Click & Eat Application

Project Report

30th April 2021

Bachelor Of Science (Honours) Software Development

(Student) - Ana Griga
(Student Number) - C00231441
(Tutor) - Dr. Chris Meudec

Institiúid Teicneolaíochta Cheatharlach
INSTITUTE of TECHNOLOGY CARLOW
At the Heart of South Leinster

Declaration

I declare that this final project report titled “Click & Eat” has been written by me under the supervision of Dr. Christophe Meudec.

This work was not presented in any previous research for the award of bachelor degree to the best of my knowledge.

The work is entirely mine and I accept the full responsibility for any errors that might be found in the work, while the reference to publish materials have been duly acknowledged.

I have provided a complete table of reference of all works and sources used in the preparation of this document.

I understand that failure to conform with the Institute’s regulations governing plagiarism represent a serious offence.

Signature: Ana Griga Date: 25.04.2021

Ana Griga (Student)
C00231441 (Student Number)

Abstract

The aim of this project was to create an application dedicated to the restaurant industry that would be a valuable asset in times like these of pandemic, by simplifying the process of ordering and paying for meals while dining in a restaurant. The application was created with the needs of both customers and restaurants in mind.

Apart from providing consumers with a simple and low-interaction way to order and pay for their meals, it also saves time for the staff and lowers restaurant labor costs.

This final report describes the project in detail, the work achieved and problems encountered during development and the great learning outcomes gained.

Table of Contents

Table of Figures

1. Introduction

The Click & Eat application has been developed to fill a specific niche in the hospitality market by providing restaurants with the ability to offer their customers an ordering and payment system used while dining in the restaurant, which will eliminate the waiting time and reduce the interaction with the staff required these days due to Covid-19 pandemic.

This application would allow restaurants to increase their business profit by lowering labor costs and offering customers a better dining experience, bringing them more back. The system allows for simple and easy management of the menu updated in the customer module in real-time and can be browsed and ordered with a simple click. Restaurant staff then process the customer's orders efficiently using an intuitive user interface.

This document will describe the submitted project, outline the problems encountered during development, what was achieved and what was not yet achieved. It will also present the learning outcomes and what would have been done differently if started again. Any deviations from the original proposed design and specification, as well as any new information, will be included in the report.

2. Description of the Submitted Project

The project called Click & Eat is a web application designed for the restaurant industry to simplify the ordering and payment for the customers, to minimise the interaction with the staff due to the COVID-19 pandemic and to reduce the restaurant's staff labor and cost.

This application is going to replace the old and inefficient restaurant ordering and payment system.

The standard ordering method is inconvenient for both employees and consumers because it necessitates a great deal of manual labor. The manual work performed by the waiters can result in certain human mistakes, such as giving the wrong bill to the customers, the waiter's handwriting being unattractive, and the order sequence being incorrect. Customers would be dissatisfied with the restaurant as a result of all of these human mistakes. As a result, this restaurant ordering system was created to assist the restaurant in better management. The time it takes to place an order and pay for it has been cut in half thanks to this ordering and payment system.

The application is intended to be used by the restaurant's customers while in the restaurant and differs from applications like Just Eat which are for take away and delivery purposes. With this application, the customer will place and pay for their order using their mobile phone with few simple clicks.

The application has two modules, and there are three types of users that can utilize it, the customers, the restaurant's administrator and the restaurant's waiters.

2.1 Customer Module Features

- QR code scanning which will bring the customer to the restaurant's menu page
- Optional Registration if the customer wishes to create and account for future visits, to get information about offers or events in the restaurant or to get discounts for being a restaurant regular
- Optional Login for customers that hold an account
- Browsing the menu that contains all the information needed
- Creating an order with the choice of adding more items and special instructions

- Paying for the order
- Leave feedback for any menu item
- Customers that are logged in can logout

2.2 Restaurant Module Features

- Login option
- The administrator can create waiter’s accounts
- The administrator can CRUD the menu
- The administrator can see the orders
- The administrator can see the sales per day, week, month and year
- The administrator can logout
- The waiter can Login into the system
- The waiter can see the daily, weekly, monthly and yearly sales
- The waiter can see the orders
- The waiter can logout

2.3 Technologies used

Click & Eat is a single page application developed as an API server on the back end with C# and Asp.Net Core and the UI modeled using ReactJs. To store the data the application uses an SQL database.

2.3.1 API server

As a single page does not incorporate page refreshes, an web Api server was needed to be developed, that could be queried in real time. This was created using C# and Asp Net core framework.

The ASP.NET Web API is a platform for creating web services that use HTTP as their communication protocol. ASP.NET Web API may be used to return data based on the data that the client has requested, for example, JSON or XML as the data format.

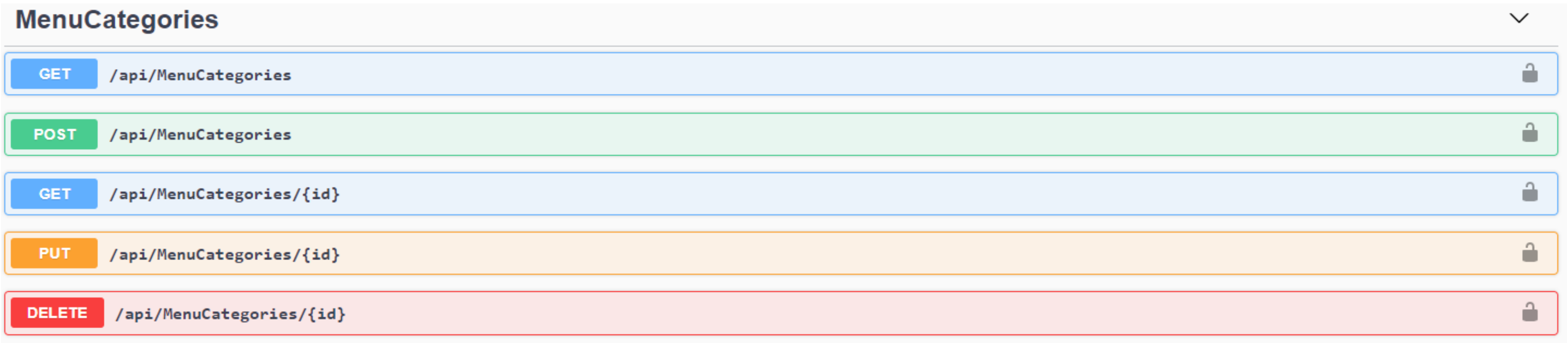
When opposed to ASP.NET MVC, routing in the ASP.NET Web API operates a little differently. The main difference between MVC and Web API routing is that Web API uses the HTTP method to pick the action rather than the URI route. A routing table is used by the Web API Framework to decide which action should be performed for each request (Packt, 2013).

The Click & Eat application's controllers are in charge of how the user communicates with the software. Simply put, a controller comprises the application's back end logic and determines what response will be sent to the user for each user's requests. Each method in the controllers is presented as an API documented using Swagger which is a JSON based interface description language for documenting API s.

Click & Eat API's expose the following HTTP methods:

- GET is used to retrieve data
- POST method is used to create data
- PUT is used to update or modify data
- DELETE method deletes the specified data

Click & Eat Menu Category example of an API with the HTTP methods documented with Swagger:



MenuCategories

GET /api/MenuCategories

POST /api/MenuCategories

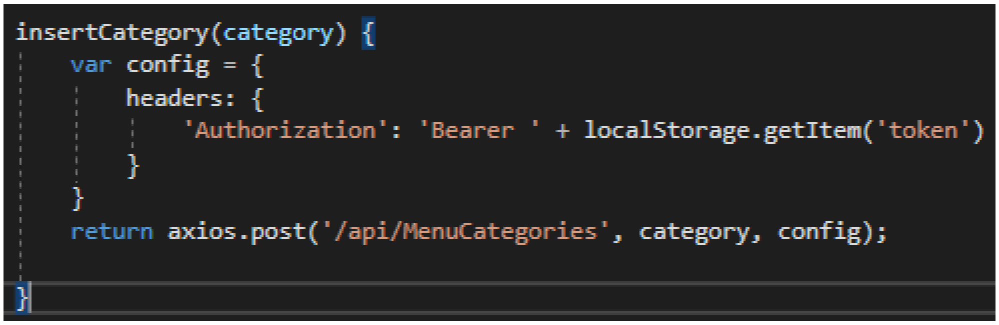
GET /api/MenuCategories/{id}

PUT /api/MenuCategories/{id}

DELETE /api/MenuCategories/{id}

Figure 1- API Example with HTTP methods
Source: Ana Griga, 2021

To retrieve data in the Click & Eat application from the APIs to be displayed on the page, the Axios library is used, which is designed to manipulate HTTP requests and responses asynchronously. Axios uses methods like get() and post(), which perform the HTTP GET and POST requests for retrieving or creating data. Below is an example code using the Axios post() method to create a new menu category by calling the MenuCategories API.



```
insertCategory(category) {
    var config = {
        headers: {
            'Authorization': 'Bearer ' + localStorage.getItem('token')
        }
    }
    return axios.post('/api/MenuCategories', category, config);

}
```

Figure 2- Axios code example
Source: Ana Griga, 2021

2.3.2 The UI with ReactJs

Being a user interface-oriented application, it was decided to use ReactJs library because it made a good fit for this type of application due to its component-oriented development, its popularity and great community and resources. ReactJs is designed as the view layer in the MVC framework, and it works well with the other services.

ReactJs is excellent because each element of the interface already built can be used anywhere else in the project.

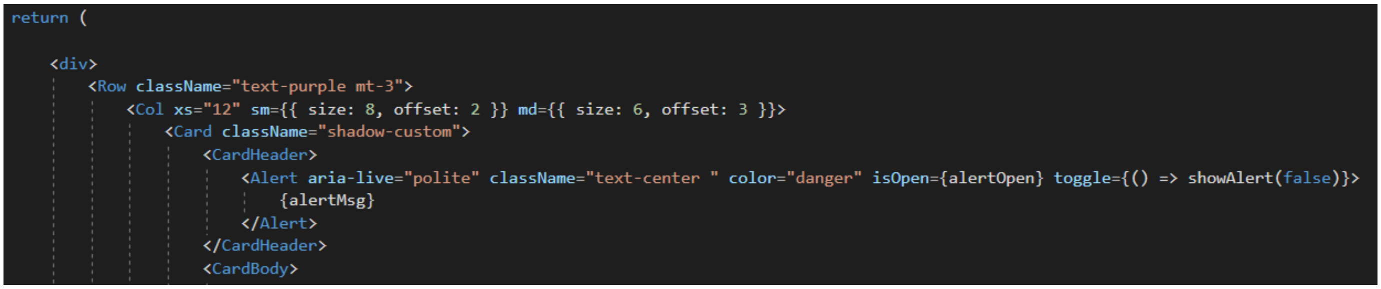
Also, because the application is built with React.JS, it can be easily ported to a React Native mobile app, and some source code can be reused.

ReactJs also allows the use of Bootstrap directly on elements and components for styling purpose

Some of ReactJS’s most important characteristics are JSX, Stateful components, Virtual DOM.

2.3.2.1 JSX

JavaScript XML (Chavan, 2021) is a syntax extension of JavaScript that makes it easier to write and add HTML in ReactJS. To convert JSX into a JavaScript that will be understood by the browser, Babel tool is used which is a JavaScript transpiler. Below is a snippet of a JSX syntax used in one of Click & Eat' s components.



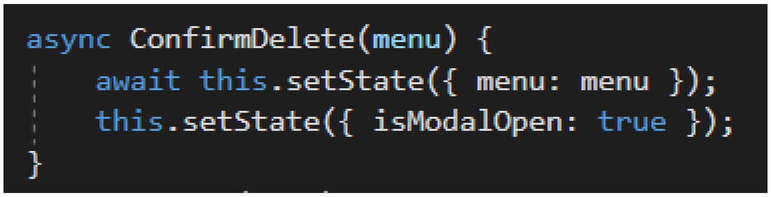
```
return (

    <div>
        <Row className="text-purple mt-3">
            <Col xs="12" sm={{ size: 8, offset: 2 }} md={{ size: 6, offset: 3 }}>
                <Card className="shadow-custom">
                    <CardHeader>
                        <Alert aria-live="polite" className="text-center " color="danger" isOpen={alertOpen} toggle={() => showAlert(false)}>
                            {alertMsg}
                        </Alert>
                    </CardHeader>
                    <CardBody>
```

Figure 3- JSX example code
Source: Ana Griga, 2021

2.3.2.2 Stateful components

React helped to break up the user interface into interchangeable React Components. React components have a render method that takes in data and outputs what should be seen. Each component has several lifecycle methods that can be overridden to run code at specific points in the process.

