INVENTORY & POINT OF SALE FOR SMALL BUSINESS

By

Sabbir Hossain
ID: CSE 048-06340

A Project Submitted in Partial Fulfillment of the Requirements for the Degree of Bachelor of Science in Computer Science & Engineering

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
STAMFORD UNIVERSITY BANGLADESH

November 2016

DECLARATION

I, Sabbir Hossain ID: CSE 048-06340, hereby, declare that the work presented in this Project is the outcome of the investigation performed by me under the supervision of Rianon Zaman, Lecturer, Department of Computer Science & Engineering, Stamford University Bangladesh. I also declare that no part of this Project and thereof has been or is being submitted elsewhere for the award of any degree or Diploma.

Countersigned	Signature
………………………………..	….………………………
(Rianon Zaman)	(Sabbir Hossain)
Supervisor	CSE 048-06340
	Candidate

ABSTRACT

Point of sale systems up the profits by improving transaction accuracy. Such systems generate detailed sales reports means small business owners can identify which items bring in the highest profits and which are under-performing. It’s easy for small business owners to immediately determine how much of a particular item sold during a prescribed time period. POS systems drastically decrease the amount of time spent on paper-based tasks. But Country like Bangladesh generally it’s a dream to use a point of sale software by a small business holder. They can’t use the software because of their small budget. So, this project is aimed at developing a desktop based Inventory & Point of sale software for a small business. This system can be used to store the details of inventory, update the inventory based on the sale details, and generate sales and inventory reports. This is one integrated system that contains user component used by sales manager. Admin component used by the sales manager for performing admin level function such as adding new items to the inventory, update of an item, dilation of an item etc. And this project software fulfil all the user requirements.

ACKNOWLEDGEMENTS

First of all I would like to thank the almighty ALLAH. Today I am successful in completing my work with such ease because He gave me the ability, chance, and cooperating supervisor.

I would like to take the opportunity to express my gratitude to Rianon Zaman, Lecturer, my respected supervisor. Although she was always loaded with several other activities, she gave me more than enough time in this work. She not only gave me time but also proper guidance and valuable advice whenever I faced with some difficulties. Her comments and guidance helped me in preparing my project report.

I would like to thank Israt Jahan Mouri and Mohammad Manzurul Islam, my respected teacher, who inspired me in every step.

I am also thankful to my teachers who helped me in a number of ways by providing various resources and moral support.

Last of all I am grateful to my family; who are, always with me in my every step of life.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Introduction

All features and procedures to develop the system will propose by this document. These documents specially containing details about objectives, scope, design model, primary requirements and finally monitoring and reporting mechanisms.

Inventory Point of Sale system is a desktop based application for the desktop implementation that can be used for small shop for proving it for usage by the sales manager of the shop. This software maintain the personal details of customer and purchase details, stock report of a product, update or deletion of a product, consign the daily sale report information.

The services provided to the sales manager can upload a new product, delete a product info, and update a product info. The service also provide sell return option to the customer.

1.2 My Project Overview

Inventory & Point of sale software is a desktop based application where user can maintain his small business. It provides more secure selling services by admin login system, storing customer information, daily sell report, purchase details and stock report. The update or deletion of any product is very user friendly. This software provide an excellent feature like Sell Return option and adjusted the database automatically. So user can Gain better control of business and grow business profit with existing business market and product lines maintaining the reputation.

1.2.1 What is inventory management system?

Inventory Management is an enterprise-wide discipline concerned with the identification and tracking of Information Services (IS) hardware and software assets. Its three main areas of concern are:

i. Acquisition
ii. Redeployment
iii. Termination

Acquisition procedures are established to assist personnel in procurement of software and hardware products. Its main purpose is to ensure that proper justifications are performed and that financial guidelines are followed.

Redeployment procedures are responsible for ensuring that assets are tracked when moved from one location to another and that budgetary considerations are adjusted as needed. Should a product be moved in conjunction with its original owner then the Inventory System is updated to reflect the new location. Should a product location and owner change, then the Inventory System must be updated to reflect the new owner and their location. In this case, the old product is deleted from the original owner's budget and added to the new owner's budget.

Termination is responsible for deleting the asset from the inventory when it is discontinued, or replaced. The owner's budget will be updated to reflect the asset termination and the asset will no longer be listed when location reports are generated.

The Inventory System is maintained within a database that ties an asset to its owner and defines the location where the asset resides. The relative importance of the asset is added to the inventory record (i.e., Criticality = 1-5, where 1 is "Most Critical"). Based on this information the contingency planning specialist can plan asset recoveries needed to support critical business operations [2].

