

Implementasi Algoritma *Random Forest* untuk Menentukan Tingkat Keberhasilan Proyek pada Sistem *Work Order* (Studi Kasus: PT XYZ)

Unggul Prasetyo Utomo¹, Hadi Zakaria²

^{1,2}Teknik Informatika, Ilmu Komputer, Universitas Pamulang

¹unggul.prasetyo.u@gmail.com, ²dosen00274@unpam.ac.id*

Abstract

PT XYZ is a company focused on technological innovation to provide modern, effective, and efficient solutions across various aspects of life. As a pioneer in the technology evolution industry, PT XYZ combines expertise in software development and the latest technologies to create positive transformation for society and businesses. In project implementation, PT XYZ faces challenges in determining project success levels objectively and measurably, particularly within the context of the work order system. This condition leads to less optimal strategic decision-making, increased risk of losses, and difficulties in conducting comprehensive, data-driven project evaluations. To address these issues, this study develops a web-based project success prediction system within the work order system by implementing the Random Forest algorithm and the Agile development approach. The Random Forest algorithm is developed using the Python programming language to classify project success levels based on several historical parameters, such as completion duration, budget, and profit percentage. The system is equipped with a user interface developed using PHP with the Laravel framework and a MySQL database, enabling efficient and integrated data processing and visualization. The results show that the implementation of the Random Forest algorithm improves prediction accuracy and provides recommendations that can support management in decision-making. The Agile approach also offers high flexibility in adapting the system to user requirements. Through this system, PT XYZ is expected to optimize work order management and proactively minimize the risk of project failure in a data-driven manner.

Keywords: random forest, project success prediction, work order, agile.

Abstrak

PT XYZ merupakan perusahaan yang berfokus pada inovasi teknologi untuk menyediakan solusi modern, efektif, dan efisien di berbagai aspek kehidupan. Sebagai pionir dalam industri evolusi teknologi, PT XYZ menggabungkan keahlian dalam pengembangan perangkat lunak dan teknologi terkini untuk menciptakan transformasi positif bagi masyarakat dan dunia usaha. Dalam pelaksanaan proyek, PT XYZ menghadapi tantangan dalam menentukan tingkat keberhasilan proyek secara objektif dan terukur, khususnya dalam konteks sistem *work order*. Hal ini menyebabkan pengambilan keputusan strategis menjadi kurang tepat, meningkatnya risiko kerugian, serta sulitnya melakukan evaluasi proyek secara menyeluruh dan berbasis data. Menanggapi permasalahan tersebut, peneliti mengembangkan sistem prediksi keberhasilan proyek pada sistem *work order* berbasis web dengan mengimplementasikan algoritma *Random Forest* dan pendekatan pengembangan Agile. Algoritma *Random Forest* dibangun menggunakan bahasa pemrograman Python untuk melakukan klasifikasi tingkat keberhasilan berdasarkan sejumlah parameter historis, seperti durasi penyelesaian, anggaran, dan persentase *profit*. Sistem ini dilengkapi antarmuka pengguna yang dikembangkan menggunakan bahasa pemrograman PHP dengan *framework* Laravel serta basis data MySQL, sehingga memungkinkan pengolahan dan visualisasi data dilakukan secara efisien dan terintegrasi. Hasil pengembangan sistem menunjukkan bahwa penerapan algoritma *Random Forest* mampu meningkatkan akurasi prediksi dan memberikan rekomendasi yang dapat digunakan oleh manajemen dalam pengambilan keputusan. Pendekatan *Agile* yang digunakan juga memberikan fleksibilitas tinggi dalam menyesuaikan sistem terhadap kebutuhan pengguna. Dengan sistem ini, PT XYZ diharapkan dapat mengoptimalkan pengelolaan *work order* serta meminimalkan risiko kegagalan proyek secara lebih proaktif dan berbasis data.

Kata kunci: *random forest*, prediksi keberhasilan proyek, *work order*, *agile*.

©This work is licensed under a Creative Commons Attribution -ShareAlike 4.0 International License

1. Pendahuluan

PT. XYZ adalah perusahaan yang bergerak di bidang inovasi teknologi dengan fokus pada pengembangan solusi modern yang efektif dan efisien di berbagai aspek kehidupan. Sebagai pelopor dalam industri evolusi teknologi, PT XYZ terus mendorong kemajuan melalui pengembangan perangkat lunak dan penerapan teknologi terkini. Namun, dalam pelaksanaan proyeknya, PT XYZ menghadapi tantangan dalam mengukur tingkat keberhasilan proyek secara objektif. Ketidadaan sistem analisis yang terintegrasi menyebabkan perusahaan kesulitan dalam

mengidentifikasi proyek yang berpotensi berhasil atau gagal sejak dini. Hal ini berdampak pada ketidaktepatan dalam pengambilan keputusan strategis dan peningkatan risiko kerugian proyek.

Implementasi adalah proses penerapan ide, konsep, kebijakan, atau inovasi ke dalam tindakan spesifik guna menghasilkan dampak yang dapat diukur. Dampak tersebut dapat berupa perubahan pada tingkat pengetahuan, peningkatan keterampilan, hingga pergeseran nilai dan sikap individu maupun kelompok yang terlibat. Implementasi mencakup rangkaian tahap sistematis dan terencana untuk memastikan bahwa

tujuan yang telah ditetapkan dapat dicapai secara efektif. Dalam praktiknya, proses ini juga melibatkan evaluasi berkelanjutan serta penyesuaian terhadap kondisi lapangan guna mengatasi hambatan yang muncul selama pelaksanaan. Dengan demikian, implementasi tidak hanya berfokus pada realisasi teknis, tetapi juga pada keberhasilan menyeluruh dari proses perubahan yang diusung [1].

Algoritma merupakan landasan logis dalam pemrosesan informasi yang berfungsi untuk menyusun langkah-langkah penyelesaian masalah secara sistematis dan terukur. Dalam konteks pengembangan sistem, algoritma tidak hanya berperan sebagai instruksi teknis, tetapi juga sebagai kerangka berpikir yang memungkinkan perumusan solusi secara efisien, konsisten, dan dapat dieksekusi oleh mesin. Algoritma disusun dengan bahasa yang logis serta mengikuti urutan langkah terbatas yang jelas, sehingga menghasilkan solusi yang dapat dijalankan secara mekanik maupun otomatis [2]. Dalam penelitian ini, algoritma digunakan sebagai pendekatan analitik utama dalam membangun model prediktif berbasis data proyek yang tersimpan pada sistem work order.

