Normalization Analysis of the Campus Insider Database

The current database design for Campus Insider's campus location rating system satisfies **Fifth Normal Form (5NF)**, also known as Project-Join Normal Form. Every relation in the schema represents a single, atomic fact about users, campus locations, or ratings, and all functional, multivalued, and join dependencies have been properly decomposed so that each table's non-key attributes depend entirely on its key, the whole key, and nothing but the key.

The schema includes the following primary entities:
- Logged_in_User(UID, username, password)
- Campus_Location(LID, name, image, …)
- Rating(RID, UID, LID, Score, Comment, Date, …)
- Rating_Equipment(RID, Equipment_Tag)
- Rating_Accessibility(RID, Accessibility_Tag)

Each row in the Rating relation represents a user's evaluation of a single campus location, with its own attributes such as score and comment. The foreign keys UID and LID connect the rating to its author and the location being rated. Because UID and LID together determine the attributes of the rating, all non-key attributes are functionally dependent on the primary key RID. This satisfies BCNF.

Originally, the design included list-type attributes such as Equipment_Tags and Accessibility_Tags inside the Rating table, along with a Ratings attribute for Campus_Locations and Logged_in_User that kept a list of RID's relevant to that location or user. This structure introduced multivalued dependencies:
- Rating.RID →→ Equipment_Tag
- Rating.RID →→ Accessibility_Tag
- Campus_Location.LID →→ Ratings
- Logged_in_User.UID →→ Ratings

The two independent sets of values in the Rating relation violated 4NF because they stored multiple unrelated multivalued attributes in the same relation. To fix this, the schema was decomposed into two separate tables:
- Rating_Equipment(RID, Equipment_Tag)
- Rating_Accessibility(RID, Accessibility_Tag)

Now, each table contains only one multivalued attribute and uses a composite key (RID, Tag) to represent one tag per rating per row. The result is that all multivalued dependencies have been isolated, and each relation satisfies 4NF. Regarding the Ratings attribute under Campus_Location and Logged_in_User, we opted to remove them and clear up any multivalued dependencies to make 4NF clearly achieved. These relationships are now represented naturally

through the foreign keys in the Rating table. As a result, the list of ratings for any user or location can be derived through queries, such as:

- SELECT * FROM Rating WHERE UID = ?
- SELECT * FROM Rating WHERE LID = ?

After decomposing multivalued attributes, no remaining relation contains nontrivial join dependencies. That is to say, each table's rows correspond to one atomic fact, which cannot be subdivided without loss of meaning or the introduction of redundancy.

- Logged_in_User: Each row represents one unique user and their personal information (e.g., username, password). All non-key attributes depend solely on the primary key UID, and there are no multivalued dependencies, since each user's identifying and descriptive attributes form a single atomic entity.
- Campus_Location: Each row represents one distinct campus location. Attributes like Name, Building, and Type are functionally dependent on the primary key LID. Any associations such as user ratings of a location are represented externally through the rating table, ensuring that each location record contains only atomic attributes about the location itself.
- Rating: Each row represents one unique instance of a user rating a specific campus location. This connects a UID and a LID together with a score, comment, timestamp, and other attributes. Each RID (rating ID) identifies one of these atomic relationships between exactly one user and one location.
- Rating_Equipment: Each row represents a single piece of equipment associated with a single rating. The composite key (RID, Equipment_Tag) ensures that each fact expresses exactly one rating-equipment association, eliminating any repeating group or lists of equipment tags within a single rating.
- Rating_Accessibility: Accomplishes the same concepts as Rating_Equipment, representing a single accessibility tag linked to a single rating. The composite key (RID, Accessibility_Tag) ensures that accessibility features are stored as separate, atomic facts rather than as lists or sets within a single rating record.

Together, these decompositions guarantee that every table holds only one type of fact, all attributes within a relation depend fully on that fact's key, and no relation contains multivalued or composite attributes. This level of decomposition is consistent with 4NF and 5NF, as no non-trivial multivalued or join dependencies remain. Since all remaining join dependencies are implied by candidate keys, the highest normal form we achieve with this design is 5NF. Any combination of users, locations, ratings, or tags can be reconstructed solely by joining on keys, with no redundancy or risk of inconsistency. This design technically meets the requirements of 6NF since there are no possible further non-trivial decompositions, but the records lack temporal attributes, with the exception of the timestamp attached to each Rating record. As such, the Campus Insider system can be classified as 5NF.