Global Distributed Software Development

Master Team Project WS 2022/23

“ReserveEat”

Milestone 1 - Project Team 01 - Hochschule Fulda

December 07

Utsav Shrestha
Team Lead, Frontend Lead
utsav.shrestha@informatik.hs-fulda.de

Sanjay George
Backend lead, Devops
sanjay.varkey-george@informatik.hs-fulda.de

Omer Zogubi
Github master, Backend Member
omer.zogubi@informatik.hs-fulda.de

Muhammad Bilal
Frontend member
muhammad.bilal@informatik.hs-fulda.de

Parthiv Jani
Frondend Member, Document master
parthiv-yogesh-kumar.jani@informatik.hs-fulda.de

Kristijan Lazeski
Database admin, Backend member
kristijan.lazeski@informatik.hs-fulda.de

History Table (Revisions)

Date Submitted	Date Revised	Revision Summary

Contents

1 Executive Summary 3

1.1 Motivation 3
1.2 Functions and Services 3
1.3 Why should you use our Website 3
1.4 About us 3

2 Personae and main Use Cases 3

2.1 Personae 3
2.1.1 Administrator 3
2.1.2 Users 3
2.1.3 Restaurant Administrator 3
2.1.4 Restaurant Host 3
2.2 Main Use Cases 4
2.2.1 Account Registration 4
2.2.2 Managing Restaurant Details 4
2.2.3 Search and Reserve Restaurants 4
2.2.4 Reviews 4
2.2.5 Chat 4

3 List of main data items and entities 4

3.1 Registration 4
3.2 Restaurant 4
3.3 Roles 5
3.4 Reviews 5
3.5 Chat 5
3.6 Flag 5

4 Initial list of functional requirements 5

5 List of non-functional requirements 6

6 Competitive analysis 7

7 High-level system architecture and technologies used 7

8 Team and roles 8

9 Checklist 8

1 Executive Summary

1.1 Motivation

Your kid has Birthday soon and wants to organise something in a restaurant? Do you need a night out with friends? Wanna enjoy yourself with a nice dinner? Or just wanna go out and grab a drink? No problem, our Website **ReservEat** has got you covered. With **ReservEat** you can book a table in any restaurant in the area. See which ones are free, and choose the one that fits you the most. There is no more need to have a phone list of all the restaurants. One does not always update it and checks the new restaurants. Don't go and seek out restaurants, let the restaurants seek you out. Best fit for the best night.

1.2 Functions and Services

Our Websites offer a very Sophisticated and easy to use User Interface (UI) that also gives the User the feeling that they have everything in grasp. Hereby is the User Experience (UX) on a really high level. As a User one can Look and Search for the Restaurants around. After choosing one the user can book a number of seats for a given date and time. Everything presented in an easy to use Graphical User Interface (GUI). After booking the User shows up at the restaurant on the chosen day at the chosen Time.

Our website has an admin panel where a restaurant owner can manage their restaurants and update their details. The restaurant management can chat with users if they have some enquiries and flag them for attendance.

1.3 Why should you use our Website

ReservEat is an application that gives you the best experience to reserve a table in your favourite restaurant in a very smooth and simple way. We have made it easier for the users to communicate with us by introducing the chat functionality. The users can always message us about the enquiries they have. Every restaurant has their own chat service where the users can communicate with. The users can leave their reviews about the restaurants and one can have an idea of how a restaurant is.

1.4 About us

We are a Team of Motivated Master Students that have a goal to bring this problem not to be a problem anymore. We Utsav, Sanjay, Bilal, O¨ mer, Parthiv and Kristijan will continue to support and bring the Website to the masses and the life of everyone. We are developing this website for the netizens of Fulda to help them search for restaurants and make it easy for them to book a restaurant.

2 Personae and main Use Cases

2.1 Personae

2.1.1 Administrator

Administrator is the person who moderates all content on the application. The responsibility of the administrator includes approving restaurant listings, images, reviews as well as rejecting content that does not meet the required standards. An Administrator should be familiar with the flow of the software.

2.1.2 Users

Users are the end consumers of the application who search for restaurants and make reservations. Users can use the functionalities of the application to find a restaurant based on their needs and reserve seats at the place for their desired date and time. Additionally, they can connect with the restaurant host to clarify queries and concerns about the restaurant, cuisine or timings. Users can then choose to leave a review about their dining experience in the application, to help other users with their decision.

