OMNIA

OUR TEAM:

Ismaeel Rahaman
u20427906
u20427906@tuks.co.za
065 814 5362

Nawailah Tarmohamed
u21437972
u21437972@tuks.co.za
067 319 1637

Deshlan Pillay
u21481084
u21481084@tuks.co.za
067 127 6684

Sameer Ghela
u21445142
u21445142@tuks.co.za
066 239 5477

Sashin Gounden
u20487062
u20487062@tuks.co.za
072 520 1968

From left to right: Sameer Ghela, Ismaeel Rahaman (Group leader), Nawailah Tarmohamed, Sashin Gounden, and Deshlan Pillay.

Team 7: Omnia	Project: Water Resort administration system	System Name: Hydrotech

Iteration 7 – Technical Specifications

This document will define the technical specifications that our system will use, and it includes all external and internal technologies.

Table of Contents

Table of Figures

1. Document Introduction

Technical specifications of a system are a detailed description of the hardware, software, and other technical components that make up the system. These specifications provide a comprehensive overview of the system's capabilities, performance, and requirements.

2. Technical Specifications

2.1. SQL

2.1.1. What is SQL?

SQL stands for Structured Query Language, and it is a programming language used to manage and manipulate relational databases. SQL allows users to create, modify, and query databases to retrieve, insert, update, and delete data.

SQL is a standard language used by many relational database management systems, such as MySQL, Oracle, and Microsoft SQL Server. It is also commonly used in web development, data analysis, and other areas where large amounts of data need to be stored and managed.

Some of the basic commands in SQL include SELECT, INSERT, UPDATE, and DELETE, which allow users to retrieve data from tables, add new data to tables, modify existing data, and remove data from tables, respectively. SQL also includes commands for creating tables, defining relationships between tables, and setting up constraints and indexes to ensure data integrity.

2.1.2. Setting up the SQL database

Based on our data definition language (DDL), we will use the create query to make a new database and new tables (entities), the drop query to delete tables inside the database and in exceedingly rare cases, the alter query to edit tables should the situation arise.

2.1.3. Database integrity

Referential integrity is a concept in SQL that ensures the consistency and accuracy of data between tables. It ensures that the data in one table is linked to the data in another table in a consistent and accurate manner. To implement referential integrity in SQL, we will use foreign keys and by ensuring these don't allow for null entries. Entity integrity is another important concept in SQL that ensures that each row in a table is unique and identifiable. It requires that each row has a primary key that uniquely identifies that row. To implement entity integrity in SQL, we will use primary keys and to uniquely identify each instance in a table.

2.1.4. Connecting SQL server to .Net Core

To connect to a SQL Server database from a .NET Core application, you can use the Microsoft.Data.SqlClient package, which is a lightweight and cross-platform SQL Server provider for .NET Core.

1. Install the Microsoft.Data.SqlClient package using the NuGet Package Manager or run the following command in the Package Manager Console:

```
mathematica                                                    Copy code

Install-Package Microsoft.Data.SqlClient
```

Figure 1:Install Microsoft Data SQLClient package

2. Install the Microsoft.Data.SqlClient package using the NuGet Package Manager or run the following command in the Package Manager Console:

```arduino
using Microsoft.Data.SqlClient;
```

Figure 2:Run command in PM console

3. Install the Microsoft.Data.SqlClient package using the NuGet Package Manager or run the following command in the Package Manager Console:

```java
string connectionString = "Server=myServerAddress;Database=myDatabase;User I
SqlConnection connection = new SqlConnection(connectionString);
```

Figure 3:Connection string

4. Replace myServerAddress, myDatabase, myUsername, and myPassword with your own values.

5. Open the connection:

```mathematica
connection.Open();
```

Figure 4:Use to open connection

2.1.5. Using SQL in .NET Core:

As mentioned, the queries, SELECT, UPDATE and DELETE will allow us to physically work with the data in our database.

