WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

BenefitLens: Transparent Government Benefit Disbursement Tracker

Thrishna R[1], Ugesh A[2], Vaira Aatheesh S[3], Vaishnavi K[4], Mrs. S. Hemalatha[5]

[1-4]UG students, Department of Computer Science and Engineering, Sri Shakthi Institute of Engineering and Technology (An Autonomous Institution), Coimbatore – 641062, Tamil Nadu, India

[5]Assistant Professor, Department of Computer Science and Engineering, Sri Shakthi Institute of Engineering and Technology, Coimbatore – 641062, Tamil Nadu, India.

ABSTRACT :

Government welfare schemes in India provide financial assistance and essential services in areas such as education, healthcare, housing, agriculture, and employment. However, information about these schemes is distributed across multiple platforms, making scheme discovery, eligibility assessment, and benefit tracking challenging for beneficiaries. This paper presents BenefitLens, a web-based prototype designed to integrate government scheme discovery, eligibility checking, application tracking, and benefit-disbursement monitoring within a unified interface. The system provides scheme search, rule-based eligibility filtering, multi-stage application tracking, and a beneficiary dashboard that compares sanctioned, disbursed, and beneficiary-reported amounts. It also includes role-specific interfaces for citizens, nodal officers, and administrators. The prototype is developed using React, Vite, and Tailwind CSS, with demonstration data maintained in a client-side JavaScript object repository, and is deployed on Netlify. Its reconciliation mechanism calculates pending amounts and identifies discrepancies requiring further review. A demonstration using illustrative records shows a ₹4,000 discrepancy and a reconciliation indicator of 75% (3 of 4 records). These results demonstrate interface functionality rather than verified government payment activity. The current prototype does not integrate with official government payment systems, verify beneficiary identities, or use live financial data. The paper discusses the system architecture, implementation, reconciliation logic, and limitations, and identifies secure backend integration, usability evaluation, and multilingual support as future work.

Keywords: Government Welfare Schemes, Direct Benefit Transfer, Benefit Disbursement Tracking, E-Governance, Eligibility Assessment, Financial Transparency, Web Application.

1. Introduction

Government welfare schemes provide financial assistance, educational support, healthcare, housing, agricultural help and employment services to citizens. Eligible people, however, often struggle to find the right scheme, because information is spread across portals and offices and eligibility conditions are written in technical language. Even after an application succeeds, a beneficiary has few ways to check whether the sanctioned amount was released and credited in full.

India has moved many welfare payments to Direct Benefit Transfer (DBT), in which funds are credited directly to beneficiaries' bank accounts [2], [4]. Official platforms such as myScheme [1], the DBT Bharat portal [2] and the Public Financial Management System (PFMS) [3] serve scheme discovery, DBT information and payment management respectively. A beneficiary-facing view that places the sanctioned amount, the released amount and the amount the beneficiary reports having received side by side is a complementary need. BenefitLens is a prototype of such a view, combined with scheme search and eligibility checking.

The objectives of the work are:

1. to give citizens one interface for searching schemes and checking eligibility;
2. to let citizens follow an application through its stages;
3. to show sanctioned, disbursed, pending and beneficiary-reported amounts for each benefit and flag discrepancies;
4. to provide separate views for citizens, nodal officers and administrators; and
5. to deliver the system as a browser-based application that needs no installation.

The contributions of this paper are (i) a reconciliation view that compares sanctioned, disbursed and beneficiary-reported amounts, (ii) a working React prototype with role-based portals running on demonstration data, and (iii) an explicit separation of implemented features from proposed ones, so that the claims made here can be checked against the prototype.

The rest of the paper is organised as follows. Section 2 reviews related work. Section 3 describes the design and methodology. Section 4 states the implementation status. Section 5 presents the results and discussion, including limitations. Section 6 concludes and lists future work.

2. Related Work

2.1 Government Scheme Discovery and Eligibility Assessment

Government initiatives increasingly use digital platforms to improve access to welfare information. The myScheme portal [1] provides information about government schemes and their eligibility requirements, while the National Portal of India [5] offers access to information about government services and initiatives. These platforms help citizens discover relevant schemes; however, beneficiaries may still need to interpret eligibility conditions and compare information across different schemes.