State is a simple JavaScript object for recording and reacting to user events. Each feature specified on the basis of a class has its own state object. When the state of a component is modified, the component and all of its child components are re-rendered immediately. States are used to store values in the component and can be passed down as props to child components.



```
async ConfirmDelete(menu) {
    await this.setState({ menu: menu });
    this.setState({ isModalOpen: true });
}
```

Figure 4- State example code
Source: Ana Griga, 2021

2.3.2.3 Virtual DOM

Since the HTML DOM was designed for static pages, it was not configured for creating dynamic user interfaces. When the DOM changes, it must update each element and re-paint the page with the new CSS and layout. It's not uncommon for a single page application to have many dynamically generated elements with event listeners. The HTML DOM must search for changes in any node data at regular intervals in dynamic pages. This has a notable impact on the application's performance. As a result of this inefficiency, the Virtual DOM was developed.

This application relies heavily on user interaction with the elements like navigation bars, buttons and these elements will have their states changed, and ReactJs manage the changing states easily and dynamically thanks to its Virtual DOM. A virtual DOM is a virtual copy of the DOM. Whenever a component in the application is updated, the Virtual DOM is also updated. The updated DOM is then compared against a real DOM using a reconciliation algorithm from ReactJS. This method is used to determine the smallest number of changes that should be made to the DOM. Rather than update dozens of elements whenever a change is made, ReactJS updates only a single element that was just modified, significantly cutting the time of the operation (Minnick, 2016).

2.3.3 SQL Database

Click & Eat application uses an SQL database to store all the data and the Microsoft SQL Server Management Studio tool. The data is modelled in the Model folder of the MVC model, and the tables are created in the database automatically using the “update-database” command in the console.

2.4 Authentication and Authorization

The application's three users each have their own set of privileges, such as a customer can only see the customer page, an administrator can only see the administrator page, and the waiter can only see the waiter page.

For the authentication and authorization, Json Web Token for securely transmitting information was used, and it was chosen as it works better with single-page applications such as Click & Eat because a session-based authentication requires cookies to be forwarded, which is not easy to do without page refreshes.

JWT is a type of token-based authentication. A token is passed for authentication for every request from a client to the server.

Token-based authentication allows users to receive a token that allows them to access a service and/or retrieve a particular resource without having to authenticate each request with their username and password. Stateless authentication is referred to as such because the token can be a self-contained object that conveys all of the necessary information for authenticating the request. In this scenario, the server does not need to keep track of a user's state.

The authenticating service generates the authentication token, which includes information to identify a specific user as well as the token's validity. To avoid interference, the token is cryptographically signed.

The token is used to define a security background for the client after it has been validated by the server, so that the service can make authorization decisions or audit operations for subsequent user requests (Gore, 2020).

2.5 The Payment Service

Braintree payment gateway was chosen for the application because it has great integration with the technologies used. It was found that the developer tools like sandbox testing cards it provides were very helpful.

The research included other payment gateways, but looking at PayPal, for example, it was found that the documentation was too complex and the Braintree documentation was precise and to the point.

Braintree sandbox testing environment is identical to the production environment, and it was useful for testing the payment of the application before going live. The developer account created allows running transactions using test credit card numbers, exploring this way the payment function implementation.

To integrate Braintree in the application, the Braintree.net SDK was installed for the server-side and the Braintree JavaScript V3 API for the client-side. The client-side API offers a secure custom Iframe for collecting sensitive data from the payment fields. Hosted Field solution is hosted by Braintree and gives the facility to customize the appearance of the form while ensuring that it is compliant with Payment Card Industry requirements (Braintree, 2021).

3. Description of Conformance to Specification and Design

3.1 The initial plan

The Click & Eat project's original idea was to create a single page web application for restaurants that would allow customers to order and pay through their mobile devices. It also included a restaurant module that allows the administrator to add, change, or remove menu items and display customer orders for both the administrator and the wait staff.

The application's structure, as shown in the specification document, consists of two modules: a customer part and a restaurant part, each with the following functionalities:

The customer module’s main functions:

- Order
- Pay
- Create account
- Give feedback

The restaurant module’s main functions:

- Login
- See orders
- CRUD menu
- See payment
- View customer
- View feedback
- CRUD Accounts

3.2 Work achieved

The project was developed in accordance with the specification and design documents, and the results will be summarized in this section.

3.2.1 Customer module

3.2.1.1 Menu

The application's customer page can be used on mobile devices, and it was designed to be responsive on any platform. The restaurant will have a QR code at each table for the customers, which will take them to the restaurant's menu page when scanned. If the consumer has an account, they can also log in without scanning the code.

The menu is beautifully laid out, with photographs of the dishes (as seen in the picture above) and the menu items organized by category.

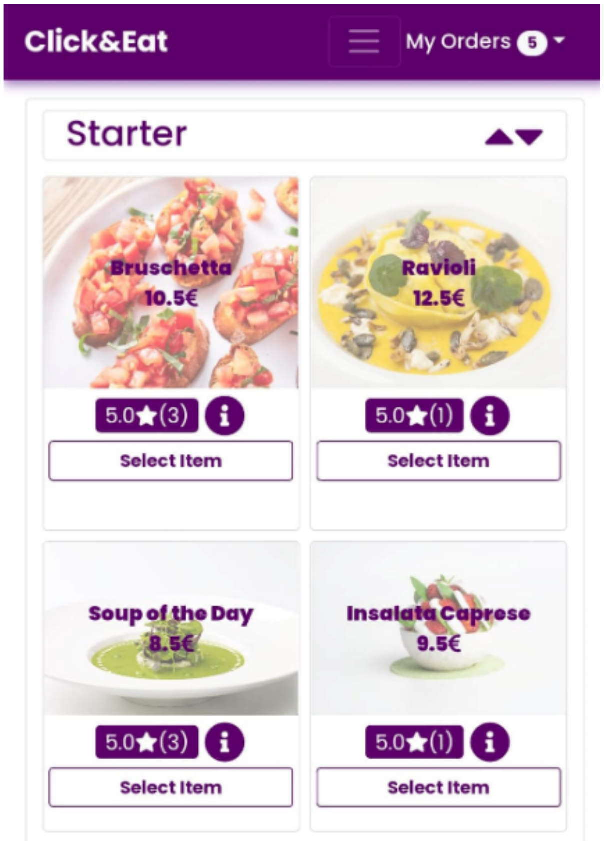
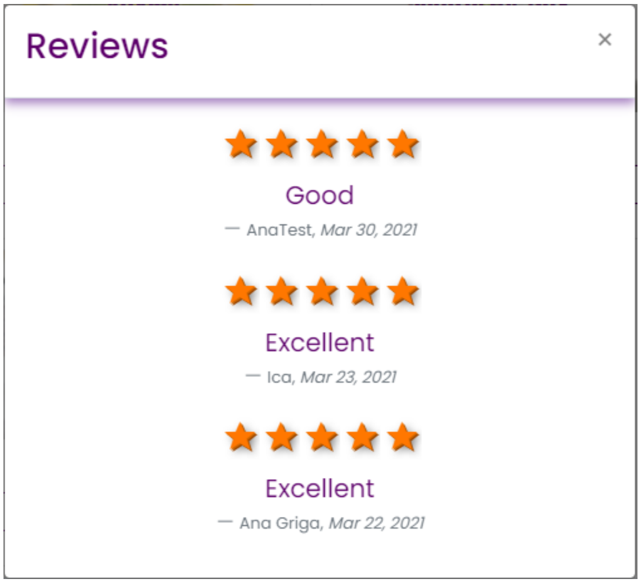


Figure 5- Menu Items Click & Eat
Source: Ana Griga, 2021

A feedback icon is also included on the menu item, which keeps the reviews for that particular item. Customers can see all of the feedback by clicking on it, as seen in the image below.



Reviews

★★★★★

Good

— AnaTest, *Mar 30, 2021*

★★★★★

Excellent

— Ica, *Mar 23, 2021*

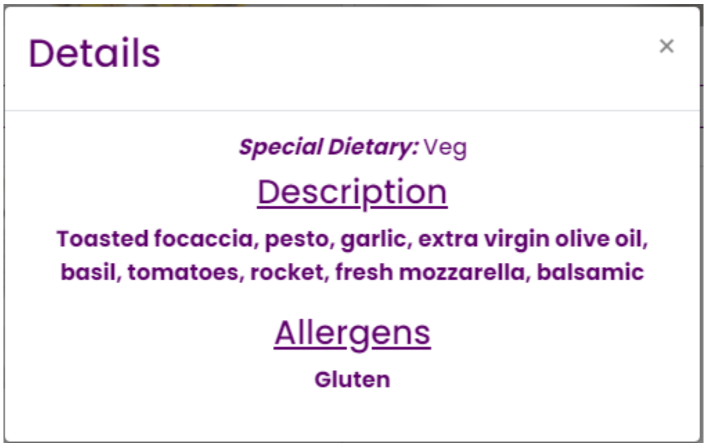
★★★★★

Excellent

— Ana Griga, *Mar 22, 2021*

Figure 6- Menu Item Reviews
Source: Ana Griga, 2021

The menu item's information is presented in a popup form when clicking on the info icon, which will reveal the details, including price, description, allergies, and special dietary options.



Details

Special Dietary: Veg

Description

Toasted focaccia, pesto, garlic, extra virgin olive oil, basil, tomatoes, rocket, fresh mozzarella, balsamic

Allergens

Gluten

Figure 7- Menu Items Details
Source: Ana Griga, 2021

The administrator creates the menu in the restaurant module, and it is modified in real time for the customer.

3.2.1.2 Order

The customers can easily place an order with the ease of a single click on the Select Item option on the menu.

The total number of items added to the order will be shown at the top of the navigation bar and in a separate bar labelled "View your order." This was introduced in response to users feedback that only the My Orders section of the navigation bar was not intuitive enough to see that the selected item was added to the order.

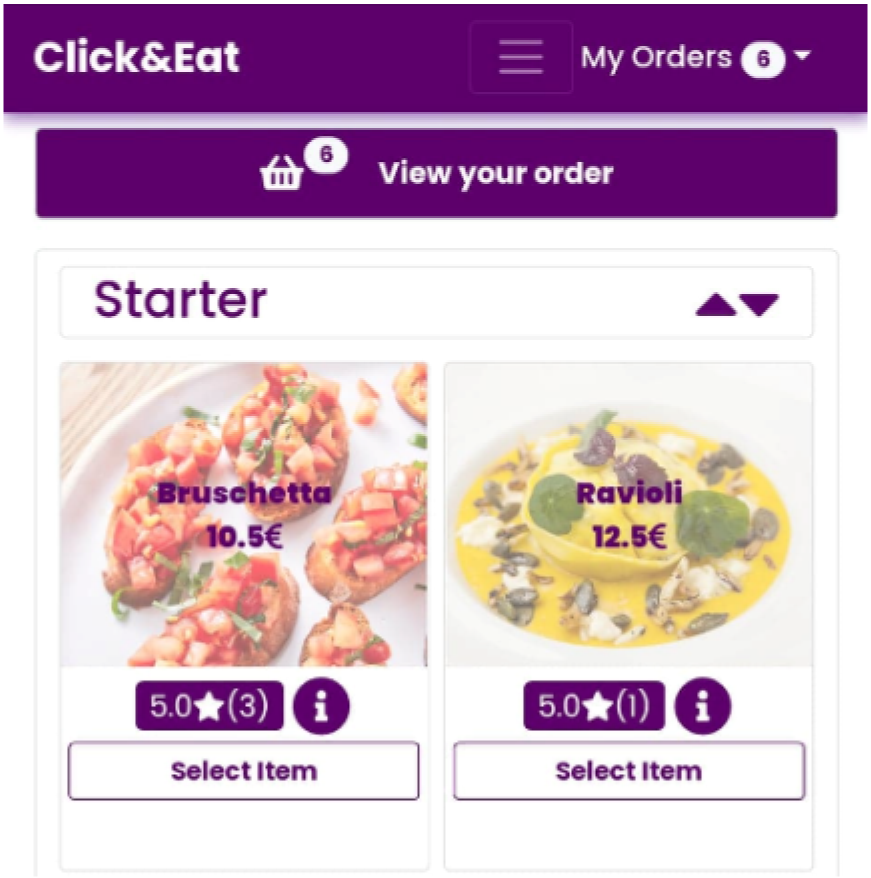
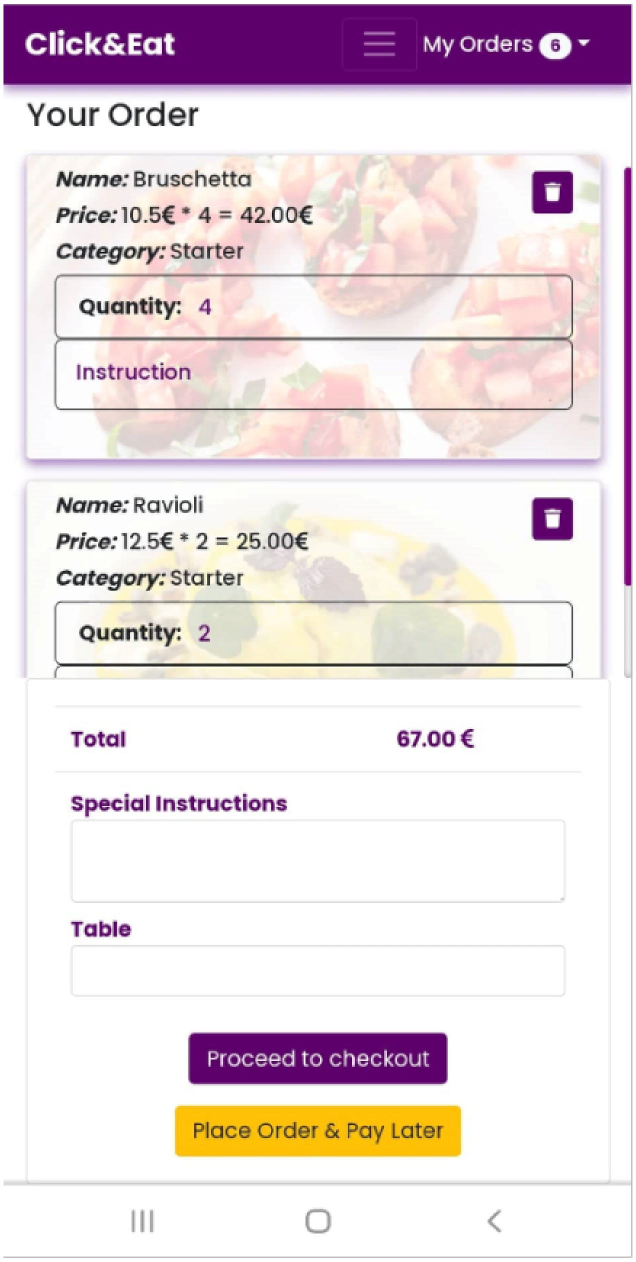


