CHAPTER:1
INTRODUCTION

1.1 OVERVIEW

The "Criteria 6 Automation System" has been developed to override the problems prevailing in the practicing manual system. This software is supported to eliminate and in some cases reduce the hardships faced by this existing system. Moreover this system is designed for the particular need of the company to carry out operations in a smooth and effective manner.

The website is reduced as much as possible to avoid errors while entering the data. It also provides error message while entering invalid data. No formal knowledge is needed for the user to use this system. Thus by this all it proves it is user-friendly. Criteria 6 Automation System, as described above, can lead to error free, secure, reliable and fast management system. It can assist the user to concentrate on their other activities rather to concentrate on the record keeping. Thus it will help organization in better utilization of resources.

1.2 PROBLEM STATEMENT

The Criteria 6 Automation system need to keep track of details of items, staff details. Hence its tremendous pressure maintaining their day to day activities, which is done manually. Entire records have to be up- dated timely even a sight could complicate things. The Criteria 6 Automation system is a complete solution for organizations, which need to manage items and staff details. Organize and track of items details provided by coverage

1.3 EXISTING SYSTEMS

The old manual system was suffering from a series of drawbacks. Since whole of the system was to be maintained with hands the process of keeping, maintaining and retrieving the information was very tedious and lengthy. The records were never used to be in a systematic order. There would always be unnecessary consumption of time while entering records and retrieving records. Once the records were entered it was very difficult to update these records. The reason behind it is that there is lot of information to be maintained. For this reason, we have provided features Present system is partially automated (computerized).

1.4 PROPOSED SYSTEMS

The aim of proposed system is to develop a system of improved facilities. The proposed system can over- come all the limitations of the existing system. The system provides proper security and reduces the manual work.

- Security of data.
- Ensure data accuracies.
- Proper control of the higher officials.
- Minimize manual data entry.
- Minimum time needed for the various processing.
- Greater efficiency.
- Better service.
- User friendliness and interactive.
- Minimum time required.

1.5 ADVANTAGES

- Faster System.
- Reliability.
- Informative.
- Easy updating of data.

CHAPTER:2

SOFTWARE REQUIREMENT

2.1 SOFTWARE USED

Operating systems: Windows 10 or 11.

Front end: HTML, CSS, PHP & Bootstrap.

Back end: Phyton, Django.

IDE: Xampp & Visual Studio Code

2.2 SOFTWARE DESCRIPTION

2.2.1 XAMPP(PhpMyAdmin)

XAMPP is a free and open-source cross-platform web server solution stack package developed by Apache Friends, consisting mainly of the Apache HTTP Server, MariaDB database, and interpreters for scripts writ- ten in the PHP and Perl programming languages. 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 suit- able environment to test and verify the working of projects based on Apache, Perl, MySQL database, and PHP through the system of the host itself.

2.2.2 THE PHYTON LANGUAGE

The Python programming language has powerful features for database programming. Python supports various databases like SQLite, MySQL, Oracle, Sybase, PostgreSQL, etc. Python also supports Data Definition Language (DDL), Data Manipulation Language (DML) and Data Query Statements. The Python standard for database interfaces is the Python DB-API. Most Python database interfaces adhere to this standard.

2.2.3 HTML

To publish information for global distribution, one needs a university-understood language, a kind of pub- lishing mother tongue that all computers may potentially understand, The publishing language used by the World Wide Web is HTML (Hyper Text Markup Language)

- Publish online documents with headings, text, tables, list, photos etc.
- Retrieve online information via hypertext links, at the click of a button.
- Design forms for conducting transactions with remote services, for use in searching information, making reservation, ordering products etc.;
- Includes spreadsheets, video clips, sound clips, and other applications directly in the documents.

2.2.3 CSS

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language such as HTML/XML CSS is designed to enable the separation of content and presentation, including layout, colours, and fonts. This separation can improve content accessibility; provide more flexibility and control in the specification of presentation characteristics; enable multiple web pages to share formatting by specifying the relevant CSS in a separate. CSS file, which reduces complexity and repetition in the structural content; and enable the. CSS file to be cached to improve the page load speed between the pages that share the file and its formatting.

- CSS is the language we use to style an HTML document.
- CSS describes how HTML elements should be displayed.
- More code means slower page speed.
- And CSS enables you to use less code.
- CSS allows you to use one CSS rule and apply it to all occurrences of a certain tag within an HTML document

2.3 SOFTWARE IMPLEMENTATION

Fig 2.3.1 (SDLC)

SDLC stands for the software development life cycle. It is a process used design, develop and test high- quality software

The system implementation for the development of the “CRITERIA-6 AUTOMATION" system would involve the following:

1. Firstly, the home page for the website needed to be made. The home page contained a navbar that would contain the links to services and a stock page.
2. The services part of the home page would contain all the services provided by the admin. And the stocks page contains page provides the user to view the items that are inserted or moved from one room to another room.
3. In addition to that navbar would also provide us with login and signup models. This login and signup take email, username, and password and is linked to the database in the backend.
4. After login the page displays the items list that are used to insert or move from one room to another room
5. Then we have to insert the items or move the items from one room to the another room by specifying the correct room number

   For ex: we have to add item=Fan to room no=223 then the page displays the items list there we have to select Fan and add it to the room no 223 and Also we can move the item=Fan from room no=224 to room no=234 etc;

   To verify:
   Check the stock list page there we can view the result of operation that we have performed etc;

CHAPTER:3

SYSTEM ANALYSIS AND DESIGN

3.1 SYSTEM ANALYSIS

System analysis is a detailed of the various operations performed by a system and their relationship within and outside the system. It is a systematic technique that defines goal and objectives. The goal of system development is to develop a system in line with the user requirement, and analysis of the system plays important role. One of the main aspects of analysis is the defining the boundaries of the system.

