk memenuhi kebutuhan pengguna.

3. HASIL DAN PEMBAHASAN

3.1 Analisis Kebutuhan Sistem

Analisis kebutuhan sistem dilakukan dengan menerapkan metode *Research and Development (R&D)* untuk memahami secara menyeluruh kebutuhan dan harapan pengguna terhadap aplikasi perencanaan keuangan pegawai berbasis web. Tujuan dari analisis ini ialah untuk memastikan bahwa fitur-fitur yang dikembangkan benar-benar relevan dan bermanfaat sesuai dengan kebutuhan pengguna.

Data yang dikumpulkan melalui metode wawancara dan observasi, kemudian akan dianalisis lebih lanjut untuk merumuskan kebutuhan fungsional dan non-fungsional yang menjadi dasar pengembangan sistem.

a. Wawancara

Metode wawancara dilaksanakan dengan semi-terstruktur terhadap tiga orang responden dari divisi operasional Nurul Fikri Academy. Tujuannya ialah untuk menggali kebutuhan, serta kendala yang dihadapi dalam hal perencanaan keuangan serta masukan terhadap aplikasi.

Dari hasil wawancara yang telah terlaksana, diketahui bahwa beberapa pegawai berinvestasi melalui emas fisik maupun digital, namun sebagian besar masih belum memahami pengelolaan keuangan secara terstruktur. Hal ini menunjukkan bahwa aplikasi perencanaan keuangan berbasis web dibutuhkan untuk edukasi serta pengelolaan finansial bagi pegawai.

b. Observasi

Metode observasi dilakukan bersamaan dengan tahap wawancara, di mana peneliti mengamati secara langsung bagaimana responden menggunakan aplikasi. Pengamatan dilakukan terhadap interaksi pengguna dengan berbagai fitur seperti kalkulator keuangan, histori perhitungan, forum diskusi, serta navigasi antarmuka.

Hasil observasi pada Tabel 1 menunjukkan bahwa mayoritas fitur aplikasi dapat diakses dan digunakan dengan mudah. Namun, terdapat beberapa masukan untuk peningkatan, seperti:

- a) Tampilan *input* angka perlu ditampilkan dalam format *currency* agar lebih mudah dipahami
- b) Area klik pada forum diskusi perlu diperluas

Tabel 1. Hasil Wawancara & Observasi

No	Pertanyaan	Jawaban
1	Bagaimana pengalaman Anda dalam melakukan registrasi dan <i>login</i> ?	Semua responden menyatakan proses <i>login</i> dan registrasi berjalan lancar. Tampilan <i>login/register</i> mudah ditemukan. Salah satu responden menyarankan perbaikan minor pada tata letak agar lebih jelas.
2	Apakah <i>login</i> dengan Google berjalan dengan lancar?	Semua responden menyatakan <i>login</i> dengan Google berjalan dengan baik tanpa kendala.
3	Bagaimana pendapat Anda tentang fitur kalkulator keuangan?	Fitur kalkulator dinilai sangat membantu dan efisien. Salah satu masukan dari responden adalah agar <i>input</i> uang ditampilkan dalam format <i>currency</i> agar tidak membingungkan.
4	Bagaimana kesan Anda terhadap tampilan dan navigasi <i>website</i> ?	Tampilan dinilai <i>clean</i> dan <i>user-friendly</i> . Salah satu responden menyarankan agar jarak antar menu pada <i>header</i> dirapikan agar lebih nyaman dilihat.

No	Pertanyaan	Jawaban
5	Apakah fitur histori perhitungan memudahkan Anda?	Semua responden setuju fitur ini sangat membantu. Salah satu responden menyarankan fitur tambahan seperti pencarian histori berdasarkan kategori atau tanggal.
6	Bagaimana pengalaman Anda dalam menggunakan forum diskusi?	Forum dinilai aktif dan mudah digunakan. Salah satu responden menyarankan agar area klik pada kartu diskusi diperluas.
7	Seberapa mudah Anda menemukan informasi di forum diskusi?	Informasi mudah ditemukan berkat fitur pencarian dan kategori. Masukan: tambahkan sistem <i>filter/sorting</i> berdasarkan tanggal atau popularitas.
8	Apakah Anda mengalami kendala teknis saat menggunakan <i>website</i> ?	Tidak ada kendala teknis yang berarti ditemukan. Semua fitur berjalan normal dan stabil.
9	Bagaimana perasaan Anda saat menggunakan <i>website</i> ini?	Pengguna merasa nyaman, puas, dan terbantu. Desain aplikasi dianggap menarik dan informatif.
10	Apakah Anda akan merekomendasikan <i>website</i> ini kepada orang lain?	Semua responden menyatakan akan merekomendasikan aplikasi ini karena manfaatnya yang besar dalam membantu perencanaan keuangan.

