YeOrada

Requirements Specification and Analysis

08.03.2020

Ahmet Can Terlikçi

Ali Haydar Konuk

Yaren Çoşkun

Fatih Mehmet İdgü

Prepared for

SOFT3102 Software Development Practice



Table of Contents

[1. Introduction 1](#_Toc496873294)

[1.1. Purpose of the System 1](#_Toc496873295)

[1.2. Scope of the System 1](#_Toc496873296)

[1.3. Objectives and Success Criteria of the Project 1](#_Toc496873297)

[1.4. Definitions, Acronyms, and Abbreviations 1](#_Toc496873298)

[1.5. Overview 1](#_Toc496873299)

[2. Current System 1](#_Toc496873300)

[3. Proposed System 1](#_Toc496873301)

[3.1. Overview 1](#_Toc496873302)

[3.2. Functional Requirements 2](#_Toc496873303)

[3.3. Nonfunctional Requirements 2](#_Toc496873304)

[Usability 2](#_Toc496873305)

[Reliability 2](#_Toc496873306)

[Performance 2](#_Toc496873307)

[Supportability 2](#_Toc496873308)

[Implementation 2](#_Toc496873309)

[Interface 2](#_Toc496873310)

[Packaging 2](#_Toc496873311)

[Legal 2](#_Toc496873312)

[3.4. System Models 2](#_Toc496873313)

[Scenarios 2](#_Toc496873314)

[Use case model 2](#_Toc496873315)

[Object model 2](#_Toc496873316)

[Dynamic model 2](#_Toc496873317)

[User interface—navigational paths and screen mock-ups 3](#_Toc496873318)

[3.5. Project Schedule 3](#_Toc496873319)

[4. Glossary 3](#_Toc496873320)

[5. References 3](#_Toc496873321)

REQUIREMENTS ANALYSIS DOCUMENT[1]

# Introduction

## Purpose of the System

Today, there are many restaurants, cafes and bars. Also, they are opening, closing, and moving. Moreover, they update their menus, concept or scopes. For a person, it is hard to follow this concept continuously.

YeOrada basicly satisfies this need and try add new value to people’s lifes by simplifying the searching good and proper restaurants. It is a user-review based restaurant search system.

## Scope of the System

YeOrada aims to match the restaurant owners and their restaurants with the users who are searching for restaurants. In general, system has 3 different types of user; Customer, Client and Admin.

Clients are the restaurants controlled by their owners. They can share and update any information about their restaurants including photos of the restaurant and its menu. Customers are the main type of users. Any visitor can register the system as a Customer and start to search restaurants. They can share their opinions about restaurants with comments and can be interact with other Customers through answering their comments. Customers also have a profile page which they can display their comments, comment photos, profile photo and etc. Admin is manager of the system. They have absolute power in the YeOrada.

YeOrada will be published as a web application. Mobile applications of the system are out of scope.

## Objectives and Success Criteria of the Project

The success of the project is depends on the following core set of objectives:

* Project schedule is completed at the planned due date of project.
* Project covers the all the specified scope.
* Client’s documented change requests are all implemented in the project.

## Definitions, Acronyms, and Abbreviations

**Client:** They are restaurant accounts which will be controlled by their owners.

**Customer:** Customer account. They are the main type of the users in the system.

**Admin:** Admin is the manager of the YeOrada. They have the absolute power.

**Restaurant:** In our context, it is the same as the Client. Restaurant is a place where meals or drinks are prepared and served to customers. Therefore, it covers bars, cafes and pubs with the real restaurants.

**Menu:** A list of foods and drinks which a restaurant can serve.

## Overview

The rest of the documents contains Current System part which explains how the tasks in the system are accomplished, Proposed System part which includes an functional overview of the system, functional and nonfunctional requirement of the system, System Models which consists of scenarios, use cases, overall use case diagram of the system and overall class diagram of the system, Glossary which explains the objects the system, and references which includes all type of sources.

# Current System

YeOrada is consists of three distinct types of users: Client, Customer and Admin. In the below, their functions and boundaries will be specified.

Each Client has a one corresponding restaurant in their account. Clients will be specifiy that restaurant through a registration and approval in the system. Approval will be done by Admins in the system. They can add photos of their restaurant and its menu to the system. Also, update profile of their restaurant if they wish.

Customers will be register to the system through a easy registration form and they no need to pass from an approval process like Clients. Customers may add comments to the profiles of restaurants and also give answers to the comments of other Customers. Comments must pass through a approvall process in the system which done by Admins. They have a profile which they can see their comments and photos which uploaded through comments. Also, Customers have a profile photo. They may change it or use default one.

Admin accounts will be created manually in the system. They have absolute power. Admins will be accept or decline client requests of the visitors and also comment requests of the Customers. They can also add new Clients to the system and also update the profiles of the Clients in the system.

# Proposed System

## Overview

## Functional Requirements

1. Clients can update their profile. Client profile is consisting of photos, name, e-mail address, address, working hours and days, menu, phone number, rate, number of rates of the client. Clients can also add some predefined tags for the client.
2. Customers can update their profile. Customer profile is consisting of name, username, e-mail address, address, profile photo, phone, their comments, their biography. They can add comments to any client. Comment is consisting of the rate, some photos and review of the client. They can search and display information about any client. They can like and comment to any comment.
3. Visitors can search and display information about any client. They can register by providing an e-mail, username and password to the system. Also, they can login to the system with their e-mail and password.
4. Admins can accept or decline of new client requests. They can accept or decline of adding new comment from Customer. They can update all client and customer accounts and add new client and customer accounts. Also, they can update their profile. Admin profiles is consisting username, e-mail address and password.