In order to use these queries in .NET Core, we would have to create an ‘SQL Command’ object and then execute it:

1. SELECT

```csharp
string sql = "SELECT * FROM myTable";
SqlCommand command = new SqlCommand(sql, connection);

using SqlDataReader reader = command.ExecuteReader();
while (reader.Read())
{
    Console.WriteLine($"{reader["id"]}, {reader["name"]}, {reader["age"]}");
}
```

Figure 5:Select query

2. UPDATE

```csharp
using Microsoft.Data.SqlClient;

string connectionString = "Server=myServerAddress;Database=myDatabase;User I
string sql = "UPDATE myTable SET name = @name, age = @age WHERE id = @id";

using SqlConnection connection = new SqlConnection(connectionString);
using SqlCommand command = new SqlCommand(sql, connection);

command.Parameters.AddWithValue("@id", 1);
command.Parameters.AddWithValue("@name", "John");
command.Parameters.AddWithValue("@age", 30);

connection.Open();
int rowsAffected = command.ExecuteNonQuery();
connection.Close();
```

Figure 6:Update query

3. DELETE

```csharp
using Microsoft.Data.SqlClient;

string connectionString = "Server=myServerAddress;Database=myDatabase;User I
string sql = "DELETE FROM myTable WHERE id = @id";

using SqlConnection connection = new SqlConnection(connectionString);
using SqlCommand command = new SqlCommand(sql, connection);

command.Parameters.AddWithValue("@id", 1);

connection.Open();
int rowsAffected = command.ExecuteNonQuery();
connection.Close();
```

Figure 7:Delete query

These will allow us to enter new, update existing and to delete data in our database.

2.1.6. Stored Procedures, Jobs, and Triggers in SQL

1. **Stored Procedures**

A stored procedure in SQL Server is a precompiled collection of SQL statements and procedural logic that is stored as a named object in the database. It can be called from a .NET Core application or any other client application to execute a specific task or set of tasks.

2. **Jobs**

In SQL Server, a job is a specified set of tasks or procedures that are scheduled to run automatically at a specific time or in response to a specific event. Jobs can be created and managed using the SQL Server Agent service, which is responsible for scheduling and running jobs on an SQL Server instance.

3. **Triggers**

A trigger in SQL Server is a special type of stored procedure that is automatically executed in response to certain database events, such as an insert, update, or delete operation on a table. Triggers can be used to enforce business rules, audit changes to data, or perform complex calculations or transformations on data.

2.2. Angular

2.2.1. What is Angular?

Angular is a popular open-source web application framework that allows developers to build dynamic, responsive, and scalable web applications. It is developed and maintained by Google and is based on the TypeScript language, which is a superset of JavaScript.

Angular provides a rich set of features for building modern web applications, including powerful data binding, dependency injection, component-based architecture, and a wide range of tools for testing, debugging, and optimizing applications. It also includes a robust set of built-in directives and services for interacting with the DOM, handling user events, and making HTTP requests.

With Angular, developers can build complex single-page applications (SPAs) that offer a seamless user experience, with fast and responsive interfaces, and a clean and maintainable codebase. Angular also provides a rich ecosystem of tools and libraries, including the Angular CLI, which simplifies the process of creating, testing, and deploying Angular applications.

2.2.2. Angular in our system

To begin, we will create an Angular application that will serve as the framework for designing and programming our system. We chose Angular due to its Object-Oriented programming approach, better structuring, and excellent tool support, which will streamline development time. Within the Angular workspace, we will use modules, routing, components, services, and pipelines to facilitate communication throughout the application. Since Angular uses HTML and TypeScript, it will be easier for us to implement, considering we are already using TypeScript. In addition, we will integrate the client's webpage into our application using HTML hyperlinks or Angular Routing, making it efficient and user-friendly. One of our system's core functions is accommodation, and we must ensure it is available to users who do not book an event. Similarly, clients that wish to book a day visit without these 2 functions should be routed to the appropriate page. Simple routing will aid in addressing this issue.

2.2.3. Angular CLI

Angular CLI (Command Line Interface) is a powerful tool that simplifies the process of creating, testing, and deploying Angular applications. It is a command-line interface that can be used to generate code, add components, services, pipes, and directives, run unit and end-to-end tests, and build and deploy the application to various platforms.