2.1.3 Restaurant Administrator

A restaurant administrator is a person who manages all data of a restaurant. It is the responsibility of the restaurant administrator to create an account for a restaurant and also authorise the restaurant host to access the reservations. Restaurant administrator adds all details about the restaurant including images and menu. They should be able to manage their hosts and synchronise with them.

2.1.4 Restaurant Host

A restaurant host has the access to view all reservations for a particular date and make necessary arrangements. The host will also receive messages from users and can clarify their queries through the chat. The host can also flag users in our application who did not show up without cancelling their reservation, which inadvertently causes loss of revenue for the restaurant. The restaurant host should have organising and communication skills to manage the customers.

2.2 Main Use Cases

2.2.1 Account Registration

All users of the application, as listed in the previous section, need to register an account to use the platform. Due to the nature and context in which the application is being built, only email ids affiliated with Hochschule Fulda and ending with ‘hs-fulda.de‘ will be allowed registration.

2.2.2 Managing Restaurant Details

A restaurant administrator can upload all details regarding his restaurant to be displayed to the users. These details will cover all basic data a user needs to make an informed decision about reserving a restaurant, such as the name of the restaurant, its location, items on the menu with price, images, etc.

2.2.3 Search and Reserve Restaurants

Users of the application can search for restaurants in a location and find all details about the restaurants available in the area. Users can reserve a restaurant for a desired date and time and also choose the required number of seats. Additionally, users can also modify or cancel their reservation. The Restaurant Host will be able to view all reservations of a day to plan and manage their customers.

2.2.4 Reviews

Users who have booked reservations through the application can leave reviews of their dining experience. The reviews can include text and images and will be useful for other users with

their decision in the future. The reviews also serve as a feedback for the restaurant and the Restaurant Host.

2.2.5 Chat

Users can chat with the Restaurant Host to clarify any queries that could range from doubts about a particular kind of cuisine to questions about organising events in the restaurant. Such a chat feature helps avoid miscommunication or misunderstandings about the services provided by the particular restaurant, thus contributing to a satisfactory experience for the user.

3 List of main data items and entities

The main data items and entities used in this project are:

3.1 Registration

A user needs to be registered and logged in to be able to reserve a restaurant. Similarly a restaurant admin should register themselves before they are able to list their restaurants in the system. The restaurant admin then can register their host/hostess in the system.

3.2 Restaurant

Restaurants are the main entity of our system. All the data about the restaurants are added and shown in the website so that a user can choose the restaurant according to their liking and reserve them.

3.3 Roles

The system has 4 roles that an actor can be assigned to. Each role has its own responsibility. The roles are listed below:

- **Users:** Searches for a restaurant and books it.
- **Restaurant Administrator:** Upload the information about the restaurant and manage the data.
- **Site Administrator:** Approves or rejects the changes made by the restaurant admins and Restaurant hosts.
- **Restaurant Host:** Reviews the reservations made by the users and manages the user activities.

3.4 Reviews

A user can post a review about a restaurant in the restaurant details page. They can rate by giving a number of stars and adding a review of how they felt about the restaurant. The review needs to be approved by system admin. The restaurant can flag the user reviews if they deem it inappropriate.

3.5 Chat

A user can chat about the more enquiries they have regarding the restaurants with the restaurant host. The chat will be available in each restaurant's details page.

3.6 Flag

A restaurant admin or host/hostess can flag a user or their reviews. The flagged user's reservation needs to be then approved by the restaurant management to finalise it.

4 Initial list of functional requirements

The list of the Functional requirements is as follows:

1. The Applications is a Vendor for Multiple Restaurants
2. User can Search for Restaurants
3. User can reserve a table and specify the number of seats in a given Restaurant
4. Users can Provide Reviews for the Restaurant which they have visited through our Application
5. Restaurants can be added, removed and updated by Restaurant owners
6. Restaurants can add Extra Services with themes through Application
7. There is a Grace Period to Cancel or update the booking of the table
8. Restaurants can change the availability status of any table at any time.
9. The number of booked seats are specified per table
10. Restaurant owners can Flag users for their lack of attendance on booking
11. Restaurant Owners can deny or approve of a booking
12. Admin can Ban Restaurants
13. User can choose Event Themes
14. User can only book two tables daily (to avoid spam reservations)
15. Users can view Menus of the Restaurants
16. User can View the Restaurant Reviews
17. Site administrator approves restaurant info for posting. Also deals with typical admin duties like managing user registrations etc.
18. Users search for restaurants, make/change/cancel reservations and also post reviews.
19. Host/hostess reviews daily calendar and checks and greets incoming guests This List might be Edited as the Project goes on.

