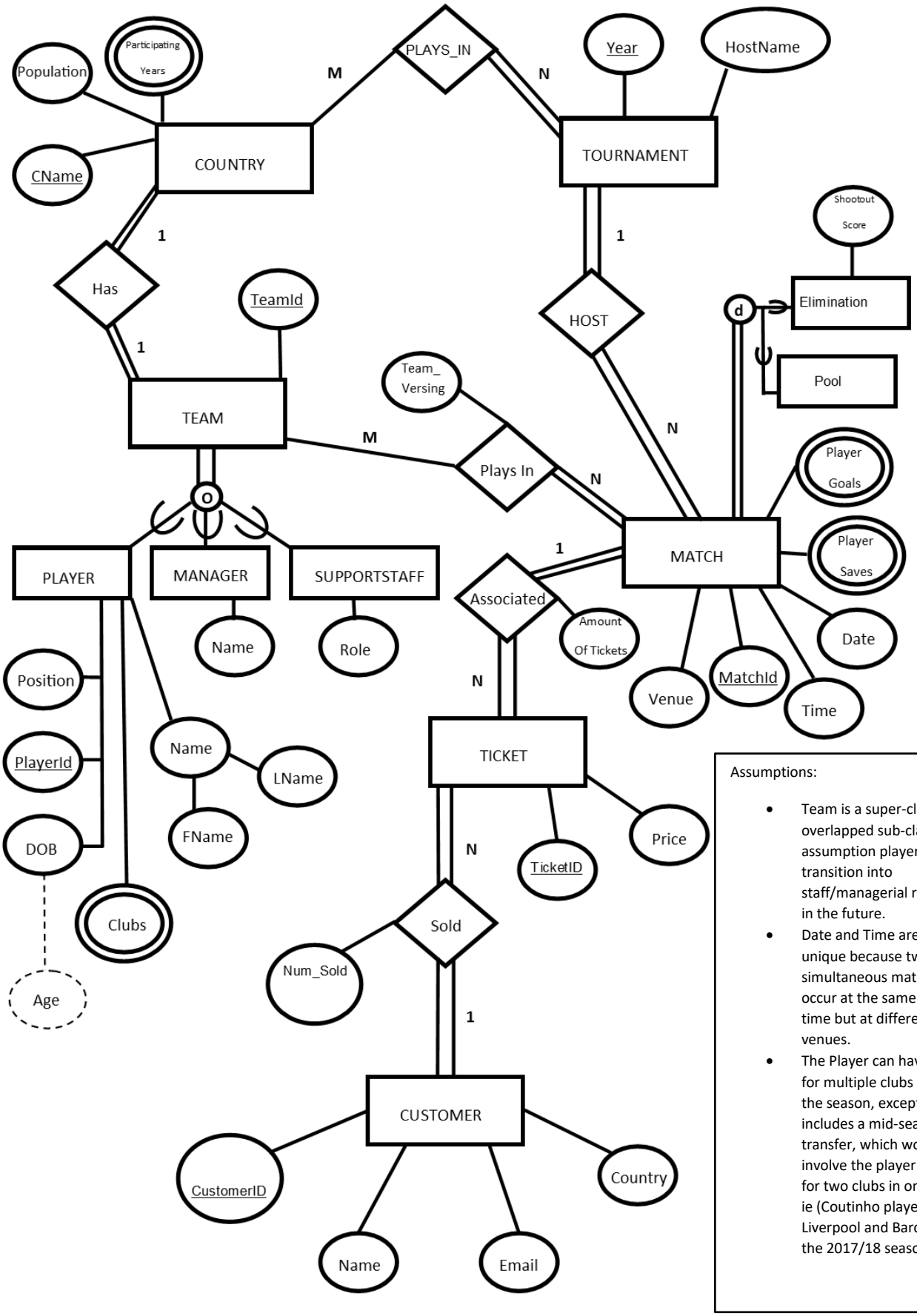
Assignment 1 Template

Group Member 1: Nikhil Naik, 45293858

Group Member 2: Chris Nebe, 45377619

YOUR ER DIAGRAM MUST FIT ON THIS PAGE



Assumptions:

- Team is a super-class with overlapped sub-class, assumption players can transition into staff/managerial roles later in the future.
- Date and Time are not unique because two simultaneous matches can occur at the same date and time but at different venues.
- The Player can have played for multiple clubs during the season, exception includes a mid-season transfer, which would involve the player playing for two clubs in one season. ie (Coutinho played for Liverpool and Barcelona in the 2017/18 season.)