Angular CLI is built on top of Node.js and provides a set of tools and utilities that streamline the development process. It includes a built-in development server that automatically reloads the browser when changes are made to the code, as well as a range of code generators that help to speed up development by automatically generating boilerplate code for components, services, and other application features.

One of the key benefits of Angular CLI is that it provides a standardized and consistent development environment across teams, ensuring that all developers are using the same toolset and best practices. It also includes a wide range of plugins and extensions that

allow developers to easily integrate with other tools and platforms, such as testing frameworks, IDEs, and cloud-based services.

Overall, Angular CLI is a powerful and flexible tool that helps to simplify the development process for Angular applications, enabling developers to focus on building high-quality, scalable, and maintainable applications without getting bogged down in the details of the development process.

2.3. .NET Core Web API

2.3.1. What is .NET Core Web API

.NET Core Web API is a framework for building web APIs using the .NET Core platform. It allows developers to create web services that can be consumed by different types of clients, such as web applications, mobile apps, or other backend services.

.NET Core is a cross-platform, open-source framework that supports building applications for Windows, macOS, and Linux operating systems. It includes a runtime, libraries, and tools to build different types of applications.

Web APIs built using .NET Core Web API can communicate with clients using various protocols such as HTTP, HTTPS, and WebSocket. It provides features like routing, model binding, middleware, and authentication to help developers build robust and secure APIs.

Developers can use C# or other .NET languages to write code for their web API, and they can deploy it to various platforms such as Docker containers, Azure cloud, or on-premises servers.

Overall, .NET Core Web API is a powerful framework for building modern web services that are cross-platform, fast, and scalable.

2.3.2. .NET Core Web API in our system

Angular will serve as the front end of our system. The API will function as the back end, providing the vital communication functions between our database and angular application. We will use the IDE Visual Studios to use this framework.

2.3.3. Connecting The API to our Angular application

Connecting an Angular application to a .NET Core Web API involves several steps. Here is an overview of the process we will use:

1. Create an Angular application: The first step is to create an Angular application using the Angular CLI. This can be done by running the command ng new <app-name> in the command prompt or terminal.
2. Install the HttpClient module: Angular uses the HttpClient module to make HTTP requests to a server. To install it, run the command npm install @angular/common @angular/compiler @angular/core @angular/forms @angular/platform-browser @angular/platform-browser-dynamic @angular/router rxjs zone.js --save in the command prompt or terminal.
3. Create a service to consume the Web API: In the Angular application, create a service that will consume the .NET Core Web API. This can be done by using the HttpClient module to make HTTP requests to the API endpoints.
4. Configure the CORS policy: By default, the .NET Core Web API does not allow cross-origin requests. To allow the Angular application to make requests to the API, you need to configure the CORS policy in the API. This can be done by adding the Microsoft.AspNetCore.Cors package to the API project and configuring the Cors middleware in the Startup.cs file.

5. Test the connection: Once the service is created and the CORS policy is configured, you can test the connection between the Angular application and the .NET Core Web API by making a request to one of the API endpoints from the Angular application.

Overall, connecting an Angular application to a .NET Core Web API requires configuring the CORS policy and creating a service that can consume the API endpoints. With these steps, you can create a robust and scalable web application that can interact with a backend API.

2.4. QR Code

2.4.1. What are QR codes?

A QR code (short for Quick Response code) is a two-dimensional barcode that contains information in the form of black and white squares or modules arranged in a square grid. It was invented in 1994 by Denso Wave, a subsidiary of Toyota, to track automotive parts during the manufacturing process.

QR codes can be scanned using a smartphone camera or a QR code reader application. The information contained in a QR code can be text, a URL, a contact information, or any other type of data. When a user scans a QR code, the data is extracted from the code and displayed on the user's device.

QR codes are used in a variety of applications such as marketing, inventory management, ticketing, and payment systems. For example, a company may use a QR code on a product label to provide customers with more information about the product, or a museum may use a QR code on an exhibit to provide visitors with audio or video content.

