



# CS353 GROUP 32

## PROJECT FINAL REPORT

ZOO DATABASE MANAGEMENT SYSTEM

### Group Members

Ramazan Melih DİKSU - 21802361

Aleyna SÜTBAŞ - 21803174

Berk TAKIT - 21803147

Sarp TEKİN - 21600813

<b>1.Description of the System</b>	<b>4</b>
<b>2.Final ER Model</b>	<b>5</b>
<b>3.RELATIONAL SCHEMAS</b>	<b>6</b>
3.1 User	6
3.2 Visitor	6
3.3 Employee	6
3.4 Keeper	6
3.5 Veterinarian	7
3.6 Coordinator	7
3.7 Comment	7
3.8 Membership	7
3.9 Comments	7
3.10 Has	8
3.11 Event	8
3.12 Group Tour	8
3.13 Educational Program	8
3.14 Conservation Organization	9
3.13 Attends	9
3.14 Creates Event	9
3.15 Donates	9
3.16 Responds to	10
3.17 Invites	10
3.18 Complaint Form	10
3.19 Schedules	10
3.20 Regularizes	11
3.21 Animal	11
3.22 Food	11
3.23 Requests	11
3.24 Belongs_to	12
3.25 Cage	12
3.26 Area	12
3.27 Shop	12
3.28 Item	13
3.29 Is_In_C	13
3.30 Is_In_S	13
3.31 Sells	13
3.32 Buys	13
3.33 Assigns	14
3.34 Creates_Complaint_Form	14

<b>4. Implementation Details</b>	<b>15</b>
<b>5. Advanced Database Features</b>	<b>16</b>
5.1 Reports	16
5.2. Views	17
5.3 Triggers	17
5.4 Constraints	17
5.5 Stored Procedures	17
5.6 Secondary Indices	17
<b>6. User's Manual</b>	<b>18</b>
6.1. Introduction	18
6.2. Login	18
6.3. Register	19
6.4. Coordinator - Home Page	20
6.5. Coordinator - Events	20
6.6. Coordinator - Create New Event	21
6.6.1. No Event Type Chosen	21
6.6.2. Group Tour Form	22
6.6.3. Educational Program Form	22
6.6.4. Conservation Organisation Form	23
6.7. Coordinator - View & Assign Cages	24
6.7.1. Unassigned Cages	24
6.7.2. Cages Assigned By You	25
6.8. Coordinator - Assign Cage	25
6.9. Keeper - Home Page	26
6.11. Keeper - Animals	28
6.12 Keeper - View Trainings	29
6.13 Visitor - Home Page	30
6.14 Visitor - Shops	30
6.15 Visitor - Items	31
6.15.1. Gift Shop Example	31
6.15.2. Toy Shop Example	31
6.16 Visitor - Membership	32
6.17 Visitor - Group Tours	32
6.18 Visitor - Comment	33
6.19 Visitor - Conservation Organizations	33

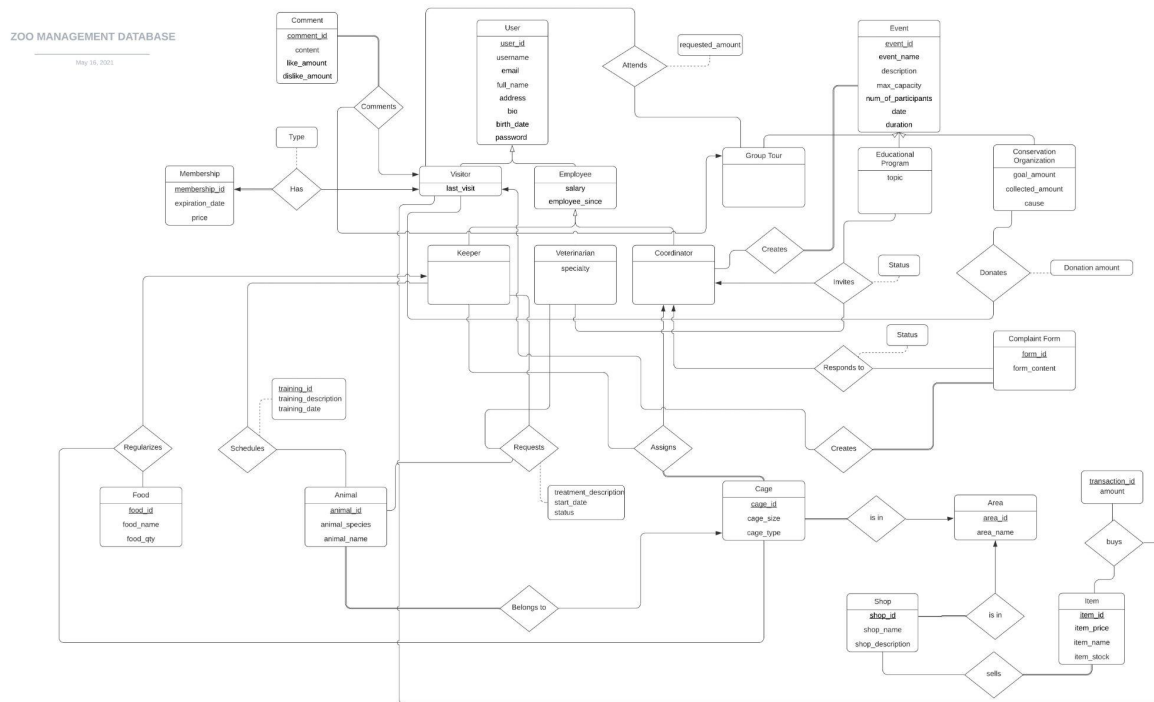
# 1. Description of the System

This project is a web-based database management application that can be used by both the employees and visitors of the zoo. The aim of the project is to increase the efficiency of communication and planning between employees and visitors.