YOUR FINAL MAPPING MUST FIT ON THIS PAGE

Schema
TEAM[TeamID]
PLAYER[PlayerID,TeamID,TeamName,FName,LName,DOB,Position,CName]
MANAGER[TeamID,TeamName,FName,LName]
SUPPORTSTAFF[TeamID,TeamName,FName,LName,Role]
COUNTRY[CName,Population,ParticipatingYears]
TOURNAMENT[Year,HostName]
PARTICIPATES[CName,Year]
PLAYS_IN[TeamID,MatchID,Teams_Versing]
Pool[MatchID,Date,Time,Venue,Year]
Elimination[MatchID,Date,Time,Venue, ShootoutScore,Year]
CUSTOMER[CustomerID,Name,Email,Country]
Ticket[TicketID,Price,MatchID,Amount of Tickets,CustomerID,Num_Sold]
PLAYER_CLUBS[PlayerID,Clubs]
COUNTRIES_YEARS_PLAYED[CName,ParticipatingYears]
PLAYER_GOALS[MatchID,PlayerGoals]
PLAYER_SAVES[MatchID,PlayerSaves]

Foreign Keys
PLAYER.TeamID -> TEAM.TeamID
MANAGER.TeamID -> TEAM.TeamID
SUPPORTSTAFF.TeamID -> TEAM.TeamID
PLAYER.CName -> COUNTRIES.CName
PARTICIPATES.CName -> COUNTRIES.CName
PARTICIPATES.Year -> TOURNAMENT.Year
PLAYS_IN.TeamID -> TEAM.TeamID
PLAYS_IN.MatchID -> MATCH.MatchID
Pool.MatchID -> MATCH.MatchID
Elimination.MatchID -> MATCH.MatchID
TICKET.MatchID -> MATCH.MatchID
TICKET.CustomerID -> CUSTOMER.CustomerID
PLAYER_CLUBS.PlayerID -> PLAYER.PlayerID
COUNTRIES_YEARS_PLAYED.CName -> COUNTRIES.CName
PLAYER_GOALS.MatchID -> MATCH.MatchID
PLAYER_SAVES.MatchID -> MATCH.MatchID

YOUR FUNCTIONAL DEPENDENCIES MUST FIT ON THIS PAGE

Functional Dependencies:

Fd1:

{ProductID} **References** (Product_Description, Product_Type, Senior_Discount_Eligibility, Alcoholic_Nature)

Fd2:

{VendorID} **References** (Vendor_Name)

Fd3:

{MatchID,VendorID } **References** (Quantity)

