Zoo Database Management System

Micheal Hua
Student ID: 918729206
GitHub Username: michealhuaa

Checkpoint #	Date Submitted
Checkpoint #1	02/21/2023
Checkpoint #2	02/28/2023
Checkpoint #3	03/14/2023
Checkpoint #4	

Table of Content

I. Project Description

I will be building a zoo management database system that collects any animal that arrives in this zoo. I will be going with this sort of database system because to give these animals a healthy life and prosper, scientists and researchers need to know all there is to each of the creatures being displayed in these zoos. With the database system that I will be building, these analysts can let the managers as well as the employees know the special traits and features that each of the animals in the zoo has. From a management and visitor's point of view, this system will act as a point of sale, or POS. Managers will be able to track ticket sales for visitors and make sure the inventory the zoo has is in stock. If there needs to have a restock in a particular category, then the manager will be able to order more of that product to keep the zoo stock for any situation. Managers will also be able to view the database of the animals to maintain the zoo and keep it under control, adding any new animals that might be arriving at the zoo. On the visitor side, they will be able to view the prices for tickets as well as purchase tickets. The system will also have a calendar system where visitors will be able to schedule those tickets on a specific day. This might be in later implementations of the database, but visitors would be able to get an overview of a particular animal they find interesting. On the other hand, analysts such as official zoo researchers and scientists will be able to make edits to the animal database. That means they'll be able to edit any comments about the animal such as specific traits, enrichment, and food type. Regular zoo employees will be able to view the animal database as well as search for a particular species in order to find the details of an animal. This might be added later on, but employees would be able to group up a representation of what animals they're taking care of.

II. Use Cases

1. Getting a new animal; Actor: Nick (Head Director)
    a. Nick is the head director who is responsible for the operation of the zoo as well as any future plans for development. He just got a new mountain lion named Geno that was transferred from a different zoo. After getting all the details from the previous zoo, he will use the data he gained to add Geno to the zoo animals database.
2. Geno's checkup; Actors: Mariah (Veterinarian), John (Veterinarian Technician)
    a. After Nick gets Geno into the animal database, He and a few other workers get him to see Mariah, who is a veterinarian. Mariah does a detailed checkup on Geno to make sure he's been treated well. During the process, she has John, her assistant, take notes of anything she found as well as help her in any case. After the checkup, Mariah had John add any comments and health records into the database where Geno was located in the system.
3. Getting Geno situated; Actors: Nick (Head Director), Ben (Curator of Exhibits), Abel (Keeper)
    a. Once Geno was checked up by Mariah and the vet crew, Nick and Ben built an exhibit for Geno to be located within the zoo. After the success of the exhibit design and creation, Abel, who is a keeper, was assigned to take care of Geno. Before Geno was placed, Abel used the database system to view the contents of Geno. She viewed things like food choices, enrichment, and specific toys that Geno was interested in. Once she gained all the information she needed, she

prepared the exhibit for Geno with the things that are included in the database system. After that, Geno was successfully placed inside his exhibit.

4. Checking on stock and back stocking; Actors: Peter (Finance Director), Nick (Head Director)
    a. In order to maintain the inventory of the zoo, Nick checked in with the curators on what needed to be stocked. Once Nick had a list of things to be ordered, he met up with Peter, the finance director, and made sure that financially, the zoo was alright. Once everything was checkmarked, Nick then used the POS part of the database system to order the things that were in that list.
5. Visiting the zoo; Actors: Leah (customer/visitor), David (Visitor Service Manager)
    a. Leah was interested in taking the opportunity on her off day to spend some quality time with her family at the zoo. Using the database system, she was able to view the pricing for the tickets, but she had questions. Therefore, she called the Visitor Service and David took the call. David answered all of the questions Leah had and she was able to buy the tickets with David using the database system to do so.

III. Existing Software Tools

One Software tool that zoos use is ZIMS. The Zoological Information Management Software, or ZIMS, is a software used by over 1000 zoos that helps zoos achieve the best management practice in zoos. ZIMS also has features such as breeding and aquatics that really benefits every zoo using it. This tool would benefit from my database system because it would also give more information about the animals from zoos that are using my system. It will give them much more research opportunities for their own software.

Another software tool that would benefit from my database system is the Association of Zoos and Aquatics. This will also give this tool more details on the behavior of animals as well as research on the unique aspects of the animal.

