Uni-Link: A Centralized Placement Data Visualization Portal

Shivam Kumar

Computer Science

KIET Group Of Institutions

Ghaziabad,India

shivam.2125cs1105@kiet.edu

Shivani

Computer Science

KIET Group of Institutions

Ghaziabad,India

shivani.cs@kiet.edu

Sudhanshu Singh

Computer Science

KIET Group of Institutions

Ghaziabad,India

sudhanshu.2125cs1128@kiet.edu

*Abstract*— Uni-Link is a centralized repository of placement data from Indian higher educational institutions, designed to present in a comprehensive and accessible manner to institutions and corporate recruiters. It aggregates information on student placements, top recruiters, salary trends, industry-wise distributions, and placement trends over time, thereby offering a very useful tool to analyse placement trends, identify emerging opportunities, and make informed career decisions.

Vaibhav Bisht

Computer Science

KIET Group Of Institutions

Ghaziabad,India

vaibhav.2125cs1119@kiet.edu

Raj Singh

Computer Science

KIET Group of Institutions

Ghaziabad,India

raj.2125cs1079@kiet.edu

In addition, the data from Uni-Link may be employed by various government agencies to monitor the trends of industries, evaluate the effect of educational policies, and make data-driven decisions to improve higher education outcomes. While the platform is primarily accessible to the institutions and corporate recruiters themselves, government agencies can request access to the data for policy analysis and research purposes.

Keywords— NextJS, PostgreSQL, Prisma, DockerHub, Centralizing Placement Data

# Introduction

Unilink is a data-rich platform designed to centralize and standardize placement information across institutions and recruiters. Therefore, it becomes a verified repository of placement records under which transparency, visibility, and trust emerge among stakeholders that include the educational institutions, corporates, and the government bodies. Thus, gluing placement data from higher education institutions in India, Unilink provides solid hints to hiring trends, company participation, and student employability.

The placement process in India is majorly not transparent, disallowing institutions to validate placement records and recruiters to rely on accurate data. Unilink counters this because it provides a one-stop shop where institutions upload placement data, after which it is verified by the recruiting companies, ensuring credibility of the data; hence it becomes an enduring reference for policy, recruitment planning, and institutional assessment.

Unilink provides end-to-end analytics that can help institutions track placement performance, assess hiring trends, and improve placement strategies. Corporates further use the platform to compare the hiring statistics across institutions for talent pooling and efficiency into their recruitment mechanism. The government can use Unilink’s dashboard in support of data-based policy-making and planning for their workforce. Incorporating data visualization techniques in the development of this platform shall ensure the presentation of insights in a user-friendly and stakeholders-friendly manner for discussing metrics of interests like company participation, salary trends, and employment distribution across industries.

Carrying out these processes will help Unilink confer transparency, better management of placement records, and better engagement between academia and industry for the good of one placement ecosystem in a more efficient and responsible manner.

# Litereature Survey

The evolution of Unilink is based on pre-existing studies and best practices in data visualization, database management, and web application development. This project aims to create a centralized, efficient display of verified placement records readily available to educational institutions, recruiters, and government bodies. Dynamic visualization using pre-existing datasets such as Excel files, in addition to data transformation techniques, is the core of Unilink. It employs Next.js[1][2] for a high-performing, server-rendered front-end and Prisma as an ORM with PostgreSQL[3][4] for the structured management of data. These create a scalable, responsive, and intuitive user experience at Unilink. Compatibility provisions enable smooth data processing with increased interactivity and good placement analytics. Thus, it is a significant resource for decision-making based on data.

G. Dinesh[7] advocates for the placement dashboards as an advanced recruitment tool based on their being a repository through which validated records could be passed on for further employability analysis. This resonates with the efforts of Unilink to create an approved platform for analytics in placement and has features for data accuracy and even transparency among the diverse stakeholders. The study further strengthens the rationale for the call for an interactive dashboard that could offer insights into trends of placements, recruiters' engagement, and metrics defining student employability

Deepmala Srivastava[5] dedicates her vast work to proving the importance of data visualization for analyzing trends and pattern making. Data visualization is categorized into spreadsheets, some special software, and programming libraries. The implementation of all these visualization techniques gets incorporated into Unilink itself to ensure that the dashboard provides actionable insights very clearly. Michael Stonebraker[3] et al. also reinforced this in their research using PostgreSQL[3]. Particular strengths lie in its extensibility, rule-based triggers, and data management capabilities. Added, this makes it a strong candidate for the Unilink database management system, allowing for faster, flexible, and secure data storage, retrieval, and security.

