

Rancang Bangun Web Service E-Budgeting Gereja Toraja Jemaat Bintaro Menggunakan Metode RESTful API Berbasis Web dan Android

Putri Hayati^{1*}, Kukuh Harsanto², Muhammad Iqbal Aulawi³, Febri Wanda⁴, Subandi⁵

^{1,3,4,5}Fakultas Teknologi Informasi, Teknik Informatika, Universitas Budi Luhur, Jakarta, Indonesia

²Fakultas Teknologi Informasi, Sistem Informasi, Universitas Budi Luhur, Jakarta, Indonesia

E-mail: ^{1*}putri.hayati@budiluhur.ac.id, ²kukuh.harsanto@budiluhur.ac.id, ³aulawi123@gmail.com,

⁴febri.wanda@budiluhur.ac.id, ⁵subandi.spd@budiluhur.ac.id

(* : corresponding author)

Abstrak

Teknologi informasi telah memberikan dampak signifikan pada berbagai bidang, termasuk dalam pengelolaan keuangan lembaga keagamaan. Gereja Toraja menghadapi kendala dalam pengelolaan anggaran yang bersifat manual, sehingga rawan terjadi kesalahan, keterlambatan, serta minimnya transparansi. Tujuan penelitian ini merancang serta membangun *web service E-Budgeting* berbasis RESTful API yang diintegrasikan dengan aplikasi *web* dan Android guna meningkatkan efisiensi, akurasi, serta transparansi pengelolaan anggaran. Penelitian ini menggunakan model pengembangan *waterfall* yang meliputi tahap analisis kebutuhan, perancangan, implementasi, serta pengujian. RESTful API dipilih sebagai metode komunikasi data karena fleksibilitasnya dalam mendukung berbagai *platform*. Perancangan sistem mencakup UML (*Unified Modeling Language*) untuk pemodelan, serta *framework* Laravel dan Flutter untuk pengembangan aplikasi *web* dan Android. Hasil penelitian menunjukkan, sistem mampu memenuhi kebutuhan pengelolaan anggaran secara efektif, dengan fitur seperti pencatatan pemasukan dan pengeluaran, perencanaan anggaran, dan laporan keuangan yang dapat diakses melalui *web* dan Android. Uji coba sistem menunjukkan tingkat keberhasilan sebesar 79% dalam kemudahan penggunaan. Kontribusi penelitian ini adalah menyediakan solusi berbasis teknologi yang mendukung pengelolaan anggaran gereja secara modern dan transparan. *Web service E-Budgeting* yang dihasilkan dapat diadopsi oleh lembaga keagamaan lainnya yang memiliki kebutuhan serupa, serta menjadi dasar pengembangan fitur yang lebih kompleks di masa depan di masa depan.

Kata kunci: web service, e-budgeting, RESTful API, android, gereja Toraja

Abstract

Information technology has had a significant impact on various fields, including the financial management of religious institutions. The Toraja Church faces obstacles in manual budget management, which is prone to errors, delays, and lack of transparency. The purpose of this research is to design and build a RESTful API-based E-Budgeting web service that is integrated with web and Android applications to improve the efficiency, accuracy, and transparency of budget management. This research uses a waterfall development model which includes the stages of needs analysis, design, implementation, and testing. RESTful API was chosen as the data communication method because of its flexibility in supporting various platforms. The system design includes UML (Unified Modeling Language) for modeling, as well as Laravel and Flutter frameworks for web and Android application development. The results show that the system is able to meet the needs of budget management effectively, with features such as recording income and expenses, budget planning, and financial reports that can be accessed via the web and Android. System trials showed a success rate of 79% in ease of use. The contribution of this research is to provide a technology-based solution that supports modern and transparent church budget management. The resulting E-Budgeting web service can be adopted by other religious institutions that have similar needs, as well as being the basis for future development of more complex features in the future.

Keywords: web service, e-budgeting, RESTful API, android, Toraja church

1. PENDAHULUAN

Gereja adalah tempat beribadah bagi umat kristiani [1]. Dalam melakukan kegiatan ibadah dan pelayanan tiap minggunya ada sejumlah pemasukan kas dan pengeluaran kas. Pemasukan kas berasal dari persembahan, perpuluhan dan sumbangan lainnya dari jemaat. Kas yang diterima

nantinya akan digunakan untuk pembiayaan kegiatan gereja diantaranya mengakomodasikan perlengkapan serta kebutuhan lainnya untuk kegiatan ibadah dan pelayanan gereja serta kegiatan kemanusiaan lainnya.

Saat ini Gereja Toraja Jemaat Bintaro menghadapi kendala dalam pengelolaan anggaran yang masih bersifat manual, sehingga rawan terjadi kesalahan, keterlambatan, serta minimnya transparan. Dengan bertambahnya minat jemaat dalam memberi bantuan sumbangan serta bertambahnya aktivitas kegiatan gereja, data keuangan yang dikelola pihak majelis gereja semakin banyak dan sulit untuk dikelola. Pencatatan keluar dan masuknya kas dengan cara manual mulai kurang efektif dalam mengelola data yang terus bertambah sehingga menimbulkan banyaknya data yang kurang lengkap, tidak akurat, dan tidak tercatat.

Penelitian ini bertujuan untuk merancang serta membangun *web service E-Budgeting* berbasis RESTful API yang dapat dihubungkan dengan aplikasi web serta Android, sehingga dapat meningkatkan efisiensi, ketepatan, dan transparansi dalam pengelolaan anggaran gereja. Proses penelitian dilakukan dengan model pengembangan sistem *waterfall*, yang mencakup tahapan analisis kebutuhan, desain sistem, implementasi, pengujian, serta pemeliharaan. RESTful API dipilih sebagai metode komunikasi data karena memiliki fleksibilitas tinggi dalam mendukung berbagai platform serta mampu beroperasi pada sistem yang terdistribusi.

