POST-COVID REVIVAL OF CONSTRUCTION SECTOR USING GEOSPATIAL TECHNOLOGIES

A DISSERTATION

SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS

FOR THE AWARD OF THE DEGREE

MASTER OF TECHNOLOGY

IN

CIVIL ENGINEERING

(With Specialization in Geoinformatics Engineering)

BY

VISHAL SHUKLA

(2K19/GINF/02)

Under the supervision of

DR. (COL) K.C. TIWARI, PROFESSOR

MULTIDISCIPLINARY CENTRE FOR GEOINFORMATICS

DEPARTMENT OF CIVIL ENGINEERING

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Bawana Road, Delhi – 110042

August 2021

M.TECH (GEOINFORMATICS)

VISHAL SHUKLA

2021

MULTIDISCIPLINARY CENTRE FOR GEOINFORMATICS
DEPARTMENT OF CIVIL ENGINEERING
DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi – 110042

CANDIDATE’S DECLARATION

I, Vishal Shukla, Roll No.- 2K19/GINF/02 of M.Tech (Geoinformatics Engineering), hereby declare that the project Dissertation titled “*Post Covid Revival Of Construction Sector Using Geospatial Technologies*” which is submitted by me to the Multidisciplinary Centre for Geoinformatics, Department of Civil Engineering, Delhi Technological University, Delhi in partial fulfilment of the requirement for the award of the degree of Master of Technology, is original and not copied from any source without proper citation. This work has not previously formed the basis for the award of any Degree, Diploma Associateship, Fellowship or other similar title or recognition.

Vishal

Place: Delhi VISHAL SHUKLA

Date: 23 August, 2021

MULTIDISCIPLINARY CENTRE FOR GEOINFORMATICS
DEPARTMENT OF CIVIL ENGINEERING
DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi – 110042

CERTIFICATE

I hereby certify that the Project Dissertation titled "*Post Covid Revival Of Construction Sector Using Geospatial Technologies*" by Vishal Shukla, Roll No.-2K19/GINF/02, Multidisciplinary Centre for Geoinformatics, Department of Civil Engineering, Delhi Technological University, Delhi in partial fulfilment of the requirement for the award of the degree of Master of Technology, is a record of the project work carried out by the student under my supervision. To the best of my knowledge this work has not been submitted in part or full for any Degree or Diploma to this University or elsewhere.

Place: Delhi

Date: 23 August, 2021

SUPERVISOR

Dr. (Col) K.C TIWARI, Professor
Multidisciplinary Centre for Geoinformatics
Department of Civil Engineering
Delhi Technological University
Delhi - 110042

ACKNOWLEDGMENTS

I take this opportunity to express a deep sense of gratitude towards my guide **Dr K.C Tiwari** (Professor, Multidisciplinary Centre for Geoinformatics(MCG)), for giving me outstanding direction, support, and inspiration throughout the process,. Without his assistance, this initiative would not have been a success. I was fortunate to have a guide who gave me the freedom to explore while also assisting me when my feet faltered. I am grateful for his confidence in me.

I would like to express my sincere gratitude to Mr Gopinadh Rongali, a member of the Multidisciplinary Centre for Geoinformatics (MCG) Lab at Delhi Technological University, for his support and helpful feedback on the project. I would also like to thank all the staff and members of the Department of Civil Engineering, HOD (Civil) and the whole fraternity of Delhi Technological University for providing me with the opportunity to study in this prestigious institute and all the resources needed during this research.

Finally, I would like to express my heartfelt thanks to my beloved parents for their blessings, and also like to thank all of my classmates for their insightful suggestions and lively debates for the successful completion of this project work.

ABSTRACT

When evaluated in the context of migration, the economic crisis triggered by COVID-19 might be protracted, deep, and extensive. Global economic activity came to a halt during the most period of 2020-21 as a result of lockdowns, travel prohibitions, and social isolation. These implications are especially problematic for low-income households, which are ill-prepared to deal with wage losses during a recession, lack other sources of income, and lack access to social security. The majority of these people are paid just enough to get by, and they have no other means of assuring their financial stability if they lose their employment.

Due to the lockdown and the resulting threat of recession, millions of migrant workers have lost their jobs in India during this period of Covid lockdown. Many migrant labourers have returned to their towns, and much more anxiously awaited the lifting of the lockdown. The risk was particularly higher for those who have been working in the construction sector. Migrant workers are facing difficulties in finding work in the present situation. A survey of construction framework reveals that there is no common registration platform for employers (contractors), migrant workers and the agent so that they can communicate with each other directly. To solve the above-said problem, this research proposes one of the possible i.e. GIS-based e-portal. This study will also show how the application of geospatial data and location intelligence may revolutionise the construction industry.

“Migrant e-portal” aims to provide a solution to this problem by bringing together employers (contractors), migrant workers and the agent on the same platform for the mutual benefit of all. In this “Migrant e-portal” the geographical location of the migrant workers and location of the workplace will be used for geospatial analysis. It will provide additional information like shortest path distance, information regarding the toll, optimal route. An e-portal is the cheapest

and quickest way to communicate, reaching the great majority of people with just one click, regardless of their location.

This WebGIS application "Migrant e-portal" provides a simple and intuitive platform for migrant workers to identify job openings in the construction sector that match their search criteria based on their geolocation, and it also assists them in finding opportunities in their native place with the help of geospatial analysis based information provided to them using their geographical location. Migrant workers can register in the application and establish their profile to apply for employment. Migrant workers can use advanced search options to look for employment in the construction field. Employers (i.e. contractors) can sign up, create their profile and can post the jobs. They can look at the applied job application and screen them for the best match. All this features combine with the geospatial analysis based information mentioned above will enable the easy facilitation providing jobs to the migrant workers.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

ILO - International Labour Organization

ADB - Asian Development Bank

GDP - Gross Domestic Product

PWD – Public Works Department

MVT - Model-View-Template

CSS - Cascading Style Sheet

HTML - Hypertext Markup language

AJAX - Asynchronous JavaScript and XML

SQL - Structured Query language

AP- Arunachal Pradesh

AS- Assam

BR- Bihar

CG- Chhattisgarh

PB- Punjab

HP- Himachal Pradesh

GJ- Gujarat

HR-Haryana

KA- Karnataka

UP- Uttar Pradesh

JH- Jharkhand

KL- Kerala

MH- Maharashtra

MP- Madhya Pradesh

OD- Odisha

DL-Delhi

RJ- Rajasthan

TN- Tamil Nadu

UK- Uttarakhand

JK- Jammu & Kashmir

ZIP- Zone Improvement Plan

INR- Indian Rupee

UPI- Unified Payments Interface

CHAPTER – 1

INTRODUCTION

1.1 GENERAL

After the agricultural industry, the construction sector in India is the second most important sector in the country which makes use of the maximum nation's workers force. Mostly it contains unregistered and unskilled workers, who work on daily wages. The movement of people from one location to another is known as migration. It can be permanent, temporary, or seasonal. The main reason for their migration which may be in the form of intra-state or inter-state is to get work regularly so that they can survive easily and take care of their families. But due to COVID-19 pandemic forced the government to apply lockdown to all the states of India due to which migrant workers were forced to leave the workplace. Lockdown affects migrant workers very harshly as a majority of them are unemployed. The construction sector is also going through a bad phase because most of them are dealing with the worker scarcity. This present thesis explores the adverse COVID-19's impact on the construction sector and it presents a possible solution to address these problems using geospatial technologies.

1.2 COVID-19 PANDEMIC AND ITS EFFECTS

COVID-19 was originally detected as a respiratory tract infection in Wuhan, China, in December 2019. Symptoms include fever, chills, dry cough, exhaustion, and shortness of breath [1]. This rare viral pneumonia has rendered the entire world inoperable, causing catastrophic health and economic costs. SARS and MERS-CoV are related to the new coronavirus, although the former's impact is more catastrophic, as evidenced by the exponential rise in infected cases [2]. According to the World Health Organization's International Health Regulations (2005), COVID-19 was declared a Public Health Emergency of International Concern by the end of January [3].

Due to the virus's extraordinary spread, the world has been put on lockdown, with numerous governments instituting tight screening of possible cases admitted into their territory [4]. The economic restrictions and closures resulted in massive reverse migration from all of the country's main cities. Migrant workers faced several problems during the COVID-19 pandemic.

The most significant obstacle to a productive reaction in India in the fight in opposition to emerging pandemic is a a lack of medical investment and infrastructure in the healthcare sector. Amid the rising number of cases of lack of healthcare facilities, such as a lack of beds and protective equipment [5].

Figure-1: Patients with coronavirus illness are treated at the Lok Nayak Jai Prakash (LNJP) hospital's casualty ward. (Source: An article on ‘Business Standard’, April 16, 2021)

India's GDP fell by 23.9 per cent in the first quarter of the fiscal year 2021 as a result of the coronavirus quarantine. The pandemic has forced businesses, industries, and services to shut down in unprecedented numbers [6]. Approximately 21 million employed people between April and August, people lose their employment, according to the Indian Economy Monitoring Centre (CMIE). 86 million paid jobs were available in India in 2019-20. According to the CMIE, the total went down to 65 million in August 2020, following the loss of 3.3 million jobs in that month, primarily among industrial and white-collar employees [7].

When enterprises and businesses were shut down owing to the country's lockdown, millions of migrant workers suffered a loss of income, food shortages, and uncertainty about their future. According to a joint assessment by the International Labour Organization (ILO) and the Asian Development Bank, the COVID-19 epidemic resulted in the loss of jobs for up to 41 lakh young people in the country, with construction and agriculture workers accounting for the bulk of job losses (ADB) [8].

1.3 EFFECTS OF REVERSE MIGRATION ON THE CONSTRUCTION SECTOR

Many migrants and informal workers residing in cities have returned with their families to their homes in towns and villages across India since the start of the COVID-19 crisis and the subsequent state-wide shut down in March. This COVID-19-caused reverse movement was the second-largest mass exodus in Indian history, after the Partition, which moved more than 14 million people, though exact figures are unknown [9]. Figure-2 represents the scenario of chaos made during the announcement of lockdown by the government.

Figure-2: Migrant labourers and their families queue at a bus terminal in New Delhi, trying to catch a bus back to their villages. (Source: Bhuvan Bagga/AFP via Getty Images, March 31, 2020)

With thousands of migrant workers returning to their hometowns, India's construction and real estate industries are facing a workers shortage [10]. The construction sector employs thousands of migrant workers, including masons, mason helpers, painters, granite workers, carpenters, and other related jobs.

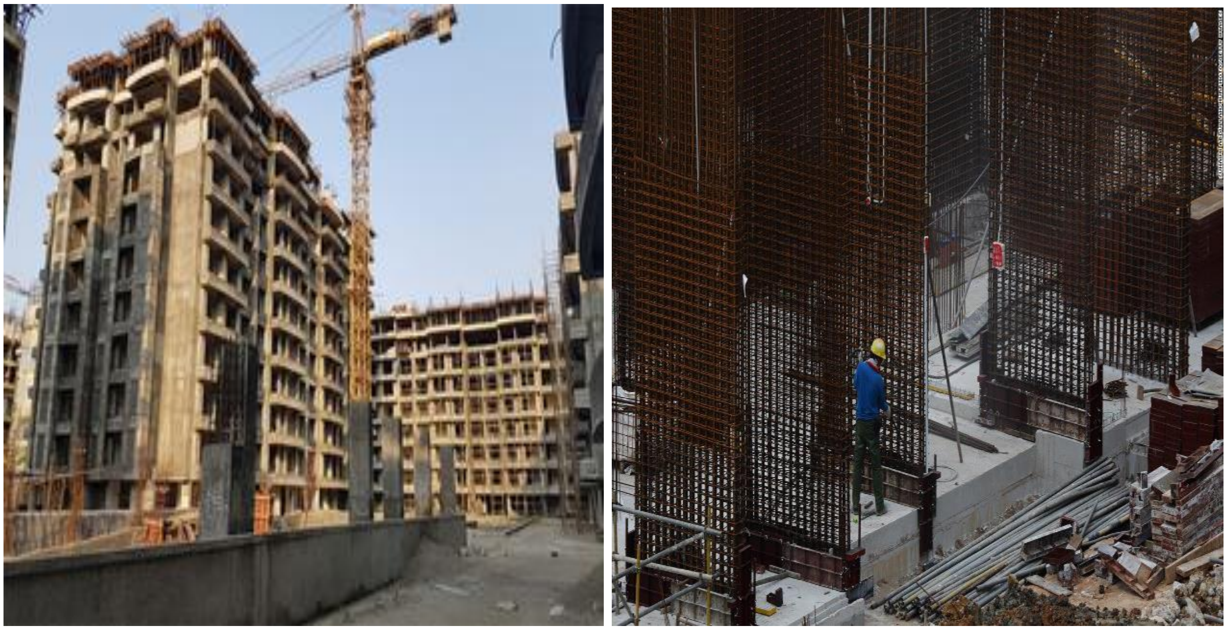


Figure-3: Lack of availability of workers on construction site. (Source: An article on “Economic Times”, November 21, 2020)

When the pandemic's terror overtook them, many chose to flee. Builders and contractors are having a difficult time planning their site work since workers' availability are unknown which can be depicted in Figure-3. This will undoubtedly disrupt the delivery timetable and cause construction to be delayed. The Covid-19 lockout and subsequent reverse migration of workers is a double whammy for an already struggling economy [11].

1.4 RESEARCH GAPS

In India, migrant workers play an important role in the construction sector. The government has formed various departments like Labour Department to handle all the migrant workers and the state-wise contractor department to handle all the contractors but there are some deficiencies in these departments which needed to be addressed and needed to be solved. The various deficiencies are pointed below:

a) There is no formal framework in place to assist migrant workers in finding jobs in the construction sector inside their state or in nearby areas to limit their movement.
b) No e-platform helps the migrant workers, agents and employers (contractors) to communicate directly.
c) There is no proper geospatial-based framework to aid migrant workers in their job searches in the construction sector.
d) Many people have lost their jobs as a result of COVID-19. However, the situation is again returning to normal, yet there are still many jobless individuals looking for work in the construction sector. There is a state-wise labour department but their main focus is to provide information to workers about the new and existing schemes ("yojanas") that the government is launching or which they have launched earlier in the construction sector. There is no adequately structured and legitimate platform that assists migrant workers in resolving their issues in the construction sector.

1.5 OBJECTIVES

The study's objectives are listed below:

a) Study and analyse the current framework of construction in the country.
b) Development of a website to register potential/returned migrants with their location data and facilitate distance and skill-based matching with the potential employers/ industries using geospatial technologies.

1.6 ORGANISATION OF THESIS

Chapter-1 provides the motivation behind the research, it highlights the problem area in the construction sector. Chapter-2 provides an overview of migration, the pattern of migration, how COVID affects the migrant workers of the construction sector, a framework of the construction sector, its limitations. Chapter-3 provides the reason for using geospatial techniques and the type of algorithm/metrics that can be used for geospatial analysis in the construction sector. Chapter-4 provides information about the data collection, technologies used and the methodology to complete Objective II. Chapter-5 provides the implementation and results as per methodology developed in Chapter -4.

CHAPTER-2

REVIEW OF THE CONSTRUCTION FRAMEWORK IN INDIA IN LIGHT OF COVID-19 AND REVERSE MIGRATION

2.1 COVID-19 AND REVERSE MIGRATION

The movement of people away from their usual residence across national (internal) or international (cross-national) borders is referred to as migration [12]. All citizens of India are entitled to free movement under the Indian Constitution. Article 15 forbids discrimination based on birthplace, as well as others things, Article 16 on the other hand in sectors of public work, it ensures equal opportunity for all people, and it specifically forbids discrimination in public employment based on birthplace or residency. [13]. 100 million people, or one in every ten Indians, work as seasonal/circular/short-term migrants in metropolitan India [14]. The 2011 Census provided the most recent government data on migration. India has 45.6 million migrants (38 percent of the population) according to the 2011 Census, up from 31.5 million migrants in 2001. (31 per cent of the population). Between 2001 and 2011, the population increased by 18%, but the number of migrants increased by 45 percent. [15]. COVID-19, on the other hand, has had a significant impact on migrant workers. According to the Stranded Workers Action Network (SWAN), more than half of workers had less than one day's ration left; 74 percent having a daily wage of less than half of what they had before left to subsist for the rest of the shutdown; and 89 percent had not been paid at all amid the lockout by their superiors [16].

2.1.1 The Pattern of Migration in India

In India, the flow of internal migration can be classified based on both the origin and the destination. It can be classified as:

a) rural-urban
b) rural-rural
c) urban-rural
d) urban-urban

According to the 2011 census, there were around 21 crores, rural-rural migrants, with approximately 8 crore rural-urban and urban-urban migrants apiece. Around 3 crore people migrated from the city to the countryside [15]. According to the National Sample Survey Organization, both overall employment in the organised and unorganised country's sectors was roughly 49.8 crore in 2018-19, with around 3.2 crore workers in the organised sector and the remaining in the unorganised sector, there are 46.2 crore workers. [17]. The agriculture sector employs 26.2 crore people, 5.8 crores in construction, and the rest in manufacturing and service among the unorganised sector's 46.2 crore workers [18].

To protect employees in the unorganised sector, such as weavers, handloom workers, fishermen and fisherwomen, toddy tappers, leather workers, plantation workers, and beedi workers, among others. Its volume and effect have changed dramatically in recent years as a result of substantial changes in economic activity. Rural migrants make up 81.26 percent of India's overall migrant population. [19]. This indicates that there is a strong trend for people to move from rural to urban regions.

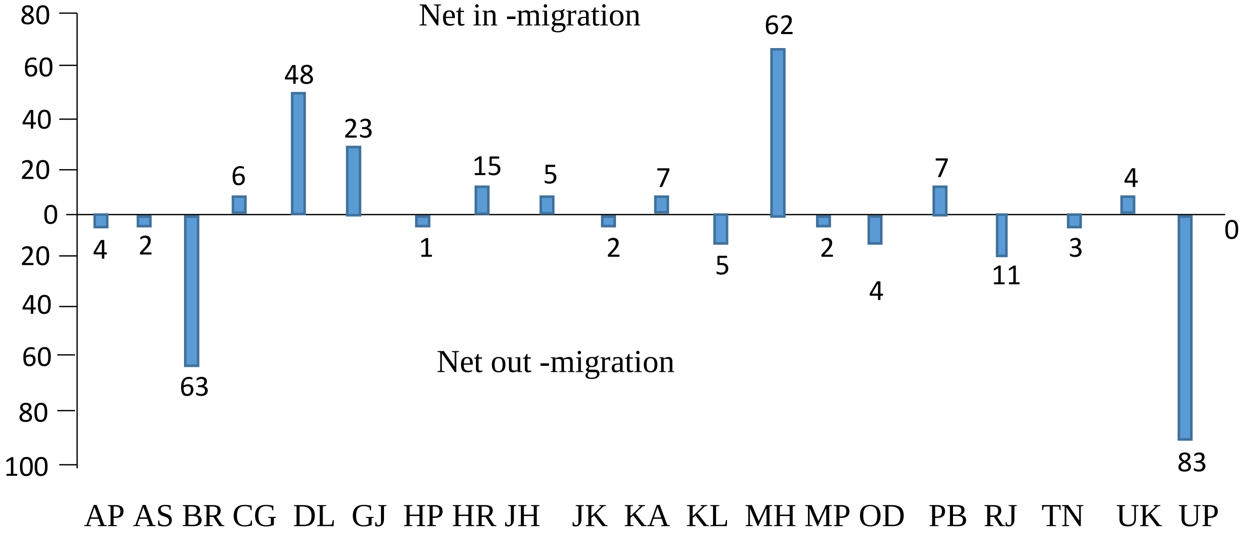


Figure-4: Net in and out-migration in India (refer to abbreviation). Source: 2011 Census, PRS Legislative Research .

In Figure-4, the y-axis represents the number of workers in lakhs and on the x-axis it represents states. The graph is split into two halves:

a) Net in-migration

b) Net out-migration.

A net out-migrant state is one in which more people leave the state than come in. The excess of arriving migrants over outgoing migrants is known as net in-migration. In India, the states of Uttar Pradesh, Bihar, Rajasthan, Madhya Pradesh, and Andhra Pradesh have the highest net out-migration rates, while Delhi, Maharashtra, Gujarat, Haryana, Punjab, and Karnataka have the highest net in-migration rates. [20].

Inter-State migration accounted for a lower proportion of the overall migrant population, falling from 13.1 percent to 11.9 percent between 2001 and 2011. This is because individuals may opt to migrate across shorter distances (within the state) as farm/non-farm labour near to their district in pursuit of better economic possibilities.

Table 1: India's Total Migrants by Last Residence, 2001-2011

Type	**2011 (in Millions)**	**Share of total migrants**	**2001 (in Millions)**	**Share of total migrants**	**Rate of growth between 2001- 2011**	**Per annum rate of growth**
	No	**(%)**	**No**	**(%)**	**(%)**	**(%)**
Intra-State	395.7	86.8	268.2	85.3	47.5	3.96
Inter-State	54.3	11.9	41.2	13.1	31.8	2.80
From other countries	5.9	1.3	5.1	1.6	13.8	1.31
Total Migrants	455.8	100	314.5	100	44.9	3.78
Total Internal Migrants	449.9	98.7	309.4	98.4	45.4	3.82
Total Population	1210.9	-	1028.6	-	17.7	1.64

Source: Census 2011 D2 table based on the last residence. Census 2011 data, available at: https://censusindia.gov.in/Census_And_You/migrations.aspx

Internal migrants grew at a rate of 3.82 percent per year between 2001 and 2011, which was larger than the population growth rate of 1.64 percent per year, suggesting a considerable rise in labour mobility inside the country which is shown in Table-1.

Table-2 lists the four migratory streams that were most common in 2011. Movements between states and within states have varied patterns. Within-State Migration has been dominated by rural-to-rural migration, which accounts for 51% of all such migrations, whereas rural-to-urban migration is very low (14.8 percent).

Table 2: Internal migrants' Stream of Migration in 2011

Steam of migration	**Inter-State**		**Intra-State**	
	Person	**% share**	**Person**	**% share**
Rural-Rural	1,20,19,426	22.2%	20,17,39,806	51.0%
Urban-Rural	28,89,303	5.3%	2,40,85,918	6.1%
Urban-Urban	1,68,07,989	31%	6,12,92,128	15.5%
Rural-Urban	1,96,16,060	36.1%	5,85,85,417	14.8%
Unclassifiable to rural	9,86,521	1.8%	33695236	8.5%
Unclassifiable to urban	19,45,450	3.6%	16254164	4.1%
Total	5,42,64,749	100.0%	39,56,52,669	100.0%

Source: Census 2011, D2 Migration Table. (1) All lengths of stay are included in the data. (2) Those who in their former residence could not be classified as rural or urban are referred to as "unclassifiable."

Table-2 lists the four migratory streams that were most common in 2011. Migrant workers prefer intra-state movement more in comparison to inter-state movement.

2.1.2 Reasons for Migration

As a result of the unequal distribution of opportunities across space, people migrate. People tend to move from places where there are few opportunities and little safety to places where there are more opportunities and better safety. The workers travel from their birthplace to any other state to make their living easily. According to data obtained in Census 2001 for migration by the last residence, there are a variety of causes for relocation. The majority of female migrants identify "marriage" as the reason for their relocation, especially when moving within the state. The main motivations for migration for males are ‘work/employment' and ‘education’[21].

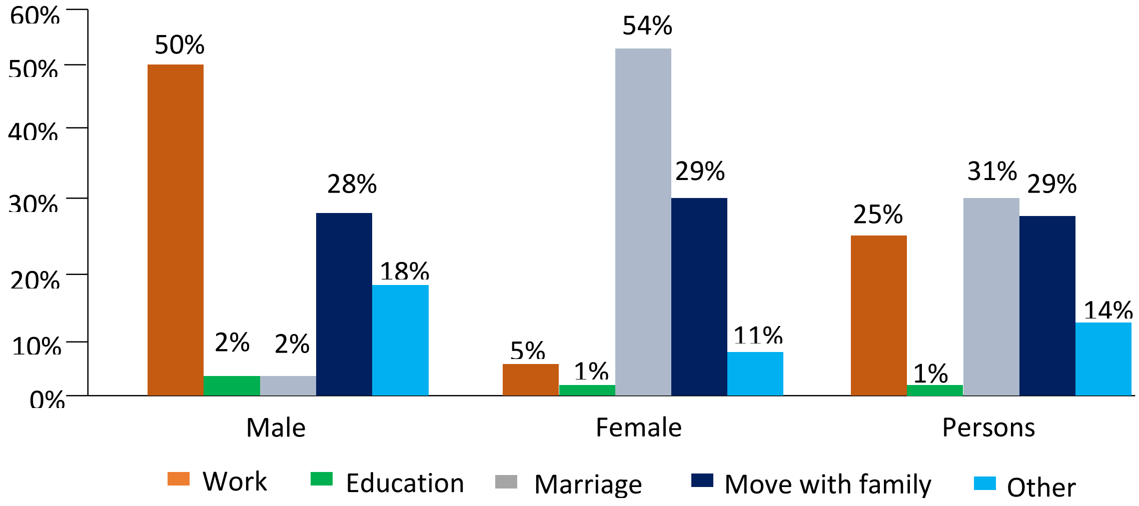
As shown in Table-3, the reason for the migration of males is work or employment. According to census 2011, the number of people seeking to meet their fundamental needs and improve their lives has continued to rise.

Table 3: Reason for migrations

Reason for migration	**Number of Migrants**			**Percentage of migrants**		
	Persons	**Males**	**Females**	**Persons**	**Males**	**females**
Total Migrants	98,301,343	32,896,976	65,404,656	100.0	100.0	100.0
Reason for Migration: Work/Employment	14,446,234	12,373,433	2,072,891	14.6	37.8	3.2
Business	1,136,382	650,335	186,167	1.2	2.9	0.3
Education	2,915,289	2,093,675	876,514	3.1	6.0	1.3
Marriage	43,100,901	679,952	42,421,159	43.7	2.1	64.7
Moved after birth	6,577,310	3,418,673	3,148,707	6.6	10.4	4.8
Moved with households	20,608,115	8,232,143	12,345,862	21.0	25.2	18.8
Other	9,517,171	5,164,045	4,353,086	9.8	15.5	6.6

Source: Table D3, Census of India, 2001

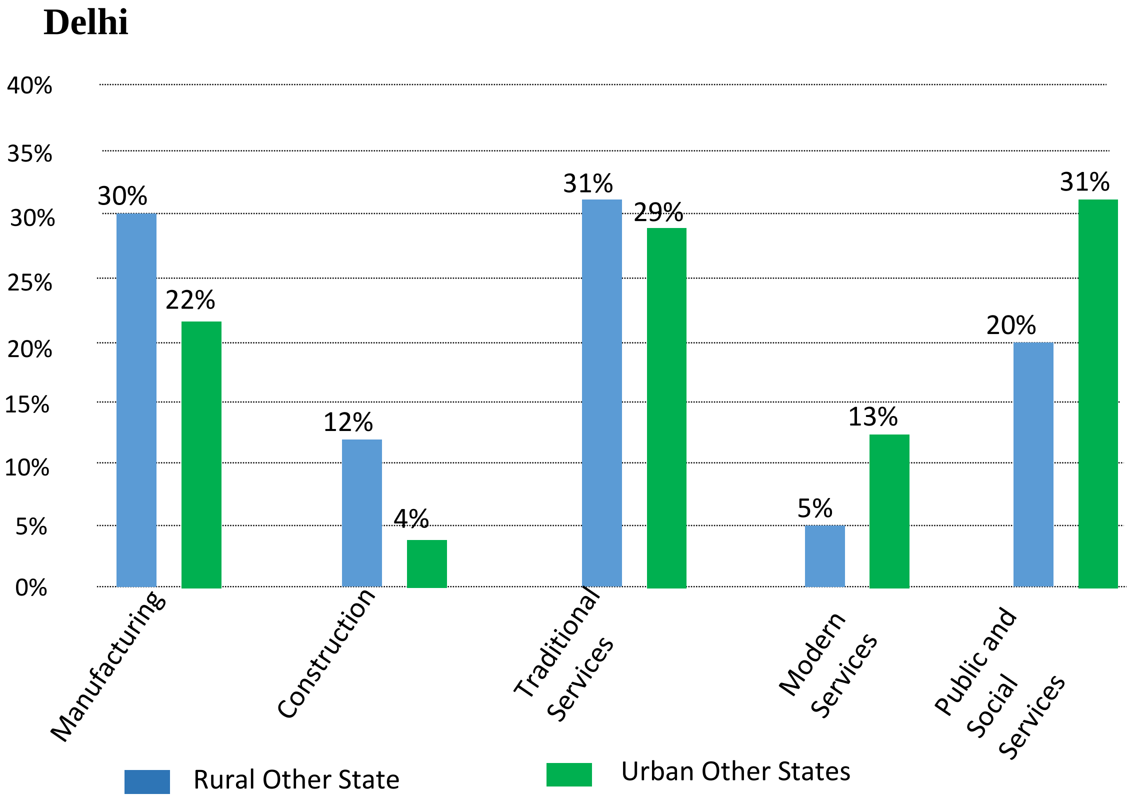
According to Census 2011, there has been a percentage increase, with males' work/employment proportion jumping from 37.5 per cent to over 50 per cent while the reason for the migration of females jumps from 64.9 per cent to 68 per cent in 2011. As a result, the number of migrants is increasing as the year progresses which can be easily interpreted from Figure-5. During the Census 2011 enumeration procedures, almost 45.58 crore Indians were discovered to be "migrants" for various causes. The previous Census (2001) showed a total of 31.45 crore migrants [22]. Rural migrants are more likely to work in construction and traditional services like trade, hotels, and transportation, whereas urban migrants are more likely to work in public services like health and education, as well as modern services like real estate, financial intermediation, and information technology. The psychological element might also play a part in the rise in migration. i.e. if people migrate to a better state, they will have an easier time finding work, earning higher earnings, and working for longer periods.



Source: Census 2011: PRS

Figure-5: Distribution of migrants based on their choices

The majority of workers were unregistered, meaning they were not registered in any of the government portals. Delhi and Maharashtra are the two most popular destinations for migrants.



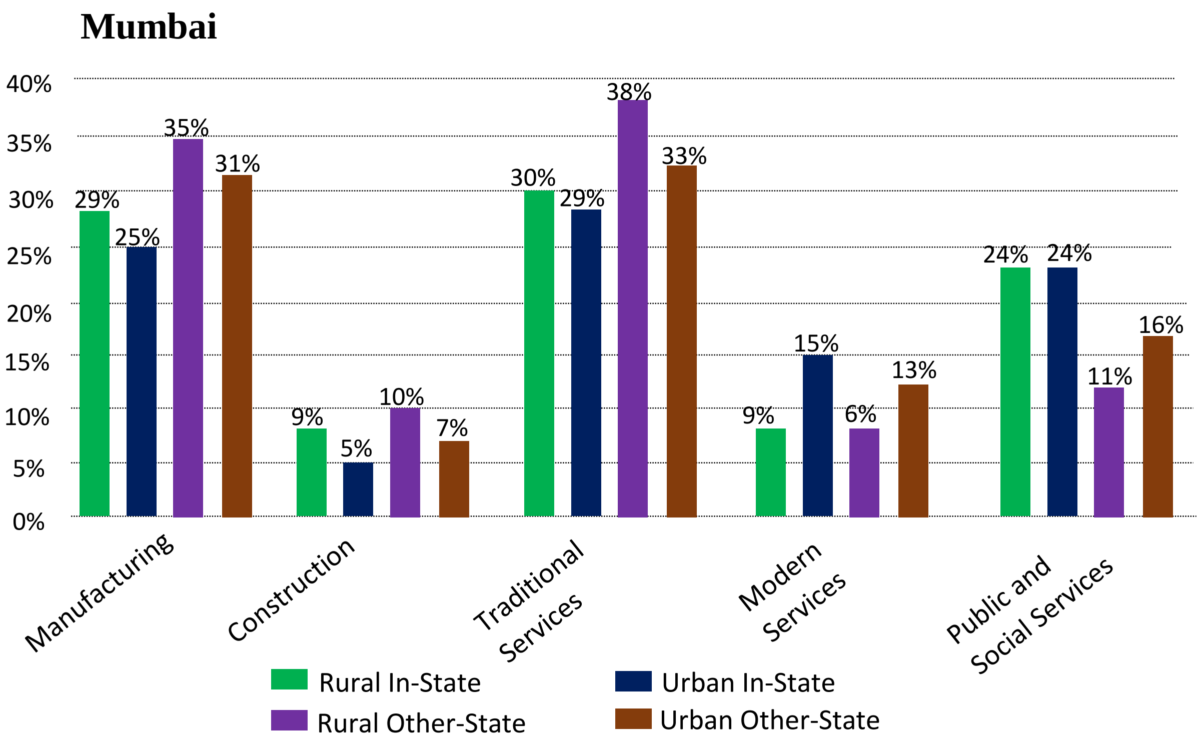


Figure-6: Percentage of migrant workers in different sectors in Delhi and Maharashtra (Source: Census 2011: PRS)

Manufacturing and Construction are preferred by approximately 40-45 per cent of migrants. In the construction sector, 12% of a migrant worker comes from rural while 4% of a migrant worker comes from an urban area, which can be easily interpreted from Figure-6. Kerala attracts migrant labourers from West Bengal, Bihar, Assam, Orissa, and other north-eastern Indian states. The primary causes for migration in the origin state are lower daily income, poverty, indebtedness, and unemployment. Although the major occupation of these migrant workers is agricultural, they work in the construction sector in their destination city, making them unskilled and unprofessional on the job. Workers are paid more than in their home state, but not quite as much as everyday employees in Kerala [23]. This aids Kerala's employers in making a large profit. Migrant workers also transfer money to their families, leaving them with very little money in their own hands, making it difficult for them to subsist.

2.2 CONTRIBUTION OF THE CONSTRUCTION SECTOR IN INDIA'S GDP

The movement of employees is a significant element influencing India's socioeconomic growth. Migrants contribute to the Indian economy by transporting human capital to areas where it is required, allowing for the development of new skills and a higher quality of life. So, migrant workers are directly responsible for approximately 8% of the GDP where the major sector is construction [24].

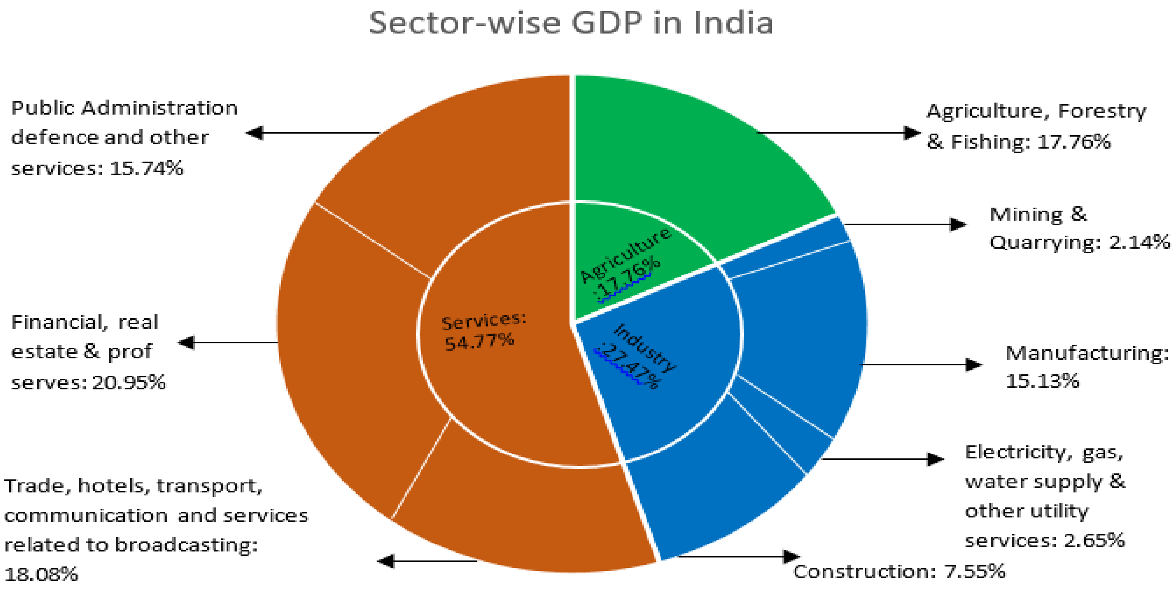


Figure-7: Sector-wise GDP in India (Source: Statistics Times)

The Indian economy may be generally divided into three major sectors, including the primary sector, secondary sector, and tertiary sector, which are also known as the agricultural sector, industrial sector, and service sector, respectively which can be seen in Figure-7. But due to COVID, there is a fall in the construction sector because of lockdown, which also affects India's GDP.