3.2 Perumusan Kebutuhan Sistem

Berdasarkan hasil wawancara dan observasi, berikut ini adalah perumusan kebutuhan fungsional dan non fungsional aplikasi:

A. Kebutuhan Fungsional

- Pengguna dapat melakukan pendaftaran dan *login* menggunakan email dan *password* atau melalui akun Google.
- Pengguna dapat menggunakan kalkulator untuk menghitung Dana Darurat, Investasi, dan Dana Pensiun.
- Pengguna dapat menyimpan, melihat, dan menghapus histori atau riwayat perhitungan keuangan.
- Pengguna dapat membaca artikel keuangan yang tersedia.
- Pengguna dapat membuat dan membalas topik dalam forum diskusi.

B. Kebutuhan Non – Fungsional

- Aplikasi harus responsif dan dapat diakses di berbagai perangkat, baik desktop maupun *mobile*.
- Data pengguna harus disimpan secara aman menggunakan *Firebase Authentication* dan *Firestore*.

- Aplikasi harus memiliki waktu respons yang cepat dan stabil untuk memastikan pengalaman pengguna yang nyaman.

3.3 Perencanaan

Pada bagian perencanaan, akan dijelaskan secara mendetail mengenai *use case* fitur utama, *class diagram* basis data, *activity diagram* meliputi fitur kalkulator investasi, dana darurat, dan dana pensiun. Setiap aspek perencanaan dan perancangan ini akan dijabarkan secara sistematis guna memberikan pemahaman yang komprehensif terkait implementasi teknologi yang digunakan dalam pengembangan aplikasi perencanaan keuangan ini sebagai berikut:

3.3.1 Use Case Fitur Utama

Bagian ini membahas mengenai *use case* fitur utama dari aplikasi perencanaan keuangan berbasis web, yang akan dijabarkan sebagai berikut:



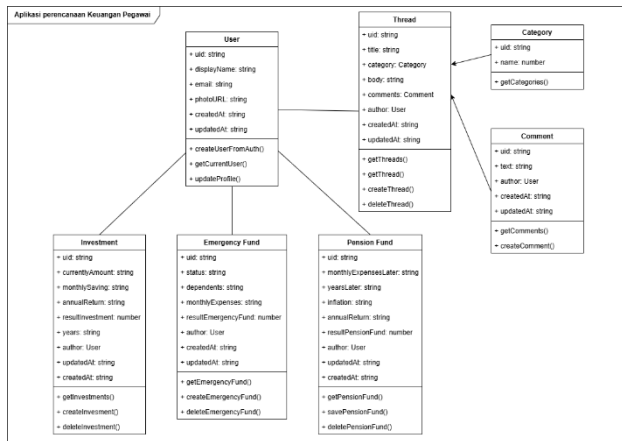
Gambar 2. Use Case Fitur Utama

Pada Gambar 2, menampilkan diagram *use case* yang menggambarkan interaksi pengguna dengan aplikasi perencanaan keuangan pegawai. Dalam aplikasi ini, pengguna dapat melakukan pendaftaran akun melalui halaman *Register*, kemudian masuk ke dalam sistem dengan fitur *Login* atau menggunakan autentikasi melalui akun Google. Setelah berhasil masuk, pengguna memiliki akses ke berbagai fitur utama, salah satunya adalah Kalkulator Keuangan, yang mencakup perhitungan investasi, dana darurat, dan dana pensiun. Selain itu, pengguna juga dapat mengakses Artikel Keuangan yang berisi berbagai informasi terkait pengelolaan keuangan serta berpartisipasi dalam Forum Diskusi sebagai wadah untuk bertukar

informasi dan mendiskusikan berbagai permasalahan keuangan dengan pengguna lainnya

