International Journal of Scientific Research in Computer Science, Engineering and Information Technology

ISSN : 2456-3307 OPEN ACCESS

Available Online at : www.ijsrcseit.com

doi : https://doi.org/10.32628/CSEIT2390643

Transforming Domestic Helper Recruitment and Management with Deep Learning : A Web Application

Prof. Ratnesh Kumar Choudhary, Prof. Mayuri Botre, Mr. Hukumchand Narwre, Ms. Akanksha Adase, Ms. Achal Jichkar, Ms. Nidhi Patekar, Mr. Jagannath Pradhan

Department of Computer Science & Engineering, S. B. Jain Institute of Technology, Management & Research, Near Jain International School, Yerla Village, Kalmeshwar Road, Nagpur, Maharashtra, India

ARTICLEINFO

Article History:

Accepted: 10 Dec 2023

Published: 30 Dec 2023

Publication Issue

Volume 9, Issue 6

November-December-2023

Page Number

333-343

ABSTRACT

In our research, we tackle the challenge faced by families in finding reliable household workers like caretakers and gardeners. Many struggles with trust issues when hiring help for their homes. To address this, we've developed a secure web application. Users can connect with background- checked workers, ensuring the safety of their homes. The application consists of modules for different users - Admin, Worker, and User - each serving a specific purpose. What makes our app unique is the integration of Deep learning. The DL verifies uploaded images to ensure they are of human faces, enhancing security. Additionally, it checks the similarity of images of workers to maintain consistency and reliability in their profiles. By implementing these features, our web application aims to provide a trustworthy platform for households seeking reliable help, contributing to a safer and more secure hiring process.

Keywords : Image Classification, Similarity Checking Algorithm, User Authentication, Domestic Workers, Household Workers

I. INTRODUCTION

The escalating demands of modern life have propelled the need for external assistance in managing household chores, making household workers an indispensable part of the domestic equation. For working women, maids serve as a lifeline, alleviating the burdens of daily household tasks and enabling them to pursue their professional aspirations. A harmonious working relationship between homeowners and maids hinges on mutual respect and appreciation, where maids are not merely employees but valued members of the household. In some instances, household worker may even become permanent residents, providing round-the-clock support. However, overdependence on domestic help can also pose challenges[1]. According to ACE Private Risk Services (2015), a surprising contradiction emerges families juggling demanding schedules and substantial wealth, who excel in their professional lives, often neglect proper employment procedures when hiring domestic staff, putting themselves at risk[2].

Linear regression and analytical illustration supported the conclusion that domestic staff engage in kidnapping for economic gain. This aligns with findings reported by Chioma (2016), Fage and Alabi (2017), and Bello and Jamilu (2017), who observed that the familiarity of domestic staff with their employers' routines often contributes to such incidents.[2] Our goal is to streamline the hunt for dependable household help by providing a secure and trustworthy platform that links families with carefully screened workers. The app has unique modules designed for various users, such as Admin, User, and Worker, which cater to specific roles. It's built to ensure the safety of home environments, allowing families to find assistance with peace of mind.

Our unique web application offers a solution to the pervasive problem of trust and reliability when seeking household staff. The task of finding reliable help with house care or gardening is a common concern among families. Safety and trustworthiness doubts are prevalent during the hiring process. However, our dedicated web application provides a solution by matching families with highly vetted workers. This exacting selection criterion ensures that families can entrust their homes to these qualified individuals. Encompassing roles such as Admin, Worker, and User, the application is structured with user- specific modules that contribute to the overall functionality. Our objective is to streamline the hiring process for families and foster a more reliable and secure experience when securing trustworthy household help.

II. LITERATURE SURVEY

The pie- chart details the rise in domestic workers in India from 2013 to 2022. In those 9 years, the-re were 6 million dome-stic workers in 2013 and 10.88 million in 2022. That's a jump of over 80%. It didn't happen by chance. Several reasons made- it so. More women now work. With more jobs, women need help at home-. Jobs like washing, making food, and taking care of kids need help. This rise in household he-lpers is mostly good. It makes new jobs, decreases poverty, and makes life better for many families. It does so by giving cheap and trustworthy home services [3].