Penelitian yang dilakukan [2], bertujuan menyediakan layanan pemantauan penyebaran wabah COVID-19 secara *real-time* untuk masyarakat. Hasil penelitian menunjukkan bahwa *web service* yang dikembangkan dapat dimanfaatkan oleh masyarakat untuk memantau penyebaran COVID-19 secara langsung. Sementara itu, penelitian [3], bertujuan menyusun struktur dokumen CV dengan menganalisis kebutuhan data serta informasi pengguna pada perangkat lunak. Hasilnya, dosen memperoleh CV yang otomatis diperbarui setiap tahun berdasarkan data yang dimasukkan ke sistem lain milik universitas. Adapun penelitian yang dilakukan oleh [4], bertujuan mengembangkan modul *inventory e-canteen* yang mudah di-*scale-up* serta terintegrasi dengan aplikasi lain. Hasilnya berupa modul *inventory* berbasis *web service* yang memungkinkan pengguna melakukan operasi CRUD pada produk kantin mereka menggunakan metode RESTful.

Berdasarkan penjelasan di atas, tujuan membuat aplikasi ini yaitu dapat memudahkan gereja dalam melakukan pencatatan kas dan menyediakan informasi keuangan kepada jemaat. Dengan adanya aplikasi ini, pihak majelis gereja dapat melakukan pengelolaan data keuangan menjadi lebih efektif dan lebih efisien. Selain itu, aplikasi ini digunakan untuk transparansi data pemasukan dan pengeluaran keuangan terkait sumbangan yang diterima pihak majelis gereja dari jemaat agar jemaat dapat mengetahui informasi yang mereka butuhkan, aplikasi terintegrasi multi-platform, yaitu *web* dan *android* dan pengelolaan anggaran yang terstruktur dan *real-time*.

2. METODE PENELITIAN

Pendekatan kualitatif digunakan pada penelitian untuk mengembangkan aplikasi *e-budgeting*. Analisa kebutuhan, perancangan sistem, implementasi, pengujian, serta pemeliharaan adalah semua bagian dari model *waterfall* dalam pengembangan perangkat lunak[5]. Tahap analisis dilakukan dengan mengumpulkan data melalui metode wawancara, observasi lapangan, serta kuesioner untuk memahami proses bisnis yang berjalan. Dari hasil analisis ini, ditemukan dua jenis kebutuhan, yaitu kebutuhan fungsional dan kebutuhan non-fungsional. Kebutuhan fungsional meliputi manajemen transaksi keuangan seperti pencatatan pemasukan, pengeluaran anggaran dan laporan keuangan.

Sementara itu, kebutuhan non-fungsional mencakup *usability* dan *portability*. *Usability* memungkinkan aplikasi mudah dipahami dan digunakan oleh pengurus gereja maupun jemaat, sedangkan *portability* memungkinkan sistem dapat diakses melalui web dan android. Tabel analisa kebutuhan dapat dilihat pada Tabel 1 [6].

Tabel 1. Analisis Kebutuhan

No	Kategori	Kebutuhan	Deskripsi
1	Kebutuhan fungsional	Pemasukan anggaran	Sistem memungkinkan melakukan pencatatan pemasukan
		Pengeluaran anggaran	Sistem memungkinkan melakukan pencatatan pengeluaran
		Laporan keuangan	Sistem menyediakan laporan keuangan yang mencakup ringkasan pemasukan, pengeluaran dan saldo
2	Kebutuhan non-fungsional	Usability	Antarmuka pengguna dirancang agar mudah dipahami dan digunakan oleh pengurus gereja maupun jemaat
		Portabilitas	Sistem dapat diakses melalui web dan android

Tahapan penelitian untuk untuk menyelesaikan masalah yang terjadi dapat dilihat pada Gambar 1.



Gambar 1. Tahapan Penelitian

2.1. Metode Pengumpulan Data

Dalam penelitian ini, menerapkan beberapa Teknik pengumpulan data yang dibutuhkan, meliputi analisa kebutuhan, observasi dan studi Pustaka [7].

a. Wawancara.

Wawancara dengan majelis Gereja Toraja Jemaat Bintaro dilakukan sebagai bagian proses kebutuhan aplikasi *e-budgeting* yang bertujuan untuk memperoleh pemahaman mendalam mengenai kebutuhan fungsional dan non fungsional. Data dari wawancara ini memberikan gambaran tentang permasalahan yang dihadapi serta manfaat dan kekurangan aplikasi dalam implementasi nyata

b. Observasi.

Observasi dilakukan untuk mendapatkan Gambaran langsung mengenai proses pengelolaan keuangan yang berjalan di Gereja Toraja Jemaat Bintaro, sehingga dapat memberikan data yang lebih akurat tentang aktivitas dan kebutuhan majelis gerja terkait sistem *e-budgeting*.

c. Kuesioner

Pada penelitian ini peneliti memakai kuesioner yang berisi pertanyaan nantinya akan diberikan kepada responden dan dikumpulkan untuk identifikasi kebutuhan sistem. Responden yang terlibat terdiri dari majelis gereja dan jemaat sebanyak 20 orang.

d. Studi Pustaka

Tahap ini dilakukan dengan penelusuran dan pemahaman artikel, buku, termasuk sumber-sumber lain yang berkaitan dengan subjek penelitian.