The various tools of structured analysis are

- Entity relationship diagram
- Table
- Table description
- Flow diagram

The structured analysis has the following attributes

- Entity relationship diagram (ER-Diagram) is a graphical representation of entities and their relationship to each other.
- Typically used in computing in regards to the organization of data within data base. Here, we are explaining the system, modules and their relationship using ER-Diagram.
- The Data Flow Diagram (DFD) presents a picture of what is being specified and is conceptually easy to understand presentation of the application.
- Table is the logical model of storing data with its attribute and data type.

ER Diagram Symbols & Notations

- Rectangles: This Entity Relationship Diagram symbol represents entity type.
- Ellipse: Symbol represent attributes
- Diamonds: This symbol represents relation types
- Lines: It links attributes to entity types and entity types with other relation
- Primary key: attributes are underlined
- Double Ellipse: Represent multi-valued attribute

3.2 ER DIAGRAM

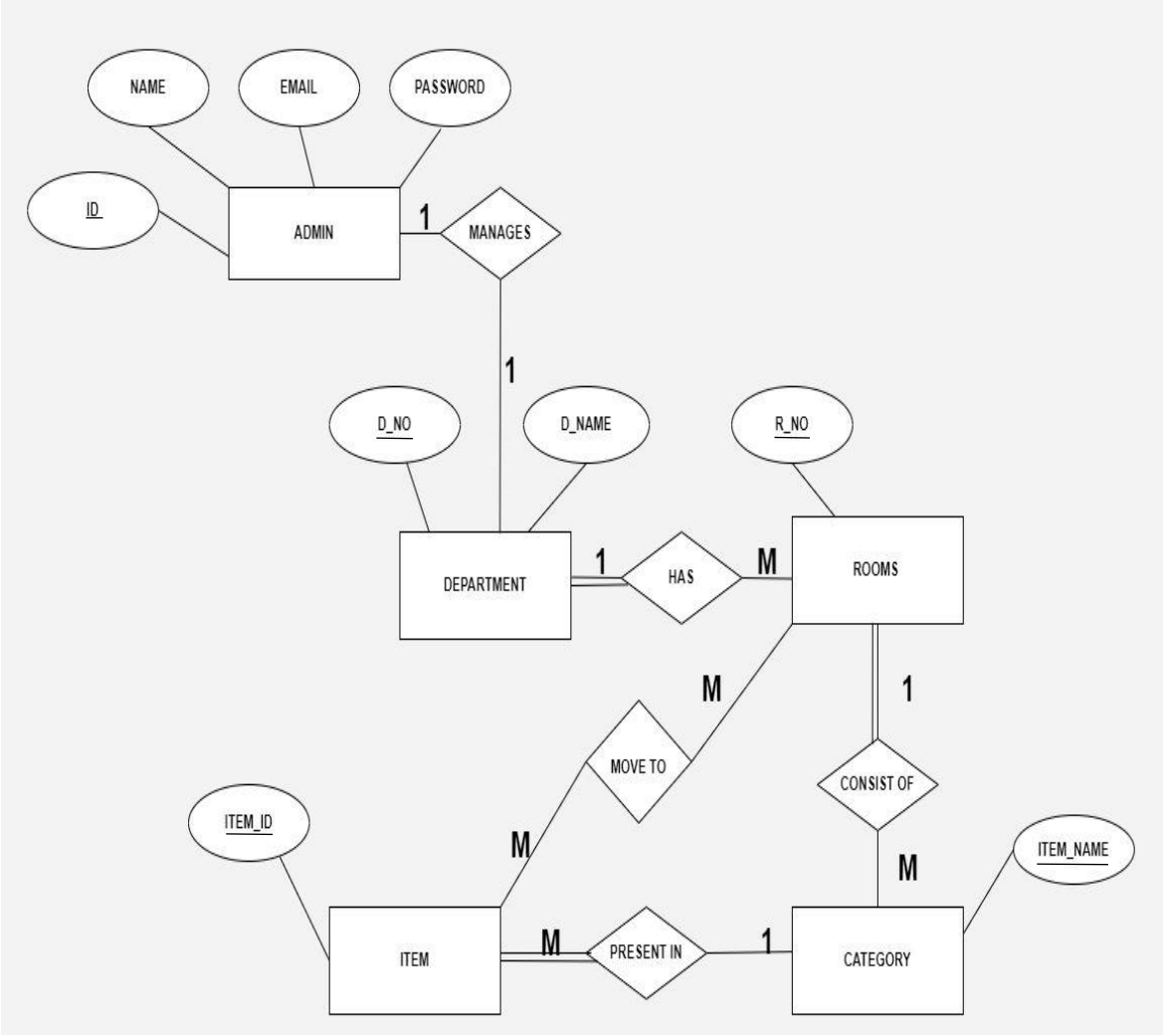


Fig: Entity relationship diagram

An ER model describes the relations between entities. It is adapted to represent relational data. Data that cannot easily be put into this form should not be modelled with an ER model. ER is aimed at designing a database from scratch; using it to describe and change a database that already exists may be more difficult to do.

