UTS Fine Dining

Online Reservation and Ordering

System Requiremens

Contents

Online Restaurant Table Reservation system

1. Product Description

UTS's Fine Dining Restaurant (UTSFD) aims to be one of the most prestigious restaurants in Sydney. The strategy is to create a simple, elegant yet friendly and vibrant environment, not just for its food but to act as a social gathering place for UTS students, academics and their friends and families of people.

To establish this UTSFD has decided to create a web site, both as an advertisement for UTSFD's offerings but also as a place where patrons can register and book tables. A second phase of the project will include table service automation with links to the kitchen and electronic payment methods.

The business team wish to avoid the more common problems:
- Too many people needed manage bookings and day to day management at the expense of the level of service in the restaurant.
- Easing the workload for staff in all capacities.
- Failure to present food and drinks in a professional manner.
- Loss of interaction with patrons after they leave the restaurant.
- Not allowing customers to reserve any table in the restaurant without the need for phone calls.
- Increased costs.

With the aim for solving the above problems UTS have employed a student group to develop the UTSFD reservation system.

This will be an agile development with the first phase addressing advertising, patron registration and table booking. Later phases will include electronic ordering to the kitchen and billing systems.

1.1 Product Vision

The new system aims to increase efficiency to smoother work flow of the restaurant so as to provide top-notch dining experience and service to the customers. It also aims to reduce overheads by avoiding manual system.

1.2 Business Requirements

The first version of the system must be available within three months.

1.3 Stakeholders and Users

Management – The UTS Program Management Board (PMB) are the owners of the new system. Weekly status update meeting will be held to communicate the progress of the project to the management PMB.

Purchaser – UTS.
User – Patron of UTSFD; management and staff of UTSFD.
Developers – The eight-member development team which includes one project manager, two programmers, two software engineers, two database analysts and one designer.
Staff – Restaurant Manager and Restaurant Supervisor who maintain and update the data such the menu and gallery pages.

1.4 Project Scope

The scope of this project is to develop a responsive web site integrating membership, and Online Table Reservation. This system allows patrons to choose their desire seats online based on the restaurant floor plan and availability.

1.5 Assumptions

Patrons will register their before they can make a booking.
There will be a set number of tables which will not be changed.
Some areas will be available for large groups.
Patron registration information will be available to the marketing team.
Bookings will be reviewed and confirmed by a UTSFD manager before confirmation
The first phase will not include financial transactions but the framework must be created to facilitate the payment of a deposit for a table booking and eventually the creation and payment of the bill.

1.6 Constraints

The system must work on any mobile device
Patron information must be secure and encrypted for commercial and privacy reasons.
Patron registration information will be automatically validated by sending a 'confirmation of registration' email. Failure to reply to the email within 24 hours will result in the information being deleted from the system.

2. Functional Requirements

2.1. General

2.1.1. The user shall only be able to perform the following operations:
- i. For customer:
  - a.   Make a reservation
  - b.   Browse menu
  - c.   View galleries of food, drink and the restaurant
  - d.   Modify/cancel reservation
- ii. For restaurant personnel:
  - a. View database

2.1.2. Navigation will allow a patron to go to any part of the web site.

2.2. Make a Reservation

2.2.1. A patron must log on to the system before making a reservation.

2.2.2. When the patrol initiates 'Make a booking', he/she must be taken to the 'Booking' page.

2.2.3. The user must be able to enter the following information
- a. Time and date
- b. Number of people

2.2.4. Once the user submits the information, he/she will be taken a page that shows the floor plan of the restaurant.

2.2.5. The page must show all the available table(s) that fit the requirements in 2.2.2 in yellow. Unqualified table(s) will be in red.

2.2.6. The user must be able to choose the table(s) in yellow only and using the mouse by clicking on the table. The selected table will be in green.

2.2.7. The user must be able to deselect the table by clicking the selected table again and the table will be in yellow again.

2.2.8. Once the requested table(s) is selected, the user must be able to click the 'Confirm' button to proceed.

2.2.9. The user must be able to enter the following information:
- a. Name (between 1 to 32 characters)
- b. Contact number
- c. Email address

2.2.10. Once the information is submitted, the selected table(s) and the information must be updated in the table reservation database.

2.2.11. If the update fails, the system must show an error message to the user notifying him/her of the failure.

2.3. Browse Menu

2.3.1. The web site must display the menu that is updated to the current day and must be coherent to the menu used in the restaurant.

