PETS CLINIC MANAGEMENT SYSTEM

Team members
فاطمة ابراهيم كامل ابراهيم
عبد المنعم حامد عبد المنعم حامد
عمرو عوني عبد التواب يوسف
عبد الرحمن عبد الهادي محمد
طه ابراهيم طه ابراهيم

SUPERVISED / DR. Shaimaa Talaat

Dept: CS

Contents

Ch1: Introduction

1.1 Introduction of the Project Pets Clinic Management System

The "Pets Clinic Management System" has been developed to override the problems prevailing in the practicing manual system. This software is supported to eliminate and, in some cases, reduce the hardships faced by this existing system. Moreover, this system is designed for the particular need of the company to carry out operations in a smooth and effective manner.

The application is reduced as much as possible to avoid errors while entering the data. It also provides error message while entering invalid data. No formal knowledge is needed for the user to use this system. Thus, by this all it proves it is user-friendly. Pets Clinic Management System, as described above, can lead to error free, secure, reliable and fast management system. It can assist the user to concentrate on their other activities rather to concentrate on the record keeping. Thus, it will help organization in better utilization of resources.

Every organization, whether big or small, has challenges to overcome and managing the information of Doctor, Clinic, Appointment, Medicines, Tests. Every Pets Clinic Management System has different Clinic need; therefore, we design exclusive employee management systems that are adapted to your managerial requirements. This is designed to assist in strategic planning and will help you ensure that your organization is equipped with the right level of information and details for your future goals. Also, for those busy executive who are always on the go, our systems come with remote access features, which will allow you to manage your workforce anytime, at all times. These systems will ultimately allow you to better manage resources.

1.2 Definition of the Project Pets Clinic Management System

The purpose of Pets Clinic Management System is to automate the existing manual system by the help of computerized equipment and full-fledged computer software, fulfilling their requirements, so that their valuable data/information can be stored for a longer period with easy accessing and manipulation of the same. The required software and hardware are easily available and easy to work with.

Pets Clinic Management System, as described above, can lead to error free, secure, reliable and fast management system. It can assist the user to concentrate on their other activities rather to concentrate on the record keeping. Thus, it will help organization in better utilization of resources. The organization can maintain computerized records without redundant entries. That means that one need not be distracted by information that is not relevant, while being able to reach the information.

The aim is to automate its existing manual system by the help of computerized equipment and full-fledged computer software, fulfilling their requirements, so that their valuable data/information can be stored for a longer period with easy accessing and manipulation of the same. Basically, the project describes how to manage for good performance and better services for the clients.

1.3 Objective of the Project Pets Clinic Management System

The main objective of the Project on Pets Clinic Management System is to manage the details of Clinic, Doctor, Pet, Appointment, Tests.

It manages all the information about Clinic, Medicines, Tests, Clinic. The project is totally built at administrative end and thus only the administrator is guaranteed the access. The purpose of the project is to build an application program to reduce the manual work for managing the Clinic, Vets, Medicines, Patient. It tracks all the details about the Patient, Appointment, Tests.

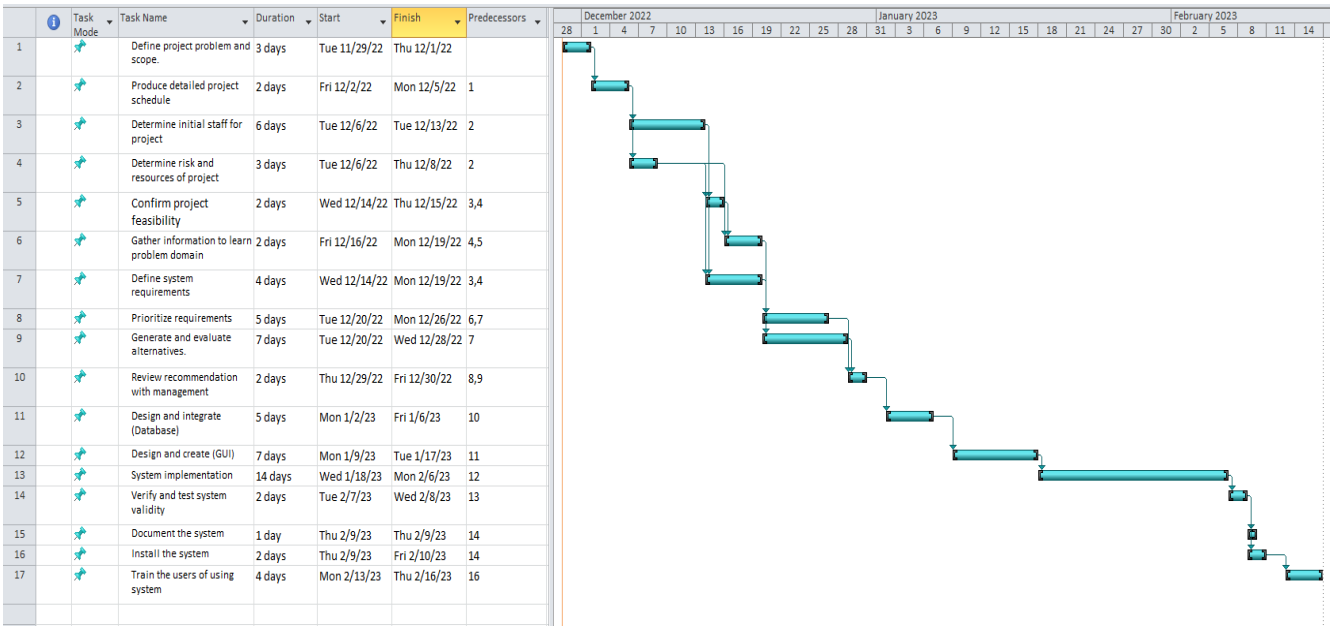
Functionalities provided by Pet Clinic Management System are as follows:

- Pets Clinic Management System manage the Medicines details, Appointment details, Tests details, Clinic.
- Manage the information of doctor
- Shows the information and description of the Clinic, Pets
- To increase efficiency of managing the Clinic.
- It deals with monitoring the information and transactions of Appointment.
- Editing, adding, and deleting of Records is improved which results in proper resource management of Clinic data.
- Manage the information of Appointments.
- It tracks all the information of Vet, Medicines, Appointment etc.

1.4 WBS of Pets Clinic Management System

1.5 Gantt chart

Gantt chart for total project:



	Task Mode	Task Name	Duration	Start	Finish	Predecessors
1		Define project problem and scope.	3 days	Tue 11/29/22	Thu 12/1/22	
2		Produce detailed project schedule	2 days	Fri 12/2/22	Mon 12/5/22	1
3		Determine initial staff for project	6 days	Tue 12/6/22	Tue 12/13/22	2
4		Determine risk and resources of project	3 days	Tue 12/6/22	Thu 12/8/22	2
5		Confirm project feasibility	2 days	Wed 12/14/22	Thu 12/15/22	3,4
6		Gather information to learn problem domain	2 days	Fri 12/16/22	Mon 12/19/22	4,5
7		Define system requirements	4 days	Wed 12/14/22	Mon 12/19/22	3,4
8		Prioritize requirements	5 days	Tue 12/20/22	Mon 12/26/22	6,7
9		Generate and evaluate alternatives.	7 days	Tue 12/20/22	Wed 12/28/22	7
10		Review recommendation with management	2 days	Thu 12/29/22	Fri 12/30/22	8,9
11		Design and integrate (Database)	5 days	Mon 1/2/23	Fri 1/6/23	10
12		Design and create (GUI)	7 days	Mon 1/9/23	Tue 1/17/23	11
13		System implementation	14 days	Wed 1/18/23	Mon 2/6/23	12
14		Verify and test system validity	2 days	Tue 2/7/23	Wed 2/8/23	13
15		Document the system	1 day	Thu 2/9/23	Thu 2/9/23	14
16		Install the system	2 days	Thu 2/9/23	Fri 2/10/23	14
17		Train the users of using system	4 days	Mon 2/13/23	Thu 2/16/23	16

1.6 Risks of the Project Pets Clinic Management System

- **Size**: The Size of the clinical database System has a very huge database that makes the system heavy, and this increases the access time of the database.
- **Time**: The time given to complete the project is only two months which is very limited for completion of such a big project.
- **Expenses**: The process of completing the whole project will require some money which will cater for Internet access and purchase of a domain to test the system.
- Clinical management System will only benefit those who know about the internet, this means the computer illiterate will not be able to use it.
- No money was allocated for the project, and this may to some extend limit the product scope.

1.7 Safety

System safety focuses on identifying hazards, their causal factors, and predicting the resultant (e.g., potential damage) severity and probability.

The goal is to reduce or eliminate the severity and probability of the identified hazards, and to minimize risk and severity where the hazards cannot be eliminated. hazard is defined as: "a real or potential condition that could lead to an unplanned event or series of events (i.e., mishap) resulting in death, injury, occupational illness, damage to or loss of equipment or property, or damage to the environment.".

An example of hazards that might occur in the pet's clinic management system -if not handled-:

> *The medical database that's holding details of the drugs, administered to the pets, errors(hazard) in this system might result in an incorrect drug dosage being taken by the pets,*
>
> *thus, jeopardizing their lives! (I. e., damage, with a high severity).*

this kind of risks needs to be considered when planning for such a critical system, risk assessment is one way to achieve that.

Since safety is a critical characteristic for modern systems to have,

System Safety can be achieved through the following steps:

- Identifying hazards
- identifying the risks using hazard analysis techniques as early as possible in the system life cycle, risk levels can be provided in the hazard analysis so that we can define the damages severity.
- Hazard avoidance, the system is designed so that hazards are avoided and Developing options to eliminate or control hazards
- Hazard detection and removal, the system is designed so that hazards are detected and removed before they result in an accident.

- Define residual risks, and that can be achieved by conducting risk assessments on regular basis.
- Damage limitation, the system may include protection features that minimize the damage that may result from an accident.

1.8 Security

- **Well defined Access Controls**: Make sure only the people who are authorized to access certain data have this privilege. Give admins the minimum number of rights they need for their specific job. This will make it much harder for unauthorized users to break into your system and steal valuable information from you.
- **Secure remote access from outside the clinic premises**: Using VPN tunnels helps to a great extent. Enforce security features like two-factor authentication, multi-factor authorization, and fingerprint scans for any type of access to the system; this will make sure that no one has unauthorized or unauthenticated remote connections.
- **Ensure data protection**: Never store sensitive information without encrypting it first as this may lead you into a situation where confidential records are stolen which could result in some serious consequences.
- **encrypting databases and folders:** encrypting databases or folders with sensitive data which could be used by hackers to get access into your system and prevent unauthorized people from handling the data and it is only used by admins.
- **Audit control capabilities**: The Audit Control (or Logging) module is designed to improve transparency and provide a thorough, accurate overview of all transactions in the application.

Chapter2: System Analysis

2.1 Introduction

In this phase, the requirements of the system are being analyzed on the existing system and all requirements that are needed to develop a new system. In this phase, the information regarding the Pets Clinic Management either interviews, articles, books, or research paper is being collected and gathered.

2.2 System Requirements

Functional requirements:

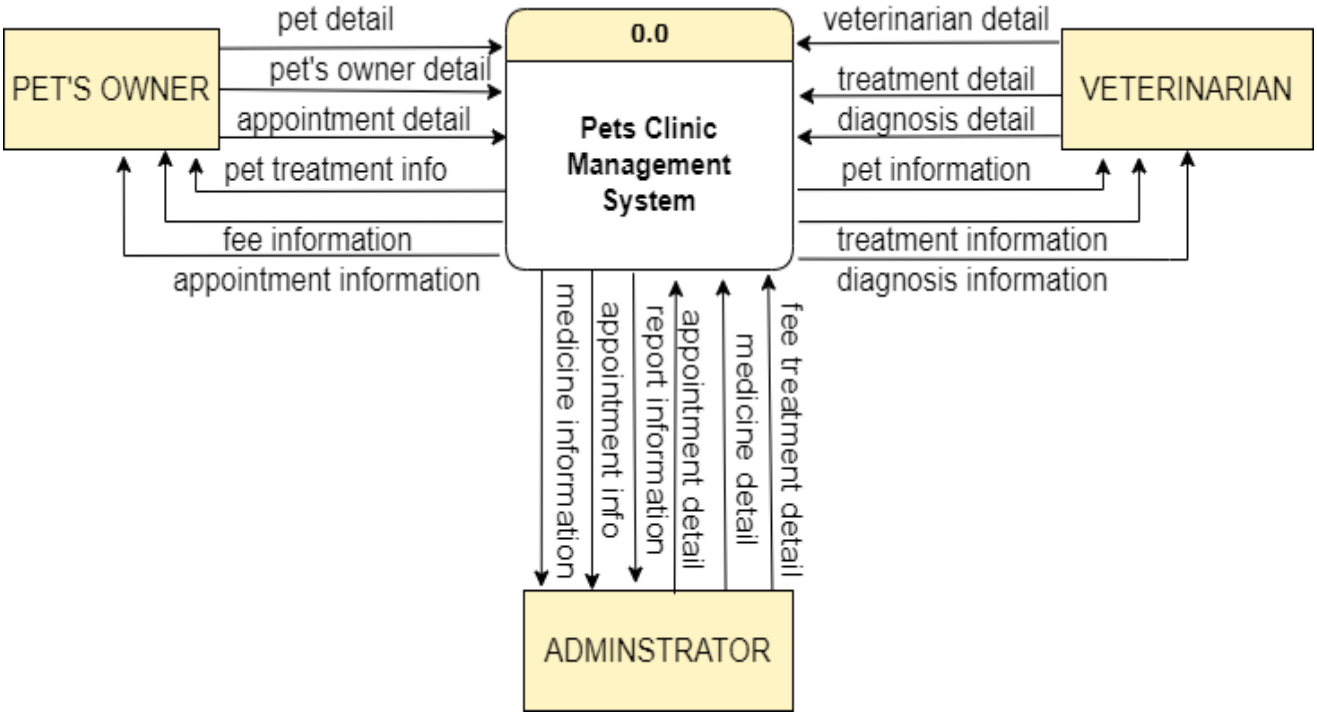
- Adding customer: the pets clinic system enables the staff to include new customers to the system.
- Record customer's basic data.
- Registration of pet details.
- search for a pet.
- Book a time for an appointment.
- Remove medication or procedure from the application.
- Add medication or a procedure to application.
- update medication or procedure from the application.
- track the detailed information of pet, booking, appointment, medicine.
- vet can view their pet’s treatment records and details easily.

Non-Functional requirements:

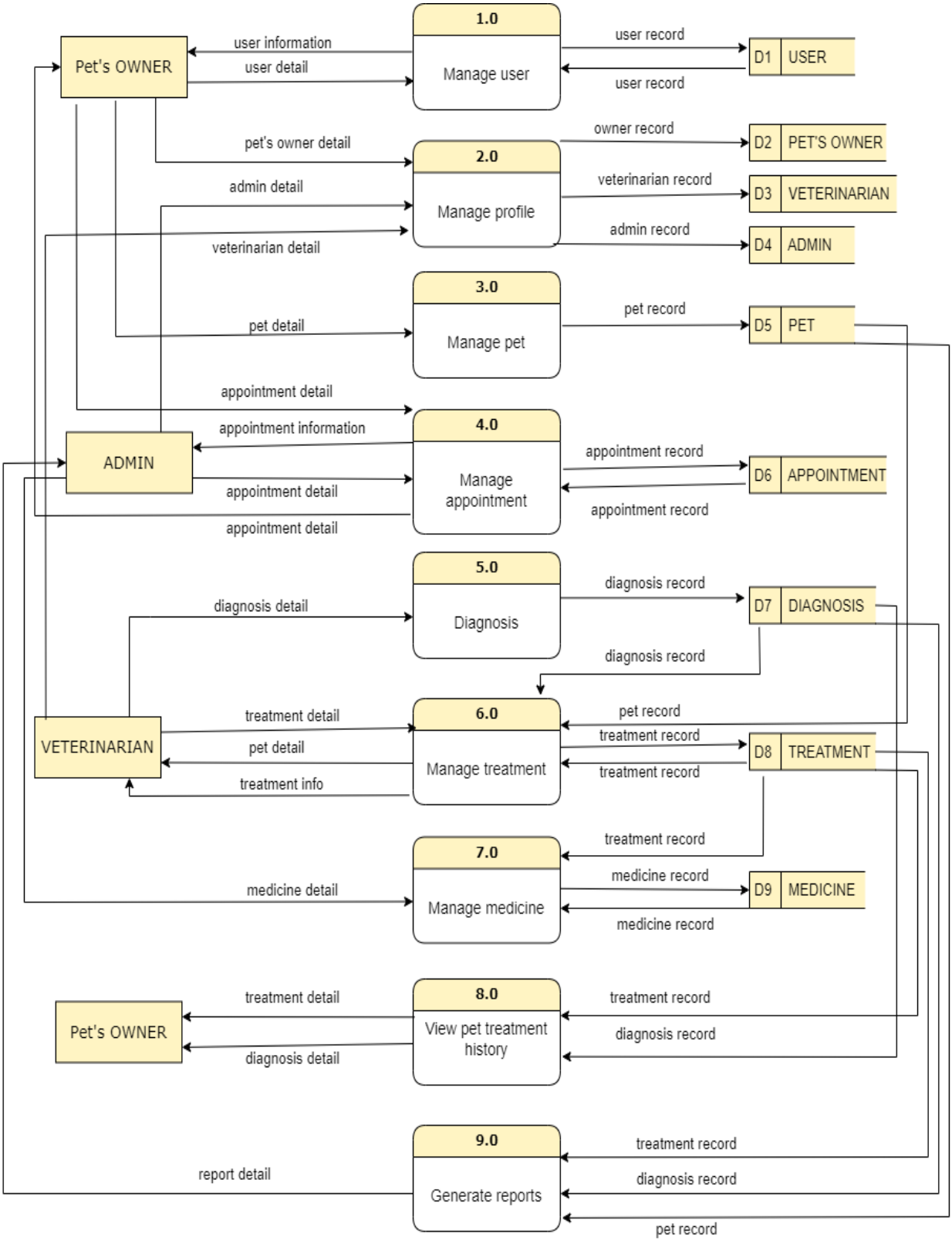
- Flexibility: The system is convenient and flexible to be used.
- Availability: The system is available all the time.
- Usability: the system is easy to use by Staff.
- Security: system should be secure.
- Maintainability: the system is easy to modify and maintain.
- Safety: system should be safe.