1.2.2 What is Point of Sale (POS)?

The **point of sale** (**POS**) or **point of purchase** (**POP**) is the time and place where a retail transaction is completed. It is the point at which a customer makes a payment to the merchant in exchange for goods or after provision of a service. At the point of sale, the merchant would prepare an invoice for the customer (which may be a cash register printout) or otherwise calculate the amount owed by the customer and provide options for the customer to make payment. After receiving payment, the merchant will also normally issue a receipt for the transaction. Usually the receipt is printed, but it is increasingly being dispensed electronically [1].

1.3 Background of the study

In Bangladesh application of communication technology increased enormously in the recent years. It is perceived that the government and non-government organizations are working in numerous sectors such as e-commerce, business, education, governance, banking and commerce. It indicates that business institutes moving towards using modern technologies to meet the demand of the business stakeholders and make them familiar with the new technologies with low budget. The main intention of this paper is to develop an inventory point of sales system with suitable features and lack of ambiguities. The system allows selling products, stock report, update a product info, delete a product info, store the daily sale report, store the customer details with payment method, and able to sell return option.

1.4 Objective of the study

The objective of inventory POS system is -

I. To be a recognized distributor and suppliers of products.
II. To grow business profit with existing business market and product lines while maintaining reputation.
III. Look for software with a user-friendly graphical interface.
IV. Knowing the stock report of products without complications.
V. Easily sell return option.
VI. Maintain a sales history to help adjust your buying decisions for seasonal purchasing trends.
VII. Automatically update inventory and accounts receivable records.

VIII. Keep tight control over cash receipts to prevent theft.
IX. Gain better control of business.

1.5 Strategies of the project

Strategies of inventory POS system is -

I. Create a long term relationship with community and clients.
II. Know the customers need and demands.

1.6 Goals of the project

Successfully fulfil the project within the limited time period and maintain the all requirement of the project. And also decrease the technical death as much as possible.

CHAPTER 2

SOFTWARE DEVELOPMENT LIFE CYCLE

2.1 What is SDLC?

Software Development life cycle (SDLC) is a process followed for a software project, within a software organization. It consists of a detailed plan describing how to develop, maintain, replace and alter or enhance specific software. The life cycle defines a methodology for improving the quality of software and the overall development process [3].

The following figure is a graphical representation of the various stages of a typical SDLC .