5 List of non-functional requirements

Hereby are a High-Level Version of the non-functional specifications (how the app is delivered and other constraints) presented:

1. Application shall be developed, tested and deployed using tools and servers approved by Class CTO and as agreed in Milestone 0. Application delivery shall be from chosen cloud server
2. Application shall be optimised for standard desktop/laptop browsers e.g. must render correctly on the two latest versions of two major browsers

3. All or selected application functions must render well on mobile devices
4. Data shall be stored in the database on the team's deployment cloud server
5. Full resolution free media shall be downloadable directly, and full resolution media for selling shall be obtained after contacting the seller/owner
6. No more than 50 concurrent users shall be accessing the application at any time
7. Privacy of users shall be protected, and all privacy policies will be appropriately communicated to the users.
8. The language used shall be English (no localization needed)
9. Application shall be very easy to use and intuitive
10. Application should follow established architecture patterns
11. Application code and its repository shall be easy to inspect and maintain
12. Google analytics shall be used (optional for Fulda teams)
13. No email clients shall be allowed.
14. Pay functionality, if any (e.g. paying for goods and services) shall not be implemented nor simulated in UI.
15. Site security: basic best practices shall be applied (as covered in the class) for main data items
16. Application shall be media rich (images, video etc.). Media formats shall be standard as used in the market today
17. Modern SE processes and practices shall be used as specified in the class, including collaborative and continuous SW development
18. For code development and management, as well as documentation like formal milestones required in the class, each team shall use their own GitHub to be set-up by class instructors and started by each team during Milestone 0
19. The application UI (WWW and mobile) shall prominently display the following exact text on all pages "Fulda University of Applied Sciences Software Engineering Project, Fall 2022 For Demonstration Only" at the top of the WWW page. (Important to not confuse this with a real application).

6 Competitive analysis

Competitors / Feature Comparison	ReservEat	DinnerBooking	Chope	Table Agent	Dish Cult
Extra Services Selection	Yes	No	No	Yes	No
Chat with Restaurant	Yes	Yes	No	No	No
Flag User	Yes	No	No	No	No
Restaurant Menu	Yes	No	No	No	Yes
User Review	Yes	No	Yes	No	Yes

ReservEat reservation system is aimed to meet the needs of today's mobile generation. This system will allow users to make online table reservations at their favourite restaurants in a hassle free manner. Users can view the relevant restaurant menu, can select extra services, edit their reservation details and even chat with the restaurant host for detailed information. They can also share their experience with a restaurant by giving review and feedback about quality, service, pricing and environment. On the other hand, the restaurant host has authority to flag users according to their reservation history and behaviour.

7 High-level system architecture and technologies used

- **Hosting:** Microsoft Azure
- **Back-end Framework:** Express JS
- **Back-end Language:** JavaScript (Node JS)
- **Front-end Framework:** React JS
- **Front-end Language:** JavaScript
- **Code Editor:** Visual Studio Code
- **Target Web Browser:** Google Chrome, Firefox, Safari
- **Database:** MySQL
- **SSL Certificate:** Certbot

8 Team and roles

Name	Role	Email
Utsav Shrestha	Team Lead, Frontend Lead	utsav.shrestha@informatik.hs-fulda.de
Muhammad Bilal	Frontend Developer	muhammad.bilal@informatik.hs-fulda.de
Sanjay George	Backend Lead, DevOps	sanjay.varkey-george@informatik.hs-fulda.de
Omer Zogubi	Backend Developer, GitHub Admin	omer.zogubi@informatik.hs-fulda.de
Parthiv Jani	Frontend Developer, Document Master	parthiv-yogesh-kumar.jani@informatik.hs-fulda.de
Kristijan Lazeski	Backend Developer, Database admin	kristijan.lazeski@informatik.hs-fulda.de

9 Checklist

No.	Task	Status
1	Team found a time slot to meet (online) outside of the class	Done
2	GitHub master chosen	Done
3	Team decided and agreed together on using the listed SW tools and deployment server	Done
4	Team ready and able to use the chosen back and front end frameworks and those who need to learn are working on learning and practising	Done
5	Team lead ensured that all team members read the final M1 and agree/understand it before submission	Done

6	GitHub organised as discussed in class (e.g. master branch, development branch, folder for milestone documents etc)	Done