2.3 Context Diagram

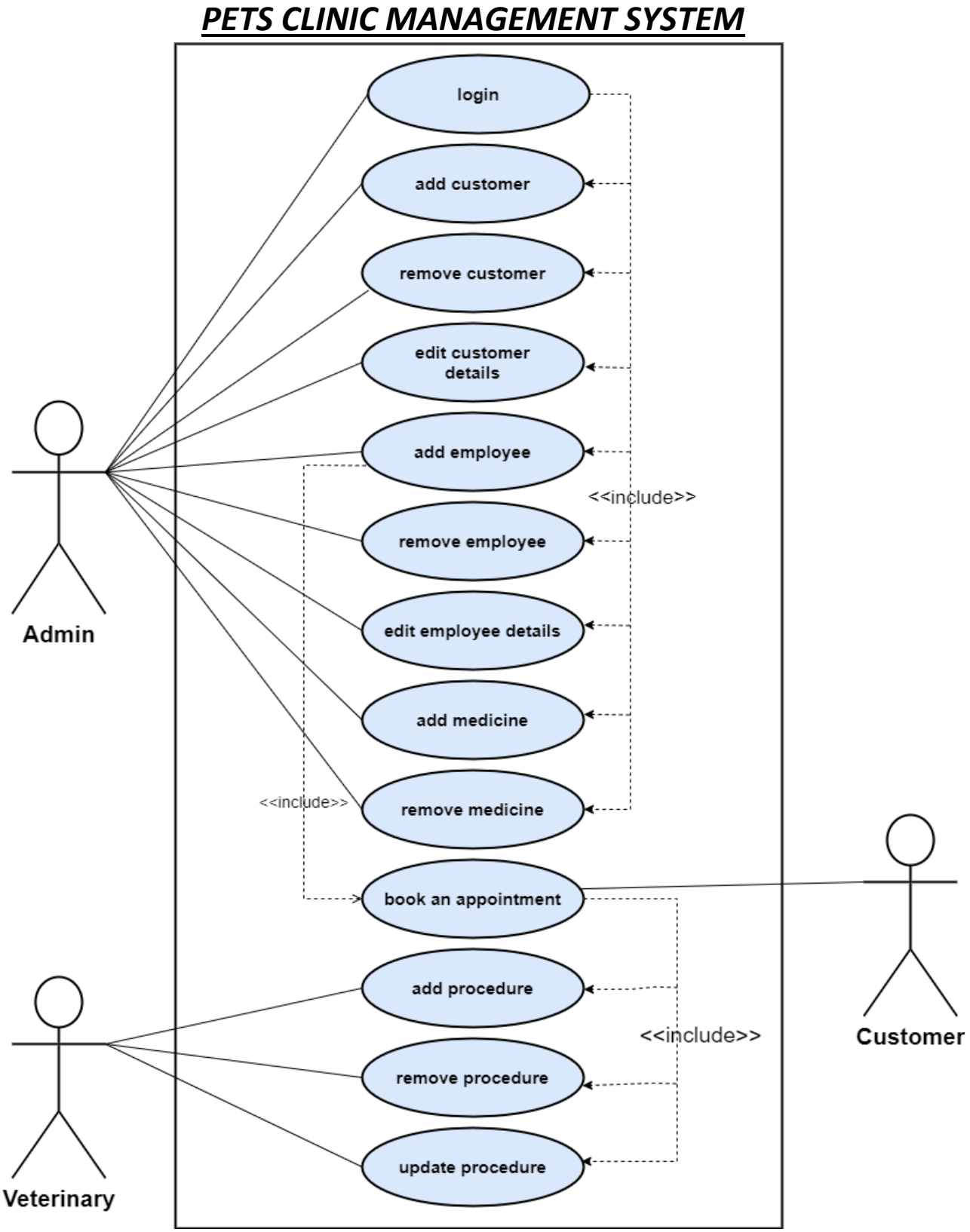
The context diagram defines as the highest level that represent the entire system. There are three external entities which are pet's owner, veterinarian, and administrator. The pet's owner needs to give their personal details and pet details to the system after they have registered or login to the system. Once successful into the pet's owner own page, they can make pet appointment, view pet treatment information and can view the free information of their pet treatment. For veterinarian can give treatment information and diagnosis information based on pet detail and their historical treatment or diagnosis information. The administrator will provide their appointment information, medicine detail, fee information and can get the report about all information.



2.4 Data flow diagram



2.5 Use case diagram



➢ Login

Use case id:	1
Use case name:	Add customer
Introduction:	This use case describes the process of logging in the system by the administrator.
Actors:	Administrator
Pre-condition:	Administrator must provide the correct password to login
Post-condition:	There will be a prompted message: ”Logging in successfully.”
Basic flow:	Get administration name Get administration password Checking the validation of password Confirm logging in process
Alternative flow:	Providing an incorrect password for the administrator (Reenter password or Error will be prompted)
Special requirements:	Administrator must provide the correct password to display admin settings.
Relationships:	The base case includes checking the validation of the password id administrator

➢ Add customer

Use case id:	2
Use case name:	Add customer
Introduction:	This use case describes the process of adding the details of a customer by the administrator.
Actors:	Administrator
Pre-condition:	Administrator must login to be able to add new customer.
Post-condition:	There will be a prompted message: ”New customer was added successfully.”
Basic flow:	Get customer and his pet details Confirm adding process
Alternative flow:	Providing an incorrect password for the administrator (Error will be prompted). or Customer details incomplete
Special requirements:	Administrator must login to add new customer
Relationships:	The base case includes checking the logging in process of administrator.

➢ Remove customer

Use case id:	3
Use case name:	Remove customer
Introduction:	This use case describes the process of removing the details of a customer by the administrator.
Actors:	Administrator
Pre-condition:	Administrator must login to be able remove the customer. Administrator must enter the correct customer id to remove the customer.
Post-condition:	There will be a prompted message: ”Customer was removed successfully.”
Basic flow:	Get customer id. Check the validation of customer id. Confirm removing process
Alternative flow:	Providing an incorrect customer id.
Special requirements:	Administrator must enter the customer id correctly to remove customer
Relationships:	The base case includes checking the validation of the customer id.

➢ Edit customer details

Use case id:	4
Use case name:	Edit customer details
Introduction:	This use case describes the process of editing the details of a customer by the administrator.
Actors:	Administrator
Pre-condition:	Administrator must login to be able to edit the customer details. Administrator must enter the correct customer id to edit the customer details.
Post-condition:	There will be a prompted message: ”Customer details was editing successfully.”
Basic flow:	Get customer id. Check the validation of customer id. Confirm editing process
Alternative flow:	Providing an incorrect customer id.
Special requirements:	Administrator must enter the customer id correctly to edit the customer details.
Relationships:	The base case includes checking the validation of the customer id.

➢ Add employee

Use case id:	5
Use case name:	Add employee
Introduction:	This use case describes the process of adding the details of an employee by the administrator.
Actors:	Administrator
Pre-condition:	Administrator must login to be able to add new employee.
Post-condition:	There will be a prompted message: ”New employee was added successfully.”
Basic flow:	Get employee details Confirm adding process
Alternative flow:	Providing an incorrect password for the administrator (Error will be prompted). or Employee details incomplete
Special requirements:	Administrator must login to add new employee
Relationships:	The base case includes checking the logging in process of administrator

➢ Remove employee

Use case id:	6
Use case name:	Remove employee
Introduction:	This use case describes the process of removing the details of an employee by the administrator.
Actors:	Administrator
Pre-condition:	Administrator must login to be able remove the employee. Administrator must enter the correct employee id to remove the employee.
Post-condition:	There will be a prompted message: ”Employee was removed successfully.”
Basic flow:	Get employee id. Check the validation of employee id. Confirm removing process
Alternative flow:	Providing an incorrect employee id.
Special requirements:	Administrator must enter the employee id correctly to remove customer
Relationships:	The base case includes checking the validation of the employee id.

➢ Edit employee details

Use case id:	7
Use case name:	Edit employee details
Introduction:	This use case describes the process of editing the details of an employee by the administrator.
Actors:	Administrator
Pre-condition:	Administrator must login to be able to edit the employee details. Administrator must enter the correct employee id to edit the employee details.
Post-condition:	There will be a prompted message: ”Employee details was editing successfully.”
Basic flow:	Get employee id. Check the validation of employee id. Confirm editing process
Alternative flow:	Providing an incorrect employee id.
Special requirements:	Administrator must enter the employee id correctly to edit the employee details.
Relationships:	The base case includes checking the validation of the employee id.

➢ Add medicine

Use case id:	8
Use case name:	Add medicine
Introduction:	This use case describes the process of adding the medicine by the administrator.
Actors:	Administrator
Pre-condition:	Administrator must login to be able to add new customer.
Post-condition:	There will be a prompted message: ”Medicine was added successfully.”
Basic flow:	Get medicine details Confirm adding process
Alternative flow:	Providing an incorrect password for the administrator (Error will be prompted). or medicine details incomplete
Special requirements:	Administrator must login to add medicine
Relationships:	The base case includes checking the logging in process of administrator

➢ Remove medicine

Use case id:	9
Use case name:	Remove medicine
Introduction:	This use case describes the process of removing the medicine by the administrator.
Actors:	Administrator
Pre-condition:	Administrator must login to be able remove the medicine. Administrator must enter the correct medicine id to remove the medicine.
Post-condition:	There will be a prompted message: ”Medicine was removed successfully.”
Basic flow:	Get medicine id. Check the validation of medicine id. Confirm removing process
Alternative flow:	Providing an incorrect medicine id.
Special requirements:	Administrator must enter the medicine id correctly to remove medicine
Relationships:	The base case includes checking the validation of the medicine id.

➢ Book an appointment

Use case id:	10
Use case name:	Book an appointment
Introduction:	This use case describes the process of booking an appointment by the Customer.
Actors:	Customer
Pre-condition:	Customer must be added before booking.
Post-condition:	There will be a prompted message: ”Appointment was booked successfully.”
Basic flow:	Get pet id. Select appointment details. Confirm booking process
Alternative flow:	Customer doesn’t register before booking.
Special requirements:	Customer must enter the pet id and select appointment details.
Relationships:	The base case includes checking if the customer register or not.

➢ Add procedure

Use case id:	11
Use case name:	Add procedure
Introduction:	This use case describes the process of adding the details of procedure by the veterinarian.
Actors:	Veterinarian
Pre-condition:	Customer must book an appointment to enable veterinarian adds a procedure. Veterinary must be added to the system by admin.
Post-condition:	There will be a prompted message: ”Procedure was added successfully.”
Basic flow:	Get procedure details Confirm adding process
Alternative flow:	Customer doesn’t book an appointment before.
Special requirements:	Customer must book an appointment to enable veterinarian adds a procedure.
Relationships:	The base case includes checking booking an appointment by the customer.

➢ Remove procedure

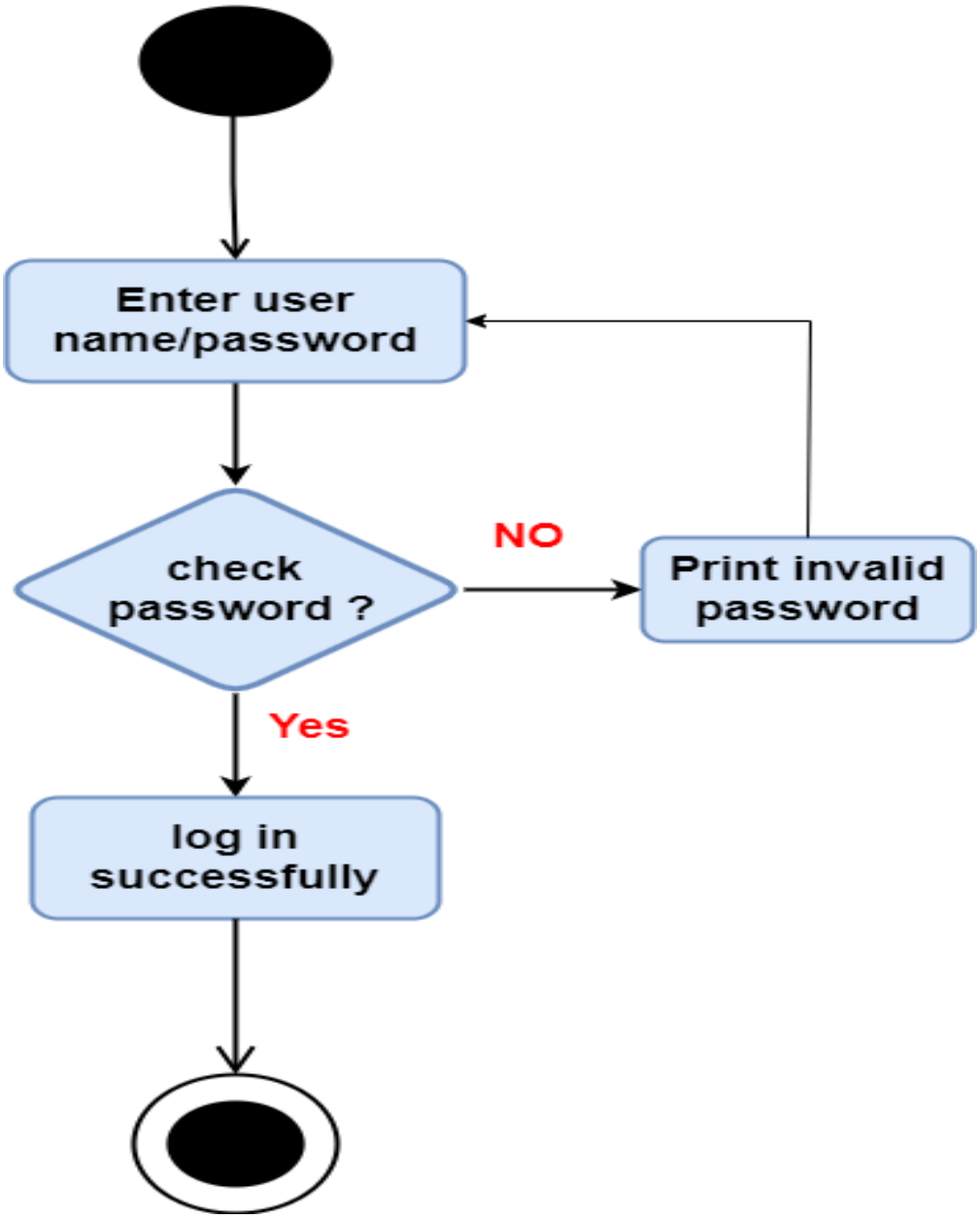
Use case id:	12
Use case name:	Remove procedure
Introduction:	This use case describes the process of removing the procedure by the veterinarian.
Actors:	Veterinarian
Pre-condition:	Customer must book an appointment to enable veterinarian removes a procedure. Veterinary must be added to the system by admin. Procedure id must be entered correctly to remove procedure.
Post-condition:	There will be a prompted message: ”Procedure was removed successfully.”
Basic flow:	Get procedure id. Check the validation of procedure id. Confirm removing process
Alternative flow:	Providing an incorrect procedure id.
Special requirements:	Procedure id must be entered correctly to remove procedure
Relationships:	The base case includes checking the validation of the procedure id.

➢ Update procedure

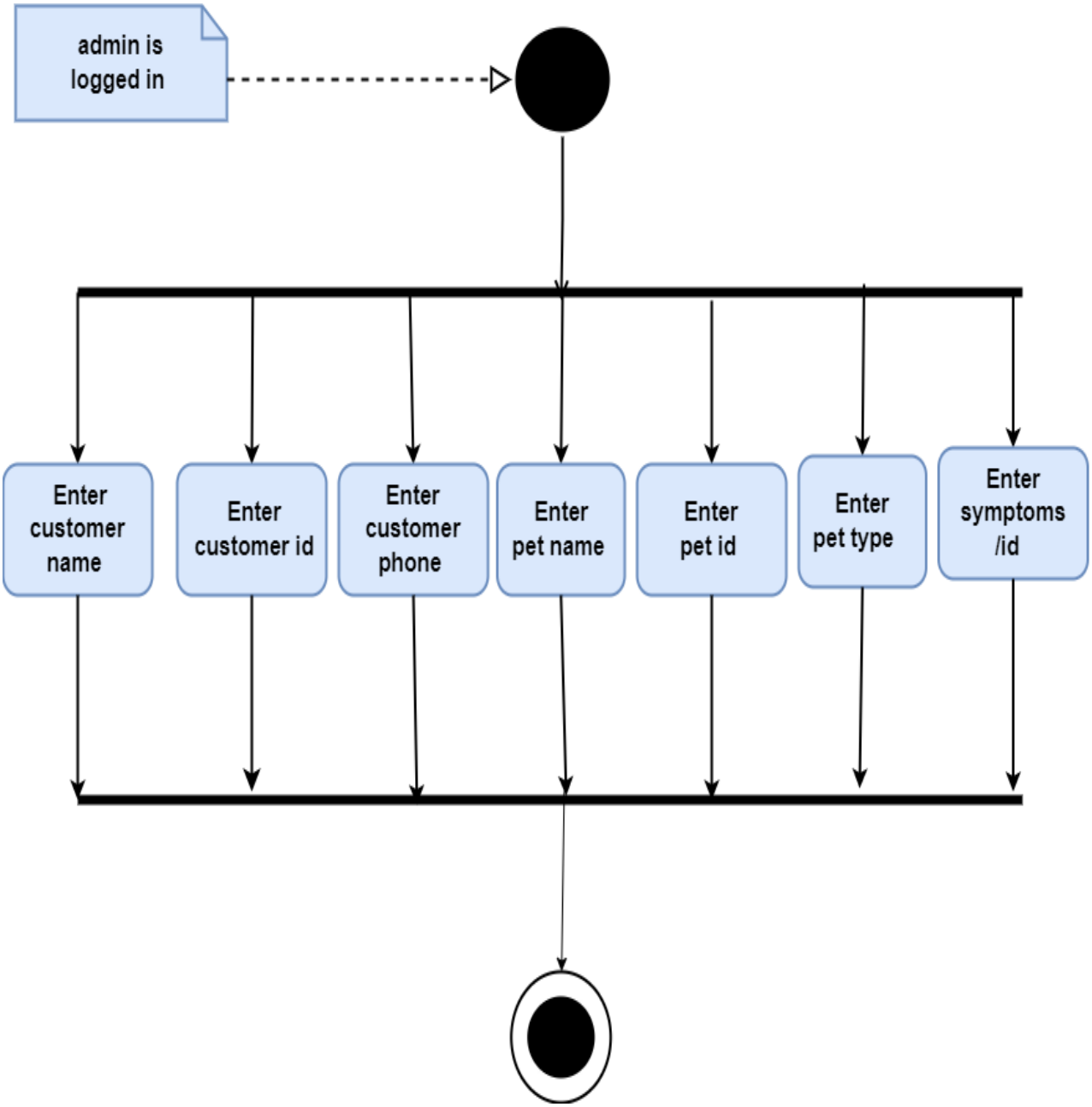
Use case id:	13
Use case name:	Update procedure
Introduction:	This use case describes the process of updating the details of procedure by the veterinarian.
Actors:	Veterinarian
Pre-condition:	Customer must book an appointment to enable veterinarian updates a procedure. Veterinary must be added to the system by admin. procedure id must be entered correctly to edit the procedure details.
Post-condition:	There will be a prompted message: ”Procedure was updated successfully.”
Basic flow:	Get procedure id Check the validation of procedure id. Update procedure details Confirm updating process
Alternative flow:	Customer doesn’t book an appointment before. Or Providing a procedure id.
Special requirements:	Customer must book an appointment to enable veterinarian updates a procedure.
Relationships:	The base case includes checking booking an appointment by the customer and checking the validation of the procedure id.

