

**DATABASE**

**SPECIFICATIONS**

*Next-Gen Restaurant Application Database System (NRADS)*

*Nicholas Forleo: nvf5178@psu.edu*

**School of Graduate Professional Studies**

Information Science Department

INSC 521 - Introduction to Database Concepts

Feb. 2024

# Document Control

## Work carried out by:

|  |  |  |
| --- | --- | --- |
| **Name** | **Email Address** | **Other** |
| **Nicholas Forleo** | **nvf5178@psu.edu** |  |
|  |  |  |
|  |  |  |

## Revision Sheet

Changes in **Bold** are new since the last submission.

|  |  |  |
| --- | --- | --- |
| **Release No.** | **Date** | **Revision Description** |
| 1.0 | 01/21/2024 | Milestone 1 Creation |
| 1.1 | 01/26/2024 | - Added username and password to Requirement No. 9  - Changed “shall” to “will”  - Added Purpose and Outcome section to Milestone 1  - Modified DR regarding logs to note location of stored logs  - Separated menu and tab into respective DR  - Modified attributes for all Core Requirements to be more specific and their referenced page numbers |
| 2.0 | 01/29/2024 | - Milestone 2 Creation |
| 2.1 | 02/07/2024 | - Updated diagram to show Staff is-a User relationship  - Deleted and re-inserted diagram to verify it will persist on download (The web version of MS word was giving me problems, will save via MS Teams and submit as PDF from this version on) |
| 3.0 | 02/13/2024 | - Milestone 3 Creation  - Modified DR1 to remove attribute *table\_layout* and replace with directory location where layouts will be saved to  - Modified DR3 to include the last 3 attributes for more accurate data records  - Removed *gratuity* from DR7  - Updated ERD to show Customer pays Transaction relationship |
| **3.1** | **02/25/2024** | **- Expanded all many-to-many type relationships in Logical Model** **(StaffLocations (Works\_At), Locations Layouts (Location\_Contains\_Layout), Layout Tables (Layout\_Contains\_Table), Tab Menu (Tab\_Has\_Menu\_Item))**  **- Changed store\_id to location\_id in Locations table**  **- Added Legend to Logical diagram**  **- Added Function Dependency Descriptor for new tables mentioned above**  **- Modified Month on Title page** |
|  |  |  |

**DATABASE SPECIFICATIONS**

**TABLE OF CONTENTS**

Document Control i

Work carried out by: i

Revision Sheet i

Milestone 1: Data Requirements 1

System Name or Title 1

Core requirements 1

Milestone 2: Conceptual Design 2

Diagram 2

Assumptions and Constraints 2

Milestone 3: Logical Design 3

Entity Relationship Diagram 3

Assumptions and Constraints 3

Milestone 4: Normalization and 4

Milestone 5: Physical Design 4

Assumptions and Constraints 4

Naming Conventions 4

Tables 4

**Examples of values** 4

**Notes** 4

Milestone 6: SQL queries and 6

# Milestone 1: Data Requirements

## System Name or Title

Next-Gen Restaurant Application Database System (NRADS)

**Purpose**

This section will describe the data requirements needed to facilitate the successful creation of the Next-Gen Restaurant Application (NRA) as defined in the Software Requirements Specification (SRS).

**Outcome**

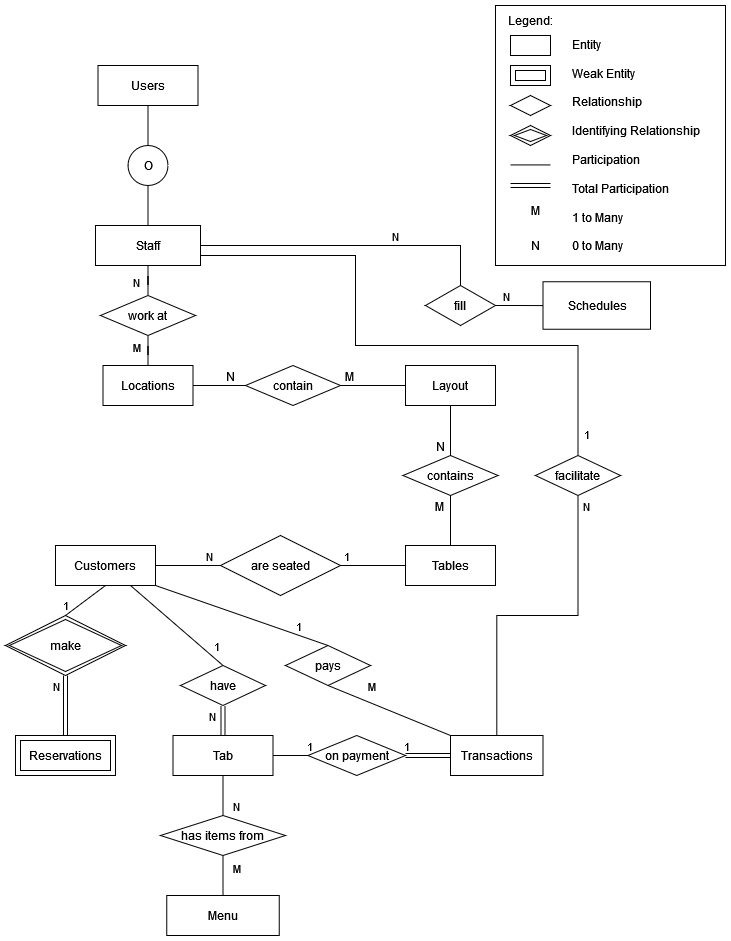
This section will define the data entities and their attributes needed to support the NRA.

