

SYNOPSIS

The project definition name describes about creating a web application that can be used to keep track of the project process.

The Project tracking system has developed in order to plan, coordinate and control the complex and diverse activities of modern industrial and commercial projects. The main goal of the project is to achieve the desired outcome at a specific end date.

Project Title : Project Tracking and Management System

Abstract : The project deals with various levels of project development and will account for the time used in analysis, design, programming, testing, verification and compute employee contribution etc.

Project Type : Website

Technology : HTML, CSS, JAVA script with PHP

Database : MySQL

Project tracking refers to the management of projects, which includes but is not limited to measuring and reporting the status of milestones, tasks and activities required in achieving the predefined project results. A project is a one-time effort to accomplish an explicit objective by a specific time. Each project is unique although similar projects may exist. Every project has definite beginning and definite ending. One of the key component of project tracking is controlling.

Project tracking requires a careful balance of monitoring to ensure that you will achieve the results and respond to any incidents or roadblocks, while avoiding micro management and reducing team member's ability to make decisions. There are various tools and techniques available to manage and track the projects. Project management software is frequently used by the companies to assist in managing the initiation, execution, tracking and closing of projects.

ACKNOWLEDGEMENT

While presenting this Database Project on "**PROJECT TRACKING AND MANAGEMENT SYSTEM**", I feel that it is our duty to acknowledge the help rendered to us by various persons.

Firstly I thank God for showering his blessings on me. I am grateful to my institution PESIT Bangalore South Campus for providing me a congenial atmosphere to carry out the project successfully.

I would like to express my heartfelt gratitude to **Dr. J SURYAPRASAD**, Principal, Bangalore, for extending his support.

I would also like to express my heartfelt gratitude to **Dr. ANNAPOORNA** Professor & HOD, Information Science and Engineering whose guidance and support was truly invaluable.

I am very grateful to my guide, RASHMA and KAKOLI BORA, Department of Information Science, for their able guidance and valuable advice at every stage of my project which helped me in the successful completion of my project.

I would also indebted to my Parents and Friends for their continued moral and material support throughout the course of project and helping me in finalize the presentation.

My heartfelt thanks to all those who have contributed bits, bytes and words to accomplish this Project.

Thanking you'll,

CHARAN KUMAR S 1PE15IS027

TABLE OF CONTENTS

Chapter1	Introduction
	1.1 Introduction to Database
	1.2 Introduction to Relational Database
	1.3 Introduction to Languages
Chapter2	Literature Survey
	2.1 Library Function
	2.2 User Defined Functions
Chapter3	Requirements Specification
	3.1 Software Requirements
	3.2 Hardware Requirements
Chapter4	Design
	4.1 ER Diagram
	4.2 Schema Diagram
Chapter5	Implementation and Snap shots and SOURCE CODE
Chapter6	Conclusion
Chapter7	Bibliography
	8.1 References
	8.2 Websites for Reference

Project Tracking and Management System

INTRODUCTION

1.1 INTRODUCTION TO DATABASE

A database is a collection of information that is organized so that it can be easily accessed, managed and updated.

Data is organized into rows, columns and tables, and it is indexed to make it easier to find relevant information. Data gets updated, expanded and deleted as new information is added. Databases process workloads to create and update themselves, querying the data they contain and running applications against it.

Computer databases typically contain aggregations of data records or files, such as sales transactions, product catalogues and inventories, and customer profiles.

Typically, a database manager provides users with the ability to control read/write access, specify report generation and analyse usage. Some databases offer **ACID** (atomicity, consistency, isolation and durability) compliance to guarantee that data is consistent and that transactions are complete.

Databases are prevalent in large mainframe systems, but are also present in smaller distributed workstations and midrange systems, such as IBM's AS/400 and personal computers.

1.2 INTRODUCTION TO RELATIONAL DATABASE

A relational database, invented by E.F. Codd at IBM in 1970, is a tabular database in which data is defined so that it can be reorganized and accessed in a number of different ways.

A relational database management system (RDBMS) is a program that are made up of a set of tables with data that fits into a predefined category and lets you create, update, and administer a relational database.

The leading RDBMS products are Oracle, IBM's DB2 and Microsoft's SQL Server. Despite repeated challenges by competing technologies, as well as the claim by some experts that no current RDBMS has fully implemented relational principles, the majority of new corporate databases are still being created and managed with an RDBMS.

The Structured Query Language (SQL) is the standard user and application program interface for a relational database. Relational databases are easy to extend, and a new data category can be added after the original database creation without requiring that you modify all the existing applications.

1.3 INTRODUCTION TO SOFTWARES AND TOOLS USED:

PHP:

The programming language used is Java, which is implemented using the software PHP is now officially known as “**PHP: Hypertext Preprocessor**”. It is a server-side scripting language usually written in an HTML context. Unlike an ordinary HTML page, a PHP script is not sent directly to a client by the server; instead, it is parsed by the PHP binary or module, which is server-side installed. HTML elements in the script are left alone, but PHP code is interpreted and executed. PHP code in a script can query databases, create images, read and write files, talk to remote servers – the possibilities is endless. The output from PHP code is combined with the HTML in the script and the result sent to the user’s web-browser, therefore it can never tell the user whether the web-server uses PHP or not, because the entire browser sees is HTML.

PHP’s support for Apache and MySQL further increases its popularity. Apache is now the most-used web-server in the world, and PHP can be compiled as an Apache module. MySQL is a powerful free SQL database, and PHP provides a comprehensive set of functions for working with it. The combination of Apache, MySQL and PHP is all but unbeatable.

That doesn’t mean that PHP cannot work in other environments or with other tools. In fact, PHP supports an extensive list of databases and web-servers. While in the mid-1990s it was ok to build sites, even relatively large sites, with hundreds of individual hard-coded HTML pages, today’s webmasters are making the most of the power of

Project Tracking and Management System databases to manage their content more effectively and to personalize their sites according to individual user preferences.

.

1.3.1 HTML

- 2 HTML or **Hyper Text Markup Language** is the standard markup language used to create web pages.
- 3 HTML was created in 1991 by Tim Berners-Lee at CERN in Switzerland. It was designed to allow scientists to display and share their research.
- 4 HTML is written in the form of HTML elements consisting of *tags* enclosed in angle brackets (like <html>). HTML tags most commonly come in pairs like <h1> and </h1>, although some tags represent *empty elements* and so are unpaired, for example . The first tag in a pair is the *start tag*, and the second tag is the *end tag* (they are also called *opening tags* and *closing tags*).
- 5 The purpose of a web browser is to read HTML documents and compose them into visible or audible web pages. The browser does not display the HTML tags, but uses the tags to interpret the content of the page. HTML describes the structure of a website semantically along with cues for presentation, making it a markup language rather than a programming language.
- 6 HTML elements form the building blocks of all websites. HTML allows images and objects to be embedded and can be used to create interactive forms. It provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. It can embed scripts written in languages such as Java Script which affect the behavior of HTML web pages.
- 7 HTML is descriptive markup language. Library of various markup languages is defined in various browsers.

1.3.1.1 HTML 5

- 1 HTML5 will be the new standard for HTML. The previous version of HTML, HTML 4.01,

Project Tracking and Management System

- 2 came in 1999. The web has changed a lot since then. HTML5 is still a work in progress.
- 3 However, the major browsers support many of the new HTML5 elements and APIs.
- 4 HTML5 is cooperation between the World Wide Web Consortium (W3C) and the Web
- 5 Hypertext Application Technology Working Group (WHATWG).
- 6 WHATWG was working with web forms and applications, and W3C was working with
- 7 XHTML 2.0. In 2006, they decided to cooperate and create a new version of HTML.
- 8 Some rules for HTML5 were established:
- 9 a) New features should be based on HTML, CSS, DOM, and JavaScript
- 10 b) Reduce the need for external plug-ins (like Flash)
- 11 c) Better error handling
- 12 d) More markup to replace scripting
- 13 e) HTML5 should be device independent
- 14 f) The development process should be visible to the public

1.3.2 CSS

CSS tutorial or CSS 3 tutorial provides basic and advanced concepts of CSS technology. Our CSS tutorial is developed for beginners and professionals. The major points of CSS are given below:

CSS stands for Cascading Style Sheet.

