iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** XI **Month of publication:** November 2026

DOI: https://doi.org/10.22214/ijraset.2026.85488

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue XI Nov 2026- Available at www.ijraset.com

LifeLink: An AI-Assisted, Explainable Organ Transplant Matching and Smart Priority-Queue Platform

Varun Kumar R[1], Vinay K[2], Chethan Gowda D S[3], Karthik R[4], Mr. Harish Kumar M[5]
Department of Artificial Intelligence & Machine Learning, Vemana Institute of Technology, Bengaluru, India

Abstract: ***Organ transplantation systems must decide who is treated first, which donor organ suits which patient, and how those decisions can be checked afterwards. In practice, government and private hospitals run separate processes, urgency claims are hard to verify, and automated ranking tools rarely explain their output. This paper presents LifeLink, a web platform in which patients choose a government priority-queue route or an urgent private route within one system. Medical urgency is certified by a doctor and, for private cases and large urgency changes, countersigned by a hospital administrator. Donor organs are ranked by a two-stage engine: hard medical filters remove impossible matches, and a trained logistic-regression model scores the rest, with an exact per-factor explanation for every score. Every status change is written to an audit log, and the platform continues through appointment, transplant completion, and an aftercare plan. A simulation study on synthetic data compares three queue policies and the model ranking against random and best-possible choices. A companion mobile application is the next development stage. The system is a prototype trained and evaluated on synthetic data; it is not a clinical system.***

Index Terms: ***Organ Transplantation, Explainable AI, Matching System, Priority Queue, Healthcare Informatics, Django, Logistic Regression, Simulation.***

I. INTRODUCTION

Organ transplantation is limited by supply. The number of patients waiting is far larger than the number of donor organs, and a donor organ remains usable for only a short time once it has been retrieved [1]. Allocation therefore has to be fast, fair, and open to review. National systems rely on structured allocation policies that weigh medical urgency, compatibility, and waiting time [2], and in India the legal basis is the Transplantation of Human Organs Act, 1994 [3]. At the hospital level the situation is less organised. Patients must contact hospitals individually, government and private hospitals operate separate processes, and a patient with a worsening condition has no simple, verifiable way to request faster treatment. Where software is used to rank organ matches, the ranking is often a black box, which is difficult to accept in a decision that affects survival [4]. LifeLink addresses these gaps in one platform. A patient registers once, chooses between the government queue and a private urgent route, and follows the case through to transplant and aftercare. Doctors certify urgency, hospital administrators countersign important decisions, and an AI module ranks compatible organs while showing exactly why each one scored as it did.

A. Objectives

- Combine government and private hospitals in a single patient-facing platform.
- Provide an urgency-ordered government queue that needs no payment.
- Verify urgency through doctor certification and hospital countersignature, with supporting documents fingerprinted.
- Rank donor organs with a trained model whose every score is explainable.
- Record every status change in an audit log.
- Support appointment scheduling, transplant completion, and an aftercare plan.
- Test the design choices through simulation, and prepare a mobile application as the next stage.

B. Existing System

Most hospitals use paper or spreadsheet waiting lists, telephone coordination, and manual urgency judgements. Patients who cannot visit several hospitals are at a disadvantage, and there is little transparency about why one patient is offered an organ before another. Where private fast-track options exist, the evidence for urgency is usually informal. Automated matching tools, when used, typically output a ranking without an explanation, so clinicians cannot check it.

©IJRASET: All Rights are Reserved

C. Proposed System

LifeLink provides one portal for four roles: patient, doctor, hospital administrator, and government administrator. The government route places the patient in a priority queue ordered by doctor-assessed urgency. The private route requires a doctor's urgency certification with an uploaded supporting document, a countersignature from a hospital administrator, and a payment step, after which matching is unlocked. Matching combines hard medical filters with a trained model, and the platform then handles hospital confirmation, scheduling, completion, and aftercare.

