Zoo-Bazaar

Project plan

Authors: *Mihail, Stanislav, Atanas, Desislav*

Date: *18.09.2023*

Team: *Group 6*

Version: *1.0*

Content

Introduction

In this project, we will aim to create a platform for Zoo-bazar, a zoo in Eindhoven. A corporation has commissioned our team to create an application that focuses on employee and zoo management, over the next 6 weeks. We will aim to be as transparent and as communicative as possible with our client representative from Jupiter Corp. and strive to complete the features in the allocated timeframe.

1. Client & Team

Client: Michiel Koehorst

Email: m.koehorst@fontys.nl

- Represents Zoo-bazaar.

Our team consists of:

- Mihail
- Stanislav
- Atanas (team leader for the first 6 weeks)
- Desislav (secretary), email: 457570@student.fontys.nl

Our team's name is the ZooManiacs. Desislav represents our group.

2. Current situation

Our client is currently using Excel sheets to save their employees and animal information. This is not efficient and wastes a lot of time. Zoo Bazaar has problems with their scheduling their employees and their animal care. We hope to solve these problems with our application.

3. Problem description

They would like to have a program to manage their schedules and animal care to have an easier time managing their zoo.

Summary of the client meeting:

3.1 User Profiles and Application Access:

- User Types: Employees, Caretakers, HR Admins.
- HR Admins can perform CRUD operations for employee management.
- Employees can view and edit their personal details, like phone number and address, via the website.
- Employee account creation is handled by HR, who also set the initial password. Employees cannot create their own accounts.

3.2 Employee Data and Access Levels:

- Data Stored: Email, ID, job position, SSN, contract number, contract type (permanent/temporary), contract history, start and end dates of contracts.
- Access Level: Employees can only view their own information.

3.3 Data Location:

- Data related to animals and their management is stored and accessed via the desktop app.

3.4 Admin Account Creation:

- HR Admins create employee accounts.

3.5 Animal Management and Viewing:

- Animals can be created and managed through the desktop app.

3.6 Roles in System Administration:

- HR tracks animals.
- Designations: HR Manager, Animal Manager, and those with combined roles.

3.7 Animal Data and Operations:

- Operations: CRUD for animals, tracking animal transfers, family, species, feeding schedules.
- Future goal: Automate the aforementioned processes.

3.8 Application Usage:

- Specific employee roles accessing the desktop app and website are not specified.

3.9 Scheduling:

- Feedings involve caretakers.
- Cleaning is planned by activity type.
- Employees can view their schedules on the website.
- Schedules are manually created with future hopes for automation.

3.10 Desired Features for Upcoming Release:

- By 12.10, the focus is on handling animals, managing employees and locations.
- Features include indoor/outdoor location management and task assignments like feeding animals.

3.11 Design Aesthetics:

- Preferred website color scheme: Friendly, positive green tones aimed at appealing to younger visitors for a fresh look.

4. Project goal

Our project would like to achieve a fully functional program to manage the zoo, so that they have a better and more efficient way to plan for their employees and a way to track their animals and their care.

5. Deliverables

We are supplying this company with:

- A Desktop application for managing staff and animals (care).
- A web page/website where their employees can login and find their schedules, and find what animals need care and when.
- Documentation on how this was made.

6. Non-Deliverables

We are not supplying this company with:

- Maintenance for this project after we deliver it.
- A manual on how to use the application & website.

7.Constraints

We are limited to the coming constraints:

- The waterfall iteration will conclude in 6 weeks / 18 weeks in total.
- The program will be written in C#, HTML, CSS and MySQL.

8. Phasing