

Efficient Platform to Hire the Labour Normally in Traditional Way Through Mobile Computing

Prof. Krishna Kulkarni
CSE
ICOER JSPM, Pune

Mr. Sumit Yadav
CSE
ICOER JSPM, Pune

Mr Pranit Malusare
CSE
ICOER JSPM, Pune

Miss. Shraddha Raut
CSE
ICOER JSPM, Pune

Abstract— The “Labour Hiring Application” is an android application which is helpful for uneducated and illiterate people as well as organizations or contractors. In the existing system, all the activities are done by the brokers. It is very much costly and time-consuming. In our proposed system, labour and contractors can log in in our app to hire or to get hired. It’s format is as simple which will be understood by labour. The data will be stored in the database. The common people or contractor can log into their

account through the app itself and update information or requirement. In our system, common people or contractor can view labour’s details such as type, cost, location etc. Other than this the advanced features are: Our system has a facility for people who don’t have an android phone we are going to provide the facility like registering them through the toll-free no so everyone can take advantage of our app.

Keywords:- (1)Labour, (2)Customer, (3)Interface, (4)Android, (5)Open-Source, (6)Software Development Kit

I. INTRODUCTION

In today’s world mobile and mobile application plays an important role in our life. All our daily works done by mobile phones, considering this we have created a mobile application to hire labour who work on daily work and wages. As we know this people is an important role in the suburb area. Basically using our application user can hire labour from nearby area. For this user and labour, both need to register themselves on the application using their basic details such as name, number, address, gender etc further labour need to fill his charges also. If any user doesn’t have an android phone he/she can register themselves from their normal phone using provided toll-free number. When any labour register, their individual work profile should get created. The labour should not need to go so far for work, in short, it should be on real-time location. Contractor adds all the requirements and hire employees on the basis of these aspects. Labour should also find their choice of work. Each worker should pay through this application.

II. LITERATURE SURVEY

In order to overcome inefficient traditional way of labours hiring and its management, we developed an efficient

platform to maintain and make it digital way. To hire and manage labours manually in the traditional way is very difficult and time consuming process. Many applications have been developed for labour management but also has its own drawbacks. All those having their advantages and disadvantages, Labour Hiring through digitally is the best replacement to bulky, time consuming manually hiring of labours those who are working on regular wages. [1]

In the current system the hiring process is mostly about the educated peoples. There is no digital system for the labours which has daily basis income and that labours do not get easily access or information about the digital platform so we try to provide easy and safe digital hiring platform for all type of the labours. [2][3]

III. EXISTING SYSTEM

When our team studied this system then we found that this is totally Unorganized sector. In that Capitalist hired worker through the broker, there is no direct contact between capitalist and worker. The worker decides all details with the broker, so there is a lack of communication between worker and Capitalist. Due to that both worker & Capitalist faced many problems as we discussed earlier. The existing system the individual profile of the labour should not get maintained. If we maintain the profile for each labour it should contain all records and his

ratings as per his work. Existing system is not that much user friendly. Existing systems have some drawbacks. The existing application compulsorily needs android mobile phones for operating application. If there is no android phone labour should not be able to use it. It does not have any other registration option like toll-free number. There is no facility of payment processes as per the work done by labour. Also, it should not contain the time record of labour that how much time he gave for work. In/her work. Existing system should not have real-time location for labour. He/She needs to go for the allocated location for there work.

IV. PROPOSED SYSTEM

The system architecture has smartphones with Android OS. The android smartphones or Tablet must use 4G or Wi-Fi network for internet connectivity to ensure better performance however 3G network also comes in use but this will add the disadvantage like time-lag. Otherwise it will be ok with 3G network.

