

PROJECT NAME  
(Lookout - Employee Recruitment System)



Submitted at  
GUJARAT UNIVERSITY  
&  
C. U. SHAH INSTITUTE OF COMPUTER APPLICATION

**Lookout - Employee Recruitment System**

**A PROJECT REPORT ON**

**Lookout - Employee Recruitment System**

**Submitted to**

**GUJARAT UNIVERSITY**



**For Partial Fulfilment of Degree of  
Bachelor of Computer Application (BCA)  
SEM-5 System Development Project-1**

**Submitted By**

**PANDYA KUNAL**

**Internal Guide:**

**ASST.PROF.DEVANGI PAREKH**

**External Guide:**

**C U. Shah Institute of Computer Application**

**Ashram Road, Ahmedabad- 380014**

**Website : <https://www.cushahbca.ac.in>**

**Contact no: +91 635-925-9669**

## **Acknowledgement**

The success and final outcome of this project required a lot of guidance and assistance from many people and I am extremely privileged to have got this all along the completion of my project. All that I have done is only due to such supervision and assistance and I would not forget to thank them.

We respect and thank **Asst. Prof. Devangi Parekh** , for providing me an opportunity to do the project work in **C.U. Shah Institute of Computer Application** and giving us all support and guidance which made me complete the project duly.

We owe my deep gratitude to our project guide **Asst. Prof. Devangi Parekh** , who took keen interest on our project work and guided us all along, till the completion of our project work by providing all the necessary information for developing a good system.

Last but not the least we are also thankful to our friends, project partners, colleagues and parents for support and understanding the provided us during the project work. We are also very much thankful to all who directly or in directly helped us.

We are also thanking our college faculties **Dipti Bhatt, Shilpa Mehta, Praful Devadiya**

**Kunal Pandya [3118]**

## INDEX

• Project Profile.....	[7]
• Existing System.....	[9]
• Proposed System.....	[9]
• Development Tools and Technologies.....	[11]
• Advantages and Disadvantages.....	[14]
• System Flow Diagram.....	[15]
• ER Diagram.....	[20]
• UML Diagram.....	[23]
• Use case Diagram.....	[26]
• Activity Diagram.....	[33]
• Sequence Diagram.....	[40]
• Class Diagram.....	[46]
• Data Dictionary.....	[49]
• References.....	[49]

## Project Profile

<b>Project Name</b>	Lookout – Employee Recruitment System
<b>Objective</b>	Jobseeker Can Apply for the Dream Jobs, Also Organizations can fulfil them staff Requirement
<b>Group Members</b>	1
<b>Group Members Name</b>	Kunal Pandya
<b>Platform</b>	Website
<b>Institute</b>	C.U. Shah Institute of Computer Application

**COMPANY PROFILE :**

Company Name:	
Address:	
Owner Name:	
Contact Number:	

### **Existing System:-**

- Many organizations are Investing in Students and giving First Priority to the students. If one organization will going in any campus and offering jobs another campus will miss that company.
- It is time Consuming & Not Guaranteed To Get skilled jobseeker.
- Many Skilled Person Cannot find good companies which are giving right salary for the work

### **Propose System: -**

- Employee Recruitment System by overall is aims to facilitate the applicant to apply for the job online. Indirectly, it is also to facilitate the managerial department of an organization for an optimized and systematic employee recruitment process.
- A quick look at the overall trends in Online recruiting shows the rise in the importance of recruiting on the web site, online training, dawn of video interviews and emergence of professional Internet Recruiters. Online recruiting systems, with its emphasis on a more strategic decision making process is fast gaining ground as a popular outsourced function.
- The system also provides the global platform for both jobseekers and the organization, where the jobseekers can find their dream jobs and organization can find the right candidate to fulfill staff requirements.
- This system has web-forms like registration form, login form and account pages like user applicant page, admin account page etc. It includes following main module Admin module. Candidate module An administrator can be a manager who has full authority over the whole system. The administrator is able to update and retrieve data from the account of candidate. The candidate is a center of this system. He has to register himself to use the services of the system.

## **1.2 Purpose:**

- The primary purpose to develop this system is to optimize the recruitment process for an organization. Besides, the qualified applicants could be sort by this system based on their qualifications and company requirements.
- The system has been designed to do a whole lot more than just reduce paperwork. It can make a significant contribution to a company's marketing and sales activities.

## **1.3 Project**

### **Scope:**

- Online Recruitment System enables the users to have the typical recruitment facilities and features at their disposal. It resolves typical issues of manual staffing processes and activities into a controlled and closely monitored work flow in the architecture of the application. This multi platform solution brings in by default, the basic intelligence and immense possibilities for further extension of the application as required by the user. The system makes it simpler to share and manage the organization's human resource requirements with higher efficiency and easiness. The objective of these websites is to serve as a common meeting ground for jobseekers and organization, both locally and globally. This kind of systems is specifically designed for organization to help in solving staffing problems and managing human resource department activities at high degree of optimization.

## **1.4 Objectives:**

- This software helps applicants to find suitable job within the organization and  
apply for that job easily.
- The software helps in managing and viewing details of interested applicants for  
the administrator.
- The system is capable of sorting and filtering best suitable candidates based



on

some criteria, . Company will not have to waste his time for finding right employee at right post

## **1.4 TECHNOLOGY AND LITERATURE REVIEW**

### **Introduction of PHP**

- PHP is a server side scripting language. PHP is just a platform on which you can create dynamic web application. It is initiative is broad-based and very grand. It is open source, which encompasses the languages and execution platform, plus extensive class libraries, providing rich built-in functionality.

### **An Overview of PHP**

PHP is a server-side scripting language designed for web development but also used as a general-purpose programming language. PHP is now installed on more than [244](#) million websites and [2.1](#) million web servers. Originally created by Rasmus Lerdorf [1995](#), the reference implementation of PHP is now produced by The PHP Group. While PHP originally stood for Personal Home Page, it now stands for PHP: Hypertext Preprocessor

PHP code is interpreted by a web server with a PHP processor module, which generates the resulting web page: PHP commands can be embedded directly into an HTML. Source document rather than calling an external file to process data. It has also evolved to include a command-line interface capability and can be used in standalone graphical applications.

PHP is free software released under the PHP License. PHP can be deployed on most web servers and also as a standalone shell on almost every operating system and platform, free of charge.

### **MySQL**

MySQL is the world's second most widely used open-source relational database

management system (RDBMS) The SQL phrase stands for Structured Query Language.

The MySQL development project has made its source code available under the terms of

the GNU General Public License, as well as under a variety of proprietary agreements.

MySQL is a popular choice of database for use in web applications, and is a central

component of the widely used LAMP open source web application software stack (and other AMP stacks).

### **LAMP:**

LAMP is an acronym for "Linux, Apache, MySQL, PHP"

The exact combination of the software included in a LAMP stack is prone to variation, for example Apache web server can be replaced by some other web server software.

Though the original authors of these programs did not design them to work as a

component of the LAMP stack, the development philosophy and tool sets are shared and were developed in close conjunction, so they work and scale very well together. The software combination has become popular because it is entirely free and open-source

software, which means that each component can be adapted to the underlying hardware and customized to meet the specification as exactly as possible, without the slightest vendor lock-in. The complete software stack is also free of cost, maximizing the available budget for tailoring the hardware and software.

Due to the nature of free and open-source software and the ubiquity of its components, each component of the LAMP stack is very well tested regarding performance and security. At the same time, there is an abundance of experienced contractors to do the tailoring required for various customizations, or for complex setups. There is also constant development going on.

-

### **Introduction to XAMPP:**

XAMPP requires only one zip, tar, 7z, or exe file to be downloaded and run, and little or no configuration of the various components that make up the web server is required. XAMPP is regularly updated to incorporate the latest releases of Apache, MySQL, PHP and Perl. It also comes with a number of other modules including OpenSSL and phpMyAdmin.

Self-contained, multiple instances of XAMPP can exist on a single computer, and any given instance can be copied from one computer to another.

It is offered in both a full, standard version and a smaller version.

### **APACHE HTTP Server**

The Apache HTTP Server, commonly referred to as Apache is a web server application notable for playing a key role in the initial growth of the World Wide Web. Originally based on the NCSA HTTPd server, development of Apache began in early [1995](#) after work on the NCSA code stalled. Apache quickly overtook NCSA HTTPd as the dominant HTTP server, and has remained the most popular HTTP server in use since April [1996](#). In [2009](#), it became the first web server software to serve more than [100](#) million websites.

Apache is developed and maintained by an open community of developers under the auspices of the Apache Software Foundation. Most commonly used on a Unix

like system, the software is available for a wide variety of operating systems.

including Unix FreeBSD, Linux, Solaris, Novel NetWare, OS X, Microsoft Windows, OS/2, TPF. OpenVMS and eComStation. Released under the Apache License, Apache is open-source software.

,

❖ **Advantages :**

- It will Provide Single Platform to Get & Give the Job.
- It will Make recruiting process of faster
- It will Show number of the current jobs with necessary Description .
- User Friendly UI
- NO bidding System and long Processes
- Easier to use

**(Diagram)**  
**System Flow Diagram**

## **System Flow Diagram**

- A System Flow Diagram is a way to show Relationships between a Business and its components, such as Customers (according to IT Toolbox).

System Flow Diagram also Known as “**Process Flow Diagram**” or “**Data Flow Diagrams**”

- **What is a System Flow?**

System Flows are Systems models that show the Activities and Decisions that Systems execute.





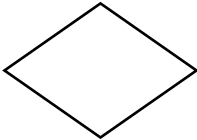

- **What is System Diagram?**

A System Diagram is a Visual model of a System, its components and their interactions. With supporting Documentation. It can Capture All the essential information of a System's Design.

## **Symbols Using For System Flow Diagram**

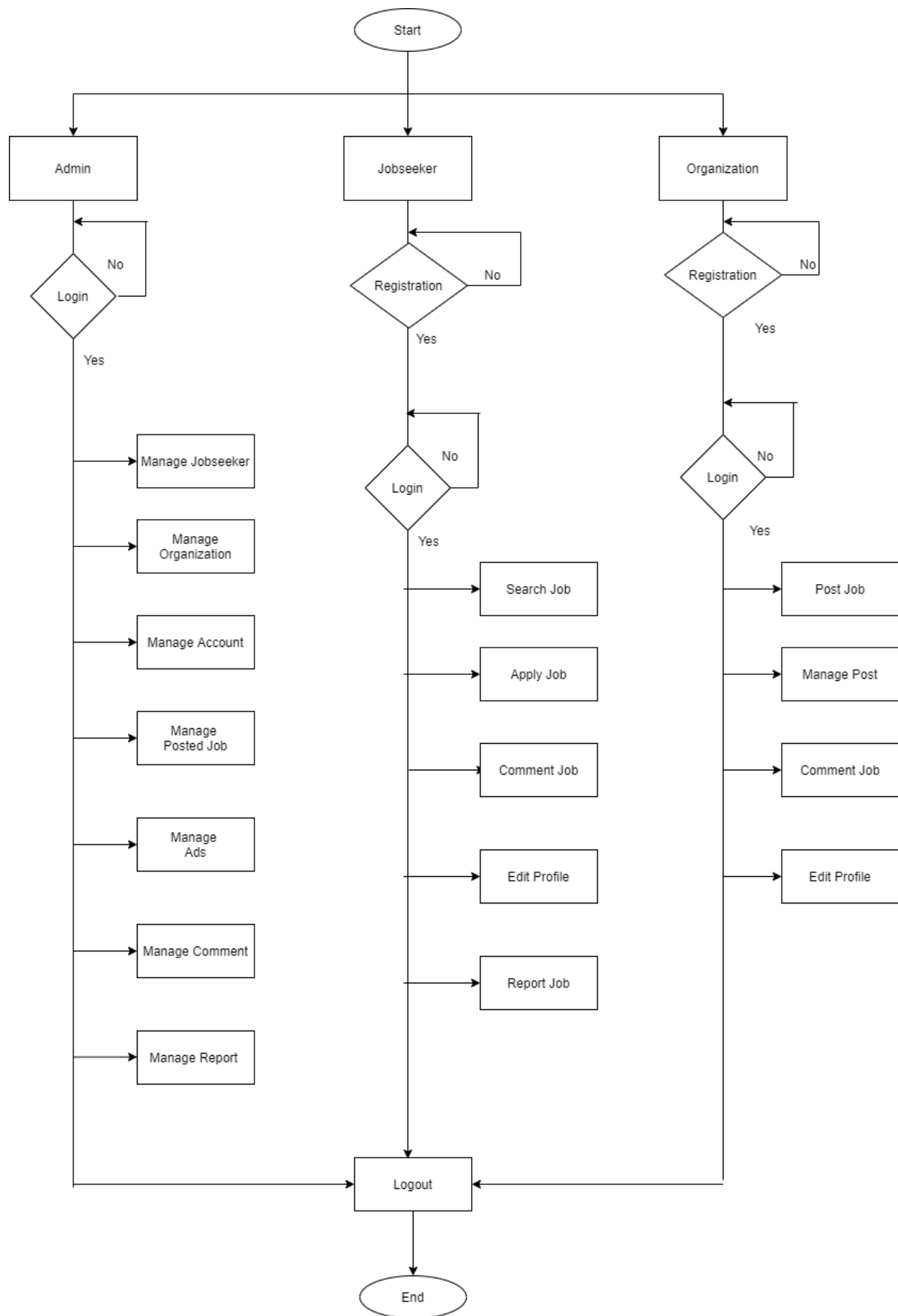
<b>Oval</b>	An End or a Beginning The oval used to Represent the Start and end of a process
<b>Rectangle</b>	A step in the Flowcharting Process The Rectangle is your go-to Symbol
<b>Arrow</b>	Directional Flow
<b>Diamond</b>	Call for a Decision