BenefitLens addresses this usability challenge through a unified search interface and a rule-based eligibility calculator. The calculator compares user-provided information against structured eligibility conditions in the demonstration dataset. Its purpose is to simplify scheme exploration rather than replace official eligibility verification or government application procedures.

2.2 Direct Benefit Transfer and Payment Monitoring

Direct Benefit Transfer (DBT) is an important mechanism for delivering government benefits directly to beneficiaries' bank accounts. The DBT Bharat portal [2] provides information about the implementation of DBT, while the Public Financial Management System (PFMS) [3] supports government financial management and related payment services.

Barnwal [4] examined India's Direct Benefit Transfer reform in the context of cooking-fuel subsidies and reported reductions in the diversion of subsidised fuel. This research provides relevant context for understanding the role of digital transfer mechanisms in improving benefit delivery.

Although these initiatives support digital welfare administration, beneficiaries may still require a consolidated interface for understanding sanctioned amounts, disbursed amounts, and amounts they report receiving. BenefitLens explores this requirement through a beneficiary-facing reconciliation dashboard. Its current implementation uses illustrative records and does not access official payment databases.

2.3 Positioning and Contribution of BenefitLens

BenefitLens is designed as a complementary prototype rather than a replacement for existing government portals. It combines scheme discovery, rule-based eligibility filtering, application tracking, and benefit-disbursement comparison in a single web interface.

The main technical contribution is the presentation of sanctioned, disbursed, and beneficiary-reported amounts alongside pending amounts and discrepancy indicators. Separate interfaces for citizens, nodal officers, and administrators illustrate how different user roles could interact with the system. Unlike a live government payment-monitoring service, the present prototype uses client-side demonstration data and local role selection. Its results therefore establish the behaviour of the implemented interface, not the accuracy of real-world payment verification.

3. System Design and Methodology

3.1 Overview and architecture

BenefitLens is a single-page application (SPA) with three user roles: citizen, nodal officer and administrator. Fig. 1 shows the architecture of the implemented prototype. The presentation layer holds the React components for the main modules: scheme search, Eligibility Calculator, Track Application, Beneficiary Dashboard and Admin Monitoring. The application logic runs in the browser and handles state, role-based access, input validation, eligibility filtering, search and category indexing, and the reconciliation calculations. The data layer is a JavaScript object repository; there is no server-side database in the current version. Features that are proposed but not implemented are shown separately at the bottom of the figure.

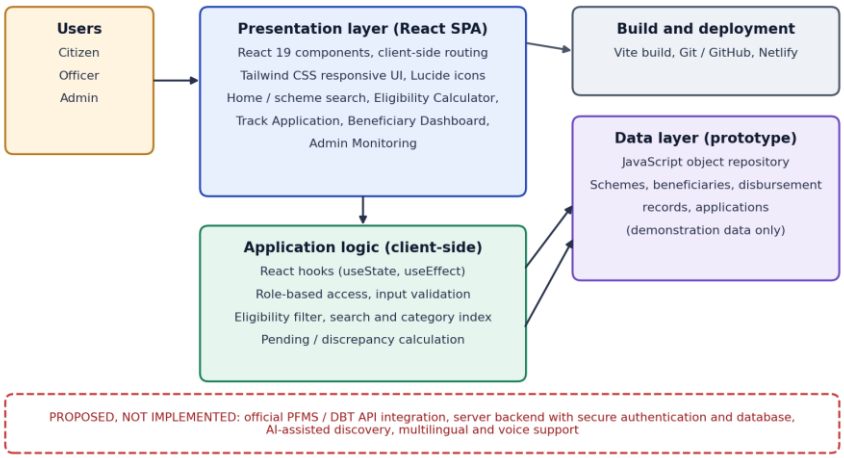


Fig 1 Architecture of the BenefitLens prototype (implemented components above, proposed components in the dashed box)sssss

3.2 Technology stack

The stack was chosen so that the prototype runs entirely in the browser and can be hosted as a static site. Table 1 lists the technologies used.

