**Inventory Management System**



**Name: Mahesh Bist**

**NCC. Id: 00176100**

**CP Proposal**

Contents

[Overview 2](#_Toc12916615)

[Introduction to system. 2](#_Toc12916616)

[Justification of the Project. 2](#_Toc12916617)

[Scopes 3](#_Toc12916618)

[Aim of the project. 3](#_Toc12916619)

[Objectives of the project. 4](#_Toc12916620)

[Overview of the Scope. 6](#_Toc12916621)

[Development Methodology 6](#_Toc12916622)

[Methodology Used (aaryan, 2009) 6](#_Toc12916623)

[Design Pattern (Jain, 2004) 8](#_Toc12916624)

[System Architecture (Janssen, 2011) 11](#_Toc12916625)

[Scheduling 12](#_Toc12916626)

[WBS (Work breakdown Structure) (cooper, 2001) 12](#_Toc12916627)

[Milestone 13](#_Toc12916628)

[Description of Milestone: 14](#_Toc12916629)

[Gantt chart through Project Liberal. 15](#_Toc12916630)

[Scheduling/Gantt chart 16](#_Toc12916631)

[Risk Management 17](#_Toc12916632)

[Configuration Management 19](#_Toc12916633)

[Conclusion 20](#_Toc12916634)

[References 21](#_Toc12916635)

# **Overview**

## **Introduction to system.**

This will be the desktop application that I am going to develop through the programming language C#. The Project that I have selected is Inventory Management system. This is Object Oriented based software or the system that will be helpful in management of small business. It is all about the management of product, Sales, Suppliers, etc.

## **Justification of the Project.**

**Background to system.**

Businessman or the Business Staffs get so many problems to manage their business by registering the data and information about the business in the Registers. Writing the data in the Register bring up so many problems like difficult to search the data in handwriting register, need more time to register the data and the information instead of that Inventory management system helps to overcome these kinds of problems and helps to make the easy run business.

Normally Inventory Management system work by managing the data like Product, Supplier data, User data, Sales data, and others in very easier manner without any heavy effort with important time saving.

* ***Overview of the purpose system.***

The developed system will need the Desktop or the computer to access the Inventory management system. It do not support on other device then the computer or Laptops because it will made for Desktop computers or Laptops only.

The user will have to get knowledge from the developer to access the system at very first phase. In this system first of all access party have to register his detail with his username and password then only he will able to access the system for the management and can deal with the supplier, customer, he can manage the sales, auto calculate the transaction and can manage sale return through the system.

# **Scopes**

## **Aim of the project.**

The cause that we are going to build the inventory management system to increase profit in the business, managing cash flow and ensuring that end customer always has a way to get their hands on the products they want and need, it helps to provide information about the data that we have sore on the management database.

It helps to save the time that the management team of the business used to waste by writing the data and information on the register and it make easier to search the data at the required time.

## **Objectives of the project.**

The main objective of the project is to build it and to keep it secure.

For the fulfillment of the each and every aim and features we should have to manage the data of user, customer, Suppliers, Sales, Sales return, category of the product, Auto transaction option and other.

The data that we store through the management system should be able to Update, Delete, View, etc.

* **Features too be included.** (Barry, 1984)
* ***User verification or identification.***

This feature is for the identification of those user who used to access the inventory management system. It require the username and password to access the system and helps in verification.

* ***User Friendly.***

User-friendly describes a hardware device or software interface that is easy to use. The system or the software is user friendly means the user should not feel difficult while accessing the management system.

* ***Shipping.***

Having good shipping capability is extremely important to overall order fulfillment and customer satisfaction levels. Standard shipping tools typically include the ability to print product labels and barcodes.

* ***Order Management.***

This management system can help you to manage your sales orders, supplier data and information. These tools allow users to customize pricing, calculating cost, print invoice, track orders, look over sales and manage returns.

* ***Integration***

This system integrate different sector together like, user management, product management option, sale management option, supplier management option and so on.

* **Auto transaction control or calculation**

The system that we are going to develop will have the ability to calculate the total cost without using any calculator.

* **Product History.**

Product history means to store all the possible and important data in the database through the management system. It helps to store all supplier data, sales data, sales return data, etc.

* **Security.**

The system that we have to develop should be secure. Because the third person can hack the important information form the system database.

## **Overview of the Scope.**

This project will include the sales, suppliers, connection or the interaction with different company for the supply and sales of product,

This may contain the problems as well. If the system get crashed the organization or the small business will lost their important data and information which will be the greater disadvantage for the system.