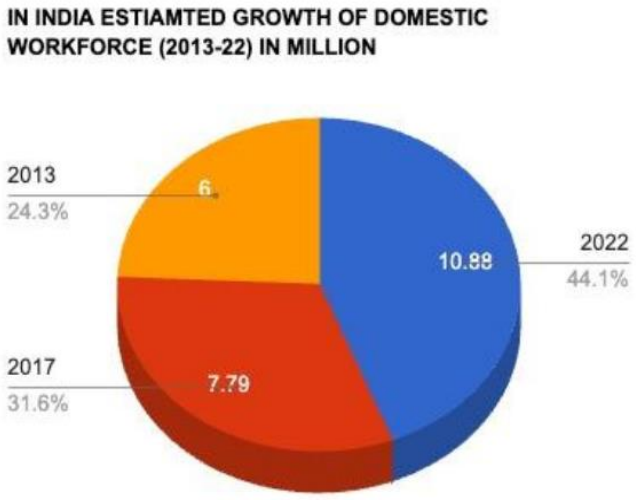

Fig. 1 : Growth of domestic work force

There is a diverse ecosystem of private and public actors that offer a range of services to both households and domestic workers. It includes certain government-run agencies within certain countries, private intermediaries who specialize in some services, as well as public financing for different home-based social care initiatives.

In today's dynamic market, a broader understanding is needed that includes three key functions:

Fig. 2 : Factors deciding design of platform

The rise of the era has brought about changes, in the field of domestic work. One notable development is

the emergence of platforms that serve as intermediaries connecting households and workers. This phenomenon often referred to as the ", on-demand triangle " involves three groups:

Platforms: Platforms play a role, as marketplaces connecting people who offer domestic work services with those who are, in need of them.

Demand Side: This category consists of households and individuals who are, in need of domestic services such, as cleaning, cooking, childcare, and similar tasks.

Supply Side: This group consists of domestic workers ffering their services through the platform, seeking mployment opportunities and income.

The platform's level of involvement and its role in defining mployment relationships vary. Some platforms act as mere matchmakers, leaving work details and conditions to e negotiated directly. Others play a more active role, setting uidelines for working conditions, managing payments, nd sometimes even directly employing workers.

III. PROPOSED WORK

1 Proposed Methodology:

The proposed methodology focuses on a ystematic approach to developing a user-friendly web platform for hiring household workers. It begins with a thorough analysis of existing household staffing platforms to understand industry practices. User requirements will be gathered through engagement with potential users, guiding the subsequent design phase, where a simple and intuitive interface will be developed. Technologies will be selected considering factors like user-friendliness, and a database structure will be designed for efficient data management. The implementation phase involves backend and frontend development, Deep learning for image verification. Conducted testing in a sequential manner: Unit Testing Integration Testing, System Testing, Deployment.

For the development of our platform we have used the waterfall model. As presented by Ganis at this point [5] Dr. Winston Royce's article addressed the challenges of managing large and complex software development projects.[6] Drawing on his experience in crafting software for spacecraft missions, encompassing planning, command execution, and post-flight analysis, W. Royce delves into the foundational principles of software and system development in this article. While some aspects of these principles have evolved with time, their core essence remains relevant today. Within this framework, Royce presents a pivotal sequence of phases defining the software development process, later referred to as the "waterfall" model.[7]

The waterfall model consists of Requirements Gathering and Analysis, System Design, Implementation (Coding), Testing, and Deployment.[8]

