

# **Car Insurance Management**

**:-Assignwise**

## Table of Contents

### **Part A - Database Logical Model3**

#### **1.Background3**

#### **2.OBJECTIVES:3**

#### **3.Problems Statement:4**

#### **4.SCOPE4**

#### **5.Hardware and Software4**

#### **6.Entity Relationship Diagram5**

#### **7.Database Schema6**

#### **8. Normalization:6**

### **PART B (DATABASE DESIGN)7**

#### **Tables7**

#### **Tables Values9**

#### **Output Table10**

### **Part C11**

#### **1.Background11**

#### **2.OBJECTIVES:11**

#### **3.Problems Statement:12**

#### **4.SCOPE12**

#### **5.Hardware and Software12**

#### **6.Entity Relationship Diagram13**

#### **7.Database Schema14**

#### **8. Normalization:14**

#### **Print screen of system user interface15**

#### **Tables15**

#### **Tables Values17**

#### **Output Table18**

#### **Conclusion19**

## **Part A - Database Logical Model**

### **1.Background**

#### **Introduction to current system:**

The current system which is used is a traditional method of storing data and information of company. This system is called as File based system. File Based System helps to collect the data in the form of files. It helps users to organized the data in a small group. This system is very easy to use. It is effective for small data and information.

#### **Problems**

1. It cannot manage and organize large number of data.
2. It consumes more time and efforts.
3. It covers more space and are not flexible.
4. It does not provide backup and it can't recover the data.
5. It lacks proper security.

#### **Significances**

1. It helps to improve data retrieval techniques.
2. Files can be edited easily.
3. It is cheap and easy to understand.
4. They are used to handle a single or a multiple file.

### **2.OBJECTIVES:**

- To create new files in computer system.
- To manage data of users.
- To store, retrieve and update data.
- To provide data security.
- To maintain consistency.

### **3.Problems Statement:**

The problems that are faced on the current system of car insurance management system are listed below:

- It consumed our lots of time and efforts.
- It is not so safe. it lacks proper security.
- It caused a lot of technical errors.
- It does not work efficiently.
- It required a lot of money and the employees.
- It is very complex and difficult to manage.

### **4.SCOPE**

- To notify and alert the customer regarding the policies, premium due date and claim updates.
- To generate policy number, car number plate, model, customer Id, insurance Id, accident Id, customer first name, customer last name, car color, accident location and date.
- To keep proper record of car insurance and its policy.
- Admin can update the required details.
- To gain insight into policy performance and show smooth customers responds.