The labour as well as contractor or capitalist will log in to the our system through Android smartphones. Customer can see all the details of the particular labour through this application. Multiple registrations can be done in-app. According to detail the constructor, organization or a common people will approach labour via SMS with all details in it. We have added many features in it Such as chatbox where labour and contractor or common people can communicate with each other to remove the any kind of confusion between them , using any language as we include all Indian language in it. We also set a Google language translator for those who cannot understand the English language. The application offers some features like trust ,safety and reliability. This application help labour to get the job on click as well as all other people get the labour on one click. We also provide facilities for labours those who don't have an android phone and this facility is like we are going to register them through the toll-free number. This could be very useful to the labours those are illiterate.and especially those labours who doesn't have the android phone.

The scheme uses the profile of each and every user maintained by CSP. This scheme is known as Profile Based Access Control Model (PrBAC). Profile creation is based on the content. CSP will create a list (CSPLST) with the profile of users. This list is a temporary list for fast data access. In CSPLST there are five attributes namely user group (USRGRP), user id (USRID), data, USRGRP time and date, user time and date. When the worker starts his allocated work, he has to press the start button on the app, so the contractor will receive notification about the work of labour. After completion of work, labour has to press the finish button so the contractor will get the correct information about time and work. This will bring fluency between labour and contractor detail in below sections. And it is also worth mentioning that many games and various hiring applications using this searching algorithm as their main center technique for searching.in our application we are using this technique for searching labours as per contractor wants.this would really helps us to find nearest location labours as well as contractor

V. IMPROVED A* ALGORITHM

A* Search algorithm is one of the best and popular techniques used in path finding and in our system we are going to use this algorithm for finding labour which is located at the nearest location.

The pseudocode of A algorithm is as follows:*

```
function find_path( Source, Destination){  
    OPEN_List = [Source];
```

```
    CLOSED_List = [];  
    Current_Node = null;  
    Neighbours = []; Path = [];  
    Source.Parent = null;  
    while ( OPEN_List is not Empty ) {  
        Current_Node = OPEN_List.remove_least_node();  
        if( Current_Node == Destination )  
            break;  
        CLOSED_List.add(Current_Node);  
        Neighbours = Map.search_neighbours(  
Current_Node );  
        foreach (n in Neighbours) {  
            examine(n);  
            n.Parent = Current_Node;  
            OPEN_List.add(n);  
        }  
    }  
    while( Current_Node is not null ) {  
        Path.add( Current_Node );  
        Current_Node = Current_Node.Parent;  
    }  
}
```

WHY A* SEARCH ALGORITHM ?

Informally speaking, A* Search algorithms, unlike other traversal techniques, it has “brains”. What it means is that it is really a smart algorithm which separates it from the other conventional algorithms. There are a number of simple optimizations or implementation details that can significantly affect the performance of an A* implementation. The first detail to note is that the way the priority queue handles ties can have a significant effect on performance in some situations. If ties are broken so the queue behaves in a LIFO manner, A* will behave like depth-first search among equal cost paths (avoiding exploring more than one equally optimal solution).

When a path is required at the end of the search, it is common to keep with each node a reference to that node's parent. At the end of the search these references can be used to recover the optimal path. If these references are being kept then it can be important that the same node doesn't appear in the priority queue more than once (each entry corresponding to a different path to the node, and each with a different cost). A standard approach here is to check if a node about to be added already appears in the priority queue. If it does, then the priority and parent pointers are changed to correspond to the lower cost path. A standard binary heap based priority queue does not directly support the operation of searching for one of its elements, but it can be augmented with a hash table that maps elements to their position in the heap, allowing this decrease-priority operation to be performed in logarithmic time. Alternatively, a Fibonacci heap can perform the same decrease-priority operations in constant amortized time.