### Symbols Using For System Flow Diagram:

Symbol	Name
	Start
	Arrows
	Input / Output
	Process
	Decision
	End



## Lookout – Employee Recruitment System (Job Portal)



## **ER Diagram**

## **ER Diagram**

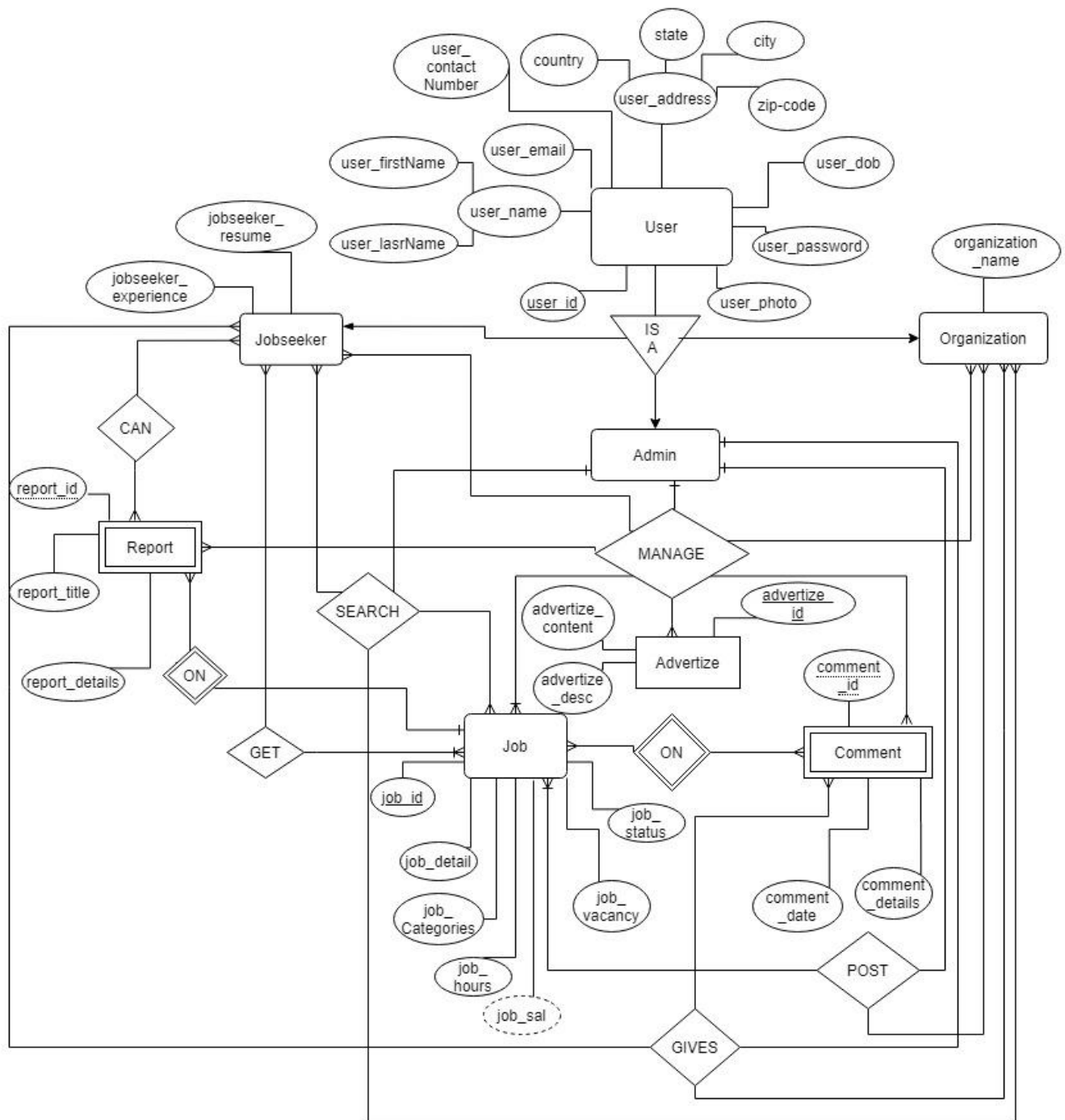
### **❖ What is ER Diagram?**

- An Entity is an Object or Component of Data.
- An Entity is represented as Rectangle in an ER Diagram.

### **❖ Why we used ER Diagram?**

- ER Diagrams are most often used to Design or Debug Relational Databases in the Fields of Software Engineering, Business information Systems, Education and Research.
- ER Diagrams Also Are often used in Conjunction with Data Flow Diagrams (DFDs), which map out the Flow of Information for Processes or Systems.

## ER Diagram



**UML Diagram**

## **UML Diagram**

- **What is UML and what is it used for?**

- A UML Diagram is a Diagram based on the UML (Unified Modelling Language) with the purpose of visually Representing a System along with its main Actors, roles, actions, artifacts or classes, in order to better understand, alter, maintain or Document information about the System.
- UML used by Software Developers.
- UML can be used to Develop Diagrams and provide users with ready-to-use, expressive modelling examples.

- **Why do we use UML Diagrams?**

- UML has Applications beyond Software Development, such as process Flow in Manufacturing.
- It is Analogous to the Blueprints used in other Fields, and Consists of different types of Diagrams.
- In the Aggregate, UML Diagrams Describe the Boundary, Structure and the behaviour of the System and the objects within it.

- **Why are UML Diagrams Important?**

- The Importance of Using UML for Modelling.
- UML is a powerful tool that can greatly improve the quality of your Systems Analysis and Design, and it is hoped that the improved practices will translate into higher quality Systems.

- **What is the need of UML Diagrams?**

- "Importance of UML Diagrams in Software Development".
- The Unified Modelling Language (UML) is a Standard Language for Specifying, Visualizing, Constructing and documenting the artefacts of Software Systems as well as for Business modelling and other Non-software Systems.

- **Advantages : -**

- Provides standard for Software Development.
- Reducing of Costs to Develop Diagrams of UML using supporting tools.
- Development time is reduced.
- The past faced issues by the Developers are no longer exists.

- **Types Of UML Diagram : -**

1. Use Case Diagram
2. Activity Diagram
3. Sequence Diagram
4. Class Diagram

## **Use case Diagram**



## **Use case Diagram**

### **❖ Purpose of a Use case Diagram: -**

- Use case Diagram is one of them and its specific Purpose is to gather System Requirements and Actors.
- Use case Diagrams specify the events of a system and their flows.
- But Use case Diagram never describes how they are implemented.
- A Use case describes a sequence of actions that provide something of measurable value to an Actor and is drawn as a Horizontal ellipse.

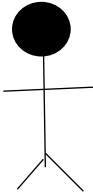
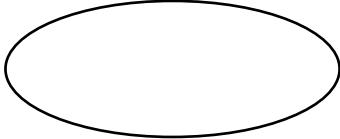
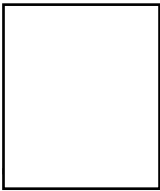
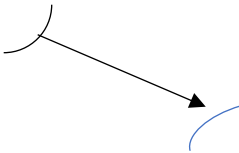
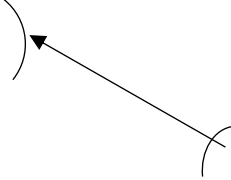
### **❖ Importance of Use cases: -**

- Use cases are important because they are in a tracking format. Hence they make it easy to comprehend about the functional Requirements in the system and also make it easy to identify the various interactions between the Users and the Systems within an Environment.

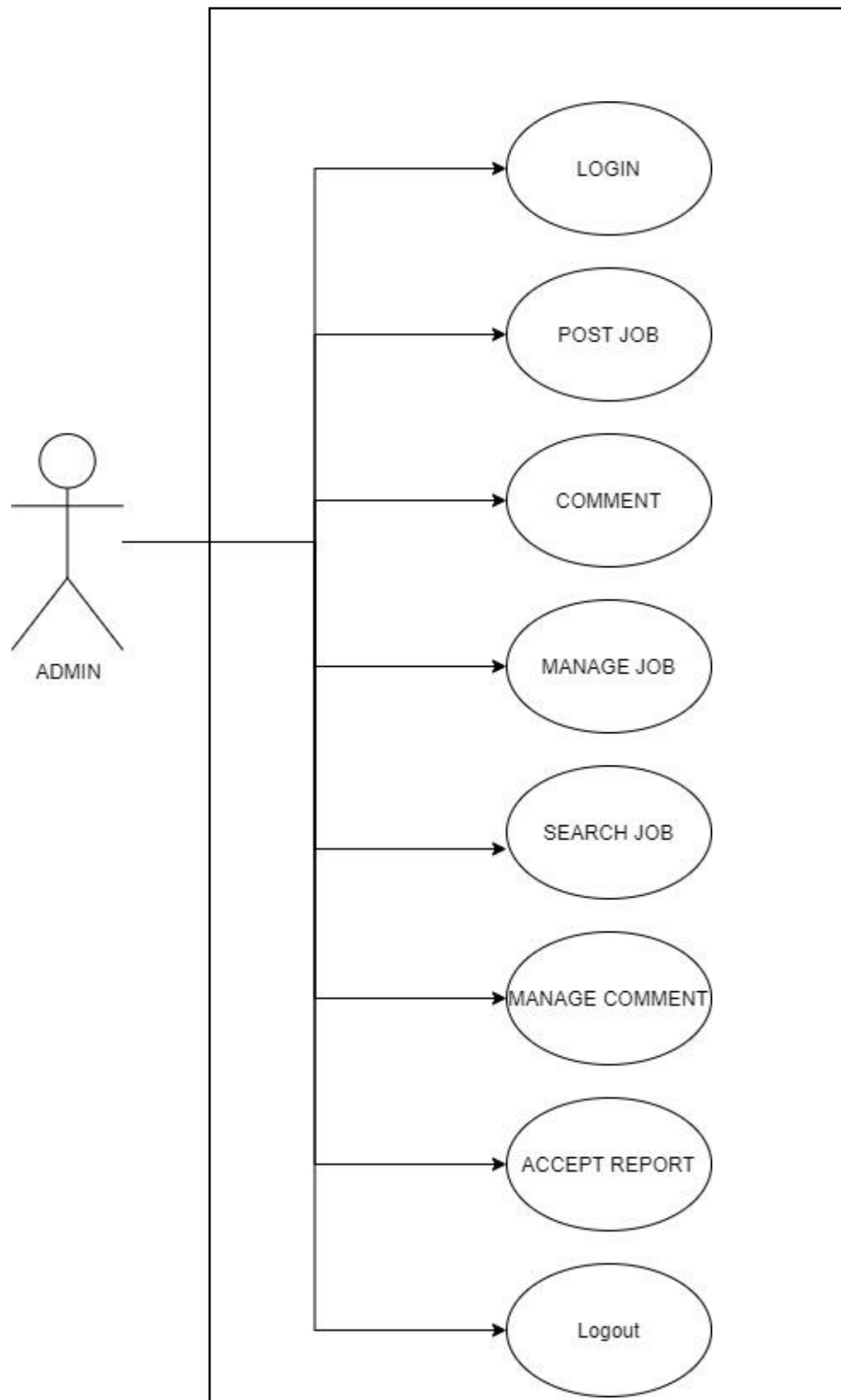
## Use case Diagram Notations

<b>Actor</b>	Actors are usually individuals involved with the system defined According to their roles.
<b>Use Case</b>	A use case describes how Actors uses a system to accomplish a particular goal.
<b>Relationship</b>	The Relationships between and among the Actors and the Use cases.
<b>System Boundary</b>	Boundary of inside whole System But Actors are Outside of the System Boundary.

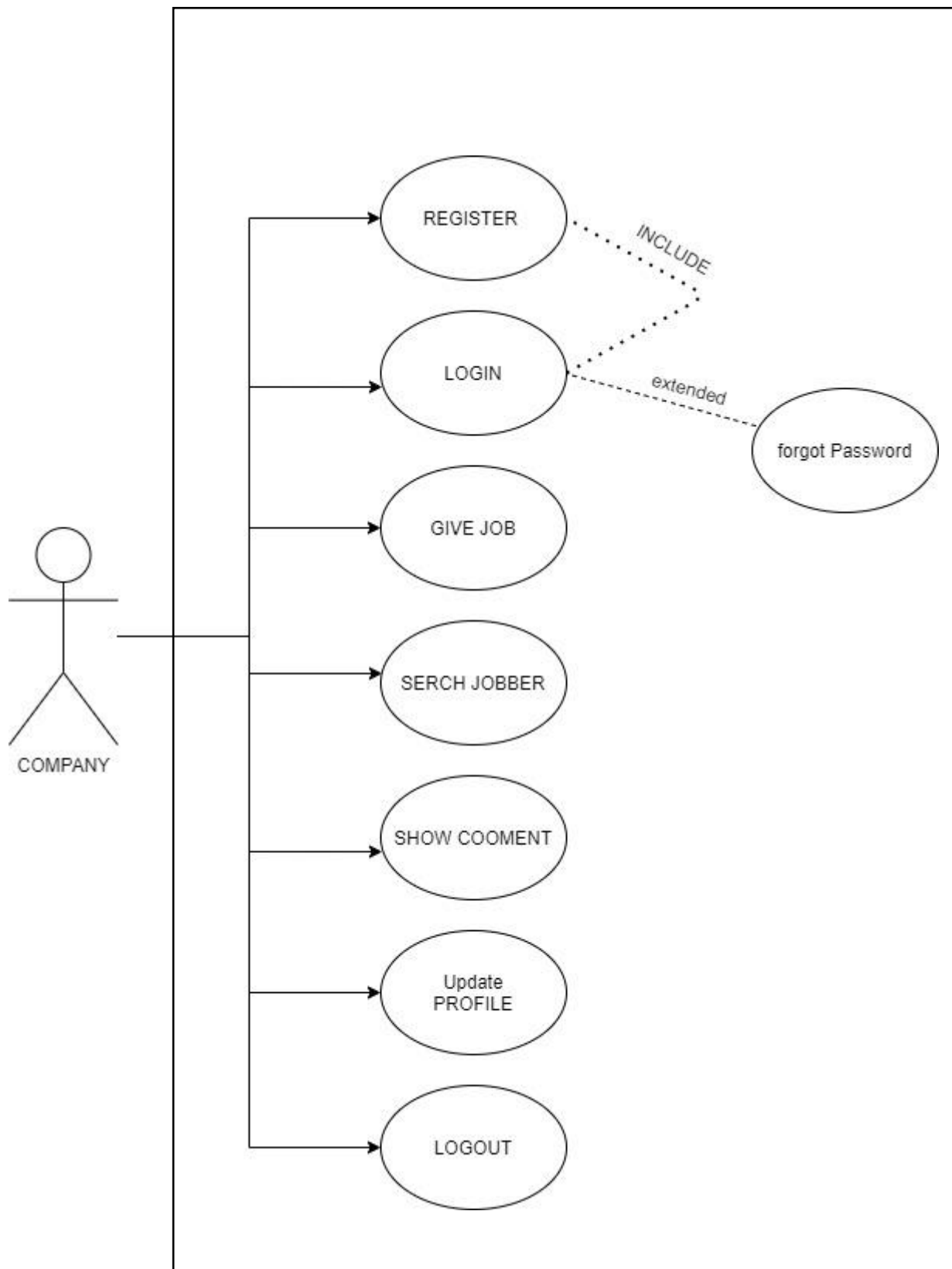
## Symbol of Use case Diagram

Symbols	Overview
 <p style="text-align: center;">Actor</p>	<p>Actors are the entities that interact with a system. Although in most cases, actors are used to represent the users of system, actors can actually be anything that needs to exchange information with the system. So, an actor may be people, computer hardware, other systems, etc.</p>
	<p>A use case represents a user goal that can be achieved by accessing the system or software application. In Visual Paradigm, you can make use of the sub-diagram feature to describe the interaction between user and sys within a use case by creating a sub-sequence diagram under a use case. You can also describe the use case scenario using the Flow of Events editor.</p>
	<p>The system can work in a system Boundary. The scope of a system can be represented by a system (shape), or sometimes known as a system boundary. The use cases of the system are placed inside the system shape, while the actor who interact with the system are put outside the system. The use cases in the system make up the total requirements of the system.</p>
 <p>&lt;&lt;include&gt;&gt;</p>	<p>An include relationship specifies how the behaviour for the inclusion use case is inserted into the behaviour defined for the base use case.</p>
 <p>&lt;&lt;Extend&gt;&gt;</p>	<p>An extend relationship specifies how the behaviour of the extension use case can be inserted into the behaviour defined for the base use case.</p>