2.6 Activity diagram

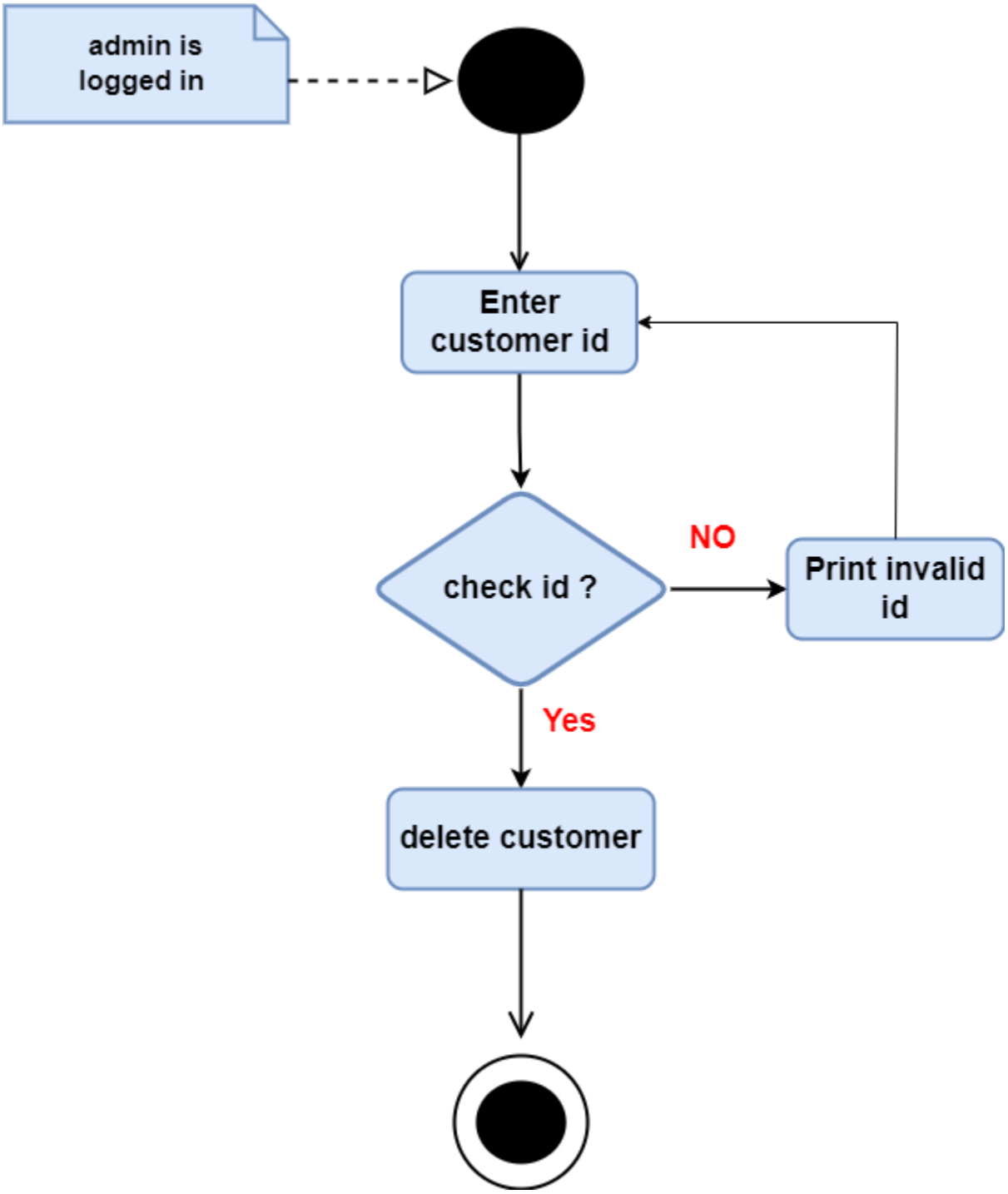
- Login



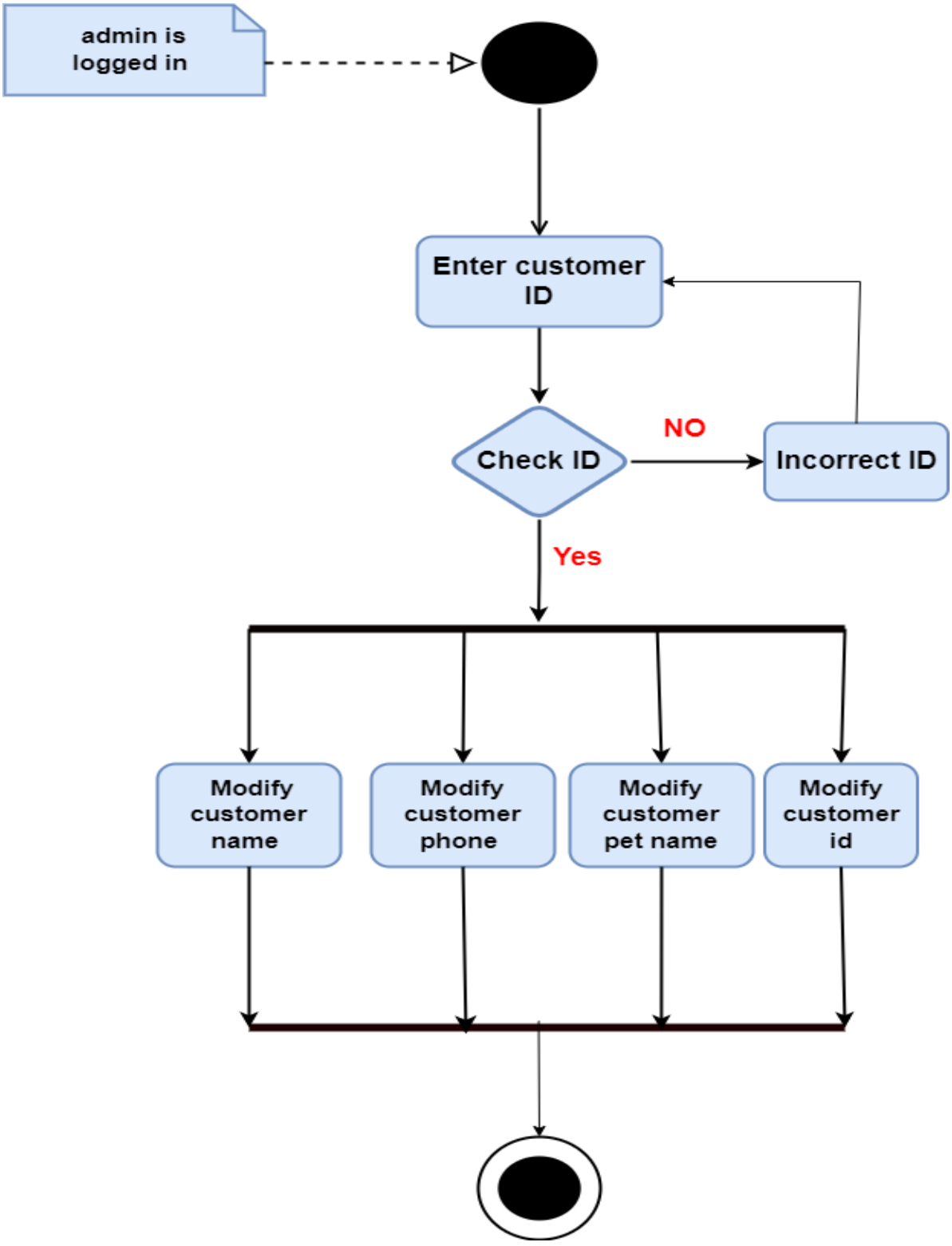
➢ Add customer



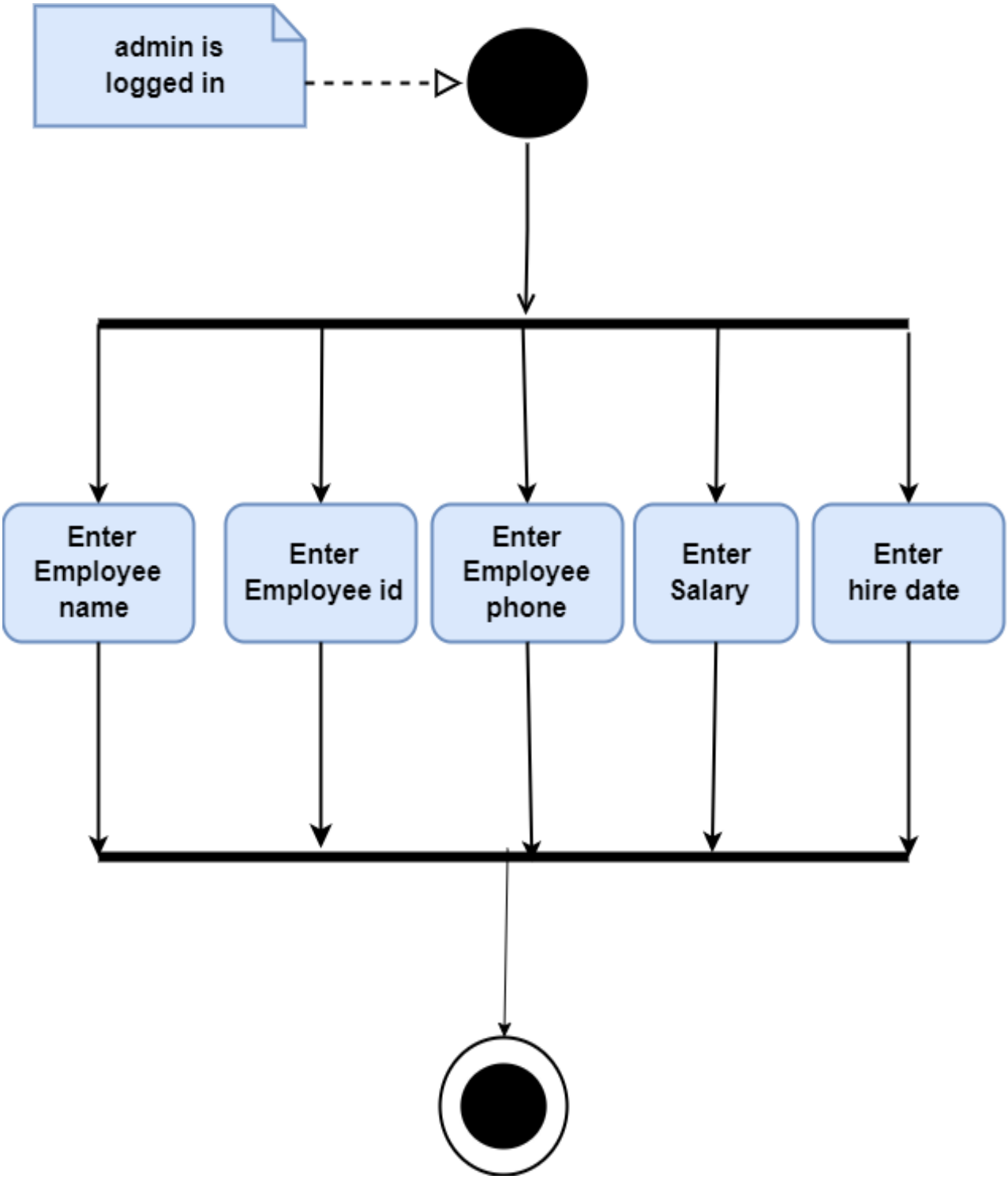
➢ Remove customer



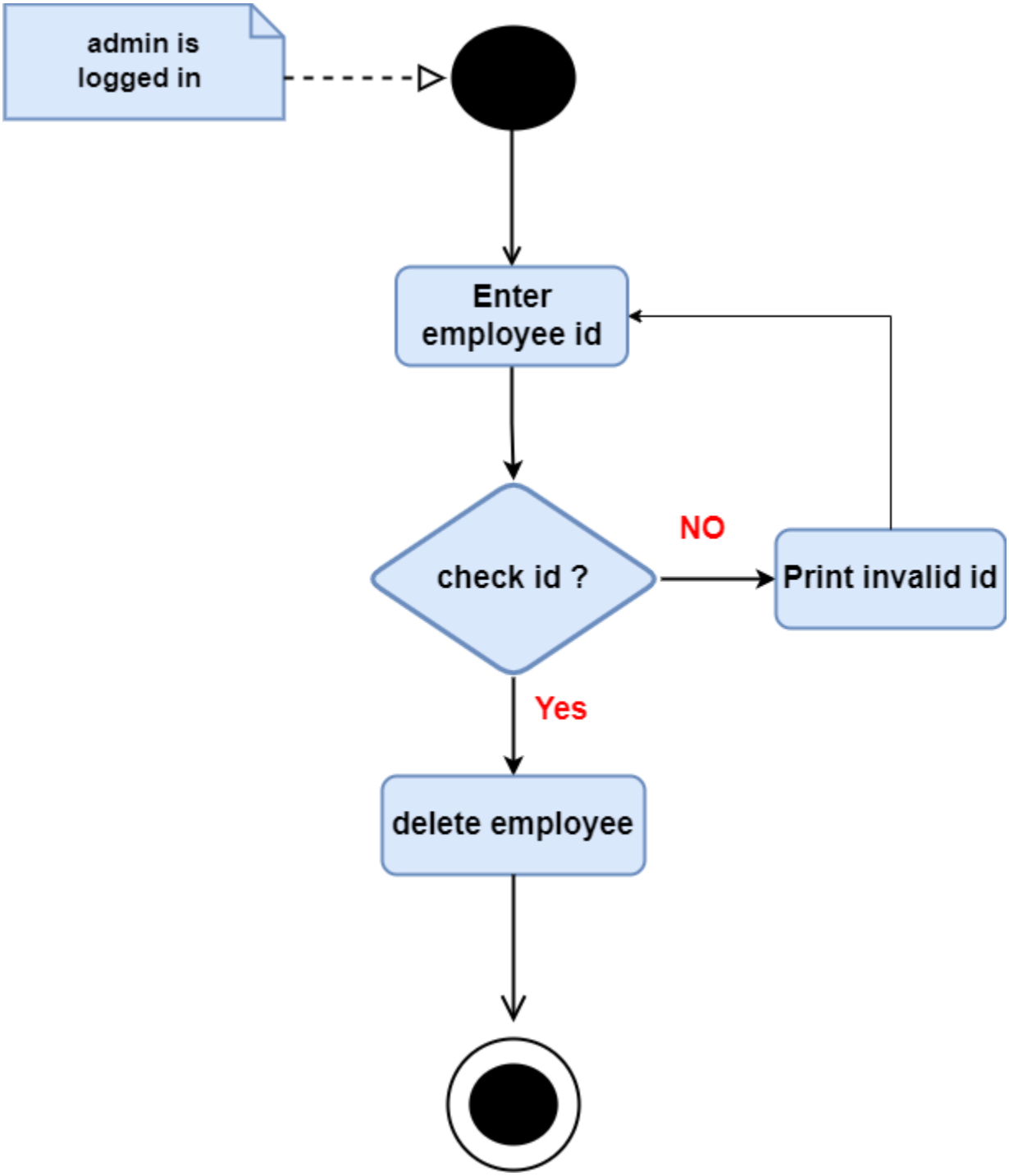
➢ Edit customer details



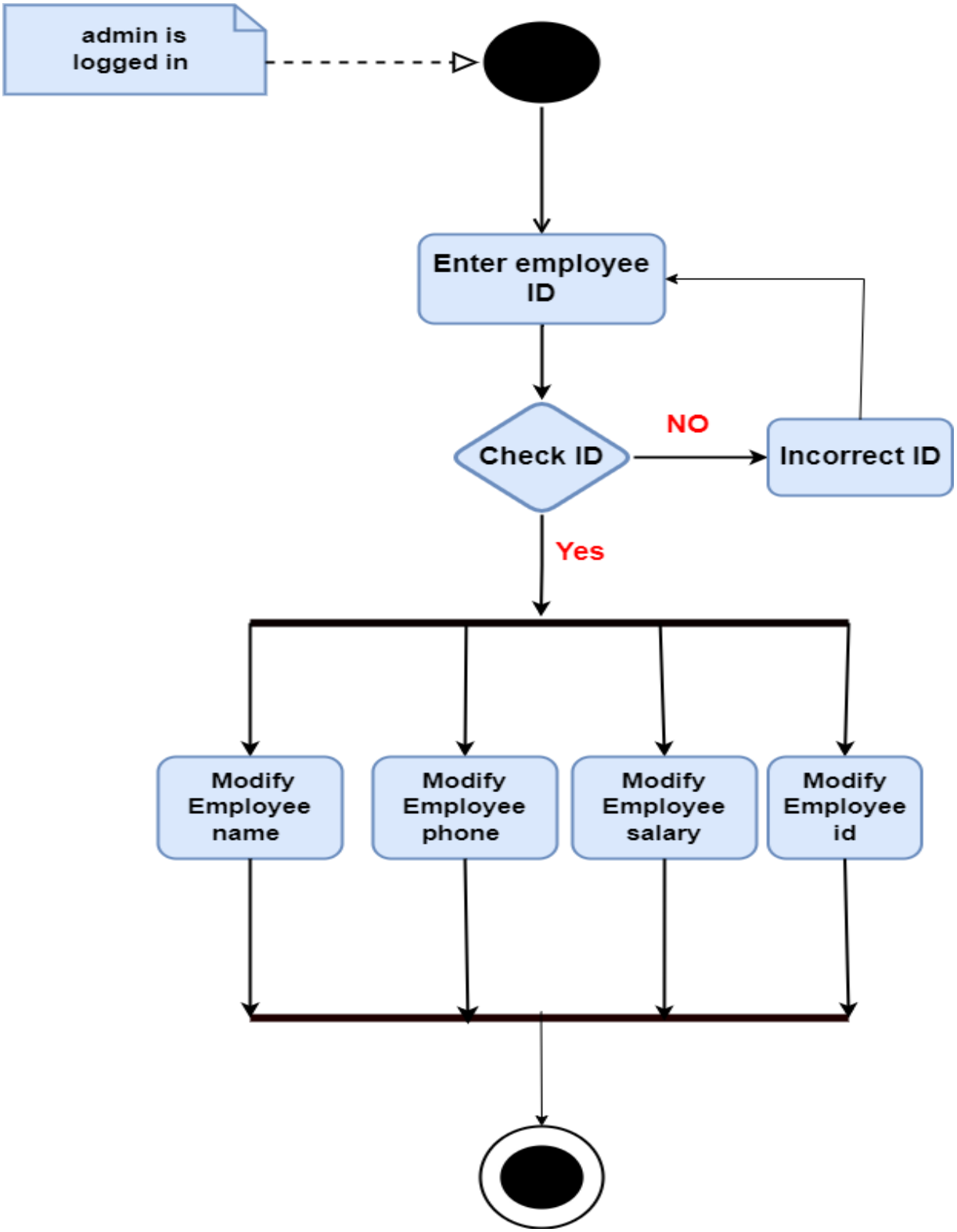
➢ Add employee



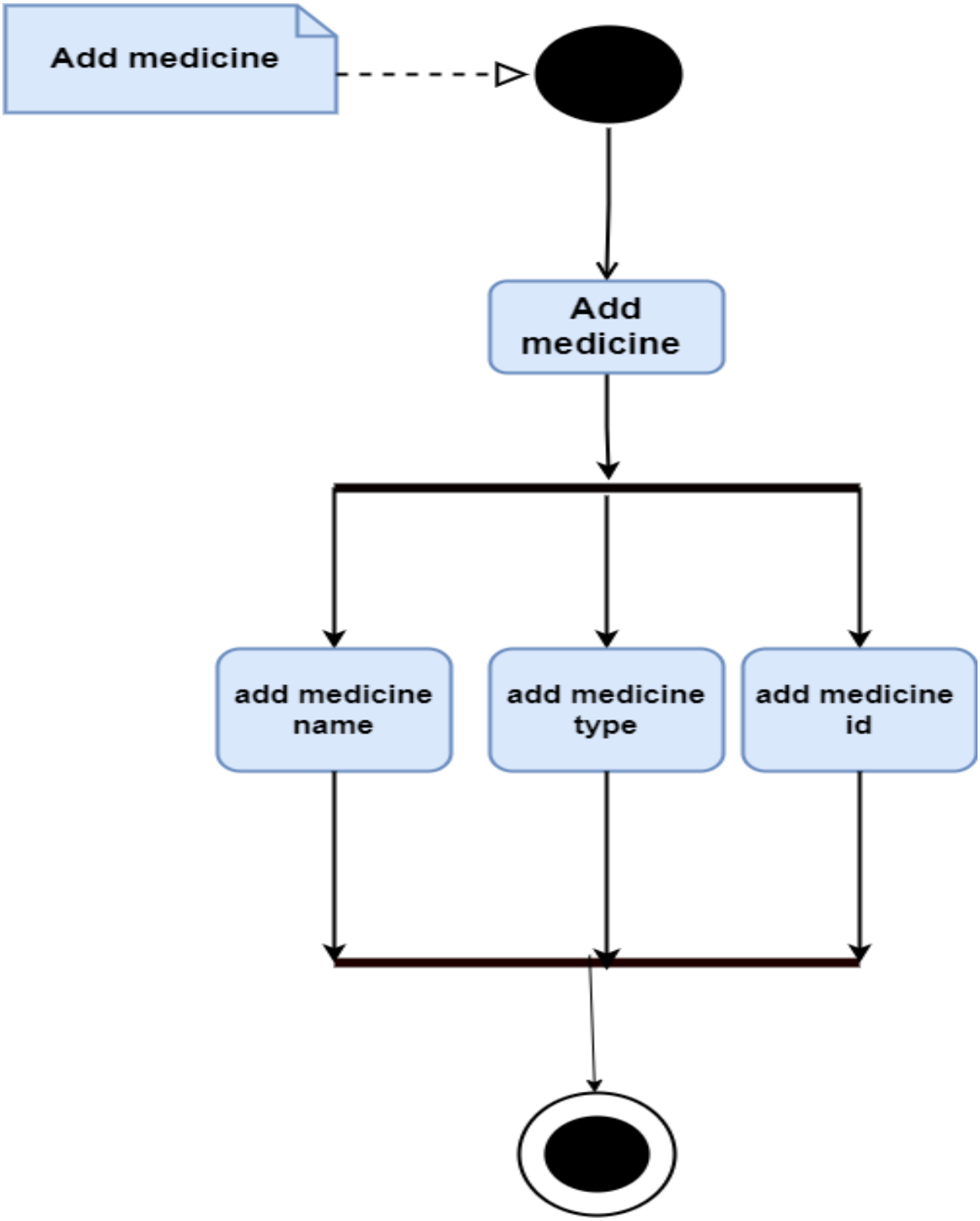
➢ Remove employee



