AISE 3309 – Database Management Systems

**Assignment 02**

Requirement Specification

[Changes to the Model from Assignment 1 1](#_Toc373641307)

[Task 1: Conceptual Model Design 2](#_Toc567079729)

[1.1 Enhanced ER Modelling Explanation 3](#_Toc2097305010)

[Task 2: Conceptual to Logical Model 4](#_Toc53617036)

[Task 3: Relational Schema Evaluation 7](#_Toc1387403023)

[3.1: Schema Flaws and Normalization 8](#_Toc2037293047)

[3.1.1 Entities in 3NF 8](#_Toc928091627)

[3.1.2 Changing Entities to Follow 3NF 13](#_Toc1900113345)

[3.1.3 Final Discussion for Selection 15](#_Toc2053813218)

[Task 3.2: Opportunities for Combining Relations 16](#_Toc553425334)

[Task 3.3: BCNF Evaluation 16](#_Toc183749117)

# Changes to the Model from Assignment 1

While creating the ER model, we made some changes to the specifications we made in assignment 1. For the sake of clarity, we decided to include the major changes we made from assignment 1..

* Foreign key attributes in entities in Assignment 1 were removed from the ER model.l.
* The word Cuisine was change dto FOood to be more concise
* LocationCoordinatesID was changed to a single type attribute to make foreign key inclusions simpler.
* Another attribute for age range was added to Demographic to be more specific.
* RegionID was removed as it was redundant, however DemographicID and PopularityStatsID was added to make foreign key inclusions simpler.
* For Query, RestaurantsOwned removed, businessAffilliation was added.
* For User, DateJoined added.

# **A diagram of a company Description automatically generated**Task 1: Conceptual Model Design

## 1.1 Enhanced ER Modelling Explanation

Aggregation is applied to the relationship between Region and Location and Demographic statistics, as changes made to a Region will result in a different set of locations and demographic statistics. Similarly, aggregation is applied to the relationship between Location and Popularity statistics since different locations may be significantly different in terms of their popularity statistics.

BusinessOwner is a subclass of User, and the relationship between them is {Optional, Or}. A user may or may not be a business owner; if they are a business owner, they have more attributes to describe them in terms of which business this user is affiliated with and what this user’s budget is. The relationship is also disjointed as there is only one subclass to which the entity occurrence can apply.

# Task 2: Conceptual to Logical Model

User (userID, firstName, lastName, dateJoined)

PK: userID

AK: -

FK: -

BusinessOwner (User, Budget)

PK: User

AK: -

FK: User references User(userID)

RegionPreference (User, Region)

PK: User + Region

AK: -

FK: User refers to userID in User

Region references Region(Name)

FoodPreference (User, Food)

PK: User + Food

AK: -

FK: User references User(userID)

Food references Food(Name)

BusinessAffiliation(Affiliation, User)

PK: Affiliation + User

AK: -

FK: User references BusinessOwner(User)

Query (queryID, Date, Time, Function, User, Result)

PK: queryID

AK: Date + Time + User

FK: User references User(userID)

Analysis (queryID, popStatID, demoStatID)

PK: queryID + popStatID + demoStatID

AK: -

FK: queryID references Query(queryID)

popStatID references PopularityStats(ID)

DemoStatID references DemographicStats(ID)

DemographicStats (ID, Region, dateCollected, 0-18, 19-35, 35-55, >55, incomeLevel)

PK: ID

AK: Region + dateCollected

FK: Region references Region(Name)

PopularityStats (ID, Date, footTraffic, realStateBracket, Location)

PK: ID

AK: Date + Location

FK: Location references Location(locCoordinates)

CountryRanks (Country, demographicID, Rank)

PK: Country + demographicID

AK: demographicID + Rank

FK: Country references Country(Name)

demographicID references DemographicStats(demoID)

Country (Name, Population, emmigrationRate, GDP)

PK: Name

AK: -

FK: -

Food (Name, countryOfOrigin)

PK: Name + countryOfOrigin

AK: -

FK: countryOfOrigin references Country(Name)