## Nonfunctional Requirements

*Usability*

1. All pages in the website must be responsive to all types of screen resolutions and sizes.
2. A Customer must be able to add a comment to a Client at most 4 steps.
3. A Customer which is never use YeOrada before must be able to find a specific client which is he/she searching for and add a comment into that Client at most 3 minutes.

*Reliability*

1. The system must prevent to show removed clients from the system in search results.
2. User must encounter with an error page or pop-up when a problem occurs in the system.

*Performance*

1. The system must be display a result through a client search at most 5 seconds.

*Implementation*

1. The system must support .jpg and .png image types.
2. The system must support .pdf and .docx document types.
3. The source code of the system must be written in Java with using JSF Framework.

## System Models

### Scenarios

**Scenario Name** Register

**Participating Actor** Kenan:Visitor

**Instances**

**Flow of Events** 1. Kenan wants to add comment to a resturant in Kadıköy named Koreköy and therefore want to the register the YeOrada website.

2. Kenan enters the register page in the website.

3. He enters his name, surname, username, email address and

password to the given form in the website. 4. Then, he clicks the register button and activates the register function of the YeOrada.

5. After, he receives the message from the website as “You Registered Succesfully.”.

**Scenario Name** SearchForClients

**Participating Actor** Kenan:User

**Instances**

**Flow of Events** 1. Kenan wants to eat some chineese food in Kadıköy and

enters the YeOrada website to do quick a search for the near

restaurants.

2. He notices the search bar in the main page of the website.

3. First, he selects the location from using a menu and

writes “chineese food” to the textfield in the bar. Then, he clicks the search button.

4. After, he receives a new page in the website that consists of a list of restaurants related the “chineese food” keyword in the

selected location.

**Scenario Name** UpdateMyAccountSettings

**Participating Actor** Kenan:Customer

**Instances**

**Flow of Events** 1. Kenan is a Customer in the YeOrada system and wants to

change his email in the website.

2. Therefore, he logins the website and then enters the Settings

page in the website.

3. He immediately sees Account Settings in the webpage and changes his email address.

4. After that, he clicks save button in the webpage and receives

a message from the system which says “Saved Succesfully.”.

**Scenario Name** AddComment

**Participating Actor** Selin:Customer

**Instances**

**Flow of Events** 1. Selin ate a Suffle in the Bahçelievler Mutlu Kafe yesterday

and likes it very much. So she wants to a write a comment about Suffle and the restaurant.

2. She searches for Bahçelievler Mutlu Kafe by using the search

panel in the main page of the YeOrada, finds it and enters into

it’s profile.

3. Then, he clicks the “New Comment” button in the profile

page and receives a pop-up which wants a rate from 1 to 10, a

comment and some photos from her.

4. After that, he clicks the “Send” button and notices a message which says that “Your comment sended to a supervisor for

approval.”.

**Scenario Name** ChangeProfilePhoto

**Participating Actor** Veli:Customer

**Instances**

**Flow of Events** 1. Veli wants to change his profile photo in the YeOrada, so he first logins the website.

2. Then, he opens Settings page.

3. On this page, currently she clicks the Change button which is located under his profile photo and a pop-up comes up which shows to him the files in his device.

4. He locates and clicks to the appropriate photo and clicks the save button.

5. After that, he notices the new photo on his Settings page.

**Scenario Name** LikeComment

**Participating Actor** Selin:Customer

**Instances**

**Flow of Events** 1. Selin searches for near restaurants in Bahçelievler, Merkez to

find a restaurant which cooks chineese food by using the search panel in the main page of the YeOrada.

2. She locates a restaurant and enters its profile. Then, she starts

to read comments of that restaurant to ensure that it cooks good or bad foods.

3. She sees a funny comment about restaurant which says

“Go to this restaurant if you want to suicide easily.”, laughes to it and likes it by using the LIKE button on this comment.

4. After that, she notices that the LIKE button now becomes blue.

**Scenario Name** EvaluteClientRequest

**Participating Actor** Metin:Admin

**Instances**

**Flow of Events** 1. Ahmet is the owner of a restaurant in Kadıköy called Kadıköy Mutlu Café and wants to add his restaurant to the YeOrada. Therefore, he enters to the main page of the system and fills out a form which includes some information about his restaurant. Then, he sends this form for approval to the system.

2. Metin checks the client requests in the system every hour and notices the client request of the Ahmet’s restaurant, Kadıköy Mutlu Café.

3. Metin clicks the button Read More and reads the all information about the restaurant. Then, he concludes that the Kadıköy Mutlu Café is a valid restaurant to add into the YeOrada.

4. He comes back to the previous tab in his browser and clicks Accept button on the client request of the Kadıköy Mutlu Café.

5. After that, client request disappears and next the request comes to its place.

**Scenario Name** UpdateMyGeneralInfo

**Participating Actor** Kadıköy Mutlu Kafe: Client

**Instances**

**Flow of Events** 1. Kadıköy Mutlu Kafe wants to add “Pet Friend” and “Breakfast” tags to the “General Information Panel” in his restaurant profile.

2. It enters the My Restaurant page from the main page of the YeOrada.

3. Then, it clicks the “Update” button on the “General Information Panel” and receives a pop-up which some predefined tags can be selected or deselected with other general information of the restaurant.