Algorithm A is optimally efficient with respect to a set of alternative algorithms **Alts** on a set of problems **P** if for every problem P in **P** and every algorithm A' in **Alts**, the set of nodes expanded by A in solving P is a subset (possibly equal) of the set of nodes expanded by A' in solving P. The definitive study of the optimal efficiency of A* is due to

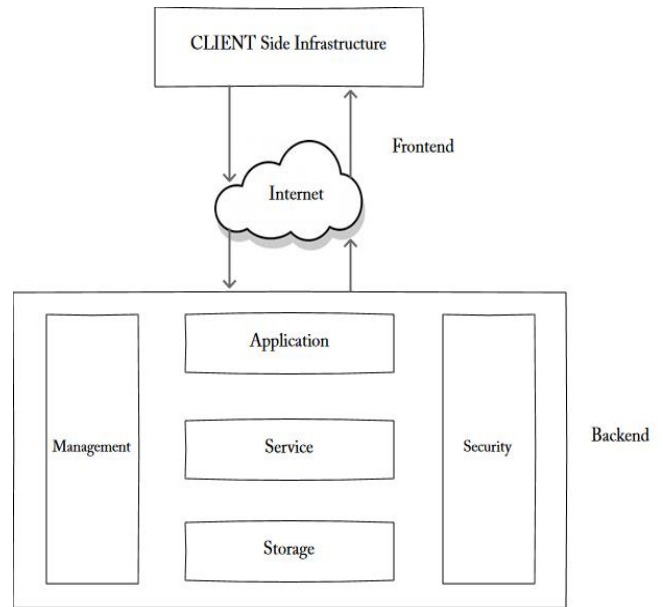
Rina Dechter and Judea Pearl.^[10] They considered a variety of definitions of **Alts** and **P** in combination with A*'s heuristic being merely admissible or being both consistent and admissible. The most interesting positive result they proved is that A*, with a consistent heuristic, is optimally efficient with respect to all admissible A*-like search algorithms on all "non-pathological" search problems. Roughly speaking, their notion of non-pathological problem is what we now mean by "up to tie-breaking". This result does not hold if A*'s heuristic is admissible but not consistent. In that case, Dechter and Pearl showed there exist admissible A*-like algorithms that can expand arbitrarily fewer nodes than A* on some non-pathological problems.

Optimal efficiency is about the *set* of nodes expanded, not the *number* of node expansions (the number of iterations of A*'s main loop). When the heuristic used is admissible but not consistent, it is possible for a node to be expanded by A* many times, an exponential number of times in the worst case. In such circumstances Dijkstra's algorithm could outperform A* by a large margin.

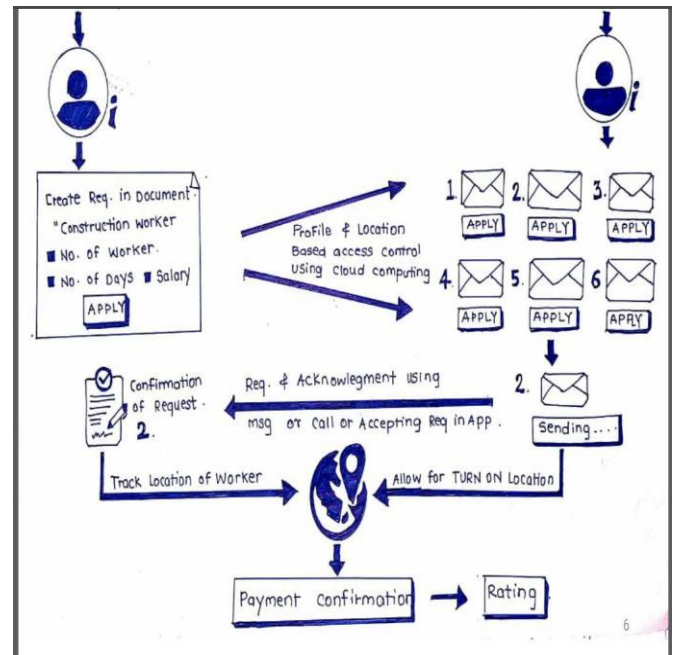
STRUCTURE OF MOBILE COMPUTING

SYSTEM Mobile cloud computing can be seen that the definition of a cloud computing of commercial calculation model is applied in mobile environment. On-line translation system front end which is designed and realized in this paper based on mobile devices, mobile internet and cloud computing of facing mobile internet is mutually assisted. Software and hardware resources obtain unprecedented intensive application through cloud computing technology and people can reach functions of traditional PC on mobile terminal. Due to its portability, small volume, etc, although mobile phone evolves to intelligence in various aspects, it cannot be compared to computer on its processing ability within a short time. Therefore, mobile internet can perfectly interpret and embody features of cloud computing, store, process and apply based on server of cloud computing mode so as to weaken processing requirement of mobile terminal device.