D. Scope

The work covers the backend services, the web interface, the matching model, a fairness simulation, and the design of a mobile client. It is a prototype. Payment is simulated, HLA matching uses a placeholder value, and all training data are synthetic (Section VI). It does not replace national allocation systems or clinical judgement; it is intended as a communication and decision-support layer that those systems could adopt.

II. LITERATURE REVIEW

This section reviews the policy and technical background that shapes the design: national allocation rules, the legal framework in India, machine-learning outcome prediction, and the case for interpretable models.

A. National Allocation Policies

Allocation in the United States is governed by policies that combine waiting time, medical urgency, and compatibility. The 2014 change to deceased-donor kidney allocation, for example, was designed using simulation to study its likely effect on outcomes [2]. This is the same approach followed here: design choices are tested in simulation before being adopted.

B. Legal Framework in India

The Transplantation of Human Organs Act, 1994, regulates organ donation and transplantation in India and prohibits commercial dealings in organs [3]. This matters for a platform that includes a private route: the payment in LifeLink is for hospital services only, never for an organ or for queue position, and an urgent private case still needs medical certification and administrative countersignature.

C. Machine Learning for Outcome Prediction

Statistical and machine-learning models are widely used to estimate the likelihood of a successful outcome from donor, recipient, and logistic factors. Logistic regression remains a standard, well-understood choice for binary outcomes [5], and it is available in mature open-source libraries [6].

D. Interpretability in High-Stakes Decisions

Rudin argues that for high-stakes decisions, inherently interpretable models should be preferred over black-box models explained after the fact [4]. Post-hoc methods such as SHAP approximate each feature's contribution to a prediction [7]. A logistic-regression model needs no approximation: its score is a weighted sum, so each feature's contribution is exact. LifeLink uses this property.

E. Gaps Identified

Table I summarises the approaches reviewed. They are qualitative descriptions, not measurements. Existing approaches usually solve one part of the problem: allocation rules without a patient-facing workflow, prediction models without explanation, or hospital tools without a verifiable urgency process. LifeLink combines them.

Comparative Analysis of Approaches

Approach	Strength	Limitation	LifeLink step
Manual hospital waiting list	Simple	Opaque, slow	Digital urgency queue
National allocation policy	Fair rules at scale	Not patient-facing	Patient portal on top
Black-box ML ranking	Accurate	Unexplained	Exact explanations
Informal private fast-track	Fast	Unverified urgency	Two-person certification
LifeLink	All of the above	Synthetic data	—

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

III. TOOLS AND TECHNOLOGIES

The tools were chosen for availability, maintainability, and ease of integration in a prototype.

A. Backend

Django and Django REST Framework provide models, serializers, and a JSON API [8]. SimpleJWT issues JSON Web Tokens for stateless authentication [9]. Role-based permissions limit each endpoint to the roles that need it. SQLite is used for local development; the schema is portable to a production relational database.

B. Machine Learning

scikit-learn supplies the preprocessing pipeline (standardisation and one-hot encoding), the logistic-regression model, and a random-forest baseline [6]. The trained pipeline is stored as a single artifact and loaded by the matching service.

C. Frontend and Mobile

The web interface is a single-page application in plain HTML, CSS, and JavaScript that calls the REST API. Because the API is JSON and token-based, a mobile client can use the same endpoints without changes to the server.