# Development Methodology

## **Methodology Used** (aaryan, 2009)

Waterfall approach was first SDLC Model to be used widely in Software Engineering to ensure success of the project. In "The Waterfall" approach, the whole process of software development is divided into separate phases. In this Waterfall model, typically, the outcome of one phase acts as the input for the next phase sequentially.

The following illustration is a representation of the different phases of the Waterfall Model.



Figure 1: Waterfall Module

The sequential phases in Waterfall model are −

* **Requirement Gathering and analysis**: We collect all the requirement that are needed to develop software and then documentation is done to complete this phase.
* **System Design**: In this phase we have to design the system by studying all the requirement. This system design helps in specifying hardware and system requirements and define the overall system architecture.
* **Implementation**: In this phase each and every unit is developed and tested to know whether it is functioning or not. Unit testing is done in this phase.
* **Integration and Testing**: All the small units that are developed in the previous phase are integrated together and tested it for the functionality.
* **Deployment of system**: After completing the testing the product is deployed in the customer environment or released into the market.
* **Maintenance**: In this phase the issues that is shown by the system should overcome in present and in the coming time. As well as in this phase upgraded is done.

**Advantages**

* Simple as well as easy to understand and use.
* Due to rigidity of the module it is easy to manage. Each phase has specific deliverables and a review process.
* Phases are processed and completed one at a time.
* Clearly defined stages.
* Much understandable milestones.
* Easy task arrangement.
* Well documented process and result.

**Disadvantage**

* No working software is produce until the whole lifecycle is completed.
* It is the poorest model for long process projects.
* It is difficult to measure the progress.
* Too much risky and uncertainty.

## **Design Pattern** (Jain, 2004)

The pattern that will have to use in this system is MVC Pattern that is model, View, and Controller. This is most frequently used design pattern which helps the programmer to coordinate the allowing of the code to compartmentalize functionality by creating the barriers. The three component contained by MVC pattern are Model, View and Controller.

***Model***

* It is data related logic that holds raw data of the system.
* Interact with database (Select, Insert, Update, Delete) like MySQL Server, Oracle etc.
* It Communicate with controller.
* Sometime it can update the view.

***View***

* Usually consist of code which helps make your system interactive.
* It also communicate with controller.
* View can pass the dynamic values form the controller.
* Template Engines.

***Controller***

* It helps to receive the input from the view or URL.
* It process all the request like get, post, put, delete.
* It helps to get data from the module.
* Controller works for passing the data to the view.

**Diagram for the MVC Pattern**



Figure 2: MVC Pattern Diagram

## **System Architecture** (Janssen, 2011)

In this management system I will be used 3-tier architecture which contain three layer which are Presentation layer, Application layer and database server. Presentation layer means the PC, Tablet, Mobile, etc. and Application Layer which deals with the server.

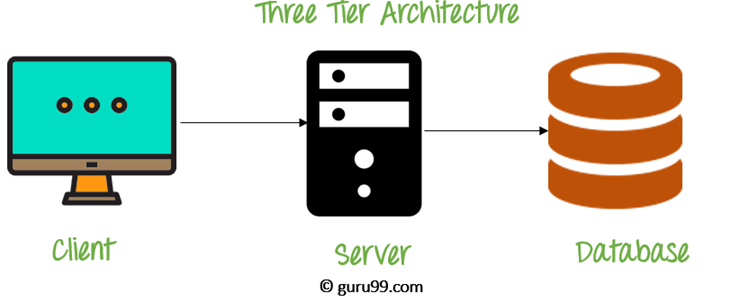
[](https://www.guru99.com/images/1/091318_0745_DBMSArchite3.png)

Figure 3: Three tier architecture.

Three tier architecture is the most popular system architecture.

**The goal of Three-tier architecture is:**

* The user application and the physical database are separated by the three-tier architecture.
* It helps to support DBMS characteristics
* Program-data independence
* It helps to support the multiple data of the system.

# **Scheduling**

## **WBS (Work breakdown Structure**) (cooper, 2001)

A work-breakdown structure is procedure where a project is sub divided into several sub units or sections reducing the complexity into several small divisions. After the breakdown of a complex problem into smaller divisions it gets too easy to work on it as different members can work on different sections and the project can be accomplished within less time.

The main objective of Work breakdown structure (WBS) is to manage time complexity for accomplishing larger and complex project in a well-managed format.

