**CHAPTER-1**

**INTRODUCTION**

* 1. **INTRODUCTION TO PROJECT**

This project is designed to facilitate a gymming and fitness center to automate its operations of keeping records and store them in form of a large and user friendly database further facilitating easy access to the personnel.

**Reason for the Project**

Organization Overview

The Gym Management requires a system that will handle all the necessary and minute details easily and proper database security accordingly to the user. They requires software, which will store data about members, employees, products, payroll, receipts of members etc & all transactions that occur in Gym and lock-up with graphical user interface(GUI).

Objective of the Project

1. The main objective of the project is to design and develop a user friendly system.
2. Easy to use and efficient computerized system.
3. To develop an accurate and flexible system, it will eliminate data redundancy.
4. Computerization can be helpful as means of saving time & money.
5. To provide better graphical user interface.
6. Less chances of information leakage.
7. Provides security to data by using login & password.
8. Scope of the Project
9. Storing information of members, employees.
10. Check validity of information provided by user.
11. Storing information of members according to their id.
12. Generating reports for different id.

Introduction to Project

We have done a project on Gym Management and database management and transactions. This system is proposed to be an automate database management & transactions. This stores employee, member, payroll, receipts, and products information. It also provides the facility of search & advanced search for searching the records efficiently & immediately. This system provides data storing & report generation with graphical user interface (GUI).

System Study

It is always necessary to study and recognize the problems of existing system, which will help in finding out the requirements for the new system. System study helps in finding different alternatives for better solution.

The project study basically deals with different operations and steps involved in generation of examination mark sheets. Ti includes:

1. Data gathering

2. Study of existing system

3. Analyzing problem

4. Studying various documents

5. Feasibility study for further improvements

Following are the steps taken during the initial study:

Initially, we collected all the information, which they wanted to store. Then we studied the working of the current system which is done manually. We noted the limitation of that system which motivated them to have new system.

With the help of these documents we got basic ideas about the system as well as input output of the developed system.

The most important thing is to study system thoroughly.

Here we are studying both existing system and proposed system so that advantages & disadvantages of both the systems can be understood

The first task was identifying how system can be computerized. Some analysis and projections was done regarding changes to be made to the existing system.

The new developed system for Gym Management is simple without complexities.

Existing System

The gym is working manually. The current system is time consuming and also it is very costly, because it involves a lot of paperwork. To manually handle the system was very difficult task. But now-a-days computerization made easy to work.

The following are the reasons why the current system should be computerized:

 To increase efficiency with reduced cost.

 To reduce the burden of paper work.

 To save time management for recording details of each and every member and employee.

 To generate required reports easily.

Limitations of existing system:

Time consumption:

As the records are to be manually maintained it consumes a lot of time.

Paper work:

Lot of paper work is involved as the records are maintained in the files & registers

Storage requirements:

As files and registers are used the storage space requirement is increased.

Less reliable:

Use of papers for storing valuable data information is not at all reliable.

Accuracy:

As the system is in manual there are lot many chances of human errors. These can cause errors in calculating mechanism or maintaining customer details.

Difficulty in keeping new records:

It is difficult for keeping all the new entries of members, their account and transaction details.

Proposed System

The proposed system is managed by the visual basic 6.0, which are user friendly windows for every user and for maintaining the database Microsoft access is used.

Scope of proposed system:

The system proposed has many advantages.

1. The proposed system is highly secured, because for login the system it requires the username and password which is different for each department therefore providing each department a different view of the customer information.

2. It provides wide range of certain criteria in each window the client is working for better and quicker solution.

3. It maintains report for all criteria and transactions.

4. Manages member information separately for all exercise and employee information separately for considering the requirements of gym.

5. Stores information about regular products.

6. This system can run on any windows operating system.

SYSTEM ANALYSIS & DESIGN

The way that is followed while gymrying on with the development application is as follows

Phase I (defining a problem)

Defining a problem is one of the important activities of the project. The objective is to define precisely the business problem to be solved & thereby determined the scope of the new system. This phase consist of 2 main tasks. The 1st task within this activity is to review the organization needs that originally initiated the project. The 2nd task is to identify, at an abstract or general level, the expected capabilities of the new system. Thus, it helps us to define the goal to be achieved & the boundary of the system. A clear understanding of the problem will help us in building a better system & reduce the risk of project failure. It also specifies the resources that have to be made available to the project.

Three important factors project goal, project bounds & the resource limits are sometimes called the project’s term of reference.

Phase II (feasibility study):

The first study aspect is whether the current project is technically feasible i.e. whether the project be gymried out with the current equipment, existing software and available personnel. If new technology is required than what is the likelihood that it can be developed?

The second study aspect is whether the project is economically feasible i.e. are there sufficient benefits in creating the system to make the cost acceptable. Are the costs of not creating the system so great that the project must be undertaken?

The third study aspect is whether the project is operationally feasible or not i.e. whether the system will be used if it is developed and implemented? Project is worth developing only if it can meet institutions operating requirements.

The feasibility study proposes one or more conceptual solutions to the problem set for the project. The objective in assessing feasibility is to determine whether a development project has a reasonable chance of success. It helps us to determine the input & output of the system. The following are the criteria that are considered to confirm the project feasibility.

The following feasibility study was undertaken for the proposed system:

Technical feasibility:

At first it’s necessary to check that the proposed system is technically feasible or not & to determine the technology and skill necessary to gymry out the project. If they are not available then find out the solution to obtain them. Hardware is already available in the collage.

Economic feasibility:

While considering economic feasibility, it is checked in points like performance, information and outputs from the system. MS Access is available in one package of the windows operating system & does not require additional software cost for the client tools. The cost incurred to develop the system is freeware & does not incur the cost to the project. Backend database technology is a freeware. This justifies economical feasibility of the system.

Social feasibility:

Although generally there is always resistance, initially to any change in the system is aimed at reliving the work load of the users to extent the system is going to facilitate user to perform operations like calculating salary amounts and deductions, generating reports with less possible errors. Thus there is no reason to make system socially unfeasible.

Operational feasibility:

The operational feasibility is obtained by consulting with the system users. Check that proposed solution satisfies the user needs or not. There is no resistance from employee since new system is helpful. The existing system is manual system, while the new system is computerized and extremely user friendly.

Software details of the proposed system:

 Front End:- Visual Basic 6.0

 Back End :- MS Access

Phase III (System Analysis):

The phase is detailed appraisal of the existing system. This appraisal includes how the system works and what it does. It also includes finding out more detail- what are the problems with the system and what user requires from the system or any new change in the system.

The output of this phase results in detail model of the system. The model describes the system functions & data & system information flow. The phase also contains the detail set of user requirements are used to set objectives for new system.

System study:

It is always necessary to study and recognize the problems of the existing system, which will help in finding out the requirements for new system. System study helps in finding different alternatives for better solution.

The project study basically deals with different operations and steps involved in generation of examination mark sheets. It includes:

1. Data gathering

2. Study of existing system

3. Analyzing problem

4. Studying various documents

5. Feasibility study for further improvements

Following are the steps taken during the initial study:

 Initially, we collected all the information, which they wanted to store.

 Then we studied the working of the current system which is done manually. We noted the limitations of that system which motivated them to have a new system

 Then we analyzed the format the reports generated by the system.

With the help these documents we got basic ideas about the system as well as input & output of the developed system.

The Spiral Model:

The spiral model, originally proposed by Boehm, is evolutionary software process model that couples the iterative nature of prototyping with the controlled and systematic aspects of the linear sequential model. It provides the potential for rapid development of incremental versions of the software. Using the spiral model, software is developed in a series of incremental releases. During early iterations, the incremental release might be a paper model or prototype. During later iterations, increasingly more complete versions of the engineered system are produced. A spiral model is divided into a number of framework activities, also called task regions.6 typically, there are between three and six task regions. Figure depicts a spiral model that contains six task regions:

• Customer communication—tasks required to establish effective communication between developer and customer.

• Planning—tasks required to define resources, timelines, and other project related information.

• Risk analysis—tasks required to assess both technical and management risks.

• Engineering—tasks required to build one or more representations of the application.

• Construction and release—tasks required to construct, test, install, and provide user support (e.g., documentation and training).

• Customer evaluation—tasks required to obtain customer feedback based on evaluation of the software representations

created during the engineering stage and implemented during the installation stage. Each of the regions is populated by a set of work tasks, called a task set, that are adapted to the characteristics of the project to be undertaken. For small projects, the number of work tasks and their formality is low. For larger, more critical projects, each task region contains more work tasks that are defined to achieve a higher level of formality. In all cases, the umbrella activities (e.g., software configuration management and software quality assurance) noted is applied. As this evolutionary process begins, the software engineering team moves around the spiral in a clockwise direction, beginning at the center. The first circuit around the spiral might result in the development of a product specification; subsequent passes around the spiral might be used to develop a prototype and then progressively more sophisticated versions of the software. Each pass through the planning region results in adjustments to the project plan. Cost and schedule are adjusted based on feedback derived from customer evaluation. In addition, the project manager adjusts the planned number of iterations required to complete the software. Unlike classical process models that end when software is delivered, the spiral model can be adapted to apply throughout the life of the computer software. An alternative view of the spiral model can be considered by examining the project entry point axis, also shown in Figure. Each cube placed along the axis can be used to represent the starting point for different types of projects.

A “concept development project” starts at the core of the spiral

and will continue (multiple iterations occur along the spiral path that bounds the central shaded region) until concept development is complete. If the concept is to be developed into an actual product, the process proceeds through the next cube (new product development project entry point) and a “new development project” is initiated. The new product will evolve through a number of iterations around the spiral, following the path that bounds the region that has somewhat lighter shading than the core. In essence, the spiral, when characterized in this way, remains operative until the software is retired. There are times when the process is dormant, but whenever a change is initiated, the process starts at the appropriate entry point (e.g., product enhancement). The spiral model is a realistic approach to the development of large-scale systems and software.

Because software evolves as the process progresses, the developer and customer better understand and react to risks at each evolutionary level. The spiral model uses prototyping as a risk reduction mechanism but, more important, enables the developer to apply the prototyping approach at any stage in the evolution of the product. It maintains the systematic stepwise approach suggested by the classic life cycle but incorporates it into an iterative framework that more realistically reflects the real world. The spiral model demands a direct consideration of technical risks at all stages of the project and, if properly applied, should reduce risks before they become problematic.

* 1. **PURPOSE OF THE PROJECT**

The new system is totally computerized system.

• A new system provides features like time efficiency to show gym details, user profiles and whatever the customer will give the feedback to the admin.

• This system provides tourism and travelling facilities.

• An inquiry is easily done by user in the system.

• It is the most software application for managing online gym rental business.

* 1. **PROBLEM IN EXISTING SYSTEM**

An existing system can provide manually paper work.

• The user has to go in the office where user can get the gym on rent and book their gym.

• In the existing system you can not provide feedback of the user to the admin online.

* 1. **SOLUTION OF THESE PROBLEMS**

A gym rental is a vehicle that can be used temporarily for a fee during a specified period.Getting a rental gym helps people get around despite the fact they do not have access to their own personal vehicle or don't own a vehicle at all. The individual who needs a gym must contact a rental gym company and contract out for a vehicle. This system increases customer retention and simplify vehicle and staff management.

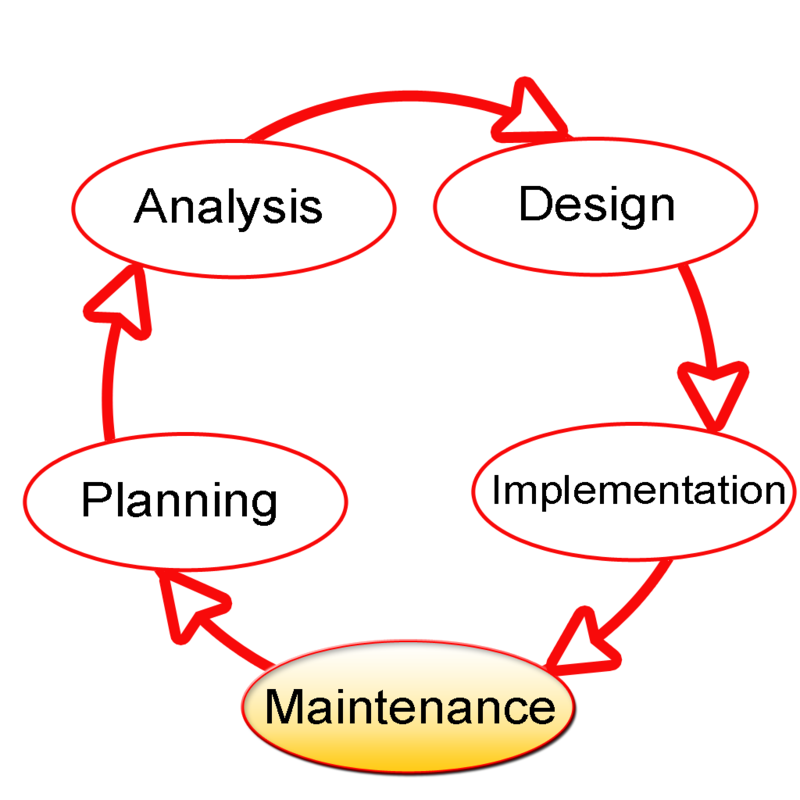
**CHAPTER-2**

**SYSTEM ANALYSIS**

* 1. **PROCESS MODELS USED WITH JUSTIFICATION**

**SDLC MODEL:**

**Software Development Lifecycle (SDLC)**

****

The software development life cycle (SDLC) for small to medium database application development efforts.

This project uses iterative development life cycle, where components of the application are developed through a series of tight iteration. The first iteration focus on very basic functionality, with subsequent iterations adding new functionality to the previous work and or correcting errors identified for the components in production.

The six stages of the SDLC are designed to build on one another, taking outputs from the previous stage, adding additional effort, and producing results that leverage the previous effort and are directly traceable to the previous stages.

During each stage, additional information is gathered or developed, combined with the inputs, and used to produce the stage deliverables. It is important to not that the additional information is restricted in scope, new ideas that would take the project in directions not anticipated by the initial set of high-level requirements or features that are out-of-scope are preserved for later consideration.

Too many software development efforts go away when development team and customer personnel get caught up in the possibilities of automation. Instead of focusing on high priority features, the team can become mired in a sea of nice to have features that are not essential to solve the problem, but in themselves are highly attractive. This is the root cause of large percentage of failed and or abandoned development efforts and is the primary reason the development team utilizes the iterativemodel.

**INPUT DESIGN**

Input design is a part of overall design. The main objective during the input design is as given below:

* To produce a cost-effective method of input
* To achieve the highest possible level of accuracy
* To ensure that the input is acceptable and understood by the user.

**INPUT STAGES:**

The main input stages before the information gets stored in the database media:

Ex: In this project voter either existing or new user data will be stored in database

as the inputs given by users…..

* Data recording, Data transcription, Data conversion, Data verification.
* Data control, Data transmission, Data validation, Data correction.

**OUTPUT DESIGN:**

Output from computer system are required primarily to communicate the results of processing to users. They are also used to provide a permanent copy of the results for later consultation. The various types of outputs in general are:

* External Outputs, whose destination is outside the organization.
* Internal Outputs, whose destination is within organization.
* Users main interface with the computer.
* Operational outputs whose use is purely within the computer department.

The outputs were needed to be generated as a hard copy and as well as queries

To be viewed on the screen. Keeping in view these outputs, the format for the output is taken from the outputs, which are currently being obtained after manual processing.

The standard printed is to be used as output media for hard copies.

**CHAPTER -3**

**Feasibility Report**

**Feasibility Report**

Preliminary investigation examine project feasibility, the likelihood the system will be useful to the organization. The main objective of the feasibility study is to test the Technical, Operational and Economical feasibility for adding new modules and debugging old running system. All system is feasible if they are unlimited resources and infinite time. There are aspects in the feasibility study portion of the preliminary investigation:

* Technical Feasibility
* Operation Feasibility
* Economical Feasibility

**3.1 Technical Feasibility**

The technical issue usually raised during the feasibility stage of the investigation includes the following:

* Does the necessary technology exist to do what is suggested?
* Do the proposed equipments have the technical capacity to hold the data required to use the new system?
* Will the proposed system provide adequate response to inquiries, regardless of the number or location of users?
* Can the system be upgraded if developed?
* Are there technical guarantees of accuracy, reliability, ease of access and data security?

Earlier no system existed to cater to the needs of ‘Secure Infrastructure Implementation System’. The current system developed is technically feasible. It is a web based user interface for audit workflow at NIC-CSD. Thus it provides an easy access to the users. The database’s purpose is to create, establish and maintain a workflow among various entities in order to facilitate all concerned users in their various capacities or roles. Permission to the users would be granted based on the roles specified.

**3.2. Operational Feasibility**

Proposed projects are beneficial only if they can be turned out into information system. That will meet the organization’s operating requirements. Operational feasibility aspects of the project are to be taken as an important part of the project implementation. Some of the important issues raised are to test the operational feasibility of a project includes the following: -

* Is there sufficient support for the management from the users?
* Will the system be used and work properly if it is being developed and implemented?
* Will there be any resistance from the user that will undermine the possible application benefits?

This system is targeted to be in accordance with the above-mentioned issues. Beforehand, the management issues and user requirements have been taken into consideration. So there is no question of resistance from the users that can undermine the possible application benefits.

**3.3. Economic Feasibility**

A system can be developed technically and that will be used if installed must still be a good investment for the organization. In the economical feasibility, the development cost in creating the system is evaluated against the ultimate benefit derived from the new systems. Financial benefits must equal or exceed the costs.

**CHAPTER -4**

**SOFTWARE REQUIRMENT SPECIFICATION**

The software, Site Explorer is designed for management of web sites from a remote location.

**INTRODUCTION:**

**Purpose:** The main purpose for preparing this document is to give a general insight into the analysis and requirement of the existing system or situation and for determining the operating characteristics of the system.

**Scope:** This document plays a vital role in the development life cycle (SDLC) and it describes the complete requirement of the system. It is meant for use by the developers and will be the basic during testing phase. Any changes made to the requirement in the future will have to go through formal change approval process.

**DEVELOPERS RESPONSIBILITIES OVERVIEW:**

The developer is responsible for:

* Developing the system, which meets the SRS and solving all the requirement of the system?
* Demonstrating the system and installing the system at clients’ location after the acceptance testing is successful.
* Submitting the required user manual describing the system interface to work on it and

also the document of the system.

* Conducting any user training that might be needed for using the system.
* Maintaining the system for a period of one year after installation.

**4.1 FUNCTIONAL REQUIREMENTS** **AND NON-FUNCTIONAL REQUIREMENTS**

**Functional Requirements**  
Requirement analysis is a software engineering technique that is composed of the various tasks that determine the needs or conditions that are to be met for a new or altered product, taking into consideration the possible conflicting requirements of the various users.

Functional requirements are those requirements that are used to illustrate the internal working nature of the system, the description of the system, and explanation of each subsystem. It consists of what task the system should perform, the processes involved,which data should the system holds and the interfaces with the user.

The functionalrequirements identified are:

a. Customer’s registration: The system should allow new users to register online and generate membership gymd.  
b. Online reservation of gyms: Customers should be able to use the system to make booking and online reservation.  
c. Automatic update to database once reservation is made or new customer registered:Whenever there’s new reservation or new registration, the system should be able update the database without any additional efforts from the admin.  
d. Feedbacks to customers: It should provide means for customers to leave feedback.

**Non-Functional Requirements**  
It describes aspects of the system that are concerned with how the system provides the functional requirements. They are:

a. Security: The subsystem should provide a high level of security and integrity of thedata held by the system, only authorized personnel of the company can gain access tothe company’s secured page on the system; and only users with valid password and username can login to view user’s page.

b. Performance and Response time: The system should have high performance rate when executing user’s input and should be able to provide feedback or response within a short time span usually 50 seconds for highly complicated task and 20 to 25  
seconds for less complicated task.

c. Error handling: Error should be considerably minimized and an appropriate error message that guides the user to recover from an error should be provided. Validation of user’s input is highly essential. Also the standard time taken to recover from an error should be 15 to 20 seconds.

d. Availability: This system should always be available for access at 24 hours, 7 days a week. Also in the occurrence of any major system malfunctioning, the system should be available in 1 to 2 working days, so that the business process is not severely affected.  
e. Ease of use: Considered the level of knowledge possessed by the users of this system,a simple but quality user interface should be developed to make it easy to understand and required less training.

**4.2 PERFORMANCE REQUIREMENTS**

Performance is measured in terms of the output provided by the application.

Requirements specification plays an important part in the analysis of system. Only when the requirement specifications are properly given, it is possible to design a system, which will fit into required environment. It rests largely in the part of the user of existing system to give the requirements specification because they are the people who finally use the system. This is because the requirements have to be known during the initial stages so that the system can be designed according to those requirements. It is very difficult to change the system once it has been designed and on the other hand designing a system, which does not cater to the requirements of the user, is of no use.

The requirement specification for any system can be broadly stated as given below:

* The system should be able to interface with the existing system
* The system should be accurate
* The system should be better than existing system

The existing system is completely dependent on the user to perform all the duties.

**4.3 HARDWARE REQUIREMENTS**

* P4 2.8GB processor and above.
* Ram 512 MB and above.
* HDD 20 GB Hard Disk and above.

**4.4 SOFTWARE REQUIREMENTS**

* Java
* HTML
* AJAX Tool kit.
* Servlet
* Oracle 10xe

**4.4.1 HTML**

**HyperText Markup Language (HTML)** is the standard [markup language](https://en.wikipedia.org/wiki/Markup_language) for creating [web pages](https://en.wikipedia.org/wiki/Web_page) and [web applications](https://en.wikipedia.org/wiki/Web_application). With [Cascading Style Sheets](https://en.wikipedia.org/wiki/Cascading_Style_Sheets) (CSS) and [JavaScript](https://en.wikipedia.org/wiki/JavaScript) it forms a triad of cornerstone technologies for the [World Wide Web](https://en.wikipedia.org/wiki/World_Wide_Web). [Web browsers](https://en.wikipedia.org/wiki/Web_browser) receive HTML documents from a [webserver](https://en.wikipedia.org/wiki/Webserver) or from local storage and render them into multimedia web pages. HTML describes the structure of a web page [semantically](https://en.wikipedia.org/wiki/Semantic) and originally included cues for the appearance of the document.