Hardware and Software Requirements

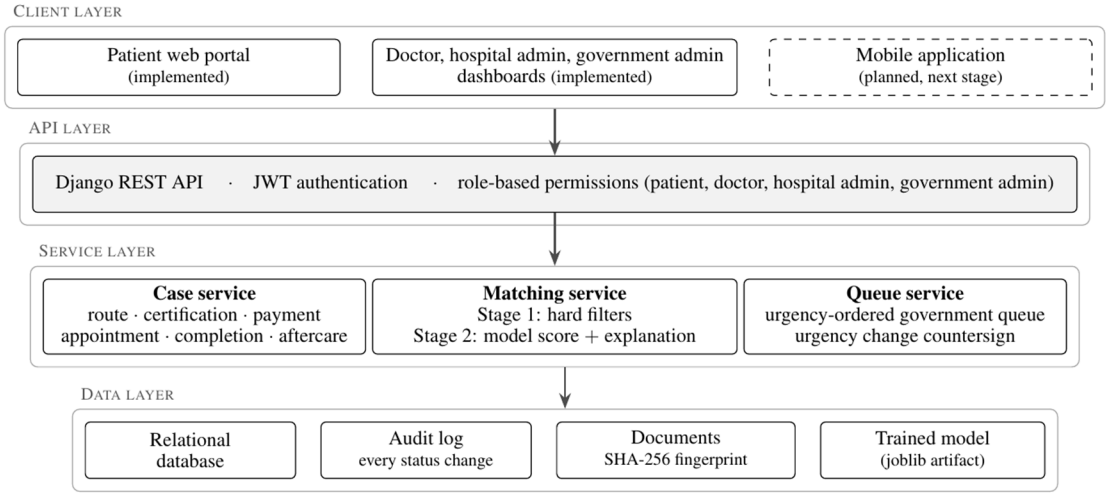
Category	Requirements
Backend	Python, Django, Django REST Framework, SimpleJWT
Database	SQLite (development), relational database (production)
AI / ML	scikit-learn (logistic regression, random forest), NumPy, pandas, joblib
Frontend	HTML, CSS, JavaScript (single-page web app)
Mobile	Planned: Flutter or React Native client using the same REST API
Security	JWT, role-based access control, SHA-256 document hashing
Development	VS Code, Git, Postman

IV. METHODOLOGY

The design separates identity, case workflow, matching, and record-keeping so that each can be developed and tested independently.

A. System Architecture

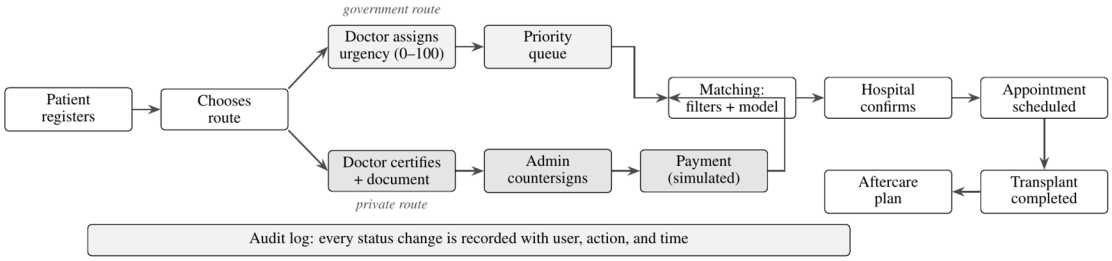
Fig. 1 shows four layers. Client applications—the web portal and, in the next stage, a mobile app—call a REST API protected by JWT authentication and role-based permissions. Behind it, a case service handles routes, certification, payment, appointments, and completion; a matching service filters and ranks organs; and a queue service orders government-route patients. A data layer holds the relational database, the audit log, uploaded documents with their SHA-256 hashes, and the trained model artifact.



System architecture of LifeLink. The dashed box marks the mobile client, which is planned and not yet built.

B. Case Workflow

Fig. 2 traces a case. After registering, the patient chooses a route. On the *government route*, a doctor assigns a medical urgency score from 0 to 100, and the patient joins a queue ordered by urgency and then by joining time. On the *private route*, a doctor certifies urgency with a justification and a signed, sealed document tied to the patient; the file is hashed with SHA-256 so the same document cannot be silently reused for another patient. A hospital administrator then countersigns, the patient makes a (simulated) payment, and matching is unlocked. Both routes continue to matching, hospital confirmation, appointment scheduling, completion, and an aftercare plan. Each status transition is written to the audit log.



