Project Proposal and Project Plan

Introduction

This Project will be a web based Booking System for Restaurants. The motivation behind the project is to eliminate customers wasting time calling a restaurant, or constantly retyping their details online to reserve a table. The purpose is to decrease, the time taken to book a reservation.

Objectives:

- Create a User Friendly Website which adapts to phone screens
- Allow users to select specific restaurants in their city
- Users to be notified when a restaurant is fully booked
- Users only having to log in once and not repeat their details to book a table
- The project must be faster than the current methods of reserving a table
- The project must be completed by the 28th April 2017

The constraints of the project will be time management, group knowledge and project complexity. Time management is important as the deadline is the 28th April 2017, where the project needs to be completed. The constraint of this, is being able to complete our individual set tasks of the project without extending our scheduled deadlines. This could be caused by poor management of other units being studied, illness or even miscommunication on group meeting times. The solution to this is to make sure we consider everyone's timetables towards the scheduling of tasks to make it as flexible as possible. For group knowledge, people have various experience in different topics, which will require extra research to complete the project. To help solve this constraint, we have organised our team into their strengths in the specific fields required. Finally Project Complexity, some areas within the project will be very complex to complete without extra time, resources and research. Therefore within the project schedule, extra time will be given to do the research for each task. We will try to assign two people to a task, so that the work will be done more efficiently.

Project Organisation

In order to ensure efficient completion of each task to time constraints, the group will be divided according to each individual's strengths and weaknesses. This will allow the work to be divided up so that multiple tasks can be worked on simultaneously.
The project is broken into six stages with varying numbers of tasks at each stage. For a complete breakdown of the assigned tasks see figure 1 at the bottom the document on page 5.

Risk analysis

Risk description and type	Risk impact	Risk Prob-ability	Mitigation	First indicator that the risk is turning into an issue	Effect
Data Loss	This could lead to loss of files	3/10	Periodic online backup and usb backup	One or more files are lost.	Catastrophic
Absence of technology knowledge	Loss of time for learning this technologies .Exponential	6/10	Technology research throughout all process of	Subsystem deadlines are not being respected	Serious

	increase of the amount of work.		development of the system.		
Time used for other modules of the year or time deadlines not respected	More hours of study every day, possible failure in submission of the final system.	4/10	Planning of time for the entire year	Time deadlines are not respected	Tolerable
Illness or Personal problems	Increase of stress and work for the entire group	3/10	Try to anticipate the work related to time deadlines	Time deadlines are not respected	Tolerable
System components do not interact as expected	System revision and change of requirements , design , development and functionality testing	6/10	Carefulness while designing components	The application functionalities are not working	Serious
Communication problems within the group	Increase of system delays and eventually system failure	5/10	Respecting group ideas and collaborate in order to resolve problems	The system does not reflect group ideas or is not working properly	Catastrophic
The size of software is underestimated	Increase of the total hours of work	6/10	Write system requirements in detail and use reusable components	Application design is bigger than expected	Serious
System lack in performance	Loss in usability of the application	6/10	Decide what data structures are more appropriate for every subsystem that require the control of a great amount of data	System is slow while executing operations	Tolerant
Reusable software components do not work properly in the system	Loss of time , design and development of the components	5/10	Analyze design of reusable components and avoid the one that are not written by experts	The components does not work in the application	Serious

Google docs is not working	Loss of time	2/10	Back up data regularly	Docs is not working for more than one day	Tolerant
Problems in understanding MVC pattern	Loss of time,change of roles	5/10	Research and analyze document written by experts in web development and MVC pattern	The pattern is not used in the proper way in the design or implementation phase	Serious
Problems in understanding the iterative approach	Loss of time,change of roles and if possible and convenient evaluation of other approaches	4/10	Study the iterative approach for different and trusted resources	The process is not being developed in an iterative way	Serious
University IT system goes down	Use different communication systems	3/10	Use github in order to have updated software	University system goes down for more days	Tolerant
One or more group people need assistance with individual tasks	Project work effort is increased	4/10	Communicate and be honest on the topics that are not clear	Time deadlines are not respected	Serious
Members using the bus miss it and turn up late	This delays progress of each individual task.	4/10	Try to arrive earlier than scheduled	When tasks don't get completed on time.	Tolerable
Members of the group can have a dispute	Tasks might be delayed if the member does not get their way.	3/10	Give every member equal contribution towards the tasks.	When a deadline is not met on certain tasks.	Serious

