**SOFTWARE PERSONNEL MANAGEMENT SYSTEM**

**PROBLEM STATEMENT:**

The Software Personnel Management System keeps data of all the employees working in the company, record their working hours, salary to be paid and the tasks operated by them. This system allows employees to edit their personal details, know the number of hours worked per day or total of all hours spent on a project and the pay received year-to-date etc. Adding, deleting and generating the performance record of the employees is also done.

**INITIAL STATE:**

**Manually maintain all records:**

All the details of the employees are maintained manually . This type of approach costs more since they have to be placed in a safe place taking care of the possible harms.

**Time Consuming Search :**

Since the records are maintained manually, one has to go through all the records to find required data. As the number of employees increases, it becomes more complicated to find information.

**Long procedures:**

Editing and deleting information of an employee involves long procedure. The candidate must first request, submit all the required proofs and wait for the acceptance.

**IDEAL STATE:**

**Easily Editable Data:**

The employees can edit their personal Information without having to go through the procedural hinderence. All the required proofs can be scanned and attached to the file for easy validation .

**Time Saving Search:**

One doesn't need to go through all the records to find the required data. Only the name of candidate and related details must be provided to search for the record.

**Transparency:**

Employees can get all the relevant data of the company within seconds without having to go to the company office and requesting for seeking information. This is efficient and avoids inaccurate information.

**Automation:**

Software personnel management system allows employees to record time card electronically and automatically generates pay slips based on number of hours worked and total amount of sales.

**SOFTWARE REQUIREMENT SPECIFICATIONS:**

1. **INTRODUCTION:**

The main objective of designing and developing a Software Personnel Management system is to provide an interface to store and access the employee details effectively. It also aims at improving the efficiency in the generation of Pay slip.

**PURPOSE:**

If the entire process of Software personnel management is done in a manual manner then it becomes cumbersome to maintain records. Also, the entire process of generating pay slips becomes time consuming. Considering the fact that the number of employee is increasing every year, a maintenance system is essential to meet the demand. So this system uses several programming and database techniques to elucidate the work involved in this process.

**SCOPE:**

* Software personnel management system allows effective management of employee details.
* This assures time saving search to find out background of an employee and his/her contribution to the organization i.e all the information about an employee will be available within seconds.
* It makes it easy to generate statistical data or custom data, line finding a certain set of employee.

**DEFINITIONS, ACRONYMS AND SYNONYMS:**

* **SPMS**

Refers to Software personnel management system.

* **ADMINISTRATOR / MANAGER**

One who administers / manages employee details

* **EMPLOYEE**

An Individual who provides labor to the company.

* **HTML**

Hyper Text Markup Language- Markup Language used for creating web pages.

* **J2EE**

Java 2 Enterprise Edition is a programming platform java platform for developing and running distributed java applications.

* **HTTP**

Hyper Text Transfer Protocol- Protocol used by the World Wide Web that defines how messages are formatted and transmitted, and what actions Web servers and browsers should take in response to various commands

1. **OVERALL DESCRIPTION:**

* **PRODUCT PERSPECTIVE:**

The Software Personnel Management system is an Interface between Employee and the Administrator responsible for generation of payment slip. It aims at improving the efficiency in the generation of Pay slip and reduces the complexities involved in it to the maximum possible extent.

* **User Interface :**
* The Software uses GUI.
* It has fields where the employees can enter their personal/professional information like name , login id , address , qualifications , post etc.
* It has buttons to ‘login’ , ‘register’ , ‘edit information’ , ‘check details’ , ‘generate pay slip’.
* **Software Interface:**
* This software interacts with the **OS.** It is compatible with Windows 7 and higher versions, UNIX, LINUX, MAC, Ubuntu.
* **Hardware Interface:**
* This software uses ***basic input/output peripherals*** (keyboard/mouse/touchscreen/joystick).
* It also uses the***Internet***.
* This software interacts with the ***Printer*** if the user intends to print pay slip/form/application.
* This software interacts with the***Scanner*** if the user intends to scan the documents as specified by the corresponding company.
* **Product functions**
* Registration
* Application
* Editing details
* Checking work hours
* Generating payslip
* **SOFTWARE REQUIREMENTS:**