2.3.2. The menu page must display the menu in tabs with the following categories:

       i.   Main course

      ii.   Entree

    iii.   Beverages

    iv.   Dessert

2.3.3. Each page must scroll so all items are available.

2.3.4. User must be able to navigate to different parts of the system using menu options at the top of each page.

2.4. Special Request

2.4.1. The system must allow a patron to identify the booking as a special event with the following categories:

    a. Birthday

    b. Wedding anniversary

    c. Graduation

    d. Farewell

    e. Free format description

2.5. Modify/ Cancel Reservation

2.5.1. The system must allow the user to modify or cancel reservation at "Modify/Cancel Reservation" Page.

2.5.2. The user must be logged in and can only change their own bookings.

2.5.3. The system must display the bookings with options to:

    a. Modify a booking

    b. Cancel a booking

2.5.4. The system must allow the user to modify any aspect of the booking (see above).

2.5.5. The system must update the database with the adjustments.

2.5.6. If the adjustment fails, the system must show an error message to the user notifying him or her of the failure.

2.5.7. If the update is successful, the system must display the following information.

    a. New Transaction ID

    b. Time and Date

    c. Number of people

    d. Table(s) reserved

    e. Special event (if any)

2.5.8. The system must display the following information of the reservation:

    a. Transaction ID

    b. Time and Date

    c. Number of people

    d. Table(s) reserved

    e. Special request(s) (if any)

2.5.9. The user must be able to cancel the reservation by clicking "Cancel Reservation".

2.5.10. Once the user selects "Cancel Reservation", the system must update the database with the adjustments.

2.5.11. If the adjustment fails, the system must show an error message to the user notifying him or her of the failure.

2.6. View Database

2.6.1. The user must choose the time and date in a drop down menu.

2.6.2. The system must display the database in tabs with the following categories:

   a. Tables reserved
   b. Wedding anniversary
   c. Special event
   d. Other (free format)

2.6.3. The system must display the floor plan showing the status of the table:

   a. Green table means reserved
   b. Red table means not reserved

2.6.4. The system must allow the user to view the following about a reserved table from the floor plan by clicking on the table:

   a. Customer name
   b. Customer contact number
   c. Number of people
   d. Link to the special request.

2.6.5. Once viewing of the database is done, the user must be exit the database by clicking on the "Exit" button.

3. Data Requirements

Data requirements describe the format, structure, type, and allowable values of data entering, leaving, or stored by the product.

3.1 The system will only accept data which are correct and not ambiguious. E.g email addresses must be correctly formatted; phone numbers must be validated.

3.2 The booking can only be submitted and processed by the system when all required fields of data have been entered.

3.3 The system should display all times in the 24-hour clock format.

3.4 The system must store customer names in fields recording first and last name.

3.5 When a customer has selected a table for booking but has not yet confirmed his booking, the system will lock out that particular table to other customers for 10 minutes.

4.   Non-functional requirements

There are requirements that are not functional in nature. Specifically, these are the constraints the system must work within.

4.1   Compatiblity
- 4.1.1   The website should be compatitible with all major web browsers on PC, Apple, Linux, SmartPhone, Table or any mobile device.

4.2 User interface
- 4.2.1   The user interface should be user friendly and intuitive e.g., it follows the UI guidelines for naming menus, buttons, and dialog boxes whenever possible.

4.3 Security
- 4.3.1   Access will be controlled with usernames and passwords
- 4.3.2   Only administrator users will have access to administrative functions, average users will not.
- 4.3.3   Database should be secured to prevent leak or loss of confidential information such as credit card details from customers..

4.4 Performance
- 4.4.1   The system should be up and running 24/7.
- 4.4.2   It should support at least 100 users using the online booking concurrently without any lag.

4.5 Backup and Recovery
- 4.5.1   There should be a backup server and database to prevent service interuption or loss of data when the main server and database are down.
- 4.5.2   Downtime should not last more than 30sec when switching from the main system to the backup system in case of a breakdown.

4.6 Reliability
- 4.6.1   The whole online booking system should achieve a 99.9% availability. i.e downtime should not be more than 0.1% of its total operating time.
- 4.6.2   System review will take place monthly. Any lack in performance or reliability will be addressed and improved on after each review.

4.7 System Maintainence
- 4.7.1   Maintenance of the system will be conducted weekly.
- 4.7.2   Maintenance will be conducted during off-peak hours e.g between 12am - 6am

5.   ***Interface Requirements***

Overview

The user interface of this restaurant booking system is a web site which can be viewed using popular web browsers. This high accessibility made it easier and more convenient for users to use the system. Users don't need to set up any additional software for the purpose of running the system. As long as an Internet connection is available, the system can be easily accessed using their mobile devices. Multi-platforms operation is also an additional advantage of this design.

One more advantage of this design is the power of the Hyper Text Markup Language (HTML). HTML provides nicer features with simple modification and configuration compared to the GUI of other languages. HTML language supports the use of other languages and technique to make dynamic objects, which can improve the vividness of the application.

