**KEY LEARNINGS FROM PROJECT:**

**1. End-to-End process of database generation:**

We learn the end to end process of database generation, right from conception of idea, to designing database, random data generation, data population, writing queries, stored procedures to documentation. Especially, random data generation part was new for me. I learnt to use Microsoft Excel for data generation. I got opportunity to write java code for data generation. Previously, I had the knowledge of Microsoft SQL Server 2005 during my undergrad. This project gave me opportunity to use tools like mySql Workbench, PostgreSQL. Working with tables having self references (like response to answers) was new learning. We also learned that the first design is not final. At the time of implementation we had to do some minor changes in design considering practical limitations.

**2. Indexing techniques**

Through course work and project work, we had the opportunity to learn different indexing techniques like B tree index, hash tables, etc. We also learnt how indexing helps to reduce query runtime and increases search efficiency.

**3. Constraints**

We were able to study and implement primary key index, foreign key index, cascading effects in presence of foreign key index. Although, in project we mainly relied on indexes auto generated by PostgreSQL. One of our applications was to score a researcher considering how active contribution he has. We had to access many table and many attributes. This query, if run for all database, which has records of about 300000 researchers, would take lot of time. We realized later that we could have designed and experimented on indexes for such applications.

**4. Normalization**

We got the good idea of process of normalization. Although as we have designed relational database and our database is normalized, we understood that normalization is not always a good idea. Especially, repetition of information in database actually helps in case of databases which are more frequently used for report generation.

**5. Handling complex queries**

We learned to handle some of the complex queries. We also learned that complex queries can be divided in small pieces/parts. Then we can combine parts to complete complex querying. We learnt the aggregates, functions that work on dates, different types of joins and their uses.

**6. Future scope**

We were bit too ambitious initially and designed a big database with large number of tables. But we were able to estimate in earlier phases itself and decided to cut some of the part as future work scope. The job searching feature was considered as future scope.