Final Project Report

by

Dilpreet Singh

# ABSTRACT

This project is aimed at developing a Web based application named Stock Management System for managing the Stock of any organization. The Stock Management System (SMS) refers to the system and processes to manage the stock of organization with the involvement of Technology system. This system can be used to store the details of the Stock, stock maintenance, update the Stock based on the sales details, generate sales and Stock report daily or weekly based. This project is categorize individual aspects for the sales and Stock management system. In this system we are solving different problem affecting to direct sales management and purchase management. Stock Management System is important to ensure quality control in businesses that handle transactions resolving around consumer goods. Without proper Stock control, a large retail store may run out of stock on an important item. A good Stock management system will alert the wholesaler when it is time to record. Stock Management System is also on important means of automatically tracking large shipment. An automated Stock Management System helps to minimize the errors while recording the stock.

# CHAPTER-1: INTRODUCTION

## Introduction to Stock Management System

The project Stock Management System is a complete Web based application designed on MVC of .Net technology using Visual Studio Software. The main aim of the project is to develop Stock Management System Model software in which all the information regarding the stock of the organization will be presented. It is an internet based Web application which has admin component to manage the Stock and maintenance of the Stock system.

This Web application is based on the management of stock of an organization. The application contains general organization profile, sales details, Purchase details and the remaining stock that are presented in the organization. There is a provision of updating the Stock also. This application also provides the remaining balance of the stock as well as the details of the balance of transaction.

Each new stock is created and entitled with the named and the entry date of that stock and it can also be update any time when required as per the transaction or the sales is returned in case. Here the login page is created in order to protect the management of the stock of organization in order to prevent it from the threads and misuse of the Stock.

## Literature Review

Products are considered as the business resources for the organization. This includes managing the product with appropriate way to review any time as per the requirement. Therefore it is important to have a computer based SMS which has the ability to generate reports, maintain the balance of the stock, details about the purchase and sales in the organization. Before developing this application we came up with 2 Stock Management System existing in the market, which helps to give the knowledge for the development of our project. These application software are only used by the large organization but so we came up with the application which can be used by the small company for the management of their stock in the production houses.

After analyzing the other Stock management system we decided to include some of common and key features that should be included in every Stock management system. So we decided to include those things that help the small organization in a way or other.

## Problem Statement

After analyzing many existing SMS we have now the obvious vision of the project to be developed. Before we started to build the application team had many challenges. We defined our problem statement as:

* To make Web based application of SMS for small organization.
* To make the system easily managed and can be secured.
* To cover all the areas of SMS like purchase details, sales details and stock management.

## Objective of the Project

* To develop an application that deals with the day to day requirement of any production organization
* To develop the easy management of the Stock
* To handle the Stock details like sales details, purchase details and balance stock details.
* To provide competitive advantage to the organization.
* To provide details information about the stock balance.
* To make the stock manageable and simplify the use of Stock in the organization.

## Features of Project

This application is used to show the stock remaining and details about the sales and purchase. The components developed in the project are described below:

**Home Page :-** this is the main page which will display as the project will open in the browse.

**Client page: -** This is the page through which entry of the client can be maintained. This is the page having creation, edition, deletion of the records of client.

**Company Page :**- By this page we can do the maintenance of the company’s information

**Stock\_in :**- through this page a stock in hand can be maintained

**Sales –** this is a page which will enter the sales and maintain it through out the program

**Payments Page :**- is for the entry of the details of the payment made by the client and company

**Stock\_out :** this page maintain the stock which is send against the order of the client

**Stock\_return:**- take the information about the return of sold stock.

**Bill Payment :**- this is a page which maintain the payment of the cill during the transaction.

**Register –** this tab is used to register a new user.

## Scope of the Application

Stock Management System (SMS) is targeted to the small or medium organization which doesn’t have many godwom or warehouses i.e. only to those organization that has single power of authority. Some of the scope are:

* Only one person is responsible in assigning the details or records.
* It is security driven.
* Company / client can be added as per the requirement.

# CHAPTER-2 BACKGROUND KNOWLEDGE

## Architectural Review

This Web based application is based on 3-tier architecture of .Net Framework. The 3-tier includes(Model, View and controller) the three hierarchy of the flow of programming logic from user interface to database and again database to user interface with the desired information requested by the clients. In between there involves the logic layer for effectively and correctly manipulating the request. The 3-tier includes the following:

### Client tier

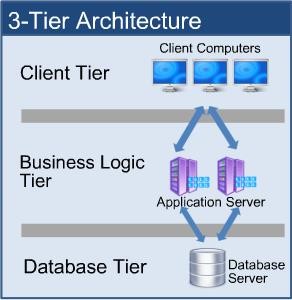
The visual part is implemented using all kinds of swing components, which does not make database calls. The main function of this tier is to display information to the user upon user’s request generated by user’s inputs such as firing button events. For example, Stock list will display when user click “display” button if he or she wants to know the list of stock remaining in the organization.

### Business tier

The middle tier, business logic, is called by the client to make database queries. It provides core function of the system as well as connectivity to the data tier, which simplify tasks that were done by the clients tier.

### Data tier

Data layer is also the class which gets the data from the business tier and sends it to the database or gets the data from the database and sends it to business tier. This is the actual DBMS access layer or object layer also called the business object. The database backend stores information which can be retrieved by using the mysql database Connectivity. Mysql database connectivity is used to manage the commi=unication between the middle tier and the backend database by issuing complex database queries.



### Figure 1: Tier Architecture

Source**:** <http://www.dbtalks.com/UploadFile/a7e1c8/3-tier-clientserver-architecture/>

## Database Theory

A database is a collection of information that is organizes so that it can easily be accessed, managed and updated. In one view, database can be classified according to types of content: bibliography, full-text, numeric, and image. In computing, database are sometime classified according to their organizational approach. A distributed database is one that can be dispersed or replicated among different points in a network.

### Relational Database

SMS has the relational database model. A relational database is a digital database whose organization is based on the relational model of data. This model organizes data into one or more tables of rows and columns. These tables here have the relation. The relation is maintained by the unique key defined in each row. The key can be primary and foreign depending on their nature of connection. The standard user and application program interface to a relational database is the structured query language (SQL). SQL statement are used both for interactive queries for information from relational database and for gathering data for reports.

### Primary Key

The primary key of a relational table uniquely identifies each record in the table. It can either be a normal attribute that is guaranteed to be unique or it can be generated by the DBMS. A primary key’s main features are:

* It must contain a unique value for each row of data.
* It cannot contain null value.

### Foreign Key

A foreign key is a column or group of column in a relational database table that provides a link between data in two tables. In foreign key reference, a link is created between two tables when the column or columns that hold the primary key value for one table are referenced by the column or column in another table thereby establishing a link between them. Creating a foreign key manually includes the following advantages:

* Changes to primary key constraints are checked with foreign key constraints in relation table.
* An index enables the Database Engine to quickly find related data in the foreign key tables.

### Structured Query Language (SQL)

The structured Query language (SQL) is the set of instructions used to interact with a relational database. In fact, SQL is the only language the most database actually understand. Whenever you interact with such a database, the software translates your commands into SQL statement that the database knows how to interpret. SQL has three major Components:

* Data Manipulation Language (DML)
* Data Definition Language (DDL)
* Data Control Language (DCL)

## ACID Property

Every database transaction obeys the following rules:

**Atomicity –** Either the effects of all or none of its operation remain (“all or nothing” semantics) when a transaction is completed (committed or aborted respectively). In other words, to the outside world a committed transaction appears (by its effects on

the database) to be indivisible, atomic, and an aborted transaction does not leave effects on the database at all, as if never existed.

**Consistency –** every transaction must leave the database in a consistent (correct) state, i.e., maintain the predetermined integrity rules of the database (constraints upon and among the database’s objects). A transaction must transform a database from one consistent state to another consistent state (however, it is the responsibility of the transaction’s programmer to make sure that the transaction itself is correct, i.e., performs correctly what it intends to perform (from the application’s point of view) while the predefined integrity rules are enforced by the DBMS). Thus since a database can be normally changed only by transactions, all the database’s states are consistent. An aborted transaction does not change the database state it has started from, as if it never existed (atomicity above).

**Isolation –** Transactions cannot interfere with each other (as an end result of their executions). Moreover, usually (depending on concurrency control method) the effects of an incomplete transaction are not even visible to another transaction. Providing isolation is the main goal of concurrency control.

**Durability –** Effects of successful (committed) transactions must persist through crashes (typically) by recording the transaction’s effects and its commit event in a non-volatile memory.

# CHAPTER-3: ANALYSIS AND DESIGN

## Background Research

We started research by identifying the need of SMS in the organization. Initially we bounded our research to find the general reasons that emerged the needs of Stock Management System. We used different techniques to collect the data that can clearly give us the overall image of the application. The techniques we used were interview with the developers, visiting online websites that are presented as the templates and visiting some organization to see their SMS application. Basically the following factors forced us to develop SMS application:

* Cost and affordability
* Lack of stock management.
* Effective flow of stock transfer and management.
* Difficulty in monitoring the stock management.

## Requirement Analysis

We collected a number of requirements for project from our primitive research, website visits, and interview to the concerned personnel and their experiences regarding the concepts of its development. We have even visited some organization in Kathmandu valley and analyze its importance and try to develop the project by fulfilling all the weakness that were found in the application. We then decided to bulid same type of application with different logic flow and new language which will be suitable for the small organization.

## SMS Requirement

The goal for the application is to manage the Stock management function of the organization. Once it is automated all the functions can be effectively managed and the organization can achieve the competitive advantage. Business requirement are discussed in the Scope section, with the following additional details:

* Helps to search the specific product and remaining stock.
* Details information about the product sales and purchase.
* Brief Information of the organization todays status in terms of news, number of present Stock as per the date entered.
* It helps to identify the total presented Stock in the company.
* To know the balance and details of sales distributed in specific date.
* There is proper transaction management of Stock.
* All transaction have specific entry date along with quantity and rate.
* Only admin can login in the page.

## Users Requirement

User requirement are categorized by the user type

### Admin

* Able to create new godwom along with date.
* Able to edit the entry as per entry.
* Able to add, modify and delete the stock entry.

### Stock management

* Able to check the stock available.
* Able to check the balance payment.
* Able to view the remaining sales stock.

## Feasibility Analysis

This software has been tested for various feasibility criterions from various point of views.

### Economic Feasibility

The system is estimated to be economically affordable. The system is medium scale Web application and has affordable price. The benefits include increased efficiency, effectiveness, and the better performance. Comparing the cost and benefits the system is found to be economically feasible.

### Technical Feasibility

Development of the system requires tools like:

* Visual Studio 2015
* .NET Framework 4.5
* Microsoft SQL server 2008, etc

Which are easily available within the estimated cost and schedule.

### Operational Feasibility

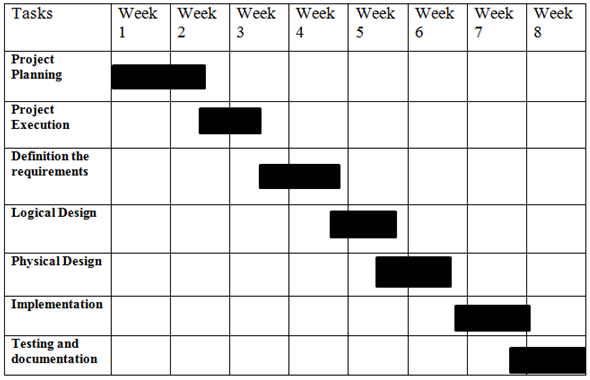
The system provides better solution to the libraries by adding the typical requirement and necessities. The solution provided by this system will be acceptable to ultimate solution for the stock management.

### Schedule Feasibility

The organized schedule for the development of the system is presented in the schedule sub-section. The reasonable timeline reveals that the system development can be finished on desired time framework.