5.1   User Interfaces

These are the fundamental features of the GUI that should be included in the websites:

A login box comprises of an account and a password text field. Users can sign in using their NRIC to check their bookings. We can provide the sign up function for long-term users so that they don't have to refill the information everytime booking is made.

A dynamic menu including the links to the homepage, the menu page, the booking page and the information page.

The menu page will have the list of food with its respective image. It can be divided into many pages to ease up the navigation. The booking page will have a clickable map for the reservation of seats. After booking is submitted, the webpage will automatically redirect to a 'thank you' page. The information page will provide additional information about the restaurant.

A slideshow or a flash of the images of the restaurant.

Images of the top ordered dishes and their respective information (e.g price, ..).

A panel for advertisements coming from our own restaurant or from other parties.

5.2   Hardware Interfaces

Describe how the software application interfaces with hardware *that exists outside the scope of the system*.

5.3   Software Interfaces
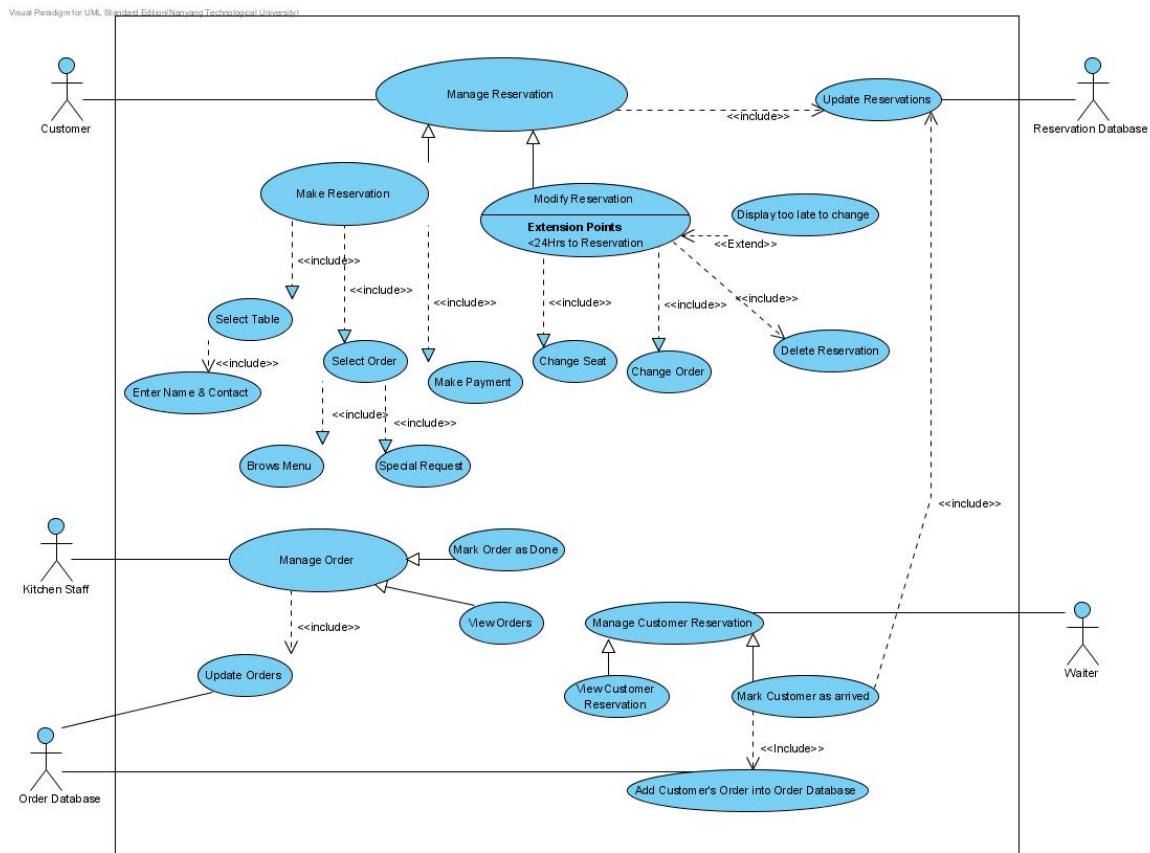
The use of web design tools such as Adobe Dreamweaver is employed to make a more professional and nicer design of the system. The code editor and the design editor is integrated in one tool, which allows easy modification as well as addition of elements onto the web pages. Interactive and dynamic objects can be created more easily within a few clicks. The platform to implement the webpage is php and mysql with the support of Apache. Another platforms to be considered are jsp, serverlet using netbean, asp.net and C# using Visual Studio. However, PHP is chosen due to its popularity, ease in coding and the availability of free scripts online.