Inventory management system

Proposal

Maintenance/

Submission

Implementation

Testing

Design

\

Analysis

Use case diagram

Database coding

Unit testing

Project Plan

Improvement

UI Design

Task Management

Integration testing

Analysis specification

GUI coding

Maintainability Unitesting

Structural model

WBS

Final Documentation

Requirement

Gathering

Black box testing

Risk Management

Database design

White box testing

Team Management

Behavioral model

Architecture

Cost estimate

The diagram conveys the clear concept of WBS where a project is divided into sub-units providing clear vision of the system.

## **Milestone**

|  |  |  |
| --- | --- | --- |
| **S.N** | **Milestone** | **Date** |
| **1.** | **Proposal**  Project plan  Task management  WBS  Risk management  Team management  Cost estimation | **06/07/2019-07/01/2019** |
| **3.** | **Analysis**  Use case diagram  Analysis specification  Requirement gathering  Architecture | **07/01/2019-07/31/2019** |
| **4.** | **Design**  UI Design  Structural model  Database Design  Behavioral model | **07/31/2019-08/29/2019** |
| **5.** | **Implementation**  Database coding  GUI coding | **08/29/2019-09/20/2019** |
| **6.** | **Testing**  Unit testing  Integrated testing  Black box testing  White box testing | **09/20/2019-09/30/2019** |
| **7.** | **Maintenance/Submission**  Improvement  Maintainability  Final Documentation | **09/30/2019-10/12/2019** |

***Table: Project time estimation.***

## **Description of Milestone:**

The broad area of works are shown at higher levels where as the exact activities are seen in the lower level break downs of respective work.

* **Proposal(25 days)**
* Project plan( 5 days)
* Scope management(4 days)
* Task management( 6days)
* Risk management(4 days)
* Team management(3 days)
* Cost estimate(3 days)
* **Analysis(30 days)**
* Use case diagram(5 days)
* Analysis specification(7 days)
* Requirement gathering(12 days)
* Architecture(6 days)
* **Design(29 days)**
* UI design(8 days)
* Structural model(7 days)
* Database design(8 days)
* Behavioral design(6days)
* **Implementation(22 days)**
* Database design(9 days)
* GUI design(13 days)
* **Testing(10 days)**
* Unit testing(4 days)
* Integrated testing(2 days)
* Black box testing(2 days)
* White box testing(2 days)
* **Maintenance/Submission(12 days)**
* Improvement(3 days)
* Maintainability(5 days)
* Final documentation(4 days)

## **Gantt chart through Project Liberal.**

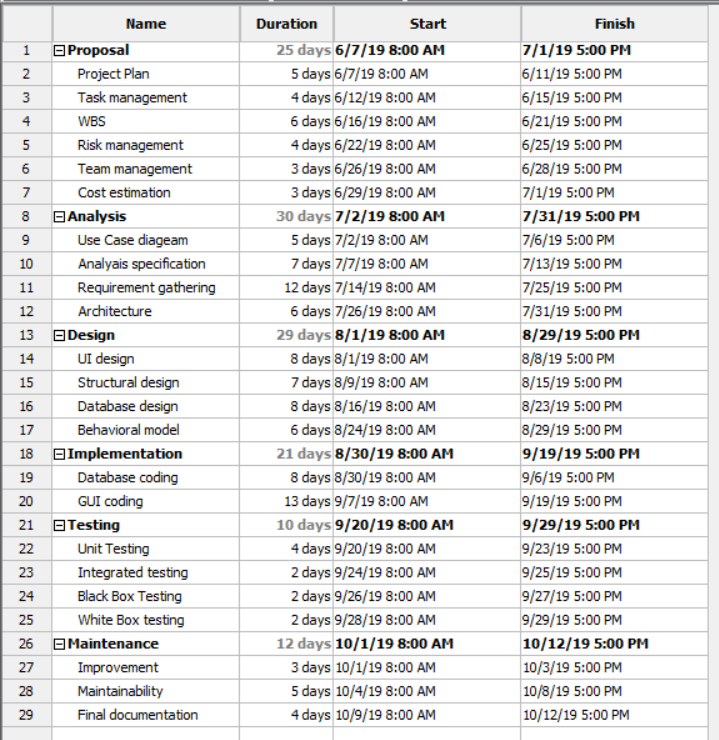
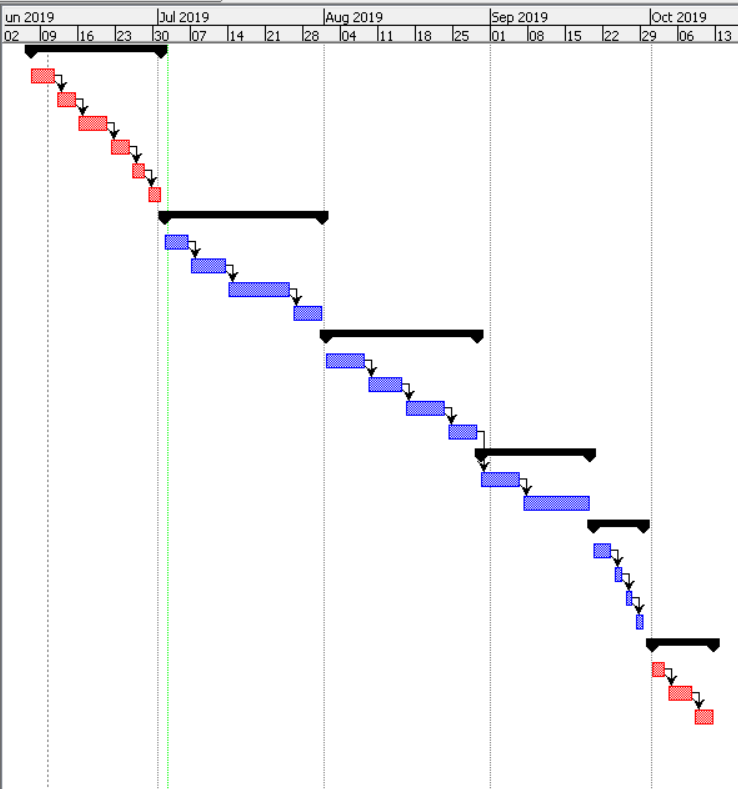