### **5.Hardware and Software**

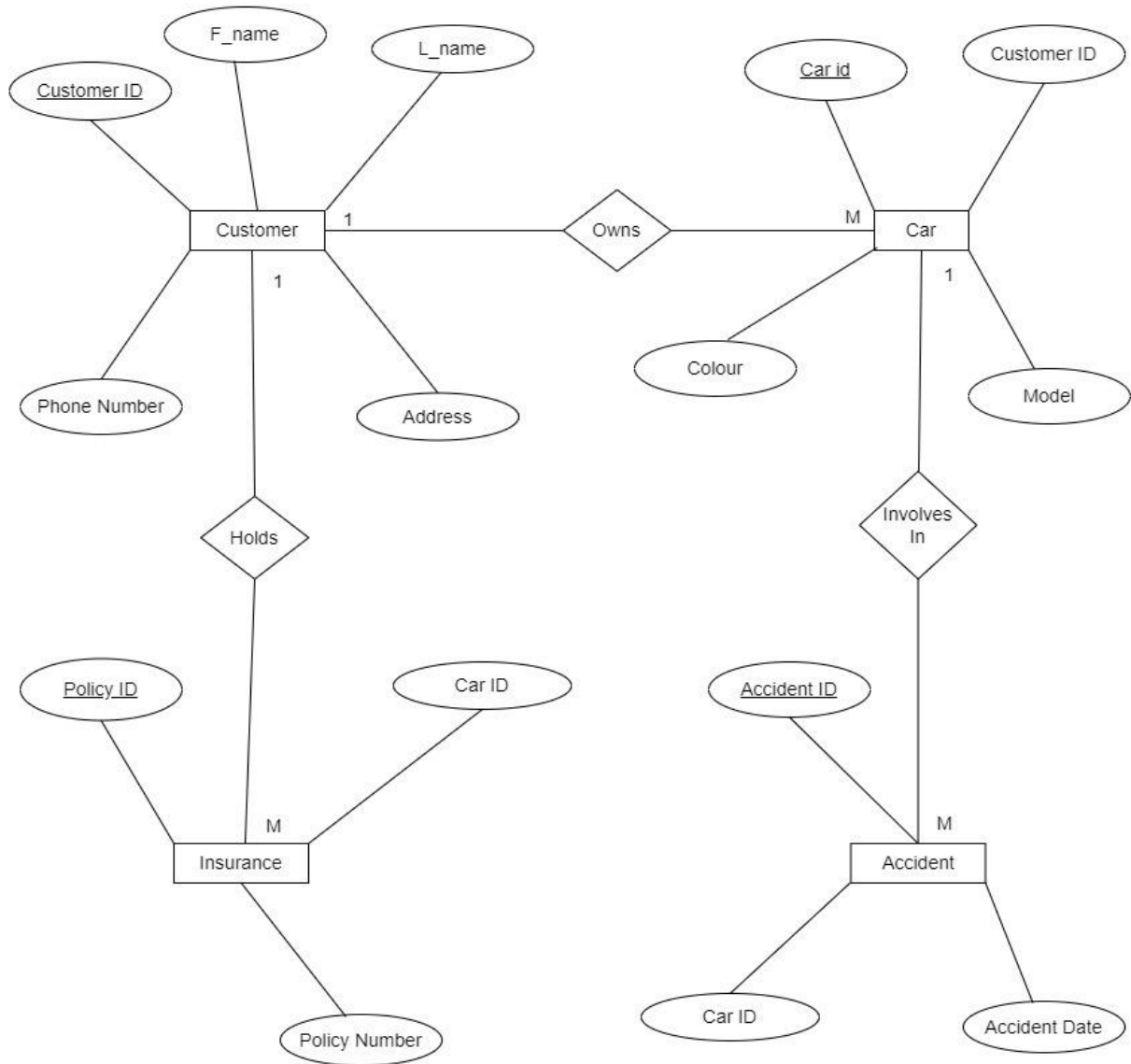
#### **Hardware:**

1. Monitor
2. Mouse
3. Keyboard
4. Printer
5. Speaker etc.

#### **Software:**

1. XAMPP Server
2. Server language PHP
3. My SQL

## 6.Entity Relationship Diagram



## 7.Database Schema

A representation of plane or theory in the form of a model.

### Customer table:

**Schema** (CustomerID, Firstname, LastName, PhoneNumber, Address);

Primary key: CustomerID

### Car Table:

**Schema** (CarID, CustomerID, colour, model);

Primary key: CarID

Foreign key: CustomerID

### Accident Table:

**Schema** (AccidentID, CarID, AccidentDate)

Primary key: AccidentID

Foreign key: CarID

### Insurance Table:

**Schema** (PolicyID, CarID, PolicyNumber)

Primary key: PolicyID

Foreign key: CarID

## 8. Normalization:

### customer table:

CustomerID	FirstName	LastName	PhoneNumber	Address
1	John	Das	9845001245	Sarlahi
2	Sita	Rai	9821221234	Chandrapur
3	Geet	Dahal	9866221830	Birgunj

### Car Table:

CarID	CustomerID	Colour	Model
1	1	red	Camry
2	2	black	Hyundai
3	3	white	Nissan

### Accident Table:

AccidentID	CarID	AccidentDate
11	1	2023-05-15
22	2	2020-2-25
33	3	2020-07-18

### Insurance Table:

PolicyID	CarID	PolicyNumber
2001	1	Pol-123456
2002	2	Pol-234567

2003	3	Pol-345678
------	---	------------

**1.First-Normal Form:** 1NF requires that each column in a table be atomic (indivisible) and that there are no repeating groups or arrays. All tables in the provided dataset are already in 1NF because they contain only single values in each cell.

