p-ISSN : 2406-7733
e-ISSN : 2597-9922

IMPLEMENTASI *RANDOM FOREST* UNTUK PENCATATAN TRANSAKSI KEUANGAN PADA APLIKASI BERBASIS ANDROID

Annisaa Irsalina Razita[1], Indah Puji Astuti[2], Adi Fajaryanto Cobantoto[3]
Teknik Informatika, Fakultas Teknik, Universitas Muhammadiyah Ponorogo
Jl. Budi Utomo No.10, Ponorogo
E-mail: annisaairsalinarazita@gmail.com[1], indahpujiastuti@umpo.ac.id[2], adifajaryanto@umpo.ac.id[3]

Abstrak – Pencatatan keuangan pribadi secara manual sering kali tidak efisien, memakan waktu, dan rentan terhadap kesalahan kategorisasi transaksi. Penelitian ini bertujuan untuk mengembangkan aplikasi pencatatan transaksi keuangan berbasis Android dengan sistem klasifikasi otomatis menggunakan algoritma *Random Forest* dan pembobotan *Term Frequency-Inverse Document Frequency* (TF-IDF). Pengembangan aplikasi ini menerapkan metode Waterfall, serta mengintegrasikan Rule-Based System untuk mengevaluasi laporan keuangan pengguna ke dalam status hemat, normal, atau boros. Model dilatih menggunakan total 1.781 data, yang terdiri atas 1.200 data dari *dataset* Kaggle dan 581 data transaksi riil yang dikumpulkan selama penelitian. Hasil pengujian model menggunakan *Confusion Matrix* menunjukkan performa yang sangat baik dengan akurasi sebesar 90,15%, *precision* 92,04%, *recall* 90,15%, dan *F1-score* 90,08%. Sementara itu, pengujian fungsional sistem melalui *Blackbox Testing* menunjukkan seluruh fitur aplikasi berjalan lancar sesuai harapan. Penelitian ini membuktikan bahwa implementasi *Random Forest* pada platform mobile efektif meminimalkan kesalahan pencatatan manual serta mengoptimalkan efisiensi manajemen keuangan personal.

Kata Kunci: Android, Keuangan Pribadi, Klasifikasi Transaksi, *Random Forest*, TF-IDF.

I. PENDAHULUAN

Manajemen keuangan pribadi merupakan salah satu aspek penting dalam menjaga stabilitas finansial seseorang. Penerapan pengelolaan keuangan yang baik melalui kegiatan pencatatan, penganggaran, menabung, dan investasi dapat membantu individu mencapai kondisi keuangan yang lebih sehat serta mempersiapkan kebutuhan di masa mendatang (Syaffiyah et al., 2025). Selain itu, perilaku pencatatan keuangan juga terbukti berpengaruh terhadap perilaku keuangan seseorang dalam mengelola pemasukan dan pengeluaran (Muasyaroh & Yoga, 2025).

Perkembangan teknologi perangkat bergerak mendorong munculnya berbagai aplikasi pengelolaan keuangan pribadi berbasis Android yang memudahkan pengguna dalam melakukan pencatatan transaksi secara digital (Lampang et al., 2023; Lin & Hoendarto, 2021). Namun, sebagian besar aplikasi masih mengharuskan pengguna menentukan kategori transaksi secara manual sehingga proses pencatatan menjadi kurang efisien dan berpotensi menimbulkan kesalahan klasifikasi (Artanto et al., 2025).

Salah satu pendekatan yang dapat digunakan untuk mengatasi permasalahan tersebut adalah penerapan algoritma *machine learning*. *Random Forest* merupakan algoritma klasifikasi yang memiliki kemampuan generalisasi yang baik, tahan terhadap *overfitting*, serta mampu menghasilkan tingkat akurasi yang tinggi pada berbagai permasalahan klasifikasi (Azzahra & Imlika, 2024; Lampang et al., 2023; Widyayanti, 2022). Oleh karena itu, penelitian ini mengimplementasikan algoritma *Random Forest* untuk melakukan klasifikasi kategori transaksi secara otomatis pada aplikasi pencatatan keuangan berbasis Android.

II. TINJAUAN PUSTAKA

Saputra dan Widayat (2025) mengembangkan aplikasi manajemen keuangan pribadi berbasis Android dengan menerapkan algoritma *Naïve Bayes* untuk mengklasifikasikan transaksi keuangan. Hasil penelitian menunjukkan bahwa penerapan algoritma klasifikasi mampu membantu pengguna dalam mengelompokkan transaksi secara otomatis sehingga proses pencatatan menjadi lebih efektif.

Penelitian oleh Fauzi et al. (2025) menerapkan algoritma *Random Forest* pada klasifikasi risiko kredit dengan mengombinasikan *feature engineering* dan *hyperparameter tuning*. Hasil penelitian menunjukkan bahwa *random forest* mampu meningkatkan performa klasifikasi sehingga menghasilkan tingkat akurasi yang baik dan dapat diterapkan pada berbagai permasalahan klasifikasi.

Penelitian oleh Lampang et al. (2025) berhasil mengembangkan aplikasi pengelolaan keuangan pribadi berbasis Android yang memudahkan pengguna dalam melakukan pencatatan transaksi. Namun, proses pengelompokan kategori transaksi masih dilakukan secara manual sehingga berpotensi mengurangi efisiensi pencatatan. Berdasarkan penelitian-penelitian tersebut, penelitian ini mengembangkan aplikasi pencatatan transaksi keuangan berbasis Android dengan menerapkan algoritma *Random Forest* untuk melakukan

p-ISSN : 2406-7733
e-ISSN : 2597-9922

klasifikasi kategori transaksi secara otomatis serta mengintegrasikan *Rule-Based System* untuk memberikan informasi kondisi keuangan pengguna.

Machine Learning

Machine Learning merupakan cabang dari kecerdasan buatan (*Artificial Intelligence*) yang memungkinkan komputer mempelajari pola dari data sehingga dapat menghasilkan prediksi atau keputusan tanpa diprogram secara eksplisit. Melalui proses pelatihan (*training*), model mampu mengenali karakteristik data dan digunakan untuk menyelesaikan berbagai permasalahan seperti klasifikasi, regresi, maupun prediksi (Daqiqil, 2021). Penerapan *machine learning* telah berkembang pada berbagai bidang, mulai dari klasifikasi teks hingga klasifikasi citra. Integrasi model *machine learning* ke dalam aplikasi juga memungkinkan proses klasifikasi dilakukan secara otomatis dan hasilnya dapat disampaikan kepada pengguna secara lebih cepat dan efisien (Cobantoro et al., 2023). Dalam perkembangannya, machine learning dan data mining telah banyak dimanfaatkan untuk mengolah data menjadi informasi yang bermanfaat. Berbagai metode data mining digunakan untuk menemukan pola dan hubungan antar data sebagai dasar dalam mendukung proses pengambilan keputusan (Astuti et al., 2022).