Case workflow. Both routes meet at the matching stage and continue to completion and aftercare.

C. Two-Stage Matching

Stage 1: hard filters. A donor organ is a candidate only if (i) its type matches the organ the patient needs, (ii) the donor and recipient blood groups are ABO-compatible, (iii) the organ is still available, (iv) it is within its viability window, and (v) the donor hospital is within 500 km. Organs that fail are not scored. Each exclusion is kept with a plain-language reason, so a "why not this organ?" view can show the clinician what was removed and why.

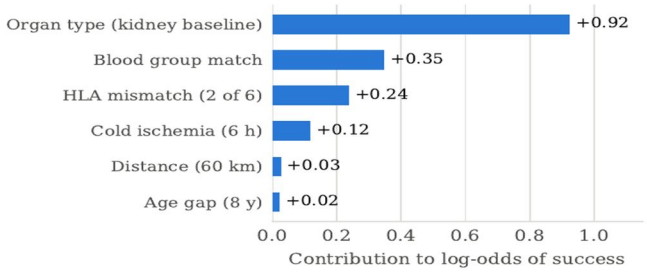
Stage 2: model ranking. Each remaining candidate is described by six features: exact blood-group match, HLA mismatch count (0–6), cold-ischemia time, donor–recipient age gap, distance, and organ type. A logistic-regression model estimates the probability of a successful transplant, and candidates are ranked by it.

D. Exact Explainability

For logistic regression the log-odds of success is

$$\log(p / (1 - p)) = b + \Sigma\, w_i\, z_i$$

where z_i is the i-th standardised feature and w_i its learned weight. The term $w_i z_i$ is therefore the exact contribution of feature i, measured against an average training case, with no approximation. Fig. 3 shows the explanation for an example kidney candidate. The platform displays these contributions beside every ranked organ.



Exact per-factor explanation for an example kidney candidate (predicted success 84%). Contributions are relative to an average training case.

E. Two-Person Rule for Urgency

Urgency decides queue position, so it is protected by a second approval. A doctor may adjust a government-route patient's urgency; if the change is 20 points or more, the queue is not changed until a hospital administrator countersigns it. A proposer cannot countersign their own request, and a newer proposal replaces an older pending one. On the private route, the doctor's certification must always be countersigned before payment can begin.

F. Role-Based Access

Patients see only their own cases; doctors assess urgency; hospital administrators countersign, confirm matches, schedule, and complete; and government administrators oversee the queue. These rules are enforced in the API, not only in the interface.

G. Aftercare

When a transplant is marked complete, the platform creates a starter plan of medication reminders, a one-week follow-up, a one-month check-up, and general guidance, all due-dated. These are reminders and general advice, not diagnosis; clinical decisions remain with the transplant team.

H. Mobile Application (Next Stage)

A mobile client is the next stage of development. It will reuse the existing REST API and JWT authentication, so no change to the server is expected. Planned features include case status and a step tracker, push notifications for approval, match, and appointment events, document upload from the phone camera, appointment reminders, and the aftercare checklist. The server has been designed mobile-ready (stateless token authentication and JSON responses), but the application itself has not yet been built or tested.

V. RESULTS AND EVALUATION

Two experiments were run on synthetic data. They show how the design choices behave under stated assumptions; they are not clinical evidence.

A. Matching Model

The model was trained on 6,000 synthetic cases generated from documented clinical relationships (ABO compatibility, HLA mismatch, organ-specific cold-ischemia sensitivity, age gap, and distance) with random outcome noise. On a held-out 20% test set (Table III), logistic regression reached an accuracy of 63.7% and an ROC AUC of 0.682. A random-forest baseline reached 65.3% and 0.692. The difference is small, so the interpretable model was chosen. Discrimination is modest because outcomes in the generated data were deliberately noisy; the figures describe the synthetic data, not real transplants.