## Gantt chart

It is one of the popular way to illustrate project schedule. A Gantt chart is a graphical representation of a project that shows each activity task as a horizontal bar whose length is proportional to its time for completion. A Gantt chart for the project deliverables within time frame. This project Gantt chart is shown below:

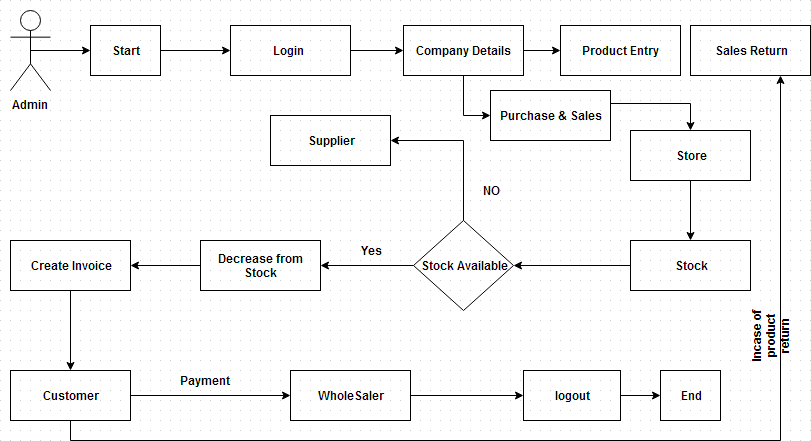


**Figure 2: Gantt chart**

# CHAPTER – 4: SYSTEM DESIGN

## Process Flow Diagram

Process Flow Diagram or Flowchart is a diagram which uses geometric symbols and arrows to define the relationships. It is a diagrammatic representation of the algorithm. The Process flow Diagram of our application is shown below:



**Figure 3 : SMS Process flow diagram**

## Use Case Diagram

Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors and their goals.

The main purpose of a use case diagram is to show what system functions are performed for which actors.

### Diagram Building Block Use cases

A use case describes a sequence of actions that provide something of measurable value to an actor and is drawn as a horizontal ellipse.

### Actor

An actor is a person, organization or external system that plays a role in one or more interactions with the system

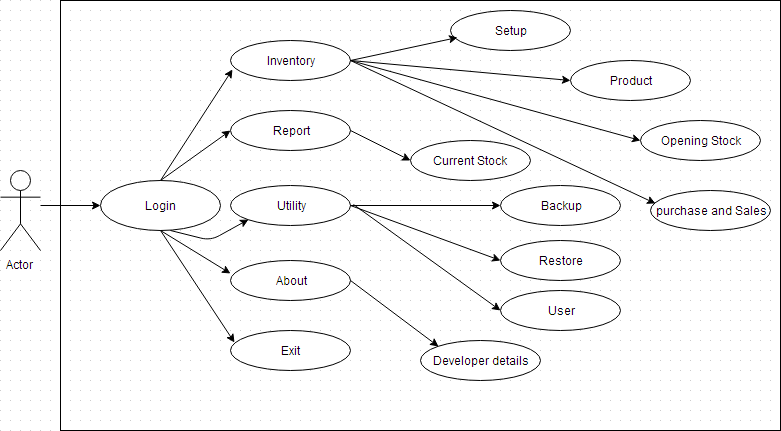
### System boundary boxes (optional)

A rectangle is drawn around the use case called the system boundary box to indicate scope of the system.

### Actor



**Use case**



**Figure 4 : SMS Use Case Diagram**

# CHAPTER – 5: TOOLS AND TECHNOLOGY USED

## Development Tools

### Microsoft visual Studio

Microsoft Visual Studio is an integrated development environment (IDE) from Microsoft. It is used to develop console and graphical user interface applications along with Windows Form applications, websites, web applications, and web services in both native code together with managed code for all platforms supported by Microsoft Window, Windows Mobile, Windows CE, .NET Framework, .NET Compact Framework and Microsoft Silverlight. Microsoft Visual Studio simplifies the basic tasks of creating, debugging and deploying applications.

Microsoft Visual Studio comes with .NET Framework and supports applications targeting Windows. It supports IBM DB2 and Oracle databases, in addition to Microsoft SQL Server. It has integrated support for developing Microsoft Silverlight applications, including an interactive designer. Microsoft Visual Studio offers several tools to make parallel programming simpler: in addition to the Parallel Extensions for the .NET Framework and the Parallel Patterns Library for native code, Visual Studio includes fools for debugging parallel applications.

The Visual Studio code editor now highlights references; whenever a symbol is selected; all other usages of the symbol are highlighted. It also offers a Quick Search feature to incrementally search across all symbols in C++, C# and VB.NET projects. Quick Search supports substring matches and camel Case searches. The Call Hierarchy feature allows the developers to see all the methods that are called from a current method as well as the methods that call the current one. IntelliSense in Visual Studio supports a consume-first mode which developers can opt into. In this mode, IntelliSense will not auto-complete identifiers; this allows the developer to use undefined identifiers (like variable or method names) and define those later. Visual Studio can also help in this by automatically defining them, if it can infer their types from usage.

We have used Visual Studio Community 2015, v 14.0.23107.10 for developing the Stock Management System Application.

### Microsoft SQL server Management Studio Express

Microsoft SQL Server Management Studio Express (SSMSE) provides a graphical management tool for SQL Server Express Edition. SSMSE user interface is a subset of SQL Management Studio that is available with other editions of SQL Server. SSMSE call also manage instance of the SQL Server Database Engine created by any edition of SQL Server. Stock Management System is developed using Microsoft SOL Server 2008.

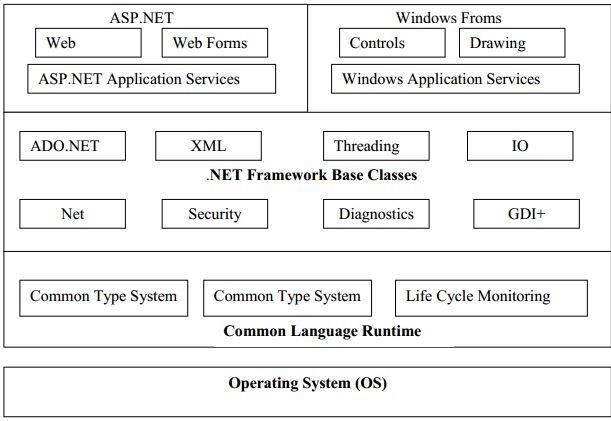
### .NET Framework 4.5

The .NET Framework is a development platform for building apps for Windows, Windows Phone, Windows Server, and Microsoft Azure. It consists of the common language runtime (CLR) and the .NET Framework class library, which includes classes, interfaces, and value types that support an extensive range of technologies. The .NET Framework provides a managed execution environment, simplified development and deployment, and integration with a variety of programming languages, including Visual Basic and Visual C#.

## Technology Used

### .NET Framework Structure

The .Net architecture is basically segregated in to three layers namely top, middle and bottom layer. The bottom layer is CLR, it is the heart of .NET Framework which provides the runtime environment in which programs are executed. The middle layer comprises the next generation of standard system services are brought under the control of the framework, making them universally available and standardizing their usage across languages. The top layer includes user and program interfaces as figure:



### Figure 5: .Net Framework Architecture

The first thing that you should notice when looking at this diagram is that the .NET Framework sits on top of the operating system. There has also been a lot of talk about

.NET being ported over by some third-party companies so that a majority of the .NET Framework could run on other platforms as well.

At the base of the .NET Framework is the Common Language Runtime (CLR). The CLR is the engine that manages the execution of the code. The next layer up is the

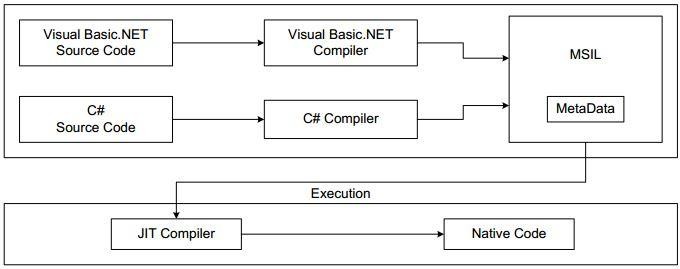
.NET Framework Base Classes. This layer contains classes, value types, and interfaces that you will use often in your development process. Most notably within the .NET Framework Base Classes is ADO.NET, which provides access to and management of data.

The third layer of the framework is ASP.NET and Windows Forms. ASP.NET should not be viewed as the next version of Active Server Pages after ASP 3.0, but as a dramatically new shift in Web application development. Using ASP.NET, it’s now possible to build robust Web applications that are even more functional than Win32 applications of the past.

### Compilation to Manage Code

Code that is compiled and targeted to the CLR is known as managed code. Managed code provides metadata that is needed for the CLR to provide the services of multi-

language support, code security, object lifetime management, and memory management. The .NET Framework requires that you use a language compiler that is targeted at the CLR, such as the Visual Basic .NET, C#, C++ .NET, or Jscript .NET compilers provided by Microsoft. So how does the code that you typed into Visual Studio .NET become the code that the user receives when he is using your application? It is fairly simple and straightforward. Figure below shows a diagram of the compilation process.



### Figure 6: Compilation to managed code

Source: Teacher’s handout

The IL is CPU-independent. This means that IL code is not reliant on the specific computer that generated it. In other words, it can be moved from one computer to another (as long as the computer supports the .NET Framework) without any complications. This is what makes XCopy, or just copying over the application, possible.

### JIT compilation

The .NET Framework contains one or more JIT compilers that compile your IL code down to machine code. This is done when the application is executed for the first time.

### The .NET Language

In the past, you chose the development language for an application based upon the functionality that you were trying to perform. Some languages were more powerful than others, but at the same time they might have required a higher level of understanding and were, in most cases, more difficult to program in.

Now the .NET Framework provides you with a language-independent programming platform. You do not have to decide which language would provide a better solution. All languages are now on a level playing field. In .NET, no one language is superior to any of the other languages. They all have equal access to everything that .NET offers.

To be part of the .NET Framework, a language only has to follow certain rules. The biggest and most important rule for inclusion is that the language needs to be an object-oriented language. Microsoft provides four languages with the .NET Framework:

Visual Basic .NET C#

C++.NET and Jscript .NET.

Microsoft also provides J# (pronounced J-sharp), but in order to use this new language that is basically Java for .NET, you need to download the language to install it on your server.

### Data Provider

The data provider is responsible for providing and maintaining the connection to the database. A database provider is a set of related components that works together to provide in an efficient and performance driven manner. Each Data provider consists of the following components classes:

* The command object which is used to execute a command.
* The Connection object which provides a connection to the database.
* The Data Reader object which provides a ready only, connects recordset.

### The Connection object

The connection object created the connection to the database. Microsoft Visual Studio

.NET provides two types of connection classes: the SQLconnection object, which is designed specifically to connect to Microsoft SQL Server.

### The command Object

The command object is represented by corresponding classes: SQL Command. Command object are used to execute commands to a database across a data connection. The command objects provides three methods that are used to execute commands on the database.

* ExecuteNonQuery: Executes commands that have no return values such as INSERT, UPDATE AND DELETE
* ExecuteScalar: Returns a single value from a database query
* ExecuteReader: Returns a result set by way of a DataReader Objects.

### The Data Reader object

The DataReadre object provides a read-only, connected stream recordset from a database. Unlike other components of the Data Provider, DataReader objects cannot be directly instantiated. Rather, the DateReader is returned as the result of the Command objectsExecute Reader method. The DataReader can provide rows of data directly to application logic when one does not need to keep the data cached in memory. Because only one row is in memory in time, the DateReader provides the lowest overhead in terms of system performance but requires the exclusive use of an open Connection object for the life time of the DataReader.

## Microsoft SQL Server

Microsoft SQL Server is an application used to create computer databases for the Microsoft Windows family of server operating systems. Microsoft SQL Server provides an environment used to generate database that can be accessed from workstations, the Internet, or other media such as a personal digital assistant (PDA). Microsoft SQL Server is used to create Web , enterprise, and web-based database applications. It is used at different levels and with various goals.

