|  |  |  |  |
| --- | --- | --- | --- |
| logo ngan.png | | **MINISTRY OF EDUCATION AND TRAINING** | |
| **FPT UNIVERSITY** | |
| **CAPSTONE PROJECT DOCUMENT** | |
| **BUILDING A WEBSITE SUPPORT THE ACTIVITIES “TIẾP SỨC MÙA THI” CAMPAIGN**  **Project Report #1 – Project Introduction** | |
|  | |
| |  |  | | --- | --- | | **Group 22** | | | **Group Members** | |  |  | | --- | --- | | SE60769 | Nguyen Duy Khoa | | SE60687 | Le Nguyen Huu Tri | | 60358 | Nguyen Dinh Tuan | | 60325 | Tran Nguyen Kim Vinh | | 60339 | Nguyen Vinh Hien | | | **Supervisor** | Nguyen Trong Tai | | **Ext Supervisor** | N/A | | **Capstone Project Code** | TSMT | | |
| - Ho Chi Minh City, January 2014 - | |

TABLE OF CONTENTS

[4. System Design Description 3](#_Toc381299333)

[4.1. Design Overview 3](#_Toc381299334)

[4.2. System Architectural Design 3](#_Toc381299335)

[4.2.1. What is MVC? 3](#_Toc381299336)

[4.2.2. Advantages and disadvantages of MVC 4](#_Toc381299338)

[4.3. Component Diagram 4](#_Toc381299339)

[4.4. Detailed Description of Component 4](#_Toc381299340)

[4.5. Sequence Diagram 4](#_Toc381299341)

[4.6. User Interface Design / Hardware Interface Design 4](#_Toc381299342)

[4.7. Database Design or Data Structures or Algorithms 4](#_Toc381299343)

[**Figure 4-1: MVC Model** 3](#_Toc381299337)

# System Design Description

## Design Overview

The System Design Description (SDD) describes the system architectural design and the detailed design, including the user interfaces of the system.

The detailed designs of the system should contain the basic principles of software design, including “high cohesion, low coupling” and “open to extension, closed to modification” principles.

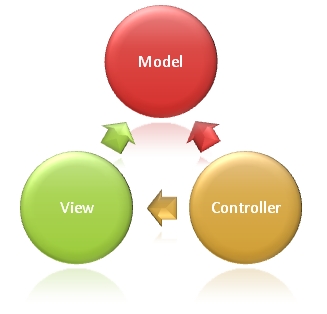
The system is structured basing on MVC framework.

## System Architectural Design

### What is MVC?

MVC stands for model – view – controller pattern. Trygve Reenskaug introduced MVC into Smalltalk-76 while visiting Xeroc Parc, in the 70s; next, in the 80s, Jim Althoff and others implemented a version of MVC for the Smalltalk-80 class library. Only later MVC was expressed as a general concept, in a 1988 article.

MVC is a software architecture pattern which separates the representation of information from user’s interaction with it. The model consists of application data, business rules, logic and functions. A view can be any output representation of data, such as a chart or a diagram. Multiple views of the same data are possible, such as bar chart for management and a tabular view for accountants. The controller mediates input, converting it to commands for the model or view. The central ideas behind MVC are code reusability and separation of concerns.



**Figure 4-1:** MVC Model

In addition to dividing the application into three kinds of components, the MVC design defines the interactions between them:

* **A controller:** can send commands to its associated view to change the view's presentation of the model (e.g., by scrolling through a document). It can also send commands to the model to update the model's state (e.g., editing a document).
* **A model:** notifies its associated views and controllers when there has been a change in its state. This notification allows the views to produce updated output, and the controllers to change the available set of commands. A *passive* implementation of MVC omits these notifications, because the application does not require them or the software platform does not support them.
* **A view:** requests from the model the information that it needs to generate an output representation to the user.

### Advantages and disadvantages of MVC

* **Advantages:** the MVC model demonstrates professionalism in programming and design analysis. It is divided into independent components to help develop applications faster, simpler, easier upgrades and maintenance.
* **Disadvantages:** for small projects applying MVC model caused cumbersome, time consuming in development process. Time consuming to transits data between components.

## Component Diagram

## Detailed Description of Component

## Sequence Diagram

## User Interface Design / Hardware Interface Design

## Database Design or Data Structures or Algorithms