Even where it is suitable in principle, ER diagram is rarely used as a separate activity. One reason for this is that there are many tools that allow diagramming and that have other design support directly on relational database management system. These tools can extract database diagrams that are very close to ER diagrams from existing databases, and they provide alternative views on the information contained in such diagrams.

3.2.1 ER MAPPING

Step 1: Mapping of regular entity type. Step 2: Mapping of weak entity type.

Step 3: Mapping of is to 1 relationship types. There is no 1 is to 1 relationship types in the ER Diagram.

Step 4: Mapping of is to N relationship types.

Step 5: Mapping of M is to N relationship types.

Step 6: Mapping of multivalued attributes.

Step 7: Mapping of N-array Relationship types.

3.3 SCHEMA DIAGRAM

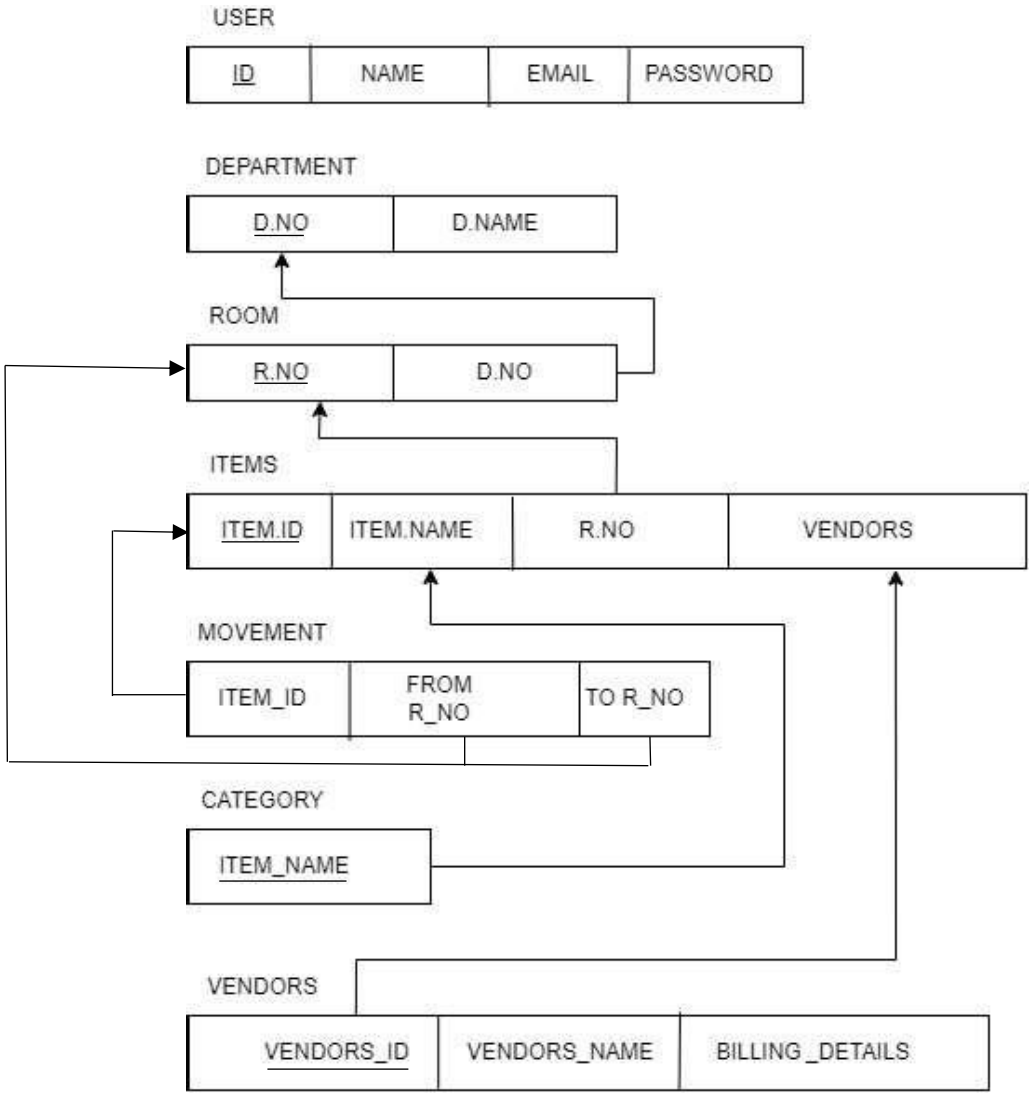


Fig: Schema diagram.