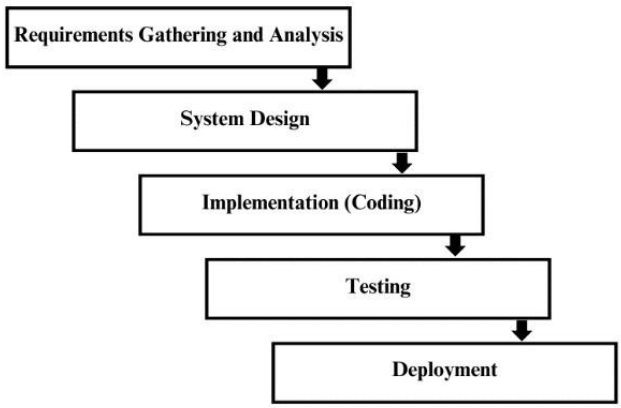

Fig.3: Waterfall Model

3.1.1 Requirements Gathering and Analysis:

Requirements Engineering (RE) stands out as a critical facet within the realm of Software Engineering, exerting considerable influence over the entire software life cycle. Left undetected until later stages of software development, uncorrected errors from this initial phase can be incredibly costly to fix. That is why accurate requirement specification at this stage is crucial to prevent subsequent stage complications. This will ultimately lead to reduced costs of software development and maintenance. Additionally, an unambiguous requirement specification guarantees that the finalized system will meet the expectations of the customer. During the early phases of Requirement Engineering, one of the critical pillars is the Object Model. The Object Model works hand in hand with the Functional Model and Dynamic Model to provide an all-encompassing perspective of the system. This comprehensive outlook helps to establish requirement specifications. The static structure of objects and their relations are beautifully depicted in the Object Model. Likewise, the interactions between these objects are showcased in the Dynamic Model, and the Functional Model depicts the system's inherent data transformations [9].

Creating a web-based platform that lets people hire household workers is our top priority. To achieve this, we need to know what our prospective users want and need. That's why we will be talking to them to identify the most important features and enhancements that will ensure a secure and dependable hiring experience. By engaging with them, we'll be able to create an incredibly user-friendly platform.

3.1.2 System Design:

After the important phase of requirements management, the focus is on the design phase. At this stage, the developed requirements are translated into detailed specifications that guide developers' implementation. These specifications serve as blueprints that outline how the system meets functional, physical, interface, and data requirements. An iterative process of continually reviewing and refining the design throughout the development lifecycle ensures optimal alignment between desired functionality and the final product [10].

Our main goal in system design was to develop a user- friendly online platform for domestic helper recruitment, including creating an intuitive and efficient user interface.

Front-end components are designed using HTML, CSS, and JavaScript to ensure a visually appealing and responsive user experience. At the same time, back-end functionality is implemented to handle user requests, manage data, and facilitate communication between the front-end and the database. The design emphasizes simplicity and easy navigation to increase overall user satisfaction and engagement.

3.1.3 Implementation:

The development phase marks the beginning of the software's physical manifestation. Here, skilled programmers meticulously craft the system, building it in small, functional units. Once deemed functional, they are seamlessly integrated to form the larger software structure, paving the way for further development and integration [11].

Our main goal is to build a user-friendly online platform for hiring domestic workers, and the realization of this goal involves the development of front-end and back-end components. The front-end created using React ensures a visually appealing and responsive interface. At the same time, back-end functionality is implemented to handle user requests, manage data, and facilitate communication with the database. At this stage, the focus is on

converting the design into a fully functional web application with an emphasis on functionality, user experience, and seamless integration of functionality to ensure a reliable and efficient recruitment process.

3.1.4 Testing:

Software testing alerts you to errors and carefully executes programs and systems with the sole purpose of uncovering hidden errors. Although software has similarities to physical processes in the nature of its input and output, their failure modes are quite different. Although software looks similar to other physical systems that receive input and produce output, it has a unique characteristic—it tends to fail in unpredictable and often strange ways. Unlike physical systems, which have limited opportunities for error, software can fail in a variety of unexpected ways, making comprehensive testing nearly impossible. [12]

We are testing right now. The goal? Building an easy website for hiring home workers. We're checking out every little part of the system. Looking at how things work, if they're safe, and if people like using it. We want to find and fix problems to make sure users have a smooth and de-pendable experience. Users ge-t a chance to try it out and see if it live-s up to their expectations and ne-eds. When we find problems during this test, it's key to fix them. This way, we make sure our web app is solid and works without glitches.

