Software Design Specification

for

VIT Team Registration Portal

Version 1.0 Approved

Prepared by

Manav Chawla (14BCE0122)
Anirudh Batra (14BCE0602)
Manthan Kapoor Jain (14BCE0894)

Vellore Institute of Technology

8th February 2017

Table of Contents

1. Introduction

1.1 Purpose

This document will outline in detail the software architecture and design for the VIT Team Registration Portal (VTRP). This document will provide several views of the system's design in order to facilitate communication and understanding of the system. It intends to capture and convey the significant architectural and design decisions that have been made for the VTRP.

1.2 Scope

This document provides the architecture and design of Release 1.0 of the VTRP. It will show how the design will accomplish the functional and non-functional requirements detailed in the project's Software Requirements Specification (SRS) document.

1.3 Intended Audience

The document is created for:

- The instructors of the course 'Software Engineering' for their review and monitoring progress of the project.
- The software development team for their use in analyzing the requirements.

1.4 References

- VDK-RIT Software Requirements Specification (SRS) Revision 1.2
- VDK-RIT Use Case Document (UCD) Revision 1.1
- VDK-RIT Vision and Scope Document Revision 1.0

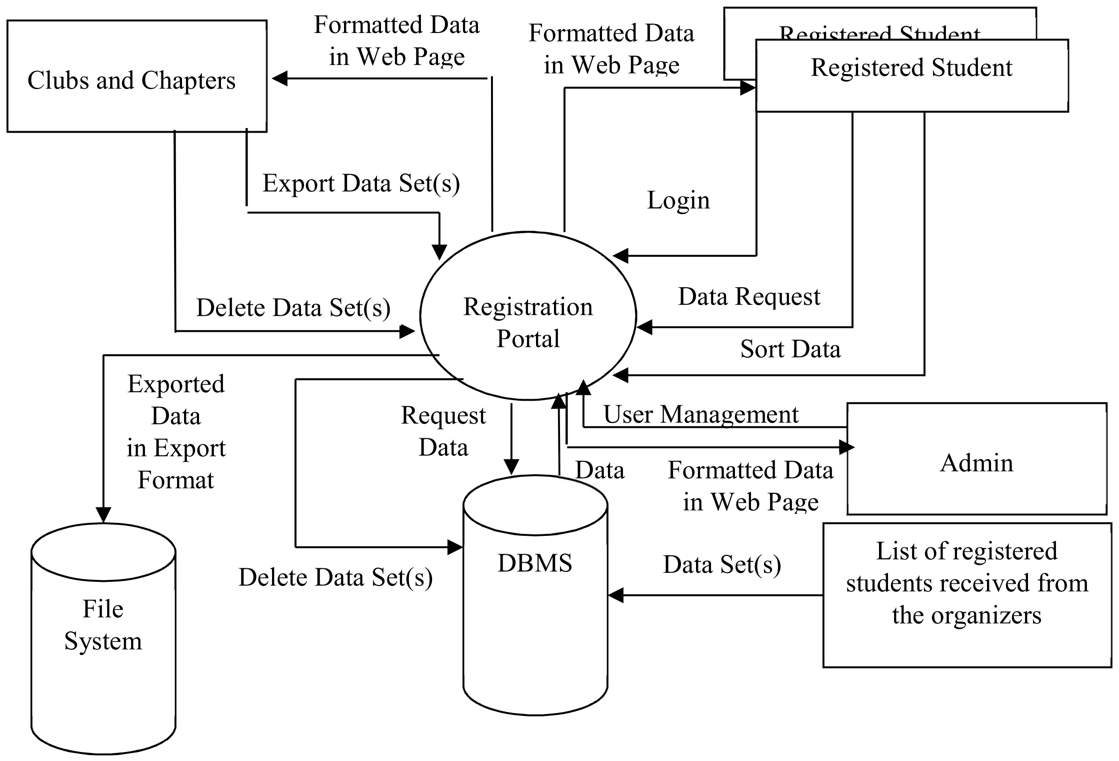
1.5 Definitions, Acronyms, and Abbreviations

- DBMS – Database Management System. A programmable interface which provides a common layer of abstraction between a physical database and a user or external program.
- HTML – Hypertext Markup Language. Set of markup symbols or codes intended for display on a World Wide Web browser page. The markup instructs a web browser on how to display words and images for a web page.
- VTRP – VIT Team Registration Portal
- PHP – Hypertext Preprocessor. An extensible scripting language, suited for web-based development, typically embedded in HTML.