Random Forest merupakan algoritma ensemble learning yang terdiri atas kumpulan decision trees yang bekerja secara paralel untuk menghasilkan prediksi yang lebih akurat. Setiap pohon keputusan dibangun menggunakan subset data dan fitur yang dipilih secara acak, sehingga meningkatkan keragaman model dan mengurangi risiko *overfitting*. Proses ini dikenal sebagai teknik *bagging* (*bootstrap aggregating*), di mana masing-masing pohon berkontribusi pada keputusan akhir melalui mekanisme majority voting untuk klasifikasi atau averaging untuk regresi [3]. Parameter utama dalam algoritma ini antara lain jumlah pohon (*n_estimators*), kedalaman maksimum pohon (*max_depth*), serta jumlah fitur yang dipertimbangkan saat membagi simpul. Dengan sifatnya yang robust terhadap data yang kompleks dan kolinearitas antar fitur, *Random Forest* menjadi pilihan yang efektif untuk membangun sistem prediksi berbasis data historis proyek dalam konteks penelitian ini.

Tingkat keberhasilan proyek merupakan indikator pencapaian tujuan proyek berdasarkan dimensi waktu, biaya, ruang lingkup, dan kualitas hasil. Dalam praktik modern, keberhasilan tidak hanya dilihat dari sisi teknis, tetapi juga dari nilai strategis yang dihasilkan bagi organisasi. Untuk mempermudah analisis berbasis data, keberhasilan proyek dapat direpresentasikan sebagai variabel klasifikasi biner, yaitu berhasil atau gagal, yang memungkinkan penerapan algoritma machine learning dalam proses prediksi [4].

Sistem dapat dipahami sebagai suatu kumpulan elemen atau komponen yang saling berkaitan dan bekerja secara bersama-sama untuk mencapai fungsi tertentu. Dalam dunia teknologi informasi, sistem terdiri dari kombinasi antara perangkat keras (*hardware*) dan perangkat lunak (*software*) yang dirancang untuk

menjalankan aplikasi dan memproses data guna mendukung operasional organisasi. Arsitektur sistem umumnya disusun secara hierarkis agar dapat memastikan integritas dan kinerja dalam pelaksanaan tugas-tugas komputasi [5]. Dalam penelitian ini, sistem yang dibangun berfungsi untuk mengelola data work order secara terpusat dan terstruktur, sekaligus menyediakan fitur analitik yang mampu memproses data historis proyek untuk mendukung prediksi keberhasilan.

Work Order (WO) merupakan dokumen resmi yang berisi instruksi untuk melaksanakan pekerjaan tertentu dalam suatu organisasi atau perusahaan. WO biasanya mencakup informasi seperti deskripsi pekerjaan, tanggal pelaksanaan, sumber daya yang dibutuhkan, dan pihak yang bertanggung jawab. Penggunaan WO yang terstruktur membantu perusahaan dalam mengelola pekerjaan secara efisien dan memastikan bahwa setiap tugas diselesaikan sesuai dengan standar yang ditetapkan [6]. Dalam sistem yang dikembangkan, data dari *work order* menjadi sumber utama untuk menghasilkan fitur prediktif yang akan digunakan oleh *algoritma Random Forest*.

Model *Agile* adalah pendekatan fleksibel dan iteratif dalam pengembangan perangkat lunak yang menekankan kolaborasi tim, respons terhadap perubahan, serta pengiriman perangkat lunak secara bertahap dan kontinu. *Agile* berlandaskan pada empat nilai utama yang terkandung dalam Manifesto *Agile*, yaitu individu dan interaksi lebih penting daripada proses dan alat, perangkat lunak yang berfungsi lebih penting daripada dokumentasi yang komprehensif, kolaborasi dengan pelanggan lebih penting daripada negosiasi kontrak, dan respons terhadap perubahan lebih penting daripada mengikuti rencana. Pendekatan ini memungkinkan pengembangan sistem yang lebih adaptif terhadap kebutuhan pengguna dan perubahan pasar [7]. Oleh karena itu, pendekatan *Agile* dipilih dalam penelitian ini agar sistem dapat dikembangkan secara dinamis dan sesuai kebutuhan operasional perusahaan.

Penelitian terkait penerapan algoritma machine learning telah banyak dilakukan dalam berbagai domain dengan hasil yang menunjukkan tingkat akurasi yang tinggi. Penelitian pada klasifikasi tutupan lahan menunjukkan bahwa algoritma *Random Forest* mampu mengungguli metode lain seperti *Decision Tree* dan *Logistic Regression* dalam mengolah data dengan kompleksitas tinggi serta fitur yang beragam [8]. Selain itu, penelitian pada klasifikasi kelayakan air minum juga membandingkan beberapa algoritma seperti *Logistic Regression*, *Decision Tree*, *Random Forest*, dan *Extra Trees*, di mana *Random Forest* memberikan performa terbaik dibandingkan metode lainnya [9].

Lebih lanjut, pada bidang kesehatan, penelitian mengenai klasifikasi penyakit jantung menunjukkan bahwa *Random Forest* memiliki kinerja yang lebih unggul dibandingkan metode *boosting* seperti

XGBoost dalam hal akurasi dan kemampuan mendeteksi data secara tepat [10]. Hal ini menunjukkan bahwa algoritma *Random Forest* memiliki keunggulan dalam menangani berbagai jenis data, baik yang bersifat linier maupun non-linier, serta mampu memberikan hasil prediksi yang stabil dan akurat.

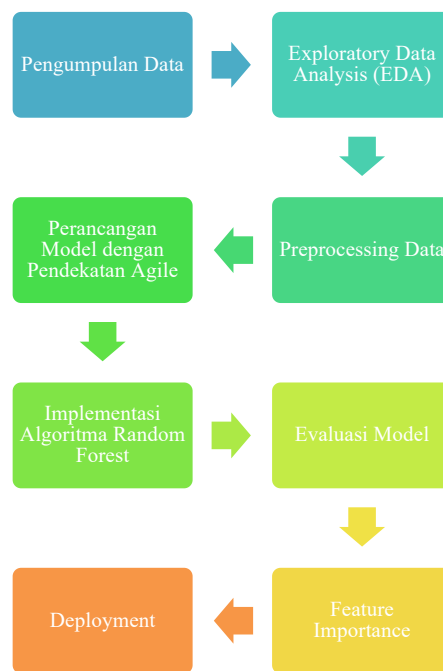
Namun demikian, penelitian-penelitian tersebut masih memiliki keterbatasan. Sebagian besar penelitian hanya berfokus pada pembangunan model prediksi tanpa mengintegrasikannya ke dalam sistem informasi yang dapat digunakan secara langsung dalam proses bisnis. Selain itu, domain yang digunakan masih terbatas pada bidang lingkungan, kesehatan, dan data sensorik, sehingga belum banyak penelitian yang mengkaji penerapan machine learning, khususnya *Random Forest*, dalam konteks manajemen proyek berbasis data *work order*.