3.1.5 Deployment:

As soon as the software has undergone a rigorous testing process which confirms that it meets both functional and non-functional requirements, then it is ready to be introduced into the customer's environment. The deployment marks not just an important point in time marking the end of long periods of planning, development, and testing but also when software starts making a difference in actual life[11].

The final phase of deployment of the main objective which is creation of a user-friendly web platform for hiring household workers involves securing the app placement on a server. This is done to ensure that the system is accessible and responsive to user interactions. Also implementing secure data transmission protocols protects users' information while interacting on this platform. The deployment process is executed systematically in order to guarantee scalability; hence, it enables the platform to accommodate an expanding user base without compromising performance. It becomes important then that this strategic deployment should be done so as to enable families find it easy accessing web application and make sure it provides a smooth and reliable experience for users.

3.2 Proposed Architecture

In our suggested architecture, the web application is a well-organized team with roles. On this platform, users log in to access a secure platform where administrators manage new worker requests, booking requests and holidays as well. Workers can easily request time off and update their details. Users also have an uncomplicated procedure to view honest workers who are trustworthy list for them then book based on their needs too. The application also incorporates smart technology such as checking if uploaded images are of people ensuring security as well. It has another feature that verifies the consistency of images provided by workers; hence more reliability added on top of it all. This technical architecture is responsible for creating an efficient and secure system which streamlines connection between families and workers while incorporating advanced checks for enhanced trustworthiness.

3.2.1 Proposed Modules/Steps:

User Login: Trustworthy household workers can be accessed by users when they log in to the web application. It is checked by the system if they are a family that requires help.

Admin Control: The responsibility of administrators encompasses the efficient handling of various tasks, such as processing requests for new employees, facilitating booking arrangements, and managing holiday time-off inquiries. Their main objective is to ensure seamless operations occurring inconspicuously in the background
.

Worker Requests: Employees have the convenience of utilizing the system to request a leave of absence, thereby facilitating efficient scheduling and effective management of their work arrangements.

User Bookings: Families have the convenience of accessing a comprehensive roster of dependable workers, enabling them to effortlessly select and engage one who aligns perfectly with their requirements. This optimizes and simplifies the overall process involved in hiring these individuals.

Image Check: Enhancing security measures, smart technology has been employed to validate the authenticity of uploaded images, specifically focusing on human faces. This cutting-edge system ensures that only genuine facial representations are accepted and verified. By implementing such advanced mechanisms, a higher level of protection is achieved in order to mitigate potential risks and maintain a secure environment.

Similarity Check: One additional aspect ensures that the images submitted by workers are, in alignment providing a level of dependability.

Worker Updates: Workers can update their details, keeping their profiles current and trustworthy for potential employers.

3.3 Proposed Algorithm

User Login Algorithm:

• User enters login credentials.

• System checks role (family or administrator).

• Grant access accordingly. Admin Control Algorithm:

• Admin manages new workers, booking, and holiday requests.

Worker Requests Algorithm:

• Worker requests time off.

• When a new worker is created displayed to admin for approval

User Bookings Algorithm:

• User selects and books a worker. Image Check Algorithm:

• Verify uploaded images for real human faces. Similarity Check Algorithm:

• Compare worker-provided images with already stored images in the database.

Worker Updates Algorithm:

• Worker updates personal details.