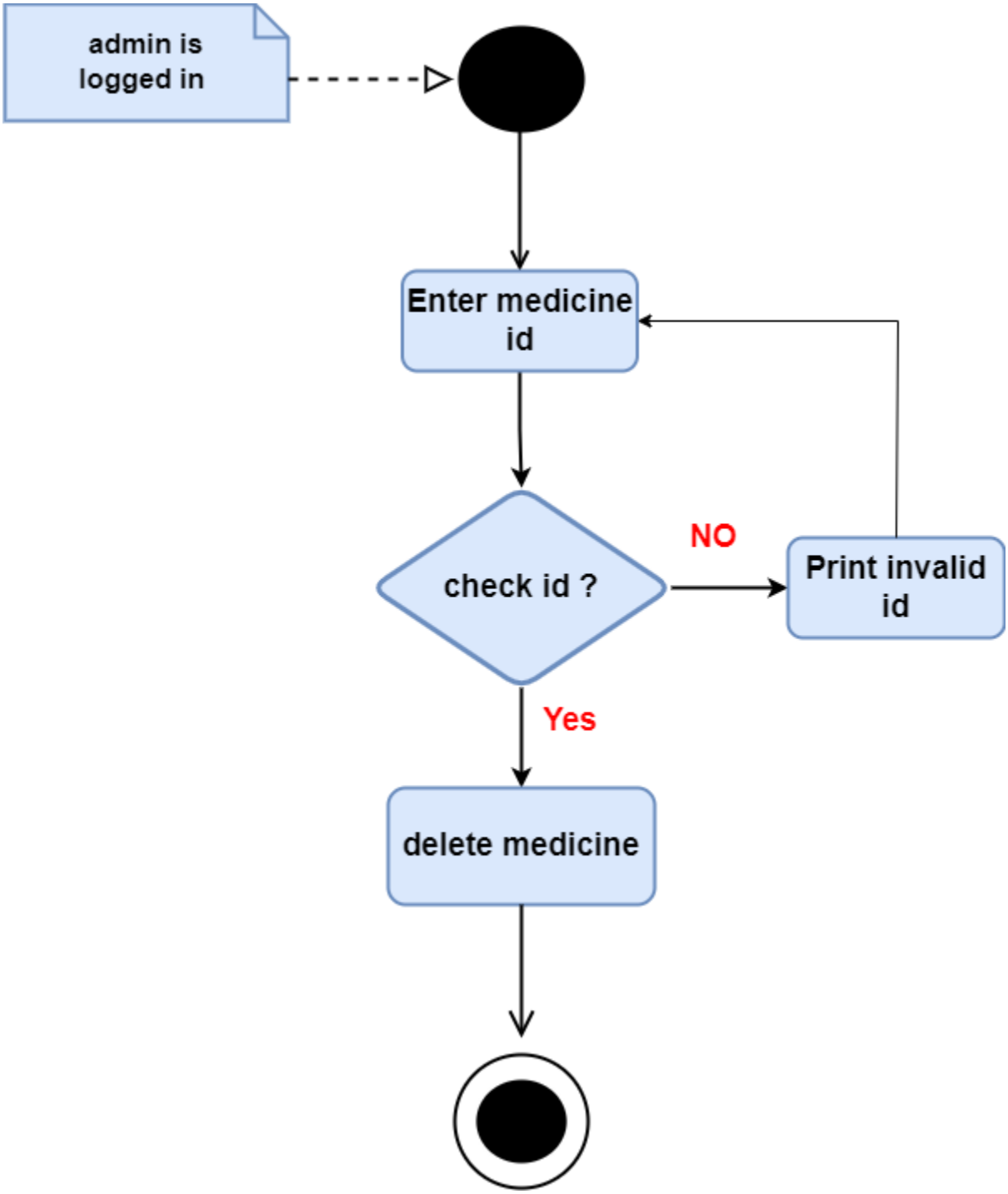
➢ Edit employee details



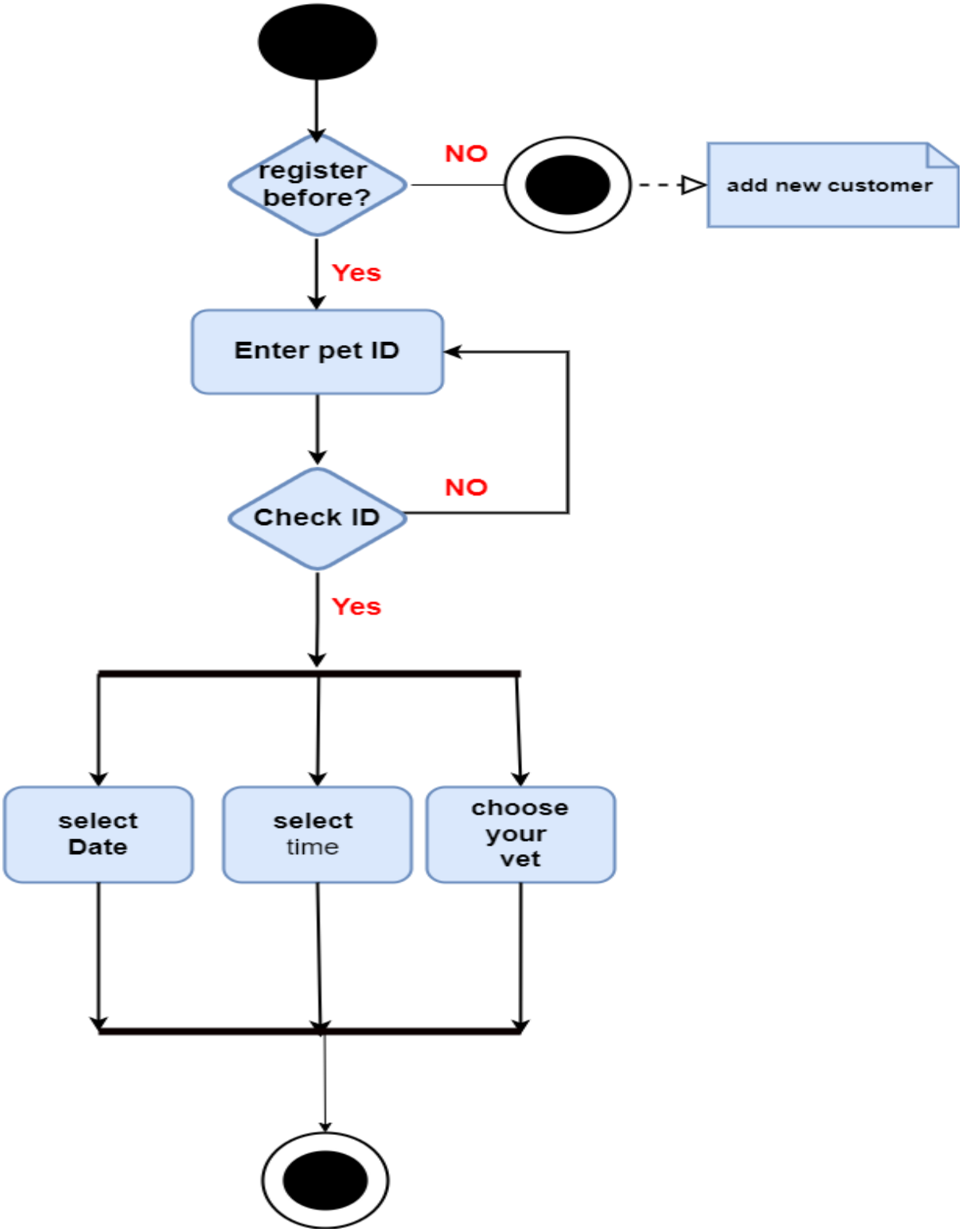
➢ Add medicine



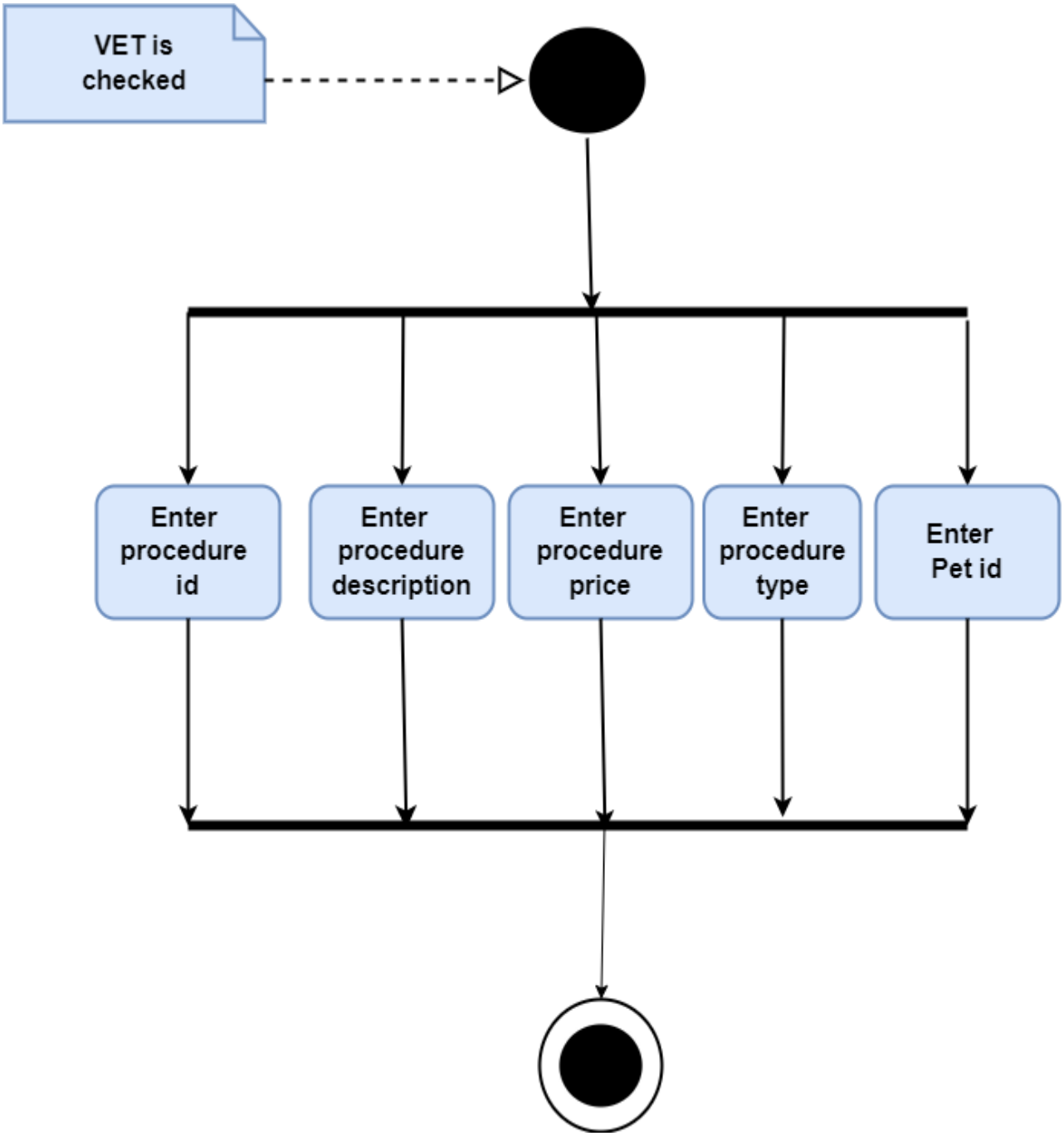
➢ Remove medicine



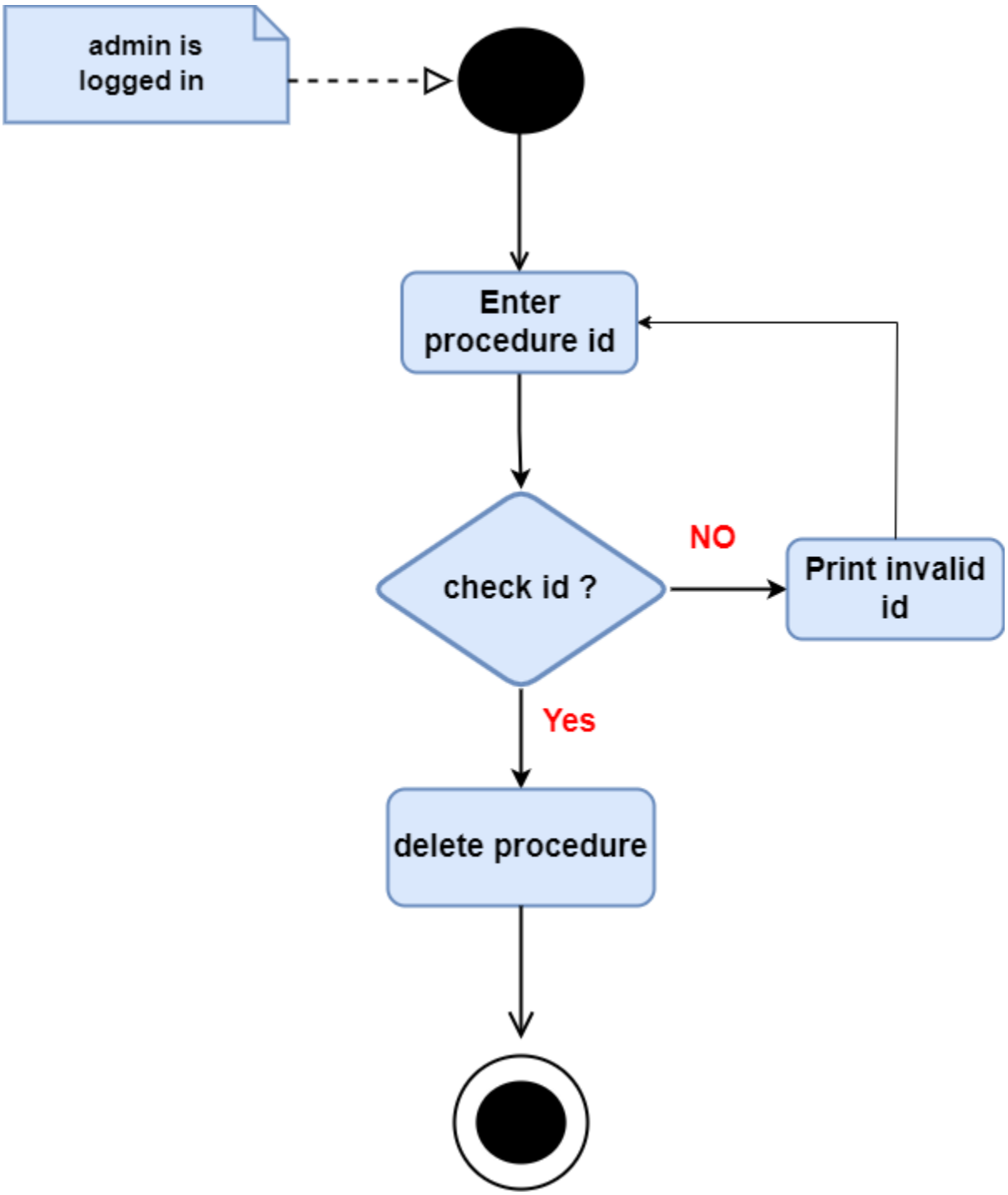
➢ Book an appointment



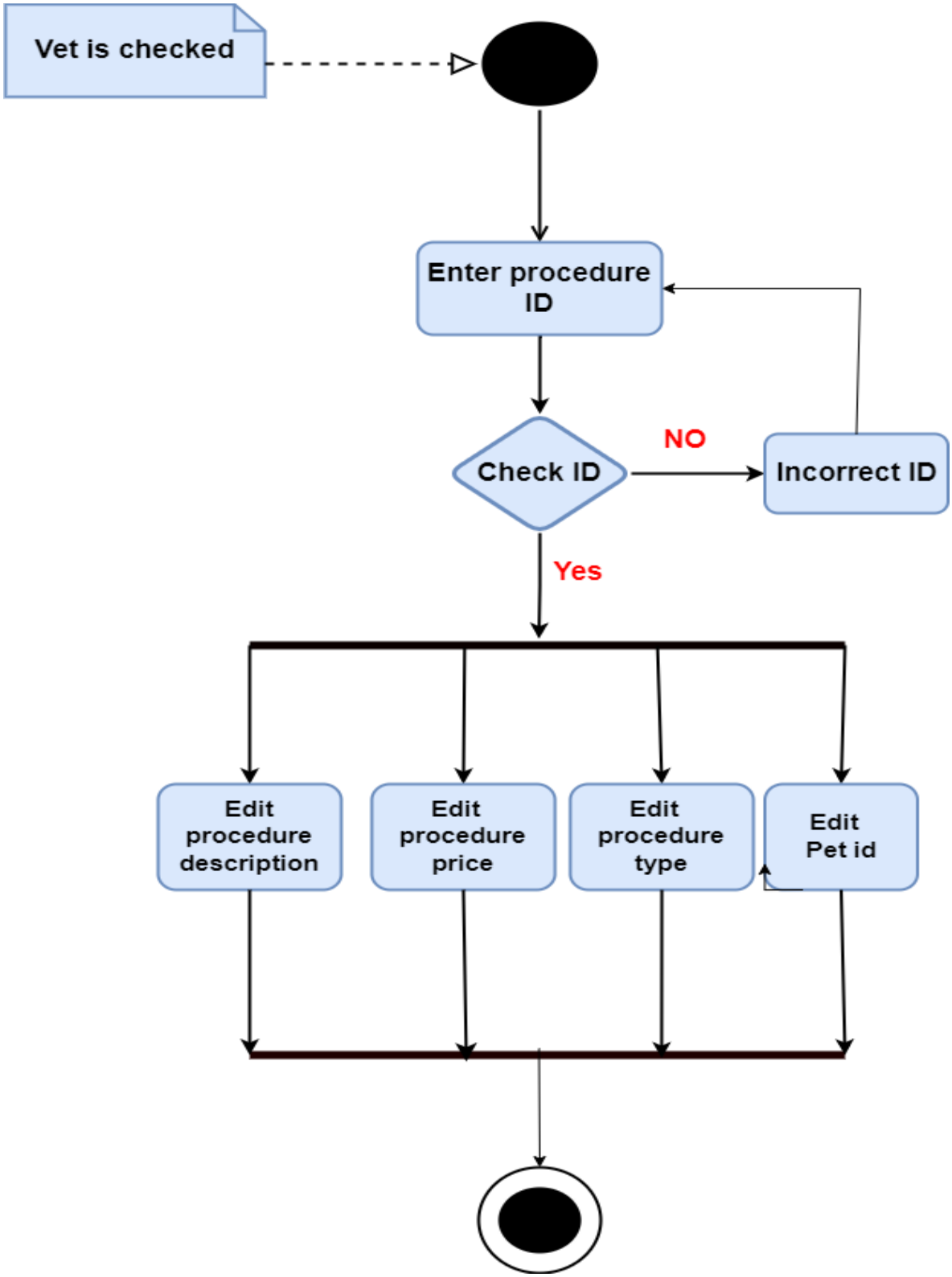
➢ Add procedure



➢ Remove procedure

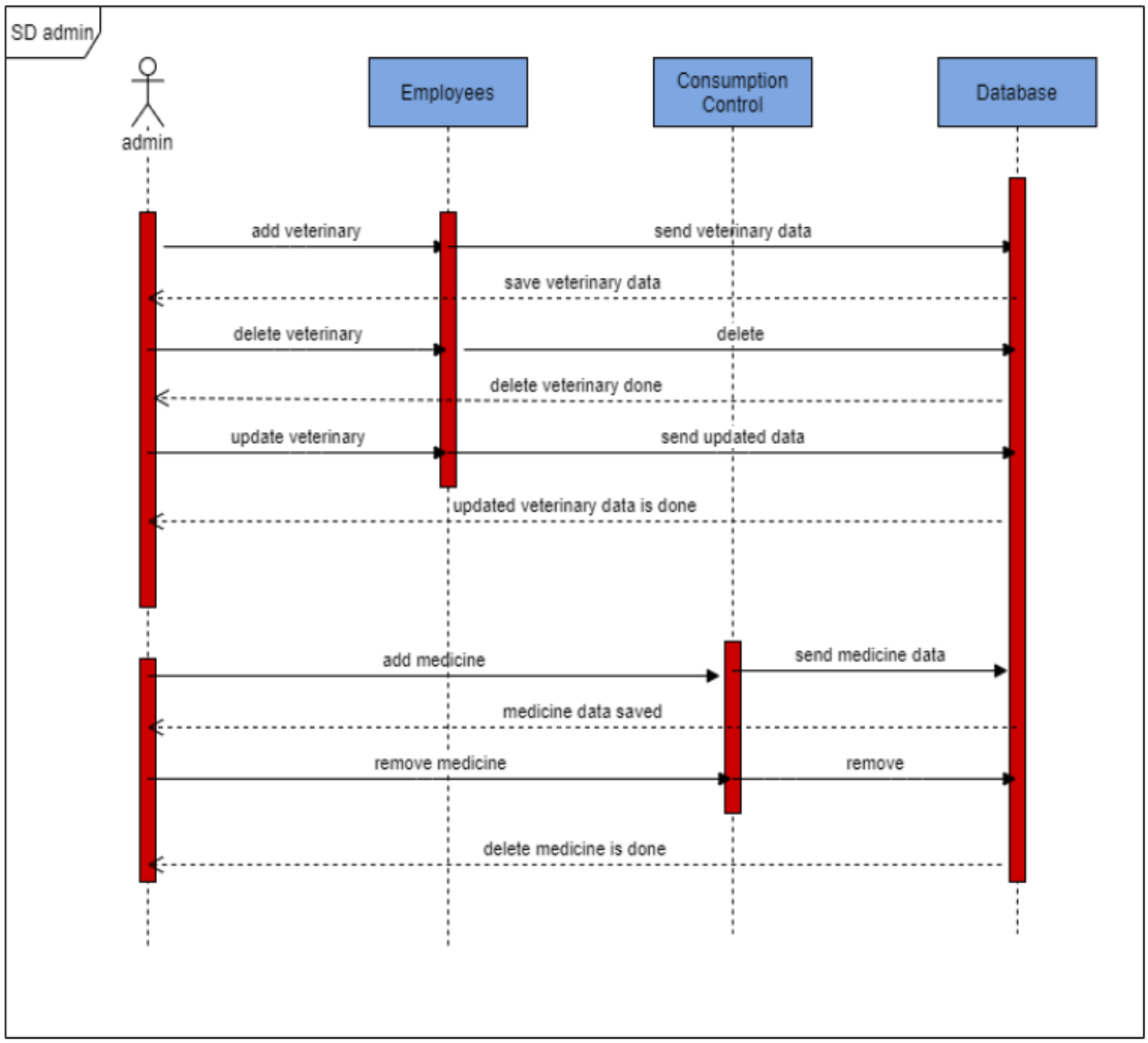


➢ Update procedure

