COM2058 Project-2

Group Members

RIBHI BISTAWI - 20290078 – ribhy2001@gmail.com
IBRAHIM ALSADI - 20290890 - ibraheemsadiii@gmail.com
Ibraheem R N Hamdan - 20290946 - ibraheemhamdan02@gmail.com
Ömer Faruk Uysal - 19290337 – uysal0113@gmail.com

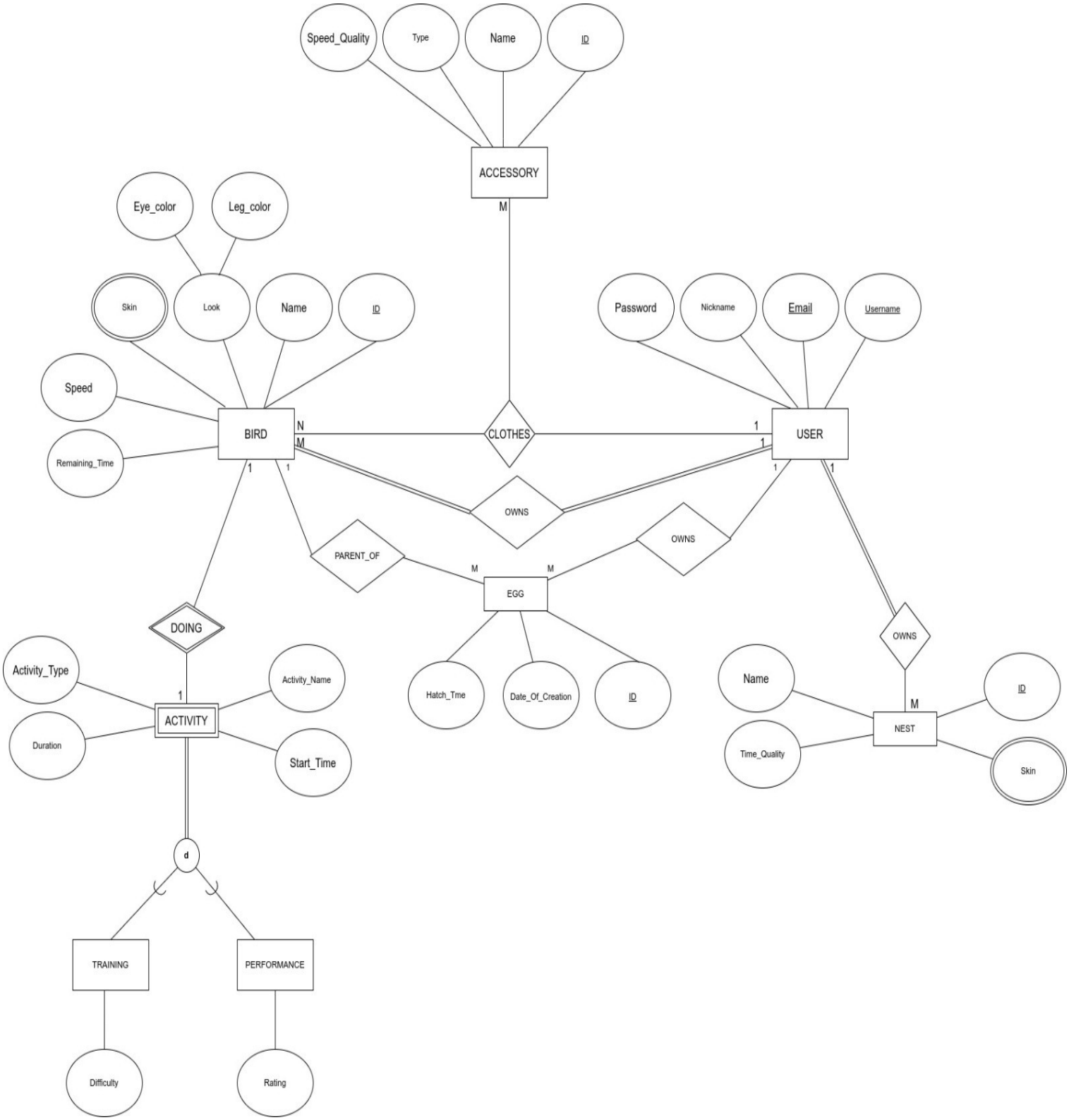
Task A: Defining Our Database

BIRD GAME DATABASE

Our project is not about storing data for a company or an establishment, it is about storing data for a game. The idea of the game is simple, each player owns a bunch of birds. The player can train them, send them to performances and breed them. Each bird has a speed quality stat. For example a speed quality of a bird can be ‘high’, ‘medium’ or ‘low’. The higher speed quality, the bird can perform another activity sooner. If a player wants new birds , they can breed their existing birds, which will result in them laying eggs. Eggs hatch after a while and produce new birds. Birds can be equipped with accessories, which improves their performance if they’re either training or doing a show. Training birds mostly improves their stats. And our database contains the following information;