## Core requirements

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Requirement | Referenced page in SRS | Referenced Section in SRS | Referenced Paragraph in Section |
| 1 | The system will store general restaurant information including store\_id, address, table layout file location (/home/user/layouts/), table\_occupancy, table\_availability, table\_type, table\_id, layout\_id | 3  5  10  11 | 1.2  2.2  3.5.2  3.5.3 | 1.2.1  2.2.1  2.2.9  all  3.5.3.6 |
| 2 | The system will store customer information including customer\_id, phone\_number, first\_name, last\_name, date\_of\_birth. | 3  5  10  11  13 | 1.2  2.2  3.5.3  3.5.3  5.3 | 1.2.2  2.2.2  3.5.3.2  3.5.3.3  3.5.3.4  5.3.1 |
| 3 | The system will store walk-in and reservation information including reservation\_id, customer\_id, party\_size, wait\_time, is\_walkin, expected\_start\_time, and actual\_start\_time.. | 3  5  5  10  11  12 | 1.2  2.1  2.2  3.5.3  3.5.3  3.5.6 | 1.2.2  3  2.2.2  2.2.7  2.2.9  3.5.3.1  3.5.3.7  3.5.3.8  all |
| 4 | The system will store staff (host/hostess, Servers, Kitchen Staff, Restaurant Management) information including staff\_id, date\_of\_birth, phone\_number, address, and availability. | 3  5  5  11 | 1.2  2.1  2.2  3.5.3 | 1.2.3  1.2.7  3  2.2.2  3.5.3.5 |
| 5 | The system will store transaction information including transaction\_id, staff\_id, customer\_id, tab\_id, total, tip, datetime, currency, payment\_method, and location. | 3  5  5 | 1.2  2.1  2.2 | 1.2.4  3  2.2.4 |
| 6 | The system will store menu information including menu\_item\_id, menu\_item\_description, menu\_item\_name, menu\_item\_ingredients, minimum\_age\_to\_order, current\_inventory, price | 3  5  5  11 | 1.2  2.1  2.2  3.5.4 | 1.2.5  3  2.2.3  2.2.5  2.2.8  all |
| 7 | The system will store tab information including sales\_tax, menu\_item\_id, customer\_id, staff\_id, tab\_total, tab\_id, datetime, customer\_id, and tab\_limit\_amount | 5  9 | 2.2  3.5 | 2.2.5  all |
| 8 | The system will store schedule structure information including required\_days, time, and required\_staff\_levels, staff\_assignment. | 3  5  5 | 1.2  2.1  2.2 | 1.2.7  3  2.2.6 |
| 9 | The system will handle user authentication including username, password, account\_type. | 13 | 5 | 5.1  5.2 |
| 10 | The system will store log information on file system in /var/log/ | 11 | 3.5.5 | all |

# Milestone 2: Conceptual Design

## Diagram



## Assumptions and Constraints

1. Staff ‘is-a’ user.
2. Staff must work at at least one location.
3. A location can have any number of staff assigned to work.
4. A location can contain different layouts.
5. A layout must belong to a location.
6. Staff can fill any number of schedule slots (or none).
7. A schedule can have any number of staff assigned to work.
8. A layout must contain at least one table.
9. A table can belong to any number of layouts.
10. A customer can only be seated at one table.
11. A table can have any number of customers seated at it.
12. A customer can make a reservation but does not have to make a reservation.
13. A reservation must belong to a customer.
14. A tab must belong to a customer.
15. A tab must be compiled of at least one menu item.
16. A menu item can be associated with any number of tabs.
17. A customer can have any number of tabs open.
18. A customer pays to create a transaction.
19. A transaction must close out a tab.
20. A tab can only have one transaction.
21. A staff member must facilitate a transaction.
22. Staff can facilitate any number of transactions.