Random Forest

Random Forest merupakan salah satu algoritma *supervised learning* yang digunakan untuk menyelesaikan permasalahan klasifikasi maupun regresi. Algoritma ini bekerja dengan membangun sejumlah *decision tree* menggunakan teknik *bootstrap sampling*, kemudian menghasilkan prediksi akhir berdasarkan mekanisme *majority voting*. Pendekatan tersebut mampu meningkatkan akurasi model, mengurangi risiko *overfitting*, serta memiliki kemampuan generalisasi yang baik pada berbagai jenis data (Daqiqil, 2021).

Term Frequency–Inverse Document Frequency (TF-IDF)

Term Frequency–Inverse Document Frequency (TF-IDF) merupakan metode pembobotan kata yang digunakan untuk mengubah data teks menjadi representasi numerik berdasarkan tingkat kepentingan suatu kata dalam dokumen. Metode ini banyak digunakan pada proses *text mining* sebagai tahapan ekstraksi fitur sebelum dilakukan proses klasifikasi menggunakan algoritma *machine learning* (Martanto, 2023).

Android

Android merupakan sistem operasi berbasis Linux yang dikembangkan untuk perangkat bergerak dan menyediakan *platform* terbuka bagi pengembang untuk membangun berbagai aplikasi *mobile*. Android mendukung integrasi dengan berbagai teknologi seperti basis data, *web service*, dan *machine learning*, sehingga banyak digunakan dalam pengembangan aplikasi pencatatan keuangan (Prabowo et al., 2020).

Metode Waterfall

Metode *Waterfall* merupakan salah satu model pengembangan perangkat lunak yang dilakukan secara bertahap dan sistematis, dimulai dari tahap analisis kebutuhan, perancangan, implementasi, pengujian, hingga pemeliharaan. Setiap tahapan harus diselesaikan sebelum melanjutkan ke tahapan berikutnya sehingga proses pengembangan menjadi lebih terstruktur (Pargaonkar, 2023).

Rule-Based System

Rule-Based System merupakan metode pengambilan keputusan yang menggunakan sekumpulan aturan (*if–then rules*) untuk menghasilkan suatu keputusan berdasarkan kondisi tertentu. Pada penelitian ini, *Rule-Based System* digunakan untuk menentukan status kondisi keuangan pengguna menjadi kategori Hemat, Normal, atau Boros berdasarkan rasio pengeluaran terhadap saldo tersedia pada periode yang dipilih (Runtu et al., 2025).

Blackbox Testing

Blackbox Testing merupakan metode pengujian perangkat lunak yang dilakukan dengan menguji fungsi-fungsi sistem berdasarkan masukan dan keluaran tanpa memperhatikan struktur kode program. Metode ini digunakan untuk memastikan seluruh fitur aplikasi berjalan sesuai dengan kebutuhan fungsional pengguna (Setiawan, 2021).

III. METODE PENELITIAN

A. Analisis Sistem

Penelitian ini bertujuan mengembangkan aplikasi pencatatan transaksi keuangan berbasis Android yang mampu mengklasifikasikan kategori transaksi secara otomatis menggunakan algoritma *Random Forest*. Sistem yang dikembangkan juga menerapkan *Rule-Based System* untuk memberikan informasi kondisi keuangan pengguna berdasarkan rasio pengeluaran terhadap saldo tersedia. Pengembangan aplikasi dilakukan menggunakan metode *Waterfall* yang terdiri dari tahapan analisis kebutuhan, perancangan sistem, implementasi, pengujian, dan pemeliharaan.

B. Analisis Masalah

Proses pencatatan keuangan pribadi masih banyak dilakukan secara manual sehingga pengguna harus menentukan kategori transaksi secara mandiri. Cara tersebut kurang efisien, terutama ketika jumlah transaksi semakin banyak. Selain itu, sebagian besar aplikasi pencatatan keuangan hanya menyediakan fitur pencatatan tanpa memberikan analisis kondisi

p-ISSN : 2406-7733
e-ISSN : 2597-9922

keuangan pengguna berdasarkan pola pengeluarannya. Oleh karena itu, diperlukan aplikasi yang mampu mengklasifikasikan transaksi secara otomatis sekaligus memberikan informasi kondisi keuangan sebagai bahan evaluasi bagi pengguna.

C. Analisis Kebutuhan Sistem

1. Data Penelitian

Penelitian menggunakan dataset sebanyak 1.781 data transaksi yang terdiri atas 1.200 data dari *Kaggle* dan 581 data transaksi riil. Dataset dikelompokkan ke dalam sembilan kategori transaksi, yaitu *Food and Drink*, *Transport*, *Shopping*, *Health*, *Education*, *Bills*, *Entertainment*, *Home*, dan *Others*.

2. Perangkat Keras

a. Laptop AMD Ryzen 5 5500U
b. RAM 8 GB
c. Smartphone Android untuk pengujian aplikasi

3. Perangkat Lunak

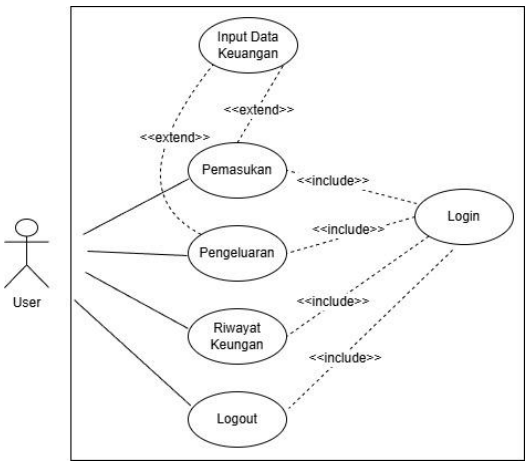
a. Windows 11
b. Android Studio
c. Java
d. Python
e. Flask
f. MySQL
g. Firebase Authentication
h. Visual Studio Code

D. Perancangan Sistem

Tahap perancangan dilakukan untuk menggambarkan proses kerja sistem sebelum diimplementasikan. Perancangan meliputi perancangan arsitektur sistem, *Use Case Diagram, Activity Diagram*, struktur basis data, serta rancangan antarmuka aplikasi.