**2.Second-Normal Form:** 2NF requires that the table is in 1NF and that there are no partial dependencies of any column on the primary key. In other words, each non-key attribute should be fully functionally dependent on the entire primary key.

The **Customer table** is already in 2NF as there are no partial dependencies.

In The **Car Table**, it appears that CarID can serve as the primary key because it uniquely identifies each row. So, CustomerID, colour, and model are functionally dependent on CarID.

The **Accident table** is already in 2NF as there are no partial dependencies.

The **Insurance table** is also already in 2NF as there are no partial dependencies.

**3.Third-Normal Form:** 3NF requires that the table is in 2NF and that there are no transitive dependencies. In other words, non-key attributes should not depend on other non-key attributes.

The **Customer table** is already in 3NF as there are no transitive dependencies.

The **Car details table** is also in 3NF because there are no transitive dependencies.

The **Accident table** is already in 3NF as there are no transitive dependencies.

The **Insurance table** is already in 3NF as there are no transitive dependencies.

So, all the tables are now in 3NF.

## PART B (DATABASE DESIGN)

### Tables

```

1 CREATE TABLE Customers (
2     CustomerID INT PRIMARY KEY AUTO_INCREMENT,
3     FirstName VARCHAR(50) NOT NULL,
4     LastName VARCHAR(50) NOT NULL,
5     PhoneNumber VARCHAR(15),
6     Address VARCHAR(255)
7 );

```

```
1 CREATE TABLE Cars (  
2     CarID INT PRIMARY KEY AUTO_INCREMENT,  
3     CustomerID INT,  
4     colour VARCHAR(50) NOT NULL,  
5     Model VARCHAR(50) NOT NULL,  
6     FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID)  
7 );|
```

```
1 CREATE TABLE Accidents (  
2     AccidentID INT PRIMARY KEY AUTO_INCREMENT,  
3     CarID INT,  
4     AccidentDate DATE NOT NULL,  
5     FOREIGN KEY (CarID) REFERENCES Cars(CarID)  
6 );|
```

```
1 CREATE TABLE Insurance (  
2     PolicyID INT PRIMARY KEY AUTO_INCREMENT,  
3     CarID INT,  
4     PolicyNumber VARCHAR(20) UNIQUE NOT NULL,  
5     FOREIGN KEY (CarID) REFERENCES Cars(CarID)  
6 );|
```



## Tables Values

```
1 INSERT INTO Customers (FirstName, LastName, PhoneNumber, Address)
2 VALUES ('John', 'Das', '9845001245', 'sarlahi'),
3 ('sita', 'Rai', '9821221234', 'chandrapur'),
4 ('Geet', 'Dahal', '9866221830', 'Birgunj');
```

```
1 INSERT INTO Cars (CarID, CustomerID, colour, Model)
2 VALUES (01, 1, 'Red', 'Camry'),
3         (02, 2, 'black', 'hyundai'),
4         (03, 3, 'white', 'Nissan');
5
```

```
1 INSERT INTO Accidents (AccidentID, CarID, AccidentDate)
2 VALUES (11, 01, '2023-05-15'),
3 (22, 02, '2020-02-25'),
4 (33, 03, '2022-07-18');
```

```
1 INSERT INTO insurance (policyID, CarID, PolicyNumber)
2 VALUES (2001, 01, 'POL-123456'),
3 (2002, 02, 'POL-234567'),
4 (2003, 03, 'POL-345678');
```