There are four different types of users which all have their unique features. Employees are divided into three sub groups as keeper, veterinarian and coordinator which have unique features in the web site that they can use. Coordinators can create events like group tours, educational programs or a conservation organization. Moreover, they can invite veterinarians to educational programs and assign keepers to the cages and respond to the complaint forms. Keepers can request treatment for an animal by the veterinarians and schedule training for animals. Veterinarians can accept or reject invitations to educational programs that were sent by the coordinators and treatment requests that were sent by the keepers. Visitors can attend group tours by paying the price of the tour that they desire and make donations to the conservation organizations. Furthermore, they can write complaint forms and comment on group tours as well.

Since there are many different users and many different request combinations, a well structured database is needed for this project. Thus, our database allows the data to be organised and retrieved more efficiently than a simple file system.

## 2.Final ER Model



URL: [https://drive.google.com/file/d/1Sc0Sg5USFwTm2BUSyQR5wzujrT5dTnMJ/view?usp=s\\_haring](https://drive.google.com/file/d/1Sc0Sg5USFwTm2BUSyQR5wzujrT5dTnMJ/view?usp=s_haring)

## 3.RELATIONAL SCHEMAS

### 3.1 User

- Relational Model:  
User(user\_id, username, email, full\_name, address, bio, birth\_date,password)
- Foreign Keys:  
-

### 3.2 Visitor

- Relational Model:  
Visitor(visitor\_id, last\_visit)
- Foreign Keys  
**FOREIGN KEY** (visitor\_id) **REFERENCES** User(user\_id)

### 3.3 Employee

- Relational Model:  
Employee(employee\_id, salary, employee\_since)
- Foreign Keys:  
**FOREIGN KEY** (employee\_id) **REFERENCES** User(user\_id)

### 3.4 Keeper

- Relational Model:  
Keeper(keeper\_id)
- Foreign Keys:  
**FOREIGN KEY** (keeper\_id) **REFERENCES** Employee(employee\_id)

### 3.5 Veterinarian

- Relational Model:  
Veterinarian(veterinarian\_id, speciality)
- Foreign Keys:  
**FOREIGN KEY** (veterinarian\_id) **REFERENCES** Employee(employee\_id)

### 3.6 Coordinator

- Relational Model:  
Coordinator(coordinator\_id)
- Foreign Keys:  
**FOREIGN KEY** (coordinator\_id) **REFERENCES** Employee(employee\_id)

### 3.7 Comment

- Relational Model:  
Comment(comment\_id, content, like\_amount, dislike\_amount)
- Foreign Keys:  
-

### 3.8 Membership

- Relational Model:  
Membership(membership\_id, expiration\_date, price)
- Foreign Keys:  
-

### 3.9 Comments

- Relational Model:  
Comments(comment\_id, user\_id, groupTour\_id)
- Foreign Keys:  
**FOREIGN KEY** (comment\_id) **REFERENCES** Comment(comment\_id)  
**FOREIGN KEY** (user\_id) **REFERENCES** Visitor(visitor\_id)  
**FOREIGN KEY** (groupTour\_id) **REFERENCES** Group\_Tour(group\_tour\_id)

### 3.10 Has

- Relational Model:  
Has(membership\_id, user\_id, type)
- Foreign Keys:  
**FOREIGN KEY** (membership\_id) **REFERENCES** Membership(membership\_id),  
**FOREIGN KEY** (user\_id) **REFERENCES** Visitor(visitor\_id)

### 3.11 Event

- Relational Model:  
Event(event\_id, description, max\_capacity, num\_of\_participants, date, duration)
- Foreign Keys:  
-

### 3.12 Group Tour

- Relational Model:  
Group\_Tour(group\_tour\_id)
- Foreign Keys:  
**FOREIGN KEY** (group\_tour\_id) **REFERENCES** Event(event\_id)

### 3.13 Educational Program

- Relational Model:  
Educational\_Program(edu\_prog\_id, topic)
- Foreign Keys:  
**FOREIGN KEY** (edu\_prog\_id) **REFERENCES** Event(event\_id)



### 3.14 Conservation Organization

- Relational Model:  
Educational\_Program(con\_org\_id, goal\_amount, collected\_amount, cause)
- Foreign Keys:  
**FOREIGN KEY** (con\_org\_id) **REFERENCES** Event(event\_id)

### 3.13 Attends

- Relational Model:  
Attends(visitor\_id, group\_tour\_id, requested\_amount)
- Foreign Keys:  
**FOREIGN KEY** (visitor\_id) **REFERENCES** Visitor(visitor\_id)  
**FOREIGN KEY** (group\_tour\_id) **REFERENCES** Group\_Tour(group\_tour\_id)

### 3.14 Creates Event

- Relational Model:  
Creates(coordinator\_id, event\_id)
- Foreign Keys:  
**FOREIGN KEY** (coordinator\_id) **REFERENCES** Coordinator(coordinator\_id)  
**FOREIGN KEY** (event\_id) **REFERENCES** Event(event\_id)

