In this chapter we are focusing on the College Event information portal. This is a simple project of Web-Base system. We are introducing this project with an introduction about the project topic and discussing the objectives of this project. Which is a brief information on the project on which basis projects work and run on the system.

* 1. **Title**

Eventure

* 1. **Introduction**

Every college website typically has a section for Events. This section contains a complete information about past and upcoming Event of the college. This information is usually general in nature and does not change often. In this scenario, the central college team and department provides the desired event information to the website administrators and they build the website section for the department as per the instructions. Any subsequent changes or additions to the website will require making a request for the same to the website administrators. This severely restricts the flexibility and usability of the website for the department and its faculty.

A computer science department can easily create its own website, but a small mca department might find this bothersome. It would be highly desirable if there was a system that could enable faculty members to update the desired information on a college website on their own and whenever they want. This way, they can use the internet to communicate with the students much more effectively. They can maintain up to date their information on the college website such as upcoming events, deadline for registration, feedback etc. This system could also support other features such as a discussion board to interact with students, etc. These desirable features were the motivation for this web based content management system.

**1.3 Objective**

The main purpose of this project is to simplify the process of handling each event by providing a web interface for admin, teacher and students. It mainly helps the student for knowing the details about the college event and extracurricular activities. This system, is help to reduce the consumption of time during maintaining the event records of college. It easily provides a information to student. Our System also provides an easy way not only to automate all functionalities of a college, but also to provide full functional reports to top management of college with the finest of details about any aspect of college.

**2.1 Existing System**

Now a days we do all work with computerized system but with current system is very complicated to keep the track of all the events at one platform. As well as this current system is time consuming & also expensive and it might be not accurate and not fastest.

This software helps you to track all the events, co-curricular activities in a college.

**2.1.1 Limitations of Current System:**

* Time consuming.
* More expensive.
* Searching problem.
* Less accuracy.
* Difficult to generate feedback report.

**2.1.2 Advantages of Computerized System:**

* Time saving.
* Less expensive.
* Powerful searching.
* Solve the problem of maintains information page.
* More accuracy.
* Easy generate feedback report.
* Contact details for all types of queries.

**2.2 Feasibility Study**

The objective is to determine whether the proposed system is feasible. There are three aspects of feasibility study to which the proposed system is subjected as discussed below.

**2.2.1 Technical Feasibility**

Technical feasibility assesses whether the current technical resources are sufficient for the new system. If they are not available, can they be upgraded to provide the level of technology necessary for the new system? It checks whether the proposed system can be implemented in the present system without supporting the existing hardware. Hardware used in this system is only a computer which is not dependent on any platform.

**2.2.2. Economic Feasibility**

Economic feasibility determines whether the time and money are available to develop the system. It also includes the purchase of new equipment, hardware, and software. A software product must be cost effective in the development, on maintenance and in the use. Since the hardware and resources are already available with the organization and the organization can afford to allocate the required resources.

**2.2.3. Operational Feasibility**

Operational feasibility determines if the human resources are available to operate the system once it has been installed. The resources that are required to implement or install are already available with the organization. The persons of the organization need no exposure to computer but have to be trained to use this particular software. A few of them will be trained. Further, training is very less.

In this chapter we listed about the analysis done for the project and give the different entity which helps to make the product. The analysis about the project we are making the data flow diagram which contain the levels 0 and level 1.

**3.1 Technologies Used**

**3.1.1 Hardware**

Processor : Intel IV or above

Mouse : Standard two button or higher

Keyboard : Standard 101-102 key keyboard

Display : 15” Monitor

**3.1.2 Software**

Operating System : Windows XP & above

Platform : Django Web Framework

User Interface : HTML,CSS,Bootstrap,Javascript

Language Used : Python

Database : SQLITE

**3.2 About Language**

**Python**

Python is an interpreter, object-oriented, high-level programming language with dynamic semantics. Its high-level built in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed.

**CSS**

**C**ascading **S**tyle **S**heets, fondly referred to as CSS, is a simple design language intended to simplify the process of making web pages presentable. CSS handles the look and feel part of a web page. Using CSS, you can control the color of the text, the style of fonts, the spacing between paragraphs, how columns are sized and laid out, what background images or colors are used, layout designs, and variations in display for different devices and screen sizes as well as a variety of other effects. CSS is easy to learn and understand but it provides powerful control over the presentation of an HTML document. Most commonly, CSS is combined with the markup languages HTML or XHTML.

**HTML**

HTML is a hypertext mark-up language which is in reality a backbone of any website. Every website can’t be structured without the knowledge of html. If we make our web page only with the help of html, than we can’t add many of the effective features in a web page, for making a web page more effective we use various platforms such as CSS. So here we are using this language to make our web pages more effective as well as efficient. And to make our web pages dynamic we are using Java script.