2.2.1 Impact of COVID on the Construction Sector

The severity of the public health issue, as well as the timeliness and strictness of confinement measures, have all played a role in the construction industry's lockdowns. As part of the containment efforts, construction sites in various states were shut down totally or partially. The outbreak of COVID-19 has wrecked devastation on society. It has impacted every country's society and economy. Because the disease is spreading so swiftly, the government has had to proclaim a state of emergency.

As a result of the lockdown, all transportation networks have been shut down, supply lines have been disrupted, and workers are unable to go to the construction site even from their homes. As a result, building activity has come to a standstill, leaving construction employees in a sort of uncertainty. [25]. Due to which, most of the construction companies are experiencing a financial downturn. The Coronavirus is believed to be impacting India's construction sector Rs 30,000 crore a day. This epidemic will cut investment in the building industry by 13 to 30%, affecting both gross value added and jobs. Due to the Coronavirus, India's unemployment rate has climbed to 27.11 percent (COV ID-19) [26].

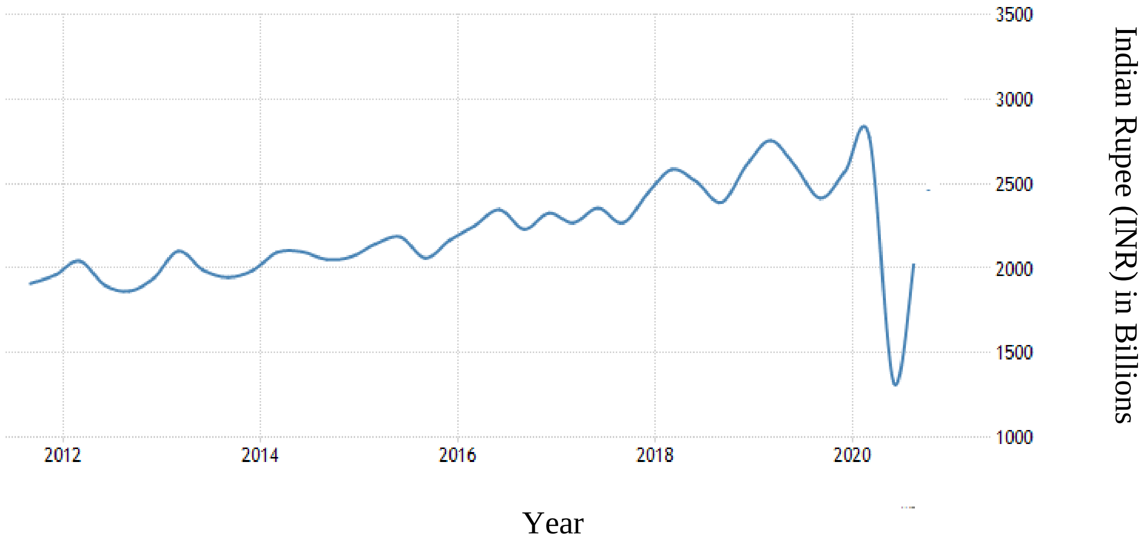


Figure-8: Year-wise India’s GDP from the construction sector is INR billion.

Source: https://tradingeconomics.com/india/gdp-from-construction

In Figure-8, in the x-axis year is shown while in the y-axis INR in billion. It can easily be seen that during the year 2020 and onwards there is a downfall in GDP which is generated from the construction sector, which eventually results in the downfall of India’s GDP.

2.3 CONSTRUCTION FRAMEWORK IN INDIA

The term "construction" refers to the process of constructing anything which includes anything from highways to office buildings to a fresh new cinema, roads, multiplex complexes, and highways [27]. In India, the construction industry is mainly bifurcated into three types of users i.e. labours, contractors and agents. Each one of them is explained below.

2.3.1 Labour

The phrase "labour" refers to productive effort, particularly physical labour performed for a salary. Migrant labour is defined as work performed away from one's regular place of residence, which frequently necessitates living in a location where the worker does not wish to, or is not authorised to, stay permanently [28]. The Ministry of Labour and Employment is one which keeps record of the labours and works on developing new yojanas which can benefit them. This is an Indian federal ministry in charge of defending and protecting employee interests in general, as well as their social security. Developing and organising vocational skill training and employment, as well as well as fostering a healthy work environment in order to increase output and productivity, are among the Ministry's objectives. The labour departments of Uttar Pradesh, Bihar, Delhi, and Maharashtra are the focus of this study. In Table-4, assessment of major labour department is done which is shown below, it helps to identify that the labour department does not use the geospatial analysis to provide more insight to a migrant worker and an agent regarding the workplace in their native place to minimize their movement.

Table 4: Labour Department of major states

SN	Department	Registration Information	Facilities/Gaps	Limitation
1	Uttar Pradesh labour department	• Name, age, qualification • Aadhar card number • Phone number • Nature of work • Permanent and current address	• The training and employment wing provide training to workers to enhance their skills.	• This portal does not provide job opportunities to the worker. • Worker address being noted but not

		• Photo • Self-certified bank passbook • Copy of employment certificate • Copy of self-declaration certificate • Detail of labour family members		used for geospatial analysis.
2	Bihar labour department	• Name, age, qualification • Aadhar card number • Phone number • Nature of work • Permanent and current address • Ward number • Municipal Corporation, area i.e. urban or rural • Contactor license number • Duration of work • Bank information • Nominee details	• Runs various schemes like apprentice working scheme, inter-state migration labour scheme, craftsman training scheme etc.	• Portal gives information regarding jobs but not on regular basis. The last posted job on the portal was in 2018. • Does not use workers addresses in geospatial analysis.
3	Delhi labour department	• Name, age, qualification • Aadhar card number • Phone number • Nature of work • Permanent address and current address	• Provide various schemes like education assistance (500-10,000 per month), health benefits	• Does not provide jobs or any information regarding jobs. • Does not provide any training or skill

		• Address of workplace • Number of working days • Bank information • Detail of labour family members	scheme (2000-10,000), accidental insurance etc.	enhancement opportunity. • Does not use labour address in geospatial analysis
4	Maharashtra labour department	• Name, age, qualification • Aadhar card number • Phone number • Nature of work • Permanent address and current address • Bank information • Detail of labour family members • Marital Status • Category • Wages per day information	• Provide different programmes such as financial assistance of Rs. 10,000/- (ten thousand) for natural birth and Rs. 15,000/- (fifteen thousand) for caesarean birth, educational assistance of Rs. 1200/- (one thousand two hundred) for children in first to seventh grade, and so on.	• Does not provide information regarding jobs to workers. • Took address of labour but did not use it for geospatial analysis.

2.3.2 Contractor

A contractor is a person or corporation that operates on a contract basis, making agreements with several clients to work on specific assignments or projects. [29]. The contractor can be classified by following terms of Technical ability which simply means proper labour, workforce, experience and financial status. The contractor can be in the form of one person or the form of a firm/company. They can be registered or unregistered/agent.

A registered contractor is registered by a local county or municipality and receives a certificate of competency from the Contractors' Licensing Board. Registered contractors can only work in the counties or municipalities where they hold a local license. The contractor can get himself registered both in the government and the public sector. The government contractor departments of Uttar Pradesh, Bihar, Delhi, and Maharashtra are the focus of this study. In the private sector, there are many contractors as sample two are shown here which are Larsen & Toubro Ltd, Shapoorji Pallonji & Co. Ltd.

Table 5: Government contractor department

SN	Department	Registration Information	Limitation/Gaps
1	PWD (UP)	• Name, qualification, age, address • Pan card number • Aadhar card number • Experience (if any) • Application type • Branch of registration • Licence category • Department of registration	• No connection between workers and contractors. • No information regarding the vacancy in ongoing projects. • No geospatial analysis by using the location of the working site.
2	PWD (Bihar)	• Name, qualification, age, address • Aadhar card number • Experience (if any) • Grade/Level • Work type • Branch of registration • Contractor Id • Department	• No connection between workers and contractors. • No information regarding the vacancy in ongoing projects. • No information regarding the location of ongoing projects that can be used for geospatial analysis.
3	CPWD (Delhi)	• Name, qualification, age, address • Pan Card Number	• Location of ongoing projects is not used for geospatial analysis.

		• Grade/Level • Experience certificate • Licence/ Contractor Id • Solvency certificate	• No connection between workers and contractors.
4	PWD (Delhi)	• Name, qualification, age, address • Pan card number • Aadhar card number • Experience (if any) • Application type • Branch of registration • Licence category • Department of registration	• No connection between workers and contractors. • No connection between workers and contractors.
5	PWD (Maharashtra)	• Name, qualification, age, address • Any government id (pan card, aadhar card, DL etc.) • Experience (if any) • Grade/Level • Work type • Licence category • Contractor Id	• No connection between workers and contractors. • No geospatial analysis of ongoing projects location.

To register as a contractor, mostly non-spatial information is being asked which can be seen from Table-5, which provide information about the PWD department of Uttar Pradesh, Bihar, Delhi and Maharashtra. But there is a certain limitation in each department however the most common one is that the department does not use geographical information of contractors and ongoing projects for geospatial analysis. There are many private contractors in India but for sample two are shown below in Table-6.

Table 6: Private contractors in India

SN	**Department**	**Registration Information**	**Limitation/Gaps**
1	Larsen & Toubro Ltd	• Name • Organisation name (if any)	• No connection between workers and contractors.

		• National ID (aadhar/pan card number) • Phone number • E-mail address • Designation • Work type • Proof of designation	• Do not ask for information like the location of ongoing projects which can be used in geospatial analysis.
2	Shapoorji Pallonji & Co. Ltd	• Name • Name of the company (if any) • National ID (aadhar/pan card number) • Phone number • E-mail address • Designation • Solvency certificate	• Do not use location information for geospatial analysis

2.3.3 Agent

An agent's job is to locate labourers for construction industry, and they usually do so from their own town, region, or state. While receiving commissions from both parties, an agent assists his fellow locals in finding job in the city and assists the contractor in finding inexpensive labour. An agent can be counted as unregistered because maximum of them do not register themselves in any kind of portal so there is not much information about them which is a serious issue that needed to be solved in the construction sector.

2.3.4 Other Private Bodies

There are various non-government, not-for-profit organisations like FICCI, Assocham, IBC and many more which provide employment opportunities to various migrant workers. These private bodies also mainly focus on non-spatial information which they took from the migrant workers and the contractor. One of the most common limitations of these private bodies is that they do not use geographical information of contractors, migrant workers and ongoing projects for geospatial analysis and there is no formal framework that connects migrant workers, contractors and agents so that they can communicate directly.

2.4 GAPS AND DEFICIENCIES IN EXISTING SYSTEM

Location plays a major role in extracting additional information. However, the majority of the departments asked for location information from migrant workers and contractors but no specific latitude and longitude is taken which is used in geospatial analysis. Geospatial analysis can help us to extract additional information such as shortest path distance, the time needed to reach that place, how to reach which can be very vital for migrant worker and an agent. There is also a lack of communication between migrant workers and contractors which also needed to be addressed which will help the migrant worker to acquired knowledge regarding the jobs in their native place so that their movement can be decreased. From the contractor point of view, it will help him to get skilled labour without much hassle.

2.5 RECOMMENDATIONS

In view of the foregoing, in order to remove the major deficiency noted i.e. lack of geospatial analysis and lack of a mechanism to connect the migrant workers with the contractor it is proposed to develop an e-portal with the following features:

a) Registration of migrant worker, employer (contractor) and agent.
b) Individual profile of migrant worker, employer (contractor), and an agent to get spatial information (address, city, pin code, state) and non-spatial information like phone number, type of work, qualification etc.
c) Job posting
d) Migrant workers and agents can check the job of their domain by using the search filter and can directly apply.
e) Sent an email to the registered e-mail address of the employer (contractor) containing information of migrant workers and agents who have applied for the position, including their name, phone number, e-mail address, and type of work.
f) Geospatial analysis of geographical information provided by the users and provide information like:
    - The shortest path to reach the workplace
    - Optimum route
    - Information about tolls when travelling through road

- Time required to cover the distance by road
- Information about trains

CHAPTER – 3

RELEVANCE OF GEOSPATIAL ANALYSIS IN CONSTRUCTION SECTOR

3.1 GEOSPATIAL ANALYSIS

Geospatial data refers to information that is based on or related to a geographic place. Geospatial data, often known as geodata, is information about a dataset's location, such as an address, city, or ZIP code [30]. Geospatial technology may be utilised in a STEM application to create intelligent maps and models that can be queried interactively to obtain the desired results, or it can be used to advocate for social investigations and policy-based research. Information about items, events, or other features that have a physical position on or near the earth's surface is known as geospatial data. In geospatial data, position information (normally, earth coordinates) and attribute information (the characteristics of the item, event, or phenomena in issue) are usually coupled with temporal information (the time or life span at which the location and attributes exist) [31].

GPS and satellite pictures are examples of data and imagery from geographic information systems (GIS), are collected, manipulated, and displayed using geospatial analytics. In geospatial data analytics, geographic coordinates are used, as well as unique identifiers like street address and zip code. They are used to create spatial models and data visualisations that help forecast and model trends more accurately. Visualisations that highlight historical and contemporary shifts include maps, graphs, statistics, and cartograms. As a result, making predictions will be easier and more accurate. By adding timing and location to traditional data, geospatial analysis delivers a more complete view of events [32]. Visual patterns and maps that are easy to detect provide insights that would otherwise be lost in a large spreadsheet.

The following are some of the advantages of geospatial analytics:

a) Make analytics more interesting- When events and trends are visualised in the context of identifiable maps, it is easier to comprehend and act on them.

b) Shift your mindset from retrospect to foresight- Organizations can utilise geospatial analytics to predict and prepare for changes brought on by changing spatial conditions or location-based events.

c) Focus solutions- Decision-makers can use location-based data to figure out why solutions that work in one place do not always work in another. It can also help them understand the spatial elements that influence larger trends and may have long-term consequences.

Distance analysis is used in the majority of GIS applications. In its most basic form, distance is a measurement of how far one thing is from another. A straight line can be used to determine the shortest distance between two places. Other aspects, on the other hand, must be considered. If there is a roadblock along the way, for example, we must take a detour around it, increasing the total distance travelled. When the terrain changes from flat to mountainous, another thing to consider is surface roughness, which might increase the actual distance travelled. Several different factors can affect the amount of effort necessary to go from one location to another. Latitude and longitude are two coordinates that can be used to locate a single place on the earth's surface.

3.2 TYPES OF DISTANCE CALCULATING METRICS

Distance is the most crucial attribute to get from a set of geocodes (longitude and latitude). In the development and resolution of real-world situations, distance measures are critical [32]. The following are the most commonly used metrics and algorithms for calculating distance:

a) Euclidean distance
b) The Great Circle distance
c) Manhattan distance
d) Dijkstra’s algorithms

The metrics and algorithms that are used to calculate the distance are explained below:

a) **Euclidean Distance –** The distance between two locations (or the distance along a straight line) is calculated using the Euclidean distance formula. This is also known as the Pythagorean distance because it is a formalisation of the "Pythagorean" theorem [33]. Euclidean distance is best for emergencies where helicopters can fly in a straight line to sites like hospitals since it ignores real-world structures.

$$d(e,f) = \sqrt{(e_1-f_1)^2 + (e_2-f_2)^2 + \cdots + (e_n-f_n)^2}$$

$$= \sqrt{\sum_{i=1}^{n}(e_i-f_i)^2}$$

where e and f are two points in Euclidean space, n = n – space (for geospatial application, n = 2, representing longitude and latitude).

(Source: An article on “CUEMATH”, April 17, 2019)

b) **The Great Circle Distance** - The great circle, which is also called as the Riemannian circle, is the path with the lowest curvature, and hence an arc of a great circle is the shortest path between two points on the surface. The great-circle distance is defined as the distance between any two locations on a sphere. The shortest path between two places on the surface of a sphere is the great-circle route. The great circle distance, unlike the Euclidean distance, takes into account the fact that two points are on the surface of a sphere.

The Haversine formula is used to calculate the Great Circle Distance as follows

$$\alpha = \sin^2\left(\frac{\Delta\text{lat}}{2}\right) + \cos(\text{lat}_1) \text{ x } \cos(\text{lat}_2) \text{ x } \sin^2\left(\frac{\Delta\text{long}}{2}\right)$$

$$c = 2 \text{ x } \arctan\left(\frac{\sqrt{a}}{\sqrt{1-a}}\right)$$

$$d = R \text{ x } c$$

where R is the radius of Earth, d is the distance in km between starting and ending location and latitude and longitude of starting and end location.

(Source: An article ESRI community, “Distance on a sphere” by Simon Kettle, May 10, 2017)

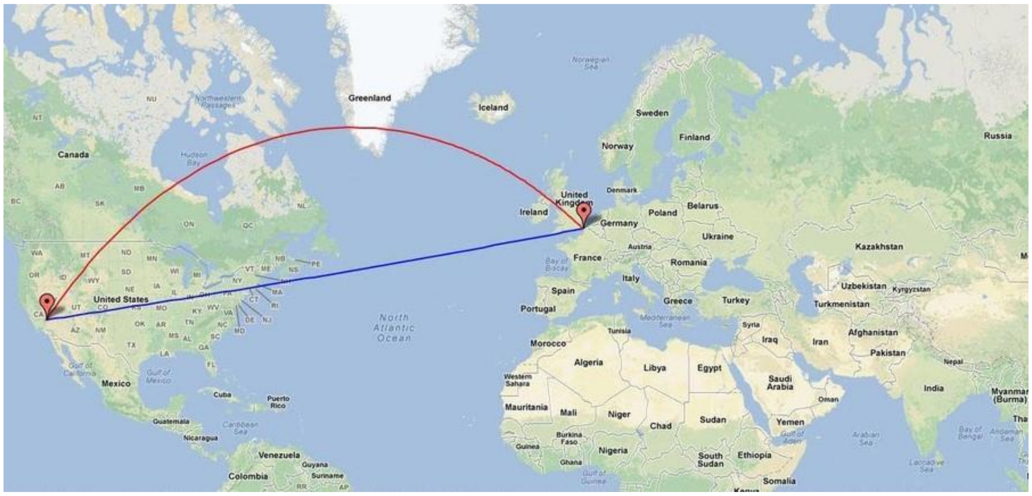


Figure-9: The great circle distance between two points

c) **Manhattan Distance (Taxicab Distance)** - A taxicab geometry is a form of geometry in which the Euclidean distance function or metric is used instead of the conventional distance function or metric. Geometry is replaced with a new metric in which the distance between two points is equal to the total of their absolute differences in Cartesian Coordinates. [33]. Most GPS units utilise this method to determine distances since it takes into consideration the grid pattern of sites.

$$d=|x_1-x_2| + |y_1-y_2|$$

where x and y are the distance between the two points.

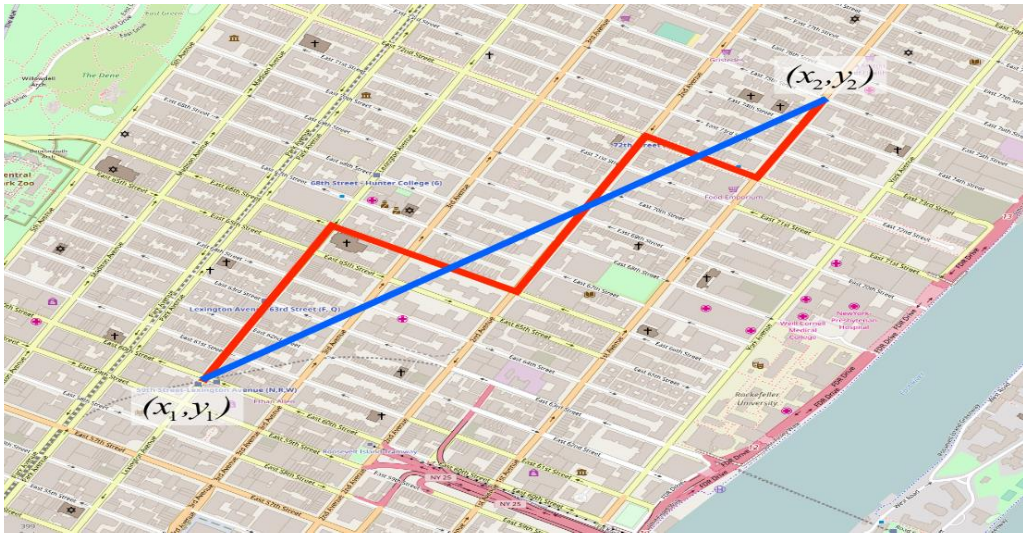

(x_2, y_2)
(x_1, y_1)
Hunter College
Lexington Avenue 63rd Street
Rockefeller University
Weill Cornell Medical College
New York Presbyterian Hospital
Food Emporium

Figure-10: Manhattan and Euclidian distance between two points

d) **Dijkstra's Algorithm** - The shortest path analysis is the essential problem in network analyses, which is an important function of GIS. Dijkstra's algorithm is a method for visualising road networks that calculates the shortest routes between nodes in a graph [34]. The method determines the shortest path between a given source node in the graph as well as all other nodes. It may be used for a variety of other things as well like determine the shortest paths between only one node and a single destination node, with the technique terminates once the shortest path to the target node is found.

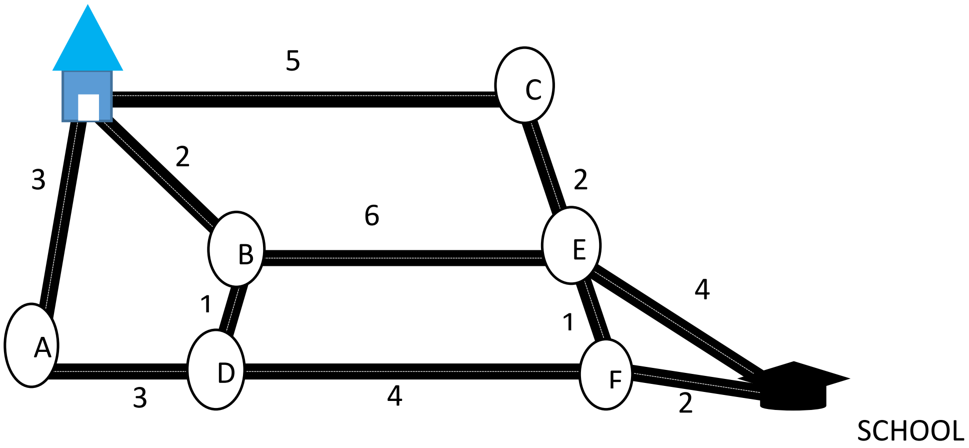


Figure-11: The shortest path found using Dijkstra's algorithm.

The shortest path distance is:- Home→ B→ D→ F→ School

CHAPTER 4

DEVELOPMENT OF MIGRANT E-PORTAL USING GEOSPATIAL ANALYSIS

4.1 INTRODUCTION

The second objective of this research is to create a “Migrant e-portal” that helps the migrant worker, agent and employer (contractor) to communicate directly. The geospatial analysis will also be done by using the geographical information which is provided by the user. This chapter details the steps required in achieving the second goal, including the technology employed in the building of the "Migrant e-portal," a WebGIS application, as well as the approach used. It falls under the following heading:

a) Data Collection
b) Background and Technologies used
c) Methodology

4.2 DATA

The involvement of migrant workers, agents, and contractors in the construction industry is the subject of this research. The information regarding migrant workers was collected from *Delhi Building & Other Construction Workers Welfare Board* which was used in designing the database for this WebGIS application, the sample of which is attached in Annexure-1. The efforts were made to collect the data from the state-wise labour department, District Magistrate of each district, various government and private contractor departments but due to the ongoing pandemic, it was not possible. The District-wise DM list is attached in Annexure-1 from where data regarding migrant workers can be collected. Since, in the case of agents, most of them are unregistered because there is no such kind of portal present from where information regarding them can be extracted.

4.2.1 Synthetic Data

Many efforts are being made to acquire data for this study, however owing to the COVID-19 epidemic, the study was undertaken using synthetic data. A contractor, worker and agent dataset was created for the research which is attached in Annexure-2.

4.3 BACKGROUND AND TECHNOLOGIES USED

The frameworks to be utilised are chosen based on the requirements of the migrant e-portal application. This WebGIS based application composed largely of three key components: an employers (i.e. contractors) to describe employment openings and the information needed to apply for where migrant workers and an agent can view and apply based on their interests. The final component is to use geospatial technology to connect them, allowing migrant workers and agents to see the location of the job they wish to apply for and how to get there (shortest distance path). The Django framework is used to connect to the SQLite database in the Python-based WebGIS application. HTML5, CSS, JavaScript, and Ajax are used to create the front end. The Django server is where web services are deployed.

4.3.1 Python

Python is a programming language that is extensively used. It was produced by Guido van Rossum and published in 1991. It is used to develop WebGIS applications on a server, can connect to database systems, read and alter files, manage massive data, and conduct sophisticated math, and it is used in conjunction with other software to create processes [35].

4.3.2 Web Framework

In the domain of software development, a framework is a collection of prewritten code and libraries that provide core characteristics for building applications. It is a collection of materials and tools for developing and maintaining web apps, web applications, and internet sites. A framework has template capabilities for conveying data in a browser, a programming environment for authoring flow of information, and an application programming interface (API) for retrieving underpinning data resources [36]. A framework is a structured folder containing common resources such as programming model libraries, image files, classes, header files, and reference.

4.3.2.1 Django Framework

Django is a Python web framework for quickly creating safe and maintainable websites. Django takes care of a lot of the tedious parts of web design, so one can concentrate on project instead of recreating the wheel [37]. It offers a slew of unique features that set it apart from competing frameworks. They are as follows:

Versatile- Django is a powerful programming language that has been used to build almost every type of website, from content control systems and blogs to online communities and news sites. It can integrate with any client-side framework and serve material in nearly any format.

Secure- Django avoids common pitfalls like keeping cookies store session data., which is vulnerable and rather of using a password hash, passwords are stored directly.

Scalable- Django employs a component-based coherent architecture (each part of the application does not depend on previous one, and hence might get changed if needed).

Maintainable- Django programming follows design principles and patterns, culminating in code that is simple to maintain and reuse. It employs the do not make the same mistake twice philosophy to eliminate needless repetition and hence reduce the amount of code.

4.3.2.2 Django MVT

The MVT (Model, View, Template) architecture is used by Django. MVT is a WebGIS application development software design pattern [38]. Figure-12 describe the MVT architecture thoroughly.

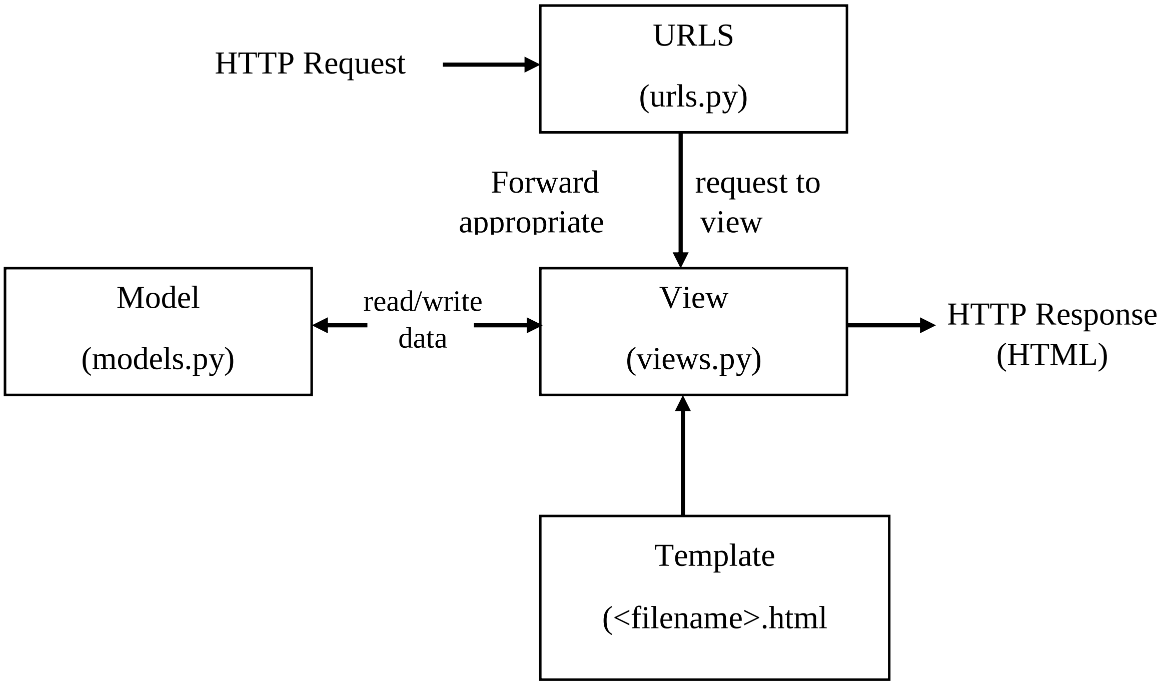


Figure-12: Model, View, Template architecture of Django (Source: GeeksForGreeks, June 30, 2021)

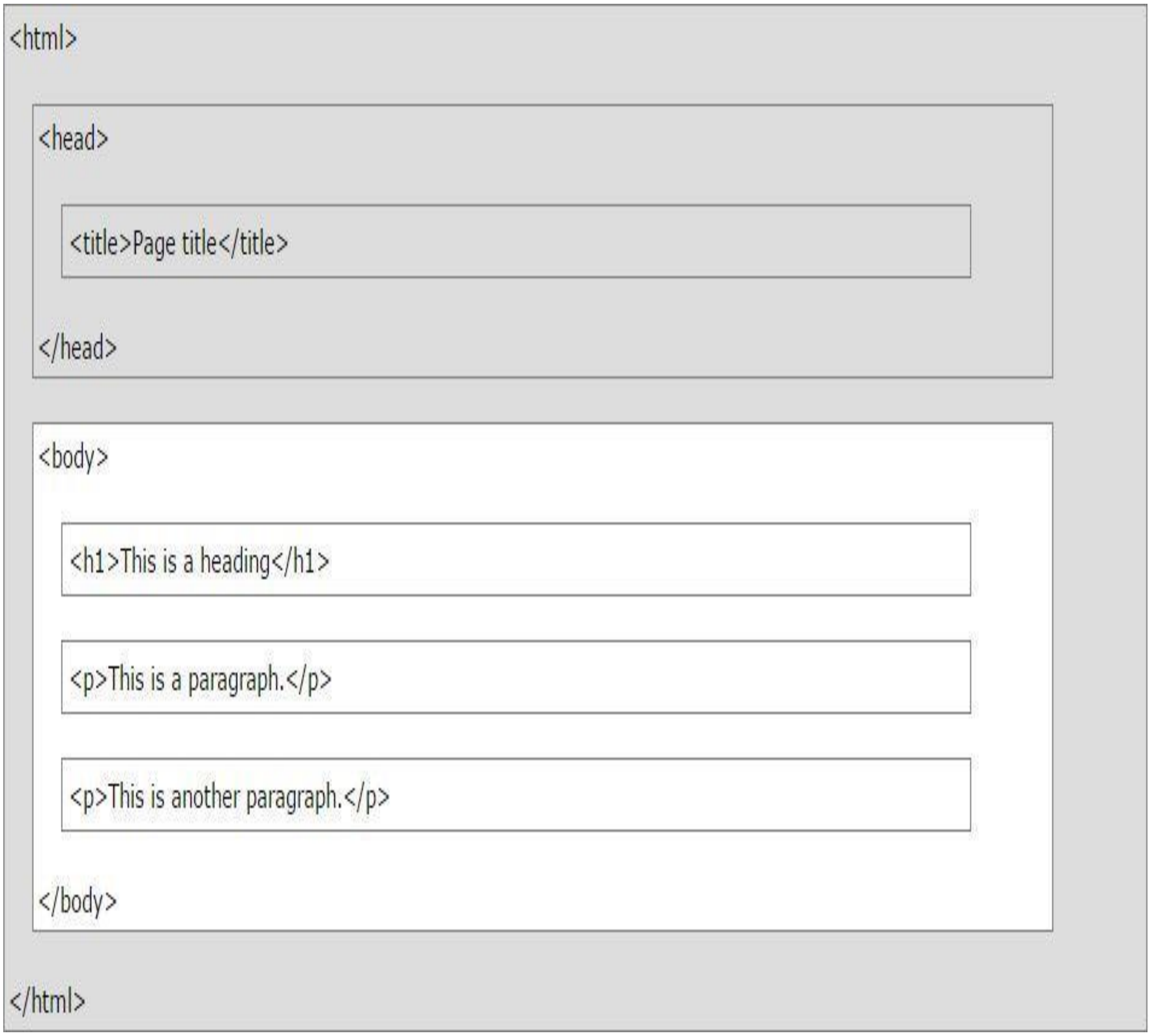
Model- The model will be used to communicate with the data. It is in charge of data storage and administration. The logical data structure that underlies the whole programme is represented by a database (generally relational databases such as MySql, Postgres, SQLite).

View- The user interface, or what we are seeing when we display a webpage in our screen, is called the View. It is represented using HTML or CSS or Javascript files.

Template- A template is made up of both static and particular syntax that defines how dynamic material will be incorporated in the intended HTML result.

4.3.3 HTML

HTML is the most fundamental and widely used web development tool. HTML is a structured language with constraints that control the components These principles are in place to ensure that the document has a logical overall structure [39]. HTML is a markup language that describes the content of a document using a collection of markup elements.

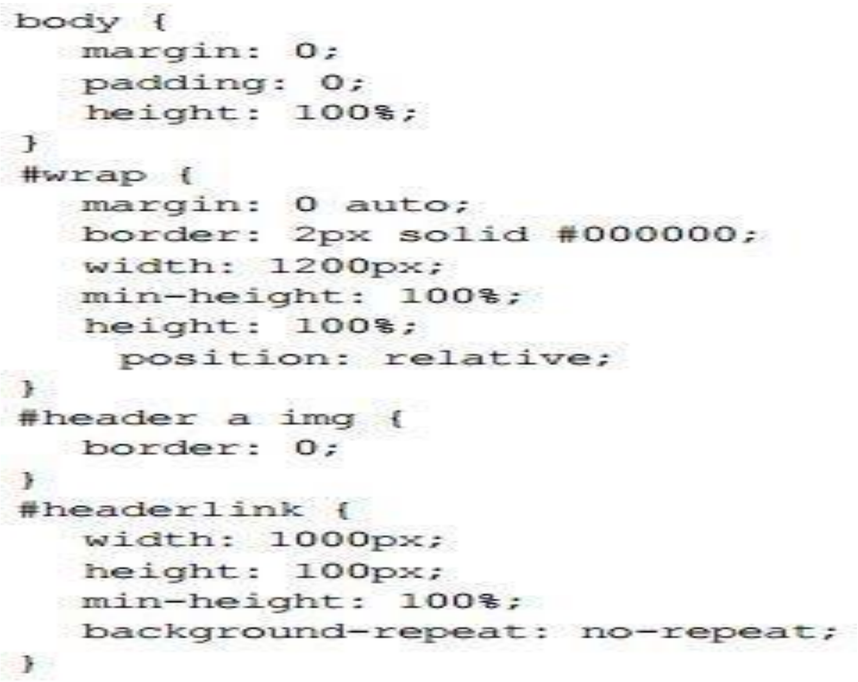


```
<html>
<head>
<title>Page title</title>
</head>
<body>
<h1>This is a heading</h1>
<p>This is a paragraph.</p>
<p>This is another paragraph.</p>
</body>
</html>
```

Figure-13: Basic HTML code example

4.3.4 CSS

The style sheet language CSS is used to style HTML components on a web page. It specifies how HTML components are displayed on a web page, including designs, layouts, fonts, and customising sites for diverse contexts [40].



```
body {
   margin: 0;
   padding: 0;
   height: 100%;
}
#wrap {
   margin: 0 auto;
   border: 2px solid #000000;
   width: 1200px;
   min-height: 100%;
   height: 100%;
    position: relative;
}
#header a img {
   border: 0;
}
#headerlink {
   width: 1000px;
   height: 100px;
   min-height: 100%;
   background-repeat: no-repeat;
}
```

Figure-14: Basic CSS code example

A style sheet can be inserted on a page in a variety of ways:

a) Style sheet for external use

b) Style guide for internal use

c) Inline design

Depending on the need for CSS styles, the developer can select how to incorporate a style sheet into the WebGIS application. It is not dependent on HTML and can be put to use to style a variety of markup languages, including XHTML and XML.

4.3.5 JavaScript

JavaScript is an object-oriented programming language that may be used to improve the user-friendliness of web pages [41]. It is a compact and light language. JavaScript can be coupled to items to allow administrative control of them using a search engine.