[HTML elements](https://en.wikipedia.org/wiki/HTML_element) are the building blocks of HTML pages. With HTML constructs, [images](https://en.wikipedia.org/wiki/Img_(HTML_element)) and other objects, such as [interactive forms,](https://en.wikipedia.org/wiki/Fieldset) may be embedded into the rendered page. It provides a means to create [structured documents](https://en.wikipedia.org/wiki/Structured_document) by denoting structural [semantics](https://en.wikipedia.org/wiki/Semantics) for text such as headings, paragraphs, lists, [links](https://en.wikipedia.org/wiki/Hyperlink), quotes and other items. HTML elements are delineated by *tags*, written using [angle brackets](https://en.wikipedia.org/wiki/Bracket#Angle_brackets). Tags such as <img /> and <input /> introduce content into the page directly. Others such as <p>...</p> surround and provide information about document text and may include other tags as sub-elements. Browsers do not display the HTML tags, but use them to interpret the content of the page.

HTML can embed programs written in a [scripting language](https://en.wikipedia.org/wiki/Scripting_language) such as [JavaScript](https://en.wikipedia.org/wiki/JavaScript) which affect the behavior and content of web pages. Inclusion of CSS defines the look and layout of content. The [World Wide Web Consortium](https://en.wikipedia.org/wiki/World_Wide_Web_Consortium) (W3C), maintainer of both the HTML and the CSS standards, has encouraged the use of CSS over explicit presentational HTML since 1997.

**4.4.2 XHTML**

**Extensible Hypertext Markup Language (XHTML)** is part of the family of [XML](https://en.wikipedia.org/wiki/XML) [markup languages](https://en.wikipedia.org/wiki/Markup_language). It mirrors or extends versions of the widely used [Hypertext Markup Language](https://en.wikipedia.org/wiki/Hypertext_Markup_Language) (HTML), the language in which [Web pages](https://en.wikipedia.org/wiki/Web_page) are formulated.

While HTML, prior to [HTML5](https://en.wikipedia.org/wiki/HTML5), was defined as an application of [Standard Generalized Markup Language](https://en.wikipedia.org/wiki/Standard_Generalized_Markup_Language) (SGML), a flexible markup language framework, XHTML is an application of [XML](https://en.wikipedia.org/wiki/XML), a more restrictive subset of SGML. XHTML documents are [well-formed](https://en.wikipedia.org/wiki/XML#Well-formedness_and_error-handling) and may therefore be parsed using standard XML parsers, unlike HTML, which requires a lenient HTML-specific parser.

XHTML 1.0 became a [World Wide Web Consortium](https://en.wikipedia.org/wiki/World_Wide_Web_Consortium) (W3C) [Recommendation](https://en.wikipedia.org/wiki/W3C_recommendation) on January 26, 2000. XHTML 1.1 became a W3C Recommendation on May 31, 2001. The standard known as XHTML5 is being developed as an XML adaptation of the HTML5 specification.

**4.4.3 SCRIPTING LANGUAGES**

**4.4.3.1 Cascading Style Sheets (CSS)** is a [style sheet language](https://en.wikipedia.org/wiki/Style_sheet_language) used for describing the [presentation](https://en.wikipedia.org/wiki/Presentation_semantics) of a document written in a [markup language](https://en.wikipedia.org/wiki/Markup_language). Although most often used to set the visual style of [web pages](https://en.wikipedia.org/wiki/Web_page) and user interfaces written in [HTML](https://en.wikipedia.org/wiki/HTML) and [XHTML](https://en.wikipedia.org/wiki/XHTML), the language can be applied to any [XML](https://en.wikipedia.org/wiki/XML) document, including [plain XML](https://en.wikipedia.org/wiki/Plain_Old_XML), [SVG](https://en.wikipedia.org/wiki/Scalable_Vector_Graphics) and [XUL](https://en.wikipedia.org/wiki/XUL), and is applicable to rendering in [speech](https://en.wikipedia.org/wiki/Speech_synthesis), or on other media. Along with HTML and [JavaScript](https://en.wikipedia.org/wiki/JavaScript), CSS is a cornerstone technology used by most websites to create visually engaging WebPages, user interfaces for [web applications](https://en.wikipedia.org/wiki/Web_applications), and user interfaces for many mobile applications.

CSS is designed primarily to enable [the separation of document content from document presentation](https://en.wikipedia.org/wiki/Separation_of_presentation_and_content), including aspects such as the [layout](https://en.wikipedia.org/wiki/Page_layout), [colors](https://en.wikipedia.org/wiki/Color), and [fonts](https://en.wikipedia.org/wiki/Typeface). This separation can improve content [accessibility](https://en.wikipedia.org/wiki/Accessibility), provide more flexibility and control in the specification of presentation characteristics, enable multiple HTML pages to share formatting by specifying the relevant CSS in a separate .css file, and reduce complexity and repetition in the structural content.

Separation of formatting and content makes it possible to present the same markup page in different styles for different rendering methods, such as on-screen, in print, by voice (via speech-based browser or [screen reader](https://en.wikipedia.org/wiki/Screen_reader)), and on [Braille-based](https://en.wikipedia.org/wiki/Braille_display) tactile devices. It can also display the web page differently depending on the screen size or viewing device. Readers can also specify a different style sheet, such as a CSS file stored on their own computer, to override the one the author specified.

Changes to the [graphic design](https://en.wikipedia.org/wiki/Graphic_design) of a document (or hundreds of documents) can be applied quickly and easily, by editing a few lines in the CSS file they use, rather than by changing markup in the documents.

The CSS specification describes a priority scheme to determine which style rules apply if more than one rule matches against a particular element. In this so-called *cascade*, priorities (or *weights*) are calculated and assigned to rules, so that the results are predictable.

The CSS specifications are maintained by the [World Wide Web Consortium](https://en.wikipedia.org/wiki/World_Wide_Web_Consortium) (W3C). Internet media type ([MIME type](https://en.wikipedia.org/wiki/MIME_media_type)) text/css is registered for use with CSS by [RFC 2318](https://tools.ietf.org/html/rfc2318) (March 1998). The W3C operates a free [CSS validation service](https://en.wikipedia.org/wiki/W3C_Markup_Validation_Service#CSS_validation) for CSS documents.

**4.4.3.2 JavaScript**

It is a [high-level](https://en.wikipedia.org/wiki/High-level_programming_language), [dynamic](https://en.wikipedia.org/wiki/Dynamic_programming_language), [untyped](https://en.wikipedia.org/wiki/Untyped_language), and [interpreted](https://en.wikipedia.org/wiki/Interpreted_language) [programming language](https://en.wikipedia.org/wiki/Programming_language). It has been standardized in the [ECMAScript](https://en.wikipedia.org/wiki/ECMAScript) language specification. Alongside [HTML](https://en.wikipedia.org/wiki/HTML) and [CSS](https://en.wikipedia.org/wiki/CSS), JavaScript is one of the three core technologies of [World Wide Web](https://en.wikipedia.org/wiki/World_Wide_Web) [content production](https://en.wikipedia.org/wiki/Content_engineering); the majority of [websites](https://en.wikipedia.org/wiki/Website) employ it, and all modern [Web browsers](https://en.wikipedia.org/wiki/Web_browser) support it without the need for [plug-ins](https://en.wikipedia.org/wiki/Browser_extension). JavaScript is [prototype-based](https://en.wikipedia.org/wiki/Prototype-based_programming) with [first-class functions](https://en.wikipedia.org/wiki/First-class_function), making it a [multi-paradigm](https://en.wikipedia.org/wiki/Multi-paradigm) language, supporting [object-oriented](https://en.wikipedia.org/wiki/Object-oriented_programming), [imperative](https://en.wikipedia.org/wiki/Imperative_programming), and [functional](https://en.wikipedia.org/wiki/Functional_programming) [programming styles](https://en.wikipedia.org/wiki/Programming_paradigm). It has an [API](https://en.wikipedia.org/wiki/Application_programming_interface) for working with text, [arrays](https://en.wikipedia.org/wiki/Array_data_type), dates and [regular expressions](https://en.wikipedia.org/wiki/Regular_expression), but does not include any [I/O](https://en.wikipedia.org/wiki/Input/output), such as networking, storage, or graphics facilities, relying for these upon the host environment in which it is embedded.

Although there are strong outward similarities between JavaScript and Java, including language name, [syntax](https://en.wikipedia.org/wiki/Syntax_(programming_languages)), and respective [standard libraries](https://en.wikipedia.org/wiki/Standard_library), the two are distinct languages and differ greatly in their design. JavaScript was influenced by programming languages such as [Self](https://en.wikipedia.org/wiki/Self_(programming_language)) and [Scheme](https://en.wikipedia.org/wiki/Scheme_(programming_language)).

JavaScript is also used in environments that are not Web-based, such as [PDF](https://en.wikipedia.org/wiki/Portable_Document_Format) documents, [site-specific browsers](https://en.wikipedia.org/wiki/Site-specific_browser), and [desktop widgets](https://en.wikipedia.org/wiki/Desktop_widget). Newer and faster JavaScript [virtual machines](https://en.wikipedia.org/wiki/Virtual_machine) (VMs) and platforms built upon them have also increased the popularity of JavaScript for [server-side](https://en.wikipedia.org/wiki/Server-side) [Web applications](https://en.wikipedia.org/wiki/Web_application). On the [client side](https://en.wikipedia.org/wiki/Client_side), developers have traditionally implemented JavaScript as an [interpreted](https://en.wikipedia.org/wiki/Interpreter_(computing)) language, but more recent browsers perform [just-in-time compilation](https://en.wikipedia.org/wiki/Just-in-time_compilation). Programmers also use JavaScript in [video-game development](https://en.wikipedia.org/wiki/Video_game_development), in crafting desktop and mobile applications, and in server-side [network programming](https://en.wikipedia.org/wiki/Computer_network_programming) with run-time environments such as [Node.js](https://en.wikipedia.org/wiki/Node.js).

**4.4.3.3 DYNAMIC JAVA**

The **Java Dynamic Management Kit** or **JDMK** is a [Java](https://en.wikipedia.org/wiki/Java_platform) technology that provides a Java [API](https://en.wikipedia.org/wiki/Application_Programming_Interface) and a collection of software tools for developing and designing [JMX](https://en.wikipedia.org/wiki/Java_Management_Extensions) based [applications](https://en.wikipedia.org/wiki/Application_software). These applications are typically called *smart agents* as they generally provide an abstraction above the communication layer and may also provide a [GUI](https://en.wikipedia.org/wiki/GUI) using [Swing](https://en.wikipedia.org/wiki/Swing_(Java)) or [SWT](https://en.wikipedia.org/wiki/Standard_Widget_Toolkit).

The Java Dynamic Management Kit (Java DMK) provides a set of Java classes and tools that allow easy development of secure monitoring and management solutions based on the Java Management Extensions (JMX) specifications and on the [SNMP](https://en.wikipedia.org/wiki/Simple_Network_Management_Protocol) standards.

Java DMK is a toolkit that allows you to build monitoring and management solutions that answer the main challenge faced by networked environments today, namely dynamic maintenance. Such solutions are based on "smart agents" and result in faster problem resolution, reduced network traffic, less human intervention, and a means of managing complete systems or networks in an integrated way.

Java DMK 5.1 is the first commercial implementation of the latest versions of the JMX standards, JMX 1.2 and JMX Remote API 1.0, and is an all-in-one offering for building secure, interoperable monitoring and management solutions on the J2SE 1.4.2 platform

**4.4.4 WEB LOGIC SERVER**

Today's business environment demands Web and e-commerce applications that accelerate your entry into new markets, help you find new ways to reach and retain customers, and allow you to introduce new products and services quickly. To build and deploy these new solutions, you need a proven, reliable e-commerce platform that can connect and empower all types of users while integrating your corporate data, mainframe applications, and other enterprise applications in a powerful, flexible, end-to-end e-commerce solution. Your solution must provide the performance, scalability, and high availability needed to handle your most critical enterprise-scale computing.As the industry-leading e-commerce transaction platform, WebLogic Server allows you to quickly develop and deploy reliable, secure, scalable and manageable applications. It manages system-level details so you can concentrate on business logic and presentation.

**J2EE Platform**

Web Logic Server contains Java 2 Platform, Enterprise Edition (J2EE) technologies. J2EE is the standard platform for developing multitier enterprise applications based on the Java programming language. The technologies that make up J2EE were developed collaboratively by Sun Microsystems and other software vendors, including BEA Systems.

J2EE applications are based on standardized, modular components. Web Logic Server provides a complete set of services for those components and handles many details of application behavior automatically, without requiring programming.

**XML Implementation**

Web Logic Server consolidates Extensible Markup Language (XML) technologies applicable to Web Logic Server and XML applications based on Web Logic Server. A simplified version of the Standard Generalized Markup Language (SGML) markup language, XML describes the content and structure of data in a document and is an industry standard for delivering content on the Internet. Typically, XML is used as the data exchange format between J2EE applications and client applications, or between components of a J2EE application. The Web Logic Server XML subsystem supports the use of standard parsers, the Web Logic FastParser, XSLT transformers, and DTDs and XML schemas to process and convert XML files.

**CHAPTER -5**

**SYSTEM DESIGN**

**5.1. Module design:**

Software design sits at the technical kernel of the software engineering process and is applied regardless of the development paradigm and area of application. Design is the first step in the development phase for any engineered product or system. The designer’s goal is to produce a model or representation of an entity that will later be built. Beginning, once system requirement have been specified and analyzed, system design is the first of the three technical activities -design, code and test that is required to build and verify software.

The importance can be stated with a single word “Quality”. Design is the place where quality is fostered in software development. Design provides us with representations of software that can assess for quality. Design is the only way that we can accurately translate a customer’s view into a finished software product or system. Software design serves as a foundation for all the software engineering steps that follow. Without a strong design we risk building an unstable system – one that will be difficult to test, one whose quality cannot be assessed until the last stage.

During design, progressive refinement of data structure, program structure, and procedural details are developed reviewed and documented. System design can be viewed from either technical or project management perspective. From the technical point of view, design is comprised of four activities – architectural design, data structure design, interface design and procedural design.

**5.2 NORMALIZATION**

It is a process of converting relation to standard form. The process is used to handle the problems that can arise due to data redundancy i.e. repetition of data in database, maintain data integrity as well as handling problems that can arise due to insertion, updation, deletion anomalies.

Decomposing is the process of splitting relations into multiple relations to eliminate

anomalies and maintain anomalies and maintain data integrity. To do this we use normal forms or rules for structuring relation.

**Insertion anomaly:** Inability to add data to the database due to absence of other data.

**Deletion anomaly:** Unintended loss of data due to deletion of other data.

**Update anomaly:** Data inconsistency resulting from data redundancy and partial update.

**Normal Forms:** These are the rules for structuring relations that eliminate anomalies.

**FIRST NORMAL FORM:**

A relation is said to be in first normal form if the values in the relation are atomic for every attribute in the relation. By this we mean simply that no attribute value can be a set of values or, as it is sometimes expressed, a repeating group.

**SECOND NORMAL FORM:**

A relation is said to be in second normal form if it is in first normal form and it should satisfy any one of the following rules.

1. Primary key is not a composite primary key.
2. No non key attributes are present
3. Every non key attribute is fully functionally dependent on full set of primary key.

**THIRD NORMAL FORM:**

A relation is said to be in third normal form if there exists no transitive dependencies.

**Transitive Dependency:** If two non key attributes depend on each other as well as on the primary key then they are said to be transitively dependent.

The above normalization principles were applied to decompose the data in multiple tables thereby making the data to be maintained in consistent state.

**5.3 DATA FLOW DIAGRAMS**

A data flow diagram is graphical tool used to describe and analyze movement of data through a system. These are the central tool and the basis from which the other components are developed. The transformation of data from input to output, through processed, may be described logically and independently of physical components associated with the system. These are known as the logical data flow diagrams. The physical data flow diagrams show the actual implements and movement of data between people, departments and workstations. A full description of a system actually consists of a set of data flow diagrams. Using two familiar notations Yourdon, Gane and Sarson notation develops the data flow diagrams. Each component in a DFD is labeled with a descriptive name. Process is further identified with a number that will be used for identification purpose. The development of DFD’S is done in several levels. Each process in lower level diagrams can be broken down into a more detailed DFD in the next level. The lop-level diagram is often called context diagram. It consists a single process bit, which plays vital role in studying the current system. The process in the context level diagram is exploded into other process at the first level DFD.

The idea behind the explosion of a process into more process is that understanding at one level of detail is exploded into greater detail at the next level. This is done until further explosion is necessary and an adequate amount of detail is described for analyst to understand the process.

Larry Constantine first developed the DFD as a way of expressing system requirements in a graphical from, this lead to the modular design.

A DFD is also known as a “bubble Chart” has the purpose of clarifying system requirements and identifying major transformations that will become programs in system design. So it is the starting point of the design to the lowest level of detail. A DFD consists of a series of bubbles joined by data flows in the system.

**DFD SYMBOLS:**

In the DFD, there are four symbols

1. A square defines a source(originator) or destination of system data
2. An arrow identifies data flow. It is the pipeline through which the information flows
3. A circle or a bubble represents a process that transforms incoming data flow into outgoing data flows.
4. An open rectangle is a data store, data at rest or a temporary repository of data

Process that transforms data flow.

Source or Destination of data

Data flow

Data Store

**CONSTRUCTING A DFD:**

Several rules of thumb are used in drawing DFD’S:

1. Process should be named and numbered for an easy reference. Each name should be representative of the process.
2. The direction of flow is from top to bottom and from left to right. Data traditionally flow from source to the destination although they may flow back to the source. One way to indicate this is to draw long flow line back to a source. An alternative way is to repeat the source symbol as a destination. Since it is used more than once in the DFD it is marked with a short diagonal.
3. When a process is exploded into lower level details, they are numbered.
4. The names of data stores and destinations are written in capital letters. Process and dataflow names have the first letter of each work capitalized

A DFD typically shows the minimum contents of data store. Each data store should contain all the data elements that flow in and out.

**SAILENT FEATURES OF DFD’S**

1. The DFD shows flow of data, not of control loops and decision are controlled considerations do not appear on a DFD.
2. The DFD does not indicate the time factor involved in any process whether the dataflow take place daily, weekly, monthly or yearly.
3. The sequence of events is not brought out on the DFD.

**TYPES OF DATA FLOW DIAGRAMS**

1. Current Physical
2. Current Logical
3. New Logical
4. New Physical

**CURRENT PHYSICAL:**

In Current Physical DFD process label include the name of people or their positions or the names of computer systems that might provide some of the overall system-processing label includes an identification of the technology used to process the data. Similarly data flows and data stores are often labels with the names of the actual physical media on which data are stored such as file folders, computer files, business forms or computer tapes.

**CURRENT LOGICAL:**

The physical aspects at the system are removed as much as possible so that the current system is reduced to its essence to the data and the processors that transform them regardless of actual physical form.

**NEW LOGICAL**:

This is exactly like a current logical model if the user were completely happy with the user were completely happy with the functionality of the current system but had problems with how it was implemented typically through the new logical model will differ from current logical model while having additional functions, absolute function removal and inefficient flows recognized.

**NEW PHYSICAL:**

The new physical represents only the physical implementation of the new system.

**RULES GOVERNING THE DFD’S**

**PROCESS:**

1. No process can have only outputs.
2. No process can have only inputs. If an object has only inputs than it must be a sink.
3. A process has a verb phrase label.

**DATA STORE**

1. Data cannot move directly from one data store to another data store, a process must move data.
2. Data cannot move directly from an outside source to a data store, a process, which receives, must move data from the source and place the data into data store
3. A data store has a noun phrase label.

**SOURCE OR SINK**

The origin and or destination of data.

1. Data cannot move direly from a source to sink it must be moved by a process
2. A source and /or sink has a noun phrase land

**DATA FLOW**

1. A Data Flow has only one direction of flow between symbols. It may flow in both directions between a process and a data store to show a read before an update. The later is usually indicated however by two separate arrows since these happen at different type.
2. A join in DFD means that exactly the same data comes from any of two or more different processes data store or sink to a common location.
3. A data flow cannot go directly back to the same process it leads. There must be at least one other process that handles the data flow produce some other data flow returns the original data into the beginning process.
4. A Data flow to a data store means update (delete or change).
5. A data Flow from a data store means retrieve or use.

A data flow has a noun phrase label more than one data flow noun phrase can appear on a single arrow as long as all of the flows on the same arrow move together as one package.

* **DATA FLOW DIAGRAMS OF PROJECT**

**MODULE-1:**

**ONLINE GYM**

**MODULE-2:**

**BOOKING FORM(HOME)**

**MODULE-2(CONTINUE):**

**SUBMIT**

**MODULE-3:**

CONTACT FORM

**SUBMIT**

**MODULE-4:**

FEEDBACK FORM

**SUBMIT**

**MODULE-5:**

**ADMIN FORM**

**SUBMIT**

**NOT VALID**

**VALID**

**FEEDBACK REPORT**

**REPORT OF CUSTOMER**

**FEEDBACK**

**REPORT OF CUSTOMER**

**CONTACT**

**REPORT OF CUSTOMER**

**BOOKED**

**LOGOUT**

**Chapter -6**

**Implementation of Project**

**Description of Technology Used in Project.**

# 6.1. INTRODUCTION TO JAVA

The JAVA Framework is a new computing platform that simplifies application development in the highly distributed environment of the Internet. The JAVA Framework is designed to fulfill the following objectives:

* To provide a consistent object-oriented programming environment whether object code is stored and executed locally, executed locally but Internet-distributed, or executed remotely.
* To provide a code-execution environment that minimizes software deployment and versioning conflicts.
* To provide a code-execution environment that guarantees safe execution of code, including code created by an unknown or semi-trusted third party.
* To provide a code-execution environment that eliminates the performance problems of scripted or interpreted environments.
* To make the developer experience consistent across widely varying types of applications, such as Windows-based applications and Web-based applications.
* To build all communication on industry standards to ensure that code based on the JAVA Framework can integrate with any other code.

The JAVA Framework can be hosted by unmanaged components that load the common language runtime into their processes and initiate the execution of managed code, thereby creating a software environment that can exploit both managed and unmanaged features. The JAVA Framework not only provides several runtime hosts, but also supports the development of third-party runtime hosts..

## FEATURES OF THE COMMON LANGUAGE RUNTIME

The common language runtime manages memory, thread execution, code execution, code safety verification, compilation, and other system services. These features are intrinsic to the managed code that runs on the common language runtime.

With regards to security, managed components are awarded varying degrees of trust, depending on a number of factors that include their origin (such as the Internet, enterprise network, or local computer). This means that a managed component might or might not be able to perform file-access operations, registry-access operations, or other sensitive functions, even if it is being used in the same active application.