Ahmad Tasnim Siddiqui[6] has discussed various tools in data visualization such as Power BI and Tableau with other related issues regarding the scalability and correctness in working with large data. These findings validate the quality of data presentations in Unilink so as to ensure stakeholders get the necessary and right-time insights. Ciprian-Octavian Truică et al. discuss asynchronous replication mechanisms in PostgreSQL, MySQL, and SQL Server. They make a point about how trade-offs are implicated when one is looking at performance and the higher end of data consistency. Their research informs Unilink's replication approach with reference to the database, ensuring that students' records are efficiently updated without compromising their availability.

Tarun, Dr. Vishal Shrivastava, and Dr. Akhil Pandey[2] drive home the points made on the advantages of Next.js; most especially, they talk about server-side rendering (SSR) for statically generated pages (SSG), apart from easy API integration. Their gleanings contribute to informing Unilink's strategy in front-end development, wherein the solution is geared toward amplifying performance, the prowess of the site at being listed by search engines in terms of SEO, and overall enhanced user experience. As per Antony Unwin[11], the major differentiators between exploratory and presentation graphics are high clarity and intuitive usage facets. It thus underscores the need to design the dashboard at Unilink, not only for effective trend analysis but also to convey the findings in an intuitive yet attractive visual manner.

Collectively, these studies validate the overall architectural choices of Unilink and act to further support its goals concerning data integrity, efficient visualization, and improved user experience. With research-backed strategies for database management, frontend performance, and data analytics, Unilink establishes itself as a robust platform for empowering institutions, recruiters, and policymakers to provide sound placement insights.

# Tools And Technology

## Visual Studio Code

VS Code (Visual Studio Code) is a crucial tool for Next.js development, providing developers with a versatile and efficient platform for coding, debugging, and project management. It complements the needs of Next.js development with its rich feature set and user-friendly interface. One of the key advantages of VS Code is its extensive language support, particularly for JavaScript and TypeScript, which are essential for both the front-end and back-end aspects of Next.js applications. Additionally, VS Code seamlessly integrates with other technologies used in Next.js development, such as JSON for configuration files and API routes for server-side functionality.

## Next.js

Next.js was a key part of our stack[1], providing robust front-end and back-end capabilities for our web application. As a powerful React framework, Next.js excels in creating optimized, fast, and dynamic web applications, offering features such as server-side rendering (SSR) and static site generation (SSG) that significantly improved the performance and SEO of our web app.

Overall, Next.js played a crucial role in enhancing the performance and flexibility of our web application. Its SSR/SSG capabilities, API route support, and built-in optimizations made it an excellent choice for building scalable, high-performance web applications. Additionally, Next.js' focus on developer experience enabled us to speed up development while maintaining a well-structured, maintainable codebase.

## PostgreSQL

One of the key factors that helped in the success of our project was that PostgreSQL offered a fully robust, relational database system that focuses highly on data integrity, with support for flexibility. Advanced SQL compliance, for example, is one of the features of PostgreSQL[3], which is perfectly aligned with structured, relational data needs. PostgreSQL helps in complex query capabilities, where it makes it easy and efficient in handling structured datasets, performing relational joins, and improving data retrieval speed as well as database performance.

Another characteristic of PostgreSQL[4] is its multiplicity in schemas in the same database instance, enhancing efficiency, especially in a single environment where there are a variety of data models or tenants to be managed.

In conclusion, it was due to PostgreSQL's relational design, its support for SQL and JSON, strong data integrity features, and scalable performance that it proved a very vital tool in our project. They offered a seamless manner in which to combine structure with flexibility to respond to the ever-changing data requirements of the project, thus contributing to the project's success.

## Prisma

Prisma was very crucial to the success of our project since it provided a robust type-safe ORM solution to connect our application with the database easier. Schema-driven data modeling combined with TypeScript guarantees type safety and quickens development. The Prisma schema file is highly concise and defines the data structures that Prisma maps directly to the database schema itself for the best data management and performance. Prisma CLI makes it much easier when it comes to database migrations and schema updates and will greatly alleviate the workload of administering these tasks for developers. Such syntax is highly effective as regards management of complex relations in relational databases like PostgreSQL.

With smooth integration into TypeScript, there will be type safety throughout the database layer. It reduces the errors and boosts productivity with intelligent code suggestions and compile-time error checks with support in the IDE. This makes coding more efficient and reliable.