## Output Table

CustomerID	FirstName	LastName	PhoneNumber	Address
1	John	Das	9845001245	sarlahi
2	sita	Rai	9821221234	chandrapur
3	Geet	Dahal	9866221830	Birgunj

CarID	CustomerID	colour	Model
1	1	Red	Camry
2	2	black	hyundai
3	3	white	Nissan

AccidentID	CarID	AccidentDate
11	1	2023-05-15
22	2	2020-02-25
33	3	2022-07-18

PolicyID	CarID	PolicyNumber
2001	1	POL-123456
2002	2	POL-234567
2003	3	POL-345678

## **Part C**

### **1.Background**

#### **Introduction to current system:**

The current system which is used is a traditional method of storing data and information of company. This system is called as File based system. File Based System helps to collect the data in the form of files. It helps users to organized the data in a small group. This system is very easy to use. It is effective for small data and information.

#### **Problems**

6. It cannot manage and organize large number of data.
7. It consumes more time and efforts.
8. It covers more space and are not flexible.
9. It does not provide backup and it can't recover the data.
10. It lacks proper security.

#### **Significances**

5. It helps to improve data retrieval techniques.
6. Files can be edited easily.
7. It is cheap and easy to understand.
8. They are used to handle a single or a multiple file.

### **2.OBJECTIVES:**

- To create new files in computer system.
- To manage data of users.
- To store, retrieve and update data.
- To provide data security.
- To maintain consistency.

### **3.Problems Statement:**

The problems that are faced on the current system of car insurance management system are listed below:

- It consumed our lots of time and efforts.
- It is not so safe. it lacks proper security.
- It caused a lot of technical errors.
- It does not work efficiently.
- It required a lot of money and the employees.
- It is very complex and difficult to manage.

### **4.SCOPE**

- To notify and alert the customer regarding the policies, premium due date and claim updates.
- To generate policy number, car number plate, model, customer Id, insurance Id, accident Id, customer first name, customer last name, car color, accident location and date.
- To keep proper record of car insurance and its policy.
- Admin can update the required details.
- To gain insight into policy performance and show smooth customers responds.

