University of British Columbia, Vancouver
Department of Computer Science

CPSC 304 Project Cover Page

Milestone #: 2

Date: 2023-07-26

Group Number: 34

Name	Student Number	CS Alias (Userid)	Preferred E-mail Address
Kyle Nguyen	76276393	k1s9g	kylnguyen@hotmail.com
Michael Paknys	30618185	k4r6m	mpaknys@icloud.com
Hoi Suen Wong	18947440	t8a8t	Johnnwong@hotmail.com

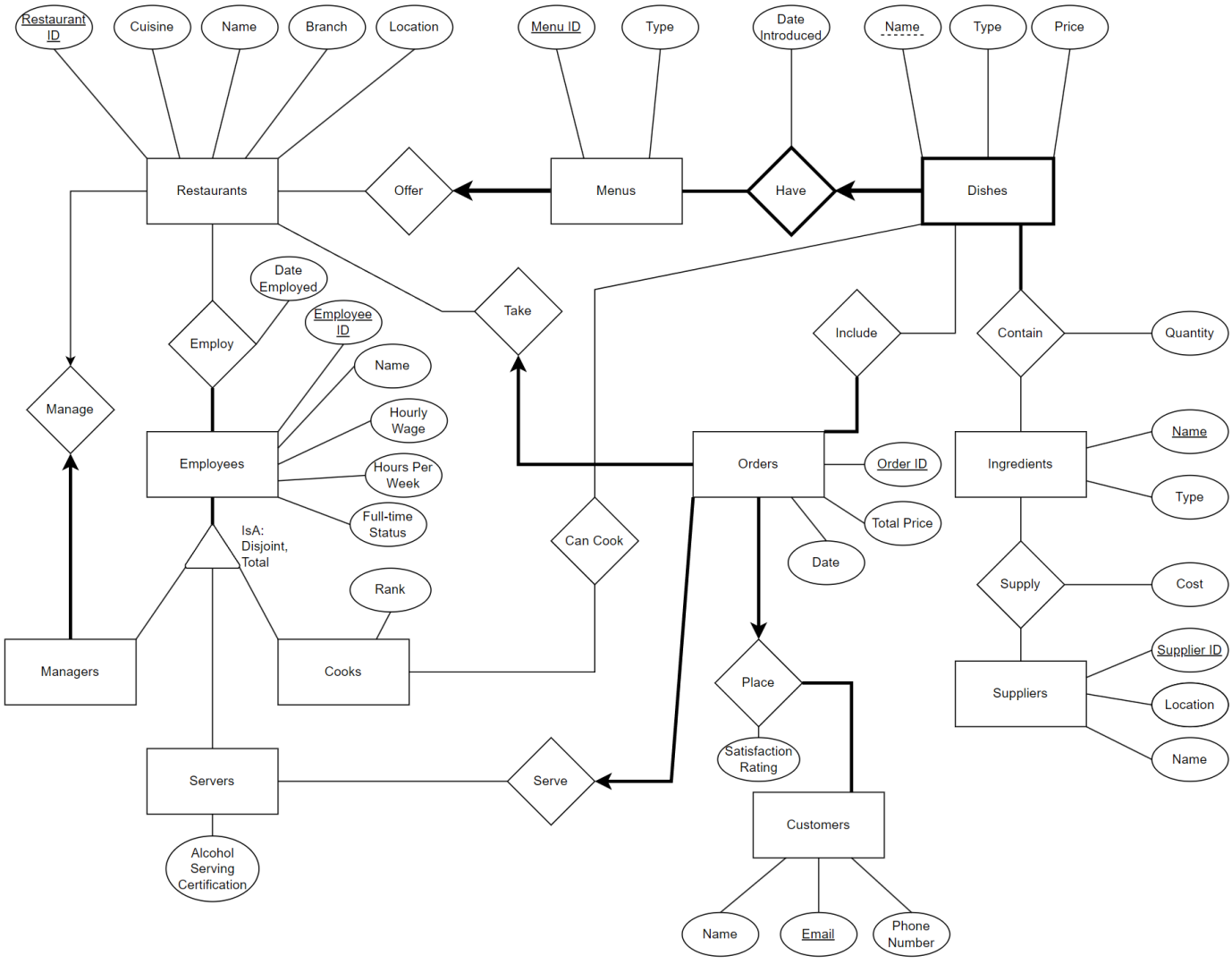
By typing our names and student numbers in the above table, we certify that the work in the attached assignment was performed solely by those whose names and student IDs are included above. (In the case of Project Milestone 0, the main purpose of this page is for you to let us know your e-mail address, and then let us assign you to a TA for your project supervisor.)