SQL Server makes simpler and easier to deploy, manage, and optimize enterprise data and analytical applications. An enterprise data management platform, it performs a single management console that enables data administrators anywhere in your organization to monitor, manage, and tune all of the databases and associated services across your enterprise. It provides an extensible management infrastructure that can be easily programmed by using SQL management objects, enabling users to customize and extend their management environment and independent software vendors to build additional tools and functionality to further extend the capabilities that come out of the box.

SQL Server simplifies management by providing integrated management console to monitor and manage the SQL Server relational database as well as integration services, analysis services, reporting services, notification services and SQL Server Mobile Edition across large number of distributed servers and databases. Database

administrator can perform several tasks at the same time, such as authorizing and executing a query, viewing server objects, managing an object, monitoring system activities, and viewing online help.

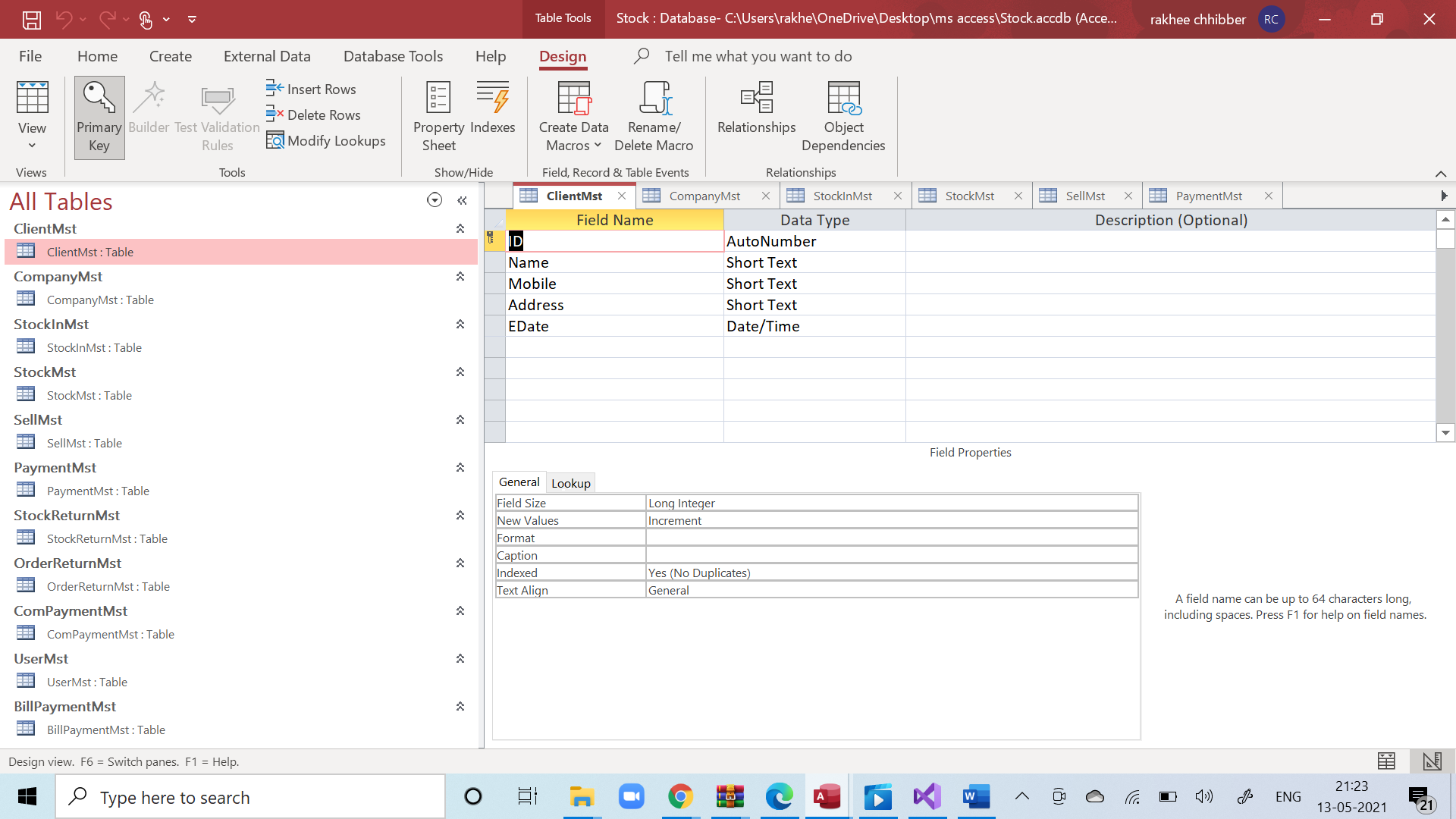
SQL Server expose more than 70 new measures of internal database performance and resource usages, ranging from memory, locking, and scheduling to transactions and network and disk I/O. these dynamic management views provide greater transparency and visibility into the database and a powerful infrastructure for proactive monitoring of database health and performance. The major characteristics are listed below: **Reliability:** achieve a more secure deployment. SQL Server provides rich security features to protect data and network resources.

**Confidentiality:** Protect your data. SQL Server clustering supports Kerberos authentication on a virtual Server and Microsoft-style policies on standard logins so that a consistent policy is applied across all accounts in the domain.

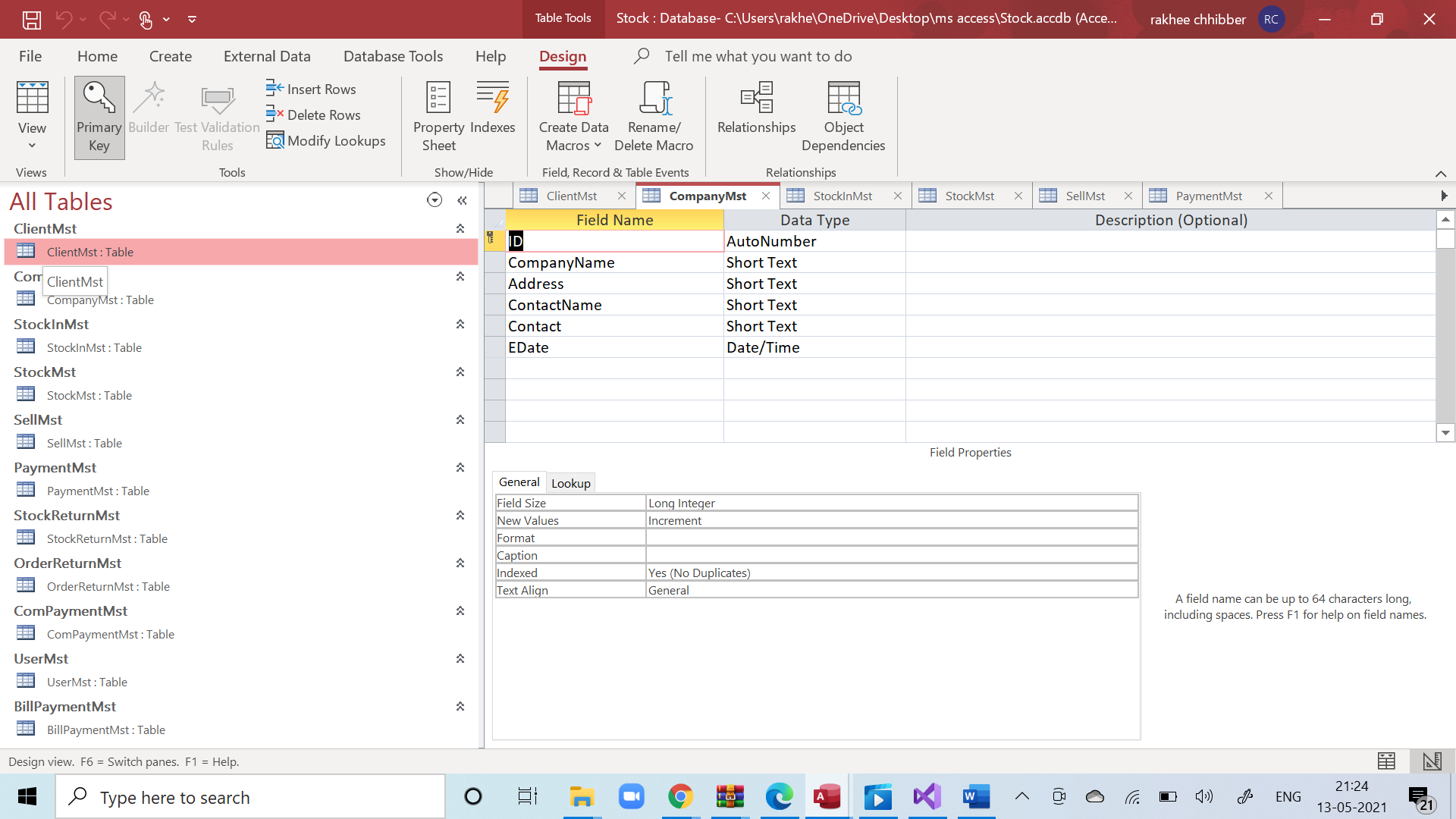
**Integrity:** SQL Server supports encryption capabilities within database itself, fully integrated with a key management infrastructure. By default, client server communications are in encrypted.