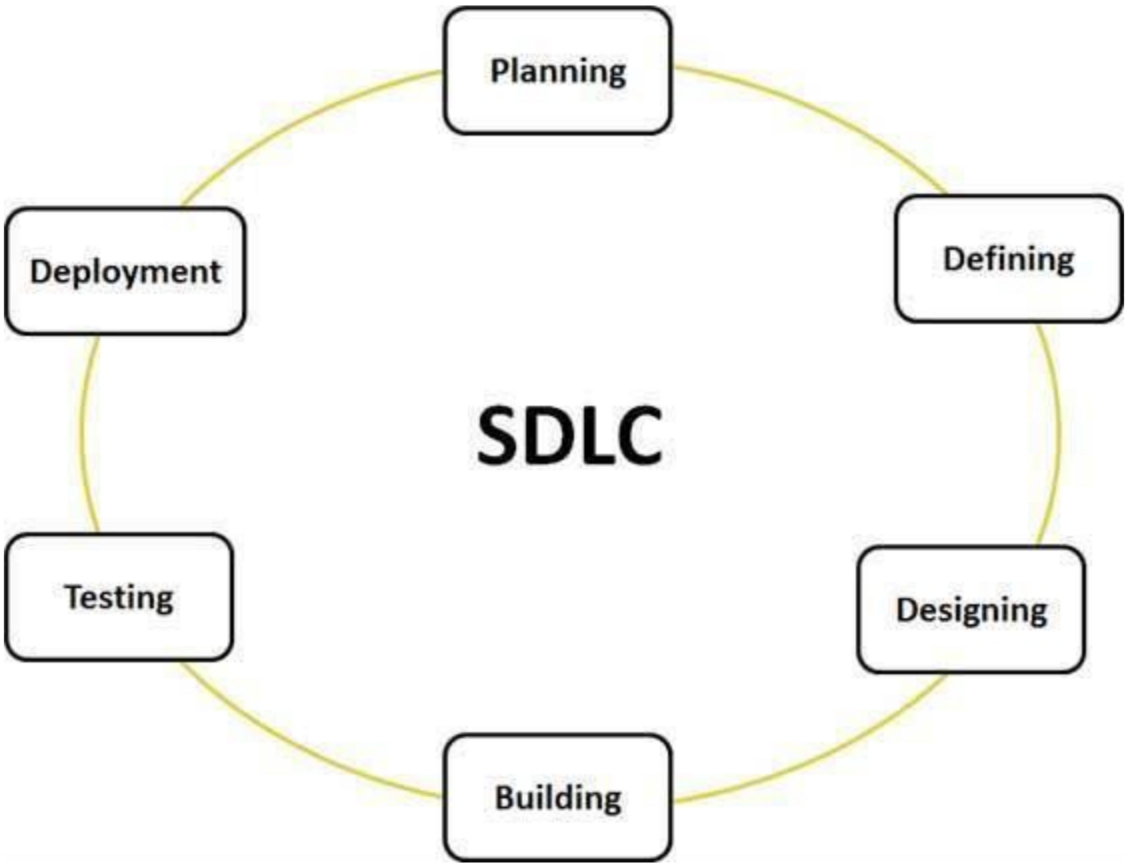


Figure 2.1 Stages of SDSL

2.2 Various Stages of SDLC

2.2.1 Stage 1: Planning and Requirement Analysis

Requirement analysis is the most important and fundamental stage in SDLC. It is performed by the senior members of the team with inputs from the customer, the sales department, market surveys and domain experts in the industry. This information is then used to plan the basic project approach and to conduct product feasibility study in the economical, operational, and technical areas.

Planning for the quality assurance requirements and identification of the risks associated with the project is also done in the planning stage. The outcome of the technical feasibility study is to define the various technical approaches that can be followed to implement the project successfully with minimum risks [3].

2.2.2 Stage 2: Defining Requirements

Once the requirement analysis is done the next step is to clearly define and document the product requirements and get them approved from the customer or the market analysts. This is done through .SRS. . Software Requirement Specification document which consists of all the product requirements to be designed and developed during the project life cycle [3].

2.2.3 Stage 3: Designing the product architecture

SRS is the reference for product architects to come out with the best architecture for the product to be developed. Based on the requirements specified in SRS, usually more than one design approach for the product architecture is proposed and documented in a DDS - Design Document Specification.

This DDS is reviewed by all the important stakeholders and based on various parameters as risk assessment, product robustness, design modularity , budget and time constraints , the best design approach is selected for the product.

A design approach clearly defines all the architectural modules of the product along with its communication and data flow representation with the external and third party modules (if any). The internal design of all the modules of the proposed architecture should be clearly defined with the minutest of the details in DDS [3].

2.2.4 Stage 4: Building or Developing the Product

In this stage of SDLC the actual development starts and the product is built. The programming code is generated as per DDS during this stage. If the design is performed in a detailed and organized manner, code generation can be accomplished without much hassle.

Developers have to follow the coding guidelines defined by their organization and programming tools like compilers, interpreters, debuggers etc. are used to generate the code. Different high level programming languages such as C#, C, C++, Pascal, Java, and PHP are used for coding. The programming language is chosen with respect to the type of software being developed [3].

2.2.5 Stage 5: Testing the Product

This stage is usually a subset of all the stages as in the modern SDLC models, the testing activities are mostly involved in all the stages of SDLC. However this stage refers to the testing only stage of the product where products defects are reported, tracked, fixed and retested, until the product reaches the quality standards defined in the SRS [3].

2.2.6 Stage 6: Deployment in the Market and Maintenance

Once the product is tested and ready to be deployed it is released formally in the appropriate market. Sometime product deployment happens in stages as per the organizations. Business strategy. The product may first be released in a limited segment and tested in the real business environment (UAT- User acceptance testing).

Then based on the feedback, the product may be released as it is or with suggested enhancements in the targeting market segment. After the product is released in the market, its maintenance is done for the existing customer base [3].

2.3 SDLC models

The selection of model has very high impact on the testing that is carried out. It will define the what, where and when of our planned testing, influence regression testing and largely determines which test techniques to use.

There are various Software development models or methodologies. They are as follows [4]:

I. Waterfall model
II. V model
III. Incremental model
IV. RAD model
V. Agile model
VI. Iterative model
VII. Spiral model
VIII. Prototype model

2.4 About inventory POS system model

Choosing right model for developing of the software product or application is very important. Based on the model the development and testing processes are carried out. For the inventory

POS system I choose the agile Kanban methodology because Kanban promotes continuous collaboration and encourages active, ongoing learning and improving by defining the best possible workflow.

2.4.1 Agile model

Agile SDLC model is a combination of iterative and incremental process models with focus on process adaptability and customer satisfaction by rapid delivery of working software product.

Agile Methods break the product into small incremental builds. These builds are provided in iterations. Each iteration typically lasts from about one to three weeks. Every iteration involves cross functional teams working simultaneously on various areas like planning, requirements analysis, design, coding, unit testing, and acceptance testing.

At the end of the iteration a working product is displayed to the customer and important stakeholders [5].