**Bootstrap**

Bootstrap is a [free and open-source](https://en.wikipedia.org/wiki/Free_and_open-source_software) front-end [framework](https://en.wikipedia.org/wiki/Application_framework) for designing [websites](https://en.wikipedia.org/wiki/Website) and [web applications](https://en.wikipedia.org/wiki/Web_application). It contains [HTML](https://en.wikipedia.org/wiki/HTML)- and [CSS](https://en.wikipedia.org/wiki/CSS)-based design templates for [typography](https://en.wikipedia.org/wiki/Typography), forms, buttons, navigation and other interface components, as well as optional [JavaScript](https://en.wikipedia.org/wiki/JavaScript) extensions. Unlike many earlier web frameworks, it concerns itself with [front-end development](https://en.wikipedia.org/wiki/Front-end_web_development) only.

**Database** SQLite

SQLite is a software library that provides a relational database management system. The lite in SQLite means light weight in terms of setup, database administration, and required resource.

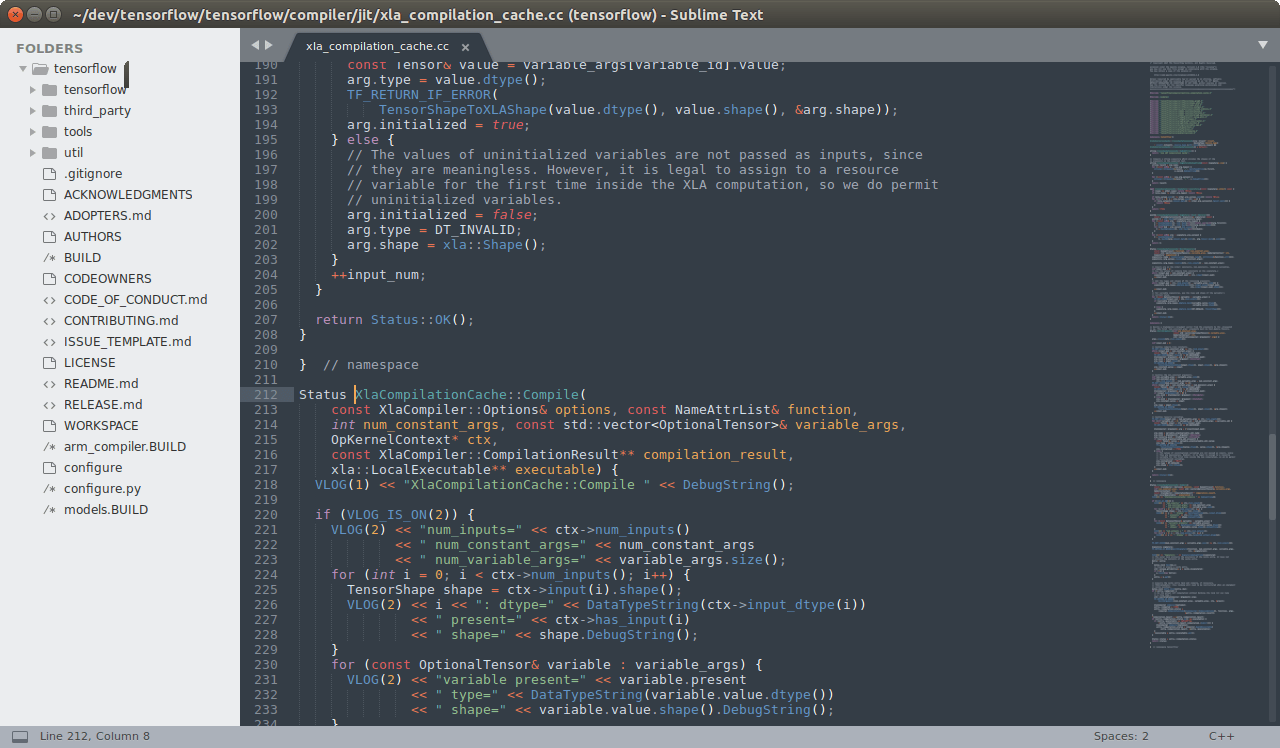
**3.3 Tools:-**

**Sublime Text 3 Editor**

Sublime Text editor is a sophisticated text editor which is widely used among developers. It includes wide features such as Syntax Highlight, Auto Indentation, File Type Recognition, Sidebar, Macros, Plug-in and Packages that make it easy for working with code base. This tutorial gives you a comprehensive coverage of concepts of Sublime Text and makes you comfortable to use it in your software development projects.

