System Requirement Specification

Preface

Version	Author(s)	Task	Description	Date
1	Everyone	All	Research other SRS documents and start tasks.	27/10/2016
1	Lucinda	Introduction		30/10/16
1	Francesco	Method		03/11/2016
1	Jamie, Lucinda, Mitchel	User Requirements	Interviewed potential stakeholders of the system and typed them up with rationales. Then organised and prioritised them.	04/11/2016 - 10/11/2016
1	Ben and Abraham	System Requirements		11/11/2016
1	Abraham	Glossary	Found all of the keywords within the SRS and defined them.	13/11/2016
2	Lucinda, Ben, Abraham	All	Refine each section.	18/11/2016

Introduction

This System Requirement Specification document (SRS) will discuss the overall requirements this system needs to be developed. The web based booking system for restaurants is essential because it allows people to book a reservation faster than the existing methods in place. People are currently having to call up restaurants or type in details into online forms to place a reservation. These methods all take time and effort. The system in discussion will provide faster services to people requiring its resources. The system will be called ‘Tablevation’, which incorporates the words, table and reservation. However, this name could change before implementation if any issues arise with it.

The system will function with the following:

- The ability to allow users to add a new account when they haven't got one.
- Users can log in using email address and a password.
- Give locations in a drop down list so users don't have to type out locations.
- Given locations will determine and generate the restaurants available.
- Provide restaurants in the area specified.
- Once logged in, no more user details will be required to make the reservation.
- Users can enter the number of people that will attend the reservation for the restaurant selected.
- Users can select the time and date of reservations
- Once booked, a confirmation email will be sent to the users email address given.

The system will be web based, so users will find it via its URL. With the appropriate domain name, Tablevation, users will be able to type it into any search engine using any device. Devices can range from smart phones to tablets so it must be responsive in all formats. From here, a login page will appear allowing people to enter their email and password, as well as a link to add a new account if the user is new. After the details are filled in, it will follow onto the home-page which allows the user to select the area they want, generating the restaurants in that specified area. Users can then select restaurants and provide the restaurant with the time, date and number of people who will attend. After confirming this, the users will be sent a confirmation email and be taken back to the home page. If the reservation is unavailable, then the user will be notified immediately and offered to change the time or another field.

There are very few similar systems out there. There is ‘Open Table’, ‘Book a Table’ and ‘Table Pouncer’ which offers the same services as Tablevation. All have a different approach where the user enters the location, number of people, time and date and then it generates bookings available from various restaurants. Whereas this system will allow users to select specific restaurants in the area before filling in

the details. This will also save time finding the restaurant the user desires. Similar functions include being able to use the location to generate restaurants in the area. Other websites have also used the drop down list to speed up the process of booking a table. Again, the same fields are required to book a table, date, time and number of people.

Glossary

Tablevation: The name of this system.

Method

In order to correctly write requirements, a model for the elicitation process is needed, this model is divided into these activities:

Requirements Discovery: To find the user requirements we will interview a range of people. The people that will be interviewed will be people who have previously used the current method of booking a reservation. To find this out, we will ask, 'Have you ever made a restaurant reservation before?', if the answer is 'yes', then the user qualifies to be interviewed. We will try and collect requirements from a range of sources differing in age, gender and career. During the interview, we will each show the user a copy of the introduction and ask them questions. The questions to ask are, 'Would you change any part of the system we propose?', 'What is important to you as a user with a system like this?', 'What unique qualities would you like this system to have?' and 'Is there anything you think will distinguish our system from others already existing?'. The answers will then be written up and any duplicate requirements will be removed. Another source of user requirements will be gathered from actual restaurant employees who will be affected by this system, so it is important we consider their requirements. Questions to ask restaurant employees will be, 'What would you like this system to do for your restaurant?', 'Would you like any limitations to user inputs?' as well as the previously stated questions.

Requirements Prioritisation and Organisation: In this phase, requirements will be prioritised, which will depend on the interactions with stakeholders. In this project, basic functionalities of Tablevation will be highly prioritised. The requirements will be placed in prioritisation categories that include, mandatory requirements, where it is compulsory they are implemented into the system. Desirable requirements, where the requirements will be implemented if it is feasible. Optional requirements will be requirements that could be implemented into the system. Finally, possible requirements, where they could be implemented if there was more time and money to do so in the future. Once prioritised, each requirement will be organised by importance within each category.

User Requirements

Mandatory Requirements

- The user must sign up in order to use the system, from here, the user enters personal details, such as name, contact number and email to proceed to the system.
- This requirement is important and will save the data for when the name and contact number is needed by the restaurant when a reservation is booked.
- The system will display a drop down list of sub areas within Portsmouth (e.g. Southsea), which will show available restaurants.
- This requirement should be implemented due to the available resources and time we have to create the system.
- Restaurant owners can create and manage an account associated with their restaurant, allowing the restaurant to keep track of bookings.
- This will help populate the site as the list of restaurants will be created by the restaurant owners themselves rather than having to manually list all restaurants in Portsmouth.
- The system will not permit users to book a table at the same time and date as another reservation already made using this system.
- This requirement has been included as it's impossible for a user to attend two reservations at the same time. This is important as it will be a form of validation.
- The restaurant should be able to update the availability of time slots and number of available spaces when other people book using different methods.
- This requirement is important because restaurants could have people booking tables in other ways (e.g. phone call).
- The user should be allowed to cancel a reservation booking up to the time slot of the reservation, which then makes the slot available to other users.