IV. Database Requirements

IV.I Functional Database Requirements

1. User
    1.1. A user shall make zero or one account
    1.2. A user can request for many ticket prices
    1.3. A user can buy many tickets
    1.4. A user can contact zero or many Visitor Service centers
    1.5. A user can be a visitor
    1.6. A user shall be answered by a Visitor Service worker
2. Head Director
    2.1. A head director shall work for one zoo
    2.2. A head director shall add zero or many animals
    2.3. A head director shall blueprint many exhibits
    2.4. A head director shall restock zero or many inventories
    2.5. A head director shall operate a zoo
    2.6. A head director shall maintain a zoo
3. Veterinarian
    3.1. A veterinarian shall checkup on zero or many animals
    3.2. A veterinarian shall make notes of zero or many animals' health?
    3.3. A veterinarian shall make zero or many comments?
    3.4. A veterinarian shall add zero or many health records?
4. Exhibit Curator
    4.1. An exhibit curator shall design many exhibits
    4.2. An exhibit curator shall assemble many exhibits
5. Keeper
    5.1. A keeper shall take care of many animals
    5.2. A keeper can view many inventories
    5.3. Keepers can talk to many visitors
    5.4. A keeper shall put out many food choices
    5.5. A keeper shall put out many enrichments
    5.6. A keeper shall put out zero or many toys
    5.7. A keeper shall maintain many exhibits
6. Finance Director
    6.1. A finance director can see many inventories
    6.2. A finance director shall manage many financial decisions
    6.3. A finance director can view many sales

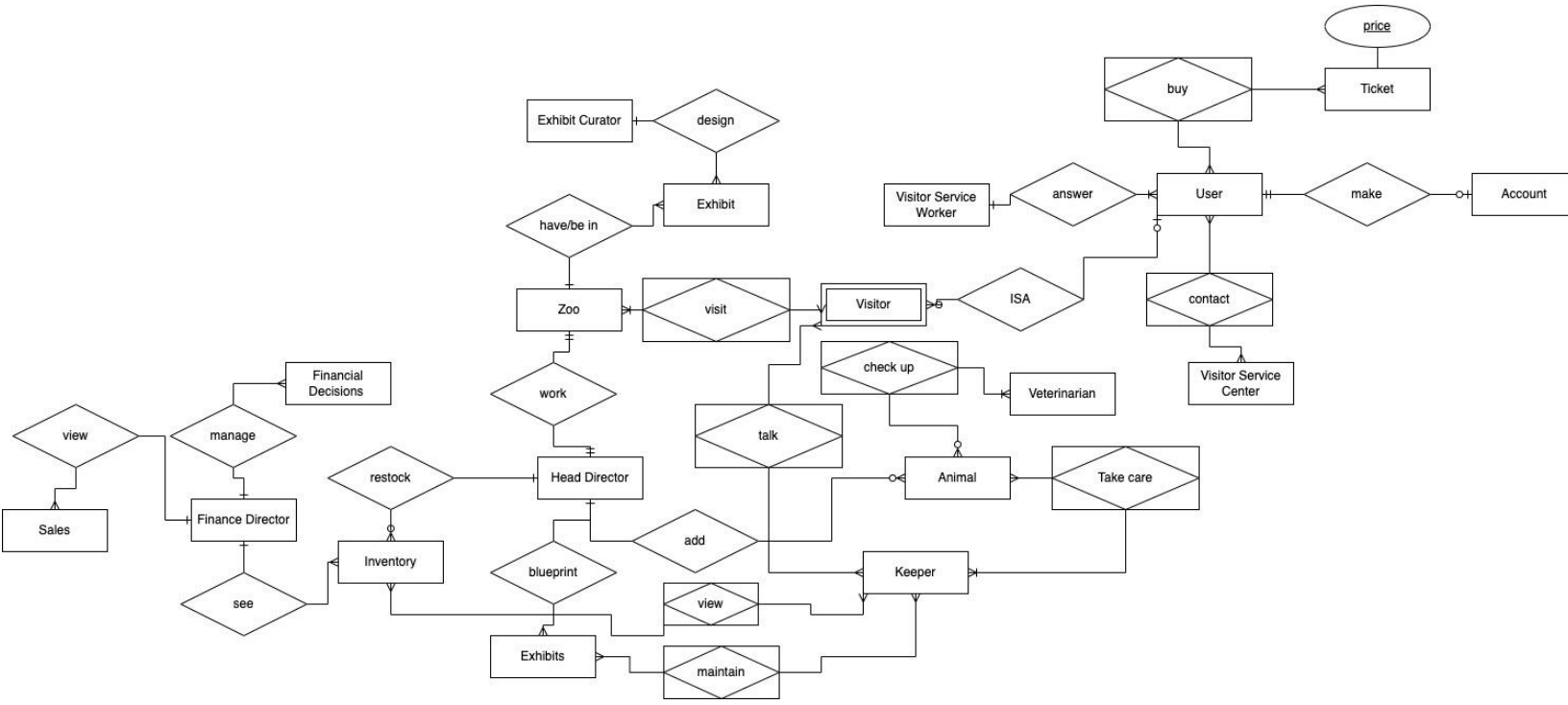
7. Visitor
    7.1. Visitors shall visit at least one zoo
    7.2. A visitor can be a user
    7.3. A visitor can talk to many keepers
8. Account
    8.1. An account shall be made by only one user
9. Ticket price
    9.1. Ticket prices can be requested by many users
10. Ticket
    10.1. Tickets can be bought by many users
11. Visitor Service centers
    11.1. A Visitor Service center can be contacted my many users
12. Visitor Service worker
    12.1. A Visitor Service worker shall answer at least one user
13. Zoo
    13.1. A zoo shall have only one head director
    13.2. A zoo shall have many visitors
    13.3. A zoo shall be operated by a head director
    13.4. A zoo shall be maintained by a head director
    13.5. A zoo shall have many exhibits
14. Animal
    14.1. An animal shall be added by a head director
    14.2. An animal shall be checked up by at least one veterinarian
    14.3. An animal shall be taken care of by at least one keeper
    14.4. An animal can have many food choices
    14.5. An animal can have multiple enrichments
    14.6. An animal can have many toys
15. Animal’s health
    15.1. An animal’s health shall be noted by many veterinarians
16. Exhibit
    16.1. An exhibit shall be planned out by a head director
    16.2. An exhibit shall be designed by an exhibit curator
    16.3. An exhibit shall be assembled by at least one exhibit curator
    16.4. An exhibit shall be maintained by many keepers
    16.5. An exhibit shall be in one zoo.
17. Inventory
    17.1. An inventory can be viewed by many workers
    17.2. An inventory shall be restocked by a head director
18. Financial decisions
    18.1. Financial decisions shall be managed by a financial director