CSS is used to design HTML tags.

CSS is a widely used language on the web.

HTML, CSS and JavaScript are used for web designing. It helps the web designers to apply style on HTML tags.
Dept. of ISE|PESIT BSC

Project Tracking and Management System

Cascading Style Sheets (CSS) is a style sheet language used for describing the look and formatting of a document written in a markup language. While most often used to style web pages and user interfaces written in HTML and XHTML, the language can be applied to any kind of XML document, including plain XML, SVG and XUL. CSS is a cornerstone specification of the web and almost all web pages use CSS style sheets to describe their presentation.

CSS is designed primarily to enable the separation of document content from document presentation, including elements such as the layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple pages to share formatting, and reduce complexity and repetition in the structural content (such as by allowing for table less web design).

CSS can also allow the same markup page to be presented in different styles for different rendering methods, such as on-screen, in print, by voice (when read out by a speech-based browser or screen reader) and on Braille-based, tactile devices. It can also be used to allow the web page to display differently depending on the screen size or device on which it is being viewed. While the author of a document typically links that document to a CSS file, readers can use a different style sheet, perhaps one on their own computer, to override the one the author has specified.

With plain HTML you define the colors and sizes of text and tables throughout your pages. If

you want to change a certain element you will therefore have to work your way through the

document and change it. With CSS you define the colors and sizes in "styles". Then as you

write your documents you refer to the styles. Therefore: if you change a certain style it will

change the look of your entire site. Another big advantage is that CSS offers much more detailed attributes than plain HTML for defining the look and feel of your site.

1.3.3 JAVA SCRIPT

JavaScript (JS) is a dynamic computer programming language. It is most commonly used as part of web browsers, whose implementations allow client-side scripts to interact with the user, control the browser, communicate asynchronously, and alter the document content that is displayed. It is also being used in server-side network programming (with Node.js), game development and the creation of desktop and mobile applications.

JavaScript is a prototype-based scripting language with dynamic typing and has first-class functions. Its syntax was influenced by C. JavaScript copies many names and naming conventions from Java, but the two languages are otherwise unrelated and have very different semantics. The key design principles within JavaScript are taken from the Self and Scheme programming languages. It is a multi-paradigm language, supporting object-oriented, imperative, and functional programming styles.

The application of JavaScript in use outside of web pages—for example, in PDF documents, site-specific browsers, and desktop widgets—is also significant. Newer and faster JavaScript VMs and platforms built upon them (notably Node.js) have also increased the popularity of JavaScript for server-side web applications. On the client side, JavaScript was traditionally implemented as an interpreted language but just-in-time compilation is now performed by recent (post-2012) browsers.

JavaScript was formalized in the ECMA Script language standard and is primarily used as part of a web browser (client-side JavaScript). This enables programmatic access to objects within a host environment.

JavaScript is the most popular programming language in the world.

It is the language for HTML, for the Web, for computers, servers, laptops, tablets, smart phones, and more.

You can use JavaScript to:

- a) Change HTML elements
 - Delete HTML elements
 - Create new HTML elements
 - Copy and clone HTML elements

1.3.4 My Sqli:



Introduction: The database has become an integral part of almost every human's life. Without it, many things we do would become very tedious, perhaps impossible tasks. Banks, universities, and libraries are three examples of organizations that depend heavily on some sort of database system. On the Internet, search engines, online shopping, and even the website naming convention would be impossible without the use of a database.

A database that is implemented and interfaced on a computer is often termed a database server.

One of the fastest SQL (Structured Query Language) database servers currently on the market is the MySQL server, developed by T.c.X. DataKonsultAB. MySQL, available for download at www.mysql.com, offers the database programmer with an array of options and capabilities rarely seen in other database servers. MySQL is free of charge for those wishing to use it for private and commercial use. Those wishing to develop applications specifically using MySQL should consult MySQL's licensing section, as there is charge for licensing the product.

LITERATURE SURVEY

2.1 LIBRARY FUNCTIONS

2.1.1 HTML Images - The Tag and the Src Attribute

In HTML, images are defined with the `` tag.

The `` tag is empty, which means that it contains attributes only, and has no closing tag.

To display an image on a page, you need to use the `src` attribute. `Src` stands for "source".

The value of the `src` attribute is the URL of the image you want to display.

Syntax for defining an image:

```

```

2.1.2 HTML FORMS

HTML forms are used to pass data to a server.

An HTML form can contain input elements like text fields, checkboxes, radio-buttons, submit buttons and more. A form can also contain select lists, textarea, fieldset, legend, and label elements.

2.1.3 Image tag ():

To add an image to an HTML document, we just need to include an `` tag with a reference to the desired image. The `` tag is an empty element i.e. it doesn't require a closing tag and we can use it to include from small icons to large images.

Syntax: ``

Dept. of ISE|PESIT BSC

2.1.4 HTML Lists:

An ordered list: <ul style="list-style-type: none">• The first list item• The second list item• The third list item	An unordered list: <ul style="list-style-type: none">• List item• List item• List item
---	--

2.1.5 SUM ():

SELECT SUM returns the sum of the data values.

Syntax: SELECT SUM(column-name)

FROM table-name;

2.1.6 DATEDIFF ():

The DATEDIFF () function returns the difference between two date values, based on the interval specified.

Syntax: DATEDIFF (*interval, date1, date2*);

2.1.7 COUNT ():

The SQL COUNT function is used to count the number of rows returned in a SELECT statement.

The COUNT () function returns the number of rows that matches a specified criterion.

2.1.8 COUNT (DISTINCT ()):

- SELECT DISTINCT returns only distinct (different) values.
- SELECT DISTINCT eliminates duplicate records from the results.

Project Tracking and Management System

- DISTINCT can be used with aggregates: COUNT, AVG, MAX, etc.
- DISTINCT operates on a single column. DISTINCT for multiple columns is not supported.
- Syntax: **SELECT DISTINCT column-name**

FROM table-name;

2.2 USER DEFINED FUNCTIONS

2.2.1 PROCEDURES : A stored procedure is a set of Structured Query Language statements with an assigned name, which are stored in relational database management system as a group. It can be reused and shared by multiple projects. In our project, the stored procedure can be implemented by finding the percentage of total hours worked by the employee and the total hours taken by the employee to complete the project. These variables will be fetched from the tables such as employee_work and project table and will be computed. Then we need to declare the variables such as employee total hours and project total hours. The total hours worked by each employee can be calculated by fetching the employee id from the employee table. The total hours to complete the project can be calculated by fetching the project id from the project table. Finally, the percentage can be calculated by dividing the employee total hours by project total hours and

multiplied by 100. In our project, the stored procedure can also be implemented by using project progress and task progress. The project progress can be calculated by fetching the distinct date like start and end date of the project from the project table which is divisible by completed days and multiplied by 100. Where the task progress can be calculated by fetching the task id from the milestone i.e., by dividing the completed task by the total task and which will be multiplied by 100.