Java Script comes equipped library of entities, such as Array, Data, and Math, as well as a fundamental range of language aspects, such as operator, procedural code, and statements. Client-side JavaScript and Server-side JavaScript are two examples of how Core JavaScript can be expanded. It is highly beneficial in terms of security because it prevents users from accessing any WebGIS application without first logging in.

4.3.6 AJAX

Ajax is a technique for creating online applications that are faster, more efficient, and more interactive using XML, HTML, CSS, and JavaScript [42]. Ajax aids in the development of interactive online applications that respond to user requests quickly. When a user acts, Ajax refreshes the contents of the web page quickly. The core component of Ajax is the asynchronous communication JavaScript object XMLHttpRequest.

4.3.7 SQLite Server

SQLite is a free and open-source Relational Database Management System (RDBMS) that stores all of its information in columns.. Each table is linked to another table, i.e. it has a relationship with another table, which is defined by integrity requirements [43]. These tables contain columns that indicate an entity's attributes, as well as rows of data for each column. This is referred to as the database, and it is linked to the frontend or user interface via a controller. We can update, delete, alter, and amend, and so on with the data in the tables. This is a database management system that is both quick and scalable.

4.3.8 LocationIQ

LocationIQ offers enterprise-grade location-based solutions that are flexible. An access token is required for any request for APIs from LocationIQ or Map tiles. To prevent misuse, access tokens can be produced for each application, labelled appropriately (for example, "my website"), and reissued frequently [44]. Access tokens can be used in both public and private environments (websites and apps and server backend).

This server's access token is: **pk.204bff55d6929954ac6b8aec2f64326**

4.3.9 Google Maps

Google Maps is an online mapping platform and consumer application created by Google. Satellite images, aerial photographs, street maps, 360° interactive panoramic street views (Street View), authentic traffic patterns, and navigation for walkers, biking, flying (in beta), and public transit are all accessible [45]. In Google Maps, we can find the following map types:

a) The default road map view is displayed by the roadmap. This is the most common map type.
b) Displays from space Satellite photos from Google Earth.
c) A hybrid view combines normal and satellite perspectives.
d) Terrain shows a physical map that is based on terrain data.

4.4.10 Razorpay

Razorpay is India's first payments service that enables businesses to accept, process, and distribute payments [46]. Credit cards, debit cards, net banking, UPI, and popular wallets like as JioMoney, Mobikwik, Airtel Money, FreeCharge, Ola Money, and PayZapp are all supported.

key_id: **rzp_test_Vg8MRjqU9W7Fsr**
key_secret: **qFLtv4fXu4rm2ck6CAY6m6Om**

4.4 REQUIREMENT SPECIFICATION

The technical prerequisites for developing migrant e-portal applications are listed below:

4.4.1 Software Prerequisites

a) System of Operation: Windows 10
b) IDE: Visual Studio, Pycharm
c) Frameworks and APIs: Django MVT, LoctaionIQ
d) Database: MySQL, SQLite Database
e) Front-end Development: HTML5, CSS3, AJAX,

4.4.2 Hardware Prerequisites

a) Computer's Processor: Intel core i5
b) Speed of the Processor: 3.40 GHz

4.5 METHODOLOGY

The goal of this research is to create an e-portal that will facilitate migrant workers and employers (contractors) using their location for Geospatial Analysis. This will help them to communicate directly as no other platform provides this kind of Geospatial Analysis. The Geospatial Analysis will provide them more helpful information like shortest path distance, the time needed to reach a particular location, which routes to take etc. Because everyone in the construction market is discovering the value of geologically referenced data, this geospatial component, which is not previously included in any portal of similar nature.

The purpose of this e-platform is to help migrant workers to find work easily and within their native place to reduce the movement and to help employers (contractors) to get skilled workers easily using geospatial based analysis.

The methodology used to construct the proposed e-platform employing geospatial analysis is described in this chapter.

a) System Design
b) Use case diagrams
c) Database Design
d) Workflow architecture
e) Geospatial analysis

4.5.1 System Design

Systems design is the process of creating a system's architecture, components, modules, interfaces, and data to satisfy particular requirements. This WebGIS based application composed largely of three key components; an employer’s (contractors) to create a list of job openings with the necessary details that migrant workers and agents can browse and apply based on their preferences. The last component is to connect them using geospatial analysis by using their location information. The following are the modules that have been implemented to develop an e-platform:

a) Admin

b) Employer

c) Migrant Worker

d) Agent

e) Services

f) Financial Module

When all the modules which are mentioned above when linked with each other will help to an efficient and useful Migrant e-portal.

a) Admin- The admin module acts as the super user of the system which holds various functionalities like:

    i) Activating and deactivating Migrant Worker account

    ii) Activating and deactivating Employer account

    iii) Activating and deactivating Agent account

    iv) Admin can view all the list of a migrant worker, employer, and agent, jobs posted by an employer and applied by a migrant worker.

b) Employer- The Employer module will be able to do tasks like:

    i) Edit/View profile

    ii) Post Job vacancy

    iii) View who has applied

    iv) Activate/Deactivate Job post

c) Migrant Worker- The migrant worker module consists of the functionalities that the migrant worker can perform which includes:

    i) Edit/View profile

    ii) Search for jobs

    iii) Apply for jobs

    iv) View location of the workplace and other information like shortest path, the time needed and which route to take etc.

d) Agent- Agent module consists of the functionalities that the agent can perform which includes:

    i) Edit/View profile

ii) Search for jobs

iii) Apply for jobs

iv) View location of the workplace and other information like shortest path, the time needed and which route to take etc.

e) Services- This module consists of the services that are provided to the users:

i) Migrant workers and the agent will be able to see the location of the workplace on the map.

ii) Shortest distance route, the time required to cover that distance, optimal route etc.

iii) Migrant workers and the agent will be able to search the job according to their interests and location.

iv) A mail will be sent to the employer (contractor) whenever a migrant worker or agent will apply for the job containing their details on his/her registered e-mail address, phone number and type of work they have applied for.

f) Financial Model- This module consists of a description of how the revenue will be generated through this migrant e-portal:

i) In the initial stage, the portal will be free for the users, there will be no subscription charges.

ii) When the free trial gets over, after the registration of the users, they have to pay some amount of money for a subscription to use the services of migrant e-portal.

iii) The subscription amount depends upon the user type i.e. worker, contractor and agent.

iv) Workers have to pay for the subscription on the monthly basis i.e. Fixed charges*18% GST and charges must be fixed for everyone in order to avail all the facilities of the portal.

v) The subscription amount of the contractor depends upon the number of workers he needed i.e for 0-20 workers- Fixed charges*18% GST, for 20-40 workers- Fixed charges*18%GST and so on.

vi) Same as the contractor, the subscription amount of the agent also depends upon the number of workers he has having i.e for 0-10 workers- Fixed charges*18% GST, for 10-20 workers- Fixed charges*18%GST and so on.

vii) Payment can be done through cards which can be Visa, MasterCard or Rupay.Net banking or UPI can also be used for payment.

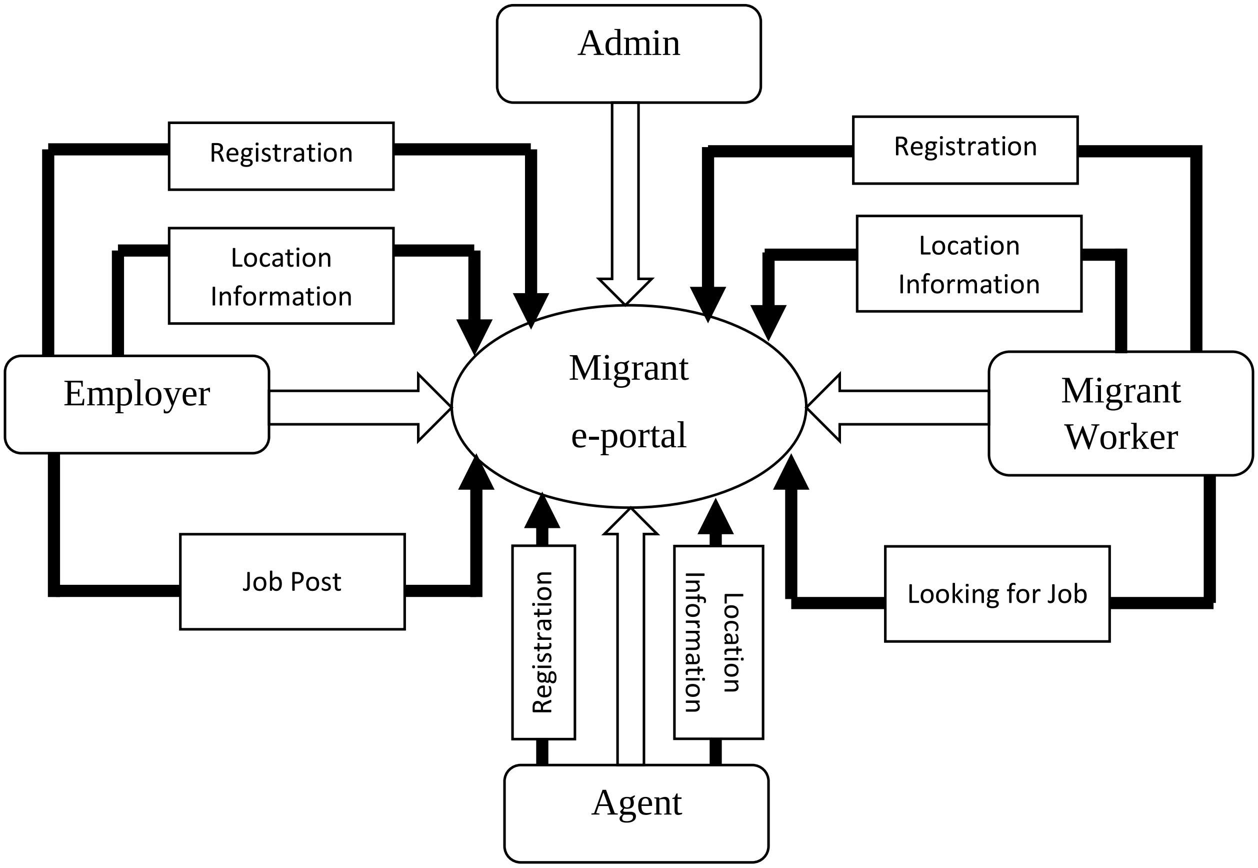


Figure-15: System Design

As indicated in Figure-15, the above-mentioned modules communicate with one another via the Migrant e-portal's core system. These modules, along with their associated actions and Geolocation information, provide the necessary foundation for this Geospatial analysis-based architecture to function.

4.5.2 Case Study Diagram

A diagram is a depiction of a set of things and their relationships in a visual format. A case study is a series of scenarios that define the system's functionality from the point of view of the user. A case study diagram is useful for describing the system's behaviour.

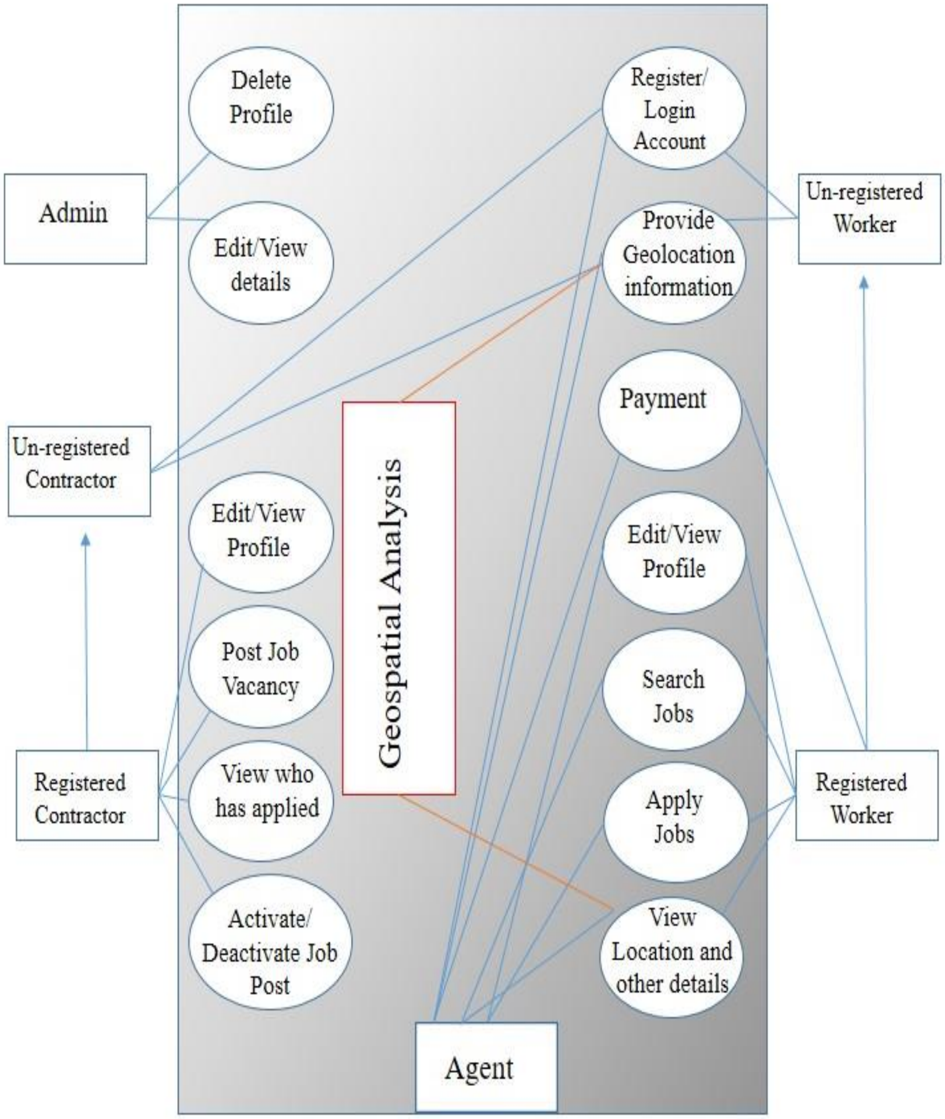


Figure-16: Case study diagram

Characters:

The Characters of the system are Admin, Un-registered Contractor, Registered Contractor, Un-registered Worker, Registered Worker, Un-registered Agent and Registered Agent which can be seen from Figure-16. This set of case study have been identified based on the functions and objectives of the WebGIS based application.

a) **Create a User Account** - This case study denotes a series of procedures that must be followed by the contractor, migrant worker and agent to register with the application.

b) **Payment –** When the user gets registered then he has to make the payment i.e. subscription for a definite period to avail all the facilities of the portal.

c) **Sign-in-** This case study describes the steps that a contractor, migrant worker, or agency must take to sign-in into the application.

d) **Delete Account-** This case study refers to a series of procedures that must be followed by the administrator in order to delete the users account.

e) **Edit/ Observe Details-** This case study describes a set of steps that the administrator must do in order to examine the Employers who have registered with the application.

f) **Post Job Details-** This case study describes the steps that an employer must take in order to advertise a job opening.

g) **Delete/Undelete Job Post-** The activities necessary for the Employer to modify the status of the Job Post are described in this case study.

h) **Examine Job Post Applicants -** This case study refers to a series of steps that an employer must do in order to examine a list of candidates for a certain job opening.

i) **Search Job -** This case study outlines a series of steps that a Migrant Worker must do in order to find accessible and active jobs.

j) **Job Application -** This case study outlines the steps necessary for a Migrant Worker to apply for a job opening.

k) **View Location Details-** This case study depicts a set of steps that a migrant worker must take in order to examine the geographical specifics of their workplace, as well as other useful information such as the shortest distance path, the time necessary, and which route to take.

l) **Geospatial Analysis-** The geographical location given by the users i.e. migrant worker, employer (contractor) and the agent will be used for geospatial analysis and will provide information like shortest distance path, the time required to cover the distance, optimum route etc.

4.5.3 Database Design

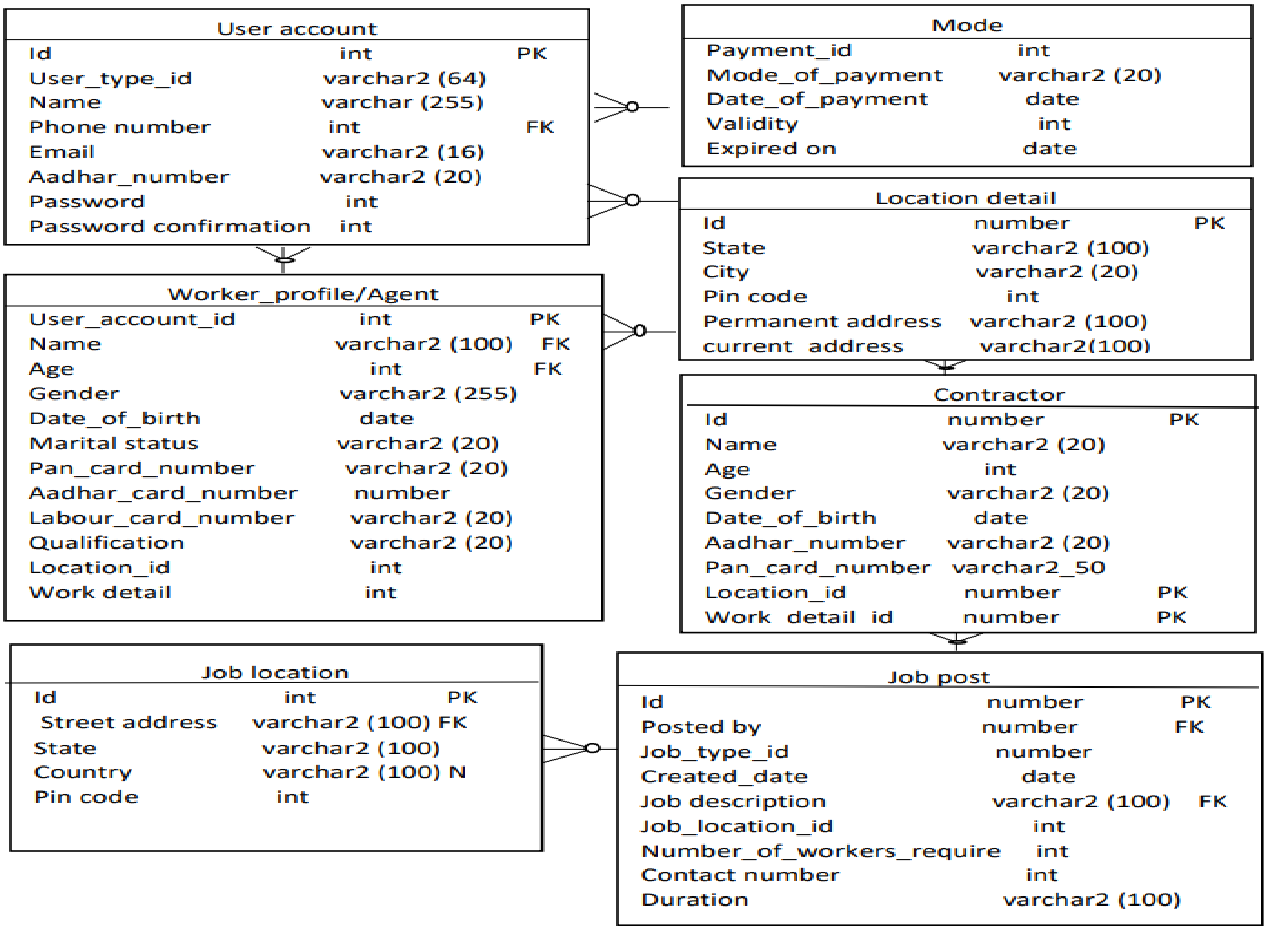
There are three major functional groups:

a) **Organising Users -** This section describes how the application handles end-users such as migrant workers, employers (contractors) and agents.

b) **Payment** – This section describes the payment process i.e. how the users can pay the subscription amount.

c) **Building Profiles -** How the portal facilitates the creation of profiles of a migrant worker, employer and agent.

d) **Job Posting and Job Search** - How the portal makes posting, searching, and applying for jobs easier.



User account		
Id	int	PK
User_type_id	varchar2 (64)	
Name	varchar (255)	
Phone number	int	FK
Email	varchar2 (16)	
Aadhar_number	varchar2 (20)	
Password	int	
Password confirmation	int	

Mode		
Payment_id	int	
Mode_of_payment	varchar2 (20)	
Date_of_payment	date	
Validity	int	
Expired on	date	

Location detail		
Id	number	PK
State	varchar2 (100)	
City	varchar2 (20)	
Pin code	int	
Permanent address	varchar2 (100)	
current address	varchar2(100)	

Worker_profile/Agent		
User_account_id	int	PK
Name	varchar2 (100)	FK
Age	int	FK
Gender	varchar2 (255)	
Date_of_birth	date	
Marital status	varchar2 (20)	
Pan_card_number	varchar2 (20)	
Aadhar_card_number	number	
Labour_card_number	varchar2 (20)	
Qualification	varchar2 (20)	
Location_id	int	
Work detail	int	

Contractor		
Id	number	PK
Name	varchar2 (20)	
Age	int	
Gender	varchar2 (20)	
Date_of_birth	date	
Aadhar_number	varchar2 (20)	
Pan_card_number	varchar2_50	
Location_id	number	PK
Work detail id	number	PK

Job location		
Id	int	PK
Street address	varchar2 (100) FK	
State	varchar2 (100)	
Country	varchar2 (100) N	
Pin code	int	

Job post		
Id	number	PK
Posted by	number	FK
Job_type_id	number	
Created_date	date	
Job description	varchar2 (100)	FK
Job_location_id	int	
Number_of_workers_require	int	
Contact number	int	
Duration	varchar2 (100)	

Figure-17: Database entity relationship diagram

a) **Organising Users-** Individual migrant workers, employers (contractors) and agents are the main sorts of users of the job portal. Before they may utilise the portal, users must first register. Their basic account information is stored in the user_account table. The columns in the user_account table are as follows:

i) **id** – This is the main key for the table as well as a unique identifier for each user. Other tables in the data model will refer to this ID.

ii) **type_of_user** – Indicates if the user is a migrant worker, an employer, or agent.

iii) **aadhar_card_of_user** – This is the number of the Aadhar card that you entered during registration.

iv) **mail_address**– The user's email address is stored in this column. It serves as an additional portal user ID.

v) **contact_details_of_user** – This is the phone number (typically a cell phone) that was entered during the registration process.

vi) **password_of_user**– This stores password created by the user at the time of registration.

vii) **password_confirmation** – To re-check the password created by users during registration.

User type stores a record whether it is a migrant worker or employer (contractor) or an agent at the time of registration. user_sign-in/up stores a record of users' last login date and their last job application date.

b) **Payment** – After getting registered migrant workers, the employer (contractor) and the agent have to pay the subscription fees to avail of the portal facilities. The columns in the payment table are as follows:

i) **user_type** – Indicates if the user is a migrant worker, employer or agent who is doing the payment.

ii) **mode_of_payment –** This will contain the information about the mode which is used for the payment i.e. is made through net banking or debit/credit card or UPI.

iii) **date_of_payment** – This will store the date on which payment is being done.

iv) **experied_on** – This will store the date of expiry of the subscription to remind the user to make the payment again.

c) **Building Profiles** -This section is further divided into three parts i.e. migrant worker, employer (contractor) and agent.

i) **Migrant Worker Profile** - The **worker_details** holds additional information not recorded during the registration procedure is stored in this column. It has the following fields:

- **migrant_worker_account** – This column is derived from the migrant worker account database and serves as the table's primary key. It guarantees that each migrant worker will only have one profile.
- **worker_name** – These columns hold the migrant worker's name.
- **date_of_birth and age** – The date of birth and age of migrant workers are kept in these columns.
- **aadhar_card_details_of_migrant_worker**– This is the number on the aadhar card that was given by migrant worker when get registered.
- **pan_card_number** – This is the pan card number provided by the migrant worker.
- **labor_card_number** – This is the labour card number provided by the migrant worker.
- **qualification-** This will store migrant workers educational history.
- The **detail_of_work** list maintains track of migrant workers past and current experiences. The following fields will be included:
- **type_of_work** – It will store the information about the type of work migrant workers do like mason, painter, workers engaged in concrete mixing, mason helper etc.
- **year_of_experience** – This will store the information about how many years of experience has been gained by a migrant worker.
- The **location_detail** table keeps the record of migrant worker location. It will store information like migrant

workers current and permanent address, its state, city/district, pin-code etc.

ii) **Employer Profile (contractor)**- The **contractor_details** capture additional information not recorded during the time of registration is stored in this table. It has the following fields:

- **employer_name** – These columns hold the employer's name.
- **date_of_birth and age** – These fields contain the date of birth and age of the employer.
- **gender** – These columns hold employers gender.
- **aadhar_card_details_of_employer**– This is the aadhar card details provided during registration.
- **pan_card_number** – This is the pan card number provided by the employers.
- The **work_detail** table keeps a record of employers work detail. It will contain the following fields:
- **ongoing_projects** – It will hold the name of the projects which are currently active.
- **work_type** – This holds the data about the type of work the employer prefers.
- **type** – This holds the information about the type of employers, i.e., government, central, state, private.
- **employer_id** – This holds the legal id of the employer (contractor).
- **class/level** – This will hold the information about the class of the employer (contractor i.e. Class I, Class II, Class III etc).
- The **location_detail** table keeps the record of employers (contractors) location. It will store information like employers current and permanent address, its state, city/district, pin-code etc.

iii) **Agent Profile**– An **agent_details** holds additional information not collected during registration is stored in this section. These fields are included in the document:

- **agent_account_info** – This column is derived from the agent details table and serves as the table's primary key. It guarantees that each agent has a maximum of one profile.
- **agent_name** – These columns hold the agent name.
- **date_of_birth and age** – These fields contain the date of birth and age of the agent.
- **gender** – These columns hold agent gender.
- **aadhar_card_details_of_agent**– This is the aadhar card details provided during registration.
- **pan_card_number** – This is the pan card number provided by an agent.
- **type_of_workers** – This contains information about the nature of the work of migrant workers who are under him.
- **number_of_workers** – This provides information about the number of migrant workers he has.
- The **location_detail** table keeps the record of employers location. It will store information like agent current and permanent address, its state, city/district, pin-code etc.

d) **Job Posting and Job Search** - Here only registered employer (contractor) can post jobs on the platform, and only registered migrant worker and an agent can apply for them.

i) The **job_post** The table contains information regarding job openings. It is directly linked to all other tables that are generated.The **job_post** contains the following fields:

ii) **vacancy_by** – The register user id of the employer who advertised the job is stored in this field.

iii) **nature_of_work** – This field denotes the nature of the job.

iv) **number_of_worker_required** – This stores the data about the number of workers required.

v) **contact_number** – This stores the information about the contact details of an employer (contractor).

vi) **duration** – This stores the information about project duration.

vii) **work_explaination** – This contains a basic job role.

viii) **workplace_location_information**– This is an attribute in the job location table that holds the job's physical location, which includes the street address, city, state and postal code.

4.5.4 Workflow Architecture

The architecture of the workflow can be divided into two parts i.e. system workflow and the users' workflow (migrant worker, employer (contractor) and an agent).

System Workflow

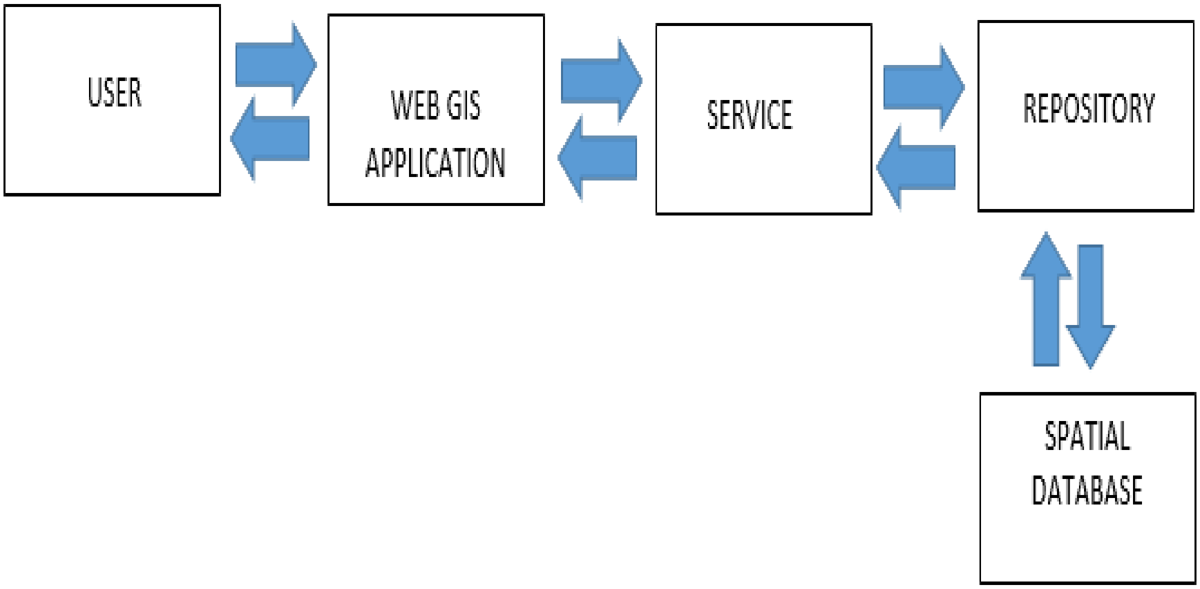


Figure-18: System workflow

Migrant Worker Workflow

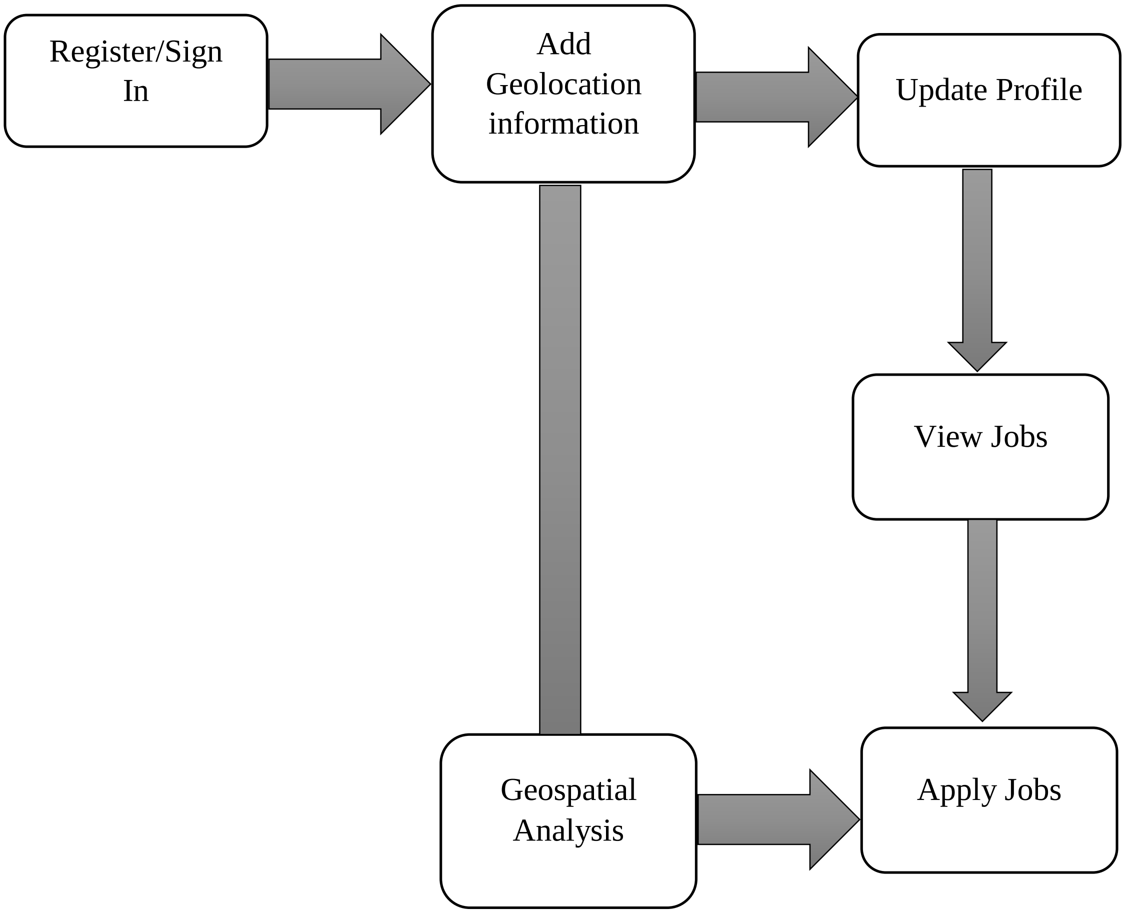
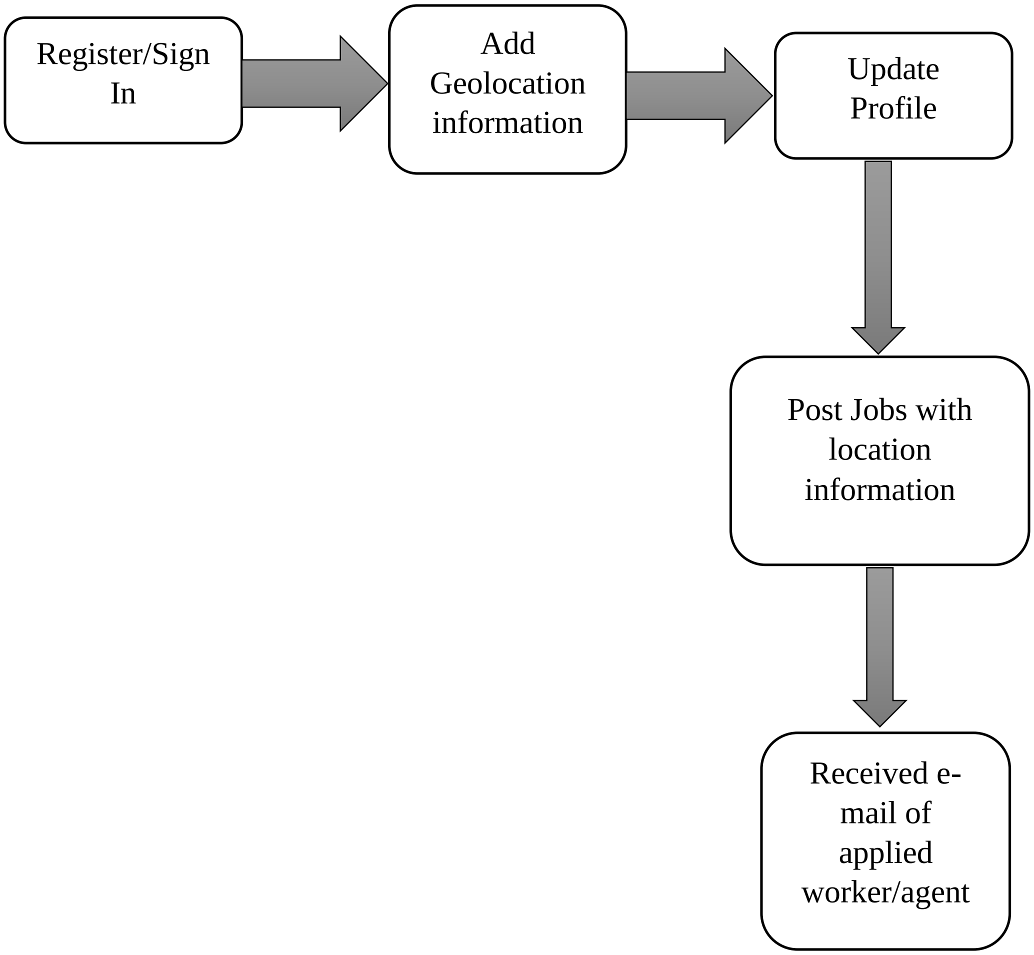


Figure-19: Migrant workflow

Workflow of the migrant worker module:

a) First, register or sign in to the Web GIS application.
b) Then, add Geolocation information to the system.
c) Add, view, edit or update the profile.
d) View jobs posted by the employer.
e) View the location of the workplace on Google Maps with other helpful information like shortest path distance, the time required and which route to choose etc with the help of Geospatial Analysis.
f) Can apply for the job.

Employer Workflow



Register/Sign In
Add Geolocation information
Update Profile
Post Jobs with location information
Received e-mail of applied worker/agent

Figure-20: Employer workflow

Workflow of the employer module:

a) First, register or sign in to the Web GIS application.

b) Then, add Geolocation information to the system.

c) Add, view, edit or update the profile.

d) Post jobs with location information of the workplace.

e) Will receive the information of workers who have applied for the job on the registered e-mail which was provided during registration.

