Web Technologies Report

Group Members: Tharidu Jayaratne (tj17908) & James Lim (jl17056)

Our website is for a Spanish themed restaurant. The site runs on port 8080 and is hosted locally using XAMPP with APACHE and MYSQL enabled.

Setup and installation are described in readme.txt

Grade Estimates

We feel that we have achieved the following grades estimates for each section.

HTML – A+

We used XHTML to deliver every page and ensure they are valid. Furthermore, we have used various HTML tags to structure our content well and achieve aesthetically pleasing pages.

The following pages created are as follow:

Index.html: The main page of the website consisting of links to the menu and about page. In addition, there are links to book a table and view a booking already made. When booking a table, a form tag is used with method=POST, which send the form data to a php page login.php which handles said data. Finally, the link "Our Restaurants" redirects you to another page with various images used to represent different branches.

Menu.html: This page consists of different tabs for different courses of meals.

About.html: This page contains a few paragraphs and links with a PNG made background.

Locations.html: Contains mainly two div elements each holding six images which can vary in length when hovering over it.

seeProfile.html: Consists of a form that takes in a name and email and uses the POST method to send data to ViewProfile.php to check the database if the entered data exists or not.

CSS – A+

We created various stylesheets to ensure each page is delivered in the most visually pleasing way possible. We made use of text shadows for the titles of our main page and menu page. In addition, we used it for each item in the mains and desserts tab. Furthermore, media queries were used wherever possible to adjust our pages according to the size of the browser window. Moreover, we made use of the translate function to achieve a smooth animation of mains and dessert item descriptions. Also, with certain tweaks with the hover selector, we created small animations when hovering over buttons. Finally, we mainly used the translateX function and manipulation of the z-index to achieve the expanding effect. The !important property was used to ignore certain styling rules applied previously. Google API’s font Montserrat and Amazon’s S3 moon font was mainly used throughout all pages.

JavaScript – A

JS components were mainly used in our main page and menu page. In our main page when clicking on “Book A Table” we used event listeners and query selectors to display and close a form by extracting an html element through their id. Moreover, in the menu page, we used the js script script.js to switch between meal courses and have a fade in animation for each row of drink items. Finally, we initially had a preloader function to display an image whist a page was loading but decided to not use it in the final product as majority of the pages rendered quickly.

PNG – A

The PNG used in the about page was made from scratch through Inkscape. We used mainly the pen tool to create a few anchor points around the canvas and dragged certain points to create a Bezier curve to have a white space for navigation. We then filled certain areas of the canvas and used a gradient tool to create a linear interpolation of colours across a desired direction. We repeated this process to using an ellipse shape to overlay other shapes made and tweaked the opacity. For the bottom of the canvas we used the united tool to combine circles spaced out to create one shape. Finally, we trimmed out any unnecessary parts of the shapes we created.

SVG - A

Server – C

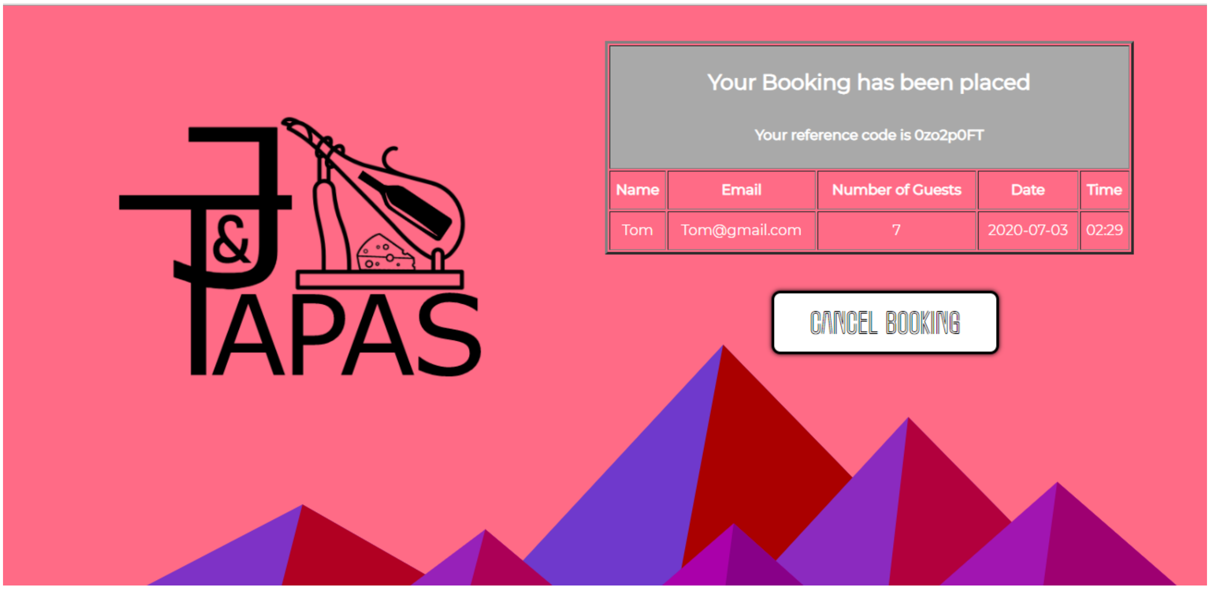
We used XAMPP, which is an open source web server which consists of interpreters for the scripts written in php that retrieve and manipulate data in the sql database. We used this as it installs Apache and MySQL, which makes it simpler to run our website in a local server.