3.3.2 Class Diagram Basis Data

Perancangan basis data *Firebase Firestore* menggunakan *Class Diagram* sebagai alat untuk merepresentasikan tipe data yang digunakan serta metode yang akan diterapkan. Oleh karena itu, hasil dari perancangan tersebut disajikan sebagai berikut:



Gambar 3. Class Diagram Basis Data

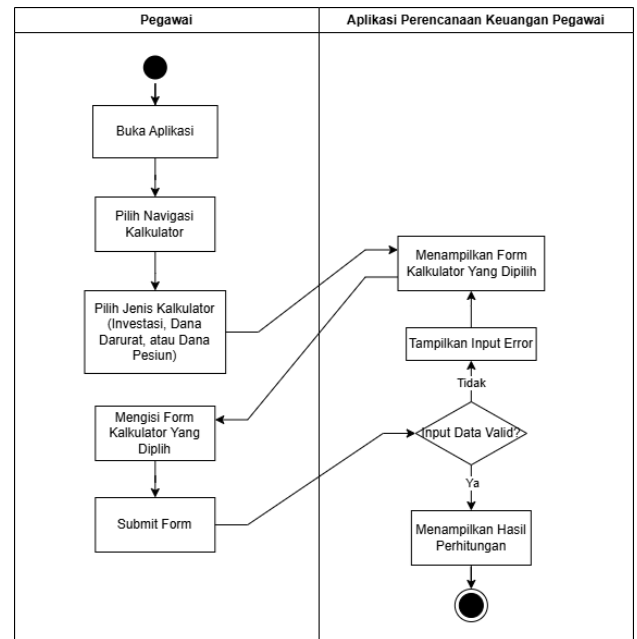
Pada Gambar 3, terdapat *use case* perancangan basis data aplikasi perencanaan keuangan pegawai berpusat pada *User* sebagai aktor utama yang mengelola informasi pribadi dan berinteraksi dengan fitur *Thread*. Fitur ini mencakup *Category* untuk mengelompokkan diskusi dan *Comment* sebagai sarana komunikasi antar pengguna. Hubungan antara entitas ini memungkinkan pertukaran informasi yang lebih efektif dalam forum diskusi terkait perencanaan keuangan.

Selain fitur diskusi, aplikasi juga menyediakan fitur kalkulasi keuangan yang terdiri dari *Investment*, *Emergency Fund*, dan *Pension Fund*. Masing-masing model membantu pengguna dalam merencanakan keuangan mereka dengan menghitung proyeksi investasi, kebutuhan dana darurat, dan dana pensiun berdasarkan berbagai parameter keuangan. Hasil perhitungan ini tersimpan dalam sistem untuk referensi di masa mendatang, sehingga membantu pengguna dalam mengambil keputusan keuangan yang lebih matang dan terarah.

3.3.3 Activity Diagram Kalkulator

Bagian *Activity Diagram* untuk fitur kalkulator terdiri dari dua skenario, yaitu ketika pengguna tidak melakukan *login* dan ketika pengguna telah *login*. Pada kondisi *login*, sistem akan menampilkan tombol Simpan Histori, yang berfungsi untuk menyimpan riwayat perhitungan investasi ke dalam basis data. Dua skenario tersebut akan dijelaskan lebih lanjut sebagai berikut:

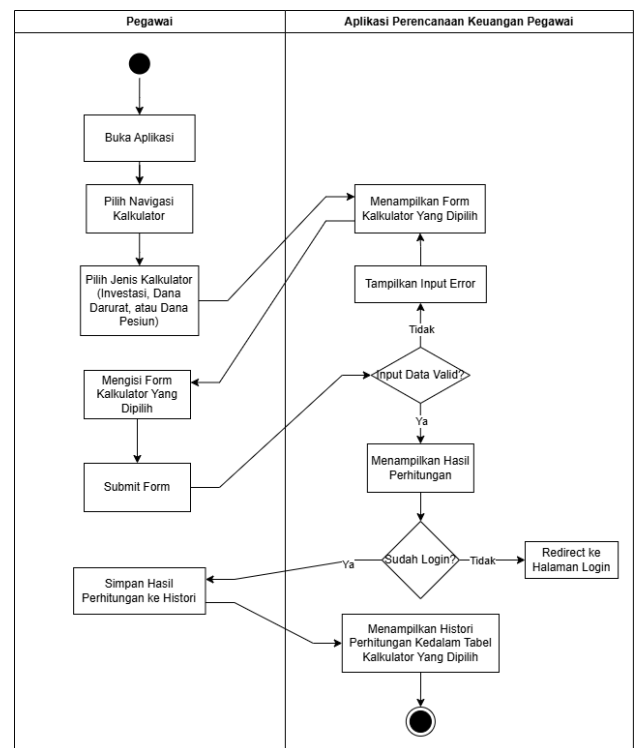
A. Skenario Tanpa Login



Gambar 4. Activity Diagram Kalkulator Skenario Tanpa Login

Pada Gambar 4, menggambarkan skenario di mana pengguna dapat melakukan perhitungan pada kalkulator investasi tanpa perlu *login* terlebih dahulu. Dengan demikian, pengguna diharapkan dapat memahami hasil perhitungan sebelum memilih untuk menyimpan riwayat perhitungan ke dalam *database*.

B. Skenario Login

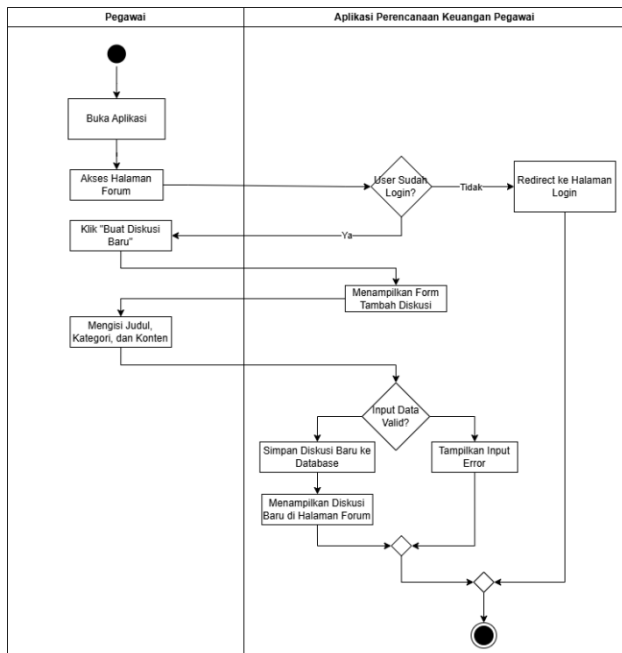


Gambar 5. Activity Diagram Kalkulator Dengan Skenario Login

Pada Gambar 5, sekian menjelaskan alur proses penggunaan kalkulator dalam menyimpan riwayat perhitungan. Dalam alur tersebut, pengguna diwajibkan untuk *login* terlebih dahulu sebelum dapat menggunakan fitur penyimpanan riwayat perhitungan pada kalkulator yang dipilih (kalkulator investasi, kalkulator dana darurat, dan kalkulator dana pensiun).

3.3.4 Activity Diagram Forum Diskusi

Activity diagram forum diskusi membuat diskusi baru, untuk detailnya sebagai berikut:



Gambar 6. Activity Diagram Menambahkan Diskusi Baru

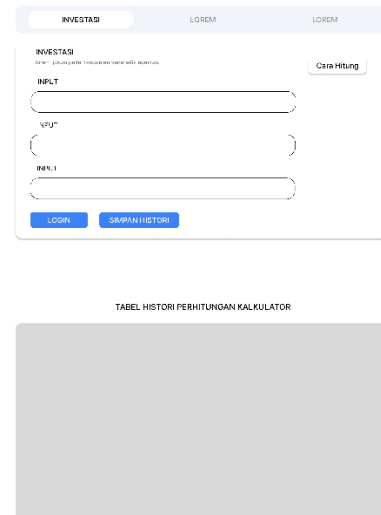
Pada Gambar 6, menggambarkan alur proses pembuatan diskusi baru dalam forum aplikasi perencanaan keuangan pegawai. Proses dimulai ketika pengguna membuka aplikasi dan mengakses halaman forum. Selanjutnya, pengguna memilih opsi "Buat Diskusi Baru," yang kemudian divalidasi oleh sistem untuk memastikan status *login* pengguna. Jika pengguna belum masuk, sistem akan mengarahkan mereka ke halaman *login* terlebih dahulu. Setelah berhasil *login*, sistem menampilkan formulir untuk menambahkan diskusi, di mana pengguna mengisi judul, kategori, dan konten diskusi. Data yang dimasukkan kemudian diperiksa valid atau tidaknya oleh sistem. Jika data valid, sistem akan menyimpan diskusi ke dalam *database* dan menampilkannya di halaman forum. Sebaliknya, jika data tidak valid, sistem akan menampilkan pesan kesalahan. Diagram ini menunjukkan bagaimana sistem menangani proses pembuatan diskusi dengan mekanisme validasi *login* dan *input* untuk memastikan kelancaran serta integritas data dalam aplikasi.