►We will keep track of all USERs, including their email [Email], account password [Password], username (unique) [Username], nickname (not unique) [Nick-name].
►Each user will have a minimum of two BIRDs[OWNS], which each have an ID [ID], name [Name], a skin which changes the appearance of bird [Skin], a speed that defines how quickly the bird can perform an activity again [Speed], remaining time that keeps the time before the bird is able to do another activity [Remaining_time] and a multi-valued look attribute [Look] that will describe the look of the bird. Each bird is related to a player and a accessory with a relation [CLOTHES] that will be explained in more details later on.
►Birds can breed. Whenever they breed, the mother lays EGGs. We keep track of those eggs. We store their ID [ID], date of creation [Date_Of_Creation], their hatch time [Hatch_Time]. We also connect each egg to its parents with a relation [PARENT OF]. We also keep track of the user owning the egg with the relationship [OWNS_EGG].
►Each player owns several NESTs [OWNS_NEST]. We keep track of the nest’s ID [ID], name [Name]. We also store the quality of the nest. The time quality attribute [Time_Quality] will affect the time the eggs hatch in. And like birds, a nest can also have a skin that changes its appearance [Skin].
►Birds can do ACTIVITIES, which is an entity dependent on the existence of a bird. Each activity has a name [Activity_Name], a starting time [Start_Time], a duration [Duration], and a type (race, singing or stunts) [Activity_Type]. Activities divide into two types, performance [PERFORMANCE] or training [TRAINING]. Each performance has rating [Rating], which defines how good the performance was. Each training has difficulty [Difficulty].
►Users can clothe their birds with ACCESSORIES [CLOTHES]. Accessories can improve a bird’s activity speed. Each accessory has an ID [ID], a name [Name], a type [Type], and speed [Speed_Quality] which describe how well this accessory can improve the bird’s activity speed.

Task B: EER Diagram



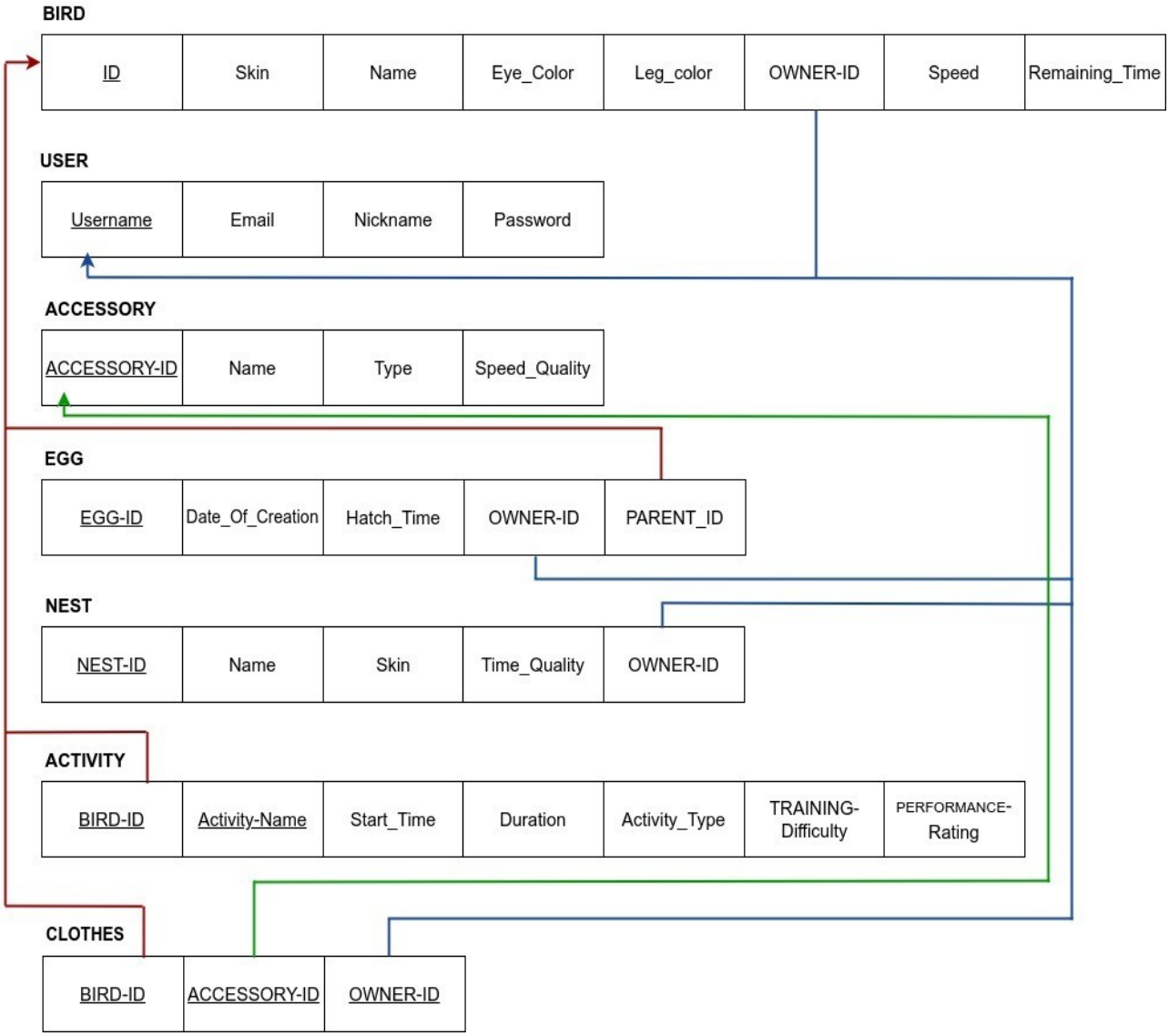
From the EER diagram we can see the followings;