Table 1 Technology stack of the BenefitLens prototype

Layer	Technology	Role
Markup, style, logic	HTML5, CSS3, JavaScript (ES6+/JSX)	Page structure, styling and application logic
UI library	React 19	Component-based interface
Build tool	Vite	Development server and production build
Styling	Tailwind CSS	Utility-first, responsive styling
Icons and feedback	Lucide React, canvas-confetti	Vector icons; visual feedback on user actions
State management	React hooks (useState, useEffect)	Client-side application state
Data store	JavaScript object repository	Central store of demonstration data, accessed through REST-inspired asynchronous calls
Access control	Local role-based access	Citizen, Officer and Admin views
Routing	SPA client-side routing	Navigation between modules without page reloads
Tools and hosting	Visual Studio Code, Git and GitHub, Netlify	Development, version control and deployment

3.3 Workflow

A citizen opens the website, searches for a scheme or selects a category, and the request is compared with the available data. If relevant information is found, the scheme and disbursement details are retrieved and the beneficiary, payment status and disbursement date are displayed; otherwise the user is shown guidance and asked to refine the search. Fig. 2 shows this workflow. The administrator branch of the workflow (record review, validation and update) is part of the design; the screens evaluated in Section 5 show the monitoring view of that branch.

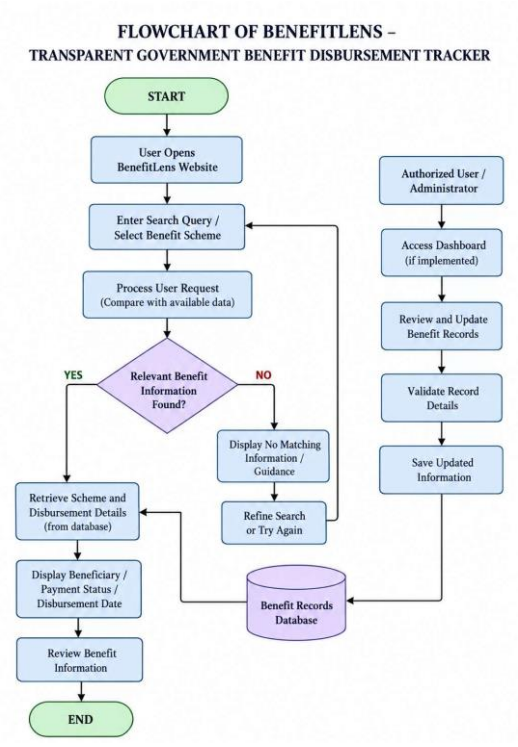


Fig 2 Workflow of BenefitLens

3.4 Data model

Each disbursement record in the repository holds the scheme name and category, the sanctioned amount S, the disbursed amount D, the beneficiary-reported received amount R, the payment status and the disbursement date. Beneficiary profiles and application records are stored in the same repository. All values in the prototype are demonstration data.

3.5 Eligibility filtering and search

Each scheme carries a set of eligibility conditions, for example an age band, an income ceiling, an occupation or an educational qualification, where these apply. The Eligibility Calculator compares the citizen's inputs with these conditions and returns the schemes whose conditions are satisfied. The search function matches the entered scheme name, ministry or entitlement against the repository, and schemes are indexed by category so that results can be browsed by area such as education, health or agriculture.

3.6 Reconciliation logic

For every disbursement record i, the prototype computes the pending amount and the discrepancy as

$$Pending_i = max(0, S_i - D_i) \quad (1)$$

$$Discrepancy_i = | S_i - R_i | \quad (2)$$

The reconciliation mechanism evaluates each disbursement record using the sanctioned amount, disbursed amount, and beneficiary-reported received amount. The pending amount is calculated as the difference between the sanctioned and disbursed amounts, subject to a minimum value of zero. The discrepancy indicator represents the absolute difference between the sanctioned amount and the amount reported as received by the beneficiary.

Records with non-zero discrepancies are flagged for further review. The dashboard also displays a reconciliation indicator based on the records classified as reconciled in the demonstration dataset. These indicators are intended to support preliminary review and improve the visibility of payment-related information.

A discrepancy does not independently establish that a payment has been withheld or incorrectly processed. The reported amount may require verification against authoritative transaction records before any conclusion can be reached. Since the current prototype uses demonstration data and does not connect to official payment systems, the reconciliation mechanism illustrates the proposed monitoring process rather than providing verified financial reconciliation.