In addition, we indicate that we are fully aware of the rules and consequences of plagiarism, as set forth by the Department of Computer Science and the University of British Columbia

ER Diagram

Changes:

- Changed IsA constraints to disjoint from overlapping to keep roles separate
- Changed primary key of Customers to just Email as it is already solely unique
- Changed participation constraints of Restaurants in all relationships with other entity sets to allow for Restaurants entities to exist without constraints
- Added Branch attribute to Restaurants to help produce a candidate key alongside name and location
- Changed Employees attribute of Annual Salary to Hourly Wage, and added Hours Per Week to introduce a functional dependency between Hours Per Week and Full Time Status



Relational Model Schema

Primary Key: PK, Candidate Key: CK, Foreign Key: FK, assume PKs are already considered as CKs

- Restaurants(RID: INTEGER, Cuisine: CHAR(32), RName: CHAR(32), Branch: CHAR(32), RLocation: CHAR(32))
    - RID is PK
    - (RName, Branch, RLocation) is CK
- MenusOffered(MID: INTEGER, MType: CHAR(32), **RID: INTEGER**)
    - MID is PK
    - **RID** is FK referencing Restaurants (cannot be null)
- Suppliers(SID: INTEGER, SName: CHAR(32), SLocation: CHAR(32))
    - SID is PK
    - (SName, SLocation) is CK
- Ingredients(IName: CHAR(32), IType: CHAR(32))
    - IName is PK
- Supplied(**IName: CHAR(32)**, **SID: INTEGER**, Cost: REAL)
    - (IName, SID) is PK
    - **IName** is FK referencing Ingredients
    - **SID** is FK referencing Suppliers
- DishesHad(**MID: INTEGER**, DName: CHAR(32), DType: CHAR(32), Price: REAL, DateIntroduced: DATE)
    - (MID, DName) is PK
    - **MID** is FK referencing MenusOffered (already cannot be null as part of PK)
        - *Menus require at least one dish, will require assertions, total participation constraint cannot be enforced for now*
- Contained(**MID: INTEGER**, **DName: CHAR(32)**, **IName: CHAR(32)**, Quantity: CHAR(32))
    - (MID, DName, IName) is PK
    - **(MID, DName)** is FK referencing DishesHad
    - **IName** is FK referencing Ingredients
        - *Dishes require at least one ingredient, will require assertions, total participation constraint cannot be enforced for now*
- Customers(CEmail: CHAR(32), PhoneNo: CHAR(16), CName: CHAR(32))
    - CEmail is PK
    - (CName, PhoneNo) is CK (in case different customers use same landline)
- Employees(EID: INTEGER, EName: CHAR(32), HourlyWage: REAL, HoursPerWeek: REAL, FulltimeStatus: BOOLEAN)
    - EID is PK
- Servers(**EID: INTEGER**, AlcoholServingCertification: BOOLEAN)
    - EID is PK
    - **EID** is FK referencing Employees