► Each user can have M number of birds (at least 2 in our game scenario) and there is total participation between BIRD and USER from both sides. So each bird has an owner and also each user owns a bird.
► Also Email and username are both specified as a candidate to primary key. Later in the mapping stage one of them will be selected as a primary key of USER.
► Each user can clothe N number of birds with M accessories but there is no participation constraints in this ternary relation.
► A bird can have M number of eggs which is represented with the relation PARENT. Again there is no participation constrain in this relation so a bird may have or may not have any eggs. Very similar relation OWNS is also shown between USER and EGG. A user may have M number of eggs.
► BIRD has a multi-valued attribute Skin. That means a bird can have more than one values for Skin attribute.
► BIRD has a composite attribute which is Look. Look consists of to sub-attributes Eye-color and Leg-color.
► Each user has M number of nests and there is total participation for users in this relation.
► NEST has a multi-valued attribute Skin. That means a nest can have more than one values for Skin attribute.
► ACTIVITY is a weak entity that means does not have its own key and BIRD's key attribute is used to identify it.
► A bird can do just one activity at a time. DOING is a 1 to 1 identifying relation and does not have any participation constrains so that a bird may or may not do an activity.
► Activities are specialized into two subclasses with a total participation constrain. Therefore each activity must be in the TRAINING subclass or PERFORMANCE subclass. Also these subclasses are disjoint so they can not overlap.

About Other Constrains

► Each bird has at least 5 skins
► Eye color and leg color for look attribute of bird must be one of RGB
► Each user has at least 2 birds
► Each parent can have only one egg at a time

Task C: Relational Schema



Data types for the table BIRD's attributes

- ID INT PK
- Skin VARCHAR(15)
- Name VARCHAR(15)
- Eye-color VARCHAR(8)
- Leg-color VARCHAR(8)
- Owner-ID INT
- Speeed DOUBLE
- Remaining_Time TIME

Data types for the table USER's attributes

- Username INT PK
- Email VARCHAR(30)
- Nickname VARCHAR(15)
- Password VARCHAR(20)

Data types for the table ACCESSORY’s attributes
► ACCESSORY-ID INT PK
► Name VARCHAR(15)
► Type VARCHAR(10)
► Speed-Quality VARCHAR(10)

Data types for the table EGG’s attributes
► EGG-ID INT PK
► Date_Of_Creation DATE
► Hatch_Time TIME
► OWNER_ID INT
► PARENT_ID INT

Data types for the table NEST’s attributes
► NEST-ID INT
► Name VARCHAR(15)
► Skin VARCHAR(15)
► Time_Quality VARCHAR(10)
► OWNER-ID INT

Data types for the table ACTIVITY’s attributes
► BIRD-ID INT PK
► Activity-Name VARCHAR(15) PK
► Start_Time TIME
► Duration DOUBLE
► Activity-Type VARCHAR(10)
► TRAINING-Difficulty VARCHAR(10)
► PERFORMANCE-Rating DOUBLE

Data types for the table CLOTHES’s attributes
► BIRD-ID INT PK
► ACCESSORY-ID INT PK
► OWNER-ID INT PK