QR codes can store up to several hundred characters of information, and they have error correction capabilities that allow them to be read even if some of the modules are damaged or obscured. QR codes have become increasingly popular in recent years due to their versatility and ease of use.

2.4.2. QR Codes in our system

QR codes will be used in the booking part of our system, namely the event booking and day visit bookings. Our system must generate QR codes when the client makes the booking online and sends them to the client. When they arrive at the resort, they provide this code for scanning and the system must check that this matches the code generated for this booking.

2.4.3. QR codes using .NET Core (Technical)

1. We firstly need to add the QR Coder NuGet Package

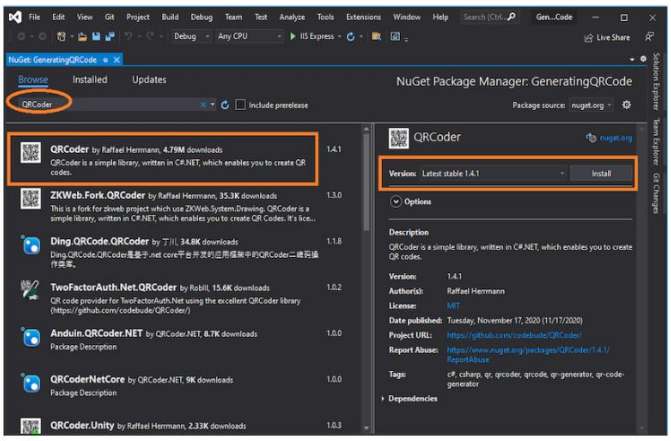


Figure 8:QR code NuGet package

2. After which we can define our QRCode model class and controller
3. The following method is the code to generate the QR Code which will be stored in the database.

.

```
using GeneratingQRCode.Models;
using Microsoft.AspNetCore.Mvc;
using QRCoder;
using System;
using System.Drawing;
using System.Drawing.Imaging;
using System.IO;

namespace GeneratingQRCode.Controllers
{
    public class HomeController : Controller
    {

        [HttpGet]
        public IActionResult CreateQRCode()
        {
            return View();
        }

        [HttpPost]
        public IActionResult CreateQRCode(QRCodeModel qRCode)
        {
            QRCodeGenerator QrGenerator = new QRCodeGenerator();
            QRCodeData QrCodeInfo = QrGenerator.CreateQrCode(qRCode.QRCodeText, QRCodeG
            QRCode QrCode = new QRCode(QrCodeInfo);
            Bitmap QrBitmap = QrCode.GetGraphic(60);
            byte[] BitmapArray = QrBitmap.BitmapToByteArray();
            string QrUri = string.Format("{0}", Convert.ToBase64S
            ViewBag.QrCodeUri = QrUri;
            return View();
        }
    }

```

Figure 9:Code to generate the QR Code

```
22          {
23              QRCodeGenerator QrGenerator = new QRCodeGenerator();
24              QRCodeData QrCodeInfo = QrGenerator.CreateQrCode(qRCode.QRCodeText, QRCodeG
25              QRCode QrCode = new QRCode(QrCodeInfo);
26              Bitmap QrBitmap = QrCode.GetGraphic(60);
27              byte[] BitmapArray = QrBitmap.BitmapToByteArray();
28              string QrUri = string.Format("{0}", Convert.ToBase64S
29              ViewBag.QrCodeUri = QrUri;
30              return View();
31          }
32      }
33
34      //Extension method to convert Bitmap to Byte Array
35      public static class BitmapExtension
36      {
37          public static byte[] BitmapToByteArray(this Bitmap bitmap)
38          {
39              using (MemoryStream ms = new MemoryStream())
40              {
41                  bitmap.Save(ms, ImageFormat.Png);
42                  return ms.ToArray();
43              }
44          }
45      }
46  }
```

Figure 10:Code to generate the QR Code

4. The next part is to configure the routing for the function.

```
app.UseEndpoints(endpoints =>
{
    endpoints.MapControllerRoute(
        name: "default",
        pattern: "{controller=Home}/{action=CreateQRCode}/{id?}");
});
    }
}
}
```

Figure 11:Configure routing

When the QR code is stored in the database, it will be stored as a URL. When the customer provides the QR code, the system will scan the code and verify that the scanned code matches the URL in the database.