Admin UML:

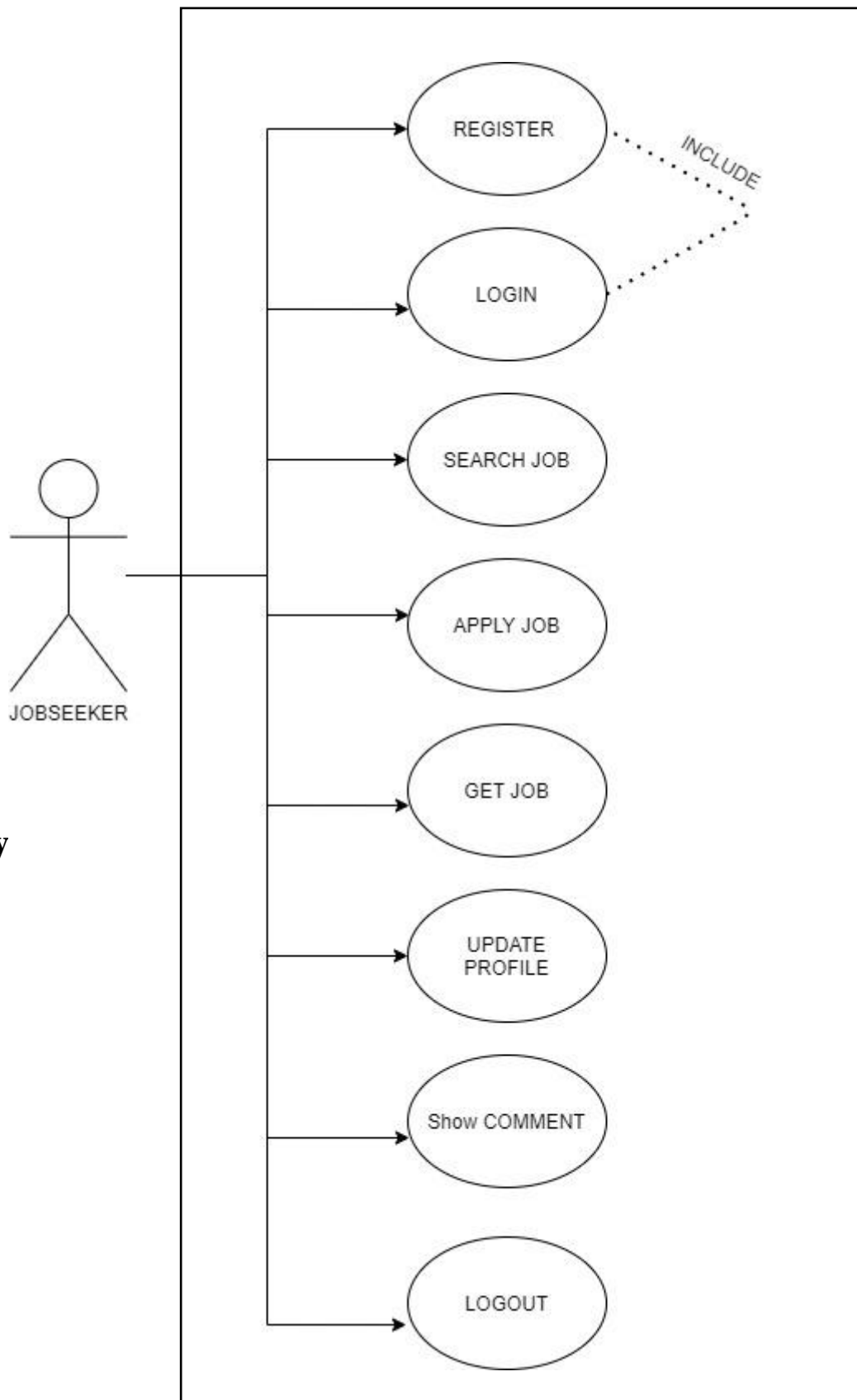


**Organization UML:**



**Jobseeker UML:**

**Activity**



**Activity Diagram**

## **Activity Diagram**

### **❖ What is Activity Diagram Definition?**

- In Unified Modelling Language (UML), an Activity Diagram is a Graphical Representation of an executed Set of Procedural System.
- Activities and considered a State Chart Diagram variation.
- Activity Diagrams Describe Parallel and Conditional Activities Use cases and System Functions at a Detailed Level.

### **❖ Why do we use Activity Diagram?**

- The Basic Usage of Activity Diagram is Similar to other Four UML Diagrams. The Specific Usage is to Model the Control Flow from one Activity to another.
- This Control Flow does not include Messages.
- Activity Diagram is Suitable for Modelling the Activity Flow of the System.

### **❖ How do you write an Activity Diagram?**

In Each of these Cases, Here's How to Draw an Activity Diagram From the Beginning.

**Step 1:** Figure out the Action steps From the Use case.

**Step 2:** Identify the Actors who are involved.

**Step 3:** Find a Flow among the Activities.

**Step 4:** Add Swim lanes.

### **❖ What are the Elements of Activity Diagram?**

Fundamental Elements of the Activity are Actions and Control Elements:

- Decision
- Division
- Merge
- Initiation etc...



## **Lookout – Employee Recruitment System (Job Portal)**

Elements are connected by so-called "Activity Edges" and Form the "Control Flow", which can also be casually called "Flow".

The execution of an Activity can contain Parallel Flows.

### **❖ What is Swim lane Diagram?**

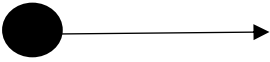



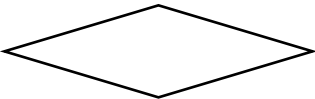
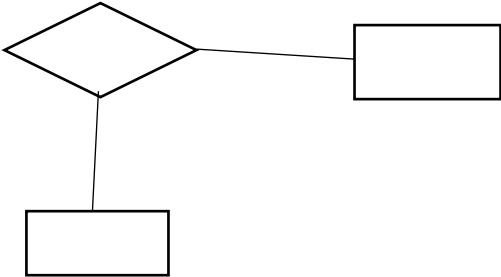
- Swim lane From Wikipedia, the Free Encyclopaedia.
- A Swim lane Diagram is used in Process Flow Diagrams or Flowcharts, that visually Distinguishes Job Sharing and Responsibilities for Sub-Processes of a Business Process.
- Swim lanes may be arranged either "Horizontally" or "Vertically".

### **❖ What is Swim lanes in Activity Diagram?**

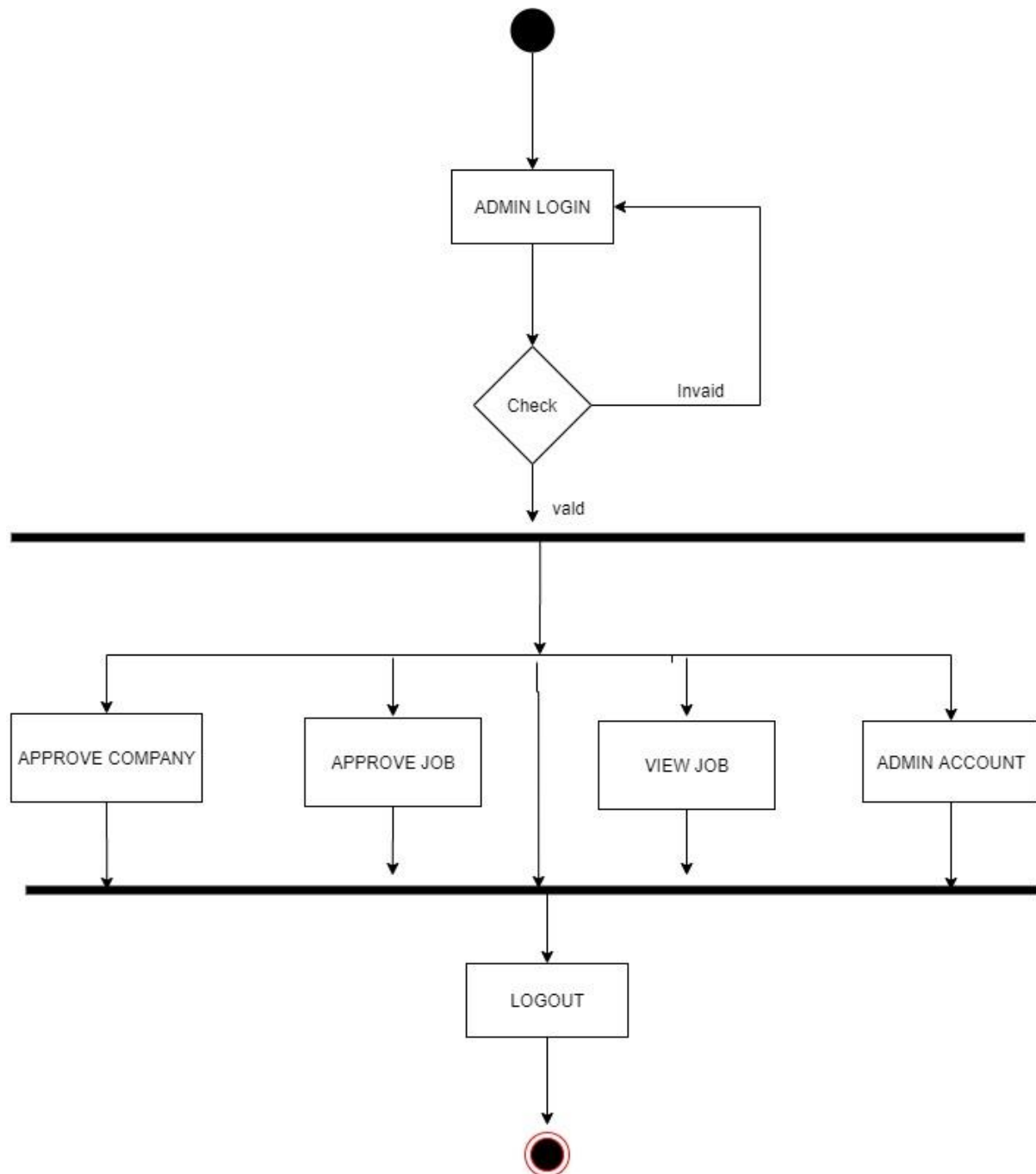
- A Swim lane Diagram is a Type of Flowchart that Delicates who does what in a Process.
- Using the Metaphor of Lanes in a Pool.

✓ A Swim lane Diagram Provides Clarity and Accountability by Placing Process Steps within the Horizontal or Vertical "Swim lanes of a Particular Employ work Group or Department.