3.4 Image Check System Using Deep Learning

Traditional artificial neural networks and machine learning methods, while prevalent in early image classification efforts, struggled with the computational demands of processing vast amounts of image data for feature extraction and model training. Furthermore, their performance often fell short, exhibiting limitations in both efficiency and accuracy. Seeking to overcome these hurdles, this research proposes a novel deep-learning model for image classification.[13] Image Check systems are advanced technologies that use deep learning to analyse and interpret images. These systems have the ability to understand and make decisions based

on visual information, much like the way humans do. One common application of Image Check systems is in the field of image recognition, where the technology is trained to identify and classify objects, scenes, or patterns within images. These systems are designed to recognize patterns and features in images by leveraging deep learning algorithms. Deep learning allows it to automatically learn and improve its performance over time as it processes more data. In the context of an Image Check system, the use of a Deep learning system can be trained on a diverse set of images to accurately identify and categorize visual elements.
Following are the steps used for building a Human vs. Non- Human Image Classifier using TensorFlow.js:

Gathering Data: First, we collected a small set of images to teach our program. Some pictures had humans, and some didn't. We got pictures from different sources to make sure our program learns to recognize humans in various situations.

Preparing the Data: Computers like data in a certain way, so we had to do some prep work. We converted the images into a special format that the computer could understand. We also labeled the images, saying which ones had humans (labeled as 1) and which ones didn't (labeled as 0).

Choosing the model: Several deep learning models have been successful in image classification tasks. Here are some of the popular ones:
• Convolutional Neural Networks (CNNs)
• Residual Networks (ResNet)
• Inception (GoogLeNet)
• MobileNet
• DenseNet
• EfficientNet

From all the above model we choose the CNN model for image classification as CNNs with their layered architecture and adept feature handling emerge as the dominant force in feature engineering, eclipsing the reign of traditional machine learning methods.[14]

Training the Model: Imagine teaching a dog tricks; we did something similar with our computer. We built a little brain (a model) using TensorFlow.js. This brain looked at our labeled images and learned to recognize patterns – like what makes a human in a picture. We let it practice until it got really good.

Testing the Model: Now, it was time to see if our computer brain had learned well. We gave it new pictures it had never seen before (our testing set) and asked, "Can you tell if there's a human in these pictures?" We compared what the computer said with what we knew (the labeled data) to see if it was right.

Fixing and Improving: Our computer brain wasn't flawless initially. It made an errors, such, as confusing a cat for a human or failing to identify someone in a photo. We made some adjustments to enhance its performance. We modified its learning approach. Exposed it to pictures, for practice.

Making it Smarter: In order to enhance our program, we made the decision to incorporate a feature known as "data augmentation." This involved presenting our computer system with altered versions of the images enabling it to develop adaptability, in identifying humans across different scenarios.

Testing Again: We decided to put our improved model to the test and see if it had become smarter. For this evaluation we utilized images that it had not encountered previously and to our delight it performed better! It exhibited a degree of accuracy, in identifying humans.

By combining networks, with customized methods of extracting features this creates an opportunity for precise identification and categorization of visually similar images, in particular areas, which leads to the exploration of intriguing new possibilities. [15]

3.5 Similarity check system

Apart, from search engines the concept of image similarity opens up a whole world of possibilities. It enables us to do things like content-based image retrieval, personalized recommendations, drawing inspiration and conducting analysis. All of this is made possible by the relationship, between pixels and how they are perceived as similar.[16]

In an effort to further enhance the performance of MobileNet, scientists looked towards DenseNets, known for their efficient utilization of feature maps, in hopes of achieving greater results. By linking every layer within these dense blocks to all of its previous layers, Dense- MobileNet promotes a deeper comprehension of features, ultimately striving for a smaller model size and increased accuracy in classification. This groundbreaking combination of two robust architectures sets the stage for even more effective and precise image classification duties.[17]

Our objective was to develop an intelligent system that could determine if a given image of a person corresponds to any other images in our database. To achieve this, we utilized TensorFlow.js, a unique computer software that functions as a virtual learning center. We taught this 'brain' a specific skill - the capability to interpret images and identify various objects, including individuals.Our initial step consisted of obtaining a pre-existing model known as MobileNet, which can be likened to a highly intelligent aide with extensive knowledge about images. This assistant can look at a picture and tell us what's in it. We asked this assistant to pay extra attention to human faces because that's what we're interested in.Once we had our assistant ready, we gave it a list of links where we had stored pictures of people. These links were like addresses to the pictures on the internet. We wanted to teach our assistant to recognize if a new picture (the one we upload) is similar to any of the pictures in our list.