2.5. MailKit- Sending Emails

2.5.1. What is MailKit?

MailKit is an open-source .NET email client library that provides a cross-platform API for sending, receiving, and managing email messages. It is a robust and feature-rich library that supports various email protocols such as SMTP, POP3, IMAP, and Exchange Web Services (EWS). MailKit also provides support for S/MIME and PGP email encryption and signing, as well as advanced features such as message threading, message signing and encryption, mailbox search, and more.

2.5.2. MailKit in our system

When the client makes a booking, we would like to send an email with details of the booking as confirmation. Information to be included in the email:

- Client Name
- Booking Date
- Booking reference number
- Amount

We chose email for this as we feel this proves most effective when it comes to providing a user-friendly interface. Another reason is that emails are a universally accepted way of communicating hence it would provide an easy way for clients to receive their booking details.

2.5.3. The technical connection

To set up MailKit in your .NET application we will follow these steps:

1. Install the MailKit package: You can install the MailKit package using the NuGet Package Manager in Visual Studio or by running the following command in the Package Manager Console:

```mathematica
Install-Package MailKit
```

Figure 12:Install MailKit package

2. Import the MailKit namespace: Once you have installed the MailKit package, you need to import the MailKit namespace in your code:

```csharp
using MailKit.Net.Smtp;
using MailKit.Security;
using MimeKit;
```

Figure 13:Import MailKit package

2.6. SMS's- Twilio

2.6.1. What is Twilio?

Twilio SMS is a feature provided by the Twilio cloud communications platform that enables developers to programmatically send and receive SMS messages in their applications. With Twilio SMS, developers can add SMS functionality to their web or mobile applications, such as sending appointment reminders, two-factor authentication codes, or marketing messages.

Twilio SMS is highly customizable and provides a range of options for sending and receiving messages. For example, developers can use short codes or long codes for sending messages, which can be branded with their business name or number. They can also use the REST API or Twilio Helper Libraries to send SMS messages from popular programming languages such as Java, Python, and C#. Additionally, Twilio SMS supports Unicode characters, which allows developers to send messages in multiple languages and include emojis.

To use Twilio SMS, developers need to create a Twilio account and obtain an API key and secret. They can then use the Twilio API to programmatically send and receive SMS messages. The Twilio platform also provides extensive documentation and support to help developers get started quickly and troubleshoot any issues that arise.

2.6.2. Twilio in our system

When the time arrives closer to a clients' booking, we would like to send them a gentle reminder of said booking via SMS. This would involve the use of a timer as well.

We have opted for an SMS for this as opposed to emails due to the fact that it gets the recipients attention easier and due to the fact that users don't check emails as often as they view their messages. Since it serves a reminder for a booking, we would like it to grab the user's attention.

2.6.3. The Technical Connection

To install the Twilio .NET Core SDK for sending SMS messages, we will follow these steps:

1. In the NuGet Package Manager, search for "Twilio" and select the "Twilio" package from the search results.
2. Click the "Install" button to install the package and its dependencies.
3. Once the installation is complete, we can start using the Twilio SDK in your project.

```csharp
using Twilio;
using Twilio.Rest.Api.V2010.Account;

// set up the Twilio client with your account SID and auth token
TwilioClient.Init("your-account-sid", "your-auth-token");

// create and send an SMS message using the Twilio API
var message = MessageResource.Create(
    body: "Hello from Twilio!",
    from: new Twilio.Types.PhoneNumber("your-twilio-phone-number"),
    to: new Twilio.Types.PhoneNumber("recipient-phone-number")
);

// print the message SID to the console
Console.WriteLine(message.Sid);
```

Figure 14: Twilio code to send SMS

Once you have the Twilio SDK installed in your .NET Core Web API project, we can use it to send SMS messages and take advantage of other Twilio features such as receiving messages and tracking message status.

