Problem Statement

P.O.G. : Gaming Cafe

P.O.G. will be opening soon to the public as an internet gaming cafe. At this cafe we provide PC's for gaming/personal usage, and serve coffee/drinks. The PC's at the cafe are updated constantly with the latest games. Also, the hardware for the PC's are to be changed every few years. There are a certain number of PC's with each PC equipped with a gaming headset and gaming chair that are to be sanitized after use by the employee's. Also, the PC's have a dividing wall in between each other in order to have privacy for the customers and each PC has an electronic monitor to order any drinks from the cafe. The hours of operation at P.O.G. are from 6am to 12pm except on the weekends where the hours of operations are from 6am to 2am. There is a server room that is for both protection against any customers that may accidentally wander in and help have connection to a network for all the PC's in the cafe.

Supplier / Inventory

The supplies at the gaming cafe are split into two categories the gaming/PC supplies and the cafe supplies. For both suppliers we will need their email addresses, phone number, and a SID.

The supplies for the gaming/PC include hardware, gaming chairs, monitors, headphones and sanitizers. The number of PC's must be equivalent to the number of gaming chairs and headphones. We must also keep track of the internet usage and servers are good at the gaming cafe. The number of games are usually 30 for each PC but customers are allowed to download and buy games for their assigned PC serial number.

As for the supplies the cafe has the coffee beans, milk, and other ingredients for specialty drinks sourced from farms, and the equipment/machinery to make coffee and specialty drinks being sourced from a manufacturer from a factory. Both farms and factories will have their own IDs. There is a menu with different prices for the drinks at the cafe that is given to the customer when they arrive at their PC or if they decide to order directly from the cafe. Also, we must keep track of the expiration dates on all consumable goods since they are perishable.