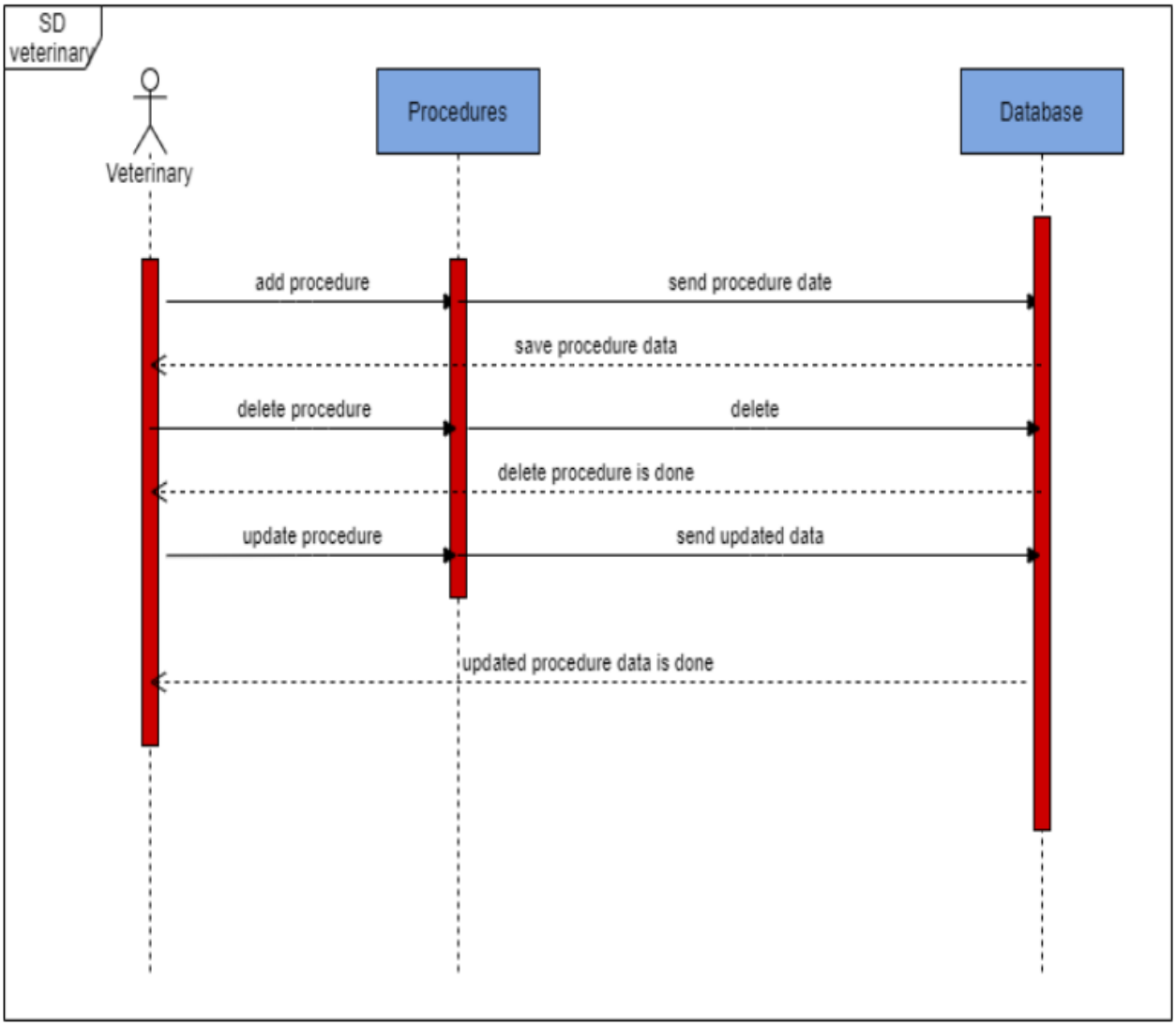


2.7 *Sequence diagram*

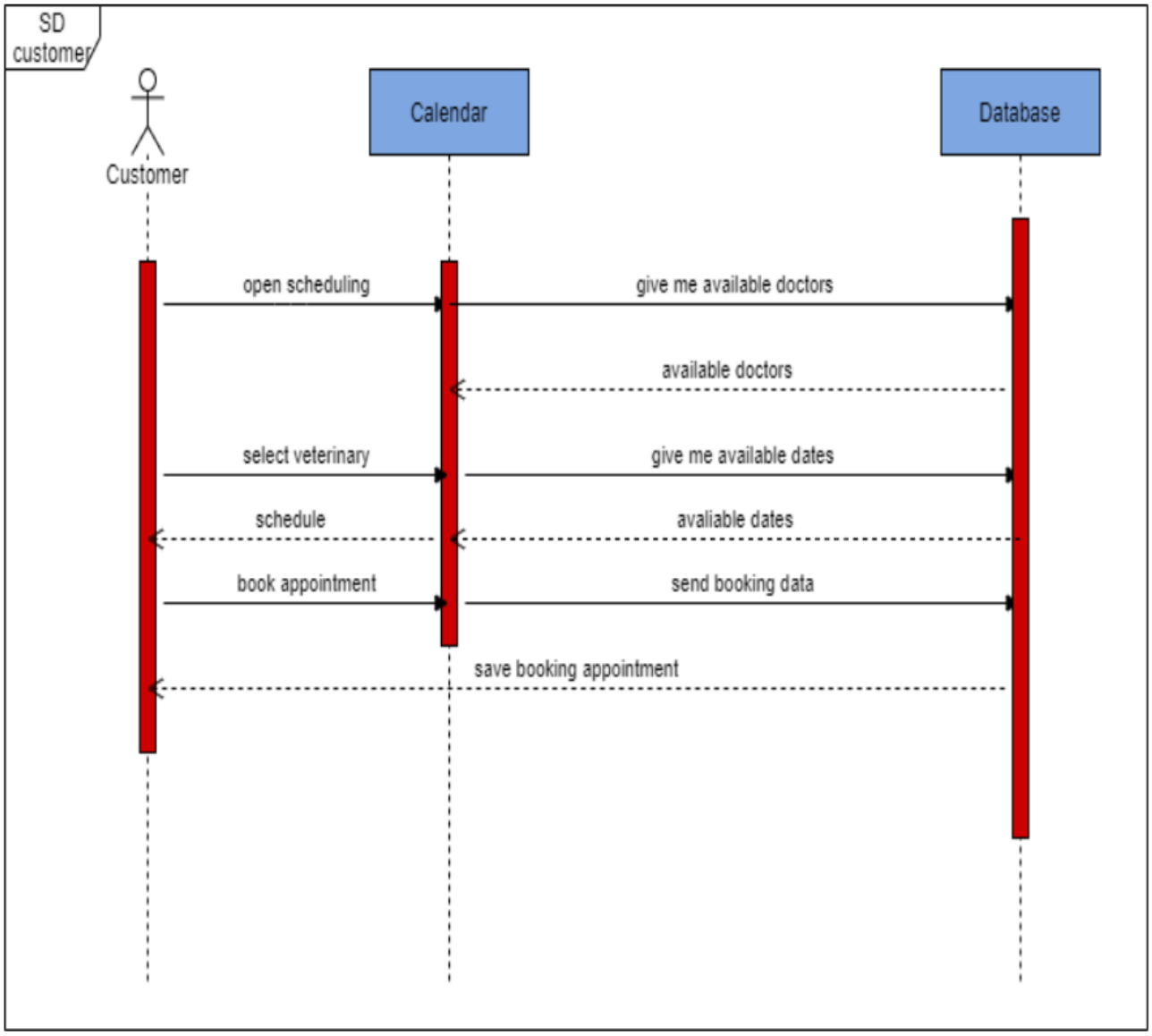
- **Admin sequence diagram**



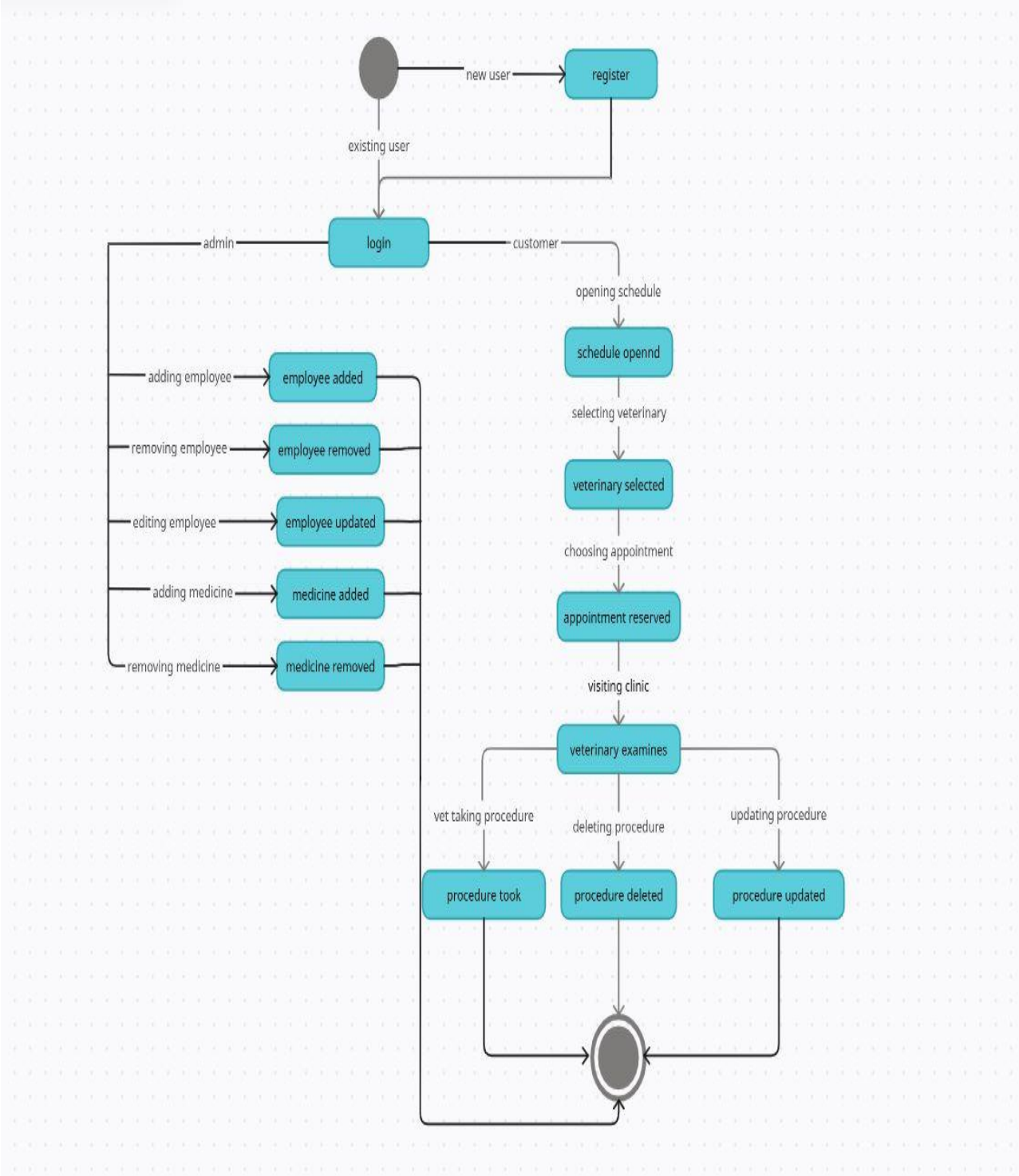
➢ Veterinary sequence diagram



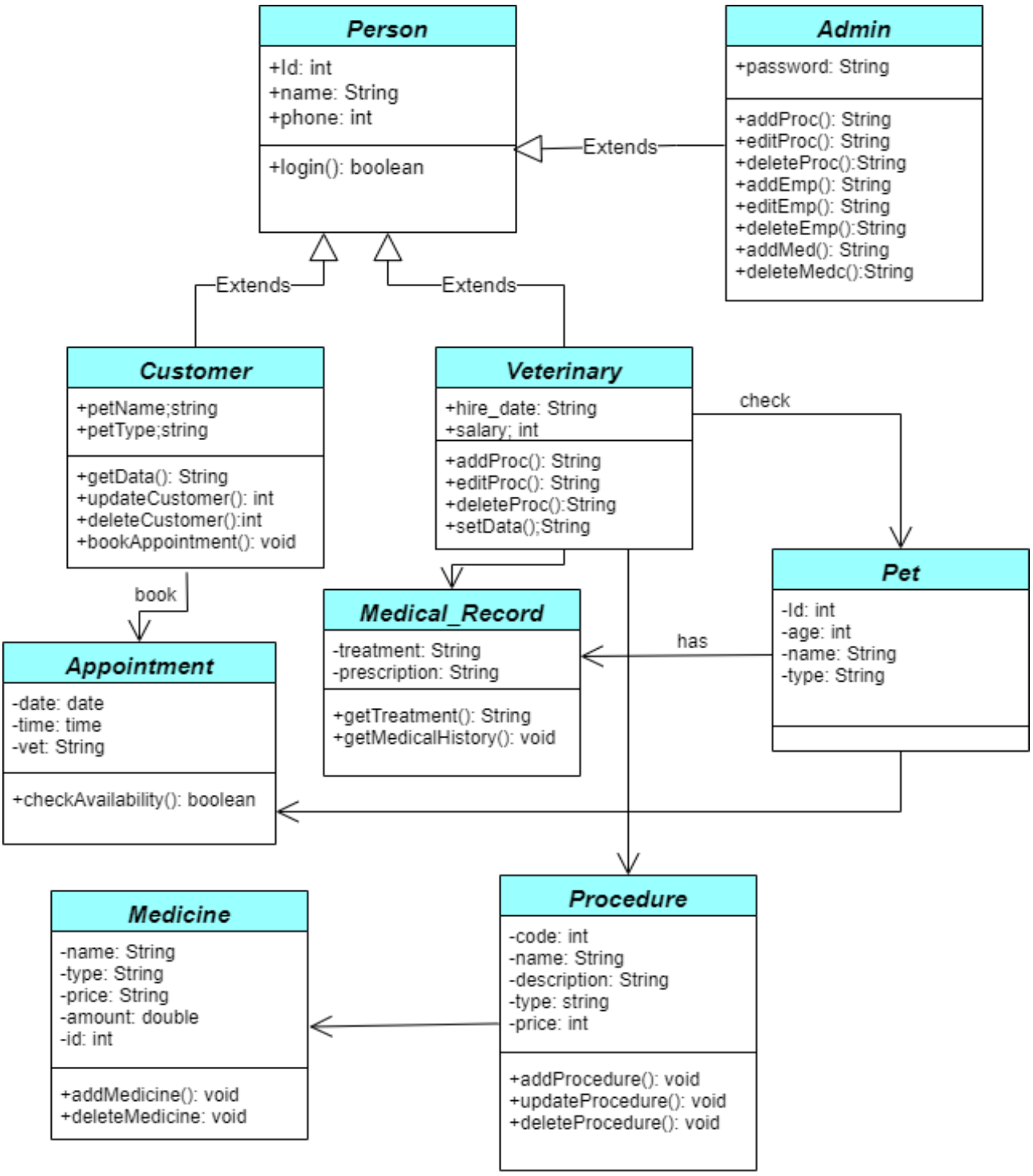
➢ Customer sequence diagram



2.8 State diagram

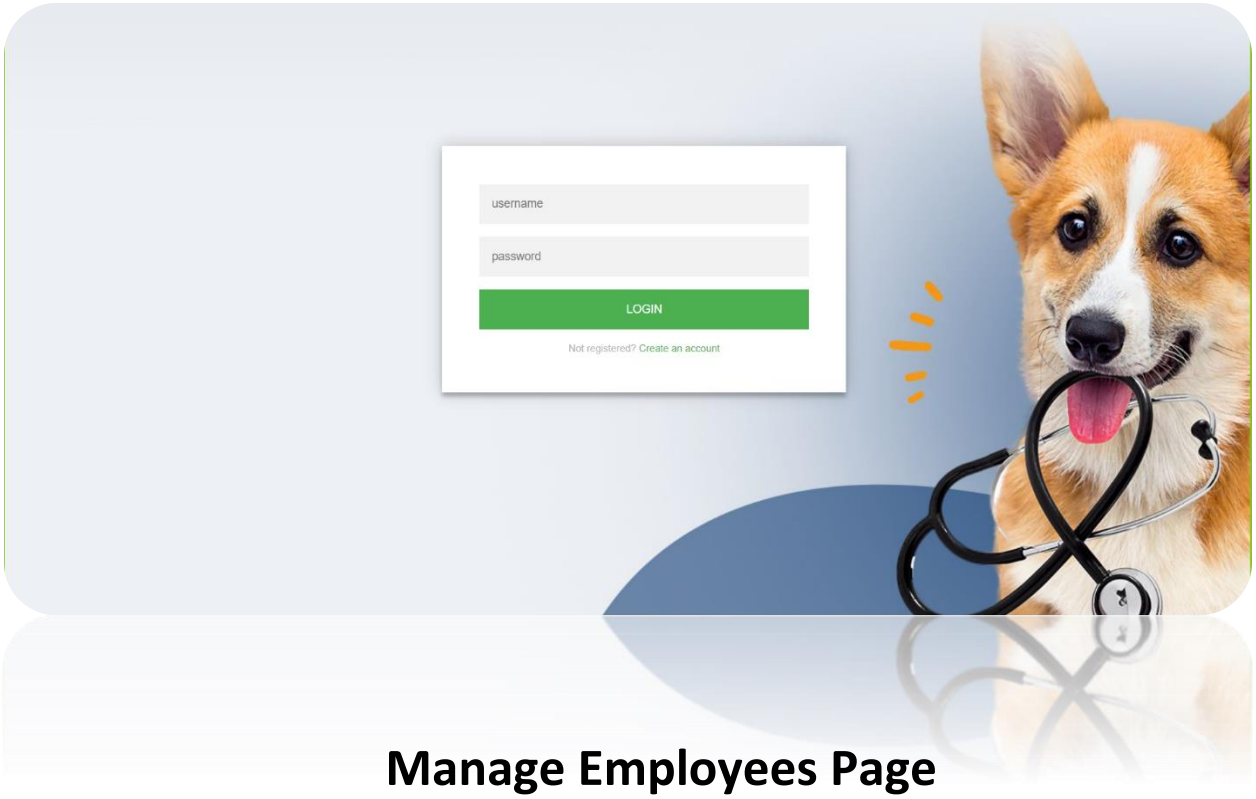


2.9 Class diagram



Ch3: GUI Design