Berdasarkan permasalahan tersebut, penelitian ini dilakukan untuk mengisi kesenjangan dengan mengimplementasikan algoritma *Random Forest* dalam sistem monitoring *work order* guna memprediksi tingkat keberhasilan proyek berdasarkan data historis. Dengan integrasi antara model prediksi dan sistem informasi, diharapkan hasil analisis dapat digunakan secara langsung dalam mendukung pengambilan keputusan strategis perusahaan.

Adapun tujuan dari penelitian ini adalah untuk membangun sistem *monitoring work order* berbasis *machine learning* menggunakan algoritma *Random Forest* yang mampu memprediksi tingkat keberhasilan proyek secara akurat, serta membantu perusahaan dalam mengidentifikasi potensi keberhasilan atau kegagalan proyek sejak tahap awal.

2. Metode Penelitian

Penelitian ini mengimplementasikan algoritma *Random Forest* dengan dukungan pendekatan Agile untuk membangun sistem prediksi tingkat keberhasilan proyek pada *Work Order* berbasis *web*. Metode *Random Forest* dipilih karena mampu menangani data tabular dengan baik, tahan terhadap *overfitting*, serta dapat memberikan informasi mengenai pentingnya setiap variabel (*feature importance*) dalam proses klasifikasi. Pendekatan *Agile* digunakan untuk mempercepat proses pengembangan melalui iterasi (*sprint*) yang fleksibel dan berorientasi pada kebutuhan pengguna.



Gambar 1. Tahapan Metode *Design Science Research* (DSR)

Berikut adalah penjelasan dari tahap-tahap yang dilakukan pada *Random Forest* dengan model pengembangan *Agile* pada penelitian ini:

2.1. Pengumpulan Data

Tahap pertama adalah pengumpulan data yang diperoleh dari sistem *Work Order* pada PT XYZ. *Dataset* diambil dari table pada sistem yang saat ini terdiri atas 50 data proyek, mencakup informasi seperti jenis pekerjaan, nilai proyek, Rencana Anggaran Biaya (RAB), margin keuntungan, lama pengerjaan, serta status keberhasilan proyek (berhasil atau gagal). Data ini menjadi dasar utama dalam membangun model prediksi dengan algoritma *Random Forest*.

2.2. *Exploratory Data Analysis* (EDA)

Setelah data terkumpul, dilakukan proses *Exploratory Data Analysis* (EDA) untuk memahami karakteristik data. Analisis ini mencakup pemeriksaan distribusi nilai pada setiap variabel, identifikasi data ekstrim (*outlier*), hubungan antar variabel, serta proporsi kelas target. EDA memberikan gambaran awal tentang kualitas data sekaligus membantu dalam menentukan strategi *preprocessing* dan pemodelan yang tepat.

Exploratory Data Analysis

```

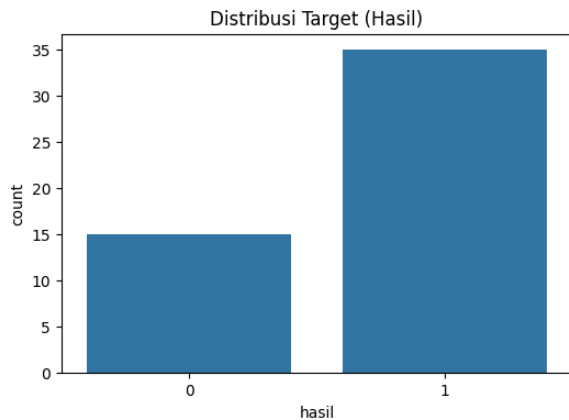
plt.figure(figsize=(6,4))
sns.countplot(x='hasil', data=df)
plt.title("Distribusi Target (Hasil)")
plt.show()

# Preprocessing before correlation heatmap
# Encode fitur kategorikal
le = LabelEncoder()
df['jenis_pekerjaan'] = le.fit_transform(df['jenis_pekerjaan'])

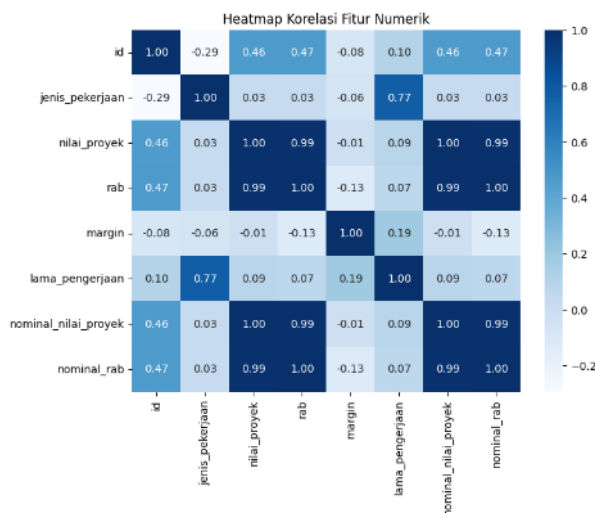
# Korelasi antar fitur numerik (after encoding)
plt.figure(figsize=(8,6))
# Select only numeric columns for correlation
  
```

```
numeric_cols = df.select_dtypes(include=np.number).columns.tolist()
# Ensure 'hasil' is not included in correlation matrix
if 'hasil' in numeric_cols:
    numeric_cols.remove('hasil')
sns.heatmap(df[numeric_cols].corr(),
            annot=True, cmap="Blues", fmt=".2f")
plt.title("Heatmap Korelasi Fitur Numerik")
plt.show()
```

```
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)
# Split data 80:20
X_train, X_test, y_train, y_test = train_test_split(X_scaled, y, test_size=0.2,
                                                    random_state=42, stratify=y)
print("\nJumlah Data Train:", len(X_train))
print("Jumlah Data Test:", len(X_test))
```



Gambar 2. Distribusi Target Hasil)



Gambar 3. Heatmap Korelasi Fitur Numerik)

2.3. Preprocessing Data

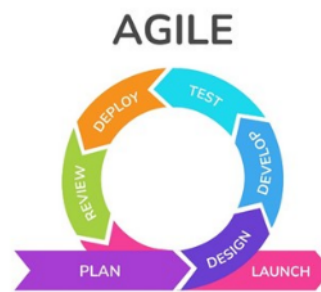
Tahap *preprocessing* dilakukan agar data siap digunakan untuk pelatihan model. Proses ini meliputi pembersihan data dari nilai kosong atau tidak konsisten, transformasi variabel kategorikal menjadi numerik menggunakan *encoding*, serta perhitungan margin jika belum tersedia. Selain itu, dilakukan normalisasi atau standarisasi terhadap fitur numerik untuk meningkatkan performa model dalam proses klasifikasi.