Figure 4: scheduling

**Chart**



## **Scheduling/Gantt chart**

Here the chart provides the overall information regarding the time period to accomplish certain different jobs providing the instant overview of a project.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.N** | **Name** | **Duration** | **Start** | **Finish** |
| **1.** | **Proposal** | **25 days** | **06/07/2019** | **07/01/2019** |
| 1.1 | Project Plan | 5 days | 06/07/2019 | 06/11/2019 |
| 1.2 | Task management | 4 days | 06/12/2019 | 06/15/2019 |
| 1.3 | WBS | 6 days | 06/16/2019 | 06/21/2019 |
| 1.5 | Risk management | 4 days | 06/22/2019 | 06/25/2019 |
| 1.6 | Team management | 3 days | 06/26/2019 | 06/28/2019 |
| 1.7 | Cost estimation | 3 days | 06/29/2019 | 07/01/2019 |
| **2.** | **Analysis** | **30 days** | **07/01/2019** | **07/31/2019** |
| 2.1 | Use Case diagram | 5 days | 07/02/2019 | 07/06/2019 |
| 2.2 | Analysis Specification | 7 days | 07/07/2019 | 07/13/2019 |
| 2.3 | Requirement gathering | 12 days | 07/14/2019 | 07/25/2019 |
| 2.4 | Architecture | 6 days | 07/25/2019 | 07/31/2019 |
| **3.** | **Design** | **29 days** | **08/1/2019** | **08/29/2019** |
| 3.1 | UI design | 8 days | 08/1/2019 | 08/08/2019 |
| 3.2 | Structural module | 7 days | 08/09/2019 | 08/15/2019 |
| 3.3 | Database design | 8 days | 08/16/2019 | 08/23/2019 |
| 3.4 | Behavioral module | 6 days | 08/24/2019 | 08/29/2019 |
| **4.** | **Implementation** | **21 days** | **08/30/2019** | **09/19/2019** |
| 4.1 | Database coding | 8 days | 08/30/2019 | 09/06/2019 |
| 4.2 | GUI coding | 13 days | 09/07/2019 | 09/19/2019 |
| **5.** | **Testing** | **13 days** | **09/20/2019** | **09/29/2019** |
| 5.1 | Unit testing | 4 days | 09/20/2019 | 09/23/2019 |
| 5.2 | Integrated testing | 2 days | 09/24/2019 | 06/25/2019 |
| 5.3 | Black box testing | 2 days | 09/26/2019 | 09/27/2019 |
| 5.4 | White box testing | 2 days | 09/28/2019 | 09/29/2019 |
| **6.** | **Maintenance** | **12 days** | **10/01/2019** | **10/12/2019** |
| 6.1 | Improvement | 3 days | 10/01/2019 | 10/03/2019 |
| 6.2 | Maintainability | 5 days | 10/04/2019 | 10/08/2019 |
| 6.3 | Final documentation | 4 days | 10/09/2019 | 10/12/2019 |

# **Risk Management**

Risk Management is one of the very important management for the project because in future the project may have to face different kinds of risky events like time shortage, budget shortage, hardware failure, or system may can contain threats, Requirement may will not meet, natural disaster etc. The likelihood of occurring these risk can be high, medium and low as well as consequences depend on the likelihood of risks.