* The webpages in the software are developed using HTML, CSS, PHP and JavaScript.
* Database management is achieved using SQL
* **HARDWARE REQUIREMENTS:**
* Basic peripherals like keyboard , mouse , printer (if payslip is to be printed) , scanner (if documents are  to be scanned) are needed.
* WLAN/Internet connection is mandatory.
* **FUNCTIONAL REQUIREMENTS:**
* **Registration:**
* Every employee in the company must register itself with the software providing all the required information and scanned documents.
* Since the software is secure and reliable , one need not worry about the misuse of the data.
* **Log in:**
* The employees once registered can log in with the software using their respective employee id and password.
* It also lets them change their details ( only with valid proofs ).
* The Manager can also log in using his manager id and can check details of any employee.
* **Checking Details:**
* An Employee can keep track of the hours spent per day on work or any project.
* Manager/Employee can check and verify project related information.
* The manager can also keep eye on the employees without having to personally watch them.

* **Generating pay slip:**
* Based on the number of hours worked by an employee or the number of projects accomplished, their salary is calculated.
* A mail is sent to all employees.
* They can also directly verify it using their id.
* **NON-FUNCTIONAL REQUIREMENTS:**

1. **USABILITY -** The software is easy to use without any particular manual or support.
2. **PORTABLE -** It can execute properly on multiple hardware platforms.
3. **SECURE –** Information is securely transferred to the server without any changes.
4. **RELIABILITY –** It assures no misuse of data and the rate of failure of this software is very low.
5. **USER CHARACTERISTICS:**

* **MANAGER MODULE:**

The Manager can:

* Check Employee details
* Trace project success
* Change project information
* Update Company details
* Verify personnel details
* Take action against any complains
* **BANK MODULE:**
* Bank needs to issue cheques of requisite amount to every employee.
* Employee’s account details are saved securely by bank.
* **EMPLOYEE MODULE:**

An Employee can:

* Register and log in
* Check their working hours
* Report complains
* Edit their personal details
* Update Information about their project
* Generate payslip

1. **ASSUMPTIONS:**

* System is virus and other malware free.
* The user knows the basic operations of the computer.
* The corresponding examination management supports the functionalities provided by the application.

**DESIGN DOCUMENTATION:**

**TITLE:** SOFTWARE PERSONNEL MANAGEMENT SYSTEM

**INTRODUCTION:**

The main objective of designing and developing a Software Personnel Management system is to provide an interface to store and access the employee details effectively. It aims at improving the efficiency in the generation of Pay slip.

This software manages data of all the employees working in the company, record their working hours, salary to be paid and the tasks operated by them. Adding, deleting and generating the performance record of the employees is also done.

**PERSON’S INVOLVED:**

* **MANAGER :**

The Manager can:

* Check Employee details
* Trace project success
* Change project information
* Update Company details
* Verify personnel details
* Take action against any complains
* **BANK :**
* Bank needs to issue cheques of requisite amount to every employee.
* Employee’s account details are saved securely by bank.
* **EMPLOYEE :**

An Employee can:

* Register and log in
* Check their working hours
* Report complains
* Edit their personal details
* Update Information about their project
* Generate payslip

**PROBLEM:**

Initially maintaining the software personnel records caused the following problems:

1. **Manually maintain all records:**

All the details of the employees are maintained manually. This type of approach costs more since they have to be placed in a safe place taking care of the possible harms.

1. **Time Consuming Search :**

Since the records are maintained manually, one has to go through all the records to find required data. As the number of employees increases, it becomes more complicated to find information.

1. **Long procedures:**

Editing and deleting information of an employee involves long procedure. The candidate must first request, submit all the required proofs and wait for the acceptance.

1. **Difficult to keep track of employee work:**

As the number of employees in most of the companies are not usually low, keeping supervision on every member’s work is difficult. It is hard to maintain records of every employee’s daily task manually.

1. **Lengthy procedure to generate pay slip:**

As mentioned above , observing every member’s work is difficult. This results in an automatic procedure delay of calculating the pay. Moreover , then every employee should be personally informed about their salary , increments and cuts ( if any ) and be given the cheque.

**EXPECTED SOLUTION:**

The above problems of maintaining the records manually can be solved using an automated software, wherein every detail is fetched in the software. The Software Personnel Management system is an Interface between Employee and the Administrator responsible for effective management of employee details.

Either the employee or the management registers with the software. Daily updates about the presence of staff, hours worked by him/her, or any development in the project are loaded. Based on the data given and criteria set, the software automatically calculates the pay.

Hence It aims at improving the efficiency in the generation of Pay slip and reduces the complexities involved in it to the maximum possible extent. It also assures time saving search to find out background of an employee and his/her contribution to the organization i.e all the information about an employee will be available within seconds.