Sublime Text Editor is a full featured Text editor for editing local files or a code base. It includes various features for editing code base which helps developers to keep track of changes. Various features that are supported by Sublime are as follows −

* Syntax Highlight
* Auto Indentation
* File Type Recognition
* Sidebar with files of mentioned directory
* Macros
* Plug-in and Packages

****

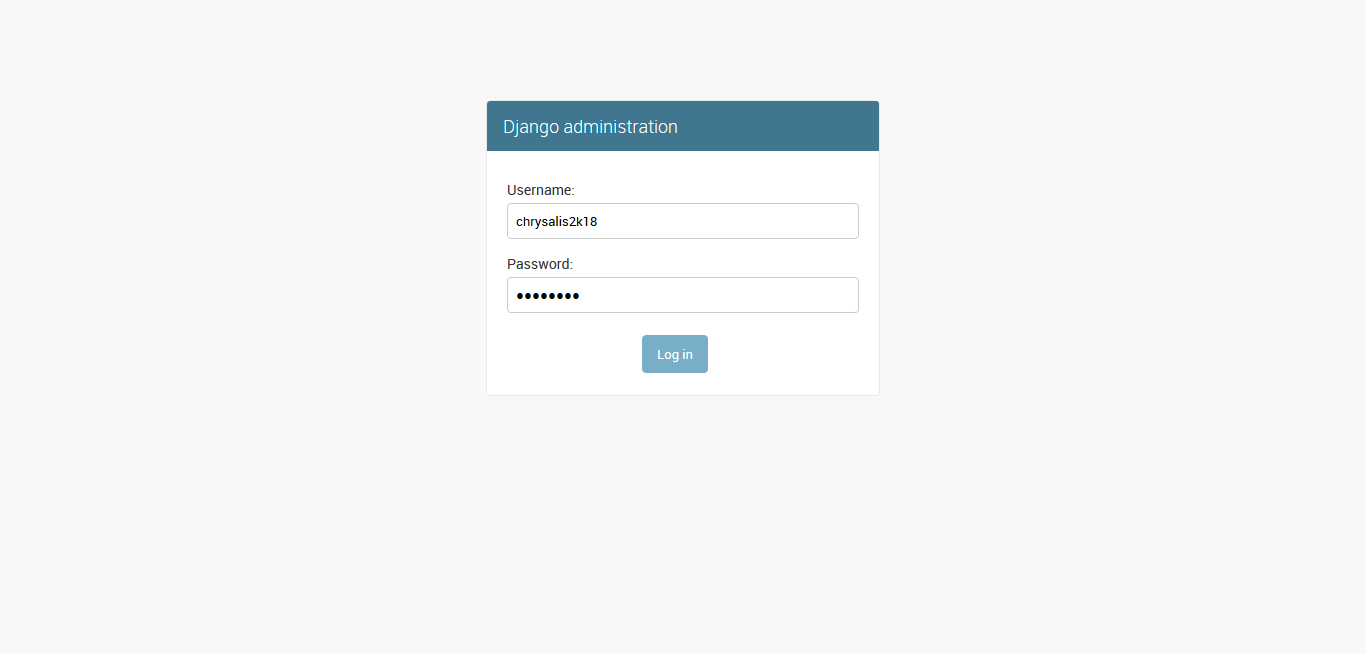
**Figure 3.1 Sublime Text 3**

**4.1 Problem Definition**

* Purpose : Eventure is designed to ease to find all event related information at one platform. .
* Objective: This system provides security by granting access rights for changing the sensitive data to administrator only.
* Scope : This type of software is suitable for all colleges. Separate division is provided to maintain Students, Teachers record, Events and Discussion etc.

**4.2 Front End Design**

* Username : User will have to enter a unique username.
* Sign In : Login if existing user.



**Figure 4.1 Front End Design**

**4.4 Back End Design**

A database design is a collection of stored data organized in such a way that the data requirements are satisfied by the database. The general objective is to make information access easy, quick, inexpensive and flexible for the user. There are also some specific objectives like controlled redundancy from failure, privacy, security and performance.

**4.5 Flowchart of System**

One of the tools of structured analysis is the diagram. A data flow diagram is a graphical representation of the system. The analyst can use dataflow diagram to explain this understanding about the system.

**Description**

* A graphical tool used to describe and analyze the movement of data through a system manual or automated including the process store of data and delays in the system.
* The transformation of data from input to output through processes may be described logically and independently of the physical component associated with system.
* The data transformation at each step before moving on to the next stage. These processing steps or transformation are programs function when data flow diagrams are used to documents a software design.