At background, storage and computing ability of cloud calculation also solve the limitation of storage capacity and information loss. With appearance of some classical internet cloud computing application, cloud computing is gradually transiting from internet to mobile internet and it is a bit different between mobile devices accessing cloud computing and fixed equipment



VI. SYSTEM ARCHITECTURE



VII. CONCLUSION

Our team completed the raw structure of the flow of the application according to various validation and authentication. We need Cloud requirement for the storage, real-time profile and location access control. Track geolocation of the worker by use of cloud computing. Our team get a feasible solution on two requirement i.e storage and real-time profile-based access control. Now currently we working on the real-time location-based access control and geo-location of the worker using cloud

VIII. REFERENCE

1. Labour Management Using Android Application Reena Chaudhari 1 , Ketan Gurav2 , Rajesh jaljal3, Akshay Kshirsagar4 |Prof. Reena Chaudhari, Dept of Computer Engineering, SSJCOE, Dombivli, India

- 2Ketan Gurav, SSJCOE, Dombivli, India 3Rajesh jaljal, SSJCOE, Dombivli, India 4Akshay Kshirsagar, SSJCOE, Dombivli, India
2. Profile Based Access Control Model in Cloud Computing Environment Suyel Namasudra¹ , Samir Nath² , Abhishek Majumder³ Department of Computer Science & Engineering Tripura University (A Central University) Suryamaninagar, Tripura West, India.
 3. K-anonymous Location-based Fine-grained Access Control for Mobile Cloud Yaser Baseri Department of Computer Science and Operations Research Universite de Montreal Abdelhakim Hafid Department of Computer Science and Operations Research Universite de Montreal,Soumaya Cherkaoui INTERLAB Research Laboratory Universite de Sherbrooke.
 4. Mobile-Based Location Tracking without Internet Connectivity using Cloud Computing Environment 1 Qassim Bani Hani 2 Julius P. Ditcher Department of Computer Science & Engineering University of Bridgeport, CT-06604, USA 1 qbanihan@my.bridgeport.edu
 5. Design and Implementation of Online English Translation System Based on Cloud Computing Model Jiang Xue North China Electric Power University Foreign Languages Department, Beijing 102206, China Jiangxue20032000@163.com
 6. Secure Framework for Data access Using Location Based Service in Mobile Cloud Computing Deepanshu Goyal University School of Communication and Technology Guru Gobind Singh Indraprastha University, New Delhi, India E-Mail: deepanshu_cs@live.com M. Bala Krishna University School of Communication and Technology Guru Gobind Singh Indraprastha University, New Delhi, India E-Mail: mbalakrishna@ipu.ac.in
 7. D. P. Gadekar, N. P. Sable, A. H. Raut, "Exploring Data Security Scheme into Cloud Using Encryption Algorithms" International Journal of Recent Technology and Engineering (IJRTE), Published By:Blue Eyes Intelligence Engineering & Sciences Publication, ISSN: 2277-3878, Volume-8 Issue-2, July2019, DOI: 10.35940/ijrte.B2504.078219, SCOPUS Journal.
 8. Sable Nilesh Popat*, Y. P. Singh," Analysis and Study on the Classifier Based Data Mining Methods" in *Journal of Advances in Science and Technology | Science & Technology*, Vol. 14, Issue No. 2, September-2017, ISSN 2230-9659.