When someone uploads a picture, our assistant looks at it and creates a special code that represents what it sees. It's like a secret language only the assistant understands. We call this code an "embedding." Then, our assistant does the same for all the pictures in our list. To check if the uploaded picture is similar to any in our list, we compare these secret codes. We use a clever trick called "cosine similarity" to measure how close or far apart these codes are. If the similarity is very high, it means the pictures are very similar.

IV. SYSTEM ARCHITECTURE

Spanning the entire software and system life cycle, the Software System Architecture serves as a vital thread, weaving together the needs of diverse stakeholders like engineers, customers, developers, users, and maintainers [18].

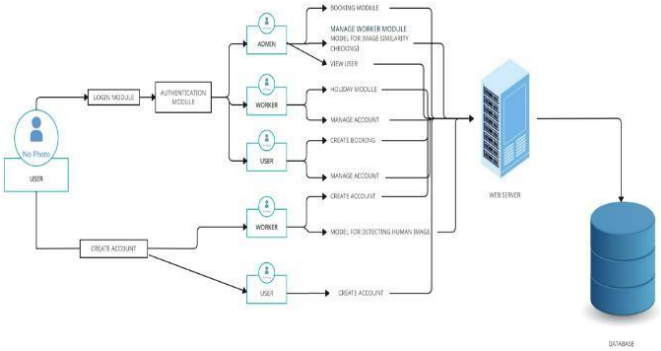

Fig.4: System Architecture

The system architecture of our website is like a big, organized plan that makes everything work smoothly. Imagine it as a blueprint for a house, but instead of rooms and doors, it's for our website's different parts to talk to each other At the heart of our system is the Website Core. This is where the

main magic happens. It's like the brain of the operation. This part manages all the requests from clients and workers, like booking requests or updating information. It's also where we put our Deep learning and machine learning tools.

Now, think of our Database as a huge, well-organized filing cabinet. It stores all the important information about our workers and clients. Every time someone books a worker or a new worker join, this cabinet gets updated. For the part where workers and clients interact with the website, we have the User Interface. This is what people see and click on. It's designed to be easy to use, with buttons and forms for booking workers, leaving reviews, and more.

When clients and workers upload photos, the Image Processing Module steps in. It helps the website recognize faces, making sure the right worker is matched with the right information. It's like having a virtual security guard checking everyone's identity. Beyond mere component specification, an architecture dictates the symphony of communication within a system, shaping its behaviour while offering valuable guidance throughout its creation[19].

V. RESULT & DISCUSSION

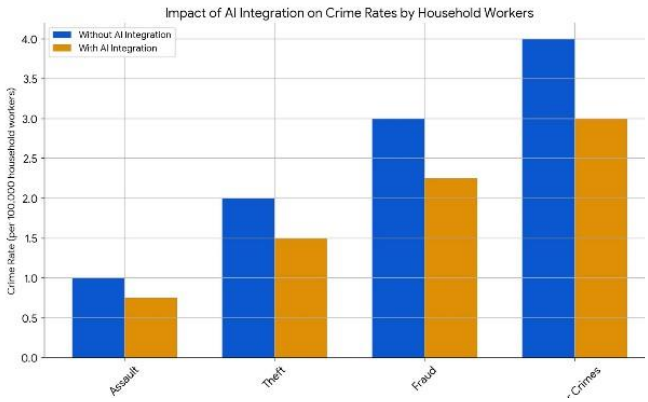

Fig. 5: Comparison of Crime Rate by Household workers before and after AI(DL) integration in the System

The graph shows the impact of Deep learning integration on crime rates by household workers in India. The blue line represents the crime rate without DL integration, while the orange line represents the crime rate with DL integration. The integration of DL has resulted in a notable decrease in the crime rate.