## Shad CN UI

ShadCN UI is highly extensible with a customizable, component-based architecture. This enabled the effective development of the product as it came with pre-built, reusable components, using an atomic design approach, which would help keep the app consistent throughout[1].

The other major plus point is the utility-first styling of ShadCN UI Tailwind CSS, which makes it easy for developers to maintain and not require maintenance of several long CSS files. Its flexible best practices and optimized performance co-align well with a user-friendly, aesthetically pleasing interface contributing much to project success.

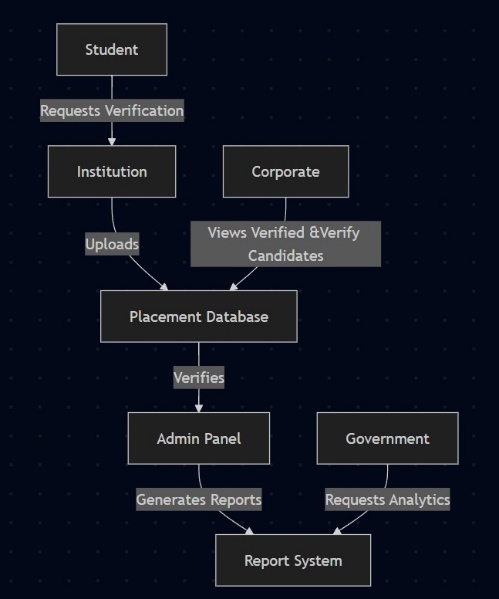
## Docker Hub

The main benefit of Docker Hub is it is centralized. [7][8]. For example, we could quickly pull frequent and diverse pre-built images from a vast library of official repositories like databases, web servers, or even development environments, reducing time and complexity. Also, custom made images built on our local system were effortlessly pushed to Docker Hub, enabling the entire team to access images in a consistent and version-controlled manner improving collaboration. Thus, Docker Hub was the bedrock of our project because it ensured a robust platform for storing and disseminating Docker images. Its ability to automate builds that are CI/CD-compliant and manage both public and private repositories made this tool indispensable for container management and deployment, multiplying the effectiveness and scale of our development process.

# Implementation

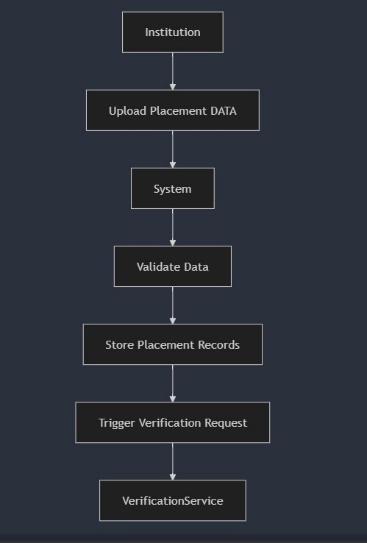
It is basically the aim of the UniLink System: to view records of placement obtained from institutions. Then receive placement verification from corporates and at last allow government authorities to see the trends of placement. To better apprehend the logical flow of information inside the system, Data Flow Diagrams (DFDs) were developed. DFDs provide visualization of the system architecture, with its interconnections to external entities, mapping the flow of information from one system module to another, and providing clarity about the system implementation from the point of view of debug and optimization of processes.

The Level 0 DFD or Context Diagram is an extremely abstract description for UniLink System, being the most abstract representation of the UniLink System. The entire system has been represented as a single process with respect to its interactions with different external entities that include the institutions, students, corporates, government authorities, and admin panel. The institutions upload placement data and send requests for verification. Students can track their placement verification and change profile information. Corporates verify the placement records submitted by the institutions and see verified candidates. Government authorities request analytics and placement reports for policy formulation. The admin panel handles user account creation and general reports for the entire system.