3.4 USE CASE DIAGRAM

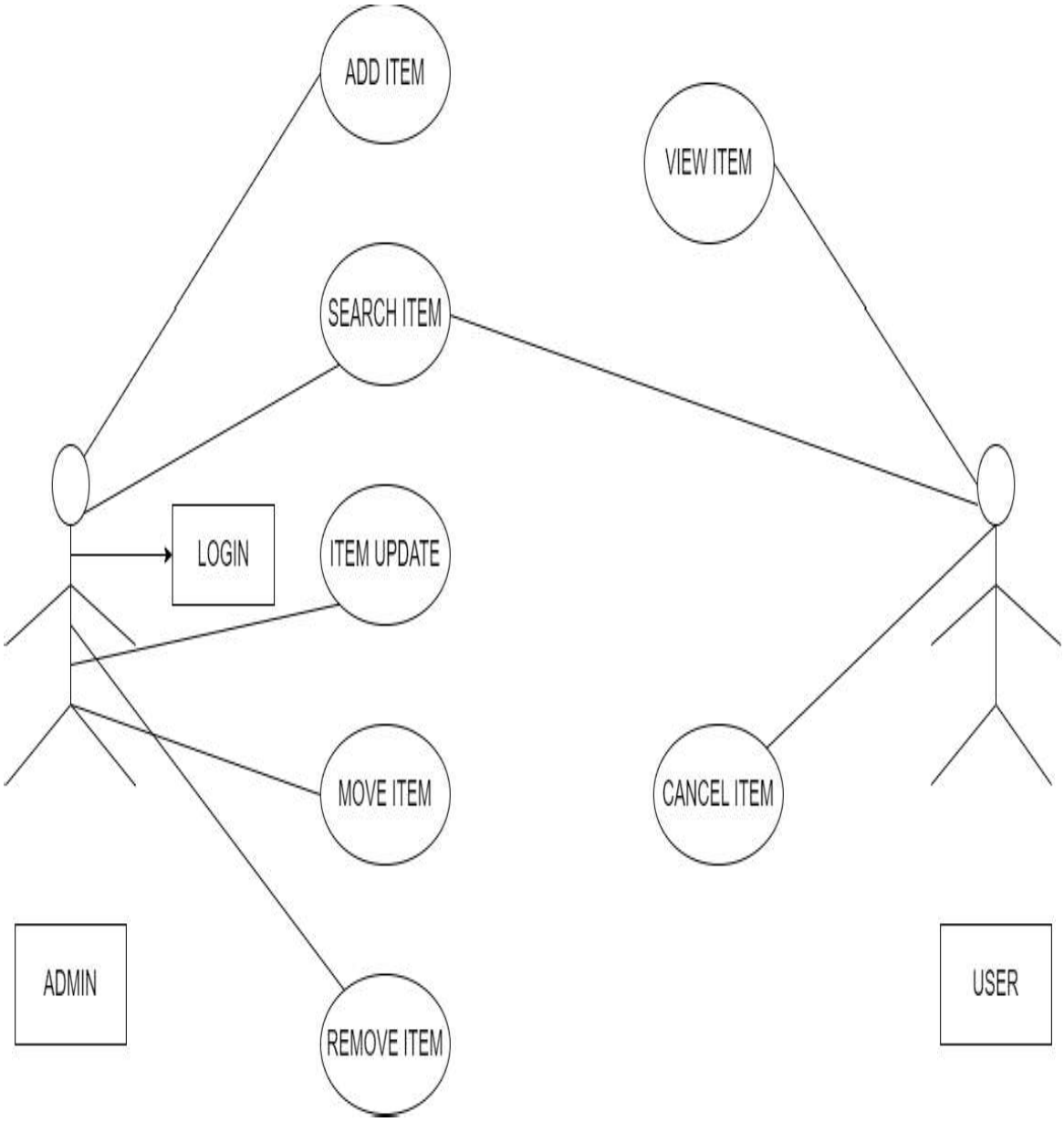


Fig: Use case diagram.

3.5 TABLES DESCRIPTION

- ## DESC USER

The “user” table provides information about the user.

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
☐	1	id	int(11)			No	*None*		AUTO_INCREMENT	Change Drop More
☐	2	name	varchar(50)	utf8mb4_general_ci		No	*None*			Change Drop More
☐	3	email	varchar(50)	utf8mb4_general_ci		No	*None*			Change Drop More
☐	4	password	varchar(1000)	utf8mb4_general_ci		No	*None*			Change Drop More

- ## DESC DEPT

The “dept” table provides information about the departments.

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
☐	1	dno	int(11)			No	*None*			Change Drop More
☐	2	dname	char(20)	utf8mb4_general_ci		Yes	*NULL*			Change Drop More

- ## DESC ROOM

The “room” table provides information about the present rooms.

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
☐	1	rno	varchar(30)	utf8mb4_general_ci		No	*None*			Change Drop More
☐	2	dno	int(11)			Yes	*NULL*			Change Drop More

- **DESC ITEMS**

The “items” table provides information about the items present in rooms.

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
☐	1	item_id	varchar(30)	utf8mb4_general_ci		No	*None*			Change Drop More
☐	2	item_name	varchar(20)	utf8mb4_general_ci		Yes	*NULL*			Change Drop More
☐	3	rno	varchar(30)	utf8mb4_general_ci		Yes	*NULL*			Change Drop More
☐	4	vid	varchar(20)	utf8mb4_general_ci		Yes	*NULL*			Change Drop More

- **DESC MOVEMENT**

The “movement” table provides information about items movement from one room to another.

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
☐	1	item_id	varchar(30)	utf8mb4_general_ci		Yes	*NULL*			Change Drop More
☐	2	from_rno	varchar(30)	utf8mb4_general_ci		Yes	*NULL*			Change Drop More
☐	3	to_rno	varchar(30)	utf8mb4_general_ci		Yes	*NULL*			Change Drop More

- **DESC CATEGORY**

The “category” table provides information about the types of items that are present.

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
☐	1	item_name	varchar(30)	utf8mb4_general_ci		No	*None*			Change Drop More

CHAPTER:4
DATABASE TECHNIQUE AND RESULTS

4.1 TRIGGER

Trigger is a stored program, which automatically executed or fixed when some events occur. Trigger are, in fact written to be executed in response to any of the following events.