2. System Overview

The project is a registration portal accessible by a network of students and event organizers to be built for VIT, a renowned college in India. The VTRP will provide an interface to make the team registration during Gravitas and Riviera much easier and less cumbersome. This system will be highly reliable, cost effective and extensive within the area of deployment. This platform will be available for every user interface. This will greatly reduce the work load of the organizers and coordinators.
The scope of the project is to make a client-server architecture for all the students of the college who can register in a team for an event they have signed up for. We will develop and provide the interface for populating the VTRP's Database Management System (DBMS) from the list of students registered for any events.

System Context Diagram



3. Detailed Description of Components

3.1 Database

We will be using Amazon Web Services as our database. It will contain tables to represent the data that will be used in the project. It is divided into two parts:

- Received Database: This contains all the entries/registrations for all events received from the club coordinators.
- Main Database: This will be the database that we create by manipulating the data from the received database.

Received Database:

TABLE 1: `Registered_Student`

(Student_name,Student_regno,Student_mobileno,Event_name)

Main Database:

TABLE 2: `Event_details`

(Event_name,Club_name,min_participants,max_participants)

TABLE 3: `Team_details`

(Student_regno,Event_name,Team_name)

TABLE 4: `Captain_details`

(Captain_regno,Event_name,Team_name)

TABLE 5: `Clublogin_details`

(Club_id,password)

TABLE 6: `Team_approval`

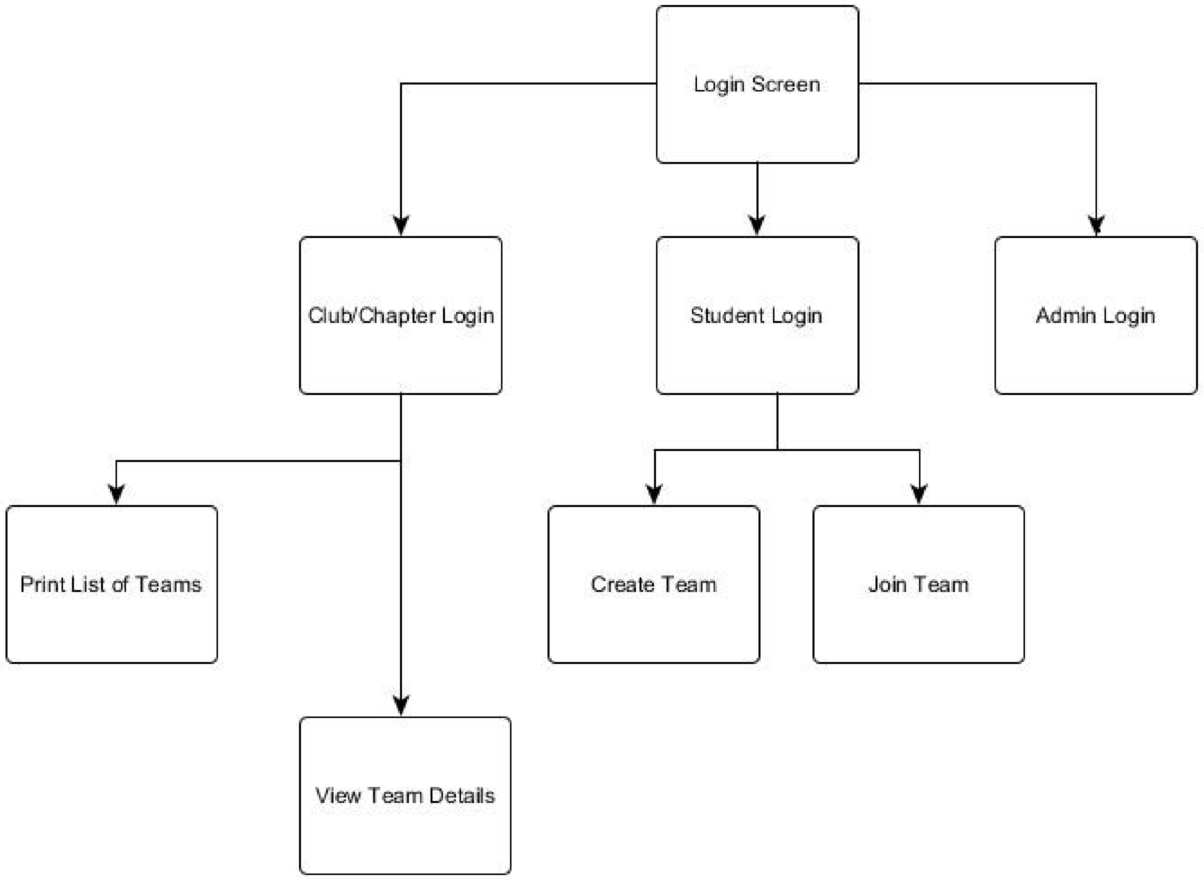
(Student_regno,Event_name,Team_name,Approval_status)