# Milestone 3: Logical Design

## Functional Dependencies

**Entity name**: Users

**Attributes**:

user\_id, name, position, access\_level

**Functional dependencies**:

user\_id → name, position, access\_level

|  |  |  |  |
| --- | --- | --- | --- |
| **Attributes not in FD** | **Attributes on the left** | **Attributes on both sides** | **Attributes on the right side** |
|  | user\_id |  | name, postion, access\_level |

**Attribute closures**:

user\_id+ = user\_id, name, position, access\_level

**Unique keys**: user\_id

**Entity name**: Staff

**Attributes**:

staff\_id, user\_id, date\_of\_birth, phone\_number, address1, address2, city, state, availability, is\_active, location\_id

**Functional dependencies**:

staff\_id → user\_id, date\_of\_birth, phone\_number, address1, address2, city, state, availability, name, is\_active, location\_id

user\_id → staff\_id, date\_of\_birth, phone\_number, address1, address2, city, state, availability, name, is\_active, location\_id

|  |  |  |  |
| --- | --- | --- | --- |
| **Attributes not in FD** | **Attributes on the left** | **Attributes on both sides** | **Attributes on the right side** |
|  |  | staff\_id, user\_id | date\_of\_birth, phone\_number, address1, address2, city, state, availability, name, is\_active, location\_id |

**Attribute closures** (if any):

user\_id+ = staff\_id, user\_id, date\_of\_birth, phone\_number, address1, address2, city, state, availability, name, is\_active, location\_id

staff\_id+ = staff\_id, user\_id, date\_of\_birth, phone\_number, address1, address2, city, state, availability, name, is\_active, location\_id

**Unique keys**: the key for this table is/are

staff\_id\*

user\_id

Both can be used, but because a user can exist without being staff, we will just use staff\_id to be explicit as possible.

**Entity name**: Scheduled

**Attributes**:

schedule\_id, staff\_id

**Functional dependencies**:

None

|  |  |  |  |
| --- | --- | --- | --- |
| **Attributes not in FD** | **Attributes on the left** | **Attributes on both sides** | **Attributes on the right side** |
| schedule\_id, staff\_id |  |  |  |

**Attribute closures** (if any):

**Unique keys**: the key for this table is/are

schedule\_id, staff\_id (super key)

**Entity name**: Works\_At

**Attributes**:

staff\_id, location\_id

**Functional dependencies**:

none

|  |  |  |  |
| --- | --- | --- | --- |
| **Attributes not in FD** | **Attributes on the left** | **Attributes on both sides** | **Attributes on the right side** |
| staff\_id, location\_id |  |  |  |

**Attribute closures** (if any):

**Unique keys**: the key for this table is/are

staff\_id, location\_id (super key)

**Entity name**: Locations

**Attributes**:

store\_id, address1, address2, city, state, layout\_id

**Functional dependencies**:

store\_id → address1, address2, city, state, layout\_id

|  |  |  |  |
| --- | --- | --- | --- |
| **Attributes not in FD** | **Attributes on the left** | **Attributes on both sides** | **Attributes on the right side** |
|  | store\_id |  | address1, address2, city, state, layout\_id |

**Attribute closures** (if any):

store\_id+ = store\_id, address1, address2, city, state, layout\_id

**Unique keys**: the key for this table is/are

store\_id

**Entity name**: Schedules

**Attributes**:

schedule\_id, required\_start\_time, required\_end\_time, position, filled\_by\_staff\_id

**Functional dependencies**:

schedule\_id → required\_start\_time, required\_end\_time, position, filled\_by\_staff\_id

|  |  |  |  |
| --- | --- | --- | --- |
| **Attributes not in FD** | **Attributes on the left** | **Attributes on both sides** | **Attributes on the right side** |
|  | schedule\_id |  | required\_start\_time, required\_end\_time, position, filled\_by\_staff\_id |

**Attribute closures** (if any):

schedule\_id+ = required\_start\_time, required\_end\_time, position, filled\_by\_staff\_id