Figure 8 - Add to order
Source: Ana Griga, 2021

When the customer is ready to place their order, the menu choices are displayed, and the customer can increase, decrease the quantity or remove any item, and provide instruction for each item along with special instructions to better fulfil their order. The total per item and the overall total of the order, including the option to proceed to checkout or place the order and pay later, is displayed.



Your Order

Name: Bruschetta
Price: 10.5€ * 4 = 42.00€
Category: Starter
Quantity: 4
Instruction

Name: Ravioli
Price: 12.5€ * 2 = 25.00€
Category: Starter
Quantity: 2

Total **67.00 €**

Special Instructions

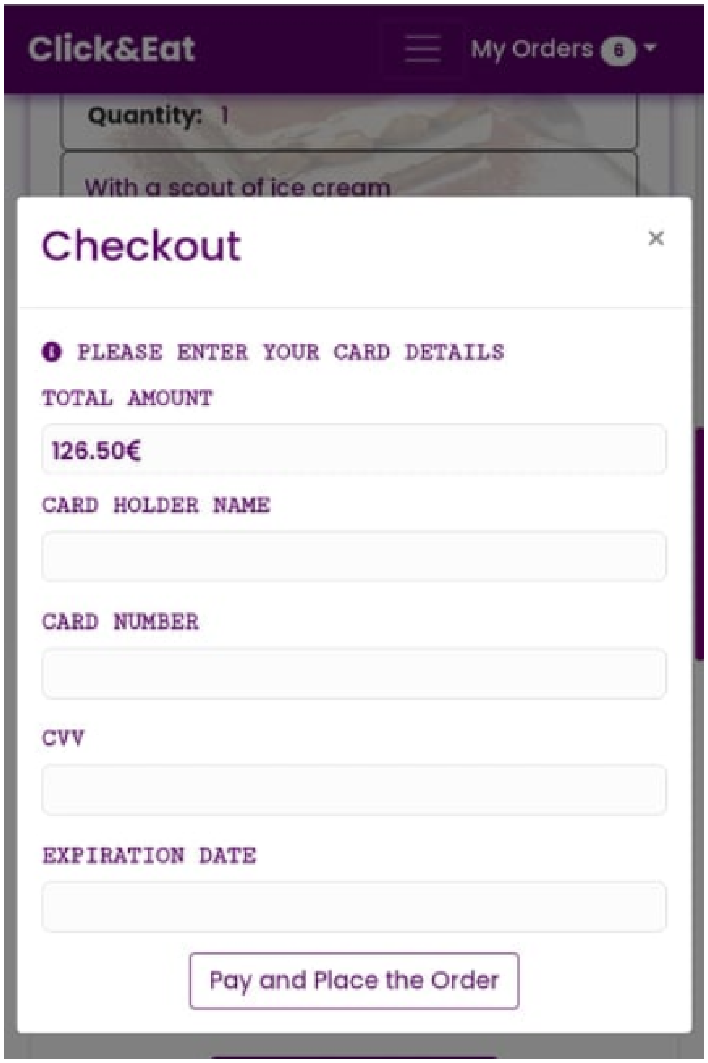
Table

Figure 9 - Your order
Source: Ana Griga, 2021

3.2.1.3 Payment

Once the customer is satisfied with their order, they can proceed to checkout.

At checkout, the customer is presented with an intuitive form to enter all the required payment information. The Braintree development environment is used to process payments. All payment information is safely processed by Braintree payment gateway.



Click&Eat

My Orders 6

Quantity: 1

With a scout of ice cream

Checkout

PLEASE ENTER YOUR CARD DETAILS

TOTAL AMOUNT

126.50€

CARD HOLDER NAME

CARD NUMBER

CVV

EXPIRATION DATE

Pay and Place the Order

Figure 10 - Payment form
Source: Ana Griga, 2021

3.2.1.4 Create account

For future visits, the customer has the optional choice of creating an account. The advantage of getting an account is that the restaurant can keep track of their visits, give them deals, and remind them of upcoming activities on the premises.

The registration is a very simple and intuitive form where the customer enters all the required information and clicks on the register button to easily create an account.

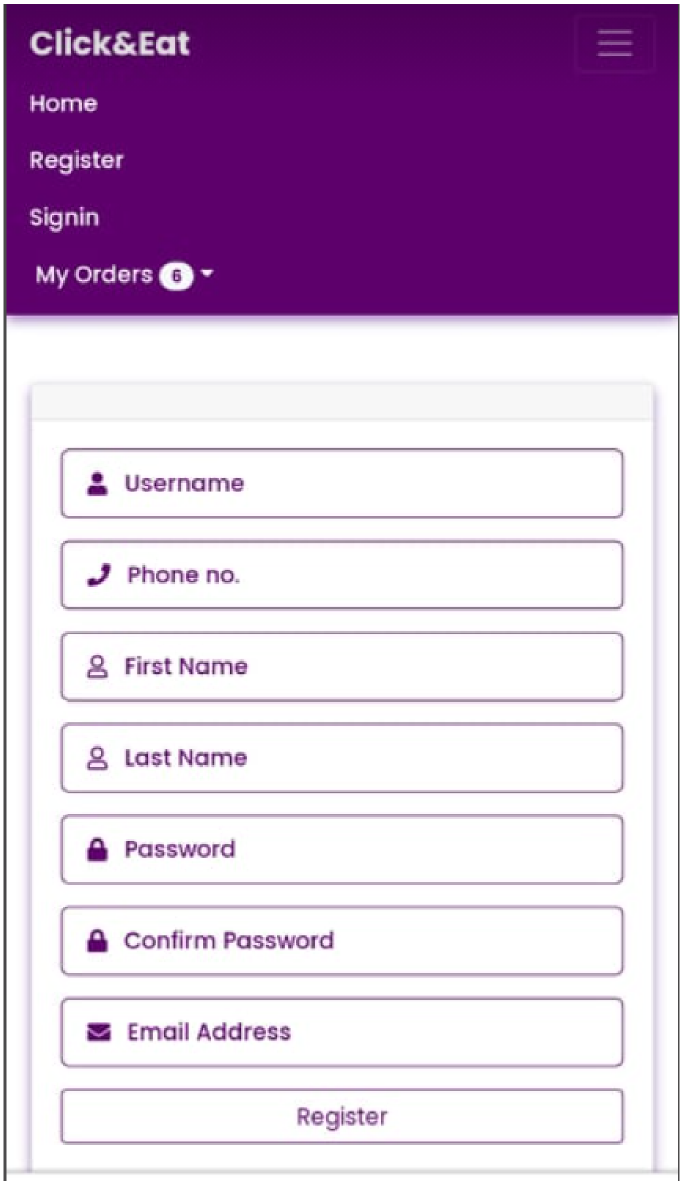


Figure 11 - Registration form
Source: Ana Griga, 2021

3.2.1.5 Give feedback

Once an order has been placed, this gets sent to the waiters and administrators, and when the order has been filled, they will mark it as served.

When the order has been served, the customers can optionally choose to leave feedback on the ordered items. This is done through a star rating form which will be displayed when the customer clicks on the Give Feedback option found on each menu item in the previous orders page.

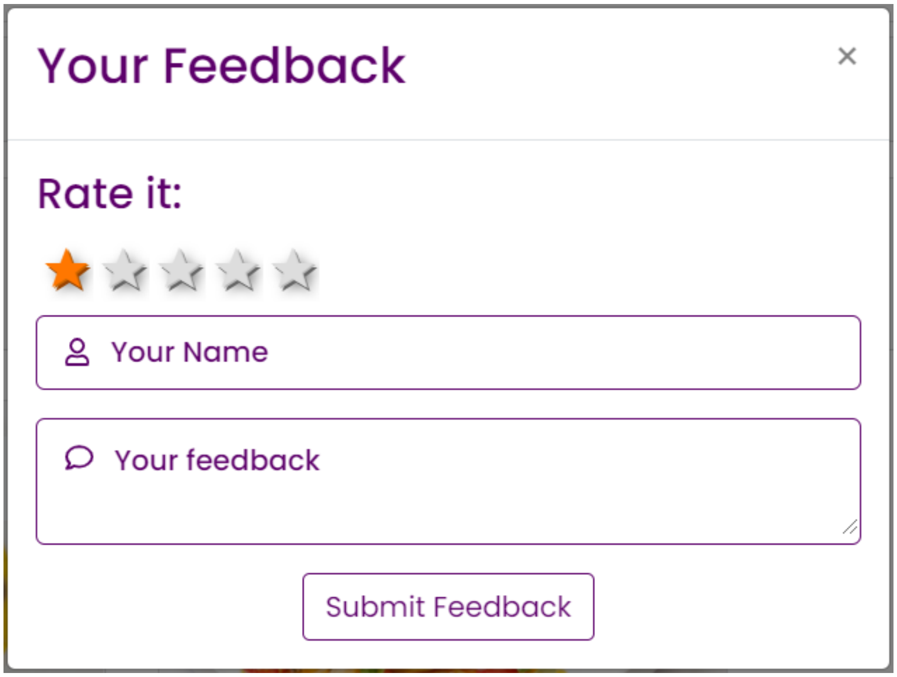


Figure 12 - Give feedback form
Source: Ana Griga, 2021

3.2.2 Restaurant module

The restaurant module has two users, the administrator and the waiter and each have privileges to see only their designated pages which have different functionalities presented below.

3.2.2.1 Login

The login form is going to be used by the administrator, the waiter and the customers which hold an account.

