

Virtual Placement Portal

Mrs. Chethana C¹, Feras Ahmed², Ishita Khetarpal³, R Hariharan⁴

Assistant Professor, CSE, BMSIT&M College, Yelahanka, India¹

Student, CSE, BMSIT&M College, Yelahanka, India²

Student, CSE, BMSIT&M College, Yelahanka, India³

Student, CSE, BMSIT&M College, Yelahanka, India⁴

Abstract: The main aim of this virtual placement portal is to unify the on-campus placement process across various institutes i.e., providing a common platform where all registered colleges can conduct the on-campus placement process virtually. Having moved to complete “virtual” placements for the 2020 batch due to the COVID-19 situation, some of the key issues were observed and hence to address such issues the online placement portal is being developed as solution that not only tackles the noted issues but also bridges the gap between the placement cell of an institute and its students. The portal helps keep the students updated about all companies that visit campuses for placements, directly by the placement cell of the institute with little to no intervention of any intermediary link between the students and the placement cell such as the "placement coordinators".

Keywords: Virtual recruitment, Web portal, Placement- management system, Web-based system

I. INTRODUCTION

The online Placement Portal is a PHP, MySQL based web application project demonstrating the functionalities required to run placements digitally without any paperwork. An aspiring candidate willing to take part in their institute’s placement session can view details of all the jobs posted by the placement cell of an institute, and can apply for the same, provided all eligibility criteria are met. The Placement Portal enables an institute to post details about the companies visiting the campus along with the job profiles offered and other such details pertaining to the job. The institute can also view and maintain information of their students such as the students resumes, 10th/12th marks card, the college marks cards and so on. This project also focuses on providing training materials to the aspiring candidates. This Online web application will be hosted on the cloud which not only makes its accessibility easier but also ensures smooth scaling of infrastructure on the backend as and when required.

Web applications- They are computer programs that utilize web browsers and web technology to perform tasks over the internet. Web applications are usually coded in browser- supported languages like JavaScript and PH. Some web applications use MySQL as its database management system due its speed and flexibility. Commonly used web applications are web-mails, online retail services, online banking.

Cloud applications- Cloud application is an evolved web application. It is equally used to access online services over the Internet like web apps but not always exclusively dependent on web browsers to work. A cloud application, is a software program where cloud-based and local components work together. It is mainly used for its simplified operation, API use, low cost, improved data sharing and security. A well-known cloud application is Drop Box where user can upload and interact with files. With the emergence of cloud computing, computing resources (i.e., networks, servers, storage, applications) are provisioned as metered on-demand services over networks, and can be rapidly allocated and released with minimal management effort. In the cloud computing paradigm, the virtual machine is one of the most commonly used resource carriers in which business services are encapsulated. Amazon EC2 provides resizable compute capacity in the cloud. It is designed to make web-scale cloud computing easier for developers and allows maximum scalability and availability for websites and web applications. Amazon EC2 changes the economics of computing by allowing you to pay only for capacity that you actually use. Our goal with this project is to highly automate the PT (Placement Training) cell. Placement portal is an application developed using HTML, CSS, JavaScript, PHP, MySQL technologies which will be hosted on the AWS cloud. This connects the institute and students where students the institute registers themselves and post jobs using the system which will be visible to all the site users whether registered or not registered. And the students can find and apply for their targeted job, only the registered students can apply for jobs and view the status of applied jobs. The placement cell can view the students resume who have applied for their posted job and accept or reject the application. The status of job application can be viewed by each applicant who has applied for the job. FAQ page can be viewed by all the candidates which will help them solve their query. FAQ will be answered by the placement cell. Students can also get updates about Training and Placement schedules, interviews and selection procedures. There will not be a risk of multiple and fake entries since the student can login with only one account. This makes our application highly reliable for the TPO to use during the entire process of placement in the college.