1. Use Case Diagram

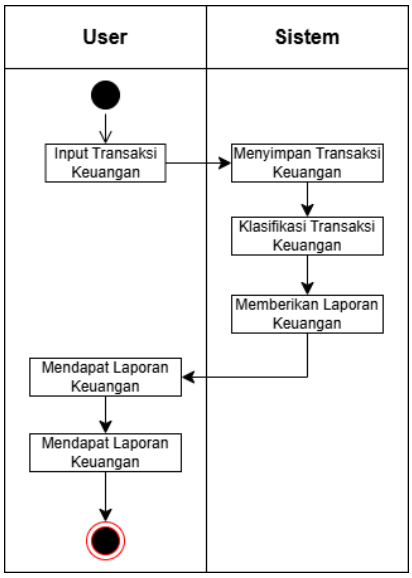
Use Case Diagram menggambarkan interaksi antara pengguna (*user*) dengan sistem pencatatan keuangan. Pengguna dapat melakukan *login*, menambahkan transaksi pemasukan maupun pengeluaran, melihat riwayat transaksi, mengelola *reminder*, melihat laporan keuangan, serta melakukan *logout*. Diagram ini digunakan untuk menunjukkan kebutuhan fungsional sistem secara keseluruhan.



Gambar 1. *Use Case Diagram*

2. Activity Diagram

Activity Diagram menggambarkan alur proses sistem mulai dari pengguna memasukkan data transaksi, penyimpanan data ke basis data, proses klasifikasi kategori transaksi menggunakan algoritma *Random Forest* melalui *web service* Flask, hingga sistem menampilkan hasil klasifikasi dan laporan keuangan kepada pengguna.

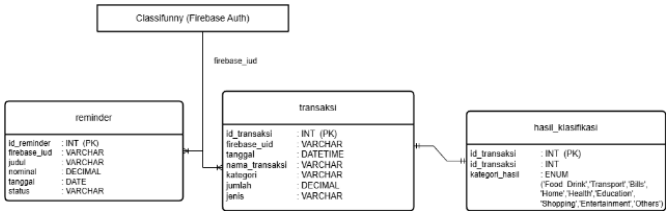


Gambar 2. *Activity Diagram*

3. *Entity Relationship Diagram* (ERD)

Entity Relationship Diagram (ERD) menggambarkan struktur basis data yang digunakan pada sistem. Basis data terdiri atas entitas *User*, Transaksi, *Reminder*, dan Hasil Klasifikasi yang saling berhubungan. Setiap pengguna dapat memiliki banyak transaksi dan *reminder*, sedangkan setiap hasil klasifikasi berhubungan dengan satu data transaksi. Struktur basis data ini mendukung proses penyimpanan data, klasifikasi kategori transaksi menggunakan algoritma *random forest*, serta penyajian laporan keuangan.

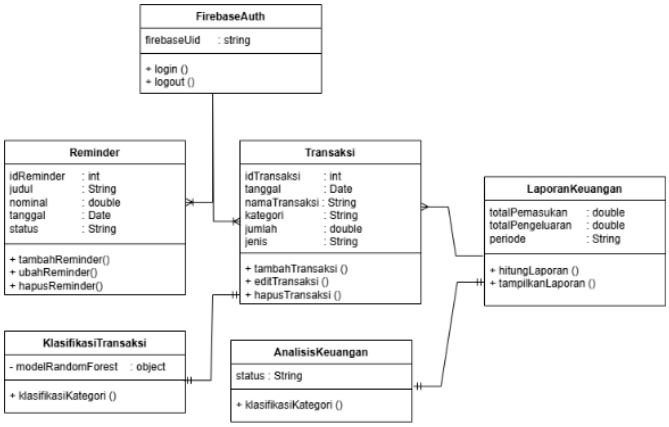
p-ISSN : 2406-7733
e-ISSN : 2597-9922



Gambar 3. *Entity Relationship Diagram*

4. *Class Diagram*

Class Diagram menggambarkan struktur kelas yang digunakan dalam pengembangan aplikasi. Diagram ini terdiri atas kelas FirebaseAuth, Transaksi, KlasifikasiTransaksi, *Reminder*, LaporanKeuangan, dan AnalisisKeuangan beserta atribut, metode, dan hubungan antar kelas. Diagram ini menjadi acuan implementasi sistem agar setiap komponen dapat berinteraksi sesuai fungsinya.



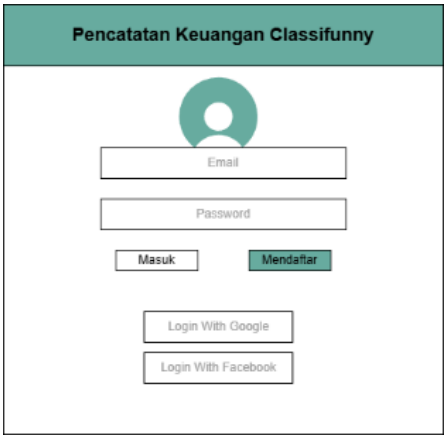
Gambar 4. *Class Diagram*

5. Perancangan *Interface*

Perancangan antarmuka dilakukan untuk memberikan gambaran tampilan aplikasi sebelum proses implementasi. Antarmuka dirancang dengan memperhatikan aspek *user friendly* sehingga pengguna dapat melakukan pencatatan transaksi, melihat hasil klasifikasi, mengelola *reminder*, serta memantau kondisi keuangan dengan mudah melalui perangkat Android.

a. Halaman *Login*

Halaman *login* merupakan halaman awal yang digunakan pengguna untuk mengakses aplikasi. Halaman ini menyediakan dua kolom masukan, yaitu Email dan Password, sebagai proses autentikasi pengguna. Selain itu, tersedia tombol Masuk untuk melakukan *login*, tombol Mendaftar bagi pengguna baru, serta fitur *Login with Google* dan *Login with Facebook* sebagai alternatif proses autentikasi.