2.7. Payments- PayPal

2.7.1. What is PayPal?

PayPal boasts a robust set of features and functionalities that allow merchants to effortlessly process online transactions. With its foundation dating back to the late 1990s, PayPal has become a global leader in digital payments. It offers a secure and user-friendly platform for businesses to receive payments through various means, including credit and debit cards, bank transfers, and PayPal account balances.

By integrating PayPal into my online business, I can ensure a streamlined payment experience for customers. PayPal provides extensive fraud protection measures, simplifies payment reconciliation, and offers responsive customer support. Integration options are versatile and cater to different needs, such as hosted payment pages, API-based custom integrations, and compatibility with major e-commerce platforms like WooCommerce, Shopify, and Magento.

Beyond standard payment processing, PayPal presents additional services like recurring payments, subscription billing, and automated invoicing. Each transaction made through PayPal incurs a transaction fee, which varies depending on the specific payment method and the volume of transactions processed monthly.

Overall, PayPal serves as a reliable and efficient payment gateway solution that aligns with my business needs. Its diverse payment options, security measures, and supplementary services make it an excellent choice for businesses seeking to facilitate online payments.

2.7.2. PayPal in our system

In order to make payments towards bookings, we have decided to make clients use the PayPal payment gateway for the following reasons:

1. Security: PayPal is committed to providing a secure payment environment for its merchants and customers.
2. Convenience: PayPal offers a variety of payment methods, including credit and debit cards, and Instant EFT, making it easy for customers to pay for goods and services online.
3. Integration: PayPal provides a variety of integration options, including a hosted payment page, custom integration with an API.
4. Cost-effective: It's free to open a PayPal account and buy something using PayPal unless it involves a currency conversion. There are no PayPal fees if you use your PayPal balance to send money in your own currency to friends and family. There may be fees for other transactions.

2.7.3. The Technical Connection

To use PayPal in an Angular application, we will follow these general steps:

1. Set up a PayPal account. This will involve providing your personal and business details and linking your bank account to your PayPal account.
2. Create a PayPal merchant ID and key in your PayPal account settings. This will allow you to authenticate your requests to PayPal.