3.2 Client Software (Registration Portal)

The client software will reside on the user's PC. Its purpose is to present data to the user as requested, and provide an interface so that the user can easily update project information. It will handle conflict resolution automatically when possible, and otherwise allow the user to choose which version of an item to keep.
The various screens used in the portal are as follows:

- Login Screen: This screen asks the login type from the user: Student, Club or Admin.
- Student Login Screen: This asks the user for his/her registration number and password.
- Club Login Screen: This asks the user for their login id and password.
- Admin Login Screen: This asks the user for their login id and password.
- Student Screen: This allows the participants to view the events they have registered in, create and join teams, view the team details.

- Club Screen: This allows the clubs to view the list of teams under their events and the team details.
- Admin Screen: This allows the admin to make required updates in the database.



4 Reuse and relationship to other products

4.1 Java and Java tools

From the beginning, we set a goal to make use of any existing code to avoid wasting time duplicating other's work. We also decided to use open source or freeware solutions wherever possible. Because we are creating software for a defined application, it is possible to use open source technologies in each area. First, we decided to implement our code in Java because it is free and allows us to run on a wide range of operating systems. Next, we chose a development environment that would allow us to edit our Java code. NetBeans was chosen because it is free and has many plug-ins which allow us to use already existing applications. It also helped us create the GUI because it allows us to drag and drop frames and edit the generated code all within the same environment.

4.2 Database and Reporting

With these decisions in place, we needed a database on which to store our records. We chose Amazon Web Services, because it is having many management tools, and includes a visual interface that is very appealing.

5. Design Decisions and Tradeoffs

5.1 Assumptions and Dependencies

- The registrations for Gravitas and Riviera are stored in an online database which we are given access to.
- The operation of VTRP depends on changes being made in the event lists or registrations by the clubs and chapters.

5.2 General Constraints

- The VIT Team Registration Portal should be small in size but productive. It should be fast and should decrease the work of the coordinators.
- The VTRP will be available only in the English language.
- All HTML code shall conform to HTML 4.0 standard.
- All scripts shall be written in PHP.

5.3 Goals and Guidelines

- Emphasis shall be placed on Usability as the User Interface will be used by users without much training.
- The system must be fully functional, tested and deployable within the scheduled time frame.
- The system must be able to be modified by the user to display the target data in various reports for various purposes.

5.4 Development Methods

This project is being constructed using an evolutionary prototyping approach.

5.5 Three Tier Design

Deciding how to judiciously divide the project between all team members was another design issue. We finally decided on a 3 tier design, which is an application program organized into three major parts, each of which is distributed to different places in a network.

The three parts are:

1. The client application
2. The server application
3. The database and programming related to managing it

A 3-tier application uses the client/server computing model. With three tiers or parts, each part can be developed concurrently by a different team of programmers. Because the programming for a tier can be changed or relocated without affecting the other tiers, the 3-tier model makes it easier for an enterprise or software packager to continually evolve an application as new needs and opportunities arise. Existing applications or critical parts can be permanently or temporarily retained and encapsulated within the new tier of which it becomes a component. This design idea was very appealing to our team, especially for portability purposes.