Gambar 5. Halaman *Login*

b. Halaman *Dashboard*



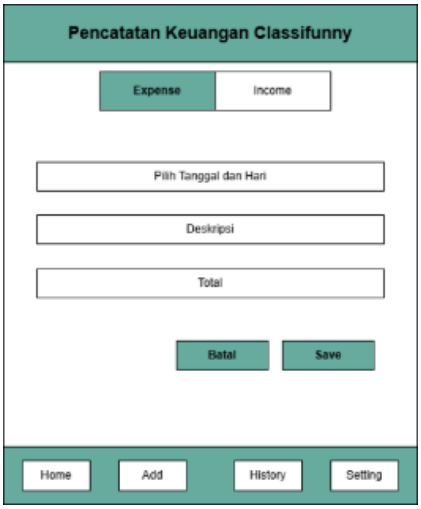
Gambar 6. Halaman *Dashboard*

Halaman *dashboard* merupakan halaman utama aplikasi yang menampilkan ringkasan informasi keuangan pengguna. Informasi yang disajikan meliputi saldo keuangan saat ini, daftar *reminder* tagihan, serta grafik pengeluaran bulanan yang memberikan gambaran kondisi keuangan pengguna secara ringkas.

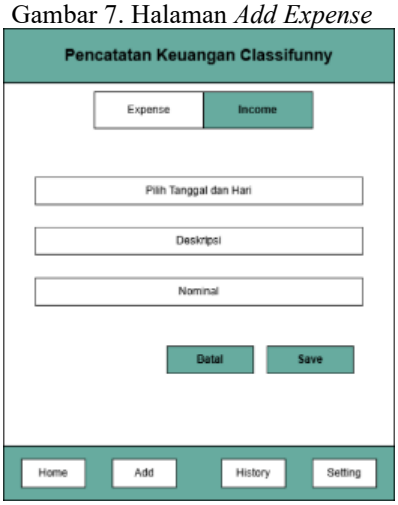
c. Halaman *Add*

Gambar 7 menunjukkan *form* pencatatan pengeluaran yang digunakan untuk memasukkan data transaksi berupa tanggal dan hari, deskripsi pengeluaran, serta total nominal pengeluaran sebelum disimpan ke dalam sistem. Gambar 8 menunjukkan *form* pencatatan pemasukan yang digunakan untuk memasukkan data transaksi berupa tanggal dan hari, deskripsi pemasukan, serta nominal pendapatan sebelum disimpan ke dalam sistem.

p-ISSN : 2406-7733
e-ISSN : 2597-9922

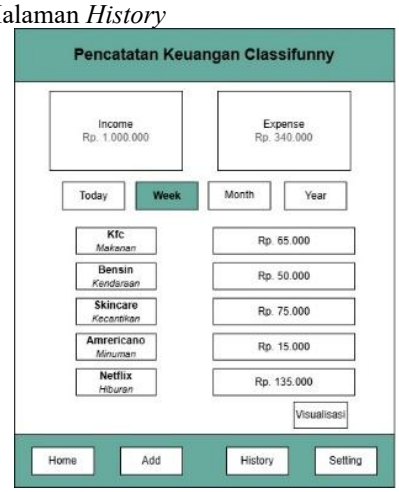


Gambar 7. Halaman *Add Expense*



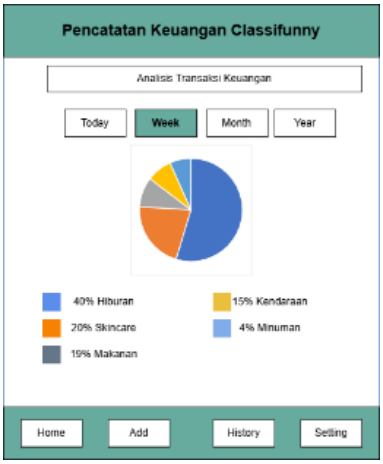
Gambar 8. Halaman *Add Income*

d. Halaman *History*



Gambar 9. Halaman *History*

Halaman tambah transaksi digunakan untuk mencatat transaksi keuangan yang terdiri atas transaksi pengeluaran (*expense*) dan pemasukan (*income*). Pengguna dapat mengisi informasi berupa tanggal transaksi, deskripsi transaksi, serta nominal yang akan disimpan ke dalam sistem. Setelah data dikirim, sistem akan melakukan proses klasifikasi kategori transaksi secara otomatis menggunakan algoritma *Random Forest*.



Gambar 10. Halaman *History*

E. Implementasi Algoritma *Random Forest*

Implementasi algoritma *Random Forest* diawali dengan proses *preprocessing* data transaksi yang meliputi *case folding* dan normalisasi teks untuk menyeragamkan penulisan deskripsi transaksi. Selanjutnya, data teks diubah menjadi representasi numerik menggunakan metode *Term Frequency-Inverse Document Frequency* (TF-IDF) sehingga dapat diproses oleh algoritma klasifikasi.

Model *random forest* dibangun menggunakan parameter *n_estimators* sebanyak 300, *max_depth* sebesar 20, *min_samples_split* sebesar 2, *min_samples_leaf* sebesar 1, dan *random_state* sebesar 42. Model yang telah selesai dilatih kemudian disimpan dalam format.*pkl* dan diintegrasikan ke dalam *web service* Flask. Aplikasi Android mengirimkan data transaksi melalui API menuju *web service*, kemudian model melakukan proses klasifikasi untuk menentukan kategori transaksi secara otomatis. Hasil klasifikasi selanjutnya disimpan ke dalam basis data dan ditampilkan pada aplikasi sebagai kategori transaksi yang sesuai.

F. Pengujian Sistem

Pengujian pada penelitian ini terdiri atas pengujian model klasifikasi dan pengujian fungsional aplikasi. Pengujian model dilakukan menggunakan *Confusion Matrix* untuk memperoleh nilai *accuracy*, *precision*, *recall*, dan *F1-score*, sehingga dapat diketahui tingkat kinerja algoritma *Random Forest* dalam mengklasifikasikan transaksi keuangan.

Selain itu, dilakukan pengujian fungsional menggunakan metode *Black Box Testing* untuk memastikan seluruh fitur aplikasi berjalan sesuai dengan kebutuhan pengguna. Pengujian meliputi proses *login*, penambahan transaksi, klasifikasi

p-ISSN : 2406-7733
e-ISSN : 2597-9922

kategori transaksi secara otomatis, pengelolaan *reminder*, riwayat transaksi, visualisasi laporan keuangan, serta pengaturan bahasa aplikasi. Hasil pengujian digunakan untuk memastikan bahwa setiap fungsi sistem telah berjalan sesuai dengan rancangan dan kebutuhan pengguna.

