A DBMS Mini-Project

on

ACADEMIC RECORDS MANAGEMENT SYSTEM

Submitted by:

Sutej R Jantali (20GACSE073)

Vinayak G M (20GACSE080)

VSem, B.Tech(CSE)

Under the Guidance of

Dr. Samyama Gunjal G H

Associate Professor

Dept. of CSE, UVCE

Deepa Patil

Guest Faculty

Dept. of CSE, UVCE

Department of Computer Science and Engineering

UNIVERSITY VISVESVARAYA COLLEGE OF ENGINEERING

KR Circle Bengaluru-560001

January-2023

Bangalore University

UNIVERSITY VISVESVARAYA COLLEGE OF ENGINEERING

KR Circle Bengaluru-560001

Department of Computer Science and Engineering

CERTIFICATE

This is to certify that **Sutej R Jantali** of V Semester B.Tech, Computer Science and Engineering bearing the register number **20GACSE073** has submitted the DBMS Mini Project Report on **“ACADEMIC RECORDS MANAGEMENT SYSTEM”,** in partial fulfilment for the DBMS Lab, prescribed by the **Bangalore University, UVCE** during the academic year 2022-23.

Dr. Samyama Gunjal G H
Associate Professor
Dept of CSE, UVCE
Bangalore

Dr. H S Vimala
Professor and Chairperson
Dept of CSE, UVCE
Bangalore

Examiner 1

……………

Examiner 2

…………...

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany the successful completion of any task would be incomplete without the mention of the people who made it possible, whose constant guidance and encouragement crowned our effort with success.

We express our gratitude to University Visvesvaraya College of Engineering, Bangalore for giving us the opportunity to carry out this project. We also like to thank our Honourable Principal Dr. H N Ramesh for giving us all the support to make this project successful.

We would like to thank our Professor and Chairperson Dr. H S Vimala for providing us all the facilities to work on this project.

We consider it as a privilege and honour to express our sincere gratitude to our professor and guide, Dr.Samyama Gunjal G H, Associate Professor, Dept. of CSE for her valuable guidance throughout the tenure of this project work, and her constant support and encouragement made this work successful. We express our gratitude and thanks to Suhasini Bagalkot, Guest Faculty, Department of Computer Science and Engineering, for her support and encouragement.

We also thank non-teaching staff of Dept. of CSE, our parents and our friends for their help, encouragement and support. Last but not the least, we thank God Almighty, without whose blessings this wouldn't have been possible.

Sutej R Jantali (20GACSE073)

Vinayak G M (20GACSE080)

ABSTRACT

Academic Records Management System is a simple and user-friendly web application Developed using HTML, CSS, JS, PHP. The Project is based on the concept of managing all the academic records of the college. Talking about the system, a user has to pass through a login system to use its features. A user can register himself/herself by entering his/her details such as name, email and department.

This project shall act as a centralised database for all academic records. Through this Interface, the user shall be able to view the available academic records. It is a simple user interface which has three types of users, namely students, faculty and alumni. The students and faculty shall be able to login into the system and add, view and modify all the records. The records can be publications, projects and patents. The interface provides a grid view which display all the details of the user and the records. An alumni can register and view all the records. The administrator can verify the details of the alumni. It serves as an online library for college publications. All the publications can be viewed by anyone from anywhere via the internet. The project is very helpful for academic purpose to all the students and research scholars.

CONTENTS

CHAPTER 1

INTRODUCTION

1.1 GENERAL OVERVIEW

Academic Records Management System is a simple and user-friendly web application developed using Html, CSS, JS, PHP. The Project is based on the concept of managing the academic records of the college. Talking about the system, a user has to pass through a login system to use its features. A user can register himself/herself by entering his/her details such as name, email and department.

The Project aims to build an academic records management system for an educational institution. The system acts as a centralized database for all the academic publications. The records here can be added and removed by the user. It serves as an online library for college publications. All the publications can be viewed by anyone from anywhere via the internet. It is helpful for academic purposes for students and researchers. An Academic Records Management System is a type of Content Management System(CMS).A CMS is software that helps users to create, manage and modify content on a website without the need for specialized technical knowledge. This system records the publications, patents and projects done by the college students and faculty members.

1.2 DATABASE MANAGEMENT SYSTEM (DBMS)

DBMS is a collection of programs that enables users to create and maintain a database The DBMS is a general-purpose software system that facilitates the processes of defining, constructing, manipulating and sharing databases among various users and applications.

A Relational database is a database that has a collection of tables of data items, all of which is formally described and organized according to the relational model. Data in a single table represents a relation, from which the name of the database type comes. In typical solutions, tables may have additionally defined relationships with each other. In the relational model, each table schema must identify a column or group of columns, called the primary key, to uniquely identify each row. A relationship can then be established between each row in the table and a row in another table by creating a foreign key, a column or group of columns in one table that points to the primary key of another table.

1.4.1 Characteristics of Database Management Systems

- Self-describing nature.
- Keeps a tight control on data redundancy.
- Enforces user defined rules to ensure that integrity of table data.
- Provides insulation between Programs and data, Data abstraction.
- Supports multiple views of the data.
- Helps sharing of data and Multi-user transaction processing.

1.4.2 Advantages of using DBMS approach

- Controlling the redundancy.
- Restricting unauthorized access.
- Providing persistent storage for program objects.
- Providing storage structures for efficient query processing.
- Providing backup and recovery.
- Providing multiple users interfaces
- Representing complex relationships among data

CHAPTER 2

LITERATURE REVIEW

2.1 EXISTING SYSTEM MODEL

The university students used to apply for intern positions manually. The interface was not developed. The system was complex as the students had to search for available positions with all the departments. All the applications need to be filed manually and lots of staff is required to process that information. Lots of documentation process is involved which increases cost of operations for university. As the existing system is manual there are chances of corruption and less transparency is there as information can be tempered by anyone during process leading to less reliable system.

2.2 PROPOSED SYSTEM MODEL

The tool is developed using PHP and MySQL Database, CSS, JS. This new interface shall enable the students to apply for internship. To use this system, the students should have registered with the database. Each student shall have a login id and password. The admin shall have the rights to create and manage these users from the back end. When the students log in to the system, a grid view shall be displayed which displays all the open job positions. It also gives the details about the start and end date. It gives a brief description about the job. Using the apply button the student shall be able to apply for the job. University can post multiple internship openings for students and they can apply for those online only which will make it very easy for students to apply for internships. It will be very simple and easy to use system for internship programs.

Advantages of Proposed System:

- It is trouble free to use.
- It is relatively fast approach to get internship.
- It is highly reliable, approximate result from user and has best user interface.

2.3 REQUIREMENTS SPECIFICATION

A software requirement definition is an abstract description of the services whichthe system should provide, and the constraints under which the system must operate. Itshould only specify the external behaviour of the system.

For the successful, efficient and problem free designing any project or program, the system should meet some requirements.

Modules

Student: This module shall have the login details of the student. They can login to their accounts to see the available internships available for them and apply based on their field.

Jobs: This module will have the details of the jobs like the name, description start and enddates etc. University staff can create new internship available for students and set thestart as well as end date.

Hardware Requirements

- Processor – Intel core Duo 2.0 GHz or more
- RAM - 512 MB or more
- Monitor
- Hard disk - 60 GB or more
- Mouse
- CD drive LG
- Keyboard Standard

Software configuration

- Operating System - Windows XP Professional
- Software – Xampp (for apache server)
- Database - MYSQL
- Front End - HTML5, CSS, JS
- Back End - PHP

2.4 STRUCTURED QUERY LANGUAGE (SQL)

SQL is a comprehensive database language. SQL is a standard language for storing, manipulating and retrieving data in databases. The ANSI standard SQL provides basic functions for data manipulation, transaction control, and record retrieval from the database. However, most end users interact with Oracle through application that provides an interface that hides the underlying SQL and its complexity. SQL uses the terms table, row, and column for relation, tuple, and attribute, respectively.

Applications of SQL:-

SQL is one of the most widely used query language over the databases that:

- Allows users to access data in the relational database management systems.
- Allows users to describe the data.
- Allows users to define the data in a database and manipulate that data.
- Allows to embed within other languages using SQL modules, libraries & pre-compilers.
- Allows users to create and drop databases and tables.
- Allows users to create view, stored procedure, functions in a database.

2.5 ENTITIES AND ATTRIBUTES

An entity may be an object with a physical existence (for e.g. A particular person, car or employee) or it may be an object with a conceptual existence (e.g. company, university).

Each entity has attributes i.e., the particular properties that describe it. The attribute values that describe each entity become a major part of the data store in the database. Whenever an attribute of one entity type refers to another entity type, a relationship exists. In the initial design of entity types, relationships are typically captured in the form of attributes. As the design is refined these attributes get converted into relationships between entity types. In the ER diagrams, the emphasis is on representing the schemas rather than the instances. This is more useful in the database design because a database schema changes rarely, whereas contents of the entity sets change frequently.

Types of Attributes

Simple attribute: An attribute is classified as a simple attribute if it cannot be partitioned into smaller components. For example, age and sex of a person. A simple attribute is represented by an oval.

Composite attribute: A composite attribute can be subdivided into smaller components which further form attributes. For example, 'name' attribute of an entity "person" can be broken down into first name and last name which further form attributes. Grouping of these related attributes forms a composite attribute.

Single valued attribute: If an attribute of a particular entity represents single value for each instance, then it is called a single-valued attribute. For example, Ramesh, Kamal and Suraj are the instances of entity ‘student’ and each of them isissued a separate roll number. A single oval is used to represent this attribute.

Multi valued attribute: An attribute which can hold more than one value, it isthen termed as multi-valued attribute. For example, phone number of a person.

Derived attribute: A derived attribute calculates its value from another attribute. For example, 'age' is a derived attribute if it calculates its value from 'current date' and 'birth date' attributes. A derived attribute is represented by a dashed oval.

Stored attribute: In some cases, two or more attribute values are related forexample, the Age and Birth_date attributes of a person. For a particular person entity, the value of Age can be determined from the current date and the value of the person's Birth_date. The Age attribute hence called derived attribute and is said to be derivable from the Birth_date attribute, which is called a Stored attribute.

Complex attribute: It is a Composite and Multivalued attribute.

2.6 RELATIONSHIPS

A relationship is defined as bond or attachment between 2 or more entities. Normally, a verb ina sentence signifies a relationship.

For example: An employee assigned a project.

Teacher teaches a student.

Author writes a book.

A diamond is used to symbolically represent a relationship in the ERdiagram.