II. REVIEW OF LITERATURE

[2] In the traditional web hosting, we had a 3-tier architecture which consisted of a presentation layer, and two persistence layers that consisted of the backend servers and the database. To scale these backend layers, we would not only need a large amount of time but would also incur huge costs to provision new servers or upgrade the existing ones. It would also cost us a great deal of time during these updates when the web application would be put to a halt while carrying out these upgrades. There may be situations where we need more computing power for only some small amount of time and the whole computing infrastructure is not in use during the normal operational periods. Hence to overcome these limitations we move to the AWS (Amazon Web Services) cloud. Amazon Web Services (AWS) provides a reliable, scalable, secure, and highly performing infrastructure for the most demanding web applications. This infrastructure matches IT costs with customer traffic patterns in real time.

[9] The key issues tackled by AWS are:

1) A Cost-Effective Alternative to Oversized Fleets Needed to Handle Peaks In the traditional hosting model, you have to provision servers to handle peak capacity. Unused cycles are wasted outside of peak periods. Web applications hosted by AWS can leverage on-demand provisioning of additional servers, so you can constantly adjust capacity and costs to actual traffic patterns.

2) A Scalable Solution to Handling Unexpected Traffic Peaks: An even more dire consequence of the slow provisioning associated with a traditional hosting model is the inability to respond in time to unexpected traffic spikes. There are many stories about web applications going down because of an unexpected spike in traffic after the site is mentioned in the popular media. The same on-demand capability that helps web applications scale to match regular traffic spikes can also handle an unexpected load. New hosts can be launched and ready in a matter of minutes, and they can be taken offline just as quickly when traffic returns to normal.

3) An On-Demand Solution for Test, Load, Beta, and Preproduction Environments the hardware costs of building out a traditional hosting environment for a production web application don't stop with the production fleet. Quite often, you need to create pre-production, beta, and testing fleets to ensure the quality of the web application at each stage of the development lifecycle. While you can make various optimizations to ensure the highest possible use of this testing hardware, a lot of expensive hardware sits unused for long periods of time. In the AWS Cloud, you can provision testing fleets as you need them. Additionally, you can simulate user traffic on the AWS Cloud during load testing. This enables quick switchover from current production to a new application version with little or no service outages.[7]

4) Cloud Security: Two services in particular help with your defense strategy. AWS Shield is a managed DDoS protection service that helps safeguard against various forms of DDoS attack vectors. This standard offering helps to defend against the most common network and transportation layer attacks. In addition to this level, the advanced offering grants higher levels of protection against your web application. [6] If we focus on the security aspect, one can get access to the AWS DDoS Response Team (DRT) to help mitigate large-scale and sophisticated attacks against your resources. AWS WAF works in line with CloudFront or Application Load Balancer, along with your custom rules, to defend against attacks such as cross-site scripting, SQL injection, and DDoS. As with most AWS services, AWS WAF comes with a fully featured API that can help automate the creation.

The client side of the project is a web browser which interprets the results sent by the server in a format which is understood by the user. The web browser development is outside the scope of this project, but what reaches the browser, that is the responses sent by the server is what we'll be touching upon in the following section. When a request is made to the server the server not only fetches the requested pages/documents, but also may perform some computation to generate dynamic results. PHP has remained one of the most versatile and pragmatic web development languages in the world today. Its range of functionalities, amazing array of add-ins to extend functionalities, its open-source nature, and tremendous online community support has made PHP a perennial favorite amongst newbies as well as established development agencies worldwide. This is commonly referred to as server-side programming where the computational logic is devised so as to provide dynamic responses as and when needed. PHP is a server-side programming language. PHP & MySQL is a great combination for creating simple as well as enterprise grade web applications that are secure, fast and scalable.

[1] There are many compelling reasons to use PHP to create your web application. Much of the syntax of PHP will be familiar to those who have experience in C and C++, because PHP was evolved from these languages. PHP is also closely related to MySQL, the most popular choice to create and access databases. PHP has been carefully developed to offer more advanced capabilities than C, but with greater emphasis on the commands that are to be used for web development. Microservices architecture is a distributed design approach intended to overcome the limitations of traditional monolithic architectures. Microservices help to scale applications and organizations while improving cycle times. However, they also come with a couple of challenges that might add additional architectural complexity and operational burden. AWS offers a large portfolio of managed services that can help product teams build microservices architectures and minimize architectural and operational complexity.