Preprocessing

```
x = df.drop('hasil', axis=1)
y = df['hasil']
# Standarisasi fitur numerik
```

2.4. Perancangan Model dengan Pendekatan Agile

Dalam penelitian ini digunakan pendekatan *Agile* untuk merancang dan mengembangkan sistem prediksi. Proses pengembangan dilakukan secara iteratif melalui sprint, di mana setiap siklus menghasilkan artefak yang dapat dievaluasi. Dengan metode ini, pengembangan sistem menjadi lebih adaptif terhadap perubahan kebutuhan pengguna dan memungkinkan adanya penyempurnaan berkelanjutan.



Gambar 4. Metode Agile

Penelitian ini menerapkan model Agile yang terdiri dari:

- *Planning* → Menentukan kebutuhan data dan tujuan model.
- *Design* → Merancang alur *preprocessing* dan algoritma *Random Forest*.
- *Development* → Implementasi model menggunakan *scikit-learn*.
- *Testing* → Menguji performa model pada data uji.
- *Review* → Melakukan analisis hasil evaluasi untuk perbaikan.

2.5. Implementasi Algoritma Random Forest

Tahap berikutnya adalah implementasi algoritma *Random Forest* sebagai metode klasifikasi. *Random Forest* dipilih karena keunggulannya dalam menangani data tabular, ketahanannya terhadap *overfitting*, serta kemampuannya memberikan interpretasi melalui *feature importance*. Model dilatih menggunakan dataset latih (*train set*) dengan variabel prediktor meliputi jenis pekerjaan, nilai proyek, RAB, margin keuntungan, dan lama pengerjaan untuk memprediksi status keberhasilan proyek.

Implementasi algoritma Random Forest

```
model = RandomForestClassifier(
    n_estimators=200,
    max_depth=None,
    max_features='sqrt',
    min_samples_leaf=1,
    min_samples_split=5,
    random_state=42)
```

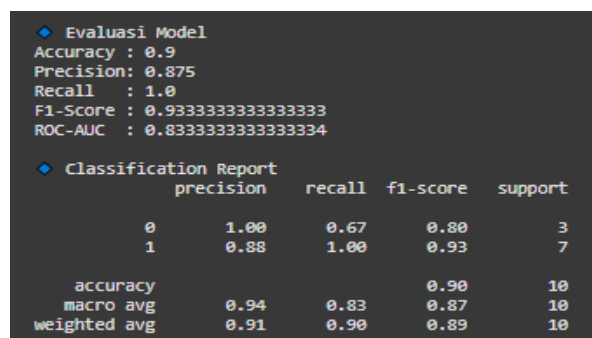
```
)
model.fit(x_train, y_train)
```

2.6. Evaluasi Model

Evaluasi model dilakukan untuk menilai kinerja algoritma *Random Forest*. *Dataset* dibagi menjadi data latih dan data uji dengan proporsi tertentu (misalnya 80:20). Model kemudian diuji menggunakan data uji, dan hasil prediksi dievaluasi menggunakan metrik seperti *accuracy*, *precision*, *recall*, *F1-score*, *confusion matrix*. Evaluasi ini bertujuan memastikan bahwa model mampu melakukan prediksi dengan tingkat akurasi yang baik.

Evaluasi Model

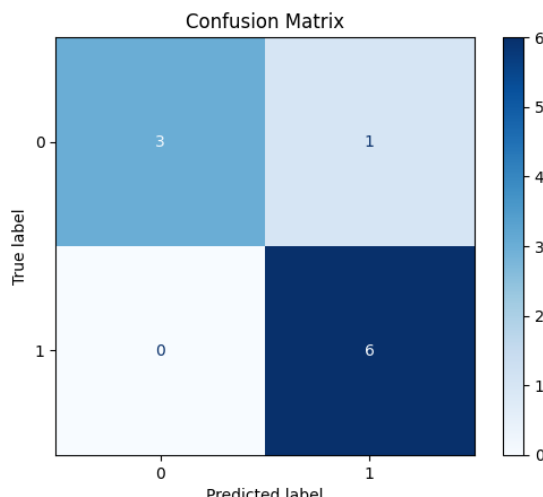
```
y_pred = rf.predict(x_test)
print("\n Evaluasi Model")
print("Accuracy :", accuracy_score(y_test,
y_pred))
print("Precision:", precision_score(y_test,
y_pred))
print("Recall :", recall_score(y_test,
y_pred))
print("F1-score :", f1_score(y_test, y_pred))
print("ROC-AUC :", roc_auc_score(y_test,
y_pred))
print("\n Classification Report")
print(classification_report(y_test, y_pred))
```



Gambar 5. Evaluasi Model & Classification Report

Confusion Matrix

```
cm = confusion_matrix(y_test, y_pred)
disp
ConfusionMatrixDisplay(confusion_matrix=cm) =
disp.plot(cmap='Blues')
plt.title("Confusion Matrix")
plt.show()
```



Gambar 6. Confusion Matrix

Dari tabel tersebut, diperoleh nilai sebagai berikut:

- True Positive (TP) = 6 → Prediksi berhasil, aktual berhasil
- True Negative (TN) = 3 → Prediksi gagal, aktual gagal
- False Positive (FP) = 1 → Prediksi berhasil, aktual gagal
- False Negative (FN) = 0 → Prediksi gagal, aktual berhasil

Kemudian, metrik evaluasi model dihitung secara manual menggunakan rumus sebagai berikut:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} = \frac{6+3}{6+3+1+0} = \frac{9}{10} = 0,9 \quad (1)$$

$$Precision = \frac{TP}{TP+FP} = \frac{6}{6+1} = \frac{6}{7} \approx 0.857 \quad (2)$$

$$Recall = \frac{TP}{TP+FN} = \frac{6}{6+0} = 1.0 \quad (3)$$

$$F1 - Score = \frac{2 \times P \times R}{P + R} = \frac{2 \times 0.8571 \times 1.0}{0.8571 + 1.0} \approx 0.9230 \quad (4)$$

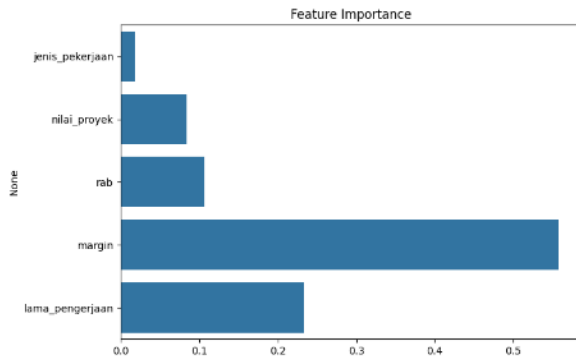
Hasil evaluasi menunjukkan bahwa model *Random Forest* memiliki performa klasifikasi yang sangat baik, dengan nilai akurasi mencapai 90% dan *F1-score* di atas 0,92. Hal ini mengindikasikan bahwa model mampu memprediksi keberhasilan proyek dengan tingkat presisi dan sensitivitas yang tinggi, serta keseimbangan yang optimal antara keduanya.

