MIDDLE EAST TECHNICAL UNIVERSITY

NORTHERN CYPRUS CAMPUS

DEPARTMENT
OF COMPUTER ENGINEERING

CNG 351
Data Management and File Structures
Assignment 1

Team Details

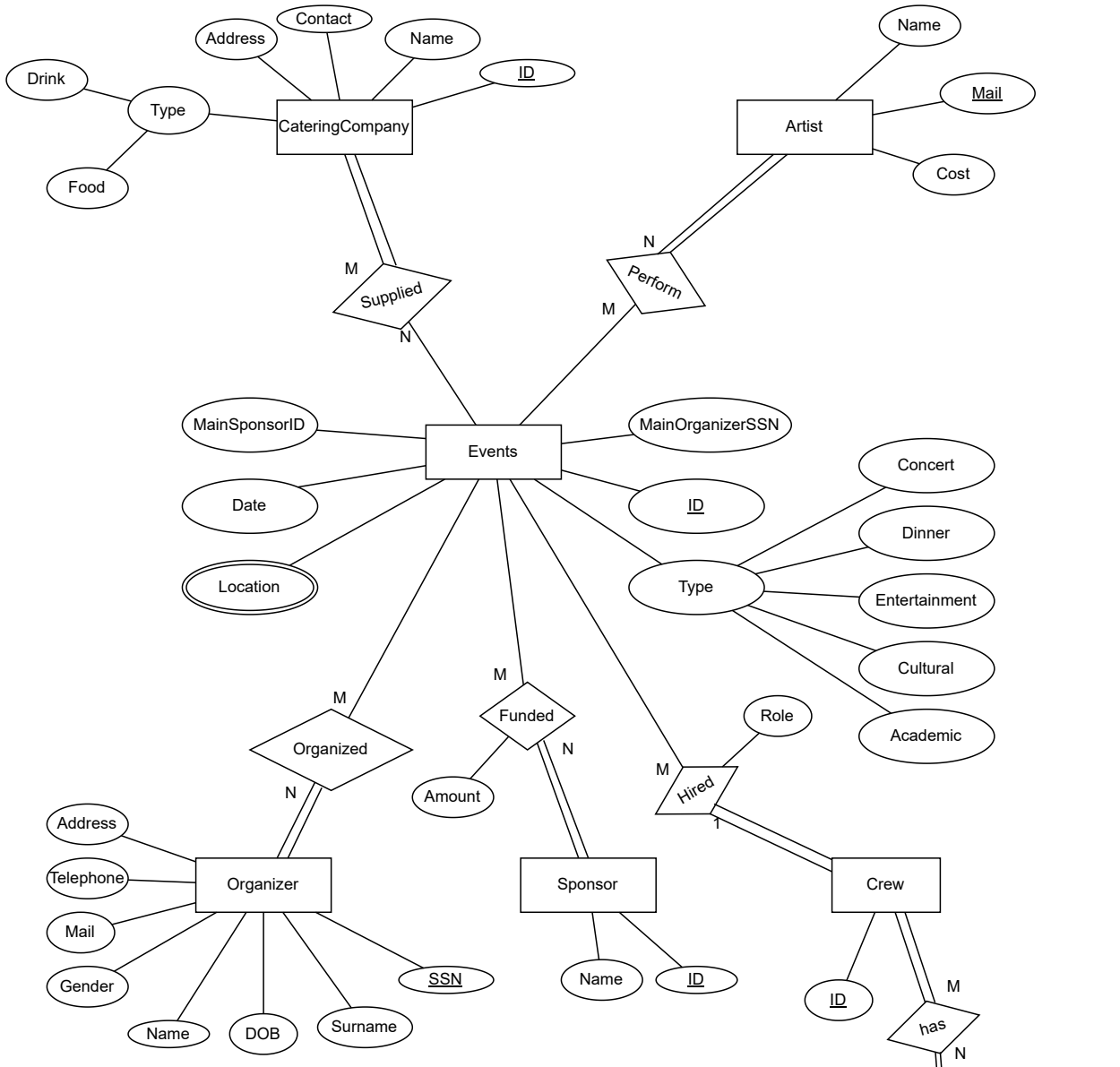
Member 1

Name: Mert Can
Surname: Bilgin
Student ID: 2453025

Member 2

Name: Hami
Surname: Karslı
Student ID: 2453322

Use Case 1 – MENTOR (Metu EveNT OrganiseR)



