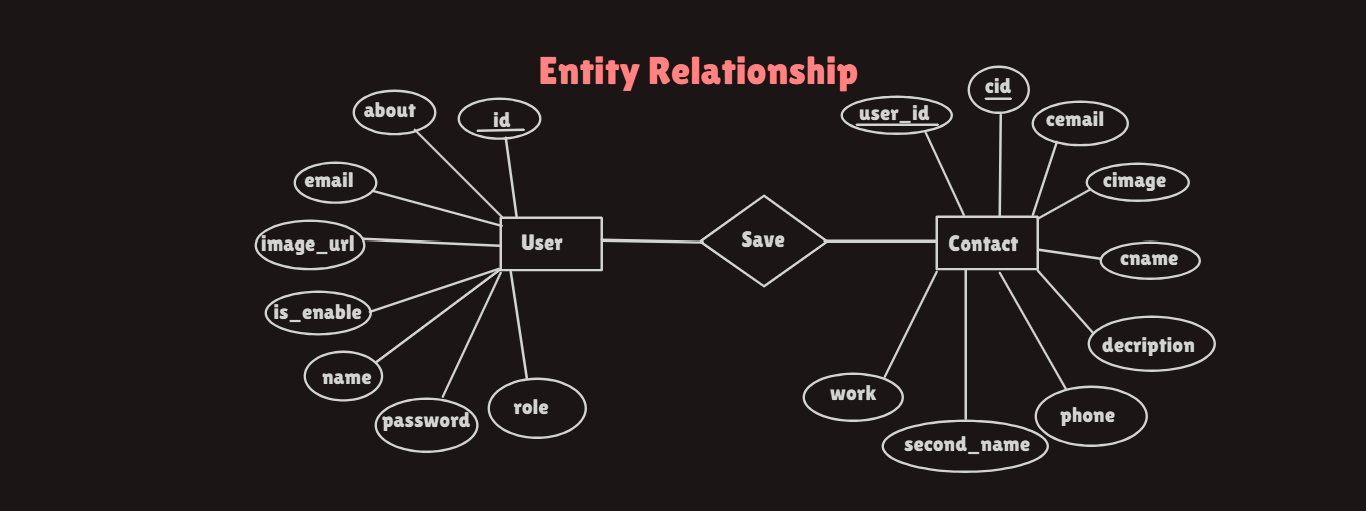
**Diagrams [Smart Contact Manager]**

1. **ER Diagram (Entity-Relationship Diagram)**

****

**Entities:**

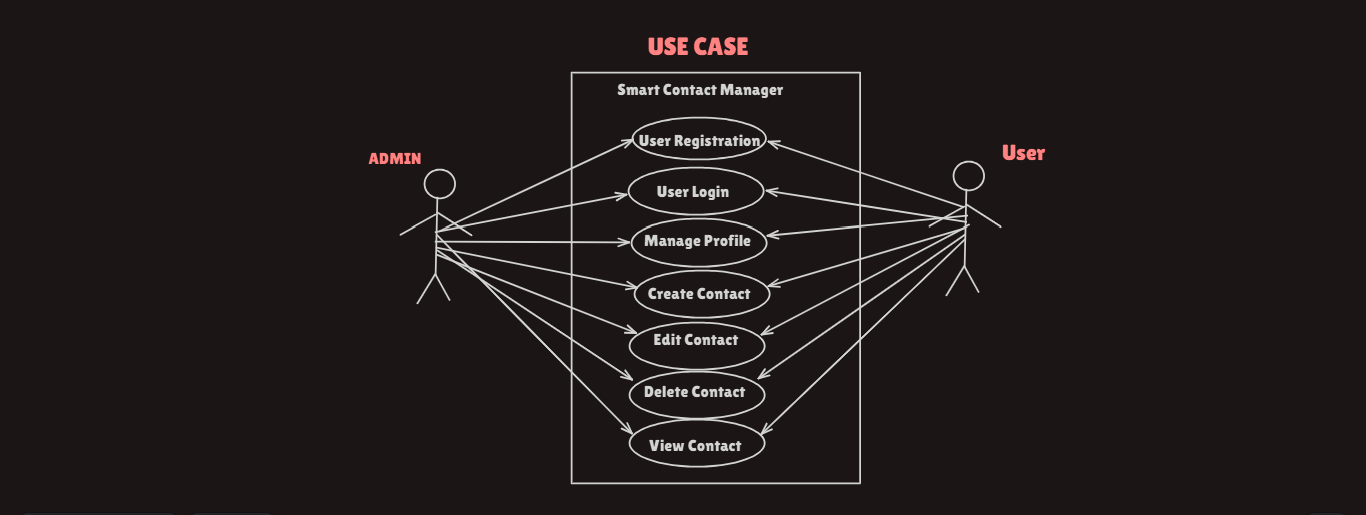
* **User**
  + Attributes: id, about, email, image\_url, is\_enable, name, password, role
* **Contact**
  + Attributes: cid, cemail, cimage, cname, decription, phone, second\_name, work, user\_id

**Relationships:**

* A **User** can have multiple **Contacts** (One-to-Many relationship).
* The user\_id in the **Contact** table acts as a foreign key referencing id in the **User** table.