2.2.1.1 EMPLOYEE CONTRIBUTION FOR A PARTICULAR PROJECT FUNCTION

delimiter \$\$

```
create procedure employee_contribution (IN emp_id_var varchar(20), IN proj_id_var varchar(20), OUT percentage_contributed int)
```

```
BEGIN
```

```
    declare emp_total_hrs int;
```

```
    declare proj_total_hrs int;
```

Project Tracking and Management System

```
select sum(worked_for_hours) into emp_total_hrs from  
EMPLOYEE_PROJECT_CONTRIBUTION where project_id=proj_id_var and  
employee_id=emp_id_var;  
  
select sum(worked_for_hours) into proj_total_hrs from  
EMPLOYEE_PROJECT_CONTRIBUTION where project_id=proj_id_var;  
  
set percentage_contributed = ((emp_total_hrs*100)/proj_total_hrs);  
  
/*select percentage_contributed;*/  
  
/*select emp_id_var;  
  
select proj_total_hrs;  
  
select emp_total_hrs;*/  
  
END$$  
  
delimiter ;
```

2.2.1.2 EMPLOYEE CONTRIBUTION IN ALL PROJECTS FUNCTION

```
delimiter $$
```

```
create procedure employee_contribution_in_all_projects (IN emp_id_var varchar(20),  
OUT percentage_contributed int)  
  
BEGIN  
  
declare emp_total_hrs int;  
  
declare proj_total_hrs int;  
  
declare proj_id_var varchar(20);  
  
declare cur1 cursor for select project_id from PROJECT_MEMBERS where  
manager=emp_id_var OR member1=emp_id_var OR member2=emp_id_var OR  
member3=emp_id_var OR member4=emp_id_var OR member5=emp_id_var OR  
member6=emp_id_var ;  
  
open cur1;  
  
beginning_of_loop : LOOP  
  
fetch cur1 into proj_id_var;
```

Project Tracking and Management System

```
    select sum(worked_for_hours) into emp_total_hrs from
EMPLOYEE_PROJECT_CONTRIBUTION where project_id=proj_id_var and
employee_id=emp_id_var;

    select sum(worked_for_hours) into proj_total_hrs from
EMPLOYEE_PROJECT_CONTRIBUTION where project_id=proj_id_var;

    set percentage_contributed = (emp_total_hrs/proj_total_hrs) * 100;

    select proj_id_var,percentage_contributed;

END LOOP;

close cur1;

/*select percentage_contributed;*/

/*select emp_id_var;

select proj_total_hrs;

select emp_total_hrs;*/

END$$

delimiter ;
```

2.2.1.3 EMPLOYEE PROJECT PROGRESS FINDER FUNCTION

```
delimiter $$

create procedure project_progress_finder(IN proj_id_var
varchar(20), OUT proj_progress int, OUT tasks_progress int)

BEGIN

declare proj_total_days int;

declare completed_days int;

declare completed_tasks int;

declare total_tasks int;

declare start_date_var date;
```

```

Project Tracking and Management System
    declare end_date_var date;

        select count(distinct on_date) into completed_days from
EMPLOYEE_PROJECT_CONTRIBUTION where project_id=proj_id_var;

        /*select completed_days;*/

        select start_date into start_date_var from PROJECT where
project_id = proj_id_var ;

        select end_date into end_date_var from PROJECT where
project_id = proj_id_var ;

        select DATEDIFF(end_date_var,start_date_var) into
proj_total_days;

        /*select proj_total_days;*/

        set proj_progress = (completed_days/proj_total_days)*100;

        select count(task_id) into total_tasks from MILESTONES where
project_id = proj_id_var;

        select count(task_id) into completed_tasks from MILESTONES
where project_id=proj_id_var and status='completed';

        set tasks_progress = (completed_tasks/total_tasks)*100;

        END$$

delimiter ;

```

2.2.2 TRIGGER:

2.2.2.1 AUTO MANAGER ADD TO PROJECT MEMBER

```

delimiter $$

create trigger member_add
    after insert on PROJECT_MEMBERS
    FOR EACH ROW
BEGIN
    update PROJECT_MEMBERS set member6 = NEW.manager where
project_id=NEW.project_id ;
END$$

```

Project Tracking and Management System

delimiter ;

2.3 EXISTING SYSTEM

- Existing system is an MS-Excel based system.
- Project Team members maintain the status of their assigned tasks on their individual desks.
- On demand reports are prepared by collecting status information from individuals, and printed or softcopy is submitted to relevant authority

2.3.1 DRAWBACKS OF EXISTING SYSTEM

- Spread sheet files are maintained on individual's desktop, that creates lot of redundancy and discrepancy.
- Report generations on ad-hoc basis is not possible.
Even to know the status of the project, required to look in to hundreds of files.

2.4 PROPOSED SYSTEM

- Proposed system is the web based project management system.

DISTINCT FEATURES

- Reflects the accurate status of the master plans, projects and tasks at any instance of time.
- Ease the work of project leader and reviewer.
- Easy to define task dependency.
- CUSTOMISED SECURITY allows to create team hierarchy with different levels of credentials.
- Enhance communication among team members, using integrated e-mails and file sharing.
- Alert team members before the task due.

CHAPTER 3

REQUIREMENTS

3.1 HARDWARE REQUIREMENTS

Processor: Intel Core i7-7700U Processor (2.81GHz 1600MHz 3MB)

Operating system: Windows 10

Memory: 8.0GB PC3L-12800 DDR3L 1600MHz

Hard drive: 1TB 7200rpm Serial ATA hard drive

Video card: Intel® HD graphics

Display: 15.6"HD WLED 1366x768

Keyboard: English keyboard without backlight

Ports: 1 USB 3.0

2 USB 2.0

1 HDMI v1.48

Slots: Kensington lock slot

Multi Card Slot (2-in-1 Card Reader)

Keyboard: Full size, spill-resistance keyboard with 10-key numeric keypad

Touchpads

Multi -touch gesture -enabled pad with integrated scrolling

Camera: HD (720p) capable webcam, microphone

Dimension: Height: 25mm (0.98) x width: 384mm (15.1) x depth: 265mm (10.4)