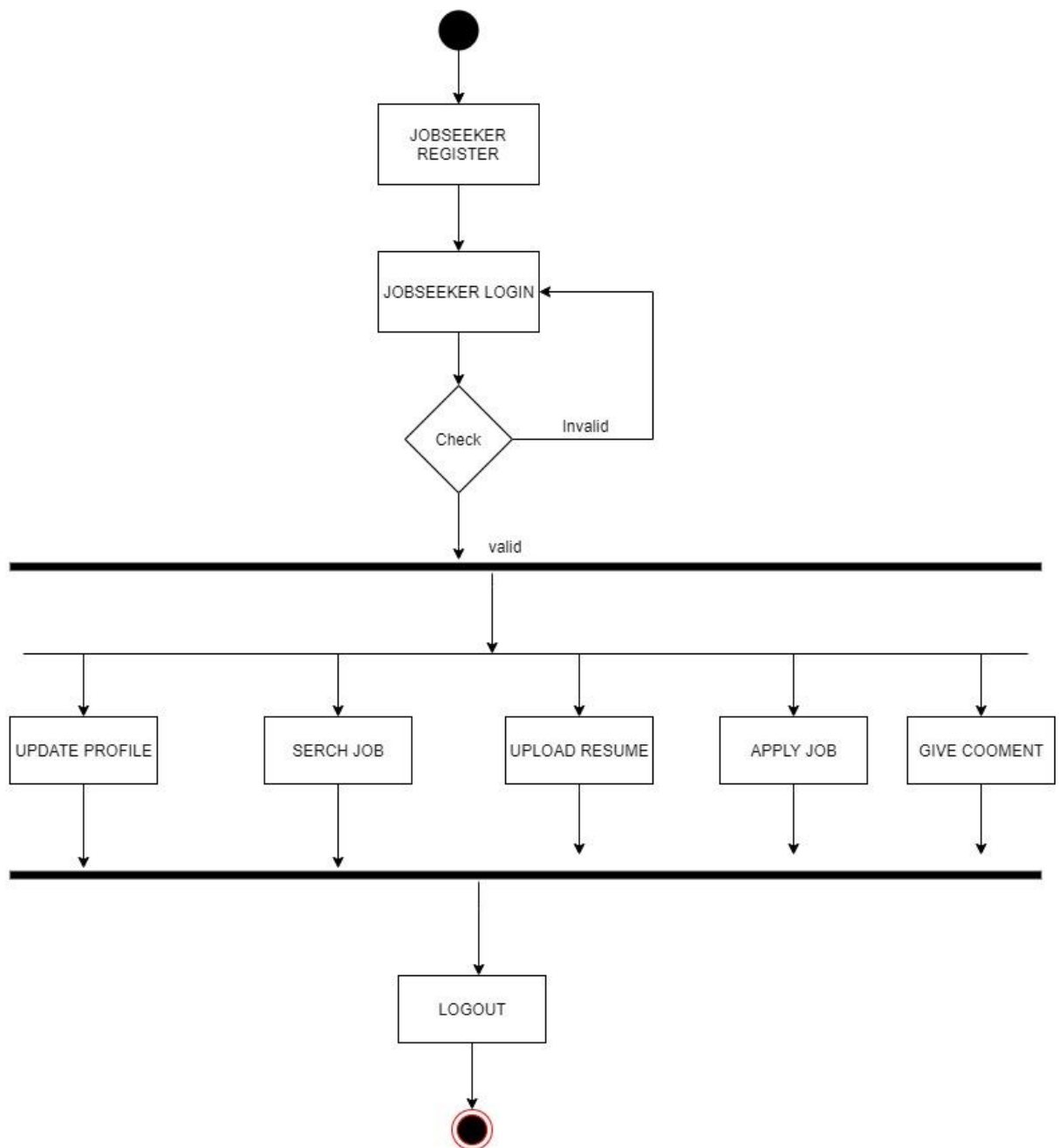
## Symbol of activity Diagram

Symbol	Overview
<p>Start point/initial state</p> 	<p>A small filled circle followed by an arrow represents the initial action state or the start point for any activity diagram. For activity diagram using swim lanes, make sure the start point is placed in the top left corner of the first column.</p>
<p>End point symbol</p> 	<p>An arrow pointing to a filled circle nested inside another circle represents the final action State (Ending Point).</p>
<p>Activity</p> 	<p>An action state represents the non-interruptible action of objects. You can draw an action state in Smart-Draw using a rectangle with rounded Corners.int could Activity.</p>
<p>Action flow</p> 	<p>Action flows, also called edges and paths, illustrate the transitions from one action state to another. They are usually drawn with an arrowed line.</p>
<p>Decision symbol</p> 	<p>A diamond represents a decision with alternate paths. When an activity requires a decision prior to moving on to the next activity, add a diamond between the two activities. The outgoing alternates should be labeled with a condition or guard expression. You can also label one of the paths "else."</p>
<p>Guard system</p> 	<p>In UML, guards are a statement written next to a decision diamond that must be true before moving next to the next activity. These are not essential, but are useful when a specific answer, such as "Yes, three labels are printed," is needed before moving forward.</p>

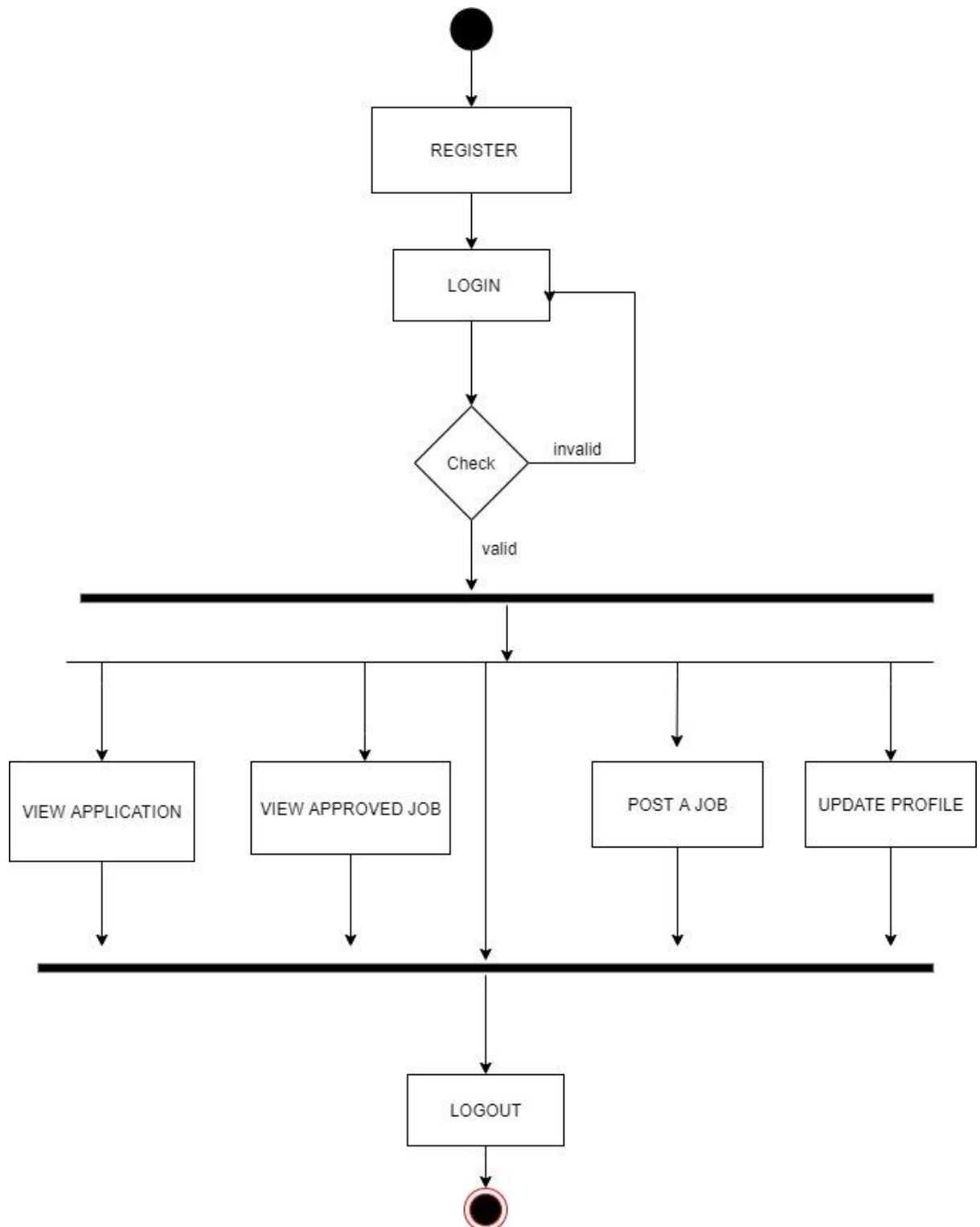
## Admin Activity Diagram



### Jobseeker Activity Diagram



## Organization Activity Diagram



## **Sequence Diagram**

## **Sequence Diagram**

### **❖ What is Sequence Diagram?**

- Sequence Diagram Tutorial.
- A Sequence Diagram Describes an interaction among a Set of Objects Participated in a Collaboration or Scenario, Arranged in a Chronological order.
- It Shows the Objects Participating in the interaction by their "Lifelines" and "Messages" that they send to each other.

### **❖ What is a Sequence Diagram used for?**

- UML Sequence Diagrams model the Flow of Logic within your System in a Visual manner, enabling you both to Document and Validate your Logic and are commonly used for both Analysis and Design Purposes.

### **❖ Why is Sequence Diagram used?**

- The Sequence Diagram is a good Diagram to use to Document a System's Requirements and to flush out a System's Design. > The Reason the Sequence Diagram is so useful is because it shows the interaction Logic.
- Between the Objects in the System in the time order that the in interactions take Place.

### **❖ Basic Sequence Diagram Notations:**

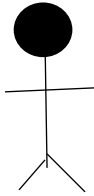
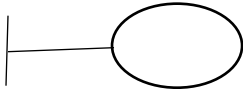
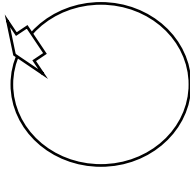
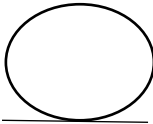
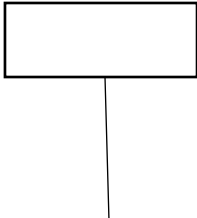
- Class Roles or Participants. Class roles describe the way an Object will behave in Context.
- Activation or Execution Occurrence. Activation boxes Represent the time an Object needs to complete a task
- Messages.
- Lifelines.
- Destroying Objects.
- Loops.
- Synchronous Message.
- Asynchronous Message.

**❖ What are the Elements of Sequence Diagram?**

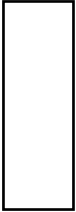
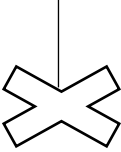
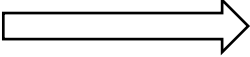

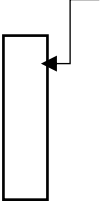
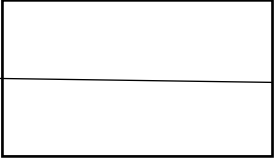
- The Following Nodes and Edges are typically Drawn in a UML Sequence Diagram :
- Lifeline
- Execution Specification
- Message
- Combined Fragment
- Interaction Use
- State invariant
- Continuation
- Destruction Occurrence.



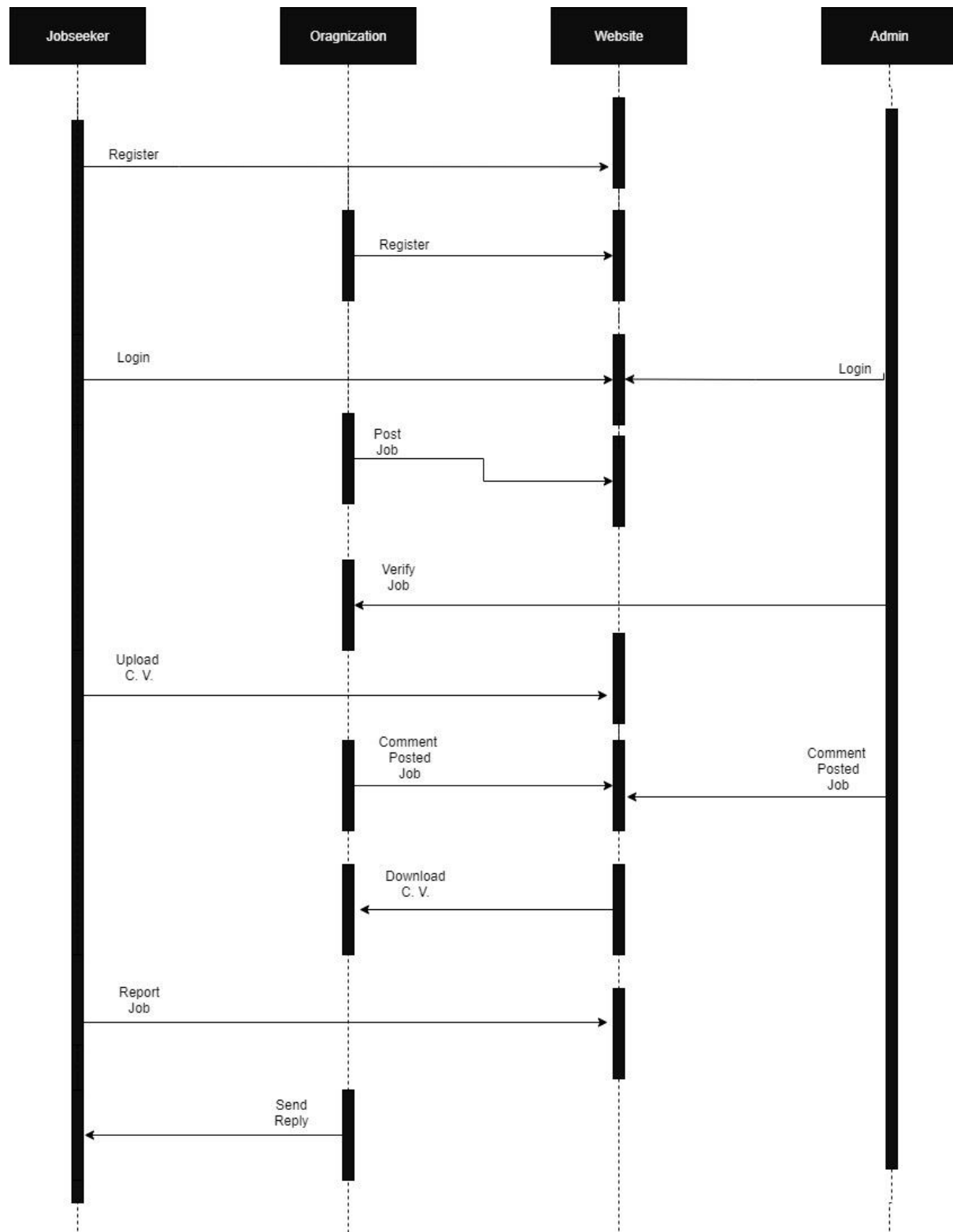
## Symbol of Sequence Diagram

Symbol	Overview
 actor	Actors are the entities that interact with a system. Although in most cases, actors are used to represent the users of system, actors can actually be anything that needs to exchange information with the system. So, an actor may be people, computer hardware, other systems, etc.
 boundary object	Objects that interface with system actors (e.g. a user or external service). Windows, screens and menus are examples of boundaries that interface with users.
 Control object	Objects that mediate between boundaries and entities. These serve as the glue between boundary elements and entity elements, implementing the logic required to manage the various elements and their interactions. It is important to understand that you may decide to implement controllers within your design as something other than objects-many controllers are simple enough to be implemented as a method of an entity or boundary class for example.
 Entity object	Objects representing system data, often from the domain model.
	A sequence diagram is made up of several of these lifeline notations that should be arranged horizontally across the top of the diagram. No two lifeline notations should overlap each other. They represent the different objects or parts that interact with each other in the system during the sequence.