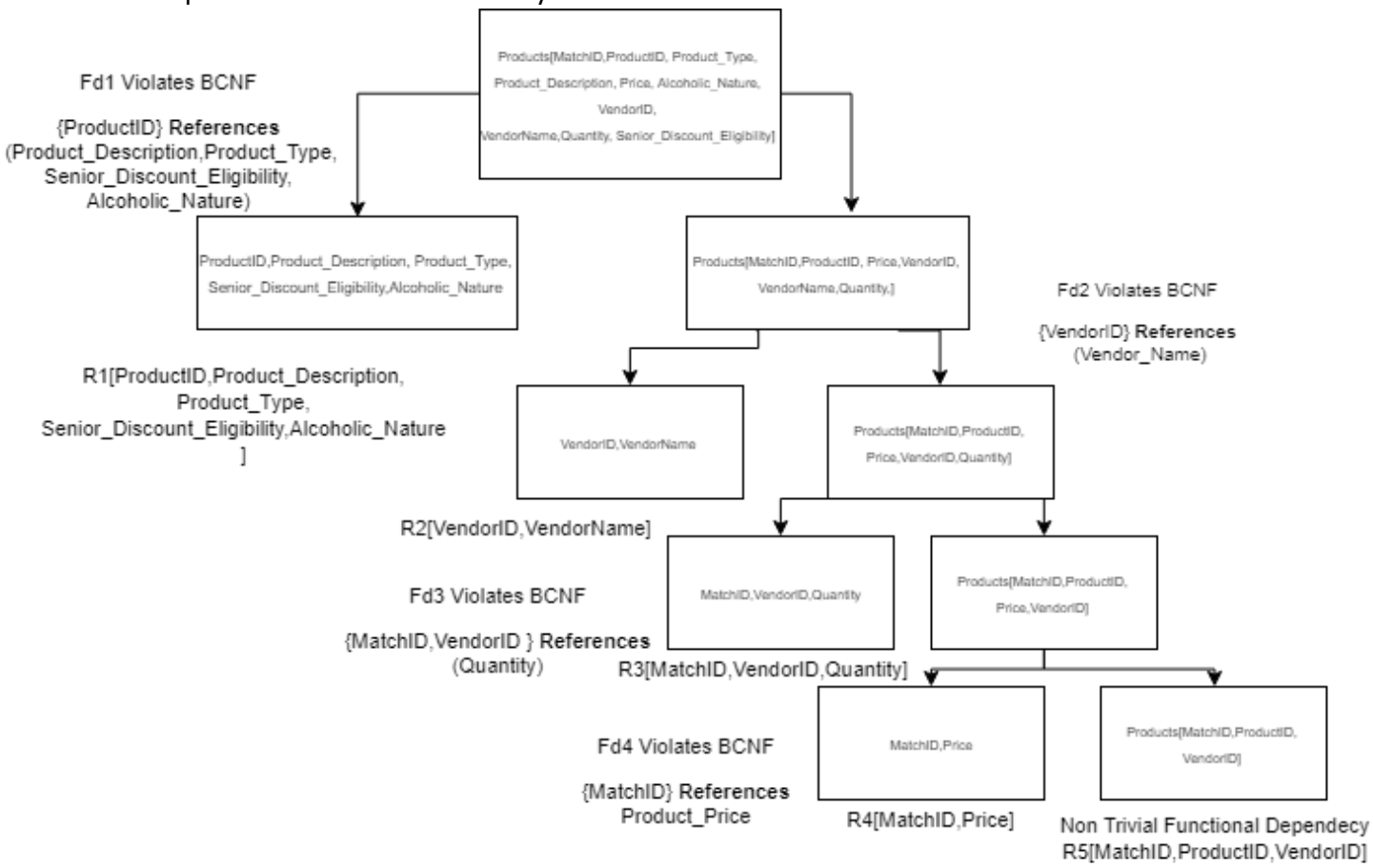
Fd4:

{MatchID} **References** Product_Price

YOUR NORMALISATION MUST FIT ON THIS PAGE

BCNF Normalisation:

1. Using functional dependencies from part three:
   Fd1:
   {ProductID} **References** (Product_Description, Product_Type, Senior_Discount_Eligibility, Alcoholic_Nature)
   Fd2:
   {VendorID} **References** (Vendor_Name)
   Fd3:
   {MatchID,VendorID } **References** (Quantity)
   Fd4:
   {MatchID} **References** Price
   PRODUCTS[MatchID, ProductID, ProductType, Description, Price, Alcoholic, VendorID, VendorName, Quantity, SeniorDiscount]
   Candidate Key: { ProductID , VendorID , MatchID }+
2. Decomposed Product Candidate Key is achieved



3. Final Answer:

R1[ProductID,Product_Description, Product_Type, Senior_Discount_Eligibility,Alcoholic_Nature	Fd1:{ProductID} References (Product_Description, Product_Type, Senior_Discount_Eligibility, Alcoholic_Nature)
R2[VendorID,VendorName]	Fd2:{VendorID} References (Vendor_Name)
R3[MatchID,VendorID,Quantity]	Fd3:{MatchID,VendorID } References (Quantity)
R4[MatchID,Price]	Fd4:{MatchID} References Product_Price
R5[MatchID,ProductID,VendorID]	Non Trivial Functional Dependecy