2.4.2 What is Agile?

Agile model believes that every project needs to be handled differently and the existing methods need to be tailored to best suit the project requirements. In agile the tasks are divided to time boxes (small time frames) to deliver specific features for a release.

Iterative approach is taken and working software build is delivered after each iteration. Each build is incremental in terms of features; the final build holds all the features required by the customer.

Here is a graphical illustration of the Agile Model [5]:

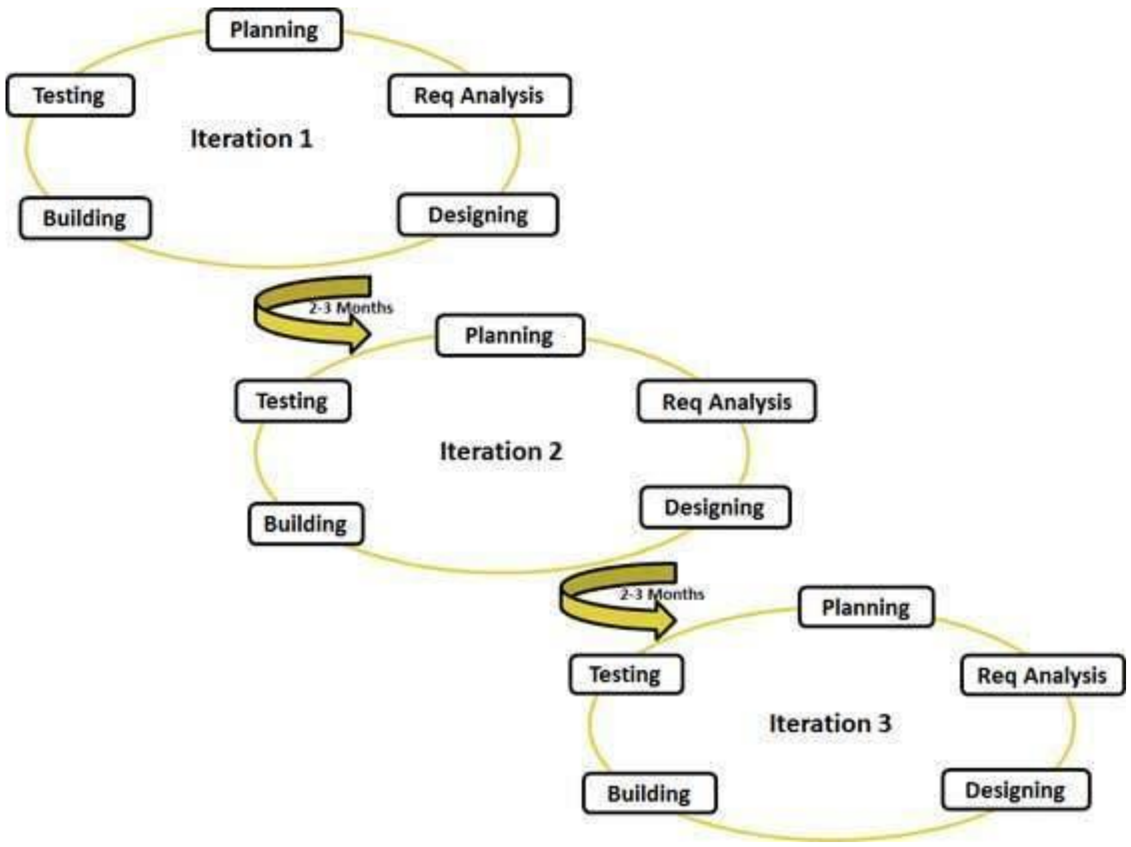


Figure 2.2. Graphical illustration of the Agile Model

2.4.3 Agile Principles

Following are the Agile Manifesto principles [5] -

- **Individuals and interactions** - in agile development, self-organization and motivation are important, as are interactions like co-location and pair programming.
- **Working software** - Demo working software is considered the best means of communication with the customer to understand their requirement, instead of just depending on documentation.
- **Customer collaboration** - As the requirements cannot be gathered completely in the beginning of the project due to various factors, continuous customer interaction is very important to get proper product requirements.
- **Responding to change** - agile development is focused on quick responses to change and continuous development.

2.4.4 Agile Scrum Methodology

Scrum is a lightweight agile project management framework with broad applicability for managing and controlling iterative and incremental projects of all types. Ken Schwaber, Mike Beedle, Jeff Sutherland and others have contributed significantly to the evolution of Scrum over the last decade. Scrum has garnered increasing popularity in the agile software development community due to its simplicity, proven productivity, and ability to act as a wrapper for various engineering practices promoted by other agile methodologies.