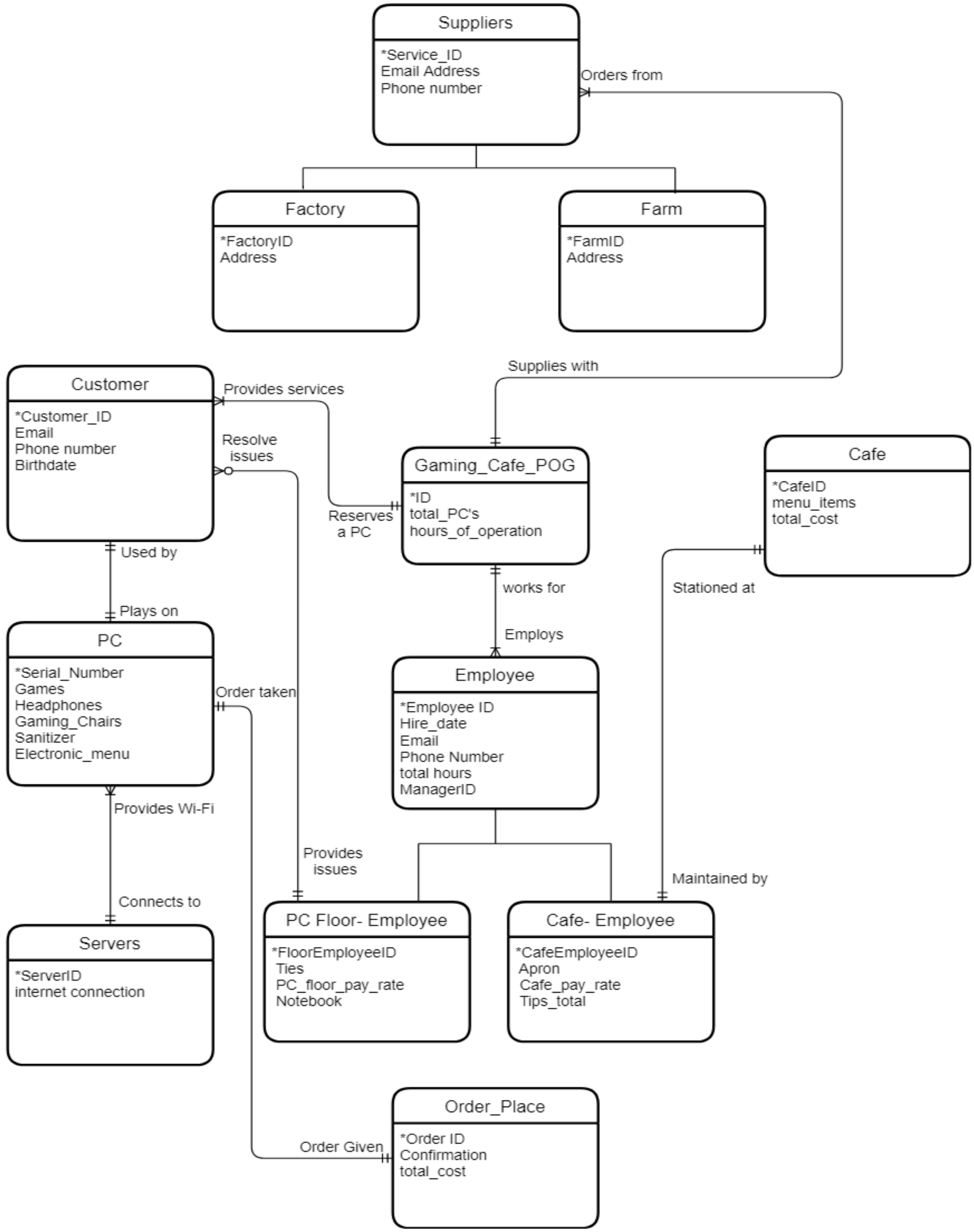
Employee's

Employee's are are tracked by a database with information on them such as their employee ID, email address, primary address, whether their position is on the PC floor or at the cafe, when did they start working for P.O.G., total amount of hours to be worked in a week. There is an employee who manages other employees and they have their own manager ID..The employees will be split into mainly two categories, the employees who work in the area with the PC's and the other employees will work at the cafe. The employees in the PC area will bring the orders placed at the customer's PC from the cafe to the customer. Also, the employees from the PC area will walk around and act as customer support while carrying a notebook to record any issues. The employees in the cafe are required to wear an apron and the employees working in the PC area are required to wear a red tie. However, if an employee is part time working in the PC area they will be wearing a white-red tie. The pay rate for employees that work in the cafe and PC area are different and only people that work at the cafe are allowed to receive tips.