4. It selects the “Pet Friend” and “Breakfast” tags, and then clicks the “Save” button.

5. After that, it receives a message which is “Saved Successfully.”.

**Scenario Name** UpdateMyMenuPhotos

**Participating Actor** Kadıköy Mutlu Kafe: Client

**Instances**

**Flow of Events** 1. Kadıköy Mutlu Kafe wants to add some new photos to the menu album in his restaurant profile.

2. He enters the My Restaurant page from the main page of the YeOrada.

3. Then, He clicks the “Update” button on the menu album and receives a pop-up. He notices that previous menu photos are currently shown to him.

4. He clicks the “Add New Photo” button and receives a new pop-up which shows him the files in his computer.

5. After that, he selects a photo from his computer and clicks “Okey” button.

6. Finally, he notices that the photo is upluoaded to the website and placed into the pop-up together with previous menu photos.

**Scenario Name** EvaluteCommentRequest

**Participating Actor** Metin:Admin

**Instances**

**Flow of Events** 1. Selin ate a Suffle in the Bahçelievler Mutlu Kafe yesterday and likes it very much. So she added a comment about Suffle and the restaurant to the profile page of the Mutlu Kafe but the comment didn’t published immediately, instead it is sended to a supervisor, an Admin, for approval.

2. Metin checks the comment requests in the system every hour and notices the comment request of the Selin’s.

3. Metin reads the all the comment and examines the photos in it. Then, he concludes that this comment is not violate any rule for comments in the system and therefore, clicks the Accept button on the comment request.

4. After that, comment request disappears and next the request comes to its place.

**Scenario Name** UpdateMyClientPhotos

**Participating Actor** Kadıköy Mutlu Kafe: Client

**Instances**

**Flow of Events** 1. Kadıköy Mutlu Kafe wants to add some new photos to the photo album in his restaurant profile.

2. He enters the My Restaurant page from the main page of the YeOrada.

3. Then, He clicks the “Update” button on the photo album and receives a pop-up. He notices that previous photos of his restaurant are currently shown to him.

4. He clicks the “Add New Photo” button and receives a new pop-up which shows him the files in his computer.

5. After that, he selects a photo from his computer and clicks “Okey” button.

6. Finally, he notices that the photo is upluoaded to the website and placed into the pop-up together with previous menu photos.

**Scenario Name** AddClient

**Participating Actor** Metin:Admin

**Instances**

**Flow of Events** 1. YeOrada searches for new restaurants all the time the extend their catalog and they found a new one called Kadıköy Mutlu Café. Some employees went to the restaurant, took appropriate information and some photos for the restaurant’s profile in the YeOrada.

2. Related documents is sended to the Metin and he is responsible for creating a profile for the Kadıköy Mutlu Café.

3. First, He enters the Add New Restaurant page in the YeOrada website and receives a form about critical information of the restaurant like name, phone, email, password and so on . He fills out the form and clicks the Save button. Then, he receives a new page which consists of a message which says “New Restaurant Created Succesfully.”. Also, this page includes a quick link for the profile of the restaurant.

**Scenario Name** UpdateClientSettings

**Participating Actor** Metin:Admin

**Instances**

**Flow of Events** 1. A restaurant in the YeOrada has changed its phone numbers and want to update his website profile as well. Instead of doing this work by own, restaurant owner contacts with the YeOrada and requests to change the phone numbers in the website.

2. Metin receives the job in the company with the appropriate documentation.

3. He locates the related restaurant’s profile page in the system by using search bar in the main page and clicks the Settings icon on the top-right corner of the page.

4. Then, he receives a pop-up which includes critical information of the restaurant like username, email, phones, address and and so on.

5. He changes the phone numbers with new ones and clicks the Save button in the pop-up.

6. After that, he notices that the previous phone numbers is changed to the appropriate ones in the profile page of the restaurant.

**Scenario Name** AnswerTheComment

**Participating Actor** Aysu:Customer

**Instances**

**Flow of Events** 1. Aysu enters to a restaurant which is named as “Tavuk Dünyası Kadıköy” while she searches all restaurants where is near to Kadıköy.

2. She interests a comment about a restaurant while she reads all comment about same restaurant. The comment is “I tried this restaurant, waitstaffs were genial :))))”

3. Then, Aysu comments to this comment with “I agree with you, they are so kind :)”

4. Finally, she writes and clicks to send button. Then, she sees her own comment.

**Scenario Name** DisplayMyComments

**Participating Actor** Aykut:Customer

**Instances**

**Flow of Events** 1. Aykut wants to see all comments he writes in the YeOrada.

2. He enters his own profile page from the main page of the website.

3. Then, he enters to “Comments” tab from opening profile page.

4. Lastly, he views all comments with date, number of likes and which restaurant was written.

**Scenario Name** DisplayMyPhotoAlbum

**Participating Actor** Oğuz:Customer

**Instances**

**Flow of Events** 1. Oğuz wants to see all photos which he adds to the YeOrada by now.

2. He enters his own profile page from the mainpage of the YeOrada.

3. Then, he enters to “Photos” tab from opening profile page.

4. After all, he views all photos with photo’s adding date and about which restaurant.

**Scenario Name** UpdateMyAdminProfile

**Participating Actor** Ayça:Admin

**Instances**

**Flow of Events** 1. Ayça wants to change her own information from her account in the YeOrada.

2. She enters the Settings page from the main page of the YeOrada and immediately sees and clicks the Account Settings tab.