3.4 Design

Bagian desain akan menjabarkan secara detail rancangan yang digunakan, mencakup *wireframe* untuk halaman,

beranda/utama, kalkulator keuangan (investasi, dana darurat, dan dana pensiun), serta forum diskusi. *Wireframe* atau desain kasar aplikasi perencanaan keuangan halaman kalkulator dan halaman forum diskusi. Berikut ini hasil pembuatan *wireframe* yang sudah dibuat:

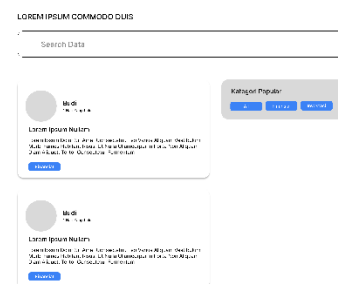
A. Halaman Kalkulator



Gambar 7. Wireframe Halaman Kalkulator

Pada Gambar 7, ditampilkan *wireframe* halaman kalkulator yang mencakup tiga fitur utama, yaitu kalkulator investasi, kalkulator dana darurat, dan kalkulator dana pensiun. Halaman ini juga dilengkapi dengan *form input*, panduan perhitungan, tombol hitung, tombol simpan histori perhitungan, serta tabel histori perhitungan untuk menampilkan riwayat hasil kalkulasi yang telah disimpan oleh pengguna.

B. Halaman Forum Diskusi



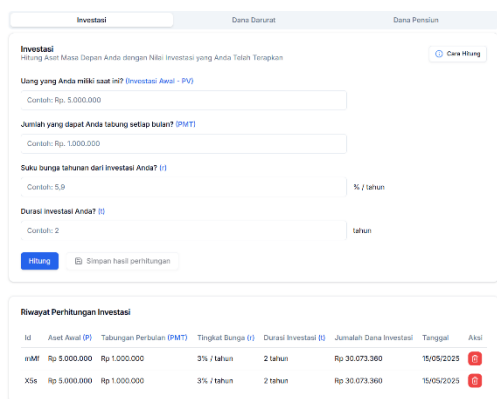
Gambar 8. Wireframe Halaman Forum Diskusi

Pada Gambar 8, ditampilkan *wireframe* halaman forum diskusi yang dilengkapi dengan fitur pencarian (*search*) yang berfungsi untuk mencari forum diskusi berdasarkan judul. Selain itu, terdapat juga fitur filter berdasarkan kategori yang memungkinkan pengguna menyaring forum diskusi sesuai dengan topik yang diinginkan.

3.5 Pengkodean

Bagian pengkodean mencakup hasil *slicing user interface* kalkulator (investasi, dana darurat, dan dana pensiun) dan halaman forum diskusi. Rincian lebih lanjut akan dijelaskan pada bagian berikut:

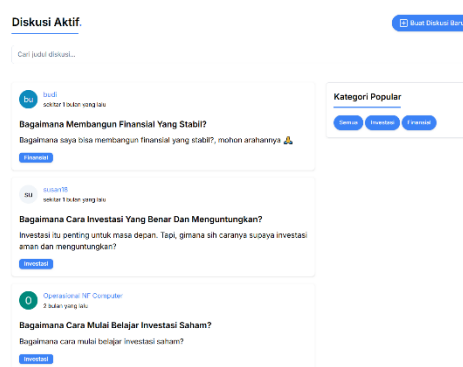
A. Slicing Halaman Kalkulator



Gambar 9. Slicing Halaman Kalkulator

Pada Gambar 9, menampilkan hasil pengerjaan *slicing UI* pada halaman kalkulator yang mencakup investasi, dana darurat, dana pensiun, serta tabel histori perhitungan pada halaman kalkulator aplikasi perencanaan keuangan berbasis web.

