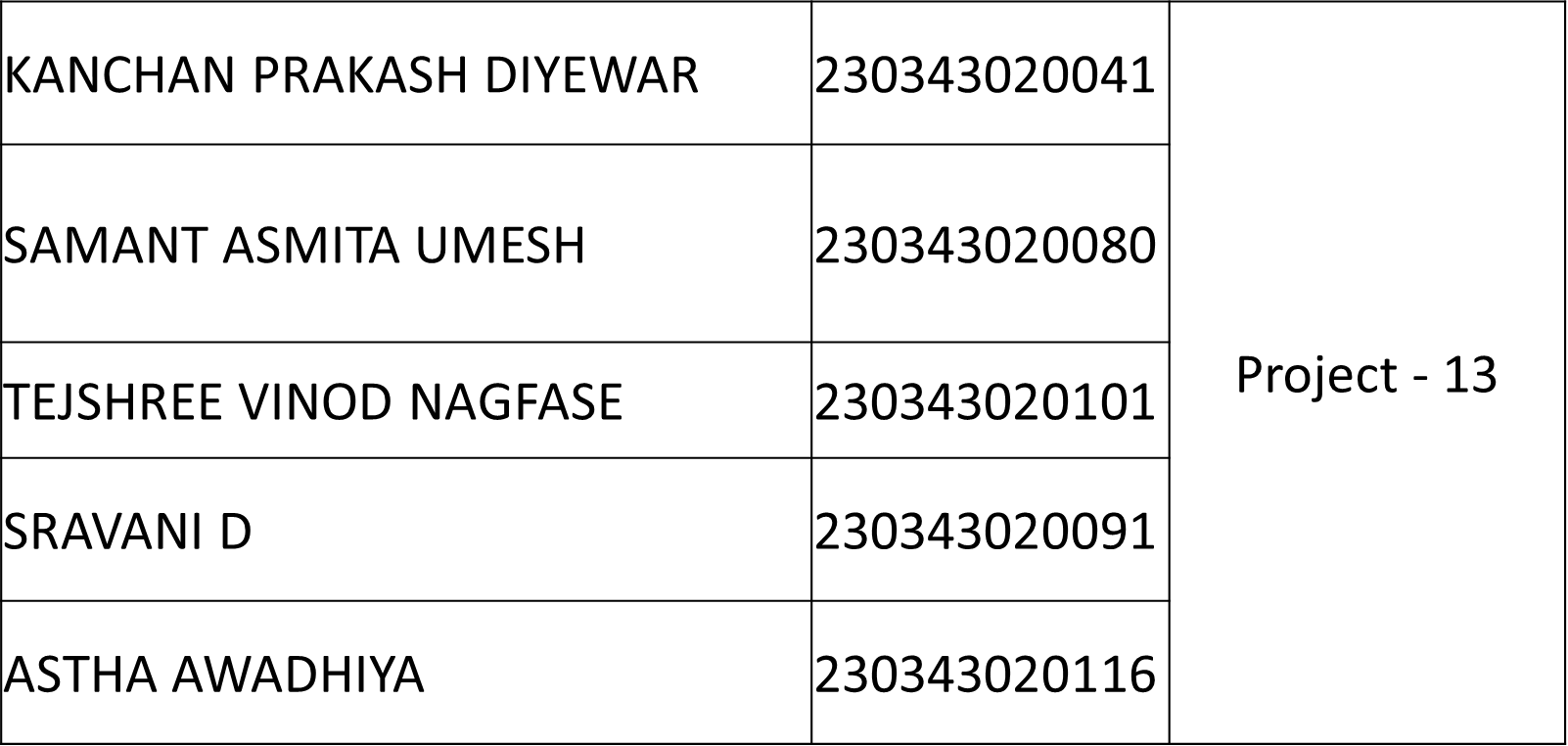
DESIGN DOCUMENT

HANDYMAN SERVICE SYSTEM



The purpose of Handyman Service system is to build a platform for both customers and small scale vendors. By using Handyman service system it is easy for customers to find out vendors for different services required. Customers can get even the reviews from different customers and verify standard rates for different household services. Vendors can reach to multiple customers by providing better services.

Tables that we have identified are:

1. Customers: This table contains the information about the customers.

2. Vendors: This table contains the information about the Vendors.

3. Services: This table contains the information about the services provided.

4. VendorServices: This table contains the information about the vendor and the service provided and the rates.

5. Appointment: if customer Book the service it will be reflected in this table.

6. Payment: It will contain the information of payments made by the customer.

7. Feedback: This table contains the feedback and rating given by the customer.

8. History: This table will show history of appointment.

TABLE RELATIONS:

Customers

1 Customer can have many appointment.

1 Customer can have multiple History records.

1 Customer can have multiple Feedback.

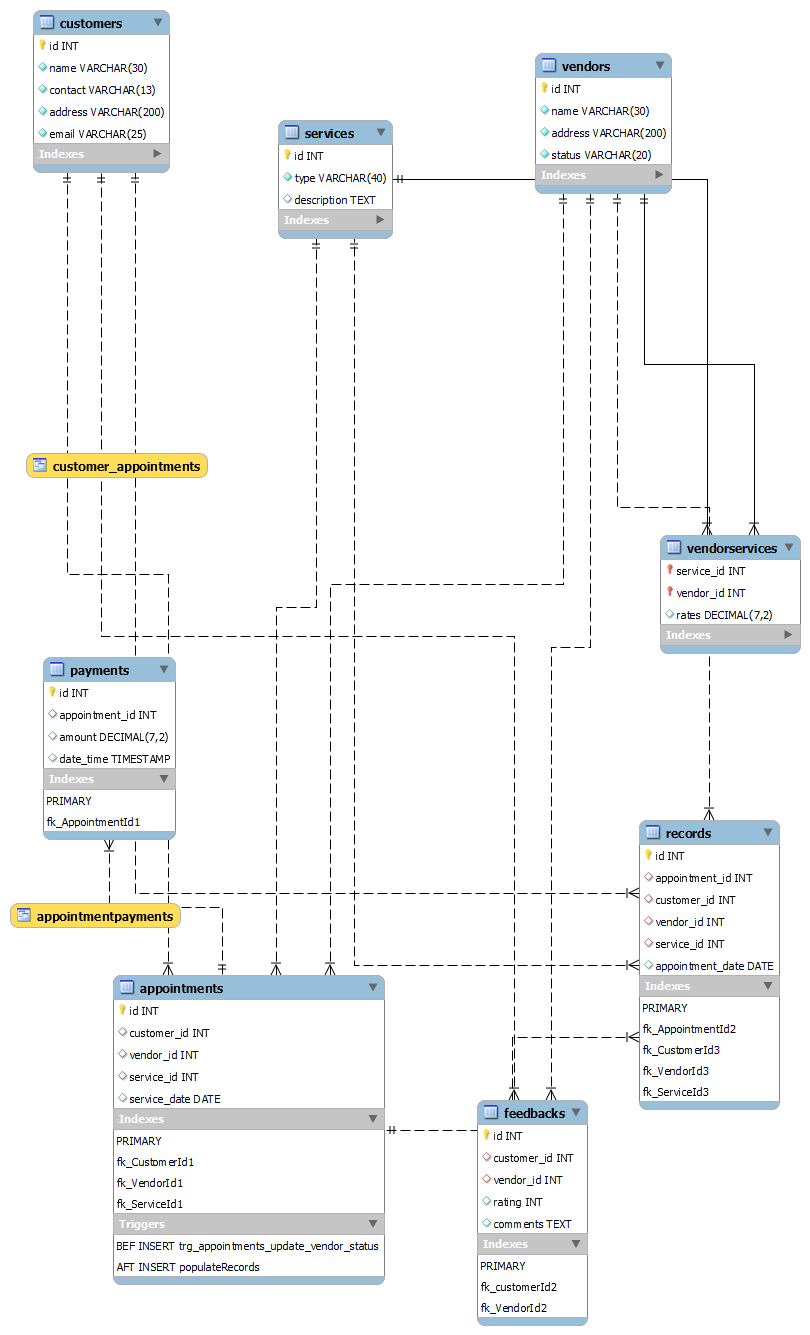
Vendors

1 Vendor can provide Many Services.

1 Vendor can have multiple appointment (depending upon vendor status)

1 Vendor can have multiple history record.

1 Vendor can have multiple feedback

**ER DIGRAM**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Field | Type | Null | Key | Default | Extra |
| id | int | No | PRI | NULL | auto\_increment |
| Customer\_name | varchar(30) | No |  | NULL |  |
| contact | varchar(13) | No |  | NULL |  |
| address | varchar(200) | No |  | NULL |  |
| email | varchar(75) | No | UNI | NULL |  |

**vendors**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Field | Type | Null | Key | Default | Extra |
| id | int | No | PRI | NULL | auto\_increment |
| Vendor\_name | varchar(30) | No |  | NULL |  |
| address | varchar(200) | No |  | NULL |  |
| vendor\_status | Varchar(20) | No |  | Available |  |

**services**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Field | Type | Null | Key | Default | Extra |
| id | int | No | PRI | NULL | auto\_increment |
| Service\_name | varchar(40) | Yes |  | NULL |  |
| description | Text | Yes |  | NULL |  |

**VendorServices**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Field | Type | Null | Key | Default | Extra |
| service\_id | int | No | PRI | NULL |  |
| vendor\_id | int | No | PRI | NULL |  |
| rates | decimal(7,2) | Yes |  | NULL |  |

**appointments**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Field | Type | Null | Key | Default | Extra |
| id | int | No | PRI | NULL | auto\_increment |
| service\_id | int | Yes | MUL | NULL |  |
| vendor\_id | int | Yes | MUL | NULL |  |
| customer\_id | int | Yes | MUL | NULL |  |
| Service\_date | date | Yes |  | NULL |  |

**payments**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Field | Type | Null | Key | Default | Extra |
| id | int | No | PRI | NULL | auto\_increment |
| service\_id | int | Yes | MUL | NULL |  |
| amount | decimal(7,2) | Yes |  | NULL |  |
| date\_time | timestamp | Yes |  | NULL |  |

**feedbacks**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Field | Type | Null | Key | Default | Extra |
| id | int | No | PRI | NULL | auto\_increment |
| customer\_id | int | Yes | MUL | NULL |  |
| vendor\_id | int | Yes | MUL | NULL |  |
| rating | int | Yes |  | NULL |  |
| comments | text | Yes |  | NULL |  |

**records**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Field | Type | Null | Key | Default | Extra |
| id | int | No | PRI | NULL | auto\_increment |
| appointment\_id | int | Yes | MUL | NULL |  |
| service\_id | int | Yes | MUL | NULL |  |
| vendor\_id | int | Yes | MUL | NULL |  |
| customer\_id | int | Yes | MUL | NULL |  |
| Appointment\_date | date | Yes |  | NULL |  |

**1. Trigger to populate data into History table.**

**2. Trigger to update vendor’s status in Vendor table after checking Vendors availability.**

**3. View to retrieve customer details along with their appointment information.**

**4.** **View to retrieve appointment details along with the corresponding payment information.**

**5. Filter the customer's appointments based on CustomerId.**