### **5.Hardware and Software**

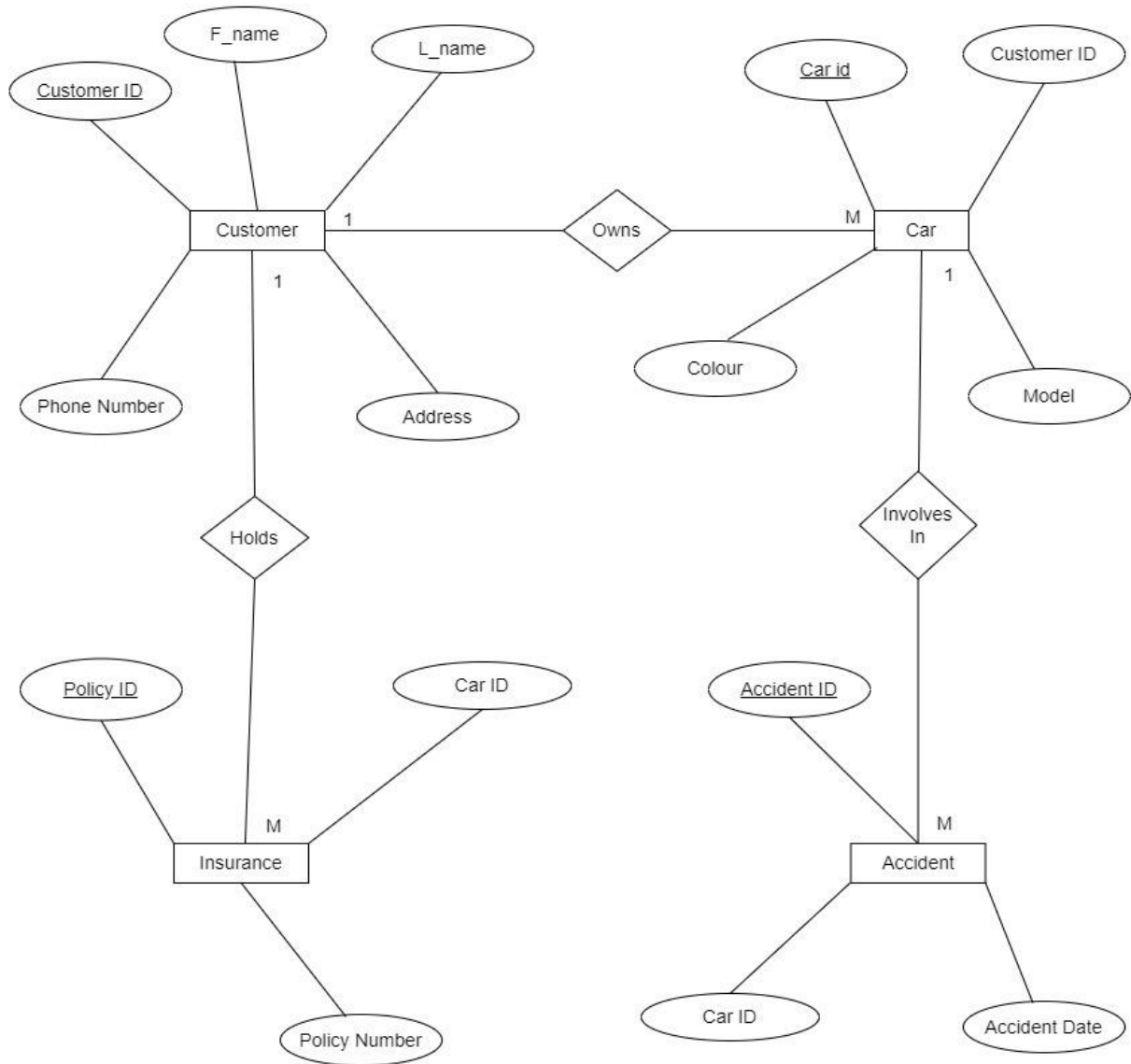
#### **Hardware:**

6. Monitor
7. Mouse
8. Keyboard
9. Printer
10. Speaker etc.

#### **Software:**

4. XAMPP Server
5. Server language PHP
6. My SQL

## 6.Entity Relationship Diagram



## 7.Database Schema

A representation of plane or theory in the form of a model.

### Customer table:

**Schema** (CustomerID, Firstname, LastName, PhoneNumber, Address);

Primary key: CustomerID

### Car Table:

**Schema** (CarID, CustomerID, colour, model);

Primary key: CarID

Foreign key: CustomerID

### Accident Table:

**Schema** (AccidentID, CarID, AccidentDate)

Primary key: AccidentID

Foreign key: CarID

### Insurance Table:

**Schema** (PolicyID, CarID, PolicyNumber)

Primary key: PolicyID

Foreign key: CarID

## 8. Normalization:

### customer table:

CustomerID	FirstName	LastName	PhoneNumber	Address
1	John	Das	9845001245	Sarlahi
2	Sita	Rai	9821221234	Chandrapur
3	Geet	Dahal	9866221830	Birgunj

### Car Table:

CarID	CustomerID	Colour	Model
1	1	red	Camry
2	2	black	Hyundai
3	3	white	Nissan

### Accident Table:

AccidentID	CarID	AccidentDate
11	1	2023-05-15
22	2	2020-2-25
33	3	2020-07-18

### Insurance Table:

PolicyID	CarID	PolicyNumber
2001	1	Pol-123456
2002	2	Pol-234567

2003	3	Pol-345678
------	---	------------

**1.First-Normal Form:** 1NF requires that each column in a table be atomic (indivisible) and that there are no repeating groups or arrays. All tables in the provided dataset are already in 1NF because they contain only single values in each cell.