B. Slicing Halaman Forum Diskusi



Gambar 10. Slicing Halaman Forum Diskusi

Pada Gambar 10, menampilkan hasil pengerjaan *slicing UI* pada halaman forum diskusi aplikasi perencanaan keuangan pegawai berbasis web.

3.6 Pengujian

Proses pengujian aplikasi menggunakan dua metode pengujian yakni, metode *blackbox testing* yang digunakan oleh penguji untuk menguji aplikasinya, kemudian *user acceptance testing* digunakan sebagai tolak ukur pengujian yang dilakukan oleh *user*. Untuk detail nya sebagai berikut ini:

A. Hasil Pengujian Black Box Testing

Berikut merupakan hasil *black box testing* yang telah dilakukan guna memastikan aplikasi berjalan dengan semestinya. Rincian lengkapnya disajikan di bawah ini:

Tabel 2. Hasil Pengujian Black Box Testing

No	Fitur Yang Diuji	Skenario Pengujian	Input	Ekspektasi Output	Hasil
1	Registrasi & Login	Pengguna dapat mendaftar dan login menggunakan email dan password	Input email & password valid	Akun berhasil dibuat, pengguna dapat masuk	Sesuai
2	Login dengan Akun Google	Pengguna dapat login menggunakan akun Google	Klik tombol "Berlogo Google"	Pengguna diarahkan ke akun Google dan berhasil masuk	Sesuai
3	Kalkulator Keuangan (Investasi, Dana Darurat, & Dana Pensiun)	Pengguna dapat melakukan perhitungan keuangan	Input data keuangan yang valid	Hasil kalkulasi muncul sesuai dengan rumus yang diterapkan	Sesuai
4	Histori Perhitungan	Pengguna dapat melihat histori perhitungan sebelumnya	Klik tombol "Simpan Histori Perhitungan"	Data histori perhitungan muncul dengan benar	Sesuai
5	Forum Diskusi	Pengguna dapat membuat, membalas, melihat diskusi, dan menghapus diskusi	Input teks dan submit	Postingan atau komentar muncul di forum	Sesuai
6	Pencarian Informasi (Cari Diskusi)	Pengguna dapat mencari diskusi pada halaman forum diskusi	Input kata kunci di kolom pencarian	Hasil pencarian relevan dengan kata kunci	Sesuai
7	Filter Diskusi Dengan Kategori	Pengguna dapat melakukan filter diskusi pada halaman forum diskusi	Klik tombol kategori diskusi	Hasil filter relevan dengan kategori diskusi	Sesuai
8	Navigasi & UI/UX	Menu navigasi bekerja dengan baik	Klik menu atau tombol navigasi	Pengguna diarahkan ke halaman yang sesuai	Sesuai

No	Fitur Yang Diuji	Skenario Pengujian	Input	Ekspektasi Output	Hasil
9	Kecepatan & Performa	Website berjalan dengan lancar tanpa lag	Akses website dengan koneksi normal	Halaman dimuat dalam waktu kurang dari 3 detik	Sesuai
10	Logout	Pengguna dapat keluar dari akun dengan aman	Klik tombol <i>logout</i>	Pengguna kembali ke halaman <i>login</i> , tidak bisa mengakses halaman sebelumnya	Sesuai

Mengacu Tabel 2, seluruh fitur yang diuji menunjukkan hasil sesuai dengan ekspektasi *output*, sehingga dapat disimpulkan bahwa semua skenario pengujian berhasil dijalankan dengan baik.

Setelah skenario uji *black box testing* didapatkan maka, nilai tingkat keberhasilannya dapat dihitung sebagai berikut:

$$\text{Tingkat Keberhasilan (\%)} = \left(\frac{\text{Jumlah Pengujian Berhasil}}{\text{Total Pengujian}} \right) \times 100$$

Dengan rincian sebagai berikut:

a. Jumlah pengujian berhasil = 10

b. Total Pengujian = 10

Maka didapatkan nilai:

$$\text{Tingkat Keberhasilan} = \left(\frac{10}{10} \right) \times 100 = 100\%$$

Kesimpulan yang didapatkan ialah tingkat keberhasilan pada pengujian *black box testing* pada aplikasi perencanaan keuangan pegawai mencapai 100%, yang menunjukkan tingkat fungsionalitas utama berjalan sesuai dengan spesifikasi dan harapan pengguna.