Dept. of ISE|PESIT BSC

3.2 SOFTWARE REQUIREMENTS

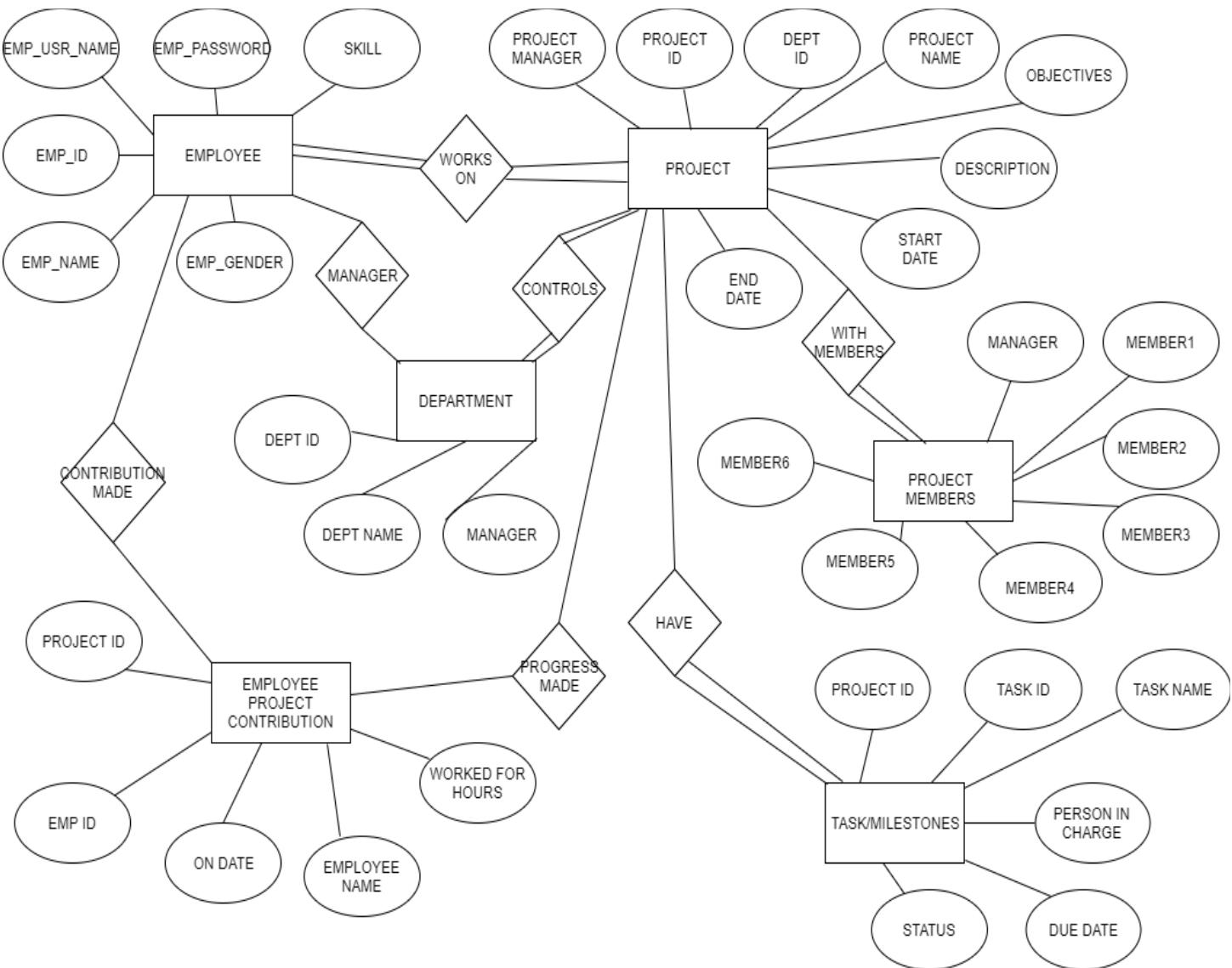
- **Language Used** : PHP, JavaScript
- **Database** : MySQL
- **User Interface Design** : HTML, CSS
- **Web Browser** : Google Chrome, Mozilla Firefox
- **Software** : XAMPP Version: 7.1.10 (XAMPP stands for Cross-Platform (X), Apache (A), Maria DB (M), PHP (P) and Perl (P))

CHAPTER 4

DESIGN ANALYSIS

4.1 ENTITY-RELATIONSHIP DIAGRAM

Project Tracking and Management System



4.2 RELATIONAL SCHEMA

Project Tracking and Management System

DEPARTMENTS

department_id	department_name	manager
---------------	-----------------	---------

EMPLOYEE

employee_id	employee_name	gender	username	password	skill
-------------	---------------	--------	----------	----------	-------

EMPLOYEE_PROJECT_CONTRIBUTION

project_id	employee_id	on_date	employee_name	worked_for_hours
------------	-------------	---------	---------------	------------------

MILESTONES

project_id	task_id	task_name	person_in_charge	due_date	status
------------	---------	-----------	------------------	----------	--------

PROJECT

project_id	project_name	objective	description	start_date	end_date	department_id	project_manager
------------	--------------	-----------	-------------	------------	----------	---------------	-----------------

PROJECT_MEMBERS

project_id	manager	member1	member2	member3	member4	member5	member6
------------	---------	---------	---------	---------	---------	---------	---------

CHAPTER 5

IMPLEMENTATION

PROJECT TRACKING AND MANAGEMENT is basically a web application that is used for tracking and monitoring the progress and project and maintain each employee contribution.

The design of this application is implemented by in HTML and PHP which is the front end and the back end is the database which is implemented by using the MySQL.

The implementation of the application follows the following steps:

The application has two domains: User and Admin
User:

1. The User logins to the application with the appropriate username and password which he/she has created.
2. Then the user will be able to view all the projects that are currently present in which he is member of.
3. He can create a new project if he is a manager and assign tasks to employees .
4. View existing projects and update them.
5. There is a page for employees where they can update their work and tasks.

Admin:

6. The Admin logins to the application with the appropriate username and password which he/she has created.
7. Then the user will be able to view all the projects that are currently present .
8. He can create a new project and assign tasks to employees .
9. View existing projects and update them.
10. There is a page for employees where they can update their work and tasks.

SOURCE CODE

Project Tracking and Management System

CREATE TABLE STATEMENTS:

```
create table EMPLOYEE(employee_id varchar(20), employee_name  
varchar(20), gender char(1), username varchar(20), password  
varchar(20), skill varchar(10));
```

```
create table PROJECT(project_id varchar(20), project_name  
varchar(30), objective varchar(50), description varchar(100),  
start_date date, end_date date, department_id varchar(20),  
project_manager varchar(20));
```

```
create table PROJECT_MEMBERS(project_id varchar(20), manager  
varchar(20), member1 varchar(20), member2 varchar(20), member3  
varchar(20), member4 varchar(20), member5 varchar(20), member6  
varchar(20));
```

```
create table MILESTONES(project_id varchar(20), task_id  
varchar(20), task_name varchar(30), person_in_charge varchar(20),  
due_date date, status varchar(20));
```

```
create table DEPARTMENTS(department_id varchar(20),  
department_name varchar(20), manager varchar(20));
```

```
create table EMPLOYEE_PROJECT_CONTRIBUTION(project_id  
varchar(20), employee_id varchar(20), on_date date, employee_name  
varchar(20), worked_for_hours int);
```

CONSTRAINTS:

```
alter table EMPLOYEE add primary key (employee_id);
```

Project Tracking and Management System

```
alter table DEPARTMENTS add primary key (department_id);

alter table DEPARTMENTS add foreign key (manager) references
EMPLOYEE(employee_id) on delete cascade on update cascade;
```



```
alter table PROJECT add primary key (project_id);

alter table PROJECT add foreign key (project_manager) references
EMPLOYEE(employee_id) on delete cascade on update cascade;

alter table PROJECT add foreign key (department_id) references
DEPARTMENTS(department_id) on delete cascade on update cascade;
```



```
alter table PROJECT_MEMBERS add foreign key (project_id)
references PROJECT(project_id) on delete cascade on update
cascade;

alter table PROJECT_MEMBERS add primary key (project_id);

alter table PROJECT_MEMBERS add foreign key (manager) references
EMPLOYEE(employee_id) on delete cascade on update cascade;

alter table PROJECT_MEMBERS add foreign key (member1) references
EMPLOYEE(employee_id) on delete cascade on update cascade;

alter table PROJECT_MEMBERS add foreign key (member2) references
EMPLOYEE(employee_id) on delete cascade on update cascade;

alter table PROJECT_MEMBERS add foreign key (member3) references
EMPLOYEE(employee_id) on delete cascade on update cascade;

alter table PROJECT_MEMBERS add foreign key (member4) references
EMPLOYEE(employee_id) on delete cascade on update cascade;

alter table PROJECT_MEMBERS add foreign key (member5) references
EMPLOYEE(employee_id) on delete cascade on update cascade;

alter table PROJECT_MEMBERS add foreign key (member6) references
EMPLOYEE(employee_id) on delete cascade on update cascade;
```



```
alter table MILESTONES add foreign key (project_id) references
PROJECT(project_id) on delete cascade on update cascade;

alter table MILESTONES add primary key (project_id, task_id);

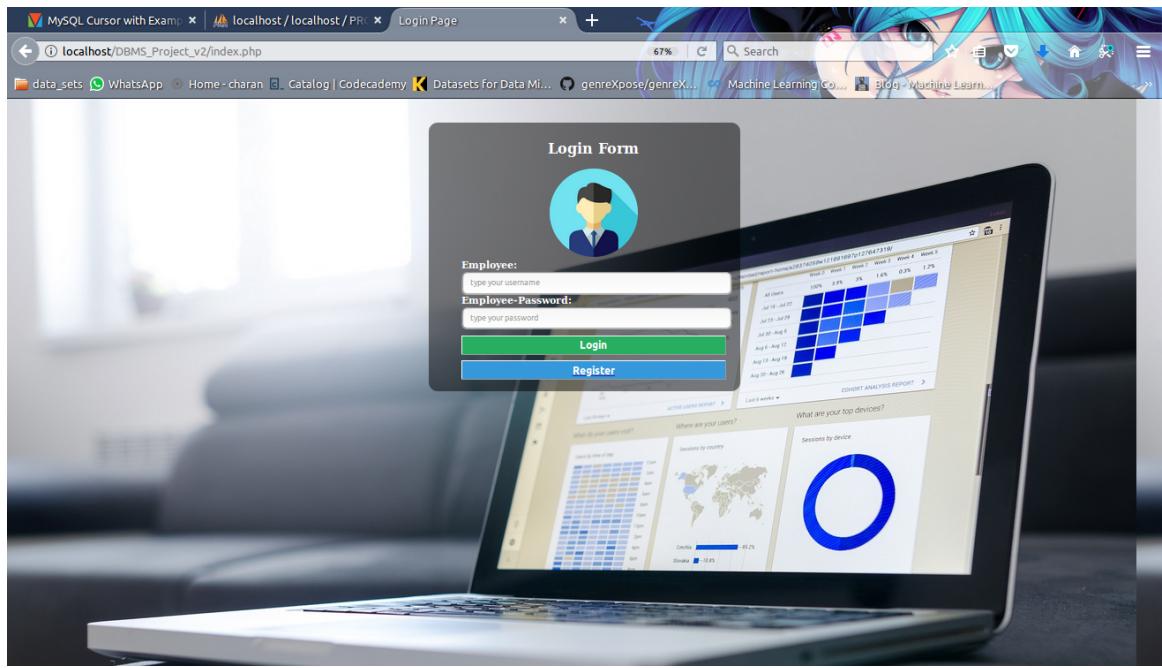
alter table MILESTONES add foreign key (person_in_charge)
references EMPLOYEE(employee_id) on delete cascade on update
cascade;
```