**The symbols that are used in the data flow diagram carry following meaning.**

**Rectangle it symbolizes the sources.**

**Ellipse/Circle it symbolizes the process.**

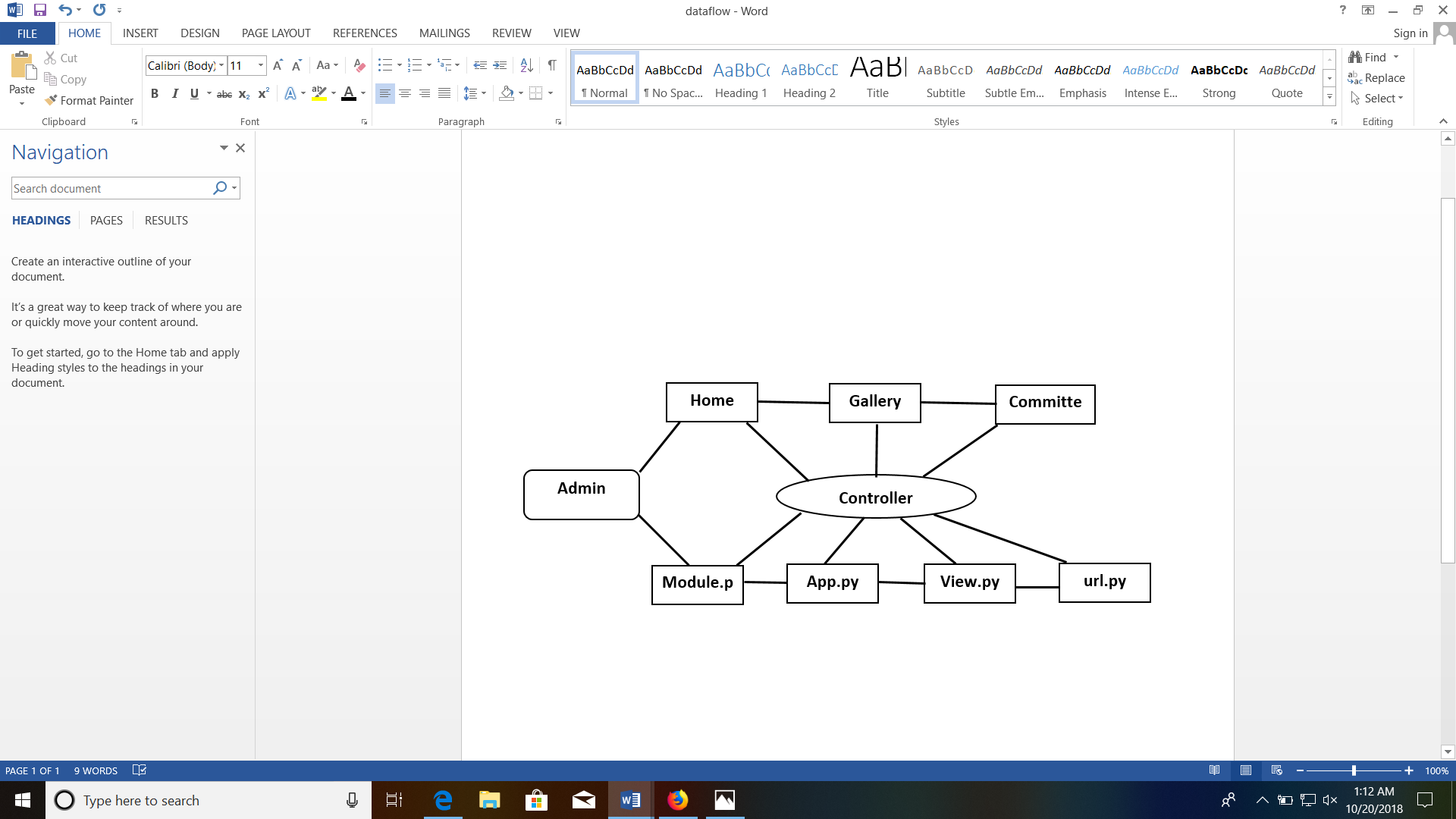
**Open Rectangle it symbolizes store data.**