DietaryCompatibility (foodName, foodOrigin, Diet)

PK: foodName + foodOrigin + Diet

AK: -

FK: foodName references Food(Name)

foodOrigin references Food(countryOfOrigin)

Practices (Country, Religion)

PK: Country + Religion

AK: -

FK: Country references Country(Name)

Religion references Religion(Name)

Religion (Name, DietaryPermittance, VegetarianY/N, AlcoholY/N)

PK: Name

AK: -

FK: -

Region (Name, Coordinates, populationSize, Area, populationDensity)

PK: Name

AK: Coordinates

FK: none

Location (locCoordinatesID, locationSize, Region)

PK: locCoordinatesID

AK: -

FK: Region references Region(Name)

Restaurant (restaurantName, Address, Rating, Franchise, Location)

PK: Address + restaurantName

AK: -

FK: Location references Location(locCoordinatesID)

ServiceStyle(serviceStyle, restaurantName, restaurantAddress)

PK: restaurantName + restaurantAddress

AK: -

FK: restaurantName references Restaurant(restaurantName)

restauratntAddress references Restaurant(Address)

Serves (Restaurant, RestAddress, Food, FoodOrigin)

PK: Restaurant + RestAddress + Food + FoodOrigin

AK: -

FK: Restaurant references Restaurant(Name)

RestAddress references Restaurant(Address)

Food references Food(Name)

FoodOrigin references Food(CountryOfOrigin)

# Task 3: Relational Schema Evaluation

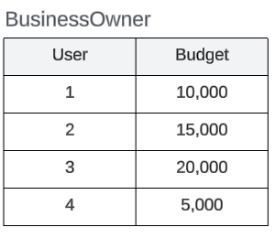
## 3.1: Schema Flaws and Normalization

We are content with our schema and do not think any significant flaws exist. Our schema possesses a few attributes that can hold null entries, which may allow us to perform operations easier in the future. Our topic is complex and has led to many challenges and various opportunities. Many of the relationships between entities were complicated, leading to us needing to create many extra relations to handle the complexity.

### 3.1.1 Entities in 3NF

Entities with only two attributes that are at 2NF are automatically in 3NF. This is because there is necessarily a need for a pair of non-prime attributes. For 3NF to be violated, some non-prime attribute must be dependent on another non-prime attribute. Hence, without at least a pair of those attributes, there cannot be a 3NF violation. This is also the case if all attributes are part of the primary key. If there are no non-prime attributes, there cannot be a violation of 3NF. Brief descriptions are provided for each entity in 3NF below.

**BusinessOwner:** Relation for the subclass of user.



**BusinessAffiliation:** It handles the multi-value attribute businessAffiliation in BusinessOwner.

A table with text on it

Description automatically generated

**FoodPreferences:** It handles the many-to-many relationships between user and food.

A table with text on it

Description automatically generated

**RegionPreferences:** It handles the many-to-many relationships between user and region.

A table with text on it

Description automatically generated

**Analysis:** It handles the many-to-many ternary relationship in which query analyses popularity and demographic stats.

A table with numbers and letters

Description automatically generated

**CountryRank:** It handles the many-to-many relationship and adds a ranking attribute

A table with black text

Description automatically generated

**Food:** It only has two attributes and therefore cannot possibly violate 3NF.

A table with names and words

Description automatically generated with medium confidence

**DietaryCompatibility:** It handles the multi-valued attribute diet. Foodname and foodOrigin are included, because they are the composite primary keys of Food.

A table with different types of food

Description automatically generated

**Practices:** It handles a many to many relationships.

A table of names with black text

Description automatically generated with medium confidence

**ServiceStyle:** It handles a multivalued attribute of restaurant.

A table with a number of restaurants

Description automatically generated

**Serves:** Handles the many-to-many relationship between restaurant and food. Each attribute is both a primary and a foreign key.

A table with text on it

Description automatically generated

**DemographicStats:** It has non-prime attributes with no relationships with each other. There cannot be a dependence between non-prime attributes because those attributes can only be in the same table if held by the primary key. This relationship does not have transitive dependence and is in 3NF.