Resource Requirements

For design documentation, the resources we will need to create our Use case diagram and Sequence Diagram is Microsoft Visual Basic. Using the skills we gained last year creating these diagrams for Databases, we will already have an understanding on what to do, which will reduce research time into these tasks. For the mock-up of the system UI we can again use Visual Basic, or even Photoshop to create the interface a user will see.

When it comes to the implementation of our database, we will write out our SQL code in notepad++. This will then allow errors to be amended and the data to be easily entered into our Virtual Machines using Putty.

For the website (front-end) we will use atom.io which is a multi-language editor. JavaScript will be used for the functionality of the website. To do so, we will use a learning based website called 'Java the Right Way' to help our understandings on how it works.

When it comes to accessing our code, we will use GitHub. This allows all members to view and edit the code from many different locations, whether it be at home or at university. Google Docs will also be used throughout the project to allow shared access of the up to date document being amended.

To record the demo video of the final project, we will use free screen recording software which will capture the way a user can interact with the Booking System. Voice recordings will be placed on top to have a narration on how it all works.

Work Breakdown

By using the Incremental Development model, we have introduced all the stages and their deliverables required in each activity. Figure 2 can be found on page 6 illustrating the model used.

- **Requirements**
- **Analysis**
  - Plan
  - Research in technologies
    - **Client side**
      - CSS
      - HTML
      - JavaScript
    - **Server side**
      - MySQL
      - Java Servlet
    - **Web design patterns**
- **Design**
  - **Basic subsystems interaction design**
  - **Client side design**
    - Components design
    - Components interaction design
  - **Server side**
    - Components design
    - Components interaction design
  - **Database design**
- **Implementation**
  - **Structure**
    - Structured programming/ structured coding
  - **Functions**
  - **Analysis**
- **Testing**
  - **System will be tested by a chosen software**
- **Hand-in**
- **Evaluation of maintainability and reusability of system components**
- **Documentation**

Project Schedule

On pages 7 to 8 you can see figures 3.1 and 3.2 which present the schedule our group will follow. As you can see, some tasks overlap which we decided to do as members are working on different tasks allowing simultaneous completion. Documentation and Research is almost carried out throughout the entirety of the project as they are areas that will be worked on continuously.

Monitoring and Reporting Mechanisms

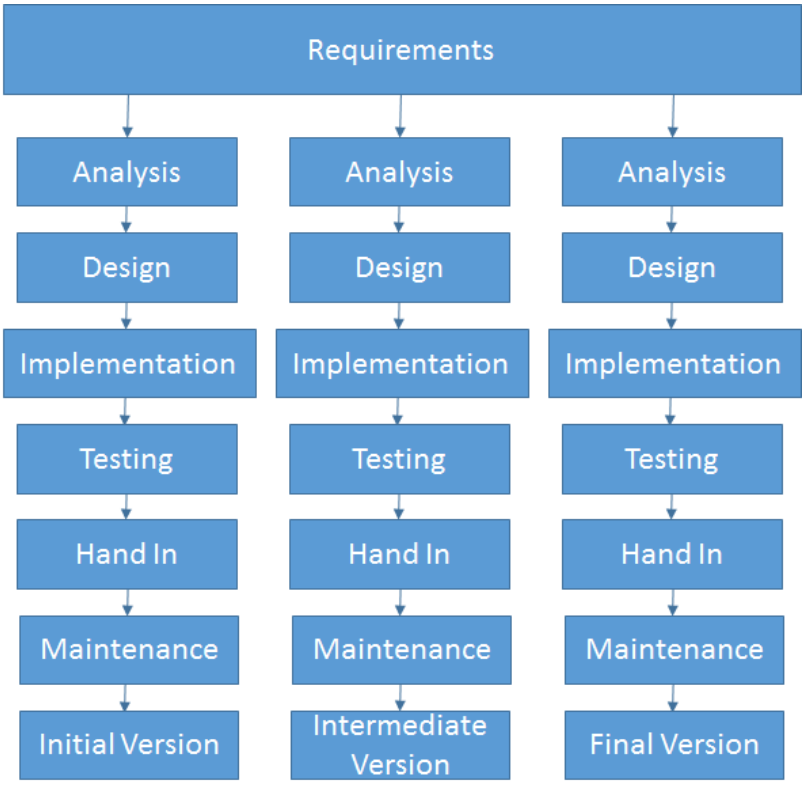
To make sure the project is on schedule, the following strategies will be introduced. Weekly meetings to discuss progress and to assist those requiring help. Using instant messaging services the group are able to communicate any arising issues that could affect development deadlines. 6 Individual deadlines for each coursework hand in throughout the Project from the 14th October 2016 to 28th April 2017. The Project Schedule will be closely monitored and will be used to schedule all activities throughout the year. Formative feedback will be received twice and used to refine the project. Google docs will be used to document all aspects of the project which can be accessed by all members of the group. This will allow members to access and amend the project from any computer whilst keeping the rest of the group updated. GitHub will also be used to store the program and provide home access to all members allowing improvements to be made.