Held-out Model Comparison (Synthetic Data)

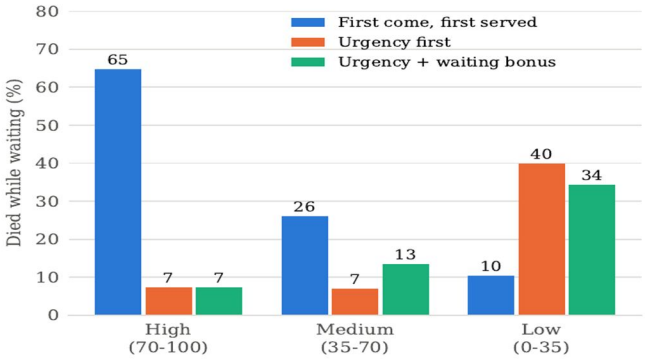
Model	Accuracy	ROC AUC
Logistic regression (used)	0.637	0.682
Random forest (baseline)	0.653	0.692

B. Experiment 1: Queue Policies

The same synthetic patients and organ arrivals (1.2 patients and 0.9 compatible organs per day, 120 replications) were replayed under three policies: first come, first served; urgency first (the current LifeLink queue); and urgency plus a waiting bonus of 0.15 points per day. Patients who outlive their survival time while waiting are counted as deaths, and more urgent patients have shorter survival times. Patients arriving in an initial burn-in period were excluded.

Fig. 4 and Table IV show the results. Urgency-based queuing sharply protects the most urgent patients, with deaths while waiting falling from 64.7% under first come, first served to 7.3%. The cost falls on low-urgency patients: 39.8% died waiting under pure urgency ordering, compared with 10.4% under first come, first served. Adding a small waiting bonus kept the benefit for urgent patients (7.3%), reduced deaths among low-urgency patients to 34.3%, and gave the best overall result (22.6% died while waiting, 76.9% transplanted). Waiting times for the lowest-urgency group, however, rose under the bonus (median 110.8 days among those transplanted), which shows a real trade-off for policy makers rather than a free improvement.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



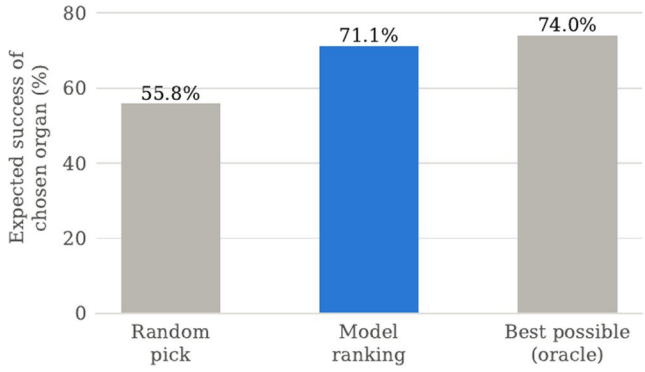
Percentage of patients who died while waiting, by urgency tier and queue policy (synthetic simulation).

Queue Policy Results (% of Patients)

Outcome	FCFS	Urgency	Urgency+wait
Transplanted (all)	74.0	72.2	76.9
Died waiting (all)	26.0	23.4	22.6
Died, high urgency	64.7	7.3	7.3
Died, medium urgency	26.1	7.0	13.5
Died, low urgency	10.4	39.8	34.3
Still waiting (all)	0.0	4.4	0.5

C. Experiment 2: Model Ranking

For each of 20,000 simulated patients, four candidate kidneys were offered. Three selection rules were compared by the true success probability of the organ chosen: a random pick, the model's top-ranked pick, and the best possible pick (an oracle that is not achievable in practice). As shown in Fig. 5, the model's choice reached an expected success of 71.1%, against 55.8% for a random pick and 74.0% for the oracle, closing about 84% of the gap. The model chose the truly best organ in 66.7% of trials, against 25% by chance. Because the evaluation uses the same clinical relationships that generated the training data, it shows that the ranking mechanism works as designed, not that it would perform equally on real patients.