### 3.15 Donates

- Relational Model:  
Donates(visitor\_id, con\_org\_id, donation\_amount)
- Foreign Keys:  
**FOREIGN KEY** (visitor\_id) **REFERENCES** Visitor(visitor\_id)  
**FOREIGN KEY** (con\_org\_id) **REFERENCES** Conversation\_Organization(con\_org\_id)

### 3.16 Responds to

- Relational Model:  
Responds\_to(form\_id, coordinator\_id, status)
- Foreign Keys:  
**FOREIGN KEY** (form\_id) **REFERENCES** Complaint\_Form(form\_id)  
**FOREIGN KEY** (coordinator\_id) **REFERENCES** Coordinator(coordinator\_id)

### 3.17 Invites

- Relational Model:  
Responds\_to(veterinarian\_id, edu\_prog\_id, coordinator\_id, status)
- Foreign Keys:  
**FOREIGN KEY** (coordinator\_id ) **REFERENCES** Coordinator(coordinator\_id)  
**FOREIGN KEY** (veterinarian\_id) **REFERENCES** Veterinarian(veterinarian\_id)  
**FOREIGN KEY** (edu\_prog\_id ) **REFERENCES** Educational\_Program(edu\_prog\_id)

### 3.18 Complaint Form

- Relational Model:  
Complaint\_Form(form\_id, form\_content)
- Foreign Keys:  
-

### 3.19 Schedules

- Relational Model:  
Schedules(training\_id, keeper\_id, animal\_id, training\_description, training\_date)
- Foreign Keys:  
**FOREIGN KEY** (keeper\_id) **REFERENCES** Keeper(keeper\_id)  
**FOREIGN KEY** (animal\_id) **REFERENCES** Animal(animal\_id)

### 3.20 Regularizes

- Relational Model:  
Regularizes(food\_id, cage\_id, keeper\_id)
- Foreign Keys:  
**FOREIGN KEY** (food\_id) **REFERENCES** Food(food\_id)  
**FOREIGN KEY** (keeper\_id) **REFERENCES** Keeper(keeper\_id)  
**FOREIGN KEY** (cage\_id) **REFERENCES** Cage(cage\_id)

### 3.21 Animal

- Relational Model:  
Animal(animal\_id, animal\_species, animal\_name)
- Foreign Keys:  
-

### 3.22 Food

- Relational Model:  
Food(food\_id, food\_name, food\_qty)
- Foreign Keys:  
-

### 3.23 Requests

- Relational Model:  
Regularizes(keeper\_id, veterinarian\_id, animal\_id, treatment\_description, start\_date, status)
- Foreign Keys:  
**FOREIGN KEY** (keeper\_id) **REFERENCES** Keeper(keeper\_id)  
**FOREIGN KEY** (veterinarian\_id) **REFERENCES** Veterinarian(veterinarian\_id)  
**FOREIGN KEY** (animal\_id) **REFERENCES** Animal(animal\_id)

### 3.24 Belongs\_to

- Relational Model:  
Belongs\_to(animal\_id, cage\_id)
- Foreign Keys:  
**FOREIGN KEY** (animal\_id) **REFERENCES** Animal(animal\_id)  
**FOREIGN KEY** (cage\_id) **REFERENCES** Cage(cage\_id)

### 3.25 Cage

- Relational Model:  
Cage(cage\_id, cage\_size, cage\_type)
- Foreign Keys:  
-

### 3.26 Area

- Relational Model:  
Area(area\_id, area\_name)
- Foreign Keys:  
-

### 3.27 Shop

- Relational Model:  
Area(shop\_id, shop\_name, shop\_description)
- Foreign Keys:  
-

### 3.28 Item

- Relational Model:  
Area(item\_id, item\_name, item\_stock)
- Foreign Keys:  
-

### 3.29 Is\_In\_C

- Relational Model:  
Is\_In(cage\_id, area\_id)
- Foreign Keys:  
**FOREIGN KEY** (area\_id) **REFERENCES** Area(area\_id)  
**FOREIGN KEY** (cage\_id) **REFERENCES** Cage(cage\_id)

### 3.30 Is\_In\_S

- Relational Model:  
Is\_In(shop\_id, area\_id)
- Foreign Keys:  
**FOREIGN KEY** (area\_id) **REFERENCES** Area(area\_id)  
**FOREIGN KEY** (shop\_id) **REFERENCES** Shop(shop\_id)

### 3.31 Sells

- Relational Model:  
Belongs\_to(shop\_id, item\_id)
- Foreign Keys:  
**FOREIGN KEY** (shop\_id) **REFERENCES** Shop(shop\_id)  
**FOREIGN KEY** (item\_id) **REFERENCES** Item(item\_id)

### 3.32 Buys

- Relational Model:  
Buys(transaction\_id, item\_id, user\_id, amount)
- Foreign Keys:  
**FOREIGN KEY** (item\_id) **REFERENCES** Item(item\_id)  
**FOREIGN KEY** (user\_id) **REFERENCES** User(user\_id)

### 3.33 Assigns

- Relational Model:  
Assigns(keeper\_id,cage\_id, coordinator\_id)
- Foreign Keys:  
**FOREIGN KEY** (coordinator\_id) **REFERENCES** Coordinator(coordinator\_id)  
**FOREIGN KEY** (keeper\_id) **REFERENCES** Keeper(keeper\_id)  
**FOREIGN KEY** (cage\_id) **REFERENCES** Cage(cage\_id)

### 3.34 Creates\_Complaint\_Form

- Relational Model:  
Create\_Complaint\_Form(form\_id,visitor\_id)
- Foreign Keys:  
**FOREIGN KEY** (form\_id) **REFERENCES** Complaint\_Form(form\_id)  
**FOREIGN KEY** (visitor\_id) **REFERENCES** Visitor(visitor\_id)

## 4. Implementation Details

The implementation details of our system are as follows:

**Environment:** For our working environment we used the XAMPP Control Panel for the Apache server and the MySQL database. We also use PHPMyAdmin, which is included in the XAMPP Control Panel, for testing SQL queries. As IDE, we mostly used VSCode.

