

Project Management System

Ashita Wadhawan¹, Vishal Kumar², Krishan Kumar³

Department of Information Technology AKTU

Inderprastha Engineering college, Ghaziabad

Corresponding Authors' email id: vishal0709kumar@gmail.com

Abstract

Current practice uses a log book to document meetings and discussions between supervisors and students. Although a detailed schedule has been proposed, there is no monitoring process to trigger specific actions to make sure all deadlines are met. To improve project monitoring supervision, a prototype of a web-based supervision management system was designed and developed. The initial prototype consists of three modules, namely user profile, project monitoring (of software development and report writing) and appointment setting. Once the prototype is completed, we plan to conduct a user acceptance test. This information could be the general details like student name, address, performance, attendance etc or specific information related to departments like collection of data.

This portal will provide an integrated view for the project in process .This will also help to increase efficiency of students. As they have to follow strict deadlines which will help them to improve efficiency and coordination of students among each others. It will include an admin section which can be only controlled by faculty. This will also provide us with the latest update of each project simultaneously .This software also provide mentors authority of approval or disapproval before the final presentation/ final assessment. Hence it will maintain record of all the activities throughout the project development.

Keywords: *Web-based supervision management system*

I. INTRODUCTION

Project management plays an important role in coordinating the planning, design and development processes of a software project. Final year project is an appropriate avenue to introduce research to the undergraduates, since students are required to undertake, research in a specific domain and develop software as partial requirement for their final year project. Students are also required to write a dissertation as part of the course requirements. Preliminary observation has shown that for the last five years, an increasing number of students failed to complete their final year project within the allocated time. Although the role of the final year project in information technology curriculum is significant, some students (and also supervisors) fail to see its importance. The quality of the prototypes developed has also decreased.

II. METHODOLOGY

Available Methodology:

While thinking about creating a Bot, there are three routes you can take which are:

- 1) The spiral model is a risk-driven process model generator for software projects. Based on the unique risk patterns of a given project, the spiral model guides a

team to adopt elements of one or more process models, such as incremental, waterfall, or evolutionary prototyping[1].

- 2) The waterfall model is a sequential (non-iterative) design process, used in software development processes, in which progress is seen as flowing steadily downwards (like a waterfall) through the phases of conception, initiation, analysis, design, construction, testing, production/implementation and maintenance
- 3) Iterative Model - Design. Iterative process starts with a simple implementation of a subset of the software requirements and iteratively enhances the evolving versions until the full system is implemented. At each iteration, design modifications are made and new functional capabilities are added.
- 4) The V-model is an SDLC model where execution of processes happens in a sequential manner in a V-shape[4]. It is also known as Verification and Validation model.

The V-Model is an extension of the waterfall model and is based on the association of a testing phase for each corresponding development stage. The V-model is an SDLC model where execution of processes happens in a sequential manner in a V-shape. It is also known as Verification and Validation model. The V-Model is an extension of the waterfall model and is based on the association of a testing phase for each corresponding development stage.

Chosen Methodology

We are choosing V-methodology for the development because:-

- Simple and easy to use
- Testing activities like planning, test designing happens well before coding. This saves a lot of time. Hence higher chance of success over the waterfall model.
- Proactive defect tracking – that is defects are found at early stage.
- Avoids the downward flow of the defects.

- Works well for small projects where requirements are easily understood

III. PROJECT PLAN

Our project has been carefully and meticulously divided into modules each of which is challenging in one way or other. We will try to execute things In the first half of the model Verification testing activity is integrated in each phase like review user requirements, System Design document & in the next half the Validations testing activity is come in picture.

Phase 1: Verification Phase

Phase 1.1: Business Requirement Analysis

This is a very important activity and needs to be managed well, as most of the customers are not sure about what exactly they need. The acceptance test design planning is done at this stage as business requirements can be used as an input for acceptance testing.