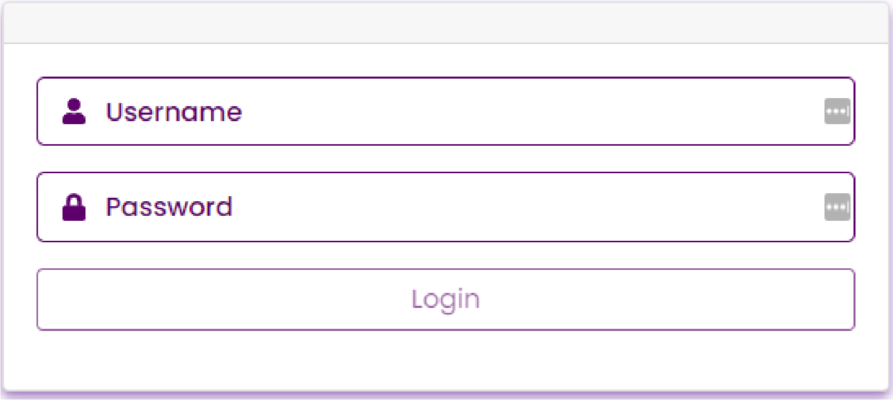
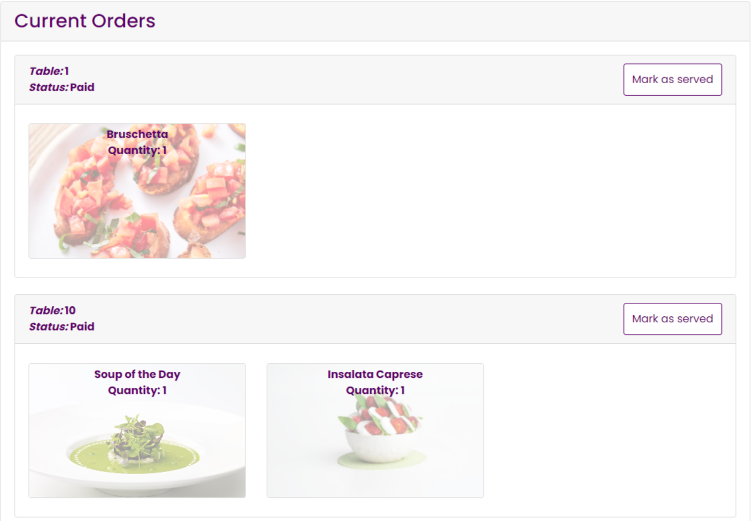


Figure 13 - Login
Source: Ana Griga, 2021

3.2.2.2 See order

This feature is common for both the administrator and the waiter. After the customer makes an order, the waiter and administrator will see the current orders with the table number, the payment status and the menu items in the order. When the order is being filled, the waiter or administrator can mark it as served with a simple click of a button.



Current Orders

Table: 1
Status: Paid

Bruschetta
Quantity: 1

Table: 10
Status: Paid

Soup of the Day
Quantity: 1

Insalata Caprese
Quantity: 1

Figure 14 - See orders
Source: Ana Griga, 2021

The waiter can access the orders on the navigation bar, and for the administrator, orders are placed under the Management drop-down menu in the navigation bar.

3.2.2.3 CRUD Menu

The administrator can add, edit and delete menu items.

To add a new item, the administrator is presented with a simple form where they can add all the information about the product along with a photo of the menu item. Every newly created item will be rendered on the customer page in real-time.

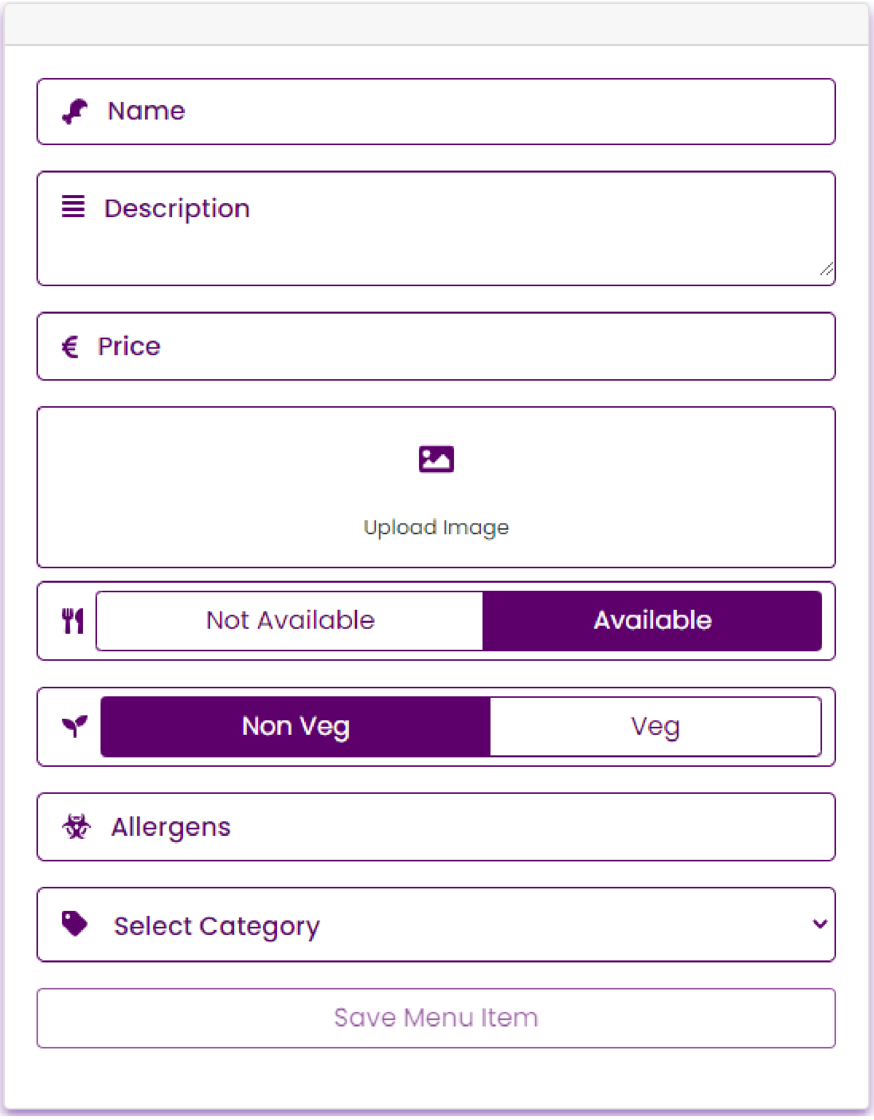
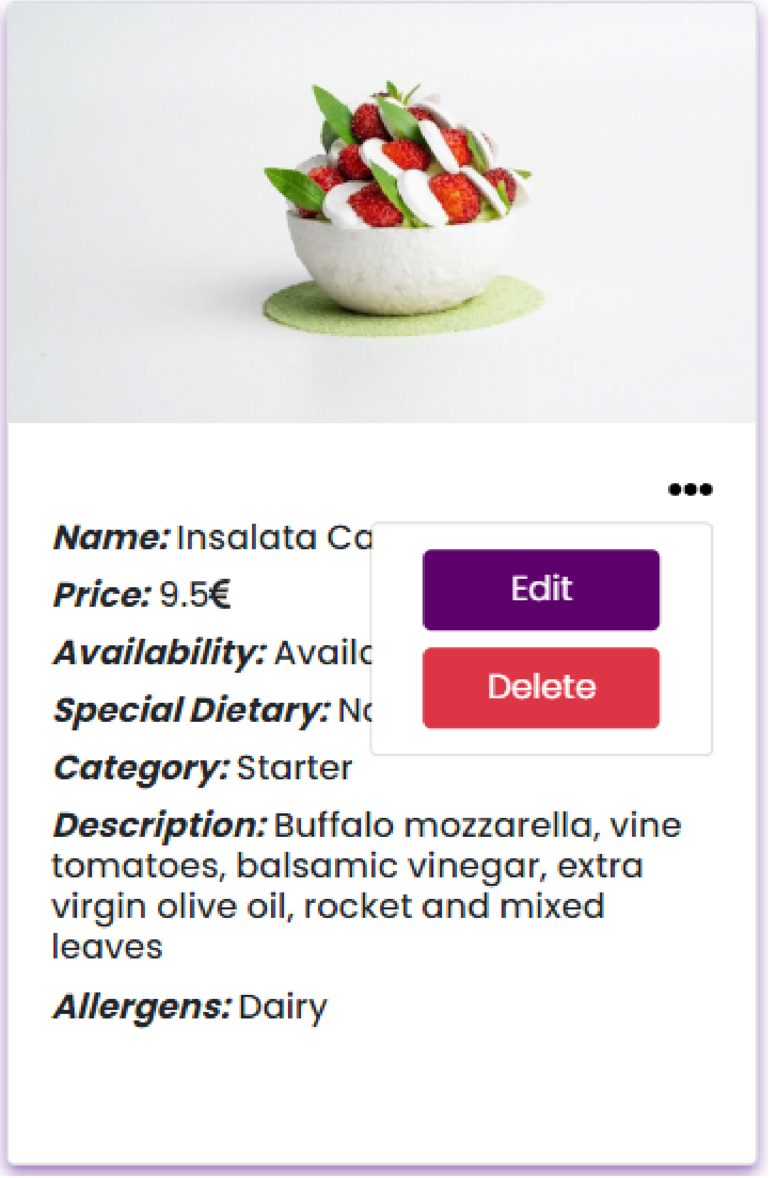


Figure 15 - Add menu item
Source: Ana Griga, 2021

Modification and deletion options are found on each menu item in the form of a three-dotted menu, which will reveal Edit and Delete options as seen in the Figure above.



Name: Insalata Ca

Price: 9.5€

Availability: Availa

Special Dietary: No

Category: Starter

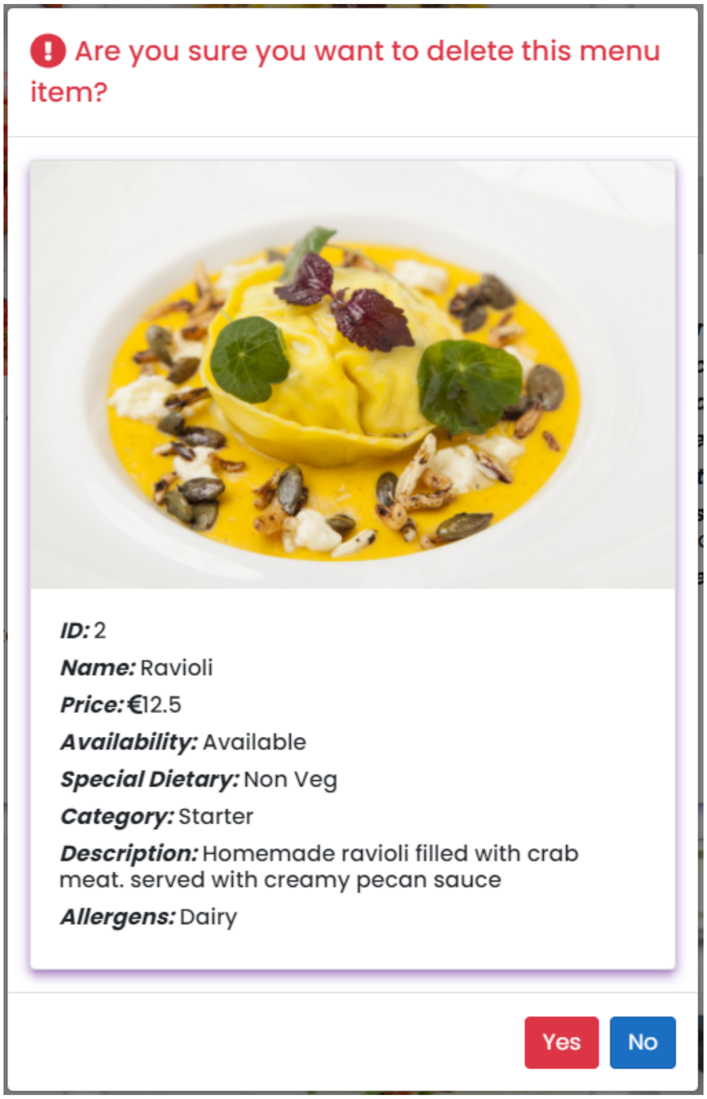
Description: Buffalo mozzarella, vine tomatoes, balsamic vinegar, extra virgin olive oil, rocket and mixed leaves

Allergens: Dairy

Figure 16 - Edit and Delete menu item
Source: Ana Griga, 2021

Edit option will show the Add menu item form presented above in Figure 13, just that all the fields are filled and can be modified.

The delete option will reveal the menu item with all the information and a confirmation message to make sure the administrator agrees to delete the product. It has two options for deletion, Yes if they want to proceed or No if they decide not to delete the product.



Are you sure you want to delete this menu item?

ID: 2
Name: Ravioli
Price: €12.5
Availability: Available
Special Dietary: Non Veg
Category: Starter
Description: Homemade ravioli filled with crab meat. served with creamy pecan sauce
Allergens: Dairy

Figure 17 - Edit and Delete menu item
Source: Ana Griga, 2021

3.2.2.4 See payment

Both the waiter and the administrator would need to know if the order has been paid, and they can do so by going to the order option, which shows the table number as well as the order's payment status.

