User Requirement Specification (URS)

Zoo Bazar

Date: 26-03-2024

Group: 2

Class: CB06

Table of Contents

Functional Requirements

ID	Name
FR-01	All employees must log in
FR-02	Admin can add, remove, edit employee
FR-03	Admin can assign shifts
FR-04	Admin and vet can access animal data
FR-05	Vet can update health status of animal
FR-06	Employees can view their assigned shifts and work schedules
FR-07	Employees can request shift changes or time-off, subject to approval by the admin
FR-10	Generate reports on employee attendance, shift coverage, and animal health statistics
FR-11	Notification system to alert employees of important updates or announcements
FR-12	Admin can add new animals to the system with details such as species, breed, age, and medical history

Non-Functional Requirements

Security: The system must employ robust authentication mechanisms to safeguard sensitive data.

Usability: User interfaces should be intuitive and user-friendly to facilitate ease of use.

Performance: The system should be responsive and able to handle multiple concurrent users without significant delays.

Scalability: The system should be designed to accommodate future growth in terms of data volume and user base.

Reliability: The system should minimize downtime and ensure data integrity through regular backups and redundancy measures.

Use Cases

Use Case 1: Purchase Tickets Online

Actors: Visitor (Customer), Admin

Main Success Scenario:

1. Visitor selects the type and quantity of tickets they wish to purchase and proceeds to checkout.
2. System requests shipping information.
3. Visitor fills in shipping information (address, preferred delivery method) and confirms.
4. System calculates and displays full pricing information, including any additional fees.
5. Visitor fills in payment information and confirms.
6. System confirms successful payment and sends a confirmation email to the visitor.

Extensions:

2a: No tickets are selected

2a.1: System displays a message indicating that no tickets have been selected.

2a.2: End of use case.

2b: Visitor is logged in

2b.1: System displays saved shipping information.

2b.2: Return to Main Success Scenario step 4.

Use Case 2: Manage Employees

Actors: Admin

Main Success Scenario:

1. Admin accesses the employee management system.
2. Admin adds, removes, or edits employee information as needed.
3. Admin assigns shifts to employees according to the schedule.

Extensions:

2a: Admin tries to remove an employee with active shifts.

2a.1: System prompts the admin with a warning about active shifts.

2a.2: Admin decides whether to proceed with the removal or cancels the action.

2b: Admin edits employee information

2b.1: System validates the changes and updates the employee information accordingly.

Use Case 3: View Work Schedule

Actors: Employee

Main Success Scenario:

1. Employee logs in to the system.
2. Employee navigates to the work schedule section.
3. System displays the assigned shifts for the employee.

Extensions:

None

Use Case 4: Request Shift Changes/Time-off

Actors: Employee, Admin

Main Success Scenario:

1. Employee logs in to the system.
2. Employee submits a request for shift changes or time-off.
3. System sends the request to the admin for approval.
4. Admin approves or denies the request.

Extensions:

2a: Employee submits an invalid request (e.g., overlapping shifts)

2a.1: System notifies the employee about the issue and prompts for corrections.

2a.2: Employee makes necessary changes and resubmits the request.

Use Case 5: Generate Reports

Actors: Admin

Main Success Scenario:

1. Admin accesses the reporting system.
2. Admin selects the type of report to generate (employee attendance, shift coverage, animal health statistics).
3. System generates the report based on the selected criteria.

Extensions:

2a: Admin requests a report with invalid parameters

2a.1: System notifies the admin about the invalid parameters.

2a.2: Admin adjusts the parameters and resubmits the request.

Use Case 6: Add New Animals

Actors: Admin

Main Success Scenario:

1. Admin accesses the animal management system.
2. Admin enters details such as species, breed, age, and medical history for the new animal.
3. System adds the new animal to the database.

Extensions:

2a: Admin tries to add an animal with incomplete information

2a.1: System prompts the admin to provide missing information.

2a.2: Admin completes the required fields and submits the information again.

Use Case 7: Update Animal Health Status

Actors: Vet

Main Success Scenario:

1. Vet logs in to the system.

2. Vet navigates to the animal health section.
3. Vet selects an animal to update its health status.
4. Vet updates the health status of the selected animal (e.g., healthy, under observation, treated).
5. System confirms the update and records the changes.

Extensions:

4a: Vet encounters an issue while updating the health status

4a.1: System displays an error message indicating the issue.

4a.2: Vet resolves the issue and resubmits the update.

Use Case 8: Handle Employee Shift Swap

Actors: Employee

Main Success Scenario:

1. Employee logs in to the system.
2. Employee navigates to the shift swap section.
3. Employee finds a colleague willing to swap shifts.
4. Employees agree on the shift swap details (date, time, roles).
5. System processes the shift swap and updates the schedules accordingly.

Extensions:

4a: Employees cannot agree on shift swap details

4a.1: System notifies the employees about the disagreement.

4a.2: Employees renegotiate the swap details or cancel the request.

Use Case 9: Manage Notifications

Actors: Admin, Employee

Main Success Scenario:

1. Admin configures notification settings for employees.
2. System sends notifications to employees for important updates or announcements.
3. Employees receive notifications via email or within the system interface.

Extensions:

2a: Admin encounters issues with notification settings configuration

2a.1: System displays an error message indicating the issue.

2a.2: Admin resolves the issue and reconfigures the settings.

Use Case 10: Monitor Animal Feeding Schedule

Actors: Zookeeper

Main Success Scenario:

1. Zookeeper logs in to the system.
2. Zookeeper accesses the animal feeding schedule.
3. System displays the feeding schedule for each animal species.
4. Zookeeper updates the feeding schedule based on animal dietary needs and requirements.
5. System confirms the update and adjusts the feeding schedule accordingly.

Extensions:

4a: Zookeeper encounters an unexpected change in animal dietary needs

4a.1: System notifies the zookeeper about the change.

4a.2: Zookeeper adjusts the feeding schedule accordingly and confirms the update.

Use Case 12: Handle Animal Medical Emergencies

Actors: Staff Member, Veterinary Team

Main Success Scenario:

1. Staff member notices an animal showing signs of distress.
2. Staff member alerts the veterinary team about the situation.
3. Veterinary team accesses the medical emergency section of the system.
4. Veterinary team reviews the animal's medical history and current symptoms.
5. Veterinarian diagnoses the condition and prescribes necessary treatment.
6. System records the medical intervention and updates the animal's health status.

Extensions:

5a: Veterinarian requires additional diagnostic tests or consultation

5a.1: System facilitates scheduling additional tests or consultations.

5a.2: Veterinarian performs necessary tests or consults with specialists for further diagnosis and treatment.