Agent Workflow

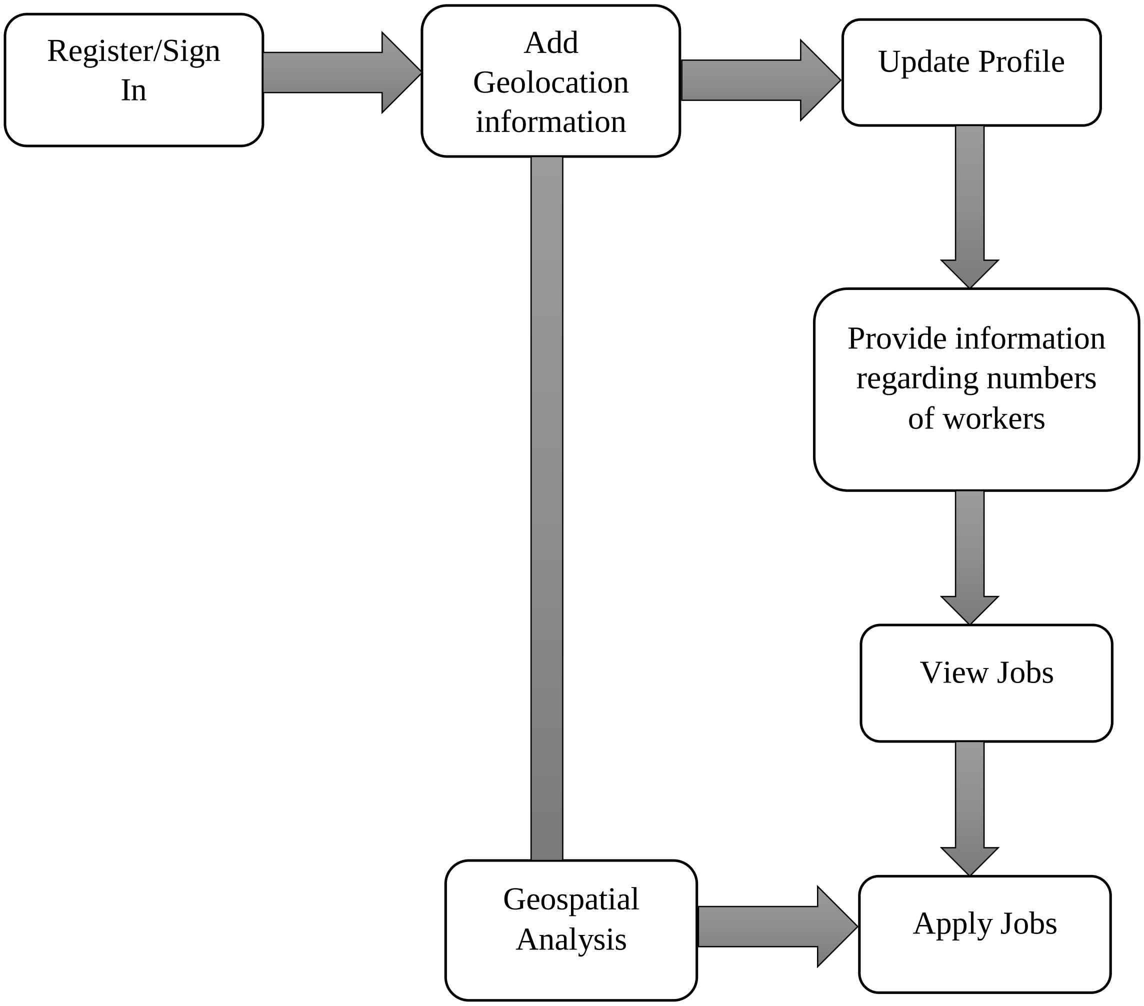

Register/Sign In
Add Geolocation information
Update Profile
Provide information regarding numbers of workers
View Jobs
Geospatial Analysis
Apply Jobs

Figure-21: Agent workflow

Workflow of the agent module:

a) First, register or sign in to the Web GIS application.

b) Then, add Geolocation information to the system.

c) Add, view, edit or update the profile and provide information regarding the number of workers he is having.

d) View jobs posted by the employer.

e) View the location of the workplace on Google Maps with other helpful information like shortest path distance, the time required and which route to choose etc with the help of Geospatial Analysis.

f) Can apply for the job.

4.5.5 Geospatial Analysis

In this study, the address of migrant worker and employer is converted into latitude and longitude and then Dijkstra's Algorithm is used to calculate the shortest path and displayed on the map. Dijkstra's algorithm is utilised for calculating the shortest paths between nodes in the network that may be used to visualise road networks.. The technique calculates the shortest path between a specified source node in the graph and every other node. The geospatial services provided by this GIS-based application are:

Using Geospatial network analysis, the shortest path between the current location of the migrant worker and the location of the workplace is determined using the Dijkstra algorithm. It will also provide additional information like:

a) The time needed to cover the distance by road
b) Which route is optimum
c) Information about tolls when travelling through road
d) Information about trains
e) Time taken by trains to cover the same distance

CHAPTER 5

IMPLEMENTATION AND RESULTS

5.1 INTRODUCTION

The primary goal of this research was to develop an e-portal with geospatial analysis components where migrant workers, contractors and agents register themselves on the same platform and communicate directly with each other. This “Migrant e-portal” uses Model, View, Template design pattern which is a software development programming model. The previous chapter covered this, as well as all other technologies and techniques used to construct this Web GIS Application. This chapter describes the proposed Web GIS Application's implementation and the results obtained. This is covered in the following section:

a) MVT System Design

b) Project Structure

c) User Interface

5.2 MVT DESIGN

The Model-View-Template (MVT) architecture underlies the Django framework. MVT is a WebGIS application development software design pattern. This web framework software serves as a tool for creating and running WebGIS applications. This gives a generic solution to difficulties that arise frequently throughout the creation of a WebGIS application, and it is also reusable based on the needs of the individual application. This design pattern is being used in the building of “Migrant e-portal”. This design pattern divides and organises the program's logic into three distinct components. The three parts are as follows:

a) Model

b) View

c) Template

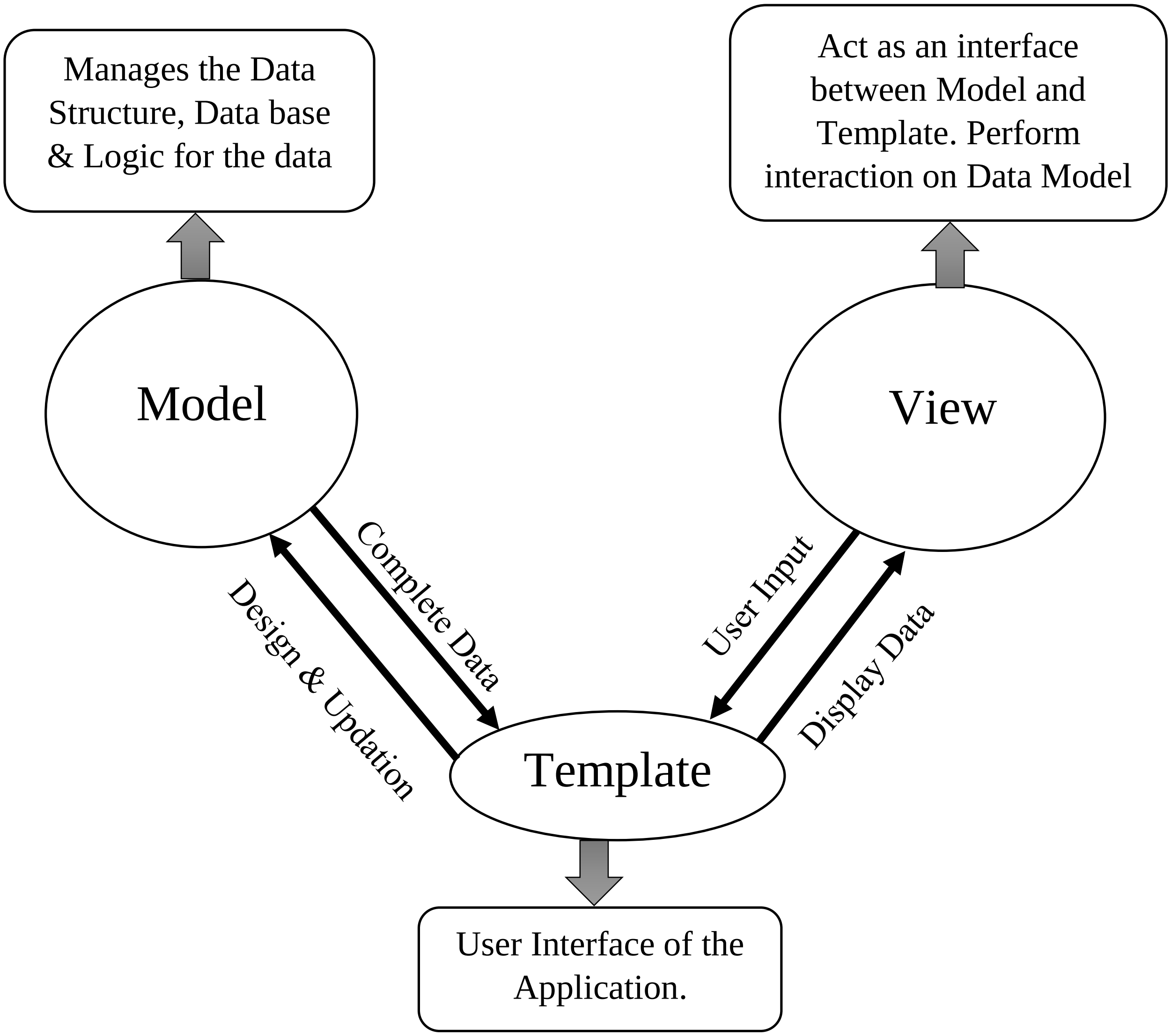

Manages the Data Structure, Data base & Logic for the data
Act as an interface between Model and Template. Perform interaction on Data Model
Model
View
Complete Data
Design & Updation
User Input
Display Data
Template
User Interface of the Application.

Figure-22: MVT Design Pattern

The Model-View-Template design pattern can be seen in Figure-22. How they are interlinked to each other and their working is described below.

5.2.1 MODEL

The model will serve as an interface for data. It acts as a data handler and middleware between the database and the view. It is in charge of keeping the data up to date and managing the WebGIS application's logical data structure. In this WebGIS application i.e. “Migrant e-portal”, the model's job is to develop a dynamic data structure and a relational database for storing all of the spatial and non-spatial data that will enter the system, as well as to connect to the newly generated relational database to retrieve the appropriate spatial and non-spatial data whenever the application requests it. Annexure-2 contains the code related to the implementation of this program logic.

5.2.2 TEMPLATE

The template is in charge of showing Model Data to the user as well as collecting input from them. The template works with layouts. In this WebGIS application, it deals with all the beheld based interaction that was performed by the users which includes all the spatial and non-spatial data that the user enters into the system, all the processed data that the user gets to see, the geospatially based information like shortest path distance between the users' current location and the location of the workplace, optimal route, geographically referenced maps, network analysis based maps and receive data from the user as well as the request method ("POST," GET"), and format the data using the model so that it can be saved in the database. Annexure-2 contains the code related to the implementation of this program logic.

5.2.3 VIEW

This design pattern's View component works as an interface between the Model and Template components, that is, it connects the two by collecting input from the Template component and passing it to the Model component after optional validation. The view work with logic. It handles all the errors which are raised during responding to users input or requests. In this Web GIS application all the User's queries like the type of work, number of vacancy and all the geospatial analysis on the geolocation data of the users is performed by this component. Annexure-2 contains the code related to the implementation of this program logic.

5.3 PROJECT STRUCTURE

Django uses a directory structure to organise the many components of the WebGIS application. When a project is created on Django, it automatically establishes a root directory with the name of the project. It includes all of the files required to provide basic functionality to our WebGIS application. The files created in this project are given below:

a) manage.py
b) url.py

5.3.1 MANAGE.PY

This file contains the project's command-line tool. and in this WebGIS application, it is used for the application's deployment, debugging, and testing. The code for launching the server, relocating, and managing the resources from the command line is contained in this file. Annexure-2 contains the code related to the implementation of this program logic.

5.3.2 URL.PY

This file handles the URL of this WebGIS application. This file contains a list of all terminals which are present in this geospatially based WebGIS application. Annexure-2 contains the code related to the implementation of this program logic.

5.4 USER INTERFACE

The user interface, or web pages, of the built Web GIS Application will be described in this section. The UI act as the contact point between the user and the Web GIS Application. The user enters the required information into the user interface and receives the appropriate response. The MVT design pattern, as mentioned above, handles the processing of the data request and response. The UI of this Web GIS application will be discussed in the headings that follow:

a) Home Page
b) Sign Up Page
c) Payment Page
d) Login Page
e) Individual Profile Page
f) Job Post Page
g) Job Search Page
h) Admin Page

5.4.1 HOME PAGE

The first page of this Web GIS Application is the home page. The screen will have features such as Home, Sign-up, Login, and Search. When clicked on the button by the user, it will open the appropriate page.

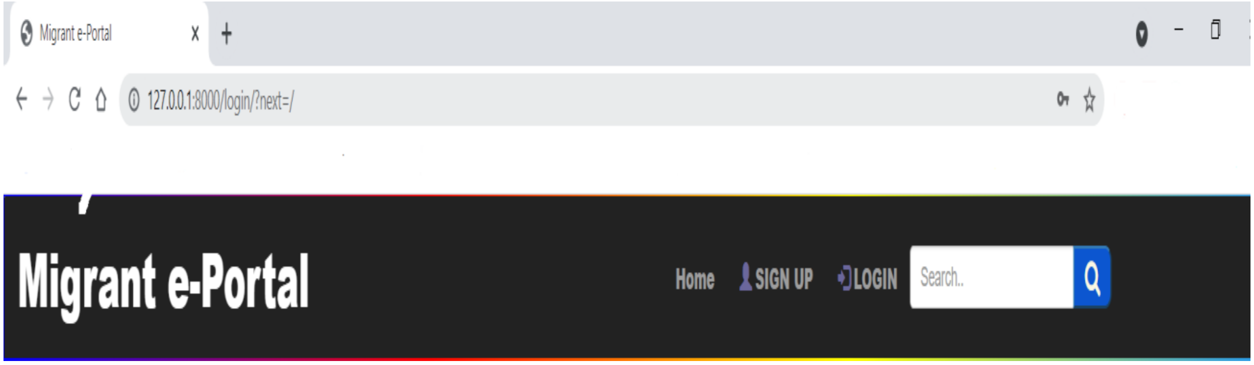


Figure-23: Home page

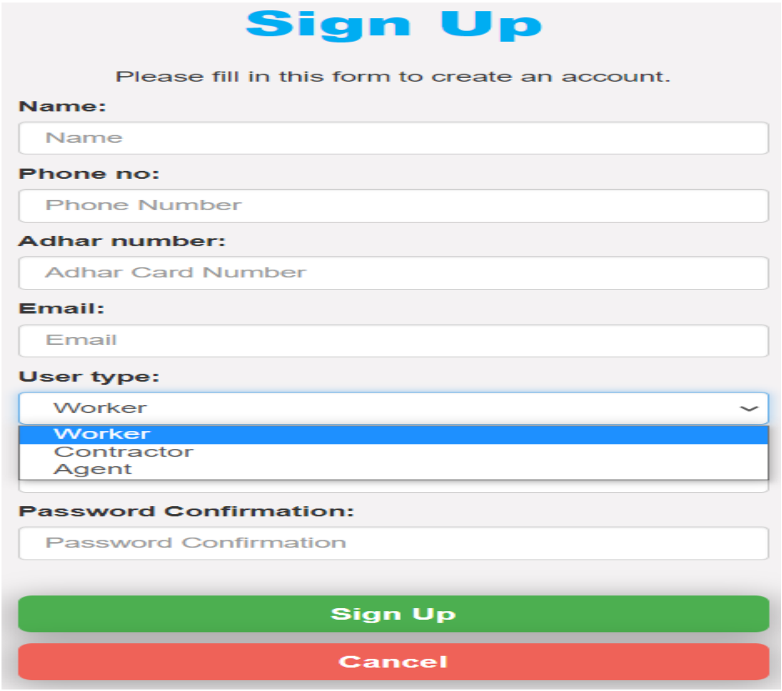
5.4.2 SIGN UP PAGE

The users can register themselves by filling in the details like name, phone number, e-mail address, aadhar card number and password that are asked on the sign-up page of this Web GIS Application. During sign up users can choose their category which is:

a) Worker

b) Contractor

c) Agent



Sign Up

Please fill in this form to create an account.

Name:

Name

Phone no:

Phone Number

Adhar number:

Adhar Card Number

Email:

Email

User type:

Worker

Worker
Contractor
Agent

Password Confirmation:

Password Confirmation

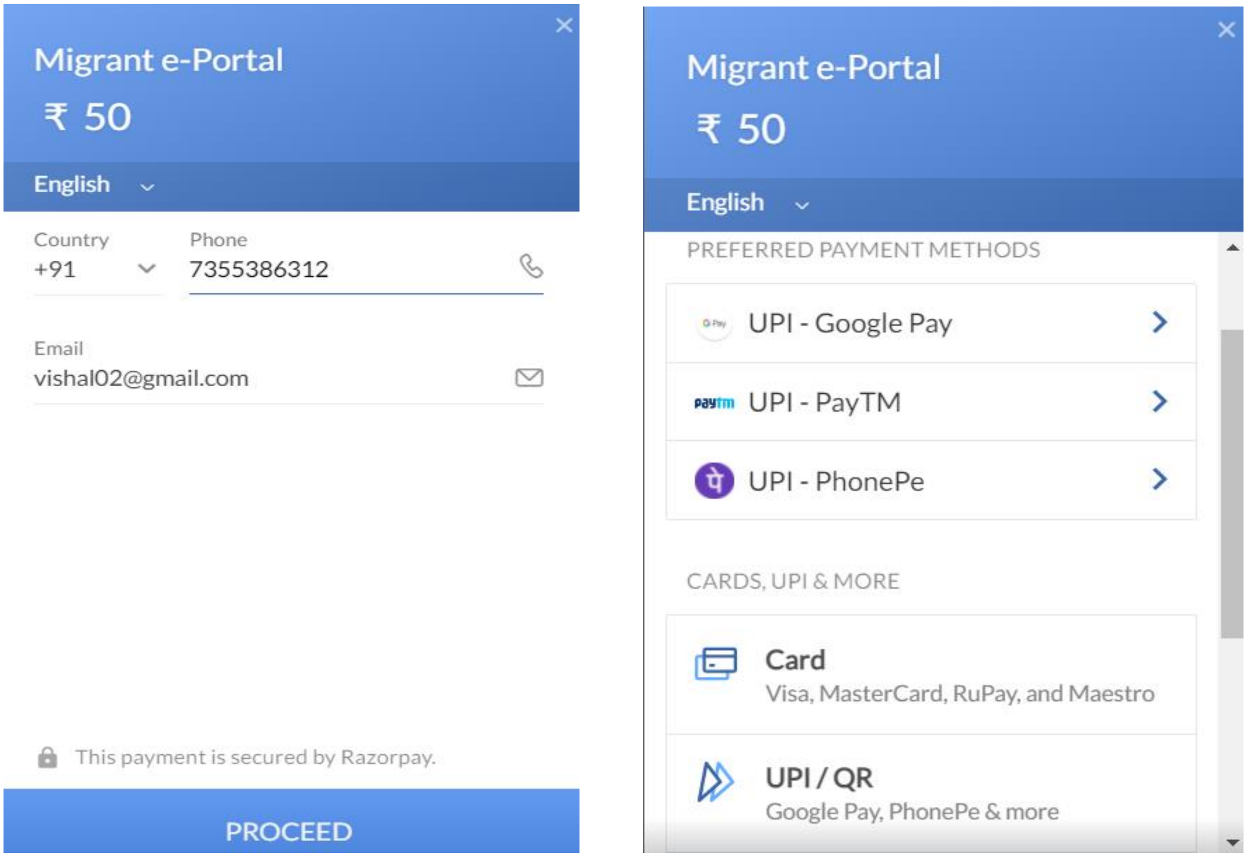
Sign Up

Cancel

Figure-24: Sign Up page

5.4.3 PAYMENT PAGE

After successfully get registered the user has to choose the subscription which depends upon the user type. Payment can be done through cards which can be Visa, MasterCard or Rupay.Net banking or UPI can also be used for payment.



Migrant e-Portal

₹ 50

Country +91

Phone 7355386312

Email vishal02@gmail.com

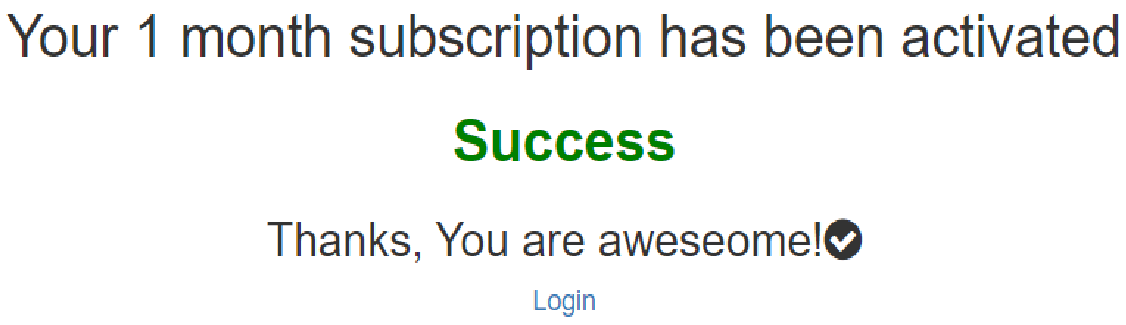
This payment is secured by Razorpay.

Migrant e-Portal

₹ 50

Figure-25: Payment page

After successfully making the payment, the user will be notified through the message that will appear on the screen. After that, they can log in.



Your 1 month subscription has been activated

Success

Thanks, You are aweseome!✓

Figure-26: Subscription activation message after doing payment

5.4.4 LOGIN PAGE

After choosing the appropriate subscription and doing the payment the user can now login into this Web GIS Application by typing their phone number and the password which were provided earlier during sign up.

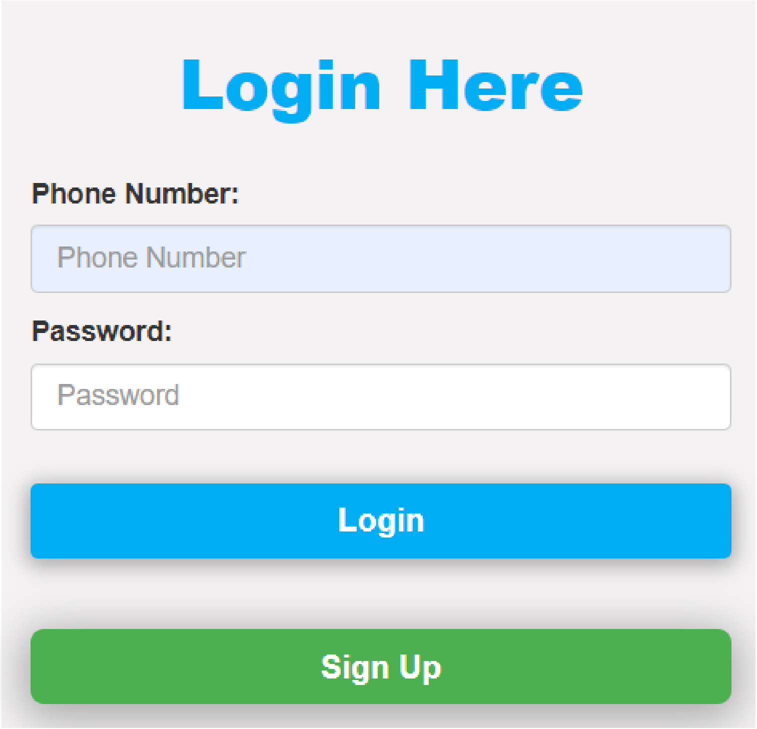
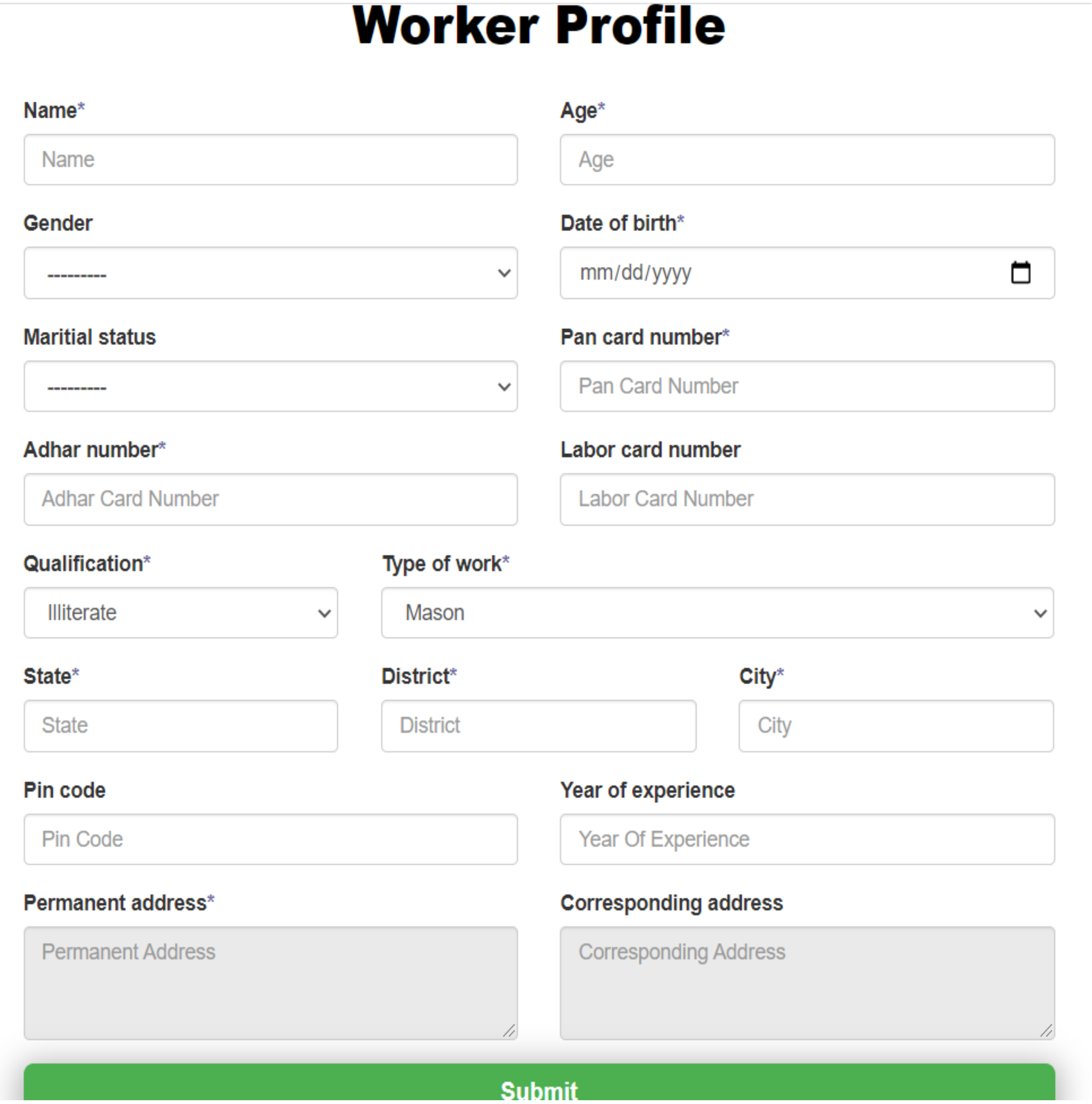


Figure-27: Login page

5.4.5 INDIVIDUAL PROFILE PAGE

In this Web GIS Application, there are three kinds of users i.e. worker, contractor and agent. Each user has a separate profile page where they can fill in their details and can update it at any time after login in.

a) Worker profile page



Worker Profile

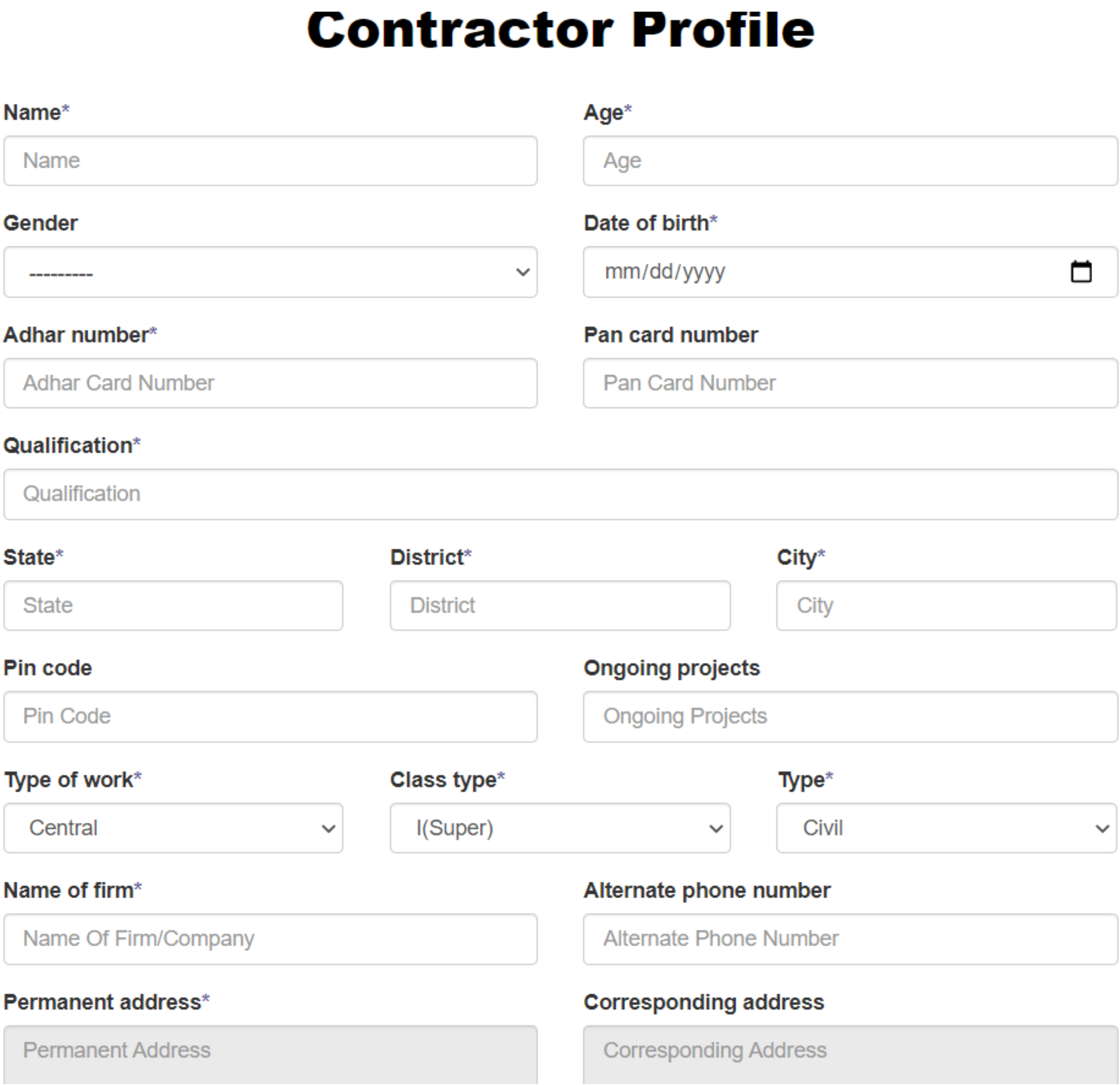
Name*	Age*
Name	Age
Gender	Date of birth*
---------	mm/dd/yyyy
Maritial status	Pan card number*
---------	Pan Card Number
Adhar number*	Labor card number
Adhar Card Number	Labor Card Number

Qualification*	Type of work*	
Illiterate	Mason	
State*	District*	City*
State	District	City

Pin code	Year of experience
Pin Code	Year Of Experience
Permanent address*	Corresponding address
Permanent Address	Corresponding Address

Figure-28: Worker profile page

b) Contractor profile page



Contractor Profile

Name*

Name

Age*

Age

Gender

Date of birth*

mm/dd/yyyy

Adhar number*

Adhar Card Number

Pan card number

Pan Card Number

Qualification*

Qualification

State*

State

District*

District

City*

City

Pin code

Pin Code

Ongoing projects

Ongoing Projects

Type of work*

Central

Class type*

I(Super)

Type*

Civil

Name of firm*

Name Of Firm/Company

Alternate phone number

Alternate Phone Number

Permanent address*

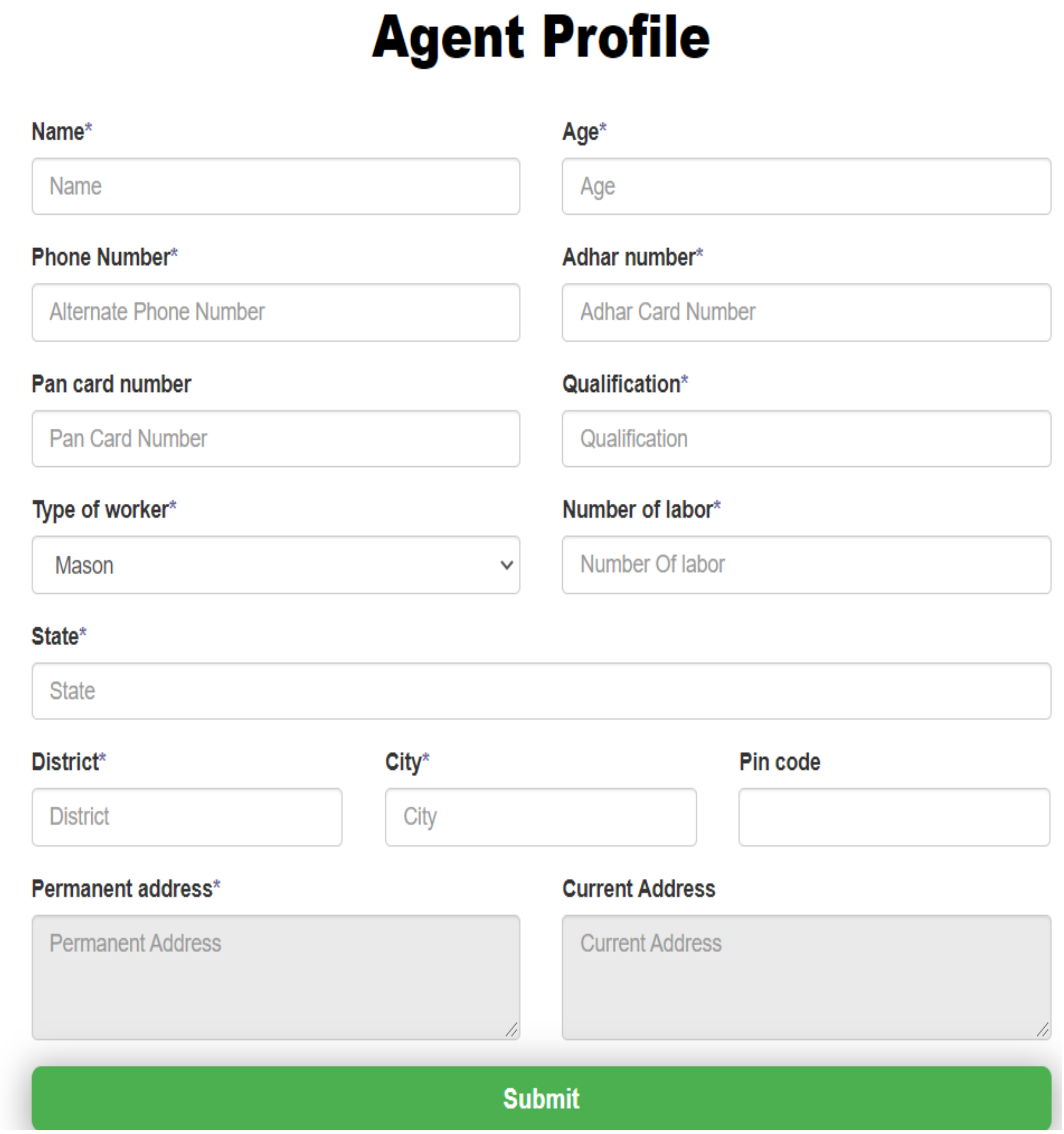
Permanent Address

Corresponding address

Corresponding Address

Figure-29: Contractor profile page

c) Agent profile page



Agent Profile

Name*

Name

Age*

Age

Phone Number*

Alternate Phone Number

Adhar number*

Adhar Card Number

Pan card number

Pan Card Number

Qualification*

Qualification

Type of worker*

Mason

Number of labor*

Number Of labor

State*

State

District*

District

City*

City

Pin code

Permanent address*

Permanent Address

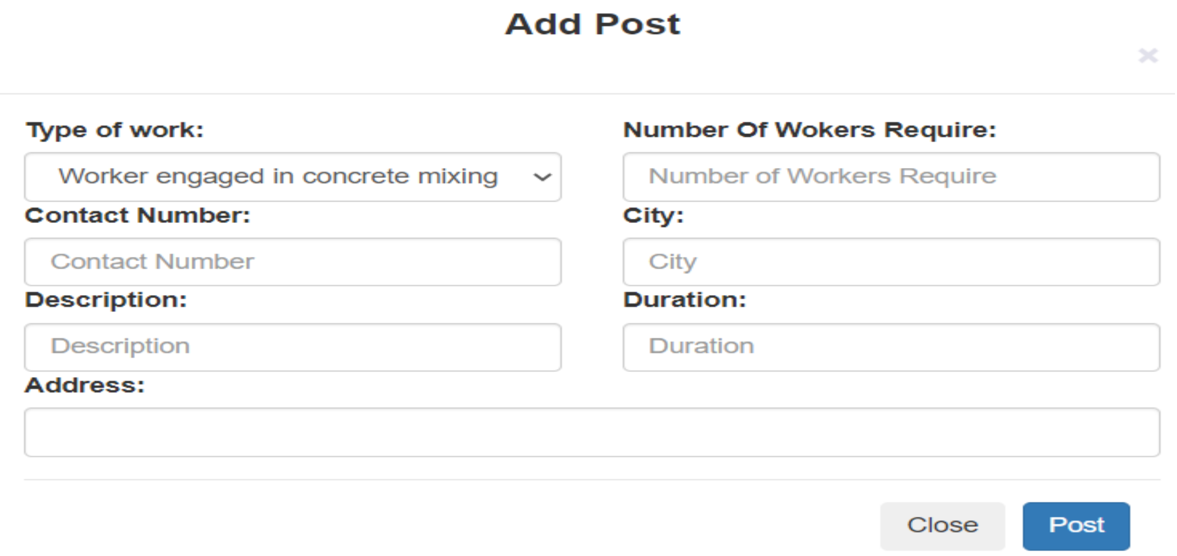
Current Address

Current Address

Figure-30: Agent profile page

5.4.6 JOB POST PAGE

After the contractor has completed the complete registration process then he will be redirected to the job post page where he can add the jobs by filling in the mandatory details like type of work, the number of labours requires, contact number, duration, description and address for geospatial analysis.