- Cooks(**EID: INTEGER**, Rank: CHAR(32))
    - EID is PK
    - **EID** is FK referencing Employees
- ManagersManaged(**EID: INTEGER**, **RID: INTEGER**)
    - EID is PK
    - **EID** is FK referencing Employees
    - **RID** is FK referencing Restaurants (cannot be null, must be unique)
- Employed(**EID: INTEGER**, **RID: INTEGER**, DateEmployed: DATE)
    - (EID, RID) is PK
    - **EID** is FK referencing Employees
    - **RID** is FK referencing Restaurants
        - *Employees require employment at one restaurant at least, will require assertions, total participation constraint cannot be enforced for now*
- CanCook(**EID: INTEGER**, **MID: INTEGER**, **DName:** CHAR(32))
    - (EID, MID, DName) is PK
    - **EID** is FK referencing Cooks
    - **(MID, DName)** is FK referencing DishesHad
- OrdersPlacedServedTaken(OID: INTEGER, TotalPrice: REAL, Date: DATE, SatisfactionRating: INTEGER, **CEmail: CHAR(32)**, **RID: INTEGER**, **EID: INTEGER**)
    - OID is PK
    - **CEmail** is FK referencing Customers (cannot be null)
        - *Customers require at least one order, will require assertions, total participation constraint cannot be enforced for now*
    - **RID** is FK referencing Restaurants (cannot be null)
    - **EID** is FK referencing Servers (cannot be null)
- Included(**OID: INTEGER**, **MID: INTEGER**, **DName: CHAR(32)**)
    - (OID, MID, DName) is PK
    - **OID** is FK referencing OrdersPlacedServedTaken
    - **(MID, DName)** is FK referencing DishesHad
        - *Orders require at least one dish from a menu, will require assertions, total participation constraint cannot be enforced for now*

Functional Dependencies

PK and CK FDs:

- RID → Cuisine, RName, Branch, RLocation
- RName, Branch, RLocation → RID, Cuisine
- MID → MType, RID
- SID → SName, SLocation
- SName, SLocation → SID
- IName → IType
- IName, SID → Cost
- MID, DName → DType, Price, DateIntroduced
- MID, DName, IName → Quantity
- CEmail → CName, PhoneNo
- CName, PhoneNo → CEmail
- EID → EName, HourlyWage, HoursPerWeek, FulltimeStatus
- EID → AlcoholServingCertification
- EID → Rank
- EID → RID
- EID, RID → DateEmployed
- OID → TotalPrice, Date, SatisfactionRating, CEmail, RID, EID

Other FDs:

- RName → Cuisine
    - E.g. Jo's Italian Deli → Italian
    - E.g. Nelly's Brunch Pantry → Brunch
- HoursPerWeek → FulltimeStatus
    - E.g. 40.00 → True
    - E.g. 20.00 → False

Normalization

Tables in the Relational Model were normalized to be in BCNF. 2 tables were broken down via lossless-join BCNF decomposition as FDs exist that weren't PK or CK dependencies, hence violating BCNF (and 2NF). The rest of the tables were preserved for satisfying BCNF.

- Restaurants(RID: INTEGER, Cuisine: CHAR(32), RName: CHAR(32), Branch: CHAR(32), RLocation: CHAR(32))
    - FD: RID → Cuisine, RName, Branch, RLocation
        - *Satisfies BCNF as RID is a PK*
    - FD: RName, Branch, RLocation → RID, Cuisine
        - *Satisfies BCNF as (RName, Branch, RLocation) is a CK*
    - FD: RName → Cuisine
        - *Violates BCNF as RName is not a superkey*

    - *Decompose to:*
        - RestaurantsMain(RID: INTEGER, RName: CHAR(32), Branch: CHAR(32), RLocation: CHAR(32))
        - RestaurantsName(Cuisine: CHAR(32), RName: CHAR(32))

- Employees(EID: INTEGER, EName: CHAR(32), HourlyWage: REAL, HoursPerWeek: REAL, FulltimeStatus: BOOLEAN)
    - FD: EID → EName, HourlyWage, HoursPerWeek, FulltimeStatus
        - *Satisfies BCNF as EID is a PK*
    - FD: HoursPerWeek → FulltimeStatus
        - *Violates BCNF as* HoursPerWeek *is not a superkey*

    - *Decompose to:*
        - EmployeesMain(EID: INTEGER, EName: CHAR(32), HourlyWage: REAL, HoursPerWeek: REAL)
        - EmployeesFT(HoursPerWeek: REAL, FulltimeStatus: BOOLEAN)