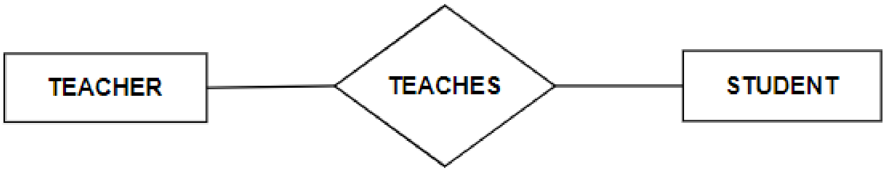


Figure: Relationship connecting two Entities.

Various terms related to relationships:

Degree of relationship: It signifies the number of entities involved in arelationship.

Degree of a relationship can be classified into following types:

Unary relationship: If only single entity is involved in a relationship then it is a unary relationship. For example, An employee (manager) supervises another employee.

Binary relationships: When two entities are associated to form a relation, then it is known as a Binary relationship. For example, A person works in a company. Most of the times we use only binary relationship in an ER diagram.

Other types of relationships are ternary and quaternary: As the name signifies a ternary relationship is associated with three entities and a quaternary relationship is associated with four entities.

Connectivity of a relationship: Connectivity of a relationship describes, how many instances of one entity type are linked to how many instances of another entity type. Various categories of connectivity of a relationship are:

One to One (1:1)-"Restaurant has menu" signifies a one-to-one relationship because only one instance of an entity is related with exactly one instance of another entitytype.

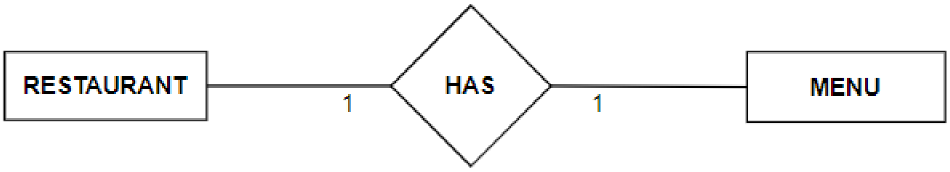


Figure: One to One Relationship

One to Many (1:M) - "A Customer places Order" is a one-to-many relationshipbecause a customer can place more than one order, but a order is related to onlyone Customer.

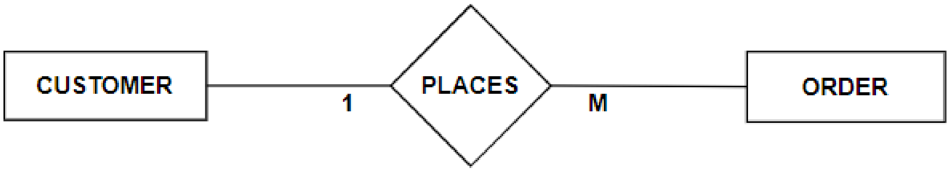


Figure: One to Many Relationship

Many to One (M:1) - "Menu has items" is a many-to-one relationship.

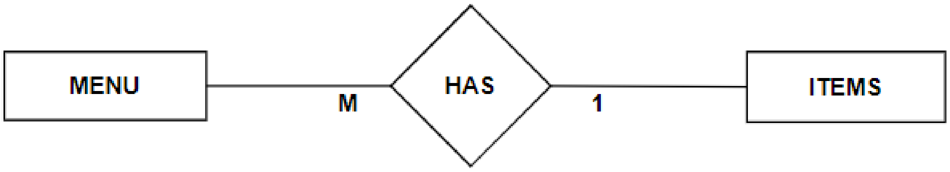


Figure: Many to One Relationship

Many to Many (M: N)-"Author writes books" is a many-to-many relationship because an author can write many books and a book can be written by many authors.

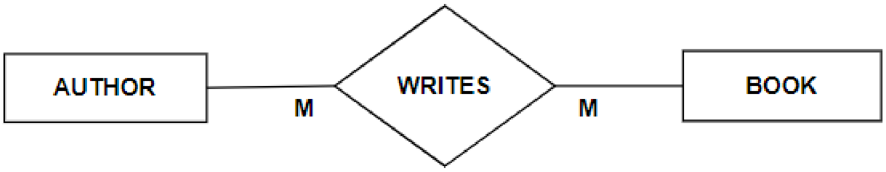


Figure: Many to Many Relationship

Total Participation and Partial Participation

Total Participation: It specifies that each entity in entity set must compulsorily participate in at least one relationship instance in that relationship set. It is also called as Mandatory Participation. The total participation is represented using a double line between the entity set and the relationship set.

Partial Participation: It specifies that each entity in entity set may or may not participate in the relationship instance in that relationship set. It is also called as Optional Participation. The partial participation is represented using a single line between the entity set and the relationship set.

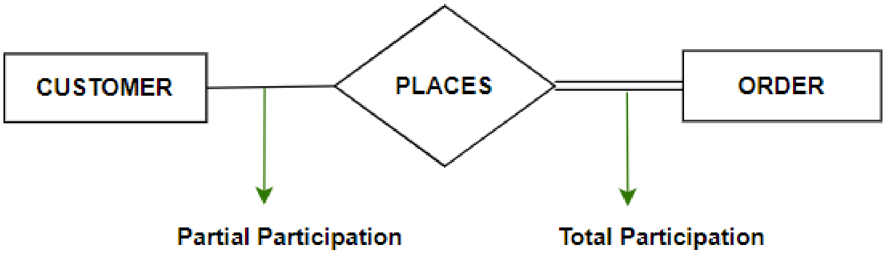


Figure: Total and Partial Participations

Types of Entities

Strong entity: A strong entity has a primary key attribute which uniquely identifies each entity. Symbol of strong entity is same as an entity.

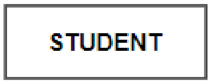


Figure: Strong Entity

Weak entity: A weak entity does not have a primary key attribute and depends on other entity via a foreign key attribute.

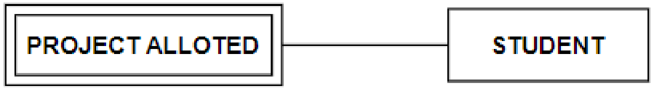


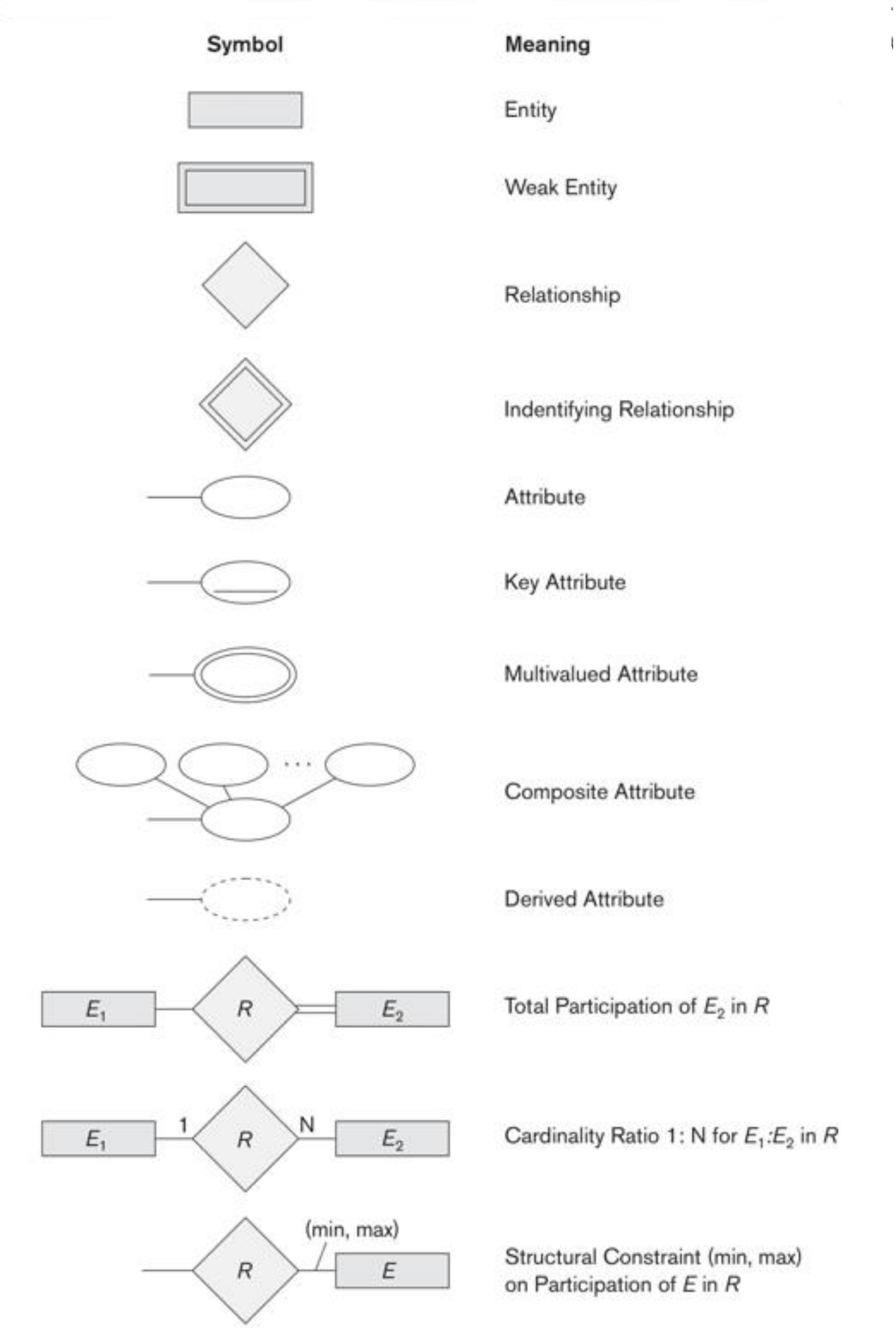
Figure: Weak Entity

2.7 ER DIAGRAM

An Entity-relationship model (ER model) describes the structure of a database with the help of a diagram, which is known as Entity Relationship Diagram (ER Diagram). An ER model is a design or blueprint of a database that can later be implemented as a database. The main components of ER model are:

- Entity set.
- Relationship set.

An ER diagram shows the relationship among entity sets. An entity set is a group ofsimilar entities and these entities can have attributes. In terms of DBMS, an entity is a table or attribute of a table in database, so by showing relationship among tables and their attributes, ER diagram shows the complete logical structure of a database. An Entity-Relationship (ER) model is an abstract way to describe a database.


