SYSTEM DESIGN

**4.1 INTRODUCTION**

The purpose of the design phase is to find a solution to the problem specified in the requirement document. This is the first step in moving from problem domain to solution domain.

The design of a system is perhaps the most critical factor affecting the quality of the software. It has a major impact on the later phases is the design document. This document is similar to a blueprint or a plan for the solution and is used later during implementation testing and maintenance.

The design activity is often divided into separate phases - system design and detailed design. System design is sometimes also called as top level design. This system design aims to identify the modules that should be present inside the system, specifications of these modules, and how they interact with each other to produce the desired output. At the end of the system design all the major data structures, file formats, and the major modules in the system and their specification are decided.

**4.2 OVERVIEW**

This website is used for arranging online event for various colleges which come under Mangalore University. The events are scheduled by the Mangalore University and the various colleges which are taking part in the event should register online using this website.

The primary purpose of the system is to implement the above stated functionalities. It will have user friendly GUI's that will guide the user to easily achieve the same. The application shall also have some report displaying in the same window.

In addition to this application features to generate different kinds of reports. This application is fully developed under SQL server using ASP.Net.

**4.3 SCOPE**

This document will provide detailed specification for designing, implementing and configuring software for **Project Name**.

**4.4 APPLICABLE DOCUMENTS**

As the project is user-friendly, it can be applied to large database with more information. This software can be used by banks to make the work simple .They can get information quickly as possible. It can handle large volume of data and present the report whenever required.

**4.5 ASSUMPTIONS AND CONSTRAINTS**

**4.5.1 ASSUMPTIONS**

* It is assumed that the college admin and students will have enough training to work with the system.
* It is assumed that enough computers are available to install the system and also they are compatible with the system.
* It is assumed that all the information that fed into the system is up-to-date.
* It is assumed that browsers and operating system will support .NET Framework 4.0

**4.6 FUNCTIONAL DECOMPOSITION**

Functional decomposition refers broadly the process of resolving a functional relationship into its constituent parts in such a way that the original function can be reconstructed from those parts by function composition. In general, this process decomposition is undertaken either for the purpose of gaining insight into the industry of the constituent components or for the purpose of obtaining a compressed representation of the global function, a task which is feasible, and a task which is feasible when the constituent process has a certain level of modularity.

**4.6.1 System Software Architecture**

**Web server (IIS)**

**SQL server**

**ASP.NET**

**Local host**

**SQL query**

**Database**

Send

Receive

Retrieve

Add

The software has following software, database and other requirements.

Operating system: Windows XP or higher.

Frontend: ASP.Net.

Backend: SQL Server 2008.

**4.6.2 SYSTEM TECHNICAL ARCHITECTURE.**

Sql Server

IIS Server

Inter College Event

Database Administrator

Database Server

Database Compiler

**4.6.3 SYSTEM HARDWARE ARCHITECTURE:**



Admin Server

Local Area Network

Users

Database Server

The system should have these hardware requirements.

Processor: Intel dual core or above

Processor Speed: 1.0GHZ or above

RAM: 1 GB RAM or above

Hard disk: 20 GB hard disk or above

Operating system: Windows XP or above

Monitor: VGA or higher resolution.

**4.7 DESCRIPTION OF THE COMPONENTS**

The notations used in the following diagram are:

**DATAFLOW DIAGRAM NOTATION:**

|  |  |
| --- | --- |
|  | **Function:** The bubble represent a process or transformation that is applied to the data which changes in some way. Each bubble is a staff of the information. |
|  | **File:**It represents the repository data that is to be stored for used by one or more process. |
|  | **Input/output:** This rectangle is called an entity, which represent the staff of the information. |
|  | **Flow:** The arrow represents the dataflow. The arrow indicates the direction of the dataflow. All arrows in the dataflow diagram are labeled. |

**4.8 CONTEXT FLOW DIAGRAM (CFD)**

A context flow diagram (CFD) is a general representation of the "flow" of context through an information system. CFD's can also used for the visualization of data processing (Structured Design).

On a CFD data item can flow from an external data source or internal data store to an internal data store, or to an external data sink via an internal process.

A CFD provides no information about the training or ordering of processing, or about whether processes will operate in sequence or in parallel. It is therefore quite different from flowchart, which shows flow of control through an algorithm.

Allowing reader to determine what operations will be performed, in what order, and under what circumstances, but not what kinds of data will be input to and from the system, or where the data will come from and go to, or where the data will be stored.

// do the design - CFD

**4.9 DATA FLOW DIAGRAMS**

Data flow diagrams are a graphical representation of the “flow“ of data through an information system. Data flow diagrams are commonly used during problem analysis. It views a system as a function that transforms a the input into desired output. A DFD shows movement of data through the different transformation or process in the system, with the help of various levels in a crystal clear way.

A DFD representations flow of data through a system. Data flow diagrams are commonly used during problem analysis. It views a system as a function that transforms the input into desired output. A DFD shows movement of data through the different transformation or processes in the system.