**2.Second-Normal Form:** 2NF requires that the table is in 1NF and that there are no partial dependencies of any column on the primary key. In other words, each non-key attribute should be fully functionally dependent on the entire primary key.

The **Customer table** is already in 2NF as there are no partial dependencies.

In The **Car Table**, it appears that CarID can serve as the primary key because it uniquely identifies each row. So, CustomerID, colour, and model are functionally dependent on CarID.

The **Accident table** is already in 2NF as there are no partial dependencies.

The **Insurance table** is also already in 2NF as there are no partial dependencies.

**3.Third-Normal Form:** 3NF requires that the table is in 2NF and that there are no transitive dependencies. In other words, non-key attributes should not depend on other non-key attributes.

The **Customer table** is already in 3NF as there are no transitive dependencies.

The **Car details table** is also in 3NF because there are no transitive dependencies.

The **Accident table** is already in 3NF as there are no transitive dependencies.

The **Insurance table** is already in 3NF as there are no transitive dependencies.

So, all the tables are now in 3NF.

## Print screen of system user interface Tables

```

1 CREATE TABLE Customers (
2     CustomerID INT PRIMARY KEY AUTO_INCREMENT,
3     FirstName VARCHAR(50) NOT NULL,
4     LastName VARCHAR(50) NOT NULL,
5     PhoneNumber VARCHAR(15),
6     Address VARCHAR(255)
7 );|

```

```
1 CREATE TABLE Cars (  
2     CarID INT PRIMARY KEY AUTO_INCREMENT,  
3     CustomerID INT,  
4     colour VARCHAR(50) NOT NULL,  
5     Model VARCHAR(50) NOT NULL,  
6     FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID)  
7 );|
```

```
1 CREATE TABLE Accidents (  
2     AccidentID INT PRIMARY KEY AUTO_INCREMENT,  
3     CarID INT,  
4     AccidentDate DATE NOT NULL,  
5     FOREIGN KEY (CarID) REFERENCES Cars(CarID)  
6 );|
```

```
1 CREATE TABLE Insurance (  
2     PolicyID INT PRIMARY KEY AUTO_INCREMENT,  
3     CarID INT,  
4     PolicyNumber VARCHAR(20) UNIQUE NOT NULL,  
5     FOREIGN KEY (CarID) REFERENCES Cars(CarID)  
6 );|
```



## Tables Values

```
1 INSERT INTO Customers (FirstName, LastName, PhoneNumber, Address)
2 VALUES ('John', 'Das', '9845001245', 'sarlahi'),
3 ('sita', 'Rai', '9821221234', 'chandrapur'),
4 ('Geet', 'Dahal', '9866221830', 'Birgunj');
```

```
1 INSERT INTO Cars (CarID, CustomerID, colour, Model)
2 VALUES (01, 1, 'Red', 'Camry'),
3         (02, 2, 'black', 'hyundai'),
4         (03, 3, 'white', 'Nissan');
5
```

```
1 INSERT INTO Accidents (AccidentID, CarID, AccidentDate)
2 VALUES (11, 01, '2023-05-15'),
3         (22, 02, '2020-02-25'),
4         (33, 03, '2022-07-18');
```

```
1 INSERT INTO insurance (policyID, CarID, PolicyNumber)
2 VALUES (2001, 01, 'POL-123456'),
3         (2002, 02, 'POL-234567'),
4         (2003, 03, 'POL-345678');
```

## Output Table

CustomerID	FirstName	LastName	PhoneNumber	Address
1	John	Das	9845001245	sarlahi
2	sita	Rai	9821221234	chandrapur
3	Geet	Dahal	9866221830	Birgunj