Login Page



Manage Employees Page

Manage **Employees**

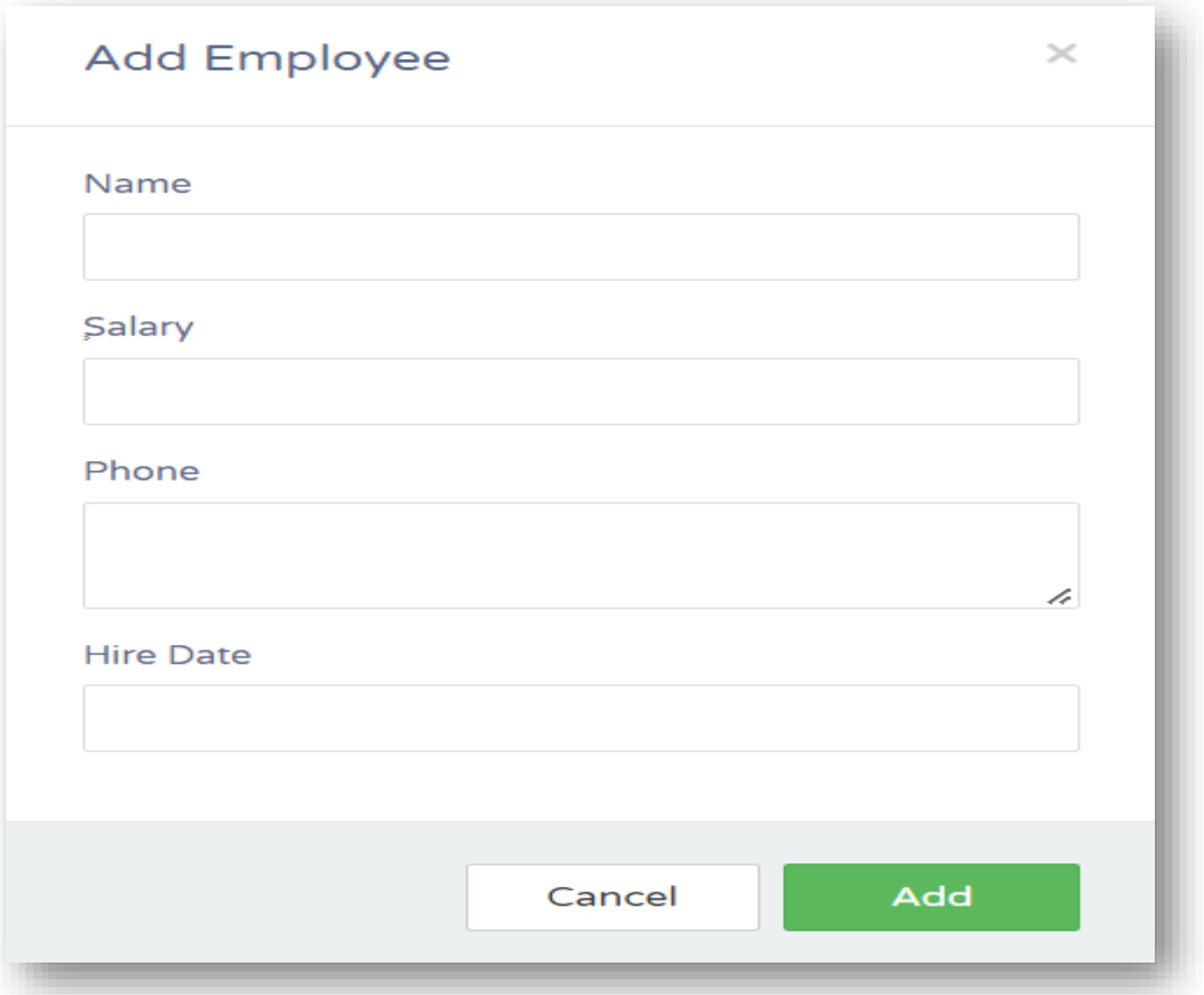
Delete | Add New Employee

ID	Name	Salary	Phone	Hire Date	Action
1	Fatma Ebrahim	12000	01148584587	3/1/2022	
2	Taha Ebrahim	14000	01148584587	8/2/2020	
3	Abdulmoneim Hamed	15000	01148584587	1/4/2022	
4	Amr Awny	9500	01148584587	13/8/2021	
5	Abdulrahman Abdelhady	10000	01148478166	4/7/2022	