**Database and Table structure**

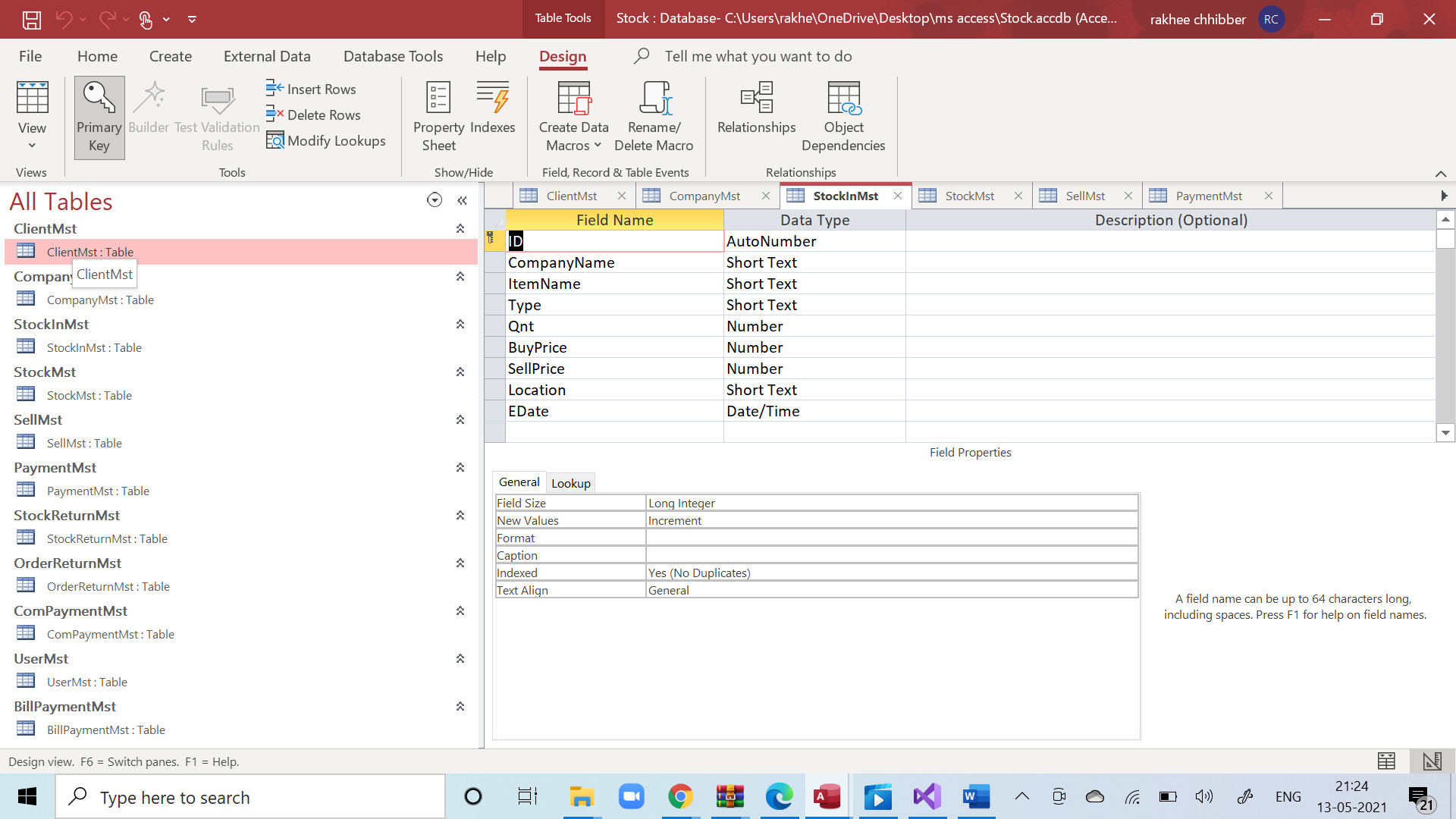
1. Client master



1. Company master



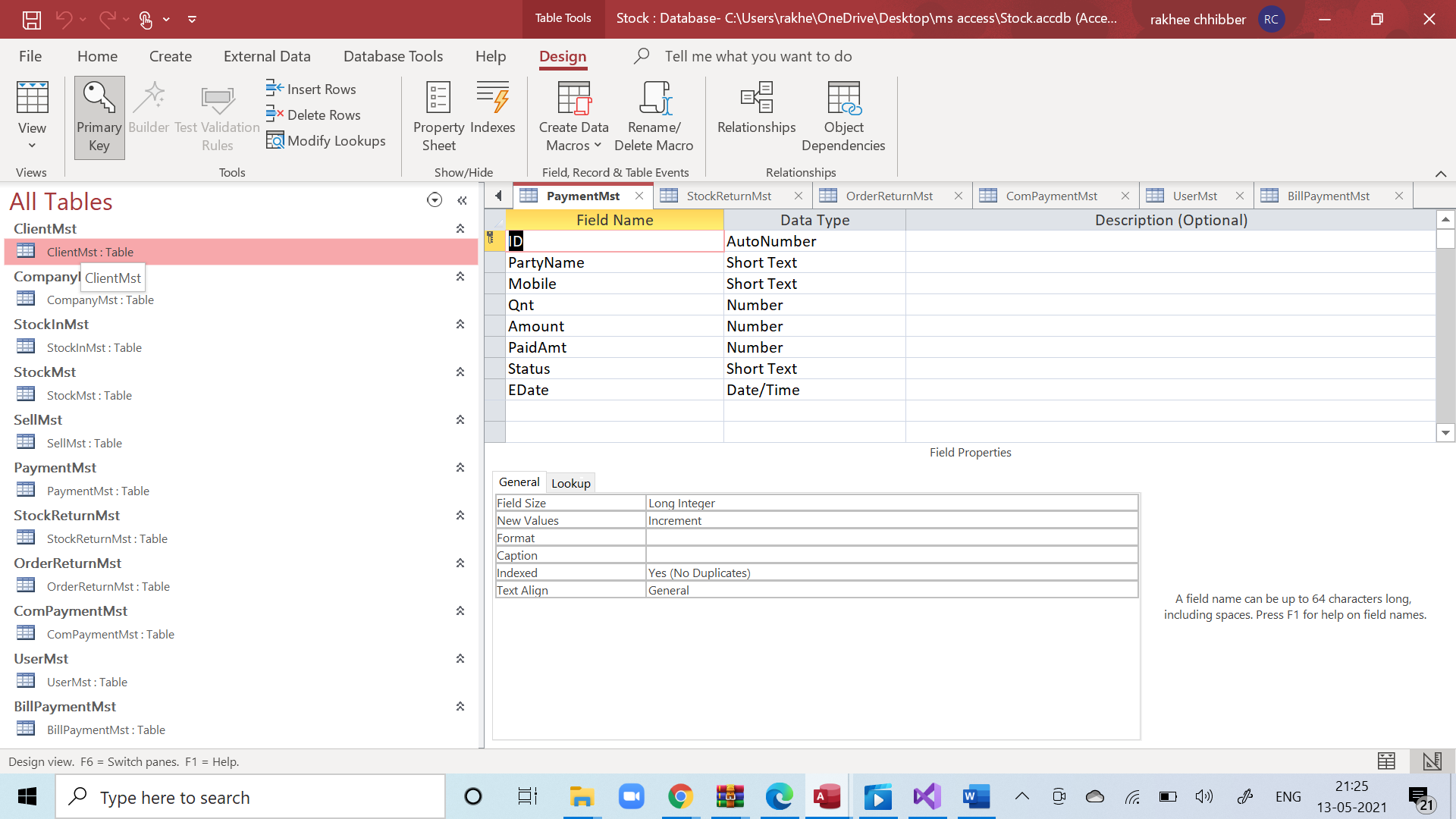
1. Stockinmst



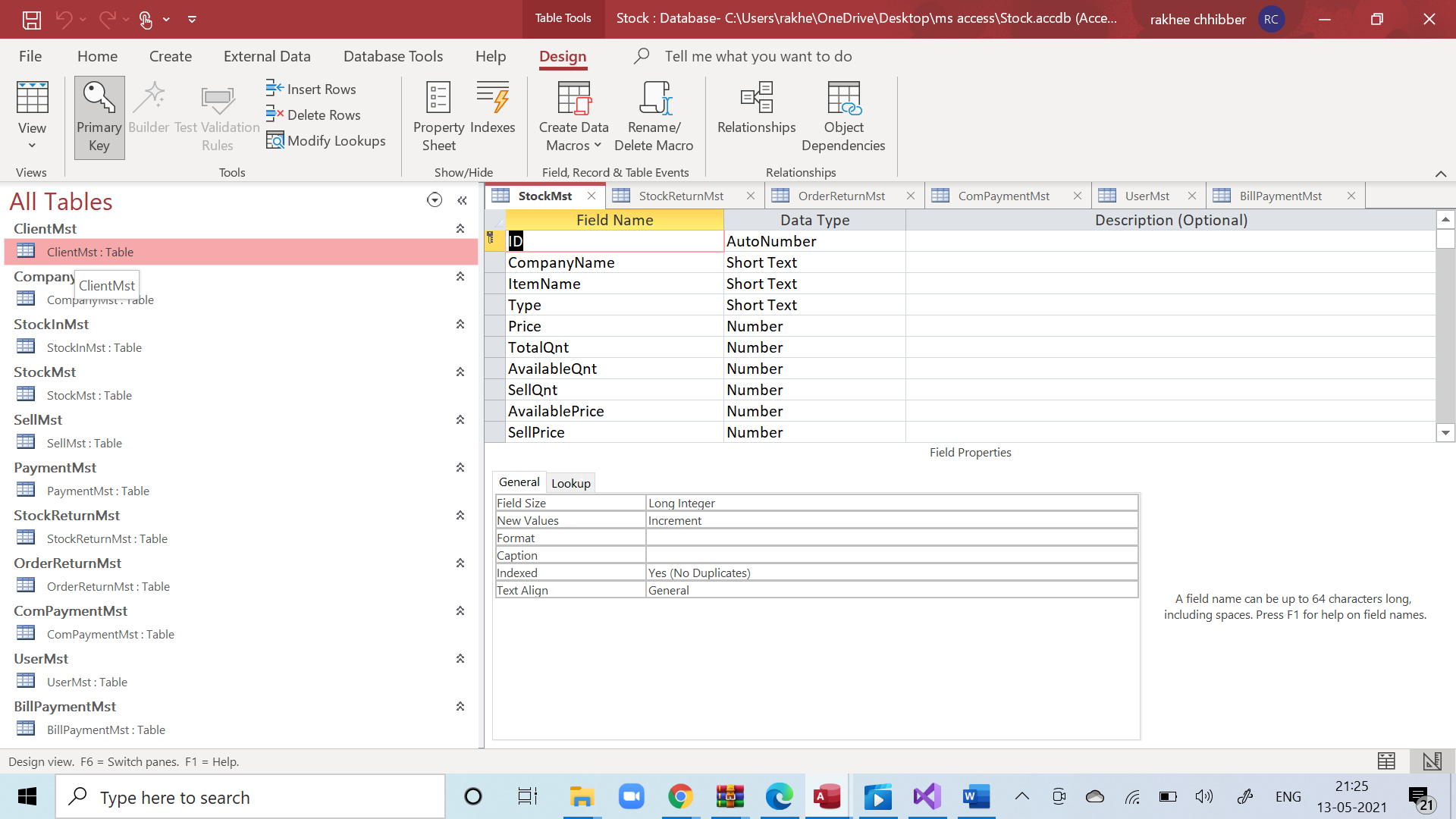
1. Selllmaster



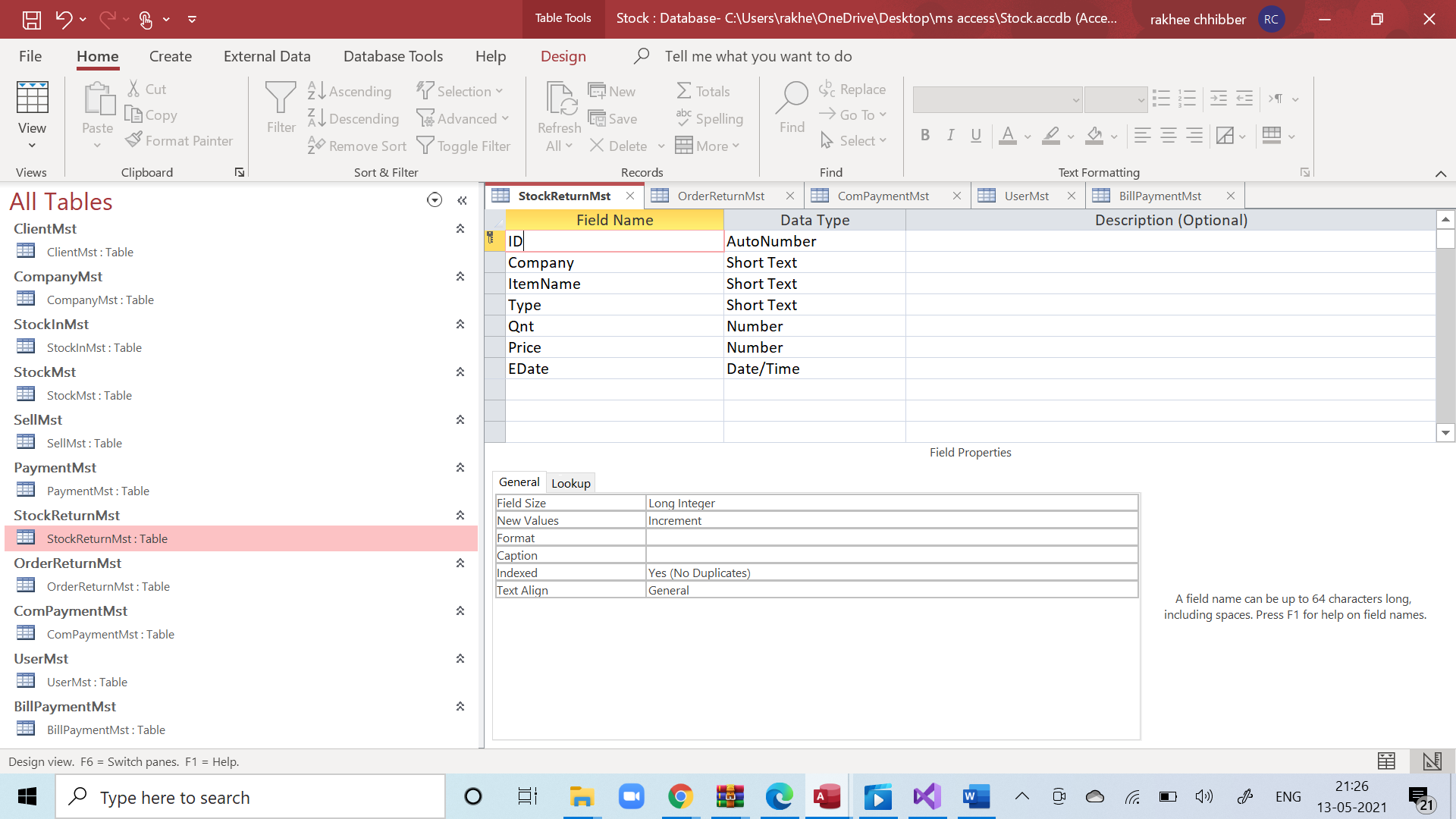