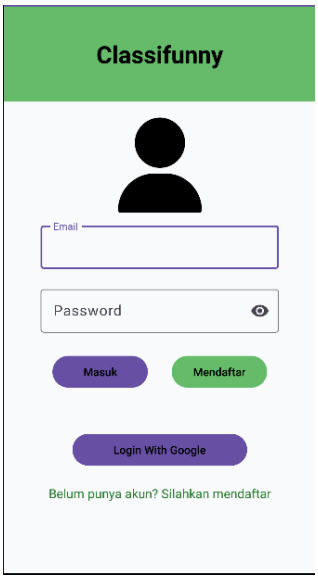
IV. HASIL DAN PEMBAHASAN

Hasil Penelitian

Hasil Tampilan

1. Halaman *Login*

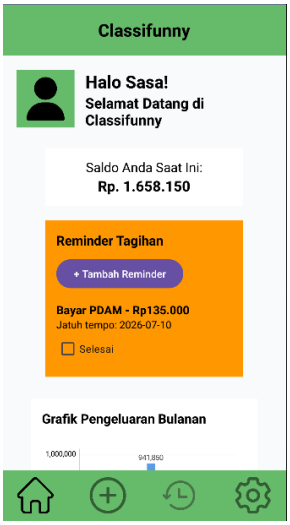
Pada Gambar 11 ditampilkan halaman *login* aplikasi Classifunny. Halaman ini digunakan sebagai proses autentikasi pengguna menggunakan Firebase Authentication. Pengguna dapat masuk menggunakan alamat email dan kata sandi yang telah terdaftar maupun menggunakan akun Google.



Gambar 11. Halaman *Login*

2. Halaman *Dashboard*

Pada Gambar 12 dan Gambar 13 ditampilkan halaman *dashboard* yang menyajikan informasi saldo keuangan, daftar *reminder*, serta grafik pengeluaran berdasarkan periode tertentu. Halaman ini menjadi pusat informasi bagi pengguna untuk memantau kondisi keuangan secara ringkas.



Gambar 12. Halaman *Dashboard*

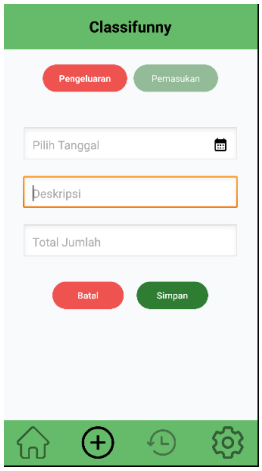


Gambar 13. Halaman *Dashboard*

3. Halaman Tambah Transaksi

Pada Gambar 14 ditampilkan halaman tambah transaksi yang digunakan untuk mencatat transaksi pemasukan maupun pengeluaran. Pengguna mengisi tanggal transaksi, deskripsi transaksi, dan nominal transaksi. Selanjutnya, data dikirim ke *web service* Flask untuk diproses menggunakan algoritma *Random Forest* sehingga kategori transaksi dapat ditentukan secara otomatis.

p-ISSN : 2406-7733
e-ISSN : 2597-9922

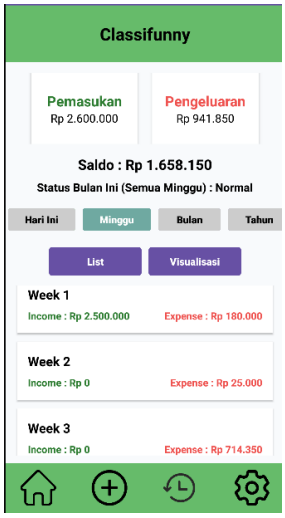


Gambar 14. Halaman *Add* Transaksi

4. Halaman *History*



Gambar 15. Halaman *History*



Gambar 16. Halaman *History*

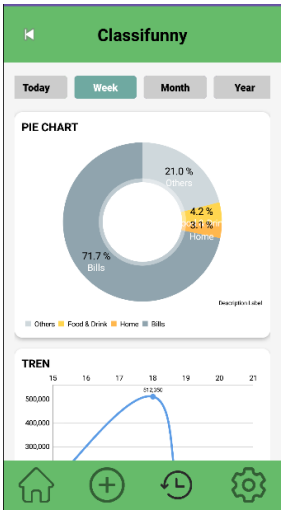


Gambar 17. Halaman Detail *History*

Pada Gambar 15, Gambar 16, dan Gambar 17 ditampilkan halaman riwayat transaksi yang memuat daftar transaksi pemasukan dan pengeluaran berdasarkan periode tertentu. Informasi yang ditampilkan meliputi tanggal transaksi, kategori, jenis transaksi, dan nominal. Selain itu, pengguna dapat memilih salah satu transaksi untuk melihat informasi secara lebih rinci melalui halaman detail transaksi.

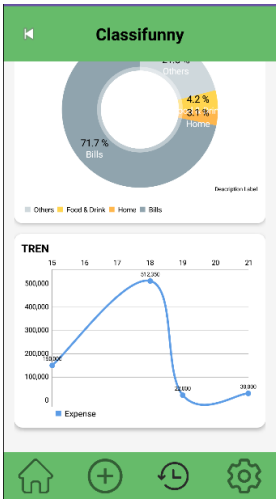
5. Halaman Visualisasi

Pada Gambar 18 dan Gambar 19 ditampilkan visualisasi laporan keuangan dalam bentuk diagram *pie*. Diagram tersebut menunjukkan persentase pengeluaran berdasarkan kategori sehingga memudahkan pengguna dalam menganalisis pola pengeluaran selama periode tertentu.



Gambar 18. Halaman Visualisasi

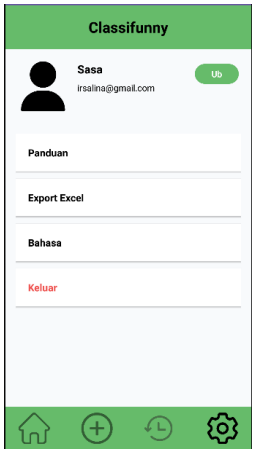
p-ISSN : 2406-7733
e-ISSN : 2597-9922



Gambar 19. Halaman Visualisasi

6. Halaman Setting

Pada Gambar 20 ditampilkan halaman pengaturan aplikasi yang menyediakan berbagai fitur, seperti perubahan bahasa aplikasi, informasi akun pengguna, serta fasilitas *logout*.



