**GRANITE BUSINESS MANAGEMENT SYSTEM**

**ABSTRACT**

A Granite management system is an main goal is to for managing the granite factory. A very risk to managing the granite factory manually so we have a create a software which handles reduce the man power work. An administrator have an username and password, after the login the administrator added the customers details and view the customer information. We had to handle the purchase details when they purchase the material even view the purchase details as well. When the customer purchase some items they should make an entry about the purchase details in the sales module then we have to calculate the stock and billing details. We can check the billing details about the granite factory.

**1. INTRODUCTION**

The main objective of this project is for managing granite factory and maintain the product material details, example for know about the purchase,sales,billing and customer details. The project is aimed to develop by JAVA as Front end and ORACLE as Back end.

**1.1 SYSTEM SPECIFICATION**

**1.1.1 HARDWARE SPECFICATION:**

* Processor : P 4 700 GHz.
* RAM : 4 GB RAM
* Hard Disk Drive : 180 GB

**1.1.2 SOFTWARE SPECIFICATION**

* Operating System : Windows 7/8/10
* Front End : JAVA
* Back End : ORACLE

1. **SYSTEM STUDY**

**2.1 EXISTING SYSTEM:**

An existing system of granite management system is a manually we can check the quantity of material count. It takes a too man power works to complete the task. We just count the stock report as a manually and which does not give an accurate result, it’s just getting irritation to the people who are doing this task. A material purchase and sales details can’t handle this system, but in this system.**2.1.1 DRAWBACKS:**

The existing system has the following drawbacks.

* It’s very risky to manage granite system
* Man power work impossible.
* Used only in low level business

**2.2 PROPOSED SYSTEM:**

A proposed system of granite management system is that for we can takes a purchase, sales, stock and billing module have to be covered. So we no need to bother about the product. In this system we can build it to anywhere. This system have to be working fully automatically.

**2.2.1 FEATURES:**

* Showing a billing report
* very easy to find the stock details
* contact with customers

1. **SYSTEM DESIGN AND DEVELOPMENT**

**3.1 FILE DESIGN**

The selection of the file system design approach is done according to the needs of the developers what are the needed requirements and specifications for the new design. It allowed us to identify where our proposal fitted in with relation to current and past file system development. Our experience with file system development is limited so the research served to identify the different techniques that can be used. The variety of file systems encountered show what an active area of research file system development is. The file systems may be from one of the two fundamental categories. In one category, the file system is developed in user space and runs as a user process. Another file system may be developed in the kernel space and runs as a privileged process. Another one is the mixed approach in which we can take the advantages of both aforesaid approaches. Each development option has its own pros and cons. In this article, these design approaches are discussed.

**3.2 INPUT DESIGN**

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy. Input Design considered the following things:’

* What data should be given as input?
* How the data should be arranged or coded?
* The dialog to guide the operating personnel in providing input.
* Methods for preparing input validations and steps to follow when error occur.

**OBJECTIVES**

* Input Design is the process of converting a user-oriented description of the input into a computer-based system. This design is important to avoid errors in the data input process and show the correct direction to the management for getting correct information from the computerized system.
* It is achieved by creating user-friendly screens for the data entry to handle large volume of data. The goal of designing input is to make data entry easier and to be free from errors. The data entry screen is designed in such a way that all the data manipulates can be performed. It also provides record viewing facilities.
* When the data is entered it will check for its validity. Data can be entered with the help of screens. Appropriate messages are provided as when needed so that the user
* will not be in maize of instant. Thus the objective of input design is to create an input layout that is easy to follow

**3.3 OUTPUT DESIGN**

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system’s relationship to help user decision-making.

1. Designing computer output should proceed in an organized, well thought out manner; the right output must be developed while ensuring that each output element is designed so that people will find the system can use easily and effectively. When analysis design computer output, they should Identify the specific output that is needed to meet the requirements.

2. Select methods for presenting information.

3. Create document, report, or other formats that contain information produced by the system.

The output form of an information system should accomplish one or more of the following objectives.