Project Tracking and Management System

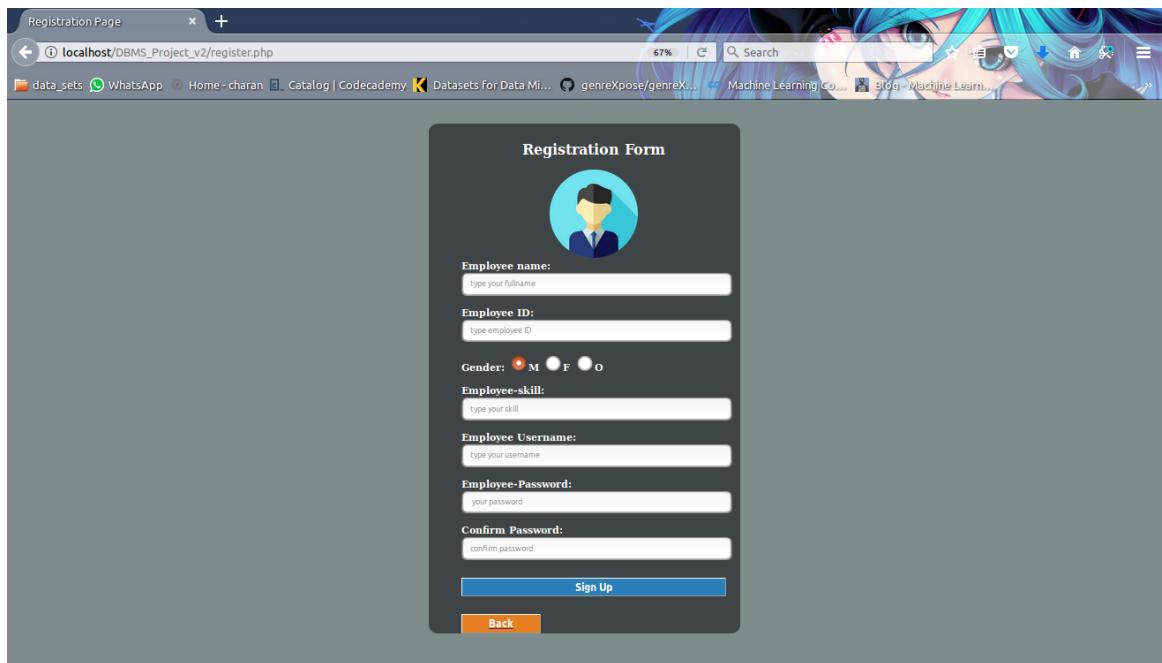
```
alter table EMPLOYEE_PROJECT_CONTRIBUTION add foreign key  
(project_id) references PROJECT(project_id) on delete cascade on  
update cascade;  
  
alter table EMPLOYEE_PROJECT_CONTRIBUTION add foreign key  
(employee_id) references EMPLOYEE(employee_id) on delete cascade  
on update cascade;  
  
alter table EMPLOYEE_PROJECT_CONTRIBUTION add primary key  
(project_id,employee_id,on_date);
```

Project Tracking and Management System

SNAPSHOTS



A user interface to login and register.



Page to register new user.

Project Tracking and Management System

The screenshot shows the homepage of the system. The top navigation bar includes links for Admin Page, localhost/DBMS_Project_v2/admin.php, data_sets, WhatsApp, Home - charan, Catalog | Codecademy, Datasets for Data Mi..., genreXpose/genreX..., Machine Learning Co..., and Blog - Machine Learn... A large banner at the top says "PROJECT TRACKING AND MANAGEMENT SYSTEM". On the left, a sidebar has links for Dashboard, CREATE NEW PROJECT, VIEW RUNNING PROJECTS, EMPLOYEE'S PAGE, and LOGOUT. The main content area has a "WELCOME TO THE HOMEPAGE : HELLO amar009" message. Below it is a "PROJECTS" section with a table:

PROJECT ID	PROJECT NAME	OBJECTIVE	DESCRIPTION	START DATE
pro594	fernandez textiles project	automate textile production management	create an app to automate the production system management	2017-04-29
pro624	royal blue sales project	production and distribution management	create an app to manage supply chain management and logistic management	2017-04-05
pro987	india post services project	database for postal services	create an appl to maintain data about all the mails in a database	2017-01-30

Below the projects table are two sections: "TASK_PROGRESS" and "TASKS/MILESTONES", each containing a table:

PROJECT ID	TASK ID	STATUS
pro594	tas5941	ongoing
pro594	tas5942	starting soon
pro594	tas5943	starting soon
pro624	tas6241	completed
pro624	tas6242	ongoing
pro624	tas6243	starting soon
pro987	tas9871	completed
pro987	tas9872	completed
pro987	tas9873	ongoing

PROJECT ID	TASK ID	TASK NAME	PERSON IN CHARGE	DUUE DATE
pro594	tas5941	requirement listing	amzn046	2017-07-29
pro594	tas5942	design and implementation	amzn037	2017-10-29
pro594	tas5943	testing	amzn070	2017-12-10
pro624	tas6241	requirement listing	amzn045	2017-12-28
pro624	tas6242	design and implementation	amzn027	2018-07-26
pro624	tas6243	testing and maintenance	amzn036	2018-12-17
pro987	tas9871	requirement listing	amzn036	2017-04-30
pro987	tas9872	design and implementation	amzn054	2017-06-14
pro987	tas9873	testing	amzn018	2017-08-30

This is the homepage after login. This gives an overview of all running projects and their progress. This also leads to creation of new project page, view projects page, employees' page.

The screenshot shows the "ADD NEW PROJECT" form. It has a header "ADD NEW PROJECT" with a user icon. The form fields are:

- Project ID: project ID
- Project NAME: project name
- OBJECTIVES: objectives
- DESCRIPTION: type description
- START DATE: yyyy-mm-dd
- END DATE: yyyy-mm-dd
- DEPARTMENT ID: department id
- PROJECT MANAGER: project manager

A "Create project" button is at the bottom.