With Scrum methodology, the "Product Owner" works closely with the team to identify and prioritize system functionality in form of a "Product Backlog". The Product Backlog consists of features, bug fixes, non-functional requirements, etc. – whatever needs to be done in order to successfully deliver a working software system. With priorities driven by the Product Owner, cross-functional teams estimate and sign-up to deliver "potentially shippable increments" of software during successive Sprints, typically lasting 30 days. Once a Sprint's Product Backlog is committed, no additional functionality can be added to the Sprint except by the team. Once a Sprint has been delivered, the Product Backlog is analyzed and reprioritized, if necessary, and the next set of functionality is selected for the next Sprint.

Scrum methodology has been proven to scale to multiple teams across very large organizations with 800+ people. See how Version One supports Scrum Sprint Planning by making it easier to manage your Product Backlog [7].

2.4.5 Agile Kanban Methodology

The Kanban Method is used by organizations to manage the creation of products with an emphasis on continual delivery while not overburdening the development team. Like Scrum, Kanban is a process designed to help teams work together more effectively.

Kanban is based on 3 basic principles:

- **Visualize what you do today (workflow):** seeing all the items in context of each other can be very informative
- **Limit the amount of work in progress (WIP):** this helps balance the flow-based approach so teams don 't start and commit to too much work at once
- **Enhance flow:** when something is finished, the next highest thing from the backlog is pulled into play

Kanban promotes continuous collaboration and encourages active, ongoing learning and improving by defining the best possible team workflow. See how Version One supports Kanban software development [7].

2.4.6 Advantages of Agile model

- Customer satisfaction by rapid, continuous delivery of useful software.
- People and interactions are emphasized rather than process and tools. Customers, developers and testers constantly interact with each other.
- Working software is delivered frequently (weeks rather than months).
- Face-to-face conversation is the best form of communication.
- Close, daily cooperation between business people and developers.
- Continuous attention to technical excellence and good design.
- Regular adaptation to changing circumstances.
- Even late changes in requirements are welcomed [6].

2.4.7 Disadvantages of Agile model

- In case of some software deliverables, especially the large ones, it is difficult to assess the effort required at the beginning of the software development life cycle.
- There is lack of emphasis on necessary designing and documentation.
- The project can easily get taken off track if the customer representative is not clear what final outcome that they want.
- Only senior programmers are capable of taking the kind of decisions required during the development process. Hence it has no place for newbie programmers, unless combined with experienced resources [6].

2.4.8 When to use Agile model

- When new changes are needed to be implemented. The freedom agile gives to change is very important. New changes can be implemented at very little cost because of the frequency of new increments that are produced.
- To implement a new feature the developers need to lose only the work of a few days, or even only hours, to roll back and implement it.
- Unlike the waterfall model in agile model very limited planning is required to get started with the project. Agile assumes that the end users' needs are ever changing in a dynamic business and IT world. Changes can be discussed and features can be newly effected or removed based on feedback. This effectively gives the customer the finished system they want or need.

- Both system developers and stakeholders alike, find they also get more freedom of time and options than if the software was developed in a more rigid sequential way. Having options gives them the ability to leave important decisions until more or better data or even entire hosting programs are available; meaning the project can continue to move forward without fear of reaching a sudden standstill [6].

CHAPTER 3

REQUIREMENT ANALYSIS

3.1 Requirement Analysis

Requirements analysis, also called **requirements** engineering, is the process of determining user expectations for a new or modified product. These features, called requirements, must be quantifiable, relevant and detailed. In software engineering, such requirements are often called functional specifications.

Requirements analysis is critical to the success or failure of a systems or software project. The requirements should be documented, actionable, measurable, testable, traceable, related to identified business needs or opportunities, and defined to a level of detail sufficient for system design [8].

Conceptually, requirements analysis includes three types of activities

- **Eliciting requirements:** (e.g. the project charter or definition), business process documentation, and stakeholder interviews. This is sometimes also called requirements gathering.

- **Analyzing requirements**: Determining whether the stated requirements are clear, complete, consistent and unambiguous, and resolving any apparent conflicts.
- **Recording requirements:** Requirements may be documented in various forms, usually including a summary list and may include natural-language documents, use cases, user stories, or process specifications.

Requirements analysis can be a long and tiring process during which many delicate psychological skills are involved. Large systems may confront analysts with hundreds or thousands of system requirements. New systems change the environment and relationships between people, so it is important to identify all the stakeholders, take into account all their needs and ensure they understand the implications of the new systems. Analysts can employ several techniques to elicit the requirements from the customer. These may include the development of scenarios (represented as user stories in agile methods), the identification of use cases, the use of workplace observation or ethnography, holding interviews, or focus groups (more aptly named in this context as requirements workshops, or requirements review sessions) and creating requirements lists. Prototyping may be used to develop an example system that can be demonstrated to stakeholders. Where necessary, the analyst will employ a combination of these methods to establish the exact requirements of the stakeholders, so that a system that meets the business needs is produced. Requirements quality can be improved through these and other methods [8] -

- **Visualization:** Using tools that promote better understanding of the desired end-product such as visualization and simulation.
- **Consistent use of templates**: Producing a consistent set of models and templates to document the requirements.
- **Documenting dependencies**: Documenting dependencies and interrelationships among requirements, as well as any assumptions and congregations.