3. She changes e-mail address which she wants to change. Then, she clicks to save button.

4. Finally, she sees updated information when she views own information.

**Scenario Name** ChangeMyPassword

**Participating Actor** Gözde:RegisteredUser

**Instances**

**Flow of Events** 1. Gözde wants to change her password of her account in the YeOrada.

2. She enters the Settings page from the main page of the YeOrada and she sees and clicks the Change Password tab.

3. She enters old password and new password. New password is wanted twice for verification.

4. Lastly, she clicks to the save button and she sees a message which is “Your password has changed successfully!”

**Scenario Name** UpdateMyClientAccount

**Participating Actor** Sivas Etli Pide & Köfte:Client

**Instances**

**Flow of Events** 1. Sivas Etli Pide & Köfte wants to change her own information from his account in the YeOrada.

2. He enters the Account Settings page from the main page of the YeOrada.

3. He changes phone number which he wants to change. Then, he clicks to save button.

4. Finally, he sees updated information when he views own information.

**Scenario Name** UpdateClientPhotos

**Participating Actor** Aziz:Admin

**Instances**

**Flow of Events** 1. Aziz wants to add new client photos for recently participant clients to improve his advertising because he has no client photos yet.

2. Aziz reaches client’s profile page from the search bar of the YeOrada.

3. He uploads new photos getting from his workmates about restaurant by using edit button in client photo field. Then, he clicks to save button.

4. Once for all, he sees updated photos in the head of client’s page.

**Scenario Name** UpdateClientMenuPhotos

**Participating Actor** Volkan:Admin

**Instances**

**Flow of Events** 1. Volkan notices a restaurant which has no menu photo. Then, he gets menu photos from his workmates.

2. He enters profile page of this mentioned restaurant to add menu photos.

3. He uploads menu photos of the restaurant with using edit button above of menu field. Then, he clicks to save button.

4. Last, he sees updated photos in the menu field of client’s page.

**Scenario Name** UpdateClientGeneralInfo

**Participating Actor** Emre:Client

**Instances**

**Flow of Events** 1. Emre notices a restaurant’s information about changing opening hours of the restaurant. Although restaurant changes opening hours, it does not update in the YeOrada.

2. He learns new opening hours of the restaurant from his workmates. Then, he finds of the restaurant’s profile page.

3. He changes opening hours of the restaurant from edit button where is in left side of the page. Then, he clicks to save button.

4. Finally, he sees updated opening hours in the general information field of client’s page.

**Scenario Name** Login

**Participating Actor** Adnan: Customer

**Instances**

**Flow of Events** 1. Adnan wants to find the chineese restaurant which he went

last week and ate a delicious salmon sushi salad but he can’t

remember the name of the restaurant. He memories that he add a comment to the profile of the restaurant in the YeOrada and

therefore enters the YeOrada to find its comment.

2. He clicks the login button on the main page of the website and receives a login form which includes e-mail address and the password fields.

3. He enters his e-mail and password and then clicks to the “Login Button”.

4. Then, he notices the main page of the YeOrada and his name

on the right-top of the website.

### Use case model

### 

*Fig 1.1: The overall use case diagram of the YeOrada.*

You can find the documentation of the each use case in the below.

**Use case name** ChangeProfilePhoto

**Participating Actors** Initiated by Customer

**Flow of Events** 1. Customer opens the Settings page from main page of the

YeOrada website. Then, clicks the “Change” button which is located under his profile photo.

2. YeOrada responds to the Customer by

displaying a pop-up which shows the files

in his/her device.

3. Customer locates and selects a photo and then clicks to the

“OK” button.

4. YeOrada responds to the Customer by closing the pop-up and displaying the new photo instead of current one as a profile photo.

**Entry condition -** Customer is logged in to the YeOrada with his/her email and password.

**Exit condition -** Customer notices the new profile photo in his/her Settings page.

**Use case name** UpdateMyAccountSettings

**Participating Actors** Initiated by Customer

**Flow of Events** 1. Customer opens the Settings page from main page of the

YeOrada website. Then, immediately receives the Account Settings as an editable form. Account Settings includes name,

surname, email, username, phone and adress. After that,

Customer changes some of the fields and clicks the “Save”

button on that page.

2. YeOrada responds to the Customer by displaying a message which says “Saved Succesfully.”

**Entry condition -** Customer is logged in to the YeOrada with his/her email and password.

**Exit condition -** Customer notices the “Saved Succesfully” message of the YeOrada.

**Use case name** AddComment

**Participating Actors** Initiated by Customer

**Flow of Events** 1. Customer enters the profile page of a restaurant in the

YeOrada through searching it using the search panel on the main page of the website. Then, clicks the “New Comment” button in

the profile page which located under the menu photos of the restaurant.

2. YeOrada responds to the Customer by

displaying a pop-up which contains a rate to give to the restaurant from 1 to 5, a short comment section and a photo section. Customer may add some photos to the photo section by pressing “Add New Photo” button on the pop-up.

3. Then, Customer gives at least a rate to the restaurant (Customer may leave photo and comment sections blank.) and then clicks the “Send” button on the pop-up.

4. YeOrada responds to the Customer by closing the pop-up and displaying a message which says “Your comment is sended to a supervisor for approval.”.

**Entry condition -** Customer is logged in to the YeOrada with his/her email and password.

**Exit condition -** Customer notices the “Your comment is sended to a supervisor for approval.” message of the YeOrada as respond.