Figure 1: Task Assignments

Job	Member(s)
Project Proposal and Plan	
Introduction	Lucinda, Abraham
Project organization	Ben
Risk analysis	Francesco
Resource requirements	Jamie
Work breakdown	Mitchel
Project schedule	Mitchel
Monitoring mechanisms	Lucinda
System Requirements Specification Document	
Version details with previous versions	Everyone
Introduction, describing system functions	Ben
Glossary of technical terms	Lucinda
Method used to achieve requirements	Francesco
User requirements	Mitchel
System requirements	Abraham, Jamie
Design Documentation	
Use case	Lucinda

Sequence diagram	Flexible (Everyone)
System architecture	Everyone
Mock-up of system UI	Ben
Implementation	
Database	Lucinda, Mitchel
Website	Abraham, Jamie
Programming	Ben, Francesco
Prototype video	Lucinda, Mitchel
Testing and Documentation	
Testing strategies	Lucinda
Testing process	Everyone
Results of testing process	Everyone
Usability evaluation	Ben
Final Report	
Video demo	Ben, Jamie
Retrospective report of project	Everyone
System documentation	Everyone
Export source code	Francesco

Figure 2: Incremental Development Diagram



Task Name	Task Number	Duration	Start	Finish
User Requirements	1	4 days	Fri 21/10/16	Wed 26/10/16
System Requirements	2	4 days	Fri 21/10/16	Wed 26/10/16
Competitor Research and Analysis	3	7 days	Thu 27/10/16	Fri 04/11/16
Research in Technologies	4	92 days	Thu 27/10/16	Fri 03/03/17
Sub-system Design and System Interaction Design	5	7 days	Sat 19/11/16	Mon 28/11/16
Graphical User Interface Design	6	5 days	Wed 23/11/16	Tue 29/11/16
Client Side Components Design	7	7 days	Fri 25/11/16	Mon 05/12/16
Server side Components Design	8	6 days	Sun 27/11/16	Fri 02/12/16
Database Design	9	10 days	Sun 27/11/16	Thu 08/12/16
Graphical User Interface Implementation	10	14 days	Sat 14/01/17	Wed 01/02/17
Database Implementation	11	10 days	Sat 14/01/17	Thu 26/01/17
Client Side Components Implementation	12	14 days	Thu 26/01/17	Tue 14/02/17
Server side Components Implementation	13	17 days	Thu 26/01/17	Fri 17/02/17
Unit Testing	14	10 days	Tue 31/01/17	Mon 13/02/17
System Testing	15	14 days	Tue 14/02/17	Fri 03/03/17
Evaluation of Reusability	16	7 days	Mon 17/04/17	Tue 25/04/17
Evaluation of Maintainability	17	7 days	Thu 20/04/17	Fri 28/04/17
Documentation	18	141 days	Fri 14/10/16	Fri 28/04/17