- It's important that the users can cancel bookings as it allows the restaurants to accommodate more table reservations and increase their revenue.
- The system will display the current available details of the restaurant they are looking at, this will include: contact number, opening times and address.
- This requirement should be included as without it the user wouldn't know where the restaurant they are booking is or the opening/closing times of that restaurant, also if anything were to happen (e.g. the system wasn't responding) the customer could call up the restaurant to manually book.

Desirable Requirements

- The system will have an option available for users on the login page to be emailed a link to reset their password.
- This is important to the system as a client who can't remember their password will not be able to access the system in order to use its services.
- The user can use a search bar to search specific restaurants which should be generated fast.
- This requirement will speed up the process of finding a specific restaurant.
- Users should be able to sort the restaurants by an Alphabetical order, Highest Ratings to Lowest and Lowest Ratings to Highest in a drop down list.
- This requirement is necessary as it will allow the users to speed up the process of finding the restaurant they would like to book a reservation at.
- When the user makes a reservation, they are able to amend it by either adding or subtracting the number of people as well as changing time and date of the reservation.
- This requirement is necessary as it will allow users to change bookings in case some people won't attend, thereby allowing the restaurant to not waste table space or time setting up for the wrong number of people.
- Users should be able to select the type of food they would like to eat on the home page where the location is selected. The selection will be in the form of a drop down list.
- Again, this will speed up the process of users finding the restaurant they would like to eat at. If they were specifically wanting to reserve an Indian restaurant then this will help them eliminate restaurants that they won't consider.
- The account passwords should be encrypted for security reasons.
- This will protect users by preventing any potential hackers from downloading a list of passwords.
- The users can report any problems with the site.
- This will help flag up any potential bugs and will allow for them to be fixed sooner.
- The site should be updated frequently with new restaurants and fixes to any errors found.
- This will ensure that security is up to date and any big bugs are found and fixed.

Optional Requirements

- Users are allowed to rate the restaurants out of 5 during their reservation time and after.
- Allowing users to rate restaurants will help give the users an idea where they would like to eat and what's worth visiting.
- When the user signs up, there should be an option where they tick whether or not they should receive emails about events or discounts restaurants are offering.
- Giving users this option might encourage them to use the system more often if they receive information about events like Wine Tastings that will require reservations to attend.

Possible Future Requirements

- When the user books a reservation, they can also enter details into an in input box. Details that can be entered cover dietary requirements, number of children that might require high chairs and birthday surprises like a cake to be kept a secret.
- This information will help out both the restaurants and the users as it will allow the reservation to be prepared for all of the user's needs.
- When the user selects a restaurant, it will provide the address of the restaurant and will allow you to link it to google maps using your GPS location when using a mobile or tablet device.
- The importance of this is to allow users to find the restaurants easily, this option should promote users using this system.

System Requirements

Functional Requirements:

- The system will contain a list of areas within the Portsmouth area.

    - The system will contain drop down menus showing the areas within Portsmouth.
    - The database will contain a list of restaurants within each area which will be displayed when a user selects an area.
- The database will contain information for each restaurant.
    - Every restaurant has a name, telephone number, address and opening/closing times.
- The system will allow customers and restaurant owners to create an account.
    - The system shall have an account creation page containing data fields for first name, last name, email, password, confirm password, contact number and whether the user is a customer or restaurant owner.
    - A restaurant owner is prompted to enter the name, address of the restaurant, opening/closing times and the Portsmouth area where it is located.
    - This data is stored onto the database. Is important to remember that password will be hashed before being stored into the database.
- The user can log into the system.
    - The user can enter email and password into data fields on the login screen.
    - The password is hashed and checked against the hashed password stored on the database associated with the entered email address.
    - If the check fails an error is displayed to the user showing that the credentials are incorrect. If the check succeeds the user is logged into the account associated with the email address.
    - The system should be able to recognise that the user is logged in for future requests.
    - The system will check if the account is normal user or restaurant and a different page will be displayed accordingly.
- The system will not allow users to book time spaces that are already booked at the same restaurant.
    - The system will check the user input against the data stored for the restaurant chosen by the user.
    - If user entry for desired time of booking is the same as an entry stored in the restaurants account then an error is output showing a time conflict.
- The user can cancel a booking.
    - The user can cancel a booking from their account after specifying a booking they wish to cancel, the field representing the number of free tables will be updated into the database.
- The user should be able to see the data of the restaurant they are currently viewing.
    - The restaurant name, address, style of food and image from the selected restaurant's account will be output on the table booking page.
- The restaurant should be able to set a table as booked/not free, and to remove a reservation.
    - The system should recognize if a table is booked or not, in particular should permit restaurants to change this state.
    - All the changes of states related to the reservations of tables, should be visible by customers and restaurants.
- The system will allow users to rate restaurants out of five stars.
    - The database shall contain an average score variable for each restaurant which is constantly updated by other users.
- The user can opt in or out of email promotions.
    - The user sign up page should have a checkbox which, when checked, marks on the user's account to include them in the mailing list.
- The user can alter their account settings.
    - The user settings page will have data fields for new password, confirm password and a check box for opting in/out of the mailing list. Once the user presses the confirm button the user will be prompted to enter their password for security.
    - If the password check succeeds the system will check if any data has been entered. If it has been it will check if the data is valid and then check if it is the same as the data stored on the user account in the database and if the data is different the old data will be overwritten. If the password check fails the system will inform the user that the password was incorrect.
- The system will have search functionality for finding specific restaurants by name.
    - The web page will have a search bar on each page allowing the user to input a restaurant name.
    - The system will use a binary search algorithm to check the entered string against the database. The restaurants which names match the input string will be shown on a search result page.
- The restaurant search results can be sorted alphabetically, or by average rating.