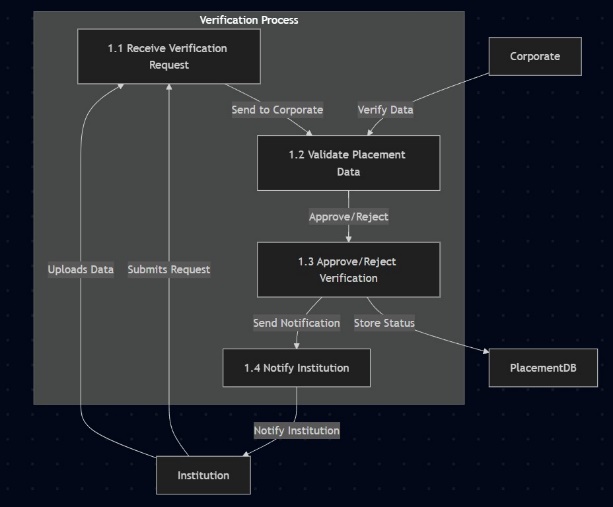
**Figure 1: Level 0 DFD**

Going in-depth for an understanding of system operations, the Level 1 DFD breaks the single process of Level 0 into the three main functional areas of placement verification, placement data handling, and reports and analytics. The placement verification will allow an institution to submit placement records for their candidates, for corporates to verify those records, and the verified placement records to be stored in the database. The placement data management process enables institutions and students to view and track placement records, allows institutions to update placement details of students if necessary, and facilitates the placement database to ensure all submitted and verified records are stored. The reports and analytics module encompasses the widest channel so that while admin panel generates reports of possible placement status at system range, government authorities request for analytics on placement trends or produce reports dynamically based on verified placement data. This modular breakdown allows for systematic implementation by ensuring that each functional area works toward its goals independently while providing the appropriate data flow between them.



**Figure 2:** Level 1 DFD

The Level 2 DFD pays attention to the finer details of placement verification in the system: A drill-down of placement record verification steps is depicted. The starting point of the process consists of institutions submitting placement records, where they are required to upload information regarding candidate names, company names, offer letters, and salary details for unverified placement records in the placement database, pending review by corporate entities. Corporate verification, in reality, involves the approach where corporates will receive a request for verification of the placement data, check details with their records, verify the records in the system, and mark the records in dispute for future action. Once verification is approved, all the changes are made in the system, and notifications are sent to students and institutions. In case of an unverified record, institutions may be required to provide further details. Finally, successful verified placement records are stored in the placement database and thereafter, admins and government authorities can use this information for reporting and analytics. The whole detailed flow of the process ensures that validated placement data alone is made available to the stakeholders thereby creating high transparency and credibility.



**Figure 3: Level 2 DFD**

# Execution

The main objective of the interface is the most efficient, most accessible, and most lucid manner in which these institutions, corporates, and governmental authorities can work with the platform. The system has a very structured workflow for placement data verification and provides transparent insights into employment trends. Users are assigned roles in the placement ecosystem, which grants them access to specific features that pertain to their operations.

*A. Authentication and Role-Based Access*

Google authentication, highly secured for this UniLink platform, enables students to use their Gmail accounts to avoid manually creating an account while enhancing security using OAuth 2.0 to ensure that only genuine individuals are in the system.

Authentication grants role-specific privileges under role-based access control (RBac)—institutes uploading placement data, corporates verifying records, government authorities analyzing reports, and admins managing system access. In addition, UniLink can leverage Google's secure login framework, thereby brings multiple advantages like denial of unauthorized access, simplifying the authentication process, and maintaining data integrity with all users.

A screenshot of a computer

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.

**Figure 4: Sign in Page**

*B. Role-Specific Dashboards and Main Interface*

Once registration is completed, users get redirected to their respective dashboards, the focal point of all activities within UniLink. The institution dashboard allows colleges to submit placement records, manage uploaded data, and track progress of verification. The corporate user is empowered to access submitted records; verify candidate details; approve or flag discrepancies. The government dashboard presents an overview of placement trends, delivers insights through multiple reports, and helps visualize analytical tools. All interfaces are designed with an easy and friendly layout so that one can minimally skill to operate respective assignments.

A screenshot of a computer

AI-generated content may be incorrect.

**Figure 5: Institute Dashboard**

*C. Placement Verification Workflow*

The main function of UniLink is that of a structured verification process for placements and ensures the accuracy and reliability of the employment records they wish to submit. Records of pending verification are being kept in the central database. The corporate users are sent requests to verify those records automatically. After verification, the status of every record gets updated to verified in the system, thus making the data available to the institutions for generating reports and to the government for review. If there are discrepancies in the records, those get flagged, and the institutions are notified for corrective action. This process lays down a pathway whereby only authenticated placement records get considered for any analysis and decision-making.

A screenshot of a black screen

AI-generated content may be incorrect.