**Framework:** We did not use any JavaScript frameworks. We used a little bit of the Bootstrap CSS framework when designing our GUI.

**Languages:** We used the following languages:

- HTML and CSS (with the Bootstrap framework) was used for designing the GUI.
- MySQL was used to build the database.
- Java with JDBC was used to initialize the database.
- PHP was used to achieve connectivity with the database for the end user.
- A very miniscule amount of JavaScript was used to prevent resubmissions of forms on refresh and provide some error messages.

During the implementation process we faced several problems with both the GUI design and database and query design parts. The way we tackled the GUI problems, which were mostly related to CSS, was to do a lot of research on relevant platforms such as StackOverflow and look at example code snippets on CodePen. For our database design problems we experimented with modifying the database using PHPMyAdmin's SQL console until we got the results we desired.

Examples of the problems that took the most time to solve and how we solved them are:

- To be able to direct the users to the correct home pages based on their type (coordinator, visitor or keeper) we first had to be able to tell a user's type given their user id. Rather than use a long and complex SQL query to check each of the corresponding tables for the three types of users, we created a stored procedure that could be called with the user id as a parameter and would return the user type.
- Create Event page had a complex process since we wanted to implement a well designed interface. Also, our event types had different attributes and we wanted to fit the create functionality to a single page instead of creating 3 separate pages for each event type. Due to its complexity; CSS, Javascript and PHP clashed with each other constantly and we faced a lot of errors. We were able to solve this problem with trial and error and having a research on how to integrate the languages together.

The work done by each member in the implementation is as follows (everyone did both the GUI and PHP implementations of their parts and contributed to database implementation):

- Aleyna Sütbaşı: Shop, item and membership.
- Berk Takıt: Donations, coordinator reports, logout and redirect functionalities.
- Ramazan Melih Diksu: Coordinator and Keeper functionalities, visitor comments.
- Sarp Tekin: Login and Register.

## 5. Advanced Database Features

### 5.1 Reports

```
SELECT event_id, event_name, event.date, COUNT(comment_id) AS num_comments
FROM group_tour LEFT JOIN comments ON group_tour.group_tour_id =
comments.groupTour_id JOIN event ON event.event_id =
group_tour.group_tour_id
WHERE DATEDIFF(CURDATE(), event.date) BETWEEN 0 AND $filter_days
GROUP BY event_id
```

- This report enables coordinators to see the comments that have been made to the group tours in a given period. The period is specified with the \$filter\_days variable.

```
SELECT shop.shop_id, shop.shop_name, SUM(amount) AS items_sold
FROM shop LEFT JOIN sells ON shop.shop_id=sells.shop_id JOIN item ON
sells.item_id = item.item_id JOIN buys ON item.item_id = buys.item_id
WHERE DATEDIFF(CURDATE(), buys.purchase_date) BETWEEN 0 AND $filter_days
GROUP BY shop.shop_id
```

- This report enables coordinators to see the items that have been purchased from the shops by the visitors in a given period. The period is specified with the \$filter\_days variable.

Zoo

HomeEventsView & Assign CagesView ReportsLogout

Reports

Comments Made for Group Tours

ID	Name	Date	Comment Count
9	Group Tour 1	2021-03-12	2
10	Group Tour 2	2021-04-10	2
11	Group Tour 3	2021-04-22	1

Items Sold by Shop

ID	Name	Total Items Sold
1	Gift_Shop	1
2	Toy_Shop	11

All

Filter Reports

- Coordinators can access the reports through the View Reports page.
- To specify the time period for the comment and item reports, the coordinators can select the “last year”, “last three months” and “last month”. Also, they can choose “All” to see all the comments and sold items without period filtering.



## 5.2. Views

```
CREATE VIEW shops_view (name, description, area_name) AS
SELECT shop_name, shop_description, area_name
FROM Shop NATURAL JOIN Is_In_S NATURAL JOIN Area
```

- This view enables users to access the shop list in the shops page. Since the users will not change any data about the shop list itself, creating a view instead of using sql query was the better choice for this page.

## 5.3 Triggers

- We have created a trigger for Has and Membership tables which makes sure that when a Membership is deleted, the related tuple on the Has table is also deleted automatically.
- We have created a trigger for Attends and Event tables which increase the number of participants attribute of the Event table by one every time a new tuple is inserted to the Attends table.

## 5.4 Constraints

- We have created constraints on all tables that include foreign keys so that they can be checked after each deletion, insertion and update.

## 5.5 Stored Procedures

- We have created a stored procedure for finding the type of a user by searching the user id in the tables of different types such as visitor, coordinator, keeper and veterinarian. We use this procedure to redirect users to the sites that they are authorized to access.