The runtime enforces code access security. For example, users can trust that an executable embedded in a Web page can play an animation on screen or sing a song, but cannot access their personal data, file system, or network. The security features of the runtime thus enable legitimate Internet-deployed software to be exceptionally featuring rich.

The runtime also enforces code robustness by implementing a strict type- and code-verification infrastructure called the common type system (CTS). The CTS ensures that all managed code is self-describing. The various Microsoft and third-party language compilers

Generate managed code that conforms to the CTS. This means that managed code can consume other managed types and instances, while strictly enforcing type fidelity and type safety.

In addition, the managed environment of the runtime eliminates many common software issues. For example, the runtime automatically handles object layout and manages references to objects, releasing them when they are no longer being used. This automatic memory management resolves the two most common application errors, memory leaks and invalid memory references.

The runtime also accelerates developer productivity. For example, programmers can write applications in their development language of choice, yet take full advantage of the runtime, the class library, and components written in other languages by other developers.

## JAVA FRAMEWORK CLASS LIBRARY

The JAVA Framework class library is a collection of reusable types that tightly integrate with the common language runtime. The class library is object oriented, providing types from which your own managed code can derive functionality. This not only makes the JAVA Framework types easy to use, but also reduces the time associated with learning new features of the JAVA Framework. In addition, third-party components can integrate seamlessly with classes in the JAVA Framework.

For example, the JAVA Framework collection classes implement a set of interfaces that you can use to develop your own collection classes. Your collection classes will blend seamlessly with the classes in the JAVA Framework.

As you would expect from an object-oriented class library, the JAVA Framework types enable you to accomplish a range of common programming tasks, including tasks such as string management, data collection, database connectivity, and file access. In addition to these common tasks, the class library includes types that support a variety of specialized development scenarios. For example, you can use the JAVA Framework to develop the mentioned types of applications and services:

**HISTORY OF JAVA**

Java language was developed by James Gosling and his team at sun micro systems and released formally in 1995. Its former name is oak. Java Development Kit 1.0 was released in 1996. To popularize java and is freely available on Internet.

**OVERVIEW OF JAVA**

Java is loosely based on C++ syntax, and is meant to be Object-Oriented Structure of java is midway between an interpreted and a compiled language. The Java compiler compiles Java programs into Byte Codes that are secure and portable across different platforms. These byte codes are essentially instructions encapsulated in single type, to what is known as a java virtual machine (JVM), which resides in standard browser.

Jvm verifies these byte codes when downloaded by the browser for integrity. Jvms available for almost all OS. JVM converts these byte codes into machine specific instructions at runtime.

**FEATURES OF JAVA**

* Java is object-oriented language and supports encapsulation, inheritance, polymorphism and dynamic binding, but does not support multiple inheritances. Everything in java is an object except some primitive data types.
* Java is portable architecture neutral that is java programs once compiled can be executed on any machine that is enabled.
* JAVA is distributed in its approach and used for Internet programming.
* Java is robust, secured, high performing and dynamic in nature.
* Java supports multithreading. There for different parts of the program can be executed at the same time

**JAVA AND INTERNET**

Java is strongly associated with Internet and known as Internet programming language. Internet users can use java to create applet programs and run them locally using java enabled browser search as hot java. Applets can be downloaded from remote machine via Internet and run it on local machine.

**JAVA AND WORLD WIDE WEB**

World Wide Web is an open-ended information retrieval system designed to be used in the distributed environment. This system contains web pages that provide both information and controls. We can navigate to a new web page in any direction. This is made possible worth HTML java was meant to be used in distributed environment such as Internet. So java could be easily incorporated into the web system and is capable of supporting animation graphics, games and other special effect. The web has become more dynamic and interactive with support of java. We can run a java program on remote machine over Internet with the support of web.

**JAVA ENVIRONMENT**

Java environment includes a large no. Of tools, which is part of the system known as java development kit (JDK) and hundreds of classes, methods, and interfaces grouped into packages forms part of java standard library (JSL).

**JAVA ARCHITECTURE**

Java architecture provides a portable, robust, high performing environment for development. Java provides portability by compiling the byte codes for the java virtual machine, which are then interpreted on each platform by the runtime environment. Java also provides stringent compile and runtime checking and automatic memory management in order to ensure solid code.

**JAVA VIRTUAL MACHINE**

When we compile the code, java compiler creates machine code (byte code) for a hypothetical machine called java virtual machine (jvm). The jvm will execute the byte code and overcomes the issue of portability. The code is written and compile for one machine and interpreted all other machines. This machine is called java virtual machine.

**PARADIGM OF JAVA**

* Dynamic down loading applets (small application programs);
* Elimination of flatware phenomenon that is providing those features of a product that user needs at a time. The remaining features of a product can remain in the server.
* Changing economic model of the software
* Up-to-date software availability
* Supports network entire computing
* Supports CORBA & DCOM

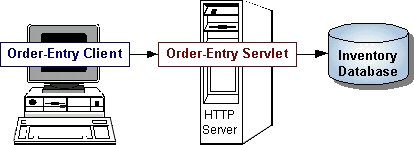
# INTRODUCTION TO SERVLETS

Servlets provide a Java(TM)-based solution used to address the problems currently associated with doing server-side programming, including inextensible scripting solutions, platform-specific APIs, and incomplete interfaces.

Servlets are objects that conform to a specific interface that can be plugged into a Java-based server. Servlets are to the server-side what applets are to the client-side -- object byte codes that can be dynamically loaded off the net. They differ from applets in that they are faceless objects (without graphics or a GUI component). They serve as platform-independent, dynamically loadable, pluggable helper byte code objects on the server side that can be used to dynamically extend server-side functionality.

## WHAT IS A SERVLET?

Servlets are modules that extend request/response-oriented servers, such as Java-enabled web servers. For example, a servlet might be responsible for taking data in an HTML order-entry form and applying the business logic used to update a company's order database.



Servlets are to servers what applets are to browsers. Unlike applets, however, servlets have no graphical user interface.

Servlets can be embedded in many different servers because the servlet API, which you use to write servlets, assumes nothing about the server's environment or protocol. Servlets have become most widely used within HTTP servers; many web servers support the Servlet API.

## USE SERVLETS INSTEAD OF CGI SCRIPTS!

Servlets are an effective replacement for CGI scripts. They provide a way to generate dynamic documents that is both easier to write and faster to run. Servlets also address the problem of doing server-side programming with platform-specific APIs: they are developed with the Java Servlet API, a standard Java extension.

So use servlets to handle HTTP client requests. For example, have servlets process data POSTed over HTTPS using an HTML form, including purchase order or credit gymd data. A servlet like this could be part of an order-entry and processing system, working with product and inventory databases, and perhaps an on-line payment system.

**OTHER USES FOR SERVLETS**

Here are a few more of the many applications for servlets:

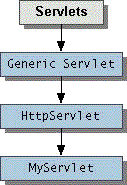
* Allowing collaboration between people. A servlet can handle multiple requests concurrently, and can synchronize requests.
* Forwarding requests. Servlets can forward requests to other servers and servlets. Thus servlets can be used to balance load among several servers that mirror the same content, and to partition a single logical service over several servers, according to task type or organizational boundaries.

## ARCHITECTURE OF THE SERVLET PACKAGE

The javax.servlet package provides interfaces and classes for writing servlets. The architecture of the package is described below.

**THE SERVLET INTERFACE**

The central abstraction in the Servlet API is the Servlet interface. All servlets implement this interface, either directly or, more commonly, by extending a class that implements it such as HttpServlet.



The Servlet interface declares, but does not implement, methods that manage the servlet and its communications with clients. Servlet writers provide some or all of these methods when developing a servlet.

**CLIENT INTERACTION**

When a servlet accepts a call from a client, it receives two objects:

* A ServletRequest, which encapsulates the communication from the client to the server.
* A ServletResponse, which encapsulates the communication from the servlet back to the client.

ServletRequest and ServletResponse are interfaces defined by the javax.servlet package.

#### The “ServletRequest” Interface

The ServletRequest interface allows the servlet access to:

* Information such as the names of the parameters passed in by the client, the protocol (scheme) being used by the client, and the names of the remote host that made the request and the server that received it.
* The input stream, ServletInputStream. Servlets use the input stream to get data from clients that use application protocols such as the HTTP POST and PUT methods.

Interfaces that extend ServletRequest interface allow the servlet to retrieve more protocol-specific data. For example, the HttpServletRequest interface contains methods for accessing HTTP-specific header information.

#### The ServletResponse Interface

The ServletResponse interface gives the servlet methods for replying to the client. It:

* Allows the servlet to set the content length and MIME type of the reply.
* Provides an output stream, ServletOutputStream, and a Writer through which the servlet can send the reply data.

Interfaces that extend the ServletResponse interface give the servlet more protocol-specific capabilities. For example, the HttpServletResponse interface contains methods that allow the servlet to manipulate HTTP-specific header information.

### Additional Capabilities of HTTP Servlets

The classes and interfaces described above make up a basic Servlet. HTTP servlets have some additional objects that provide session-tracking capabilities. The servlet writer can use these APIs to maintain state between the servlet and the client that persists across multiple connections during some time period. HTTP servlets also have objects that provide cookies. The servlet writer uses the cookie API to save data with the client and to retrieve this data.

**SYSTEM TESTING**

**SYSTEM INTEGRATION AND TESTING**

The proposed system is tested parallel with the software effort that consists of its own phases of analysis, implementation, testing and maintenance

**UNIT TESTING**

Unit testing comprises the set of tests performed by an individual programmer prior to integration of the unit into a large system.

Coding and debugging -> Unit testing -> Integration testing

There are four categories of tests should be performed.

* Functional Testing
* Performance Testing
* Stress Testing
* Structure testing

**Function test** cases involve exercising the code with the nominal input values for which the expected results are known, as well as boundary values maximum.

**Performance testing** determines the amount of execution spent in various parts of the unit program throughput, response time and device utilization by the program unit.

**Stress tests** are those tests designing to initially break the unit.

**Structure tests** are con concerned with exercising the internal logic of a program and traversing particular execution path.

Establishing a test completion criterion is another difficulty encountered in the unit testing of real programs. Unit testing includes.

* **Statement Converge**
* **Branch Converge**
* **Logical path Converge**

Using **Statement Converge** programmer attempts to find a set of test cases that will execute each statement in a program at least once.

Using **Branch Converge** as the test completion criterion the programmer attempts to find a set of cases that will execute each branching statement in each direction at least once.

**Logical Path Converge** acknowledges that the order in which the branches are executed during a test is an important factor in determining the test outcome.

**INTEGRATION TESTING**

**Integration testing is of three types:**

* **Bottom up Integration**
* **Top down Integration**
* **Sandwich Integration**

**Bottom up integration** testing consists of unit testing followed by system testing. Unit testing has the goal of testing individual modules in the system. Subsystem testing is concerned with verifying the operation of the interfaces between modules in the sub systems.

**System Testing** is concerned with subtleties in the interfaces, decision logic, and control flow recovery procedure, throughput, capacity and timing characteristics.

Top down integration starts with the main routine and one or two immediately subordinate routines in the system structure. Top down integration requires the use of program stubs to simulate the effect of lower level routines that are called by those being tested.

Top down method has the fallowing advantages:

* System integration is distributed through the implementation phase. Modules are integrated as they are developed.
* Top-level interfaces are tested first and mist often.
* The top-level routine provides a natural test harness for lower level routines.
* Errors are localized to the new modules and interfaces that are being added.

**Sandwich integration** is predominately top down, but bottom up techniques are used on some modules and sub system. This mix alleviates many of the problems encountered in pure top down and retains the advantages of the top down integration at the subsystem and system level.

For example, the Windows Forms classes are a comprehensive set of reusable types that vastly simplify Windows GUI development. If you write an JAVA Web Form application, you can use the Web Forms classes.

## CLIENT APPLICATION DEVELOPMENT

Client applications are the closest to a traditional style of application in Windows-based programming. These are the types of applications that display windows or forms on the desktop, enabling a user to perform a task. Client applications include applications such as word processors and spreadsheets, as well as custom business applications such as data-entry tools, reporting tools, and so on. Client applications usually employ windows, menus, buttons, and other GUI elements, and they likely access local resources such as the file system and peripherals such as printers.

Another kind of client application is the traditional ActiveX control (now replaced by the managed Windows Forms control) deployed over the Internet as a Web page. This application is much like other client applications: it is executed natively, has access to local resources, and includes graphical elements.

In the past, developers created such applications using C/C++ in conjunction with the Microsoft Foundation Classes (MFC) or with a rapid application development (RAD) environment such as Microsoft® Visual Basic®. The JAVA Framework incorporates aspects of these existing products into a single, consistent development environment that drastically simplifies the development of client applications.

The Windows Forms classes contained in the JAVA Framework are designed to be used for GUI development. You can easily create command windows, buttons, menus, toolbars, and other screen elements with the flexibility necessary to accommodate shifting business needs.

For example, the JAVA Framework provides simple properties to adjust visual attributes associated with forms. In some cases the underlying operating system does not support changing these attributes directly, and in these cases the JAVA Framework automatically recreates the forms. This is one of many ways in which the JAVA Framework integrates the developer interface, making coding simpler and more consistent.

## Server Application Development

Server-side applications in the managed world are implemented through runtime hosts. Unmanaged applications host the common language runtime, which allows your custom managed code to control the behavior of the server. This model provides you with all the features of the common language runtime and class library while gaining the performance and scalability of the host server.

The following illustration shows a basic network schema with managed code running in different server environments. Servers such as IIS and Oracle can perform standard operations while your application logic executes through the managed code.

**SERVER-SIDE MANAGED CODE**

JAVA is the hosting environment that enables developers to use the JAVA Framework to target Web-based applications. However, JAVA is more than just a runtime host; it is a complete architecture for developing Web sites and Internet-distributed objects using managed code. Both Web Forms and XML Web services use IIS and JAVA as the publishing mechanism for applications, and both have a collection of supporting classes in the JAVA Framework.

XML Web services, an important evolution in Web-based technology, are distributed, server-side application components similar to common Web sites. However, unlike Web-based applications, XML Web services components have no UI and are not targeted for browsers such as Internet Explorer and Netscape Navigator. Instead, XML Web services consist of reusable software components designed to be consumed by other applications, such as traditional client applications, Web-based applications, or even other XML Web services. As a result, XML Web services technology is rapidly moving application development and deployment into the highly distributed environment of the Internet.

**ACTIVE SERVER PAGES**

JAVA is a programming framework built on the common language runtime that can be used on a server to build powerful Web applications. JAVA offers several important advantages over previous Web development models:

* **Enhanced Performance.** JAVA is compiled common language runtime code running on the server. Unlike its interpreted predecessors, JAVA can take advantage of early binding, just-in-time compilation, native optimization, and caching services right out of the box. This amounts to dramatically better performance before you ever write a line of code.
* **World-Class Tool Support.** The JAVA framework is complemented by a rich toolbox and designer in the Visual Studio integrated development environment. WYSIWYG editing, drag-and-drop server controls, and automatic deployment are just a few of the features this powerful tool provides.
* **Power and Flexibility.** Because JAVA is based on the common language runtime, the power and flexibility of that entire platform is available to Web application developers. The JAVA Framework class library, Messaging, and Data Access solutions are all seamlessly accessible from the Web. JAVA is also language-independent, so you can choose the language that best applies to your application or partition your application across many languages. Further, common language runtime interoperability guarantees that your existing investment in COM-based development is preserved when migrating to JAVA.
* **Simplicity.** JAVA makes it easy to perform common tasks, from simple form submission and client authentication to deployment and site configuration. For example, the JAVA page framework allows you to build user interfaces that cleanly separate application logic from presentation code and to handle events in a simple, Visual Basic - like forms processing model. Additionally, the common language runtime simplifies development, with managed code services such as automatic reference counting and garbage collection.
* **Manageability.** JAVA employs a text-based, hierarchical configuration system, which simplifies applying settings to your server environment and Web applications. Because configuration information is stored as plain text, new settings may be applied without the aid of local administration tools. This "zero local administration" philosophy extends to deploying JAVA Framework applications as well. An JAVA Framework application is deployed to a server simply by copying the necessary files to the server. No server restart is required, even to deploy or replace running compiled code.
* **Scalability and Availability.** JAVA has been designed with scalability in mind, with features specifically tailored to improve performance in clustered and multiprocessor environments. Further, processes are closely monitored and managed by the JAVA runtime, so that if one misbehaves (leaks, deadlocks), a new process can be created in its place, which helps keep your application constantly available to handle requests.
* **Customizability and Extensibility.** JAVA delivers a well-factored architecture that allows developers to "plug-in" their code at the appropriate level. In fact, it is possible to extend or replace any subcomponent of the JAVA runtime with your own custom-written component. Implementing custom authentication or state services has never been easier.
* **Security.** With built in Windows authentication and per-application configuration, you can be assured that your applications are secure.

#### LANGUAGE SUPPORT

The Microsoft JAVA Platform currently offers built-in support for three languages: C#, Visual Basic, and JScript.

**WHAT IS JAVA WEB FORMS?**

The JAVA Web Forms page framework is a scalable common language runtime programming model that can be used on the server to dynamically generate Web pages.

**CODE-BEHIND WEB FORMS**

JAVA supports two methods of authoring dynamic pages. The first is the method shown in the preceding samples, where the page code is physically declared within the originating .aspx file. An alternative approach--known as the code-behind method--enables the page code to be more cleanly separated from the HTML content into an entirely separate file.

**INTRODUCTION TO JAVA SERVER CONTROLS**

In addition to (or instead of) using <% %> code blocks to program dynamic content, JAVA page developers can use JAVA server controls to program Web pages. Server controls are declared within an .aspx file using custom tags or intrinsic HTML tags that contain a **runat="server"** attributes value. Intrinsic HTML tags are handled by one of the controls in the **System.Web.UI.HtmlControls** namespace. Any tag that doesn't explicitly map to one of the controls is assigned the type of **System.Web.UI.HtmlControls.HtmlGenericControl**.

Server controls automatically maintain any client-entered values between round trips to the server. This control state is not stored on the server (it is instead stored within an **<input type="hidden">** form field that is round-tripped between requests). Note also that no client-side script is required.

In addition to supporting standard HTML input controls, JAVA enables developers to utilize richer custom controls on their pages. For example, the following sample demonstrates how the **<asp:adrotator>** control can be used to dynamically display rotating ads on a page.

1. JAVA Web Forms provide an easy and powerful way to build dynamic Web UI.
2. JAVA Web Forms pages can target any browser client (there are no script library or cookie requirements).
3. JAVA Web Forms pages provide syntax compatibility with existing ASP pages.
4. JAVA server controls provide an easy way to encapsulate common functionality.

#### JAVA OVERVIEW

JAVA is an evolution of the data access model that directly addresses user requirements for developing scalable applications. It was designed specifically for the web with scalability, statelessness, and XML in mind.

JAVA uses some objects, such as the **Connection** and **Command** objects, and also introduces new objects. Key new JAVA objects include the **DataSet**, **DataReader**, and **DataAdapter**.

The important distinction between this evolved stage of JAVA and previous data architectures is that there exists an object -- the **DataSet** -- that is separate and distinct from any data stores. Because of that, the **DataSet** functions as a standalone entity. You can think of the DataSet as an always disconnected recordset that knows nothing about the source or destination of the data it contains. Inside a **DataSet**, much like in a database, there are tables, columns, relationships, constraints, views, and so forth.

A **DataAdapter** is the object that connects to the database to fill the **DataSet**. Then, it connects back to the database to update the data there, based on operations performed while the **DataSet** held the data. In the past, data processing has been primarily connection-based. Now, in an effort to make multi-tiered apps more efficient, data processing is turning to a message-based approach that revolves around chunks of information. At the center of this approach is the **DataAdapter**, which provides a bridge to retrieve and save data between a **DataSet** and its source data store. It accomplishes this by means of requests to the appropriate SQL commands made against the data store.

The XML-based **DataSet** object provides a consistent programming model that works with all models of data storage: flat, relational, and hierarchical. It does this by having no 'knowledge' of the source of its data, and by representing the data that it holds as

* **Connections**. For connection to and managing transactions against a database.
* **Commands**. For issuing SQL commands against a database.
* **Data Readers**. For reading a forward-only stream of data records from an Oracle data source.
* **Datasets**. For storing, Removing and programming against flat data, XML data and relational data.
* **Data Adapters**. For pushing data into a **Dataset**, and reconciling data against a database.

When dealing with connections to a database, there are two different options: Oracle JAVA Data Provider (System.Data.SqlClient) and OLE DB JAVA Data Provider (System.Data.OleDb). In these samples we will use the Oracle JAVA Data Provider. These are written to talk directly to Microsoft Oracle. The OLE DB JAVA Data Provider is used to talk to any OLE DB provider (as it uses OLE DB underneath).

**Connections:**

Connections are used to 'talk to' databases, and are represented by provider-specific classes such as **SqlConnection**. Commands travel over connections and resultsets are returned in the form of streams which can be read by a **Data Reader** object, or pushed into a **Dataset** object.

**Commands:**

Commands contain the information that is submitted to a database, and are represented by provider-specific classes such as **SqlCommand**. A command can be a stored procedure call, an UPDATE statement, or a statement that returns results. You can also use input and output parameters, and return values as part of your command syntax. The example below shows how to issue an INSERT statement against the **Northwind** database.

**Data Readers:**

The **Data Reader** object is somewhat synonymous with a read-only/forward-only cursor over data. The **Data Reader** API supports flat as well as hierarchical data. A **Data Reader** object is returned after executing a command against a database. The format of the returned **Data Reader** object is different from a recordset. For example, you might use the **Data Reader** to show the results of a search list in a web page.

**DATA SETS AND DATA ADAPTERS:**

**Datasets**  
 The **Data Set** object is similar to the **Record set** object, but more powerful, and with one other important distinction: the **Data Set** is always disconnected. The **Data Set** object represents a cache of data, with database-like structures such as tables, columns, relationships, and constraints. However, though a **Data Set** can and does behave much like a database, it is important to remember that **Data Set** objects do not interact directly with databases, or other source data. This allows the developer to work with a programming model that is always consistent, regardless of where the source data resides. Data coming from a database, an XML file, from code, or user input can all be placed into **Data Set** objects. Then, as changes are made to the **Data Set** they can be tracked and verified before updating the source data. The **Get Changes** method of the **Data Set** object actually creates a second **Dat Set** that contains only the changes to the data. This **Data Set** is then used by a **DataAdapter** (or other objects) to update the original data source.

The **Data Set** has many XML characteristics, including the ability to produce and consume XML data and XML schemas. XML schemas can be used to describe schemas interchanged via WebServices. In fact, a **Data Set** with a schema can actually be compiled for type safety and statement completion.