2.2. Perancangan

Perancangan digunakan untuk melihat gambaran secara keseluruhan mengenai bagaimana sistem akan dibangun, mulai dari komponen-komponen sistem hingga hubungan antar komponen tersebut. Aplikasi *e-budgeting* Gereja Toraja Jemaat Bintaro dirancang menggunakan pendekatan arsitektur layanan *three-tier architecture* dengan RESTful API, serta diakses melalui aplikasi web dan android. Pada bagian *presentation layer* aplikasi web dibangun menggunakan *framework* Laravel dan digunakan oleh pengurus gereja untuk mengelola anggaran[3], kemudian pembuatan aplikasi *mobile* menggunakan *framework flutter*[8], memungkinkan jemaat mengakses informasi anggaran dan laporan keuangan secara *real-time*. Pada bagian *application layer* untuk memproses logika bisnis, yang diimplementasikan menggunakan *web service* berbasis RESTful API [9]. RESTful API ini akan menangani permintaan dari aplikasi web dan android, memproses data dan mengirimkan respons dalam format JSON [10] dan Laravel digunakan untuk membangun *web service* karena kemampuannya dalam pengembangan aplikasi berbasis API yang efisien. Pada bagian *data layer* bertugas menyimpan semua data terkait sistem, seperti data anggaran, transaksi pemasukan dan pengeluaran serta laporan keuangan, *database* yang digunakan MySQL. Alur komunikasi dalam sistem ini adalah *client* yang berupa aplikasi android melakukan

permintaan ke *web services* dengan menggunakan *HTTP Connection* ke RESTful API menggunakan *endpoint* GET, POST, PUT, DELETE. Proses permintaan dari *client* yang berupa data yang diambil dari *database* aplikasi *e-budgeting*, kemudian *web service* mengirimkan *response* yang diterima dalam format JSON. Proses terakhir aplikasi web dan android menampilkan data kepada pengguna berdasarkan *response* yang diterima.

2.3. Implementasi

Pada tahap implementasi dilakukan dengan merealisasikan rancangan sistem menjadi aplikasi yang dapat digunakan. Pengembangan *backend* menggunakan *framework* Laravel, implementasi *web service* dan *endpoint* RESTful API untuk memenuhi kebutuhan fungsional sistem. Pengembangan *frontend* aplikasi web berbasis Laravel yang digunakan admin dan aplikasi android berbasis *flutter*. Tujuan dari tahap ini memastikan sistem yang diimplementasi sesuai dengan desain yang dibuat dan sistem berjalan dengan baik[11].

2.4. Pengujian

Pada tahap ini dilakukan pengujian secara keseluruhan menggunakan *user acceptance test* sesuai kebutuhan fungsional, dimana hasilnya diharapkan dapat diterima oleh *user*. Pengujian menggunakan kuisisioner, kemudian disebarakan ke responden, yaitu majelis gereja dan jemaat sebanyak 20 orang untuk perhitungan menggunakan skala likert. Pada Tabel 2 menunjukkan indeks skala likert.

Tabel 2. Indeks Skala likert

No	Skala Likert	Bobot Nilai
1	Sangat: Mudah/Bagus/Sesuai/Jelas/Setuju/Baik	5
2	Mudah/Bagus/Sesuai/Jelas/Setuju/Baik	4
3	Cukup: Mudah/Bagus/Sesuai/Jelas/Setuju/Baik	3
4	Kurang: Bagus/Sesuai/Jelas/Setuju/Baik	2
5	Sangat: Sulit/Jelek/Tidak Sesuai/Tidak Jelas/Tidak Setuju/Tidak Baik	1

Tanggapan responden terhadap penerimaan aplikasi *e-budgeting* diambil berdasarkan jawaban dari responden. Jawaban tersebut kemudian diukur menggunakan rumus 1[12]:

$$\% \text{ skor aktual} = \frac{\text{skor aktual}}{\text{skor ideal}} \times 100\% \quad (1)$$

Keterangan umum:

Skor aktual : jawaban seluruh responden yang telah diajukan

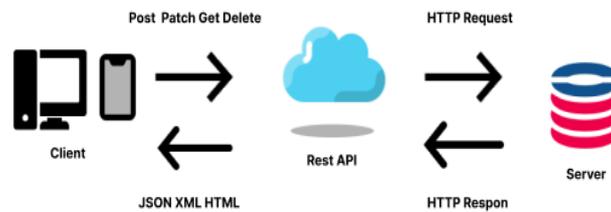
Skor ideal : nilai maksimal atau tertinggi untuk semua responden

3. HASIL DAN PEMBAHASAN

Sesuai dengan analisa kebutuhan yang telah dilakukan, maka dirancang sebuah aplikasi *e-budgeting*.