**Figure 6: Verification of Students by Company**

*D. Data Visualization and Analytics*

UniLink utilizes advanced data visualization mechanisms to show placement trends and metrics of institutional performance. It dynamically produces reports based on verified records and helps government and institutions to understand placement success rates, trends in industry hiring, and patterns of student employability. Interactive graphs and charts allow stakeholders to view the performance of various placement initiatives in detail; this will help them to strategize in the future for better results[9].

**Figure 7: Governemnt Records**

**Figure 8: Governemnt Visualization (Placeholder)**

*E. Report generation*

These reports shall be composed by way of an automated generation module developed by UniLink to solicit reports from government agencies on matters of institutional placements, employment success rates, and career trajectories[9]. The system lets the administrators ask for data-driven insight targeted toward specific policy needs. The formats are exportable, and that means getting along with different governmental data management systems. With this insight, there will be information for policy-making, skill development schemes, and workforce planning; hence, this will contribute to a more data-driven approach to improving higher education and employment outcomes. Tracking placement success rates over time will significantly inform evolving industry-academia collaborations for improved career opportunities for students.

# CONCLUSION

Uni-Link embodies the high-tech yet user-friendly maintenance of college students' placement records. This framework is designed so that students, institutions, and recruiters can easily use it when queries arise.

Uni-Link uses its strong database of placement records to retrieve and analyze those records. Furthermore, data visualization is dynamic and interactive via the dashboard. Users also enjoy strong searching and filtering options, and they can build compelling reports that inform decision-making and shed light on critical actions about placing a job.

The hallmark of Uni-Link setting itself apart from traditional placement platforms is the focus of the whole experience at the user on making use of the evidence they have[11]. It promotes collaboration and interchange between all (including students, educational institutions, and recruiters) into synergism for a better placement ecosystem.

In such an evolving educational environment, wherein it is important for colleges to help students realize their career goals while also assisting the institutions in achieving their strategic objectives, Uni-Link uses the powers of emerging technologies to transform itself to enclose the changing needs of the user base and to facilitate transformation in the college placement system. The vision does not only tackle immediate challenges but is an invitation to further enhancement in making the process of placement efficient, insightful, and impactful for all stakeholders in the future.

# REFERENCES

1. Mochammad Fariz Syah Lazuardy and Dyah Anggraini, “Modern Front End Web Architectures with React.Js and Next.Js,” International Research Journal of Advanced Engineering and Science, Volume 7, Issue 1, pp. 132-141, 2022.
2. Tarun, Shrivastava, V., & Pandey, A. (2024, March). Introduction on Next.js and its benefits and analysis. Department of Computer Science & Engineering, Arya College of Engineering & I.T., Jaipur, India.
3. Michael Stonebraker, Michael Hirohama, & Lawrence A. Rowe (1990). The implementation of Postgres. IEEE Transactions on Knowledge and Data Engineering, 2(1), 125–142. <https://doi.org/10.1109/69.50912>
4. Truica, C.O., Boicea, A., & Radulescu, F. (2013). Asynchronous replication in Microsoft SQL Server, PostgreSQL and MySQL.
5. Deepmala Srivastava. (2023). *An introduction to data visualization tools and techniques in various domains*. University of Texas at Dallas.
6. Ahmad Tasnim Siddiqui, “Data Visualization: A Study of Tools and Challenges,” *BGI, Faizabad; Department of Computer Science & Applications, Sherwood College of Professional Management*, Lucknow, India, 2021.
7. G. Dinesh, M. N. V. Surekha, D. Harika, A. Sunand, U. Dayanand Kumar, and A. Naveen, “Placements Analytics and Dashboard,” *Sri Vasavi Engineering College (Autonomous)*, Pedatadepalli, Tadepalligudem, Andhra Pradesh – 534101, India, 2024.
8. Chander Velu, S. E. Madnick & Van Alstyne, M. (2013). Centralizing data management with considerations of uncertainty and information-based flexibility. *Journal of Management Information Systems, 30*(3), 179-212. <https://doi.org/10.2753/MIS0742-1222300307>
9. Zatin Gupta, Rishabh Garg, Priyanshi Saxena & Ansh Bansal. (2022). *Data Analysis based Digital Step Towards Waste Management* Tirunelveli, India.
10. [Weiming Zhu](https://onlinelibrary.wiley.com/authored-by/Zhu/Weiming) (2021). *A study of big-data-driven data visualization and visual communication design patterns.*
11. Antony Unwin (2020). *Why is data visualization important? What is important in data visualization?*