This page is to create a new project.

Project Tracking and Management System

ADD MEMBERS

localhost/DBMS_Project_v2/add_members.php

67% Search

data_sets WhatsApp Home - charan Catalog | Codecademy Datasets for Data Mi... genreXpose/genreX... Machine Learning Co... Blog - Machine Learn...

ADD MEMBERS

PROJECT ID: Enter project id

MANAGER: type employee name

MEMBER 1: type employee name

MEMBER 2: type employee name

MEMBER 3: type employee name

MEMBER 4: type employee name

MEMBER 5: type employee name

DONE ADD TASK

This is to add members to the project.

ADD NEW TASK

localhost/DBMS_Project_v2/add_task.php

67% Search

data_sets WhatsApp Home - charan Catalog | Codecademy Datasets for Data Mi... genreXpose/genreX... Machine Learning Co... Blog - Machine Learn...

ADD NEW TASK

PROJECT ID: Enter project id

TASK ID: Enter task id

TASK NAME: Enter task

PERSON-IN-CHARGE: person name

DUE DATE: yyyy-mm-dd

STATUS:

NEW PENDING

ADD TASK Back

This page is for assigning tasks.

Project Tracking and Management System

The screenshot shows a web browser window titled "VIEW PROJECTS". The URL is "localhost/DBMS_Project_v2/viewproject.php". The page displays a table of projects with columns: PROJECT ID, PROJECT NAME, OBJECTIVE, DESCRIPTION, START DATE, END DATE, DEPARTMENT ID, and PROJECT MANAGER. Below the table is a search form titled "VIEW PARTICULAR PROJECT" with fields for "Project ID:" and buttons for "Search" and "back".

PROJECTS							
							
PROJECT ID	PROJECT NAME	OBJECTIVE	DESCRIPTION	START DATE	END DATE	DEPARTMENT ID	PROJECT MANAGER
pro594	fernandez textiles project	automate textile production management	create an app to automate the production system management	2017-04-29	2017-12-31	dep594	amzn027
pro624	royal blue sales project	production and distribution management	create an app to manage supply chain management and logistic management	2017-04-05	2019-01-05	dep624	amzn009
pro987	india post services project	database for postal services	create an appl to maintain data about all the mails in a database	2017-01-30	2017-06-05	dep987	amzn063

VIEW PARTICULAR PROJECT

Enter the project ID of Project to display its details and its Tasks

Project ID:

Search

back

This page shows the view of all projects present and user can enter the project ID of the project to be viewed to get a detailed view of every project.

The screenshot shows a web browser window titled "Project Page". The URL is "localhost/DBMS_Project_v2/view_detailed.php?pro_id=pro987&login=Search". The page displays a table of projects with columns: PROJECT ID, PROJECT NAME, OBJECTIVE, DESCRIPTION, START DATE, END DATE, DEPARTMENT ID, and PROJECT MANAGER. Below the table is a table of tasks with columns: PROJECT ID, TASK ID, TASK NAME, PERSON IN CHARGE, DUE DATE, and STATUS. Further down are sections for "PROJECT PROGRESS" and "MEMBERS' CONTRIBUTIONS".

PROJECT ID	PROJECT NAME	OBJECTIVE	DESCRIPTION	START DATE	END DATE	DEPARTMENT ID	PROJECT MANAGER
pro987	india post services project	database for postal services	create an appl to maintain data about all the mails in a database	2017-01-30	2017-06-05	dep987	amzn063

PROJECT ID	TASK ID	TASK NAME	PERSON IN CHARGE	DUe DATE	STATUS
pro987	tas9871	requirement listing	amzn036	2017-04-30	completed
pro987	tas9872	design and implementation	amzn054	2017-06-14	completed
pro987	tas9873	testing	amzn018	2017-08-30	ongoing

PROJECT PROGRESS

Start Date : 2017-01-30
End Date : 2017-06-05
Project Progress : 12%
Task Progress : 67%

MEMBERS' CONTRIBUTIONS

- MANAGER :
Employee ID : amzn063
Percentage contributed : 19%
- Employee ID : amzn045
Percentage contributed : 10%
- Employee ID : amzn036
Percentage contributed : 8%
- Employee ID : amzn054
Percentage contributed : 10%
- Employee ID : amzn018
Percentage contributed : 19%
- Employee ID : amzn029
Percentage contributed : 14%
- Employee ID : amzn037
Percentage contributed : 12%

back

This page shows the detailed view of the project the user is looking for and shows all the project progress and employee contribution to the project in APPROX. percentage.

Project Tracking and Management System

The screenshot shows a web browser window titled "EMPLOYEE PAGE". The URL is "localhost/DBMS_Project_v2/employee.php". The page has a header "EMPLOYEES" with a circular profile icon. Below it is a table:

EMPLOYEE ID	EMPLOYEE NAME	GENDER	SKILL
amzn009	Amar	O	c
amzn018	Pooja charan	F	unix shell
amzn027	Charan Kumar	M	java
amzn029	harshitha	F	css
amzn036	monica	F	PIZZA
amzn037	kavya	F	javascript
amzn045	johny	M	maya
amzn046	Sanju Sam Mathew	M	python
amzn054	Moh.Hussain	M	.net
amzn063	Puneeth	M	html
amzn070	Ranjan	M	c#

Below the table is a form titled "VIEW PARTICULARS OF EMPLOYEE" with instructions: "Enter the employee ID of Employee to display his/her details and to update work done". It contains fields for "Employee ID:" (with placeholder "employee ID") and "Password:", a "View" button, and a "back" button.

This page shows all the employees in the company and a login for them to update their work progress and task progress.

The screenshot shows a web browser window titled "Project Page". The URL is "localhost/DBMS_Project_v2/employee_details.php". The page has a header "EMPLOYEE DETAILS" with a circular profile icon. It displays the following information for employee ID amzn027:

Employee ID : amzn027
Employee Name : Charan Kumar
Employee Gender : M
Employee username : charan027
Employee Password : web123
Employee skill : java

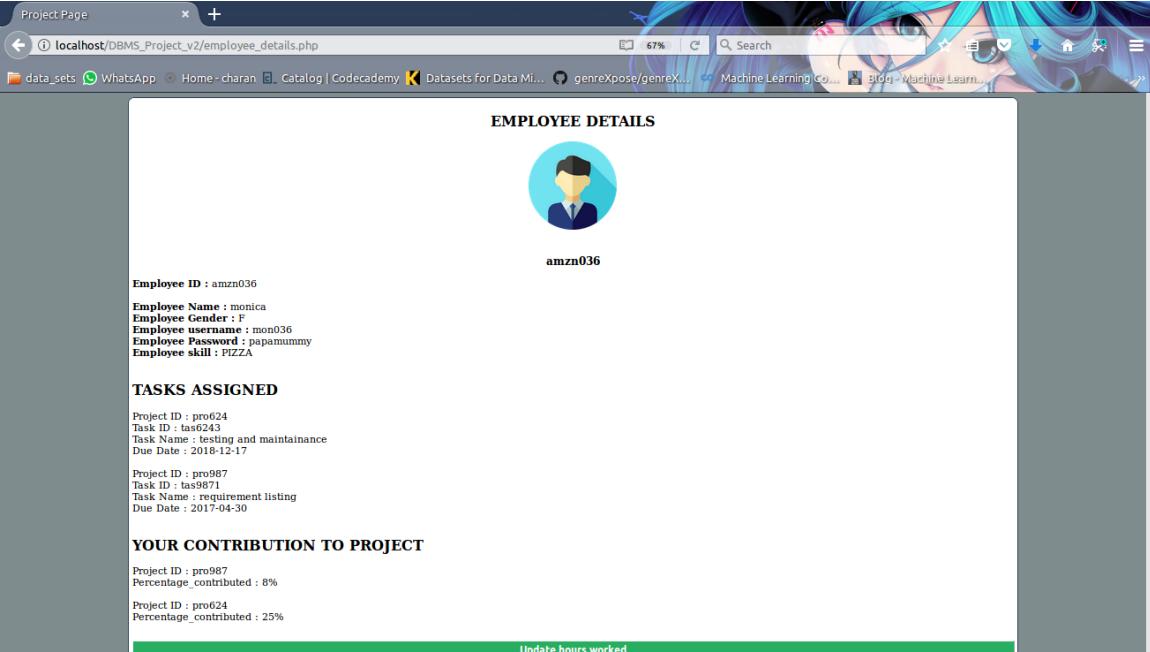
TASKS ASSIGNED
Project ID : pro624
Task ID : tas6242
Task Name : design and implementation
Due Date : 2018-07-26