3.1. Arsitektur Web Service

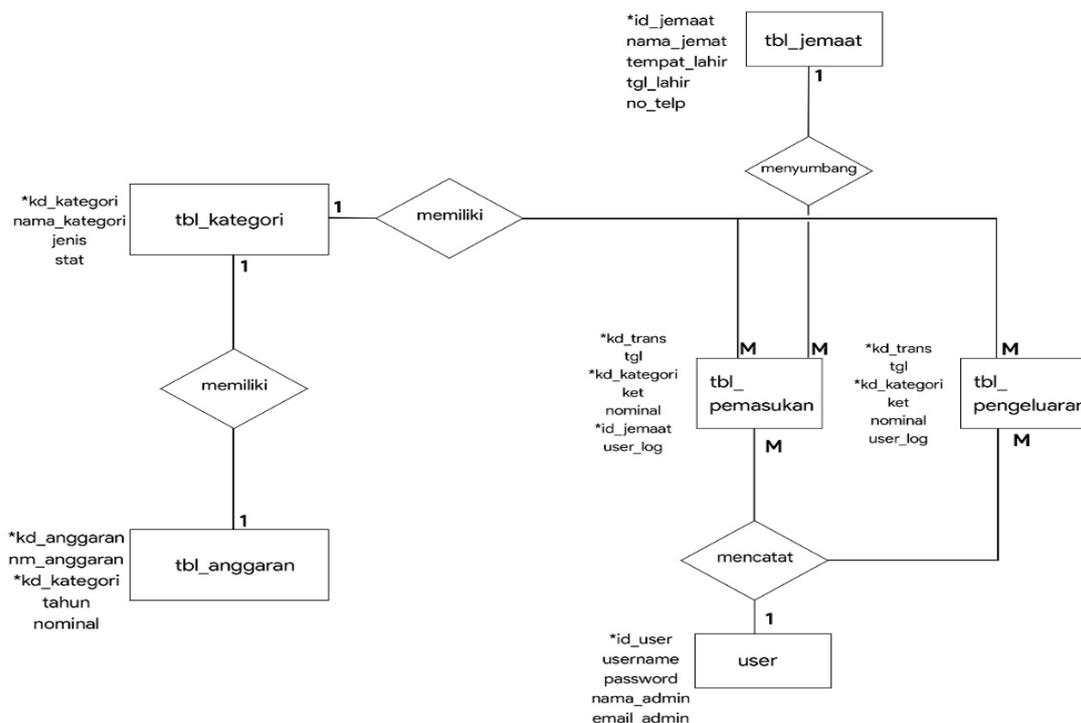
Berdasarkan perancangan maka didapatkan alur komunikasi dalam sistem. *Client* yang berupa aplikasi android melakukan permintaan ke *web services* dengan menggunakan *HTTP Connection* ke RESTful API menggunakan *endpoint* GET, POST, PUT, DELETE. Proses permintaan dari *client* yang berupa data yang diambil dari *database* aplikasi *e-budgeting*, kemudian *web service* mengirimkan *response* yang diterima dalam format JSON. Proses terakhir aplikasi web dan android menampilkan data kepada pengguna berdasarkan *response* yang diterima, dapat dilihat pada Gambar 2.



Gambar 2. Arsitektur Web Service

3.2. Entity Relationship Diagram (ERD)

ERD (*Entity Relationship Diagram*) digunakan untuk gambaran hubungan antara penyimpanan informasi di dalam *Data Flow Diagram*. ERD menggunakan sejumlah gambar untuk menggambarkan secara terstruktur dan hubungan antar informasi. ERD menggunakan tiga jenis gambar, khususnya Substansi, Kualitas, dan Hubungan[13]. ERD dari sistem aplikasi *E-Budgeting* Gereja Toraja Jemaat Bintaro dapat dilihat pada Gambar 3. Entitas yang digunakan untuk pembuatan basis data pada aplikasi *e-budgeting*, yaitu *tbl_anggaran* terdiri dari beberapa atribut seperti *kd_anggaran*, *nm_anggaran*, *kd_kategori*, *tahun* dan *nominal*, *tbl_kategori* terdiri dari beberapa atribut seperti *kd_kategori*, *nama_kategori*, *jenis* dan *stat*, *tbl_jemaat* terdiri dari beberapa atribut seperti *id_jemaat*, *nama_jemaat*, *tempat_lahir*, *tgl_lahir* dan *no_telp*, *tbl_pemasukan* terdiri dari beberapa atribut seperti *kd_trans*, *tgl*, *kd_kategori*, *ket*, *nominal*, *id_jemaat* dan *user_log*, *tbl_pengeluaran* terdiri dari beberapa atribut seperti *kd_trnas*, *tl*, *kd_kategori*, *ket*, *nominal* dan *user_log*, kemudian *user* terdiri dari beberapa atribut seperti *id_user*, *username*, *password*, *nama_admin* dan *email_admin*.

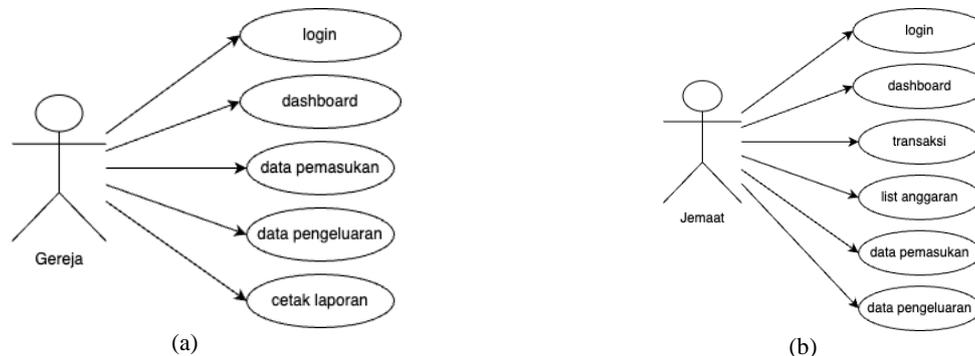


Gambar 3. Entity Relationship Diagram

3.3. Use Case Diagram

Pada Gambar 4 dapat dilihat *use case diagram* gereja dan jemaat. Aktor gereja dapat melakukan *login*, masuk ke halaman *dashboard*, kemudian aktor dapat membuka menu data

pemasukan, data pengeluaran dan cetak laporan. Aktor jemaat dapat melakukan *login*, masuk ke halaman *dashboard*, kemudian aktor dapat membuka menu transaksi, menu list anggaran, data pemasukan dan data pengeluaran.