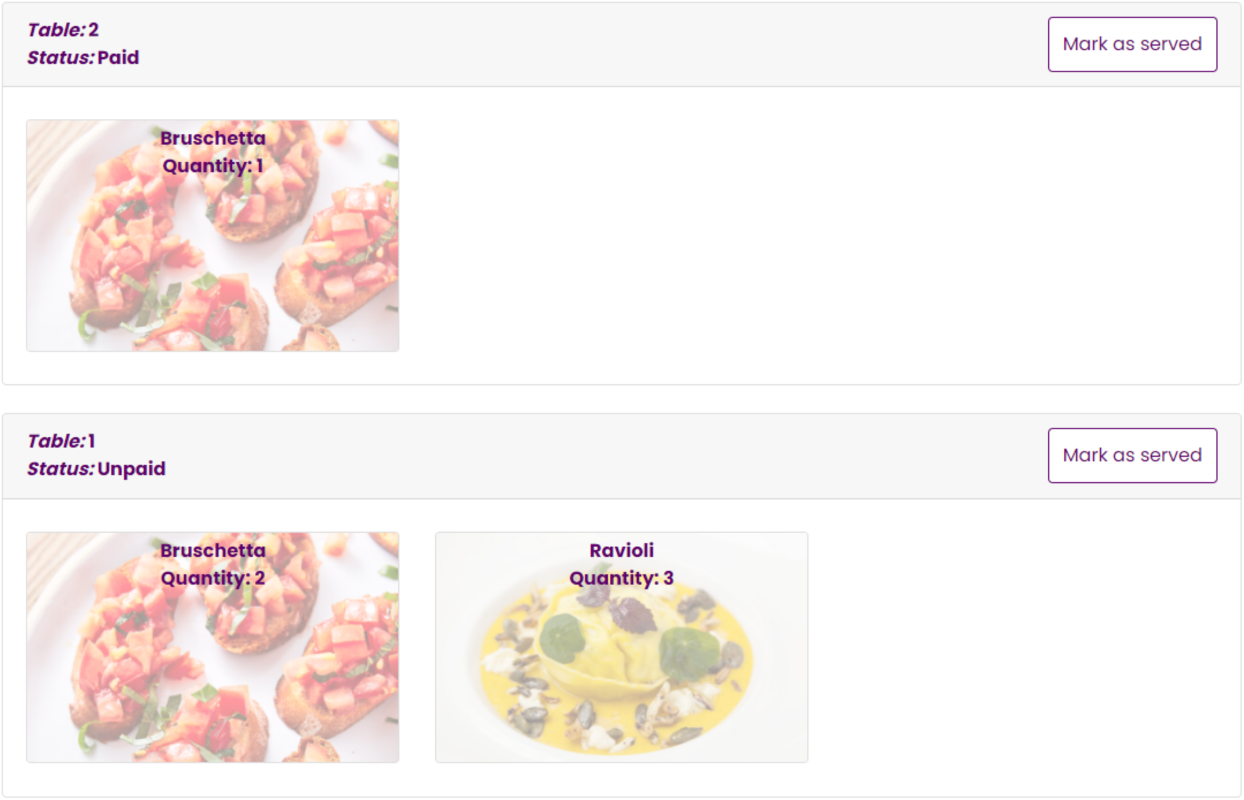


Table: 2
Status: Paid

Bruschetta
Quantity: 1

Table: 1
Status: Unpaid

Bruschetta
Quantity: 2

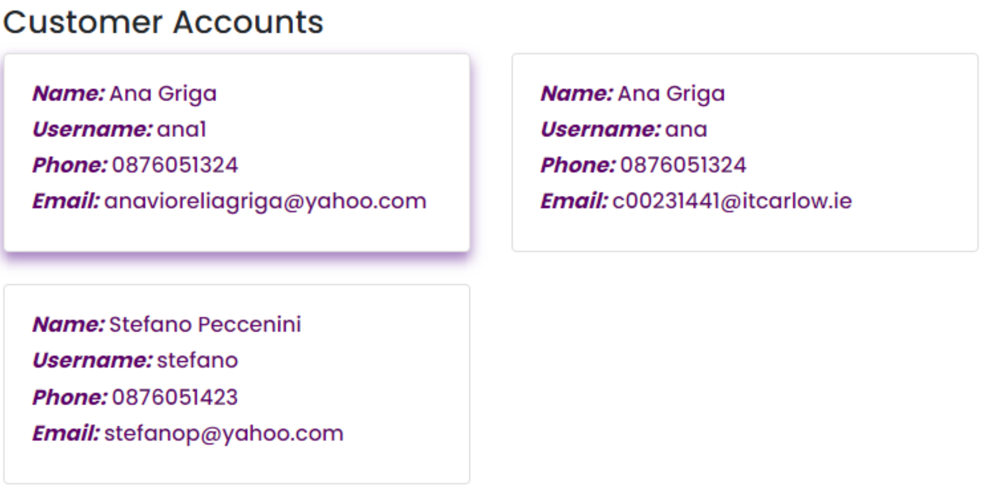
Ravioli
Quantity: 3

Figure 18 - See payment
Source: Ana Griga, 2021

3.2.2.5 View customer

The administrator has access to all of the customers who have created accounts, as well as all of their personal information. As previously stated, getting the customer's details allows the restaurant to send marketing promotions, identify loyal customers for special deals and bonuses, and keep them informed about events taking place on their premises.

The administrator can see the customer's account accessing the Accounts page situated on the navigation bar.



Customer Accounts

Name: Ana Griga
Username: ana1
Phone: 0876051324
Email: anavioreliagriga@yahoo.com

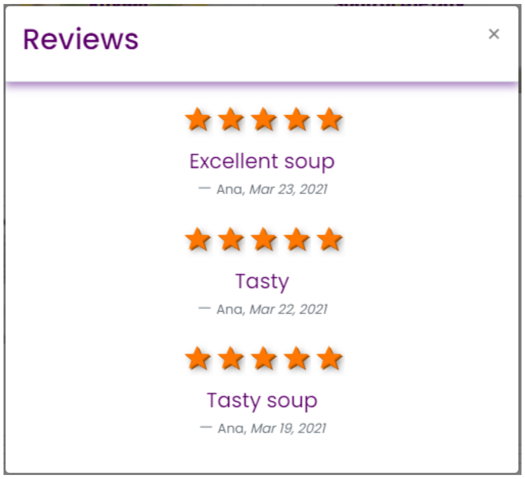
Name: Ana Griga
Username: ana
Phone: 0876051324
Email: c00231441@itcarlow.ie

Name: Stefano Peccenini
Username: stefano
Phone: 0876051423
Email: stefanop@yahoo.com

Figure 19 - See customers
Source: Ana Griga, 2021

3.2.2.6 View customer feedback

The administrator and waiter have access to all of the customer reviews on each of the restaurant's menu items. Customer's feedback is very important for the restaurant as they can tailor the menu according to the customer's preferences.



Reviews ×

★★★★★
Excellent soup
— Ana, *Mar 23, 2021*

★★★★★
Tasty
— Ana, *Mar 22, 2021*

★★★★★
Tasty soup
— Ana, *Mar 19, 2021*

Figure 20 - See customers
Source: Ana Griga, 2021

All of the functions in the Specification document that were defined at the start of the project were completed, and all were described in this section.

3.2.2.7 CRUD Accounts

The administrator is responsible for creating waiter accounts. They can create new accounts and delete existing ones. This feature is accessible via the Accounts tab on the navigation bar.

The form for registering a waiter account, as well as a list of all waiter accounts, are provided below.

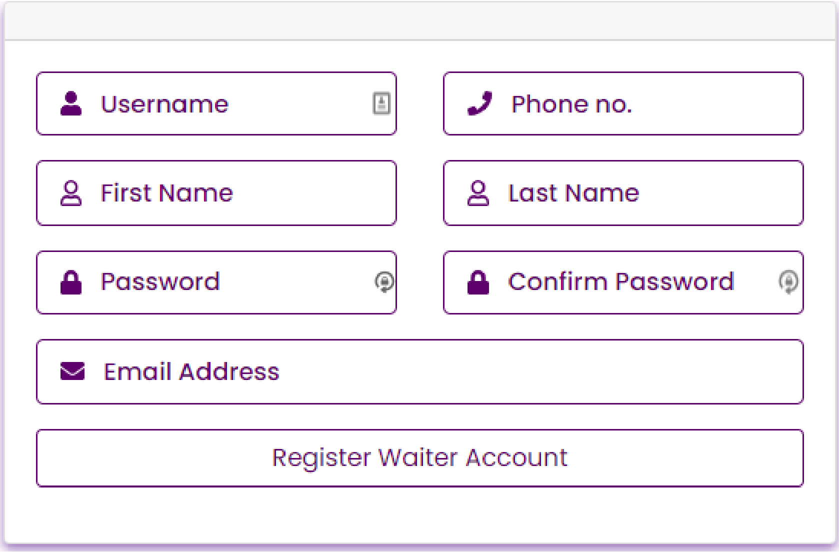
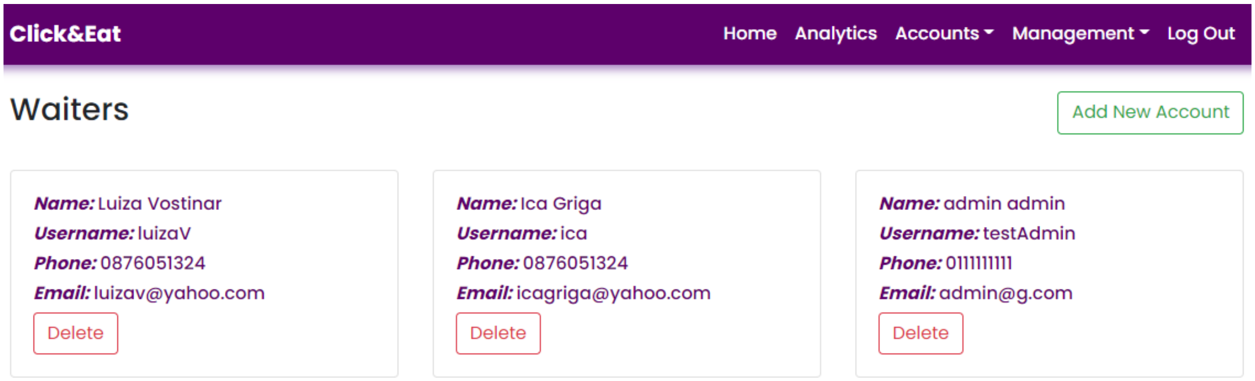


Figure 21 - Register waiter form
Source: Ana Griga, 2021



Waiters

Name: Luiza Vostinar
Username: luizaV
Phone: 0876051324
Email: luizav@yahoo.com

Name: Ica Griga
Username: ica
Phone: 0876051324
Email: icagriga@yahoo.com

Name: admin admin
Username: testAdmin
Phone: 0111111111
Email: admin@g.com

Figure 22 -Waiter accounts
Source: Ana Griga, 2021

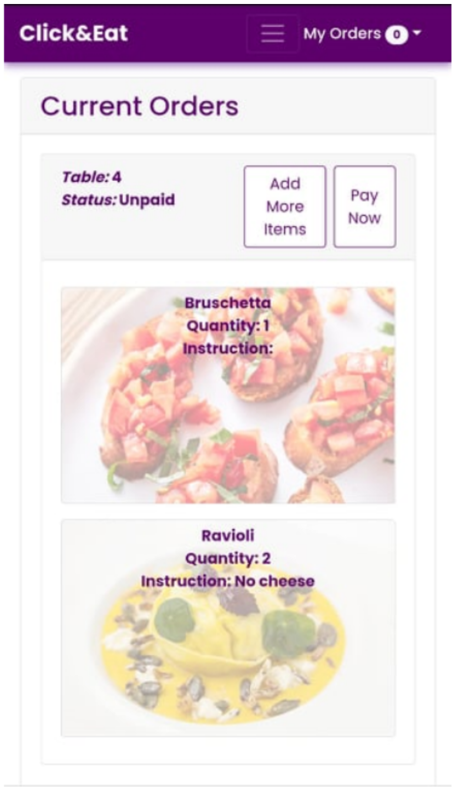
3.3 Additional features

Besides the features stipulated in the specification document when the project began, new features were introduced. Some were required because others relied on them, while others were included because they were thought to be extremely helpful to the application. This section will go over all of the new functions in detail.