**Note: Impact = Likelihood \*Consequence**

* Risk likelihood values are:

|  |  |
| --- | --- |
| **Likelihood** | **Value** |
| Low | 1 |
| Medium | 2 |
| High | 3 |

Table: Likelihood

* Risk consequence values are:

|  |  |
| --- | --- |
| **Consequence** | **Value** |
| Very low | 1 |
| Low | 2 |
| Medium | 3 |
| High | 4 |
| Very high | 5 |

Different kinds of Risks with variable Likelihood and consequences

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **R.N** | **Risk** | **Likeli-**  **hood** | **Cons**  **equence** | **Impact** | **Action** |
| 1 | Hardware Failure | 2 | 5 | 10 | Data backup should be done after the completion of certain work or the piece of work. Qualitative hardware should be use. |
| 2 | Time Limitation | 2 | 4 | 8 | Time allocation and  100% should be given. |
| 3 | Budget shortage |  |  |  | Estimation should be done in time. |
| 4 | Misdirection of project and  Project resource not met | 1 | 4 | 4 | Requirement gathering as well as frequently analysis should be done. |
| 6 | Incremental growth of project scope at any point of project causing cost or time overrun. | 1 | 5 | 5 | Remind oneself of impending deadlines using reminder application and wall calendars. |
| 7 | Natural disaster | 1 | 5 | 5 | Backup entire project directory regularly in google drive. |

***Table: Consequences***

# **Configuration Management**

(mandon, 1998)

A configuration is the set of characteristics that define a final product or deliverable. This includes all functional and physical specifications. Physical specifications may include the color, size, weight, shape, and materials.

**Five key steps for the configuration management.**

* Planning should be done to manage record, track and to audit configuration.
* All configuration requirement on the project should be identify and recorded.
* The configuration must be assessed, approved, and documented. This is normally done with the project change control process.
* Project Audit should be done. This include any tests to prove that the product conform to the configuration requirement.

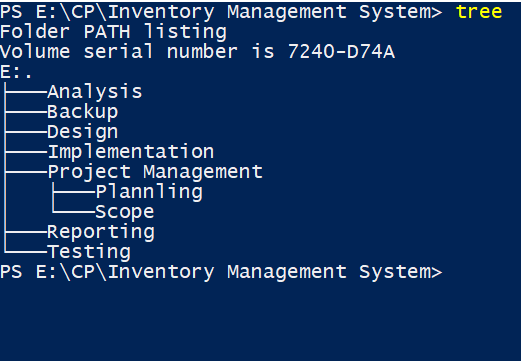


Figure 5: Configuration management

# **Conclusion**

So, in conclusion an inventory management system is a system designed to help businesses automate their inventory management, whether the inventory is the company’s assets or merchandise. These management system is used by both large and small business.

In short, whole process contain every description about the project that we are going to develop. For the project development I am going to use the MVC pattern with waterfall design model and 3-tier architecture.

# **References**

aaryan, s., 2009. *www.tutorialspoint.com.* [Online]   
Available at: https://www.tutorialspoint.com/sdlc/sdlc\_waterfall\_model.htm

Barry, B., 1984. *www.fcbco.com.* [Online]   
Available at: https://www.fcbco.com/blog/top-6-features-to-look-for-in-an-inventory-management-system

cooper, d., 2001. *www.workbreakdownstructure.com.* [Online]   
Available at: https://www.workbreakdownstructure.com

Jain, S., 2004. *www.geeksforgeeks.org.* [Online]   
Available at: https://www.geeksforgeeks.org/mvc-design-pattern/

Janssen, D., 2011. *www.techopedia.com/.* [Online]   
Available at: https://www.techopedia.com/definition/24649/three-tier-architecture

mandon, j., 1998. *www.wrike.com.* [Online]   
Available at: 1. https://www.wrike.com/project-management-guide/faq/what-is-configuration-management-in-project-management/