**DATA ADAPTERS (OLEDB/SQL)**

The **Data Adapter** object works as a bridge between the **Data Set** and the source data. Using the provider-specific **SqlData Adapter** (along with its associated **SqlCommand** and **SqlConnection**) can increase overall performance when working with a Microsoft Oracle databases. For other OLE DB-supported databases, you would use the **OleDbDataAdapter** object and its associated **OleDbCommand** and **OleDbConnection** objects.

The **Data Adapter** object uses commands to update the data source after changes have been made to the **Data Set**. Using the **Fill** method of the **Data Adapter** calls the SELECT command; using the **Update** method calls the INSERT, UPDATE or DELETE command for each changed row. You can explicitly set these commands in order to control the statements used at runtime to resolve changes, including the use of stored procedures. For ad-hoc scenarios, a **Command Builder** object can generate these at run-time based upon a select statement. However, this run-time generation requires an extra round-trip to the server in order to gather required metadata, so explicitly providing the INSERT, UPDATE, and DELETE commands at design time will result in better run-time performance.

1. JAVA is the next evolution of for the JAVA Framework.
2. JAVA was created with n-Tier, statelessness and XML in the forefront. Two new objects, the **Data Set** and **Data Adapter**, are provided for these scenarios.
3. JAVA can be used to get data from a stream, or to store data in a cache for updates.
4. There is a lot more information about JAVA in the documentation.
5. Remember, you can execute a command directly against the database in order to do inserts, updates, and deletes. You don't need to first put data into a **Data Set** in order to insert, update, or delete it.
6. Also, you can use a **Data Set** to bind to the data, move through the data, and navigate data relationship.
   1. **ORACLE**

A database management, or DBMS, gives the user access to their data and helps them transform the data into information. Such database management systems include dBase, Oracle. These systems allow users to create, update and extract information from their database.

A database is a structured collection of data. Data refers to the characteristics of people, things and events. Oracle stores each data item in its own fields. In Oracle, the fields relating to a particular person, thing or event are bundled together to form a single complete unit of data, called a record (it can also be referred to as raw or an occurrence). Each record is made up of a number of fields. No two fields in a record can have the same field name.

During an Oracle Database design project, the analysis of your business needs identifies all the fields or attributes of interest. If your business needs change over time, you define any additional fields or change the definition of existing fields.

**ORACLE TABLES**

Oracle stores records relating to each other in a table. Different tables are created for the various groups of information. Related tables are grouped together to form a database.

**PRIMARY KEY**

Every table in Oracle has a field or a combination of fields that uniquely identifies each record in the table. The Unique identifier is called the Primary Key, or simply the Key. The primary key provides the means to distinguish one record from all other in a table. It allows the user and the database system to identify, locate and refer to one particular record in the database.

**RELATIONAL DATABASE**

Sometimes all the information of interest to a business operation can be stored in one table. Oracle makes it very easy to link the data in multiple tables. Matching an employee to the department in which they work is one example. This is what makes Oracle a relational database management system, or RDBMS. It stores data in two or more tables and enables you to define relationships between the tables and enables you to define relationships between the tables.

**FOREIGN KEY**

When a field is one table matches the primary key of another field is referred to as a foreign key. A foreign key is a field or a group of fields in one table whose values match those of the primary key of another table.

**REFERENTIAL INTEGRITY**

Not only does Oracle allow you to link multiple tables, it also maintains consistency between them. Ensuring that the data among related tables is correctly matched is referred to as maintaining referential integrity.

**DATA ABSTRACTION**

A major purpose of a database system is to provide users with an abstract view of the data. This system hides certain details of how the data is stored and maintained. Data abstraction is divided into three levels.

**Physical level**: This is the lowest level of abstraction at which one describes how the data are actually stored.

**Conceptual Level**: At this level of database abstraction all the attributed and what data are actually stored is described and entries and relationship among them.

**View level**: This is the highest level of abstraction at which one describes only part of the database.

**ADVANTAGES OF RDBMS**

1. Redundancy can be avoided
2. Inconsistency can be eliminated
3. Data can be Shared
4. Standards can be enforced
5. Security restrictions can be applied
6. Integrity can be maintained
7. Conflicting requirements can be balanced
8. Data independence can be achieved.

**DISADVANTAGES OF DBMS**

A significant disadvantage of the DBMS system is cost. In addition to the cost of purchasing of developing the software, the hardware has to be upgraded to allow for the extensive programs and the workspace required for their execution and storage.

**FEATURES OF ORACLE (RDBMS**)

ORACLE is one of the leading database management systems (DBMS) because it is the only Database that meets the uncompromising requirements of today’s most demanding information systems. From complex decision support systems (DSS) to the most rigorous online transaction processing (OLTP) application, even application that require simultaneous DSS and OLTP access to the same critical data, Oracle leads the industry in both performance and capability

**ENTERPRISE WIDE DATA SHARING**

The unrivaled portability and connectivity of the ORACLE DBMS enables all the systems in the organization to be linked into a singular, integrated computing resource.

**PORTABILITY**

ORACLE is fully portable to more than 80 distinct hardware and operating systems platforms, including UNIX, MSDOS, OS/2, Macintosh and dozens of proprietary platforms. This portability gives complete freedom to choose the database server platform that meets the system requirements.

**OPEN SYSTEMS**

ORACLE offers a leading implementation of industry –standard SQL. Oracle’s open architecture integrates ORACLE and non –ORACLE DBMS with industry’s most comprehensive collection of tools, application, and third party software products Oracle’s Open architecture provides transparent access to data from other relational database and even non-relational database.

**DISTRIBUTED DATA SHARING**

Oracle’s networking and distributed database capabilities to access data stored on remote server with the same ease as if the information was stored on a single local computer. A single SQL statement can access data at multiple sites. You can store data where system requirements such as performance, security or availability dictate.

**UNMATCHED PERFORMANCE**

The most advanced architecture in the industry allows the ORACLE DBMS to deliver unmatched performance.

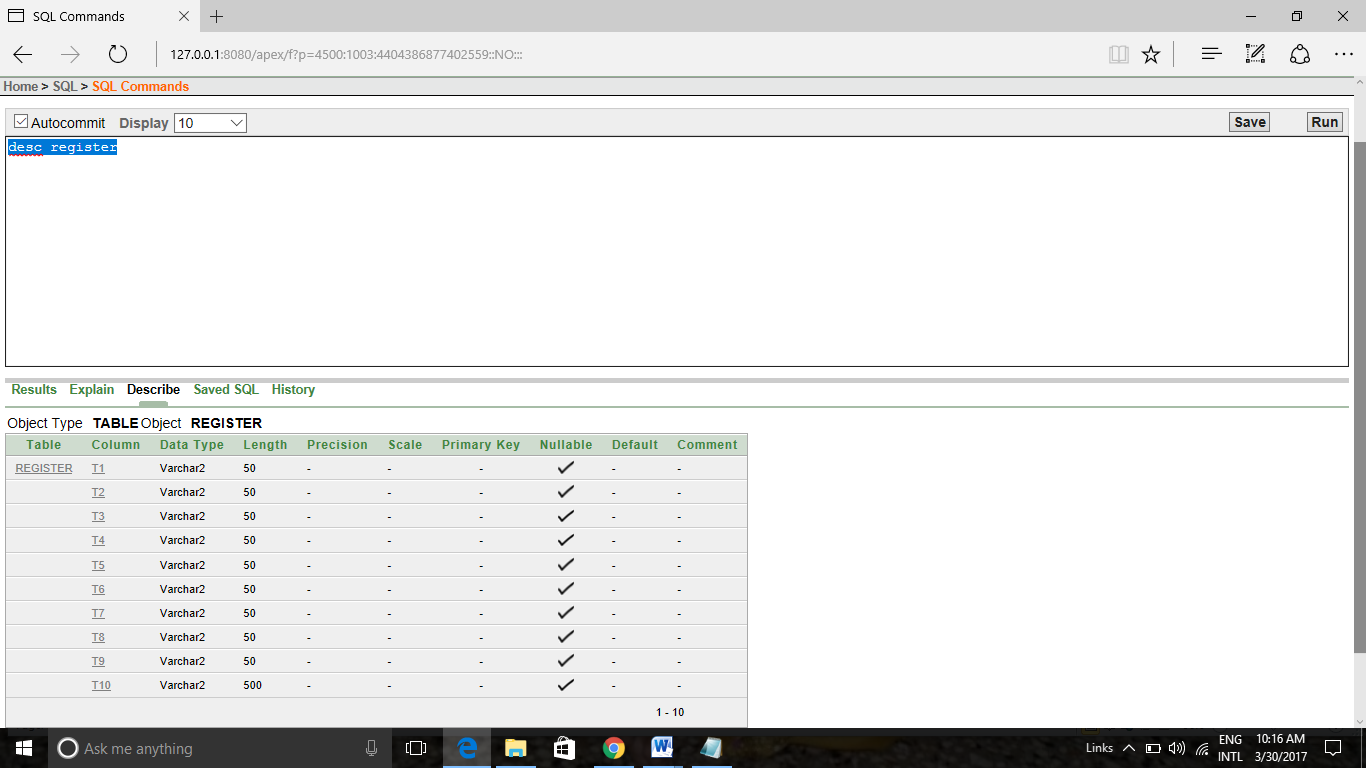
**SOPHISTICATED CONCURRENCY CONTROL**

Real World applications demand access to critical data. With most database Systems application becomes “contention bound” – which performance is limited not by the CPU power or by disk I/O, but user waiting on one another for data access . Oracle employs full, unrestricted row-level locking and contention free queries to minimize and in many cases entirely eliminates contention wait times.

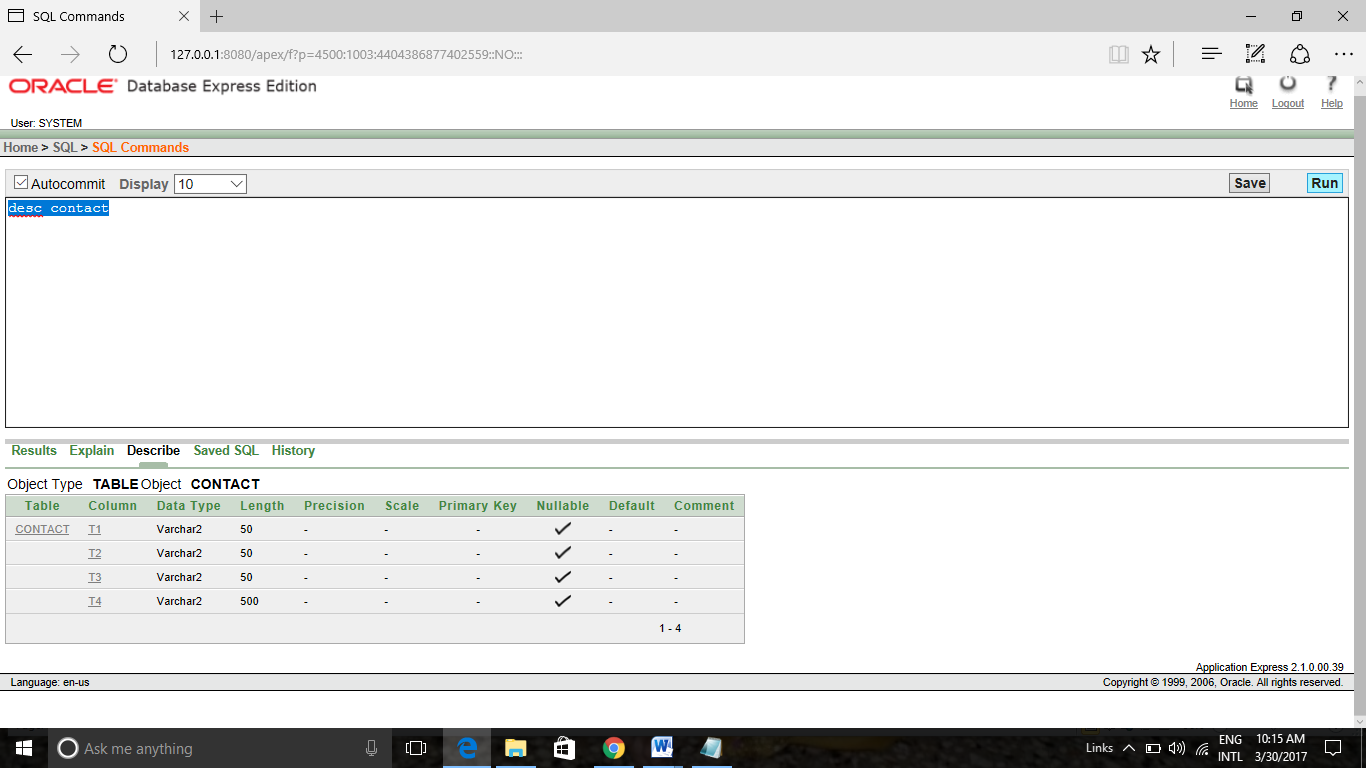
**NO I/O BOTTLENECKS**

Oracle’s fast commit groups commit and deferred write technologies dramatically reduce disk I/O bottlenecks. While some database write whole data block to disk at commit time, Oracle commits transactions with at most sequential log file on disk at commit time, On high throughput systems, one sequential writes typically group commit multiple transactions. Data read by the transaction remains as shared memory so that other transactions may access that data without reading it again from disk. Since fast commits write all data necessary to the recovery to the log file, modified blocks are written back to the database independently of the transaction commit, when written from memory to disk.

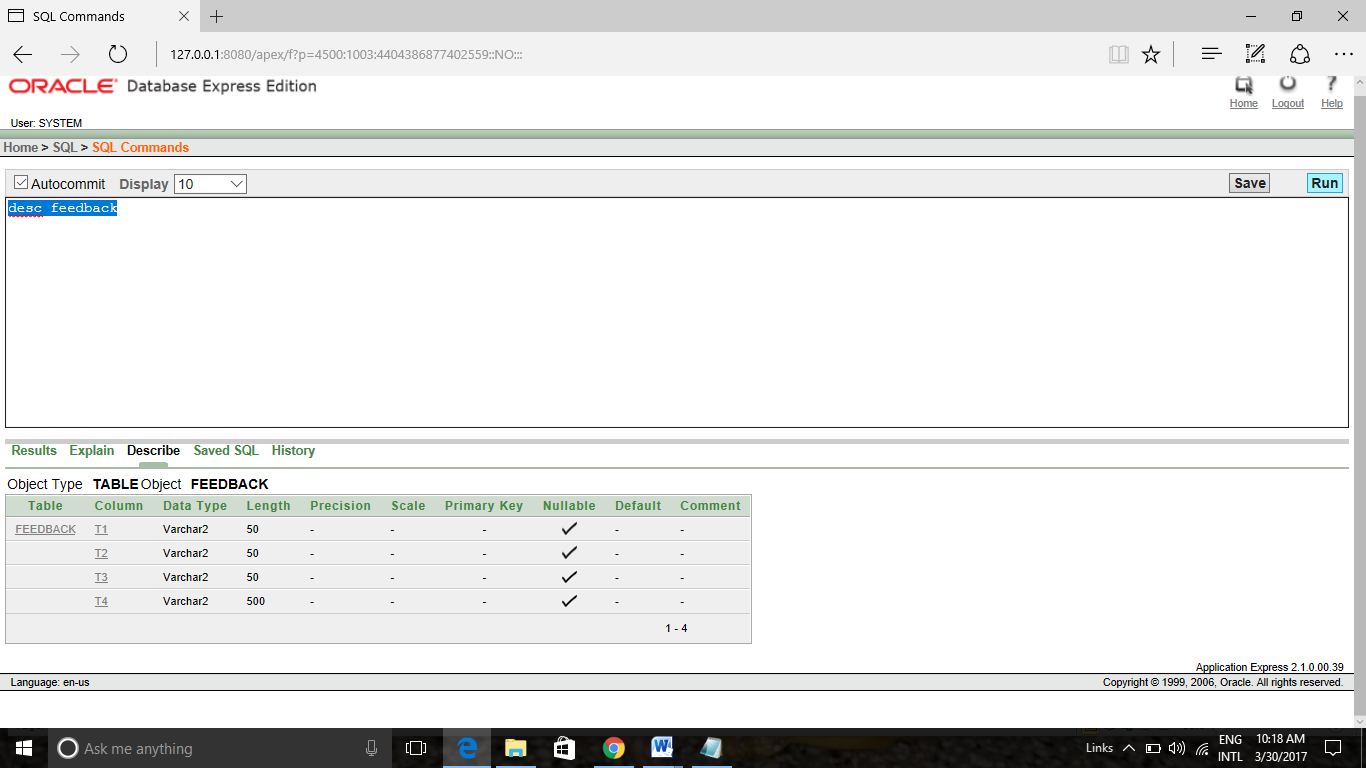
* **TABLES OF PROJECT:**
* **Go to windows->Oracle Database 10g Express…->Go to Database Home page->Database Login->SQL->SQL Commands**
* **Run.**

**For Register Table:**

**For Contact Table:**



**For Feedback Table:**



**CHAPTER-7**

**SOURCE CODE(.HTML)**

**Home Page Coding:**

<!DOCTYPE html>

<html lang="en">

<head>

<title>Home</title>

<meta charset="utf-8">

<meta name = "format-detection" content = "telephone=no" />

<link rel="icon" href="images/favicon.ico">

<link rel="shortcut icon" href="images/favicon.ico" />

<link rel="stylesheet" href="booking/css/booking.css">

<link rel="stylesheet" href="css/camera.css">

<link rel="stylesheet" href="css/owl.gymousel.css">

<link rel="stylesheet" href="css/style.css">

<script src="js/jquery.js"></script>

<script src="js/jquery-migrate-1.2.1.js"></script>

<script src="js/script.js"></script>

<script src="js/superfish.js"></script>

<script src="js/jquery.ui.totop.js"></script>

<script src="js/jquery.equalheights.js"></script>

<script src="js/jquery.mobilemenu.js"></script>

<script src="js/jquery.easing.1.3.js"></script>

<script src="js/owl.gymousel.js"></script>

<script src="js/camera.js"></script>

<!--[if (gt IE 9)|!(IE)]><!-->

<script src="js/jquery.mobile.customized.min.js"></script>

<!--<![endif]-->

<script src="booking/js/booking.js"></script>

<script>

function alpha(e)

{

var k;

document.all ? k=e.keycode:k=e.which;

return((k>64 && k<91)||(k>96 && k<123) || k==8 || k==46);

}

$(document).ready(function(){

jQuery('#camera\_wrap').camera({

loader: false,

pagination: false ,

minHeight: '444',

thumbnails: false,

height: '28.28125%',

caption: true,

navigation: true,

fx: 'mosaic'

});

$().UItoTop({ easingType: 'easeOutQuart' });

});

</script>

<!--[if lt IE 8]>

<div style=' clear: both; text-align:center; position: relative;'>

<a href="http://windows.microsoft.com/en-US/internet- explorer/products/ie/home?ocid=ie6\_countdown\_bannercode">

<img src="http://storage.ie6countdown.com/assets/100/images/banners/warning\_bar\_0000\_us.jpg" border="0" height="42" width="820" alt="You are using an outdated browser. For a faster, safer browsing experience, upgrade for free today." />

</a>

</div>

<![endif]-->

<!--[if lt IE 9]>

<script src="js/html5shiv.js"></script>

<link rel="stylesheet" media="screen" href="css/ie.css">

<![endif]-->

</head>

<script>

function fun()

{

var t1=f1.t1.value;

var t3=f1.t3.value;

var t2=f1.t2.value;

var t4=f1.t4.value;

var t5=f1.t5.value;

var t6=f1.t10.value;

if(t1=="")

{

alert("name is empty");

f1.t1.focus();

}

else if(t3=="")

{

alert("vechile no is empty");

f1.t3.focus();

}

else if(t2=="")

{

alert("phone no is empty");

f1.t2.focus();

}

else if (isNaN(t2)) //is not a number

{

alert("Please Enter a Number");

f1.t2.focus();

}

else if(t2.length!=10)

{

alert("Please give 10 digits only");

f1.t2.focus();

}

else if(t4=="")

{

alert("place is empty");

f1.t4.focus();

}

else if(t5=="")

{

alert("Time is empty");

f1.t5.focus();

}

else if(t6=="")

{

alert("Message is empty");

f1.t10.focus();

}

else

{

alert("Data submited successfully");

f1.submit();

}

}

</script>

<body class="page1" id="top">

<div class="main">

<!--==============================header=================================-->

<header>

<div class="menu\_block ">

<div class="container\_12">

<div class="grid\_12">

<nav class="horizontal-nav full-width horizontalNav-notprocessed">

<ul class="sf-menu">

<li class="current"><a href="index.html">Home</a></li>

<li><a href="index-1.html">About</a></li>

<li><a href="index-2.html">Gyms</a></li>

<li><a href="index-3.html">Services</a></li>

<li><a href="index-4.html">Contacts</a></li>

<li><a href="index-5.html">Admin</a></li>

<li><a href="feedback.html">Feedback</a></li>

</ul>

</nav>

<div class="clear"></div>

</div>

<div class="clear"></div>

</div>

</div>

<div class="container\_12">

<div class="grid\_12">

<h1>

<a href="index.html">

<img src="images/logo.png" alt="Your Happy Family">

</a>

</h1>

</div>

</div>

<div class="clear"></div>

</header>

<div class="slider\_wrapper ">

<div id="camera\_wrap" class="">

<div data-src="images/HD-Jeep.jpeg" height="500" width="500" ></div>

<div data-src="images/first.jpg" ></div>

<div data-src="images/second1.jpg" ></div>

<div data-src="images/third.jpg"></div>

</div>

</div>

<div class="container\_12">

<div class="grid\_4">

<div class="banner">

<div class="maxheight">

<div class="banner\_title">

<img src="images/icon1.png" alt="">

<div class="extra\_wrapper">Fast&amp;

<div class="color1">Safe</div>

</div>

</div>

The objective and scope of my Project Gym Parking System is to record the details various activities of user. It will simplifies the task and reduce the paper work. During implementation every user will be given appropriate training to suit their specific needs.

<a href="#" class="fa fa-share-square"></a>

</div>

</div>

</div>

<div class="grid\_4">

<div class="banner">

<div class="maxheight">

<div class="banner\_title">

<img src="images/icon2.png" alt="">

<div class="extra\_wrapper">Best

<div class="color1">Prices</div>

</div>

</div>

The objective and scope of my Project Gym Parking System is to record the details various activities of user. It will simplifies the task and reduce the paper work. During implementation every user will be given appropriate training to suit their specific needs.