## 5.6 Secondary Indices

- We have created a secondary index on the username attribute of the User table so that we can access different users through their usernames faster in the login process.
- We have created a secondary index on the date attribute of the Event table so that we can access events in certain periods faster by filtering the date attribute.

## 6. User's Manual

### 6.1. Introduction


Welcome to ZOOKEEPER! We offer you an easy to use interface that enables functionalities based on the user types. If you are a visitor, you can easily surf through the zoo page, buy items, donate money, leave a comment on the events you joined. Likewise, if you are an employee, you can easily access your data on your work, adjust and update them anytime.

### 6.2. Login

Regardless of whether you are an employee or a visitor you will be directed to the login page first. Clicking the login button after typing your user name and password will direct you to your home page.

Zoo

LOGIN REGISTER



login

password

LOG IN

URL:

<https://drive.google.com/file/d/1wfBlhmGbbrSVZGvhZf6sbGKG39EzWbIx/view?usp=sharing>

### 6.3. Register

If you are a visitor and have not been signed up yet, you can click on “REGISTER” in the header which will direct you to a form. You need to fully fill the form. If you pick a username or an e-mail address that is used for sign up before you will face an alert and your sign up will not be valid. If you fill the form with the needed requirements and click the submit button, you will successfully create a new account. You will be automatically directed to the login page. If for any reason you choose to log in with an existing account, you can click on “LOGIN” in the header to move to the login page.

Zoo

LOGINREGISTER

Name\*

Username\*

Birthday\*

E-Mail\*

Password\*

Confirm Password\*

Address\*

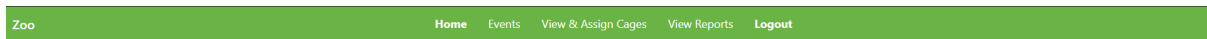
SUBMIT

URL:

[https://drive.google.com/file/d/1fDVx9Glgzr\\_bMiazjRF7X7nzzuwCnvxy/view?usp=sharing](https://drive.google.com/file/d/1fDVx9Glgzr_bMiazjRF7X7nzzuwCnvxy/view?usp=sharing)

## 6.4. Coordinator - Home Page

After login, coordinators will be directed to their home page. They can navigate freely to the desired page from the header. Clicking on “Logout” which will direct you to the login page.



Welcome, berkT



URL:

<https://drive.google.com/file/d/175WW2UxkJT0DPicGadvvNuFKKCwL85FJ/view?usp=sharing>

## 6.5. Coordinator - Events

In this page, coordinators can view the details of the events. A new event can be created through clicking “Create Event Button” which will direct you to a new page.



Events

ID	Name	Date	Start/End Time	Current Participants	Max Capacity	Description
1	Future Event 1	2022-05-08	10:00 / 12:00	21	100	None
2	Future Event 2	2022-07-08	10:00 / 12:00	31	50	None
7	Con. Org. 1	2022-06-08	10:00 / 11:00	40	150	Fund raise
8	Con. Org. 2	2022-07-03	10:00 / 11:00	30	50	Fund raise
9	Group Tour 1	2021-03-12	15:45 / 17:45	10	11	None
10	Group Tour 2	2021-04-10	12:50 / 15:50	9	11	None
11	Educational Program 1	2021-04-22	12:50 / 15:50	11	11	None

Create A New Event

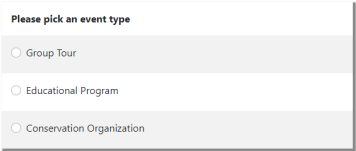
URL:

[https://drive.google.com/file/d/12jO-ggoQol-vXAIXwDXt0j6kRMEfJ\\_2p/view?usp=sharing](https://drive.google.com/file/d/12jO-ggoQol-vXAIXwDXt0j6kRMEfJ_2p/view?usp=sharing)

## 6.6. Coordinator - Create New Event

In order to create a new event, coordinators must specify the event type by clicking on the radio button near the options first. Upon clicking one of the radio buttons, a different form will be shown for each event type. Coordinator must at least fill the information boxes marked with \* to create an event. Clicking the button “Create Event” will successfully create a new event and you will remain on the same page. Clicking on “Back To Event List” will redirect you to the events page.

### 6.6.1. No Event Type Chosen



The screenshot shows a web application interface for a zoo. At the top is a green navigation bar with the text 'Zoo' on the left and links for 'Home', 'Events', 'View & Assign Cages', 'View Reports', and 'Logout' on the right. Below the navigation bar is a white form titled 'Please pick an event type'. The form contains three radio button options: 'Group Tour', 'Educational Program', and 'Conservation Organization'. Below the form is a green button labeled 'Back To Event List'.

URL:

[https://drive.google.com/file/d/1QscY3cyVkLRH-vbVxfEX-o6YTV\\_6iATR/view?usp=sharing](https://drive.google.com/file/d/1QscY3cyVkLRH-vbVxfEX-o6YTV_6iATR/view?usp=sharing)

## 6.6.2. Group Tour Form

Zoo

HomeEventsView & Assign CagesView ReportsLogout

Please pick an event type

☒ Group Tour

☐ Educational Program

☐ Conservation Organization

Back To Event List

Event Name\*

Max Capacity\*

Date\*

gg.aa.yyyy

Start Time\*

--:--

End Time\*

--:--

Description

Create Event

URL:

<https://drive.google.com/file/d/1VAjFOogcblTHRhf60GI6bQQ4u1BdJTm8/view?usp=sharing>

