

Chapter No 09

Database Structure

9.1 Database design of our application

Database design is the organization of data according to a database model. The designer determines what data must be stored and how the data elements interrelate. With this information, they can begin to fit the data to the database model. This data model all the needed logical and physical design choices. It also contains physical storage parameters needed to generate a design in a data definition language, which can then be used to create a database. A fully attributed data model contains detailed attributes for each entity.

We are using normalized database “Firebase Real-time Database” which is hosted at Firebase console. It provides faster query performance.

The process of doing database design generally consists number of steps which will be carried out by database designer. Usually, the designer must:

- Determine the data to be stored in database.
- Determine the relationship the different data elements.
- Superimpose a logical structure upon the data on the basis of these relationships.

Our database structure is made up of two levels

1. Conceptual level

2. Normalization

9.1.1 Conceptual level

The conceptual level describes the Database structure of the whole database for the community of users. This schema hides information about the physical storage structures. It is a complete description of the data stored in database. It stores the complete data that's why it is also known as community view of database. It focuses on describing:

- **Entities:** which are distinct in system.
- **Relationship:** define meaningful interaction between entities.
- **Attributes:** properties of entities, have relationship.

9.1.2 Normalization

Database normalization is the technique of organizing the data in the database. Normalization is a systematic approach of decomposing tables to eliminate data redundancy and undesirable characteristics like Insertion, Update and Deletion Anomalies. It is a multistep process that puts data into tabular form by removing duplicated data from the relation tables.

Normalization rule are divided into following normal form:

1. First Normal Form
2. Second Normal Form
3. Third Normal Form

9.1.2.1 First Normal Form

As per First Normal Form, in our tables no two rows of data contain repeating group of information i.e. each set of columns have a unique value, such that multiple columns cannot be used to fetch the same row.

9.1.2.2 Second Normal Form

As per Second Normal Form, there is no any partial dependency of any column on primary key. It means that for a table that has concatenated primary key, each column in the table that is not part of the primary key must depend upon the entire concatenated key for its existence.

9.1.2.3 Third Normal Form

Third Normal Form applies that every no prime attribute of table is dependent on primary key, or we can say that, there is not nay case in which a non-prime attribute is determined by another non-prime attribute. So, this transitive functional dependency is removed from the tables and also the tables are in Second Normal form.

9.2 Firebase Real-time Database Structure

9.2.1 Firebase Root Structure

Firebase main structure showing users, trips, and notifications nodes.



Figure9.1 Root Structure

9.2.2 Table Name: Trips Table

Trip table showing details like departure, destination, trip name, invite code, and trip ID.



Figure9.2 Trip Table

9.2.3 Table Name: User Details Table

User table storing personal data including name, email, coordinates, emergency contact, and device token.



Figure 9.3 User details

9.2.4 Table Name: User List Table View

A list view of all users with email, registration dates, and Firebase IDs.

imanfatima212311008...		5 May 2025	5 May 2025	1Yq5NUSNvOUEbXrqIaNrltHw...
smeerab53@gmail.com		5 May 2025	22 May 2025	FFCsiqDKOcTKBtAqoQS9Hy2i...
emanfatima4010@gma...		5 May 2025	24 May 2025	MLfRmxXMjZQNkx0FP2z1Bk0...
aqsan4847@gmail.com		3 May 2025	24 May 2025	6yBySwfsELfvkETTU3afwNjnP...
meerabshahbaz20@gm...		1 May 2025	24 May 2025	HE2WI5Fm7XQJ3e0ykd5PFR...

Figure9.4 User List Table View