</div>

</div>

</div>

<div class="grid\_4">

<div class="banner">

<div class="maxheight">

<div class="banner\_title">

<img src="images/icon3.png" alt="">

<div class="extra\_wrapper">Package

<div class="color1">Delivery</div>

</div>

</div>

The objective and scope of my Project Gym Parking System is to record the details various activities of user. It will simplifies the task and reduce the paper work. During implementation every user will be given appropriate training to suit their specific needs.

<a href="#" class="fa fa-share-square"></a>

</div>

</div>

</div>

<div class="clear"></div>

</div>

<div class="c\_phone">

<div class="container\_12">

<div class="grid\_12">

<div class="fa fa-phone"></div>+ 8197913304

<span>ORDER NOW!</span>

</div>

<div class="clear"></div>

</div>

</div>

<!--==============================Content=================================-->

<div class="content"><div class="ic"></div>

<div class="container\_12">

<div class="grid\_5">

<h3>Booking Form</h3>

<form action=register id="bookingForm" method=get name=f1>

<div class="fl1">

<div class="tmInput">

<input name="t1" placeHolder="Name:" type="text" onkeypress="return alpha(event)">

</div>

<div class="tmInput">

<input name="t2" placeHolder="Phone no:" type="number">

</div>

</div>

<div class="fl1">

<div class="tmInput">

<input name="t3" placeHolder="Vechile no:" type="text" >

</div>

<div class="tmInput mr0">

<input name="t4" placeHolder="Place:" type="text" onkeypress="return alpha(event)" >

</div>

</div>

<div class="clear"></div>

<strong>Time</strong>

<div class="tmInput">

<input name="t5" placeHolder="" type="text" >

</div>

<div class="clear"></div>

<strong>Date</strong>

<label class="tmDatepicker">

<input type="text" name="t6" placeHolder='20/10/2016' data-constraints="@NotEmpty @Required @Date">

</label>

<div class="clear"></div>

<div class="tmRadio">

<p>Online Gym Rental</p>

<input name="t7" type="radio" id="tmRadio0" data-constraints='@RadioGroupChecked(name="Comfort", groups=[RadioGroup])' checked/>

<span>General</span>

<input name="t7" type="radio" id="tmRadio1" data-constraints='@RadioGroupChecked(name="Comfort", groups=[RadioGroup])' />

<span>Sedan</span>

<input name="t7" type="radio" id="tmRadio2" data-constraints='@RadioGroupChecked(name="Comfort", groups=[RadioGroup])' />

<span>Luxary</span>

</div>

<div class="clear"></div>

<div class="fl1 fl2">

<em>Weeks</em>

<select name="t8" class="tmSelect auto" data-class="tmSelect tmSelect2" data-constraints="">

<option>0</option>

<option>1</option>

<option>2</option>

<option>3</option>

<option>4</option>

<option>5</option>

<option>6</option>

</select>

<div class="clear height1"></div>

</div>

<div class="fl1 fl2">

<em>Days</em>

<select name="t9" class="tmSelect auto" data-class="tmSelect tmSelect2" data-constraints="">

<option>0</option>

<option>1</option>

<option>2</option>

<option>3</option>

<option>4</option>

<option>5</option>

<option>6</option>

</select>

</div>

<div class="clear"></div>

<div class="tmTextarea">

<textarea name="t10" placeHolder="Message"></textarea>

</div>

<input type=button value=submit onclick='fun()'>

</form>

</div>

<div class="grid\_6 prefix\_1">

<a href="index2.html" class="type"><img src="images/econo.jpg" alt=""><span class="type\_caption">Economy</span></a>

<a href="index2.html" class="type"><img src="images/standard.jpg" alt=""><span class="type\_caption">Standard</span></a>

<a href="index2.html" class="type"><img src="images/luxx.jpg" alt=""><span class="type\_caption">Lux</span></a>

</div>

<div class="clear"></div>

</div>

</div>

</div>

<!--==============================footer=================================-->

<footer>

<div class="container\_12">

<div class="grid\_12">

<div class="f\_phone"><span>Call Us:</span> +8197913304</div>

<div class="socials">

<a href="#" class="fa fa-twitter"></a>

<a href="#" class="fa fa-facebook"></a>

<a href="#" class="fa fa-google-plus"></a>

</div>

<div class="copy">

<div class="st1">

</div>

<div class="clear"></div>

</div>

</footer>

<script>

$(function (){

$('#bookingForm').bookingForm({

ownerEmail: '#'

});

})

$(function() {

$('#bookingForm input, #bookingForm textarea').placeholder();

});

</script>

</body>

</html>

**About Page Coding:**

<!DOCTYPE html>

<html lang="en">

<head>

<title>About</title>

<meta charset="utf-8">

<meta name = "format-detection" content = "telephone=no" />

<link rel="icon" href="images/favicon.ico">

<link rel="shortcut icon" href="images/favicon.ico" />

<link rel="stylesheet" href="css/style.css">

<script src="js/jquery.js"></script>

<script src="js/jquery-migrate-1.2.1.js"></script>

<script src="js/script.js"></script>

<script src="js/superfish.js"></script>

<script src="js/jquery.ui.totop.js"></script>

<script src="js/jquery.equalheights.js"></script>

<script src="js/jquery.mobilemenu.js"></script>

<script src="js/jquery.easing.1.3.js"></script>

<script>

$(document).ready(function(){

$().UItoTop({ easingType: 'easeOutQuart' });

});

</script>

<!--[if lt IE 8]>

<div style=' clear: both; text-align:center; position: relative;'>

<a href="http://windows.microsoft.com/en-US/internet-explorer/products/ie/home?ocid=ie6\_countdown\_bannercode">

<img src="http://storage.ie6countdown.com/assets/100/images/banners/warning\_bar\_0000\_us.jpg" border="0" height="42" width="820" alt="You are using an outdated browser. For a faster, safer browsing experience, upgrade for free today." />

</a>

</div>

<![endif]-->

<!--[if lt IE 9]>

<script src="js/html5shiv.js"></script>

<link rel="stylesheet" media="screen" href="css/ie.css">

<![endif]-->

</head>

<body class="" id="top">

<div class="main">

<!--==============================header=================================-->

<header>

<div class="menu\_block">

<div class="container\_12">

<div class="grid\_12">

<nav class="horizontal-nav full-width horizontalNav-notprocessed">

<ul class="sf-menu">

<li><a href="index.html">Home</a></li>

<li class="current"><a href="index-1.html">About</a></li>

<li><a href="index-2.html">Gyms</a></li>

<li><a href="index-3.html">Services</a></li>

<li><a href="index-4.html">Contacts</a></li>

<li><a href="index-5.html">Admin</a></li>

<li><a href="feedback.html">Feedback</a></li>

</ul>

</nav>

<div class="clear"></div>

</div>

<div class="clear"></div>

</div>

</div>

<div class="container\_12">

<div class="grid\_12">

<h1>

<a href="index.html">

<img src="images/logo.png" alt="Your Happy Family">

</a>

</h1>

</div>

</div>

<div class="clear"></div>

</header>

<!--==============================Content=================================-->

<div class="content"><div class="ic">More Website Templates @ TemplateMonster.com - April 07, 2014!</div>

<div class="container\_12">

<div class="grid\_7">

<h3>A Few Words About Us</h3>

<img src="images/aboutima.jpg.jpg" alt="" class="img\_inner fleft">

<img src="images/wall1.jpg" alt="" class="img\_inner fleft">

</div>

<div class="extra\_wrapper">

<div class="text1 color2">

<a href="#">The main limitation of the previous system of Gym Parking System:</a>

The existing system only provides text-based interface, which is not as user-friendly as Graphical user Interface.<span class="color1"><a href="http://blog.templatemonster.com/free-website-templates/" rel="dofollow">link</a></span>.</p>

Since the system is implemented in Manual, so the response is very slow.The transactions are executed in off-line mode, hence on-line data capture and modification is not possible.

<span class="color1"><a href="http://www.templatemonster.com/properties/topic/society-people/" rel="nofollow">gym Parking system</a></span>

</div>

<div class="clear cl1"></div>

<p>The main objective of Gym Parking System is to enhance and upgrade the existing system by increasing its efficiency and effectiveness. The software improves the working methods by replacing the existing manual system with the computer-based system.</p>

The Gym Parking System automates each and every activity of the manual system and increases its throughput. Thus the response time of the system is very less and it works very fast.

<p>SELF DRIVE GYM RENTALS:

A any city with a young urban population. Besides the all-inclusive feel, the city is witnessing a shifting culture that calls for an equal focus on work and leisure. Almost every work day ends in a relaxing meal with friends, and frequent road trips are the norm. With youngsters increasingly viewing driving as a pleasure rather than a chore, self drive gym rentals have emerged as the answer to a prayer. </p>

<p>ABOUT GYM RENTAL:

No more worries about petrol mileage, fuel costs, insurance, and gym breakdowns! gym rental has enabled driving convenience for travellers around the country and is fast expanding its reach to cities including Ahmedabad, Bangalore, Chandigarh, Chennai, Coimbatore, Delhi-NCR, Hyderabad, Jaipur, Kochi, Kolkata, Mangalore, Mumbai and Mysore.

Self drive gyms from gym rental have given customers more control, privacy, and freedom. Book a self drive gym in any city you visit with the gym rental app on your phone and feel at home wherever you go. </p>

</div>

<div class="grid\_4 prefix\_1">

<h3>Why Choose Us</h3>

<ul class="list li">

<li class="list\_count">1</li>

<li class="extra\_wrapper">

<div class="text1 color2"><a href="#">Availability</a></div>

The Gym Parking System provides the uses a quick response with very accurate information regarding the users etc. Any details or system in an accurate manner, as and when required.

</li>

</ul>

<ul class="list li">

<li class="list\_count">2</li>

<li class="extra\_wrapper">

<div class="text1 color2"><a href="#">Enhancement</a></div>

The software Gym Parking System has a very user-friendly interface. Thus the users will feel very easy to work on it. The software provides accuracy along with a pleasant interface. Make the present manual system more interactive, speedy and user friendly.

</li>

</ul>

<ul class="list li">

<li class="list\_count">3</li>

<li class="extra\_wrapper">

<div class="text1 color2"><a href="#">Automation</a></div>

The transaction reports of the system can be retried as and when required. Thus, there is no delay in the availability of any information, whatever needed, can be captured very quickly and easily.

</li>

</ul>

<ul class="list li">

<li class="list\_count">4</li>

<li class="extra\_wrapper">

<div class="text1 color2"><a href="#">Accuracy</a></div>

The system is very user friendly and it is anticipated that functions of the system will be easily accessed by administrators, academics, students and applicants.

</li>

</ul>

</div>

<div class="clear"></div>

<div class="grid\_12">

<h3 class="h3\_\_ind1">Travel Quotes</h3>

</div>

<div class="grid\_4">

<blockquote class="bq1">

<p><i>We live in wonderful world that is full of beauty,charm and adventure.There is no end to the adventures we can have if only we seek them with our eyes open.</i></p>

<div class="color2">Jawaharial Nehru</div>

</blockquote>

</div>

<div class="grid\_4">

<blockquote class="bq1">

<p><i>Adventure is a path.Real adventure-self-determined,self-motivated,ofen risky-forces you to have firsthand encounters with the world.The world the way it is,not the way you imagine it.Your body will collide with the earth and you will bear witness.In this way you will be compelled to grapple with the limitless and bottomless cruelty of humankind-and perhaps realise that you yourself are capable of both.This will change you.Nothing will ever again be black-and-white.</i></p>

<div class="color2">Mark jenkins</div>

</blockquote>

</div>

<div class="grid\_4">

<blockquote class="bq1">

<p><i>It is better to travel well than to arrive.</i></p>

<div class="color2">Buddha</div>

</blockquote>

</div>

<div class="clear"></div>

</div>

</div>

</div>

<!--==============================footer=================================-->

<footer>

<div class="container\_12">

<div class="grid\_12">

<div class="f\_phone"><span>Call Us:</span> + 8197913304</div>

<div class="socials">

<a href="#" class="fa fa-twitter"></a>

<a href="#" class="fa fa-facebook"></a>

<a href="#" class="fa fa-google-plus"></a>

</div>

<div class="copy">

<div class="st1">

</div>

<div class="clear"></div>

</div>

</footer>

</body>

</html>

**Gyms Page Coding:**

<!DOCTYPE html>

<html lang="en">

<head>

<title>Gyms</title>

<meta charset="utf-8">

<meta name = "format-detection" content = "telephone=no" />

<link rel="icon" href="images/favicon.ico">

<link rel="shortcut icon" href="images/favicon.ico" />

<link rel="stylesheet" href="css/touchTouch.css">

<link rel="stylesheet" href="css/style.css">

<script src="js/jquery.js"></script>

<script src="js/jquery-migrate-1.2.1.js"></script>

<script src="js/script.js"></script>

<script src="js/superfish.js"></script>

<script src="js/jquery.ui.totop.js"></script>

<script src="js/jquery.equalheights.js"></script>

<script src="js/jquery.mobilemenu.js"></script>

<script src="js/jquery.easing.1.3.js"></script>

<script src="js/touchTouch.jquery.js"></script>

<script>

$(document).ready(function(){

$().UItoTop({ easingType: 'easeOutQuart' });

$('.gallery a.gal').touchTouch();

});

</script>

<!--[if lt IE 8]>

<div style=' clear: both; text-align:center; position: relative;'>

<a href="http://windows.microsoft.com/en-US/internet-explorer/products/ie/home?ocid=ie6\_countdown\_bannercode">

<img src="http://storage.ie6countdown.com/assets/100/images/banners/warning\_bar\_0000\_us.jpg" border="0" height="42" width="820" alt="You are using an outdated browser. For a faster, safer browsing experience, upgrade for free today." />

</a>

</div>

<![endif]-->

<!--[if lt IE 9]>

<script src="js/html5shiv.js"></script>

<link rel="stylesheet" media="screen" href="css/ie.css">

<![endif]-->

</head>

<body class="" id="top">

<div class="main">

<!--==============================header=================================-->

<header>

<div class="menu\_block ">

<div class="container\_12">

<div class="grid\_12">

<nav class="horizontal-nav full-width horizontalNav-notprocessed">

<ul class="sf-menu">

<li><a href="index.html">Home</a></li>

<li><a href="index-1.html">About</a></li>

<li class="current"><a href="index-2.html">Gyms</a></li>

<li><a href="index-3.html">Services</a></li>

<li><a href="index-4.html">Contacts</a></li>

<li><a href="index-5.html">Admin</a></li>

<li><a href="feedback.html">Feedback</a></li>

</ul>

</nav>

<div class="clear"></div>

</div>

<div class="clear"></div>

</div>

</div>

<div class="container\_12">

<div class="grid\_12">

<h1>

<a href="index.html">

<img src="images/logo.png" alt="Your Happy Family">

</a>

</h1>

</div>

</div>

<div class="clear"></div>

</header>

<!--==============================Content=================================-->

<div class="content"><div class="ic">More Website Templates @ TemplateMonster.com - April 07, 2014!</div>

<div class="container\_12">

<div class="grid\_12">

<h3>Economy</h3>

</div>

<div class="clear"></div>

<div class="gallery">

<div class="grid\_4"><a href="images/big1.jpg" class="gal"><img src="images/eco.jpg" alt=""></a></div>

<div class="grid\_4"><a href="images/big2.jpg" class="gal"><img src="images/figo.jpg" alt=""></a></div>

<div class="grid\_4"><a href="images/big3.jpg" class="gal"><img src="images/alto1.jpg" alt=""></a></div>

</div>

<div class="grid\_12">

<h3>Standard</h3>

</div>

<div class="clear"></div>

<div class="gallery">

<div class="grid\_4"><a href="images/big4.jpg" class="gal"><img src="images/page3\_img4.jpg" alt=""></a></div>

<div class="grid\_4"><a href="images/big5.jpg" class="gal"><img src="images/standard1.jpg" alt=""></a></div>

<div class="grid\_4"><a href="images/big6.jpg" class="gal"><img src="images/toyota.jpg" alt=""></a></div>

</div>

<div class="grid\_12">

<h3>Lux</h3>

</div>

<div class="clear"></div>

<div class="gallery">

<div class="grid\_4"><a href="images/big7.jpg" class="gal"><img src="images/lu.jpg" alt=""></a></div>

<div class="grid\_4"><a href="images/big8.jpg" class="gal"><img src="images/benz.jpg" alt=""></a></div>

<div class="grid\_4"><a href="images/big9.jpg" class="gal"><img src="images/bmw.jpg" alt=""></a></div>

</div>

<div class="clear"></div>

</div>

</div>

</div>

<!--==============================footer=================================-->

<footer>

<div class="container\_12">

<div class="grid\_12">

<div class="f\_phone"><span>Call Us:</span> + 8197913304</div>

<div class="socials">

<a href="#" class="fa fa-twitter"></a>

<a href="#" class="fa fa-facebook"></a>

<a href="#" class="fa fa-google-plus"></a>

</div>

<div class="copy">

<div class="st1">

</div>

<div class="clear"></div>

</div>

</footer>

</body>

</html>

**Services Page Coding:**

<!DOCTYPE html>

<html lang="en">

<head>

<title>Services</title>

<meta charset="utf-8">

<meta name = "format-detection" content = "telephone=no" />

<link rel="icon" href="images/favicon.ico">

<link rel="shortcut icon" href="images/favicon.ico" />

<link rel="stylesheet" href="css/style.css">

<script src="js/jquery.js"></script>

<script src="js/jquery-migrate-1.2.1.js"></script>

<script src="js/script.js"></script>

<script src="js/superfish.js"></script>

<script src="js/jquery.ui.totop.js"></script>

<script src="js/jquery.equalheights.js"></script>

<script src="js/jquery.mobilemenu.js"></script>

<script src="js/jquery.easing.1.3.js"></script>

<script>

$(document).ready(function(){

$().UItoTop({ easingType: 'easeOutQuart' });

});

</script>

<!--[if lt IE 8]>

<div style=' clear: both; text-align:center; position: relative;'>

<a href="http://windows.microsoft.com/en-US/internet-explorer/products/ie/home?ocid=ie6\_countdown\_bannercode">

<img src="http://storage.ie6countdown.com/assets/100/images/banners/warning\_bar\_0000\_us.jpg" border="0" height="42" width="820" alt="You are using an outdated browser. For a faster, safer browsing experience, upgrade for free today." />

</a>

</div>

<![endif]-->

<!--[if lt IE 9]>

<script src="js/html5shiv.js"></script>

<link rel="stylesheet" media="screen" href="css/ie.css">

<![endif]-->

</head>

<body class="" id="top">

<div class="main">

<!--==============================header=================================-->

<header>

<div class="menu\_block ">

<div class="container\_12">

<div class="grid\_12">

<nav class="horizontal-nav full-width horizontalNav-notprocessed">

<ul class="sf-menu">

<li><a href="index.html">Home</a></li>

<li><a href="index-1.html">About</a></li>

<li><a href="index-2.html">Gyms</a></li>

<li class="current"><a href="index-3.html">Services</a></li>

<li><a href="index-4.html">Contacts</a></li>

<li><a href="index-5.html">Admin</a></li>

<li><a href="feedback.html">Feedback</a></li>

</ul>

</nav>

<div class="clear"></div>

</div>

<div class="clear"></div>

</div>

</div>

<div class="container\_12">

<div class="grid\_12">

<h1>

<a href="index.html">

<img src="images/logo.png" alt="Your Happy Family">

</a>

</h1>

</div>

</div>

<div class="clear"></div>

</header>

<!--==============================Content=================================-->

<div class="content"><div class="ic">More Website Templates @ TemplateMonster.com - April 07, 2014!</div>

<div class="container\_12">

<div class="grid\_12">

<h3>Services Overview</h3>

</div>

<div class="grid\_4">

<div class="box">

<div class="maxheight">

<img src="images/rsz\_diesel-fuel-2.jpg" alt="">

<div class="text1 color2">

<a href="#">Fuel Cost Included</a>

</div>

Don’t worry about mileage! All fuel costs are included. If you refill fuel, we’ll pay you back!

<br>

</div>

</div>

</div>

<div class="grid\_4">

<div class="box">

<div class="maxheight">

<img src="images/hidden\_charges.jpg" alt="">

<div class="text1 color2">

<a href="#">No Hidden Fees </a>

</div>

Our prices include taxes and insurance.

What you see is what you really pay!

<br>

</div>

</div>

</div>

<div class="grid\_4">

<div class="box">

<div class="maxheight">

<img src="images/flexi.jpg" alt="">

<div class="text1 color2">

<a href="#">Flexi Pricing Packages </a>

</div>

One size never fits all! Choose a balance of time and kilometers that works best for you.

<br>

</div>

</div>

</div>

<div class="grid\_4">

<div class="box">

<div class="maxheight">

<img src="images/logo-goanywhere.png" alt="">

<div class="text1 color2">

<a href="#">Go Anywhere</a>

</div>

Our gyms have all-India permits.Just remember to pay state tolls and entry taxes.

<br>

</div>

</div>

</div>

<div class="grid\_4">

<div class="box">

<div class="maxheight">

<img src="images/roadside.jpg" alt="">

<div class="text1 color2">

<a href="#">24x7 Roadside Assistance</a>

</div>

We have round-the-clock, pan India partners. Help is never far away from you.

<br>

</div>

</div>

</div>

<div class="grid\_4">

<div class="box">

<div class="maxheight">

<img src="images/gym-512.png" alt="">

<div class="text1 color2">

<a href="#">Damage Insurance</a>

</div>

All your bookings include damage insurance! Drive safe, but don’t worry!

<br>

</div>

</div>

</div>

<div class="grid\_4">

<div class="box">

<div class="maxheight">

<img src="images/gymlite.png" alt="">

<div class="text1 color2">

<a href="#"> FLEXI PRICING GYM LITE </a>

</div>

Fuel Included,

5 km/hour,

~25% Cheaper Total Cost,

IDEAL FOR

Long Outstation Trips

Multiday Bookings

Office Commute.

<br>

</div>

</div>

</div>

<div class="grid\_4">

<div class="box">

<div class="maxheight">

<img src="images/classic.png" alt="">

<div class="text1 color2">

<a href="#">GYM CLASSIC </a>

</div>

Fuel Included,

10 km/hour,

Balance of Kilometer & Time,

IDEAL FOR

General Usage

Weekend Trips.