CarID	CustomerID	colour	Model
1	1	Red	Camry
2	2	black	hyundai
3	3	white	Nissan

AccidentID	CarID	AccidentDate
11	1	2023-05-15
22	2	2020-02-25
33	3	2022-07-18

PolicyID	CarID	PolicyNumber
2001	1	POL-123456
2002	2	POL-234567
2003	3	POL-345678

## Conclusion

According to the Database Management System (DBMS), Car Insurance Management is the management of car insurance that helps to keep the records

### **PART A – DATABASE LOGICAL DESIGN**

related to the insurance policy. It records all the essential information, data which are required for the insurance. Car Insurance helps to cover the damages that happened in accident or any other reason. Car Insurance Management is important for person as well as the insurance company. It convey complex challenges, provides protection to the data of customer and also ensure regulative deference. It is the traditional way of file management. It indicates the management of car insurance policies, customer information, details of car, and other linked data of insurance company. This management is very much important to ensure the process of car insurance smoothly. Car insurance management system has both advantages as well as the disadvantages. They are listed below:

#### **ADVANTAGES :**

- It helps to ensure data accuracy by executing data validation rule and reducing the risk error.
- It helps to maintain the problems that impact daily operations.
- It handle and manages the large number of data.
- It helps to store the essentials data and also helps in backup and recovery.

#### **DISADVANTAGES:**

- It required regular maintenance and updates to work smoothly.
- It is expensive and complex.
- It required more employees to work on the given task.
- It is very much time consuming.

#### **FUTURE WORK:**

We will try to make it work smoothly with our best work. We will be updating the system whenever it is required. We will check it out on regular basis.

NO	TASK	MARKS
1.	<b>Background</b> Background should include the following: <ul style="list-style-type: none"> <li>• Introduction to current system</li> <li>• Problem with the existing system</li> <li>• How the problem could be solve using the proposed system</li> </ul>	[     /5]
2.	<b>Objective</b> The objective of current system <b>SHOULD NOT BE MORE THAN FIVE AND LESS THAN THREE</b> and it should be listed in bullet form.	[     /5]
3.	<b>Problem Statement</b> -State what the problems with current process / system are	[     /5]
4.	<b>Scope</b> -Define scope for your proposed system	[     /5]
5.	<b>Hardware &amp; Software specification</b> -Identify the hardware and software requirements to develop the system	[     /5]
6.	<b>Entity Relationship Diagram</b> -Draw entity relationship diagram with its attributes	[     /15]
7.	<b>Database Schema</b> -Provide the data schema for the proposed system. Data schema should be documented using the entity relationship diagram. - Identify the primary and foreign key	[     /10]
8.	<b>Normalization</b> - Normalize the table into 3NF	[     /10]
	<b>PART A TOTAL</b>	[     /50]

Part B	System Development	Mark
1.	Achieve the scope mentioned in Part A	[     /3]
2.	Able to run with free of errors	[     /3]
3.	Has data validation	[     /3]
4.	Enable Entity integrity and referential integrity	[     /3]
5.	Overall design of the system / creativity	[     /3]
	<b>PART B TOTAL</b>	[     /15]
Part C	Documentation	Mark
1.	All components listed in Part A	[     /5]
2.	Print screen of result	[     /5]

3.	Conclusions -Must consist of brief introduction of your system -Advantage of your developed system -Limitations of your system -Future work / Suggestion	[ /5]
<b>PART C TOTAL</b>		[ /15]
<b>Part D</b>	<b>Description (Presentation)</b>	<b>Mark</b>
1.	The presentation contained a brief and effective introduction	[ /2]
2.	The explanation of the System design was clear and logical	[ /2]
3.	Additional recommendations for expansion plan were helpful	[ /1]
4.	The conclusions was satisfying and gave a sense of closure	[ /1]
5.	Demonstration of the system	[ /2]
6.	Able to answer question	[ /1]
7.	Attire and cooperation of the group	[ /1]
<b>PART D TOTAL</b>		[ /10]
<b>TOTAL A+B+C+D</b>		[ /100]