Symbol
Meaning
Entity
Weak Entity
Relationship
Indentifying Relationship
Attribute
Key Attribute
Multivalued Attribute
. . .
Composite Attribute
Derived Attribute
E_1
R
E_2
Total Participation of E_2 in R
E_1
1
R
N
E_2
Cardinality Ratio 1: N for E_1:E_2 in R
(min, max)
R
E
Structural Constraint (min, max) on Participation of E in R

Notations for ER Diagram[1]

2.8 RELATIONAL MAPPING

A relational schema for a database is an outline of how data is organized. It can be a graphic illustration or another kind of chart used by programmers to understand how each table is laid out, including the columns and the types of data they hold and how tables connect. A database schema usually specifies which columnsare primary keys in tables and which other columns have special constraints such as being required to have unique values in each record. In this section we describe the steps of an algorithm for ER to relational mapping. The above ER diagram is used to derive the respective relational schema. The mapping will create tables with simple single valued attributes. The steps are:

- Mapping of Regular Entity Types.
- Mapping of Weak Entity Types.
- Mapping of Binary 1: 1 Relationship Types.
- Mapping of Binary 1: N Relationship Types.
- Mapping of Binary M: N Relationship Types.
- Mapping of Multivalued Attributes.
- Mapping of N-ary Relationship Types.

Mapping of 1: N Relationship type:

For each regular binary 1:N relationship type R, identify the relation S that represents the participating entity type at the N-side of the relationship type. Include as the foreign key in S the primary key of relation T that represents other entity type in R. We have to do this because each entity instance on N side is related to at most one entity instance on 1-sideof relationship type.

Mapping of M: N Relationship type:

For each min relationship type R, create a new relation S to represent R. Include as foreign key attributes in S the primary keys of the relations that represent the participating entity type, their combination will form the primary key of S. Also include any simple attributesof m: n relationship type.

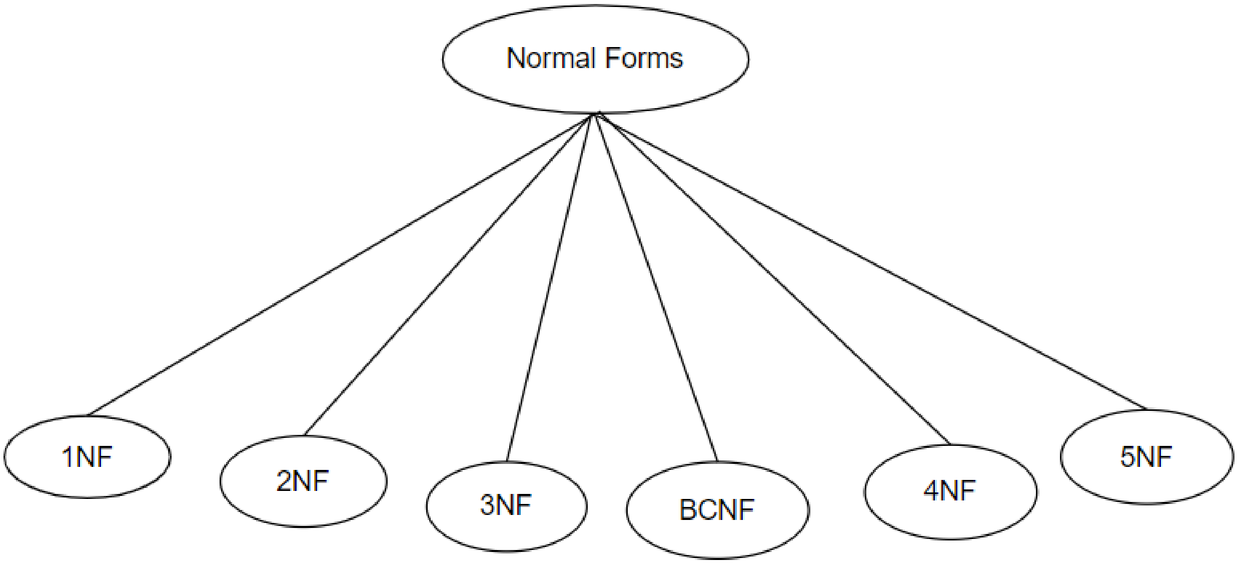
Mapping of 1:1 Relationship type:

For each 1:1 relationship type R can be migrated to any participating entity types. This relationship type ensures that each user in the database can lodge one complaint

2.9 NORMALIZATION

- Normalization is the process of organizing the data in the database.
- Normalization is used to minimize the redundancy from a relation or set of relations.
- It is also used to eliminate the undesirable characteristics like Insertion,Update andDeletion Anomalies.
- Normalization divides the larger table into the smaller table and links themusing relationship.
- The normal form is used to reduce redundancy from the database table.

Types of Normal forms



First Normal Form (1NF)

For a table to be in the First Normal Form, it should follow the following 4 rules:

- It should only have single(atomic) valued attributes/columns.
- Values stored in a column should be of the same domain.
- All the columns in a table should have unique names.
- And the order in which data is stored, does not matter.

Second Normal Form (2NF)

For a table to be in the Second Normal Form,

- It should be in the First Normal form.
- It should not have Partial Dependency.

Third Normal Form (3NF)

A table is said to be in the Third Normal Form when,

- It is in the Second Normal form.
- It doesn't have Transitive Dependency.

Boyce and Codd Normal Form (BCNF)

Boyce and Codd Normal Form is a higher version of the Third Normal form. This form deals with certain type of anomaly that is not handled by 3NF. A 3NF table which does not have multiple overlapping candidate keys is said to be in BCNF. For a table to be in BCNF, following conditions must be satisfied:

1. R must be in 3rd Normal Form
2. For each functional dependency (X→Y), X should be a super Key.

2.10 TOOLS AND ENVIRONMENTS USED

Our whole system is designed via PHP and MySQL database. This is web-based application so if we host it in some live web server then only system requirement will be internet and a device. Though the listed items are used to build this project:

Xampp

XAMPP is one of the widely used cross-platform web servers, which helps developers to create and test their programs on a local webserver. It was developed by the **Apache Friends,** and its native source code can be revised or modified by the audience. It consists of **Apache HTTP Server, MariaDB, and interpreter** for the different programming languages like PHP and Perl. It is available in 11 languages and supported by different platforms such as the IA-32 package of Windows and x64 package of macOS and Linux.

XAMPP is an abbreviation where *X stands for Cross-Platform, A stands for Apache, M stands for MYSQL and the P's stand for PHP and Perl,* respectively. It is an open-source package of web solutions that includes Apache distribution for many servers and command-line executables along with modules such as Apache server, MariaDB, PHP, and Perl.

XAMPP helps a local host or server to test its website and clients via computers and laptops before releasing it to the main server. It is a platform that furnishes a suitable environment to test and verify the working of projects based on Apache, Perl, MySQL database, and PHP through the system of the host itself. Among these technologies, Perl is a programming language used for web development, PHP is a backend scripting language, and MariaDB is the most vividly used database developed by MySQL. The detailed description of these components is given below.

Many other components are also part of this collection of software and are explained below.

Cross-Platform: Different local systems have different configurations of operating systems installed in it. The component of cross-platform has been included to increase the utility and audience for this package of Apache distributions. It supports various platforms such as packages of Windows, Linus, and MAC OS.

Apache: It is an HTTP a cross-platform web server. It is used worldwide for delivering web content. The server application has made free for installation and used for the community of developers under the aegis of Apache Software Foundation. The remote server of Apache delivers the requested files, images, and other documents to the user.

MariaDB: Originally, MySQL DBMS was a part of XAMPP, but now it has been replaced by MariaDB. It is one of the most widely used relational DBMS, developed by MySQL It offers online services of data storage, manipulation, retrieval, arrangement, and deletion.

PHP: It is the backend scripting language primarily used for web development. PHP allows users to create dynamic websites and applications. It can be installed on every platform and supports a variety of database management systems. It was implemented using C language. PHP stands for

Hypertext Processor. It is said to be derived from Personal Home Page tools, which explains its simplicity and functionality.

Perl: It is a combination of two high-level dynamic languages, namely Perl 5 and Perl 6. Perl can be applied for finding solutions for problems based on system administration, web development, and networking. Perl allows its users to program dynamic web applications. It is very flexible and robust.

phpMyAdmin: It is a tool used for dealing with MariaDB. Administration of DBMS is its main role.

OpenSSL: It is the open-source implementation of the Secure Socket Layer Protocol and Transport Layer Protocol.

XAMPP Control Panel: It is a panel that helps to operate and regulate upon other components of the XAMPP. A detailed description of the control panel will be done in the next section of the tutorial.

Webalizer: It is a Web Analytics software solution used for User logs and provide details about the usage.

Mercury: It is a mail transport system. It is a mail server, which helps to manage the mails across the web.

Tomcat: It is a servlet based on JAVA to provide JAVA functionalities.

Filezilla: It is a File Transfer Protocol Server, which supports and eases the transfer operations performed on files.

MySQL

MySQL is a fast, easy-to-use RDBMS being used for many small and big businesses. MySQL is developed, marketed and supported by MySQL AB, which is a Swedish company. MySQL is becoming so popular because of many good reasons-

- MySQL is released under an open-source license. So you have nothing to pay to use it.
- MySQL is a very powerful program in its own right. It handles a large subset of the functionality of the most expensive and powerful database packages.
- MySQL uses a standard form of the well-known SQL data language.
- MySQL works on many operating systems and with many languagesincluding PHP, PERL, C, C++, JAVA, etc.
- MySQL works very quickly and works well even with large data sets.
- MySQL is very friendly to PHP, the most appreciated language for webdevelopment.
- MySQL supports large databases, up to 50 million rows or more in a table. The default file size limit for a table is 4GB, but you can increase this (if your operating system can handle it) to a theoretical limit of 8 million terabytes (TB).
- MySQL is customizable. The open-source GPL license allows programmers to modify the MySQL software to fit their own specific environments.

PHP

The PHP Hypertext Preprocessor (PHP) is a programming language that allows web developers to create dynamic content that interacts with databases. PHP isbasically used for developing web based software applications. This tutorial helps you to build your base with PHP.PHP started out as a small opensource project that evolved as more and more people found outhow useful it was. Rasmus Lerdorf unleashed the first version of PHP way back in 1994.