<br>

</div>

</div>

</div>

<div class="grid\_4">

<div class="box">

<div class="maxheight">

<img src="images/xl.png" alt="">

<div class="text1 color2">

<a href="#">GYM XL </a>

</div>

Fuel Included

15 km/hour,

~15% Cheaper Per Kilometer,

IDEAL FOR

Outstation Day Trips

Running Around The City

The Joy Of Driving.

<br>

</div>

</div>

</div>

<div class="grid\_4">

<div class="box">

<div class="maxheight">

<img src="images/zero.jpg" alt="">

<div class="text1 color2">

<a href="#">ZERO SECURITY DEPOSIT </a>

</div>

Supermilers don't have to pay our standard security deposit ever.

<br>

</div>

</div>

</div>

<div class="grid\_4">

<div class="box">

<div class="maxheight">

<img src="images/earn-points.png" alt="">

<div class="text1 color2">

<a href="#">EARN Z POINTS </a>

</div>

Earn Z - point for every Booking and redeem for free rides

<br>

</div>

</div>

</div>

<div class="grid\_4">

<div class="box">

<div class="maxheight">

<img src="images/call.png" alt="">

<div class="text1 color2">

<a href="#">PRIORITY CALL CENTER SUPPORT</a>

</div>

24\*7 priority customer service with minimum wait time

<br>

</div>

</div>

</div>

<div class="grid\_4">

<div class="box">

<div class="maxheight">

<img src="images/benz.jpg" alt="">

<div class="text1 color2">

<a href="#">BOOK </a>

</div>

Sign up and book your machine, all on our site! You decide the duration, location, and gym. It’s all super easy.

<br>

</div>

</div>

</div>

<div class="grid\_4">

<div class="box">

<div class="maxheight">

<img src="images/license.png" alt="">

<div class="text1 color2">

<a href="#">UPLOAD LICENSE</a>

</div>

Upload your driver’s license, pay the small security deposit and your booking is confirmed.

<br>

</div>

</div>

</div>

<div class="grid\_4">

<div class="box">

<div class="maxheight">

<img src="images/Unlock.png" alt="">

<div class="text1 color2">

<a href="#">UNLOCK</a>

</div>

We will SMS your gym’s number plate 20 minutes before your booking starts. Unlock it via the Zoomgym app or SMS.

<br>

</div>

</div>

</div>

<div class="grid\_4">

<div class="box">

<div class="maxheight">

<img src="images/return-button.png" alt="">

<div class="text1 color2">

<a href="#">RETURN</a>

</div>

Bring the gym back to the same location, and fill the return checklist to end your booking. You are all set!

<br>

</div>

</div>

</div>

<div class="clear"></div>

</div>

</div>

</div>

<!--==============================footer=================================-->

<footer>

<div class="container\_12">

<div class="grid\_12">

<div class="f\_phone"><span>Call Us:</span> + 8197913304</div>

<div class="socials">

<a href="#" class="fa fa-twitter"></a>

<a href="#" class="fa fa-facebook"></a>

<a href="#" class="fa fa-google-plus"></a>

</div>

<div class="copy">

<div class="st1">

</div>

</div>

<div class="clear"></div>

</div>

</footer>

</body>

</html>

**Contacts Page Coding:**

<!DOCTYPE html>

<html lang="en">

<head>

<script>

function ValidateForm()

{

var X=document.forms["myForm"]["t2"].value;

var atpos=x.indexOf("@");

var dotpos=x.lastIndexOf(".");

if (atpos<1 || dotpos<atpos+2 || dotpos<atpos+3 || dotpos+3>=x.length)

{

alert("Not a valid e-mail address");

return false;

}

}

function alpha(e)

{

var k;

document.all ? k=e.keycode:k=e.which;

return((k>64 && k<91)||(k>96 && k<123) || k==8 || k==46);

}

function fun()

{

var t1=f1.t1.value;

var t2=f1.t2.value;

var t3=f1.t3.value;

var t4=f1.t4.value;

if(t1=="")

{

alert("name is empty");

f1.t1.focus();

}

else if(t2=="")

{

alert("email is empty");

f1.t2.focus();

}

else if(t3=="")

{

alert("phone no is empty");

f1.t3.focus();

}

else if (isNaN(t3)) //is not a number

{

alert("Please Enter a Number");

f1.t3.focus();

}

else if(t3.length!=10)

{

alert("Please give 10 digits only");

f1.t3.focus();

}

else if(t4=="")

{

alert("message is empty");

f1.t4.focus();

}

else

{

alert("Thank you for Contact");

f1.submit();

}

}

</script>

</head>

<title>Contact</title>

<meta charset="utf-8">

<meta name = "format-detection" content = "telephone=no" />

<link rel="icon" href="images/favicon.ico">

<link rel="shortcut icon" href="images/favicon.ico" />

<link rel="stylesheet" href="css/form.css">

<link rel="stylesheet" href="css/style.css">

<script src="js/jquery.js"></script>

<script src="js/jquery-migrate-1.2.1.js"></script>

<script src="js/script.js"></script>

<script src="js/superfish.js"></script>

<script src="js/jquery.ui.totop.js"></script>

<script src="js/jquery.equalheights.js"></script>

<script src="js/jquery.mobilemenu.js"></script>

<script src="js/jquery.easing.1.3.js"></script>

<script src="js/TMForm.js"></script>

<script>

$(document).ready(function(){

$().UItoTop({ easingType: 'easeOutQuart' });

});

</script>

<!--[if lt IE 8]>

<div style=' clear: both; text-align:center; position: relative;'>

<a href="http://windows.microsoft.com/en-US/internet-explorer/products/ie/home?ocid=ie6\_countdown\_bannercode">

<img src="http://storage.ie6countdown.com/assets/100/images/banners/warning\_bar\_0000\_us.jpg" border="0" height="42" width="820" alt="You are using an outdated browser. For a faster, safer browsing experience, upgrade for free today." />

</a>

</div>

<![endif]-->

<!--[if lt IE 9]>

<script src="js/html5shiv.js"></script>

<link rel="stylesheet" media="screen" href="css/ie.css">

<![endif]-->

</head>

<body class="" id="top">

<div class="main">

<!--==============================header=================================-->

<header>

<div class="menu\_block ">

<div class="container\_12">

<div class="grid\_12">

<nav class="horizontal-nav full-width horizontalNav-notprocessed">

<ul class="sf-menu">

<li><a href="index.html">Home</a></li>

<li><a href="index-1.html">About</a></li>

<li><a href="index-2.html">Gyms</a></li>

<li><a href="index-3.html">Services</a></li>

<li class="current"><a href="index-4.html">Contacts</a></li>

<li><a href="index-5.html">Admin</a></li>

<li><a href="feedback.html">Feedback</a></li>

</ul>

</nav>

<div class="clear"></div>

</div>

<div class="clear"></div>

</div>

</div>

<div class="container\_12">

<div class="grid\_12">

<h1>

<a href="index.html">

<img src="images/logo.png" alt="Your Happy Family">

</a>

</h1>

</div>

</div>

<div class="clear"></div>

</header>

<!--==============================Content=================================-->

<div class="content"><div class="ic">More Website Templates @ TemplateMonster.com - April 07, 2014!</div>

<div class="container\_12">

<div class="grid\_12">

<h3>contacts</h3>

<div class="map">

<figure>

<iframe src="https://www.google.com/maps/embed?pb=!1m14!1m12!1m3!1d24214.807650104907!2d-73.94846048422478!3d40.65521573400813!2m3!1f0!2f0!3f0!3m2!1i1024!2i768!4f13.1!5e0!3m2!1sen!2sus!4v1395650655094" style="border:0"></iframe>

</figure>

</div>

</div>

<div class="grid\_5">

<h3>Contact Info</h3>

<div class="map">

<div class="text1 color2">Gym Parking System</div>

<p>The main objective of Gym Parking System is to enhance and upgrade the existing system by increasing its efficiency and effectiveness. The software improves the working methods by replacing the existing manual system with the computer-based system.<span class="color1"><a href="http://www.templatemonster.com/website-templates.php" rel="nofollow"> </a></span> only. Free ones lack this privilege.</p>

<p>Use <span class="color1"><a href="http://www.templatetuning.com/" rel="nofollow"> </a></span> services if you need any help in customization of your gym Parking.</p>

<address>

<dl>

<dt>Gym Parking System<br>

<br>

RT Nagar

</dt>

<dd><span>Telephone:</span> +8197913304</dd>

<dd><span>FAX:</span> +8197913304</dd>

<dd>E-mail: <a href="#" class="color1">vaishu9520@gmail.com</a></dd>

</dl>

</address>

</div>

</div>

<div class="grid\_6 prefix\_1">

<h3>Contact Form</h3>

<form id="form" name="f1" action="index-4.html" onsubmit="return validateForm();"method="post">

<div class="success\_wrapper">

<div class="success-message">Contact form submitted</div>

</div>

<label class="name">

<input type="text" name=t1 placeholder="Name:" onkeypress="return alpha(event)" />

</label>

<label class="email">

<input type="text" name=t2 placeholder="E-mail:" />

</label>

<label class="phone">

<input type="number" name=t3 placeholder="Phone:" />

</label>

<label class="message">

<textarea placeholder="Message:" name=t4 ></textarea>

</label>

<div>

<div class="clear"></div>

<div class="btns">

<input type="submit" value="submit" onclick='fun()'>

</div>

</div>

</form>

</div>

<div class="clear"></div>

</div>

</div>

</div>

<!--==============================footer=================================-->

<footer>

<div class="container\_12">

<div class="grid\_12">

<div class="f\_phone"><span>Call Us:</span> + 8197913304</div>

<div class="socials">

<a href="#" class="fa fa-twitter"></a>

<a href="#" class="fa fa-facebook"></a>

<a href="#" class="fa fa-google-plus"></a>

</div>

<div class="copy">

<div class="st1">

</div>

<div class="clear"></div>

</div>

</footer>

</body>

</html>

**Admin Form:**

<!DOCTYPE html>

<html lang="en">

<head>

<title>admin</title>

<script type="text/javascript">

function validate(form)

{

var userName=form.t1.value;

var password=form.t2.value;

if(userName.length===0)

{

alert("you must enter a username");

return false;

}

if(password.length<6)

{

alert("you must 6 char");

return false;

}

return true;

}

</script>

<meta charset="utf-8">

<meta name = "format-detection" content = "telephone=no" />

<link rel="icon" href="images/favicon.ico">

<link rel="shortcut icon" href="images/favicon.ico" />

<link rel="stylesheet" href="css/style.css">

<script src="js/jquery.js"></script>

<script src="js/jquery-migrate-1.2.1.js"></script>

<script src="js/script.js"></script>

<script src="js/superfish.js"></script>

<script src="js/jquery.ui.totop.js"></script>

<script src="js/jquery.equalheights.js"></script>

<script src="js/jquery.mobilemenu.js"></script>

<script src="js/jquery.easing.1.3.js"></script>

<script>

$(document).ready(function(){

$().UItoTop({ easingType: 'easeOutQuart' });

});

</script>

<!--[if lt IE 8]>

<div style=' clear: both; text-align:center; position: relative;'>

<a href="http://windows.microsoft.com/en-US/internet-explorer/products/ie/home?ocid=ie6\_countdown\_bannercode">

<img src="http://storage.ie6countdown.com/assets/100/images/banners/warning\_bar\_0000\_us.jpg" border="0" height="42" width="820" alt="You are using an outdated browser. For a faster, safer browsing experience, upgrade for free today." />

</a>

</div>

<![endif]-->

<!--[if lt IE 9]>

<script src="js/html5shiv.js"></script>

<link rel="stylesheet" media="screen" href="css/ie.css">

<![endif]-->

</head>

<script>

function fun()

{

var t1=f1.t1.value;

var t2=f1.t2.value;

if(t1=="")

{

alert("user name is empty");

f1.t1.focus();

}

else if(t2=="")

{

alert("password is empty");

f1.t2.focus();

}

else

{

alert("data submited successfully");

f1.submit();

}

}

</script>

<body class="" id="top">

<div class="main">

<!--==============================header=================================-->

<header>

<div class="menu\_block ">

<div class="container\_12">

<div class="grid\_12">

<nav class="horizontal-nav full-width horizontalNav-notprocessed">

<ul class="sf-menu">

<li><a href="index.html">Home</a></li>

<li><a href="index-1.html">About</a></li>

<li><a href="index-2.html">Gyms</a></li>

<li><a href="index-3.html">Services</a></li>

<li><a href="index-4.html">Contacts</a></li>

<li class="current"><a href="index-5.html">Admin</a></li>

<li><a href="feedback.html">Feedback</a></li>

</ul>

</nav>

<div class="clear"></div>

</div>

<div class="clear"></div>

</div>

</div>

<div class="container\_12">

<div class="grid\_12">

<h1>

<a href="index.html">

<img src="images/logo.png" alt="Your Happy Family">

</a>

</h1>

</div>

</div>

<div class="clear"></div>

</header>

<!--==============================Content=================================-->

<div class="content"><div class="ic">More Website Templates @ TemplateMonster.com - April 07, 2014!</div>

<div class="container\_12">

<div class="grid\_12">

<h3>Admin Page</h3>

</div>

<div class="grid\_4">

<div class="box">

<div class="maxheight">

<img src="images/econo.jpg" alt="">

<div class="text1 color2">

<a href="#">Economy. </a>

</div>

Finding a solid deal on a Economy gym for rent in India can be tough, but that's why we're here. Travelocity has worked with Economy to bring you cheap rates on great rental gyms in India.These deals are going to send you off in a great Economy India rental without giving your wallet too much punishment. So enjoy your vacation or your next business trip in knowing you found the low rate gym rental in India from Economy you wanted.

<br>

</div>

</div>

</div>

<div class="grid\_4">

<div class="box">

<div class="maxheight">

<img src="images/Figo-Composite.jpg" alt="">

<div class="text1 color2">

<a href="#">Standard. </a>

</div>

When it comes to renting a gym these days, there are so many choices available: Gym rental companies, gym classes, pick up/dropoff locations, cheap rates or expensive rates.We know you’re interested in finding a low rate on a Standard gym from Economy so we’ve put together all the info you would need to book a Economy Standard gym rental, all in one place.You can find more deals for Economy Standard gyms below, but don’t let that stop you. You can also check out top airports and cities where Economy operates along with other gym class options, if you’re open to change.

<br>

</div>

</div>

</div>

<div class="grid\_4">

<div class="box">

<div class="maxheight">

<img src="images/bmbez.jpg" alt="">

<div class="text1 color2">

<a href="#">Lux. </a>

</div>

When it comes to renting a gym these days, there are so many choices available: Gym rental companies, gym classes, pick up/dropoff locations, cheap rates or expensive rates.We know you’re interested in finding a low rate on a Luxury gym from Economy so we’ve put together all the info you would need to book a Economy Luxury gym rental, all in one place.You can find more deals for Economy Luxury gyms below, but don’t let that stop you. You can also check out top airports and cities where Economy operates along with other gym class options, if you’re open to change

<br>

</div>

</div>

</div>

<div class="clear"></div>

</div>

</div>

</div>

<a href="#" class="btn">Admin Login</a>

<form action=admin method="get" onsubmit="return validate(this);">

<table border=2>

<tr><td> Enter your Username<td><input type=text name=t1 size=20></tr>

<tr><td> <td></tr>

<tr><td> Enter your Password<td><input type=password name=t2 size=20></tr>

<tr><td><td><input type=submit value=submit ></tr>

</table>

</form>

<!--==============================footer=================================-->

<footer>

<div class="container\_12">

<div class="grid\_12">

<div class="f\_phone"><span>Call Us:</span> + 8197913304</div>

<div class="socials">

<a href="#" class="fa fa-twitter"></a>

<a href="#" class="fa fa-facebook"></a>

<a href="#" class="fa fa-google-plus"></a>

</div>

<div class="copy">

<div class="st1">

</div>

</div>

<div class="clear"></div>

</div>

</footer>

</body>

</html>

**Feedback Form:**

<html lang="en">

<head>

<title>feedback</title>

<meta charset="utf-8">

<meta name = "format-detection" content = "telephone=no" />

<link rel="icon" href="images/favicon.ico">

<link rel="shortcut icon" href="images/favicon.ico" />

<link rel="stylesheet" href="css/style.css">

<script src="js/jquery.js"></script>

<script src="js/jquery-migrate-1.2.1.js"></script>

<script src="js/script.js"></script>

<script src="js/superfish.js"></script>

<script src="js/jquery.ui.totop.js"></script>

<script src="js/jquery.equalheights.js"></script>

<script src="js/jquery.mobilemenu.js"></script>

<script <!DOCTYPE html>

src="js/jquery.easing.1.3.js"></script>

<script>

$(document).ready(function(){

$().UItoTop({ easingType: 'easeOutQuart' });

});

</script>

<!--[if lt IE 8]>

<div style=' clear: both; text-align:center; position: relative;'>

<a href="http://windows.microsoft.com/en-US/internet-explorer/products/ie/home?ocid=ie6\_countdown\_bannercode">

<img src="http://storage.ie6countdown.com/assets/100/images/banners/warning\_bar\_0000\_us.jpg" border="0" height="42" width="820" alt="You are using an outdated browser. For a faster, safer browsing experience, upgrade for free today." />

</a>

</div>

<![endif]-->

<!--[if lt IE 9]>

<script src="js/html5shiv.js"></script>

<link rel="stylesheet" media="screen" href="css/ie.css">

<![endif]-->

</head>

<body class="" id="top">

<div class="main">

<!--==============================header=================================-->

<header>

<div class="menu\_block ">

<div class="container\_12">

<div class="grid\_12">

<nav class="horizontal-nav full-width horizontalNav-notprocessed">

<ul class="sf-menu">

<li><a href="index.html">Home</a></li>

<li><a href="index-1.html">About</a></li>

<li><a href="index-2.html">Gyms</a></li>

<li><a href="index-3.html">Services</a></li>

<li><a href="index-4.html">Contacts</a></li>

<li><a href="index-5.html">Admin</a></li>

<li class="current"><a href="feedback.html">Feedback</a></li>

</ul>

</nav>

<div class="clear"></div>

</div>

<div class="clear"></div>

</div>

</div>

<div class="container\_12">

<div class="grid\_12">

<h1>

<a href="index.html">

<img src="images/logo.png" alt="Your Happy Family">

</a>

</h1>

</div>

</div>

<div class="clear"></div>

</header>

<!--==============================Content=================================-->

<div class="content"><div class="ic">More Website Templates @ TemplateMonster.com - April 07, 2014!</div>

<div class="container\_12">

<div class="grid\_12">

<h3>Feedback</h3>

</div>

<div class="grid\_4">

<div class="box">

<div class="maxheight">

<img src="images/comfort.jpg" alt="">

<div class="text1 color2">

<a href="#">Comfortable zone feedback </a>

</div>

<a href="feedback1" class="btn">Report of Customer FeedBack</a>

</div>

</div>

</div>

<div class="grid\_4">

<div class="box">

<div class="maxheight">

<img src="images/family.jpg" alt="">

<div class="text1 color2">

<a href="#">Family trip feedback </a>

</div>

<a href="contact1" class="btn">Report of Customer Contact</a>

</div>

</div>

</div>

<div class="grid\_4">

<div class="box">

<div class="maxheight">

<img src="images/aboutimg1.jpg" alt="">

<div class="text1 color2">

<a href="#">Friends trip feedback </a>

</div>

<a href="register1" class="btn">Report of Customer Booked</a>

<a href="index.html" class="btn">Logout</a>

</div>

</div>

</div>

<div class="clear"></div>

</div>

</div>

</div>

<!--==============================footer=================================-->

<footer>

<div class="container\_12">

<div class="grid\_12">

<div class="f\_phone"><span>Call Us:</span> + 8197913304</div>

<div class="socials">

<a href="#" class="fa fa-twitter"></a>

<a href="#" class="fa fa-facebook"></a>

<a href="#" class="fa fa-google-plus"></a>

</div>

<div class="copy">

<div class="st1">

</div>

</div>

<div class="clear"></div>

</div>

</footer>

</body>

</html>

**CHAPTER-8**

**CODING PART(.JAVA)**

**Admin Coding:**

import java.sql.\*;

import java.io.\*;

import javax.servlet.\*;

import javax.servlet.http.\*;

public class admin extends HttpServlet

{

PreparedStatement st=null;

Connection con=null;

ResultSet rs=null;

int i=0;

public void init()

{

System.out.println("init");

try

{

Class.forName("oracle.jdbc.driver.OracleDriver");

con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:XE","system","1234");

}

catch(Exception ae)

{}

}

public void doGet(HttpServletRequest req, HttpServletResponse res)throws ServletException, IOException

{

res.setContentType("text/html");

PrintWriter out = res.getWriter();

String a=req.getParameter("t1");

String b=req.getParameter("t2");

if(a.equals("admin")&& b.equals("admin"))

res.sendRedirect("index-6.html");

else

out.println("Login failure");

out.println("<a href=index-5.html>Login Again</a>");

out.println("</body>");

out.println("</html>");

}

}

**Contact Coding:**

import java.sql.\*;

import java.io.\*;

import javax.servlet.\*;

import javax.servlet.http.\*;

public class contact extends HttpServlet

{

PreparedStatement st=null;

Connection con=null;

public void init()

{

System.out.println("init");

try

{

Class.forName("oracle.jdbc.driver.OracleDriver");

con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:XE","system","1234");

}

catch(Exception ae)

{}

}

public void doGet(HttpServletRequest req, HttpServletResponse res)throws ServletException, IOException

{

res.setContentType("text/html");

PrintWriter out = res.getWriter();

String a=req.getParameter("t1");

String b=req.getParameter("t2");

String c=req.getParameter("t3");

String d=req.getParameter("t4");

try

{

st=con.prepareStatement("insert into contact values(?,?,?,?)");

st.setString(1,a);

st.setString(2,b);

st.setString(3,c);

st.setString(4,d);

st.execute();

}

catch(Exception at)

{}

res.sendRedirect("index.html");

out.println("</body>");

out.println("</html>");

}

}

**Feedback Coding:**

import java.sql.\*;

import java.io.\*;

import javax.servlet.\*;

import javax.servlet.http.\*;

public class feedback extends HttpServlet

{

PreparedStatement st=null;

Connection con=null;

public void init()

{

System.out.println("init");

try

{

Class.forName("oracle.jdbc.driver.OracleDriver");

con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:XE","system","1234");

}

catch(Exception ae)

{}

}

public void doGet(HttpServletRequest req, HttpServletResponse res)throws ServletException, IOException

{

res.setContentType("text/html");

PrintWriter out = res.getWriter();

String a=req.getParameter("t1");

String b=req.getParameter("t2");

String c=req.getParameter("t3");