2.7. Feature Importance & ROC Curve

Feature importance untuk mengetahui kontribusi tiap variabel dalam prediksi. Model juga divalidasi menggunakan *ROC Curve* dan *AUC (Area Under Curve)* guna mengukur kemampuan model dalam membedakan antara kelas berhasil dan gagal.

Feature Importance

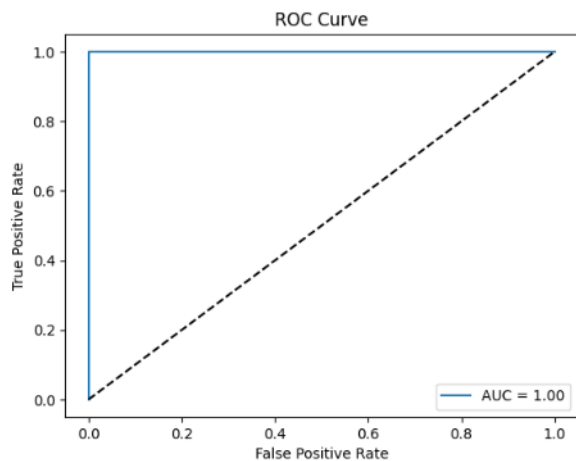
```
plt.figure(figsize=(8, 5))
sns.barplot(x=model.feature_importances_,
y=X.columns)
plt.title("Feature Importance")
plt.tight_layout()
plt.show()
```



Gambar 7. Feature Importance

ROC Curve

```
fpr, tpr, _ = roc_curve(y_test, y_proba)
roc_auc = auc(fpr, tpr)
plt.plot(fpr, tpr, label=f"AUC = {roc_auc:.2f}")
plt.plot([0, 1], [0, 1], 'k--')
plt.xlabel("False Positive Rate")
plt.ylabel("True Positive Rate")
plt.title("ROC Curve")
plt.legend()plt.show()
```



Gambar 8. ROC Curve

2.8. Deployment (Implementasi ke Web)

Tahap terakhir adalah *deployment* atau implementasi model ke dalam sistem berbasis *web*. Proses ini dilakukan agar hasil penelitian dapat digunakan secara praktis oleh pengguna dalam mendukung pengambilan keputusan pada *Work Order*. Model *Random Forest* yang telah dilatih diintegrasikan ke dalam aplikasi web sehingga dapat melakukan prediksi secara langsung terhadap data proyek baru.

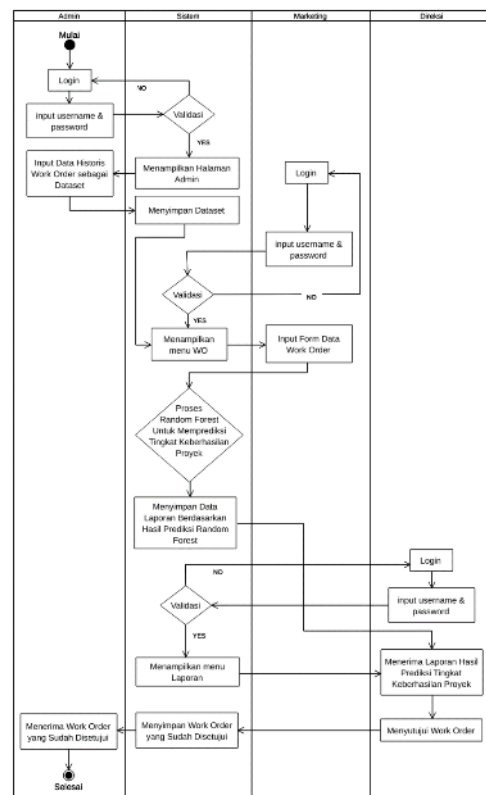
3. Hasil dan Pembahasan

Analisis sistem merupakan proses untuk memecah suatu sistem informasi menjadi komponen-komponen yang lebih kecil dengan tujuan untuk mengidentifikasi permasalahan, peluang, dan hambatan yang ada. Proses ini bertujuan untuk memahami kebutuhan spesifik yang harus dipenuhi untuk meningkatkan efisiensi dan efektivitas sistem. Hasil dari analisis sistem ini akan

menjadi dasar dalam merancang perbaikan yang sesuai dengan kebutuhan organisasi atau pengguna [11].

3.1. Analisa Sistem Usulan

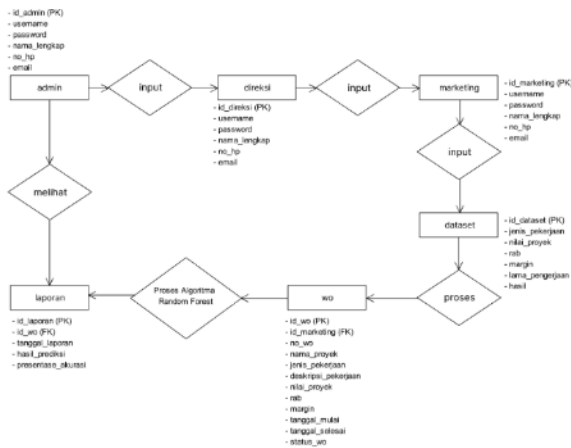
Analisis sistem yang diusulkan menggambarkan proses digital yang terstruktur dalam pengelolaan proyek berbasis *Work Order* (WO), dengan penerapan algoritma machine learning untuk memperkirakan tingkat keberhasilan proyek. Sistem ini dirancang untuk menjawab permasalahan evaluasi proyek yang selama ini masih dilakukan secara manual dan subjektif, serta belum memanfaatkan data historis secara optimal. Dengan mengintegrasikan metode prediksi menggunakan algoritma *Random Forest*, sistem mampu memberikan penilaian yang lebih objektif dan berbasis data terhadap potensi keberhasilan proyek sebelum proyek dijalankan.