YOUR CONTRIBUTION TO PROJECT
Project ID : pro594
Percentage_contributed : 24%
Project ID : pro624
Percentage_contributed : %

Buttons for "Update hours worked" and "Update Task Assigned", and a "back" button.

This is a sample of the employee details page which illustrates all the tasks and contributions made by the employee.

Project Tracking and Management System



The screenshot shows the 'EMPLOYEE DETAILS' page for employee 'amzn036'. It includes a profile picture placeholder, employee ID, and a table of assigned tasks and project contributions.

EMPLOYEE DETAILS

amzn036

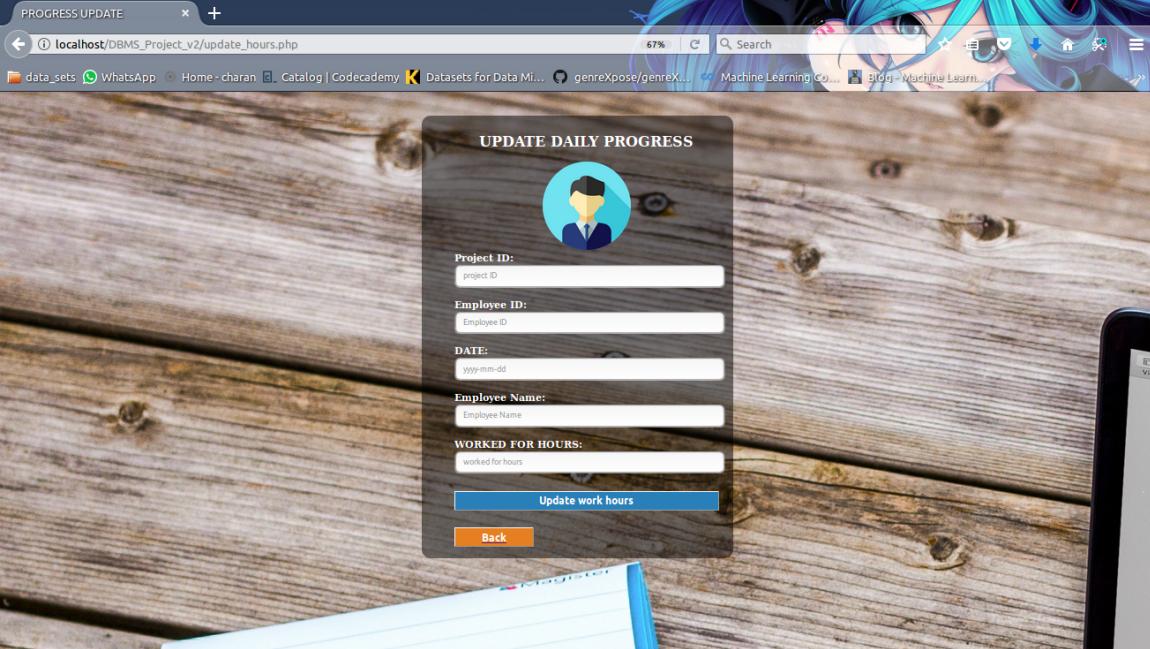
TASKS ASSIGNED	
Project ID : pro624	Task ID : tas6243
Task Name : testing and maintenance	Due Date : 2018-12-17
Project ID : pro987	Task ID : tas9871
Task Name : requirement listing	Due Date : 2017-04-30

YOUR CONTRIBUTION TO PROJECT

Project ID : pro987	Percentage_contributed : 8%
Project ID : pro624	Percentage_contributed : 25%

Update hours worked

This is another sample for the details page.



The screenshot shows the 'UPDATE DAILY PROGRESS' page. It features a profile picture placeholder and a form with fields for Project ID, Employee ID, Date, Employee Name, and Worked For Hours, along with 'Update work hours' and 'Back' buttons.

UPDATE DAILY PROGRESS

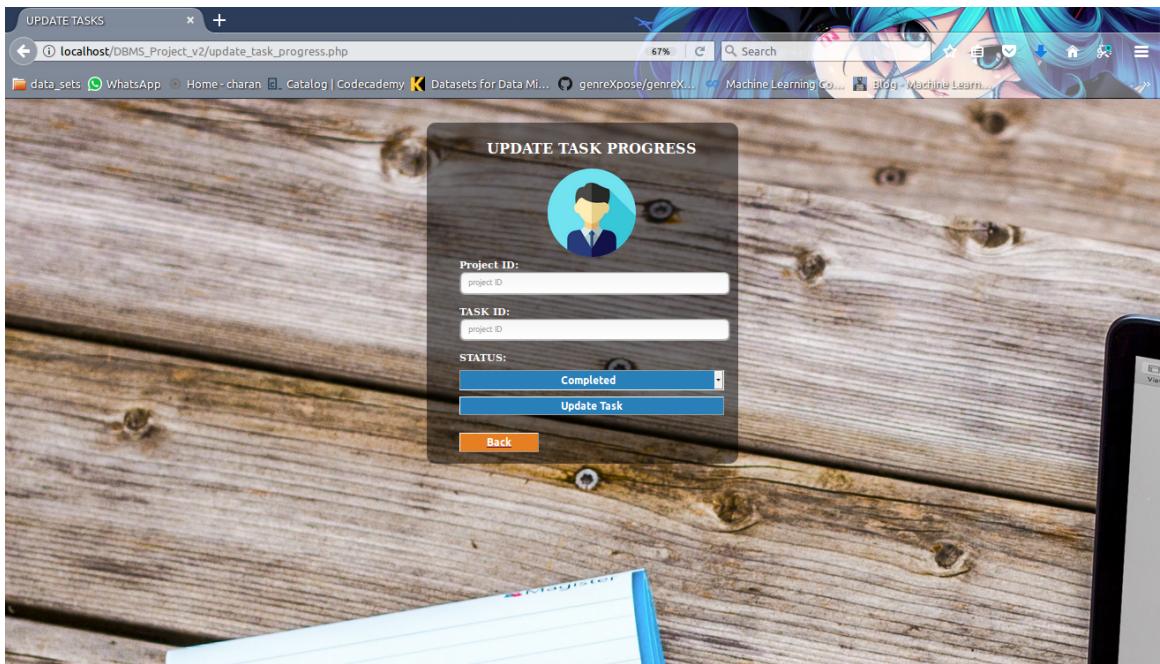
Project ID:	project ID
Employee ID:	Employee ID
DATE:	Www-mm-dd
Employee Name:	Employee Name
WORKED FOR HOURS:	worked for hours

Update work hours

Back

This page is provided with a form to update the daily progress of the employees.