Gambar 20. Halaman Setting

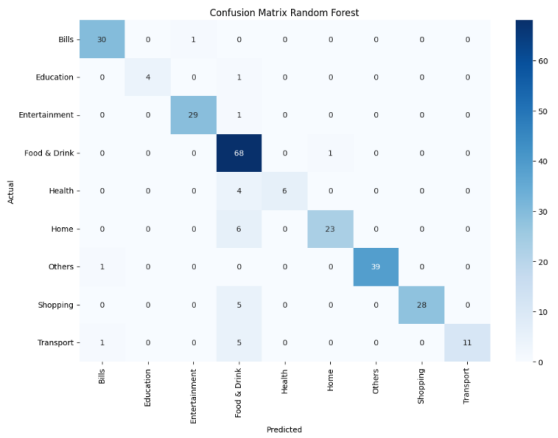
7. Hasil Evaluasi Model *Random Forest*

Tabel 1. Hasil Evaluasi Model *Random Forest*

Metrik	Nilai
Accuracy	90,15%
Precision	92,04%
Recall	90,15%
F1-Score	90,08%

Berdasarkan hasil evaluasi pada Tabel 1, algoritma *Random Forest* memperoleh nilai *accuracy* sebesar 90,15%, *precision* sebesar 92,04%, *recall* sebesar 90,15%, dan *F1-score* sebesar 90,08%. Hasil tersebut menunjukkan bahwa model memiliki kemampuan yang baik dalam mengklasifikasikan kategori transaksi keuangan. Nilai *precision* yang tinggi menunjukkan bahwa sebagian besar hasil prediksi telah sesuai dengan kategori sebenarnya, sedangkan nilai *recall* menunjukkan kemampuan model dalam mengenali data pada setiap kategori dengan baik. Selain itu, nilai *F1-score* yang tinggi menunjukkan keseimbangan antara *precision* dan *recall* sehingga model dapat dikatakan memiliki performa klasifikasi yang baik.

Selanjutnya, hasil klasifikasi divisualisasikan menggunakan *Confusion Matrix* sebagaimana ditunjukkan pada Gambar 21.



Gambar 21. *Confusion Matrix*

Berdasarkan Gambar 21, sebagian besar data transaksi berhasil diklasifikasikan ke dalam kategori yang benar. Hal tersebut ditunjukkan oleh dominasi nilai pada diagonal utama *confusion matrix* yang merepresentasikan prediksi benar (*true prediction*). Sementara itu, nilai di luar diagonal menunjukkan masih terdapat beberapa kesalahan klasifikasi (*misclassification*) pada beberapa kategori transaksi. Meskipun demikian, jumlah kesalahan tersebut relatif kecil dibandingkan jumlah prediksi yang benar, sehingga dapat disimpulkan bahwa algoritma *Random Forest* memiliki kemampuan yang baik dalam membedakan karakteristik setiap kategori transaksi.

8. Hasil Pengujian *Black Box*

Pengujian fungsional dilakukan menggunakan metode *Black Box Testing* untuk memastikan setiap fitur aplikasi berjalan sesuai dengan kebutuhan pengguna. Hasil pengujian ditunjukkan pada Tabel 2.

Tabel 2. Hasil Pengujian *Black Box*

No	Fitur	Hasil Pengujian	Status
1.	*Login*	Berhasil	Valid
2.	Tambah Transaksi	Berhasil	Valid
3.	Klasifikasi Otomatis	Berhasil	Valid
4.	Riwayat Transaksi	Berhasil	Valid
5.	*Reminder*	Berhasil	Valid

p-ISSN : 2406-7733
e-ISSN : 2597-9922

No	Fitur	Hasil Pengujian	Status
6.	Visualisasi Laporan	Berhasil	Valid
7.	Pengaturan Bahasa	Berhasil	Valid
8	*Logout*	Berhasil	Valid

2. Pembahasan

Implementasi algoritma *Random Forest* pada aplikasi Classifunny memberikan keuntungan dalam proses klasifikasi transaksi karena mampu mengolah data dengan karakteristik yang beragam melalui pembentukan sejumlah *decision tree*. Pendekatan tersebut memungkinkan sistem menghasilkan prediksi yang lebih stabil dibandingkan penggunaan satu pohon keputusan, sehingga mampu mengurangi risiko *overfitting* dan meningkatkan kemampuan generalisasi model terhadap data baru. Hal ini sejalan dengan penelitian Fauzi et al. (2025) yang menunjukkan bahwa penerapan *Random Forest* yang dikombinasikan dengan proses *feature engineering* mampu meningkatkan performa klasifikasi pada berbagai kasus, termasuk klasifikasi risiko kredit. Selain itu, penggunaan algoritma *Random Forest* dipilih karena memiliki kemampuan menangani data berdimensi tinggi serta tidak sensitif terhadap keberadaan atribut yang kurang relevan sehingga sesuai diterapkan pada proses klasifikasi transaksi keuangan yang memiliki variasi deskripsi transaksi cukup beragam.

Penggunaan metode TF-IDF sebagai teknik pembobotan kata juga memberikan kontribusi terhadap peningkatan kualitas model klasifikasi. TF-IDF mampu merepresentasikan tingkat kepentingan setiap kata pada deskripsi transaksi sehingga model *Random Forest* dapat membedakan karakteristik masing-masing kategori transaksi dengan lebih baik. Kata-kata yang sering muncul pada kategori tertentu akan memperoleh bobot yang lebih representatif dibandingkan kata-kata umum yang muncul pada hampir seluruh dokumen. Dengan demikian, proses ekstraksi fitur menggunakan TF-IDF mampu menghasilkan representasi numerik yang lebih informatif sebelum dilakukan proses klasifikasi menggunakan algoritma machine learning.