1. **Use Case DiagramActors:**

* **Admin/User**: Manages users and their contact

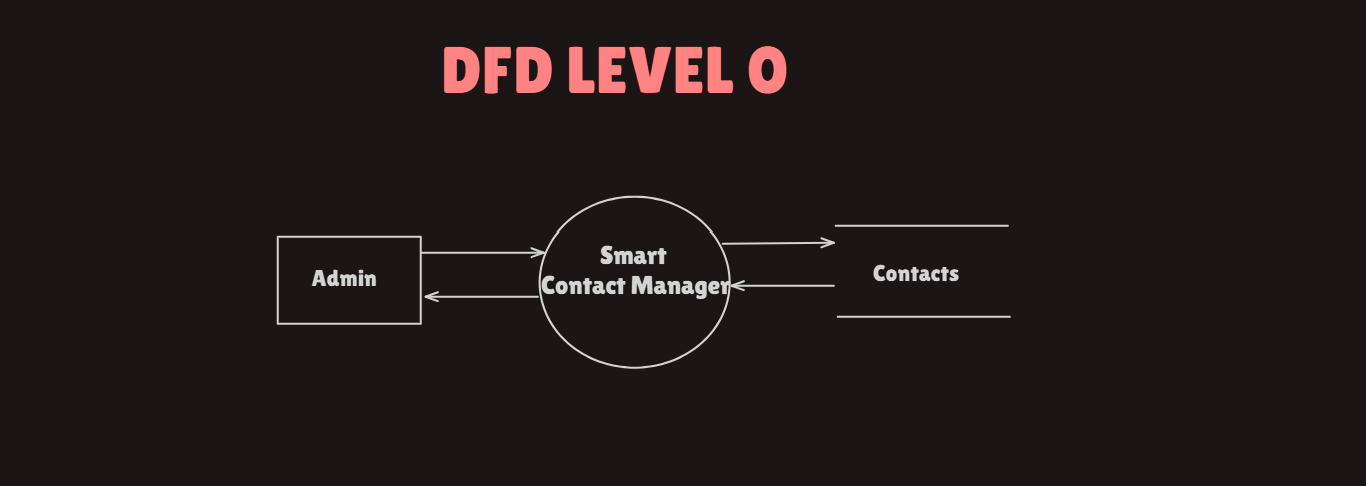
****

**Use Cases:**

1. **User Registration**: Add a new user.
2. **User Login**: Authenticate using email and password.
3. **Manage Profile**: Update details like about, image\_url, name, etc.
4. **Create Contact**: Add a new contact for the logged-in user.
5. **Edit Contact**: Modify an existing contact's details.
6. **Delete Contact**: Remove a contact.
7. **View Contact List**: Fetch all contacts for the logged-in user.

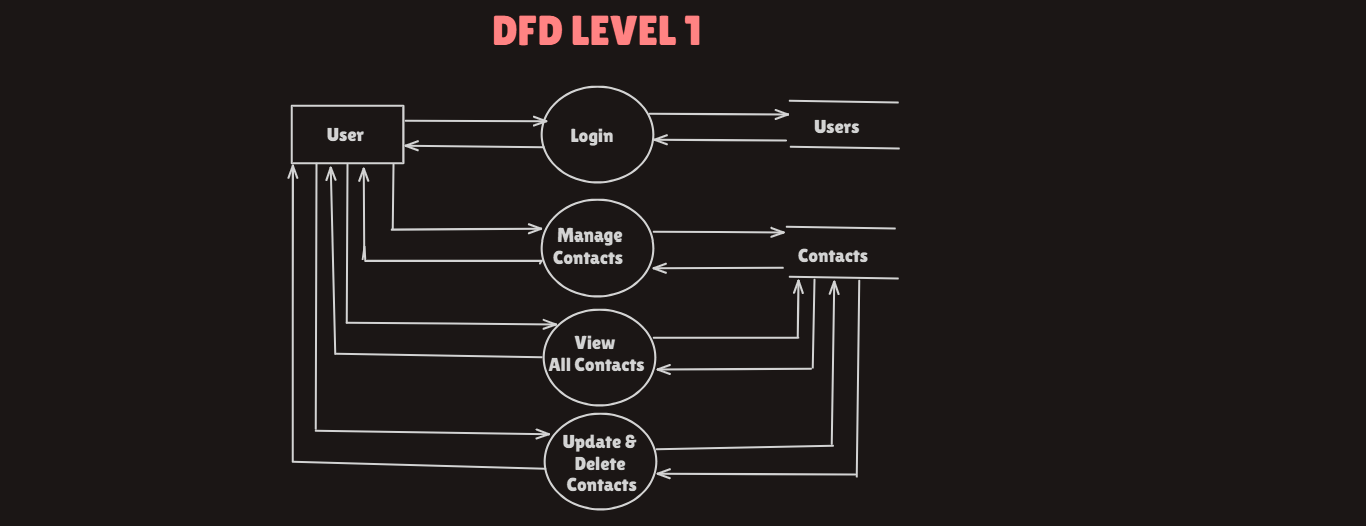
**3. Data Flow Diagram (DFD)**

**Level 0 (Context Diagram):**

****

* **External Entities**: User
* **Processes**: Manage User and Contact Information
* **Data Stores**: User Table, Contact Table
* **Data Flows**:
  + From User: Registration details, Login credentials, Contact details
  + To User: Profile details, Contact list, Contact details

**Level 1 (Detailed DFD):**

****

* **Processes**:
  1. **User Authentication**: Validates email and password for login.
  2. **Manage Profile**: Updates user details.
  3. **Contact Management**:
     + Add a new contact.
     + Update an existing contact.
     + Fetch all contacts for a user.
     + Delete a contact.
* **Data Stores**:
  1. User Table: Stores user-related data.
  2. Contact Table: Stores contacts for each user.