**Use case name** LikeComment

**Participating Actors** Initiated by Customer

**Flow of Events** 1. Customer enters the profile page of a restaurant in the

YeOrada through searching it using the search panel on the main page of the website. Then, locates a comment in the profile page of that restaurant and clicks the “LIKE” button.

2. YeOrada responds to the Customer by

changing the color of the “LIKE” button to blue.

**Entry condition -** Customer is logged in to the YeOrada with his/her email and password.

**Exit condition -** Customer notices that the color of the “LIKE” button is changed to blue.

**Use case name** EvaluateClientRequest

**Participating Actors** Initiated by Admin

**Flow of Events** 1. Admin enters the Requests page of the YeOrada which is only

allowed for the Admins. Then, immediately receives the summaries of client requests which is sorted in increasing order by their dates. Admin locates a request and then clicks the “Accept” button on the request. Client requests also have a “Decline” button which Admin may use. Decline button declines the client request by don’t adding it to the YeOrada system. Furthermore, Admins may read the full version of the client request by pressing the “Read More” button on the request.

2. If Admin clicks to the “Accept” or “Decline” buttons, YeOrada responds to the Admin by removing the corresponding request from the list on the page. If Admin clicks the “Read More” button, YeOrada responds to the Admin by displaying the profile page of the corresponding client in a new of his/her browser.

**Entry condition -** Admin is logged in to the YeOrada with his/her email and password.

**Exit condition -** Admin notices that corresponding request is removed from the list on the Requests page.

**Use case name** EvaluateCommentRequest

**Participating Actors** Initiated by Admin

**Flow of Events** 1. Admin enters the Requests page of the YeOrada which is only

allowed for the Admins. Then, clicks to the Comment Requests tab on the left-column of the page.

2. YeOrada responds to the Admin by displaying the comment requests which is sorted in increasing order by their dates.

3. Admin locates a request and then clicks the “Accept” button on the request. Comment requests also have a “Decline” button which Admin may use. Decline button declines the comment request by don’t adding it to the corresponding restaurant’s profile page.

4. YeOrada responds to the Admin by removing the corresponding request from the list on the page.

**Entry condition -** Admin is logged in to the YeOrada with his/her email and password.

**Exit condition -** Admin notices that corresponding comment request is removed from the list on the Comment Requests tab of the Requests page.

**Use case name** AddClient

**Participating Actors** Initiated by Admin

**Flow of Events** 1. Admin enters the “Add New Restaurant” page of the YeOrada

which is only allowed for the Admins.

2. YeOrada responds to the Admin by

displaying a blank form which includes name, username, email, phones, city, state and the address.

3. Admin fills out at least the name, email and username fields of the form and then clicks the “Save” button.

4. YeOrada responds to the Admin by displaying message which says “Saved Succesfully”.

**Entry condition -** Admin is logged in to the YeOrada with his/her email and password.

**Exit condition -** Admin notices a message of the YeOrada as a respond which says “Saved Succesfully”.

**Use case name** UpdateClientSettings

**Participating Actors** Initiated by Admin

**Flow of Events** 1. Admin enters the profile page of restaurant through search it

from the search panel on the main page of the YeOrada. Then, clicks the Settings icon on the top-right corner of the page.

2. YeOrada responds to the Admin by

displaying a pop-up with editable fields of name, username, email, phones, city, state and the address of the corresponding restaurant.

3. Admin changes some fields of the form and then clicks the “Save” button.

4. YeOrada responds to the Admin by closing the pop-up.

**Entry condition -** Admin is logged in to the YeOrada with his/her email and password.

**Exit condition -** Admin notices that Settings pop-up is closed by the YeOrada.

**Use case name** UpdateClientGeneralInfo

**Participating Actors** Initiated by Admin

**Flow of Events** 1. Admin enters the profile page of restaurant through search it

from the search panel on the main page of the YeOrada. Then, clicks the Pen icon on the left column of the page which is the “General Information Panel”.

2. YeOrada responds to the Admin by

displaying a pop-up with editable fields of description, extra information, client tags, opening-closing days and hours.

3. Admin changes some fields of the form and then clicks the “Save” button.

4. YeOrada responds to the Admin by closing the pop-up.

**Entry condition -** Admin is logged in to the YeOrada with his/her email and password.

**Exit condition -** Admin notices that corresponding pop-up is closed by the YeOrada.

**Use case name** UpdateMyClientAccount

**Participating Actors** Initiated by Client

**Flow of Events** 1. Client enters the Settings page from the main page of the

YeOrada.

2. YeOrada responds to the Client by

displaying a form with Client’s current information. The form includes name, email, username, phones, city, state and address of the client.

3. Client changes some of the fields on the form, then clicks the “Save” button on the page.

4. YeOrada responds to the Client by displaying a message which says “Saved Succesfully.”

**Entry condition -** Client is logged in to the YeOrada with his/her email and password.

**Exit condition -** Client notices the respond of the YeOrada as a message which says “Saved Succesfully.”.

**Use case name** AnswerTheComment

**Participating Actors** Initiated by Customer

**Flow of Events** 1. Customer search a restaurant from the search bar.

2.YeOrada responds to Customer with displaying a list which is relative with keywords of searching.