* Convey information about past activities, current status or projections of the
* Future.
* Signal important events, opportunities, problems, or warnings.
* Trigger an action.
* Confirm an action.

**3.4 DATABASE DESIGN**

Today's businesses depend on their databases to provide information essential for day-to-day operations, especially in case of electronic commerce businesses who has a definite advantage with up-to-date database access. Good design forms the foundation of any database, and experienced hands are required in the automation process to design for optimum and stable performance.

Software Solutions have been constantly working on these platforms and have attained a level of expertise. We apply proven methodologies to design, develop, integrate and implement database systems to attain its optimum level of performance and maximize security to meet the client's business model.

### Business needs addressed:

* Determine the basic objects about which the information is stored
* Determine the relationships between these groups of information and the objects
* Effectively manage data and create intelligent information
* Remote database administration or on site administrative support
* Database creation, management, and maintenance
* Information retrieval efficiency, remove data redundancy and ensure data security

**3.5 SYSTEM DEVELOPMENT**

**3.5.1 DESCRIPTION OF MODULES**

1. Customer Registration Module
2. Purchase Module
3. Sales Module
4. Billing Module
5. Stock Module

**1.Customer Registration**

A customer registration module is used collect the information details. Which is used to when the granite shop owner contact the customer this will help to give the contact details about the customer.

**2.Purchase Module**

when the manager purchasing the products or material they should register this purchase module. It’s managed by shop owner then collect the information about the product details and price etc… After the purchasing only should sales the product.

**3.Sales Module**

This module calling when the customer have to be purchased and before billing. It will show the all the details about the product. Which is basically used to find the stock entry details. A sales module collect the customer information so we can contact the customer easily.

**4. Stock module**

This Stock module will be helps to find the what are all the products are available in shop and how many products are in stock also we can check. No need to manually count a check and verify the product. Systematically it will calculate and give an report.

**5. Billing Module**

This module shows an billing detail, what are the items are sales in date wise and how much are they gain. Total entire application will be showing in this module.

1. **SYSTEM TESTING AND IMPLEMENTATION**

**SYSTEM TESTING**

System testing is the process of exercising software with the intent of finding and ultimately correcting errors. This fundamental philosophy does not change for web applications, because Web-based systems and application reside on a network and interoperate with many different operating system, browsers, hardware platforms, and communication protocols; the search for errors represents a significant challenge for web application.

The distributed nature of client\server environments, the performance issues associated with transaction processing, the potential presence of a number of different hardware platforms, the complexities of network communication, the need to serve multiple clients from a centralized database and the requirements imposed on the server all combine to make testing of client\server architectures.

System testing is actually a series of different tests whose primary purpose is to fully exercise the computer based system. System testing is the state of implementation that is aimed at assuring that the system works accurately and efficiently. Testing is the vital to the success of the system. System testing makes the logical assumption that if all the parts of the system are correct, the goal will be successfully achieved.

**The objective of testing is as follows:**

* + Testing is the process of executing a program with the intent of finding an error.
  + A successful test is that one of the cover of undiscovered error.

### TESTING ISSUES

* Client GUI considerations
* Target environment and platform diversity considerations
* Distributed database considerations
* Distributed processing considerations

**TESTING METHODOLOGIES**

System testing is state of implementation, which is aimed at ensuring that the system works accurately and efficiently as expect before live operation commences. It certifies that the whole set of programs hang together.

System testing requires a test plan that consists of several key activities and step for run program, string, system and user acceptance testing. The implementation of newly designed package is important in adopting a successful new system

Testing is the important stage in software development. the system test in implementation stage in software development process. The system testing implementation should be confirmation that all is correct and an opportunity to show the users that the system works as expected. It accounts the largest percentage of technical effort in the software development process.

Testing phase in the development cycle validates the code against the functional specification testing is vital to achievement of the system goals. The objective of the testing is to discover errors to fulfills this objective a series of test step unit, integration. validation and system tests were planned and executed the test steps are:

**System Testing**

Testing is an important phase in project development. System testing makes a logical assumption that if all parts of the system are correct, and the goal will be achieved successfully. The software must meet the user specification and it must satisfy according to the needs of the users.