## 6.6.3. Educational Program Form

Zoo

HomeEventsView & Assign CagesView ReportsLogout

Please pick an event type

☐ Group Tour

☒ Educational Program

☐ Conservation Organization

Back To Event List

Event Name\*

Topic\*

Max Capacity\*

Date\*

gg.aa.yyyy

Start Time\*

--:--

End Time\*

--:--

Description

Create Event

URL:

[https://drive.google.com/file/d/1xIw9nIHLqjK0UQ7qJAI2VUN6kINp\\_h/view?usp=sharing](https://drive.google.com/file/d/1xIw9nIHLqjK0UQ7qJAI2VUN6kINp_h/view?usp=sharing)

## 6.6.4. Conservation Organisation Form

Zoo

HomeEventsView & Assign CagesView ReportsLogout

Please pick an event type

☐ Group Tour

☐ Educational Program

☒ Conservation Organization

Back To Event List

Event Name\*

Cause\*

Max Capacity\*

Goal Amount\*

Date\*

Start Time\*

End Time\*

Description

Create Event

URL:

<https://drive.google.com/file/d/1x1EuCBr77pb6ggIFJYMymeakyNmxlvsD/view?usp=sharing>

## 6.7. Coordinator - View & Assign Cages

Clicking on “View & Assign Cages” from the header will direct the coordinator to a page displaying an unassigned cages list with the required details given. Coordinator can click on the “Assign Cage” button to assign an unassigned cage to a keeper which will direct the coordinator to a new page. Coordinator can click on “View Cages Assigned by You” to view the list of cages that were assigned by that coordinator with the required information. Upon clicking on this button the “Assign Cage” button will still be available however, “View Cages Assigned by You” will be replaced by the “Back To Unassigned Cages” button which will display the unassigned cages list again.

### 6.7.1. Unassigned Cages

Zoo

Home

Events

View & Assign Cages

View Reports

Logout

Unassigned Cages

Cage ID	Cage Size	Cage Type	Animal Species
4	500	circle	monkey
7	2000	rectangular	horse
8	5000	square	elephant
9	600	square	alligator

Assign Cage

View Cages Assigned By You

URL:

[https://drive.google.com/file/d/1FRbw5OKtmMjgMf1XD8vOR\\_O9dVgNVBdj/view?usp=sharing](https://drive.google.com/file/d/1FRbw5OKtmMjgMf1XD8vOR_O9dVgNVBdj/view?usp=sharing)



## 6.7.2. Cages Assigned By You

Zoo

HomeEventsView & Assign CagesView ReportsLogout

Cages Assigned By You

Cage ID	Keeper ID	Cage Size	Cage Type	Animal Species
1	2	2000	circle	lion
3	2	300	circle	eagle
5	2	100	circle	snake
6	2	1500	rectangular	zebra

Assign Cage

Back to Unassigned Cages

URL:

<https://drive.google.com/file/d/13nQmZ-P08NOKh8v2EM93F4XrYQ-iCqUr/view?usp=sharing>

## 6.8. Coordinator - Assign Cage

Coordinators can pick a cage and keeper pair from their separate lists and complete the assigning process by clicking “Confirm”. Process will fail if either cage or keeper is not selected. Coordinator can direct to the previous page by clicking “Back To Assigned Cages”.

Zoo

HomeEventsView & Assign CagesView ReportsLogout

Assign Cage

Cage List

	Cage ID	Cage Type	Cage Size	Species In Cage
<input type="radio"/>	4	500	circle	monkey
<input type="radio"/>	7	2000	rectangular	horse
<input type="radio"/>	8	5000	square	elephant
<input type="radio"/>	9	600	square	alligator

Keeper List

	Keeper ID	Keeper Name
<input type="radio"/>	2	melihD
<input type="radio"/>	5	inciA
<input type="radio"/>	8	baykamS

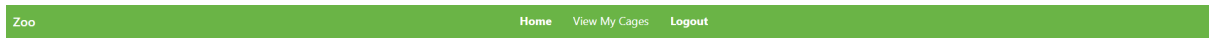
Confirm

Back To Assigned Cages

URL: <https://drive.google.com/file/d/1N8s-eltJ5LI91bEaMiGigImthu8XuKni/view?usp=sharing>

## 6.9. Keeper - Home Page

After login, keepers will be directed to their home page. They can navigate freely to the desired page from the header. Clicking on “Logout” which will direct you to the login page.



Welcome, melihD

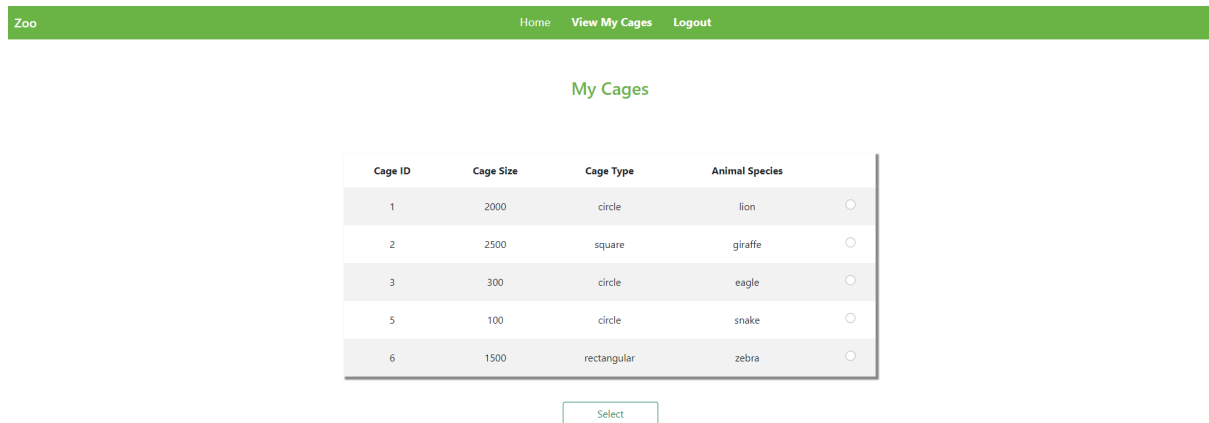


URL:

<https://drive.google.com/file/d/1MwRhFrNt0OMZboszConPVmGbTyheWp4c/view?usp=sharing>

## 6.10. Keeper - View My Cages

In this page, keepers can view the list of the cages assigned to them with detailed information. A cage can be selected and the animals residing in that cage can be listed after clicking the “Select” button.



URL:

[https://drive.google.com/file/d/10Le8VPBerhJAGE6-tQ-TiH\\_WkuLpwbox/view?usp=sharing](https://drive.google.com/file/d/10Le8VPBerhJAGE6-tQ-TiH_WkuLpwbox/view?usp=sharing)

## 6.11. Keeper - Animals

In this page, the list of the animals residing in the selected cage will be listed. An animal can be selected and their trainings can be viewed after clicking on the “Select” button. “Back To Cages” will direct the keeper to the previous page.

Zoo

HomeView My CagesLogout

Animals

Animal ID	Animal Name	Animal Species	
1	Jack	lion	<input type="radio"/>
2	Celine	lion	<input type="radio"/>
3	Kendrick	lion	<input type="radio"/>
4	Brian	lion	<input type="radio"/>
5	Tommy	lion	<input type="radio"/>

Select

Back To Cages

URL:

[https://drive.google.com/file/d/1U4MhloooOLpyGvt\\_EvcZglb\\_GuwETJng/view?usp=sharing](https://drive.google.com/file/d/1U4MhloooOLpyGvt_EvcZglb_GuwETJng/view?usp=sharing)

## 6.12 Keeper - View Trainings

In this page, keepers can view the trainings for the animals and update their dates by selecting a training, a date and clicking the “Update” button or remove the training by clicking the “Remove” button. Keepers can direct to the previous page by clicking “Back To Animals”.

Zoo

HomeView My CagesLogout

Trainings

Training Topic	Training Date	
move around	2021-05-15	<input type="radio"/>
turn	2021-05-15	<input type="radio"/>
sit	2021-05-15	<input type="radio"/>

gg.aayyy

Update

Remove

Back To Animals

URL:

<https://drive.google.com/file/d/15UtWFK1OYYBtsB9JB0EPmtVQtyPdgFX-/view?usp=sharing>

## 6.13 Visitor - Home Page

After login, visitors will be directed to their home page. They can navigate freely to the desired page from the header. Clicking on “Logout” which will direct you to the login page.

Welcome, aleynaS



URL:  
<https://drive.google.com/file/d/1WwsgLFeolXvHeyP-UJTg8UvkIVhcGq8M/view?usp=sharing>

## 6.14 Visitor - Shops

In this page, visitors can see the shops and their details and go to their specific pages for shopping by clicking the “Go” buttons.

Zoo			
	Home	Conservation Organizations	Group Tours
		<b>Shops</b>	Animals
			Membership
			<b>Logout</b>
	Shop Name	Area	Description
<a href="#">Go</a>	Gift_Shop	Gift_Shop	Key chains, magnets...
<a href="#">Go</a>	Toy_Shop	Toy_Shop	Zoo themed toys

URL:  
<https://drive.google.com/file/d/1aLnwHYDpW8SPLatn9K4rEV5FSTSI6X73/view?usp=sharing>

## 6.15 Visitor - Items

This page is created based on the shop choice in the shops page. Items of the shops and their prices and stocks are listed. Visitors can choose the amount of any item that they want and click the “buy items” button to buy them. Also, they can click the “see my funds” button to check their total amount of money.

### 6.15.1. Gift Shop Example

Zoo			
Home   Conservation Organizations   Group Tours <b>Gift Shop</b> Animals   Membership   Logout			
Amount	Item Name	Price(\$)	Stock
<input type="text"/>	key chain	5	10
<input type="text"/>	magnet	10	15
<input type="text"/>	book seperator	3	20

Back to the ShopsBuy ItemsSee my Funds

URL:

<https://drive.google.com/file/d/1UWNIIJiMByLSTMapKk8aEYBEh4HTUyUt/view?usp=sharing>

### 6.15.2. Toy Shop Example

Zoo			
Home   Conservation Organizations   Group Tours <b>Toy Shop</b> Animals   Membership   Logout			
Amount	Item Name	Price(\$)	Stock
<input type="text"/>	leo figure	15	1
<input type="text"/>	zebra figure	10	1