A table with numbers and numbers

Description automatically generated

**PopularityStats:** Has non-prime attributes with no relationships to each other. There cannot be a dependence between non-prime attributes because those attributes can only be in the same table if held by the primary key. This relationship does not have transitive dependence and is in 3NF.

A table with numbers and numbers

Description automatically generated

**Country:** Country has one primary key, name, and the other attributes are unrelated statistics. Therefore, country is in 3NF.

A table with numbers and text

Description automatically generated

**Religion:** Does not violate one or 2NF because there is only one primary key. It does not violate 3NF because dietaryPermissions only sometimes tell when the religion is also vegetarian or allows alcohol. For example, Catholicism and Hinduism both have ‘none’ for dietaryPermissions but Catholicism allows alcohol, but Hinduism does not.

A table with text on it

Description automatically generated

**User:** In user, there is only one primary key. Additionally, multiple users with the same full name could join on the same day. Also, the dateJoined is not enough information to tell a user’s name since several users could join on the same date. Therefore, it is already in 3NF.

A table of names

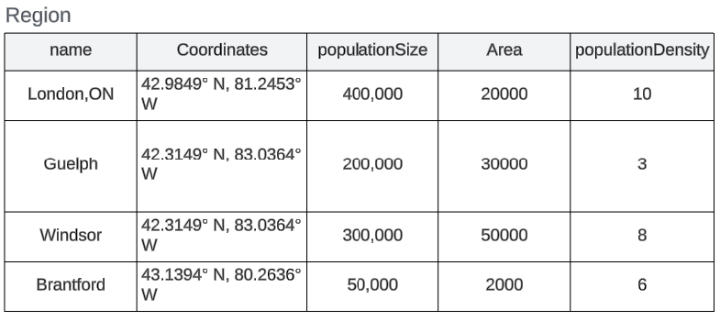
Description automatically generated

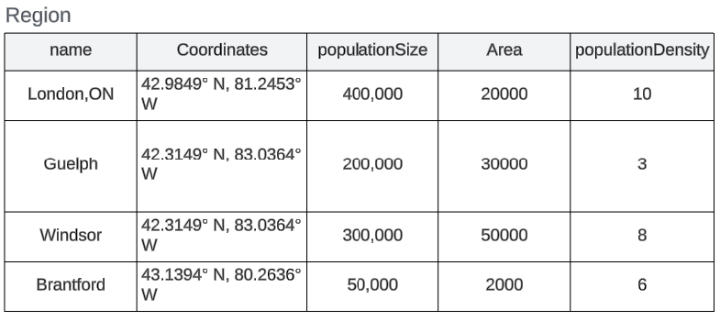
**Location:** It has only one primary key, and locationSize cannot provide the region, and vice versa. That means it is already in 3NF.

A table with numbers and letters

Description automatically generated

**Region:** Region has an alternate composite key of regCoordinates and totalArea, which together can determine the name of a region and hence its populationSize and populationDensity. It is important to note that populationDensity is stored data. Therefore, the values of populationSize and totalArea may or may not derive populationDensity and are not considered a dependency of populationDensity. Therefore, the table is already in 3NF.





### 3.1.2 Changing Entities to Follow 3NF

**Query:** Query possesses a transitive dependency between function, date, and result. The function of a query and the date it is performed dictate the query's result. Since this is done over a queryID primary key, the solution is to create a queryResult table with queryID and result while extracting the result attribute from the query relation.

A table with text and numbers

Description automatically generated with medium confidence

A screenshot of a computer

Description automatically generated

Now, since the same function can be performed at different times and the different functions can be performed at the same time, then there are no more possible transitive dependences, and the relations are in 3NF.

**Restaurants:** Restaurant has partial dependencies as the name will tell if the restaurant is a franchise, and the address can be used to determine the location. This can be resolved by breaking the relation into three, one for the main restaurant attributes, one for address and location, and one for name and franchise.