String d=req.getParameter("t4");

try

{

st=con.prepareStatement("insert into feedback values(?,?,?,?)");

st.setString(1,a);

st.setString(2,b);

st.setString(3,c);

st.setString(4,d);

st.execute();

}

catch(Exception at)

{}

res.sendRedirect("index.html");

out.println("</body>");

out.println("</html>");

}

}

**Register1 Coding:**

import java.sql.\*;

import java.io.\*;

import javax.servlet.\*;

import javax.servlet.http.\*;

public class register1 extends HttpServlet

{

Statement st=null;

Connection con=null;

ResultSet rs;

public void init()

{

System.out.println("init");

try

{

Class.forName("oracle.jdbc.driver.OracleDriver");

con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:XE","system","1234");

}

catch(Exception ae)

{

}

}

public void doGet(HttpServletRequest req, HttpServletResponse res)throws ServletException, IOException

{

res.setContentType("text/html");

PrintWriter out = res.getWriter();

out.println("<html>");

out.println("<head>");

out.println("<h1><center>Gym Parking Report </h1><hr>");

out.println("</head>");

out.println("<body bgcolor=yellow>");

out.println("<img src=page4\_img4.jpg border=0 height=200 width=820>");

out.println("<h3><center>Admin Report</h3><hr>");

out.println("<table border=2>");

out.println("<tr>");

out.println("<th>NAME</th><th>PHONE NO</th><th>VEHICLE NO</th><th>PLACE</th><th>TIME</th><th>DATE</th><th>WEEKS</th><th>DAYS</th><th>MESSAGE</th>");

out.println("</tr>");

try

{

st=con.createStatement();

rs=st.executeQuery("select \* from register");

while(rs.next())

{

out.println("<tr><td>");

out.println(rs.getString(1));

out.println("<td>");

out.println(rs.getString(2));

out.println("<td>");

out.println(rs.getString(3));

out.println("<td>");

out.println(rs.getString(4));

out.println("<td>");

out.println(rs.getString(5));

out.println("<td>");

out.println(rs.getString(6));

out.println("<td>");

out.println(rs.getString(8));

out.println("<td>");

out.println(rs.getString(9));

out.println("<td>");

out.println(rs.getString(10));

out.println("</tr>");

}

}

catch(Exception at)

{}

out.println("</body>");

out.println("</html>");

}

public void destroy()

{

System.out.println("this is destroy");

}}

**CHAPTER-9**

**Coding part(.css)**

**Style.css:**

@import "reset.css";

@import "grid.css";

@import "font-awesome.css";

@import "superfish.css";

@import url(//fonts.googleapis.com/css?family=Open+Sans:400);

@import url(//fonts.googleapis.com/css?family=Open+Sans:300);

@import url(//fonts.googleapis.com/css?family=Open+Sans:600);

@import url(//fonts.googleapis.com/css?family=Open+Sans:400italic);

@import url(//fonts.googleapis.com/css?family=Roboto+Condensed:400);

@import url(//fonts.googleapis.com/css?family=Roboto+Condensed:700);

@import url(//fonts.googleapis.com/css?family=Marvel:400);

@import url(//fonts.googleapis.com/css?family=Marvel:700);

html {

width: 100%;

}

a[href^="tel:"] {

color: inherit;

text-decoration: none;

}

\* {

-webkit-text-size-adjust: none;

}

body {

position: relative;

background-color:#4a4a4a;

color: #878787;

font: 14px/20px 'Open Sans', sans-serif;

}

.ic {

border:0;

float:right;

background:#fff;

color:#f00;

width:50%;

line-height:10px;

font-size:10px;

margin:-220% 0 0 0;

overflow:hidden;

padding:0

}

strong {

font-weight: 700;

}

address {

font-style: normal;

}

p {

margin-bottom: 20px;

}

input {

border-radius: 0 !important;

outline: none !important;

}

img {

max-width: 100%;

}

/\*\*\*\*Heads\*\*\*\*/

h1, h2, h3, h4, h5, h6 {

font-weight: normal;

color: #3e454c;

font-family: 'Roboto Condensed', sans-serif;

}

h3 {

font-size: 36px;

line-height: 36px;

color: #313030;

padding-top: 14px;

margin-bottom: 20px;

}

.page1 h3 {

margin-bottom: 46px;

padding-top: 82px;

}

.h3\_\_ind1 {

padding-top: 76px;

margin-bottom: 39px;

}

/\*\*\*\*Lists\*\*\*\*/

ul {

padding: 0;

margin: 0;

list-style: none;

}

.list {

}

.list li {

overflow: hidden;

}

.list\_count {

float: left;

text-align: center;

color: #fff;

width: 45px;

height: 45px;

margin-top: 4px;

margin-right: 13px;

background-color: #fdc903;

font: 36px/44px 'Roboto Condensed', sans-serif;

}

.list li .text1 {

margin-bottom: 3px;

}

.list li+li {

margin-top: 19px;

}

/\*\*\*\*Links\*\*\*\*/

a {

text-decoration: none;

color: inherit;

outline: none;

transition: 0.5s ease;

-o-transition: 0.5s ease;

-webkit-transition: 0.5s ease;

}

a:hover {

color: #fdc903;

}

.btn {

display: inline-block;

background-color: #fdc903;

color: #fff;

font-size: 14px;

line-height: 20px;

padding: 10px 26px 8px 25px;

}

.btn:hover {

background-color: #585757;

}

#bookingForm .btn {

margin-top: 13px !important;

}

/\*\*\*\*classes\*\*\*\*/

.mb0 {

margin-bottom: 0px !important;

}

.m0 {

margin: 0 !important;

}

.pad0 {

padding: 0 !important;

}

.pad1 {

}

.img\_inner {

max-width: 100%;

-moz-box-sizing: border-box;

-webkit-box-sizing: border-box;

-o-box-sizing: border-box;

box-sizing: border-box;

margin-bottom: 17px;

margin-top: 7px;

}

.fleft {

float: left;

width: auto !important;

margin-right: 32px;

margin-bottom: 0px;

margin-top: 4px;

}

.page1 .fleft {

margin-top: 5px;

}

.page1 .fleft {

margin-right: 16px;

}

.oh {

overflow: hidden;

}

.fright {

float: right !important;

}

.upp {

text-transform: uppercase;

}

.alright {

text-align: right;

}

.center {

text-align: center;

}

.wrapper, .extra\_wrapper {

overflow: hidden;

}

.clear {

float: none !important;

clear: both;

}

.nowrap {

white-space: nowrap;

}

/\*header\*/

.main {

background-color: #fff;

}

header {

display: block;

}

header h1 {

position: relative;

text-align: center;

padding-top: 55px;

}

.page1 header h1 {

margin-bottom: 20px;

}

header h1 a {

display: inline-block;

overflow: hidden;

width: 251px;

height: 122px;

font-size: 0;

line-height: 0;

text-indent: -999px;

transition: 0s ease;

-o-transition: 0s ease;

-webkit-transition: 0s ease;

}

header h1 a img {

display: block;

}

.banner {

font-family: 'Roboto Condensed', sans-serif;

background-color: #e5e5e7;

margin: 55px 0 87px;

overflow: hidden;

position: relative;

padding: 0 30px 24px 40px;

}

.banner\_title {

overflow: hidden;

padding-top: 19px;

color: #2a2929;

font-size: 46px;

line-height: 48px;

margin-bottom: 9px;

}

.banner img {

position: relative;

top: -1px;

float: left;

margin-right: 17px;

}

.banner a {

font-size: 30px;

float: right;

position: relative;

margin-top: -10px;

color: #959596;

}

.banner a:hover {

color: #6b6b6c;

}

.c\_phone {

font-family: 'Roboto Condensed', sans-serif;

position: relative;

padding: 62px 0 61px;

text-align: center;

font-size: 65px;

line-height: 24px;

text-transform: uppercase;

color: #fff;

background-color: #fdc903;

}

.c\_phone span {

padding-top: 30px;

padding-left: 45px;

font-size: 50px;

display: block;

}

.c\_phone .fa {

font-size: 48px;

line-height: 48px;

margin-right: 10px;

position: relative;

top: -9px;

}

.c\_phone>div>div {

position: relative;

}

.c\_phone>div>div:after {

position: absolute;

background: url(../images/tax\_bg.png) 0 0 no-repeat;

width: 183px;

height: 53px;

content: '';

left: 0;

top: 50%;

margin-top: -31px;

}

.c\_phone>div>div:before {

position: absolute;

background: url(../images/tax\_bg.png) 0 0 no-repeat;

width: 183px;

height: 53px;

content: '';

right: 0;

top: 50%;

margin-top: -31px;

}

/\*\*Content\*\*/

.content {

padding-bottom: 113px;

}

.page1 .content {

padding-bottom: 82px;

}

.type {

border: 1px solid #ebeaea;

position: relative;

margin-top: 84px;

display: block;

font-family: 'Roboto Condensed', sans-serif;

}

.type\_caption {

position: absolute;

left: 0;

bottom: 0;

display: block;

width: 312px;

-moz-box-sizing: border-box;

-webkit-box-sizing: border-box;

-o-box-sizing: border-box;

box-sizing: border-box;

background: url(../images/capt\_bg.png) 0 0 repeat;

color: #fff;

font-size: bold;

font-size: 30px;

line-height: 30px;

padding: 7px 30px 8px;

transition: 0.5s ease;

-o-transition: 0.5s ease;

-webkit-transition: 0.5s ease;

}

.type:hover .type\_caption {

width: 100%;

}

.type+.type {

margin-top: 11px;

}

.type+.type+.type {

margin-top: 10px;

}

.text1 {

margin-bottom: 23px;

font-size: 20px;

line-height: 20px;

}

.cl1 {

height: 24px;

}

.bq1 {

padding: 61px 38px 26px 46px;

background: url(../images/quotes.png) 31px 33px no-repeat #e5e5e7;

}

.bq1 .color2 {

color: #727171;

text-align: right;

font: 18px/20px 'Roboto Condensed', sans-serif;

}

.bq1:after {

content: '';

display: block;

position: absolute;

width: 0px;

height: 0px;

bottom: -27px;

left: 67px;

border-style: solid;

border-width: 27px 0 0 40px;

border-color: #cfcfd0 transparent transparent transparent;

}

.gallery {

overflow: hidden;

padding-top: 27px;

}

a.gal {

background: url(../images/magnifier.png) center center no-repeat #fdc903;

margin-right: 1px;

margin-bottom: 27px;

display: block;

border: 1px solid #ebeaea;

border-right-width: 2px;

}

a.gal img {

width: 100%;

transition: 0.5s ease;

-o-transition: 0.5s ease;

-webkit-transition: 0.5s ease;

box-shadow: 0 0 0 #fff;

}

a.gal:hover img {

opacity: 0.2;

}

.box {

margin-bottom: 23px;

margin-top: 4px;

background-color: #e5e5e7;

padding: 23px 22px 28px;

}

.box img {

margin-bottom: 25px;

}

.box a.btn {

padding-left: 13px;

padding-right: 13px;

margin-top: 23px;

}

/\*\*Map\*\*/

.color1 {

color: #fdc903;

}

.color1 a:hover , a.color1:hover {

color: #4a4a4a;

}

.color2 {

color: #4a4a4a;

}

.map {

overflow: hidden;

position: relative;

}

.map figure {

padding: 27px 0 10px;

position: relative;

display: block;

width: 100%;

-moz-box-sizing: border-box;

-webkit-box-sizing: border-box;

-o-box-sizing: border-box;

box-sizing: border-box;

}

.map figure iframe {

width: 100%;

height: 399px;

max-width: 100%;

}

.map address {

display: block;

}

.map .text1 {

padding-top: 8px;

}

.map address dt {

}

address dd span {

display: inline-block;

text-align: left;

}

/\*\*Footer\*\*/

footer {

display: block;

padding: 0px 0 60px;

font-size: 14px;

color: #aaa;

line-height: 18px;

font-family: 'Roboto Condensed', sans-serif;

}

.copy {

font-family: 'Open Sans', sans-serif;

}

.st1 {

padding-top: 80px;

color: #e2e2e2;

margin-bottom: 8px;

}

.brand {

position: relative;

top: -7px;

font: 30px/18px 'Roboto Condensed', sans-serif;

display: inline-block;

font-size: 30px;

}

.f\_phone {

padding-top: 72px;

font-size: 40px;

line-height: 36px;

float: right;

color: #e2e2e2;

}

.f\_phone span {

position: relative;

top: 4px;

font-size: 30px;

}

.socials {

margin-right: 96px;

padding-top: 82px;

float: right;

overflow: hidden;

}

.socials a {

display: block;

float: left;

margin: 0 6px 0 7px;

color: #fff;

text-align: center;

background-color: #fdc903;

border-radius: 500px;

width: 58px;

height: 58px;

font-size: 24px;

line-height: 58px;

}

.socials a:hover {

color: #fdc903;

background-color: #585757;

}

#toTop {

display: none;

text-decoration: none;

position: fixed;

bottom: 40px;

left: 51%;

margin-left: 610px;

overflow: hidden;

width: 61px;

height: 75px;

border: none;

text-indent: -999px;

z-index: 20;

background: url(../images/totop.png) no-repeat left 0;

transition: 0s ease;

-o-transition: 0s ease;

-webkit-transition: 0s ease;

}

#toTop:hover {

outline: none;

background-position: right 0;

}

/\*==================================RESPONSIVE LAYOUTS===============================================\*/

@media only screen and (max-width: 995px) {

body {

min-width: 768px;

}

.fl1 {

float: none;

margin-left: 0 !important;

width: auto;

}

.map figure, .map figure iframe, #form input, #form textarea, #form .success {

width: 100% !important;

float: none !important;

}

#form .success {

-moz-box-sizing: border-box;

-webkit-box-sizing: border-box;

-o-box-sizing: border-box;

box-sizing: border-box;

}

.map figure {

height: auto !important;

}

.extra\_wrapper {

overflow: visible;

}

.map figure {

margin-bottom: 15px;

}

.nowrap {

white-space: normal;

}

.caption .price {

display: none;

}

.caption {

padding-top: 20px;

}

.camera\_prev {

margin-left: -374px;

}

.camera\_next {

margin-right: -374px;

}

.fl1.fl2 {

clear: both;

}

}

@media only screen and (max-width: 767px) {

body {

min-width: 420px;

}

header {

-webkit-background-size: auto auto !important;

-moz-background-size: auto auto !important;

-ms-background-size: auto auto !important;

-o-background-size: auto auto !important;

background-size: auto auto !important;

}

#bookingForm {

margin-right: 0;

}

.tmRadio p {

padding-left: 0;

padding-top: 10px;

}

.tmRadio {

margin-left: 0 !important;

}

.tmRadio strong {

margin-left: 0;

}

#bookingForm > strong + .controlHolder {

width: 280px;

}

.camera\_prev {

margin-left: 0;

left: 20px;

bottom: 60px;

top: auto;

margin-top: 0;

}

.camera\_next {

margin-right: 0;

right: 20px;

bottom: 60px;

top: auto;

margin-top: 0;

}

.banner img {

width: 100%;

}

.banner {

margin-bottom: 25px;

}

.block1 {

margin-bottom: 30px;

}

.map address +address {

margin-left: 0;

float: none;

margin-top: 20px;

}

.socials {

float: none;

overflow: hidden;

}

.copy {

float: none;

padding-top: 40px;

}

header h1 {

display: block;

width: auto;

position: relative;

height: auto;

margin-bottom: 20px;

}

.img\_inner {

width: 100% !important;

float: none !important;

margin-right: 0 !important;

margin-bottom: 20px !important;

}

h1 a {

max-width: 90%;

}

.img\_inner img {

width: 100%;

}

header h1 {

float: none;

text-align: center;

}

header h1 a {

width: auto;

display: inline-block;

height: auto;

}

.map {

padding-right: 0px !important;

}

.map figure iframe {

width: 100%;

height: 300px;

}

.map address {

margin-right: 0;

}

.noresize {

width: auto !important;

float: left !important;

margin-right: 20px !important;

margin-top: 4px !important;

}

.img\_inner {

margin-top: 30px;

}

.map address {

float: none;

}

.ui-datepicker {

width: 280px;

}

}

@media only screen and (max-width: 479px) {

body {

min-width: 300px;

}

#bookingForm > strong + .controlHolder {

width: 210px;

}

.ui-datepicker {

left: 50% !important;

margin-left: -140px;

}

}

**CHAPTER-10**

**CODING PART (.JS)**

**jquery.equalheights:**

/\*parsed HTML\*/

$(function(){

$(".maxheight").each(function(){

$(this).contents().wrapAll("<div class='box\_inner'></div>");

})

$(".maxheight1").each(function(){

$(this).contents().wrapAll("<div class='box\_inner'></div>");

})

$(".maxheight2").each(function(){

$(this).contents().wrapAll("<div class='box\_inner'></div>");

})

})

/\*add event\*/

$(window).bind("resize", height\_handler).bind("load", height\_handler)

function height\_handler(){

if($(window).width()>767){

$(".maxheight").equalHeights();

}else{

$(".maxheight").css({'height':'auto'});

}

if($(window).width()>767){

$(".maxheight1").equalHeights();

}else{

$(".maxheight1").css({'height':'auto'});

}

if($(window).width()>767){

$(".maxheight2").equalHeights();

}else{

$(".maxheight2").css({'height':'auto'});

}

}

/\*glob function\*/

(function($){

$.fn.equalHeights=function(minHeight,maxHeight){

tallest=(minHeight)?minHeight:0;

this.each(function(){

if($(">.box\_inner", this).outerHeight()>tallest){

tallest=$(">.box\_inner", this).outerHeight()

}

});

if((maxHeight)&&tallest>maxHeight) tallest=maxHeight;

return this.each(function(){$(this).height(tallest)})

}

})(jQuery)

**Script:**

$(function(){

// IPad/IPhone

var viewportmeta = document.querySelector && document.querySelector('meta[name="viewport"]'),

ua = navigator.userAgent,

gestureStart = function () {viewportmeta.content = "width=device-width, minimum-scale=0.25, maximum-scale=1.6";},

scaleFix = function () {

if (viewportmeta && /iPhone|iPad/.test(ua) && !/Opera Mini/.test(ua)) {

viewportmeta.content = "width=device-width, minimum-scale=1.0, maximum-scale=1.0";

document.addEventListener("gesturestart", gestureStart, false);

}

};

scaleFix();

// Menu Android

if(window.orientation!=undefined){

var regM = /ipod|ipad|iphone/gi,

result = ua.match(regM)

if(!result) {

$('.sf-menu li').each(function(){

if($(">ul", this)[0]){

$(">a", this).toggle(

function(){

return false;

},

function(){

window.location.href = $(this).attr("href");

}

);

}

})

}

}

});

var ua=navigator.userAgent.toLocaleLowerCase(),

regV = /ipod|ipad|iphone/gi,

result = ua.match(regV),

userScale="";

if(!result){

userScale=",user-scalable=0"

}

document.write('<meta name="viewport" content="width=device-width,initial-scale=1.0'+userScale+'">')

var currentYear = (new Date).getFullYear();

$(document).ready(function() {

$("#copyright-year").text( (new Date).getFullYear() );

});

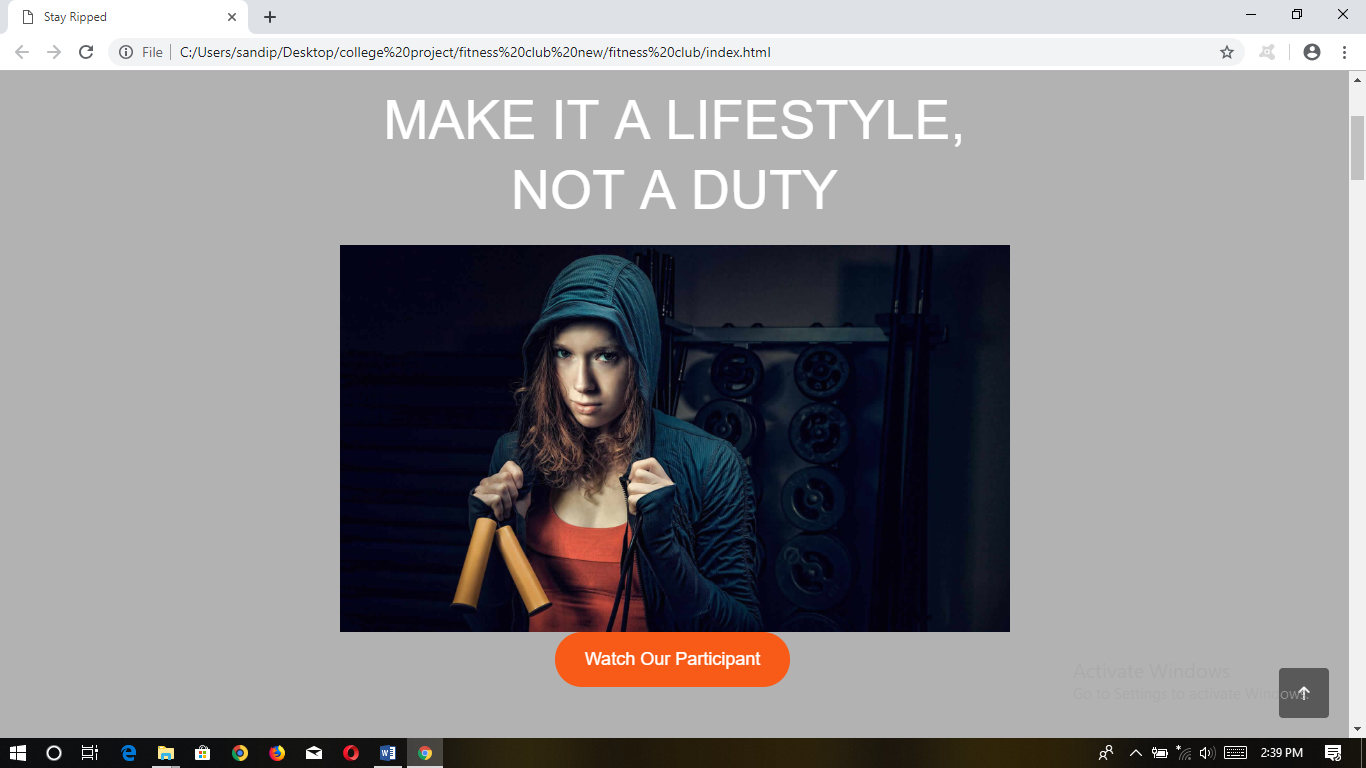
$(function(){

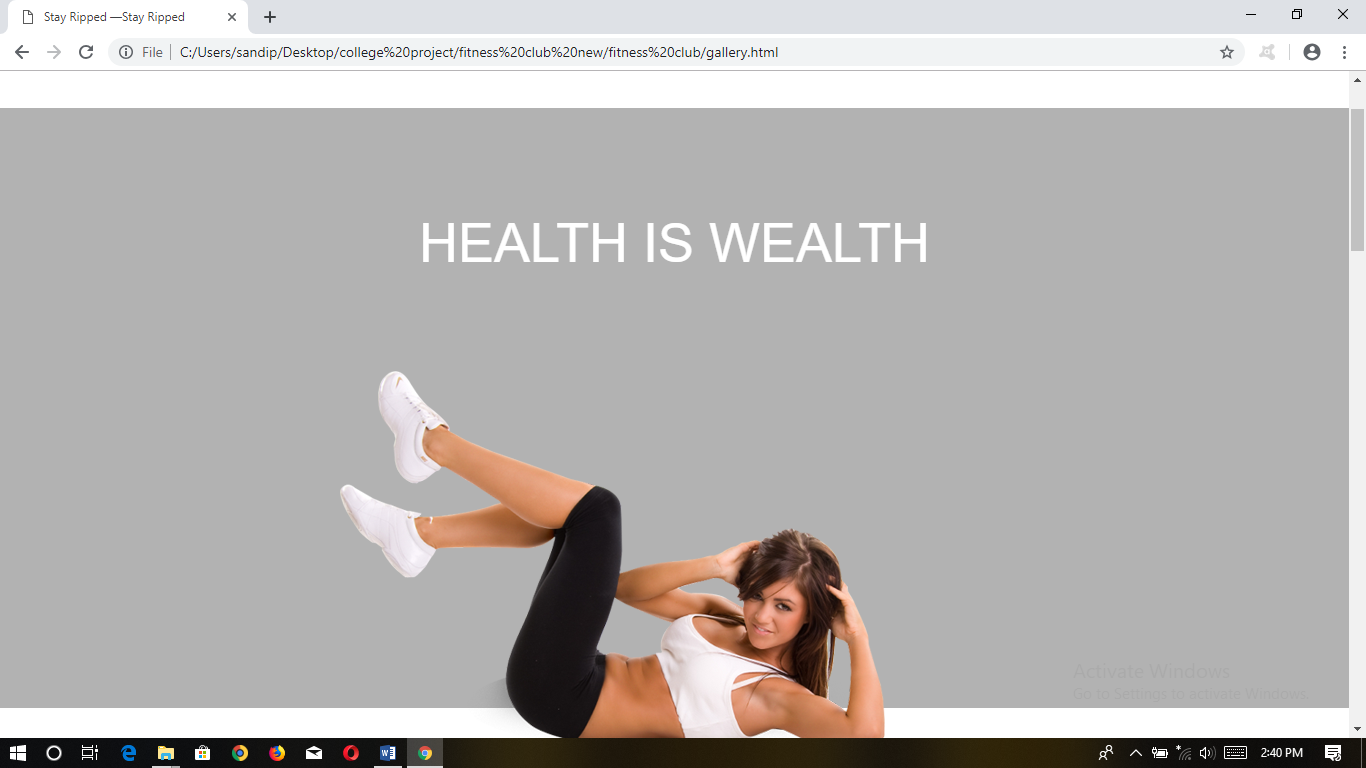
$('.sf-menu').superfish({autoArrows: true})