1. Paymentmst



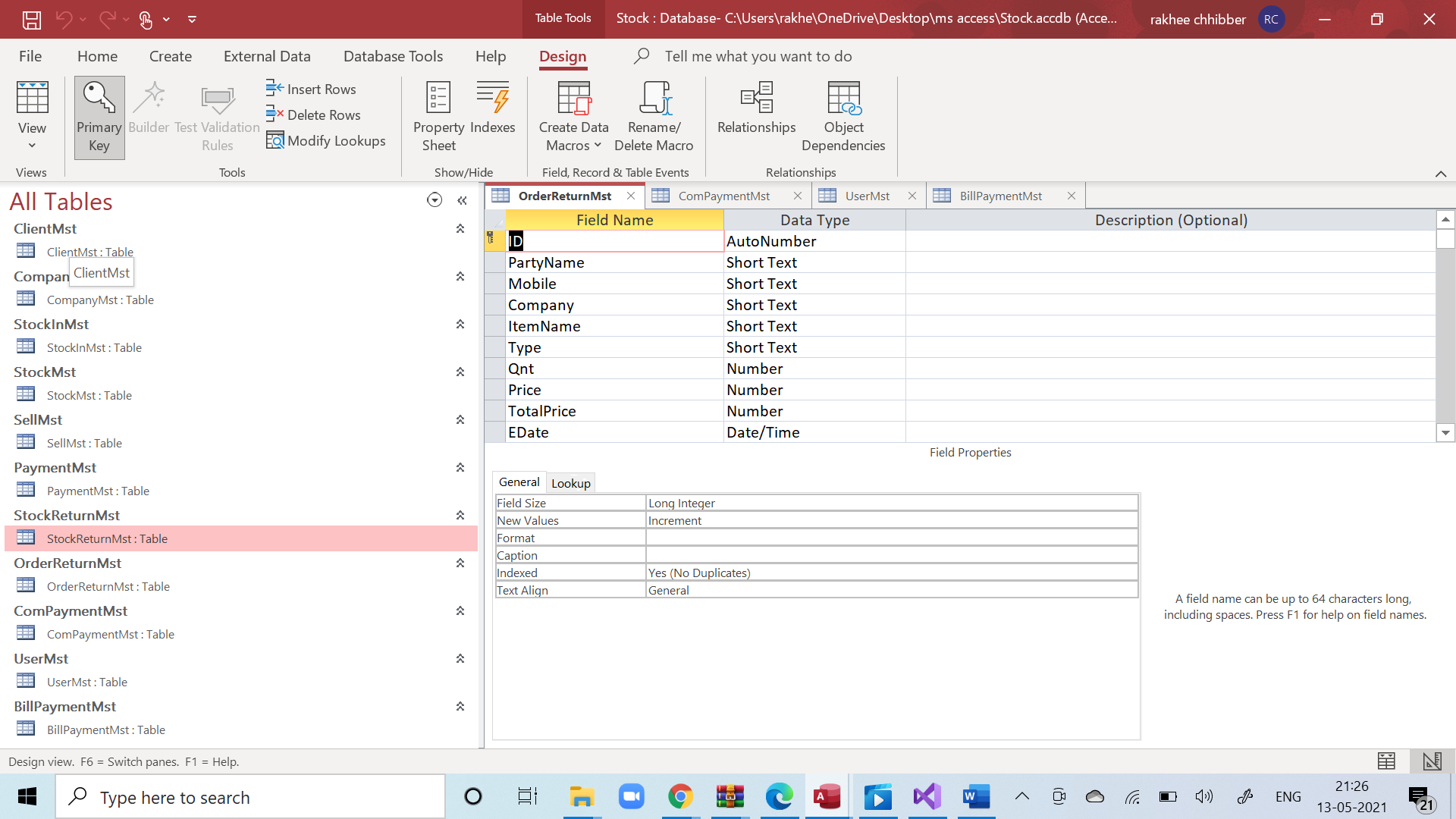
1. Stockmst



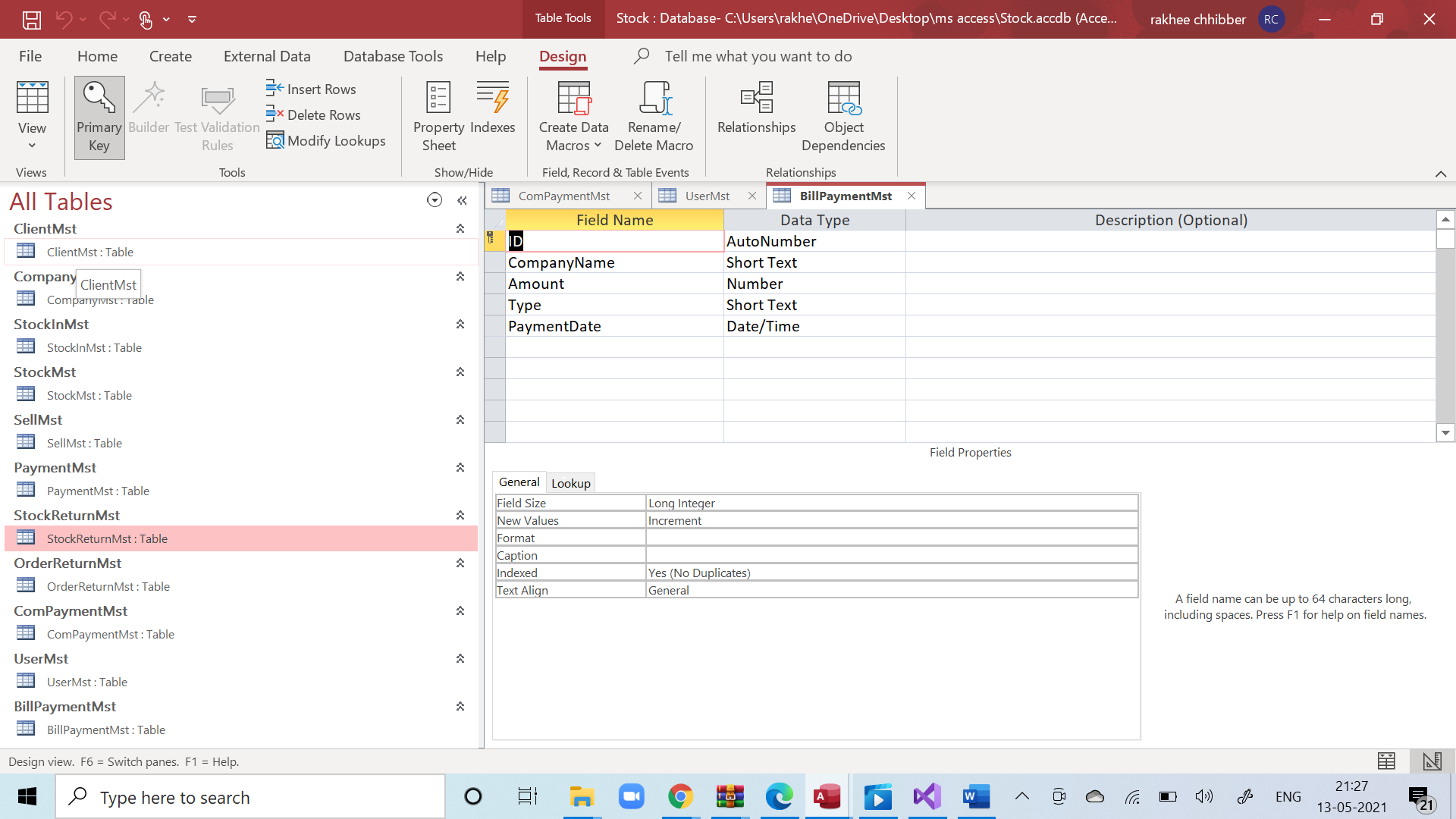
1. Stockreturnmst



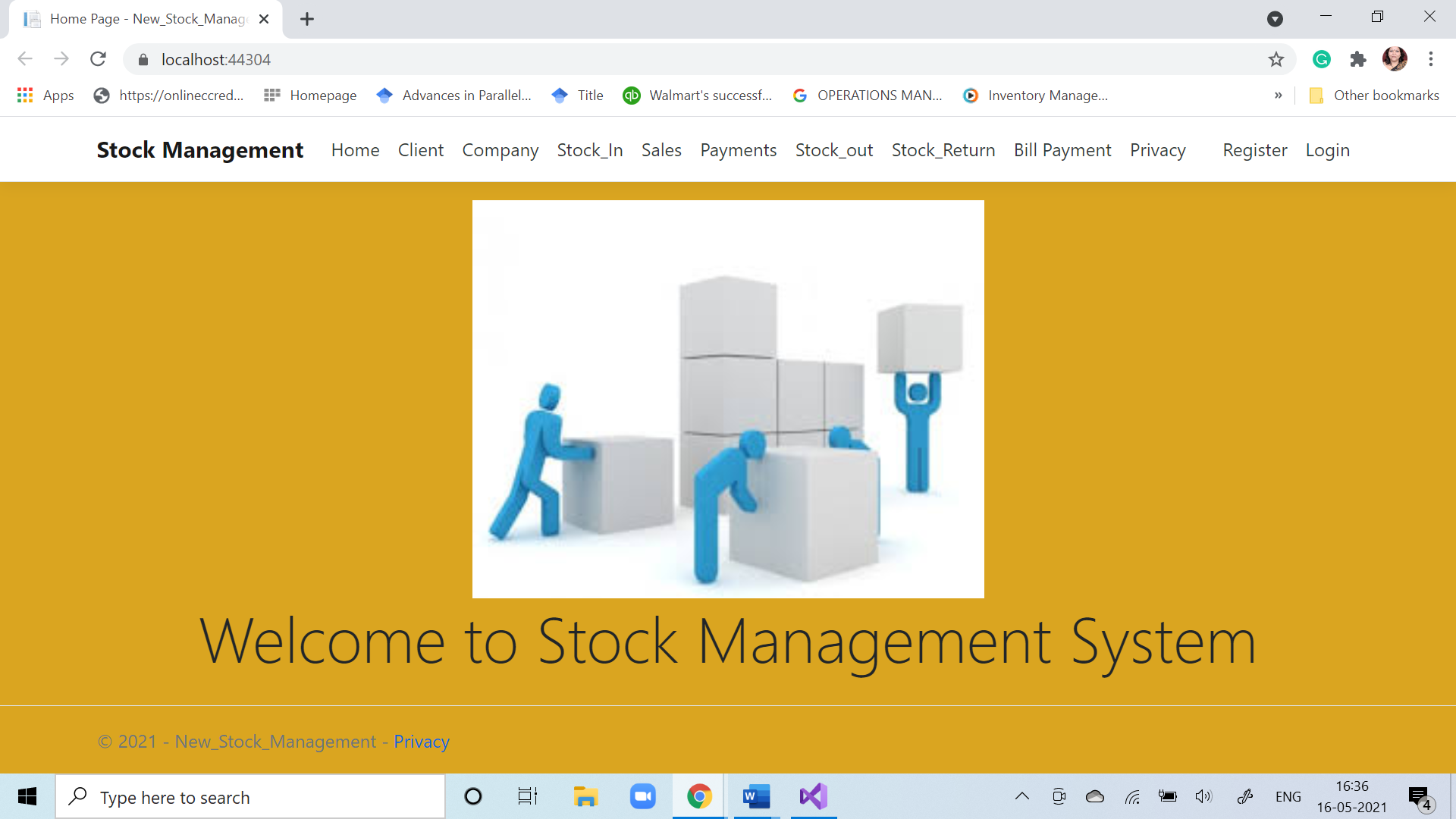
1. Orderreturnmst



1. Billpaymentmst



## Project Screenshot



### Figure 7: Home page

### 

### 

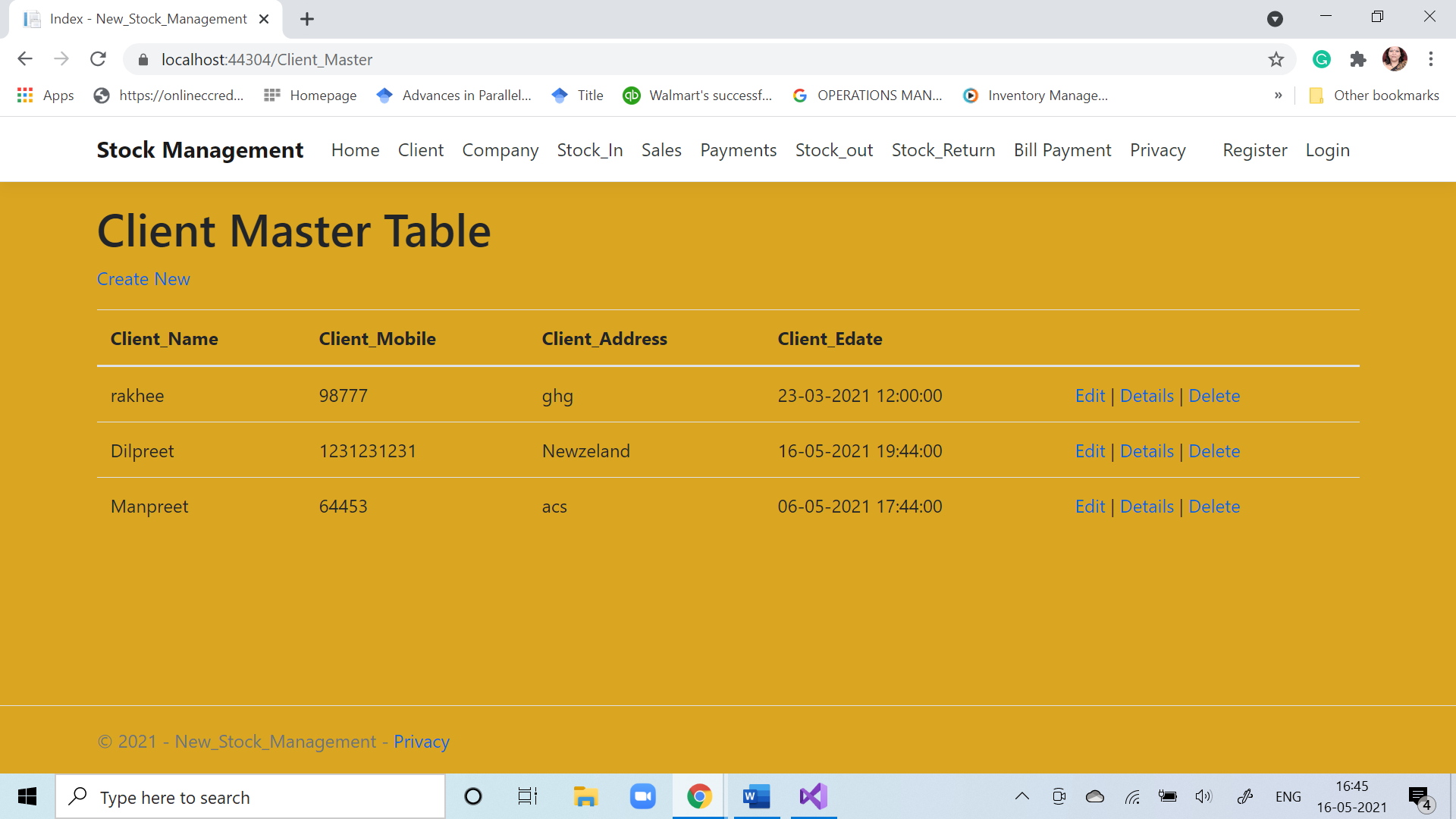