3.7 Algorithm

The processing steps of the prototype are:

1. **Role selection and sign-in.** The user chooses the Citizen, Officer or Admin role and signs in; the views available depend on the role.
2. **Query input.** The user enters a search term, selects a scheme or category, or completes the eligibility form.
3. **Validation.** Inputs are validated before processing.
4. **Matching.** The query is compared with the repository, and eligibility conditions are evaluated.
5. **Retrieval.** Matching scheme and disbursement records are retrieved.
6. **Reconciliation.** Pending amount and discrepancy are computed per record using Eqs. (1) and (2), and records are flagged.
7. **Display.** Scheme details, payment status, dates, indicators and alerts are shown.
8. **Refinement.** If nothing matches, guidance is displayed and the user is asked to refine the query.

3.8 Access control and deployment

Access control is local to the browser. The sign-in screen offers three portal roles and "quick demonstration access" profiles for a beneficiary, a nodal officer and an administrator, so that all three views can be demonstrated without a backend. This arrangement is suitable for a demonstration only (see Section 5.4). The application is built with Vite, kept under Git and GitHub version control, and deployed as a static site on Netlify.

4. Implementation Status

Table 2 states which features are present in the prototype described in this paper and which are proposed. Only the features in the first group are evaluated in Section 5.

Table 2 Implemented and proposed features

Feature	Status	Remark
Scheme search and category browsing	Implemented	Search by scheme name, ministry or entitlement
Multi-criteria eligibility calculator	Implemented	Rule-based filter over structured scheme data
Multi-stage application tracking	Implemented	Track Application module
Role-based sign-in (Citizen, Officer, Admin)	Implemented	Local, with demonstration profiles
Beneficiary dashboard with reconciliation	Implemented	Sanctioned, disbursed, pending, discrepancy, audit indicator
Payment ledger and audit-trail view	Implemented	Shown as a dashboard tab
Admin monitoring dashboard	Implemented	Evaluated with an empty dataset
Static deployment on Netlify	Implemented	Browser access, no installation
Integration with official PFMS / DBT data	Proposed	Needs authorised API access
Server backend, database, secure authentication	Proposed	Replaces the local repository and demo sign-in
AI-assisted natural-language discovery	Proposed	Use of the Gemini API has been considered; not evaluated here
Multilingual and voice-based interaction	Proposed	Not part of the current prototype
Notifications for payment updates	Proposed	Requires real-time data sources
Usability study with real beneficiaries	Proposed	Not yet conducted

5. Results and Discussion

The BenefitLens prototype was examined through a walkthrough of its implemented interface using demonstration data. The evaluation focused on the availability of the main modules, the presentation of scheme information, the operation of the eligibility interface, application tracking, role-specific views, and the display of benefit-disbursement indicators.

The results presented in this section describe the behaviour visible in the prototype. They are intended to illustrate the implementation of the proposed system rather than establish its accuracy, performance, usability, or effectiveness in real-world government benefit delivery. No formal user study, live payment verification, or integration test with official government systems was conducted.

5.1 Prototype walkthrough

Homepage (Fig. 3). The homepage introduces BenefitLens as a tracker that aims to make government benefits transparent, traceable and accountable. It offers a search box for scheme name, ministry or entitlement. The navigation bar gives access to Schemes & Welfare, Eligibility Calculator, Track Application, Beneficiary Dashboard and Admin Monitoring, together with sign-in and beneficiary registration.



Fig 3 Homepage of BenefitLens

Sign-in (Fig. 4). The secure gateway lets the user choose the Citizen, Officer or Admin role, enter a registered mobile number or government e-mail with a password, or register as a new beneficiary. The quick-access profiles represent a beneficiary (farmer), a nodal officer and an IAS administrator (audit).

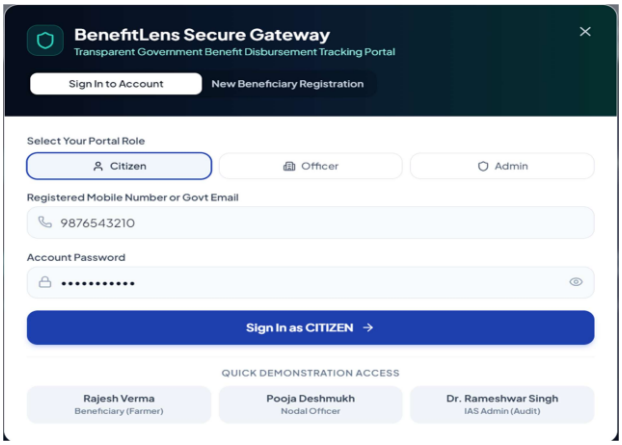


Fig 4 Role-based sign-in screen with demonstration profiles

Beneficiary dashboard (Fig. 5). For a demonstration beneficiary the dashboard shows an e-KYC status badge, a masked identifier and the jurisdiction, followed by five indicators: total sanctioned, total disbursed, total pending, discrepancy amount and a gateway-audit indicator. A banner reports that a shortfall between the sanctioned amount and the self-reported bank credit has been detected and that the nodal officer has been alerted. Tabs give access to the sanctioned-versus-received comparison, the payment ledger and audit trail, beneficiary confirmations, eligible schemes, applications, and profile and KYC details.

