



ZOO BAZAAR

Project Plan_v0.4

Date: 12/7/2022

Alaa Bolbol – 3948374

Erfan Alizade – 4592964 Aya

Shikh Suliman – 4668839

Table of versions:

Version	Status	Start date	Release date
0.1	Released	10/14/2022	10/21/2022
0.2	Released	10/23/2022	10/23/2022
0.3	Released	11/16/2022	11/18/2022
0.4	Released	12/06/2022	12/06/2022
0.5	Released	01/12/2023	01/12/2023

Table of Contents:

Table of versions:	2
Client:	3
Team:	4
Current situation:	4
Waterfall release:	4
Iteration 1:	5
Iteration 2:	5
Iteration 3:	5
Problem description:	6
Project goal:	6
Deliverables:	6
Non-deliverables:	6
Constraints:	6
Phasing:	6

Client:

The owner of the “Zoo Bazaar” company Mariëlle Fransen, is the client that asked us for a software solution for the problems that they are facing. We as “S2-CB01-Group 4” will be working on a solution for these problems.

Team:

Our team consists of four members: Aya Shikh Suliman (leader), Erfan Alizada, Alaa Bolbol, Alexander Petrov. The client can contact our leader with a phone number, email address, or by planning a physical meeting.

We also made a logo for our team:



Current situation:

Waterfall release:

Zoo Bazaar is a new zoo that is going to open soon. This zoo needs an online platform and an application to be able to start its activities.

Zoo Bazaar needs an IT solution for the problems they are facing. The zoo is going to have employees and animals. They should be able to manage the employees and the animals, and that's the most important thing in their requirements. They should be able to have a feeding schedule for animals and assign employees to feed animals. The employees should be scheduled on the right days and their working day preferences should be considered.

The company also wants a web application in which their clients can find more information online or buy tickets for visiting the zoo. Moreover, this web application should also be a platform where the employees can find the working schedule and feeding table online from anywhere.

Some of the features above are going to be implemented in the future. Of course, if we have the needed budget and time. On the other side, it is also important that the customer is satisfied with the solution we offer. There are other competitors out there, that is why we are going to try to offer the best solution and quality.

To keep everything to our client's wishes, which is also related to the quality of the work we offer, we are going to plan some meetings with the client to make sure everything is going the way that is expected from us.

Iteration 1:

Currently, we have developed a desktop application that contains the basic features. It includes basic authentication and authorization for employees, adding, deleting, promoting, editing, and assigning employees to a shift. In addition, we can now add, delete, and search through registered animals.

We have two login options in our desktop application. You can log in as an administrator, or as a scheduler. An administrator is the one who manages the employees and animals. And the scheduler is the one that is going to create the schedules for employees and animals.

For now, the UI is almost what the client needs. There are still some minor changes that we must make to have a UI that the client is satisfied with.

This is what we have implemented so far. In the next iteration we are going to mostly focus on the animal part, since managing animals is the priority for the client.

Iteration 2:

At this moment, we have implemented the functionalities for managing and scheduling animals. The administration can add, remove, and edit an animal. They can also view the feeding timetable and the health situation table. On the other side, the scheduler can view the general information and create a feeding time for an animal.

For the website, we made the paper prototype according to the needs of the client.

After creating everything, we discussed all the new functionalities with the client, and we got feedback that we are going to apply in the next iteration. In addition, we are going to implement the functionalities for managing and scheduling the employees. We are also going to make the design and the employee page for the website.

Iteration 3:

In the past iteration, we couldn't fully implement what we planned to. Our plan was to apply the feedback we got from the client, make the employee schedule, make the design of the website, and the employee page for the website. We were able to apply the client's feedback and make the design of the website, and the employee page on the website, but we couldn't finish the employee schedule because it was more than expected. We have to take more than one thing into consideration while implementing the schedule. We implemented the first step, but we still have to take more things into consideration.

For this iteration, we are going to finish implementing the employee schedule and make the design of the website functional.

Iteration 4:

In the last iteration we are finalizing our project so we are implementing what we have planned for this iteration. For the website we are making the report feature. We also made the schedule of an employee shown in the schedule page. In addition, there is a possibility for employees to send their preference working day and also they can ask for vacation or report sickness. For the desktop application, we are making the employee schedule and the feeding timetable automatic.

Problem description:

The biggest challenge that Zoo Bazaar's management foresees, is keeping track of their employees and animals. The employees should be scheduled for the right activities and their working day's preferences should be considered. They should be able to keep track of animals, feed them, and make a daily report on the animal's status. Therefore, an administrative system is needed. It should allow resources to keep track of the (e.g., employees' animals, locations, etc.). Furthermore, it should be possible to allow the users to create a feeding timetable for the animals. Ideas such as the addition of (online) ticket sales, statistics, other types of timetables, customer complaint handling, and a dedicated website for employees to view information about their schedule also feature that is required but can be added in future versions. But for now, they would like to focus on the management of employees and animals.

Project goal:

The goal is to deliver the features that are asked from our management system for managing employees and animals, a website for clients to buy tickets, handle customers' complaints and reviews, and to be used as an online platform for employees to find their schedule online.