Gambar 4. Use Case Aktor Gereja (a), Aktor Jemaat (b)

3.4. Rancangan API (*Application Program Interface*)

Rancangan API yang digunakan dalam pembuatan aplikasi *e-budgeting*, dapat dilihat pada Tabel 3.

Tabel 3. Rancangan API

No	Nama	Input	Metode	Path	Output
1	<i>login</i>	Username, password	<i>post</i>	<i>api/login</i>	status, <i>xuser</i>
2	<i>insert data pemasukan</i>	tgl, kd_kategori, <i>user_log</i> , nominal, ket, id_jemaat	<i>post</i>	<i>api/pemasukan</i>	Status, message
3	<i>update data pemasukan</i>	kd_trans, tgl, kd_kategori, <i>user_log</i> , nominal, ket, id_jemaat	<i>patch</i>	<i>api/pemasukan/{pengeluaran}</i>	Status, message
4	<i>delete data pemasukan</i>	Kd_trans	<i>delete</i>	<i>api/pemasukan/{pengeluaran}</i>	Status, message
5	data jemaat		<i>get</i>	<i>api/jemaat</i>	Id_jemaat, nama_jemaat, tempat_lahir, tgl_lahir, no_tlp
6	<i>insert data jemaat</i>	nama_jemaat, tempat_lahir, tgl_lahir, no_tlp	<i>post</i>	<i>api/jemaat</i>	Status, message
7	<i>update data jemaat</i>	Id_jemaat	<i>patch</i>	<i>api/jemaat/edit</i>	Status, message
8	<i>delete data jemaat</i>	Id_jemaat	<i>delete</i>	<i>api/jemaat/delete</i>	Status, message

Pada desain API (*Application Program Interface*), pembuatan endpoint RESTful API dilakukan dengan bertujuan menyediakan layanan komunikasi data antara aplikasi web dan android. Setiap langkah dijalankan secara terstruktur dan mampu menentukan hasil yang benar atau salah [14]. Beberapa *endpoint* Restful yang digunakan pada sistem aplikasi *e-budgeting* Gereja Toraja Jemaat Bintaro adalah sebagai berikut:

a. *Endpoint* RESTful API *Login*

Endpoint RESTful API untuk *login* dengan alamat */api/login*. API ini digunakan untuk majelis gereja dan jemaat melakukan *login*. *Endpoint* RESTful API dapat dilihat pada Tabel 4.

Tabel 4. *Endpoint* RESTful API *Login*

Method	POST
Path	<i>api/login</i>
Header	-
Body	-
Response	{ "status": "sukses", "message": "login berhasil" }

b. *Endpoint* Data Pemasukan

Endpoint RESTful API untuk data pemasukan dengan alamat /api/pemasukan dapat dilihat pada Tabel 5. *Endpoint* data pemasukan menampilkan proses semua data pemasukan yang tercatat dalam sistem.

Tabel 5. *Endpoint* RESTful API Data Pemasukan

Method	GET
Path	api/pemasukan
Header	-
Body	-
Response	<pre>{ "status": "success", "message": "Detail data pemasukan berhasil diambil", "data": { "id": 1, "tanggal": "2025-01-01", "jumlah": 5000000, "sumber": "Sumbangan jemaat", "keterangan": "Sumbangan dari jemaat pada ibadah minggu" } }</pre>

c. *Endpoint* Data Pengeluaran

Endpoint RESTful API untuk data pengeluaran dengan alamat/api/pengeluaran dapat dilihat pada Tabel 6. *Endpoint* data pengeluaran menampilkan proses semua data pengeluaran yang tercatat dalam sistem.

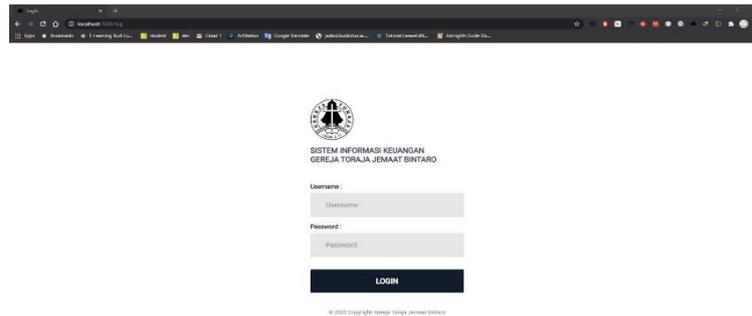
Tabel 6. *Endpoint* RESTful API Data Pengeluaran

Method	GET
Path	api/pengeluaran
Header	-
Body	-
Response	<pre>{ "status": "success", "data": [{ "id": 1, "tanggal": "2025-01-03", "jumlah": 1500000, "kategori": "Biaya operasional", "keterangan": "Listrik dan air" }, { "id": 2, "tanggal": "2025-01-07", "jumlah": 500000, "kategori": "Kegiatan ibadah", "keterangan": "Persiapan acara" }] }</pre>

3.5. Tampilan Layar

a. Tampilan *Login Website Admin*

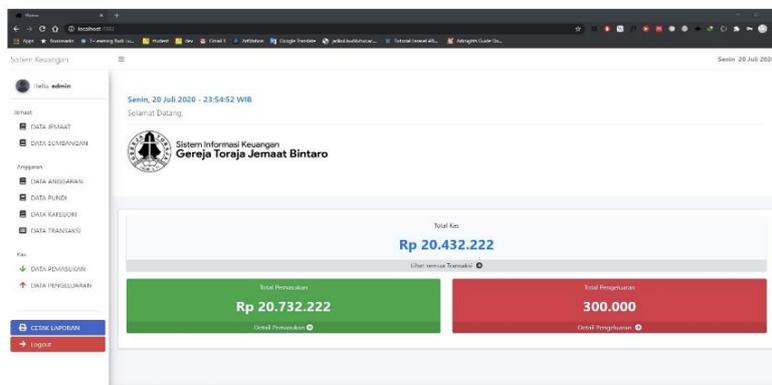
Tampilan halaman *login* di *Website Admin* keuangan Gereja Toraja Jemaat Bintaro dapat dilihat pada Gambar 5. Hanya admin yang memiliki akses untuk masuk ke *website* dengan memasukkan *username* serta *password*.