Figure 3.1: Project Schedule

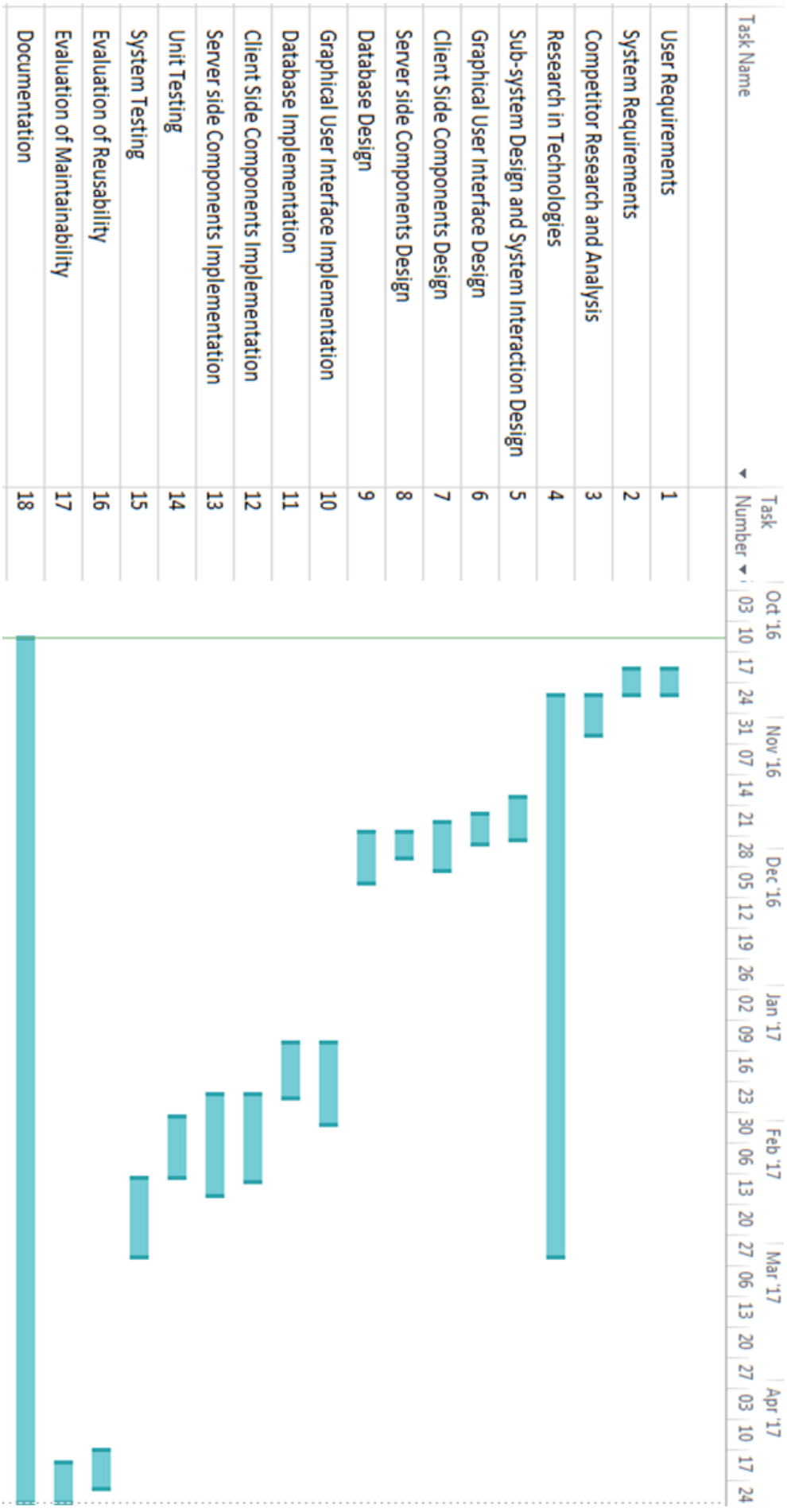


Figure 3.2: Project Schedule Timeline