 Therefore, It makes it easy to generate statistical data or custom data, line finding a certain set of employee.

Following are the sections where this management system helped in improvement:

1. **Easily Editable Data:**

The software provides the functionality for employees to edit their personal Information without having to go through the procedural hinderence. All the required proofs can be scanned and attached to the file for easy validation.

1. **Time Saving Search:**

One doesn't need to go through all the records to find the required data. Only the name of candidate and related details must be provided to search for the record.

1. **Transparency:**

Employees can get all the relevant data of the company within seconds without having to go to the company office and requesting for seeking information. This is efficient and avoids inaccurate information.

1. **Traceable Employee Work:**

The manager can easily verify the work done by each individual employee. The software doesn’t provide this feature to the employees.

1. **Automation:**

Software personnel management system allows employees to record time card electronically and automatically generates pay slips based on number of hours worked and total amount of sales.

**TABLE:**

|  |  |  |
| --- | --- | --- |
| TASK | STARTING DATE | COMPLETION  DATE |
| Developing the home page with all the desired options | Jan 11,2019 | Jan 12,2019 |
| Developing web page for registration of employees | Jan 13, 2019 | Jan 15,2019 |
| Adding additional Registration fields for the manager | Jan 16,2019 | Jan 16,2019 |
| Designing a log in page | Jan 17,2019 | Jan 17,2019 |
| Writing code for categorizing and sorting employees | Jan 18,2019 | Jan 21,2019 |
| Pictorial representation of information | Jan 22,2019 | Jan 23,2019 |
| Code for generating the pay | Jan 24,2019 | Jan 26,2019 |
| Linking code with  the webpages | Jan 27,2019 | Jan 28,2019 |
| Providing temporary dataset | Jan 29,2019 | Jan 29,2019 |
| Testing and Improvements | Jan 29,2019 | Feb 2,2019 |
| Handing over the software to the company | Feb 3,2019 | - |
| Updating | Whenever required | - |

**TOOLS USED:**

* Rational rose

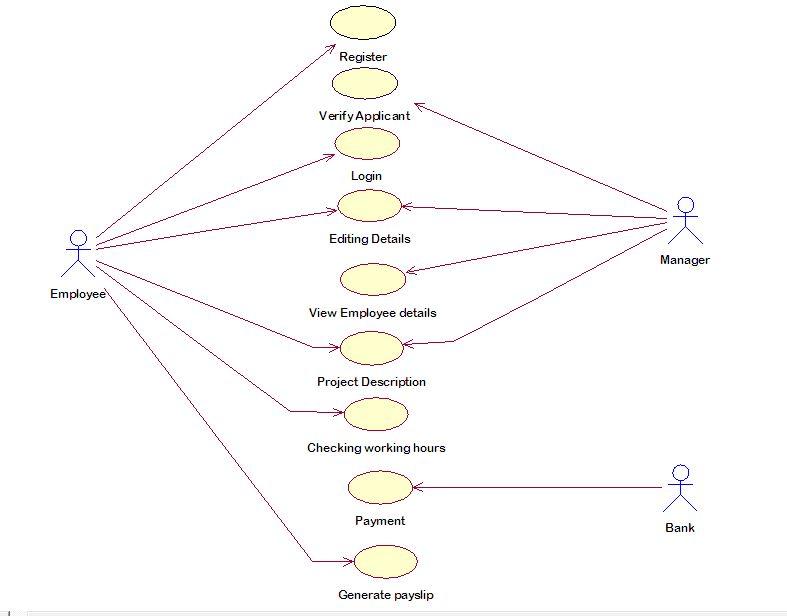
**DIAGRAMS:**

**1)USECASE DIAGRAM:**

* In the Unified Modeling Language (UML), a use case diagram can summarize the details of your system's users (also known as actors) and their interactions with the system.
* An effective use case diagram represents