Gambar 5. Tampilan Layar Login Website Admin

b. Tampilan Halaman Utama Website Admin

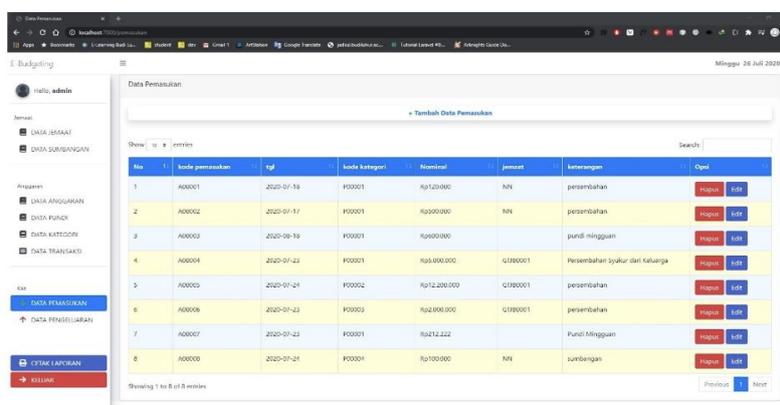
Pada Gambar 6 tampilan layar terdapat menu Data Jemaat, Data Sumbangan, Data Anggaran, Data Pundi, Data Kategori, Data Transaksi, Data Pemasukan, Data Pengeluaran dan Cetak Laporan.



Gambar 6. Tampilan Halaman Utama Website Admin

c. Tampilan Data Pemasukan Website Admin

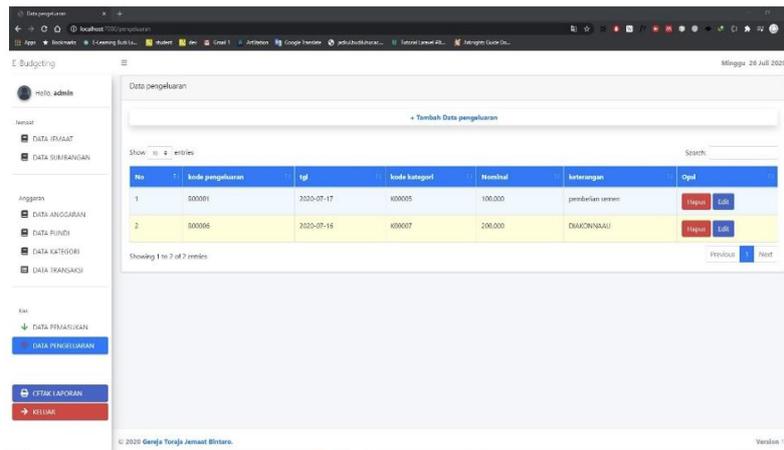
Tampilan Data Pemasukan Website Admin pada Gambar 7 terdapat menu Data, dimana admin dapat melihat dan mengelola data pemasukan Gereja Toraja Jemaat Bintaro.



Gambar 7. Tampilan Data Pemasukan Website Admin

d. Tampilan Data Pengeluaran Website Admin

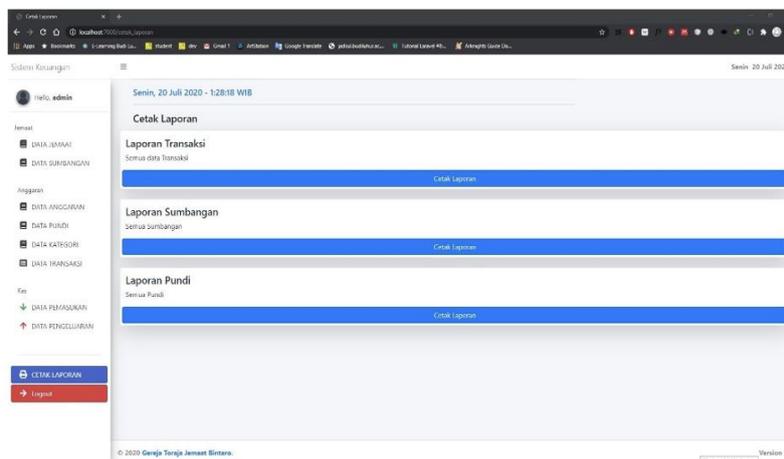
Tampilan Data Pengeluaran Website Admin pada Gambar 8, admin dapat melihat dan mengelola data pengeluaran di Gereja Toraja Jemaat Bintaro.



Gambar 8. Tampilan Data Pengeluaran *Website Admin*

e. Tampilan Cetak Laporan *Website Admin*

Tampilan Cetak Laporan *Website Admin* pada Gambar 9, admin dapat mencetak data laporan.