Expected success of the organ chosen by three selection rules (20,000 simulated patients, four candidates each).

D. Workflow Validation

The full flow was exercised end to end on a local deployment with seeded demonstration users: login, route choice, doctor assessment, countersignature, matching with explanations, match confirmation, appointment, completion, and aftercare. Matching rules are enforced on the server: a patient can only match their own case, a confirmed match cannot be re-run, a matched patient leaves the queue, and an organ cannot be booked twice. This validation is a functional demonstration, not a usability or clinical study.

VI. LIMITATIONS AND FUTURE WORK

1) Synthetic data. Real outcome registries require formal data-use agreements and ethical review. All model and simulation results describe synthetic data.
2) Circular validation. The model is checked against the relationships used to generate its training data.
3) HLA placeholder. HLA mismatch is a deterministic placeholder derived from record identifiers, not a laboratory result.
4) Simulated payment. No real payment gateway is connected.
5) Prototype security. Production use would need data-protection compliance, encrypted storage, and an independent security review.
6) Mobile application. The client is planned, not built, and has not been tested with users.

Future work includes building the mobile application with push notifications, adding a waiting-time bonus to the live queue (supported by Experiment 1), validating the model on real, de-identified data under ethical approval, integrating verified hospital and registry data, and a usability study with clinicians.

VII. CONCLUSION

LifeLink shows how a transplant platform can unite government and private routes while keeping urgency verifiable, matching explainable, and every action auditable. The two-stage matching engine and its exact explanations give clinicians something they can check, and the simulation study shows both the benefit of urgency-based queuing and its cost to low-urgency patients, which a waiting-time bonus can partly offset. The system is a prototype built on synthetic data; its value is in the design and in a working end-to-end workflow that real data, clinical review, and a mobile client can build on.