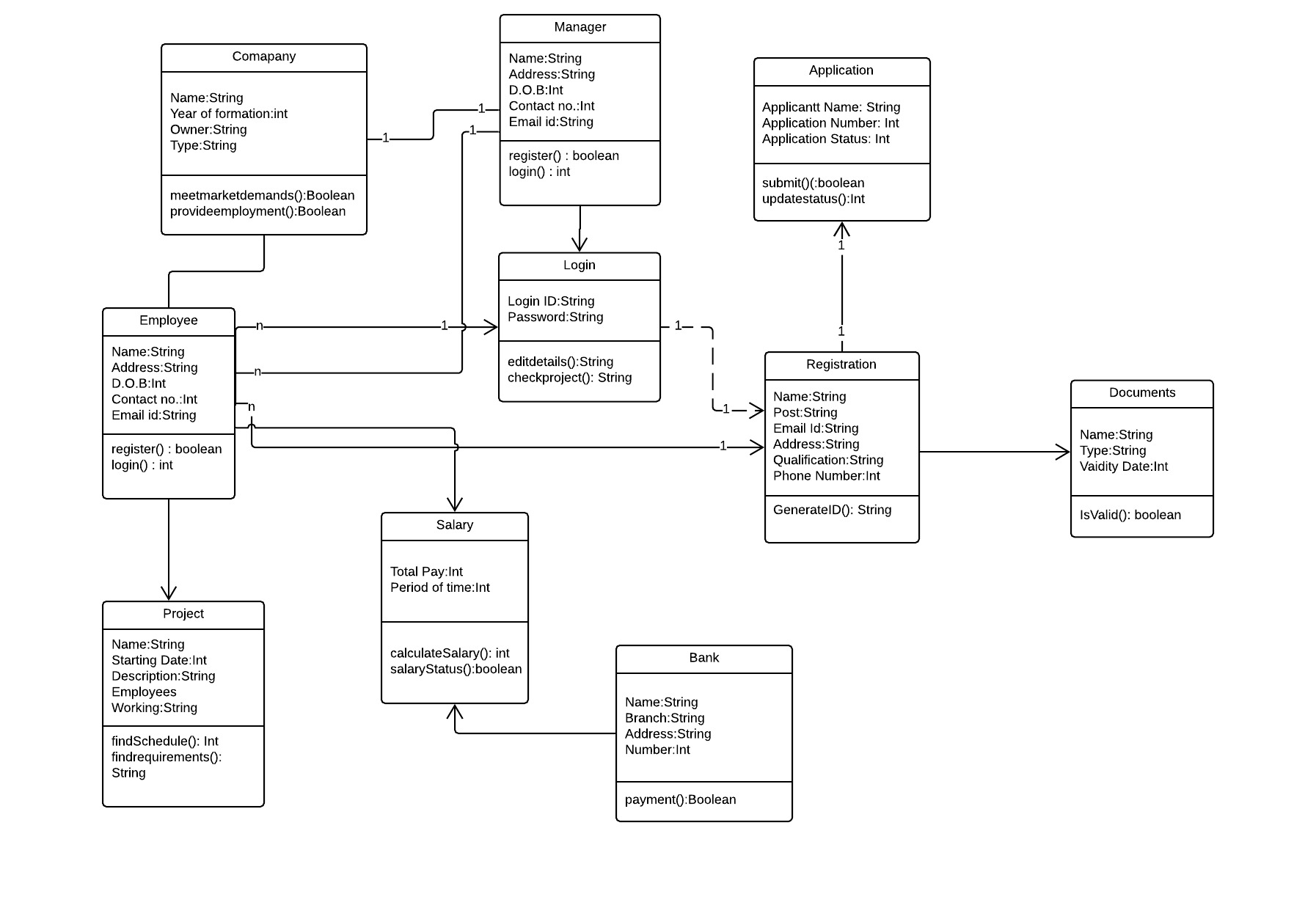
scenarios in which your system or application interacts with people, organizations, or external systems.

* Basic notations used in use case diagram:
* **Actors:** The users that interact with a system. An actor can be a person, an organization, or an outside system that interacts with your application or system. They must be external objects that produce or consume data.
* **System:** A specific sequence of actions and interactions between actors and the system. A system may also be referred to as a scenario.
* **Use Case** : They represent the system's functions.