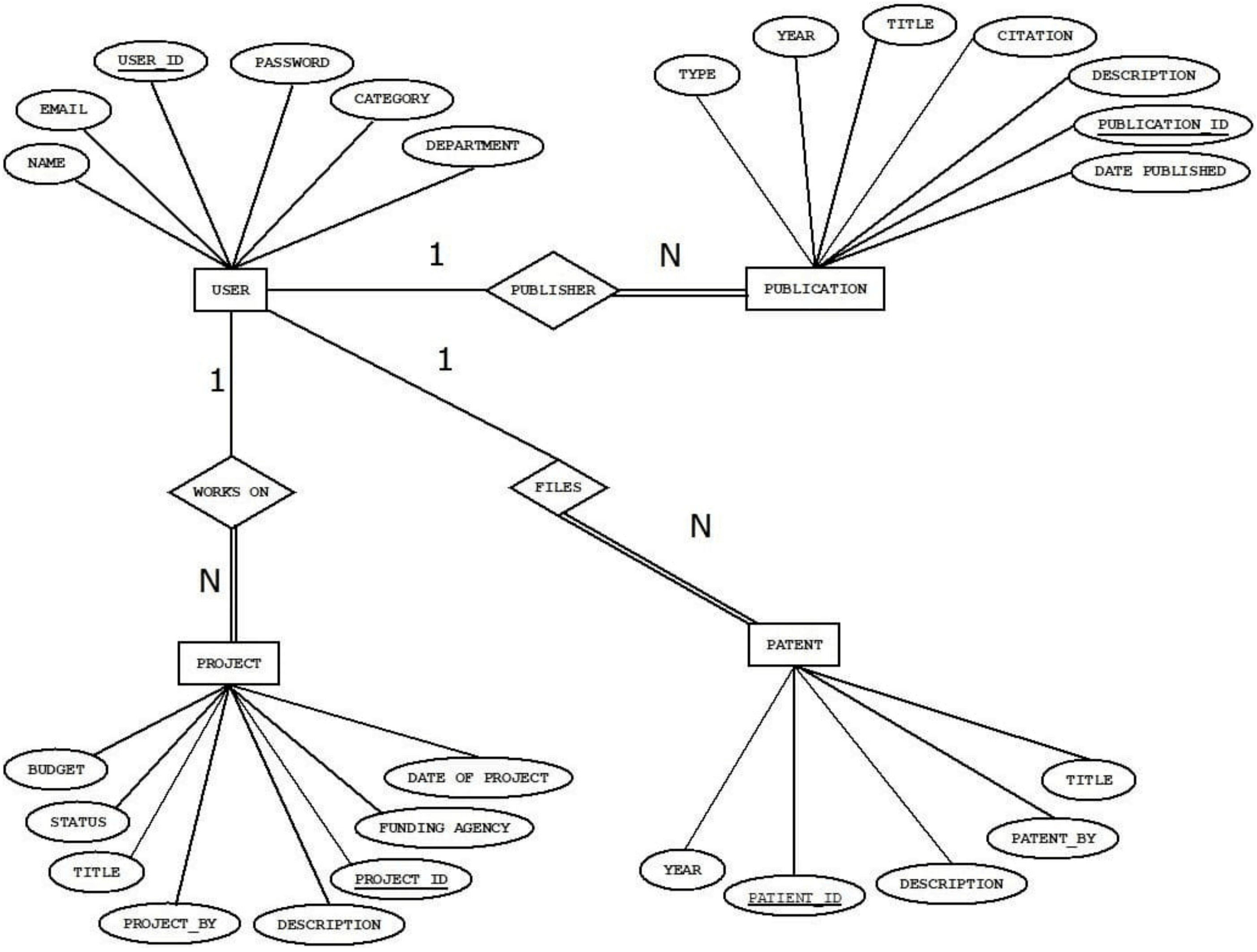
PHP is a must for students and working professionals to become a great Software Engineer specially when they are working in Web Development Domain. I will list down some of the key advantages of learning PHP:

- PHP is a recursive acronym for "PHP: Hypertext Preprocessor".
- PHP is a server side scripting language that is embedded in HTML. It is usedto manage dynamic content, databases, session tracking, even build entire e-commerce sites.
- It is integrated with a number of popular databases, including MySQL, PostgreSQL, Oracle, Sybase, Informix, and Microsoft SQL Server.
- PHP is pleasingly zippy in its execution, especially when compiled as an Apache module on the Unix side. The MySQL server, once started, execute seven very complex queries with huge result sets in record-setting time.
- PHP supports a large number of major protocols such as POP3, IMAP, and LDAP. PHP4 added support for Java and distributed object architectures (COM and CORBA), making n-tier development a possibility for the first time.
- PHP is forgiving: PHP language tries to be as forgiving as possible.
- PHP Syntax is C-Like.

CHAPTER 3

PROPOSED WORK

3.1 ER MODEL



3.2 RELATIONAL MODEL

REGISTRATION

s_no	email	otp	timestamp

USER

user_id	name	email	password	category	department

PATENT

user_id	patent_id	title	description	year	patent_by

PUBLICATION

user_id	publication_id	title	type	year	author	description	date_published	citation

PROJECT

user_id	project_id	title	project_by	description	date_of_project	budget	status	funding_agency

ALUMNI

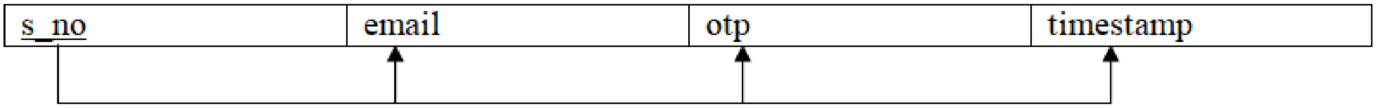
user_id	name	college_id	department	batch	phno	company	designation	verified

3.3 NORMALIZATION

The normalization process was proposed by Codd, it takes a relation schema through a series of tests to certify whether if satisfies a certain normal form. The process proceeds in a top-down fashion by evaluating each relation against the criteria for normal forms and decomposing relations as necessary can thus be considered a relational design by analysis. Normalization of data can be looked upon as a process of analyzing the given relation schemas based on their Functional Dependencies and primary keys to achieve the desirable properties of:

- Minimizing redundancy.
- Minimizing the insertion, deletion, and update anomalies.

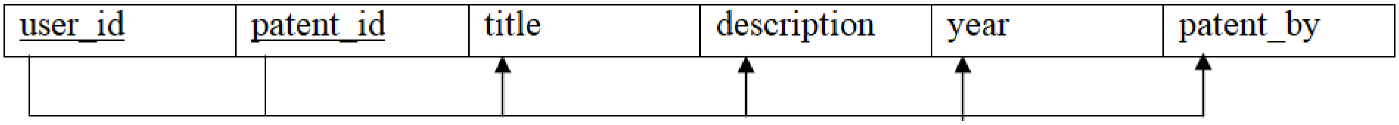
REGISTRATION



FD1:{s_no}{email, otp, timestamp}

- The registration relation is in 1NF because there are no multivalued attributes in the relational schema.
- It is in 2NF because all the attributes in the relational schema are fully functional dependent on the primary key.
- It is in 3NF because there are no transitive dependencies.

PATENT

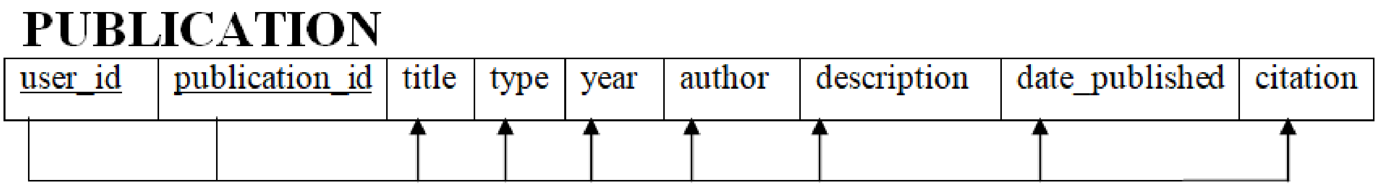


FD1:{user_id, patent_id}{title, description, year, patent_by}

- The patent relation is in 1NF because there are no multivalued attributes in the relational schema.

- It is in 2NF because all the attributes in the relational schema are fully functional dependent on the primary key.

- It is in 3NF because there are no transitive dependencies.



PUBLICATION

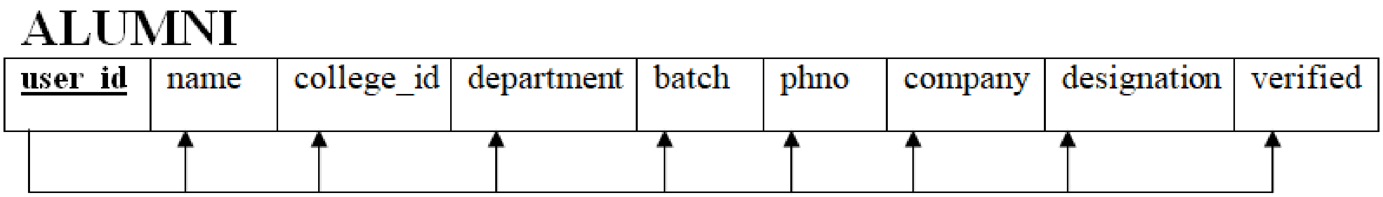
user_id	publication_id	title	type	year	author	description	date_published	citation

FD1:{user_id, publication_id}{title, type , year, author, description, date_published, citation}

- The publication relation is in 1NF because there are no multivalued attributes in the relational schema.

- It is in 2NF because all the attributes in the relational schema are fully functional dependent on the primary key.

- It is in 3NF because there are no transitive dependencies.



ALUMNI

user_id	name	college_id	department	batch	phno	company	designation	verified

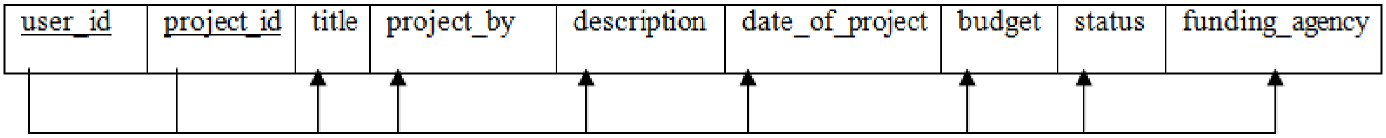
FD1:{user_id}{name,college_id, department, batch, phno, company, designation, verified}

- The alumni relation is in 1NF because there are no multivalued attributes in the relational schema.

- It is in 2NF because all the attributes in the relational schema are fully functional dependent on the primary key.

- It is in 3NF because there are no transitive dependencies.