Selain menghasilkan proses klasifikasi secara otomatis, aplikasi yang dikembangkan juga memberikan nilai tambah berupa visualisasi laporan keuangan berdasarkan kategori transaksi. Informasi tersebut membantu pengguna dalam mengidentifikasi pola pengeluaran selama periode tertentu sehingga pengguna dapat melakukan evaluasi terhadap kebiasaan finansialnya. Penyajian informasi dalam bentuk grafik dan ringkasan transaksi juga meningkatkan kemudahan pengguna dalam memahami kondisi keuangan dibandingkan hanya menampilkan daftar transaksi secara tekstual. Pendekatan ini sejalan dengan penelitian Lampang et al. (2023) yang menyatakan bahwa aplikasi pengelolaan keuangan berbasis Android mampu membantu pengguna dalam melakukan pencatatan dan pemantauan kondisi keuangan secara lebih praktis.

Berdasarkan hasil pengujian, aplikasi tidak hanya berhasil mengimplementasikan model klasifikasi dengan tingkat akurasi sebesar 90,15%, tetapi juga mampu menjalankan seluruh fungsi utama sesuai kebutuhan pengguna berdasarkan hasil pengujian *Black Box Testing*. Kondisi tersebut menunjukkan bahwa integrasi antara aplikasi Android, *web service* Flask, Firebase Authentication, dan model *Random Forest* dapat berjalan dengan baik tanpa mengurangi performa sistem. Hasil ini membuktikan bahwa pendekatan yang digunakan mampu menghasilkan aplikasi pencatatan keuangan yang tidak hanya akurat dalam proses klasifikasi, tetapi juga layak digunakan sebagai media pendukung pengelolaan keuangan pribadi.

Aplikasi Classifunny berhasil dikembangkan sebagai sistem pencatatan keuangan pribadi berbasis Android yang terintegrasi dengan Firebase Authentication untuk autentikasi pengguna, *web service* Flask sebagai media komunikasi antara aplikasi dan model klasifikasi, MySQL sebagai basis data penyimpanan transaksi, serta algoritma *Random Forest* untuk melakukan klasifikasi kategori transaksi secara otomatis. Integrasi antar komponen tersebut memungkinkan proses pencatatan dan pengelolaan transaksi dilakukan secara lebih cepat dan efisien karena pengguna tidak perlu menentukan kategori transaksi secara manual.

Proses klasifikasi diawali dengan tahap *preprocessing* berupa *case folding* dan normalisasi teks untuk menyeragamkan deskripsi transaksi. Selanjutnya, deskripsi transaksi diubah menjadi representasi numerik menggunakan metode *Term Frequency-Inverse Document Frequency* (TF-IDF) sebelum diproses oleh model *Random Forest*. Hasil klasifikasi kemudian dikirim kembali ke aplikasi Android melalui *Application Programming Interface* (API) Flask dan disimpan ke dalam basis data sehingga dapat ditampilkan pada halaman riwayat transaksi maupun visualisasi laporan keuangan.

Berdasarkan hasil evaluasi pada Tabel 1, model *Random Forest* memperoleh nilai accuracy sebesar 90,15%, precision sebesar 92,04%, recall sebesar 90,15%, dan F1-score sebesar 90,08%. Nilai tersebut menunjukkan bahwa model memiliki kemampuan yang baik dalam mengklasifikasikan kategori transaksi keuangan. Hal ini juga didukung oleh hasil confusion matrix yang memperlihatkan bahwa sebagian besar data transaksi berhasil diprediksi pada kategori yang benar. Meskipun masih terdapat beberapa kesalahan klasifikasi pada kategori tertentu, jumlahnya relatif kecil sehingga tidak memengaruhi performa model secara keseluruhan.

p-ISSN : 2406-7733
e-ISSN : 2597-9922

Selanjutnya, hasil pengujian fungsional menggunakan metode *Black Box Testing* pada Tabel 2 menunjukkan bahwa seluruh fitur utama aplikasi, meliputi proses *login*, pencatatan transaksi, klasifikasi kategori secara otomatis, pengelolaan *reminder*, riwayat transaksi, visualisasi laporan keuangan, pengaturan bahasa, dan *logout*, telah berjalan sesuai dengan kebutuhan pengguna. Dengan demikian, sistem yang dikembangkan tidak hanya memiliki performa klasifikasi yang baik, tetapi juga mampu memenuhi kebutuhan fungsional sebagai aplikasi pencatatan keuangan pribadi berbasis Android.

V. KESIMPULAN DAN SARAN

Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan, diperoleh beberapa kesimpulan sebagai berikut.

1. Telah berhasil dikembangkan aplikasi Classifunny berbasis Android sebagai sistem pencatatan keuangan pribadi yang terintegrasi dengan Firebase Authentication, *web service* Flask, dan basis data MySQL, sehingga pengguna dapat mengelola data keuangan secara lebih praktis.
2. Algoritma *Random Forest* berhasil diimplementasikan untuk mengklasifikasikan kategori transaksi keuangan secara otomatis berdasarkan deskripsi transaksi. Sebelum proses klasifikasi dilakukan, data diproses melalui tahap case folding, normalisasi teks, dan TF-IDF sebagai representasi fitur.
3. Berdasarkan hasil evaluasi model, algoritma *Random Forest* memperoleh *accuracy* sebesar 90,15%, *precision* sebesar 92,04%, *recall* sebesar 90,15%, dan *F1-score* sebesar 90,08%. Hasil tersebut menunjukkan bahwa model memiliki performa yang baik dalam mengklasifikasikan kategori transaksi keuangan.
4. Berdasarkan hasil pengujian menggunakan *Black Box Testing*, seluruh fitur aplikasi, meliputi *login*, pencatatan transaksi, klasifikasi otomatis, *reminder*, riwayat transaksi, visualisasi laporan keuangan, pengaturan bahasa, dan *logout*, telah berjalan sesuai dengan kebutuhan fungsional sistem.
5. Kelebihan sistem yang dikembangkan adalah mampu melakukan klasifikasi kategori transaksi secara otomatis sehingga membantu pengguna dalam proses pencatatan keuangan. Adapun keterbatasan penelitian ini yaitu model klasifikasi masih bergantung pada data latih yang digunakan sehingga akurasi dapat dipengaruhi oleh variasi dan jumlah data transaksi.

Saran

Berdasarkan hasil penelitian yang telah dilakukan, beberapa saran untuk pengembangan penelitian selanjutnya adalah sebagai berikut.