**Unique keys**: the key for this table is/are

schedule\_id

**Entity name**: Location\_Contains\_Layouts

**Attributes**:

location\_id, layout\_id, active

**Functional dependencies**:

location\_id, layout\_id → active

|  |  |  |  |
| --- | --- | --- | --- |
| **Attributes not in FD** | **Attributes on the left** | **Attributes on both sides** | **Attributes on the right side** |
|  | location\_id, layout\_id |  | active |

**Attribute closures** (if any):

location\_id, layout\_id+ = active

**Unique keys**: the key for this table is/are

location\_id, layout\_id (super key)

**Entity name**: Layouts

**Attributes**:

layout\_id, layout\_description, layout\_filename, layout\_file\_type, table\_id

**Functional dependencies**:

layout\_id, table\_id → layout\_description, layout\_filename, layout\_file\_type

|  |  |  |  |
| --- | --- | --- | --- |
| **Attributes not in FD** | **Attributes on the left** | **Attributes on both sides** | **Attributes on the right side** |
|  | layout\_id, table\_id |  | layout\_description, layout\_filename, layout\_file\_type |

**Attribute closures** (if any):

layout\_id, table\_id+ = layout\_id, table\_id, layout\_description, layout\_filename, layout\_file\_type

(layout\_id, table\_id) is a superkey

**Unique keys**: the key for this table is/are

layout\_id, table\_id

**Entity name**: Layout\_Contains\_Tables

**Attributes**:

layout\_id, table\_id

**Functional dependencies**:

none

|  |  |  |  |
| --- | --- | --- | --- |
| **Attributes not in FD** | **Attributes on the left** | **Attributes on both sides** | **Attributes on the right side** |
| layout\_id, table\_id |  |  |  |

**Attribute closures** (if any):

**Unique keys**: the key for this table is/are

layout\_id, table\_id (super key)

**Entity name**: Tables

**Attributes**:

table\_id, table\_occupancy, table\_type, table\_notes

**Functional dependencies**:

table\_id → table\_occupancy, table\_type, table\_notes

|  |  |  |  |
| --- | --- | --- | --- |
| **Attributes not in FD** | **Attributes on the left** | **Attributes on both sides** | **Attributes on the right side** |
|  | table\_id |  | table\_occupancy, table\_type, table\_notes |

**Attribute closures** (if any):

table\_id+ = table\_id, table\_occupancy, table\_type, table\_notes

**Unique keys**: the key for this table is/are

table\_id

**Entity name**: Customers

**Attributes**:

customer\_id, date\_of\_birth, phone\_number, first\_name, last\_name, address1, address2, city, state, table\_id

**Functional dependencies**:

customer\_id → date\_of\_birth, phone\_number, first\_name, last\_name, address1, address2, city, state, table\_id

|  |  |  |  |
| --- | --- | --- | --- |
| **Attributes not in FD** | **Attributes on the left** | **Attributes on both sides** | **Attributes on the right side** |
|  | customer\_id |  | date\_of\_birth, phone\_number, first\_name, last\_name, address1, address2, city, state, table\_id |

**Attribute closures** (if any):

customer\_id+ = customer\_id, date\_of\_birth, phone\_number, first\_name, last\_name, address1, address2, city, state, table\_id

**Unique keys**: the key for this table is/are

customer\_id

**Entity name**: Reservations

**Attributes**:

reservation\_id, customer\_id, party\_size, wait\_time, is\_walkin, expected\_start\_time, actual\_start\_time, arrival\_time

**Functional dependencies**:

reservation\_id → customer\_id, party\_size, wait\_time, is\_walkin, expected\_start\_time, actual\_start\_time, arrival\_time

|  |  |  |  |
| --- | --- | --- | --- |
| **Attributes not in FD** | **Attributes on the left** | **Attributes on both sides** | **Attributes on the right side** |
|  | reservation\_id |  | customer\_id, party\_size, wait\_time, is\_walkin, expected\_start\_time, actual\_start\_time, arrival\_time |

**Attribute closures** (if any):

reservation\_id+ = reservation\_id, customer\_id, party\_size, wait\_time, is\_walkin, expected\_start\_time, actual\_start\_time, arrival\_time

**Unique keys**: the key for this table is/are

reservation\_id

**Entity name**: Tabs

**Attributes**:

tab\_id, sales\_tax, menu\_item\_id, customer\_id, staff\_id, datetime, tab\_limit\_amount