3.3.1 Place order and pay later feature

This function was introduced to allow customers to add more products to their order after they had already placed their initial order. There are several occasions where a customer changes their mind and wants to add more to their order, and Click & Eat can provide this choice to their customers.

After the order was created, the customer will be guided to the payment form if they want to proceed to checkout. The "Place Order & Pay Later" choice forwards the customer to the current order page, where they can add more products or pay. The order's status will be set to unpaid, as seen in the Figure below.



Current Orders

Table: 4
Status: Unpaid

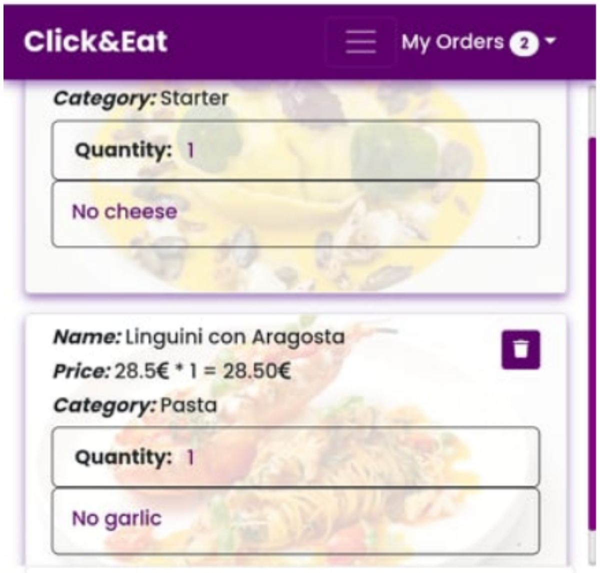
Bruschetta
Quantity: 1
Instruction:

Ravioli
Quantity: 2
Instruction: No cheese

Figure 23 - Place order and pay later
Source: Ana Griga, 2021

3.3.2 Item instructions

From feedback, it was suggested that adding instructions per item, in addition to the special instructions for the order, was a great idea in case the customer had special requirements for some of the items.



Category: Starter

Quantity: 1

No cheese

Name: Linguini con Aragosta

Price: 28.5€ * 1 = 28.50€

Category: Pasta

Quantity: 1

No garlic

Figure 24 - Instruction per items
Source: Ana Griga, 2021

3.3.3 Create account at the end of the meal

Since it was previously discussed how important it is for restaurants to retain customer details for marketing purposes, in addition to the Register option on the navigation bar, this option was added at the end of the meal, after the customer had reviewed an item. This will give him another chance to register with the system, having an account being beneficial for both the customer and the restaurant.

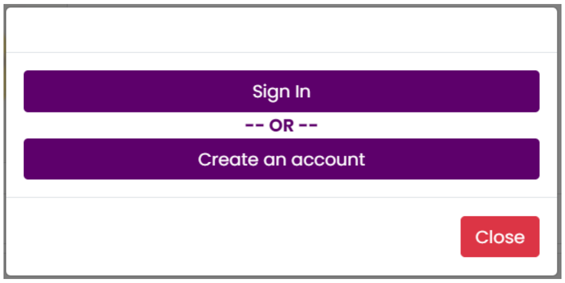


Figure 25 - Create account
Source: Ana Griga, 2021

3.3.4 Menu Category

Because it was not anticipated at the start of the project that the menu items would be divided into categories, this new feature was necessary to be added even if it was not mentioned in the specification document.

This functionality is only available to administrators and can be found on the navigation bar under the Management tab. The administrator must first create categories, followed by as many menu items as the restaurant has that correspond to a specific menu category. The feature allows the administrator to create, edit and delete menu categories.

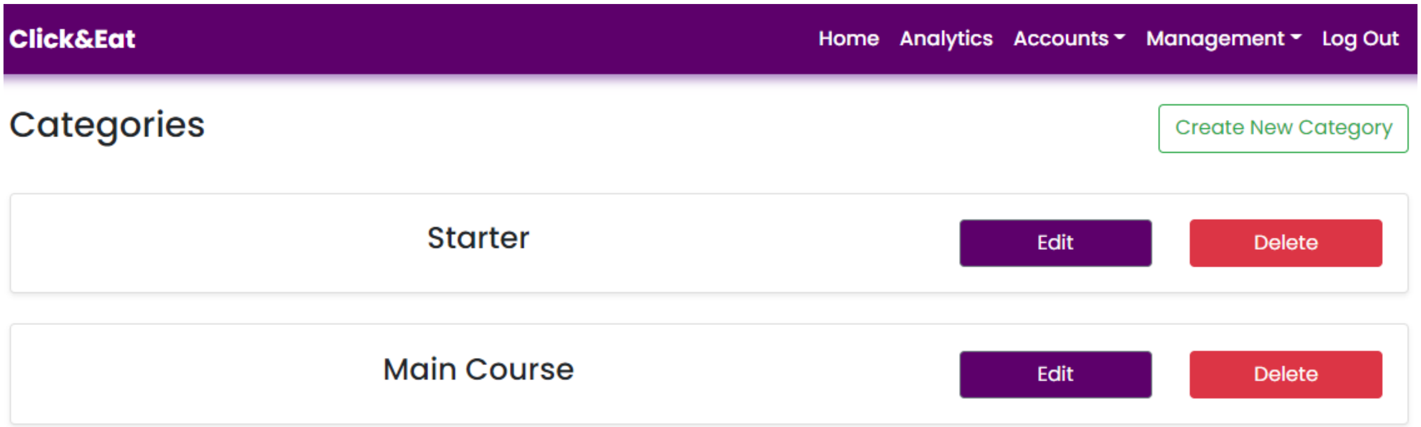


Figure 26 - Menu category
Source: Ana Griga, 2021

The form for adding a new menu category is provided below.

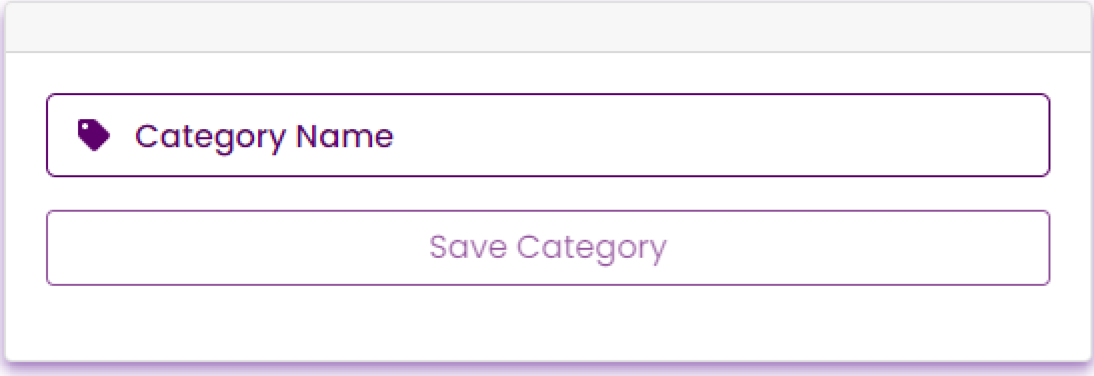
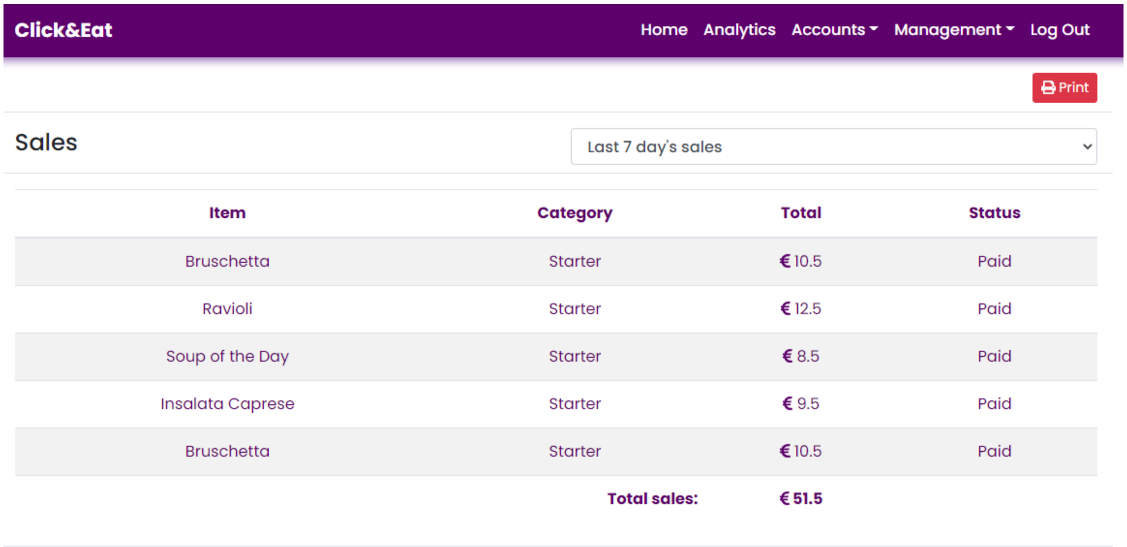


Figure 27 - Create new category
Source: Ana Griga, 2021

3.3.5 Analytics

Following feedback from the hospitality industry, it was determined that it was important for the administrator to be able to see the restaurant's sales, so this functionality was created to allow the administrator to see daily, weekly, monthly, and yearly sales in the form of a table that can be exported as a pdf document. The option is available as a tab on the navigation bar called Analytics.



Sales

Last 7 day's sales

Item	Category	Total	Status
Bruschetta	Starter	€ 10.5	Paid
Ravioli	Starter	€ 12.5	Paid
Soup of the Day	Starter	€ 8.5	Paid
Insalata Caprese	Starter	€ 9.5	Paid
Bruschetta	Starter	€ 10.5	Paid
	Total sales:	€ 51.5	

Figure 28 - Analytics
Source: Ana Griga, 2021

3.3.6 Change password

Even though it was not specified in the initial phase of gathering the project's specifications, a change password function was required because the application contains account creation features.

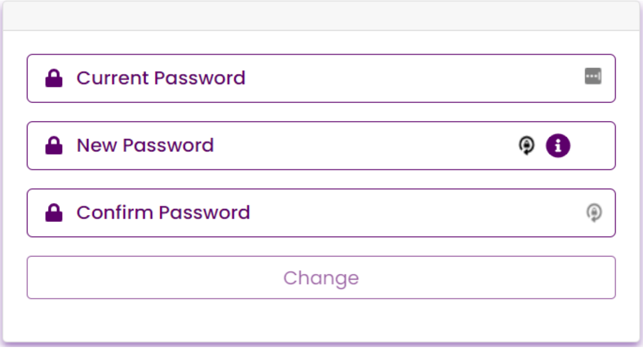


Figure 29 - Change password
Source: Ana Griga, 2021

3.4 Milestones

A milestone (Smith, 2020) is a particular point in a project's life cycle used to monitor the project's progress toward its end goal. Milestones are used as signal posts for essential events, decision points, or deliverables in project management, such as the project's start date or the project's deadline.

Setting up milestones from the beginning of the project helped in productivity by recognizing and working towards specific goals. The milestones were used in every step of the project and gave an understanding of what tasks will lead the work in the right direction to achieve the project's standards and deadlines.

For developing this project, each delivery and implementation on essential features were considered milestones and demonstrated to the supervisor.