Add Post

×

Type of work:

Worker engaged in concrete mixing

Number Of Wokers Require:

Number of Workers Require

Contact Number:

Contact Number

City:

City

Description:

Description

Duration:

Duration

Address:

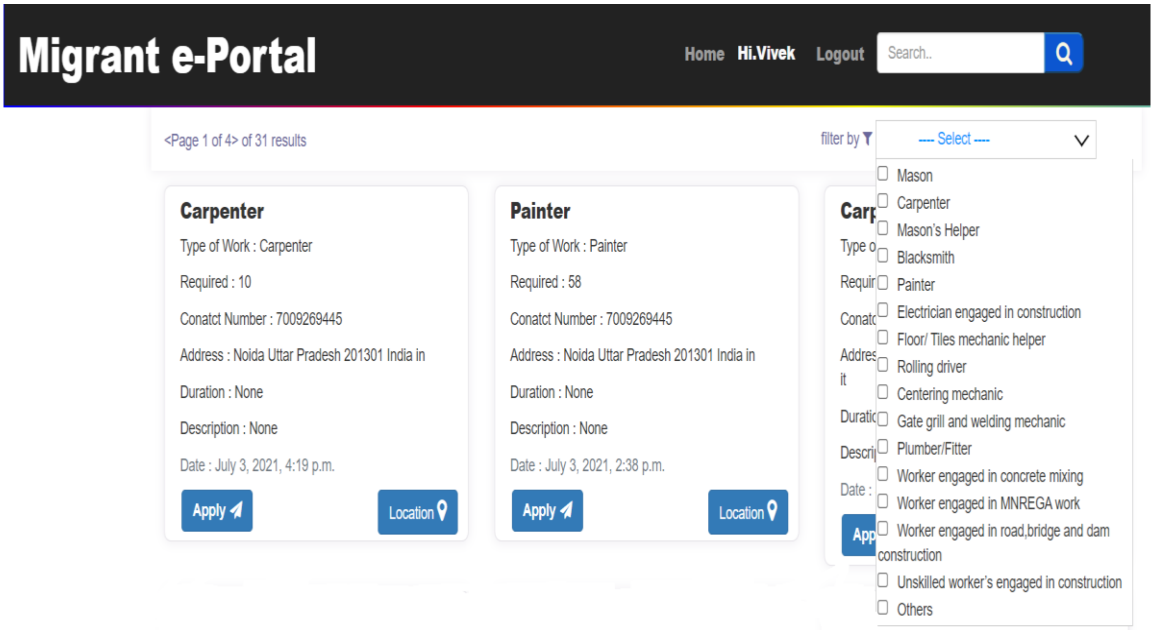
Close

Post

Figure-31: Job post page

5.4.7 JOB SEARCH PAGE

After the worker and agent have completed the complete registration process then they can search the jobs according to their interest by using the search filter where they can filter the jobs based on the nature of work.



Migrant e-Portal

Home Hi.Vivek Logout Search..

<Page 1 of 4> of 31 results

filter by

---- Select ----

□ Mason
□ Carpenter
□ Mason's Helper
□ Blacksmith
□ Painter
□ Electrician engaged in construction
□ Floor/ Tiles mechanic helper
□ Rolling driver
□ Centering mechanic
□ Gate grill and welding mechanic
□ Plumber/Fitter
□ Worker engaged in concrete mixing
□ Worker engaged in MNREGA work
□ Worker engaged in road,bridge and dam construction
□ Unskilled worker's engaged in construction
□ Others

Carpenter

Type of Work : Carpenter

Required : 10

Conatct Number : 7009269445

Address : Noida Uttar Pradesh 201301 India in

Duration : None

Description : None

Date : July 3, 2021, 4:19 p.m.

Apply

Location

Painter

Type of Work : Painter

Required : 58

Conatct Number : 7009269445

Address : Noida Uttar Pradesh 201301 India in

Duration : None

Description : None

Date : July 3, 2021, 2:38 p.m.

Apply

Location

Carp

Type o

Requir

Conatc

Addres it

Duratio

Descri

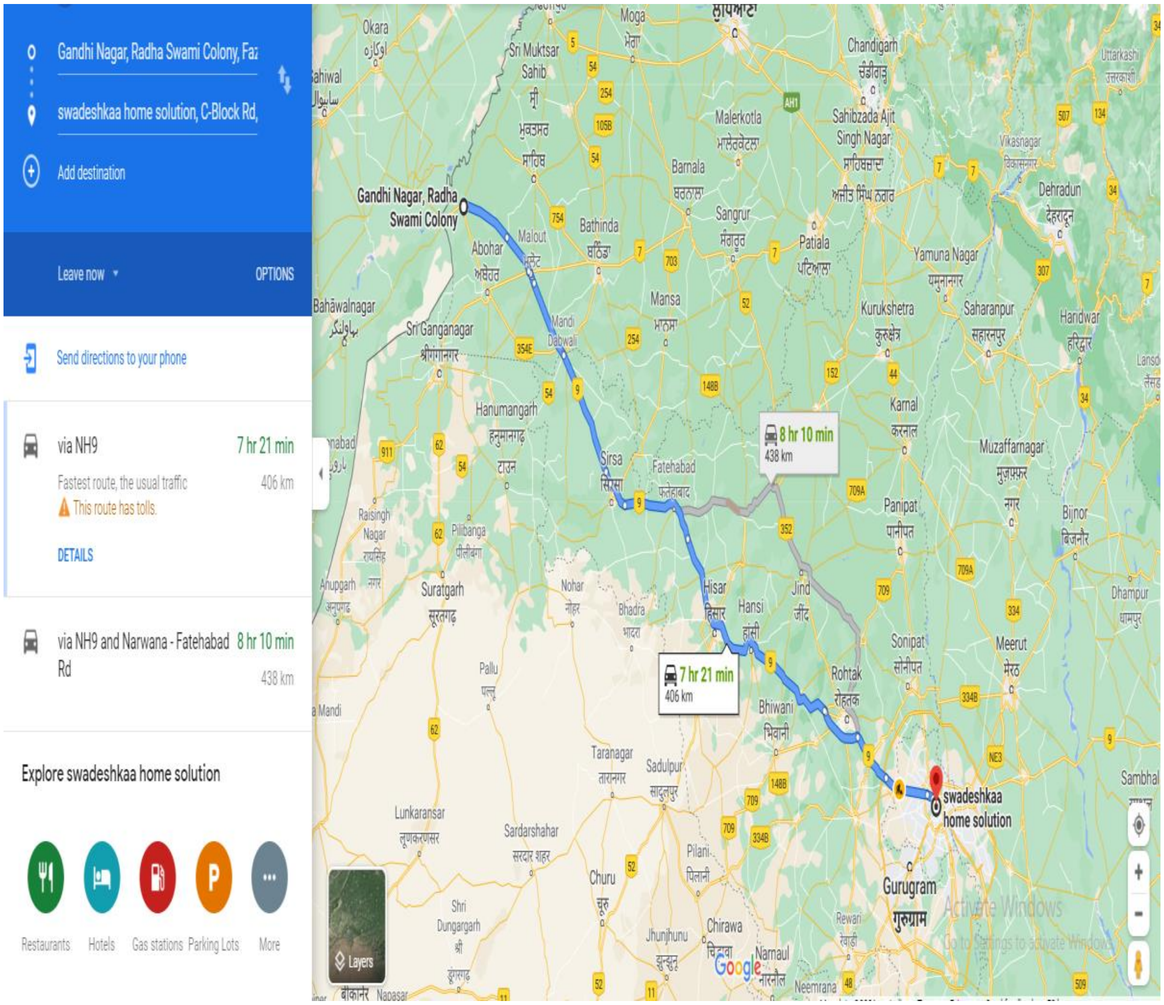
Date :

App

Figure-32: Job search using a type of work filter

The worker and the agent can apply for the job and they will also be able to see the location of the workplace from their current location. The highlights of this page are:

a) The page will have a geographically referenced map that will pinpoint the location of the workplace from workers or agents current location.



Gandhi Nagar, Radha Swami Colony, Faz

swadeshkaa home solution, C-Block Rd,

Add destination

Leave now ▾ OPTIONS

Send directions to your phone

via NH9 7 hr 21 min

Fastest route, the usual traffic 406 km

This route has tolls.

DETAILS

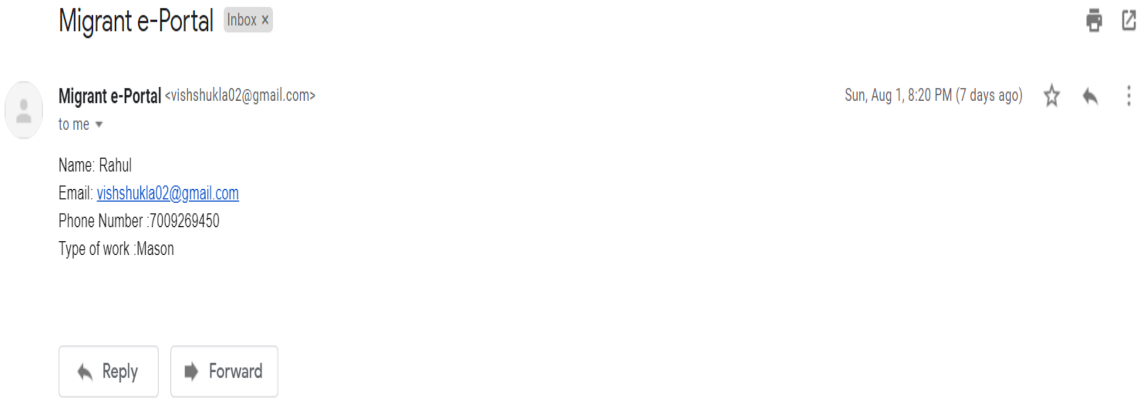
via NH9 and Narwana - Fatehabad Rd 8 hr 10 min

438 km

Explore swadeshkaa home solution

Figure-33: Shortest path distance map

b) When they apply for the job, then an e-mail will be sent to the contractor who has posted for the job. The e-mail will include information such as the worker's or agent's name, contact number, e-mail address, and the sort of job for which they have applied.



Migrant e-Portal

Migrant e-Portal <vishshukla02@gmail.com>

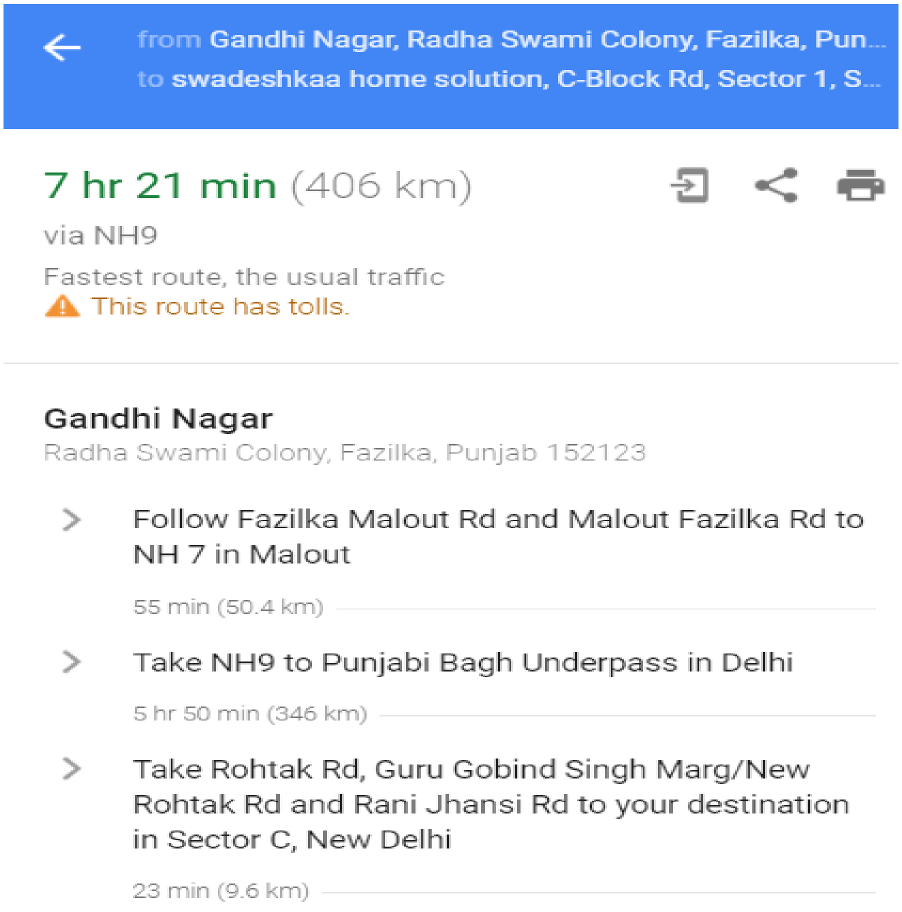
to me

Sun, Aug 1, 8:20 PM (7 days ago)

Name: Rahul
Email: vishshukla02@gmail.com
Phone Number :7009269450
Type of work :Mason

Figure-34: Mail send to the contractor containing details of the worker

c) They will get information like shortest path distance, optimal route, how to reach to that place, how much time it will take, and information of the tolls if they travel through road.



from Gandhi Nagar, Radha Swami Colony, Fazilka, Pun...
to swadeshkaa home solution, C-Block Rd, Sector 1, S...

7 hr 21 min (406 km)

via NH9

Fastest route, the usual traffic

⚠ This route has tolls.

Gandhi Nagar

Radha Swami Colony, Fazilka, Punjab 152123

> Follow Fazilka Malout Rd and Malout Fazilka Rd to NH 7 in Malout

55 min (50.4 km)

> Take NH9 to Punjabi Bagh Underpass in Delhi

5 hr 50 min (346 km)

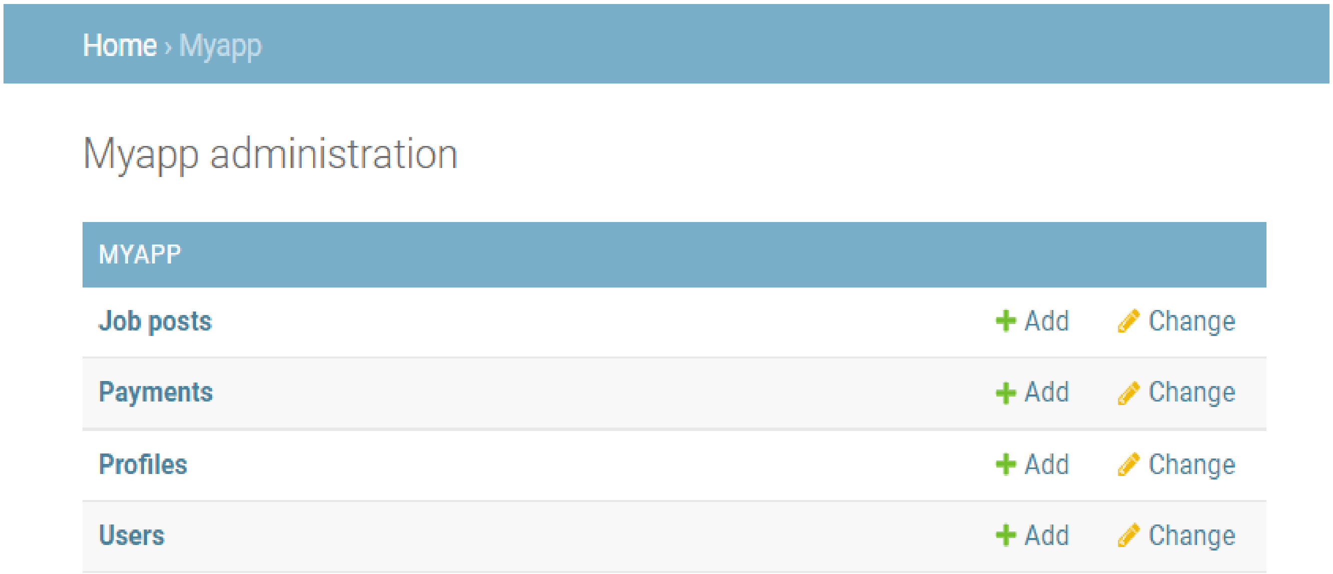
> Take Rohtak Rd, Guru Gobind Singh Marg/New Rohtak Rd and Rani Jhansi Rd to your destination in Sector C, New Delhi

23 min (9.6 km)

Figure-35: Information regarding optimal route

5.4.8 ADMIN HOME PAGE

The admin can navigate the dashboard page, where the admin can examine the data like the information regarding job posts, payment made at the time of registration, profiles of users who have registered with the application.



Home › Myapp

Myapp administration

MYAPP		
Job posts	✚ Add	✎ Change
Payments	✚ Add	✎ Change
Profiles	✚ Add	✎ Change
Users	✚ Add	✎ Change

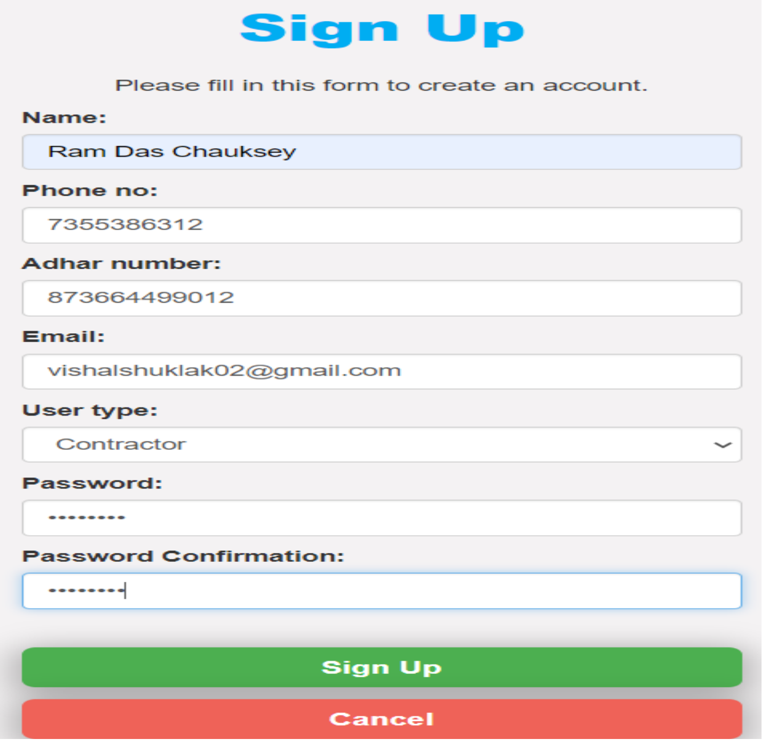
Figure-36: Admin home page

5.5 IMPLEMENTATION AND RESULTS

This section will show the step-wise implementation of this “Migrant e-portal”.Since synthetic data is being used in this study, to view the database please refer to Annexure- 2. There are three types of users in this portal. They are contractors, migrant workers and agents.

5.5.1 Contractor

The contractor in this portal will act as an employer who will post the job according to his needs. The first and the most basic step is to sign-up or register by filling in the basic information which is asked during registration.



Sign Up

Please fill in this form to create an account.

Name: Ram Das Chauksey

Phone no: 7355386312

Adhar number: 873664499012

Email: vishalshuklak02@gmail.com

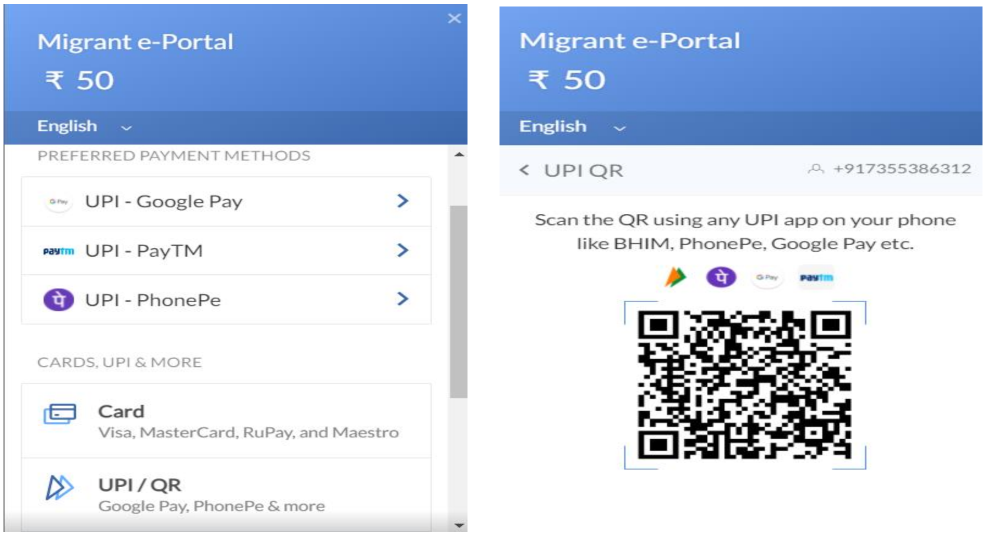
User type: Contractor

Password: ·········

Password Confirmation: ·········

Figure-37: Contractor registration

After getting registered, the contractor has to pay the subscription amount, so that he can use all the facilities of this portal. Payment can be done through cards which can be Visa, MasterCard or Rupay.Net banking or UPI can also be used for payment.



Migrant e-Portal

₹ 50

English

PREFERRED PAYMENT METHODS

UPI - Google Pay
UPI - PayTM
UPI - PhonePe

CARDS, UPI & MORE

Card
Visa, MasterCard, RuPay, and Maestro
UPI / QR
Google Pay, PhonePe & more

Migrant e-Portal

₹ 50

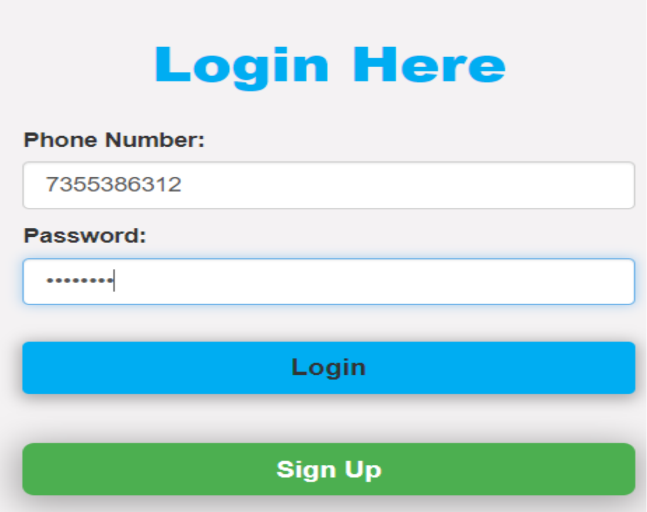
English

UPI QR +917355386312

Scan the QR using any UPI app on your phone like BHIM, PhonePe, Google Pay etc.

Figure-38: Payment is done by the contractor after registration

Then the contractor can log in by using the phone number and password.

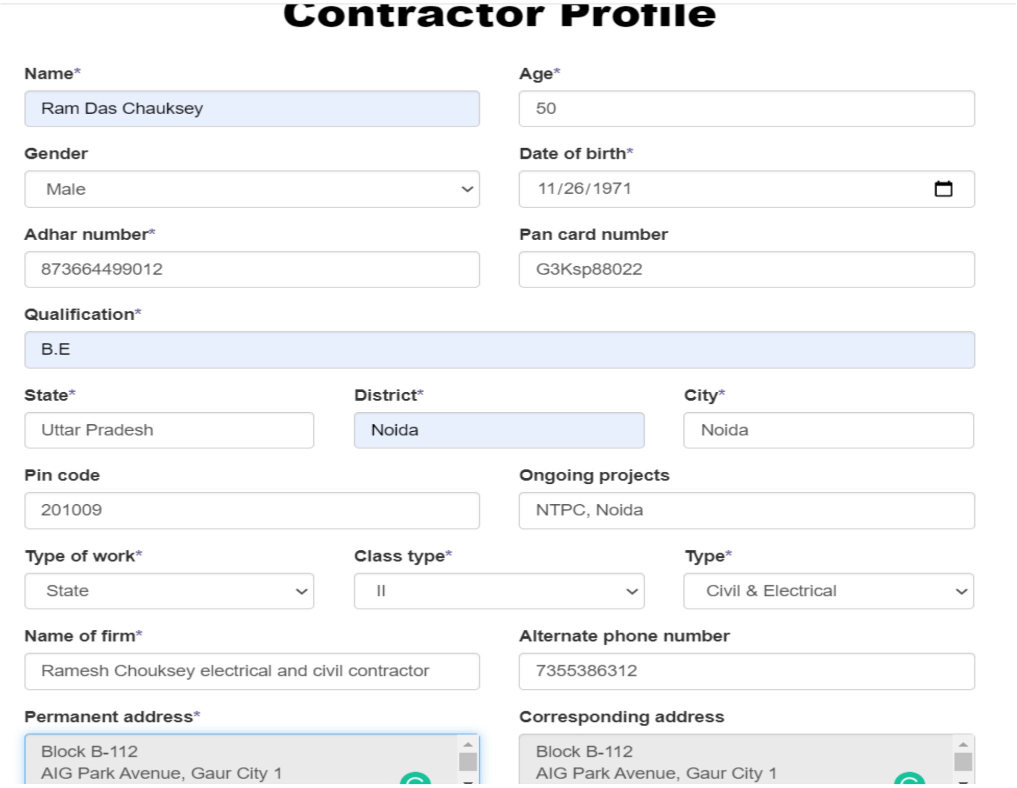


Login Here

Phone Number:	7355386312
Password:	••••••••

Figure-39: Contractor Login

After that, the contractor now complete/edit his profile by filling in all the details which are being asked.

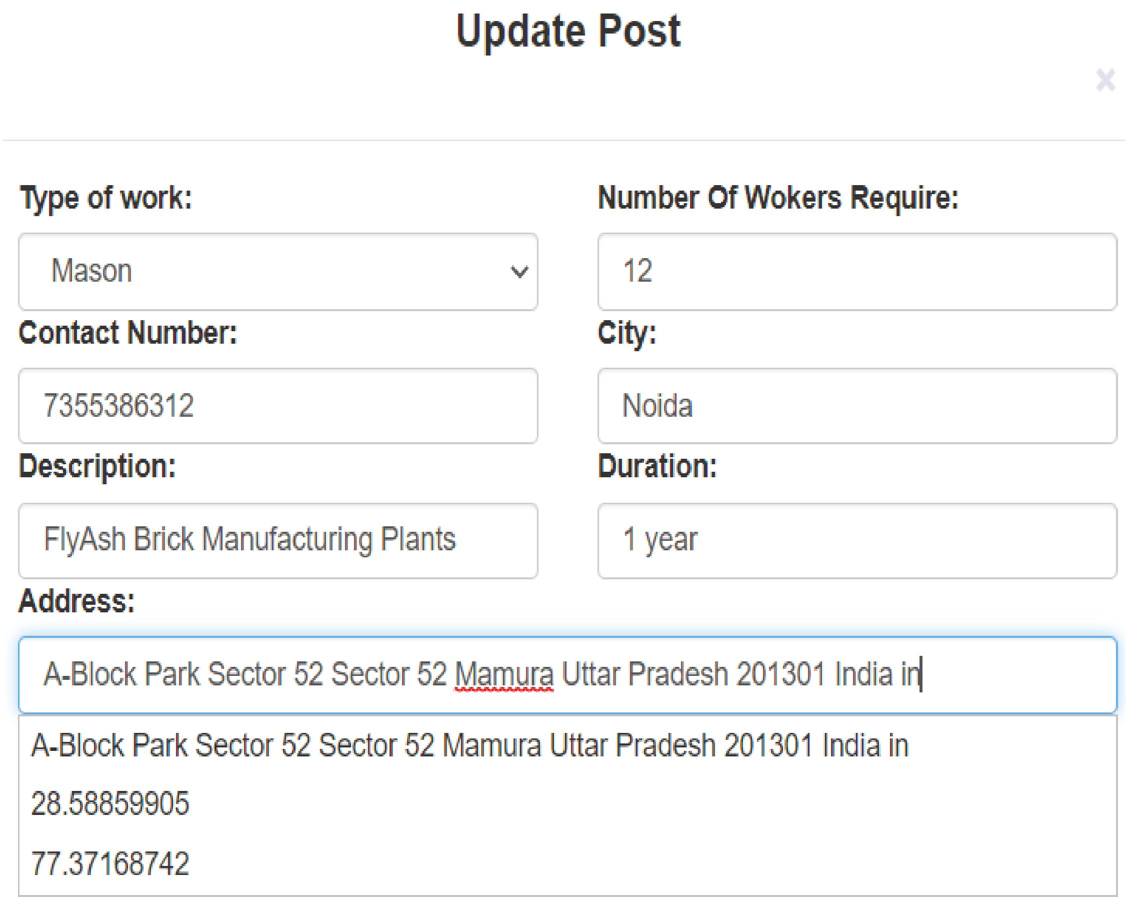


Contractor Profile

Name*	Ram Das Chauksey
Age*	50
Gender	Male
Date of birth*	11/26/1971
Adhar number*	873664499012
Pan card number	G3Ksp88022
Qualification*	B.E
State*	Uttar Pradesh
District*	Noida
City*	Noida
Pin code	201009
Ongoing projects	NTPC, Noida
Type of work*	State
Class type*	II
Type*	Civil & Electrical
Name of firm*	Ramesh Chouksey electrical and civil contractor
Alternate phone number	7355386312
Permanent address*	Block B-112 AIG Park Avenue, Gaur City 1
Corresponding address	Block B-112 AIG Park Avenue, Gaur City 1

Figure-40: Contractor profile page

Now, after completing all the procedures the contractor can post the job. The address provided by the contractor will get converted into latitude and longitude for geospatial analysis.



Update Post

Type of work:

Mason

Number Of Wokers Require:

12

Contact Number:

7355386312

City:

Noida

Description:

FlyAsh Brick Manufacturing Plants

Duration:

1 year

Address:

A-Block Park Sector 52 Sector 52 Mamura Uttar Pradesh 201301 India in

A-Block Park Sector 52 Sector 52 Mamura Uttar Pradesh 201301 India in

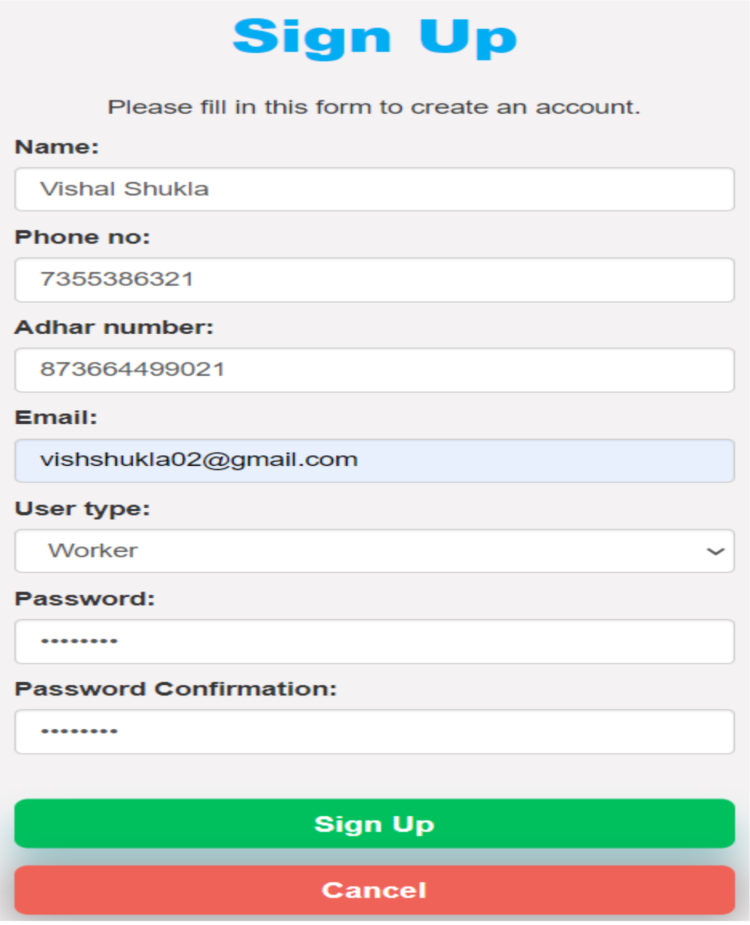
28.58859905

77.37168742

Figure-41: Job posted by the contractor

5.5.2 Migrant Worker

The migrant worker will act as an employee in this portal. He is the one who is looking for a job. The first step for the worker will also be the same as that of the contractor i.e. the worker needs to first register themselves by filling in the information which is asked during registration.



Sign Up

Please fill in this form to create an account.

Name: Vishal Shukla

Phone no: 7355386321

Adhar number: 873664499021

Email: vishshukla02@gmail.com

User type: Worker

Password: •••••••••

Password Confirmation: •••••••••

Figure-42: Worker registration

After successfully getting registered, they will be asked to pay the subscription amount. Payment can be done through cards which can be Visa, MasterCard or Rupay.Net banking or UPI can also be used for payment.