3. Customer selects a restaurant from the search list.

4. YeOrada redirects the customer to profile page of the restaurant.

5. Customer sees a comment and he/she replies to comment.

6. YeOrada displays the answer under the main comment.

**Entry condition -** The Customer is logged in to the YeOrada with his/her username and password.

**Exit condition -** The answer is added to specific comment.

**Use case name** DisplayMyPhotoAlbum

**Participating Actors** Initiated by Customer

**Flow of Events** 1. Customer opens his/her own profile page.

2.YeOrada redirects the Customer to his/her own profile page.

3. Customer selects “Photos” from the tabs.

4. YeOrada displays all photos which are uploaded through comments by the Customer until now.

**Entry condition -** The Customer is logged in to the YeOrada with his/her username and password.

**Exit condition -** The photo album of the Customer is displayed by the system.

**Use case name** UpdateMyMenuPhotos

**Participating Actors** Initiated by Client

**Flow of Events** 1. Client opens his/her own profile page.

2.YeOrada redirects the Client to his/her own profile page.

3. Client clicks to “Update” button from menu field.

4. YeOrada displays a pop-up which contains current menu photos and “Add New Photo” button.

5. Client clicks to “Add New Photo” button.

6. YeOrada displays a panel to provide uploading new photo.

7. Client selects photo and “Okey” button.

8. YeOrada displays new menu photos on client’s page.

**Entry condition -** The Client is logged in to the YeOrada with his/her username and password.

**Exit condition -** New menu photos of the Client are added to the system.

**Use case name** UpdateMyClientPhotos

**Participating Actors** Initiated by Client

**Flow of Events** 1. Client opens his/her own profile page.

2.YeOrada redirects the Client to his/her own profile page.

3. Client clicks to “Update” button from restaurant's photo album.

4. YeOrada displays a pop-up which contains current photos of the restaurant and “Add New Photo” button.

5. Client clicks to “Add New Photo” button.

6. YeOrada displays a panel to provide uploading new photo.

7. Client selects photos and “Okey” button.

8. YeOrada displays new photos on client's photo album.

**Entry condition -** The Client is logged in to the YeOrada with his/her username and password.

**Exit condition -** New photos is added to the restaurant's photo album.

**Use case name** UpdateMyGeneralInfo

**Participating Actors** Initiated by Client

**Flow of Events** 1. Client opens his/her own profile page.

2.YeOrada redirects the Client to his/her own profile page.

3. Client clicks to “Update” button from “General Information Panel”.

4. YeOrada displays a pop-up which contains Extra Information, Client Tags, Working hours, Working Days and Short Description of the restaurant.

5. Client changes some information and clicks to “Save” button.

6. YeOrada displays a message that “Saved Successfully.” and updates information of the features of the restaurant.

**Entry condition -** The Client is logged in to the YeOrada with his/her username and password.

**Exit condition -** General Information Panel are updated in the system.

**Use case name** UpdateMyAdminProfile

**Participating Actors** Initiated by Admin

**Flow of Events** 1. Admin opens Account Settings page of the YeOrada.

2.YeOrada displays a form of editable fields to the Admin. The page is consists of the fields name, surname, email and username of the Admin.

3. Admin updates some information and clicks to “Save” button.

4. YeOrada display a message to the Admin which says “Saved Succesfully.”

**Entry condition -** The Admin is logged in to the YeOrada with his/her username and password.

**Exit condition -** Admin is received the corresponding message of the YeOrada as a respond.

**Use case name** UpdateClientPhotos

**Participating Actors** Initiated by Admin

**Flow of Events** 1. Admin opens a Client’s profile page.

2.YeOrada displays the corresponding Client’s profile page to the Admin.

3. Admin clicks to “Update” button on the photo album of the client’s profile.

4. YeOrada displays a pop-up which shows previous photos of the client.

5. Admin clicks the the “Add New Button” on the pop-up.

6. YeOrada displays another pop-up which shows the Admin the files in his/her computer.

7. Admin selects a photo and clicks the “Ok” button on the pop-up.

8. YeOrada closes the pop-up and displays the new photo in the previous photo album pop-up with other uploaded photos of the client.

9. Admin clicks the Save button on the photo album pop-up.

10. YeOrada closes the photo album pop-up and displays the new photo in the client’s profile.

**Entry condition -** The Admin is logged in to the YeOrada with his/her username and password.

**Exit condition -** Client is received the new client photo on his/her profile’s photo album.

**Use case name** UpdateClientMenuPhotos

**Participating Actors** Initiated by Admin

**Flow of Events** 1. Admin opens a Client’s profile page.

2.YeOrada displays the corresponding Client’s profile page to the Admin.

3. Admin clicks to “Update” button on the menu album of the client’s profile.

4. YeOrada displays a pop-up which shows previous menu photos of the client.

5. Admin clicks the the “Add New Button” on the pop-up.

6. YeOrada displays another pop-up which shows the Admin the files in his/her computer.

7. Admin selects a photo and clicks the “Ok” button on the pop-up.

8. YeOrada closes the pop-up and displays the new photo in the previous menu album pop-up with other uploaded menu photos of the client.

9. Admin clicks the Save button on the menu album pop-up.

10. YeOrada closes the menu album pop-up and displays the new photo in the client’s profile.

**Entry condition -** The Admin is logged in to the YeOrada with his/her username and password.

**Exit condition -** Client is received the new menu photo on his/her profile’s menu album.

**Use case name** Register

**Participating Actors** Initiated by Visitor

**Flow of Events** 1. Visitor opens the register page in the website.

2. YeOrada responds to the visitor by

displaying a register form which is included username, name, surname, e-mail and password, all fields are required.