PROJECT



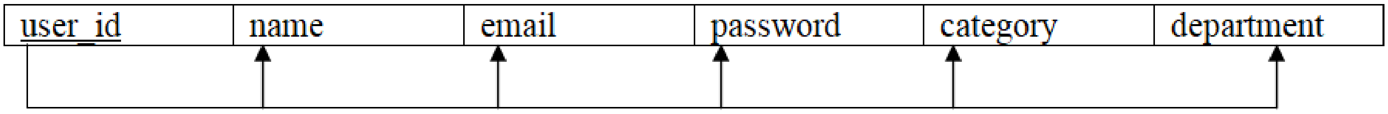
FD1:{project_id, user_id}{title, project_by, description, date_of_project, budget, status, funding_agency, user_id}

- The project relation is in 1NF because there are no multivalued attributes in the relational schema.

- It is in 2NF because all the attributes in the relational schema are fully functional dependent on the primary key.

- It is in 3NF because there are no transitive dependencies.

USER



FD1:{user_id}{name, email, password, category, department}

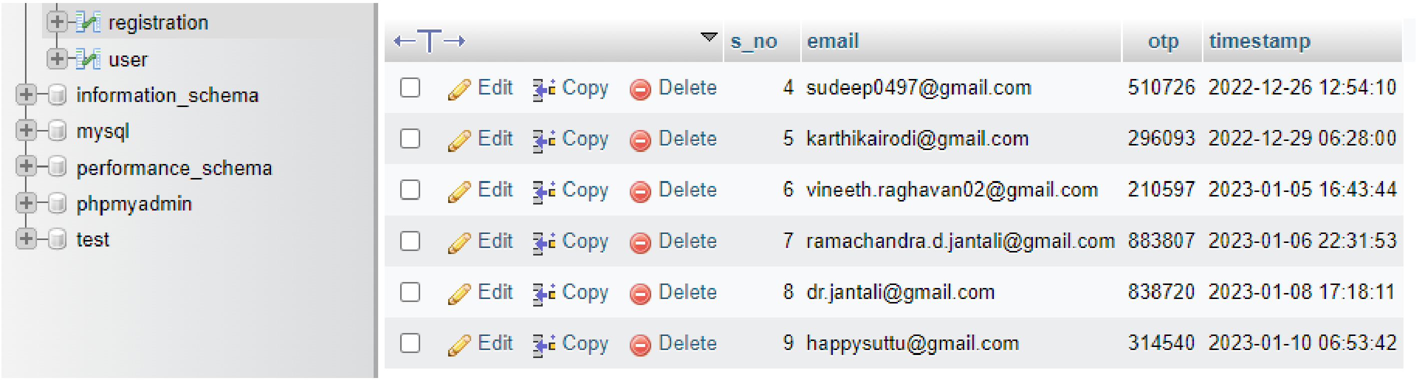
- The user relation is in 1NF because there are no multivalued attributes in the relational schema.

- It is in 2NF because all the attributes in the relational schema are fully functional dependent on the primary key.

- It is in 3NF because there are no transitive dependencies.

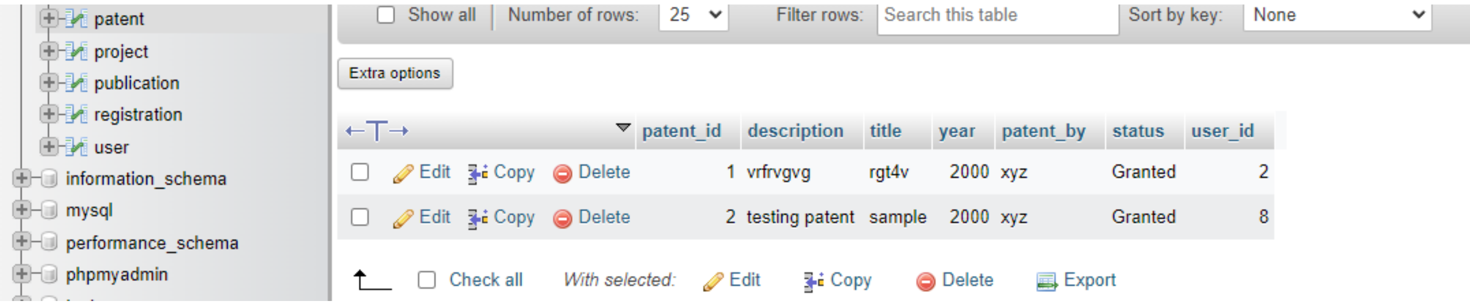
3.4 DESCRIPTION OF TABLES

Registration



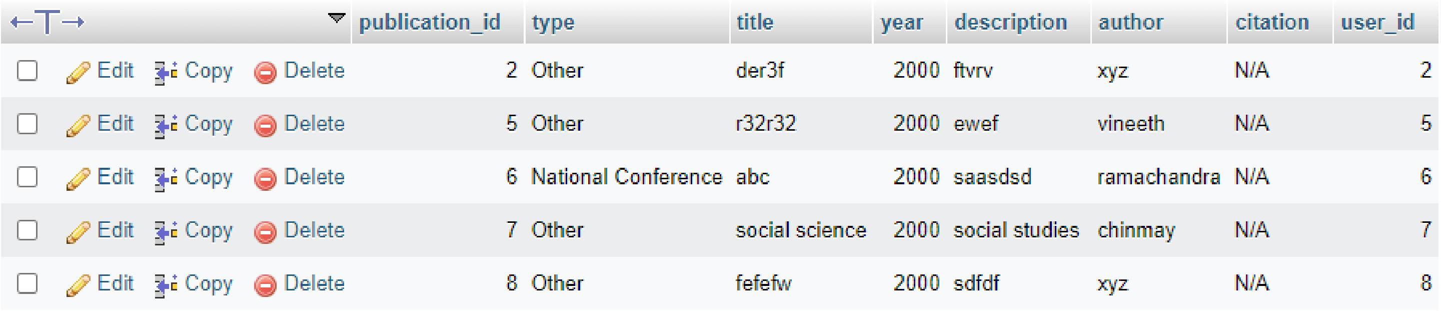
←T→	s_no	email	otp	timestamp
□ Edit Copy Delete	4	sudeep0497@gmail.com	510726	2022-12-26 12:54:10
□ Edit Copy Delete	5	karthikairodi@gmail.com	296093	2022-12-29 06:28:00
□ Edit Copy Delete	6	vineeth.raghavan02@gmail.com	210597	2023-01-05 16:43:44
□ Edit Copy Delete	7	ramachandra.d.jantali@gmail.com	883807	2023-01-06 22:31:53
□ Edit Copy Delete	8	dr.jantali@gmail.com	838720	2023-01-08 17:18:11
□ Edit Copy Delete	9	happysuttu@gmail.com	314540	2023-01-10 06:53:42

Patent



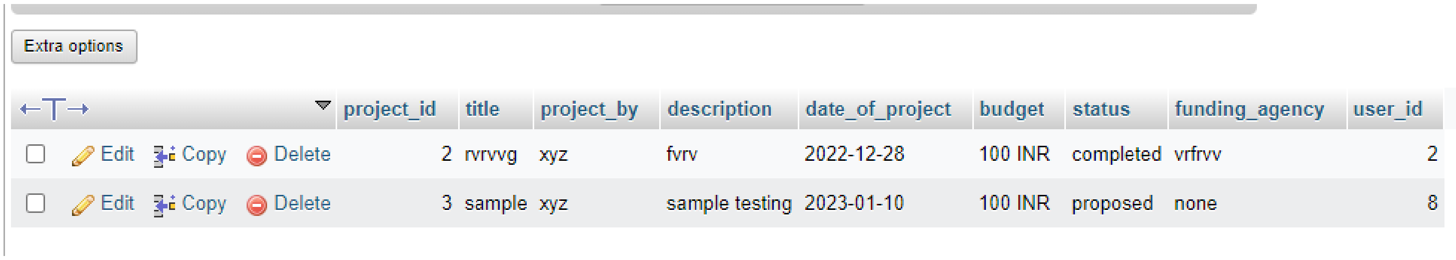
←T→	patent_id	description	title	year	patent_by	status	user_id
□ Edit Copy Delete	1	vrfrvgvg	rgt4v	2000	xyz	Granted	2
□ Edit Copy Delete	2	testing patent	sample	2000	xyz	Granted	8

Publication



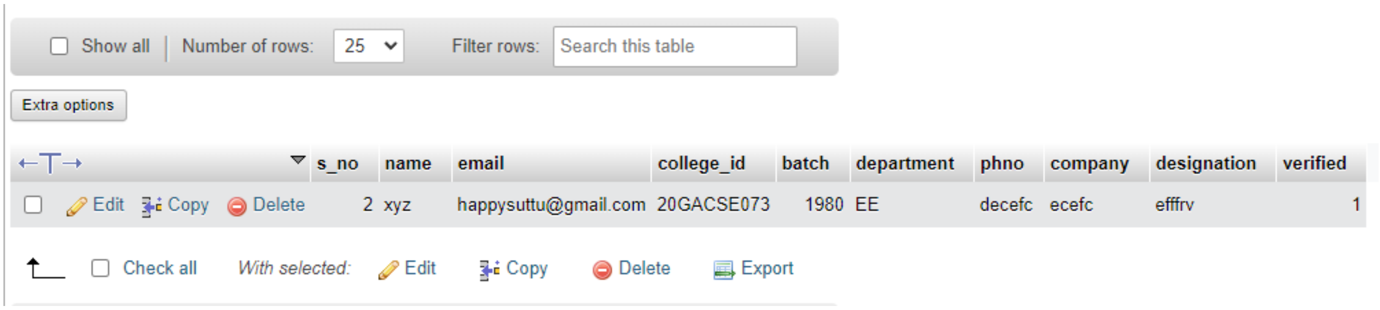
←T→	publication_id	type	title	year	description	author	citation	user_id
□ Edit Copy Delete	2	Other	der3f	2000	ftvrv	xyz	N/A	2
□ Edit Copy Delete	5	Other	r32r32	2000	ewef	vineeth	N/A	5
□ Edit Copy Delete	6	National Conference	abc	2000	saasdsd	ramachandra	N/A	6
□ Edit Copy Delete	7	Other	social science	2000	social studies	chinmay	N/A	7
□ Edit Copy Delete	8	Other	fefefw	2000	sdfdf	xyz	N/A	8

Project



←T→ ▼	project_id	title	project_by	description	date_of_project	budget	status	funding_agency	user_id
□ Edit Copy Delete	2	rvrvvg	xyz	fvrv	2022-12-28	100 INR	completed	vrfrvv	2
□ Edit Copy Delete	3	sample	xyz	sample testing	2023-01-10	100 INR	proposed	none	8