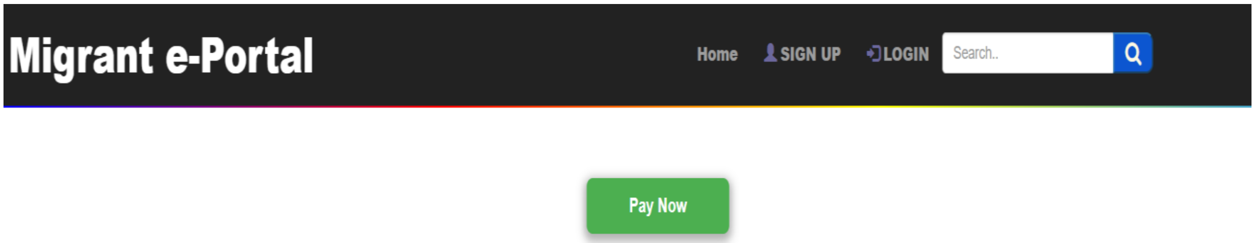
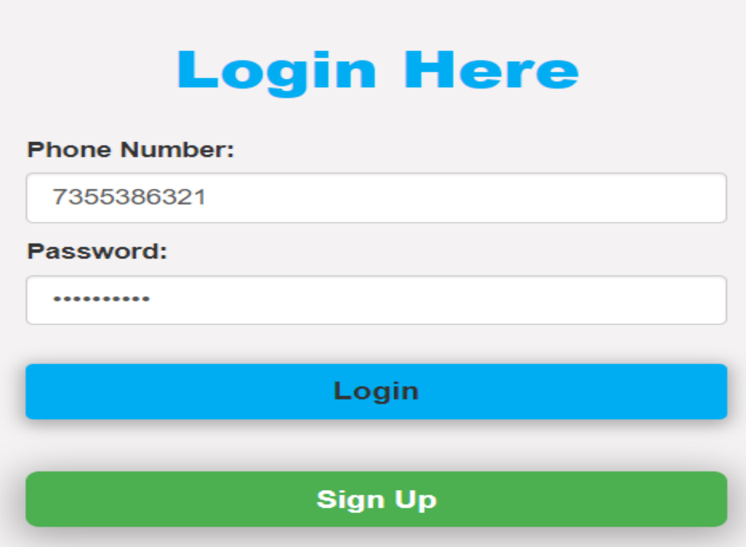


Figure-43: After registration, the worker has to pay the subscription amount

After which they can log in by filling in the phone number and password which was provided by them during the registration.

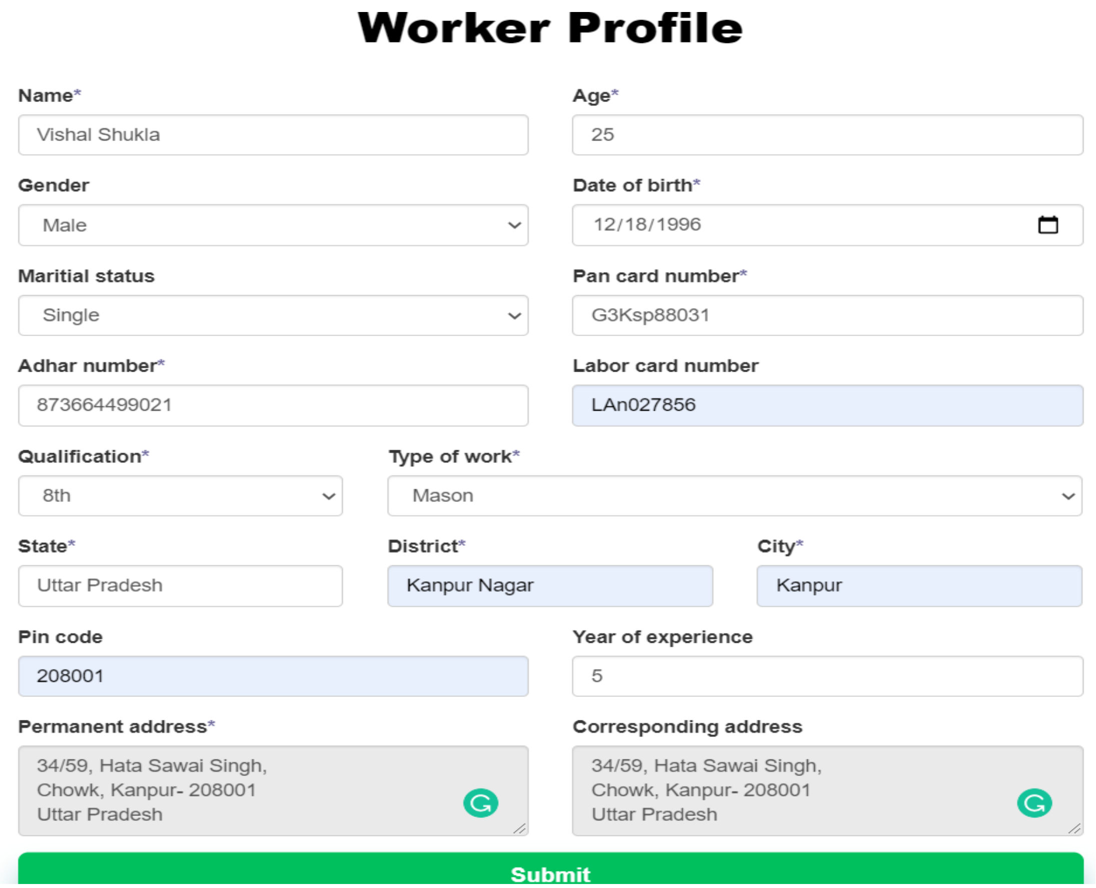


Phone Number:
7355386321

Password:
•••••••••••

Figure-44: Worker Login

Now, they can complete their profile by providing the information which is asked in it.



Worker Profile

Name*
Vishal Shukla

Age*
25

Gender
Male

Date of birth*
12/18/1996

Maritial status
Single

Pan card number*
G3Ksp88031

Adhar number*
873664499021

Labor card number
LAn027856

Qualification*
8th

Type of work*
Mason

State*
Uttar Pradesh

District*
Kanpur Nagar

City*
Kanpur

Pin code
208001

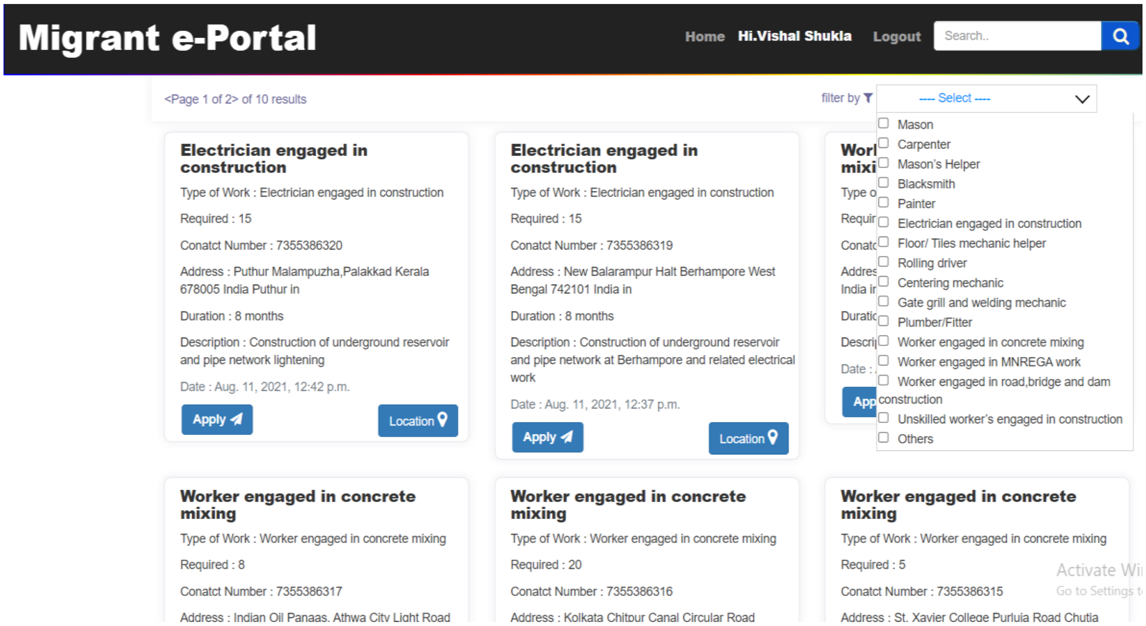
Year of experience
5

Permanent address*
34/59, Hata Sawai Singh,
Chowk, Kanpur- 208001
Uttar Pradesh

Corresponding address
34/59, Hata Sawai Singh,
Chowk, Kanpur- 208001
Uttar Pradesh

Figure-45: Worker filling the information

Now, they will be able to see the jobs which are active and can use the search filter to find the job of their interest.



Migrant e-Portal   Home   Hi.Vishal Shukla   Logout   Search..

<Page 1 of 2> of 10 results   filter by   ---- Select ----

□ Mason
□ Carpenter
□ Mason's Helper
□ Blacksmith
□ Painter
□ Electrician engaged in construction
□ Floor/ Tiles mechanic helper
□ Rolling driver
□ Centering mechanic
□ Gate grill and welding mechanic
□ Plumber/Fitter
□ Worker engaged in concrete mixing
□ Worker engaged in MNREGA work
□ Worker engaged in road,bridge and dam construction
□ Unskilled worker's engaged in construction
□ Others

Electrician engaged in construction

Type of Work : Electrician engaged in construction

Required : 15

Conatct Number : 7355386320

Address : Puthur Malampuzha,Palakkad Kerala 678005 India Puthur in

Duration : 8 months

Description : Construction of underground reservoir and pipe network lightening

Date : Aug. 11, 2021, 12:42 p.m.

Apply   Location

Electrician engaged in construction

Type of Work : Electrician engaged in construction

Required : 15

Conatct Number : 7355386319

Address : New Balarampur Halt Berhampore West Bengal 742101 India in

Duration : 8 months

Description : Construction of underground reservoir and pipe network at Berhampore and related electrical work

Date : Aug. 11, 2021, 12:37 p.m.

Apply   Location

Worl mixi

Type o

Requir

Conatc

Addres India ir

Duratic

Descri

Date :

App

Worker engaged in concrete mixing

Type of Work : Worker engaged in concrete mixing

Required : 8

Conatct Number : 7355386317

Address : Indian Oil Panaas. Athwa Citv Liaht Road

Worker engaged in concrete mixing

Type of Work : Worker engaged in concrete mixing

Required : 20

Conatct Number : 7355386316

Address : Kolkata Chitpur Canal Circular Road

Worker engaged in concrete mixing

Type of Work : Worker engaged in concrete mixing

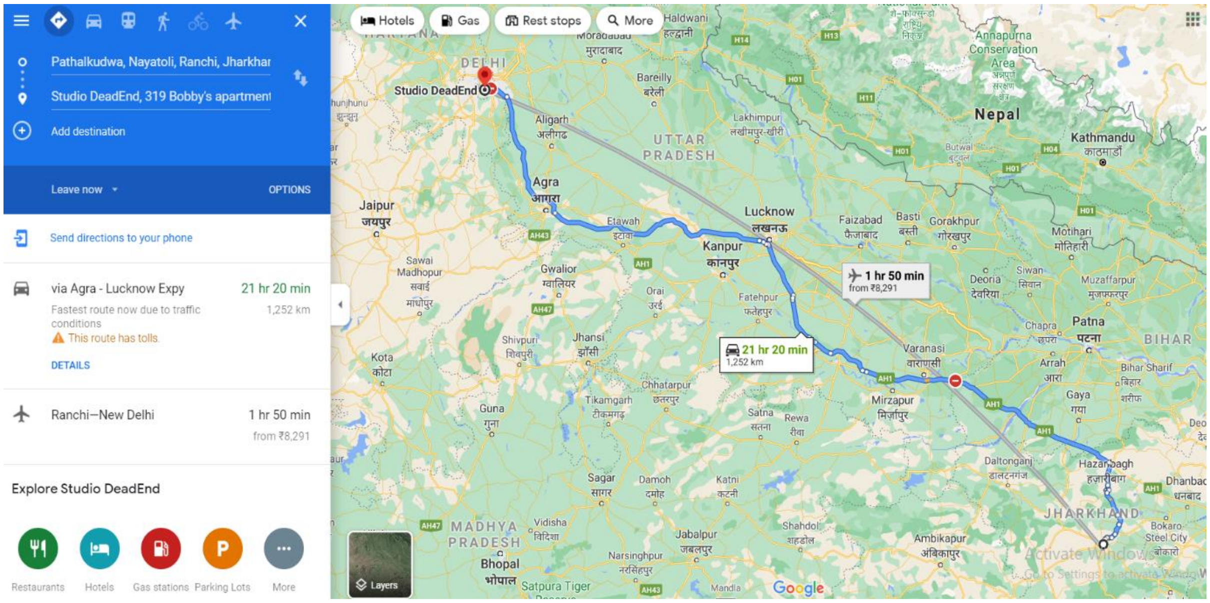
Required : 5

Conatct Number : 7355386315

Address : St. Xavier College Purluia Road Chutia

Figure-46: Worker searching for the job

By clicking on the location button, a geographically referenced map will open which will show the location of the workplace and will also provide some additional information like shortest path distance, optimal route, how much time it will take to reach that location etc.



Pathalkudwa, Nayatoli, Ranchi, Jharkhar

Studio DeadEnd, 319 Bobby's apartment

Add destination

Leave now   OPTIONS

Send directions to your phone

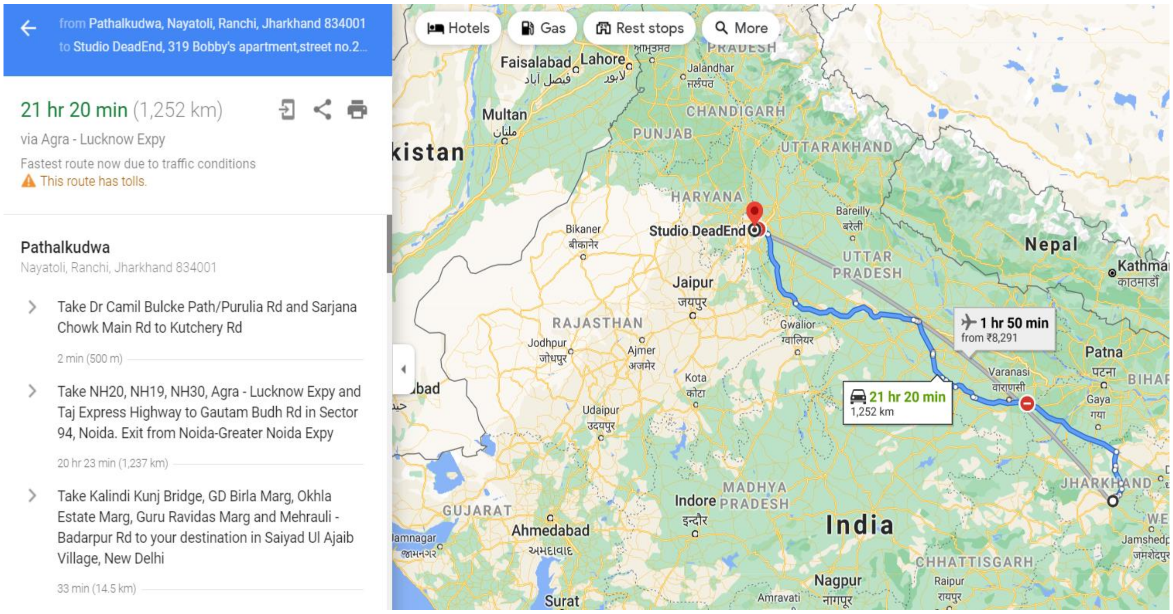
via Agra - Lucknow Expy   21 hr 20 min
Fastest route now due to traffic conditions   1,252 km
⚠ This route has tolls.
DETAILS

Ranchi–New Delhi   1 hr 50 min
from ₹8,291

Explore Studio DeadEnd

Restaurants   Hotels   Gas stations   Parking Lots   More

Figure-47: Optimal route map



from Pathalkudwa, Nayatoli, Ranchi, Jharkhand 834001
to Studio DeadEnd, 319 Bobby's apartment,street no.2...

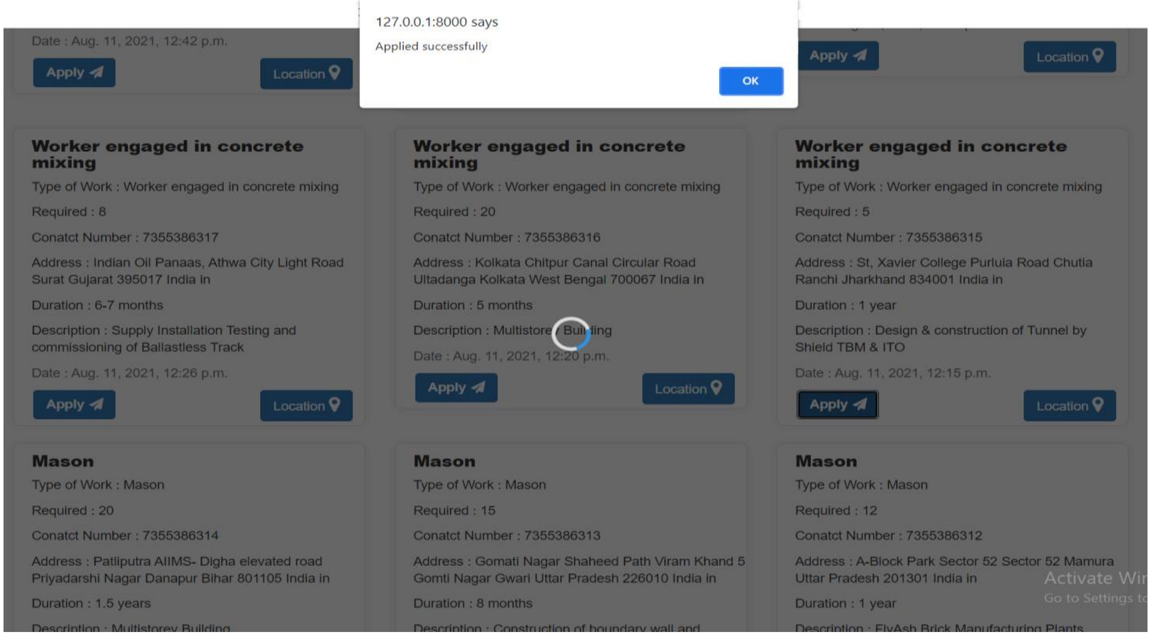
21 hr 20 min (1,252 km)
via Agra - Lucknow Expy
Fastest route now due to traffic conditions
⚠ This route has tolls.

Pathalkudwa
Nayatoli, Ranchi, Jharkhand 834001

- Take Dr Camil Bulcke Path/Purulia Rd and Sarjana Chowk Main Rd to Kutchery Rd
  2 min (500 m)
- Take NH20, NH19, NH30, Agra - Lucknow Expy and Taj Express Highway to Gautam Budh Rd in Sector 94, Noida. Exit from Noida-Greater Noida Expy
  20 hr 23 min (1,237 km)
- Take Kalindi Kunj Bridge, GD Birla Marg, Okhla Estate Marg, Guru Ravidas Marg and Mehrauli - Badarpur Rd to your destination in Saiyad Ul Ajaib Village, New Delhi
  33 min (14.5 km)

Figure-48: Optimal route information

The worker can apply for the job just by clicking on the apply button and the mail will be sent to the contractor at his registered e-mail address.



127.0.0.1:8000 says
Applied successfully
OK

Date : Aug. 11, 2021, 12:42 p.m.
Apply
Location

Apply
Location

Worker engaged in concrete mixing
Type of Work : Worker engaged in concrete mixing
Required : 8
Conatct Number : 7355386317
Address : Indian Oil Panaas, Athwa City Light Road Surat Gujarat 395017 India in
Duration : 6-7 months
Description : Supply Installation Testing and commissioning of Ballastless Track
Date : Aug. 11, 2021, 12:26 p.m.
Apply
Location

Worker engaged in concrete mixing
Type of Work : Worker engaged in concrete mixing
Required : 20
Conatct Number : 7355386316
Address : Kolkata Chitpur Canal Circular Road Ultadanga Kolkata West Bengal 700067 India in
Duration : 5 months
Description : Multistorey Building
Date : Aug. 11, 2021, 12:20 p.m.
Apply
Location

Worker engaged in concrete mixing
Type of Work : Worker engaged in concrete mixing
Required : 5
Conatct Number : 7355386315
Address : St, Xavier College Purluia Road Chutia Ranchi Jharkhand 834001 India in
Duration : 1 year
Description : Design & construction of Tunnel by Shield TBM & ITO
Date : Aug. 11, 2021, 12:15 p.m.
Apply
Location

Mason
Type of Work : Mason
Required : 20
Conatct Number : 7355386314
Address : Patliputra AIIMS- Digha elevated road Priyadarshi Nagar Danapur Bihar 801105 India in
Duration : 1.5 years
Description : Multistorey Building

Mason
Type of Work : Mason
Required : 15
Conatct Number : 7355386313
Address : Gomati Nagar Shaheed Path Viram Khand 5 Gomti Nagar Gwari Uttar Pradesh 226010 India in
Duration : 8 months
Description : Construction of boundary wall and

Mason
Type of Work : Mason
Required : 12
Conatct Number : 7355386312
Address : A-Block Park Sector 52 Sector 52 Mamura Uttar Pradesh 201301 India in
Duration : 1 year
Description : FlyAsh Brick Manufacturing Plants

Figure-49: Worker has successfully applied for the job

CHAPTER 6

CONCLUSION AND FUTURE SCOPE

6.1 CONCLUSION

Due to the COVID-19, the country was put under lockdown by the government to protect the individuals from getting affected by the virus. The adverse effect of this decision was that many individuals lost their jobs but most of them were migrant workers which work on daily wages. In the current scenario, the situation is getting back to normal but still, most migrant workers are jobless. This research mainly focused on the construction sector and how COVID affect this sector which is thoroughly explained in Chapter 2. To overcome this problem, this research recommends a GIS-based WebGIS application, “Migrant e-portal”, which helps the migrant workers, contractors and agents to register themselves on the same platform from where they can communicate with each other directly.

The important findings of this research are how geospatial technologies can be useful in the construction sector and how they can be used to minimise the movement of workers by providing jobs in their native place. The geographical information which will be provided by the migrant worker, contractor and agent can be used for geospatial analysis which helps them to gain additional information like shortest path distance, optimal route, tolls information etc. The portal which is built in this research will help the migrant worker to get the job according to their interest and in their native or nearby places and it will also help the contractor to get skilled and efficient workers. Geospatial analysis is becoming more popular, and everyone has started to recognise the value of geographically referenced data. This research has sketched the basic idea of using geospatial technologies in the construction sector and its benefits.

6.2 FUTURE SCOPE

COVID-19 hits the country very hard and make us realise the limitations of the existing system in the construction sector and provide us with the opportunity to think of new ideas/solutions to solve those limitations. This research put one foot forward in this direction and proposed one possible solution that can be implemented in the construction sector. Many more ideas can be added to this research to make it more efficient, but time limitation does not allow to explore those ideas. In future, this study can be further extended using ideas that are mentioned below:

- Machine learning and Deep Learning algorithms also can be used to understand the trend like in which city/district/state most of the migrant workers prefer to work, what type of work they prefer to do so that future prediction can be done by studying the past trends.
- This study can further get extended in the agriculture sector because there is also no formal framework that can help migrant workers to get jobs in this sector.
- This portal in future can also be used for labours emigration.
- Due to COVID-19, synthetic data is being used in this research but when we get the genuine data from the various competent authorities then this data can be used to check how many of them are registered and how many are not and can ask them to get registered in their respective government portal to avail all the facilities provided by their government.
- Data can also be used to understand the trend like from which state maximum migration took place. This information can be used by the government of the respective state to make changes in the policies that are already been made so that migration can get minimised.

REFERENCES

[1] C. C. Lai, T. P. Shih, W. C. Ko, H. J. Tang, and P. R. Hsueh, "Severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2) and coronavirus disease-2019 (COVID-19): The epidemic and the challenges," *Int. J. Antimicrob. Agents*, vol. 55, no. 3, p. 105924, 2020, doi: 10.1016/j.ijantimicag.2020.105924.

[2] Y. Liu, A. A. Gayle, A. Wilder-Smith, and J. Rocklöv, "The reproductive number of COVID-19 is higher compared to SARS coronavirus," *J. Travel Med.*, vol. 27, no. 2, pp. 1–6, 2020, doi: 10.1093/jtm/taaa021.

[3] World Health Organization (WHO), "Statement on the second meeting of the International Health Regulations (2005) Emergency Committee regarding the outbreak of novel coronavirus (2019-nCoV)," *Geneva, Switz.*, no. 2005, pp. 1–6, 2005, Accessed: Jul. 08, 2021. [Online]. Available: https://www.who.int/news/item/30-01-2020-statement-on-the-second-meeting-of-the-international-health-regulations-(2005)-emergency-committee-regarding-the-outbreak-of-novel-coronavirus-(2019-ncov).

[4] J. Kaplan, L. Frias, and M. McFall-Johnsen, "Countries That Are on Lockdown Because of Coronavirus," *INSIDER*, 2020. https://www.businessinsider.com/countries-on-lockdown-coronavirus-italy-2020-3?r=US&IR=T#many-countries-have-also-closed-borders-to-prevent-international-travelers-from-spreading-the-virus-29. (accessed Jul. 08, 2021).

[5] The Lancet, "India under COVID-19 lockdown," *Lancet*, vol. 395, no. 10233, p. 1315, 2020, doi: 10.1016/S0140-6736(20)30938-7.

[6] G. of I. Central Statistics Office, "Press Note on Provisional Estimates of Annual National

Income 2014-15 and Quarterly Estimates of Gross Domestic Product for the Fourth Quarter (Q4) 2014-15 Central Statistics Office Ministry of Statistics & Programme Implementation," no. May, 2015, [Online]. Available: http://mospi.nic.in/Mospi_New/upload/nad_press_release_29may15.pdf.

[7] https://www.cmie.com/, "CMIE." https://www.cmie.com/ (accessed Aug. 04, 2021).

[8] International Labour Organization, *World employment and social outlook: trends 2020*. 2020.

[9] K. E. Y. Messages, G. Overview, O. F. L. Return, and O. F. Migrant, "Reverse migration to rural areas of origin in the context of the COVID-19 pandemic," *Reverse Migr. to Rural areas Orig. Context COVID-19 pandemic*, no. May, pp. 1–18, 2021, doi: 10.4060/cb4712en.

[10] A. Khanna, "Impact of Migration of Labour Force due to Global COVID-19 Pandemic with Reference to India," *J. Health Manag.*, vol. 22, no. 2, pp. 181–191, 2020, doi: 10.1177/0972063420935542.

[11] N. Barker *et al.*, "Migration and the labour market impacts of COVID-19," no. WIDER Working Paper No. 2020/139, 2020.

[12] H. Dingle and V. Alistair Drake, "What is migration?," *Bioscience*, vol. 57, no. 2, pp. 113–121, 2007, doi: 10.1641/B570206.

[13] L. A. Pastalan, "Demographic characteristics," *Pastalan, L. A. (1983). Demogr. Charact. J. Hous. Elderly, 1(1), 85–87. https//doi.org/10.1300/J081V01N01_08Journal Hous. Elder.*, vol. 1, no. 1, pp. 85–87, 1983, doi: 10.1300/J081V01N01_08.

[14] L. Chakraborty and E. Thomas, “Covid-19 and macroeconomic uncertainty: Fiscal and monetary policy response,” *Econ. Polit. Wkly.*, vol. 55, no. 15, pp. 15–18, 2020.

[15] R. O. P. E. S. The Census of India, “The Census of india 2011 report pdf,” *Goi*, 2011, [Online].Available:https://www.google.co.in/search?client=safari&rls=en&q=census+of+India+2011+report&ie=UTF-8&oe=UTF-8&gfe_rd=cr&ei=dia4VqrsJceL8QeXuom4Aw#q=census+of+india+2011+report+pdf.

[16] J. Gregg, *No Country for Earthmen*, no. June. 2021.

[17] National Statistical Office, “Annual Report: Periodic Labour Force Survey, 2018-19,” *Annu. Rep. Period. Labour Force Surv. 2018-19*, pp. 0–643, 2019.

[18] K. Mahajan and R. Nagaraj, “Rural construction employment boom during 2000-12: Evidence from NSSO surveys,” *Econ. Polit. Wkly.*, vol. 52, no. 52, pp. 54–63, 2017.

[19] N. V. Naveen P. Singh, “Key words : Labour out-migration , Indo Gangetic plains , Logit model , farm efficiency , Data Envelopment Analysis . Introduction,” pp. 1–24.

[20] P. Waingankar, R. Taralekar, and P. Thatkar, “A Study to Assess Pattern of Migration across India Based on Census Data,” *Int. J. Recent Trends Sci. Technol.*, vol. 5, no. 2, pp. 2012–74, 2012, [Online]. Available: https://www.researchgate.net/publication/328334146.

[21] UNESCO-UNICEF, *Internal Migration in India Initiative National Workshop on Internal Migration and Human Development in India*, vol. 2, no. December. 2011.

[22] Census of India, “Census of India : Population Enumeration Data (Final Population),” *The Registrar General & Census Commissioner, New Delhi, Ministry of Home Affairs, Government of India.*, 2011.

https://censusindia.gov.in/2011census/population_enumeration.aspx (accessed Aug. 04, 2021).

[23] A. Pandey, “Laws related to Migrant Labourers in India - iPleaders,” *Article*, 2017. https://blog.ipleaders.in/laws-related-to-migrant-labourers-in-india/ (accessed Jul. 11, 2021).

[24] N. S. OFFICE, M. S. & PROGRAMME, IMPLEMENTATION, and G. O. INDIA, “Quarterly GDP Growth of India - StatisticsTimes.com,” 2020. https://statisticstimes.com/economy/country/india-gdp-growth-sectorwise.php (accessed Aug. 04, 2021).

[25] ILO, “Impact of COVID-19 on the construction sector The,” no. January, pp. 1–12, 2021.

[26] A. Biswas, A. Ghosh, A. Kar, T. Mondal, B. Ghosh, and P. K. Bardhan, “The impact of COVID-19 in the construction sector and its remedial measures,” *J. Phys. Conf. Ser.*, vol. 1797, no. 1, 2021, doi: 10.1088/1742-6596/1797/1/012054.

[27] J. Myren and G. Hellers, “Introduction and acknowledgements,” *Scand. J. Gastroenterol.*, vol. 24, no. S161, p. 1, 1989, doi: 10.3109/00365528909091048.

[28] P. Verma, “Informal Labour and Dynamics of the Construction Sector in India,” *ritimo*, pp. 1–5, 2013, Accessed: Aug. 04, 2021. [Online]. Available: https://www.ritimo.org/Informal-Labour-and-Dynamics-of-the-Construction-Sector-in-India.

[29] I. M. Ishaq, R. Omar, and M. Y. Yahya, “Improving communication between client and contractor during a construction project in the Nigerian construction industry.,” *J. Technol. Manag. Bus.*, vol. 6, pp. 60–75, 2019.

[30] P. L. Michael John De Smith, Michael F. Goodchild, *Geospatial Analysis: A Comprehensive Guide to Principles, Techniques and ... - Michael John De Smith, Michael F. Goodchild, Paul Longley - Google Books*. 2007.

[31] O. Buyukokkten, J. Cho, H. Garcia-Molina, L. Gravano, and N. Shivakumar, "Exploiting Geographical Location Information of Web Pages," *Proc. ACM SIGMOD Work. Web Databases - WebDB'99*, pp. 91–96, 1999, [Online]. Available: http://ilpubs.stanford.edu:8090/415/1/1999-5.pdf.

[32] T. M. Rhyne and A. MacEachren, "Visualizing geospatial data," *ACM SIGGRAPH 2004 Course Notes, SIGGRAPH 2004*, 2004, doi: 10.1145/1103900.1103931.

[33] IEEE UAE Student Branch, "Distance in Classification," 2016.

[34] D. K. Fan and P. Shi, "Improvement of Dijkstraa's algorithm and its application in route planning," *Proc. - 2010 7th Int. Conf. Fuzzy Syst. Knowl. Discov. FSKD 2010*, vol. 4, no. Fskd, pp. 1901–1904, 2010, doi: 10.1109/FSKD.2010.5569452.

[35] F. Menczer, S. Fortunato, and C. A. Davis, "Python Tutorial," in *A First Course in Network Science*, 2020, pp. 221–237.

[36] W.-F. Foundation, "Web Frameworks - Full Stack Python," *Text*, 1998. https://www.fullstackpython.com/web-frameworks.html (accessed Aug. 04, 2021).

[37] D. S. Foundation, "Django Documentation," 2012, Accessed: Aug. 04, 2021. [Online]. Available: https://docs.djangoproject.com/en/3.2/.

[38] NaveenArora and Annianni, "Django Project MVT Structure - GeeksforGeeks." 2004, Accessed: Aug. 04, 2021. [Online]. Available: https://www.geeksforgeeks.org/django-

project-mvt-structure/.

[39] E. Einführung and T. Z.- Absatzformatierung, “HTML Tutorial,” *Text*, vol. 1, pp. 1–9, 2002, Accessed: Aug. 04, 2021. [Online]. Available: https://www.w3schools.com/html/.

[40] E. Einführung, “CSS Tutorial,” *Text*, 2002. https://www.w3schools.com/css/ (accessed Aug. 04, 2021).

[41] javascript.org, “The Modern JavaScript Tutorial,” *Text*, 2020. https://javascript.info/ (accessed Aug. 04, 2021).

[42] W. You and S. Already, “AJAX Introduction «,” *text*, vol. II, pp. 1–27, 2005, Accessed: Aug. 04, 2021. [Online]. Available: https://www.w3schools.com/xml/ajax_intro.asp.

[43] Sqlservertutorial, “SQL Server Tutorial,” *Sqlservertutorial*, 2020. https://www.sqlservertutorial.net/ (accessed Aug. 04, 2021).

[44] LocationIQ, “LocationIQ - Free & Fast Geocoding, Reverse Geocoding and Maps service,” *text*, 2007. https://locationiq.com/docs (accessed Aug. 04, 2021).

[45] Google LLC, “About – Google Maps,” 2020. https://www.google.com/maps/about/#!/ (accessed Aug. 04, 2021).

[46] Subham Kumar, “Razorpay Docs,” *Site*, 2006. https://razorpay.com/docs/ (accessed Aug. 04, 2021).