### Figure 8: Register page

### 

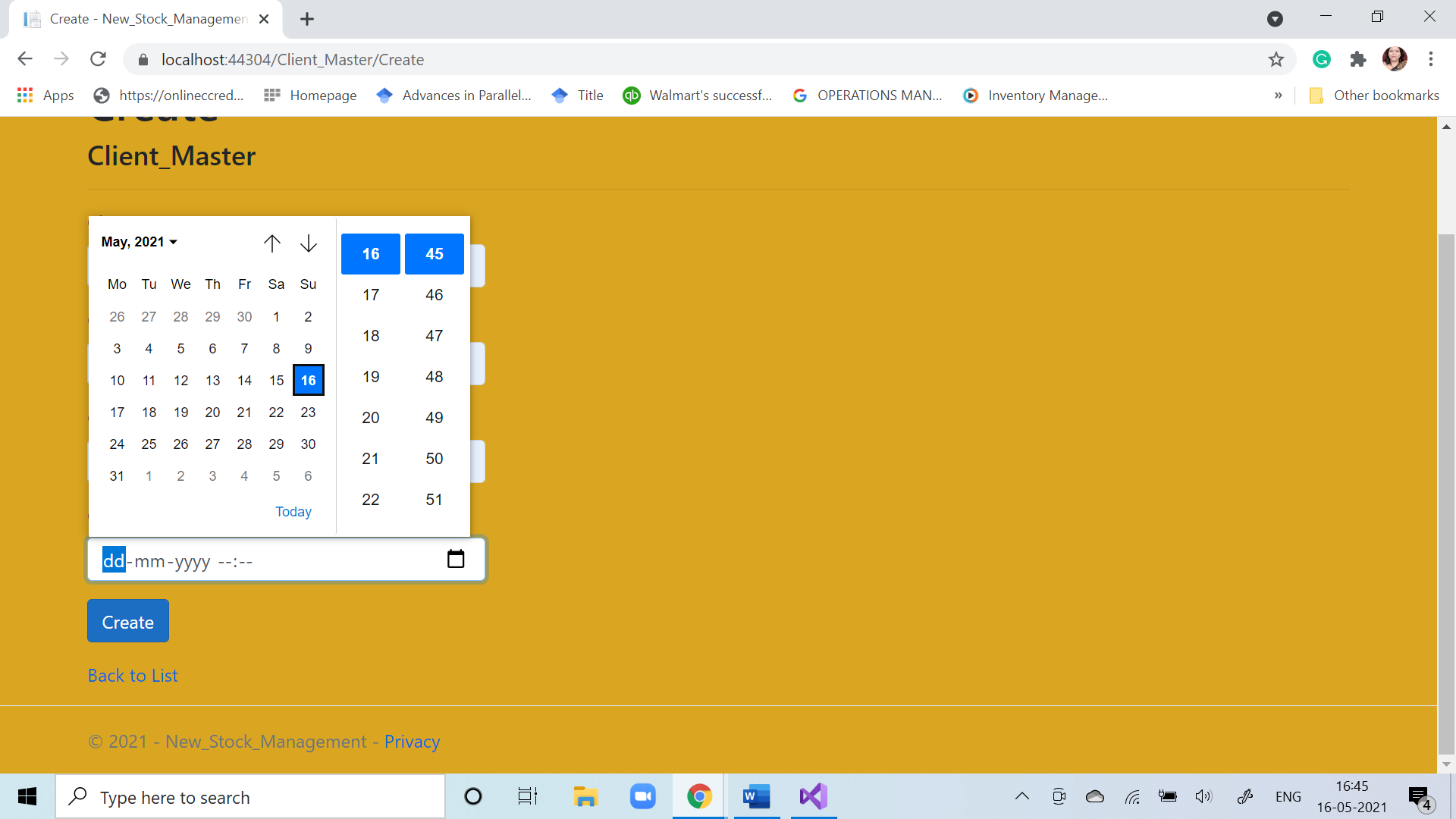
### Figure 9: Password Check

### 

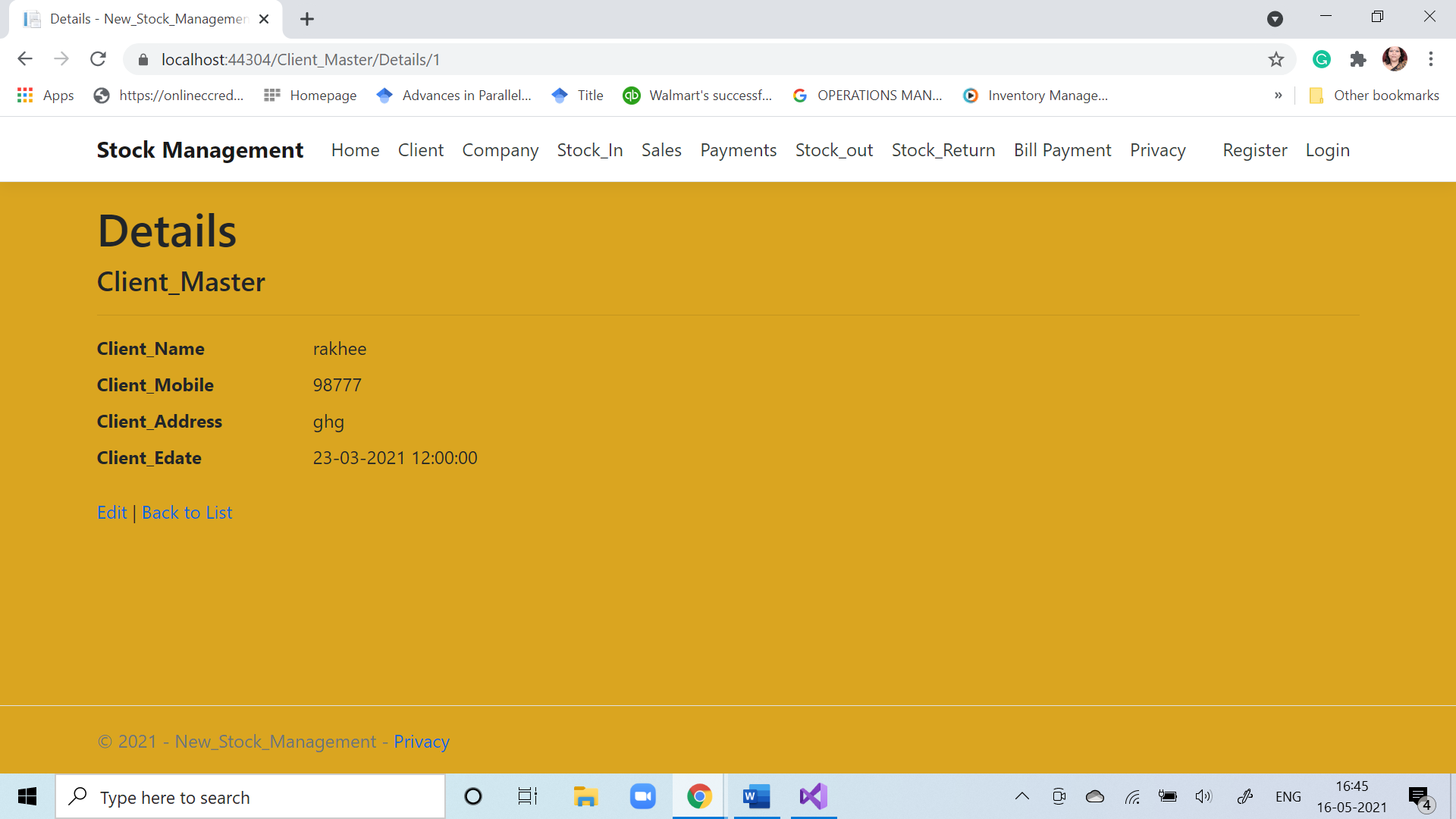
### Figure 10: Registration Page



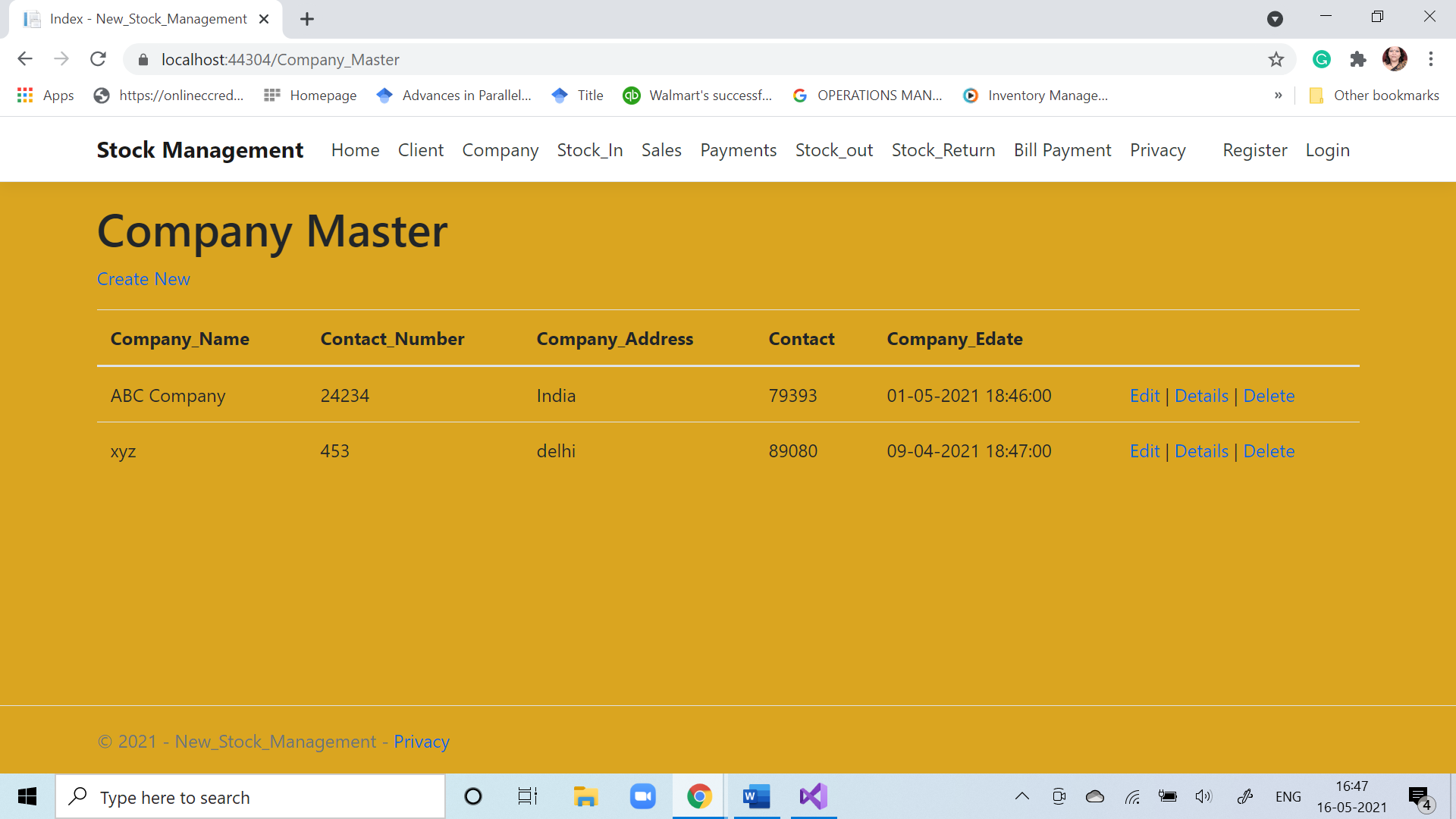