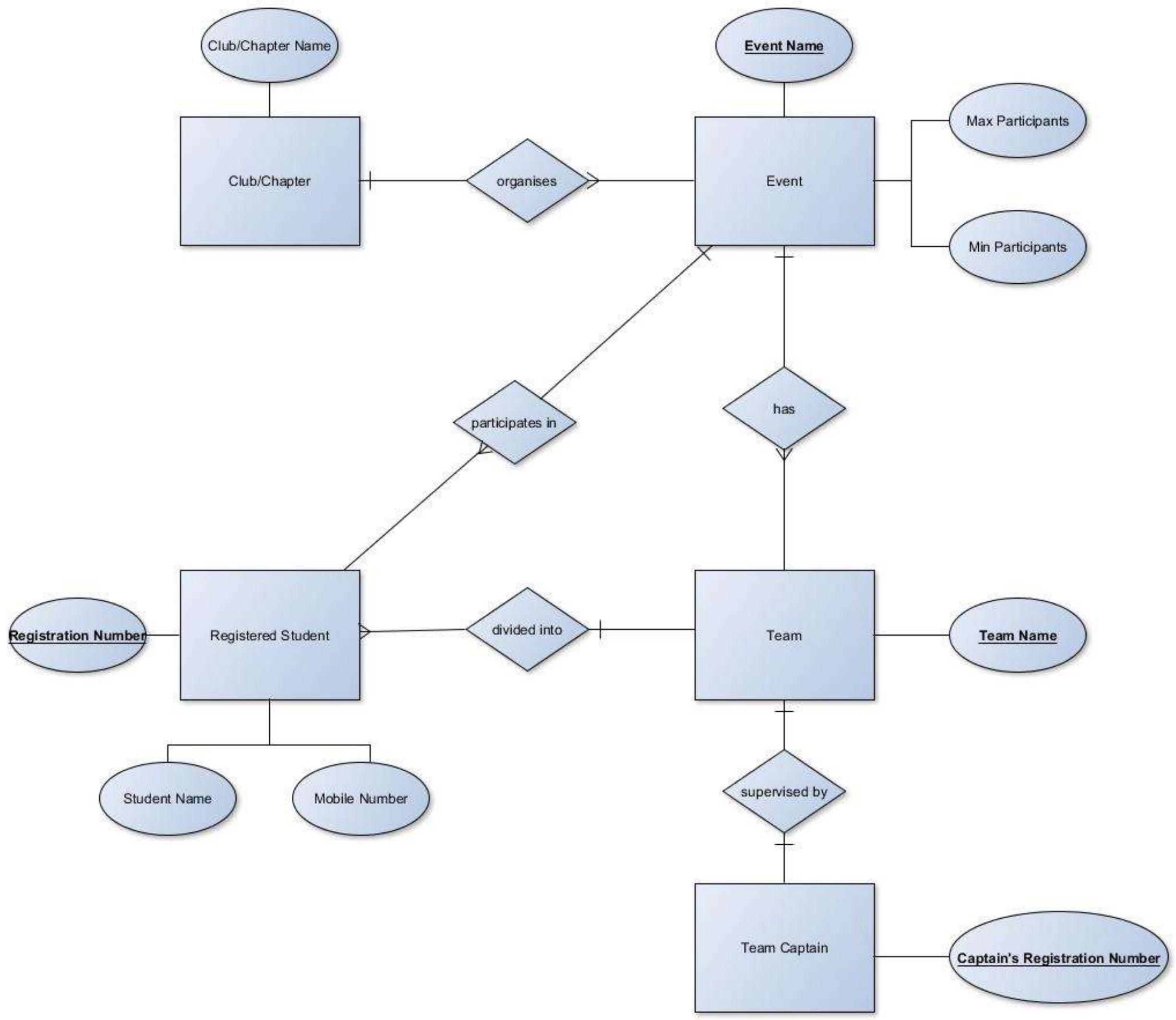
5.6 Java/Android Debate

The first issue we dealt with as a group was whether to develop in Java or .NET. Two of our members wanted to develop in .NET because they are familiar with it and enjoy working with it more. Also, PDA development is easier with .NET. However, while some members of the group do not have .NET experience, all have some Java coding experience. Another important issue we considered in selecting a code was the fact that .NET does not work in all standard operating environments. The range of operating systems that support Java is much larger. Operating systems of interest were: Windows, Unix, Linux, and Pocket PC, all of which are supported by Java and not by .NET.

5.7 Project/Project Items

We also considered using one class for both projects and sub-tasks. This would allow the user to easily upgrade a sub-task to project status, or to assign the sub-task to another user. The only difference between the two would be the amount of information that the user provided. However, after examining the options, we decided that it will be easier to create two different tasks and provide a means by which the user can "upgrade" a sub-task to project status, at which point they can reassign it if they so desire.

6. Entity Relationship Diagram (ERD):


Club/Chapter Name
Event Name
Max Participants
Club/Chapter
organises
Event
Min Participants
participates in
has
Registration Number
Registered Student
divided into
Team
Team Name
supervised by
Student Name
Mobile Number
Team Captain
Captain's Registration Number