Gambar 9. Tampilan Cetak Laporan *Website Admin*

f. Tampilan Halaman Utama dan Transaksi Aplikasi Android Jemaat

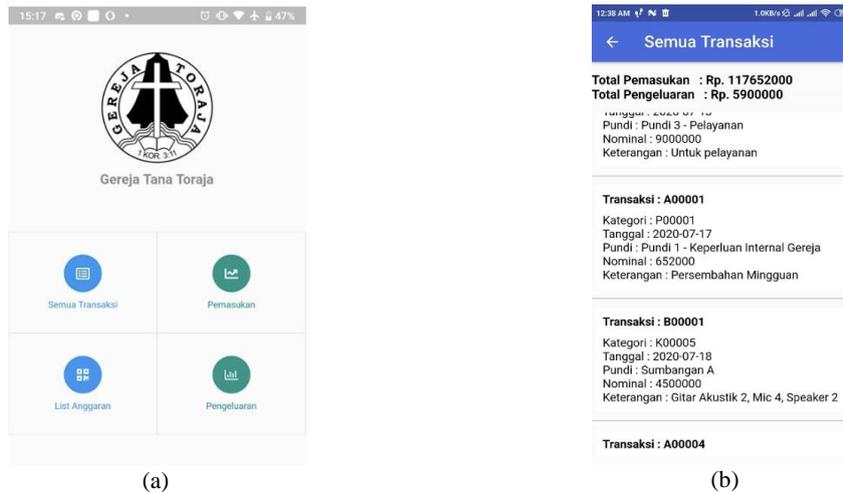
Pada Gambar 10(a) merupakan halaman utama aplikasi android jemaat, terdapat beberapa *button* yang dapat dipilih sesuai data yang inginkan, seperti semua transaksi, list anggaran, pemasukan dan pengeluaran. Sedangkan Gambar 10(b) merupakan tampilan keseluruhan transaksi yang dapat dilihat berdasarkan data pemasukan dan data pengeluaran.

g. Tampilan List Anggaran dan Pemasukan Aplikasi Android Jemaat

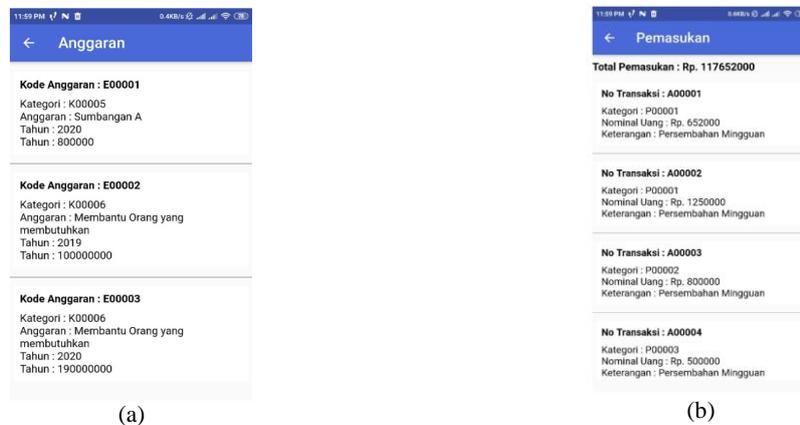
Tampilan yang ditunjukkan pada Gambar 11(a), jemaat dapat melihat data anggaran yang telah diinput oleh majelis gereja, sedangkan pada Gambar 11(b) jemaat dapat melihat semua data pemasukan.

h. Tampilan Pengeluaran Aplikasi Android Jemaat

Tampilan pengeluaran aplikasi android jemaat, untuk melihat semua data pengeluaran jemaat gereja yang ditunjukkan pada Gambar 12.



Gambar 10. Tampilan Halaman Utama (a), dan Tampilan Transaksi (b) Aplikasi Android Jemaat



Gambar 11. Tampilan Halaman Transaksi (a), dan Tampilan Pemasukan (b) Aplikasi Android Jemaat



Gambar 12. Tampilan Pengeluaran Aplikasi Android Jemaat

3.6. Pengujian

Pengujian dilakukan menggunakan metode pengujian, yaitu *user acceptance test*. Pengujian ini untuk mengukur tingkat kegunaan menurut penerimaan pengguna. Nilai skala likert dapat dilihat pada Tabel 7 [15]:

Tabel 7. Nilai Skala Likert

Nilai	Kriteria
0%-19,99%	Tidak Baik
20%-39,99%	Kurang Baik
40%-59,99%	Cukup
60%-79,99%	Baik
80%-100%	Sangat Baik

Total skor tertinggi untuk nilai sangat baik adalah $5 \times 20 = 100$, sedangkan skor terkecil nilai tidak baik $1 \times 20 = 20$, setelah kuesioner dibagikan kepada 20 responden, untuk hasil rekap dapat dilihat pada Tabel 8.

Tabel 8. Nilai Kepuasan Pengguna

No	Pertanyaan	% Nilai	Keterangan
1	Apakah tampilan aplikasi <i>e-budgeting</i> ini mudah dipahami?	82%	Sangat Baik
2	Apakah menu-menu aplikasi <i>e-budgeting</i> mudah dipahami?	74%	Baik
3	Apakah aplikasi <i>e-budgeting</i> mudah dipahami?	79%	Baik
4	Apakah aplikasi <i>e-budgeting</i> ini mempermudah mendapatkan informasi khususnya keuangan?	75%	Baik
5	Apakah tampilan antarmuka aplikasi <i>e-budgeting</i> sesuai dengan kategori pengguna?	79%	Baik
6	Apakah tidak membutuhkan waktu lama memahami aplikasi <i>e-budgeting</i> ?	82%	Sangat Baik
7	Apakah aplikasi <i>e-budgeting</i> memberikan pesan jika terdapat kesalahan?	78%	Baik
8	Apakah aplikasi <i>e-budgeting</i> menampilkan halaman yang tidak sesuai dengan halaman yang diinginkan?	77%	Baik
9	Apakah aplikasi <i>e-budgeting</i> bersifat <i>user friendly</i> ?	79%	Baik
10	Apakah aplikasi <i>e-budgeting</i> sudah cukup baik?	78%	Baik

Pada Tabel 8 menunjukkan nilai kepuasan pengguna dari masing masing pertanyaan kuesioner yang di ajukan. Berdasarkan dari presentasi di atas rata-rata total keseluruhan dari presentasi tersebut berjumlah 78.3% dengan baik, sehingga aplikasi *e-budgeting* dapat diterima oleh majelis gereja dan jemaat.