Alumni

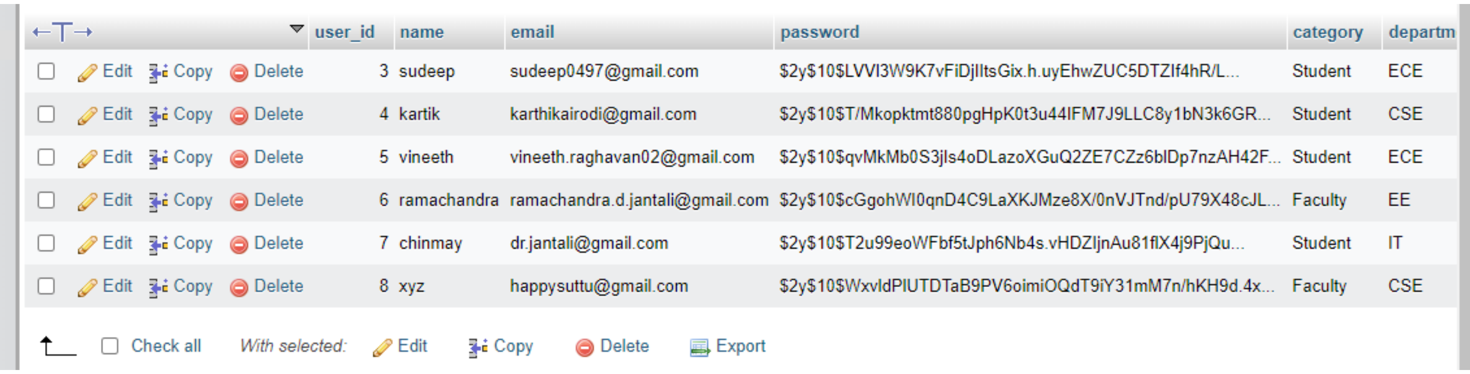


□ Show all | Number of rows: 25 Filter rows: Search this table
Extra options

←T→ ▼	s_no	name	email	college_id	batch	department	phno	company	designation	verified
□ Edit Copy Delete	2	xyz	happysuttu@gmail.com	20GACSE073	1980	EE	decefc	ecefc	efffrv	1

□ Check all With selected: Edit Copy Delete Export

User



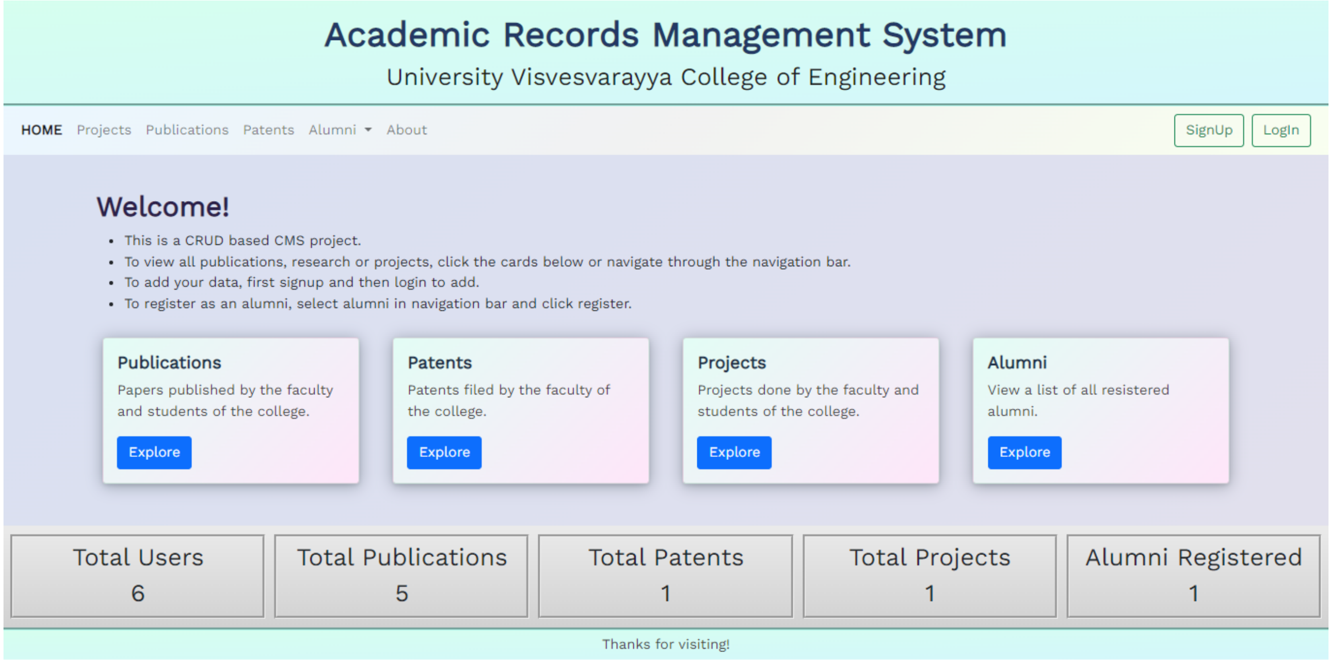
←T→ ▼	user_id	name	email	password	category	departm
□ Edit Copy Delete	3	sudeep	sudeep0497@gmail.com	$2y$10$LVVI3W9K7vFiDjIltsGix.h.uyEhwZUC5DTZIf4hR/L...	Student	ECE
□ Edit Copy Delete	4	kartik	karthikairodi@gmail.com	$2y$10$T/Mkopktmt880pgHpK0t3u44IFM7J9LLC8y1bN3k6GR...	Student	CSE
□ Edit Copy Delete	5	vineeth	vineeth.raghavan02@gmail.com	$2y$10$qvMkMb0S3jls4oDLazoXGuQ2ZE7CZz6blDp7nzAH42F...	Student	ECE
□ Edit Copy Delete	6	ramachandra	ramachandra.d.jantali@gmail.com	$2y$10$cGgohWI0qnD4C9LaXKJMze8X/0nVJTnd/pU79X48cJL...	Faculty	EE
□ Edit Copy Delete	7	chinmay	dr.jantali@gmail.com	$2y$10$T2u99eoWFbf5tJph6Nb4s.vHDZljnAu81flX4j9PjQu...	Student	IT
□ Edit Copy Delete	8	xyz	happysuttu@gmail.com	$2y$10$WxvldPlUTDTaB9PV6oimiOQdT9iY31mM7n/hKH9d.4x...	Faculty	CSE

CHAPTER 4

RESULT

The results of the project are shown in form of snapshots below. Each table describes a part of output of project when application is run. A Snapshot is the state of the system at a particular point in time. It can refer to the actual copy of a state of a system.

SNAPSHOTS OF PROJECT


Academic Records Management System

University Visvesvarayya College of Engineering

Welcome!

- This is a CRUD based CMS project.
- To view all publications, research or projects, click the cards below or navigate through the navigation bar.
- To add your data, first signup and then login to add.
- To register as an alumni, select alumni in navigation bar and click register.

Publications

Papers published by the faculty and students of the college.

Patents

Patents filed by the faculty of the college.

Projects

Projects done by the faculty and students of the college.

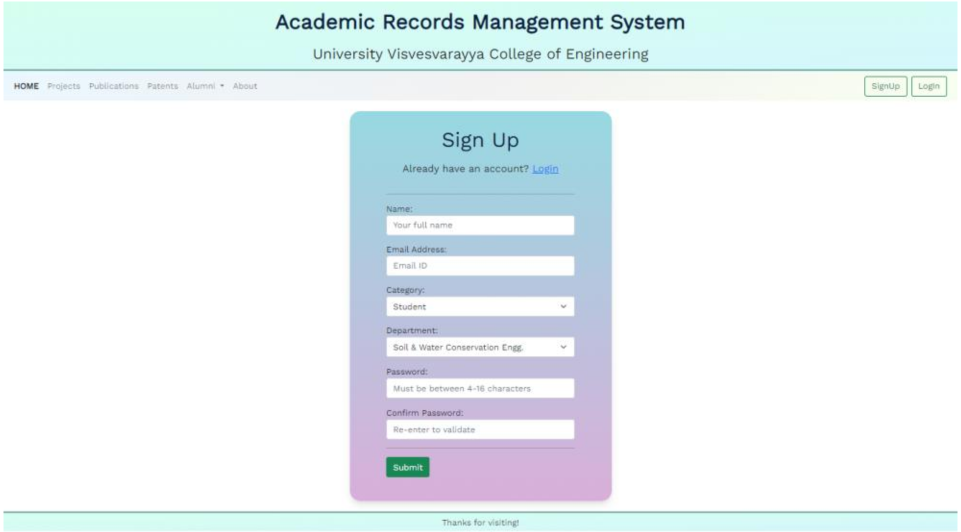
Alumni

View a list of all resistered alumni.

Total Users	Total Publications	Total Patents	Total Projects	Alumni Registered
6	5	1	1	1

Thanks for visiting!

4.1 Introduction Page of the Project


Academic Records Management System

University Visvesvarayya College of Engineering

Sign Up

Already have an account? Login

Name: Your full name

Email Address: Email ID

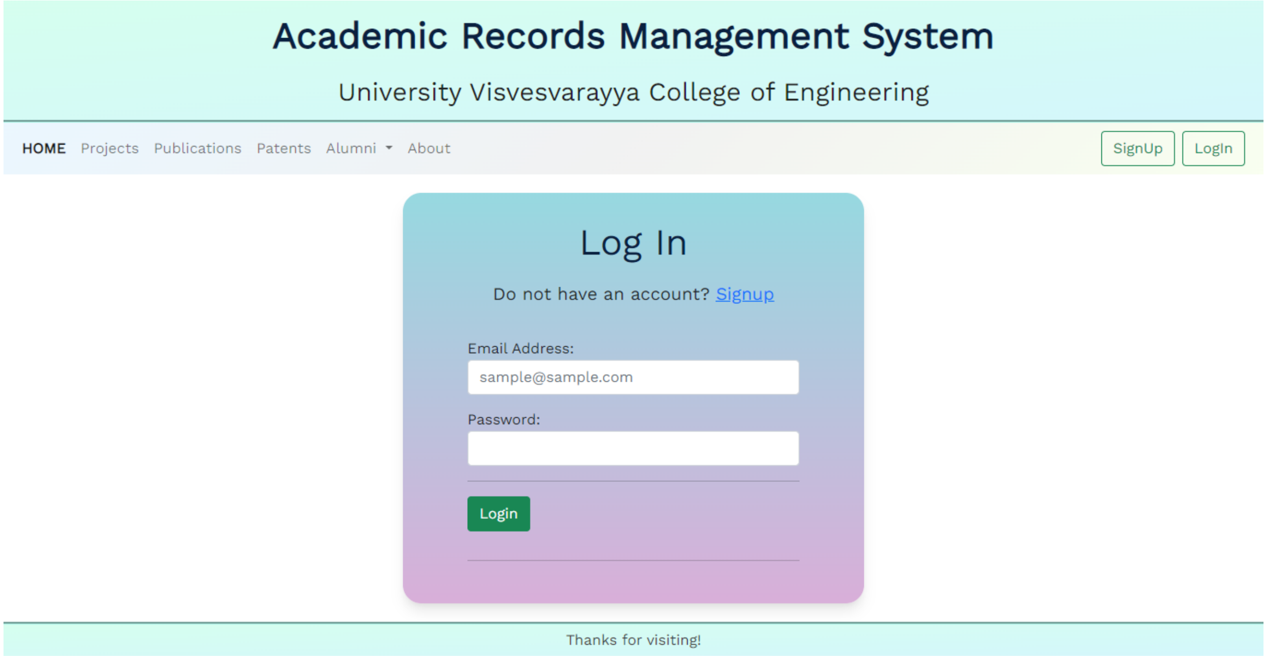
Category: Student

Department: Soil & Water Conservation Engg.