B. Hasil Pengujian *User Acceptance Testing*

Bagian UAT menggunakan pendekatan kualitatif yang digunakan untuk mengukur apakah aplikasi sudah berjalan sesuai dengan harapan pengguna. Berikut merupakan hasil pengujian yang sudah terlaksana dengan skema pengujian dengan detail sebagai berikut:

a. Skoring Jawaban

Tiap jawaban dari responden akan dinilai:

- Positif / Sesuai harapan dengan skor 1
- Negatif / Tidak sesuai harapan dengan skor 0.5 (untuk jawaban netral/kritik ringan)
- Sangat tidak sesuai dengan skor 0

b. Penilaian dari sisi pengguna

Tabel 3. Hasil Pengujian *User Acceptance Testing*

No	Pertanyaan	R1	R2	R3	Skor Total
1	Pengalaman registrasi dan login?	1	1	1	3
2	Login menggunakan Google dapat berjalan lancar?	1	1	1	3
3	Pendapat tentang fitur kalkulator keuangan	0.5	1	1	2.5
4	Kesan terhadap tampilan dan navigasi	0.5	1	1	2.5
5	Fitur histori perhitungan memudahkan pengguna?	1	1	1	3
6	Pengalaman menggunakan forum diskusi	0.5	1	1	2.5
7	Kemudahan dalam menemukan informasi pada forum diskusi	1	1	0.5	2.5
8	Kendala teknis saat menggunakan website	1	1	1	3
9	Perasaan saat menggunakan website	1	1	1	3
10	Rekomendasi kepada orang lain	1	1	1	3
Total Skor					27
Skor Maksimal (10 Pertanyaan x 3 Responden)					30

Merujuk pada Tabel 3, yang melibatkan 3 orang responden dari divisi operasional Nurul Fikri Academy, dengan total skor ialah 27. Dari data tersebut kemudian dihitung untuk mendapatkan persentase keberhasilan aplikasi sebagai berikut:

$$\text{Persentase keberhasilan} = \frac{\text{Total Skor}}{\text{Skor Maksimal}} \times 100\%$$

Maka didapatkan nilai presentasi keberhasilan:

$$\text{Presentase keberhasilan} = \frac{27}{30} \times 100\% = 90\%$$

Persentase nilai keberhasilan yang didapatkan yakni 90% telah melampaui batas keberhasilan minimal sebesar 80%, yang artinya bahwa aplikasi perencanaan keuangan telah berhasil memenuhi harapan dan kebutuhan pengguna.

4. KESIMPULAN

Berdasarkan hasil dari proses pengembangan, implementasi, dan pengujian aplikasi perencanaan keuangan berbasis web menggunakan React JS dan Firebase studi kasus Nurul Fikri Academy, maka dapat disimpulkan bahwa:

- a. Pengembangan aplikasi Perencanaan Keuangan Pegawai dilakukan menggunakan metode *Extreme Programming* (XP) melalui tahapan iteratif, yaitu perencanaan, desain, pengkodean, dan pengujian. Proses dimulai dari penyusunan *user story* dan diagram kebutuhan sistem, dilanjutkan dengan pembuatan *wireframe* antarmuka, implementasi fitur menggunakan React JS, TanStack React Query, dan Firebase Firestore, serta pengujian melalui *Blackbox Testing* dan *User Acceptance Testing* (UAT).
- b. Hasil implementasi menunjukkan aplikasi berhasil berfungsi dengan baik. Berdasarkan UAT terhadap tiga pegawai operasional Nurul Fikri Academy, tingkat keberhasilan mencapai 90%, melebihi ambang batas minimal 80% yang ditetapkan. Hal ini membuktikan bahwa aplikasi telah diterima pengguna dan efektif dalam membantu perencanaan keuangan secara digital.

Untuk pengembangan aplikasi ke depan agar lebih optimal dan adaptif terhadap kebutuhan pengguna, terdapat beberapa saran yang dapat dipertimbangkan. Pertama, penambahan fitur ekspor data dalam format PDF atau Excel dari histori perhitungan, guna memudahkan pengguna dalam mendokumentasikan atau mencetak hasil perencanaan keuangan. Kedua, perlu dilakukan pengujian lebih lanjut terhadap aspek keamanan, khususnya autentikasi dan kontrol akses pada Firebase, untuk meningkatkan kepercayaan pengguna terhadap aplikasi.