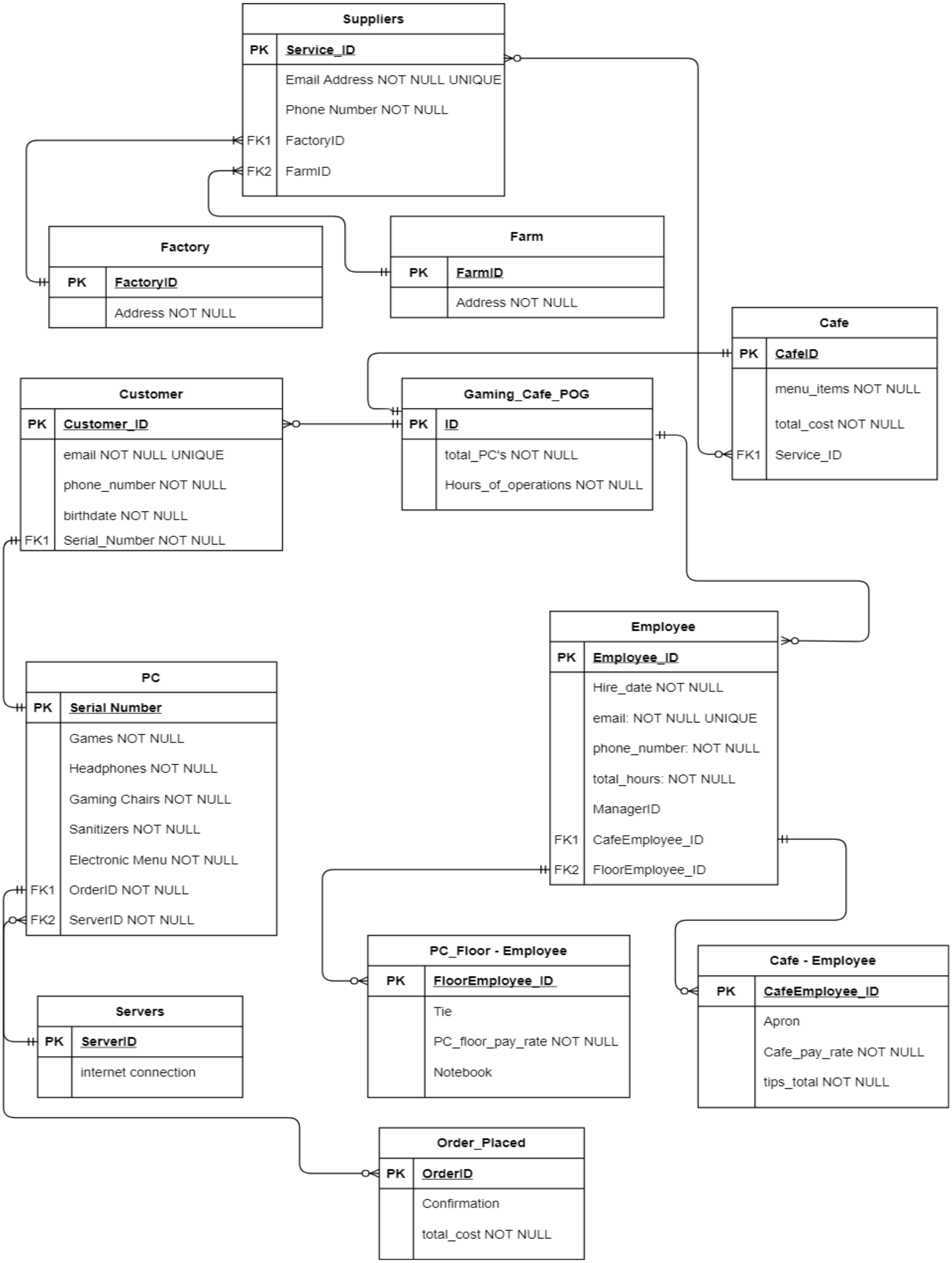
Customers

When customers first arrive at the cafe for the first time they are assigned a certain PC based on the PC's serial number to use and make an account on the PC to save their browsing/gaming progress. So, from this it means that only one customer is assigned to only one specific PC. Customers are also given an ID after checking in their first visit providing their email address, phone number, name, and date of birth in order to keep a log of the customers history at the gaming cafe. Customers are expected to be clean and are offered sanitizer at their PC in case they need to sanitize. The conditions of things such as headphones, gaming chair, and amount of sanitizer are checked within the POG cafe databases.

ERD



Relational Model Diagram



Suppliers	
PK	Service_ID
	Email Address NOT NULL UNIQUE
	Phone Number NOT NULL
FK1	FactoryID
FK2	FarmID

Factory	
PK	FactoryID
	Address NOT NULL

Farm	
PK	FarmID
	Address NOT NULL

Cafe	
PK	CafeID
	menu_items NOT NULL
	total_cost NOT NULL
FK1	Service_ID

Customer	
PK	Customer_ID
	email NOT NULL UNIQUE
	phone_number NOT NULL
	birthdate NOT NULL
FK1	Serial_Number NOT NULL

Gaming_Cafe_POG	
PK	ID
	total_PC's NOT NULL
	Hours_of_operations NOT NULL

Employee	
PK	Employee_ID
	Hire_date NOT NULL
	email: NOT NULL UNIQUE
	phone_number: NOT NULL
	total_hours: NOT NULL
	ManagerID
FK1	CafeEmployee_ID
FK2	FloorEmployee_ID

PC	
PK	Serial Number
	Games NOT NULL
	Headphones NOT NULL
	Gaming Chairs NOT NULL
	Sanitizers NOT NULL
	Electronic Menu NOT NULL
FK1	OrderID NOT NULL
FK2	ServerID NOT NULL