****

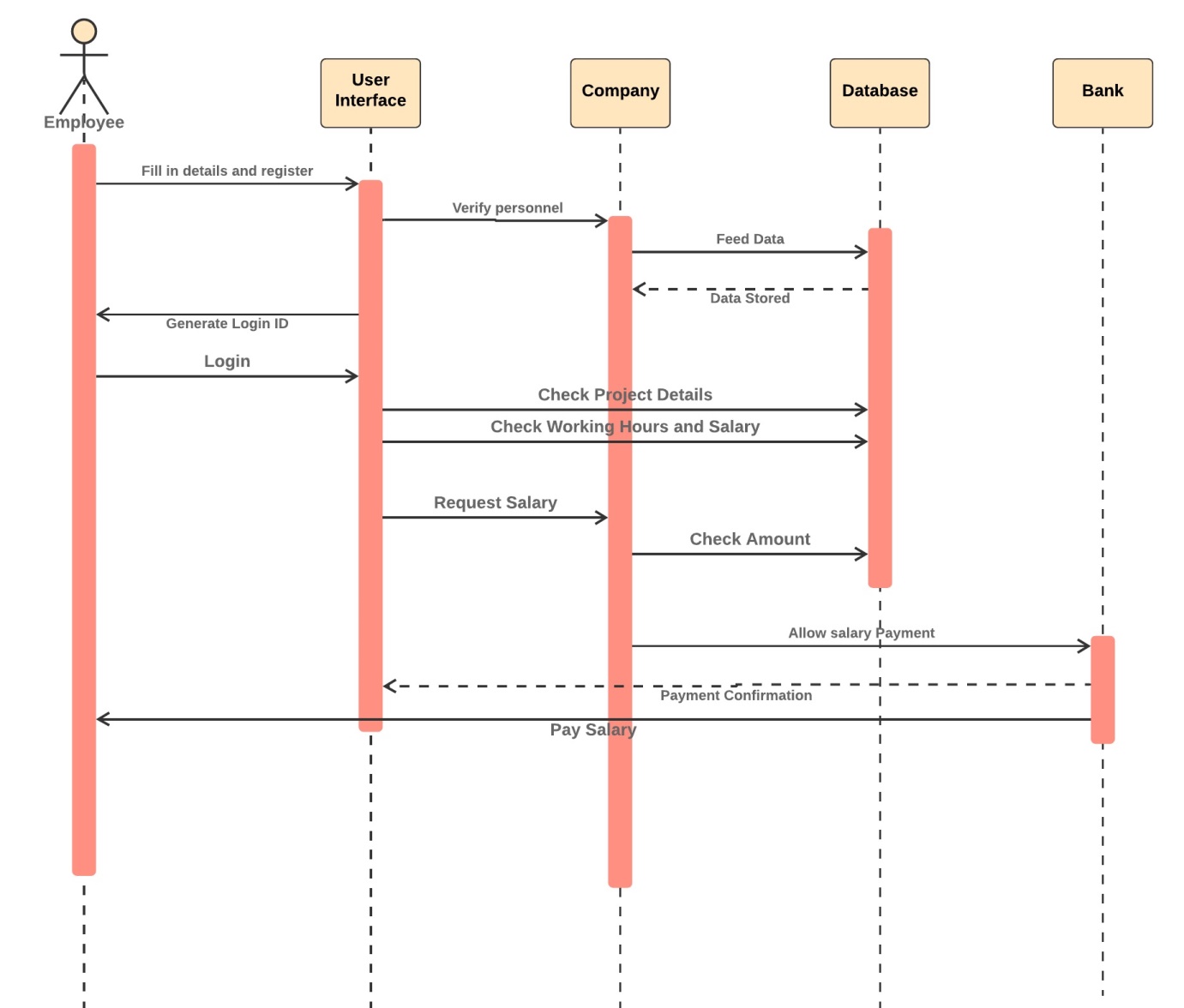
**2)CLASS DIAGRAM:**

* A class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects.
* A UML class diagram is made up of:
  + A set of classes and
  + A set of relationships between classes
* A class notation consists of three parts:
  + Class Name: The name of the class appears in the first partition.
  + Class Attributes: Attributes are shown in the second partition and map onto member variables (data members) in code.
  + Class Operations (Methods): Operations are shown in the third partition. They are services the class provides and map onto class methods in code
  + Class Relationships: A class may be involved in one or more relationships with other classes.