## Lookout – Employee Recruitment System (Job Portal)

	<p>Activation boxes represent the time an object needs to complete a task. When an object is busy executing a process or waiting for a reply message, use a thin gray rectangle placed vertically on its lifeline. To</p>
	<p>This Symbol is Known as lifeline end symbol. This symbol Work with lifeline.</p>
<p>message</p> 	<p>This Symbol is Known as lifeline end symbol. This symbol Work with lifeline.</p>
<p>return</p> 	<p>A reply message is drawn with a dotted line and an open arrowhead pointing back to the original lifeline.</p>
	<p>A message an object sends to itself, usually shown as a U shaped arrow pointing back to itself.</p>
	<p>This Symbol is could as AlterNet. This symbol have two parts valid and invalid part. This symbol is check condition of message when condition is true ten pass positive response otherwise pass negative response,</p>

## Sequence Diagram



## **Class Diagram**

## **Class Diagram**

### **❖ What is Class Diagram?**

- A Class Diagram is an illustration of the Relationships and Source code Dependencies among Classes in the Unified Modelling Language (UML).
- In this Context, a lass Defines the Methods and Variables in an Object .which is a Specific Entity in a Program or the unit of Code Representing that Entity.

### **❖ How do you identify Classes in Class Diagram?**

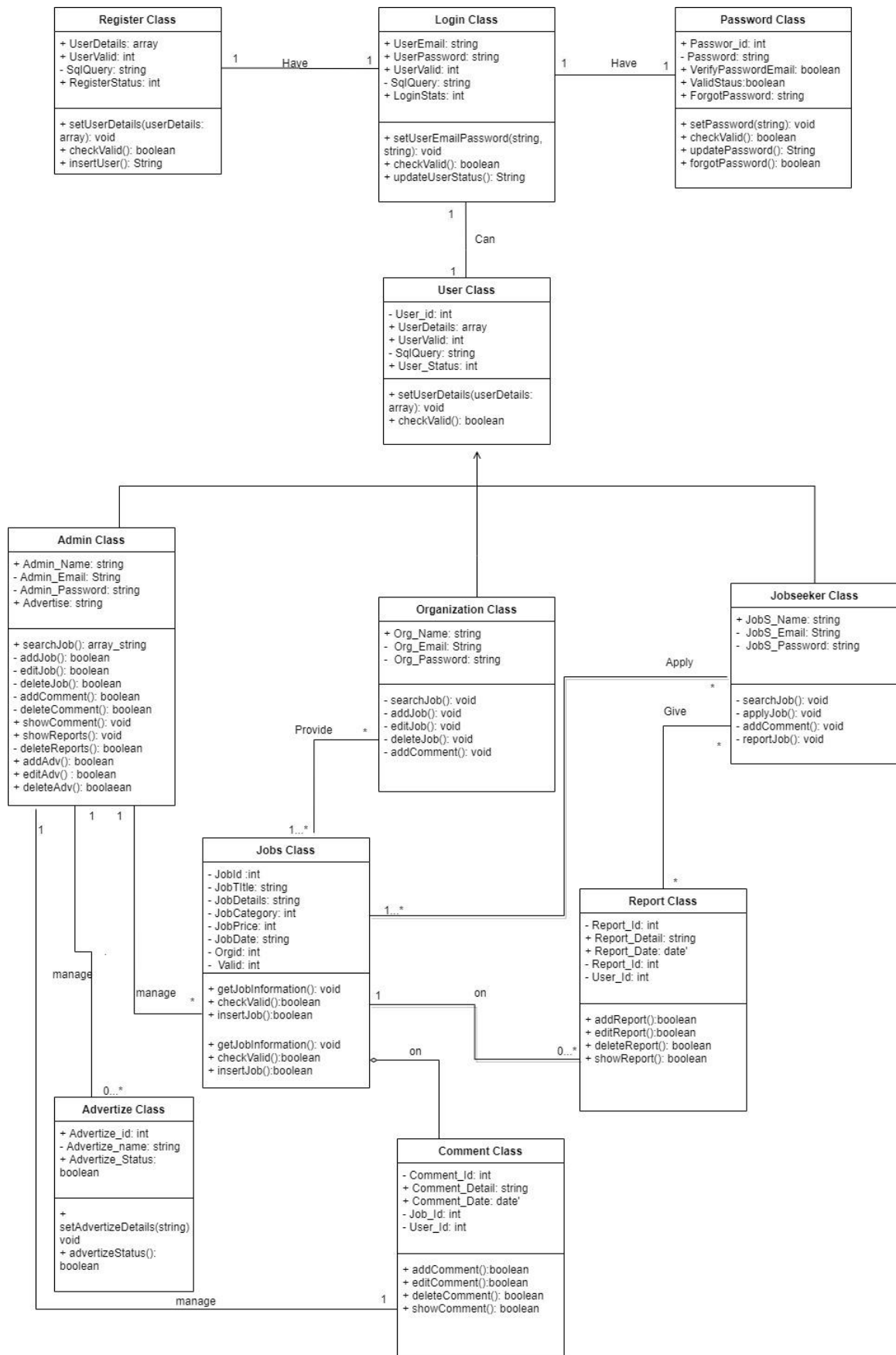
- Analysis Activities include.
- Identifying Objects (often from use cases as a starting point).
- Identifying Associations between Objects.
- Identifying General Attributes and Responsibilities of Objects.
- Modelling interactions between Objects.
- Modelling how individual Objects change state helps identify Operations.

### **❖ What are the Components of Class Diagram?**

In Class Diagrams as work with the following elements

- Class. A Class Represents a Relevant Concept from the Domain, a set of Persons, Objects or Ideas that are depicted in the IT System.
- Attribute.
- Generalization.
- Association.
- Multiplicity.
- Aggregation.

## Class Diagram



## **Data Dictionary**

## **Data Dictionary**

### **❖ What is Data Dictionary?**

- Data dictionary is a centralized repository of metadata. Metadata is data about.
- Data. Some examples of what might be contained in an organization's data dictionary include: The data types, e.g., integer, real, character, and image of all fields in the organization's databases.

### **❖ What are the benefits of Data Dictionary?**

- There are a number of advantages of using Data Dictionary in computer system analysis and design.
- The advantages are: consistency, clarity; reusability; completeness; increase in sharing and integration; and ease of use for the developer.

### **❖ Why do we need a Data Dictionary?**

- A successful data dictionary can improve the reliability and dependability of an organization's data, reduce redundancy, improve documentation and control, and make it easier to analyse data.
- Use it to make evidence-based care decisions like those common in accountable care organizations.



## Look Out Data Dictionary

### □ Tables List

- |                      |                   |
|----------------------|-------------------|
| 1. lo_tblUsers       | 5. lo_tblComments |
| 2. lo_tblProfileUser | 6. lo_tblReports  |
| 3. lo_tblCategory    | 7. lo_tblAds      |
| 4. lo_tblJobs        | 8. lo_tblOtp      |

#### 1. lo\_tblUsers:

FIELDS NAME	DATATYPE	SIZE	CONSTRAINTS	DESCRIPTION
Id	Medium int	10	<ul style="list-style-type: none"> <li>• P.K.,</li> <li>• Auto-increment,</li> <li>• Not null</li> </ul>	User id.
user_fname	Varchar	20	<ul style="list-style-type: none"> <li>• Not Null</li> </ul>	User First name.
user_lname	Varchar	20	<ul style="list-style-type: none"> <li>• Not null</li> </ul>	User Last name.
user_email	Varchar	150	<ul style="list-style-type: none"> <li>• U.K.,</li> <li>• Not null</li> </ul>	User Email.
user_contactNumber	Int	10	<ul style="list-style-type: none"> <li>• Not Null</li> </ul>	User Contact number
user_dob	timestamp		<ul style="list-style-type: none"> <li>• Not null</li> </ul>	User Date-of-birth
user_address	Varchar	100	<ul style="list-style-type: none"> <li>• Not Null</li> </ul>	User Address
user_state	Varchar	30	<ul style="list-style-type: none"> <li>• Not Null</li> </ul>	User State
user_city	Varchar	20	<ul style="list-style-type: none"> <li>• Not Null</li> </ul>	User city
user_country	Varchar	20	<ul style="list-style-type: none"> <li>• Not Null</li> </ul>	User country
user_photo	Varchar	200	<ul style="list-style-type: none"> <li>• Null</li> </ul>	User photo (optional)
user_password	Varchar	25	<ul style="list-style-type: none"> <li>• Not null</li> </ul>	User password
user_type	Enum ('A','O','J')	1	<ul style="list-style-type: none"> <li>• Default ('J')</li> <li>• Not null</li> </ul>	User type('Admin', 'Organization', 'jobseeker')
is_live	Enum ('Y','N')	1	<ul style="list-style-type: none"> <li>• Default ('N')</li> </ul>	User status('Yes', 'No')
is_deleted	Enum ('Y','N')	1	<ul style="list-style-type: none"> <li>• Default ('N')</li> </ul>	User suspended
date_created	Timestamp		<ul style="list-style-type: none"> <li>• Not null</li> </ul>	User Created date
date_updated	Timestamp		<ul style="list-style-type: none"> <li>• Not null,</li> <li>• On update Current Timestamp</li> </ul>	User lastly edited

## Lookout – Employee Recruitment System (Job Portal)

### 2. lo\_ tblProfileUser:

FIELDS NAME	DATATYPE	SIZE	CONSTRAINTS	DESCRIPTION
Id	Medium int	10	<ul style="list-style-type: none"><li>• P.K.,</li><li>• Auto-increment,</li><li>• Not null</li></ul>	Profile User id.
user_id	Medium int	10	<ul style="list-style-type: none"><li>• F.K</li><li>• Not Null</li></ul>	User id.
profile_userName	Varchar	20	<ul style="list-style-type: none"><li>• U.K.</li><li>• Not null</li></ul>	User name
jobS_resume	Varchar	255	<ul style="list-style-type: none"><li>• Not null</li></ul>	Job seeker resume file.
jobS_occupation	Varchar	50	<ul style="list-style-type: none"><li>• Null</li></ul>	JobSeeker occupation.
jobS_exp	Medium int	20	<ul style="list-style-type: none"><li>• Null</li></ul>	JobSeeker experience
org_name	Varchar	20	<ul style="list-style-type: none"><li>• NULL</li></ul>	Organization name
date_created	Timestamp		<ul style="list-style-type: none"><li>• Not null</li></ul>	Profile User Created date
date_updated	Timestamp		<ul style="list-style-type: none"><li>• Not null,</li><li>• On update Current Timestamp</li></ul>	Profile User lastly edited

### 3. lo\_ tblCategory:

FIELDS NAME	DATATYPE	SIZE	CONSTRAINTS	DESCRIPTION
id	Medium int	10	<ul style="list-style-type: none"><li>• P.K.,</li><li>• Auto-increment,</li><li>• Not null</li></ul>	Category id.
category_name	Varchar	20	<ul style="list-style-type: none"><li>• Not null</li></ul>	Category name
category_subname	Varchar	20	<ul style="list-style-type: none"><li>• Not null</li></ul>	Category Sub name
date_created	Timestamp		<ul style="list-style-type: none"><li>• Not null</li></ul>	Category Created date
date_updated	Timestamp		<ul style="list-style-type: none"><li>• Not null,</li><li>• On update Current Timestamp</li></ul>	Category lastly edited

## Lookout – Employee Recruitment System (Job Portal)

### 4. Lo\_ tblJobs

FIELDS NAME	DATATYPE	SIZE	CONSTRAINTS	DESCRIPTION
id	Medium int	10	<ul style="list-style-type: none"><li>• P.K.,</li><li>• Auto-increment,</li><li>• Not null</li></ul>	Job id.
user_id	Medium int	10	<ul style="list-style-type: none"><li>• F.K</li><li>• Not Null</li></ul>	Profile User id.
job_title	Varchar	200	<ul style="list-style-type: none"><li>• Not null</li></ul>	Job title.
job_desc	Varchar	255	<ul style="list-style-type: none"><li>• Not null</li></ul>	Job description.
job_amt	INT	10	<ul style="list-style-type: none"><li>• NOT null</li></ul>	Job price.
job_hours	Timestamp	5	<ul style="list-style-type: none"><li>• NOT Null</li></ul>	Job hours.
job_enddate	Timestamp	20	<ul style="list-style-type: none"><li>• Null</li></ul>	Job end date
job vacancy	Medium int	5	<ul style="list-style-type: none"><li>• Not NULL</li></ul>	Job vacancy
category_id	Medium int	10	<ul style="list-style-type: none"><li>• F.K.,</li><li>• Not Null</li></ul>	Job category
is_reported	Enum (‘Y’, ‘N’)	1	<ul style="list-style-type: none"><li>• Default (‘N’)</li></ul>	Job is reported
job_createdby	Enum (‘A’, ‘O’)	1	<ul style="list-style-type: none"><li>• Default Null</li></ul>	Who created this job
job_lastlyedited	Enum (‘A’, ‘O’)	1	<ul style="list-style-type: none"><li>• Default Null</li></ul>	Who lastly edited this job
date_created	Timestamp		<ul style="list-style-type: none"><li>• Not null</li></ul>	Job Created date
date_updated	Timestamp		<ul style="list-style-type: none"><li>• Not null,</li><li>• On update Current Timestamp</li></ul>	Job lastly edited

## Lookout – Employee Recruitment System (Job Portal)

### 5. Lo\_tblComments

<b>FIELDS NAME</b>	<b>DATE TYPE</b>	<b>SIZE</b>	<b>CONSTRAINTS</b>	<b>DESCRIPTION</b>
id	Medium int	10	<ul style="list-style-type: none"><li>• P.K.,</li><li>• Auto-increment,</li><li>• Not null</li></ul>	Comment id.
user_id	Medium int	10	<ul style="list-style-type: none"><li>• F.k.</li><li>• Not null</li></ul>	Comment Content
job_id	Medium int	10	<ul style="list-style-type: none"><li>• F.k.</li><li>• Not null</li></ul>	Comment Name
comment_desc	Varchar	255	<ul style="list-style-type: none"><li>• Not null</li></ul>	Comment description
comment_cretedby	Enum (‘A’,’O’,’J’)	1	<ul style="list-style-type: none"><li>• Not null</li></ul>	Comment done by Admin, organization, jobseeker
date_created	Timestamp		<ul style="list-style-type: none"><li>• Not null</li></ul>	Comment Created date
date_updated	Timestamp		<ul style="list-style-type: none"><li>• Not null,</li><li>• On update Current Timestamp</li></ul>	Comment lastly edited