Testing is the process of executing a project within the intend of finding errors. A good test case is one that has a high probability of finding an undiscovered error.

**Unit Testing**

Unit testing focuses verification efforts on the smallest unit of software design of the module. This is also known as “module testing”. This testing is carried out during programming stage itself. In this testing step, each module is found to be working satisfactorily as regards to the expected output of the modules.

**In Project**, Each module such customer registration module, branch module, service details module, billing module, vehicle module and customer detail modules are tested individually for example, Customer details module can contain the more forms to maintain the information so all forms could be tested like entered information store appropriately in database access page or not. If correctly accessed means the testing of registration module successfully completed. Likewise all modules are tested successfully.

**Integration Testing**

Data can be lost across an interface, one module can have adverse effect on another sub function when combined it may not produce the desired major functions. Integration testing is a systematic testing for constructing test to uncover errors associated within an interface.

The objectives taken from unit tested modules and a program structure is built for integrated testing. All the modules are combined and the test is made.

A correction made in this testing is difficult because the vast expenses of the entire program complicated the isolation of causes. In this integration testing step, all the errors are corrected for next testing process.

**In Project,** Integration of two modules can be tested together such as customer registration details and customer login module for verification purposes providing proper accessibility to users. The communication of Registration and Login module can test and executed successfully.

**Validation Testing**

After the completion of the integrated testing, software is completely assembled as a package; interfacing error has been uncovered and corrected and a final series of software test validation begins.

Validation testing can be defined in many ways but a simple definition is that validation succeeds when the software function in a manner that can be reasonably expected by the customer. After validation test has been conducted, one of two possible conditions exists:

**In this project,** Admin login details form Enter without username and password in textbox enter the submit button then Login failed message otherwise checks the both textbox value that is true means valid page displayed. Enter Password Displaying password character \*.if it displays the characters security is not availed so testing of software is failed.

**Output Testing**

The next process of validation testing, is output testing of the proposed system, since no system could be successful if it does not produce the required output in the specified format. Asking the user about the format required, list the output to be generated or displayed by the system under considerations.

Output testing is a different test whose primary purpose is to fully exercise the computer based system although each test has a different purpose all the work should verify that all system elements have been properly integrated and perform allocated functions.

The output format on the screen is found to be corrected as the format was designed in the system design phase according to the user needs for the hard copy also; the output testing has not resulted in any correction in the system.

**In project** All the forms are tested as it gives the necessary output to the user’s search such as view response details.

1. **CONCLUSION**

In this granite shop management system we can managing the purchase, sales and stock details. Very easy to handle and maintain stock details entry. Which is commonly used to maintain the furniture shops.

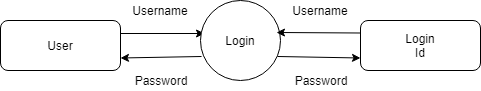
**BIBLIOGRAPHY**

* **CH97**  
  Gary Cornell and Cay S. Horstmann, Core Java, second ed., SunSoft Press, 1997
* **Gea99b**  
  David M. Geary Graphic Java 2: Mastering the JFC, vol. II, Swing, third ed., Sun Microsystems Press, 1999.
  + **Gea99c**  
    David M. Geary Graphic Java 2: Mastering the JFC, vol. III, Advanced Swing, third ed., Sun Microsystems Press, 1999(?).