ASSUMPTIONS

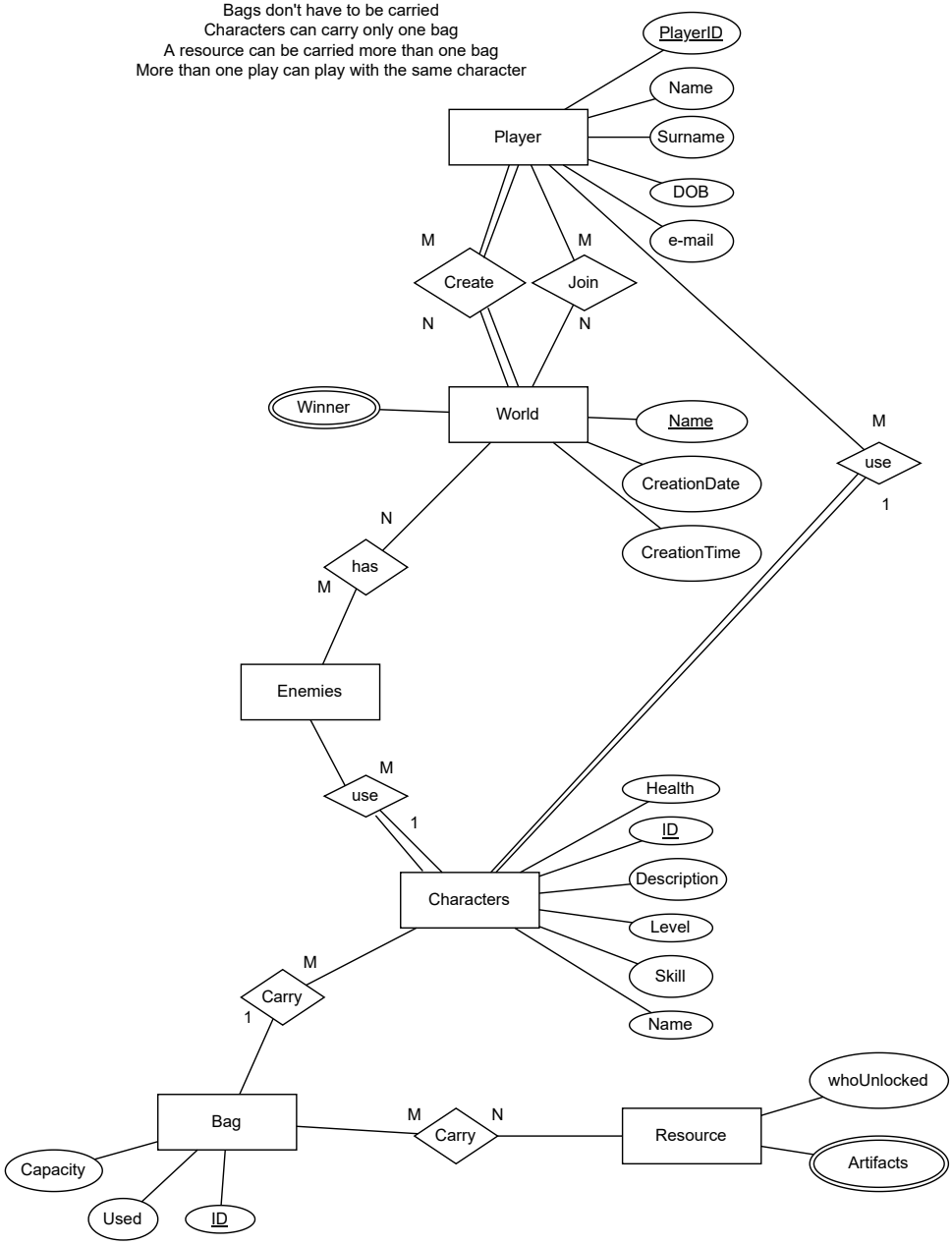
- SSN is a unique identifier
- ID is a unique identifier for Events
- Sponsor has a unique identifier named as ID
- Crew has a unique identifier named as ID
- Artists has unique mail address
- Catering Companies has unique IDs
- Each member has to work in a crew
- Artists don't have to perform in an event

First of all, we started to design database with Events. The attributes are drew according to the requirements. Because SSN is a unique identifier we assigned it as a super key. We also used unique identifiers for Sponsor, Crew and Catering Companies. For the Artists, we defined the e-mail address as unique because we assumed the address is not shareable. In addition to that, Artists don't have to perform in any events, so we used optional participation in Perform relation. Because a crew consists of members we assumed that a crew must have members.