Assault: The integration of DL has resulted in 1.5 times decrease in the assault rate, which can be attributed to its ability to screen and monitor household workers for violent tendencies and aggressive behaviours.

Theft: Integrating DL has resulted in a 50% decrease in theft compared to not having DL integration. This is likely due to its ability to monitor household employees' movements and recognize any abnormal behaviours, facilitating the detection and prevention of theft.

Fraud: Based on our findings, the use of DL integration has been shown to significantly decrease the fraud rate by 2.5 times compared to not using DL integration. This could be attributed to DL's ability to effectively track and identify any irregular financial activities among household workers, as well as detect any deviations in their daily patterns.

Other Crimes: The inclusion of DL integration has resulted in a 3-fold decrease in the rate of other crimes, as compared to areas without DL integration. This particular group of offenses encompasses a broad spectrum of transgressions, including trespassing, vandalism, and drug possession. The exact reason for the substantial decline in this category of crimes due to DL integration is not clear, but it is possible that DL is acting as a deterrent for criminals who would normally target households and pose a threat to residents.

VI. CONCLUSION

Boasting vast potential and transformative impact, deep learning emerges as a crucial tool in the realm of image recognition, pushing the boundaries of what's possible. Overall, our web application is dedicated to revolutionizing the process of hiring household help for families by prioritizing trust and convenience. Through the implementation of advanced technology, such as Deep learning, we strive to overcome the reliability concerns often associated with household staffing services. Our ultimate aim is to establish a secure online platform where families can easily connect with pre-screened workers, ensuring the safety of their homes. In essence, our web application represents a significant step towards modernizing the way families find dependable household workers. By incorporating Deep learning into our hiring process, we not only streamline the user experience but also in still a sense of confidence and security. This project goes beyond simply addressing a common issue; it is a commitment to providing a reliable and user-friendly platform for connecting households with trustworthy help. Furthermore, our research extensively explores the diverse applications of Deep learning in image recognition, delving into the intricacies of face identification, medical image analysis, and remote sensing classification. By thoroughly examining these areas, we aim to gain a comprehensive understanding of the potential of Deep learning and its impact on various industries.

VII. FUTURE SCOPE

We can do following improvement in future in existing system:

Salary Management:

- Implement a transparent and efficient system for managing worker salaries.
- Ensure timely and fair compensation for services rendered.
- Provide a user-friendly interface for both families and workers to handle salary transactions securely.

Attendance Tracking:

- Introduce a feature to track worker attendance for better scheduling and reliability.
- Enable families to monitor and manage worker schedules seamlessly through the platform.
- Enhance the overall efficiency of household staffing by promoting punctuality and dependability.

VIII. Review System

- Integrate a robust review system to allow families to share their experiences with workers.
- Provide a platform for constructive feedback, fostering accountability among workers.
- Assist families in making informed decisions by offering insights from the experiences of others.

IX. REFERENCES

[1]. Peshave, Archana Kherde Ms & Dr. Milind. (2020). "A Study on Challenges Faced by Household Owners Managing Domestic Workers." Unpublished manuscript.

[2]. Attah, F. M., Agba, A. M. O., Ibiam, A. A., Kaburise, P. K., & Kulo, C. (2021). "Information on Domestic staff utilisation and household crimes." Jinav Journal of Information Visualization, 2(2), 61-68. DOI: 10.5281/zenodo.1234567.

[3]. National Skill Development Corporation. (2015). "Human Resource and Skill Requirements in the Domestic Help Sector (2013-17) and (2017-22)."

[4]. Fudge, Judy, & Hobden, Claire. (2018). "Conceptualizing the role of intermediaries in formalizing domestic work." International Labour Office, Inclusive Labour Markets, Labour Relations

and Working Conditions Branch, (95). ISBN: 9789221328313.

[5]. Ganis, Matt. (2010). "Agile methods: Fact or fiction."