Selain itu, disarankan agar sistem mendukung pengelolaan berbasis peran, seperti admin atau superadmin, yang bertugas mengelola pengguna serta konten artikel dan forum. Dukungan berupa *dashboard* khusus admin/superadmin juga perlu dikembangkan agar pengelolaan sistem menjadi lebih efisien, terstruktur, dan terkontrol secara menyeluruh.

DAFTAR PUSTAKA

- [1] Otoritas Jasa Keuangan, "Visi misi," Accessed: Feb. 15, 2025. [Online]. Available: <https://ojk.go.id/id/tentang-ojk/Pages/Visi-Misi.aspx>
- [2] Otoritas Jasa Keuangan, "Infografis Hasil Survei Nasional Literasi dan Inklusi Keuangan Tahun 2022," Accessed: Sep. 28, 2024. [Online]. Available: <https://ojk.go.id/id/berita-dan-kegiatan/info-terkini/Pages/Infografis-Survei-Nasional-Literasi-dan-Inklusi-Kuangan-Tahun-2022.aspx>
- [3] Nurul Fikri Academy, "About Us," Accessed: Sep. 28, 2024. [Online]. Available: <https://nfacademy.id/about-us/>
- [4] T. Sulistyorini, E. Sova, and R. Ramadhan, "Pemantauan Kasus Penyebaran Covid-19 Berbasis Website Menggunakan Framework React Js Dan Api," *J. Ilm. Multidisiplin*, vol. 1, no. 04, Jul. 2022, doi: 10.56127/jukim.v1i04.137.
- [5] R. A. Putra and R. Maulana, "Perancangan Aplikasi Layanan Pengaduan Customer Berbasis Web Menggunakan React Js: Studi Kasus Pada PT Inovatif 78," *Repos. STT Terpadu Nurul Fikri*, 2024, [Online]. Available: <https://repository.nurulfikri.ac.id/id/eprint/511/>
- [6] S. Saharuddin, A. E. Burhandenny, A. Harjanto, H. Nograho, and P. Wirawan, "Rancang Bangun Sistem Internet of Things (IoT) Alat Pengendalian Kekeruhan Air Pada Kolam Terpal Ikan Lele Memanfaatkan Firebase: Indonesia," *AJIE Asian J. Innov. Entrep.*, pp. 33–38, Jan. 2023, doi: 10.20885/ajie.vol7.iss1.art5.
- [7] Q. N. Farhana, I. Arwani, and D. W. Brata, "Pengembangan Sistem Informasi Pembayaran Biaya Pendidikan Menggunakan Teknologi Payment Gateway (Studi Kasus: SD Islam Terpadu Nurul Fikri Pati)," *J. Pengemb. Teknol. Inf. Dan Ilmu Komput.*, vol. 6, no. 6, Jul. 2022.
- [8] A. Rustamana, K. H. Sahl, D. Ardianti, and A. H. S. Solihin, "Penelitian dan Pengembangan (Research & Development) dalam Pendidikan," *J. Bima Pus. Publ. Ilmu Pendidik. Bhs. Dan Sastra*, vol. 2, no. 3, pp. 60–69, Jun. 2024, doi: 10.61132/bima.v2i3.1014.
- [9] T. Ardiansah, "Perancangan Sistem Persediaan Menggunakan Metode Extreme Programming," *J. Ilm. Inform. Dan Ilmu Komput. JIMA-Ilk.*, vol. 1, no. 1, Mar. 2022, doi: 10.58602/jima-ilkom.v1i1.1.
- [10] N. M. D. Febriyanti, A. A. K. Oka Sudana, and I. N. Piarsa, "Implementasi Black Box Testing pada

Sistem Informasi Manajemen Dosen,” *JITTER J. Ilm. Teknol. Dan Komput.*, vol. 2, no. 3, p. 535, Nov. 2021, doi: 10.24843/JTRTI.2021.v02.i03.p12.

[11] A. Mujahid, M. Y. Abdullah, S. Suharya, and A. R. Adriansyah, “Analisis Dan Pengembangan Sistem

Informasi Pengelolaan Masjid Berbasis Mobile Dengan Teknologi Api Web Service,” *J. Inform. Terpadu*, vol. 7, no. 2, Sep. 2021, doi: 10.54914/jit.v7i2.368.