Use Case 2 – Survive

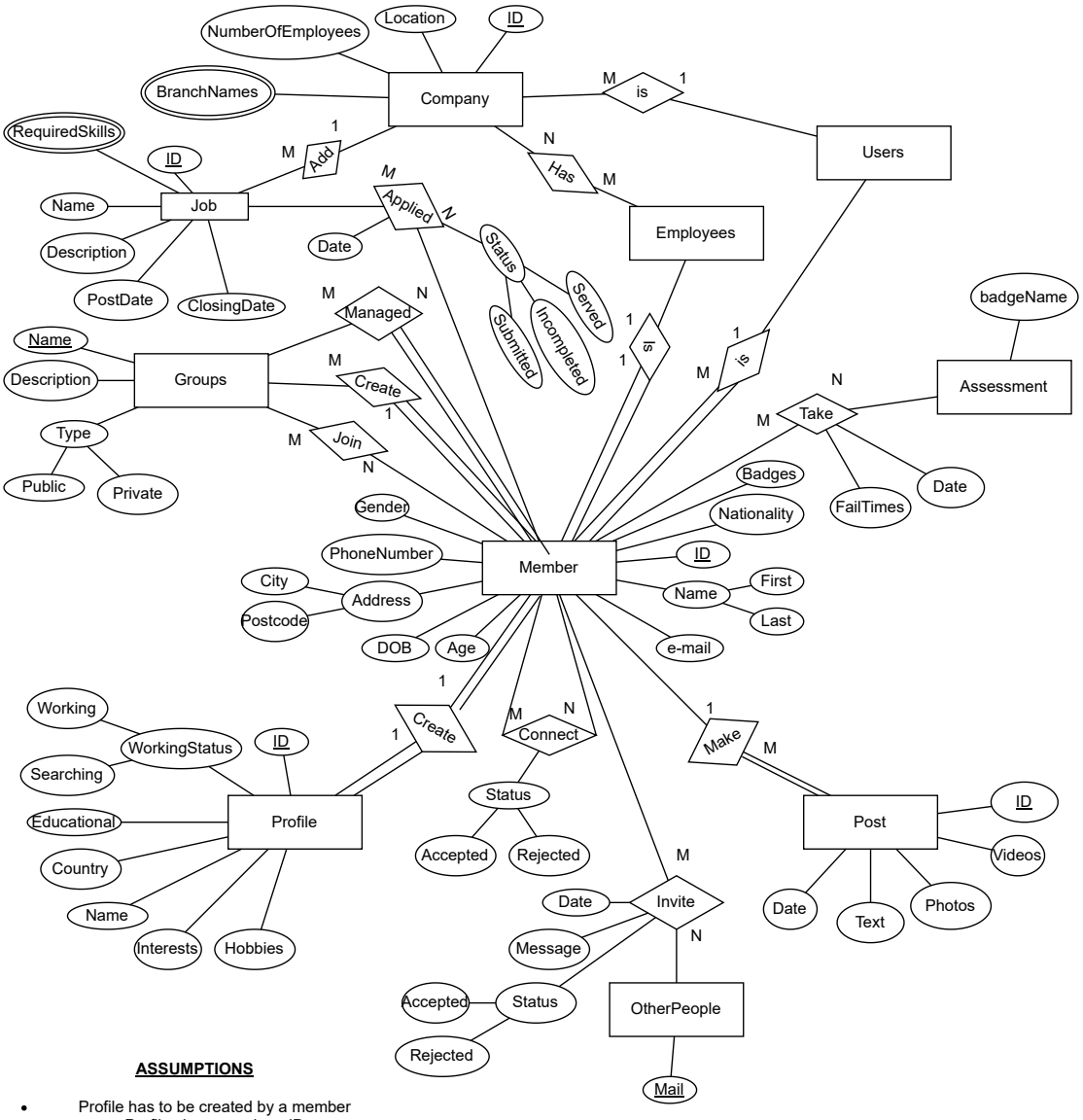
ASSUMPTIONS

- A player can play in more than one world
- Each player has a unique ID
- Bags have a unique ID
- Each enemy has a unique ID
- There might be more than one winner if the number of artifacts is equal in different players' bag
- The world needs to have players
- Characters have ID
- Bags don't have to be carried
- Characters can carry only one bag
- A resource can be carried more than one bag
- More than one play can play with the same character



By designing the database of Survive game, we started with Player details. Then, we added the attributes according to requirements. The hard part was connecting the relation between Characters and Enemies and Players. To decide the relation, we created different relations between world and player to know which player created the world. After that, we created the Characters entity under the Enemies and Player, so instead of writing Characters' details twice we used only one as common. Also, we assumed that Characters can carry only one bag, but the bag is shareable. In addition, we assumed players can use the same characters in the game, for example, there might be three Character X in the game. We also assumed that the end of the game, players might have equal number of artifacts, so there can be more than one winner.