**Unmarked Line it symbolizes connectivity**

**Without a dataflow.**

**Marked Line it symbolizes connectivity**

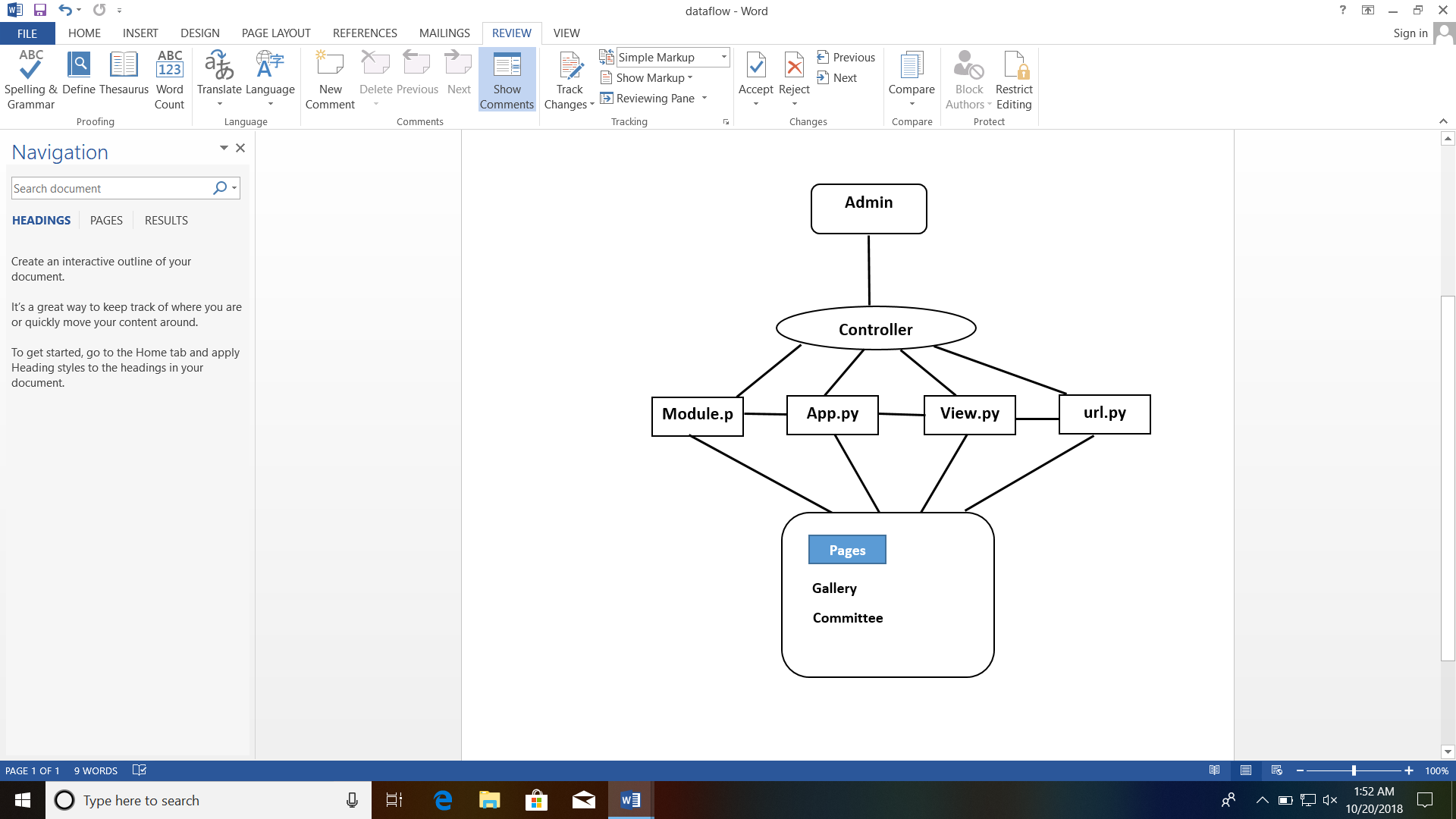
**With a dataflow.**



**Figure 4.5 System Flow Chart**

**4.6 Data Flow Diagram**

* Data flows are an intuitive way of showing how data is processed by a system.
* At analysis level analyst should be used to model the way in which data is processed in the existing system.
* Data flow models are used to show how data flows through a sequence of processing steps.



**Figure 4.6 Data Flow Diagram**

In this chapter we are discussing about the system implementation, which include project Code of application. In this section, we are giving the information about code with comment line to understand what kind of working is done for what purpose. The output screen shots are also shown which gives a brief idea about the project.

**5.1 Coding Explanation**

**Main Project (chrysalis2k18)**

**url.py**

from django.conf.urls import url, include

from django.contrib import admin

from django.conf import settings

from django.conf.urls.static import static

from registration.views import index

urlpatterns = [

url(r'^$', index),

url(r'^admin/', admin.site.urls),

url(r'^home/', include('home.urls')),

url(r'^events/', include('events.urls')),

url(r'^gallery/', include('gallery.urls')),

] + static(settings.MEDIA\_URL, document\_root = settings.MEDIA\_ROOT)

**Home page**

**url.py**

from django.conf.urls import url

from . import views

urlpatterns = [

url(r'^$', views.index, name="home"),

]

**gallery page**

**url.py**

from django.conf.urls import url

from . import views

urlpatterns = [

url(r'^', views.index, name="gallery"),

]

**model.py**

from django.db import models

class gallery(models.Model):

title = models.CharField(max\_length=250, blank=True);

description = models.CharField(max\_length=1000, blank=True)

file = models.FileField(max\_length=250)

def \_\_str\_\_(self):

return 'Title : ' + self.title + ' ,' + 'Description : ' + self.description

**Views.py**

from django.http import HttpResponse

from django.shortcuts import render

from .models import gallery

def index(request):

all\_images = gallery.objects.all().order\_by('-id')

context = {'all\_images' : all\_images}

return render(request, 'gallery/index.html', context)

**committee page**

**url.py**

from django.conf.urls import url

from . import views

urlpatterns = [

url(r'^', views.index, name="index"),

#url(r'^(?P<record\_id>[0-9]+)/$', views.detail, name="detail"),

]