Gambar 9. Analisa Sistem Usulan

3.2. Entity Relationship Diagram (ERD)

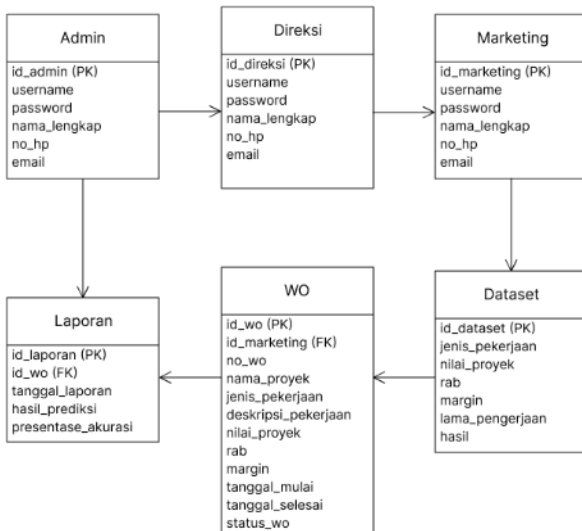
Entity Relationship Diagram (ERD) merupakan gambaran visual yang merepresentasikan keterkaitan antar entitas dalam suatu sistem atau basis data. Diagram ini memperlihatkan struktur entitas, atribut yang dimiliki, serta tingkat hubungan (kardinalitas) di dalam lingkungan basis data relasional [12].



Gambar 10. Entity Relationship Diagram (ERD)

3.3. Logical Relationship Structure (LRS)

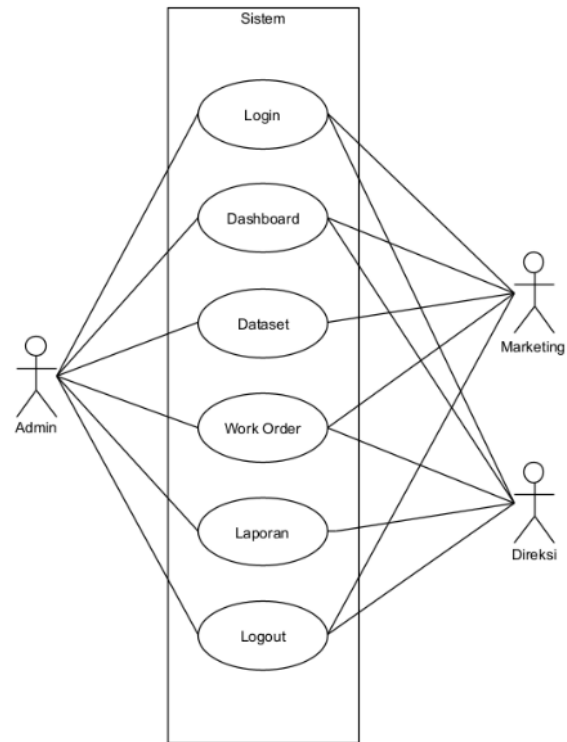
Logical Relational Structure (LRS) adalah hasil dari transformasi ERD yang mencakup entitas dan atribut terkait. LRS menyusun hubungan antar entitas dengan lebih jelas dan sistematis sesuai dengan persyaratan database, sehingga memudahkan implementasi desain basis data dalam bentuk tabel relasional yang saling terhubung [13].



Gambar 11. Logical Relationship Structure (LRS)

3.4. Use Case Diagram

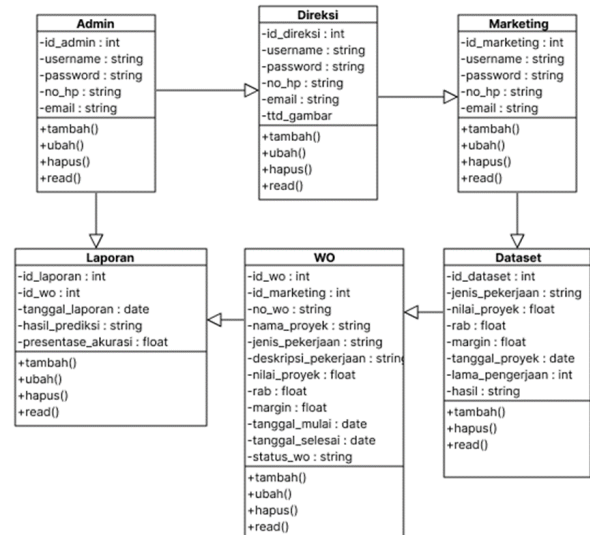
Diagram use case merupakan representasi visual yang memperlihatkan kemungkinan interaksi antara sistem dan pengguna. Diagram ini menampilkan skenario-skenario penggunaan (use case) beserta peran atau aktor yang terlibat, sehingga membantu dalam memahami fitur-fitur utama yang harus dimiliki oleh [14].



Gambar 12. Use Case Diagram

3.5. Class Diagram

Class Diagram merupakan representasi struktural yang menampilkan kelas, paket, dan objek dalam suatu sistem beserta relasi di antara elemen-elemen tersebut. Diagram ini menjelaskan spesifikasi masing-masing kelas yang mencakup atribut serta fungsionalitasnya.

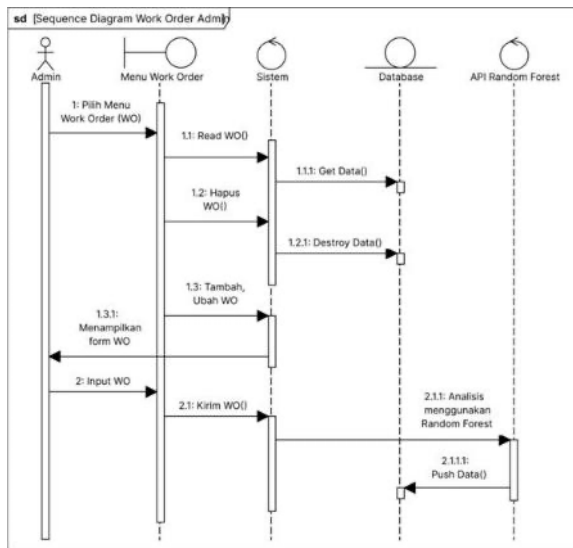


Gambar 13. Class Diagram

3.6. Sequence Diagram

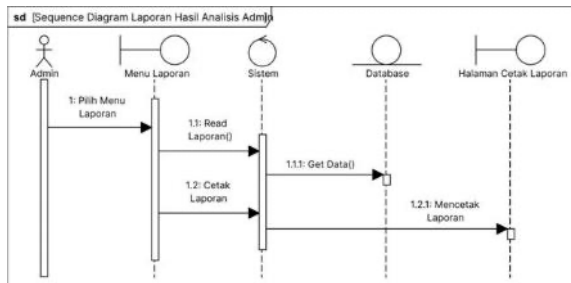
Sequence Diagram menggambarkan urutan interaksi antar aktor dan komponen sistem dalam menjalankan suatu proses. Diagram ini menunjukkan aliran komunikasi antara admin, sistem, database, serta API

dalam pengelolaan *Work Order* dan pembuatan laporan.