**Functional dependencies**:

tab\_id, menu\_item\_id → sales\_tax, customer\_id, staff\_id, datetime, tab\_limit\_amount

|  |  |  |  |
| --- | --- | --- | --- |
| **Attributes not in FD** | **Attributes on the left** | **Attributes on both sides** | **Attributes on the right side** |
|  | tab\_id, menu\_item\_id |  | sales\_tax, customer\_id, staff\_id, datetime, tab\_limit\_amount |

**Attribute closures** (if any):

tab\_id, menu\_item\_id+ = tab\_id, sales\_tax, menu\_item\_id, customer\_id, staff\_id, datetime, tab\_limit\_amount

**Unique keys**: the key for this table is/are

tab\_id, menu\_item\_id

**Entity name**: Tab\_Has\_Menu\_Item

**Attributes**:

tab\_id, menu\_item\_id

**Functional dependencies**:

none

|  |  |  |  |
| --- | --- | --- | --- |
| **Attributes not in FD** | **Attributes on the left** | **Attributes on both sides** | **Attributes on the right side** |
| tab\_id, menu\_item\_id |  |  |  |

**Attribute closures** (if any):

**Unique keys**: the key for this table is/are

tab\_id, menu\_item\_id (super key)

**Entity name**: Menu

**Attributes**:

menu\_item\_id, menu\_item\_description, menu\_item\_name, menu\_item\_ingredients, minimum\_age\_to\_order, current\_inventory, price

**Functional dependencies**:

menu\_item\_id → menu\_item\_description, menu\_item\_name, menu\_item\_ingredients, minimum\_age\_to\_order, current\_inventory, price

|  |  |  |  |
| --- | --- | --- | --- |
| **Attributes not in FD** | **Attributes on the left** | **Attributes on both sides** | **Attributes on the right side** |
|  | menu\_item\_id |  | menu\_item\_description, menu\_item\_name, menu\_item\_ingredients, minimum\_age\_to\_order, current\_inventory, price |

**Attribute closures** (if any):

menu\_item\_id+ = menu\_item\_description, menu\_item\_name, menu\_item\_ingredients, minimum\_age\_to\_order, current\_inventory, price

**Unique keys**: the key for this table is/are

menu\_item\_id

**Entity name**: Transactions

**Attributes**:

transaction\_id, staff\_id, customer\_id, tab\_id, total, tip, datetime, currency, payment\_method, location

**Functional dependencies**:

transaction\_id → staff\_id, customer\_id, tab\_id, total, tip, datetime, currency, payment\_method, location

|  |  |  |  |
| --- | --- | --- | --- |
| **Attributes not in FD** | **Attributes on the left** | **Attributes on both sides** | **Attributes on the right side** |
|  | transaction\_id |  | staff\_id, customer\_id, tab\_id, total, tip, datetime, currency, payment\_method, location |