Project Tracking and Management System



This page is for updating the progress in task made by the employees.



this snapshot shows the database and its elements. And the snapshot below shows the trigger implemented

A screenshot of the MySQL Workbench interface. The title bar says "Server: localhost » Database: PROJECT_TRACKING". The top menu has tabs for "Structure", "SQL", "Search", "Query", "Export", "Import", "Operations", and "Triggers". The "Triggers" tab is selected. A table lists a single trigger: "member_add" on the "PROJECT_MEMBERS" table, triggered "AFTER INSERT". The "Action" column shows icons for Edit, Export, and Drop. Below the table is a "Check all" checkbox and a "With selected:" dropdown with "Export" and "Drop" options. At the bottom left is a "New" button and a "Add trigger" button.

Project Tracking and Management System

The screenshot shows the MySQL Workbench interface with the PROJECT table selected. The table has columns: project_id, project_name, objective, description, start_date, end_date, department_id, and project_manager. Three rows of data are displayed:

project_id	project_name	objective	description	start_date	end_date	department_id	project_manager
pro594	fernandez textiles project	automate textile production management	create an app to automate the production system ma...	2017-04-29	2017-12-31	dep594	amzn027
pro624	royal blue sales project	production and distribution management	create an app to manage supply chain management an...	2017-04-05	2019-01-05	dep624	amzn009
pro987	india post services project	database for postal services	create an appl to maintain data about all the mail...	2017-01-30	2017-06-05	dep987	amzn063

This is a proof for the tables created in the database with populated data.

The screenshot shows the MySQL Workbench interface with the MILESTONES table selected. The table has columns: project_id, task_id, task_name, person_in_charge, due_date, and status. Ten rows of data are displayed:

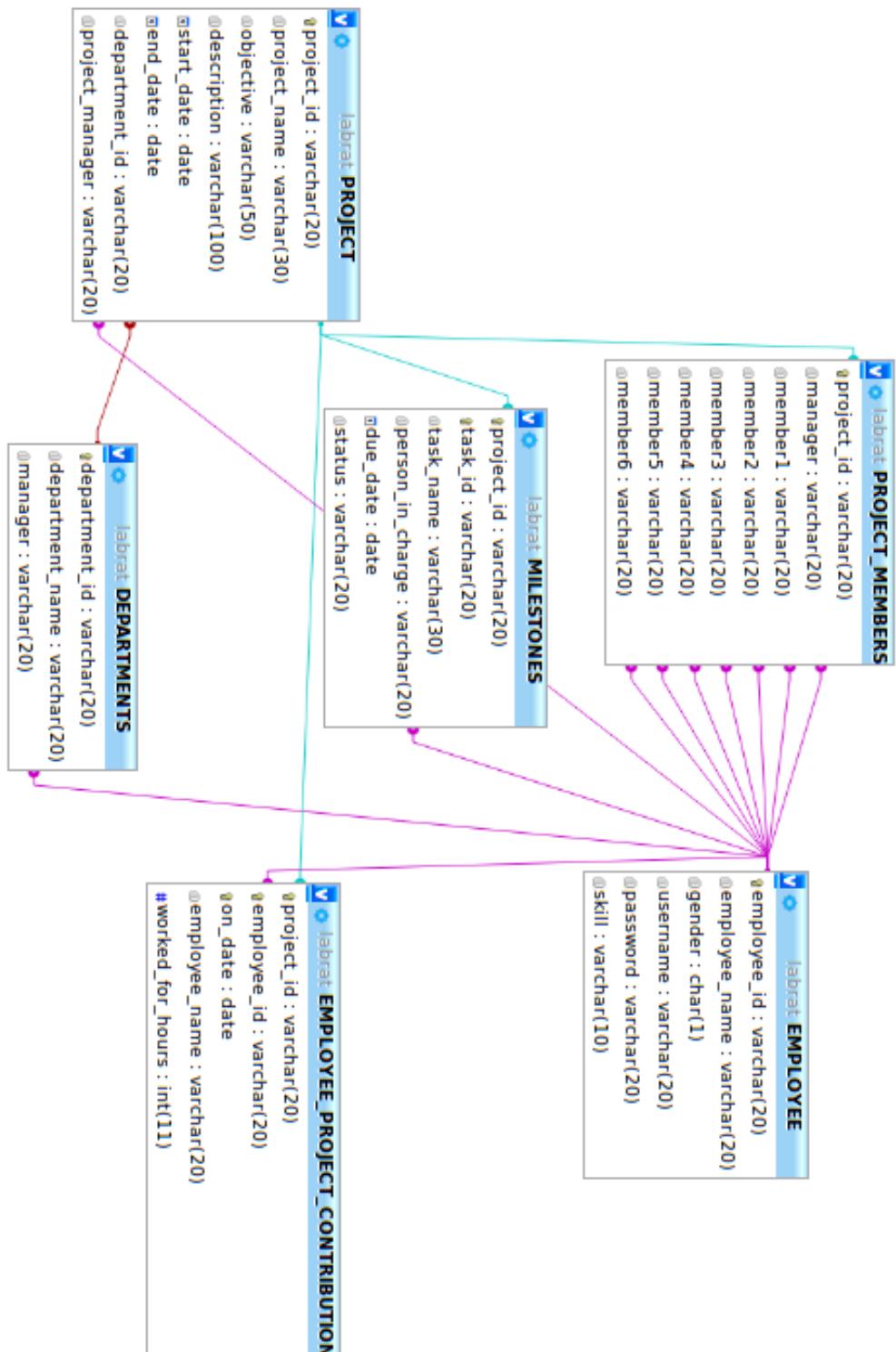
project_id	task_id	task_name	person_in_charge	due_date	status
pro594	tas5941	requirement listing	amzn046	2017-07-29	ongoing
pro594	tas5942	design and implementation	amzn037	2017-10-29	starting soon
pro594	tas5943	testing	amzn070	2017-12-10	starting soon
pro624	tas6241	requirement listing	amzn045	2017-12-28	completed
pro624	tas6242	design and implementation	amzn027	2018-07-26	ongoing
pro624	tas6243	testing and maintainance	amzn036	2018-12-17	starting soon
pro987	tas9871	requirement listing	amzn036	2017-04-30	completed
pro987	tas9872	design and implementation	amzn054	2017-06-14	completed
pro987	tas9873	testing	amzn018	2017-08-30	ongoing

This is the table which maintains the MILESTONES for each of the project.

Project Tracking and Management System

This is the schema obtained from the database from phpMyAdmin.

Project Tracking and Management System



CHAPTER 6

CONCLUSION

A plan is not a project, it's a tool: You need to remember that 35% of all projects are consumed in overheads of administration and project failures will happen.

Gain Knowledge before starting: You should always gain knowledge of the project before you start the project. Know who's in your project environment and manage them so projects do not run into politics. If you understand who's involved keep them informed by setting direction and act by keeping the project moving forward.

Monitor: Keep track of your project at all times by monitoring and controlling any risks that should arise.

Know your position: Be sure to ask how much money you have, how much time and when are the deliverables are expected. Know what 'you own and agree with the customer as to what critical success factor are.

And finally Project Managers are rarely hands on! Managers need to manage, minimize and control and if there is anything to do then do it!

BIBLIOGRAPHY

8.1 References

1. Database system models, languages, Designs and Application programming – RamezElmasri

And Shamkant B. Navathe, 7th Edition,2017, Pearson. (SQL Programming using PHP CHAPTER-13).

2. Database Management Systems – Ramakrishnan, and Gehrke, 3rdEdition, 2014, McGraw Hill

8.2 Reference Websites

1. **Website referred:** <https://www.slideshare.net>

HTML Learning:

- <https://www.codecademy.com/>
- <https://dash.generalassemb.ly/>
- <https://www.w3schools.com/>
- <https://www.youtube.com/watch?v=E9MyRFjh8oo/>

PHP Learning:

- <http://www.tutorialspoint.com/php/>
- <https://killerphp.com>
- <https://www.w3schools.com/>

Project Tracking and Management System