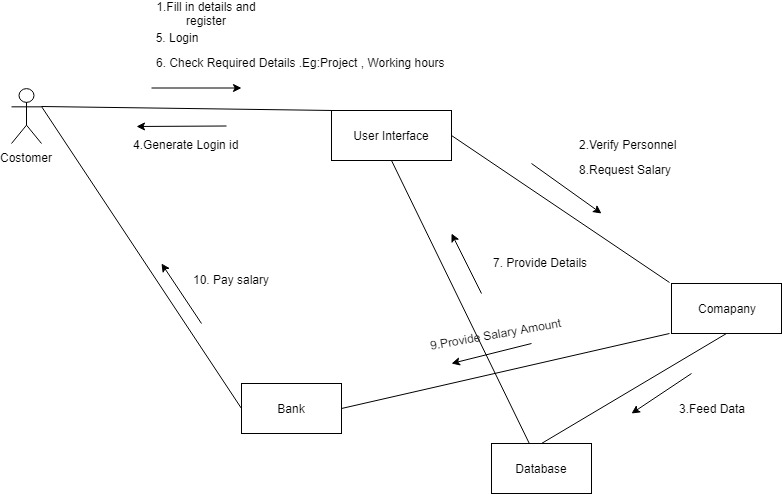
**3)SEQUENCE DIAGRAM**

Sequence Diagrams are interaction diagrams that detail how operations are carried out. They capture the interaction between objects in the context of a collaboration. Sequence Diagrams are time focus and they show the order of the interaction visually by using the vertical axis of the diagram to represent time what messages are sent and when



**4)COLLABORATION DIAGRAM**

A collaboration diagram, also called a communication diagram or interaction diagram, is an illustration of the relationships and interactions among [software](https://searchmicroservices.techtarget.com/definition/software) [object](https://searchmicroservices.techtarget.com/definition/object)s in the Unified Modeling Language (UML). A collaboration diagram resembles a [flowchart](https://whatis.techtarget.com/definition/flowchart) that portrays the roles, functionality and behavior of individual objects as well as the overall operation of the system in [real time](https://whatis.techtarget.com/definition/real-time). Objects are shown as rectangles with naming labels inside. These labels are preceded by colons and may be underlined. The relationships between the objects are shown as lines connecting the rectangles. The [message](https://whatis.techtarget.com/definition/message)s between objects are shown as arrows connecting the relevant rectangles along with labels that define the message sequencing.



**5)ACTIVITY DIAGRAM**

Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system. The control flow is drawn from one operation to another. This flow can be sequential, branched, or concurrent. Activity diagrams deal with all type of flow control by using different elements such as fork, join, etc.

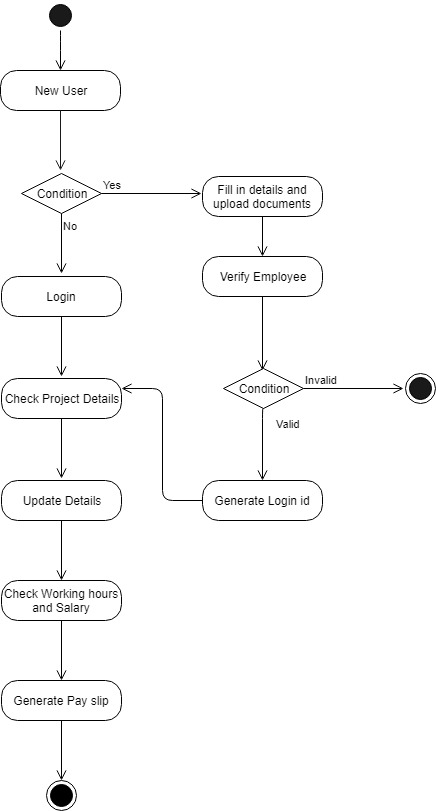
The purpose of an activity diagram can be described as −

* Draw the activity flow of a system.
* Describe the sequence from one activity to another.
* Describe the parallel, branched and concurrent flow of the system.

Before drawing an activity diagram, we should identify the following elements

* Activities
* Association
* Conditions
* Constraints

Once the above-mentioned parameters are identified, we need to make a mental layout of the entire flow. This mental layout is then transformed into an activity diagram.



**6)COMPONENT DIAGRAM**

Component diagrams are different in terms of nature and behavior. They are used to model the physical aspects of a system. Physical aspects are the elements such as executables, libraries, files, documents, etc. which reside in a node.

Component diagrams are used to visualize the organization and relationships among components in a system. These diagrams are also used to make executable systems.

Their purpose can be summarized as −

* Visualize the components of a system.
* Construct executables by using forward and reverse engineering.
* Describe the organization and relationships of the components.

**7)DEPLOYMENT DIAGRAM**

A deployment diagram is a UML diagram type that shows the execution architecture of a system, including nodes such as hardware or software execution environments, and the middleware connecting them. They are typically used to visualize the physical hardware and software of a system. Using it you can understand how the system will be physically deployed on the hardware.

Deployment diagrams help model the hardware topology of a system compared to other UML diagram types which mostly outline the logical components of a system.