In this project 3 triggers has been used.

Insert into logs values (null, NEW. item_ id, ' Inserted' , NOW())

Insert into logs values(null, OLD. item_ id, 'Updated', Now())

Insert into logs values(null, OLD. item_ id, ' Deleted', Now())

	Name	Time	Event			
☐	deleteLog	BEFORE	DELETE	Edit	Export	Drop
☐	insertLog	AFTER	INSERT	Edit	Export	Drop
☐	updateLog	AFTER	UPDATE	Edit	Export	Drop

		id	item_id	action	cdate
☐	Edit Copy Delete	1	0	Inserted	2023-01-30 11:24:03
☐	Edit Copy Delete	2	MITM/ISE/008/PC3000	Inserted	2023-01-30 11:26:19
☐	Edit Copy Delete	3	MITM/ISE/226/CCTV010	Updated	2023-01-30 11:30:15
☐	Edit Copy Delete	4	MITM/ISE/008/PC3000	Deleted	2023-01-30 11:33:34

4.2 ASSERTION

An assertion is a piece of SQL which makes sure a condition is satisfied, else or it stops the action being taken on a database.

An assertion is a constraint that might be dependent upon multiple rows of multiple tables.

Domain constraints, functional dependency and referential integrity are special forms of assertion are dependent (involve) on single row of a table a time.

Any modification to a database is allowed only if it would not cause any assertion are checked only when UPDATE or INSERT actions are performed against the table.

4.3 PROCEDURE

A procedure (often called a stored procedure) is a subroutine like a subprogram in a regular computing language, stored in database. There are many useful applications of SQL procedures within a database or database application architecture. SQL procedures can be used to create simple scripts for quickly querying transforming, updating data, generating basic reports, improve application performance, modularizing applications, and improve overall database design, and database security.

4.4 VIEWS

In a database, a view is the result set of a stored query on the data which the database users can query just as they would in persistent database collection object. This pre-established query command is kept in the database dictionary. Unlike ordinary base tables in a relational database, a does not form part of the physical schema: as a result set, it is a virtual table computed or collated dynamically from data in the database when access to that view is requested. Changes applied to the data in a relevant underlying table are reflected in the data shown in subsequent invocations of the view.

4.5 NORMALIZATION

The complete tables of the database in the project is normalized, obeying all the rules of normalization

1NF

1NF disallows relations within relations or relations as attribute values within tuples.

The only attribute values permitted by INF are single atomic (or indivisible) values.

2NF

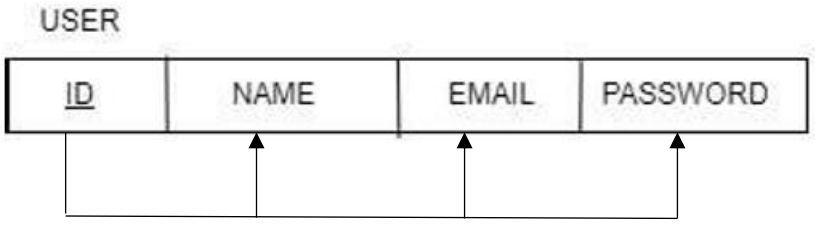
A functional dependency X Y is a full functional dependency if removal of any attribute A from X means that the dependency does not hold any more; that is, for any attribute A epsilon X, (X-{A}) does not functionally determine Y.

3NF

Transitive functional dependency

A functional dependency X Y in a relation schema R is a transitive dependency if there exists a set of attribute Z. that are neither a primary nor a subset of any key of R(candidate key) and both X Z and Y Z holds

Definition: A relation schema R is in third normal form (3NF) if it is in 2NF and no non prime attribute A in R is transitively dependent on the primary key.



1NF: In the above table there are no multi valued attributes. Thus, the functional dependency FDI and relation satisfies 1NF

2NF: There are no partial dependencies found in the above defined functional dependencies. Thus, we can say that relation satisfies 2NF.

3NF: There are no transitive dependencies found in the above defined functional dependencies. Thus, we can say that relation satisfies 3NF.

CHAPTER:5
RESULT ANALYSIS

5.1 TEST CASES

TEST CASE ID	TEST CASES	EXPECTED OUTPUT	ACTUAL OUTPUT	STATUS
1	**Login with wrong user id and wrong password**	**Invalid user id and password**	**As expected**	**Fail**
2	**Login with wrong user id and correct password**	**Invalid user id and password**	**As expected**	**Fail**
3	**Login with correct user id and wrong password**	**Invalid user id and password**	**As expected**	**Fail**
4	**Login with correct user id and correct password**	**Login successful**	**As expected**	**Pass**