The main goal of the project is to offer a solution for solving the problem of managing animals and employees.

Deliverables:

Based on the information we got from the interviews with our client, we are going to offer a desktop application and a web application as a solution. We are also going to deliver the URS, test plan, and test report to the client.

Non-deliverables:

There are some documents and files that we are not going to deliver to the client. For example, documents that have to do with our future implementation. In addition, we are also not going to deliver the UML diagram because it is too technical for the client, and it does not explain anything that the client must do.

Constraints:

The programming languages that we are going to use are C Sharp, HTML5, and CSS3. We might also need hardware which we can arrange by using the budget that the "Zoo Bazaar" company will give us for this purpose.

Phasing:

We made a diagram and wrote an explanation about each activity in each phase **Phase**

1: waterfall:

Activities:

Activity 1:

Project plan: week 1:

- Analyse the reader
- Interview the client
- Write version 0.1
- Get feedback
- Improve project plan
- Get approval **Activity**

2:

URS: week 2:

- Write down the agreements with the client
- Decide what functionalities the application is going to have
- Write down the functional & non-functional requirements
- Write down the needed use cases
- Get feedback
- Improve document
- Get approval **Activity**

3:

Design: week 3:

- Make paper prototype
- Get feedback
- Improve the paper prototypes
- Make windows form design
- Get feedback
- Improve the UI
- Get approval

Activity 4:

Implementation: week 4:

- Make a UML diagram

Get feedback
Improve UML
Get approval
Implement classes
Implement features
Get feedback
Improve
Get approval **Activity**

5:

Testing: week 5:

Test application with group members
Improve according to bugs we see **Activity**

6:

Deliverable 1: week 6:

A presentation about the latest version of the software solution with the client.

Phase 2:

Iteration 1:

Activities:

Activity 1:

Project plan: week 7:

Analyse the reader
Interview the client
Write project plan
Get feedback
Improve project plan
Get approval **Activity**

2:

URS: week 7:

Write URS

Get feedback

Improve URS

Get approval **Activity**

3:

Database design: week 8:

Decide what tables we need

Create database

Create database design

Get feedback

Improve database design

Get approval

Activity 4:

The implementation of database: week 8:

Implement database

Get feedback

Improve code

Get approval **Activity**

5:

Implement the general page for animals: week 8 - week 9:

Implement page

Get feedback

Improve code

Get approval **Activity**

6:

Implement the feeding timetable for animals: week 9:

Implement feeding timetable

Get feedback

Improve code

Get approval **Activity**

7:

Implement the health situation page for animals: week 9:

Implement page

Get feedback

Improve code

Get approval **Activity**

8:

Deliverable 2:

A feedback review form: week 10:

Read the form

Fill the form

Submit the form

Iteration 2:

Activities:

Activity 1:

Project plan: week 11:

Interview the client

Write project plan

Get feedback

Improve project plan

Get approval **Activity**

2:

URS: week 11:

Write URS

Get feedback

Improve URS

Get approval

Activity 3:

Edit the implementation for the animal general page: week 12:

Edit implementation
Get feedback
Improve code
Get approval **Activity**

4:

Edit the implementation for the feeding timetable page: week 12:

Edit implementation
Get feedback
Improve code
Get approval

Activity 5:

Add a search feature for the health situation page: week 12:

Edit implementation
Get feedback
Improve code
Get approval **Activity**

6:

Implement the schedule page for employees: week 12:

Implement page
Get feedback
Improve code
Get approval **Activity**

7:

Implement the attendance page for employees: week 12:

Implement page
Get feedback
Improve code

Get approval **Activity**

8:

Make the UI for the website: week 12:

Make the UI

Get feedback

Improve UI

Get approval **Activity**

9:

Implement employee page on the website: week 12:

Implement page

Get feedback

Improve code

Get approval

Activity 10:

Write process document: week 12:

Write document **Activity**

11:

Make the presentation: week 12:

Make the presentation

Divide the roles **Activity**

12:

Deliverable 3:

A presentation: week 12:

Present

Get feedback

Iteration 3:

Activities:

Activity 1:

Project plan: week 13:

Interview the client

Write project plan

Get feedback

Improve project plan

Get approval **Activity**

2:

URS: week 13:

Write URS

Get feedback

Improve URS

Get approval

Activity 3:

Finish implementing the employee schedule: week 14:

Implement schedule

Get feedback

Improve code

Get approval **Activity**

4:

Implement features on website: week 14:

Implement features

Get feedback

Improve code

Get approval

Activity 5:

Fill up peer review document: week 15:

Activity 6:

Deliverable 4:

A presentation/discussion about our findings/feedback: week 12:

Present

Get feedback

Iteration 4:

Activities:

Activity 1:

Project plan: week 17:

Finalize project plan

Get approval

Activity 2:

URS: week 17:

Finalize URS

Get approval

Activity 3:

Implement features on desktop: week 17:

Implement tickets management

Implement report management

Implement attendance page

Implement automatic scheduling

Get approval

Activity 4:

Implement features on website: week 14:

Implement schedule

Implement report

Implement workday's preference

Implement day off option

Implement real email submission with tickets.

Get approval