    - The system will collect all results and, upon the user selecting a sorting method, iterate through the collected results checking the selected variable then outputting the results appropriately.
- The login page will show a link for users who have forgotten their password.
    - The link will direct the user to a page with an entry box for their email
    - Once the user clicks confirm the system will check if the user has entered data and that the data corresponds with an account in the database. If so the system will email the user a link where they can reset their password.
    - If the email is incorrect or there is no data entered when the confirm button is pressed the system will output an error message telling the user the data is incorrect.
- The user can choose what style of food they wish to eat.
    - The restaurant owner will be prompted to enter the style of food served in their restaurant upon creation of the account.
    - This will be saved as a variable on the restaurant's account.
    - The search page will contain a drop down box showing different styles of food. Once one is selected and the search button is pressed the system will only display restaurants with the same type as the type entered by the user.
- The user can enter additional information at the booking page including the number of high chairs required for children or any specific dietary requirements.
    - There will be a text entry box on the booking confirmation page asking for any additional information. The text entered will be sent to the restaurant via the email stored on the restaurant account. If there is no text entered no email will be sent.
- The booking page will contain a google map link for the restaurant which is being viewed.
    - The restaurant owner can link their address to a google maps link.
    - The user can click the link and therefore map a route using google maps from their current GPS location to the restaurant.

Non-Functional Requirements

- The user passwords will be encrypted stored on the database as hashes for security purposes.
    - To test this we will try and ensure that the hashed passwords can't be decrypted easily.
- The user will be able to report errors on the site using an error report form.
    - The form will have a two data fields, title and description of problem, and will be converted into an email and sent to an administrator.
    - To ensure this works, we will report several errors and see if the email is sent and received.
- A restaurant search will take less than 5 seconds to return results.
    - Testing will involve timing how long it takes before all results are generated.
    - If a search takes over 5 seconds, an error report is logged and the administrator is notified.
- The website will display differently on mobile devices and tablets to allow for a more user friendly experience.
    - This will be tested by accessing it on multiple screen sizes and ensuring there is responsive design.
- The site will contain a navigation bar which will be the same on each page allowing for easy navigation through the site.
    - Testing user friendliness will involve counting the number of times user's get lost on the system until they reach their desired page.
    - Alterations will be made until the number of times they get lost are reduced to less than 4.
- The website will go through maintenance checks during midnight on a weekly basis so that the site can be updated.
- The website will be able to take up to 2000 users at a time on the site. This includes those accessing the site through mobile phone, computer, and other internet accessible devices.
    - To test this, it will include creating 2000 accounts and seeing if the system responds well to simultaneous use, if it reacts poorly then improvements will be made.
- The website will be available for all users to use from morning till midnight.
    - The only time where the website cannot be used is when the website is going through maintenance.
    - Testing this will be by accessing the website at all times of the day and checking if it is available.

Declarations:

- I, Lucinda Ashdown, declare my contribution to the Introduction and the User Requirements of this SRS document.

Lucinda Ashdown 

- I, Jamie Toloui, declare my contribution to the User Requirements of this SRS document.

Jamie Toloui

- I, Benjamin Harris, declare my contribution to the System Requirements of this SRS document.

Benjamin Harris

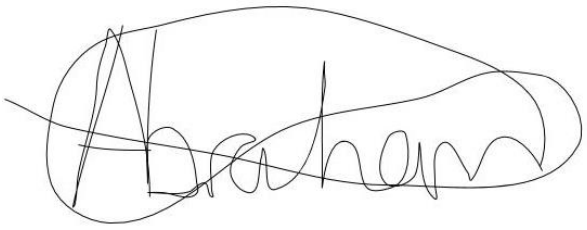
- I, Francesco David Nota, declare my contribution to the Method of this SRS document.

Francesco David Nota 

- I, Mitchel Tendai Gwaze, declare my contribution to the User Requirements of this SRS document.

Mitchel Tendai Gwaze *Mitchel Gwaze*

- I, Abraham Nwokolo, declare my contribution to the Glossary and System Requirements of this SRS document.



- Abraham Nwokolo