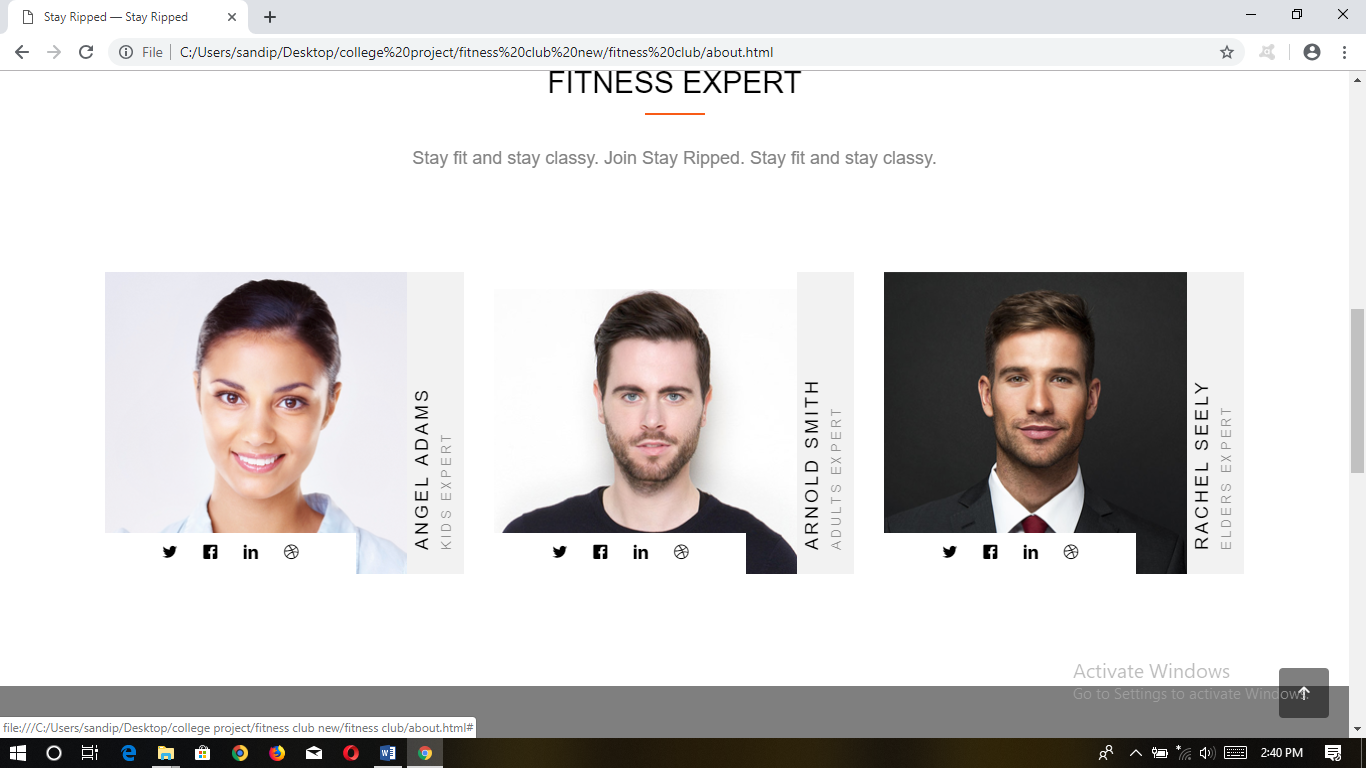
})

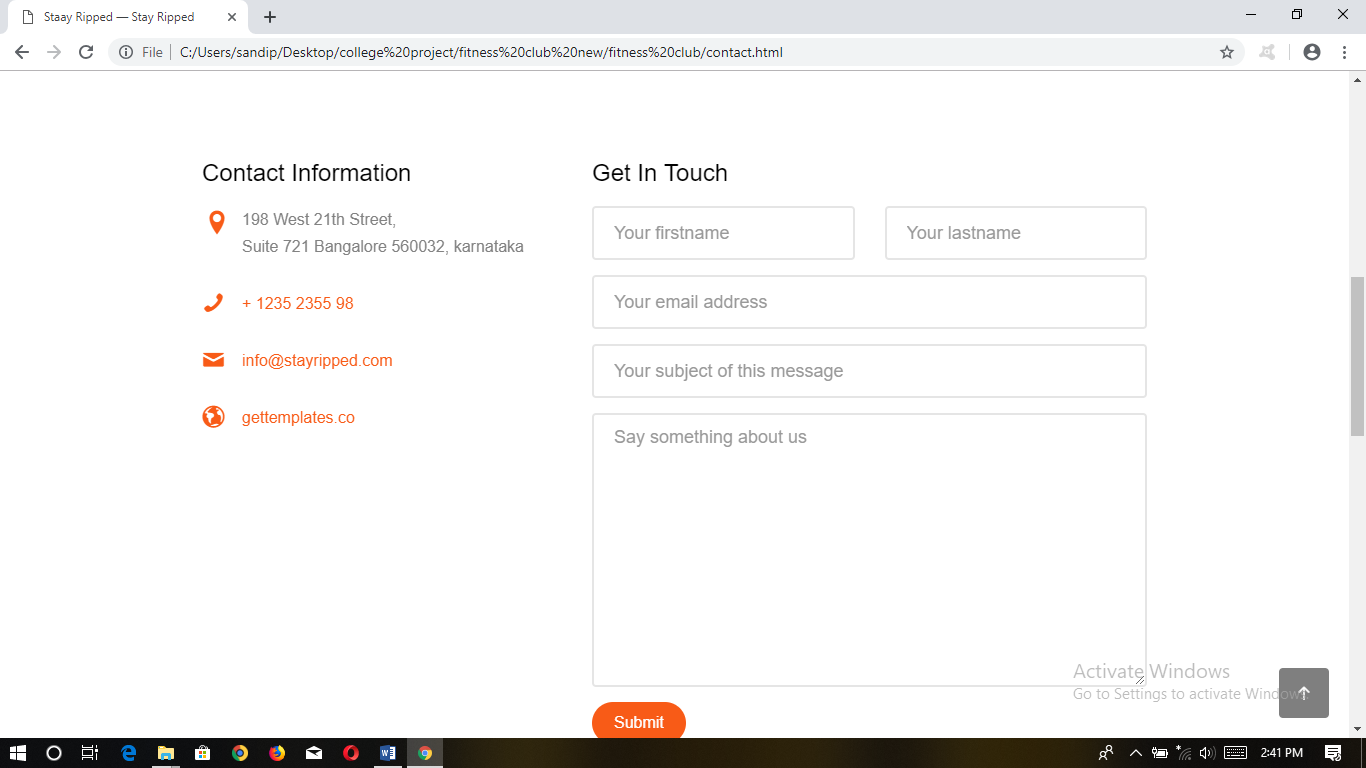
**CHAPTER-11**

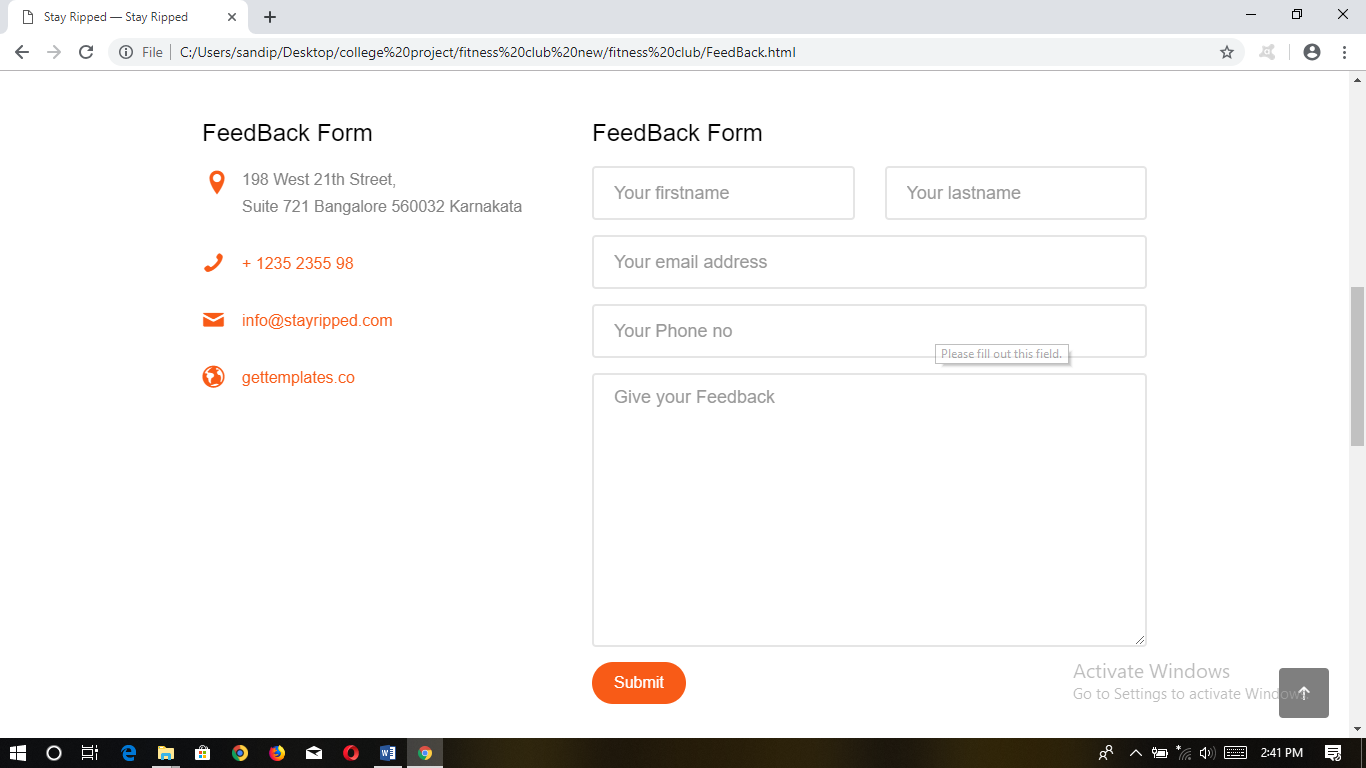
**OUTPUTS SCREENShots**











**CHAPTER-12**

**SYSTEM TESTING AND IMPLEMENTATION**

**12.1. INTRODUCTION**

Software testing is a critical element of software quality assurance and represents the ultimate review of specification, design and coding. In fact, testing is the one step in the software engineering process that could be viewed as destructive rather than constructive.

A strategy for software testing integrates software test case design methods into a well-planned series of steps that result in the successful construction of software. Testing is the set of activities that can be planned in advance and conducted systematically. The underlying motivation of program testing is to affirm software quality with methods that can economically and effectively apply to both strategic to both large and small-scale systems.

**12.2. SOFTWARE TESTING**

The software engineering process can be viewed as a spiral. Initially system engineering defines the role of software and leads to software requirement analysis where the information domain, functions, behaviour, performance, constraints and validation criteria for software are established. Moving inward along the spiral, we come to design and finally to coding. To develop computer software we spiral in along streamlines that decrease the level of abstraction on each turn.

A strategy for software testing may also be viewed in the context of the spiral. Unit testing begins at the vertex of the spiral and concentrates on each unit of the software as implemented in source code. Testing progress by moving outward along the spiral to integration testing, where the focus is on the design and the construction of the software architecture. Talking another turn on outward on the spiral we encounter validation testing where requirements established as part of software requirements analysis are validated against the software that has been constructed. Finally we arrive at system testing, where the software and other system elements are tested as a whole.

**Unit testing:**

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

**Integration testing:**

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

**Functional test:**

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted.

Invalid Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised.

Systems/Procedures : interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

**System Test**:

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

**White Box Testing:**

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

**Black Box Testing:**

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

**Unit Testing:**

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

**Test strategy and approach**

Field testing will be performed manually and functional tests will be written in detail.

**Test objectives**

* All field entries must work properly.
* Pages must be activated from the identified link.
* The entry screen, messages and responses must not be delayed.

**Features to be tested**

* Verify that the entries are of the correct format
* No duplicate entries should be allowed
* All links should take the user to the correct page.

# Integration Testing:

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

**Test Results:**

All the test cases mentioned above passed successfully. No defects encountered.

**Acceptance Testing:**

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

**Test Results:**

All the test cases mentioned above passed successfully. No defects encountered.

UNIT TESTING

MODULE TESTING

SUB-SYSTEM TESING

SYSTEM TESTING

ACCEPTANCE TESTING

Component Testing

Integration Testing

User Testing

**Unit Testing**

Unit testing focuses verification effort on the smallest unit of software design, the module. The unit testing we have is white box oriented and some modules the steps are conducted in parallel.

**1. WHITE BOX TESTING**

This type of testing ensures that

* All independent paths have been exercised at least once
* All logical decisions have been exercised on their true and false sides
* All loops are executed at their boundaries and within their operational bounds
* All internal data structures have been exercised to assure their validity.

**2. BASIC PATH TESTING**

Established technique of flow graph with Cyclomatic complexity was used to derive test cases for all the functions. The main steps in deriving test cases were:

Use the design of the code and draw correspondent flow graph.

Determine the Cyclomatic complexity of resultant flow graph, using formula:

V (G) =E-N+2 or

V (G) =P+1 or

V (G) =Number of Regions

Where V (G) is Cyclomatic complexity,

E is the number of edges,

N is the number of flow graph nodes,

P is the number of predicate nodes.

Determine the basis of set of linearly independent paths.

**3. CONDITIONAL TESTING**

In this part of the testing each of the conditions were tested to both true and false aspects. And all the resulting paths were tested. So that each path that may be generate on particular condition is traced to uncover any possible errors.

**4. DATA FLOW TESTING**

This type of testing selects the path of the program according to the location of definition and use of variables. This kind of testing was used only when some local variable were declared. The definition-use chain method was used in this type of testing. These were particularly useful in nested statements.

1. **LOOP TESTING**

In this type of testing all the loops are tested to all the limits possible. The following exercise was tested for all loops:

* All the loops were tested at their limits, just above them and just below them.
* All the loops were skipped at least once.
* For nested loops test the inner most loop first and then work outwards.
* For concatenated loops the values of dependent loops were set with the help of connected loop.
* Unstructured loops were resolved into nested loops or concatenated loops and tested as above.

Each unit has been separately tested by the development team itself and all the input have been validated.

**CHAPTER-13**

**SYSTEM SECURITY**

**13.1. INTRODUCTION**

The protection of computer based resources that includes hardware, software, data procedures and people against unauthorized use or natural

Disaster is known as System Security.

System Security can be divided into four related issues

* Security
* Integrity
* Privacy
* Confidentiality

**System Security** refers to the technical innovations and procedures applied to the hardware and operation systems to protect against deliberate or accidental damage from a defined threat.

**Data Security** is the protection of data from loss, disclosure, modification and destruction.

**System integrity** refers to the power functioning of hardware and programs, appropriate physical security and safety against external threats such as eavesdropping and wiretapping.

**Privacy** defines the rights of the user or organizations to determine what information they are willing to share with or accept from others and how the organization can be protected against unwelcome, unfair or excessive dissemination of information about it.

**Confidentiality** is a special status given to sensitive information in a database to minimize the possible invasion of privacy. It is an attribute of information that characterizes its need for protection.

**13.2** **SECURITY IN SOFTWARE**

System Security refers to various validations on data in form of checks and controls to avoid the system from failing. It is always important to ensure that only valid data is entered and only valid operations are performed on the system. The system employees have two types of checks controls:

**CLIENT SIDE VALIDATION**

Various client side validation are used to ensure on the client side that only valid data is entered. Client side validation saves server time and load to handle invalid data. Some checks imposed are:

* JavaScript is used to ensure those required fields are filled with suitable data only.

Maximum lengths of the fields of the forms are appropriately defined.

* Forms cannot be submitted without filling up the mandatory data so that manual mistakes of submitting empty fields that are mandatory can be stored out at the client side to save the server time and load.
* Tab-indexes are set according to the need and taking into account the ease of user while working with the system

**SERVER SIDE VALIDATION**

Some checks cannot be applied at client side. Server side checks are necessary to save the system from failing or intimating the user that some invalid operations has been performed or the performed operation is restricted. Some of the server side checks imposed is:

* Server constraint has been imposed to check for the validity of primary key and foreign key. A primary key value cannot be duplicated. Any attempt to duplicate the primary value results in a message intimating the user about those values through the forms using foreign key can be updated only by the existing foreign key values.
* User is intimating through appropriate messages about the successful operations or exception occurring at server side.
* Various Access Control Mechanisms have been built so that one user may not agitate on another. Access permissions to various types of users are controlled according to the organizational structure. Only permitted users can log on to the system and can have access according to their category. User-name, passwords and permissions are controlled by the server side.
* Using server side validation, constraints on several restricted operations are imposed.

**CHAPTER-14**

**CONCLUSION**

gym business has emerged with a new goodies compared to the past experience where every activity concerning gym rental business is limited to a physical location only.

Even though the physical location has not been totally eradicated; the nature of functions and how these functions are achieved has been reshaped by the power of internet.

Nowadays, customers can reserve gyms online, rent gym online, and have the gym brought to their door step once the customer is a registered member or go to the office to pick the gym.

The web based gym rental system has offered an advantage to both customers as well as Gym Rental Company to efficiently and effectively manage the business and satisfies customers need at the click of a button.

**CHAPTER-15**

**FUTURE SCOPE**

This project traverses a lot of areas ranging from business concept to computing field, and required to perform several researches to be able to achieve the project objectives.

The area covers include:

•gym rental industry: This includes study on how the gym rental business is being done, process involved and opportunity that exist for improvement.

•PHP Technology used for the development of the application.

•General customers as well as the company’s staff will be able to use the system effectively.

•Web-platform means that the system will be available for access 24/7 except when there is a temporary server issue which is expected to be minimal.

**CHAPTER-16**

**BIBLIOGRAPHY**

* **FOR JAVA INSTALLATION**

[www.support.oracle.com](http://www.support.oracle.com)

* **FOR DEPLOYMENT AND PACKING ON SERVER**

[www.developer.com](http://www.developer.com)

[www.15seconds.com](http://www.15seconds.com)

* **FOR ORACLE**

[www.msdn.oracle.com](http://www.msdn.oracle.com)

* **FOR JAVA**

[www.msdn.microsoft.com/net/quickstart/aspplus/default.com](http://www.msdn.microsoft.com/net/quickstart/aspplus/default.com)

[www.JAVA](http://www.asp.net)

[www.fmexpense.com/quickstart/aspplus/default.com](http://www.fmexpense.com/quickstart/aspplus/default.com)

[www.asptoday.com](http://www.asptoday.com)

[www.aspfree.com](http://www.aspfree.com)

[www.4guysfromrolla.com/index.aspx](http://www.4guysfromrolla.com/index.aspx)

**INFORMATION LINK USED IN PROJECT:**

* [**http://www.academia.edu/16644874/115442368-49930505-Gym-Rental-System-Project-Report**](http://www.academia.edu/16644874/115442368-49930505-Car-Rental-System-Project-Report)
* **https://www.freeprojectz.com/project-synopsis/synopsis-gym-rental-system/introduction-gym-rental-system**
* **http://www.academia.edu/16644874/115442368-49930505-Gym-Rental-System-Project-Report**