PC_Floor - Employee	
PK	FloorEmployee_ID
	Tie
	PC_floor_pay_rate NOT NULL
	Notebook

Cafe - Employee	
PK	CafeEmployee_ID
	Apron
	Cafe_pay_rate NOT NULL
	tips_total NOT NULL

Servers	
PK	ServerID
	internet connection

Order_Placed	
PK	OrderID
	Confirmation
	total_cost NOT NULL

Tables

Gaming_Cafe_P.O.G. *(ID PRIMARY KEY, total_PC's NOT NULL, Hour_ of_operation NOT NULL)*

Employee (*Employee_ID PRIMARY KEY, Hire_date NOT NULL, email NOT NULL UNIQUE, phone_number NOT NULL, total_hours NOT NULL, ManagerID, CafeEmployee_ID REFERENCES Cafe_Employee.CafeEmployee_ID, FloorEmployee_ID REFERENCES PC_Floor_Employee.FloorEmployee_ID*)

PC Floor - Employee *(FloorEmployee_ID PRIMARY KEY, Tie, PC_floor_pay_rate NOT NULL, Notebook)*

Cafe - Employee *(CafeEmployee_ID PRIMARY KEY, Apron, Cafe_pay_rate NOT NULL, tips_total NOT NULL)*

Cafe *(CafeID PRIMARY KEY, menu_items NOT NULL, total_cost NOT NULL, ServiceID REFERENCES Suppliers.Service_ID)*

Customer *(Customer_ID PRIMARY KEY, email NOT NULL UNIQUE, phone_number NOT NULL, birthdate NOT NULL, Serial_Number REFERENCES PC.Serial_Number)*

PC *(Serial Number PRIMARY KEY, Games NOT NULL, Headphones NOT NULL, Gaming_chairs NOT NULL, Sanitizer NOT NULL, Electronic_menu NOT NULL, OrderID REFERENCES Order_Placed.OrderID, ServerID REFERENCES Servers.ServerID)*

Order_Placed *(OrderID PRIMARY KEY, Confirmation, total_cost)*

Servers *(ServerID PRIMARY KEY, internet_connection)*

Suppliers *(Service_ID PRIMARY KEY, Email_Address NOT NULL UNIQUE, Phone_number NOT NULL, FactoryID REFERENCES Factory.FactoryID, FarmID REFERENCES Farm.FarmID)*

Factory *(FactoryID PRIMARY KEY, Address NOT NULL)*

Farms *(FarmID PRIMARY KEY, Address NOT NULL)*

Functional Dependencies

Gaming_Cafe_P.O.G. *(ID, total_PC's, Hour_ of_operation)*

F = {ID -> total_PC's, Hour_ of_operation }

Employee (*Employee_ID, Hire_date, email, phone_number, total_hours, ManagerID, CafeEmployee_ID, FloorEmployee_ID*)

F = {Employee_ID -> Hire_date, email, phone_number, total_hours, ManagerID,

CafeEmployee_ID, FloorEmployee_ID}

PC Floor - Employee *(FloorEmployee_ID, Tie, PC_floor_pay_rate, Notebook)*

F = {Floor*Employee_ID -> Tie, PC_floor_pay_rate, Notebook*}

Cafe - Employee *(CafeEmployee_ID, Apron, Cafe_pay_rate, tips_total)*

F = {CafeEmployee_ID -> Apron, Cafe_pay_rate, tips_total}

Cafe *(CafeID ,menu_items, total_cost, Service_ID)*

F = $\{CafeID\ -> menu\_items,\ total\_cost,\ Service\_ID\}$

Customer *(Customer_ID, email, phone_number, birthdate, Serial_Number)*

F = {Customer_ID -> email, phone_number, birthdate, Serial_Number}

PC *(Serial_Number, Games, Headphones, Gaming_chairs, Sanitizer, Electronic_menu,*

OrderID, ServerID)

F = {Serial_Number -> Games, Headphones, Gaming_chairs, Sanitizer, Electronic_menu,