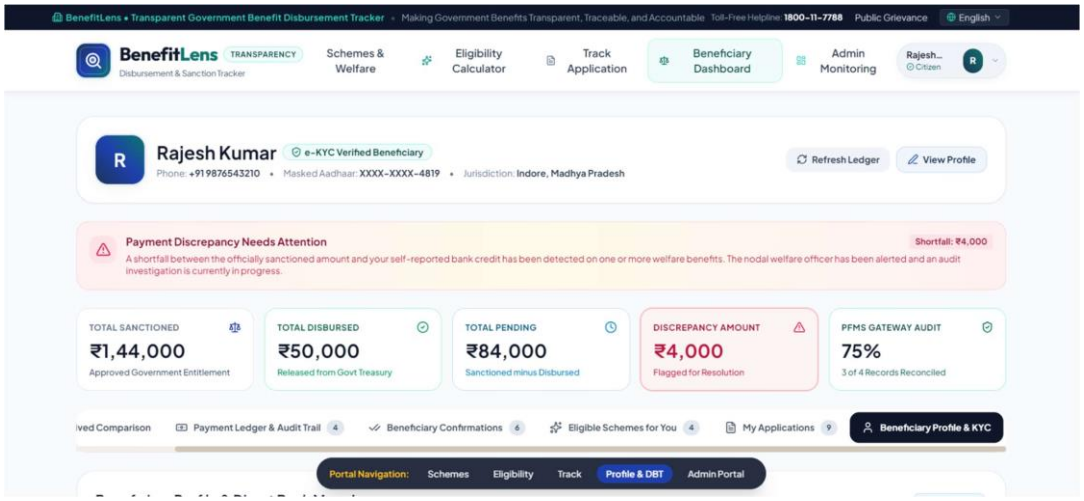


Fig 5 Beneficiary dashboard showing a flagged discrepancy

Admin monitoring. In the admin view captured for this study no benefit records were loaded, so all totals read ₹0 and the page reports that no disbursement records were found. This shows the empty state of the interface, not payment activity. With no records the audit indicator reads 100% (0 of 0 records reconciled); this is a display default and not evidence of reconciliation.

5.2 Worked Example

The beneficiary dashboard provides an illustrative example of how BenefitLens presents benefit-disbursement information. Table 3 summarises selected indicators displayed for the demonstration beneficiary.

The dashboard displays a total sanctioned amount of ₹1,44,000, a total disbursed amount of ₹50,000, and a discrepancy indicator of ₹4,000. The reconciliation indicator reports that three of four records are reconciled, corresponding to 75% of the records in the demonstration dataset.

These values illustrate the presentation of payment-related information and the identification of records requiring further review. The discrepancy value represents the result shown by the prototype and should not be interpreted as an independently verified financial loss or confirmed payment shortfall. Similarly, the reconciliation indicator reflects the classification of demonstration records and does not establish the accuracy of actual government transactions.

The example demonstrates how a consolidated dashboard can present multiple indicators in one interface. Verification against authoritative transaction records and evaluation using real-world data remain necessary before the approach can be used for operational payment monitoring.

Table 3 Selected dashboard indicators for the demonstration beneficiary

Indicator	Value shown	Meaning in the interface
Total sanctioned	₹1,44,000	Approved government entitlement
Total disbursed	₹50,000	Released from government treasury
Discrepancy amount	₹4,000	Shortfall between sanctioned amount and beneficiary-reported credit, flagged for resolution
Gateway audit indicator	75%	3 of 4 records reconciled

5.3 Qualitative comparison with manual searching

Table 4 contrasts the conventional portal-by-portal process described in Section 2 with the BenefitLens prototype. The comparison is qualitative, reflects the authors' description of conventional practice rather than a systematic survey, and is not based on timed trials.