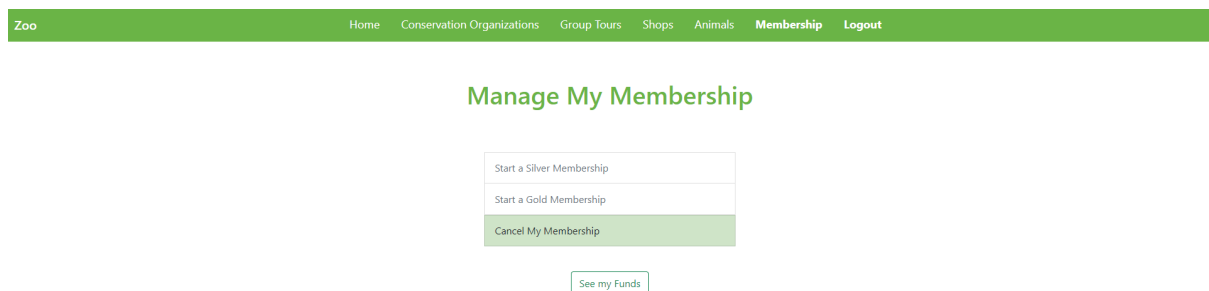
Back to the ShopsBuy ItemsSee my Funds

URL:

<https://drive.google.com/file/d/1OIjG-EY-MwBC978D-Rc83XA3KAWVmGW-/view?usp=sharing>

## 6.16 Visitor - Membership

In this page, users will be able to start or cancel a membership. If a user already has a membership of any kind, only “Cancel My Membership” button will be enabled. Otherwise, “Cancel My Membership” button will be disabled and “Start a Membership” buttons will be enabled. Also, users will be able to see their total amount of money by pressing the “See my Funds” button.

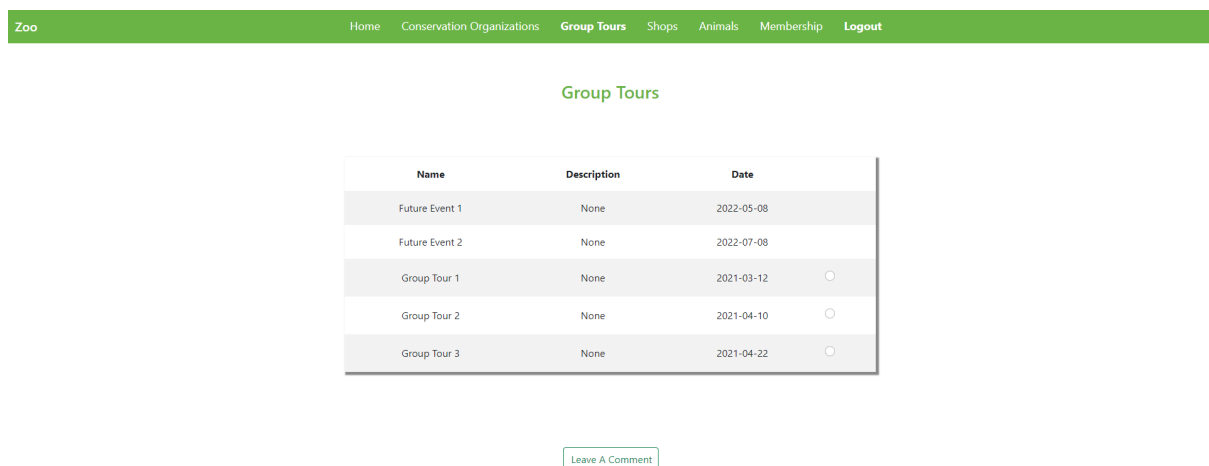


URL:

[https://drive.google.com/file/d/1lecMPnKb\\_6EGIUQQ\\_U4ip3OXfDIHW1EZ/view?usp=sharing](https://drive.google.com/file/d/1lecMPnKb_6EGIUQQ_U4ip3OXfDIHW1EZ/view?usp=sharing)

## 6.17 Visitor - Group Tours

In this page, visitors can see the upcoming and past group tours that they have attended or will attend. They can leave comments to the group tours that they have attended by selecting the tour and clicking the “Leave a Comment” button.



URL:

<https://drive.google.com/file/d/1m5ZwZ1lt3NDaGURhpbzG3F8OMTaSONRM4/view?usp=sharing>



## 6.18 Visitor - Comment

In this page, visitors can see the comments that have been made to the selected group tour and submit their own comments by typing them and clicking the “Comment” button.

Zoo

Home

Conservation Organizations

Group Tours

Shops

Animals

Membership

Logout

Add Your Comment

Comment

Back To Event List

Comments

Username	Content	Date	Likes	Dislikes
Aleyna Sütbaşı	This was fun!	2021-05-16	0	0
Batuhan Koçak	I want to join again!	2021-05-16	0	0

URL:

<https://drive.google.com/file/d/1QXaU1tgtpkMECLG3ISEeZtWButca7boa/view?usp=sharing>

## 6.19 Visitor - Conservation Organizations

In this page, visitors can see the conservation organizations, their details and donate to them by writing the amount and clicking the “Donate” button.

Zoo

Home

Conservation Organizations

Group Tours

Shops

Animals

Membership

Logout

Conservation Organizations

Name	Cause	Description	Goal	Collected
Con. Org. 1	New Lion Cages	Fund raise	\$10000	\$5000 <input type="text"/>
Con. Org. 2	Street Animal Food Fund	Fund raise	\$10000	\$5000 <input type="text"/>

Remaining Money: \$300

Donate

URL:

[https://drive.google.com/file/d/1JLW40mYn4lxtT\\_rFF5\\_6YifdDC2DEPza/view?usp=sharing](https://drive.google.com/file/d/1JLW40mYn4lxtT_rFF5_6YifdDC2DEPza/view?usp=sharing)