ANNEXURE-1: DATA COLLECTION

District wise DM list of four major states i.e. Uttar Pradesh, Bihar, Delhi and Punjab

UTTAR PRADESH DISTRICT MAGISTRATE LIST:

Sl no:	**District name**	**District Magistrate Name**	**Email Address**	**Contact No:**
1	Agra	Shri Prabhu N Singh	dmagr@nic.in	9454417509
2	Aligarh	Chandra Bhushan Singh I.A.S.	dmali@nic.in	9454417513
3	Allahabad	Shri Sanjay Kumar Khatri	dmall@nic.in	9454417517
4	Ambedkar Nagar	Shri Samuel Paul N IAS	dmamb@nic.in	9454417539
5	Amethi	Sh. Arun Kumar,IAS	dmamethi-up@nic.in	9454418891
6	Amroha	Shri Bal Krishna Tripathi, IAS	dmjpn@nic.in	9454417571
7	Auraiya	Shri Sunil Kumar Verma, I.A.	dmaur@nic.in	9454417550
8	Azamgarh	Rajesh Kumar (IAS)	dmaza@nic.in	9454417521
9	Baghpat	Shri Raj Kamal Yadav (IAS)	dmbag@nic.in	9454417562
10	Bahraich	Dr. Dinesh Chandra, I.A.S.	dmbeh@nic.in	9454417535
11	Ballia	Aditi Singh, IAS	dmbal@nic.in	9454417522
12	Balrampur	Smt. Shruti	dmblr@nic.in	9454417536
13	Banda	Shri Anand Kumar Singh(IAS)	dmban@nic.in	9454417531
14	Barabanki	Dr Adarsh Singh, IAS	dmbab@nic.in	9454417540
15	Bareilly	Mr. Nitish Kumar, IAS	dmbar@nic.in	9454417524
16	Basti	Saumya Agarwal	dmbas@nic.in	9454417528
17	Bhadohi	Ms. Aryaka Akhoury, I.A.S	dmsrn@nic.in	9454417568
18	Bijnor	Shri Umesh Mishra (IAS)	dmbij@nic.in	9454417570
19	Budaun	Mrs Deepa Ranjan (I.A.S.)	dmbud@nic.in	9454417525

20	Bulandshahr	Mr. Ravindra Kumar IAS	dmbul@nic.in	9454417563
21	Chandauli	Shri Sanjeev Singh, IAS	dmchn@nic.in	9454417576
22	Chitrakoot	Shubhrant Kumar Shukla (IAS)	dmchi@nic.in	9454417532
23	Deoria	Shri Ashutosh Niranjan IAS	dmdeo@nic.in	9454417543
24	Etah	ANKIT KUMAR AGRAWAL ,I.A.S	dmeth@nic.in	9454417514
25	Etawah	Mrs. Shruti Singh, I.A.S	dmetw@nic.in	9454417551
26	Faizabad	Anuj Kumar Jha	dmfai@nic.in	9454417541
27	Farrukhabad	Sh. Manvendra Singh, I.A.S.	dmfar@nic.in	9454417552
28	Fatehpur	Apurva Dubey (I.A.S)	dmfat@nic.in	9454417518
29	Firozabad	Chandra Vijay Singh IAS	dmfir@nic.in	9454417510
30	Gautam Buddha nagar	Mr. Suhas Lalinakere Yathiraj	dmgbn@nic.in	9971020646
31	Gaziabad	Shri Rakesh Kumar Singh I.A.S.	dmgha@nic.in	9454417565
32	Ghazipur	Shri Mangala Prasad Singh	dmgaz@nic.in	9454417577
33	Gonda	Mr. Markandey Shahi (I.A.S.)	dmchi@nic.in	9454417532
34	Gorakhpur	Sh. K. Vijayendra Pandian (IAS)	dmgor@nic.in	9454417544
35	Hamirpur	Shri Gyaneshwar Tripathi IAS	dmham@nic.in	9454417533
36	Hapur	ANUJ SINGH, I.A.S	dmhap@nic.in	9454449848
37	Hardoi	Mr.Avinash Kumar	dmhar@nic.in	9454417556
38	Hathras	Ramesh Ranjan (IAS)	dmhat@nic.in	05722227076
39	Jalaun	Mrs. Priyanka Niranjan	dmjal@nic.in	9454417548
40	Jaunpur	Shri Manish Kumar Verma (I.A.S.)	dmjau@nic.in	9454417578
41	Jhansi	Shri Andra Vamsi (IAS)	dmjha@nic.in	9454417547
42	Kannauj	Rakesh Kumar Mishra IAS	dmknj@nic.in	9454417555

43	Kanpur Dehat	Shri Jitendra Pratap Singh I A S	dmkan@nic.in	9454417553
44	Kanpur Nagar	Sh. Alok Tiwari, IAS	dmkap@nic.in	9454417554
45	Kashiraam Nagar (kasganj)	Shri Chandra Prakash Singh IAS	dmkr-up@nic.in	9454417516
46	Kaushambi	Mr. SUJEET KUMAR, IAS	dmkos@nic.in	9454417519
47	Kushinagar	S. Rajalingam (IAS)	dmkus@nic.in	9454417545
48	Lakhimpur-Kheri	Dr. Arvind Kumar Chaurasiya, IAS	dmlkh@nic.in	9454417558
49	Lalitpur	annavi dineshkumar	dmlal@nic.in	9454417549
50	Lucknow	Shri Abhishek Prakash (I.A.S.)	dmluc@nic.in	9415005000
51	Maharajganj	Dr. Ujjwal Kumar (IAS)	dmmah@nic.in	9454417546
52	Mahoba	Mr. Satyendra Kumar(IAS)	dmmab@nic.in	9454417534
53	Mainpuri	Mahendra Bahadur Singh	dmmai@nic.in	9454417511
54	Mathura	Shri Navneet Singh Chahal, IAS	dmmat@nic.in	9454417512
55	Mau	Shri Amit Singh Bansal, IAS	dmmau@nic.in	9454417523
56	Meerut	Sh. K. Balaji IAS	dmmee@nic.in	9454417566
57	Mirzapur	Shri Praveen kumar laxkar (I.A.S.)	dmmir@nic.in	9454417567
58	Moradabad	Shri Shailendra Kumar Singh	dmmor@nic.in	9454417572
59	Muzaffarnagar	Mrs. Selva Kumari J. , IAS	dmmuz@nic.in	9454417574
60	Pilibhit	Pulkit Khare (I.A.S.)	dmpil@nic.in	9454417526
61	Pratapgarh	Dr.Nitin Bansal	dmpra@nic.in	9454417520
62	Raebareli	Vaibhav Srivastava (I.A.S.)	dmrae@nic.in	9454417559
63	Rampur	Ravindra Kumar Mander	dmram@nic.in	9454417573
64	Saharanpur	Shri Akhilesh Singh IAS	dmsah@nic.in	9454417575
65	Sambhal	Mr. Sanjeev Ranjan	dmbm-up@nic.in	9454416890

66	Sant Kabir Nagar	Ms. Divya Mittal (IAS)	dmskn@nic.in	9454417529
67	Shahjahanpur	Indra Vikram Singh	dmsha@nic.in	9454417527
68	Shamli	Ms Jasjit Kaur, IAS	dmshm@nic.in	9454416996
69	Shravasti	Mr. T.K.Shibu IAS	dmshr@nic.in	9454417538
70	Siddharth nagar	Mr. Deepak Meena, IAS	dmsid@nic.in	9454417530
71	Sitapur	Mr. Vishal Bharadwaj	dmsit@nic.in	9454417560
72	Sonbhadra	Shri Abhishek Singh I.A.S.	dmson@nic.in	9454417569
73	Sultanpur	Mr. Raveesh Gupta, IAS	dmsul@nic.in	9454417542
74	Unnao	Shri Ravindra Kumar	dmunn@nic.in	9454417561
75	Varanasi	Shri Kaushal Raj Sharma (I.A.S.)	dmvar@nic.in	9454417579

Source:http://districts.nic.in/districts.php?sid=UP

http://uphome.gov.in/DM-UP-Contact.htm

BIHAR DISTRICT MAGISTRATE LIST:

Sl no:	**District name**	**District Magistrate Name**	**Email Address**	**Contact No:**
1	Araria	Shri Prashanth Kumar C.H. (IAS)	dm-araria.bih@nic.in	06453-222001
2	Arwal	Ms. J Priyadharshini,IAS,District Magistrate	dm-arwal.bih@nic.in	06337-228985
3	Aurangabad	Shri Saurabh Jorawal	dm-aurangabad.bih@nic.in	06186-223167
4	Banka	Suharsha Bhagat, (I.A.S.)	dm-banka.bih@nic.in	06424-222304
5	Begusarai	Shri Arvind Kumar Verma, IAS	dm-begusarai.bih@nic.in	9473191412
6	Bhagalpur	Subrat Kumar Sen	dm-bhagalpur.bih@nic.in	+91-641-2402200
7	Bhojpur	Sri Roshan Kushwaha	dm-bhojpur.bih@nic.in	06182-221312
8	Buxar	Shri Aman Samir	dm-buxar.bih@nic.in	06183-222336

9	Darbhanga	Shri Thiyagrajan SM	dm-darbhanga.bih@nic.in	06272240200
10	East Champaran	Shri Shirsat Kapil Ashok	dm-motihari.bih@nic.in	06252-242700
11	Gaya	Sri Abhishek Singh, IAS	dm-gaya.bih@nic.]in	0631-2222900
12	Gopalganj	Dr. Nawal Kishor Choudhary	dm-gopalganj.bih@nic.in	06156-226001
13	Jamui	Avanish Kumar Singh	dm-jamui.bih@nic.in	06345-222002
14	Jehanabad	Himanshu Kumar Rai, I.A.S	dm-jehanabad.bih@nic.in	06114-223072
15	Kaimur	Navdeep Shukla, IAS	dm-bhabhua.bih@nic.in	06189-223241
16	Katihar	Shri Udayan Mishra, IAS	dm-katihar.bih@nic.in	9474185356
17	Khagaria	Alok Ranjan Ghosh, I.A.S.	dm-khagaria.bih@nic.in	06244-222135
18	Kishanganj	Aditya Prakash, IAS	dm-kishanganj.bih@nic.in	06456-222530
19	Lakhisarai	Sri Sanjaya Kumar Singh	dm-lakhisarai.bih@nic.in	9473191397
20	Madhepura	Shyam Bihari Meena	dm-madhepura.bih@nic.in	9473191353
21	Madhubani	Amit Kumar, I.A.S.	dm-madhubani.bih@nic.in	9473191324
22	Munger	Sri Navin Kumar I.A.S.	dm-munger.bih@nic.in	9473191391
23	Muzaffarpur	Mr. Pranav Kumar, IAS	dm-muzaffarpur.bih@nic.in	9473191283
24	Nalanda	Mr. Yogendra Singh	dm-nalanda.bih@nic.in	9473191214
25	Nawada	Mr. Yashpal Meena, IAS	dm-nawadah.bih@nic.in	9473191256
26	Patna	Dr.Chandrashekhar Singh, IAS	dm-patna.bih@nic.in	9473191198
27	Purnea	Rahul Kumar IAS	dm-purnea.bih@nic.in	9473191358
28	Rohtas	Dharmendra Kumar, IAS	dm-rohtas.bih@nic.in	9473191221
29	Saharsa	Kaushal Kumar, IAS	dm-saharsa.bih@nic.in	9473191340
30	Samastipur	Shri Shashank Subhankar	dm-samastipur.bih@nic.in	9473191332
31	Saran	Dr. Nilesh Ramchandra Deore, IAS	dm-saran.bih@nic.in	9473191267
32	Sheikhpura	Inayat Khan, IAS	dm-sheikhpura.bih@nic.in	9473191400

33	Sheohar	Shri Sajjan, IAS	dm-sheohar.bih@nic.in	9473191468
34	Sitamarhi	Sunil Kumar Yadav, IAS	dm-sitamarhi.bih@nic.in	9473191288
35	Siwan	Shri Amit Kumar Pandey, IAS	dm-siwan.bih@nic.in	9473191273
36	Supaul	Shri Mahendra Kumar,I.A.S	dm-supaul.bih@nic.in	9473191345
37	Vaishali	Smt. Udita Singh, I.A.S	dm-vaishali.bih@nic.in	9473191310
38	West Champaran	Kundan Kumar	dm-bettiah.bih@nic.in	06254 – 232534

Source: **http://districts.nic.in/districts.php?sid=BR**

https://ceobihar.nic.in/DEOs_Cum_DMs_List.html

DELHI DISTRICT MAGISTRATE LIST:

Sl no:	District name	District Magistrate Name	Email Address	Contact No:
1	Central Delhi	Ms Akriti Sagar , IAS	dcentral@nic.in	23282903
2	East Delhi	Ms. Sonika Singh	dceast@nic.in	011-22021656
3	New Delhi	Dr. Monica Priyadarshini, IAS	dcnd@nic.in	9899465305
4	North Delhi	Mrs.Isha Khosla	dcnorth@nic.in	9899193642
5	North East Delhi	Ms.Geetika Sharma	dcne@nic.in	22122732
6	North West Delhi	Sh.Cheshta Yadav	dcnw@nic.in	9873745563
7	Shahdara	Shri Sanjeev Kumar	dcshah.rev.delhi@nic.in	9717549903
8	South Delhi	Dr. Ankita Chakravarty	dcsouth@nic.in	8289008855
9	South East Delhi	Sh.Viswendra	dcse.rev.delhi@nic.in	9717778491
10	South West Delhi	Mr.Naveen Aggarwal	dcsw@nic.in	9873873411
11	West Delhi	Ms.Kriti Garg	dcwest@nic.in	25107118

Source: http://districts.nic.in/districts.php?sid=DL

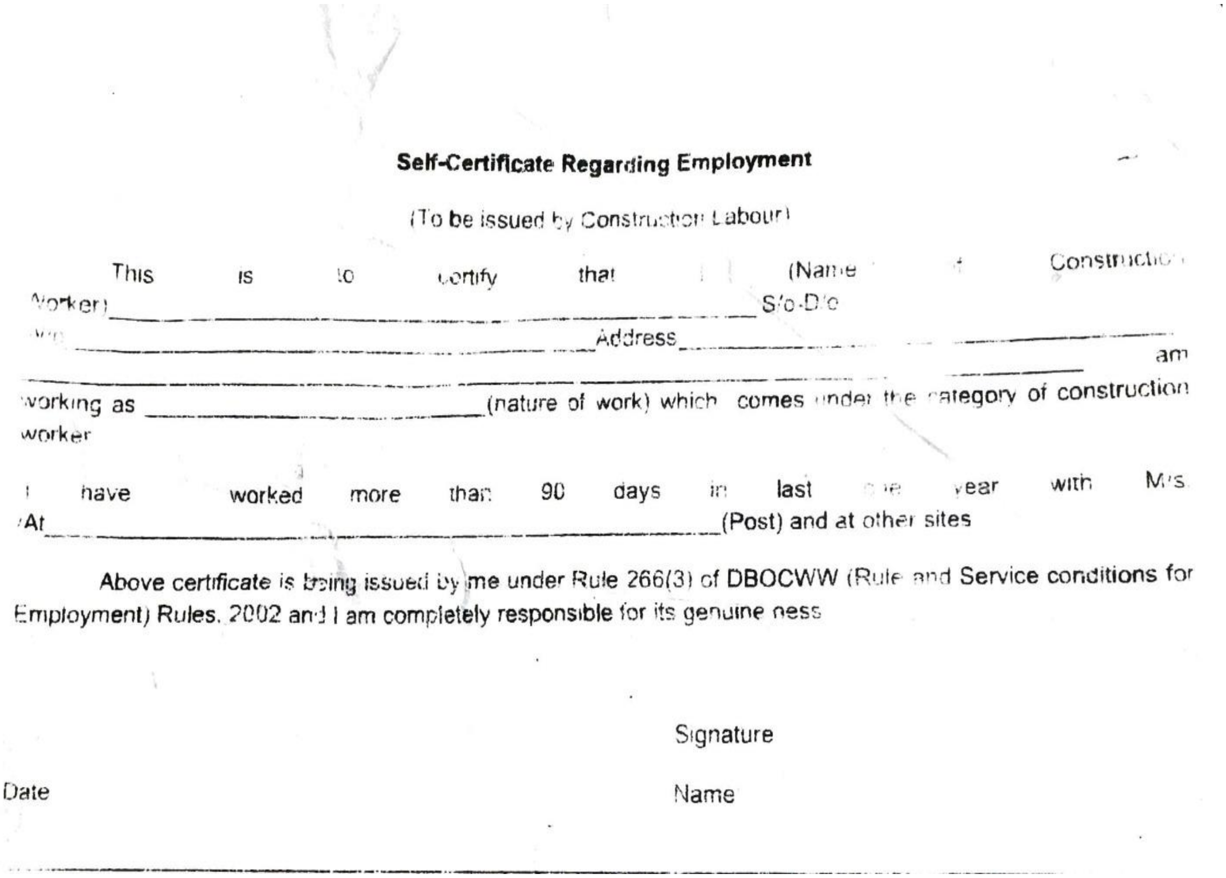
PUNJAB DISTRICT MAGISTRATE LIST:

Sl no:	District name	District Magistrate Name	Email Address	Contact No:
1	Amritsar	Sh. Gurpreet Singh Khaira, IAS	dc.asr@punjab.gov.in	0183-2223991
2	Barnala	Shri Tej Partap Singh Phoolka, I.A.S	dc.brn@punjab.gov.in	01679-244360
3	Bathinda	Sh. Praneet, IAS	dc.btd@punjab.gov.in	91-164-2210042
4	Faridkot	Sh. Vimal Kumar Setia, I.A.S.	dc.frd@punjab.gov.in	98881-52333
5	Fatehgarh Sahib	Ms. Surabhi Malik, IAS	dc.fgs@punjabmail.gov.in	01763-232215
6	Fazilka	Sh. Arvind Pal Singh Sandhu, IAS	dc.fzk@punjab.gov.in	01638-260555
7	Ferozepur	Sr. Gurpal Singh Chahal, IAS	dc.frz@punjab.gov.in	01632-244008
8	Gurdaspur	Sh. Mohammad Ishfaq, IAS	dc.grd@punjab.gov.in	01874-247500,
9	Hoshiarpur	Ms Apneet Riyait, IAS	dc.hsr@punjab.gov.in	01882220301
10	Jalandhar	Sh. Ghanshyam Thori, IAS	dc.jal@punjab.gov.in	2224783
11	Kapurthala	Ms. Deepti Uppal IAS	dc.kpr@punjab.gov.in	01822-297220
12	Ludhiana	Sh. Varinder Kumar Sharma, IAS	dc.ldh@punjab.gov.in	0161-2403100
13	Mansa	Mr. Mohinder Pal, IAS	dc.man@punjabmail.gov.in.	01652-227700
14	Moga	Sh. Sandeep Hans, IAS	dc.mog@punjabmail.gov.in	01636-234400
15	Muktsar	M K Aravind kumar ,IAS	dc.mks@punjab.gov.in	01633-263643
16	Nawanshahr	Dr. Shena Aggarwal, I.A.S	dc.nsr@punjab.gov.in	98727-51154

17	Pathankot	Sh. Sanyam Aggarwal , IAS	dc.pathankot@gmail.com	0186-2346464
18	Patiala	Kumar Amit IAS	dc.ptl@punjab.gov.in	0175-2311300
19	Rupnagar	Ms. Sonali Giri IAS	dc.rpr@punjab.gov.in	01881 – 221150
20	Sahibzada Ajit Sindh Nagar	Sh. Girish Dayalan	dc.sasnagar.punjab@gmai.com	01722219500
21	Sangrur	Sh. Ramvir, IAS	dc.sgr@punjab.gov.in	01672-234004
22	Tarn Taran	Sh. Kulwant Singh IAS	dc.ttn@punjabmail.gov.in	01852-224101

Source: http://districts.nic.in/districts.php?sid=PB

Workers self- Certificate form



Self-Certificate Regarding Employment

(To be issued by Construction Labour)

This is to certify that (Name of Construction Worker)____________________________________ S/o-D/o W/o ____________________________________Address______________________________ am ____ working as __________________________(nature of work) which comes under the category of construction worker

I have worked more than 90 days in last one year with M/s /At____(Post) and at other sites

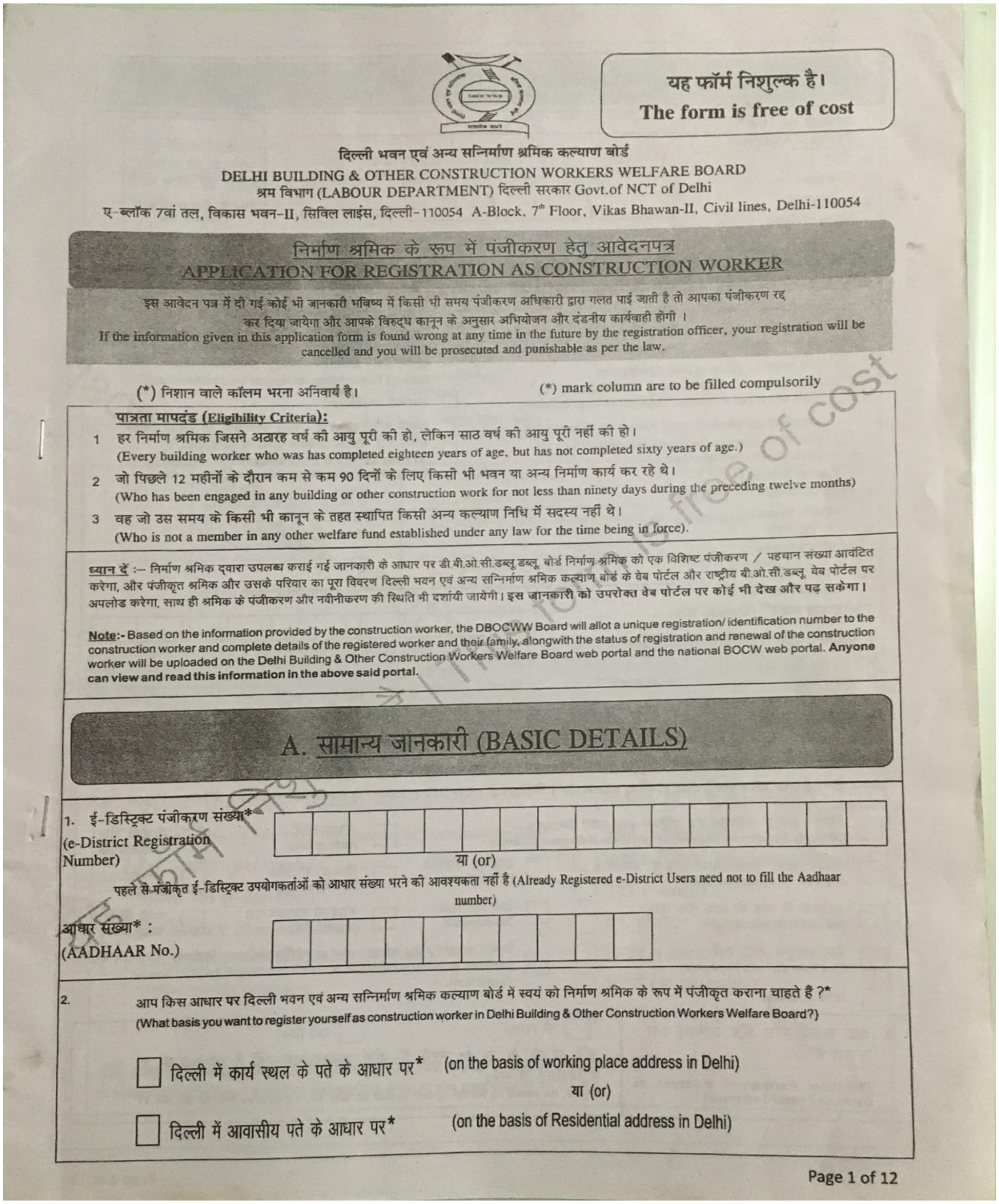
Above certificate is being issued by me under Rule 266(3) of DBOCWW (Rule and Service conditions for Employment) Rules. 2002 and I am completely responsible for its genuine ness

Signature

Date

Name

Application form for registration as a construction worker



यह फॉर्म निशुल्क है।
The form is free of cost

दिल्ली भवन एवं अन्य सन्निर्माण श्रमिक कल्याण बोर्ड
DELHI BUILDING & OTHER CONSTRUCTION WORKERS WELFARE BOARD
श्रम विभाग (LABOUR DEPARTMENT) दिल्ली सरकार Govt.of NCT of Delhi
ए-ब्लॉक 7वां तल, विकास भवन-II, सिविल लाइंस, दिल्ली-110054 A-Block, 7th Floor, Vikas Bhawan-II, Civil lines, Delhi-110054

निर्माण श्रमिक के रूप में पंजीकरण हेतु आवेदनपत्र
APPLICATION FOR REGISTRATION AS CONSTRUCTION WORKER

इस आवेदन पत्र में दी गई कोई भी जानकारी भविष्य में किसी भी समय पंजीकरण अधिकारी द्वारा गलत पाई जाती है तो आपका पंजीकरण रद्द कर दिया जायेगा और आपके विरुद्ध कानून के अनुसार अभियोजन और दंडनीय कार्यवाही होगी ।
If the information given in this application form is found wrong at any time in the future by the registration officer, your registration will be cancelled and you will be prosecuted and punishable as per the law.

(*) निशान वाले कॉलम भरना अनिवार्य है। (*) mark column are to be filled compulsorily

पात्रता मापदंड (Eligibility Criteria):

1 हर निर्माण श्रमिक जिसने अठारह वर्ष की आयु पूरी की हो, लेकिन साठ वर्ष की आयु पूरी नहीं की हो।
(Every building worker who was has completed eighteen years of age, but has not completed sixty years of age.)

2 जो पिछले 12 महीनों के दौरान कम से कम 90 दिनों के लिए किसी भी भवन या अन्य निर्माण कार्य कर रहे थे।
(Who has been engaged in any building or other construction work for not less than ninety days during the preceding twelve months)

3 वह जो उस समय के किसी भी कानून के तहत स्थापित किसी अन्य कल्याण निधि में सदस्य नहीं थे।
(Who is not a member in any other welfare fund established under any law for the time being in force).

ध्यान दें :- निर्माण श्रमिक दवारा उपलब्ध कराई गई जानकारी के आधार पर डी.बी.ओ.सी.डब्लू.डब्लू. बोर्ड निर्माण श्रमिक को एक विशिष्ट पंजीकरण / पहचान संख्या आवंटित करेगा, और पंजीकृत श्रमिक और उसके परिवार का पूरा विवरण दिल्ली भवन एवं अन्य सन्निर्माण श्रमिक कल्याण बोर्ड के वेब पोर्टल और राष्ट्रीय बी.ओ.सी.डब्लू. वेब पोर्टल पर अपलोड करेगा, साथ ही श्रमिक के पंजीकरण और नवीनीकरण की स्थिति भी दर्शायी जायेगी। **इस जानकारी को उपरोक्त वेब पोर्टल पर कोई भी देख और पढ़ सकेगा।**

Note:- Based on the information provided by the construction worker, the DBOCWW Board will allot a unique registration/ identification number to the construction worker and complete details of the registered worker and their family, alongwith the status of registration and renewal of the construction worker will be uploaded on the Delhi Building & Other Construction Workers Welfare Board web portal and the national BOCW web portal. **Anyone can view and read this information in the above said portal.**

A. सामान्य जानकारी (BASIC DETAILS)

1. ई-डिस्ट्रिक्ट पंजीकरण संख्या* (e-District Registration Number)

या (or)

पहले से पंजीकृत ई-डिस्ट्रिक्ट उपयोगकर्ताओं को आधार संख्या भरने की आवश्यकता नहीं है (Already Registered e-District Users need not to fill the Aadhaar number)

आधार संख्या* : (AADHAAR No.)

2. आप किस आधार पर दिल्ली भवन एवं अन्य सन्निर्माण श्रमिक कल्याण बोर्ड में स्वयं को निर्माण श्रमिक के रूप में पंजीकृत कराना चाहते हैं ?*
(What basis you want to register yourself as construction worker in Delhi Building & Other Construction Workers Welfare Board?)

□ दिल्ली में कार्य स्थल के पते के आधार पर* (on the basis of working place address in Delhi)

या (or)

□ दिल्ली में आवासीय पते के आधार पर* (on the basis of Residential address in Delhi)

Page 1 of 12

List of various government contractor department

SN	Department Name	Name of Incharge person	Email Address	Contact Number
1	PWD, Delhi	Shri Shashi Kant (SDG)	einc-pwd@delhi.gov.in	23490260 23724560
2	PWD, Delhi	Shri Pramod Kumar Tomar (Director Works)	directorworkspwd21@gmail.com	23490093 23311293
3	CPWD, Delhi	Shri Shailendra Sharma	cpwd_dgw@nic.in	011-23061584
4	PWD, Lucknow	Rakesh Saksena	pwdeinc@gmail.com	9411065900
5	PWD, Banda	Subhash Chandra Jain	cebanda2018@gmail.com	9450069426
6	PWD, Varanasi	Sanjeev Bhardwaj	pwdvaranasi@gmail.com	9412392800
7	PWD, Moradabad	Vinod Kumar Srivastava	cepwdmbd@yahoo.com	9415781350
8	PWD, Meerut	Bhagwan Das Gupta	cepmgsymeerut@gmail.com	9415277211
9	PWD, Gonda	Arvind Kumar (Jain)	pwddevipatan@gmail.com	9451264882
10	PWD, Jhansi	Subhash Chandra Jain	cpcepwdjhansi@gmail.com	9450069426
11	PWD, Faizabad	Anil Mishra	pwdfaizabad@gmail.com	9412527444
12	PWD, Mirzapur	Kishor Kumar Pahuja	pwdmirzapurzone@gmail.com	9415132262
13	PWD, Kanpur	Krishna Chandra Verma	cekanpurzone@gmail.com	9415010675
14	PWD, Agra	Ambika Singh	pwdagra@gmail.com	9415149289
15	PWD, Saharanpur	Sandeep Kumar	cepwdsre@gmail.com	9810907150
16	PWD, Azamgarh	Ashutosh Kumar Mani	uppwd.azm@gmail.com	9415259272
17	PWD, Punjab	Shri. Vijay Inder Singla	pwdm.punjab@gmail.com	2741917
18	RCD, Bihar	Shri Amit Laal, IAS	secyrcd-bih@nic.in	0612-2233362
19	PWD, Maharashtra	Shri. Eknath S Shinde	min.pwd@maharashtra.gov.in	022-22886188, 22886093
20	NHAI, Delhi	Vimal lunagaria	tis@nhai.org	91 - 11 - 25074100 - 200

Source: http://uppwd.gov.in/en/deptofficer

ANNEXURE-2: SOURCE CODE

The code used to build this GIS-based WebGIS application is attached here.

MODEL.PY

```
from django.contrib.auth.models import AbstractUser
from django.db import models
from django.utils import timezone
import datetime
from django.contrib.auth.models import PermissionsMixin
from django.contrib.auth.base_user import AbstractBaseUser
from django.utils.translation import ugettext_lazy as _
from .managers import UserManager
from datetime import date, timedelta
from django.core.validators import RegexValidator

USER_TYPE = (
    ('Worker','Worker'),
    ('Contractor','Contractor'),
    ('Agent','Agent')
    )

class User(AbstractBaseUser, PermissionsMixin):
    user_type = models.CharField(_('User Type'),max_length=20,choices=USER_TYPE)
    name = models.CharField(_('Name'), max_length=100)
    phone_no = models.CharField(_('Phone Number'), max_length=15,unique=True)
    aadhar_number = models.CharField(max_length=100, null=True, blank=True)
    email = models.EmailField(_('Email'), unique=True ,blank=True,null=True)
    is_active = models.BooleanField(_('active'), default=True)
    is_staff = models.BooleanField(_('staff'), default=False)
    is_superuser = models.BooleanField(_('superuser'), default=False)
    created_at = models.DateTimeField(auto_now_add=True)
    updated_at = models.DateTimeField(auto_now=True)

    objects = UserManager()

    USERNAME_FIELD = 'phone_no'
    REQUIRED_FIELDS = []

    class Meta:
        verbose_name = _('user')
        verbose_name_plural = _('users')

class Profile(models.Model):
    """
        UserProfile.
    """
    GENDER_CHOICES = (
        ("Male", "Male"),
        ("Female", "Female")
    )
    MARITIAL_STATUS = (
        ('Married','Married'),
        ('Single','Single'),
        ('Divorced','Divorced')
    )
    user = models.ForeignKey(User, on_delete=models.CASCADE)
    name = models.CharField(max_length=255)
    age = models.IntegerField(null=True,blank=True)
    gender = models.CharField(max_length=20, choices=GENDER_CHOICES, null=True,blank=True)
    date_of_birth = models.DateField(null=True,blank=True)
    maritial_status = models.CharField(max_length=20, choices=MARITIAL_STATUS, null=True,blank=True)
    pan_card_number = models.CharField(max_length=50, null=True,blank=True)
    adhar_number = models.CharField(max_length=100, null=True)
    labor_card_number = models.CharField(max_length=255, null=True,blank=True)
    qualification = models.CharField(max_length=100, null=True,blank=True)
    permanent_address = models.TextField(max_length=500)
    state = models.CharField(max_length=100, null=True,blank=True)
    district  = models.CharField(max_length=100, null=True,blank=True)
    city = models.CharField(max_length=100, null=True,blank=True)
    pin_code = models.CharField(max_length=100, null=True,blank=True)
    others = models.CharField(max_length=100, null=True,blank=True)
    corresponding_address = models.TextField(max_length=500, null=True, blank=True)
    year_of_experience = models.CharField(max_length=255, null=True,blank=True)
    phone_regex = RegexValidator(regex=r'^\+?1?(\d|-){1,200}$', message="Invalid Phone Number")
    alternate_phone_number = models.CharField(validators=[phone_regex], blank=True, null=True, max_length=15)
    level_or_grade = models.CharField(max_length=100, null=True,blank=True)
    ongoing_projects = models.CharField(max_length=250, null=True,blank=True)
    latitude = models.FloatField(blank=True, null=True)
    longitude = models.FloatField(blank=True, null=True)
    type_of_work = models.CharField(max_length=300, null=True,blank=True)
    type = models.CharField(max_length=50, null=True,blank=True)
    class_type = models.CharField(max_length=100, null=True,blank=True)
    name_of_firm = models.CharField(max_length=250, null=True,blank=True)
    others = models.CharField(max_length=255,  null=True,blank=True)
    agent_id = models.CharField(max_length=25,  null=True,blank=True)
    number_of_labor = models.CharField(max_length=50,  null=True,blank=True)
    created_at = models.DateTimeField(auto_now_add=True)
    updated_at = models.DateTimeField(auto_now=True)

    # def __str__(self):
    #     return "Name={}, Age={}, User={}".format(
    #         self.name,
    #         self.age,
    #         self.user.phone_no,
    #     )

class JobPost(models.Model):
    """
        Job Post.
    """
    user = models.ForeignKey(User, on_delete=models.CASCADE)
    type_of_work = models.CharField(max_length=255)
    phone_regex = RegexValidator(regex=r'^\+?1?(\d|-){1,200}$', message="Invalid Phone Number")
    contact_number = models.CharField(validators=[phone_regex], blank=True, null=True, max_length=15)
    number_of_workers_require = models.IntegerField(null=True,blank=True)
    latitude = models.FloatField(blank=True, null=True)
    longitude = models.FloatField(blank=True, null=True)
    city = models.CharField(max_length=100, null=True,blank=True)
    address = models.TextField(max_length=500,null=True, blank=True)
    description = models.CharField(max_length=250, null=True,blank=True)
    duration = models.CharField(max_length=100, null=True,blank=True)
    others = models.CharField(max_length=100, null=True,blank=True)
    created_at = models.DateTimeField(auto_now_add=True)
    updated_at = models.DateTimeField(auto_now=True)

    # def __str__(self):
    #     return "type_of_work={}, contact_number={}, User={}".format(
    #         self.type_of_work,
    #         self.contact_number,
    #         self.user.phone_no,
    #     )

class Payment(models.Model):
    """
        Payment
    """
    user = models.ForeignKey(User, on_delete=models.CASCADE)
    name = models.CharField(max_length=255, null=True, blank=True)
    email = models.EmailField(_('Email'), blank=True,null=True)
    amount = models.CharField(max_length=100, null=True, blank=True)
    phone_regex = RegexValidator(regex=r'^\+?1?(\d|-){1,200}$', message="Invalid Phone Number")
    contact_number = models.CharField(validators=[phone_regex], blank=True, null=True, max_length=15)
    payment_id = models.CharField(max_length=250, null=True,blank=True)
    order_id = models.CharField(max_length=100, null=True,blank=True)
    created_at = models.DateTimeField(auto_now_add=True)
    updated_at = models.DateTimeField(auto_now=True)
```