3.2 Stakeholder identification

Stakeholder analysis for a discussion of people or organizations (legal entities such as companies, standards bodies) that have a valid interest in the system. They may be affected by it either directly or indirectly. A major new emphasis in the 1990s was a focus on the identification of *stakeholders*. It is increasingly recognized that stakeholders are not limited to the organization employing the analyst. Other stakeholders will include:

- anyone who operates the system (normal and maintenance operators)
- anyone who benefits from the system (functional, political, financial and social beneficiaries)
- Anyone involved in purchasing or procuring the system. In a mass-market product organization, product management, marketing and sometimes sales act as surrogate consumers (mass-market customers) to guide development of the product
- organizations which regulate aspects of the system (financial, safety, and other regulators)
- people or organizations opposed to the system (negative stakeholders; see also Misuse case)
- organizations responsible for systems which interface with the system under design
- Those organizations who integrate horizontally with the organization for whom the analyst is designing the system [8].

3.3 Stakeholder interviews

Stakeholder interviews are a common technique used in requirement analysis. Though they are generally idiosyncratic in nature and focused upon the perspectives and perceived needs of the stakeholder, often this perspective deficiency has the general advantage of obtaining a much richer understanding of the stakeholder's unique business processes, decision-relevant business rules, and perceived needs. Consequently, this technique can serve as a means of obtaining the highly focused knowledge that is often not elicited in Joint Requirements Development sessions, where the stakeholder's attention is compelled to assume a more cross-functional context, and the desire to avoid controversy may limit the stakeholder’s willingness to contribute. Moreover, the in-person nature of the interviews provides a more relaxed environment where lines of thought may be explored at length [8].

3.4 Measurable goals

Best practices take the composed list of requirements merely as clues and repeatedly ask "why?" until the actual business purposes are discovered. Stakeholders and developers can then devise tests to measure what level of each goal has been achieved thus far. Such goals change more slowly than the long list of specific but unmeasured requirements. Once a small set of critical, measured goals has been established, rapid prototyping and short iterative development phases may proceed to deliver actual stakeholder value long before the project is half over [8].

3.5 Types of Requirements

Requirements are categorized in several ways. The following are common categorizations of requirements that relate to technical management:

Customer Requirements: Statements of fact and assumptions that define the expectations of the system in terms of mission objectives, environment, constraints, and measures of effectiveness and suitability (MOE/MOS). The customers are those that perform the eight primary functions of systems engineering, with special emphasis on the operator as the key customer. Operational requirements will define the basic need and, at a minimum, answer the questions posed in the following listing:

- Operational distribution or deployment: Where will the system be used?
- Mission profile or scenario: How will the system accomplish its mission objective?
- Performance and related parameters: What are the critical system parameters to accomplish the mission?
- Utilization environments: How are the various system components to be used?
- Effectiveness requirements: How effective or efficient must the system be in performing its mission?
- Operational life cycle: How long will the system be in use by the user?
- Environment: What environments will the system be expected to operate in an effective manner?

Architectural Requirements: Architectural requirements explain what has to be done by identifying the necessary systems architecture of a system.

Structural Requirements: Structural requirements explain what has to be done by identifying the necessary structure of a system.

Behavioral Requirements: Behavioral requirements explain what has to be done by identifying the necessary behavior of a system.

Functional Requirements: Functional requirements explain what has to be done by identifying the necessary task, action or activity that must be accomplished. Functional requirements analysis will be used as the top-level functions for functional analysis.

Non-functional Requirements: Non-functional requirements are requirements that specify criteria that can be used to judge the operation of a system, rather than specific behaviors.

Core Functionality and Ancillary Functionality Requirements: Murali Chemuturi defined requirements into Core Functionality and Ancillary Functionality requirements. Core Functionality requirements are those without fulfilling which the product cannot be useful at all. Ancillary Functionality requirements are those that are supportive to Core Functionality.

The product can continue to work even if some or all of the Ancillary Functionality requirements are fulfilled but with some side effects. Security, safety, user friendliness and so on are examples of Ancillary Functionality requirements.