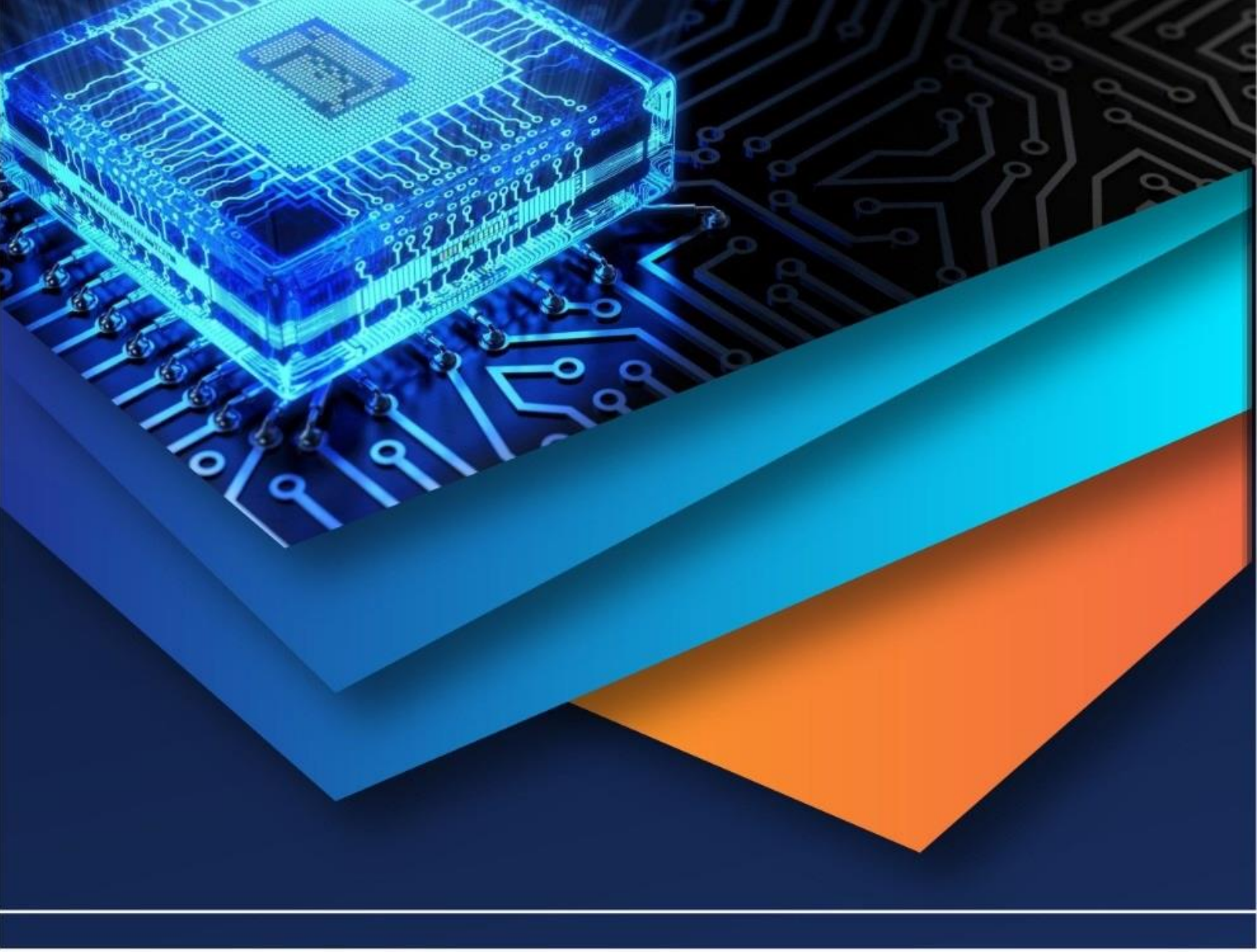
VIII. ACKNOWLEDGMENT

We thank Visvesvaraya Technological University, Belagavi, for the opportunity to undertake this work, and Vemana Institute of Technology, Bengaluru, for the facilities provided. We are grateful to the Principal and to the Head of the Department of Artificial Intelligence & Machine Learning for their encouragement and support. We sincerely thank our guide, Mr. Harish Kumar M, Department of Artificial Intelligence & Machine Learning, for his guidance, valuable suggestions, and constant support throughout this work. We also thank the faculty members, our classmates, friends, and families for their support.

REFERENCES

[1] World Health Organization and Spanish Transplant Organization (ONT), “Global Observatory on Donation and Transplantation.” [Online]. Available: https://www.transplant-observatory.org
[2] A. K. Israni et al., “New national allocation policy for deceased donor kidneys in the United States and possible effect on patient outcomes,” J. Am. Soc. Nephrol., vol. 25, no. 8, pp. 1842–1848, 2014.
[3] Government of India, “The Transplantation of Human Organs Act, 1994 (Act 42 of 1994),” Ministry of Health and Family Welfare, New Delhi, India.
[4] C. Rudin, “Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead,” Nature Mach. Intell., vol. 1, no. 5, pp. 206–215, May 2019.
[5] D. W. Hosmer, S. Lemeshow, and R. X. Sturdivant, Applied Logistic Regression, 3rd ed. Hoboken, NJ, USA: Wiley, 2013.
[6] F. Pedregosa et al., “Scikit-learn: Machine learning in Python,” J. Mach. Learn. Res., vol. 12, pp. 2825–2830, 2011.
[7] S. M. Lundberg and S.-I. Lee, “A unified approach to interpreting model predictions,” in Proc. Adv. Neural Inf. Process. Syst., vol. 30, 2017.
[8] Django Software Foundation, “Django documentation.” [Online]. Available: https://docs.djangoproject.com
[9] M. Jones, J. Bradley, and N. Sakimura, “JSON Web Token (JWT),” IETF RFC 7519, May 2015.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)