OrderID, ServerID}

Order_Placed *(OrderID, Confirmation, total_cost)*

F = {OrderID -> Confirmation, total_cost}

Servers *(ServerID, Internet_connection)*

F = {ServerID -> Internet_connection}

Suppliers *(Service_ID, Email_Address, Phone_number, FactoryID, FarmID)*

F = {Service_ID -> Email_Address, Phone_number, FactoryID, FarmID}

Factory *(FactoryID, Address)*

F = {FactoryID -> Address}

Farms *(FarmID, Address)*

F = **{***FarmID -> Address}*

Decomposition Into BCNF/4NF

Gaming_Cafe_P.O.G. *(ID, total_PC's, Hour_ of_operation)*

F = {ID -> total_PC's, Hour_ of_operation }

$CKs = \{ID\}$

This relation is in BCNF because the antecedent of the only functional dependency is a super key for the whole relation.

Employee (*Employee_ID, Hire_date, email, phone_number, total_hours, ManagerID, CafeEmployee_ID, FloorEmployee_ID*)

F = {Employee_ID -> Hire_date, email, phone_number, total_hours, ManagerID,

CafeEmployee_ID, FloorEmployee_ID}

CK = {Employee_ID}

This relation is in BCNF because the antecedent of the only functional dependency is a prime attribute.

PC Floor - Employee *(FloorEmployee_ID, Tie, PC_floor_pay_rate, Notebook)*

F = {Floor*Employee_ID -> Tie, PC_floor_pay_rate, Notebook*}

CK = {FloorEmployee_ID}

This relation is in BCNF because the antecedent of the only functional dependency is a super key for the whole relation.

Cafe - Employee *(CafeEmployee_ID, Apron, Cafe_pay_rate, tips_total)*

F = {CafeEmployee_ID -> Apron, Cafe_pay_rate, tips_total}

CK = {CafeEmployee_ID}

This relation is in BCNF because the antecedent of the only functional dependency is a super key for the whole relation.

Cafe *(CafeID ,menu_items, total_cost, Service_ID)*

F = $\{CafeID\ \ -> \ menu\_items,\ total\_cost,\ Service\_ID\}$

CK = {CafeID}

This relation is in BCNF because the antecedent of the only functional dependency is a super key for the whole relation.

Customer *(Customer_ID, email, phone_number, birthdate, Serial_Number)*

F = {Customer_ID -> email, phone_number, birthdate, Serial_Number}

CK = {CustomerID}

This relation is in BCNF because the antecedent of the only functional dependency is a super key for the whole relation.

PC *(Serial_Number, Games, Headphones, Gaming_chairs, Sanitizer, Electronic_menu, OrderID, ServerID)*

F = {Serial_Number -> Games, Headphones, Gaming_chairs, Sanitizer, Electronic_menu,

OrderID, ServerID}

CK = {Serial_Number}

This relation is in BCNF because the antecedent of the only functional dependency is a super key for the whole relation.

Order_Placed *(OrderID, Confirmation, total_cost)*

F = {OrderID -> Confirmation, total_cost}

CK = {OrderID}

This relation is in BCNF because the antecedent of the only functional dependency is a super key for the whole relation.

Servers *(ServerID, Internet_connection)*

F = {ServerID -> Internet_connection}

$CKs = \{ServerID\}$

This relation is in BCNF because the antecedent of the only functional dependency is a super key for the whole relation.

Suppliers *(Service_ID, Email_Address, Phone_number, FactoryID, FarmID)*

F = {Service_ID -> Email_Address, Phone_number, FactoryID, FarmID}

CKs = {Service_ID}

This relation is in BCNF because the antecedent of the only functional dependency is a super key for the whole relation.

Factory *(FactoryID, Address)*

F = {FactoryID -> Address}

CKs = {FactoryID}

This relation is in BCNF because the antecedent of the only functional dependency is a super key for the whole relation.

Farms *(FarmID, Address)*

F = **{***FarmID -> Address}*

CKs = {FarmID}

This relation is in BCNF because the antecedent of the only functional dependency is a super key for the whole relation.