Performance Requirements: The extent to which a mission or function must be executed; generally measured in terms of quantity, quality, coverage, timeliness or readiness. During requirements analysis, performance (how well does it have to be done) requirements will be interactively developed across all identified functions based on system life cycle factors; and characterized in terms of the degree of certainty in their estimate, the degree of criticality to system success, and their relationship to other requirements.

Design Requirements: The “build to,” “code to,” and “buy to” requirements for products and “how to execute” requirements for processes expressed in technical data packages and technical manuals.

Derived Requirements: Requirements that are implied or transformed from higher-level requirement. For example, a requirement for long range or high speed may result in a design requirement for low weight.

Allocated Requirements: A requirement that is established by dividing or otherwise allocating a high-level requirement into multiple lower-level requirements [8].

3.6 Requirements analysis issues

3.6.1 Stakeholder issues

Steve McConnell, in his book *Rapid Development*, details a number of ways users can inhibit requirements gathering:

- Users do not understand what they want or users don't have a clear idea of their requirements
- Users will not commit to a set of written requirements
- Users insist on new requirements after the cost and schedule have been fixed
- Communication with users is slow
- Users often do not participate in reviews or are incapable of doing so
- Users are technically unsophisticated
- Users do not understand the development process
- Users do not know about present technology

This may lead to the situation where user requirements keep changing even when system or product development has been started. It is also means that the requirement that's is under action process [8].

3.6.2 Engineer/developer issues

Possible problems caused by engineers and developers during requirements analysis are:

- A natural inclination towards writing code can lead to implementation beginning before the requirements analysis is complete, potentially resulting in inelegant refactoring to meet actual requirements once they are known.
- Technical personnel and end-users may have different vocabularies. Consequently, they may wrongly believe they are in perfect agreement until the finished product is supplied.
- Engineers and developers may try to make the requirements fit an existing system or model, rather than develop a system specific to the needs of the client.
- Analysis may often be carried out by engineers or programmers, rather than personnel with the domain knowledge to understand a client's needs properly [8].

3.6.3 Attempted solutions

One attempted solution to communications problems has been to employ specialists in business or system analysis.

Techniques introduced in the 1990s like prototyping, Unified Modeling Language (UML), use cases, and agile software development are also intended as solutions to problems encountered with previous methods.

Also, a new class of application simulation or application definition tools have entered the market. These tools are designed to bridge the communication gap between business users and the IT organization — and also to allow applications to be 'test marketed' before any code is produced. The best of these tools offer:

- electronic whiteboards to sketch application flows and test alternatives
- ability to capture business logic and data needs
- ability to generate high fidelity prototypes that closely imitate the final application
- interactivity
- capability to add contextual requirements and other comments
- Ability for remote and distributed users to run and interact with the simulation [8].

3.7 Requirement Analysis for Inventory POS

3.7.1 User Requirement

- User-friendly graphical interface
- Store customer information
- Check stock report of products

- Update product list
- Delete a product
- Transaction method
- Check daily sell report
- Sell return option

3.7.2 Hardware Requirement

- Ram 1GB or more
- Hard Drive 250GB or more

3.7.3 Software Requirement

- Compatible operating system : windows 7, windows 8, windows 8.1, windows 10
- Adobe PDF
- Microsoft SQL Server

CHAPTER 4

OUR TECHNOLOGIES

4.1 What is C#?

C# is an elegant and type-safe object-oriented language that enables developers to build a variety of secure and robust applications that run on the .NET Framework. You can use C# to create Windows client applications, XML Web services, distributed components, client-server applications, database applications, and much, much more. Visual C# provides an advanced code editor, convenient user interface designers, integrated debugger, and many other tools to make it easier to develop applications based on the C# language and the .NET Framework. [9]

4.2 C# Language

C# syntax is highly expressive, yet it is also simple and easy to learn. The curly-brace syntax of C# will be instantly recognizable to anyone familiar with C, C++ or Java. Developers who know any of these languages are typically able to begin to work productively in C# within a very short time. C# syntax simplifies many of the complexities of C++ and provides powerful features such as null-able value types, enumerations, delegates, lambda expressions and direct memory access, which are not found in Java. C# supports generic methods and types, which provide increased type safety and performance, and iterators, which enable implementers of collection classes to define custom iteration behaviors that are simple to use by client code. Language-Integrated Query (LINQ) expressions make the strongly-typed query a first-class language construct.

As an object-oriented language, C# supports the concepts of encapsulation, inheritance, and polymorphism. All variables and methods, including the Main method, the application's entry point, are encapsulated within class definitions. A class may inherit directly from one parent class, but it may implement any number of interfaces. Methods that override virtual methods in a parent class require the **override** keyword as a way to avoid accidental redefinition. In C#, a struct is like a lightweight class; it is a stack-allocated type that can implement interfaces but does not support inheritance.