## Lookout – Employee Recruitment System (Job Portal)

### 6. Lo\_tblReports

<b>FIELDS NAME</b>	<b>DATATYPE</b>	<b>SIZE</b>	<b>CONSTRAINTS</b>	<b>DESCRIPTION</b>
id	Medium int	10	<ul style="list-style-type: none"><li>• P.K.,</li><li>• Auto-increment,</li><li>• Not null</li></ul>	Job id.
user_id	Medium int	10	<ul style="list-style-type: none"><li>• F.K</li><li>• Not Null</li></ul>	Profile User id.
job_id	Medium Int	10	<ul style="list-style-type: none"><li>• Not null</li></ul>	Job id.
report_title	varchar	50	<ul style="list-style-type: none"><li>• Null</li></ul>	Report title
report_desc	Varchar	255	<ul style="list-style-type: none"><li>• NOT null</li></ul>	Report Description
date_created	Timestamp		<ul style="list-style-type: none"><li>• Not null</li></ul>	Report Created date
date_updated	Timestamp		<ul style="list-style-type: none"><li>• Not null,</li><li>• On update Current Timestamp</li></ul>	Report lastly edited

### 7. Lo\_tblAds

<b>FIELDS NAME</b>	<b>DATE TYPE</b>	<b>SIZE</b>	<b>CONSTRAINTS</b>	<b>DESCRIPTION</b>
id	Medium int	10	<ul style="list-style-type: none"><li>• P.K.,</li><li>• Auto-increment,</li><li>• Not null</li></ul>	Ads id.
ads_content	Varchar	255	<ul style="list-style-type: none"><li>• Not null</li></ul>	Ads Content
ads_title	Varchar	50	<ul style="list-style-type: none"><li>• Not null</li></ul>	Ads Name
date_created	Timestamp		<ul style="list-style-type: none"><li>• Not null</li></ul>	category Created date
date_updated	Timestamp		<ul style="list-style-type: none"><li>• Not null,</li><li>• On update Current Timestamp</li></ul>	Category lastly edited

## Lookout – Employee Recruitment System (Job Portal)

### 8. Lo\_tblOtp

FIELDS NAME	DATATYPE	SIZE	CONSTRAINTS	DESCRIPTION
id	Medium int	10	<ul style="list-style-type: none"><li>• P.K.,</li><li>• Auto-increment,</li><li>• Not null</li></ul>	Job id.
user_id	Medium int	10	<ul style="list-style-type: none"><li>• F.K</li><li>• Not Null</li></ul>	Profile User id.
verify_code	Tiny int	6	<ul style="list-style-type: none"><li>• Not null</li></ul>	Verify code.
verify_status	Enum(0,1)	1	<ul style="list-style-type: none"><li>• Default (0)</li></ul>	Verify Status.
is_verfy	Enum(0,1)	1	<ul style="list-style-type: none"><li>• Default (0)</li><li>• NOT null</li></ul>	Verified

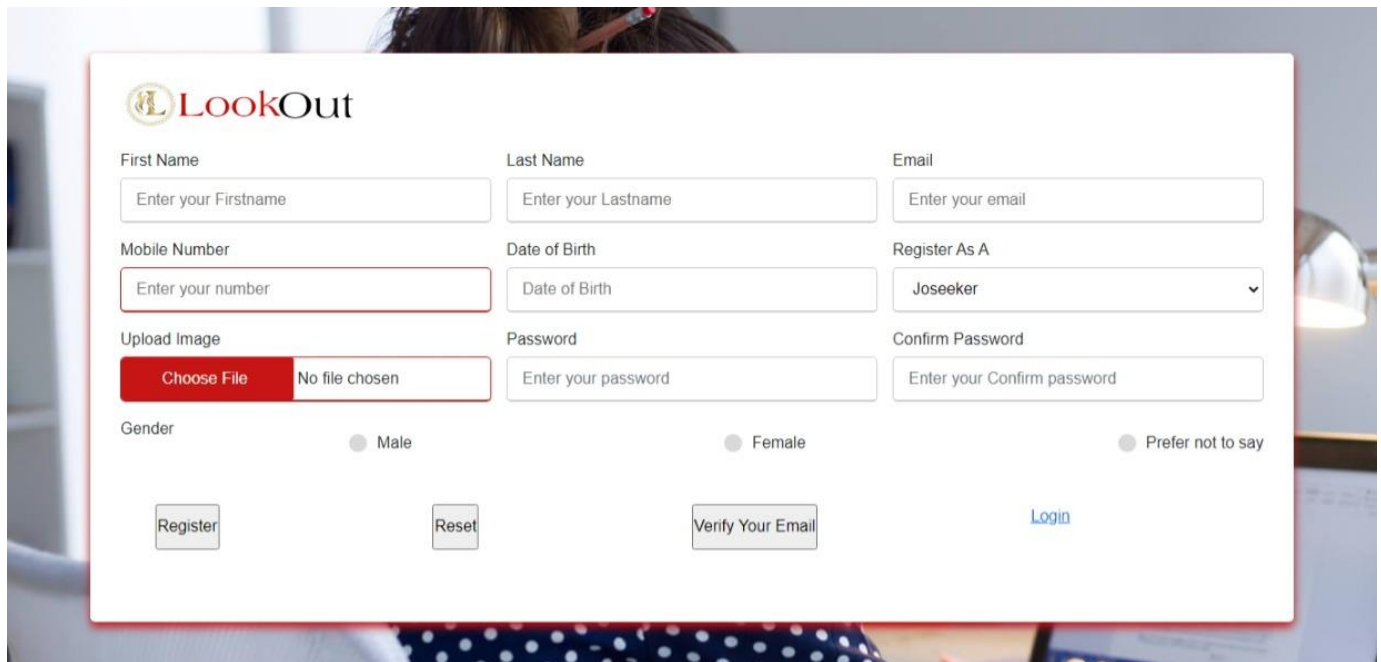
### Reference:

W3schools : <https://www.w3schools.com>

**Screen shot layout**

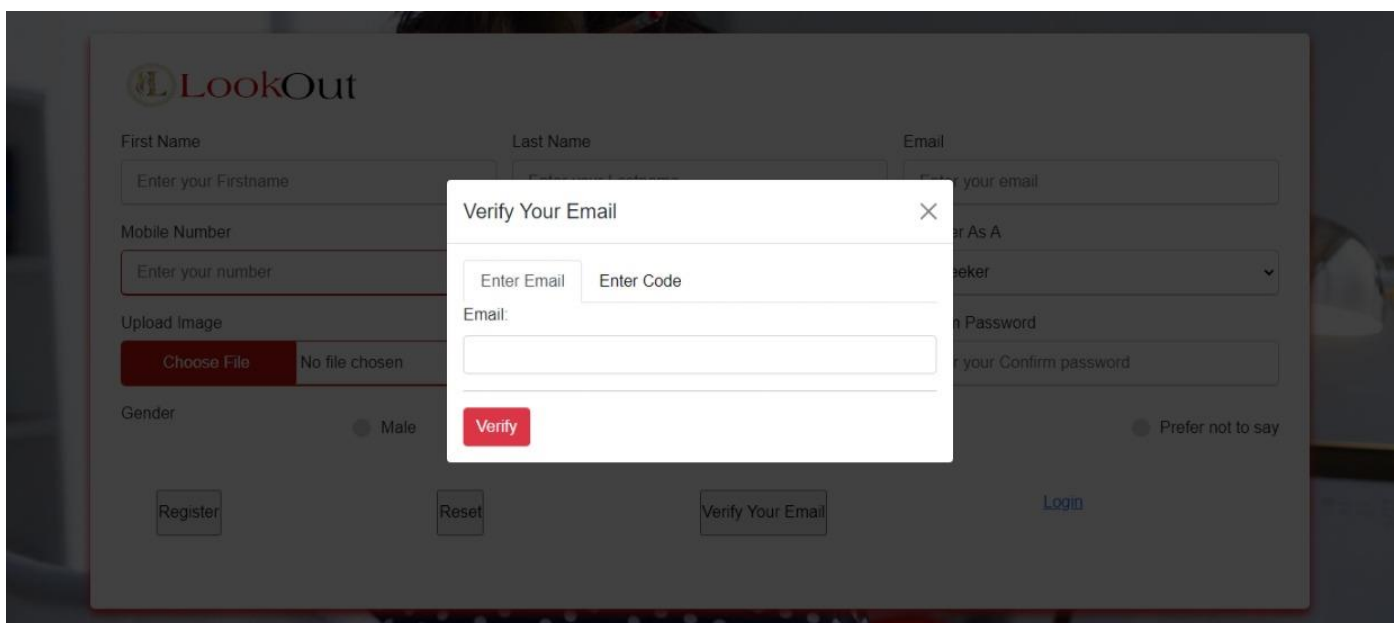
## Lookout – Employee Recruitment System (Job Portal)

### 2.1 Register Page:



The screenshot shows the 'LookOut' registration form. It includes fields for First Name, Last Name, Email, Mobile Number, Date of Birth, and a 'Register As A' dropdown menu set to 'Jobseeker'. There are also fields for Password and Confirm Password, and an 'Upload Image' section with a 'Choose File' button. Gender selection is available with radio buttons for Male, Female, and Prefer not to say. At the bottom, there are buttons for 'Register', 'Reset', 'Verify Your Email', and a 'Login' link.

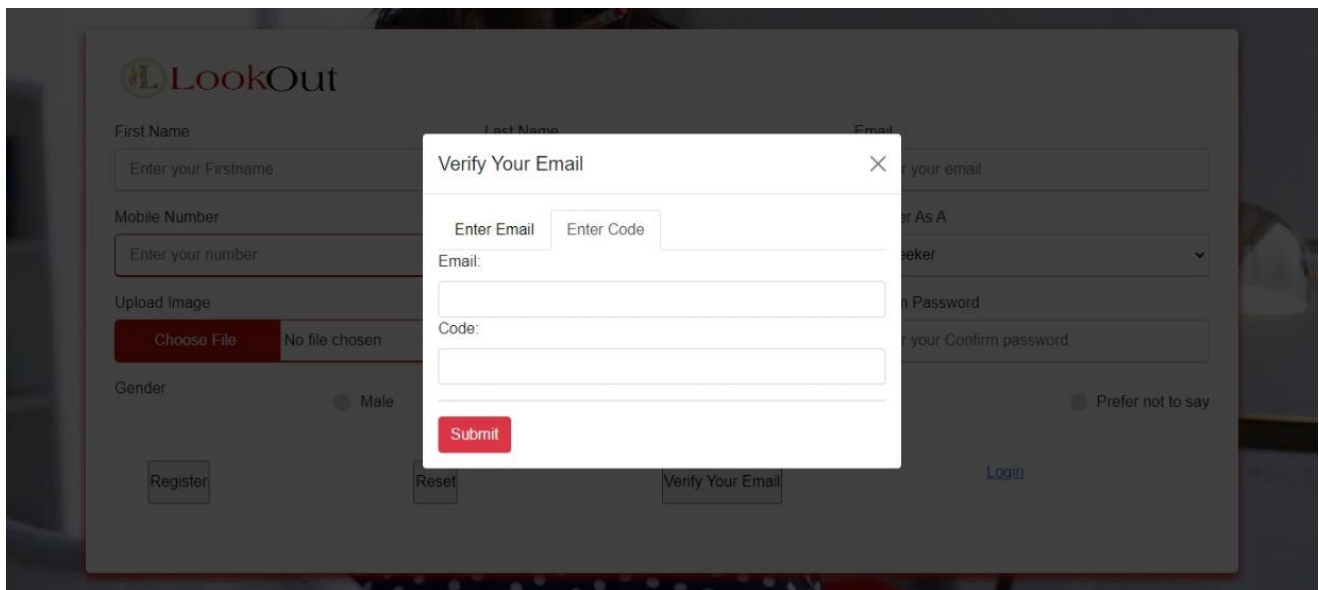
This Above are the Screenshot of the Register Page For Organization & Jobseeker they have to verify an email for successfully Register themselves



The screenshot shows the 'LookOut' registration form with a 'Verify Your Email' modal open. The modal has a close button (X) and contains fields for 'Enter Email' and 'Enter Code'. Below these fields is a 'Verify' button. The background form is dimmed.

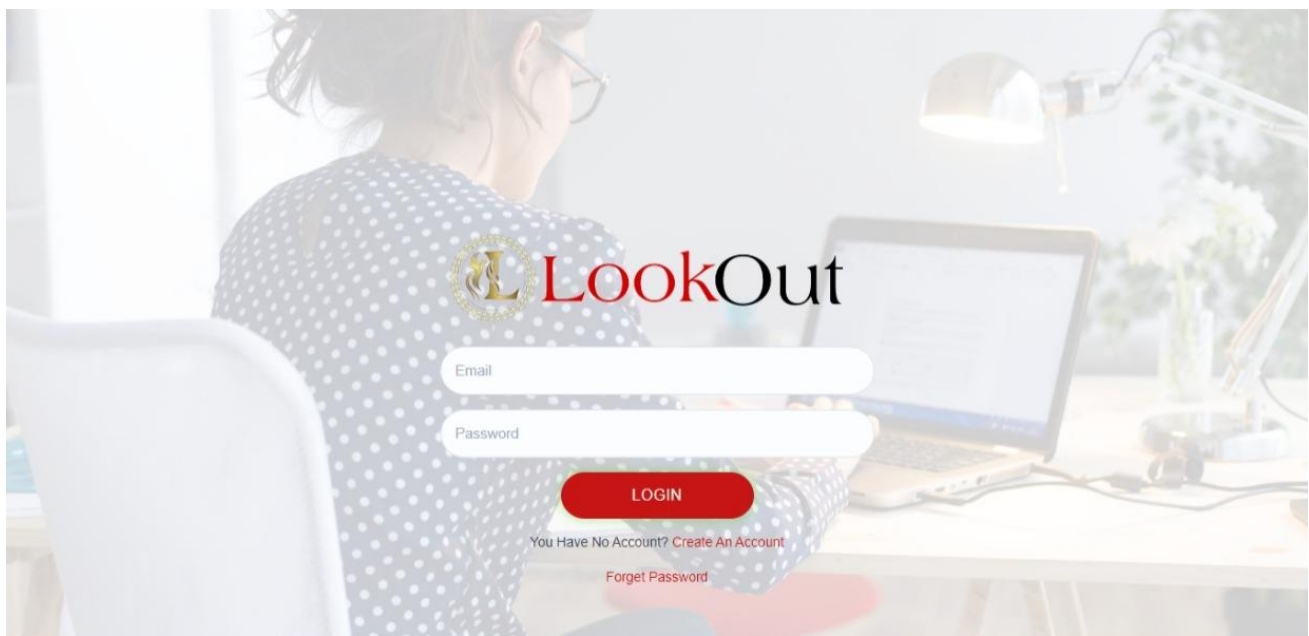


## Lookout – Employee Recruitment System (Job Portal)



These Above Are the Screenshots of the Register Email verification It will sent an 6-bit Verification code.