Password: Must be between 4-16 characters

Confirm Password: Re-enter to validate

Thanks for visiting!



Academic Records Management System

University Visvesvarayya College of Engineering

Log In

Do not have an account? Signup

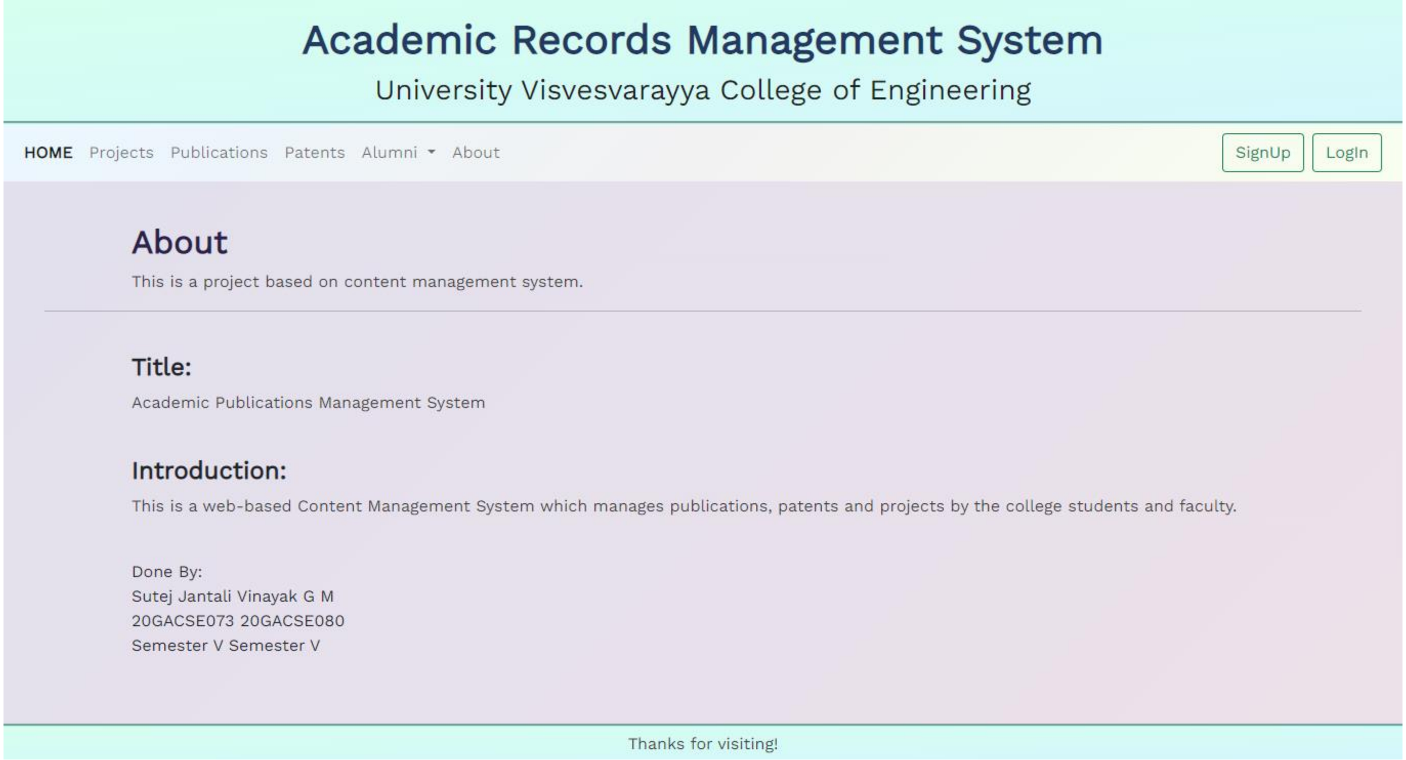
Email Address: sample@sample.com

Password:

Thanks for visiting!

4.2

Sign Up page & Log in page



Academic Records Management System

University Visvesvarayya College of Engineering

About

This is a project based on content management system.

Title:

Academic Publications Management System

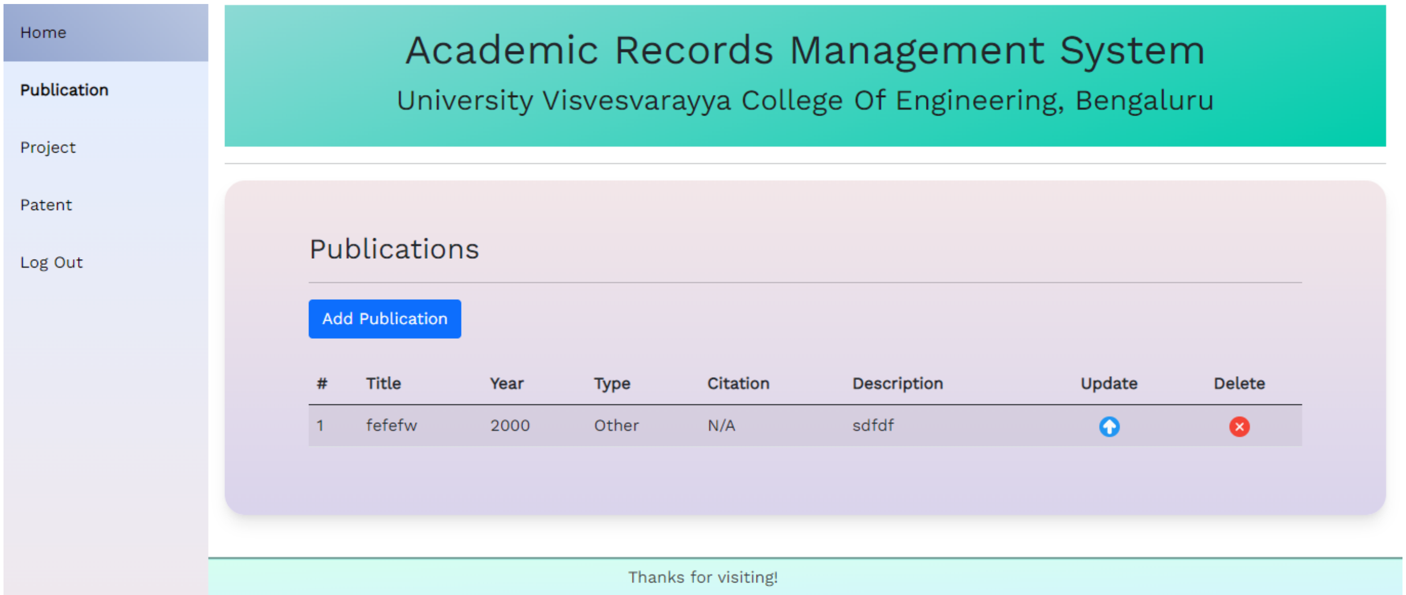
Introduction:

This is a web-based Content Management System which manages publications, patents and projects by the college students and faculty.

Done By:
Sutej Jantali Vinayak G M
20GACSE073 20GACSE080
Semester V Semester V

Thanks for visiting!