Gambar 14. *Sequence Diagram Work Order*

Sequence Diagram ini menggambarkan proses pengelolaan *Work Order* (WO) oleh *admin*, dimulai dari pemilihan menu, pengambilan data, hingga proses tambah, ubah, dan hapus data WO. Selain itu, sistem juga melakukan analisis menggunakan algoritma *Random Forest* melalui API untuk memproses data yang telah diinput dan menghasilkan prediksi tingkat keberhasilan proyek.



Gambar 15. *Sequence Diagram Laporan*

Sequence Diagram ini menggambarkan proses pembuatan laporan hasil analisis. Admin memilih menu laporan, kemudian sistem mengambil data dari database dan menampilkan hasil analisis yang selanjutnya dapat dicetak dalam bentuk laporan.

3.7. Pengujian

Tahap pengujian dilakukan untuk memastikan bahwa model *Random Forest* dan aplikasi *web* yang dikembangkan dapat berfungsi sesuai dengan tujuan penelitian, yaitu memprediksi tingkat keberhasilan proyek secara akurat, praktis, dan *real-time*. Pengujian dilakukan melalui beberapa pendekatan yang mencakup pengujian model, pengujian fitur, serta pengujian aplikasi dan penerimaan pengguna.

Pengujian akurasi model dilakukan dengan membagi dataset ke dalam data latih (*training*) dan data uji (*testing*). Model *Random Forest* dievaluasi

menggunakan beberapa metrik, antara lain confusion matrix untuk mengetahui jumlah prediksi benar dan salah pada masing-masing kelas proyek berhasil dan gagal, nilai akurasi untuk mengukur tingkat ketepatan model secara keseluruhan, serta precision, recall, dan *F1-score* untuk menilai kualitas prediksi pada setiap kelas. Selain itu, performa model juga dianalisis menggunakan *ROC curve* dan nilai *Area Under the Curve (AUC)* guna mengevaluasi kemampuan model dalam membedakan antara proyek yang berhasil dan tidak berhasil.

Selanjutnya, pengujian feature importance dilakukan untuk mengidentifikasi variabel yang paling berpengaruh terhadap tingkat keberhasilan proyek, seperti nilai proyek, RAB, lama pengerjaan, jenis pekerjaan, dan margin. Hasil pengujian ini memberikan informasi yang bermanfaat bagi manajemen dalam memahami faktor-faktor kunci yang memengaruhi keberhasilan proyek.

Pengujian fungsional aplikasi web dilakukan menggunakan metode *black box testing* dengan melibatkan administrator server PT. XYZ sebagai pengguna langsung sistem. Pengujian ini bertujuan untuk memastikan seluruh fungsi utama aplikasi berjalan dengan baik, mulai dari proses unggah dataset proyek dalam format CSV, proses prediksi proyek baru yang menampilkan hasil klasifikasi beserta nilai probabilitasnya, hingga penyajian visualisasi data seperti grafik *feature importance*, *confusion matrix*, dan *ROC curve*. Selain itu, sistem juga diuji dalam menyimpan dan menampilkan laporan serta riwayat hasil prediksi proyek.

Tahap pengujian selanjutnya adalah *User Acceptance Testing (UAT)* yang melibatkan pengguna langsung, seperti manajer proyek atau staf PT XYZ. Pada tahap ini, pengguna mengevaluasi sistem berdasarkan kemudahan penggunaan, kecepatan sistem dalam menghasilkan prediksi, kesesuaian hasil prediksi dengan kondisi di lapangan, serta kualitas tampilan visualisasi dan laporan yang dihasilkan.

Seluruh hasil pengujian kemudian didokumentasikan secara sistematis dalam bentuk tabel dan laporan pengujian untuk mendukung analisis dan kesimpulan penelitian.

Tabel 1. Hasil Pengujian

Fitur yang Diuji	Hasil yang Diharapkan	Hasil Aktual	Status
Upload dataset CSV	Data terbaca di sistem	Sesuai	Valid
Prediksi proyek baru	Sistem menampilkan hasil prediksi dan probabilitas	Sesuai	Valid
Visualisasi <i>confusion matrix</i>	Grafik tampil semua hasil evaluasi model	Sesuai	Valid
Laporan hasil prediksi	Hasil prediksi tersimpan ke database	Sesuai	Valid

3.8. Implementasi Model ke Aplikasi *Web*

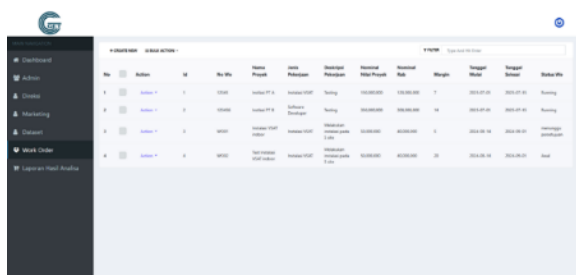
Implementasi model *Random Forest* ke dalam aplikasi web merupakan tahap penerapan hasil pemodelan yang sebelumnya dibangun dan diuji. Pada tahap ini, model *Random Forest* yang telah dilatih dengan data historis *Work Order* diintegrasikan ke dalam sistem berbasis web agar dapat digunakan secara langsung oleh pengguna. Proses implementasi meliputi penerjemahan hasil pelatihan model ke dalam kode program, integrasi dengan database proyek, serta pembuatan antarmuka web yang interaktif.

Dengan implementasi ini, model *Random Forest* yang awalnya hanya berupa perhitungan dan simulasi di lingkungan pengembangan dapat dimanfaatkan secara nyata. Pengguna dapat mengakses sistem untuk melakukan prediksi tingkat keberhasilan proyek baru secara praktis, cepat, dan *real-time*, sehingga mendukung pengambilan keputusan pada manajemen proyek di PT XYZ.



Gambar 16. Halaman *Dashboard*

Halaman *dashboard* menampilkan ringkasan informasi terkait *Work Order*, seperti jumlah proyek berhasil, gagal, disetujui, dan ditolak. Informasi ini disajikan dalam bentuk visual sehingga memudahkan pengguna dalam memantau kondisi proyek secara keseluruhan.



Gambar 17. Halaman *Work Order*

Halaman *Work Order* digunakan untuk mengelola data proyek, meliputi penambahan, perubahan, dan penghapusan data. Selain itu, halaman ini juga menampilkan informasi detail proyek seperti nama proyek, jenis pekerjaan, nilai anggaran, margin, serta status *Work Order* yang sedang berjalan.



Gambar 18. Halaman Laporan Hasil Analisis

Halaman laporan hasil analisis menampilkan output prediksi dari algoritma *Random Forest* terhadap tingkat keberhasilan proyek. Informasi yang ditampilkan meliputi hasil prediksi serta tingkat akurasi, sehingga pengguna dapat mengevaluasi potensi keberhasilan proyek berdasarkan data yang tersedia.