Use Case 3 – FindJob

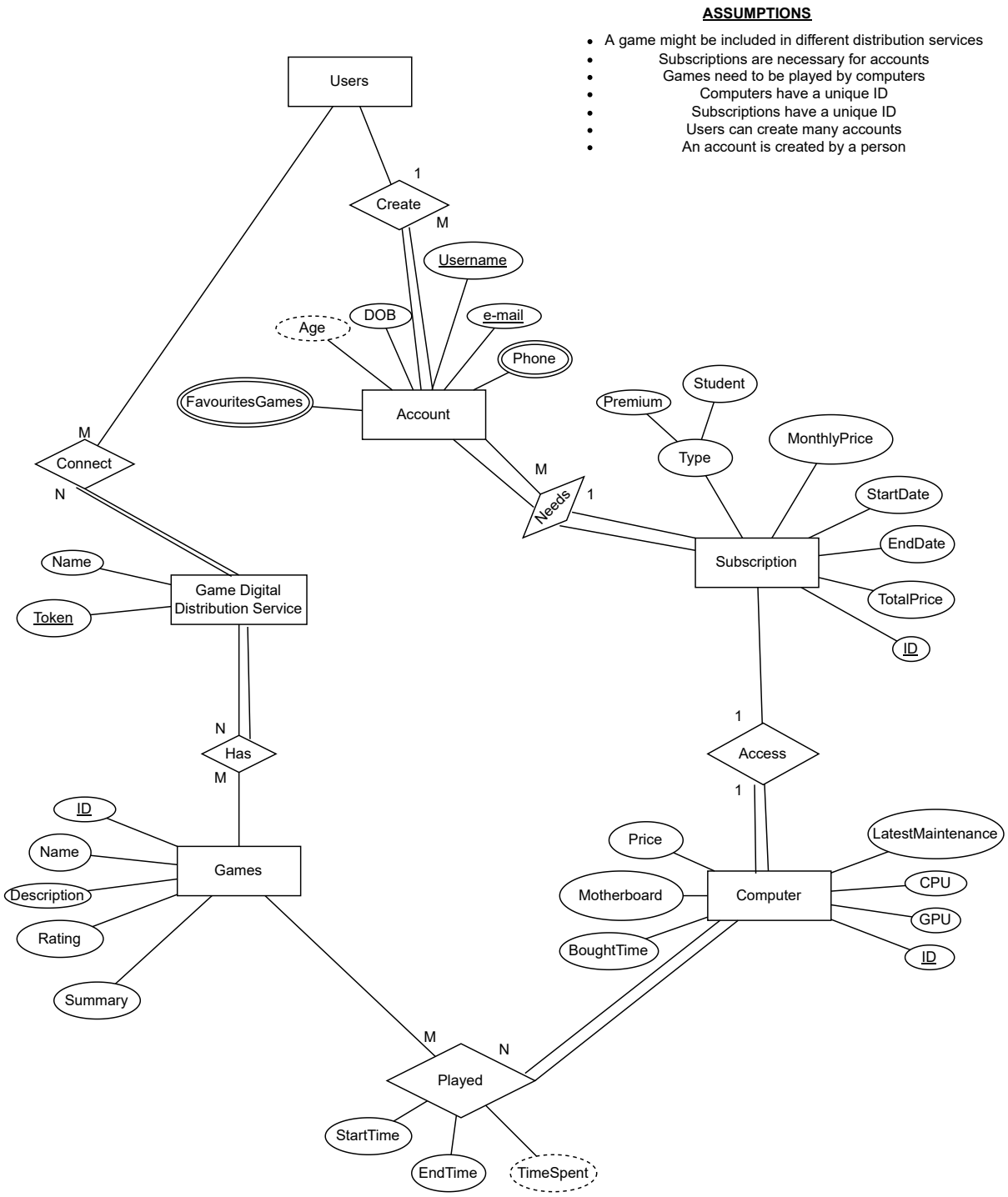


ASSUMPTIONS

- Profile has to be created by a member
- Profiles have a unique ID
- Members have a unique ID
- Videos, photos and text assumed as content
- Post ID is unique
- A group can only be created by a member
- A group needs to be managed
- Company has a unique ID
- Company or Member accounts used by one User

Use Case 3 is very similar to LinkedIn. We assumed Profile, Member and Post has unique identifiers. Also, the photos, videos and text are assumed as content, so we didn't use composite attributes for them. To create Groups, we assumed it needs to be managed by a member, and for the Connections we used recursive relation for that part.

Use Case 4 – NCCCloud



For NCCCloud, we used derived attributes for age and TimeSpent because these attributes can be derived from DOB and Start-EndTimes. We also assumed that subscriptions are necessary for accounts to use computers, and Computers, Subscriptions have unique IDs. In addition, we assumed that a game might be distributed in different services.