**model.py**

from django.db import models

# Create your models here.

class events(models.Model):

name = models.CharField(max\_length=50);

team = models.CharField(max\_length=50);

def \_\_str\_\_(self):

return ' Name ' + self.name + 'Team : ' + self.team

class students(models.Model):

student\_name = models.CharField(max\_length=50);

student\_team = models.CharField(max\_length=50);

def \_\_str\_\_(self):

return ' Name ' + self.student\_name + 'Team : ' + self.student\_team

**Views.py**

from django.http import HttpResponse

from .models import events, students

from django.template import loader

def index(request):

records = events.objects.all()

students\_records = students.objects.all()

template = loader.get\_template('events/index.html')

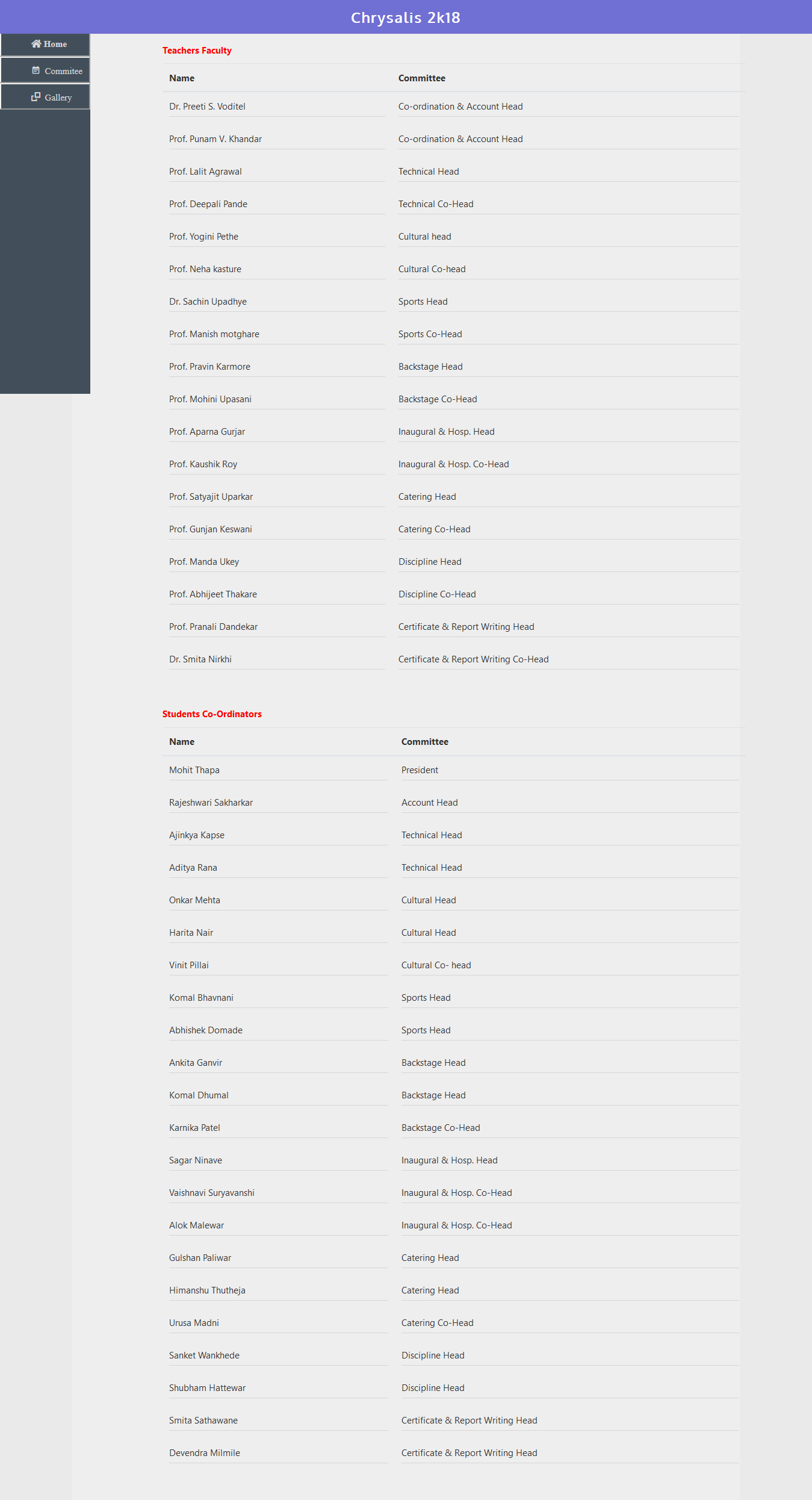
context = {'records' : records, 'students\_records' : students\_records}

return HttpResponse(template.render(context, request))