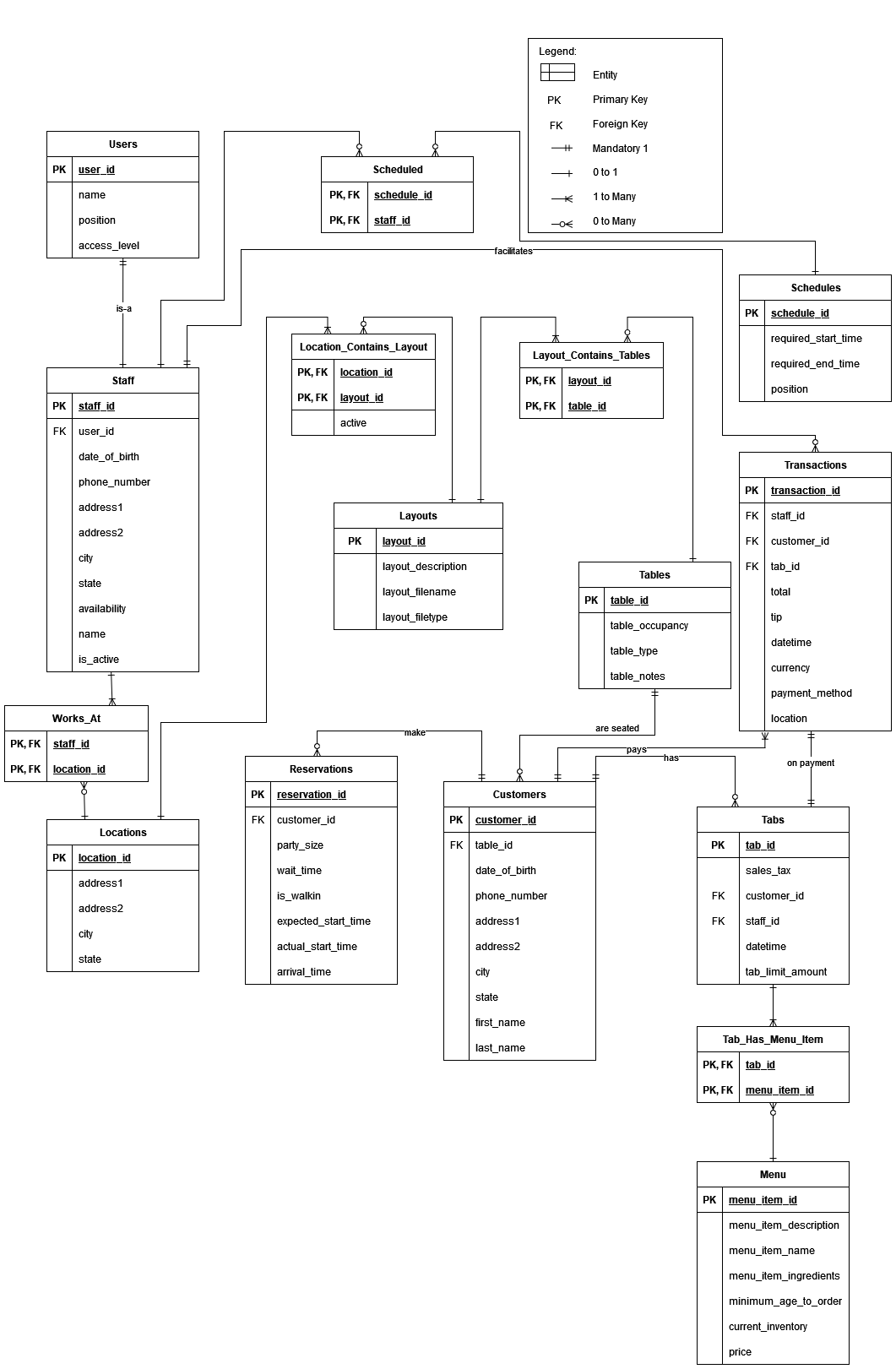
**Attribute closures** (if any):

transaction\_id+ = transaction\_id, staff\_id, customer\_id, tab\_id, total, tip, datetime, currency, payment\_method, location

**Unique keys**: the key for this table is/are

transaction\_id

## Diagram



## Assumptions and Constraints

1. Unless otherwise noted, functional dependencies that only return themselves were not written to keep the document concise.
2. Table availability can be computed based on which customers are seated at a given table.
3. Customer age can be calculated on the fly.
4. Tab total can be calculated on the fly.

# 

# Milestone 4: Normalization and

# Milestone 5: Physical Design

## Assumptions and Constraints

## Naming Conventions

Discuss the naming standards and conventions that you have used for table creation.

## Tables

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | ***Name of the table*** | ***Employee*** | | | | |
|  | **Description** | an employee is a person that works for our university. There are three types of employees: Faculty, staff, and contract employees. | | | | |
|  | **Attribute** | **Description** | | **Type** | **Examples of values** | **Notes** |
|  |  |  | |  |  |  |
|  |  |  | |  |  |  |
|  |  |  | |  |  |  |
|  |  |  | |  |  |  |
|  | **…** |  | |  |  |  |
|  | **Functional Dependencies and Keys** | | | | | |
|  | **Functional dependencies** |  | | | | |
|  | **Candidate keys** | **id** | | | | |
|  | **Normalization** | | | | | |
|  | **1NF** | **Yes** | Reason… | | | |
|  | **2NF** | **Yes** | Reason... | | | |
|  | **3NF** | **Yes** | Reason... | | | |
|  | **BCNF** | **Yes** | Reason... | | | |
|  | **Physical Design** | | | | | |
|  | **Primary Key** |  | | | | |
|  | **Foreign Keys** | **-** | | | | |
|  | **SQL Code** |  | | | | |
|  | **Count of records in the table** | **Note**: Please make sure you add 2 records in each table. | | | | |

...

# Milestone 6: SQL queries and

**Note**: Please make sure you add/have 25 records in each table, on average.

|  |  |
| --- | --- |
| **Query 1** |  |
| **English version** |  |
| **Source for the query need in the SRS document** |  |
| **SQL sentence** |  |
| **Example of returned rows (cropped screen caption)** |  |

…