### 2.2 Login Page:



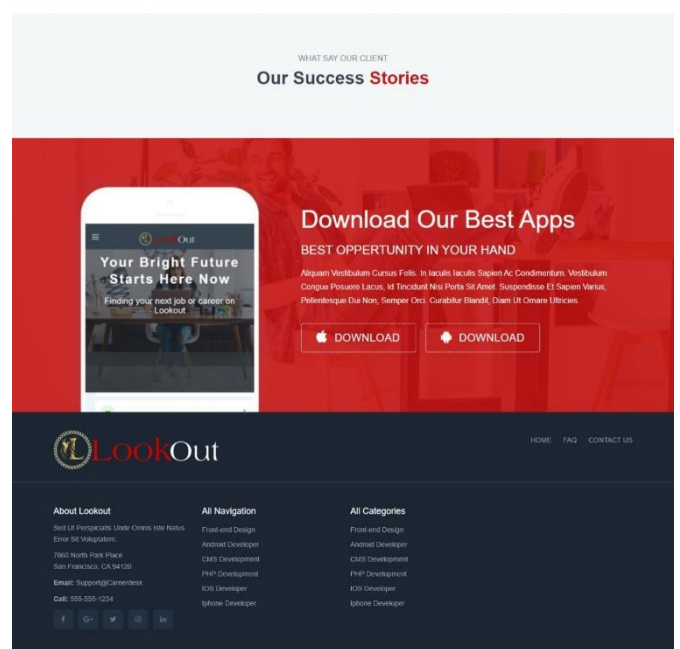
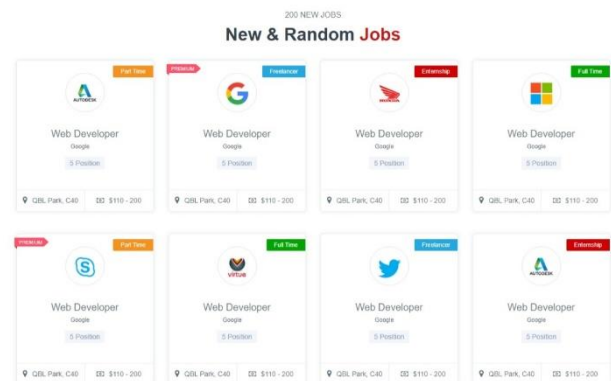
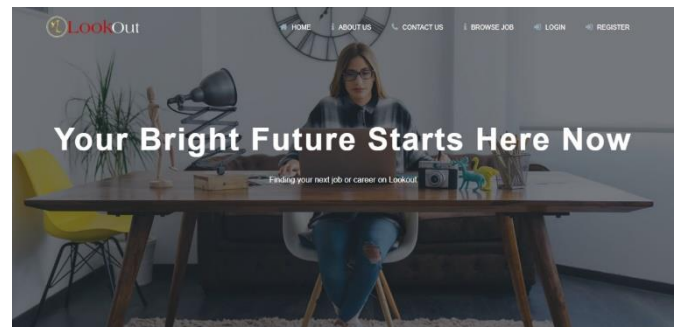
This Is Common Login Page For All users (Jobseeker, Organization, Admin) User have to Enter Registered Email & Valid Password

## Lookout – Employee Recruitment System (Job Portal)

### 2.3 Home Page:

This Above is Screenshot of Main Index page of Lookout Jobportal

We Tried To Make it Very Simple & Attractive With Navbar, Job Posts, Our Introductory Video and Footer.



### 2.4 Create JobPost Page:

**LookOut** DASHBOARD LOGOUT

# Create Job

**Job Information**

Job Title Location, e.g. London Quel Mark

Job Description

Full Time SELECT CATEGORY

Job Vacancy

Job Responsibilities

Skill Requirement

Minimum Experience Salary

**SUBMIT JOB**

**LookOut** HOME FAQ CONTACT US

**About Lookout**  
Sed Ut Perspiciatis Unde Omnis Iste Natus  
Error Sit Voluptatem.  
7860 North Park Place  
San Francisco, CA 94120  
Email: Support@Careerdesk  
Call: 555-555-1234

**All Navigation**  
Front-end Design  
Android Developer  
CMS Development  
PHP Development  
IOS Developer  
Iphone Developer

**All Categories**  
Front-end Design  
Android Developer  
CMS Development  
PHP Development  
IOS Developer  
Iphone Developer

This is our Post job Page, organization can post job from here.

## Lookout – Employee Recruitment System (Job Portal)

### 2.5 Job Details Preview From Organization Side:

The screenshot displays the LookOut Job Portal interface. At the top, the LookOut logo is on the left, and navigation links for DASHBOARD, POST JOB, and LOGOUT are on the right. The main header features a large image of a woman talking on a phone, with the text "Job Detail" overlaid. Below this, a white card contains the job details for a "CEO" position. The card includes a Google logo, the text "By: NULL", and job specifics: "Job Vacancy: 1", "Minimum Experience: 8 years", and "Job Category: WEB DEVELOPER". To the right of the card, a "Get in Touch" section provides contact information: a location pin, email "akkipandya2580@gmail.com", "Full Time" status, and a salary of "\$30000". Below the card, there are sections for "Description" (containing the text "THIS IS THE TESTING JOB") and "Comments" (with an "Add Comment" input field and an "Add" button). The footer of the page contains the LookOut logo, navigation links (HOME, FAQ, CONTACT US), and three columns of information: "About Lookout" (address, email, call), "All Navigation" (Front-end Design, Android Developer, CMS Development, PHP Development, IOS Developer, Iphone Developer), and "All Categories" (Front-end Design, Android Developer, CMS Development, PHP Development, IOS Developer, Iphone Developer). Social media icons for Facebook, Google+, Twitter, and LinkedIn are also present.

**LookOut** DASHBOARD POST JOB LOGOUT

# Job Detail

**CEO**  
By: NULL

Job Vacancy: 1 Minimum Experience: 8 years  
Job Category: WEB DEVELOPER

**Get in Touch**  
akkipandya2580@gmail.com  
Full Time  
\$30000

**Description**  
THIS IS THE TESTING JOB

**Comments**  
Add Comment  
0/200  
Add  
No Comments

**LookOut** HOME FAQ CONTACT US

**About Lookout**  
Sed Ut Perspiciatis Unde Omnis Iste Natus  
Error Sit Voluptatem.  
7860 North Park Place  
San Francisco, CA 94120  
Email: Support@Careersdesk  
Call: 555-555-1234

**All Navigation**  
Front-end Design  
Android Developer  
CMS Development  
PHP Development  
IOS Developer  
Iphone Developer

**All Categories**  
Front-end Design  
Android Developer  
CMS Development  
PHP Development  
IOS Developer  
Iphone Developer


This is our Detail View of Job, Users can view job details from here.

### 2.6 Organization Dashboard:

The screenshot displays the 'Organization Profile' page for Google on the Lookout recruitment system. The header features the Lookout logo and navigation links: DASHBOARD, POST JOB, CONTACT US, and LOGOUT. The main content area shows the Google logo and the name 'GOOGLE'. Below this, two statistics are presented: '1 Job Post' and '0 Job Approved'. A horizontal menu includes options like MY INFO, JOB POSTED (highlighted), JOB REQUEST, EDIT PROFILE, CHANGE PASSWORD, and SOCIAL MEDIA. The 'JOB POSTED' section shows a new job listing for 'CEO' with a Google logo, indicating 1 vacancy and a creation date of 03/23/2022. The category is listed as 'Full Time'. The footer contains the Lookout logo, contact information, and three columns of links: 'About Lookout', 'All Navigation', and 'All Categories'.

**LookOut** DASHBOARD POST JOB CONTACT US LOGOUT

# Organization Profile




GOOGLE

**1**  
Job Post

**0**  
Job Approved

MY INFO **JOB POSTED** JOB REQUEST EDIT PROFILE CHANGE PASSWORD SOCIAL MEDIA

**NEW**  **CEO** Category: Full Time  
Vacancy: 1  
Created On: 03/23/2022

**LookOut** HOME FAQ CONTACT US

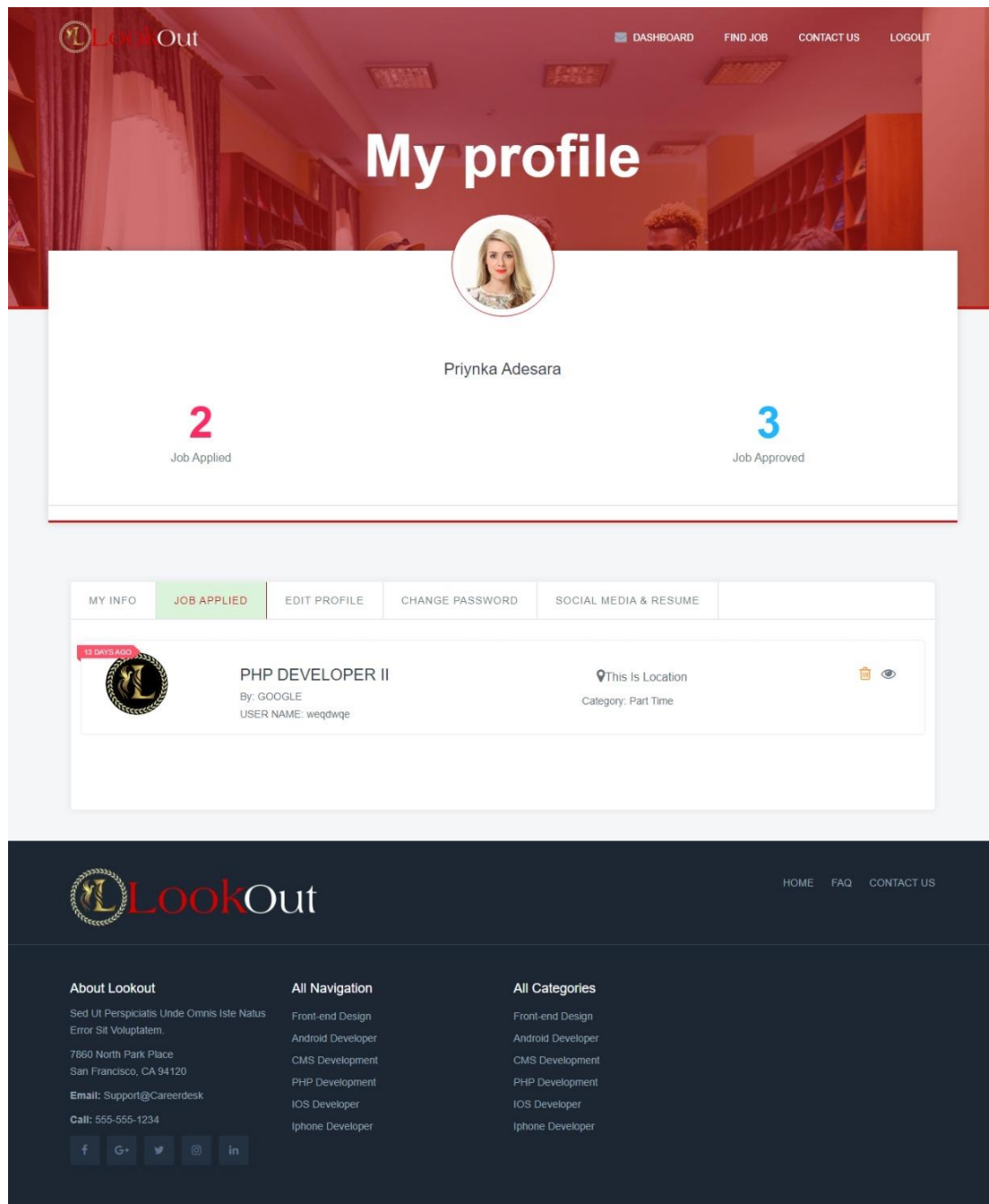
**About Lookout**  
Sed Ut Perspiciatis Unde Omnis Iste Natus  
Error Sit Voluptatem.  
7860 North Park Place  
San Francisco, CA 94120  
Email: Support@Careerdesk  
Call: 555-555-1234

**All Navigation**  
Front-end Design  
Android Developer  
CMS Development  
PHP Development  
IOS Developer  
Iphone Developer

**All Categories**  
Front-end Design  
Android Developer  
CMS Development  
PHP Development  
IOS Developer  
Iphone Developer

## Lookout – Employee Recruitment System (Job Portal)

### 2.7 Jobseeker Dashboard:



This is Jobseeker UI , jobseeker can perform their activity from here.

- Jobseeker manage profile
- Find & Apply Job Job.
- Upload Resume

## Lookout – Employee Recruitment System (Job Portal)

### 2.8 Admin Dashboard:

The screenshot displays the 'Lookout' Admin Dashboard. The left sidebar contains navigation links: Main, Dashboard, Organization (selected), Jobseeker, Post, Reports, Banned Accounts, and Logout. The main content area is titled 'Organization' and includes a search bar with a 'Go' button and a 'Short By' dropdown menu. A list of organizations is shown, each with a logo, name, type, location, and a '10 Hour Ago' timestamp. Each entry also has a trash icon for deletion.

