ADVANCE SOFTWARE SYSTEM DESIGN

Name : A.A.T.Suyama

Index : 1040

Reg No : itt-18-19-083

1.What you understand by CURD

CRUD operations are a fundamental concept in database management and software development. CRUD stands for Create, Read, Update, and Delete, which are the basic operations that can be performed on data in a database or any persistent storage system. Here's a brief explanation of each operation:

1. **Create**: This operation involves adding new data records to the database. For example, you might create a new entry in a user database when a new user signs up for an account on a website.

2. **Read**: Reading, or retrieving, involves fetching existing data records from the database. This operation allows you to access the data stored in the database for various purposes, such as displaying it to users or using it in computations or analysis.

3. **Update**: Updating involves modifying existing data records in the database. For instance, you might update a user's profile information when they change their email address or update the quantity of a product in an inventory system.

4. **Delete**: Deleting involves removing existing data records from the database. This operation is used to eliminate unnecessary or outdated data from the system, such as deleting a user account or removing a product that is no longer available for sale.

CRUD operations are essential for creating, managing, and manipulating data in virtually any software application that interacts with a database or persistent storage system. They form the basis for most data management tasks and are supported by most modern database management systems and programming frameworks.

3. Update can be done via PUT and PATCH. What the difference on each

PUT and PATCH are both HTTP methods used for updating resources on a web server, but they have different semantics and use cases.

1. **PUT**:

   - **Purpose**: PUT is typically used when you want to update a resource completely, meaning that the entire resource should be replaced with the new representation provided in the request.

- **Idempotent**: PUT requests are idempotent, which means that making the same PUT request multiple times should have the same effect as making it once. In other words, subsequent identical requests should not have additional side effects.

- **Example**: If you have a resource representing a user profile, a PUT request might be used to replace the entire user profile with a new set of data.

2. **PATCH**:

- **Purpose**: PATCH is used when you want to apply a partial update to a resource, meaning that you want to change only some of the resource's attributes without affecting the others.

- **Not necessarily idempotent**: PATCH requests are not necessarily idempotent. Making the same PATCH request multiple times might have different effects depending on the state of the resource and the specific patch operations being applied.

- **Example**: If you have a resource representing a user profile, a PATCH request might be used to update only the user's email address or only the user's phone number without touching the rest of the profile data.