Databases – A

We created an SQL database with the entries stored in TandJTapasDB.sql. When a form consisting of data entered by a user to make a booking the data is sent to login.php which calls dbh.inc.php. This file establishes a connection with the server and initializes a database and table. The database consists of one table which stores the user name, email, date, time and an incremental user_id. The login.php file assigns variables to the form data and accordingly inserts the data into the database, preventing repeated entries. A function in updateDB.php is used to parse in information about the database and update the sql database file.

Dynamic Pages – A

There are multiple dynamic pages implemented in our site where the server extracts information from the database and inserts them into html templates with CSS styling. For instance when you enter information to place a booking, you are redirected to bookingthanks.php, which takes in the latest entry to the database and displays it onto a table with a booking reference generated from a js function as shown below:

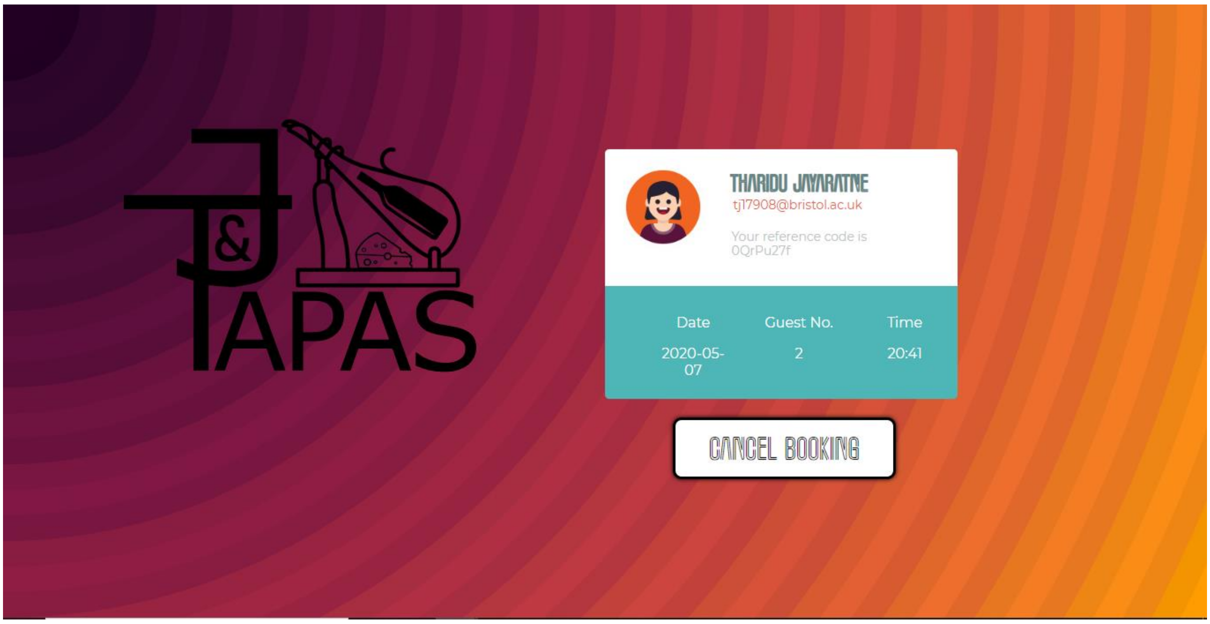


Your Booking has been placed

Your reference code is 0zo2p0FT

Name	Email	Number of Guests	Date	Time
Tom	Tom@gmail.com	7	2020-07-03	02:29

Furthermore, if you decide to view a booking that has been placed, you will redirected to seeProfle.html which takes in a name and email, sends it to ViewProfile.php and if a record is present it will display the booking details as shown below, otherwise an alert message is shown stating that the entered record is not present:



THARIDU JAYARATNE
tj17908@bristol.ac.uk

Your reference code is 0QrPu27f

Date	Guest No.	Time
2020-05-07	2	20:41

If you click on cancel booking, you will be redirected to the main page with your record removed from the database through delele.php.