Phase 1.2: System Design

Once you have the clear and detailed product requirements, it is time to design the complete system. The system design will have the understanding and detailing

the complete hardware and communication setup for the product under development.

Phase 1.3: Architectural Design

Architectural specifications are understood and designed in this phase. Usually more than one technical approach is proposed and based on the technical and financial feasibility the final decision is taken. The system design is broken down further into modules taking up different functionality. This is also referred to as High Level Design (HLD).

Phase 1.4: Module Design

In this phase, the detailed internal design for all the system modules is specified, referred to as Low Level Design (LLD). It is important that the design is compatible with the other modules in the system architecture and the other external systems.

Phase 1.5: Coding Phase

The actual coding of the system modules designed in the design phase is taken up in the Coding phase. The best suitable programming language is decided based on the system and architectural requirements.

Phase 2: Validation Phase

Phase 2.1: Unit Testing

Unit tests designed in the module design phase are executed on the code during this validation phase. Unit testing is the testing at code level and helps eliminate bugs at an early stage, though all defects cannot be uncovered by unit testing.

Phase 2.2: System Testing

System testing is directly associated with the system design phase. System tests check the entire system functionality and the communication of the system under development with external systems. Most of the software and hardware compatibility issues can be uncovered during this system test execution.

Phase 2.3: Acceptance Testing

Acceptance testing is associated with the business requirement analysis phase and involves testing the product in user environment. Acceptance tests uncover the compatibility issues with the other systems available in the user environment.

WEB DESIGNING



Figure 1: home page of the web portal



Figure 2: Student login page



Figure 3: Supervisor login page

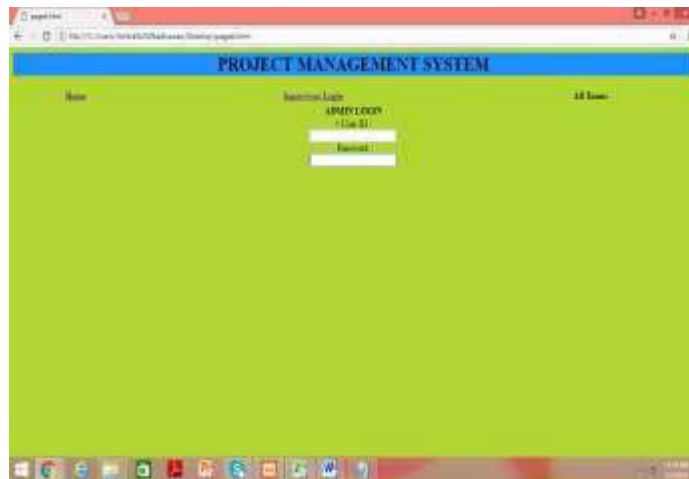


Figure 4: Admin login page

IV. ACKNOWLEDGMENT

I would like to thank individuals and organizations. I would like to extend my sincere thanks to all of them. I would like to express my gratitude towards my parents & training teachers for their kind co-operation and encouragement which help me in completion of this project.

I would like to express my special gratitude and thanks to industry persons for giving me such attention and time. My thanks and appreciations also go to my colleague in developing the project and people who have willingly helped me out with their abilities

REFERENCES

- I. [https://blog.prototypr.io/the-7-best-prototyping-tools-for-ui-and-](https://blog.prototypr.io/the-7-best-prototyping-tools-for-ui-and-ux-designers-in-2016-701263ae65e8)

[ux-designers-in-2016-701263ae65e8](https://blog.prototypr.io/the-7-best-prototyping-tools-for-ui-and-ux-designers-in-2016-701263ae65e8)

- II. [https://msdn.microsoft.com/en-us/library/s4yys16a\(v=vs.90\).aspx](https://msdn.microsoft.com/en-us/library/s4yys16a(v=vs.90).aspx)
- III. <https://dzone.com/articles/java-different-types-of-references>
- IV. https://www.tutorialspoint.com/sdlc/sdlc_v_model.htm