### Figure 11 : Password Check



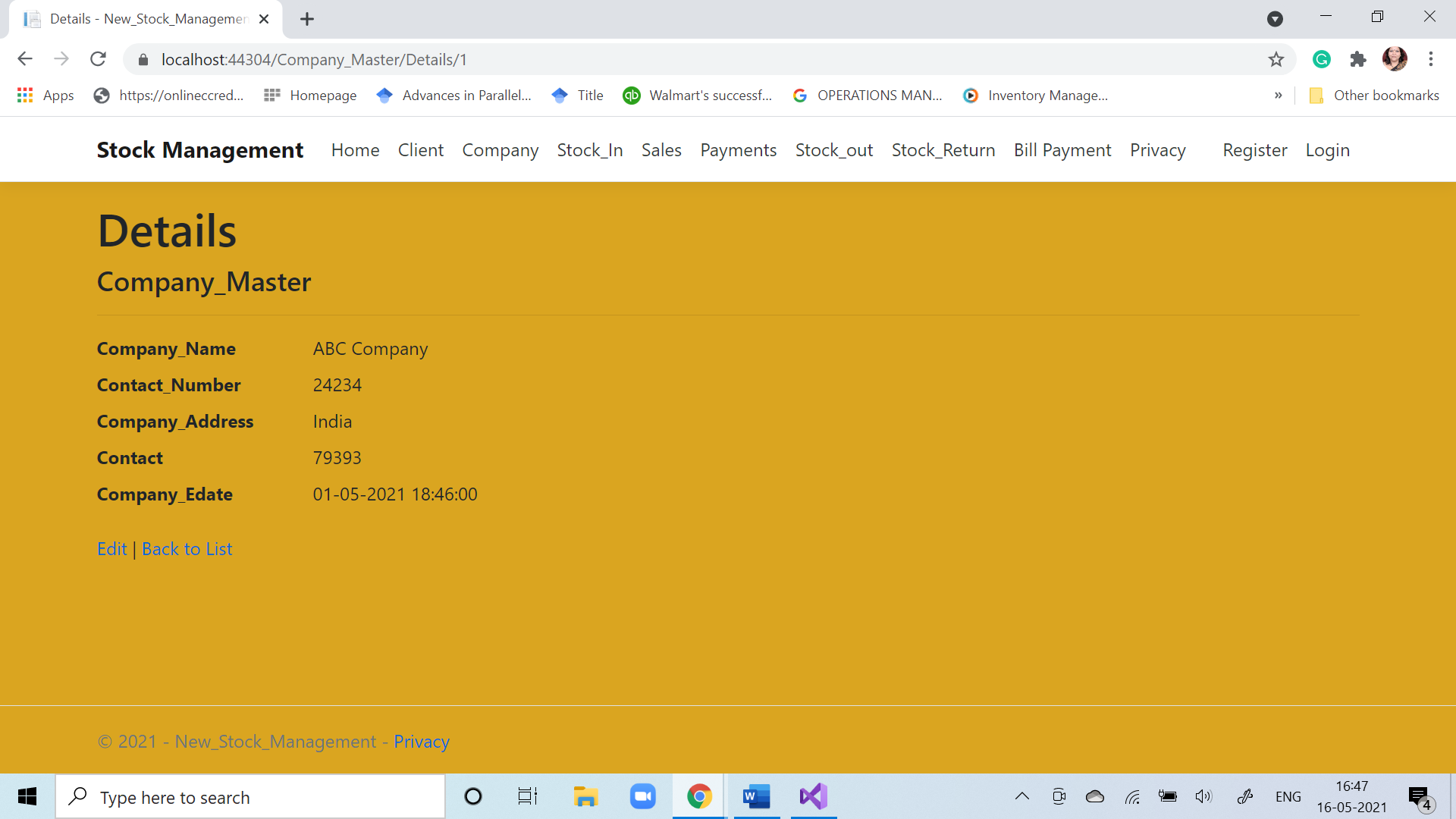
### Figure 12 : Client Master



### Figure 13 : Client Master Details

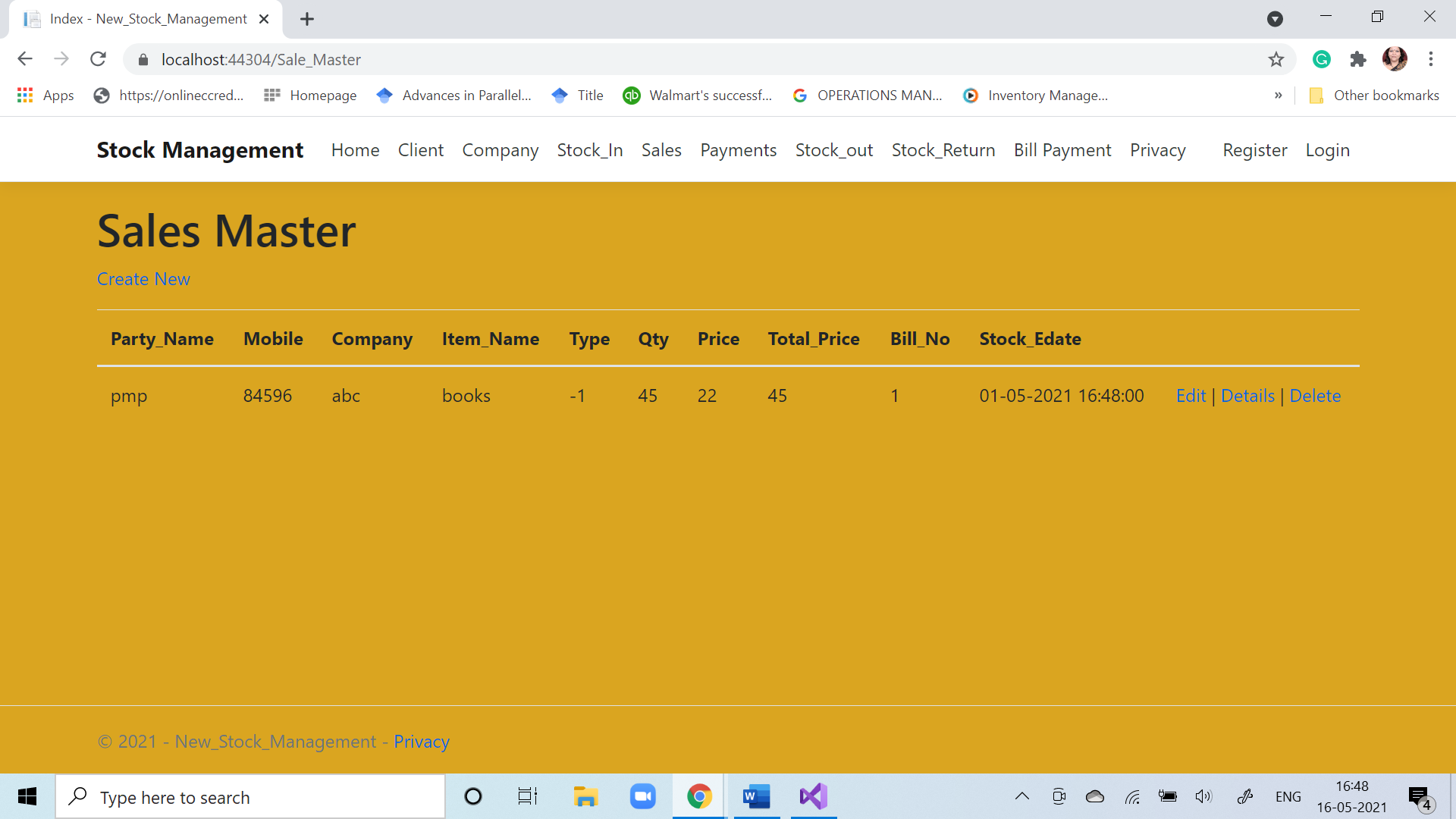


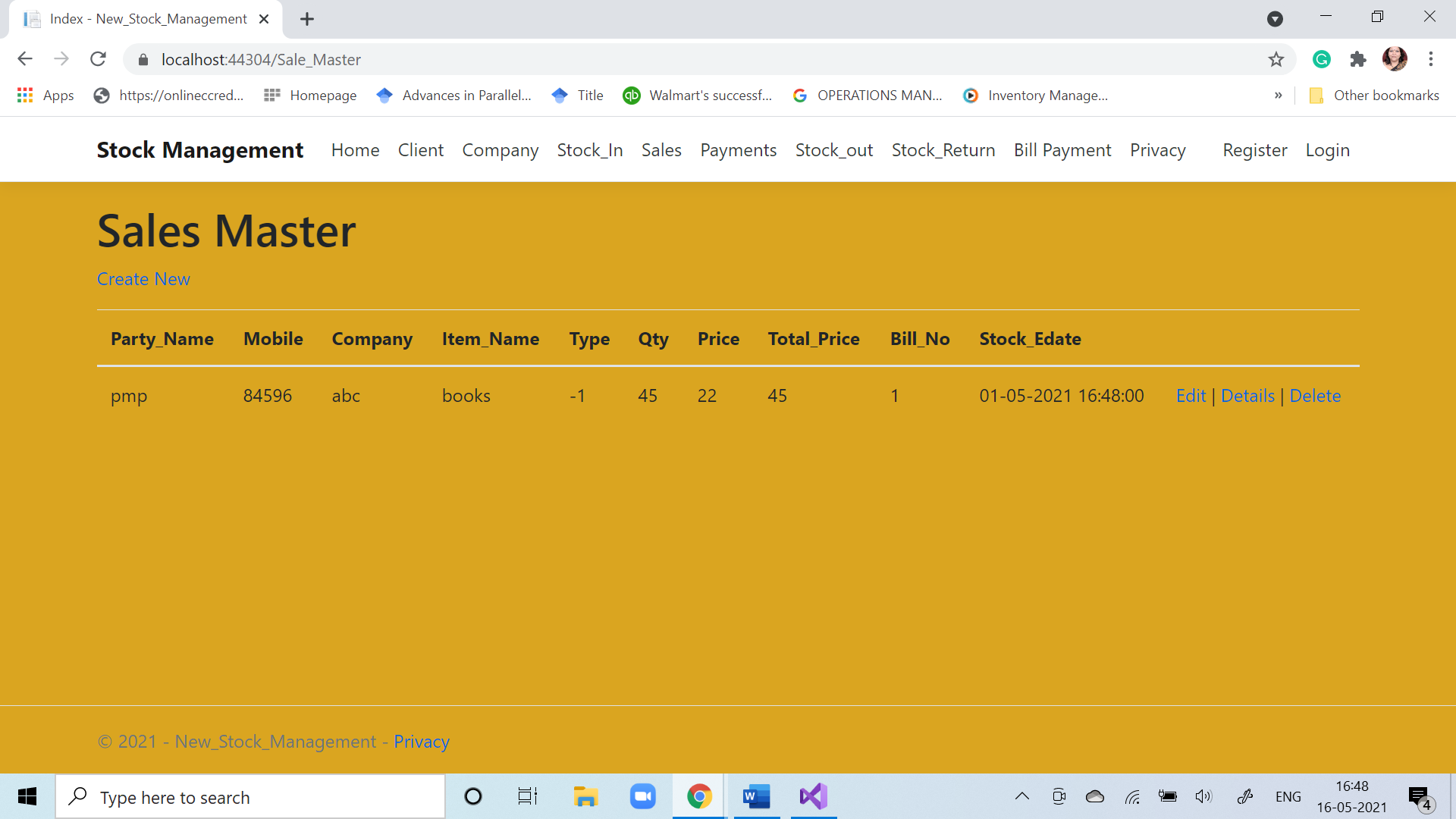
### Figure 14 : Company Master Data Entry



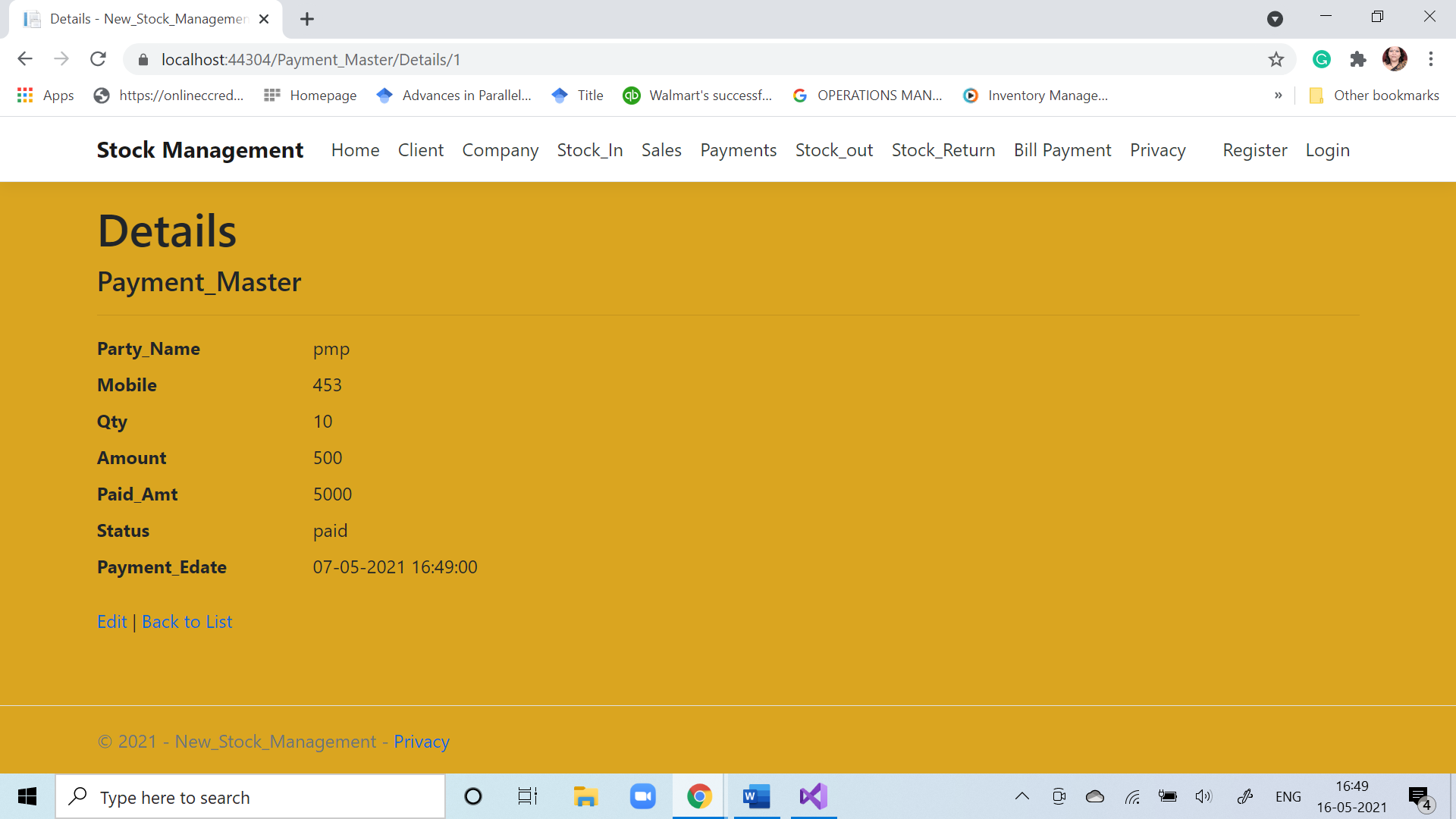