There are four kinds of system components:

* Processes
* External entities
* Data flows
* Data stores

**DFD Level 0:// do the design**

**DFD Level 1:**

**DFD Level 2:**

**DFD Level 3:**

**DFD Level 4:**

**4.10 DESCRIPTION OF COMPONENTS**

**FUNCTIONAL COMPONENT 1: LOGIN MODULE:**

**Introduction:** This module allows valid admin or student or college administrator to access the functionalities provided by the website.

**Input:** student or admin or college admin can logon to the website by entering usernameand password.

**Output:** Login successful and student or admin or college admin is allowed to log in to the website. Otherwise it will display an error message.

**FUNCTIONAL COMPONENT 2: COLLEGE MODULE:**

**Introduction:** This module allows college to register themselves into the website and then get approved by admin.

**Input:** college name, address, contact number, email id, doc proof etc.

**Output:** College information will be added in the database. Otherwise it will display an error message.

**FUNCTIONAL COMPONENT 3: STUDENT MODULE:**

**Introduction:** This module allows Colleges register all the students of their college.

**Input:** Student name, address, reg. number, contact no, etc.

**Output:** student information will be added into database.

**FUNCTIONAL COMPONENT 4: EVENT MODULE:**

**Introduction:** This module allows admin to post all event information.

**Input:** event name, date, event detail, etc.

**Output:** Event information will be updated into respective table.

**FUNCTIONAL COMPONENT 5: EVENT SCHEDULE MODULE:**

**Introduction:** Event details will be updated by respective college admin.

**Input:** event date, location, description etc.

**Output:** Event detail will be added to the database.

**FUNCTIONAL COMPONENT 6: ELECTION MODULE:**

**Introduction:**  inert college Student council election details will be updated using this module by university admin.

**Input:** post detail, election date, result date etc.

**Output:** election details will be added to the database.

**FUNCTIONAL COMPONENT 7: CANDIDATE MODULE:**

**Introduction:** In this module college administrator will Nominate candidate details for election.

**Input:** candidatename, reg. no, contact number, photo, id proof, postid

**Output:** Candidate details will be added to the database.

**FUNCTIONAL COMPONENT 8: CHANGE PASSWORD:**

**Introduction:** Change password menu will help student/admin/college admin in changing his password.

**Input:** Enter the Current Password, new password, confirm password.

**Output:** Password will be changed.

**FUNCTIONAL COMPONENT 9: UPDATE PROFILE:**

**Introduction:** Student/ College Admin can update his/her profile using this module.

**Input:** name, contact number, photo etc.

**Output:** Student/ College Admin Profile will be updated

**FUNCTIONAL COMPONENT 10: REQUEST FUND MODULE:**

**Introduction:** If any student needs Fund to participate in any inter-college events, they can request for the fund. The students of any colleges who are willing to contribute funds can pay online using their registration no..

**Input:** student registration number, reason for fund request, amount, etc.

**Output:** Details will be updated into database.

**FUNCTIONAL COMPONENT 11: COURSE MODULE:**

**Introduction:** University headorAdministrator will update course detail using this module

**Input:** course name.

**Output:** course detail will be updated into database.

**FUNCTIONAL COMPONENT 12: LOGOUT MODULE:**

**Introduction:** This module allows the user to logout from their account.

**Input:** Click logout button.

**Output:** User will be logged out.

**4.9 ENTITY RELATIONSHIP MODELS (E-R DIAGRAM)**

E-R model is a general data model, as is the relational model that underlies SQL.It provides another way of thinking about and organizing data. The E-R model is used to derive an abstract model of the data i.e. the implemented with a set of tables that conform to relational principles. E-R diagram enables to represent the conceptual design of a system’s database.

**ELEMENTS OF ER-MODEL:**

**Entities**

The first basic concept in E-R diagram is the entity. They are the items that have a definable existence in the world-Persons, Jobs, Salaries etc. They are logical categories of items rather than specific items.

Entities can be further characterized as strong or weak. A strong entity is a well defined without reference to any other entity in the model, where as a weak one requires reference to something else in order for its individual instances to be meaningful or to be identifiable.

**Relationships**

Entities have relationships to one another. Relationships have a couple of important features. One of these is the degree of the relationship. This refers to the number of entities that participate in the relationship.

Another important characteristic of relationships is their connectivity, which refers to the mapping of instances of entity A to entity B.

There are 3 possibilities:

* One -to –one (written 1:1).For each A there is no more than one B.
* One –to-many (written 1: n).For each A, there are zero or more Bs, but for each B, there is only one A.
* Many –to-many (written m: n).For each A,m there are zero or more Bs, and for each B, there are zero or more As.

**Attributes**

These can apply to either entities or relationships. An attribute is a traceable characteristic of the entity or relationship. An attribute can be single- valued, meaning it has one value per instance of the entity or multi-valued, meaning it can have several such values.

**SYMBOLS USED IN ER-DIAGRAM**

|  |  |
| --- | --- |
| **Symbol** | **Meaning** |
|  | ENTITY |
|  | RELATIONSHIP |
|  | ATTRIBUTE |
|  | KEY ATTRIBUTE |
|  | CARDINALITY  RATIO 1:N FOR E1:E2  IN R |
|  | CARDINALITY  RATIO M:N FOR E1:E2  IN R |

**ER-DIAGRAM**

//Do the design