4. Kesimpulan

Berdasarkan hasil penelitian mengenai implementasi algoritma *Random Forest* untuk menentukan tingkat keberhasilan proyek pada sistem *work order* dengan model *Agile* di PT XYZ, dapat disimpulkan bahwa sistem yang dirancang mampu menggantikan proses manual dan subjektif dalam mengevaluasi keberhasilan proyek dengan pendekatan yang lebih objektif dan berbasis data. Melalui pemanfaatan data historis *work order*, sistem ini dapat mengolah berbagai parameter proyek seperti durasi, anggaran, dan margin keuntungan untuk menghasilkan prediksi yang lebih akurat dan dapat diandalkan. Implementasi algoritma *Random Forest* terbukti mampu membangun model klasifikasi yang efektif dalam mengidentifikasi potensi keberhasilan proyek sejak tahap awal, sehingga memberikan nilai tambah dalam proses evaluasi proyek.

Secara praktis, hasil penelitian ini memberikan dampak signifikan terhadap proses pengambilan keputusan di PT XYZ, di mana manajemen dapat menggunakan hasil prediksi sebagai dasar dalam menentukan prioritas proyek, mengurangi risiko kerugian, serta meningkatkan efisiensi pengelolaan sumber daya. Selain itu, penerapan model dalam sistem berbasis web memungkinkan proses analisis dilakukan secara lebih cepat dan terintegrasi dalam alur kerja perusahaan.

Untuk pengembangan selanjutnya, sistem ini dapat ditingkatkan dengan mengintegrasikan model prediksi ke dalam sistem *real-time* agar mampu memberikan analisis secara langsung terhadap data proyek yang terus diperbarui. Selain itu, pengembangan juga dapat dilakukan dengan menambahkan variasi algoritma pembandingan atau melakukan optimasi parameter model untuk meningkatkan akurasi prediksi, sehingga sistem dapat memberikan hasil yang lebih optimal dan adaptif terhadap perubahan kondisi proyek di masa mendatang.

Ucapan Terimakasih

Penulis mengucapkan terima kasih yang sebesar-besarnya kepada Allah SWT, kedua orang tua, saudara,

dan terkhusus kepada dosen pembimbing, serta seluruh dosen prodi Teknik Informatika Universitas Pamulang dan sahabat-sahabat penulis.

Daftar Rujukan

- [1] I. Magdalena, A. Salsabila, D. A. Krianasari, and S. F. Apsarini, "Implementasi Model Pembelajaran Daring Pada Masa Pandemi Covid-19 Di Kelas Iii Sdn Sindangsari Iii," *J. Pendidik. dan Dakwah*, vol. 3, no. 1, p. 120, 2021, [Online]. Available: <https://ejournal.stitpn.ac.id/index.php/pandawa>
- [2] S. T. M. K. Indah Purnama Sari, *Algoritma dan Pemrograman*. umsu press, 2024. [Online]. Available: <https://books.google.co.id/books?id=UGI3EQAAQBAJ>
- [3] R. F. Putra *et al.*, *Algoritma Pembelajaran Mesin: Dasar, Teknik, dan Aplikasi*. PT. Sonpedia Publishing Indonesia, 2024. [Online]. Available: <https://books.google.co.id/books?id=MX7-EAAAQBAJ>
- [4] M.-W. Hsu, N. Dacre, and P. Senyo, "Applied Algorithmic Machine Learning for Intelligent Project Prediction: Towards an AI Framework of Project Success," *SSRN Electron. J.*, no. April, 2021, doi: 10.2139/ssrn.3823900.
- [5] J. M. Fernandes, *Essentials of computing systems*. 2022. doi: 10.21814/uminho.ed.93.
- [6] Hariyanto, Jenica Noor Lasmaji, and Marini, "Rancang Bangun Sistem Informasi Entry Work Order Pada PT.Hi-lex Indonesia," *J. Ilm. Sains Teknol. Dan Inf.*, vol. 2, no. 1, pp. 1–8, 2024, doi: 10.59024/jiti.v2i1.541.
- [7] K. Fajri, A. Saputra, Z. Umar, and I. Albana, "Analisis Pendekatan Metode Agile Dalam Manajemen Proyek Pada Sistem Informasi," *J. Mhs. Sist. Inf.*, vol. 6, no. 1, pp. 74–83, 2024, doi: 10.24127/jmsi.v6i1.7631.
- [8] J. Fasilkom, "Peningkatan Akurasi Klasifikasi Tutupan Lahan Menggunakan Random Forest pada Data Sentinel-2 di Jambi Author : Akhiyar Waladi," vol. 15, no. 1, pp. 17–24, 2025. <https://ejournal.umri.ac.id/index.php/JIK/article/view/8886/3478>
- [9] U. R. Zalti, D. R. Darmakusuma, M. Ridwansyah, E. Ismanto, and E. Ismanto, "Analisis dan Prediksi Kelayakan Air Minum Menggunakan Algoritma Random Forest," vol. 15, no. 2, pp. 312–317, 2025. <https://ejournal.umri.ac.id/index.php/JIK/article/view/9906/3794>
- [10] C. Arafat *et al.*, "Perbandingan Algoritma Random Forest Dan Xgboost Untuk Klasifikasi Penyakit Jantung Berdasarkan Data Medis," vol. 15, no. 2, pp. 430–435, 2025. <https://ejournal.umri.ac.id/index.php/JIK/article/view/9927/3810>
- [11] R. E. Indrajit, "Konsep Sistem Informasi," *J. Adm. Pendidik.*, vol. 3, pp. 1–14, 2015, [Online]. Available: <https://ejournal.upi.edu/index.php/JAPSPs/article/viewFile/6095/4116>
- [12] S. S. Bagui and R. W. Earp, *Database Design Using Entity-Relationship Diagrams*. CRC Press, 2022. [Online]. Available: <https://books.google.co.id/books?id=5qJ5EAAAQBAJ>
- [13] A. Priadi and M. Feizal, "Rancang Bangun Sistem Informasi Perpustakaan Digital Berbasis Web Pada Sekolah Dasar (SD) Negeri Paku Jaya 2," *OKTAL J. Ilmu Komput. dan Sci.*, vol. 1, no. 8, pp. 1155–1164, 2022, [Online]. Available: <https://journal.mediapublikasi.id/index.php/oktal>
- [14] B. John, "An Analysis of the Significance of UML Diagrams Focus on Medium-Sized Projects," no. February, pp. 1–5, 2025, [Online]. Available: <https://www.researchgate.net/publication/388683775>