## A

**PROJECT REPORT**

ON

## Faculty Management System

UNDER

## NON SYLLABUS PROJECT

## 

DEPARTMENT OF ARTIFICIAL INTELLIGENCE & DATA SCIENCE POORNIMA INSTITUTE OF ENGINEERING & TECHNOLOGY, JAIPUR

(Academic Year 2022-23)(Odd)

**Submitted To: Submitted by:**

Ms. Chanchal Tiwari Ma’am Yaduttam Pareek **(PGI20CS086)**

(Assistant Professor)

**DECLARATION**

I hereby declare that the report entitled “**FACULTY MANAGEMENT SYSTEM**” was carried out and written by me under the guidance of ”Ms. Chanchal Tiwari Ma’am”, Assistant Professor, DEPARTMENT OF ARTIFICIAL INTELLEGENCE & DATA SCIENCE, Poornima Institute of Engineering & Technology, Jaipur. This work has not been previously formed the basis for the award of any degree or diploma or certificate nor has been submitted elsewhere for the award of any degree or diploma.

Place: - Student Name: - Yaduttam Pareek

Date: - Registration Number: - PGI20CS086

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| S. No. | Description | Page No. |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**Chapter 1**

**Introduction to Project**

* 1. **Objective of Project**

As an NSP project, FACULTY MANAGEMENT SYSTEM is created using Java.

The name, salary, department, and other information linked to faculties may be handled by this application or project.

This software manages data in the form of a database, or group of tables.

* 1. **Types of User**

Currently, it only has one type of user, an admin or manager, who can manage the records and is only permitted after a successful login that includes the necessary combination of username and password, directing the person to the main page to maintain the data.

* 1. **Constraints and Dependencies**

This project's dependencies are as follows:

* A user with basic knowledge of system
* User should have knowledge of English language.
  1. **Project Life Cycle**

The software development model used in making of this project is a spiral model which can be explained as: -

A Spiral model is an evolutionary software process paradigm that blends the iterative nature of prototyping with the controlled and systematic components of the linear sequential model. It has the ability to quickly create new software versions. The spiral process is used to create the software in a succession of incremental releases. During the early stages, the extra release may be a paper model or prototype. Later cycles yield progressively complete versions of the desired system.

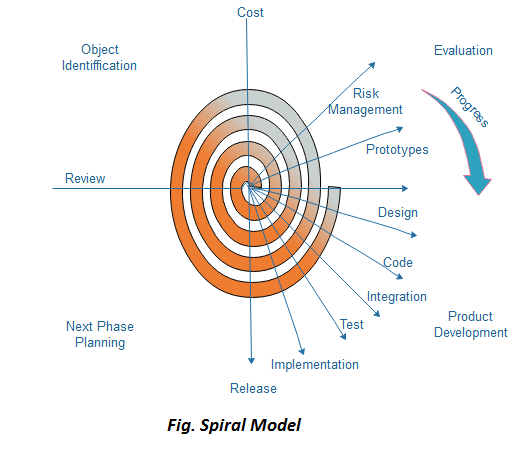


Fig 1.1: Spiral Model

* + 1. **Advantages**

The following are the benefits of employing this model:

1. A significant amount of risk analysis
2. This is useful for large, mission-critical initiatives.
   * 1. **Disadvantages**

The disadvantages of employing this model are as follows:

1. It might be an expensive model to utilize.
2. Risk analysis need extremely specialized knowledge.
3. It is ineffective for smaller tasks.

**Chapter 2**

**Requirement Analysis**

* 1. **Functional requirement**

The functional requirements of the project are: -

1. Whenever a user logs into the system, he or she must authenticate.
2. Administrative functions.
3. Transaction corrections, adjustments and cancellations.
   1. **Non-Functional requirements**

The non-functional requirements of the project are: -

1. The processing of each request should be done within 10 seconds.
2. Every entry should be done as desired like id can only have Integer values.
3. Data fetching and manipulation should be fast.
   1. **Technology Used**

The technology used in making of this project are:-

1. **JAVA: -**

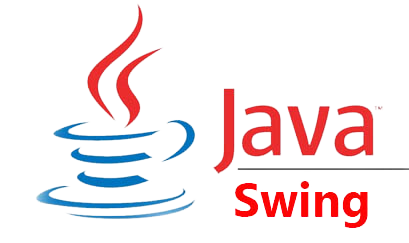
Java is a high-level, class-based, object-oriented programming language with a low number of implementation dependencies. It is a general-purpose programming language designed to allow programmers to write once and run anywhere (WORA), which means that generated Java code may run on any platform that supports Java without the need for recompilation. Java programs are often compiled to bytecode that can run on any Java virtual machine (JVM), independent of computer architecture.

1. **My SQL: -**

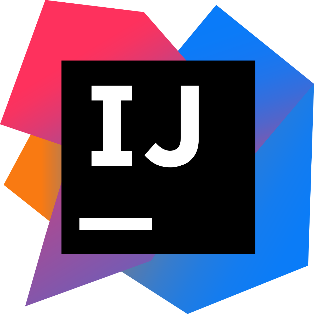
MySQL is a relational database management system that is free and open source (RDBMS). A relational database organizes data into one or more data tables, each of which can be coupled to another; these relationships assist shape the data. SQL is a programming language that allows programmers to build, change, and retrieve data from relational databases, as well as control user access to the databases. An RDBMS, in addition to relational databases and SQL, works with an operating system to implement a relational database in a computer's storage system, manages users, allows for network access, and simplifies verifying database integrity and backup generation.

1. **JAVA Swing: -**

Swing is a Java GUI widget toolkit. It's a component of Oracle's Java Foundation Classes (JFC), which provides an API for creating graphical user interfaces (GUIs) for Java programs.

Swing was created to give a more comprehensive collection of graphical user interface components than the previous Abstract Window Toolkit (AWT). Swing offers a pluggable look and feel that allows applications to have a look and feel that is unconnected to the underlying platform, as well as a look and feel that emulates the look and feel of numerous platforms. It includes more powerful and adaptable components than AWT. Swing features various complex components, such as tabbed panels, scroll panes, trees, tables, and lists, in addition to traditional components such as buttons, check boxes, and labels.

1. **IntelliJ IDEA: -**

IntelliJ IDEA is an integrated development environment (IDE) written in Java for developing computer software written in Java, Kotlin, Groovy, and other JVM-based languages. It is developed by JetBrains (formerly known as IntelliJ) and is available as an Apache 2 Licensed community edition, and in a proprietary commercial edition. Both can be used for commercial development.

**System Requirements: -**

Table 2.1: - System requirements for IntelliJ IDEA

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Windows** | **macOS** | **Linux** |
| **OS Version** | 64 Bit Microsoft Windows 8 or later | macOS 10.13 or later | Any Linux distribution that supports Gnome, KDE, or Unity DE |
| **RAM** | 2 GB RAM minimum, 8 GB RAM recommended | | |
| **Disk space** | 2.5 GB and another 1 GB for caches minimum, solid-state drive with at least 5 GB of free space recommended | | |
| **JDK Version** | Add support for Java 16 | | |
| **JRE Version** | JRE 11 is bundled. | | |
| **Screen resolution** | 1024×768 minimum screen resolution. 1920×1080 is a recommended screen resolution. | | |

**Chapter 3**

**Design**

* 1. **DFD (DATA FLOW DIAGRAM)**