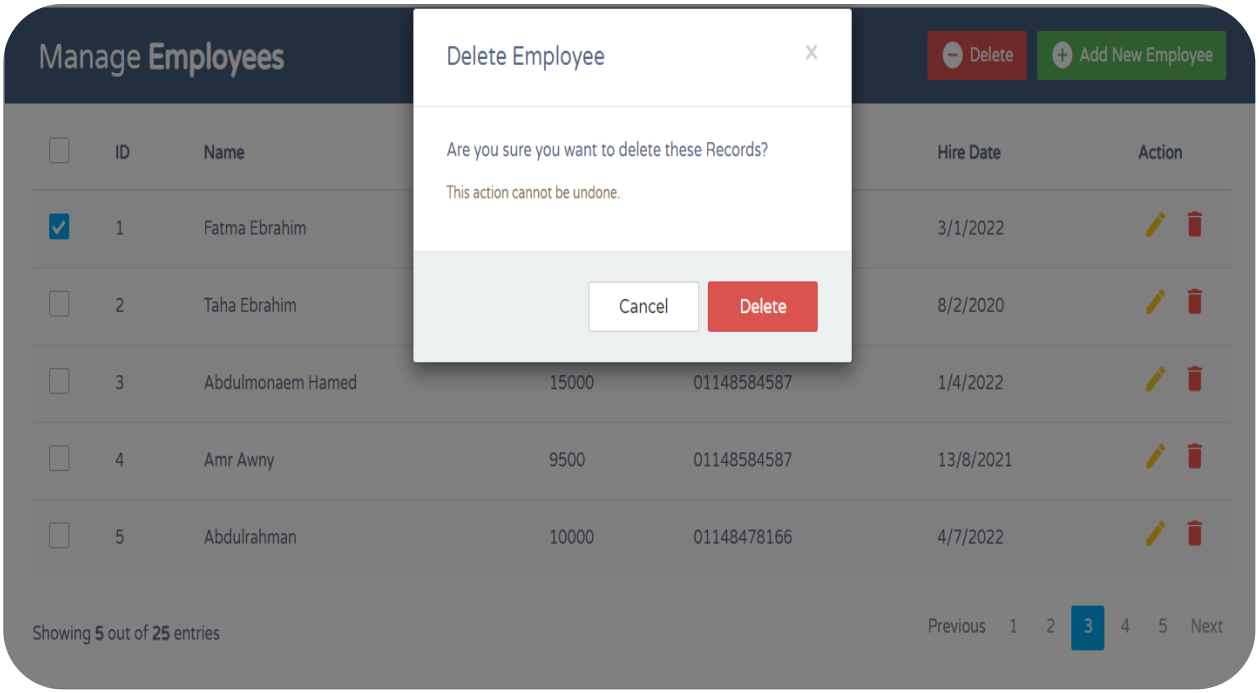
Showing **5** out of **25** entries

Previous 1 2 3 4 5 Next

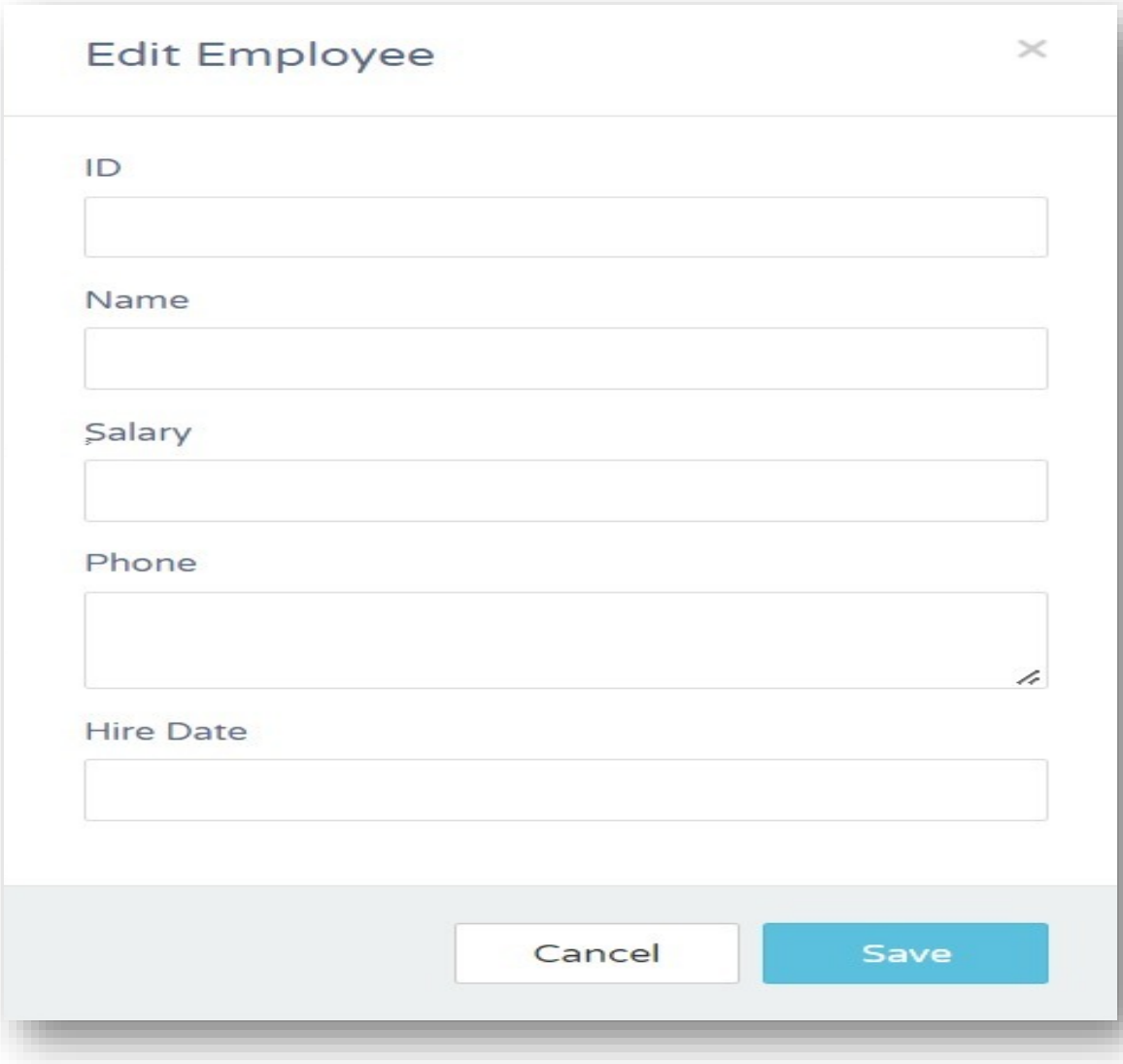
Add Employee Page



Remove Employee Page

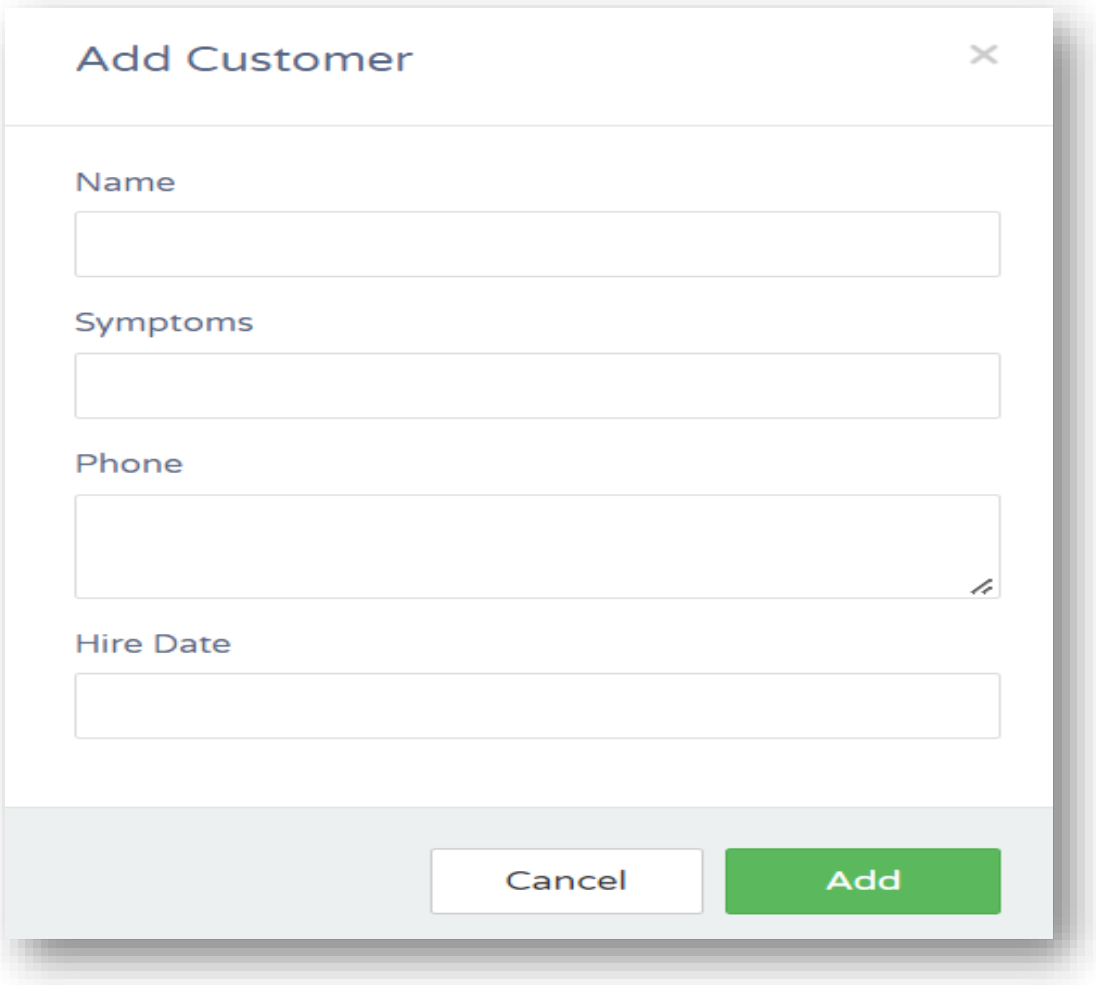


Edit Employee Page

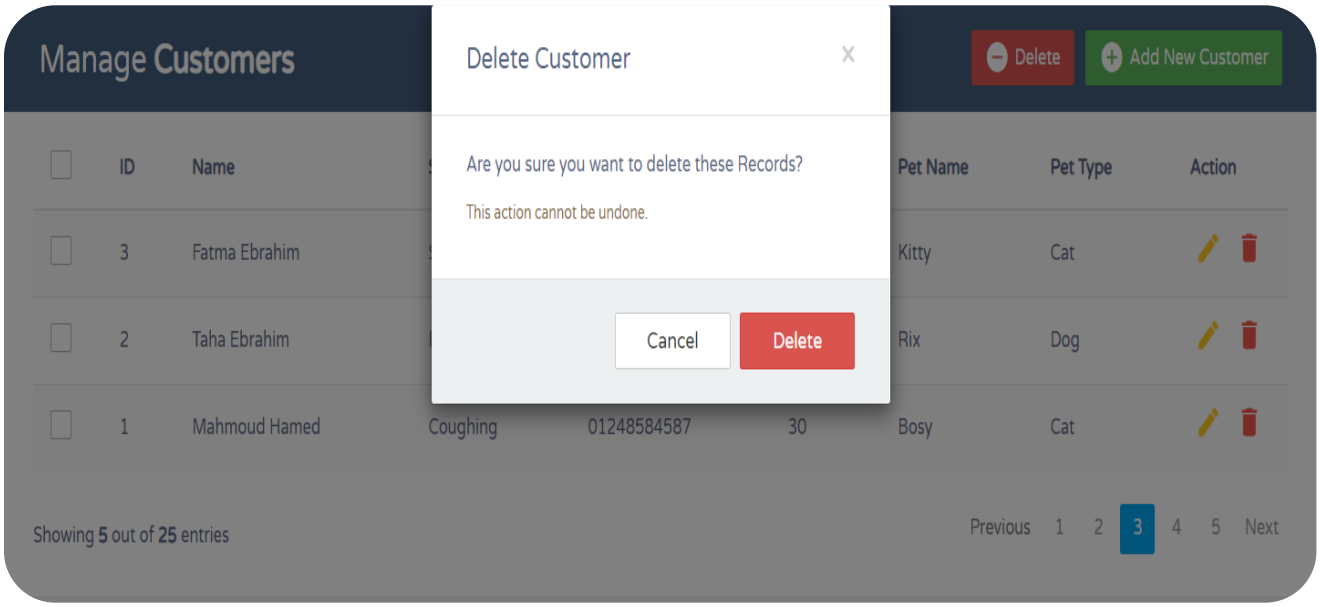


Manage customer Page

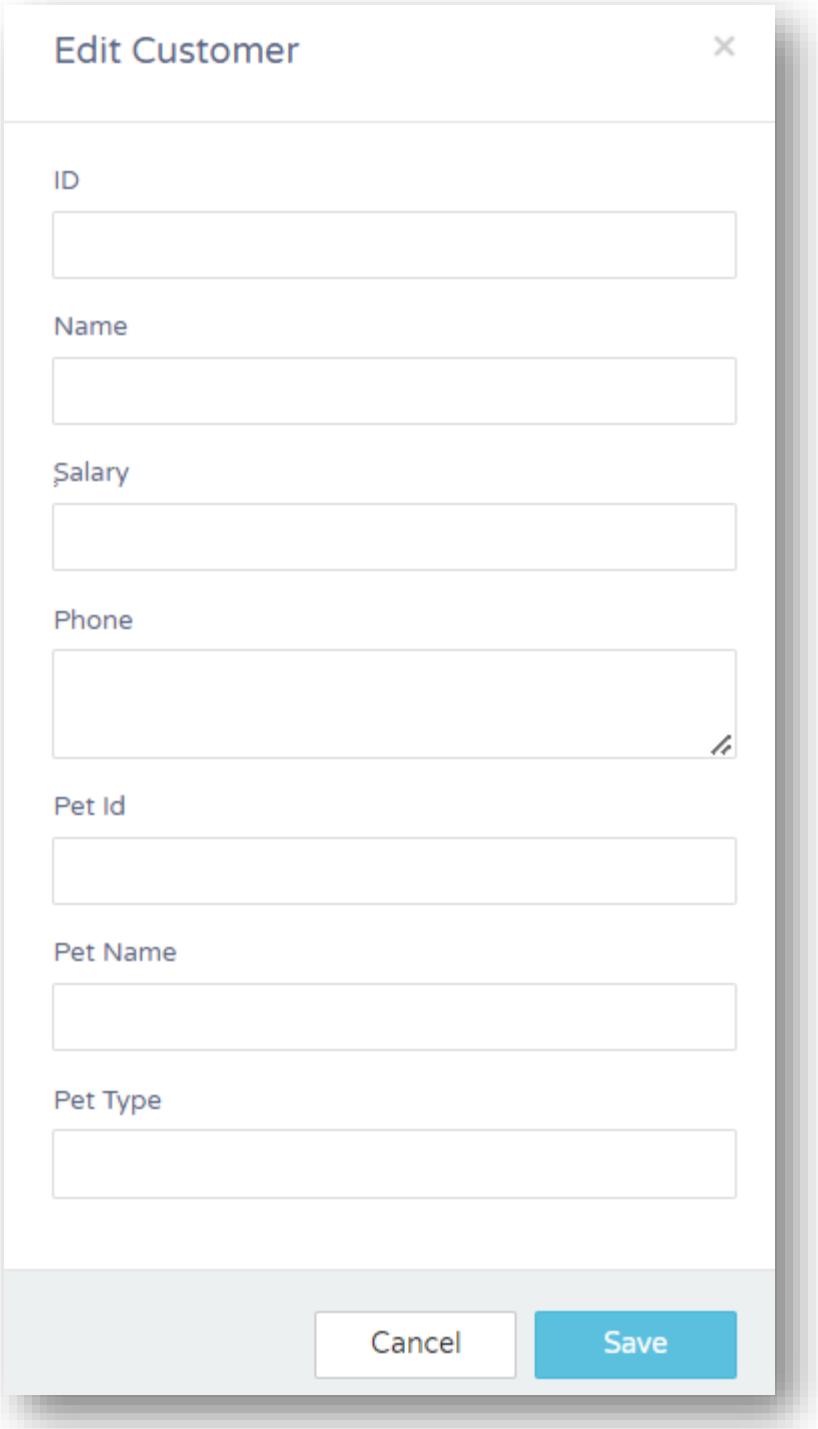
Add customer Page



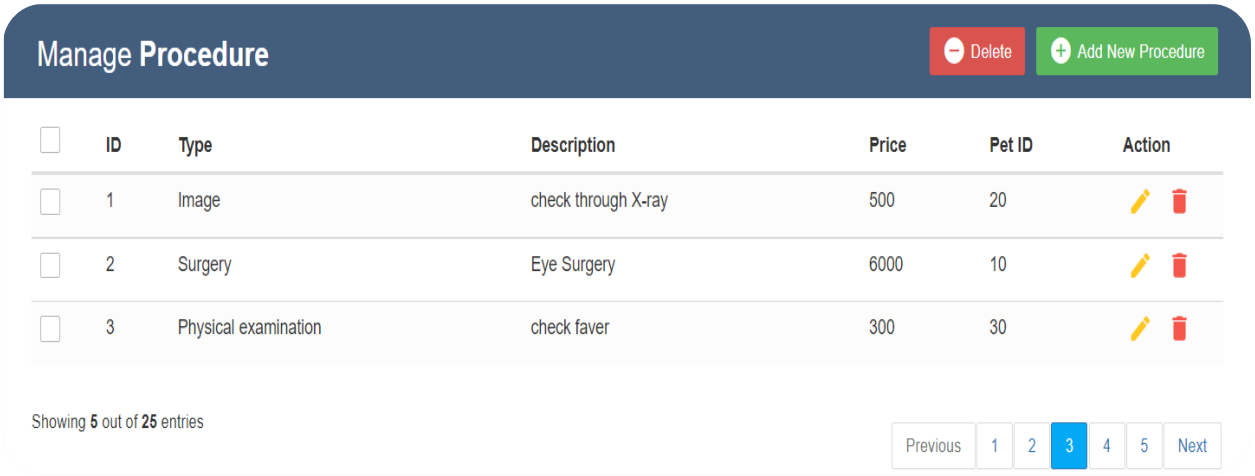
Delete customer Page



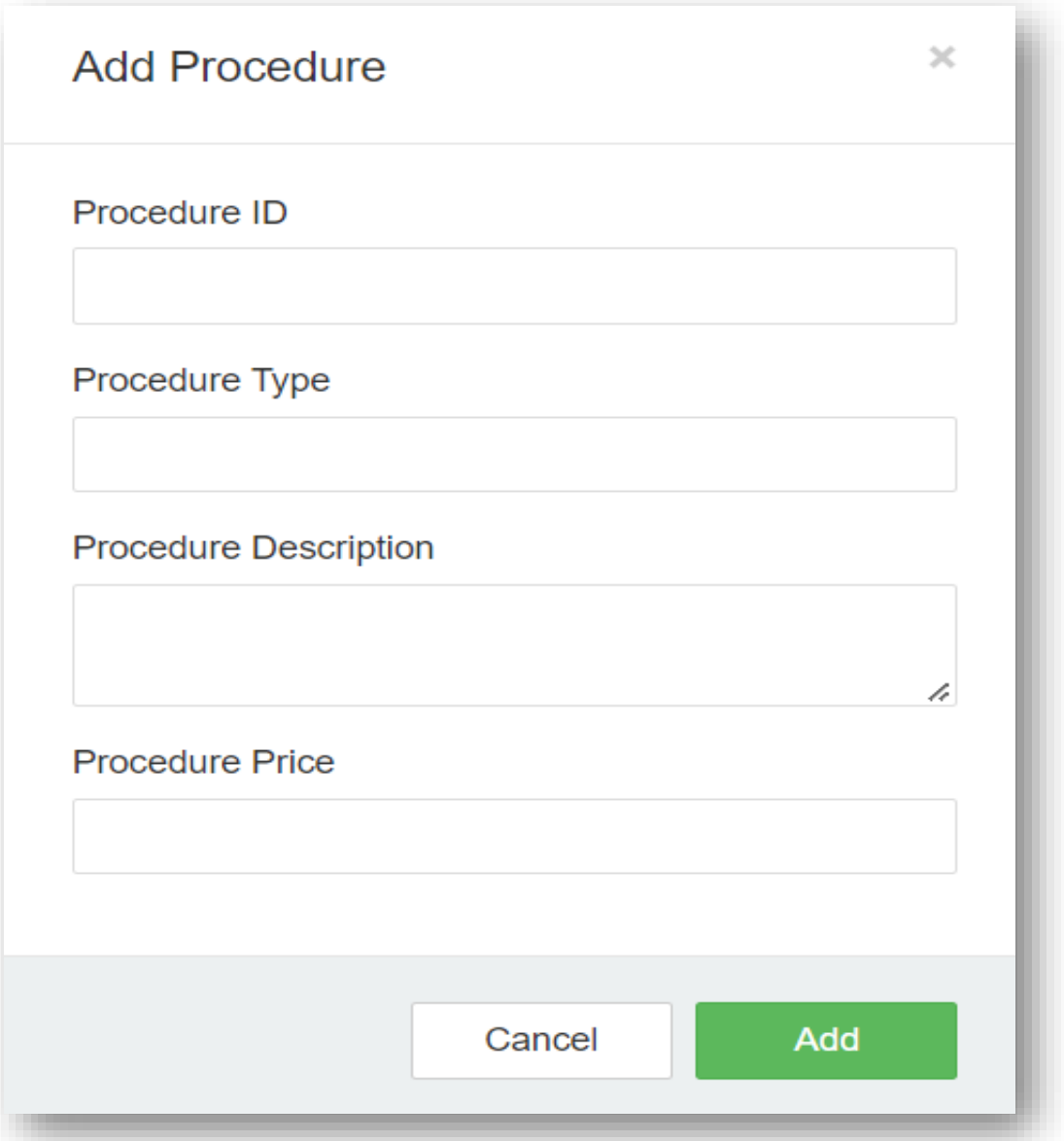
Edit customer Page



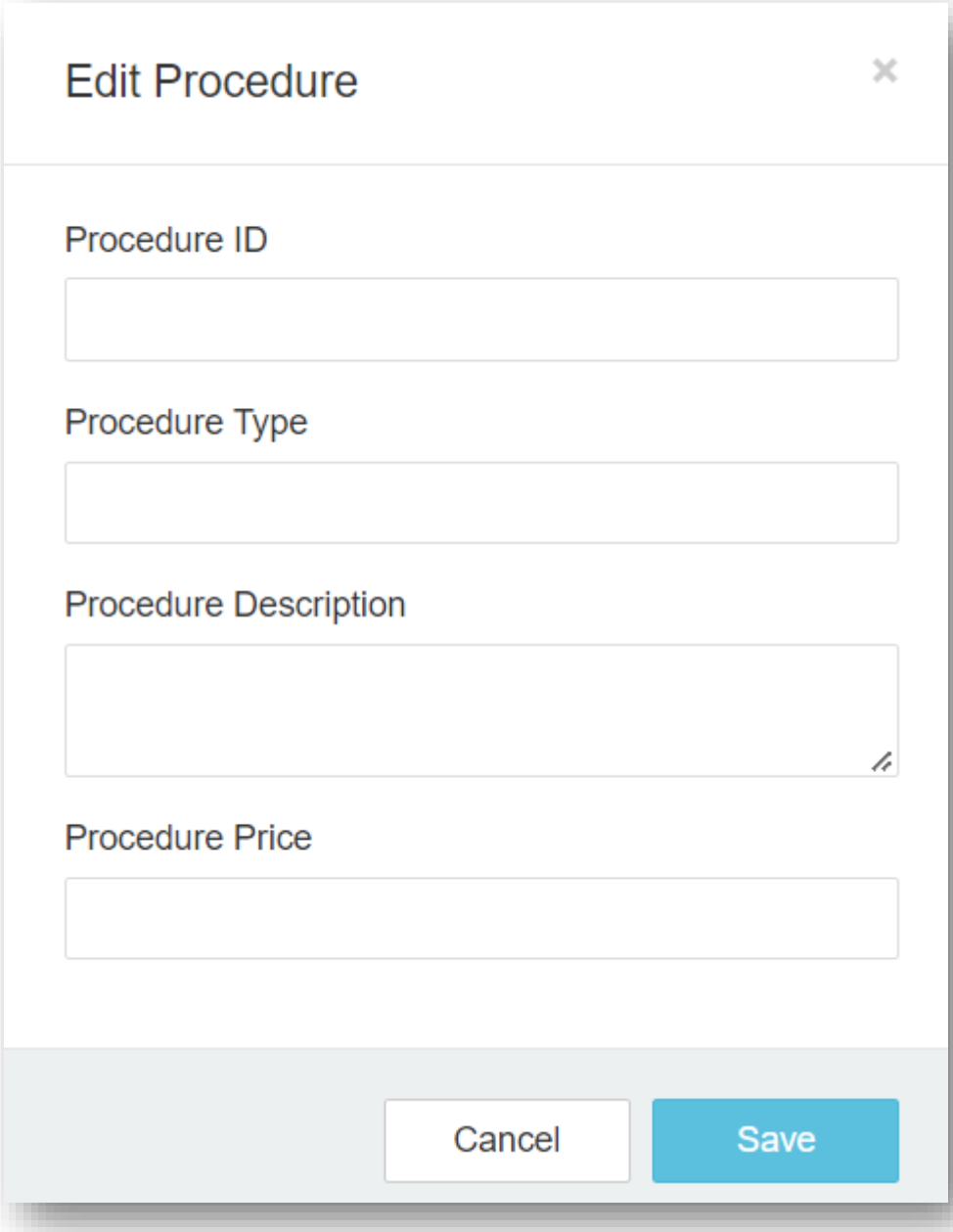
Manage procedure Page



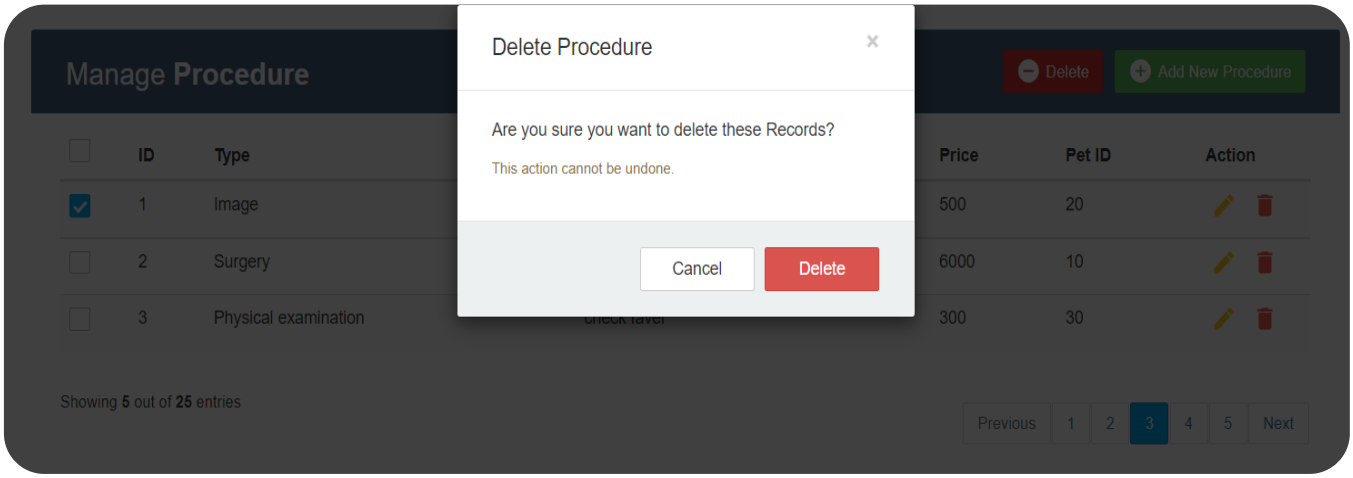
Add procedure Page

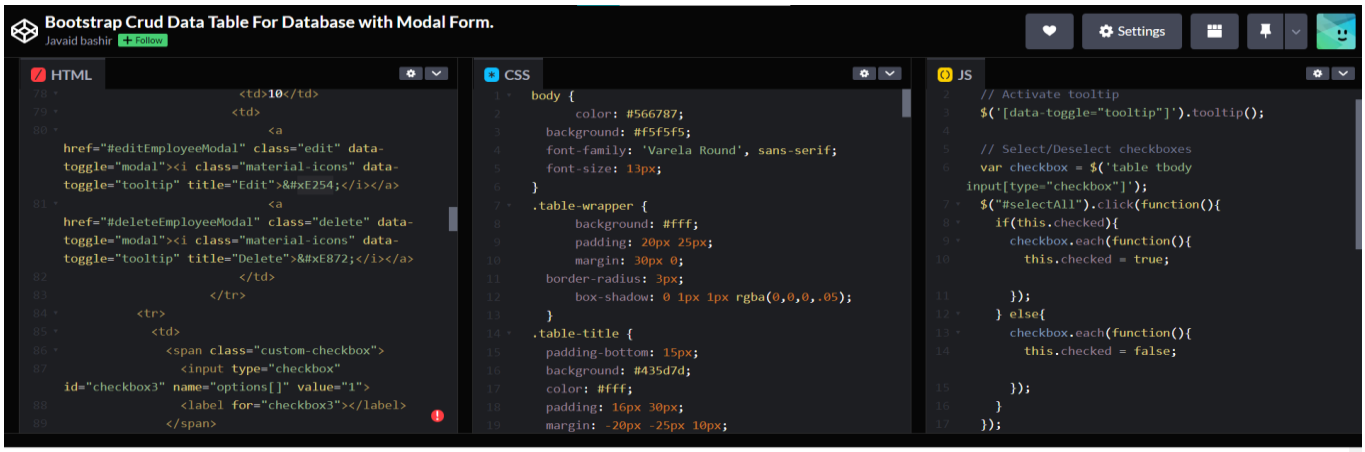


Edit procedure Page



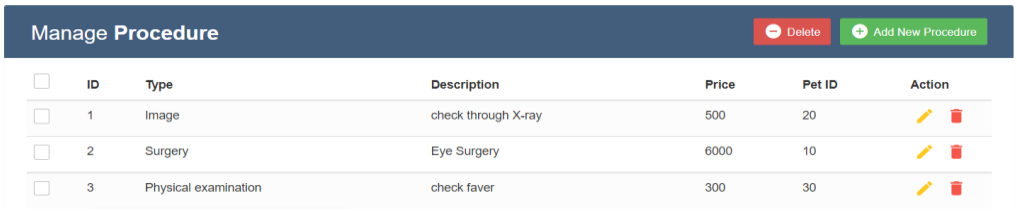
Delete procedure Page

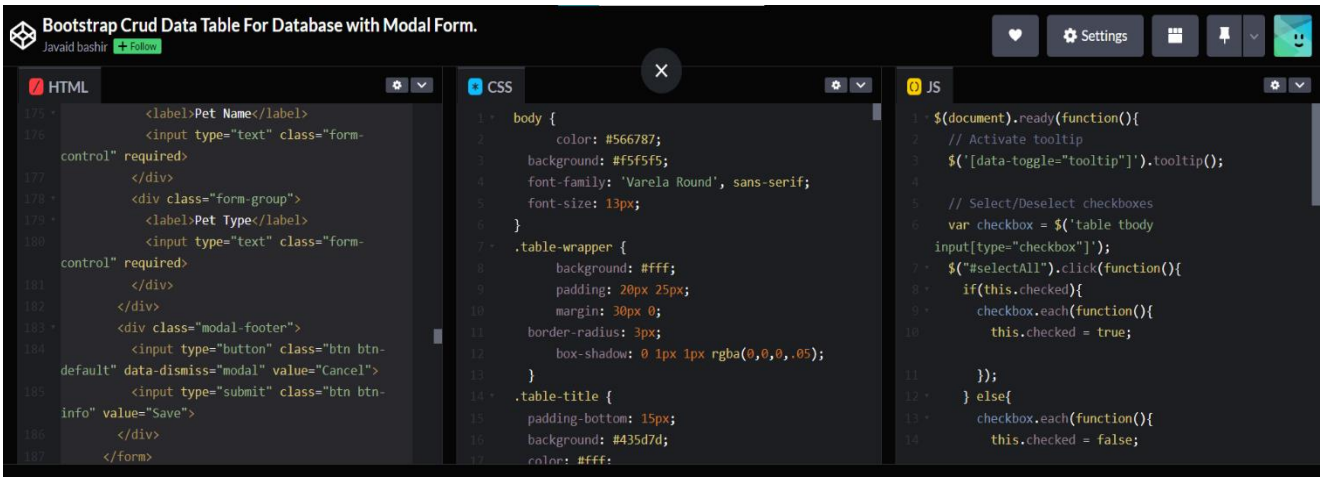



Bootstrap Crud Data Table For Database with Modal Form.
Javaid bashir
Follow
Settings
HTML
CSS
JS
<td>10</td>
<td>


</td>
</tr>
<tr>
<td>

<input type="checkbox" id="checkbox3" name="options[]" value="1">
<label for="checkbox3"></label>

body {
color: #566787;
background: #f5f5f5;
font-family: 'Varela Round', sans-serif;
font-size: 13px;
}
.table-wrapper {
background: #fff;
padding: 20px 25px;
margin: 30px 0;
border-radius: 3px;
box-shadow: 0 1px 1px rgba(0,0,0,.05);
}
.table-title {
padding-bottom: 15px;
background: #435d7d;
color: #fff;
padding: 16px 30px;
margin: -20px -25px 10px;
// Activate tooltip
$('[data-toggle="tooltip"]').tooltip();
// Select/Deselect checkboxes
var checkbox = $('table tbody input[type="checkbox"]');
$("#selectAll").click(function(){
if(this.checked){
checkbox.each(function(){
this.checked = true;
});
} else{
checkbox.each(function(){
this.checked = false;
});
}
});

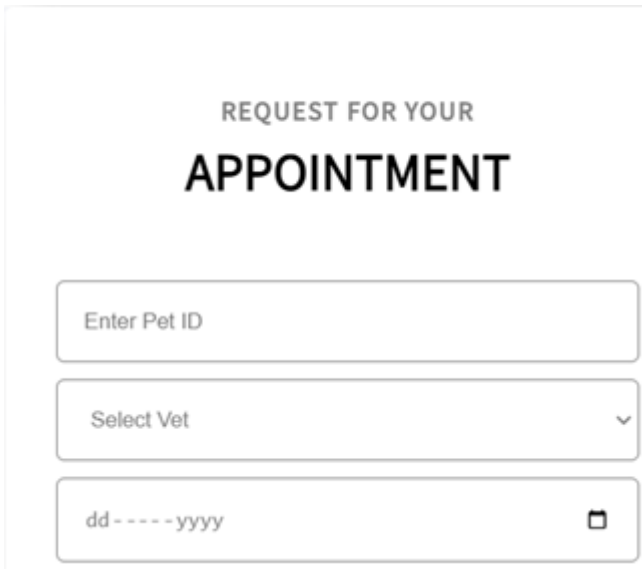

Manage Procedure
Delete
Add New Procedure
ID
Type
Description
Price
Pet ID
Action
1
Image
check through X-ray
500
20
2
Surgery
Eye Surgery
6000
10
3
Physical examination
check faver