[6]. Royce, Winston W. (1970). "Managing the development of large software systems." In Proceedings of IEEE WESCON, (Vol. 8, pp. 328–338). Los Angeles.

[7]. Bell, Thomas E., & Thayer, Thomas A. (1976). "Software requirements: Are they really a problem?" In Proceedings of the 2nd International Conference on Software Engineering, (pp. 61–68). IEEE Computer Society Press.

[8]. Van Casteren, Wilfred. (2017). "The Waterfall Model and the Agile Methodologies: A comparison by project characteristics." Research Gate, 2, 1-6.

[9]. Chakraborty, Abhijit et al. (2012). "The role of requirement engineering in software development life cycle." Journal of Emerging Trends in Computing and Information Sciences, 3(5), 1-6.

[10]. Lemke, Gillian. (2018). "The software development life cycle and its application." Senior Honors Theses & Projects, 589. Retrieved from: [Repository Name].

[11]. Shylesh, S. (2017). "A study of software development life cycle process models." In National Conference on Reinventing Opportunities in Management, IT, and Social Sciences.

[12]. Tuteja, Maneela, & Dubey, Gaurav. (2012). "A research study on importance of testing and quality assurance in software development life cycle (SDLC) models." International Journal of Soft Computing and Engineering (IJSCE), 2(3), 251-257.

[13]. Lv, Qing, Zhang, Suzhen, & Wang, Yuechun. (2022). "Deep Learning Model of Image Classification Using Machine Learning." Advances in Multimedia, 2022, Article ID 3351256, 12 pages. DOI: 10.1155/2022/3351256.

[14]. Al-Saffar, Ahmed Ali Mohammed, Tao, Hai, & Talab, Mohammed Ahmed. (2017). "Review of deep convolution neural network in image classification." 2017 International Conference on Radar, Antenna, Microwave, Electronics, and Telecommunications (ICRAMET). IEEE.

[15]. Liu, Zhizhe, Sun, Luo, & Zhang, Qian. (2022). "High Similarity Image Recognition and Classification Algorithm Based on Convolutional Neural Network." Computational Intelligence and Neuroscience, 2022, Article ID 2836486, 10 pages. DOI: 10.1155/2022/2836486.

[16]. Appalaraju, Srikar, & Chaoji, Vineet. (2017). "Image similarity using deep CNN and curriculum learning." arXiv preprint arXiv:1709.08761.

[17]. Wang, Wei, Li, Yutao, Zou, Ting, Wang, Xin, You, Jieyu, & Luo, Yanhong. (2020). "A Novel Image Classification Approach via Dense-MobileNet Models." Mobile Information Systems, 2020, Article ID 7602384, 8 pages. DOI: 10.1155/2020/7602384.

[18]. Abd-Allah, Ahmed, Gacek, Cristina, Clark, Brad, & Boehm, Barry. (1997). "On the Definition of Software System Architecture."

[19]. Luckham, David C., Vera, James, & Meldal, Sigurd. (1995). "Three concepts of system architecture." Computer Systems Laboratory, Stanford University.

[20]. Li, Y. (2022). "Research and Application of Deep Learning in Image Recognition." In 2022 IEEE 2nd International Conference on Power, Electronics and Computer Applications (ICPECA), (pp. 994-999). Shenyang, China. DOI: 10.1109/ICPECA53709.2022.9718847.

Cite this article as :

Prof. Ratnesh Kumar Choudhary, Prof. Mayuri Botre, Mr. Hukumchand Narwre, Ms. Akanksha Adase, Ms. Achal Jichkar, Ms. Nidhi Patekar, Mr. Jagannath Pradhan, "Transforming Domestic Helper Recruitment and Management with Deep Learning : A Web Application", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 9, Issue 6, pp.333-343, November-December-2023. Available at doi : https://doi.org/10.32628/CSEIT2390643

Journal URL : https://ijsrcseit.com/CSEIT2390643