1. Menambahkan jumlah dan variasi data latih agar model *Random Forest* mampu menghasilkan tingkat akurasi klasifikasi yang lebih tinggi.
2. Membandingkan performa *Random Forest* dengan algoritma klasifikasi lain, seperti *Support Vector Machine* (SVM), *Naïve Bayes*, atau *XGBoost*, sehingga dapat diketahui algoritma yang paling sesuai untuk klasifikasi transaksi keuangan.
3. Mengembangkan fitur analisis keuangan yang lebih lengkap, seperti prediksi pengeluaran, rekomendasi penghematan, target tabungan, serta notifikasi keuangan yang lebih adaptif.
4. Mengembangkan aplikasi agar mendukung sinkronisasi data secara *real-time* serta kompatibel pada berbagai platform selain Android.

DAFTAR PUSTAKA

Artanto, I. S. T., Sidhi, T. A. P., & Ardanari, P. (2025). Pembangunan Sistem Pengelolaan dan Analisis Keuangan Pribadi Berbasis Mobile. *Jurnal Informatika Atma Jogja*, *6*(1), 41–50. https://doi.org/10.24002/jiaj.v6i1.11598

Astuti, I. P., Buntoro, G. A., Studi, P., Informatika, T., Teknik, F., & Ponorogo, U. M. (2022). Analisis Minat Siswa Siswi SMP / MTS Terhadap Ekstrakurikuler MA Muhammadiyah 1 Ponorogo. *KOMPUTEK*, *6*(1), 91–105. https://doi.org/doi.org/10.24269/jkt.v6i1.1145

Azzahra, F. I., & Imlika, L. H. D. (2024). Pengaruh Literasi Keuangan, Inklusi Keuangan, Kontrol Diri, dan Teman Sebaya Terhadap Perilaku Menabung Generasi Z. *Jurnal MEDIA AKADEMIK (JMA)*, *2*(12), 1–13. https://doi.org/10.62281/v2i12.1421

Cobantoro, A. F., Masykur, F., & Rosyadi, M. R. (2023). Implementation of Bot Telegram as Broadcasting Media Classification Results of Convolutional Neural Network (CNN) Images of Rice Plant Leaves. *Journal of Computer Networks , Architecture and High Performance Computing*, *5*(1), 1–9. https://doi.org/doi.org/10.47709/cnahpc.v5i1.1976

Fauzi, N. P. N., Khomsah, S., & Wicaksono, A. D. P. (2025). Penerapan Feature Engineering dan Hyperparameter Tuning untuk Meningkatkan Akurasi Model *Random Forest* pada Klasifikasi Risiko Kredit. *Jurnal Teknologi Informasi Dan Ilmu Komputer (JTIIK)*, *12*(2), 251–262. https://doi.org/10.25126/jtiik.2025128472

Ibnu Daqiqil Id. (2021). *Machine Learning : Teori,*

p-ISSN : 2406-7733
e-ISSN : 2597-9922

Studi Kasus dan Implementasi Menggunakan Python (1st ed.). UR Press. https://doi.org/10.5281/zenodo.5113507

Lampang, M. F., Alam, T. H. I., & Amri, I. (2023). Rancang Bangun Aplikasi Pengelolaan Keuangan Pribadi Berbasis Android. *Framework : Jurnal Ilmu Komputer Dan Informatika*, *1*(02), 146–155. https://doi.org/10.33506/jiki.v1i02.2737

Lin, S., & Hoendarto, G. (2021). Aplikasi Mobile Money Management Dengan Fitur Optical Character Recognition Menggunakan Framework React Native. *Metik Jurnal*, *5*(2), 19–27. https://doi.org/10.47002/metik.v5i2.291

Luthfi Fajri Martanto, M. (2023). *Analisis Sentimen Ulasan Aplikasi M-Paspor pada Google Play Store Menggunakan Metode Multinomial Naive Bayes* [Universitas Multimedia Nusantara]. https://kc.umn.ac.id/id/eprint/33712/

Muasyaroh, H., & Yoga, I. (2025). Dampak literasi, pencatatan, dan penganggaran keuangan terhadap perilaku keuangan mahasiswa. *EKUILNOMI: Jurnal Ekonomi Pembangunan*, *7*(2), 535–546. https://doi.org/10.36985/jv2gcn95

Pargaonkar, S. (2023). A Comprehensive Research Analysis of Software Development Life Cycle (SDLC) Agile & Waterfall Model Advantages, Disadvantages, and Application Suitability in Software Quality Engineering. *International Journal of Scientific and Research Publications*, *13*(8), 120–124. https://doi.org/10.29322/ijsrp.13.08.2023.p14015

Prabowo, I. A., Wijayanto, H., Yudanto, B. W., & Nugroho, S. (2020). Buku Ajar Pemrograman Mobile Berbasis Android. In *Angewandte Chemie International Edition*. Lembaga Penelitian dan Pengabdian Kepada Masyarakat Universitas Dian Nuswantoro Semarang.

Runtu, R. R., Maramis, G. D. P., & Hasibuan, A. (2025). Implementasi Algoritma Rule-Based Classification pada Aplikasi Arsip Web Menggunakan Metode RAD dan Evaluasi ISO / IEC 25010. *Jurnal Minfo Program*, *14*(2), 2792–2798. https://doi.org/10.33395/jmp.v14i2.15505

Saputra, F. A. A., & Widayat, W. (2025). Pengembangan Aplikasi Android untuk Manajemen Keuangan Pribadi Berbasis Algoritma Naïve Bayes. *Jurnal Pendidikan Dan Teknologi Indonesia (JPTI)*, *5*(6), 1699–1711. https://doi.org/10.52436/1.jpti.863

Setiawan, R. (2021). *Black Box Testing Untuk Menguji Perangkat Lunak*. Dicoding. https://www.dicoding.com/blog/black-box-testing/

Syaffiyah, Astuti, R. P., & Puspita, C. D. (2025). Manajemen Keuangan Pribadi Untuk Mencapai Stabilitas Finansial Dengan Strategi : Menabung Dan Berinvestasi. *Jurnal Ilmiah Ekonomi Manajemen Bisnis Dan Akuntansi*, *2*(3), 88–94. https://doi.org/10.61722/jemba.v2i3.882

Widyayanti, E. R. (2022). Pelatihan Manajemen Usaha : Pencatatan Keuangan Usaha Menggunakan Aplikasi Keuangan Digital. *Journal of Community Service and Empowerment*, *3*(1), 1–10. https://doi.org/10.32639/jcse.v3i1.46