300
30

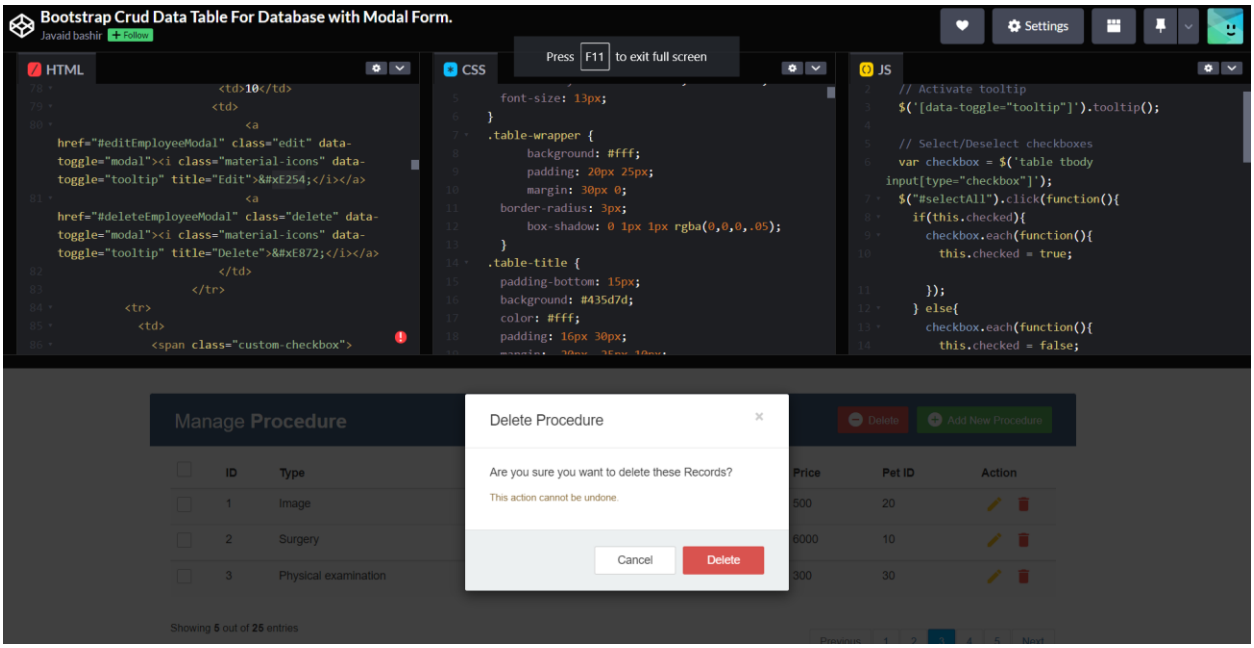

Bootstrap Crud Data Table For Database with Modal Form.
Javaid bashir
Follow
Settings
HTML
CSS
JS
<label>Pet Name</label>
<input type="text" class="form-control" required>

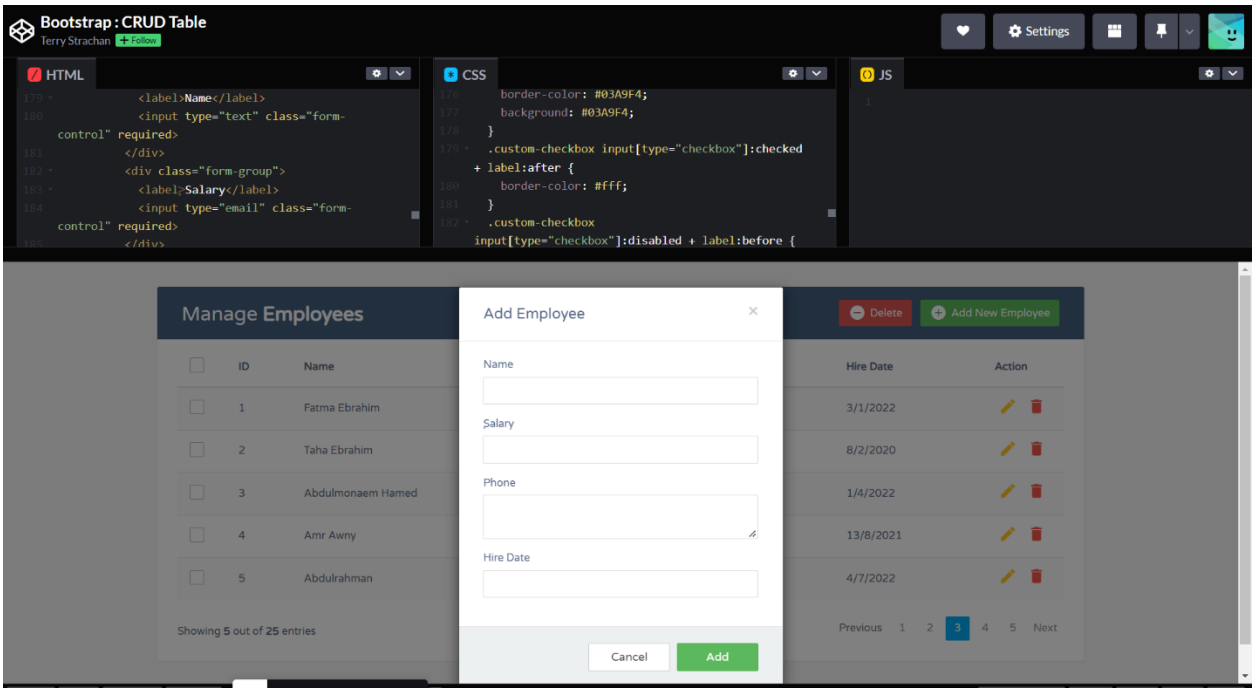
<label>Pet Type</label>
<input type="text" class="form-control" required>

<input type="button" class="btn btn-default" data-dismiss="modal" value="Cancel">
<input type="submit" class="btn btn-info" value="Save">

</form>
body {
color: #566787;
background: #f5f5f5;
font-family: 'Varela Round', sans-serif;
font-size: 13px;
}
.table-wrapper {
background: #fff;
padding: 20px 25px;
margin: 30px 0;
border-radius: 3px;
box-shadow: 0 1px 1px rgba(0,0,0,.05);
}
.table-title {
padding-bottom: 15px;
background: #435d7d;
$(document).ready(function(){
// Activate tooltip
$('[data-toggle="tooltip"]').tooltip();
// Select/Deselect checkboxes
var checkbox = $('table tbody input[type="checkbox"]');
$("#selectAll").click(function(){
if(this.checked){
checkbox.each(function(){
this.checked = true;
});
} else{
checkbox.each(function(){
this.checked = false;


REQUEST FOR YOUR
APPOINTMENT
Enter Pet ID
Select Vet
dd - - - - - yyyy


Bootstrap Crud Data Table For Database with Modal Form.
Javaid bashir
Follow
Settings
Press F11 to exit full screen
HTML
CSS
JS
Manage Procedure
Delete Procedure
Are you sure you want to delete these Records?
This action cannot be undone.
Cancel
Delete
Add New Procedure
ID
Type
Price
Pet ID
Action
1
Image
500
20
2
Surgery
6000
10
3
Physical examination
300
30
Showing 5 out of 25 entries


Bootstrap : CRUD Table
Terry Strachan
Follow
Settings
HTML
CSS
JS
Manage Employees
Add Employee
Name
Salary
Phone
Hire Date
Cancel
Add
Delete
Add New Employee
ID
Name
Hire Date
Action
1
Fatma Ebrahim
3/1/2022
2
Taha Ebrahim
8/2/2020
3
Abdulmonaem Hamed
1/4/2022
4
Amr Awny
13/8/2021
5
Abdulrahman
4/7/2022
Showing 5 out of 25 entries
Previous
1
2
3
4
5
Next