### Figure 15 : Company Master Details

### Figure 16 : Sales Master Data Entry

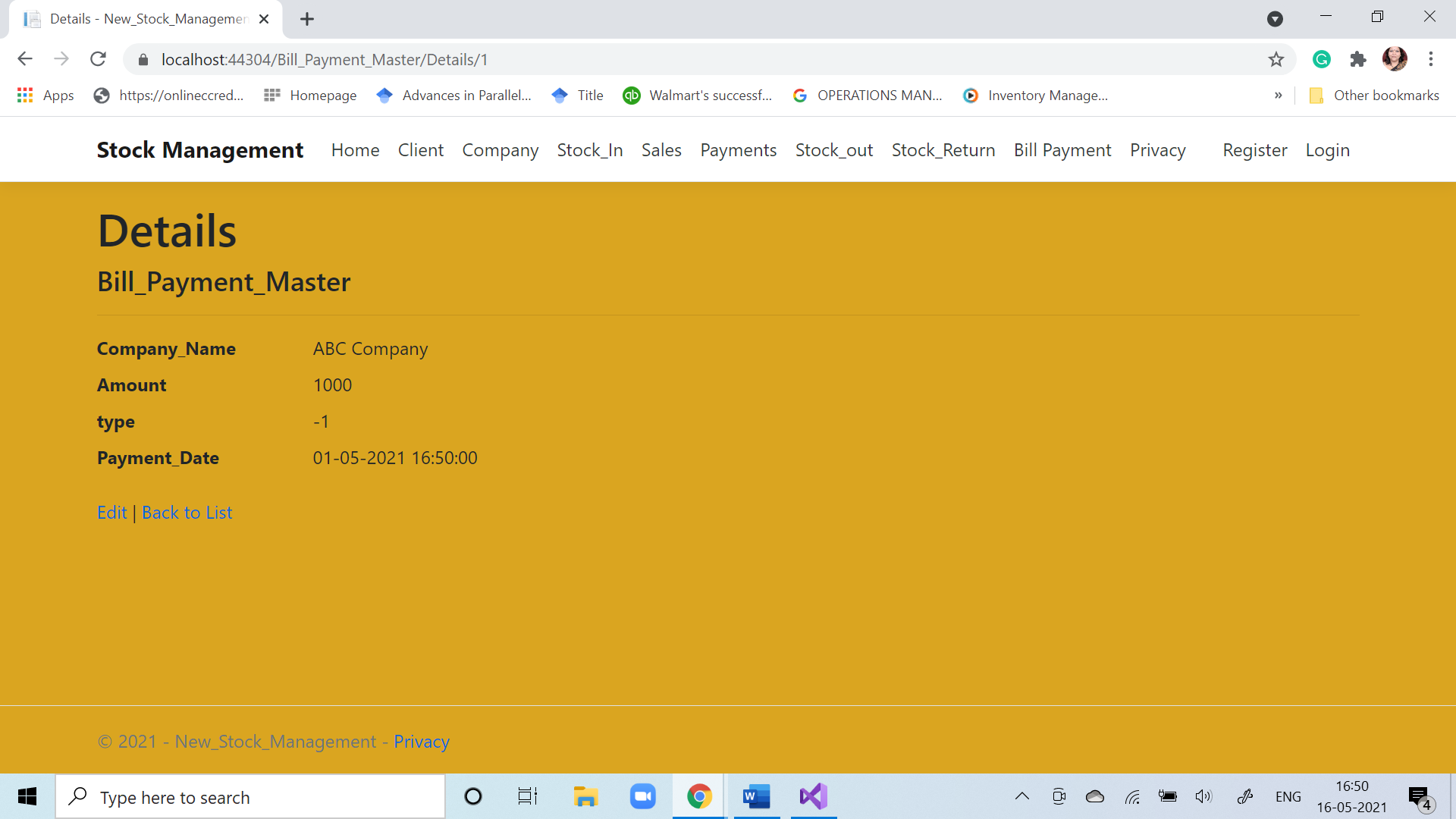




**Figure 17 : Sales Master Details**



**Figure 18 : Payment Master Details**



**Figure 19 : Bill Payment Details**