4.3 About Us Page



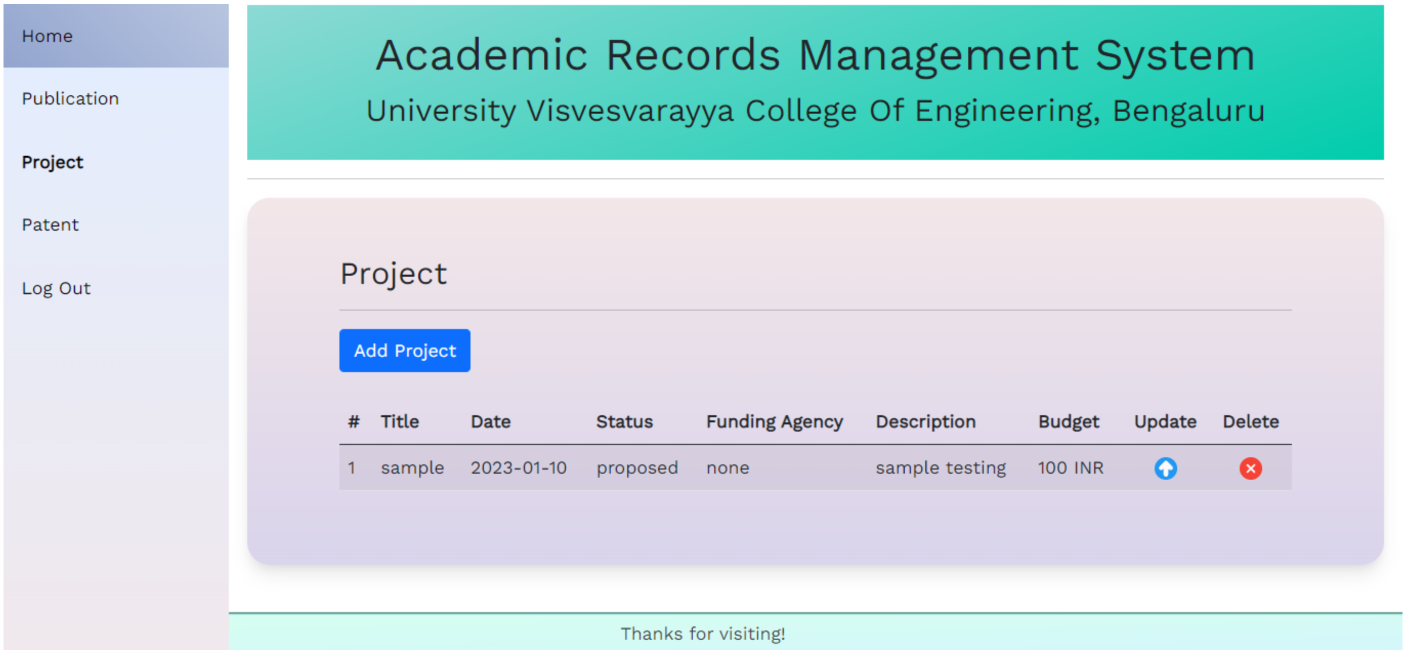
Academic Records Management System

University Visvesvarayya College Of Engineering, Bengaluru

Publications

#	Title	Year	Type	Citation	Description	Update	Delete
1	fefefw	2000	Other	N/A	sdfdf	⬆	✖

4.4 Publications page



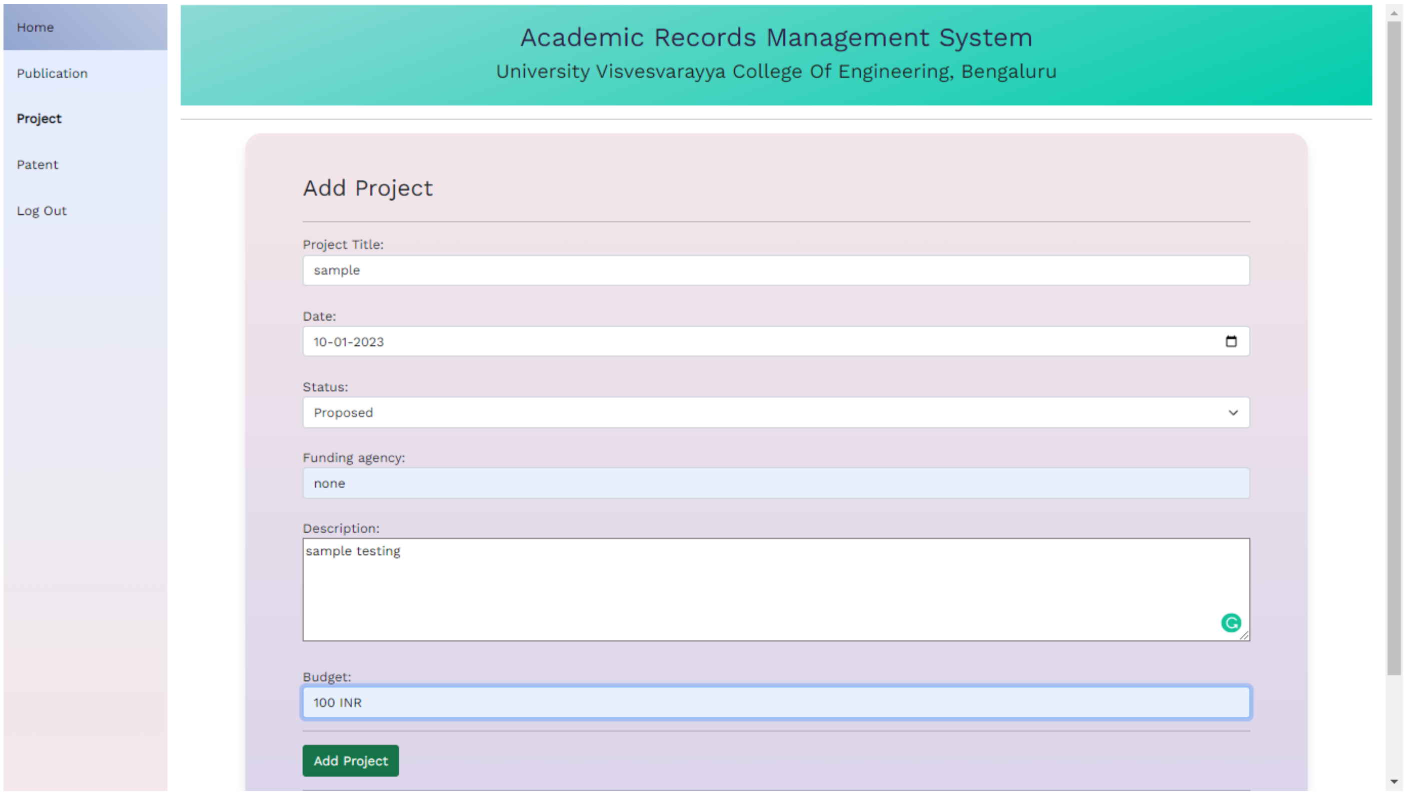
Academic Records Management System

University Visvesvarayya College Of Engineering, Bengaluru

Project

#	Title	Date	Status	Funding Agency	Description	Budget	Update	Delete
1	sample	2023-01-10	proposed	none	sample testing	100 INR	⬆	✖

4.5 Project page



Academic Records Management System

University Visvesvarayya College Of Engineering, Bengaluru

Add Project

Project Title: sample

Date: 10-01-2023

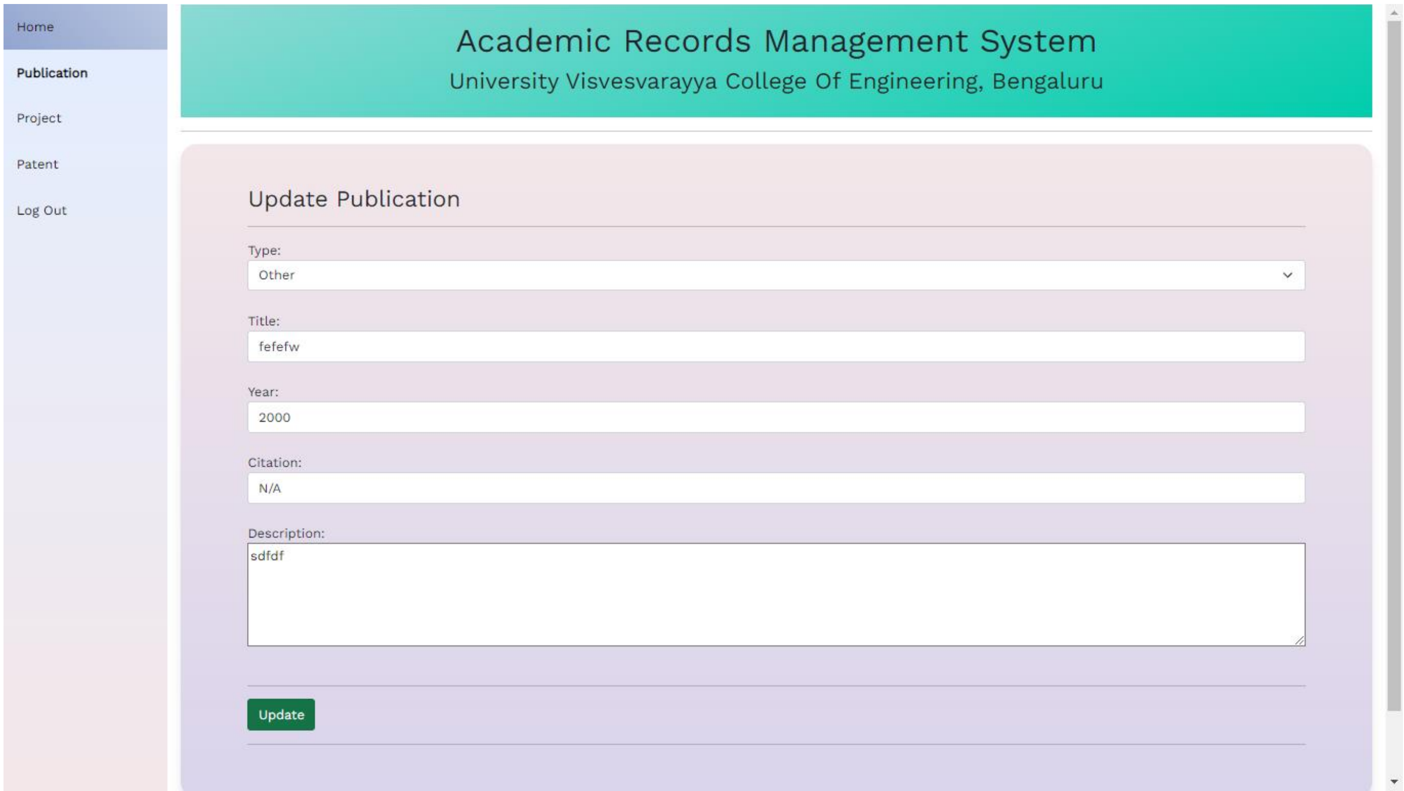
Status: Proposed

Funding agency: none

Description: sample testing

Budget: 100 INR

4.6 Adding Project



Academic Records Management System

University Visvesvarayya College Of Engineering, Bengaluru

Update Publication

Type: Other

Title: fefefw

Year: 2000

Citation: N/A

Description: sdfdf

4.7 Updating an existing Publication

CHAPTER 5

CONCLUSION

The Academic Records Management System is a web-based application that has high integrity to make process of accessing academic records easier and user-friendly. It is going to be the need and optimum option for comfortable experience in future. This project is carried out to analyse the internet applications on database management through the use of MySQL and the features of PHP and the way it can be used and interfaced with MySQL as a standalone database application. This one is useful for students and research scholars for academic purpose. This portal reduces a lot of complexity involved in searching all different types of records.

This application acts as an online library. Instead of keeping lot of paperwork and maintaining different records manually, the portal is helpful for the maintainance of a single platform for all the academic records.

FUTURE ENHANCEMENTS:

1. To improve the interface of the application.
2. To make application more interactive and user friendly
3. To add more different types of records like research papers and seminar videos
4. To add more detailed description about the records.

CHAPTER 6

BIBLIOGRAPHY

Books:

[1] Ramez Elmasri and Shamkant B Navathe, ‘Fundamental of Database Systems’, Addison Wesley, 2011, Sixth Edition.

[2] Abraham Silberschatz, Henry F. Korth, S.Sudarshan, ‘Database System Concepts’, Tata McGraw-Hill, 2010,Sixth Edition.

[3] Robin Nixon, ‘Learning PHP, MySQL, JavaScript, CSS & HTML5 – A Step By Step Guide To Creating Dynamic Websites’, Third Edition.

Websites:

https://www.javatpoint.com/dbms-tutorial

https://www.geeksforgeeks.org/dbms/?ref=lbp

https://www.techtarget.com/searchdatamanagement/definition/database-management-system

https://www.w3schools.com/html/

https://developer.mozilla.org/en-US/docs/Web/HTML

https://www.w3.org/Style/CSS/Overview.en.html

https://www.javascript.com/

https://en.wikipedia.org/wiki/PHP