3. Install the credit card package in your Angular application. You can use npm to install the package as follows:

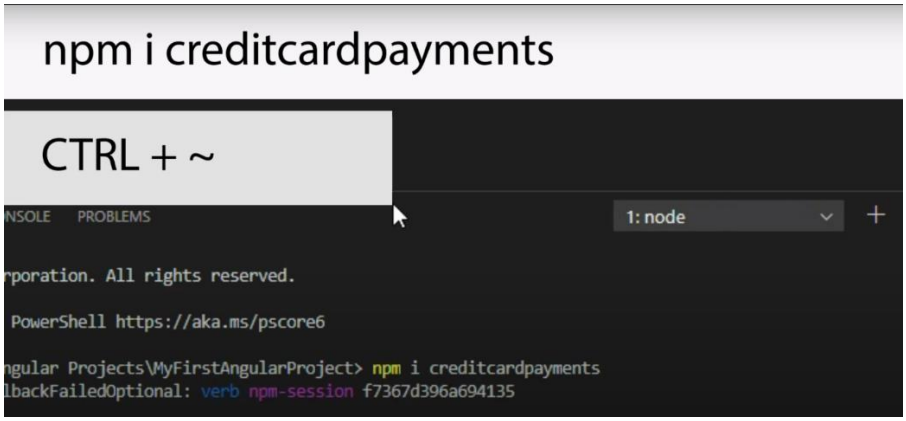


Figure 15:Install credit card package in angular

4. Import the PayPal API in your Angular app:

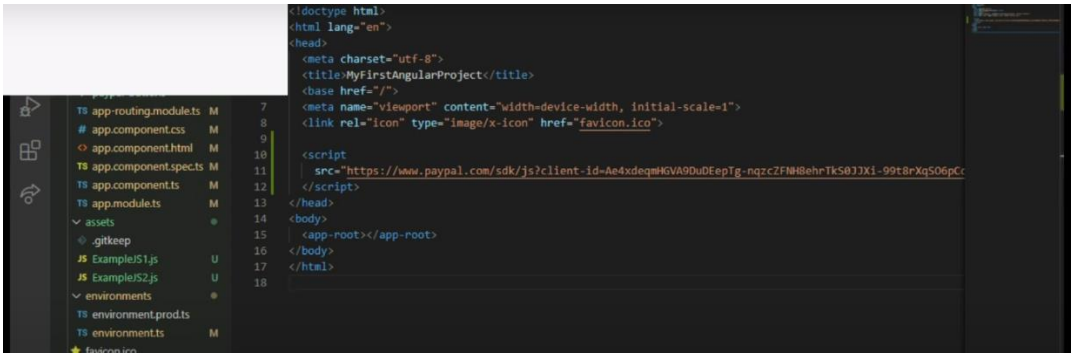


Figure 16:Import PayPal API in angular app

5. Replace "your-merchant-id" and "your-merchant-key" with our credentials.

6. Use the PayPal Service in your Angular component to generate a payment request and redirect the user to the PayPal payment page:

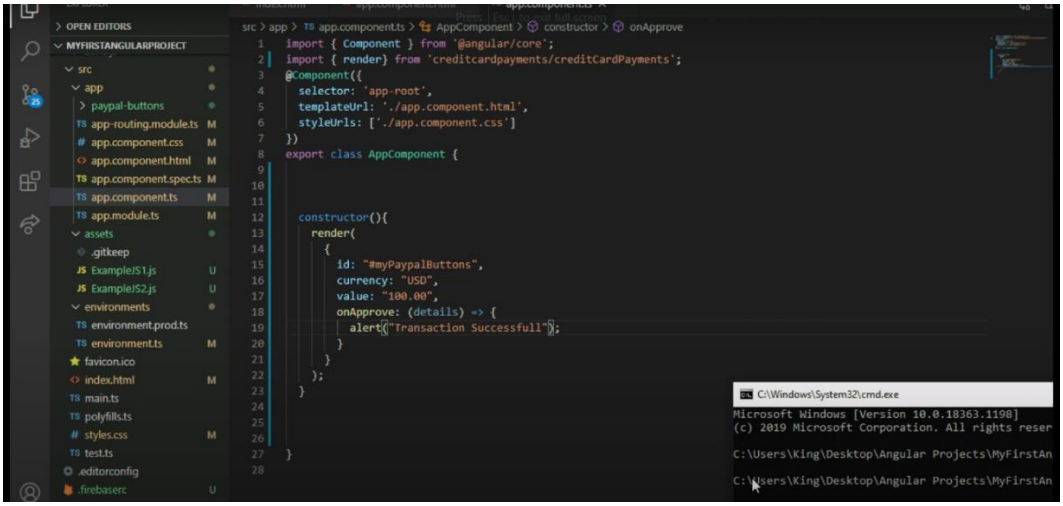


Figure 17:Code to generate a payment request and redirect the user to the PayPal payment page

3. Hardware And Software Specification

3.1. Hardware Specifications

At present, Platinum Island resort operates without much technological support, relying solely on a manual paper-based system to carry out its business operations. The resort does not utilize any hardware for its operations, with the exception of the owner's desktop computer and the specialised printer that generates the wristbands for day-visits.

For the company to utilize the system to its full potential and allow for a seamless integration into the businesses current operations, the resort would need access to additional smart devices as well as an active internet connection. To successfully run the operations of the business, with the integration of the HydroTech system, the business would be recommended to acquire the following hardware:

1. Firstly, a **Desktop Computer** would serve as the interface between the customer and the system. Through the computer, the customer would be able to access aspects of the system and manage their business operations. The following hardware specifications would be recommended:
    - Intel Core i3 generation CPU processor.
    - AMD Ryzen 3 3300x CPU processor
    - 4GB – 8GB of RAM.
    - HDD with at least 16GB storage space.
    - Any dedicated GPU with WDDM 1.0 driver (Windows Display Driver Model).
    - Built in NIC (Network Interface Card)
    - Runs Windows or Linux OS.
2. Similarly, a **Personal Laptop** would function in a similar capacity with identical hardware specification. However, the laptop would allow employees of the organisation to work remotely.
3. **Mobile Smartphone** are a portable and convenient component to the system. Platinum Island currently manually sells tickets at the front desk for day visits. We plan to implement online bookings for day visits with bar codes which allow for a more efficient approach as the guard will simply scan the bar codes on the tickets using our Ionic app on a mobile smartphone.The following hardware specification would be recommended:
    - 2GB – 4GB of RAM.
    - Capable of running IOS or Android,
    - Any device capable of network connectivity and access to an app store.
4. Lastly, Platinum Island Resort does not own any servers to store their data, as all of this occurs on an excel spreadsheet and would prefer to keep their data stored digitally rather on a physical server, thus they will not need any physical servers. Furthermore, the resort has internet accessibility readily available to customers and employees.

3.2. Software Specifications

Operating System- Windows 11

We have chosen this version of windows as the operating system to run our application as it is the most recent and stable version of windows. This allows access to any updates that can fix bugs that may arise. Windows 11 is also the current operating system that our client uses hence the client will be accustomed to the interface. For our mobile applications, it will be made available on both Android and Apple devices. The required software versions are as follows:

Android: Android 12 OS or later
Apple: IOS 15 or later

DBMS- SQL Server 2019

Our client currently does not have a database of any sort implemented. Taking into consideration the resources available, we have chosen to use SQL server 2019 to implement our database. SQL Server 2019 Standard edition delivers basic data management and business intelligence database for departments and small organizations to run their applications and supports common development tools for on-premises and cloud enabling effective database management with minimal IT resources. It is efficient and easy to maintain.

Development Environment

Web API hosted on Azure- We will be making use of .Net Core Web API and Entity Framework 6.
Backend- Angular v15 and Typescript v5.0.4
Frontend- Ionic v7.0.0 and Ionic Capacitor 4.7.0

4. Document Conclusion

The technical specifications will provide the basis for our system's development, outlining the technologies to be utilized from the database to the APIs and front-end development. This may also include the selection and installation of plugins and packages to enhance the system's functionality.

6. Team sign-off

I, Ismaeel Rahaman, confirm that I have read the requirement changes of the Platinum Island Resort and agree to all of the information contained in this document. I confirm that the information stated above is factual and that I am aware of any and all changes made to this document between drafts. Furthermore, I confirm that I have contributed to this document equally as much as my teammates.

……..

Ismaeel Rahaman

I, Nawailah Tarmohamed, confirm that I have read the requirement changes of the Platinum Island Resort and agree to all of the information contained in this document. I confirm that the information stated above is factual and that I am aware of any and all changes made to this document between drafts. Furthermore, I confirm that I have contributed to this document equally as much as my teammates.

……..

Nawailah Tarmohamed

I, Deshlan Pillay, confirm that I have read the requirement changes of the Platinum Island Resort and agree to all of the information contained in this document. I confirm that the information stated above is factual and that I am aware of any and all changes made to this document between drafts. Furthermore, I confirm that I have contributed to this document equally as much as my teammates.

……..

Deshlan Pillay

I, Sameer Ghela, confirm that I have read the requirement changes of the Platinum Island Resort and agree to all of the information contained in this document. I confirm that the information stated above is factual and that I am aware of any and all changes made to this document between drafts. Furthermore, I confirm that I have contributed to this document equally as much as my teammates.

……..

Sameer Ghela

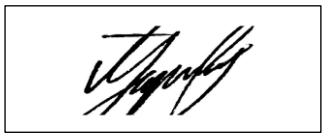
I, Sashin Gounden, confirm that I have read the requirement changes of the Platinum Island Resort and agree to all of the information contained in this document. I confirm that the information stated above is factual and that I am aware of any and all changes made to this document between drafts. Furthermore, I confirm that I have contributed to this document equally as much as my teammates.

……..

Sashin Gounden

7. Client sign-off

I Ya'qoob Tayob, on behalf of the Platinum Island resort, acknowledge that I have received and reviewed the work and confirm that it is up to quality standards.



Signature