5.2 SNAPSHOTS

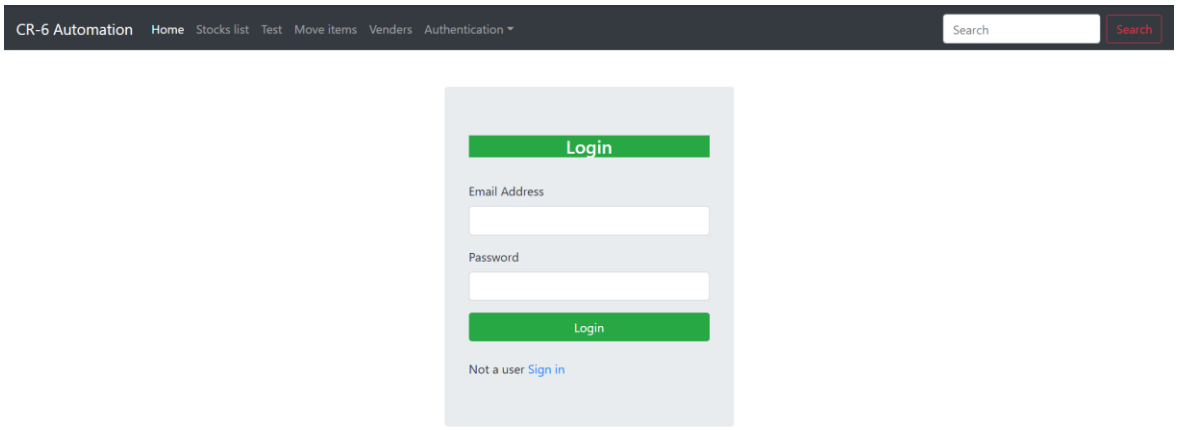


FIG: ADMIN LOGIN PAGE

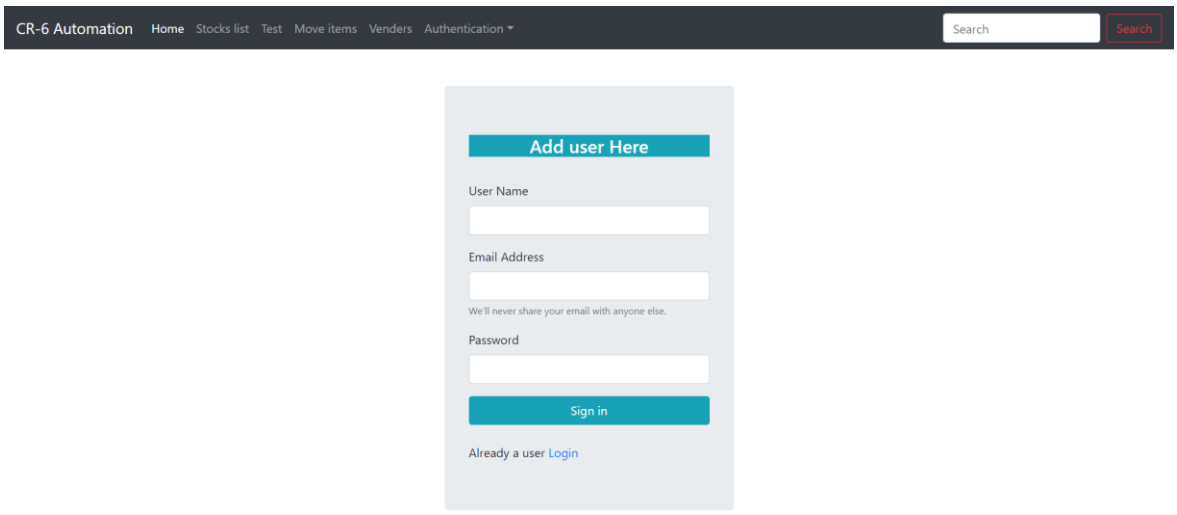


FIG: ADD USER PAGE

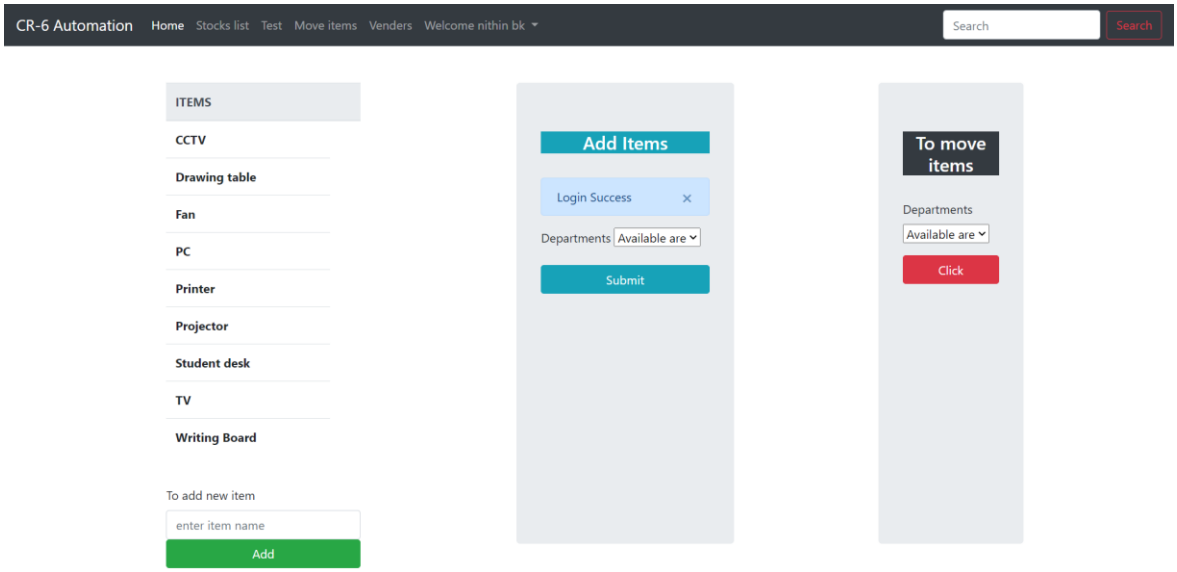


FIG: HOME PAGE

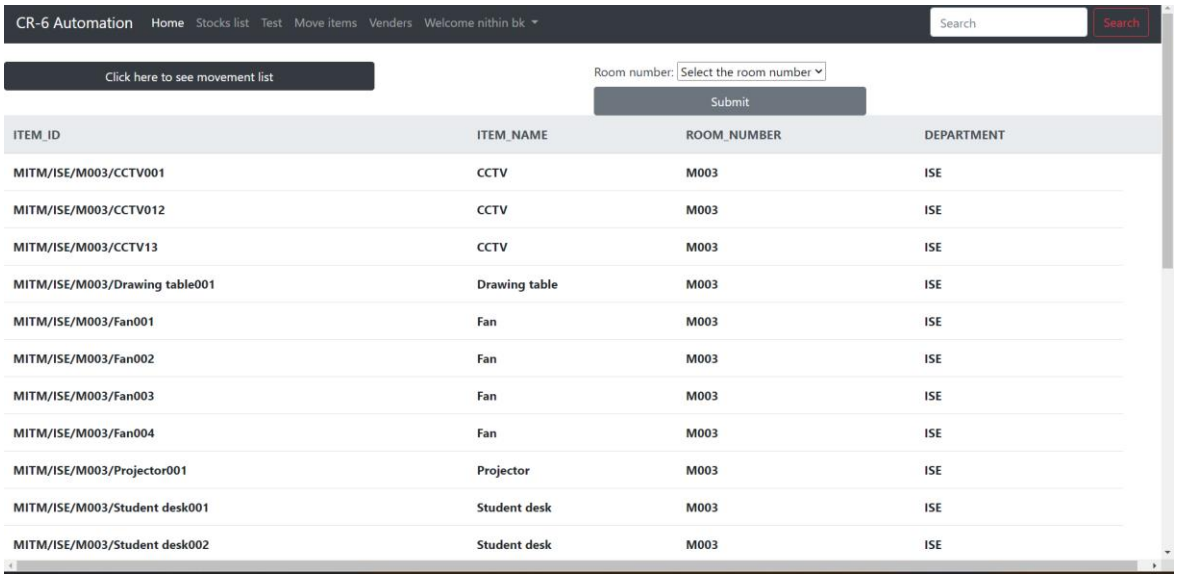


FIG: STOCK LIST

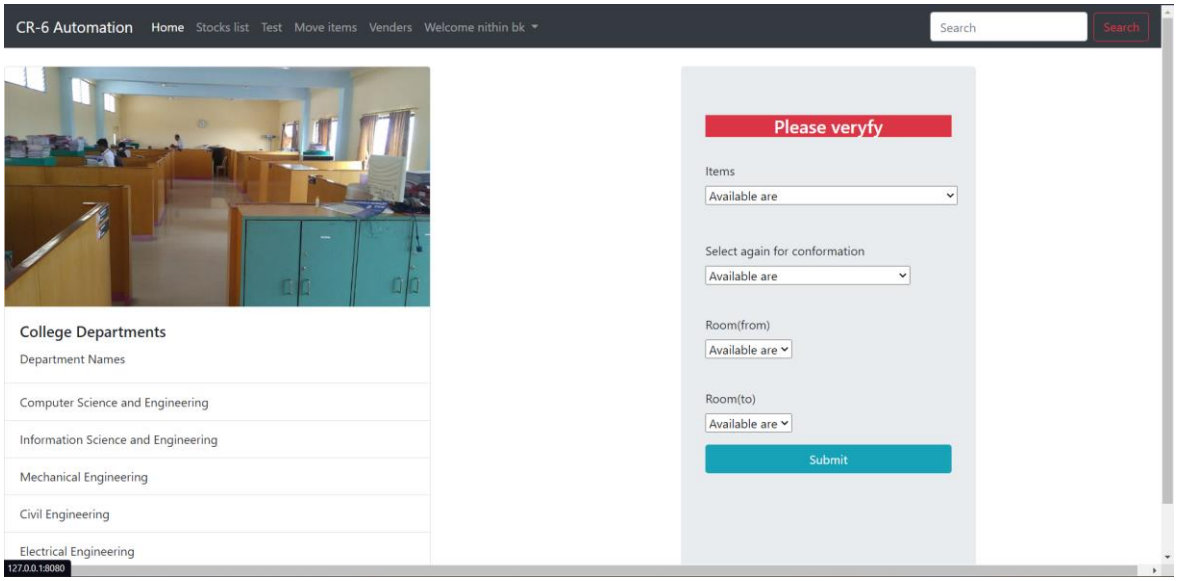


FIG: MOVING THE ITEMS

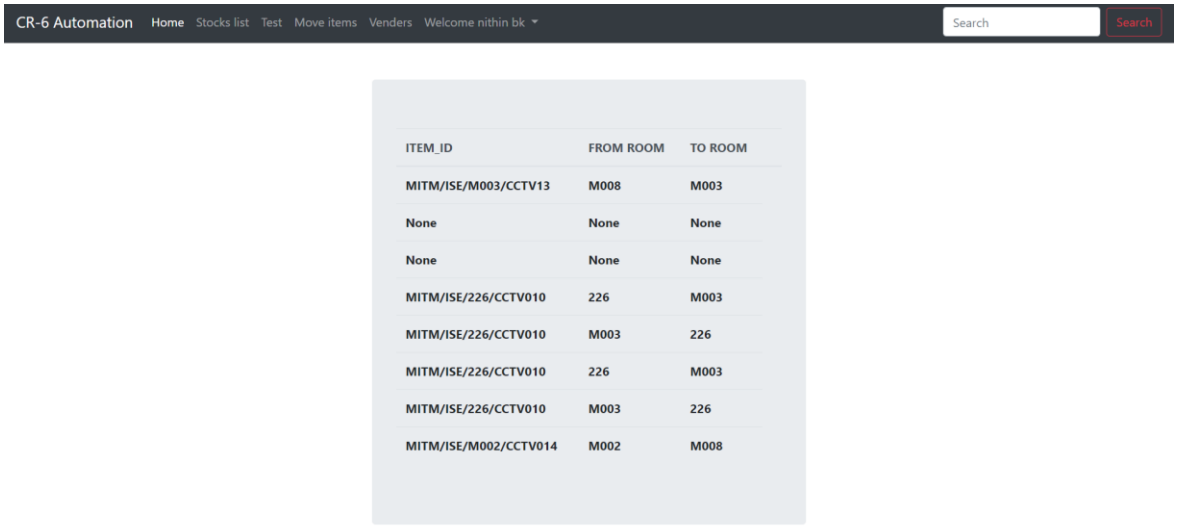


FIG: MOVEMENT LOGS

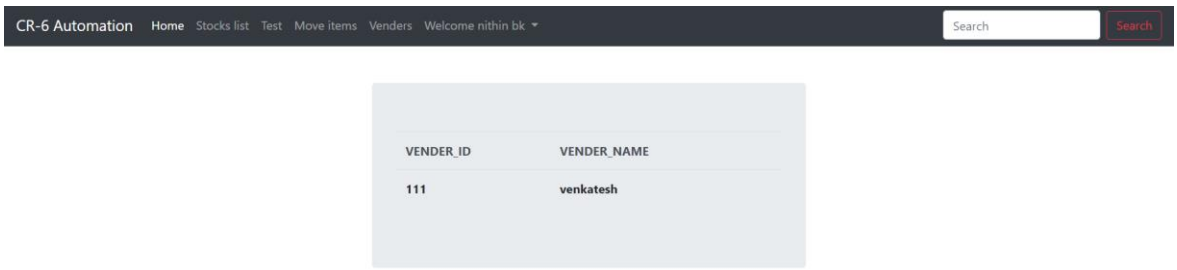


FIG: VENDORS INFORMATION

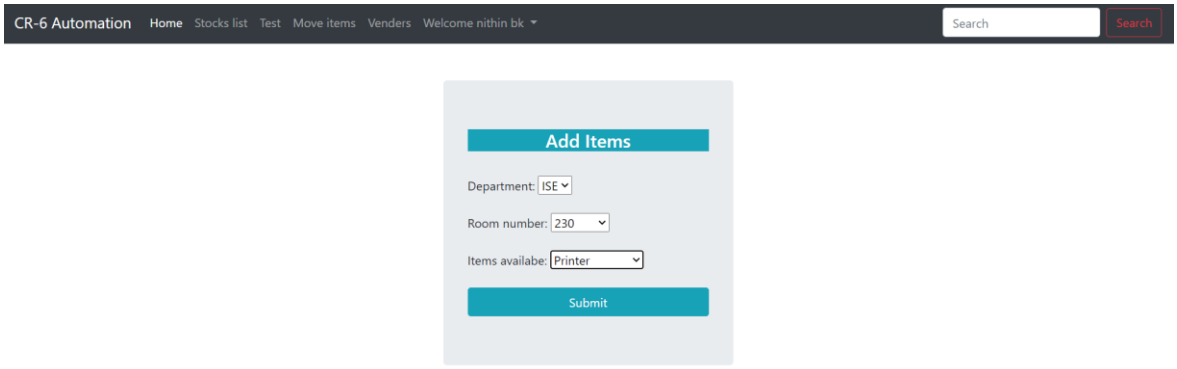


FIG: ADDING NEW ITEMS

CHAPTER:6
CONCLUSIONS AND FUTURE WORK

6.1 CONCLUSION

The main objective of this project is to help the department to maintain the stock list easily. Admin were facing difficulties while fetching particular items or to keep the track record of the various items that are present int the department. It is been done manually by maintaining logs and register books. Hence we designed this system to reduced that work by automating the system, using this CRITERIA 6 AUTOMATION system admin will be able to keep track of all record of the items and details of all the items which are newly added or moved from one room to another room even after several months/years.

6.2 FUTURE WORK

In future days, we thought to improve this project by adding more functionality like validating forms, in project details by also providing online payment by displaying the barcode ,that make more efficient for the user to track the records of the items. Initially we have designed this system for only one branch, It can be extended to many in future. The new items that are bought and the billings of those items can also be maintained here itself.

REFERENCES

For XAMPP:

- http://www.apachefriend.org/download.html

For HTML:

- http://www.w3schools.com/html

For PHP:

- http://www.php.net/

IDE: Visual Studio Code

- https://code.visualstudio.com/
- https://getbootstrap.com/

YOUTUBE CHANNEL:

- https://youtube.com/playlist?list=PL2aJidc6QnyPJtMlWFDHA9nHlVUl9DUwm