EXERCISE NO. 3

AIM: To Understand and Analysis of the Functional Point.

REQUIREMENTS:

Hardware Interfaces

▪ Core*i* CPU 3.0 GHz, 4GB RAM

▪ Screen resolution of at least 800 x 600 required for proper and complete viewing of screens. Higher resolution would not be a problem.

▪ Pen drive / cloud storage

Software Interfaces

▪ Any window-based operating system (Windows 10)/ MAC OS

▪ WordPad or Microsoft Word

THEORY:

Included here in the ppt file. FPA_ Function Point Analysis

Exercise Example:

1.1 OBJECTIVE:

A **course management system (CMS)** is a collection of software tools providing an online environment for course interactions. A CMS typically includes a variety of online tools and environments, such as:

·An area for faculty posting of class materials such as course syllabus and handouts

·An area for student posting of papers and other assignments

·A grade book where faculty can record grades and each student can view his or her grades

·An integrated email tool allowing participants to send announcement email messages to the entire class or to a subset of the entire class

·A chat tool allowing synchronous communication among class participants

- A threaded discussion board allowing asynchronous communication among participants.

In addition, a CMS is typically integrated with other databases in the university so that students enrolled in a particular course are automatically registered in the CMS as participants in that course.

The Course Management System (CMS) is a web application for department personnel, Academic Senate, and Registrar staff to view, enter, and manage course information formerly Submitted via paper. Departments can use CMS to create new course proposals, submit changes for existing courses, and track the progress of proposals as they move through the stages of online approval.

Problem Analysis and Project Planning

A course management system is a set of tools that enables an online environment for course interaction i.e. to create online course content and post it on the Web without having to handle HTML or other programming languages.

Course management system become an integral a part of the upper education system. They create teaching and course management easier by providing a framework and set of tools for faculties and for students. The executive aspects of such systems could include class rosters (a group of people or things) and therefore the ability to record students' grades. With relevance the teaching aspects, however, it might include learning objects, class exercises, quizzes and tests. The CMS might also include tools for real-time chat, integrated email tool allowing participants to send announcement email messages to entire class or to a subset of the entire class. The CMS tool additionally focuses on all aspects of teaching, learning and teacher-student interaction.

1.2 RESOURCE:

Software

Requirement

Analysis (1) Module

Summary:

(1.1) Administrator Module:

Admin can produce accounts for college students and faculties and make course programmed list and add faculties and students to it course list.

Admin can produce course details exploitation course creation kind that consists in fact name, course id, and choose student. Using Student creator kind student details are entered to information. User name, adapt username, password, given name and name, ID. After accounts are produced supported every students and instructors are divided and accessorial to list exploitation create missing students kind.

(1.2) Faculty Module:

It can check student's papers, their assignments and assign grades for work. This module accommodates preparation menu, choose student for grades.

(1.3) Students Module: Student can register with application or the proposed system and login with user name and password. He will check and submit assignment and his/her grade. Every student can have id.

1.2 PROCEDURE:

(2) Functional and Non-Functional Requirements (2.1) Functional Requirements:

(2.1.1) Creating Courses

Integration with registration system: The system shall periodically upload the latest registrar's classes list to determine courses that offered in the current semester.

The system shall generate course for each class that registered and determine the current set of students that enrolled in that class.

The system shall allow course instructor to update course content.

(2.1.2) Grade Management

a. Allow grades to be entered online: The system shall allow instructors to enter and modify grades online.

b. Allow students to access their grades online: The system shall allow student to log in their account and check their grades at any time.

c. The system shall provide statistical information such as averages, standard deviation, and median about student's grades.

d. Track and Handle Re-grade Requests: The system shall be able to track and handle requests for re- grades, and all information about re-grades shall be available to the student, and the course instructor.

(2.1.3) Paper and Assignment Submission

a. Accept submissions in multiple formats: The system shall accept submissions in multiple formats, including .zip, .cpp, .txt, .doc,etc.

b. Support for late submissions: The system shall provide information about late submissions, and also disallow submissions after a certain period of time.

c. Integration with grade management: The homework submission system shall be integrated with the grade management by using online grading templates that can be filled out, and automatically annotating code with line numbers.

1. Assignment grades can be automatically posted to student account.

2. Grader comments can be sent along with the grades.

(2.1.4) Create Accounts

a. The system shall automatically create accounts for each class.

1. Create one account for course instructor regardless to the number of classes that he/she teaches.

2. The account username is course name and its number.

3. The account password is the same password that in Academic Information System (AIS).
4. Any change in the password in AIS the system shall reflect it on the instructor account password in CMS.

5. Create one account for each student that registered in this class.
6. The account username is course name and its number.
7. The account password is the same password that in Student Information System (SIS).
8. Any change in the password in SIS the system shall reflect it on the student account password in CMS.

b. Instructor account contain the classes that he/she teach, each class contain list of student that ordered based on student serial number.

c. Instructor can modify student grades from his/her account.

(2.2) Non-Functional Requirements:

(2.2.1) Response Time

a. Average response time shall be less than 2 second.

(2.2.2) Throughput

a. The system shall accommodate 1000 booked per minute.

(2.2.3) Recovery Time

a. In case of a system failure, redundant system shall resume operations within 30 sec.

b. Average repair time shall be less than 1 hour.

(2.2.4) Start-up/Shutdown Time

a. The system shall be operational within 1 minute of starting-up.

(2.2.5) Capacity

a. The system accommodates 4000 concurrent users.

(2.2.6) Utilization of Resources

a. The system shall store in the database no more than one million transactions.

b. If the database grows over this limit, old transaction shall be backed up and deleted from the operational database.

(2.2.7) Security

a. Firewall Protection: The course management software system shall run inside a firewall.

b. Support different roles: The system shall support different roles for users, such as Instructors, Students, and administrative staff, the user logged in with given role should only be allowed access consistent with that role. For example a student shall only be allowed to see he/she grades not to modify it.

(2.2.8) Reliability

a. The system shall not be down more 2 times in year.

(2.2.9) Scalability

a. Scaling the system to large number of users: large courses will have hundreds of students.

b. The system shall be able to handle the load for such courses, especially near assignment deadlines when many students can be expected to access the course management system.

Report Submission:

Create your own project FPA and submit it as a report.