Logo	Organization Name	Type	Location	Timestamp	Action
	Autodesk	(Software Company)	Street #210, Make New London	10 Hour Ago	
	Google	(Software Company)	Street #210, Make New London	10 Hour Ago	
	Honda	(Motor Agency)	Street #210, Make New London	10 Hour Ago	
	Microsoft	(Software Company)	Street #210, Make New London	10 Hour Ago	
	Skype	(Software Company)	Street #210, Make New London	10 Hour Ago	
	Virtue	(Development Company)	Street #210, Make New London	10 Hour Ago	
	Twitter	(Social Media Company)	Street #210, Make New London	10 Hour Ago	
	Twitter	(Social Media Company)	Street #210, Make New London	10 Hour Ago	
	Twitter	(Social Media Company)	Street #210, Make New London	10 Hour Ago	
	Twitter	(Social Media Company)	Street #210, Make New London	10 Hour Ago	

Showing 1 to 10 of 14 entries

Previous 1 2 3 4 5 6 Next

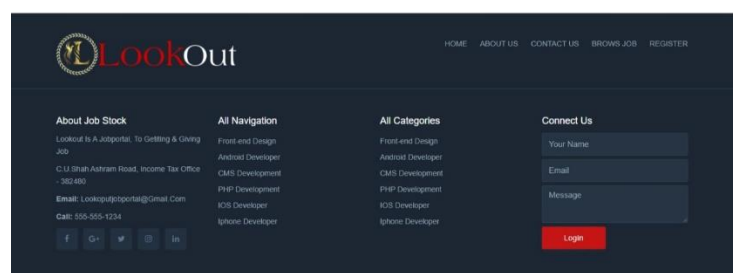
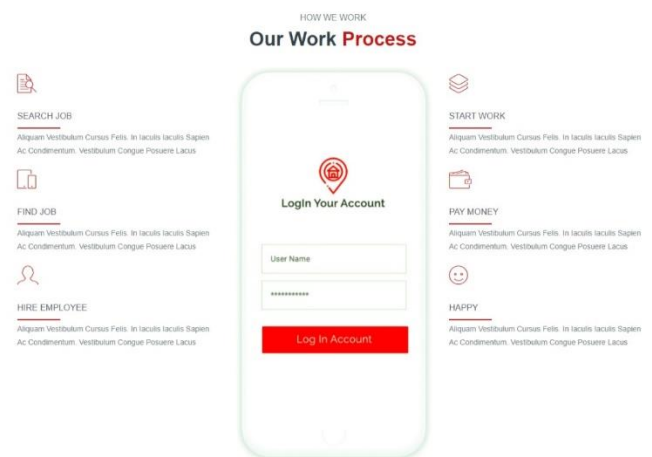
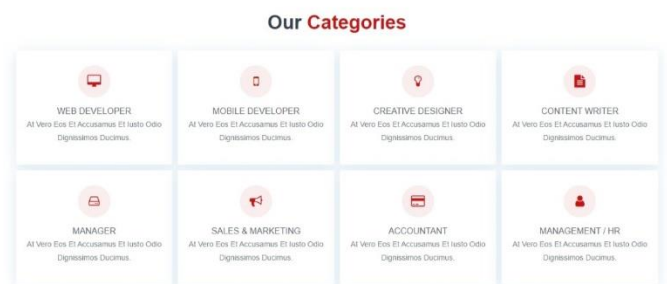
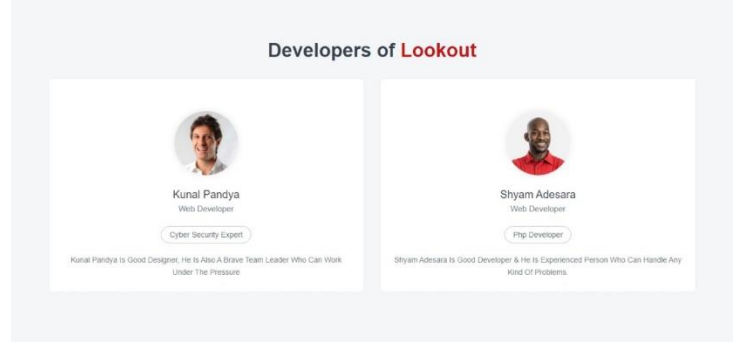
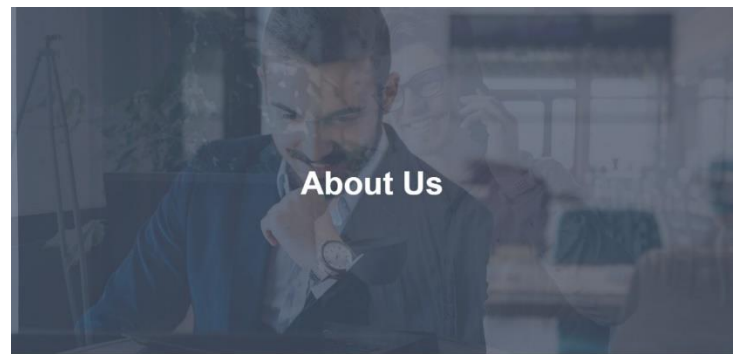


## Lookout – Employee Recruitment System (Job Portal)

### 2.9 About us:

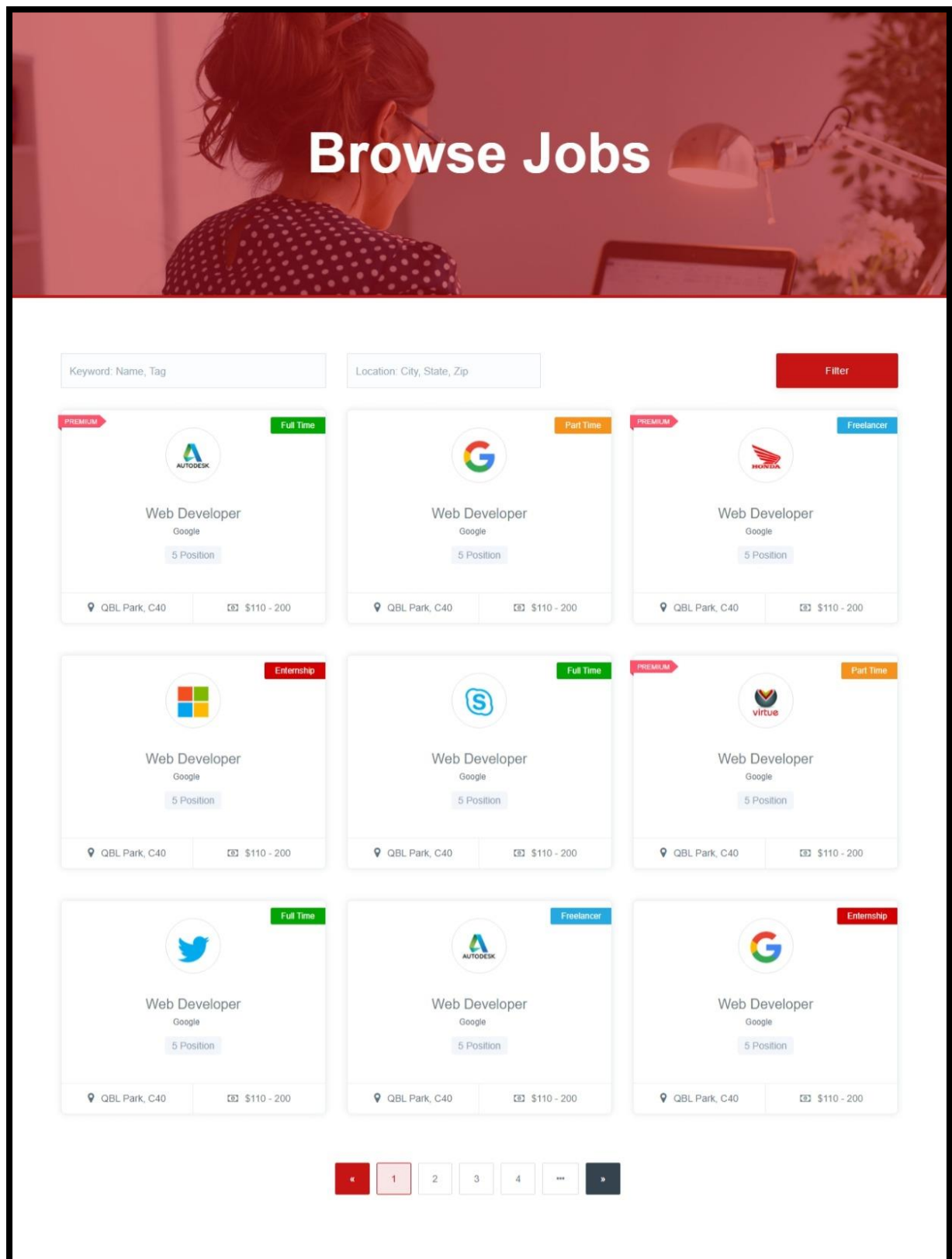
This Above is Screenshot of about us Page.

This page conduct information about Developers, Categories, Our working Process.





### 2.10 Browse Jobs:



## **Report Layout**

### **3.1 Product Functions**

This Employee Recruitment System project will have different modules.

The login section will have login facility also for the admin who will operate this system and online Managing system of Posting Job and Applying Job for Jobportal.

#### **➔ Jobseeker role**

it will take all the details of its Organization who is Posting the Job and all the details for the recipient such as its Address, Name, Email, Mobile number.

#### **➔ Organization role**

it will take all the details of its Jobseeker who is Applying for the Job and all the details for the recipient such as its Address, Name, Email, Mobile number.

#### **➔ Administration role**

Managing the Accounts, Posts & Getting Reports Day-by-Day

### **3.2. User Characteristics**

#### **End Users**

All specific knowledge or skills are required from the feeder.

Educational level: Users should be comfortable with the English language.

Experience: Users should have prior information regarding the online job posting

Skills: Users should have basic knowledge and should be comfortable.

#### **Administrator**

Administrator must be capable to manage user rights This system will not take care of any virus problem which might occur either on the Client or the server system Avoiding the use of pirated software and ensuring that floppies and other removable media are scanned for viruses before use could minimize the possibility of viral infection.

### **3.3. Constraints**

The Information of all users, subjects and allocations must be stored in a database that is accessible by every connected system. MySQL used for database.

Users may access from any system connected to the online database.

Users must have their correct use names and passwords to enter into there accounts.

### **3.4 System Study**

The system was tested and the performance of the system was found to be acceptable. All the necessary output was created. The system was found to be user-friendly with a help message for the Organization & Jobseeker. System was developed to the Post a Job and Apply for Job .

### **3.5. Assumptions and Dependencies**

The Software needs the following third-party products. PHP code for development of project. Xampp for database connectivity Although basic password authentication and role-based security mechanisms will be used to protect OPMS from unauthorized access functionality such as email notifications are assumed to be sufficiently protected under the existing security policies applied by the University network team Redundant Database is setup as the role of backup Database Server when primary database & future.

### **3.6 Technical Feasibilities**

Before beginning a project, there is a crucial decision that must be made: Is the proposal technically feasible? That is, will the technology actually work?

The answer of the above question is: “This project is technically feasible. It successfully satisfies the users’ basic requirements. The tools and application software’s used in this project are very popular and easily available across the world.”

All of these factors of technical, technological and general feasibility are taken into account in this project as properly as possible. These factors are discussed in detail in the following chapter, Courier management system.

### **3.7. Operational Feasibility**

This system will have easy to understand interface for different modules. It does not require any programming skill to use the system. After a little training, the users will be able to work with it at ease.

### **3.8 Economic Feasibility**

A system can be developed technically and that will be used if installed must still be a good investment for the organization. In the economic feasibility, the development cost in creating the system is evaluated against the ultimate benefit derived from the new systems. Financial benefits must equal or exceed the costs.

### **3.9 interfaces**

In computing an interface is a shared boundary across which three separate components of computer system exchange information

#### **User interfaces:**

The application will have a user friendly and menu-based interface.

### **3.10 Hardware Requirements**

Intel I-3 2.4 GHz Processor and Above

RAM 4 GB and Above

HDD 20GB harddisk space and above.

### **3.11 Software Requirements**

WINDOWS OS (Windows 7, 8,10) Or Linux

Visual studios code IDE Is been used for code.

MYSQL database is been used.

### **3.12 System Analysis**

System Analysis is a detailed study of the various operations performed by a system and their relationships within and outside of the system.

Here the key question is- what all problems exist in the present system?

What must be done to solve the problem? Analysis begins when a user or manager begins a study of the program using existing system.

## **4.1 software Requirement Specification (SRS)**

The software. Site Explorer is designed for management of website from a remote location This section provides software requirements to a level of detail sufficient to enable designers to design the system a tester to test the system.

This section contains all of the functional and quality requirements of the system. It gives a detailed description of the system and all as features.

## **4.2. Communication Interfaces**

The HTTP protocol will be used to facilitate communication between the client server. The system supports Google Chrome and Mozilla Firefox web browsers.

### **4.3. Memory Constraints**

Minimum memory of 512MB is required to run the exe file without any lags. This constraint does not possess an issue now a days as the minimum present RAM in a common system is 1GB. At least 512 MB RAM and 5 MB space on hard disk will be required for running the program.

### **4.4. Operations**

The normal and special operations required by the user such as:

- >The various modes of operations in the user organization.
- >Periods of interactive operations and periods of unattended operations.
- >Data processing support functions.
- >Backup and recovery operations.

### **4.5. Site adaptation requirements**

There should be no site adaptation requirement since the web application server was setup.



## **Future Enhancement**

In future we will add online payment system access for easily payment because in this time of customers now more paying digital cash but unfortunately, we will add soon.

Our client want to include the use of modern technological gadgets to easily process and save time in the in the same so we try to implement that to.

## **Conclusion**

The main objective of the project was to develop a website to make courier services home and more flexible. And customers get the most out of the least effort they make.

## **References**

[www.google.com](http://www.google.com)

[www.w3schools.com/bootstrap/](http://www.w3schools.com/bootstrap/)

[www.w3schools.com/php/](http://www.w3schools.com/php/)