To edit the images and make the flash, it is recommended to utilize Adobe Photoshop, Flash SlideShow maker and Adobe(Macromedia) Flash Player. This requires some code to make the dynamic contents. It's also used to create icons and graphics to enhance the interface.

6.   *Use Case Model*
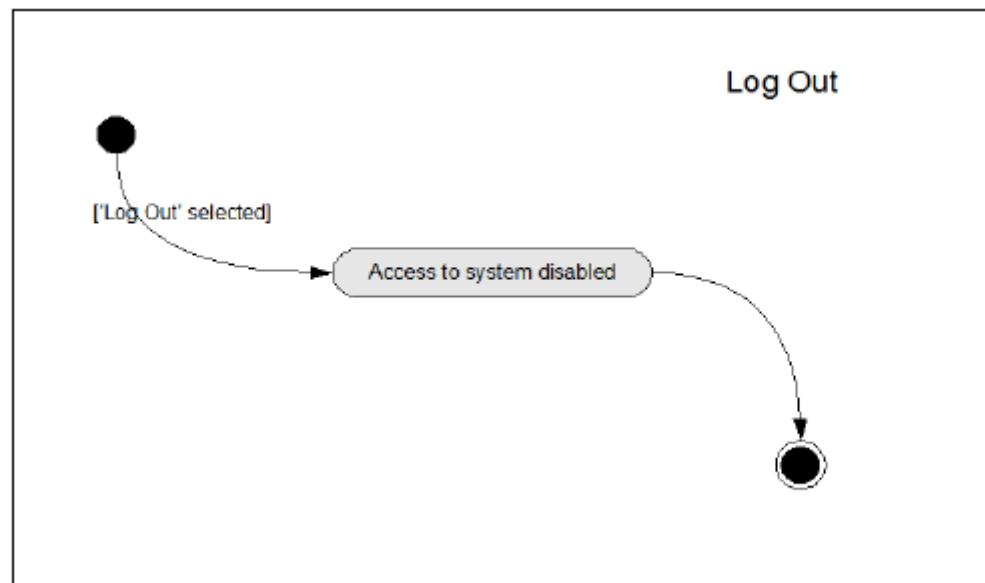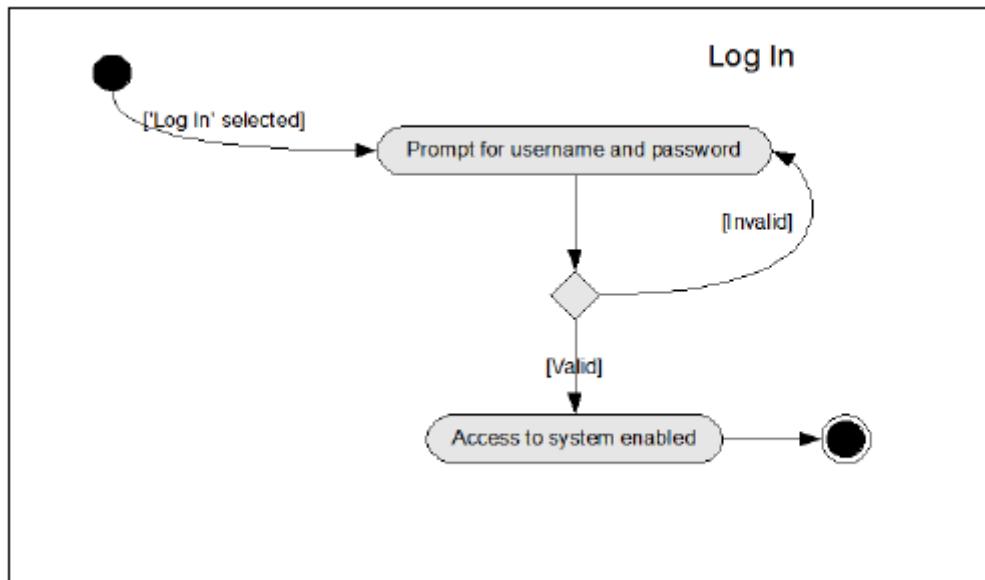
Provide the top-level use case diagram, followed by the use case description for each use case.

6.1   Use Case Diagram

6.2   Process diagrams

7. Glossary

Define all terms and acronyms required to interpret the SRS properly. This is the (problem) domain dictionary.

8. References

Provide a list of all documents and other sources of information referenced in the SRS and utilized in developing the SRS. Include for each the document number, title, date and author.

Document No.	Document Title	Date	Author

9. Revision History

Identify changes to the SRS.

Version	Date	Name	Description