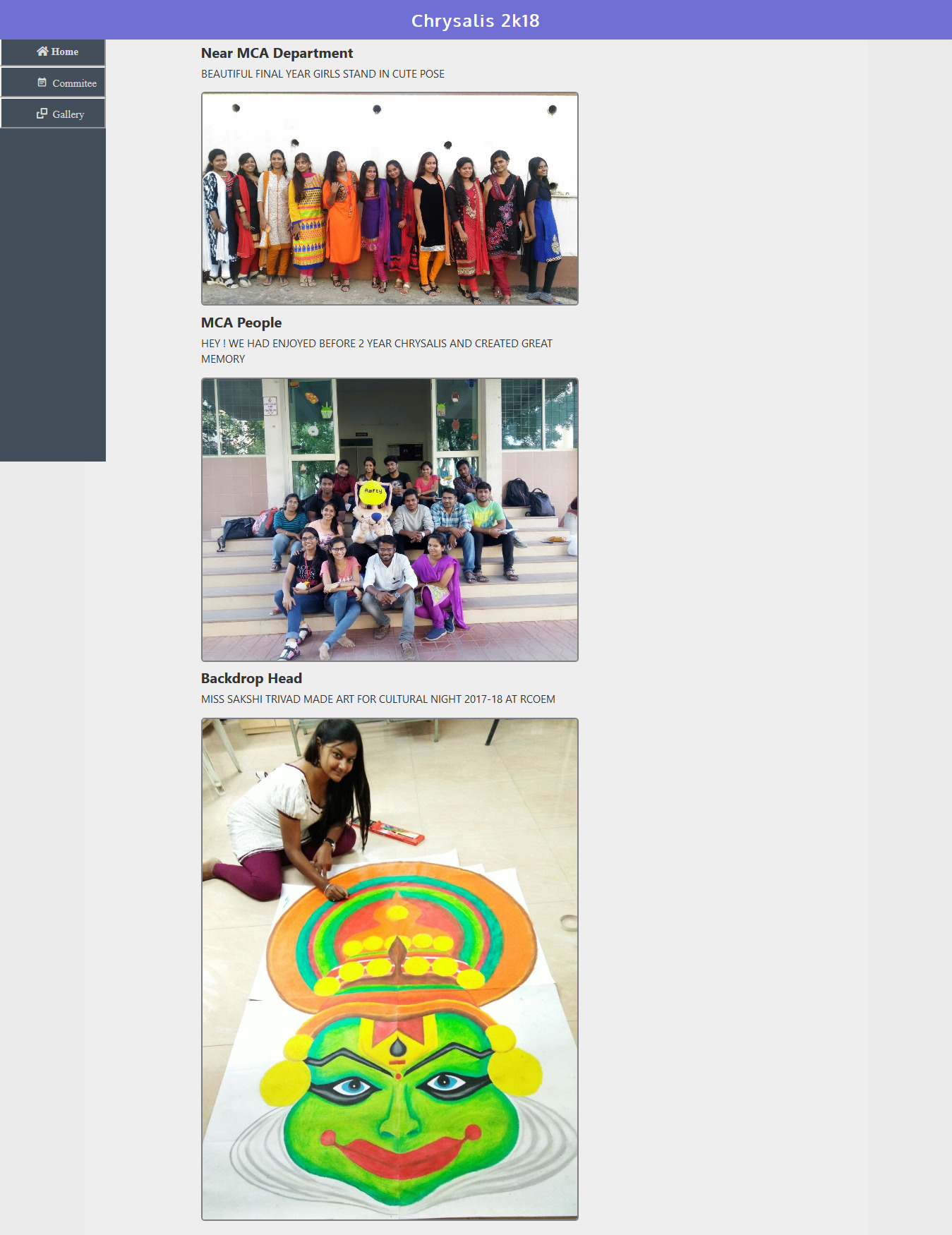
**Screenshot of Output**



**Fig 5.2.1 Home Page**



**Fig 5.2. Comittee Page**



**Fig 5.2.3 Gallery Page**

**Program design**

In this stage, a solution to the problem is designed by defining a logical sequence of steps that will achieve each of the stated system objectives. Such a sequence of steps is often referred to as an algorithm. Some of the methods used to define program algorithms are described later in this section, and include flowcharts and pseudo code. These tools allow the program designer to break a given problem down into a series of small tasks which the computer can perform to solve the problem. The user interface will also be designed during this stage, and will determine how input is obtained, how output is displayed, and what controls are available to the user.

**Program coding**

This stage, sometimes known as the implementation stage, is where the algorithms are translated into a programming language, and tends to be the longest phase of the development life-cycle. In this case, we are using Visual Basic to write the program.

**Documentation and testing**

The documentation of the program fulfils two main objectives. The first is to provide a technical reference to facilitate ongoing maintenance and development of the software itself. The second is to provide user documentation, i.e. a set of instructions that inform The user about the features of the software and how to use them. The aim of software testing is to find any errors ("bugs") in the program, to eliminate those errors (a process known as "Debugging"), and as far as is reasonably practicable should be sufficiently rigorous to ensure that the software will function as expected under all foreseeable circumstances.