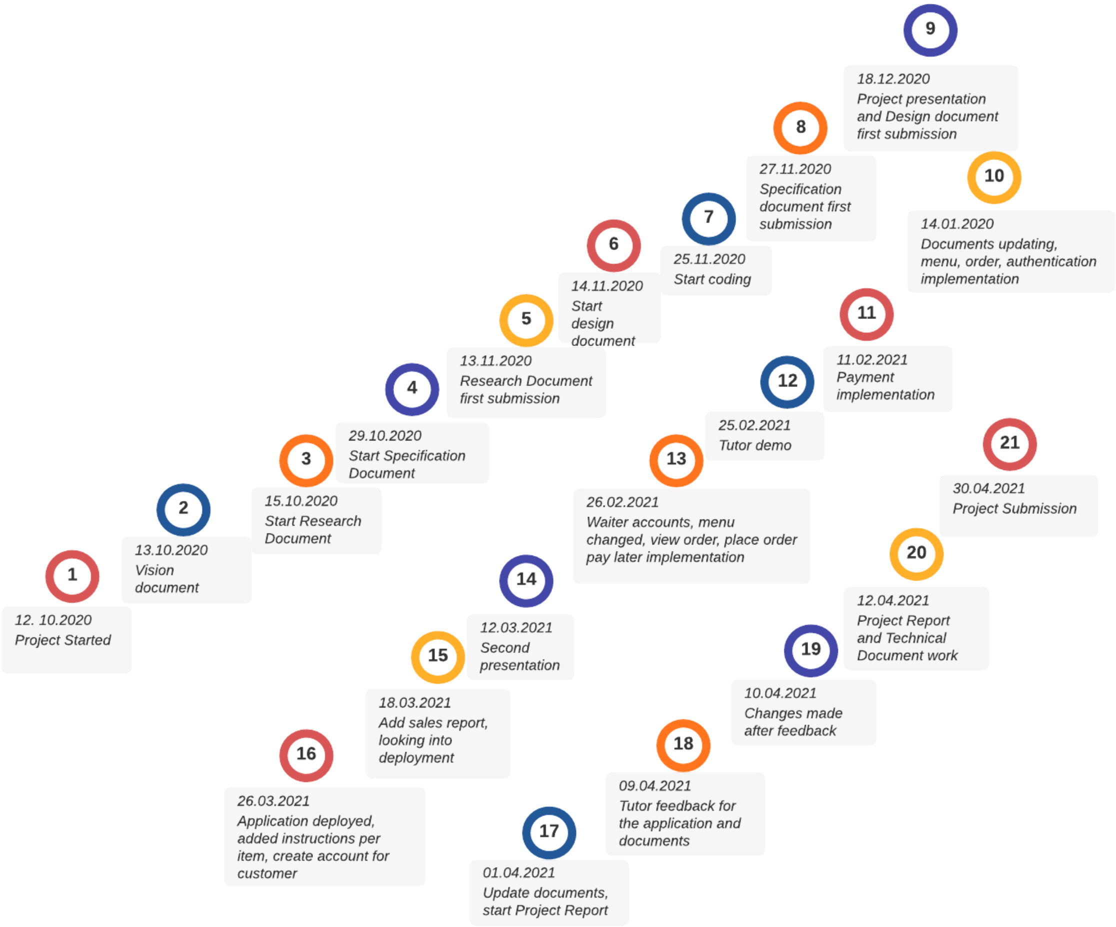

1
12. 10.2020
Project Started
2
13.10.2020
Vision document
3
15.10.2020
Start Research Document
4
29.10.2020
Start Specification Document
5
13.11.2020
Research Document first submission
6
14.11.2020
Start design document
7
25.11.2020
Start coding
8
27.11.2020
Specification document first submission
9
18.12.2020
Project presentation and Design document first submission
10
14.01.2020
Documents updating, menu, order, authentication implementation
11
11.02.2021
Payment implementation
12
25.02.2021
Tutor demo
13
26.02.2021
Waiter accounts, menu changed, view order, place order pay later implementation
14
12.03.2021
Second presentation
15
18.03.2021
Add sales report, looking into deployment
16
26.03.2021
Application deployed, added instructions per item, create account for customer
17
01.04.2021
Update documents, start Project Report
18
09.04.2021
Tutor feedback for the application and documents
19
10.04.2021
Changes made after feedback
20
12.04.2021
Project Report and Technical Document work
21
30.04.2021
Project Submission

Figure 30 - Project Milestones
Source: Ana Griga, 2021

As seen in the Figure above, the following are the project's milestones:

- 12.10.2020 Project started
- 13.10.2020 Vision document
- 15.10.2020 Start research and the research document
- 29.10.2020 Start specification document
- 13.11.2020 Research document first submission
- 14.11.2020 Start design document
- 25.11.2020 Start coding
- 27.11.2020 Specification document first submission
- 18.12.2020 Project presentation and design document first submission
- 22.12.2020 Updating the documentation, menu, order, authentication implementation
- 11.02.2021 Payment implementation and documented in research
- 25.02.2021 Tutor demonstration of the work
- 26.02.2021 Waiter accounts, costumer menu changed, view order implementation, place order and pay later
- 12.03.2021 Second project presentation
- 18.03.2021 Add sales report for admin and looking into deployment
- 26.03.2021 Application deployed, added instructions per item, create account for customers
- 01.04.2021 Updating specification, design document and start Project Report
- 09.04.2021 Tutor feedback for the application and documents
- 10.04.2021 Follow up the tutor's feedback and make changes accordingly
- 12.04.2021 Working on the Project report and the Technical document
- 30.04.2021 Project submission

3.5 Problems encountered

Software development necessitates a delicate balance of planning and structure, as well as experimentation and imagination. If this balance is recognized, managing and developing software projects can become easier and safer.

The Click & Eat project followed the agile methodology which helped in meeting deadlines by assigning work frames for research, implementing new features, getting feedback and following up on the feedback, updating documents.

There were challenges and problems encountered along the way, as with any project, and particularly because it was the first individual complex project. The main difficulties found will be discussed in the following section.

3.5.1 Authentication and Authorization

The most challenging aspect was API protection with JWT described in the research and the design document, which had limited documentation in the Microsoft.net Core resources. The research was difficult and time-consuming, and it also took time to find out how to implement it into the application.

3.5.2 UI development

A great UI design can turn potential visitors into buyers due to the excellent interaction with the application. With this in mind, a lot of effort was put into creating a great UI that will satisfy all types of customers.

A good and powerful image choice can dramatically increase the emotion and create a feeling of connection and willingness to order more. Adding beautiful photos for each menu item can increase the restaurant's sales because of the prevalent saying "we eat with our eyes".

The photos were intentionally transparently reshaped to add a note of curiosity, making the customer even more eager to order the products. If the imagery is not suitable for the style or does not match the content, then the web application will not be as effective.

Even a choice of a colour on a button can sometimes double the impact on the users.

However, user interface design is difficult; it takes time, is prone to errors, and UI flaws are noticeable and humiliating. It takes time and effort to develop a good user interface because it must be built with all types of users in mind. While the programmer can understand the UI's workflow, this does not guarantee that all users will.

The UI for Click & Eat was updated a few times based on feedback from various categories of users, including those with technological experience, those who are not very good with technology, and those who work in the hospitality industry.

The ordering process was the most problematic aspect of the Click & Eat UI. Selecting an object and adding it to the order is easily done with a simple click on the item. As shown in the Figure below, the number of items added to the order is shown on the right side of the navigation bar.

Most reviewers saw this feature as not intuitive because they were unable to notice the order and the item count increasing.

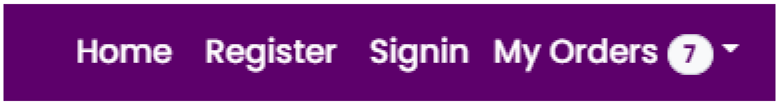
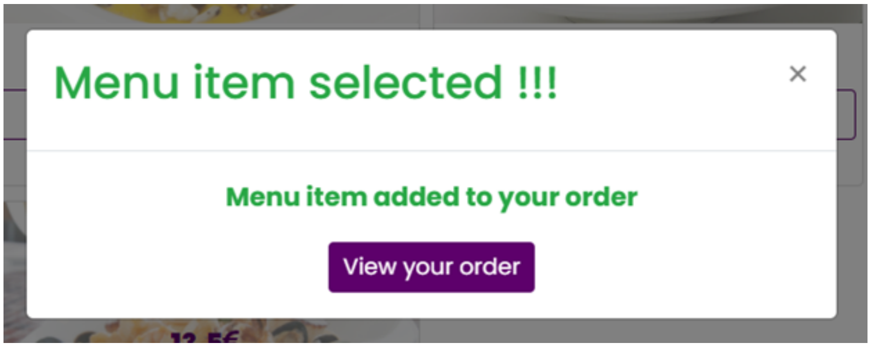


Figure 31 - Order feature
Source: Ana Griga, 2021

To fix the problem, a pop-up box was introduced to indicate that a menu item was added to the order, but this caused an even bigger problem because the customer had to close it after each item selected to the order, which was very inconvenient.


Menu item selected !!!

Menu item added to your order

Figure 32 - Order feature pop up
Source: Ana Griga, 2021

Finally, inspired by other similar applications, it was decided to add a banner to the screen that would update the order whenever a new product was added and, when clicked, would display the entire order, as seen in the Figure below.

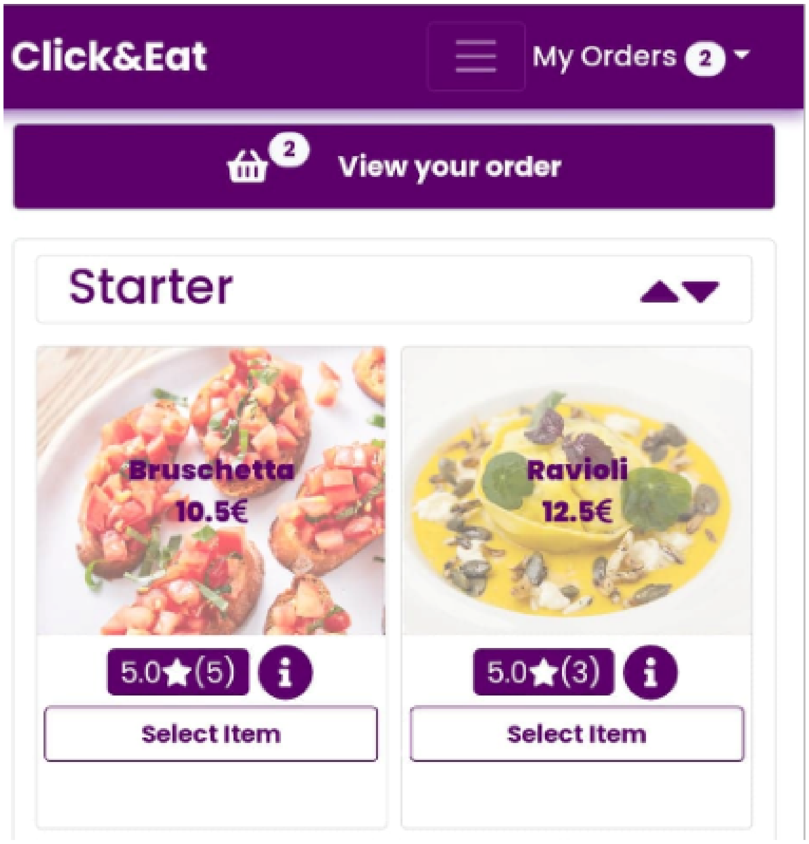


Figure 33 - Order feature banner
Source: Ana Griga, 2021

Understanding concepts such as reusable components, the component's lifecycle (mounting, updating, unmounting), the state of a component (which is an entity that contains specific information that can change during the component's lifetime), props (properties used to transfer data from one component to another) was difficult because ReactJs was a new technology never used before.

3.5.3 Payment integration

Working with APIs was a new experience during the development of the application, so incorporating the payment gateway was a challenge as well.

Initially, it was agreed to use the PayPal payment gateway, but the documentation was too complicated to comprehend, in contrast to the chosen payment gateway, Braintree, which had precise and to-the-point documentation.

Braintree offers great integration with the technologies used in the development of the Click & Eat application. It has excellent developer tools, including a sandbox environment with testing cards that is almost similar to the production environment.

Braintree's API credentials, including the API keys, were entered in the server-side code to connect the API calls to the Braintree gateway. More precisely, to integrate the Braintree gateway in the application, the Braintree.net SDK was installed on the server-side and the Braintree JavaScript V3 for the client-side. The client-side API offers a secure custom IFrame for collecting sensitive data from the payment fields.

Hosted Field solution is hosted by Braintree and gives the facility to customize the appearance of the form while ensuring that it is compliant with Payment Card Industry requirements.

The Braintree-hosted Hosted Field solution allows customisation of the form's presentation while also ensuring that it complies with Payment Card Industry standards.

4. Description of Learning

"The beautiful thing about learning is that no one can take it away from you."

- B.B. King

Fresh resolutions on milestones and progress by learning are made at a certain stage in life. It was decided four years ago to embark on a new journey on a different route. The intention was to go as far as possible, and the objectives were the steps that were taken to get there.

Developing the final year project was the most important and challenging step toward achieving the desired outcome. It was not easy, but the learning and knowledge gained fed the mind and made the effort worthwhile. From initial conception to completion, the project offered a great experience and both technological and personal learning results were achieved.

4.1 Technical

4.1.1 C# and Asp Net Core

C# and Asp Net Core were chosen for developing the server-side of the Click & Eat application because they are widely used in the industry according to the stack overflow 2020 developer survey, which means that they have been tested in production by multiple organizations, and they have large community support.