3. Visitor fills form all of the fields on opening page and clicks register button.

4. YeOrada displays a message such “You Registered Succesfully.”.

**Entry condition -** The Visitor fills all fields with given format and his/her e-mail is not registered YeOrada before.

**Exit condition -** The Visitor received a message from YeOrada about registration.

**Use case name** SearchForClients

**Participating Actors** Initiated by User

**Flow of Events** 1. User selects a location through a list of avaliable locations in

the YeOrada and enters the name of the restaurant to the textfield in search panel. Then, he clicks the search button.

2. YeOrada displays a list of restaurants related the what User entered to the search panel.

**Entry condition -** User is entered to the main page of the YeOada.

**Exit condition -** The User receives a list of restaurants related with what he/she write on search bar.

**Use case name** ChangeMyPassword

**Participating Actors** Initiated by RegisteredUser

**Flow of Event** 1. RegisteredUser clicks the “Settings” page from the main page of the YeOrada.

2. YeOrada responds to the customer by

showing “Account Settings” tab.

3. RegisteredUser clicks “Change Password” tab.

4. YeOrada displays a form which is consists of old password and new password of the RegisteredUser.

5. RegisteredUser enters his/her old password and new password for twice to verify it and then, clicks the “Save Button”.

6.YeOrada, displays a message such “Your password has changed successfully!”.

**Entry condition -** The RegisteredUser is logged in to the YeOrada

with his/her username and password before.

**Exit condition -** The RegisteredUser received a message from the YeOrada about the changing his/her password successfully.

**Use case name** DisplayMyComments

**Participating Actors** Initiated by Customer

**Flow of Events** 1. Customers enters his/her Profile page in the YeOrada from the main page of the website.

2. YeOrada displays a list of Comments of the corresponding RegisteredUser.

**Entry condition -** The Customer is logged in to the YeOrada

with his/her username and password before.

**Exit condition -** The RegisteredUser receives a list of his/her comments.

**Use case name** Login

**Participating Actors** Initiated by Customer

**Flow of Events** 1. Customer clicks to “Login Button” on main page of YeOrada.

2.YeOrada displays a form which includes e-mail and password, both of them are required.

3. Customer writes his/her own e-mail and password and then, clicks “Login Button” on the page.

4.YeOrada checks his/her e-mail and password after verified them, YeOrada displays home page.

**Exit condition -** The Customer notices his/her name and surname on the top-right menu of the YeOrada.

### Object model

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

*Fig 1.2: The Class Diagram of the YeOrada.*

### Dynamic model

The corresponding sequence diagrams of the each use case can be found in the provided visual paradigm project. Also, you can find three samples of sequence diagrams in the YeOrada.

ekran görüntüsü içeren bir resim

Açıklama otomatik olarak oluşturuldu

*Fig 1.3: The sequence diagram of the AddClient use case.*

*harita içeren bir resim

Açıklama otomatik olarak oluşturuldu*

*Fig 1.4: The sequence diagram of the UpdateClientSettings use case.*

*harita içeren bir resim

Açıklama otomatik olarak oluşturuldu*

*Fig 1.5: The sequence diagram of the AddComment use case.*

### 

### User interface—navigational paths and screen mock-ups

Screen mock-ups is prepared as a Balsamiq Mockups project. You can find the corresponding project in the project file. Four major examples of the mockups are given below:

ekran görüntüsü içeren bir resim

Açıklama otomatik olarak oluşturuldu

*Fig 1.6: The screen mockup of the Homepage.*

*ekran görüntüsü içeren bir resim

Açıklama otomatik olarak oluşturuldu*

*Fig 1.7: The screen mockup of the Restaurant Profile page.*

*ekran görüntüsü içeren bir resim

Açıklama otomatik olarak oluşturuldu*

*Fig 1.8: The screen mockup of the Customer Account Settings page.*

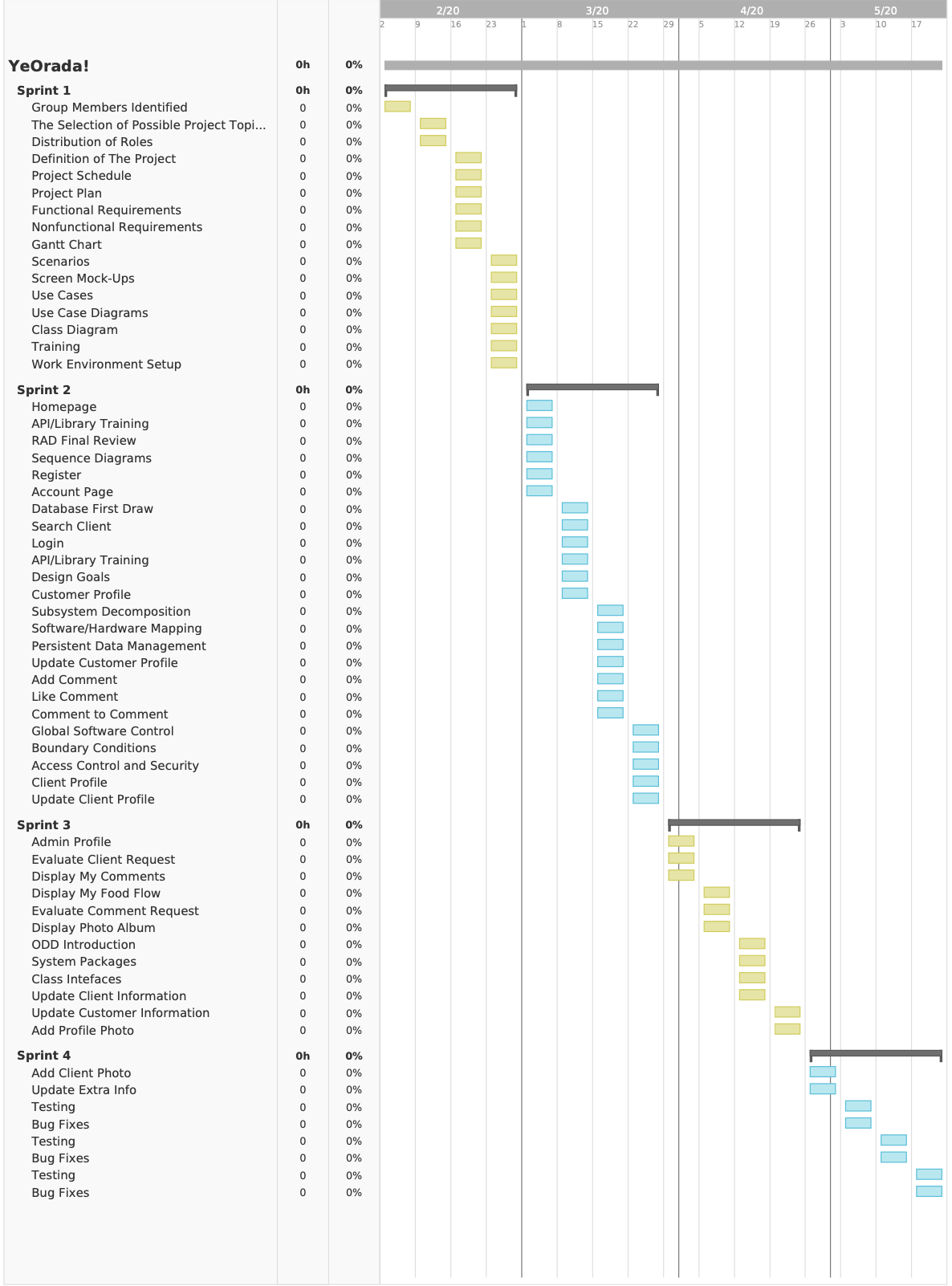
ekran görüntüsü içeren bir resim

Açıklama otomatik olarak oluşturuldu

*Fig 1.9: The screen mockup of the Customer Profile page.*

## Project Schedule

The project schedule is specified by using a Gantt Chart. You also can find the Gantt Chart as a .pdf document in the project file.



# Glossary

*Entity Objects*

**User:** All type of users. Registered and non-registered.

**Visitor:** Can visit to website without logging in to web site.

**RegisteredUser:** Registered users in the system. It has three types. These are Customer, Client and Admin.

**Client:** A restaurant account which controlled by its owner.

**Customer:** Customer account in the system.

**Admin:** Admin account in the system.

**Comment:** Comment is a opinion of the any customer that point out in a restaurant page.

**Photo:** Photo covers three different types of photo objects which are ProfilePhoto, ClientPhoto and MenuPhoto.

**ProfilePhoto:** It holds the profile photos of the Customers.

**ClientPhoto:**  It holds the photos of the Clients.

**MenuPhoto:** It holds the menu photos of the Clients.

**CommentPhoto:** Comment Photo is an image which is added to website when customer writes a comment.

**Request:** Requests made by users to the administrator. The requests are divided into two types as client request and comment request.

**Client Request:** Visitors sends client request to register the YeOrada as Clients.

**CommentRequest:** Comments of the customers must pass through from an approval. Their comments send as comment requests to the Admins and Admins may accept or decline them.

*Boundry Objects*

**NewClientForm:** It is a client form for admin to create a new client.

**RestaurantList:** It is a list of restaurants filtered with their names or locations.

**RestaurantProfile/ClientProfile:** It is the profile of the restaurants in the system which Customers may see and interact.

**AccountSettingsTab:** A page to change the information in the account.

**ChangePasswordTab:** A page to change the password of the account.

**CustomerProfilePage:** It is profile page of the customers which consits of their profile photo, comments and photo album.

**CustomerPhotos**: It is a pop-up to show photo albums.

**RequestList:** It is a list of comment requests and client registration requests.

**LoginPage:** It is a page provide login to website.

**HomePage:** The main page of the website.

**RegisterPage:** Customer registration form.

**GeneralInformationPanel:** Contains general information about the restaurant.

**ClientMenuAlbum:** It is a photo album which includes client's menu photos.

**AdminSettings:** It is a settings page to change admin’s account settings.

*Controllers*

**NewRestaurantController:** A controller which add a new client by the Admin to the website.

**SearchController:** A controller which creates the list of restaurants with a keyword or a location.

**SettingsController:** A controller which provides an editable form that includes user's information.

**ProfileController:** A controller which controls the update and display functions of the Customer’s profile page.

**RequestController:** A controller which provides to evaluate Comment and Client Registration request.

**LoginController:** A controller which provides to authenticate of the login request.

**RegisterController**: A controller which provides to create a registration form for a Visitor.

**ClientController:** A controller which provides to update a Client profile as an Admin.

# References

1. Bruegge B. & Dutoit A.H.. (2010). *Object-Oriented Software Engineering Using UML, Patterns, and Java*, Prentice Hall, 3rd ed.