III. PROBLEM DEFINITION

Currently either there are institutions that have placement portals only specific to them or we have institutions that don't have any provision as such. Those who do have it, have mostly implemented it on on-premises infrastructure that can bear workloads only pertaining to their institutes. These portals don't provide the source to various placement training materials as well. 2. The generalized "job portals" out there mainly focus on the off-campus jobs which are directly posted by the recruiting companies. The generalized existing systems are either institute oriented or candidate oriented, not a system that balances the two perfectly. 3. Monolithic approach has a limitation in size and complexity, the size of the application can slow down the start-up time and they can also be difficult to scale when different modules have conflicting resource requirements. Another problem with monolithic applications is reliability. Monolithic applications have a barrier to adopting new technologies. For institutes without a portal of their own, they manage the entire process manually by usually setting up a team of students known as the placement coordinators. For carrying out tasks like broadcasting notices, groups are used which sends emails to the entire group. Placement coordinators have to make groups and they are asked to send the information to the students in the group. If the subscription is not successful the student will not be able to enroll for activities and will not receive notices. Verifications have to be to see if the numbers or emails they have provided are their primary numbers or emails. For maintaining the students record excel sheets are used for each and every activity separately and it does not allow easy automatic computation.

VI. IMPLEMENTATION METHODOLOGY

This project has 3 modules.

1) Log-in/Registration:

Initially an organization must register. Once done the students of that organization can register too. It is only after the registration of an organization that students of that organization can register as well.

2) Organization's Page:

The placement cell of the organization or any designated person who is utilizing our service can do the following: View the list of registered students under their organization, Post job applications and allow the interested candidates to apply for the same, Mark candidates as placed/unplaced so as to disallow candidates of the organization from applying for jobs when they are already placed, View and verify candidate details such as resumes, 10th/12th marks cards and college marks cards, post any other required notifications to keep students notified or any events.

3) Candidates Page:

The candidate of any registered organization can utilize our services as follows: View job postings and apply for those eligible. Upload and update resumes or marks cards online so as to make it easier for the placement cell of the organization to have up to date valid information about the candidate they can view the FAQ page and post their query on that. They can get placement training on the same platform.