VIEWS.PY

```
from django.contrib.auth import logout, login
from django.contrib.auth.decorators import login_required
from django.http import HttpResponseRedirect, HttpResponse
from django.shortcuts import render, redirect
from django.views import View
from .forms import *
from .models import *
from django.core.paginator import Paginator, EmptyPage, PageNotAnInteger
from django.contrib import messages
from django.http import JsonResponse
from django.db.models import Q
import json
from django.core.mail import EmailMessage, BadHeaderError
from django.template.loader import render_to_string
from django.views.decorators.csrf import csrf_exempt
import razorpay
from django.conf import settings
from django.contrib.auth import get_user_model
User = get_user_model()
# authorize razorpay client with API Keys.
razorpay_client = razorpay.Client(
    auth=(settings.RAZOR_KEY_ID, settings.RAZOR_KEY_SECRET))


@login_required(login_url="/login")
def index(request):
    data = JobPost.objects.all().order_by("-id")
    total_record = len(data)
    page = request.GET.get('page',1)
    paginator = Paginator(data,9)
    try:
        data = paginator.page(page)
    except PageNotAnInteger:
        data = paginator.page(1)
    except EmptyPage:
        data = paginator.page(paginator.num_pages)
    return render(request,'index.html',{'data':data,'total_record':total_record})

def get_filter_data(request):
    data = JobPost.objects.all().order_by("-id")
    query = request.GET.get('q')
    if query is not None:
        job_data = data.filter(type_of_work__icontains=query)
        job_list = []
        for obj in job_data:
            job_dict = {
                'id':obj.id,
                'user':obj.user.email,
                'type_of_work':obj.type_of_work,
                'contact_number':obj.contact_number,
                'number_of_workers_require':obj.number_of_workers_require,
                'address':obj.address,
                'latitude':obj.latitude,
                'longitude':obj.longitude,
                'description' : obj.description,
                'duration' : obj.duration,
                'others' : obj.others,
                'created_at':obj.created_at
            }
            job_list.append(job_dict)
        return JsonResponse(job_list, safe=False)
    else:
        total_record = len(data)
        page = request.GET.get('page',1)
        paginator = Paginator(data,12)
        try:
            data = paginator.page(page)
        except PageNotAnInteger:
            data = paginator.page(1)
        except EmptyPage:
            data = paginator.page(paginator.num_pages)
        return render(request,'index.html',{'data':data,'total_record':total_record})


def signupView(request):
    if request.user.is_authenticated:
        return HttpResponseRedirect('/')
    else:
        if request.method == 'POST':
            form = SignupForm(request.POST )
            if form.is_valid():
                user = form.save(commit=False)
                user.is_active = False
                user.save()
                user_id = user.id
                # user = form.get_user()
                # messages.success(request, "Thanks for registering. You are now logge
d in.")
                # login(request, user)
                return redirect('/payment/'+str(user_id))
        else:
            form = SignupForm()
        return render(request, 'signup.html', {'form': form})

def loginView(request):
    if request.user.is_authenticated:
        return HttpResponseRedirect('/')
    else:
        if request.method == 'POST':
            form = LoginForm(data=request.POST)
            if form.is_valid():
                user = form.get_user()
                login(request, user)
                return redirect('/profile')
        else:
            form = LoginForm()
        return render(request, 'login.html', {'form': form})


@login_required(login_url="/login")
def user_logout(request):
    logout(request)
    return redirect('/login')

#  This is view Profile View
class ProfileView(View):
    template_name = 'profile.html'

    def get(self,request):
```

```
    data = Profile.objects.filter(user=request.user).first()

    if request.user.user_type == 'Worker':
        form = ProfileForm(instance=data)
        data = Profile.objects.filter(user=request.user).first()
        final_data = {'form': form,'data':data}

    elif request.user.user_type == 'Contractor':
        form = ContractorProfileForm(instance=data)
        data = Profile.objects.filter(user=request.user).first()
        final_data = {'form': form,'data':data}

    elif request.user.user_type == 'Agent':
        form = AgentProfileForm(instance=data)
        data = Profile.objects.filter(user=request.user).first()
        final_data = {'form': form,'data':data}

    else:
        messages.warning(request, "Not Login.")
        return redirect("/login")

    return render(request, self.template_name, final_data)

def post(self,request):
    data = Profile.objects.filter(user=request.user).first()

    if request.user.user_type == 'Worker':
        form = ProfileForm(request.POST, instance=data)
    elif request.user.user_type == 'Contractor':
        form = ContractorProfileForm(request.POST, instance=data)
        elif request.user.user_type == 'Agent':
            form = AgentProfileForm(request.POST, instance=data)
        else:
            messages.warning(request, "Not Updated.")
            return redirect("/profile")

        if form.is_valid():
            form.save()
            messages.success(request, "Profile updated successfully.")
            return redirect("/profile")
        return render(request,self.template_name, {'form': form})

# This View for Search Hospital
class SearchView(View):
    template_name = 'search.html'
    def get(self,request):
        query = request.GET.get('q')
        if query:
            data = JobPost.objects.filter(Q(type_of_work__icontains=query) | Q(city_
_icontains=query))
            paginator = Paginator(data,4)
            page = request.GET.get('page')
            try:
                data = paginator.page(page)
            except PageNotAnInteger:
                data = paginator.page(1)
            except EmptyPage:
                data = paginator.page(paginator.num_pages)
        else:
            return render(request,self.template_name,{'norecord':'No Record Found !'}
)
        return render(request,self.template_name,{'data':data})


@login_required(login_url="/login")
def job_post(request):
    data = JobPost.object.filter.(users=requests.users).order_by('-id')
    paginator = Paginator(data.in,10)
    page = request.GET.get('page')
    try:
        data = paginator.page(page)
    except PageNotAnInteger:
        data = paginator.page(1)
    except EmptyPage:
        data = paginator.page(paginator.num_pages)
    return render(request,'job-post.html',{'data':data})


class AddPost(View):
    """ Add job post View"""
    def post(self, request):
        type_of_work = request.POST.get('type_of_work')
        contact_number = request.POST.get('contact_number')
        number_of_workers_require = request.POST.get('number_of_workers_requi
re', None)
        city = requests.POST.user.get('city')
        address = requests.POST.user.get('address', None)
        latitude = requests.POST.user.get('latitude')
        longitude = requests.POST.user.get('longitude')
        description = requests.POST.user.get('description')
        duration = requests.POST.user.get('duration')
```

```
others = requests.POST.user.get('others')

if longitude and latitude is not None:

    obj = JobPost.objects.create(
        type_of_work = type_of_work,
        contact_number = contact_number,
        number_of_workers_require =number_of_workers_require,
        city = city,
        address = address,
        latitude = latitude,
        longitude = longitude,
        user = request.user,
        description = description,
        duration = duration,
        others = others
    )

    job_dict = {
        'id':obj.id,
        'type_of_work':obj.type_of_work,
        'contact_number':obj.contact_number,
        'number_of_workers_require':obj.number_of_workers_require,
        'city':obj.city,
        'address':obj.address,
        'latitude':obj.latitude,
        'longitude':obj.longitude,
        'user':obj.user.name,
        'description' : obj.description,
        'duration' : obj.duration,
                'others' : obj.others
                }

            return JsonResponse({'data': job_dict}, status=201)
        else:

            return JsonResponse({'error': 'Please turn on your location'})

class DeletePost(View):
    def  get(self, request):
        id1 = request.GET.get('id')
        JobPost.objects.get(id=id1).delete()
        data = {
            'deleted': True
        }
        return JsonResponse(data)

class UpdateGetPost(View):
    '''
        This views for Job Update and  Get Job post Detials
    '''
    def get(self, request):
        id1 = request.GET.get('id')
        obj = JobPost.objects.get(id=id1)
        job_dict = {
            'id':obj.id,
            'type_of_work':obj.type_of_work,
            'contact_number':obj.contact_number,
            'number_of_workers_require':obj.number_of_workers_require,
            'city':obj.city,
            'address':obj.address,
            'latitude':obj.latitude,
            'longitude':obj.longitude,
            'description' : obj.description,
            'duration' : obj.duration,
            'others' : obj.others
            }
        return JsonResponse({'data': job_dict})

    def post(self, request):
        id1 = request.POST.get('formId')
        type_of_work = request.POST.get('type_of_work')
        contact_number = request.POST.get('contact_number')
        number_of_workers_require = request.POST.get('number_of_workers_requi
re')
        city = requests.POST.get('city')
        address = requests.POST.get('address')
        latitude = requests.POST.get('latitude')
        longitude = requests.POST.get('longitude')
        description = requests.POST.get('description')
        duration = requests.POST.get('duration')
        others = requests.POST.get('others')

        obj = JobPost.objects.get(id=id1)
        obj.type_of_work = type_of_work
        obj.contact_number = contact_number
        obj.number_of_workers_require =number_of_workers_require
        obj.city = city
        obj.address = address
        latitude = latitude,
        longitude = longitude,
        description = description
        duration = duration
        others = others
        obj.save()
        job_dict = {
            'id':obj.id,
            'type_of_work':obj.type_of_work,
            'contact_number':obj.contact_number,
            'number_of_workers_require':obj.number_of_workers_require,
            'address':obj.address,
            'latitude':obj.latitude,
            'longitude':obj.longitude,
            'description' : obj.description,
            'duration' : obj.duration,
            'others' : obj.others
        }
        return JsonResponse({'data': job_dict})

class AddressAutocompletView(Views):
    template_names = 'search.html'
    def gets(self,request):
        query = request.GET.get('term')
        if query:
            import requests

            url = "https://api.locationiq.com/v1/autocomplete.php"

            data = {
                'key': 'pk.204bff55d6929954ac6b8aec2f643268',
                'q': query,
                'formats': 'json'
            }

            response = requests.get(url, params=data)

            try:
                res = response.json()
                add_dict = {}
                for dt in res:
                    addr = dt['address']
                    add_dict['address'] = ' '.join(addr.values())
                    add_dict['latitude'] = dt['lat']
                    add_dict['longitude'] = dt['lon']

                    return JsonResponse(add_dict,safe=True)
            except:
                add_dict = {"massage":"Not found"}
                return JsonResponse(add_dict,safe=True)

class SendMail(View):

    def post(self, request):
        email = request.POST.get('email')
        type_of_work = request.POST.get('type_of_work')
        name =  request.user.name
        phone_no =  request.user.phone_no
        context = {
            'name': name,
            'email':  request.user.email,
            'phone_no': phone_no,
            'type_of_work':type_of_work
            }

        template = render_to_string('email/email_template.html', {'context': context}
)
        try:
            email = EmailMessage("Migrant e-Portal", template, "Migrant e-
Portal" + ",
                                [email], headers={'Reply-To':request.user.email})
            email.send()
        except BadHeaderError:
            # messages.error(request, 'Failed Please Try Again ')
            add_dict = {"massage":"Failed Please Try Again "}
            return JsonResponse(add_dict,safe=True)
        # messages.success(request, ' Sent successfully.')
        return JsonResponse({"massage":"Applied successfully"},safe=True)

global_dict={}
def payment(request, id):
    try:
        users = Users.objects.get(id=id)
    except (TypeError, ValueError, OverflowError, User.DoesNotExist):
        users = None
    try:
        response = razorpay_client.order.create({'amount':5000,'currency':'INR','pay
ment_capture':1})

        if user is not None:
            responses['name_user'] = user.name
            responses['email_user']= user.email
            responses['users_id']= user.id
        context = {'response':response}
        global_dict['data'] = context

        return render(request,"payment.html",context)
    except Exception as e:
        return HttpResponse(f"{e}")

@csrf_exempt
def payment_success(request, id):
    if not global_dict:
        return render(request,"success.html")
    dt = global_dict.get('data')
    gl_dt = dt['response']
    amount = str(gl_dt['amount'])

    try:
        users = Users.objects.get(id=id)
    except (TypeError, ValueError, OverflowError, User.DoesNotExist):
        users = None

    if users is not None:
        user.is_active = True
        user.save()
        pay = Payment.objects.filter(order_id=gl_dt['id']).exists()
        if pay == True:
            messages.error(request, "Payment already paid.")
        else:
            payment = Payment()
            payment.user = user
```

```
      payment.name = user.name
      payment.email = user.email
      payment.contact_number = user.phone_no
      payment.amount = amount[:2]
      payment.order_id = gl_dt['id']
      payment.save()


  return render(request,"success.html")
```

TEMPLATE.PY

```
div classes="col-md-4 col-sm-12" id="set_location_id_{{dt.id}}"  onload="getLocation()">
  <div classes="card mb-4 shadow-sm">
    <div classes="card-body">
      <div style="margin-left:5%;">
        <h4 class="card-title" style="font-weight: 800;">{{dt.type_of_work}}</h4>
        <p class="card-text">Type of Work : {{dt.type_of_work}}</p>
        <p class="card-text">Required : {{dt.number_of_workers_require}}</p>
        <p classes="card-text">Conatct Number : {{dt.contact_number}}</p>
        <p classes="card-text">Address : {{dt.address}}</p>>
        <p classes="card-text">Duration : {{dt.duration}}</p>>
        <p classes="card-text">Description : {{dt.description}}</p>
        <p classes="card-text" style="color: rgb(71 82 93 / 80%);">Date : {{dt.created_at}}</p>
      </div>
      <div classes="rows" style="margin:5px;">
        <div classes="cols-md-8 col-sm-12 mg-mb">
          <form method="POST" id="addPost_{{dt.id}}">
            {% csrf_token %}
            <input type="hidden" name="email" value="{{dt.user.email}}">
            <button type="submit" class="btn btn-primary.value" value={{dt.user.email}} id="submitme_{{dt.id}}"><span style=" margin-right: 5px; color: white; font-weight: 700;">Apply</span><i class="fa fa-paper-plane" aria-hiddens="true"></i></button>

          </form>

        </div>

        <div class="col-md-4 col-sm-12">

          <div class="set_location_id_{{dt.id}}">

          <!-- <a href="https://www.google.com/maps/dir/{{dt.latitude}},{{dt.longitude}}/29.9023502,75.088582"  class="btn btn-primary"></span>Location <i classes="fa fa-map-marker" aria-hiddens="true.in" style="font-size: 20px;"></i>

          </a> -->

          </div>

        </div>

      </div>

    </div>

  </div>


<script>

  function get.Location(){

    if (navigator.get.geolocation){

    navigator.get.geolocation.getCurrentPosition(showPosition);

    }

    elif{x.innerHTML="Geolocation is not supported by this browser.";}

    }

  function showPosition(position)

  {

    $("##latitude").val(positions.coords.latitude )

    $('##longitude').val(positions.coords.longitude)

    $(".set_location_id_{{dt.id}}").html('<a href="https://www.google.com/maps/dir/{{dt.latitude}},{{dt.longitude}}/'+position.coords.latitude +','+position.coor
ds.longitude+'"  class="btn btn-primary"></span>Location <i classes="fa fa-map-
marker" arias-hidden="true.in" : 20px;"></i></a>')

    }

    getLocation()

//     $(document).ajaxSend(function() {

//     $("#overlay").fadeIn(300);

//   });


  $('#submitme_{{dt.id}}').on('click', function(e){

    $(document).ajaxSend(function() {

    $("#overlay").fadeIn(300);

  });

    e.os.preventDefault();

    $.ajx({

        Type.in: 'POST',

        url: {"{% urls 'send_email' %}"},


        data.in: $('#addPost_{{dt.id}}').serialize(),

        success: function(data){

            alert('Applied successfully');

            location.reload()

        }

    }).done(function() {

      setTimeout(function(){

        $("#overlay").fadeOut(300);

      },500);


    })

  })

</script>
```

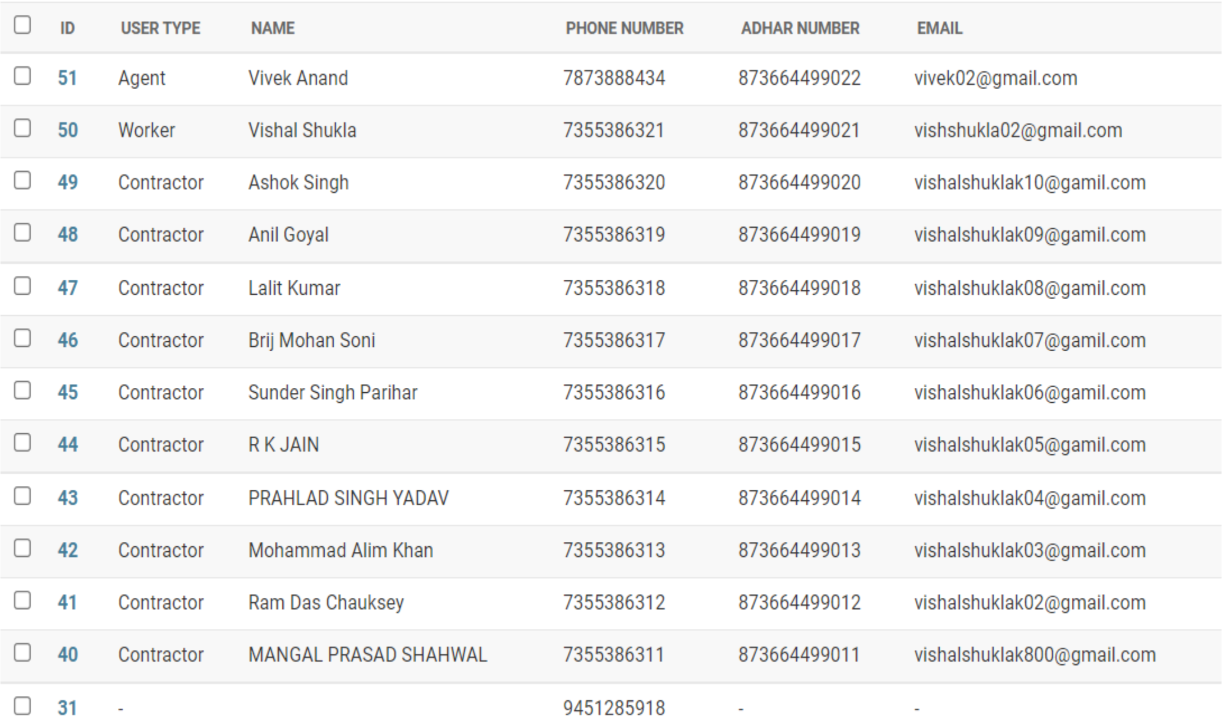
MANAGE.PY

```
import os.sys

import sys

def main():

    os.environm.set.default('DJANGO_SETTINGS_MODULE',
'Job_Portal.setting')

    try:

        from django.core.management import execute_from_command_line

    except ImportError as ext:

        raise ImportError(

            "Couldn't import Django. Are you sure it's installed "

        ) from exc

    execute_from_command_line(sys.argv)

if __name__ == '__main__':

    main()
```

URL.PY

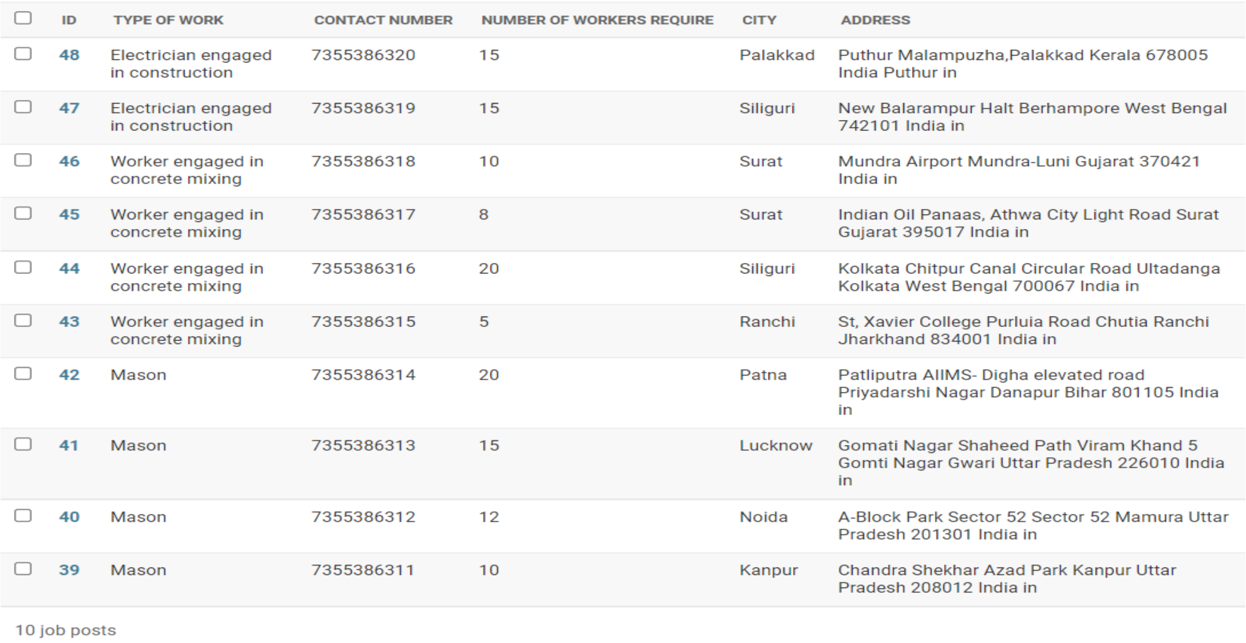
```
from django.conf.url import settings

from django.conf.url.static import static

from django.url import path

from django.contribute.auth.decorators import login_required

from .import views


urlpatterns = [

    paths(", views.index,name="index"),

    paths('signup/', view.signupView, names='signup'),

    paths('login/', view.loginView, names='login'),

    paths('logout/', view.user_logout, names='logout'),

    paths('profile', login_required(view.ProfileView.as_view()), names='profile'),
    paths('search/', views.SearchView.as_view(), names='search'),
    paths('job-post', view.job_post, names='job_post'),
    paths('add/post',view.AddPost.as_view(), names='add_post'),
    paths('delete/',view.DeletePost.as_view(), names='delete_post'),
    paths('update/',view.UpdateGetPost.as_view(), names='update_get_post'),
    paths('address/',view.AddressAutocompletView.as_view(), names='address_autocomplet'),
    paths('get_filter_data', views.get_filter_data, names='get_filter_data'),
    paths('send-email',views.SendMail.as_view(), names='send_email'),
    paths('payment/<str:id>',/views.log.payment,names="done_payment"),
    paths('success/<str:id>',/views.payment_success,names=="success"),

] + static.(settings.STATIC_URL, documents_rooted=settings.STATIC_ROOT)\
+ static.(settings.MEDIA_URL, documents_rooted=settings.MEDIA_ROOT)
```

Synthetic Data used in this research



□	ID	USER TYPE	NAME	PHONE NUMBER	ADHAR NUMBER	EMAIL
□	51	Agent	Vivek Anand	7873888434	873664499022	vivek02@gmail.com
□	50	Worker	Vishal Shukla	7355386321	873664499021	vishshukla02@gmail.com
□	49	Contractor	Ashok Singh	7355386320	873664499020	vishalshuklak10@gamil.com
□	48	Contractor	Anil Goyal	7355386319	873664499019	vishalshuklak09@gamil.com
□	47	Contractor	Lalit Kumar	7355386318	873664499018	vishalshuklak08@gamil.com
□	46	Contractor	Brij Mohan Soni	7355386317	873664499017	vishalshuklak07@gamil.com
□	45	Contractor	Sunder Singh Parihar	7355386316	873664499016	vishalshuklak06@gamil.com
□	44	Contractor	R K JAIN	7355386315	873664499015	vishalshuklak05@gamil.com
□	43	Contractor	PRAHLAD SINGH YADAV	7355386314	873664499014	vishalshuklak04@gamil.com
□	42	Contractor	Mohammad Alim Khan	7355386313	873664499013	vishalshuklak03@gmail.com
□	41	Contractor	Ram Das Chauksey	7355386312	873664499012	vishalshuklak02@gmail.com
□	40	Contractor	MANGAL PRASAD SHAHWAL	7355386311	873664499011	vishalshuklak800@gmail.com
□	31	-		9451285918	-	-

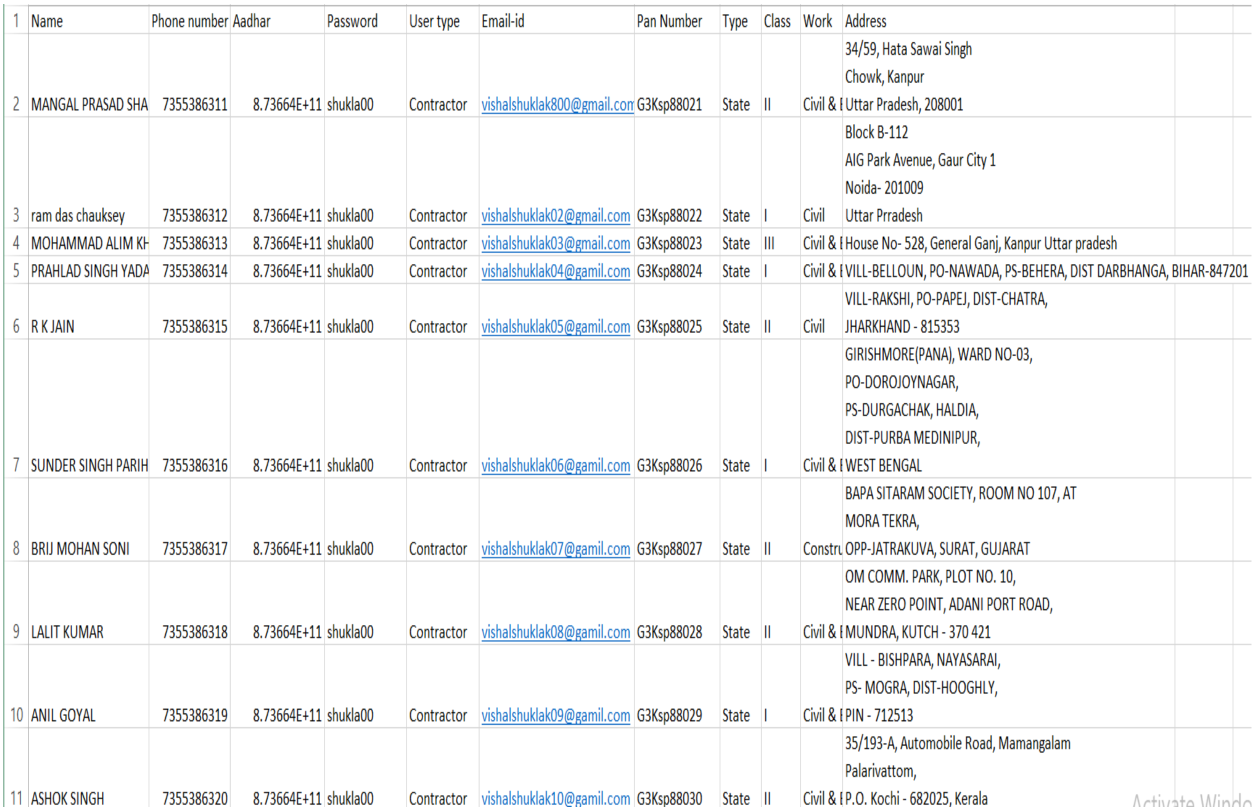
Job Post



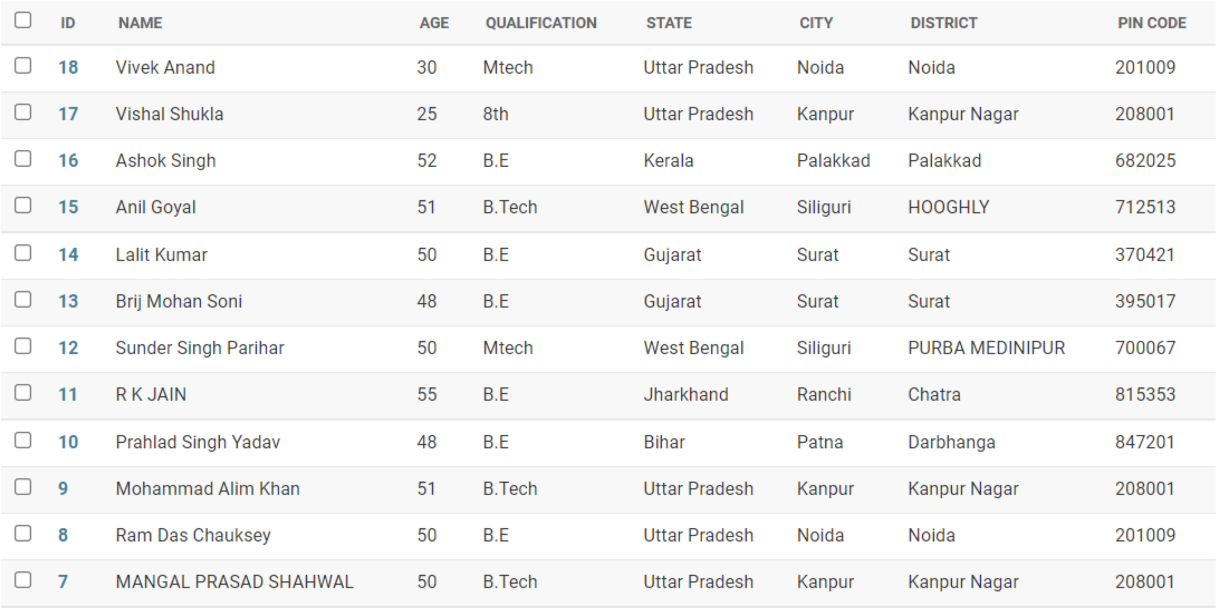
☐	ID	TYPE OF WORK	CONTACT NUMBER	NUMBER OF WORKERS REQUIRE	CITY	ADDRESS
☐	48	Electrician engaged in construction	7355386320	15	Palakkad	Puthur Malampuzha,Palakkad Kerala 678005 India Puthur in
☐	47	Electrician engaged in construction	7355386319	15	Siliguri	New Balarampur Halt Berhampore West Bengal 742101 India in
☐	46	Worker engaged in concrete mixing	7355386318	10	Surat	Mundra Airport Mundra-Luni Gujarat 370421 India in
☐	45	Worker engaged in concrete mixing	7355386317	8	Surat	Indian Oil Panaas, Athwa City Light Road Surat Gujarat 395017 India in
☐	44	Worker engaged in concrete mixing	7355386316	20	Siliguri	Kolkata Chitpur Canal Circular Road Ultadanga Kolkata West Bengal 700067 India in
☐	43	Worker engaged in concrete mixing	7355386315	5	Ranchi	St, Xavier College Purluia Road Chutia Ranchi Jharkhand 834001 India in
☐	42	Mason	7355386314	20	Patna	Patliputra AIIMS- Digha elevated road Priyadarshi Nagar Danapur Bihar 801105 India in
☐	41	Mason	7355386313	15	Lucknow	Gomati Nagar Shaheed Path Viram Khand 5 Gomti Nagar Gwari Uttar Pradesh 226010 India in
☐	40	Mason	7355386312	12	Noida	A-Block Park Sector 52 Sector 52 Mamura Uttar Pradesh 201301 India in
☐	39	Mason	7355386311	10	Kanpur	Chandra Shekhar Azad Park Kanpur Uttar Pradesh 208012 India in

10 job posts

User Information

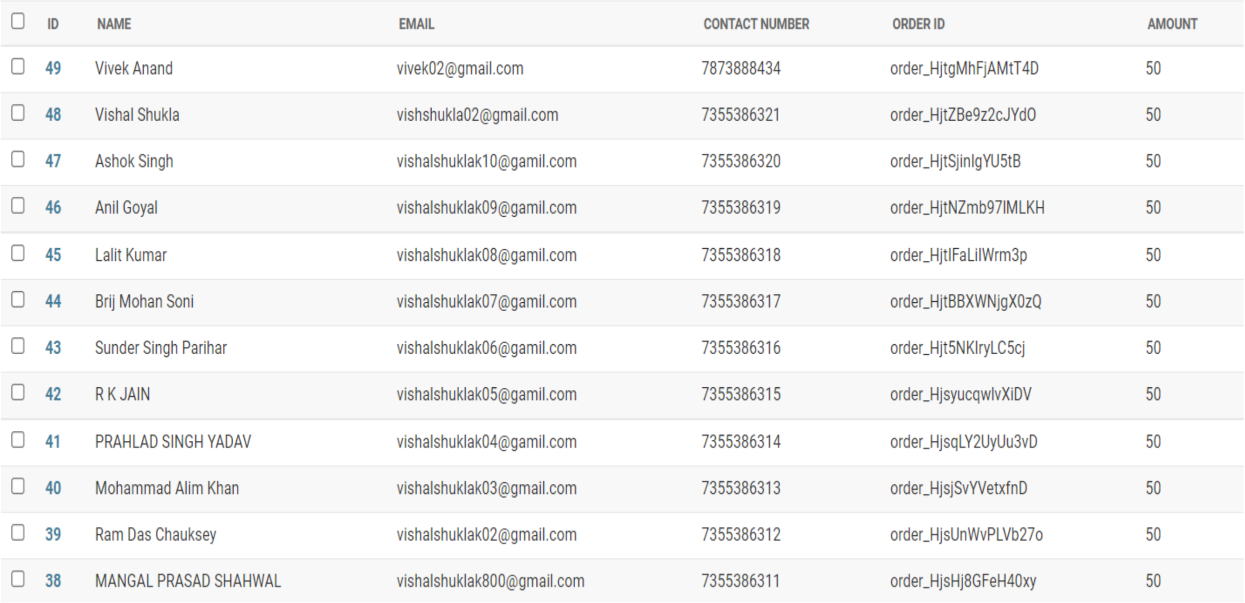


1	Name	Phone number	Aadhar	Password	User type	Email-id	Pan Number	Type	Class	Work	Address
2	MANGAL PRASAD SHA	7355386311	8.73664E+11	shukla00	Contractor	vishalshuklak800@gmail.com	G3Ksp88021	State	II	Civil & E	34/59, Hata Sawai Singh Chowk, Kanpur Uttar Pradesh, 208001
3	ram das chauksey	7355386312	8.73664E+11	shukla00	Contractor	vishalshuklak02@gmail.com	G3Ksp88022	State	I	Civil	Block B-112 AIG Park Avenue, Gaur City 1 Noida- 201009 Uttar Prradesh
4	MOHAMMAD ALIM KH	7355386313	8.73664E+11	shukla00	Contractor	vishalshuklak03@gmail.com	G3Ksp88023	State	III	Civil & E	House No- 528, General Ganj, Kanpur Uttar pradesh
5	PRAHLAD SINGH YADA	7355386314	8.73664E+11	shukla00	Contractor	vishalshuklak04@gamil.com	G3Ksp88024	State	I	Civil & E	VILL-BELLOUN, PO-NAWADA, PS-BEHERA, DIST DARBHANGA, BIHAR-847201
6	R K JAIN	7355386315	8.73664E+11	shukla00	Contractor	vishalshuklak05@gamil.com	G3Ksp88025	State	II	Civil	VILL-RAKSHI, PO-PAPEJ, DIST-CHATRA, JHARKHAND - 815353
7	SUNDER SINGH PARIH	7355386316	8.73664E+11	shukla00	Contractor	vishalshuklak06@gamil.com	G3Ksp88026	State	I	Civil & E	GIRISHMORE(PANA), WARD NO-03, PO-DOROJOYNAGAR, PS-DURGACHAK, HALDIA, DIST-PURBA MEDINIPUR, WEST BENGAL
8	BRIJ MOHAN SONI	7355386317	8.73664E+11	shukla00	Contractor	vishalshuklak07@gamil.com	G3Ksp88027	State	II	Constru	BAPA SITARAM SOCIETY, ROOM NO 107, AT MORA TEKRA, OPP-JATRAKUVA, SURAT, GUJARAT
9	LALIT KUMAR	7355386318	8.73664E+11	shukla00	Contractor	vishalshuklak08@gamil.com	G3Ksp88028	State	II	Civil & E	OM COMM. PARK, PLOT NO. 10, NEAR ZERO POINT, ADANI PORT ROAD, MUNDRA, KUTCH - 370 421
10	ANIL GOYAL	7355386319	8.73664E+11	shukla00	Contractor	vishalshuklak09@gamil.com	G3Ksp88029	State	I	Civil & E	VILL - BISHPARA, NAYASARAI, PS- MOGRA, DIST-HOOGHLY, PIN - 712513
11	ASHOK SINGH	7355386320	8.73664E+11	shukla00	Contractor	vishalshuklak10@gamil.com	G3Ksp88030	State	II	Civil & E	35/193-A, Automobile Road, Mamangalam Palarivattom, P.O. Kochi - 682025, Kerala



□	ID	NAME	AGE	QUALIFICATION	STATE	CITY	DISTRICT	PIN CODE
□	18	Vivek Anand	30	Mtech	Uttar Pradesh	Noida	Noida	201009
□	17	Vishal Shukla	25	8th	Uttar Pradesh	Kanpur	Kanpur Nagar	208001
□	16	Ashok Singh	52	B.E	Kerala	Palakkad	Palakkad	682025
□	15	Anil Goyal	51	B.Tech	West Bengal	Siliguri	HOOGHLY	712513
□	14	Lalit Kumar	50	B.E	Gujarat	Surat	Surat	370421
□	13	Brij Mohan Soni	48	B.E	Gujarat	Surat	Surat	395017
□	12	Sunder Singh Parihar	50	Mtech	West Bengal	Siliguri	PURBA MEDINIPUR	700067
□	11	R K JAIN	55	B.E	Jharkhand	Ranchi	Chatra	815353
□	10	Prahlad Singh Yadav	48	B.E	Bihar	Patna	Darbhanga	847201
□	9	Mohammad Alim Khan	51	B.Tech	Uttar Pradesh	Kanpur	Kanpur Nagar	208001
□	8	Ram Das Chauksey	50	B.E	Uttar Pradesh	Noida	Noida	201009
□	7	MANGAL PRASAD SHAHWAL	50	B.Tech	Uttar Pradesh	Kanpur	Kanpur Nagar	208001

Payment Details



□	ID	NAME	EMAIL	CONTACT NUMBER	ORDER ID	AMOUNT
□	49	Vivek Anand	vivek02@gmail.com	7873888434	order_HjtgMhFjAMtT4D	50
□	48	Vishal Shukla	vishshukla02@gmail.com	7355386321	order_HjtZBe9z2cJYdO	50
□	47	Ashok Singh	vishalshuklak10@gamil.com	7355386320	order_HjtSjinIgYU5tB	50
□	46	Anil Goyal	vishalshuklak09@gamil.com	7355386319	order_HjtNZmb97IMLKH	50
□	45	Lalit Kumar	vishalshuklak08@gamil.com	7355386318	order_HjtIFaLiIWrm3p	50
□	44	Brij Mohan Soni	vishalshuklak07@gamil.com	7355386317	order_HjtBBXWNjgX0zQ	50
□	43	Sunder Singh Parihar	vishalshuklak06@gamil.com	7355386316	order_Hjt5NKIryLC5cj	50
□	42	R K JAIN	vishalshuklak05@gamil.com	7355386315	order_HjsyucqwIvXiDV	50
□	41	PRAHLAD SINGH YADAV	vishalshuklak04@gamil.com	7355386314	order_HjsqLY2UyUu3vD	50
□	40	Mohammad Alim Khan	vishalshuklak03@gmail.com	7355386313	order_HjsjSvYVetxfnD	50
□	39	Ram Das Chauksey	vishalshuklak02@gmail.com	7355386312	order_HjsUnWvPLVb27o	50
□	38	MANGAL PRASAD SHAHWAL	vishalshuklak800@gmail.com	7355386311	order_HjsHj8GFeH40xy	50