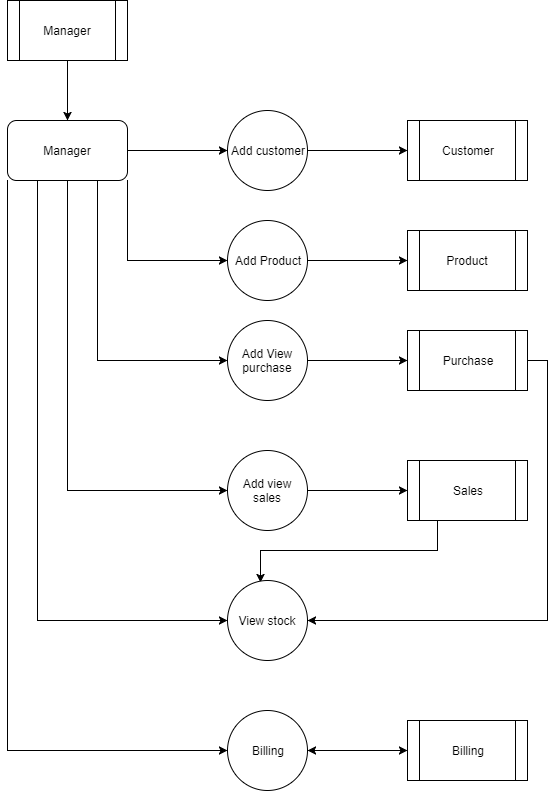
**APPENDICES**

1. **DATA FLOW DIAGRAM**

LEVEL 0:



LEVEL 1:



B. **TABLE STRUCTURE**

**TABLE NAME : manager**

**PRIMARY\_KEY : manager\_id**

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD** | **DATA TYPE** | **SIZE** | **DESCRIPTION** |
| Manager\_id | Integer |  | Manager id |
| Username | Varchar | 30 | Manager username |
| password | Varchar | 30 | Manager password |

**TABLE NAME : CUSTOMER**

**PRIMARY KEY : cid**

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD** | **DATA TYPE** | **SIZE** | **DESCRIPTION** |
| Cid | Integer | 10 | Customer id |
| Name | Varchar | 30 | Customer name |
| Address | Varchar | 30 | Address |
| Contactno | Varchar | 10 | Contact number |
| Gender | Varchar | 10 | Gender |
| Email | Varchar | 10 | eamil |

**TABLE NAME : PRODUCT**

**PRIMARY KEY : product\_id**

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD** | **DATA TYPE** | **SIZE** | **DESCRIPTION** |
| Product\_id | Integer | 10 | Product id |
| Company | Varchar | 10 | Company name |
| Product type | Varchar | 30 | Product type |
| Weight | Integer | 10 | weight |
| Price | Integer | 10 | price |

**TABLE NAME : PURCHASE**

**PRIMARY KEY : purchase\_id**

**FOREIGN KEY : product\_id**

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD** | **DATA TYPE** | **SIZE** | **DESCRIPTION** |
| Purchase\_id | Integer | 10 | Purchase id |
| Product\_id | Integer | 10 | Product id |
| Quantity | Integer | 10 | quantity |
| Price | Integer | 10 | price |

**TABLE NAME : SALES**

**PRIMARY KEY : sales \_id**

**FOREIGN KEY : product\_id**

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD** | **DATA TYPE** | **SIZE** | **DESCRIPTION** |
| sales\_id | Integer | 10 | Sales id |
| Product\_id | Integer | 10 | Product id |
| Quantity | Integer | 10 | quantity |
| Price | Integer | 10 | price |

**TABLE NAME : STOCK**

**PRIMARY KEY : stock \_id**

**FOREIGN KEY : product\_id**

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD** | **DATA TYPE** | **SIZE** | **DESCRIPTION** |
| stock\_id | Integer | 10 | stock id |
| Product\_id | Integer | 10 | Product id |
| Quantity | Integer | 10 | quantity |
| Price | Integer | 10 | price |

**TABLE NAME : BILLING**

**PRIMARY KEY : billing \_id**

**FOREIGN KEY : product\_id,sales\_id**

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD** | **DATA TYPE** | **SIZE** | **DESCRIPTION** |
| billing\_id | Integer | 10 | stock id |
| Product\_id | Integer | 10 | Product id |
| Sales\_id | Integer | 10 | Sales id |
| Quantity | Integer | 10 | quantity |
| Price | Integer | 10 | price |

**C. SAMPLE CODING**