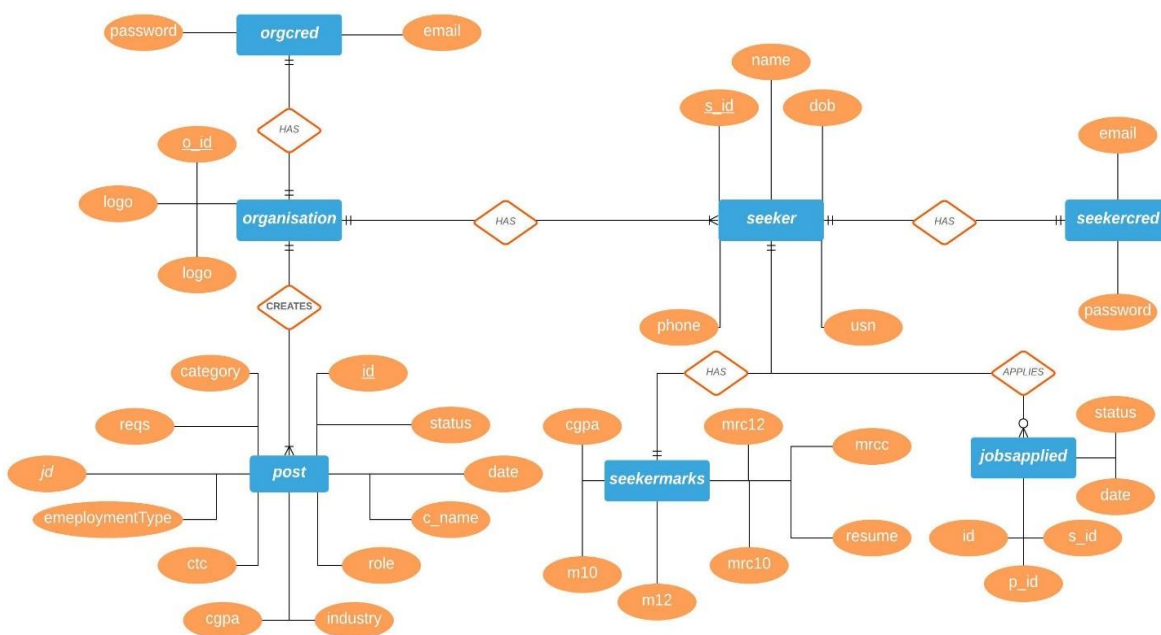


Fig. 1. ERD Diagram

System Architecture: There are 2 sides namely Client side and the server side. The client will access the web application through a web browser where he/she can use the services from this project. The server side will handle the incoming requests and perform the required computations.