Table 4 Qualitative comparison (design-level, not measured)

Task	Conventional process	BenefitLens prototype
Finding schemes	Search several websites and offices by keyword	One search box and category browsing
Checking eligibility	Read and compare conditions for each scheme	Eligibility calculator filters schemes from user inputs
Following an application	Depends on the portal or office concerned	Track Application module in the same interface
Checking that money arrived	No single view of sanctioned, released and received amounts	Dashboard compares the three and flags a discrepancy

5.4 Limitations

The current BenefitLens prototype has several limitations.

1. **Demonstration data:** The application uses illustrative records stored in a client-side JavaScript object repository. It does not retrieve live government scheme or payment information.
2. **Authentication and access control:** Role selection and demonstration sign-in are handled locally in the browser. These mechanisms do not provide production-grade authentication, authorisation, or protection against unauthorised access.
3. **Unverified beneficiary-reported amounts:** The received amount is based on demonstration information and is not independently verified against bank statements or official payment records.
4. **Identity verification:** The displayed e-KYC status and masked identifiers are demonstration interface elements. The prototype does not perform actual identity verification.
5. **Limited evaluation:** The current results are based on a prototype walkthrough. Formal usability testing, performance benchmarking, and evaluation with real beneficiaries and government officers have not been conducted.
6. **External system integration:** The application does not connect to PFMS, DBT Bharat, or other official government systems. Any future integration would require authorised access, suitable security controls, and compliance with applicable legal requirements.

These limitations mean that the present implementation should be considered a proof of concept. Further development and evaluation are necessary before it can support real-world government benefit monitoring.

6. Conclusion and Future Work

This paper presented BenefitLens, a web-based prototype designed to integrate government scheme discovery, eligibility checking, application tracking, and benefit-disbursement monitoring within a unified interface. Developed using React, Vite, and Tailwind CSS, the application provides role-specific views and a beneficiary dashboard that compares sanctioned, disbursed, and beneficiary-reported amounts.

The prototype demonstrates how pending amounts and discrepancies can be presented to support the review of benefit-related records. The walkthrough using illustrative data shows the operation of the interface and its reconciliation indicators. However, these results do not establish the accuracy of real-world payment monitoring because the application does not use verified government transaction data or independently confirmed beneficiary records.

Future work will focus on developing a secure backend with a persistent database, implementing server-side authentication and authorisation, and exploring authorised integration with relevant government data sources. Further work will also include systematic functional and performance testing, usability evaluation with intended users, multilingual support, and natural-language scheme discovery. These improvements would help assess the feasibility of extending the current prototype into a more reliable and accessible benefit-monitoring system.

Acknowledgement

The authors thank Mrs. S. Hemalatha, Assistant Professor, and Dr. K. E. Kannammal, Professor and Head, Department of Computer Science and Engineering, Sri Shakthi Institute of Engineering and Technology, Coimbatore, for their guidance and for the facilities provided.

REFERENCES :

1. Government of India, "myScheme: National Platform for Government Schemes," Ministry of Electronics and Information Technology. [Online]. Available: https://www.myscheme.gov.in
2. Government of India, "DBT Bharat Portal." [Online]. Available: https://dbtbharat.gov.in
3. Controller General of Accounts, Ministry of Finance, Government of India, "Public Financial Management System (PFMS)." [Online]. Available: https://pfms.nic.in
4. P. Barnwal, "Curbing leakage in public programs: Evidence from India's Direct Benefit Transfer policy," *American Economic Review*, vol. 114, no. 12, pp. 3812–3846, Dec. 2024, doi: 10.1257/aer.20161864.
5. Government of India, "National Portal of India." [Online]. Available: https://www.india.gov.in
6. Meta Platforms, Inc., "React Documentation." [Online]. Available: https://react.dev
7. Vite, "Vite Documentation." [Online]. Available: https://vite.dev
8. Tailwind Labs, "Tailwind CSS Documentation." [Online]. Available: https://tailwindcss.com/docs
9. Mozilla Developer Network, "JavaScript Guide," MDN Web Docs. [Online]. Available: https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide
10. Netlify, "Netlify Docs." [Online]. Available: https://docs.netlify.com
11. Government of India, "The Digital Personal Data Protection Act, 2023 (Act No. 22 of 2023)."