A screenshot of a computer screen

Description automatically generated

A screenshot of a computer

Description automatically generated

In the new form, the Restaurant relations do not have possible transitive dependencies, meaning they are in 3NF.

### 3.1.3 Final Discussion for Selection

We will select the schemas for future use after normalization is applied. Our project relies on large amounts of statistics that may be updated daily. Using the 3NF forms will reduce the possibility of errors emerging while adding large amounts of data.

## Task 3.2: Opportunities for Combining Relations

Merging is only reasonable when the two relations contain attributes that are often used together and are in a 1…1 relationship. If one of the above conditions is not met, then merging is pointless or will create redundancy.

The only relations that could be considered for merging are the user and business owner. This is because each business owner must be exactly one user, and each user cannot be more than one business owner. However, merging these relations would lead to some user relations having null attributes, which is a situation we have been trying to avoid.

All other relations cannot be merged because they are connected by 0…\* or 1…\* relationships on at least one side. Merging any of those relationships would result in redundancy.

## Task 3.3: BCNF Evaluation

If all the attributes in the table are primary keys, then the entity is already in BCNF, as there are no dependencies between the attributes, meaning it cannot violate BCNF. Any entity with only two attributes, one being the primary key, cannot violate BCNF. These two general rules encompass businessAffiliation, BusinessOwner, regionPreferences, foodPreferences, Analysis, queryResult, countryRanks, location-region, franchise, service, addrLocation, serves, food, practices, and dietaryCompatibility.

This leaves the following entities for analysis: user, query, popularity stats, Demographic Stats, region, location, restaurant, country, and Religion.

In the case of user, all attributes depend on UserID, the primary key. Since there are no other determinants, this satisfies BCNF. The same can be said about the Query relation since all attributes are dependent on the primary key, QueryID.

Even though date, footTraffic, and realestateBracket in popularityStats are candidate keys if they were a single composite key, they satisfy BCNF since they depend only on PopularityID.

Restaurants possess three attributes: name, address, and averageRating, where name and address are composite keys that point to averageRating. AverageRating cannot determine the name of the restaurant or where it is located, nor can any variations that do not include name and address. Address and name alone cannot determine each other or a restaurant's rating. Therefore, the only dependency is that of the composite keys, meaning this entity is in BCNF.

Country has three non-prime attributes that, by definition, have no dependency on each other. Therefore, even making combinations to create dependencies leads nowhere, as they are unrelated data if not for the country's name, which is the primary key. Therefore, similarly to Restaurant, there is only one dependency, and it is derived from the determinant, meaning the table is in BCNF. Similarly, Religion's primary key, "name," is the only attribute from which other attributes draw dependencies.

With popularityStats, the only dependencies exist with the primary key, the ID. There cannot be any violations of the BCNF as there are no other dependencies. The same is true for demographicStats; the only dependencies present are with the ID, which is the primary key.

The remaining attributes of these two entities are data sourced that, alone or combined, cannot determine the remaining attributes.

After dealing with the 3NF violation of Query, the query's function, date, and time can all be found uniquely using queryID. However, the same function can be taken at different times and dates. Also, the same sense that the same time and date can have different functions depending on the queryID. In other words, function, date, and time have no dependencies, meaning this is already at BCNF.

Region has two candidate keys: the primary key "name" and the alternate key "regCoordinates." By definition of the attributes, these are the only determinants that apply in the region context. PopulationSize, totalArea, and populationDensity are sourced data that may or may not relate to each other. Most certainly, regCoordinates or name do not need another determinant to point to the non-attribute keys, making them the only determinants and the table in BCNF.

The location relation also only contains the locCoordinatesID determinant, which is the primary key. Since region has many locations, then multiple equal locationSizes can be drawn for a region, and each locationSize could belong to different regions. That said, they have no dependency on each other and are related by the location coordinates, which makes it the only determinant. Therefore, this is also already in BCNF.

Therefore, it is concluded that once all the attributes in the schema were put into 3NF, they were all already in BCNF.