19. Sales
    19.1. Sales can be viewed by directors
20. Food Choice
    20.1. A food choice can be put out by a keeper
    20.2. A food choice can be used by many animals
21. Enrichment
    21.1. An enrichment can be used for multiple animals
    21.2. An enrichment shall be placed out by a keeper
22. Toys
    22.1. A toy can be given out to many animals
    22.2. A toy shall be put out by a keeper
23. Health record
    23.1. A health record shall be added by multiple veterinarians

IV.II Non-Functional Requirements

1. Security
    1.1. Passwords need to be encrypted
    1.2. Account will be locked after 5 attempts
    1.3. Backup file shall be reset every day at 11:59pm
2. Legal
    2.1. Keep regular checks with PETA
3. Performance
    3.1. Query response must return data in 1ms or more
4. Capability
    4.1. Each table shall assign no more than 3mb of memory
5. Backup
    5.1. Database shall be backed up once every day
6. Environmental
    6.1. Database system shall be kept at cool temperatures
7. Usability
    7.1. Interface shall be creative, but easy to use

V. Entity Relationship Diagram (ERD)



VI. Entity and Attributes Description

1. User (Strong)
    a. user_id: key, numeric
    b. name: composite, alphanumeric
    c. age: numeric
2. Head Director (Strong)
    a. zoo: key, alphanumeric
    b. director_id: key, numeric
    c. age: numeric
    d. email: alphanumeric
    e. name: composite, alphanumeric
3. Veterinarian (Strong)
    a. vet_id: key, numeric
    b. email: alphanumeric
4. Exhibit Curator (Strong)
    a. curator_id: key, numeric
    b. curator_name: alphanumeric
    c. age: numeric
5. Keeper (Strong)
    a. keeper_id: key, numeric
    b. name: alphanumeric
    c. age: numeric
6. Finance Director (Strong)
    a. director_id: key, numeric
    b. name: alphanumeric
    c. age: numeric
    d. email: alphanumeric
7. Visitor (Weak)
    a. user: key, numeric
    b. zoo: key, numeric

    c. visitor_id: key, numeric
    d. age: numeric
8. Account (Weak)
    a. user: key, numeric
    b. account_id: key, numeric
    c. email: alphanumeric
9. Ticket (Strong)
    a. price: numeric
    b. ticket_id: key, numeric
10. Visitor Service Center (Strong)
    a. center_id: key, numeric
11. Visitor Service worker (Strong)
    a. Visitor_Service_Center: key, numeric
    b. worker_id: key, numeric
    c. name: alphanumeric
    d. age: numeric
12. Zoo (Strong)
    a. zoo_id: numeric
    b. zoo_name: alphanumeric
13. Animal (Strong)
    a. animal_id: key, numeric
    b. name: alphanumeric
    c. species: alphanumeric
    d. health: alphanumeric
14. Exhibit (Strong)
    a. animal: key, numeric
    b. Exhibit_id: key, numeric
    c. zoo: key, numeric
    d. exhibit_curator: key, numeric
15. Inventory (Strong)
    a. inventory _id: key, numeric

   b. name: alphanumeric
   c. quantity: numeric

16. Sales (Strong)
   a. sales_id: key, numeric
   b. name: alphanumeric
   c. quantity: numeric
   d. quantity_sold: numeric
   e. total_money_made: numeric
17. Food choice (Strong)
   a. food_id: key, numeric
   b. name: alphanumeric
   c. quantity: numeric
18. Enrichment (strong)
   a. enrichment_id: key, numeric
   b. name: alphanumeric
   c. quantity: numeric
19. Toys (strong)
   a. toy_id: key, numeric
   b. name: alphanumeric
   c. quantity: numeric