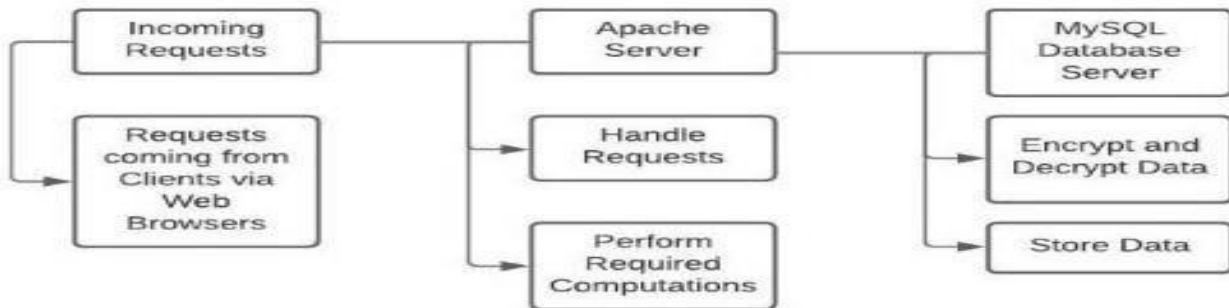


Fig.2. Server side

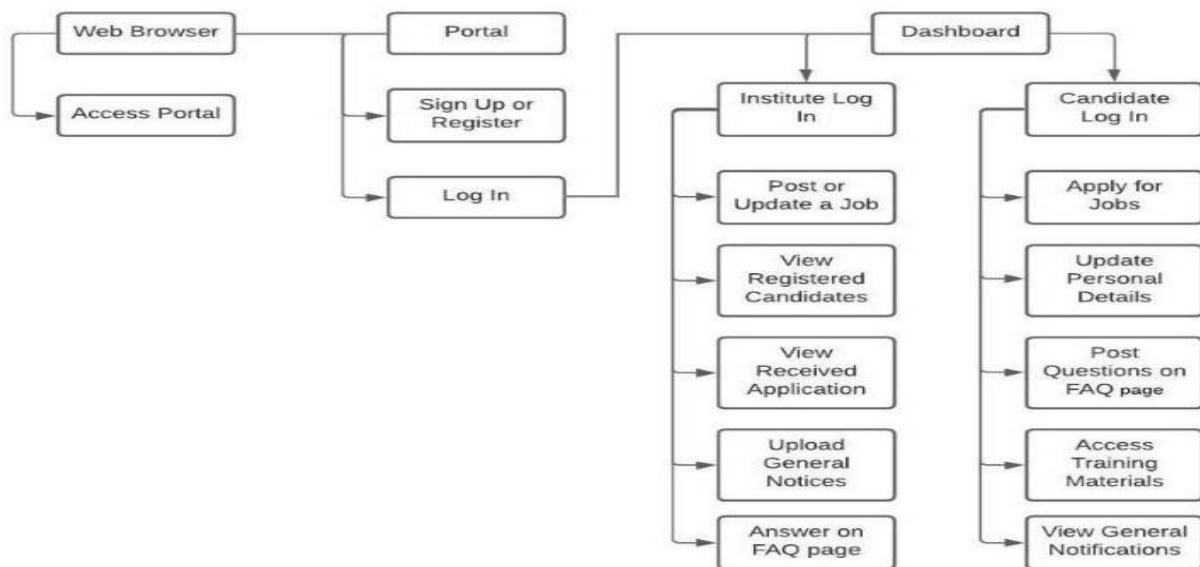


Fig. 3. Client side

V. CONCLUSION

This project provides a uniform platform that allows various institutes to carry out their placement session with ease. A platform that keeps candidates updated at all times about the placement activities in their institutes and helps clarify all their doubts pertaining to placements with the help of a common FAQs page common to all. It provides a fast and responsive microservice web application on cloud ensuring strong backend security to keep all data safe and confidential.

ACKNOWLEDGEMENT

The authors would like to thank the editor, mysterious reviewers for their valuable suggestions that appreciably improved the quality of this paper. We would also like to thank our parents, friends and our guide for their constant encouragement and providing us overwhelming support and guidance to write this paper.

REFERENCES

- [1] V. Yadav, U. Gewali, S. Khatri, S. R. Rauniyar and A. Shakya, "Smart Job Recruitment Automation: Bridging Industry and University," 2019 Artificial Intelligence for Transforming Business and Society (AITB), Kathmandu, Nepal, 2019.
- [2] N. Mangla, J. Singh and M. Singh, "Improving performance of web applications using cloud resources," Proceedings of 3rd International Conference on Reliability, Infocom Technologies and Optimization, Noida, 2014
- [3] M. Villamizar et al., "Evaluating the monolithic and the microservice architecture pattern to deploy web applications in the cloud," 2015 10th Computing Colombian Conference (10CCC), Bogota, 2015, pp. 583-590, doi: 10.1109/ColumbianCC.2015.7333476.
- [4] D. Fortunato and J. Bernardino, "Progressive web apps: An alternative to the native mobile Apps," 2018 13th Iberian Conference on



- Information Systems and Technologies (CISTI), Caceres, 2018, pp. 1-6, doi: 10.23919/CISTI.2018.8399228.
- [5] A. Weber and H. Mahringer, "Choice and success of job search methods," Empirical Economics, vol. 35, no. 1, pp. 153-178, 2018.
- [6] Mansourvar, Marjan & Mohd Yasin, Norizan. Development of a Job Web Portal to Improve Education Quality. International Journal of Computer Theory and Engineering. 43-46. 10.7763/IJCTE. 2014.V6.834, 2014.
- [7] K. Kopuri, G.M.H. Aqueel, A.S. Jabeen and T.K. Shaik Shavali, "An Online Job portal management system", International Journal of Innovative Research in Technology, vol. 3, no. 9, February 2017.
- [8] Developer guide from
<https://docs.aws.amazon.com/waf/latest/developerguide/what-is-aws-waf.html>
- [9] Web Application Hosting in the AWS Cloud: Best Practices,
<https://docs.aws.amazon.com/whitepapers/latest/web-application-hosting-best-practices/>
- [10] Documentation from
<https://www.apachefriends.org/dos/hosting-xampp-on-aws.html>

AUTHORS PROFILE

- 1) **Mrs. Chethana C**, Assistant Professor
BMSIT&M College, Yelahanka, India
- 2) **Feras Ahmed**, Student
BMSIT&M College, Yelahanka, India
- 3) **Ishita Khetarpal**, Student
BMSIT&M College, Yelahanka, India
- 4) **Hariharan R**, Student
BMSIT&M College, Yelahanka, India