**Software testing** is an investigation conducted to provide stakeholders with information about the quality of the product or service under test.Software testing can also provide an objective, independent view of the software to allow the business to appreciate and understand the risks of software implementation. Test techniques include, but are not limited to the process of executing a program or application with the intent of finding software bugs (errors or other defects).

**6.1 Testing**

Software testability is simply how easily computers programs can be tested. The checklist that follows provides a set of characteristics that lead to testable software.

* Portability.
* Observable.
* Controllability.
* Decomposability.
* Simplicity.
* Stability.
* Understandability.

**6.1.1 Why Testing is Important?**

* Meets the requirements that guided its design and development,
* Works as expected,
* Can be implemented with the same characteristics, and
* Satisfies the needs of stakeholders.

Software testing, depending on the testing method employed, can be implemented at any time in the software development process. Traditionally most of the test effort occurs after the requirements have been defined and the coding process has been completed, but in the agile approaches most of the test effort is on-going. As such, the methodology of the test is governed by the chosen software development methodology.

**6.1.2 Testing Strategies**

The philosophy behind testing is to find errors. Test cases are devised with this purpose in mind. Test case is a set of data that the system will process as normal input.

**Unit Testing**

The primary goal of unit testing is to take the smallest piece of testable software in the application, isolate it from the remainder of the code, and determine whether it behaves exactly as you expect. Each unitis tested separately before integrating them into modules to test the interfaces between modules. Unit testing has proven its value in that a large percentage of defects are identified during its use.

**Integration Testing**

Integration testing, also known as integration and testing (I&T), is a software development process which program units are combined and tested as groups in multiple ways. Integration testing can expose problems with the interfaces among program components before trouble occurs in real-world program execution. Integration testing is a component of Extreme Programming (XP), a pragmatic method of software development that takes a meticulous approach to building a product by means of continual testing and revision.

**Validation Testing**

At the validation level, testing focuses on user visible actions and user recognizable output from the system. Validations testing is said to be successful when software functions in a manner that can be reasonably expected by the customer.

**Regression Testing**

Regression testing aims at verifying the functionality of the software that is previously tested and to which changes are made. It is to ensure the old software still works with new changes.

**Acceptance Testing**

Acceptance testing is conducted to verify if the system compliance the business requirements.level, testing focuses on user visible actions and user recognizable output from the system. Validations testing is said to be successful when software functions in a manner that can be reasonably expected by the customer.

Five types of validation testing:

**Black Box Testing**

This is concerned with the proper execution of the program specification. In this testing, each function or sub program used in the main is first identified. It is complementary to White Box Testing. It uncovers a different class of errors which are not discovered by White box Methods.

**White Box Testing**

This is concerned with implementation of the program. In this, different programming structures and data structures used in program are tested for missing function, performance errors, Errors in external database access, Initializing and terminating errors.

**Interface Testing-**

This is concerned with the both program specification and program implementation. This type of testing is particularly important for object oriented system.

**Alpha Testing**

This is simulated or actual operational testing by potential users/customers or an independent test team at the developers' site. Alpha testing is often employed for off-the-shelf software as a form of internal acceptance testing, before the software goes to beta testing.

**Beta Testing**

Beta testing comes after alpha testing. Versions of the software, known as beta version, are released to a limited audience outside of the programming team. The software is released to groups of people so that further testing can ensure the product has few faults or bugs.

7.1 **Conclusion:-**

This project is developed to nature the needs of a user in a collage event updates by collage event committee. User is able to upload event date time, committee members, event photos, delete or alter information from the database.

Faculty and Committee is able visualize the data information and grab the needed information from the application. They are able to take actions as per the information and can do the changes in application data. They are able to schedule the events from student’s response and update the event information in the application.

**7.2 Future Work**

Future version of this website will still be much enhanced than the current version 1.0 thus the event website is developed and executed successfully.

This website is provides event information and all over activities organized by collage.

This is able to upload and delete image and update information by admin side . Future work of this website to upload images by web forms and manage user sessions as well as uploaded image will be analyzed and identify object in a picture.

User will get time to time notification from the website and see the changes on the website.

**8.1. Reference**

[**https://media.readthedocs.org/pdf/django/2.0.x/django.pdf**](https://media.readthedocs.org/pdf/django/2.0.x/django.pdf)

[**http://gsl.mit.edu/media/programs/south-africa-summer-2015/materials/djangobook.pdf**](http://gsl.mit.edu/media/programs/south-africa-summer-2015/materials/djangobook.pdf)

<https://www.youtube.com/watch?v=qgGIqRFvFFk&list=PL6gx4Cwl9DGBlmzzFcLgDhKTTfNLfX1IK>

<https://www.youtube.com/watch?v=Fc2O3_2kax8&list=PLw02n0FEB3E3VSHjyYMcFadtQORvl1Ssj>

<https://www.youtube.com/user/CodingEntrepreneurs>