Even though C# has similarities to Java, which was learned in the first and second years, there were limitations since it was a new language and environment that was only tried for a few months during the internship last summer, so it was an opportunity to break through at this point of limitation.

It was discovered early on in the project that Asp Net Core includes a ReactJS Single Page Application template. When Click & Eat was developed as a project in Visual Studio using the SPA template, all of the packages needed for ReactJS were installed automatically using NPM, which was extremely convenient.

All the projects developed in different programming languages contained a folder for the static files such as images, CSS, Scripts files named according to the preferences. Asp Net Core offers a default folder called wwwroot for all the static files, which was a new thing to learn.

It was useful to learn about the action filters that are supported in Asp Net Core for authorization, error handling, and API routing when developing the API server.

Another pleasant surprise was the discovery of Swagger OpenAPI in the ASP.NET Framework. It was extremely helpful in automatically creating API documentation.

It was learned about the importance of asynchronous methods that do not block the UI for the next operation and that Asp Net Core allows for extensive use of the asynchronous pattern, which, according to research, makes the application faster and provides a better user experience.

A very important learning outcome about .Net Framework was the use of Entity Framework, which allows dealing with the data in the form of objects and properties without the worry of the database tables and columns. Essentially, the Entity framework offers a high level of abstraction, allowing developers to build and maintain data-oriented applications with less code than conventional applications.

4.1.2 ReactJS

ReactJS learning path was new, and many concepts were necessary to be understood, but the ReactJS community has lots of documentation and tutorial videos that were a great help. It was indeed an exhausting list to learn when working with this library.

The first was understanding that the App component is the main component in the application, which serves as a container for all the other components. All Click & Eat applications are made up of reusable pieces of code called components.

When creating the first components, it can be confusing to create HTML like code inside JavaScript, a term known as JSX, which is simply a syntax extension to JavaScript. JXS pretty much describes what the UI should look like.

Then the concept of props and state was discovered. Props stand for properties are arbitrary inputs for components that play a significant part in making the components reusable. Props are read-only, and the component can never modify its own props.

State allows ReactJS components to change their output over time which rerenders the UI. Once an understanding of state was gained, it was necessary to understand how to modify the state. It was also essential to understand what causes a component to rerender and how to hook into this using the setState() function.

Learning this library made it possible to understand how to create an application that is responsible on different devices by scaling smoothly on different sized screens such as laptops or mobile phones.

Click & Eat application contains around twenty components, each using all the concepts described above and the logic functionality correspondent, which significantly improved the experience using ReactJs for developing an application from start to finish.

4.1.3 Deployment

To make the application available to the user, it had to be deployed on a server that would run it continuously, which was a new process that necessitated research and learning on how to do it. The application was published using Microsoft Azure services, and luckily, the documentation was clear, detailed, and helpful, making the process simple.

4.1.4 UML

Modelling with UML was taught in college and developing this project made the importance of using it very understandable.

There are nine use cases presented in the specification document for each main feature and they are all described in detailed use case scenarios which present each step for performing the specific feature.

The system sequence diagrams, which display the events that the actors produce, the order of events, and the interactions with the system for the features described by each use case scenario, are included in the design document.

The set of diagrams used in the design of the Click & Eat application were developed to help for specifying, visualizing, constructing, and documenting the features of the system and improved the knowledge gained in the UML classes during college.

4.1.5 Agile Approach

Learning how important it is to work in an agile way was another important outcome in developing the final year project. From the start, the work was done in an agile like style incrementally through an iterative process, and the feedback received after each iteration improved the quality of the features considerably.

All work was divided into small tasks, and time was allocated for developing and documenting all the progress. This approach was a great help in successfully developing the application, and it made understanding how helpful agile is in organizing the work and refining each feature at its best with the great help of each feedback.

4.2 Personal

4.2.1 Time management

An increased amount of work with a large number of responsibilities can be overwhelming, so the lesson learned after working on the final year project was to prioritize the tasks and use the time wisely to get more accomplished each day. This resulted in a reduction in tension and an increase in productivity.

Creating a plan at the start of the day with a list of the important tasks needed to be completed was critical in the productivity of the work done and learning how to prioritize important tasks kept the focus in accomplishing it.

Making a schedule at the start of the day with a list of essential tasks to do was crucial to the success of the work accomplished, and understanding how to prioritize important tasks emphasised on completing them.

It was also understood that starting early gives a clearer and more focused energy and creativity because as the day passes the energy levels start going down affecting productivity and motivation. Also taking regular breaks helped in relaxing and getting back to work with regained energy and more willingness to complete the desired tasks.

It was also recognized that getting up early provides clearer, more concentrated energy and inspiration, because as the day progressed, energy levels began to decline, impacting efficiency and motivation. Taking daily breaks often assisted in relaxing and returning to work with renewed enthusiasm and a greater ability to complete the tasks.

Good time management was taught, and using the above techniques aided in the completion of the project, despite the fact that there were occasions when it seemed that there was insufficient time due to other school exam preparations and other tasks to complete.

4.2.2 Project management

Managing the project was another learning outcome gained after developing the Click & Eat application. This was an ability that was learned without realizing it from the beginning of the project with the planning and scheduling to implement the work that needed to be done, delivering new features as they were completed, checking the features, and remodelling them after each feedback received.

4.2.3 Perseverance and motivation

The secret to success is perseverance and motivation. It is important to learn how to deal with and overcome challenges, and this was taught in the process of developing this project. Perseverance developed after a series of failures and each time returning to try again, as exercising and practising increased overall resilience.

Every choice has a price, the choice of following this course was an ambition of doing something good for the future and motivation helped in getting so far even when the road was sometimes bumpy. The project helped in understanding that motivation is essential in changing the dreams into reality.

Every decision comes with a cost; in this case, the cost was the desire to do something positive for the future, and motivation and discipline helped in getting this far despite the bumps in the road. The project provided insight into the importance of motivation in turning dreams into reality.

4.2.4 Academic writing

All other projects developed during the course of college did not necessitate writing a lot of documentation. Writing academic papers in a professional manner was another important skill gained from the final year project.

Writing in a structured manner without using the first person was new and challenging at first, and referencing all of the work in the proper format was another ability learned from all of the documents produced. Academic writing made use of logic as well, and it was important to present the arguments in a logical order in order to draw conclusions.

Another skill gained was reading the papers aloud, which guaranteed their accuracy. When hearing it, it was clear when the sentences were too long, the structure was too complicated, or the punctuation was too sparse.

4.2.5 Research

The capacity to perform research on the different topics was another skill gained as research was essential in finding information and creating an outline for the project documents.

The research was a big task, and sometimes it became intimidating and overwhelming because the resources needed to be recognized as a quality resource judging by the web site or the book and the author's reliability.

Writing the project's documents with a lot of research was a wonderful opportunity to learn about various subjects and gain experience on them, and the writing process helped in recalling what was learned and understanding it on a deeper level.

Knowing how to research is essential in writing good scholarly documents, and it is a valuable skill that will serve well throughout the career and the project helped in learning it in depth.

4.2.6 Presentation

Because of the anxiety and nervousness present, giving a successful and efficient presentation is difficult, but a lot of practice improves the ability to control the tension and be more enthusiastic and comfortable. After a few presentations this year, it was discovered that engaging the audience, showing passion for what is presented, adding some humour and a smile and practising before are the key features in delivering a successful presentation.

5. Review of the Project

Project analysis activities were used to find flaws in code or document deliverables at various points in the development process. As a result, it can be said that periodically checking the source code and project documents helped in purifying the project.

This is the final assessment, conducted after the project's completion, to decide if the project's objectives were met.

5.1 Project’s Success

In its current state, the application accomplishes the original goals detailed as per features implemented for the three types of users and other aspects considered as a success:

5.1.1 The customer

- Can scan a QR code provided by the restaurant or optionally can create an account and login
- The QR code or the login will give the access to the menu
- Can create an order by selecting menu items
- Can add instructions per item and special instructions per order
- Can pay for the order
- Can leave feedback for any menu item after the order was filled
- Can log out if they were initially logged in
- Can change the password if they have an account

5.1.2 The waiter

- Can see the daily, weekly, monthly and yearly sales
- Can see the orders
- Can see the instructions for each menu item and the special instructions of the order
- Can see the payment status
- Can see all the feedback for any menu items
- Can change their account’s password

5.1.3 The administrator

- Can CRUD menu categories
- Can CRUD menu items

- Can CRUD waiter accounts
- Can see customer accounts
- Can see the daily, weekly, monthly and yearly sales
- Can see the orders
- Can see the instructions for each menu item and the special instructions of the order
- Can see the payment status
- Can see all the feedback for any menu items
- Can change their account's password

5.1.4 Application FURPS success

- The application has valid redirects of the pages and all pages are loading correctly
- The workflow of the application corresponds to the specification requirements
- The intuitiveness and ease of the application was verified with the help of feedback and manual testing
- The browser compatibility was checked and the application is running on any browser
- The application is compatible on laptops or desktops and on mobile devices
- The response times for different operations corresponds with the specified goals in the initial phase of the project
- The payments is secured by Braintree gateway and the Hosted Fields form

5.2 Project's Failures

The ordering page of the application may be a failure after reviewing all of the features and feedback obtained. Owing to a lack of time and difficulty, the results were not as expected after attempting to execute it as best as possible. It was planned to introduce an animation that would show adding each menu item to the order, but due to time constraints and the difficulty of the process, it was abandoned.

Another issue that was considered a failure was not testing the software. It was decided at the start of the project that an automated testing using Selenium will be performed and even a unit testing but due to the lack of time this task was not achieved.

It was determined at the outset of the project that automated testing and even unit testing would be conducted, but due to a lack of time, this task was not completed.

6. Potential places of expansion

Several expansion ideas arose during the project's development, and while they would have been nice to be incorporated, they are planned to be added in the future.

First and foremost, another evaluation and improvement of the ordering feature will be completed as soon as possible, as this function was not a real success.

Making the application capable of serving several restaurants at once is a function that is considered a good addition to the application and was addressed with the tutor. The application currently only has one database, and refactoring all the code to support multiple databases for each restaurant was too challenging and the time was not enough.

Another feature that is being examined for potential implementation is the addition of a booking system to the app, which will enable customers to reserve a table in advance and help the restaurant to have a clear picture of table assignments for each day.

It was considered to incorporate a service that would allow the application to support multiple languages, which would be useful for tourists who do not speak English, but it was never researched; however, it is something that would be investigated in the future.

Another useful function that will be introduced in the future is the ability for customers and restaurant workers to monitor orders by time, which will tell them how much longer they have to wait for their order to be served.

7. Acknowledgements

Throughout the final year project called Click & Eat and all my four years in college, I have received valuable support.

First, I want to express my deep and sincere gratitude to my tutor, Dr Chris Meudec, for providing invaluable guidance throughout this project. Thank you for teaching me the methodology to carry out better the research and the development of the project and to present the work as clearly as possible.

I must thank, with all my heart, all of the Institute of Technology Carlow lecturers, whom I have worked with over the last four years, for teaching me what it means to be dedicated, for giving me their time, energy and expertise and for helping me make my dream come true.

Without the help of my family, lecturers and friends, this path would not have been possible. Thank you for supporting me in all of my endeavours and helping me to pursue my goals.

Thank you to all my friends and colleagues for encouraging me in all my pursuits and inspiring me to follow my dreams. Thank you all for listening, offering me advice, support and help through this entire journey.

8. Bibliography

Chavan, Y., 2021. *JSX in React – Explained with Examples.* [Online]
Available at: https://www.freecodecamp.org/news/jsx-in-react-introduction/
[Accessed 10 April 2021].

Gore, A., 2020. *How JSON Web Token (JWT) Secures Your API.* [Online]
Available at: https://dzone.com/articles/how-json-web-token-jwt-secures-your-api
[Accessed 10 April 2021].

Johnson, M., 2020. *Braintree Payments Review.* [Online]
Available at:
https://www.merchantmaverick.com/reviews/braintree-payment-solutions-review/
[Accessed 10 April 2021].

Minnick, C., 2016. *The Real Benefits of the Virtual DOM in React.js.* [Online]
Available at:
https://www.accelebrate.com/blog/the-real-benefits-of-the-virtual-dom-in-react-js
[Accessed 10 April 2021].

Packt, 2013. *Working with ASP.NET Web API.* [Online]
Available at: https://hub.packtpub.com/working-aspnet-web-api/
[Accessed 10 April 2021].

Smith, T., 2020. What Are 8 Milestones Every Software Project Should Have?. [Online]
Available at:
https://soltech.net/what-are-8-milestones-every-software-project-should-have/
[Accessed 15 April 2021].