Normalized Relational Model Schema

- RestaurantsMain(RID: INTEGER, RName: CHAR(32), Branch: CHAR(32), RLocation: CHAR(32))
    - RID is PK
    - (RName, Branch, RLocation) is CK
- RestaurantsName(Cuisine: CHAR(32), RName: CHAR(32))
    - RName is PK
- MenusOffered(MID: INTEGER, MType: CHAR(32), **RID: INTEGER**)
    - MID is PK
    - **RID** is FK referencing RestaurantsMain (cannot be null)
- Suppliers(SID: INTEGER, SName: CHAR(32), SLocation: CHAR(32))
    - SID is PK
    - (SName, SLocation) is CK
- Ingredients(IName: CHAR(32), IType: CHAR(32))
    - IName is PK
- Supplied(**IName: CHAR(32)**, **SID: INTEGER**, Cost: REAL)
    - (IName, SID) is PK
    - **IName** is FK referencing Ingredients
    - **SID** is FK referencing Suppliers
- DishesHad(**MID: INTEGER**, DName: CHAR(32), DType: CHAR(32), Price: REAL, DateIntroduced: DATE)
    - (MID, DName) is PK
    - **MID** is FK referencing MenusOffered (already cannot be null as part of PK)
        - *Menus require at least one dish, will require assertions, total participation constraint cannot be enforced for now*
- Contained(**MID: INTEGER**, **DName: CHAR(32)**, **IName: CHAR(32)**, Quantity: CHAR(32))
    - (MID, DName, IName) is PK
    - **(MID, DName)** is FK referencing DishesHad
    - **IName** is FK referencing Ingredients
        - *Dishes from a menu require at least one ingredient, will require assertions, total participation constraint cannot be enforced for now*
- Customers(CEmail: CHAR(32), PhoneNo: CHAR(16), CName: CHAR(32))
    - CEmail is PK
    - (CName, PhoneNo) is CK (in case different customers use same landline)
- EmployeesMain(EID: INTEGER, EName: CHAR(32), HourlyWage: REAL, HoursPerWeek: REAL)
    - EID is PK
- EmployeesFT(HoursPerWeek: REAL, FulltimeStatus: BOOLEAN)
    - HoursPerWeek is PK

- Servers(**EID: INTEGER**, AlcoholServingCertification: BOOLEAN)
    - EID is PK
    - **EID** is FK referencing EmployeesMain
- Cooks(**EID: INTEGER**, Rank: CHAR(32))
    - EID is PK
    - **EID** is FK referencing EmployeesMain
- ManagersManaged(**EID: INTEGER**, **RID: INTEGER**)
    - EID is PK
    - **EID** is FK referencing EmployeesMain
    - **RID** is FK referencing RestaurantsMain (cannot be null, must be unique)
- Employed(**EID: INTEGER**, **RID: INTEGER**, DateEmployed: DATE)
    - (EID, RID) is PK
    - **EID** is FK referencing EmployeesMain
    - **RID** is FK referencing RestaurantsMain
        - *Employees require employment at one restaurant at least, will require assertions, total participation constraint cannot be enforced for now*
- CanCook(**EID: INTEGER**, **MID: INTEGER**, **DName:** CHAR(32))
    - (EID, MID, DName) is PK
    - **EID** is FK referencing Cooks
    - **(MID, DName)** is FK referencing DishesHad
- OrdersPlacedServedTaken(OID: INTEGER, TotalPrice: REAL, Date: DATE, SatisfactionRating: INTEGER, **CEmail: CHAR(32)**, **RID: INTEGER**, **EID: INTEGER**)
    - OID is PK
    - **CEmail** is FK referencing Customers (cannot be null)
        - *Customers require at least one order, will require assertions, total participation constraint cannot be enforced for now*
    - **RID** is FK referencing RestaurantsMain (cannot be null)
    - **EID** is FK referencing Servers (cannot be null)
- Included(**OID: INTEGER**, **MID: INTEGER**, **DName: CHAR(32)**)
    - (OID, MID, DName) is PK
    - **OID** is FK referencing OrdersPlacedServedTaken
    - **(MID, DName)** is FK referencing DishesHad
        - *Orders require at least one dish from a menu, will require assertions, total participation constraint cannot be enforced for now*

Relation Instances

Please see .sql file for the DDL on creating and inserting into tables.
Please see excel file for the mock tables.