4. KESIMPULAN DAN SARAN

Hasil implementasi menunjukkan bahwa aplikasi *e-budgeting* mendapatkan respons positif dari pengguna, dengan tingkat penerimaan sebesar 78,3%. Kemudahan penggunaan sistem ini dinilai sangat baik oleh pengguna dengan nilai 79%, Sistem ini berhasil mengatasi berbagai permasalahan dalam kesalahan pencatatan, kurangnya transparansi, serta membutuhkan waktu yang lama dalam penyusunan laporan keuangan. Sistem ini mempermudah pengurus gereja dalam mengelola anggaran, serta meningkatkan transparansi keuangan dengan memberikan akses laporan keuangan secara *real-time* kepada jemaat. Penelitian selanjutnya dapat dikembangkan lebih lanjut untuk menambahkan fitur-fitur, seperti donasi digital dan integrasi dengan layanan pembayaran *online*.

DAFTAR PUSTAKA

- [1] N. Ritonga, "Teologi Sebagai Landasan Bagi Gereja Dalam Mengembangkan Pendidikan Agama Kristen," *Jurnal Shanana*, vol. 4, no. 1, pp. 21–40, 2020.
- [2] F. Al Isfahani, et al, "Implementasi Web Service untuk Aplikasi Pemantau Coronavirus Disease 2019 (COVID-19)," *Innovation in Research of Informatics (INNOVATICS)*, vol. 2, no. 1, pp. 33–39, 2020.
- [3] D. I. Ulumi and A. Perwitasari, "Web Service pada Sistem Curriculum Vitae (CV) Dosen Terintegrasi Berbasis Arsitektur REST API," *Jurnal Sistem dan Teknologi Informasi (JUSTIN)*, vol. 12, no. 2, pp. 309-314, 2024.
- [4] M. Saifudin and I. A. Kautsar, "Perancangan Aplikasi Inventory E-Canteen Berbasis Web Service," *Journal of Internet and Software Engineering*, vol. 1, no. 1, pp. 1-18, 2024.

- [5] K. Kurniawati and M. Badrul, "Penerapan Metode waterfall untuk Perancangan Sistem Informasi Inventory Pada Toko Keramik Bintang Terang," *Jurnal PROSISKO*, vol. 8, no. 2, pp. 52–57, 2021.
- [6] G. A. D.S.S, "Analisa Kebutuhan Kebutuhan Sistem Informasi Manajemen Perusahaan Dagang," *Kurawal - Jurnal Teknologi, Informasi dan Industri*, vol. 4, no. 1, pp. 17–30, 2021.
- [7] Y. Firmansyah, et al, "Implementasi Model Prototye Dalam Pembuatan Aplikasi Pemesanan dan Perawatan Taxi Berbasis Website," *Nuasa Informatika*, vol. 18, no. 2, pp. 1-10, 2024.
- [8] M. Z. Abdullah, et al, "Rancang Bangun Sistem Informasi Akuntansi Berbasis Website menggunakan Framework Laravel," *Jurnal Sains, Teknologi dan Industri*, vol. 18, no. 1, pp. 49-56, 2020.
- [9] I. Setiawan, "Rancangan Layanan Informasi Anggota DPRD Kabupaten Brebes Berbasis Web Service dengan SOAM L," *Jurnal Sistem Cerdas*, vol. 2, pp. 218–229, 2019.
- [10] A. Rachmaniar, D. Diana, and M. Saefudin, "Mobile Application for Reservation Of Meeting Rooms And Event Spaces At Marquee Executive Offices," *Journal of Information System, Informatics and Computing*, vol. 8, no. 1, pp. 14-28, 2024.
- [11] I. Saputri, M. D. Irawan, and M. Alfarisi, "Implementasi Metode Waterfall Dalam Sistem Aplikasi Money Recording Berbasis Web," *Bulletin of Computer Science Research*, vol. 4, no. 2, pp. 181–187, 2024.
- [12] T. Tugiman, D. M. U. Atmaja, and N. Suwaryo, "Analisis dan Perancangan Sistem Elektronik Toko (E-Toko) Kelontong," *INNOVATIVE: Journal of Social Science Research*, vol. 4, no. 2, pp. 775–789, 2024.
- [13] P. A. Bernanda, S. N. Asmah, and A. Maulana, "Rancang Bangun Sistem Informasi Monitoring Perkembangan Balita berbasis MultiPlatform," *Jurnal Komputer Antartika*, vol. 1, no. 3, pp. 99-107, 2023.
- [14] P. I. Daulay and Yahfizham, "Penerapan Algoritma Pemrograman dalam Pembelajaran Ilmu Komputer," *Jurnal Arjuna: Publikasi Ilmu Pendidikan, Bahasa dan Matematika*, vol. 1, no. 6, pp. 91–103, 2023.
- [15] A. S. Kusuma, M. N. Solichin, and M. E. Yuliana, "Perancangan Aplikasi Duta Bangsa E-Kantin Sebagai Media Komunikasi Digital Berbasis Android," *IDEALIS: Indonesia Journal Information System*, vol. 7, no. 2, pp. 128–135, 2024.