In addition to these basic object-oriented principles, C# makes it easy to develop software components through several innovative language constructs, including the following:

- Encapsulated method signatures called *delegates*, which enable type-safe event notifications.
- Properties, which serve as assessors for private member variables.
- Attributes, which provide declarative metadata about types at run time.
- Inline XML documentation comments.
- Language-Integrated Query (LINQ) which provides built-in query capabilities across a variety of data sources.

If you have to interact with other Windows software such as COM objects or native Win32 DLLs, you can do this in C# through a process called "Interop." Interop enables C# programs to do almost anything that a native C++ application can do. C# even supports pointers and the concept of "unsafe" code for those cases in which direct memory access is absolutely critical [9].

4.3 Why Use C#?

C# is an elegant, simple, type-safe, object-oriented language that allows enterprise programmers to build a breadth of applications.

C# also gives you the capability to build durable system-level components by virtue of the following features [10]:

- Full COM/Platform support for existing code integration.
- Robustness through garbage collection and type safety.
- Security provided through intrinsic code trust mechanisms.
- Full support of extensible metadata concepts.

You can also interoperate with other languages, across platforms, with legacy data, by virtue of the following features: [10]

- Full interoperability support through COM+ 1.0 and .NET Framework services with tight library-based access.
- XML support for web-based component interaction.
- Versioning to provide ease of administration and deployment.

4.4 C# WinForm

Windows Forms is a graphical user interface application programming interface (API) included as a part of Microsoft's .NET Framework. As of 13 May 2008, Mono's System.Windows.Forms 2.0 is API complete. Simply put, WinForms is a library for creating GUI applications [11].

4.5 Why use WinForm?

I choose WinForms because [12]:

- It's been around for a long time and you have a *large* control supply that you can use.
- There are *a lot* of good resources on WinForms to learn from and to get new controls from.
- WinForms has some pretty nice "drag-n-drop" features for data manipulation and listing
- Windows Forms is certainly more of a mature technology
- WinForms is very good for developing Windows Look and Feel Applications

4.6 What is a SQL Server?

SQL Server is a Microsoft product used to manage and store information. Technically, SQL Server is a “relational database management system” (RDMS). Broken apart, this term means two things. First, that data stored inside SQL Server will be housed in a “relational database”, and second, that SQL Server is an entire “management system”, not just a database. SQL itself stands for Structured Query Language. This is the language used to manage and administer the database server [13].

4.7 Why use a SQL Server?

SQL Server is an application for storing information inside a “table” structure, let’s examine the reasons why you would use a Database rather than a spreadsheet or some other program for data storage.

Imagine you’re creating an application for storing sales transactions. We’ll start by saving just a few columns of information such as the Item Sold, Quantity, Price, Sale Date, and the Customer sold to. One of the first storage options to consider is saving this information in a large text file. There are benefits to text file saves such as quick write times. The problem with text files is during a read, if the text file is large, it can take quite a bite of time to open and scan the contents of the file looking for what we want. Also, if we wanted to see all the sales to a specific customer, the entire text file would have to be read, and every line occurrence of the customer name would need to be saved in some temporary place until we had them all. If we saved to a spreadsheet instead of a text file, we would have a Sort feature built in. So we may be able to find all the sales to a specific customer quicker, but again, if the file was large, opening the spreadsheet could take a great deal of time.

In addition, what if we wanted to save the customers address as well as their name, now instead of saving five pieces of information (Item Sold, Quantity, Price, Sale Date, and the Customer sold to), we’ll be saving nine columns of information (all the previous plus Address, City, State, and Zip). This means were going to almost double the size of our text file or spreadsheet to accommodate this additional customer data. However, if we used a database, we could save the sales data and the customer address data in two separate places, so the size of the sales data wouldn’t get any larger. When we wanted a report showing the customers’ address, we could “Relate” or link the address data to the sales data.

Not only would our sales information be smaller in a database, but the actual address data would be smaller as well. In a spreadsheet or text file, each sales line would include a complete address. In a database, the address would only be recorded once. It wouldn’t matter if the customer made 100 or 100,000 purchases. All sales records would point to, or “Relate” to, that same single address line.

This “Relating” of data, so sizes stay small is one benefit of a database. In addition, reading and writing to database is very fast. Plus, many databases support having multiple users access the same data at the same time. Something text files and spreadsheets don’t do. Also, the amount or volume of information a database can store is almost unlimited, unlike a spread sheet where there is a fixed number or rows that can be saved [13].

4.8 Three Layer Architecture

3-layer architecture is an emergent architecture that provides a number of strong benefits. Layer indicates logical separation of components, such as having distinct namespaces and classes for the Database Access Layer, Business Logic Layer and User Interface Layer [14].

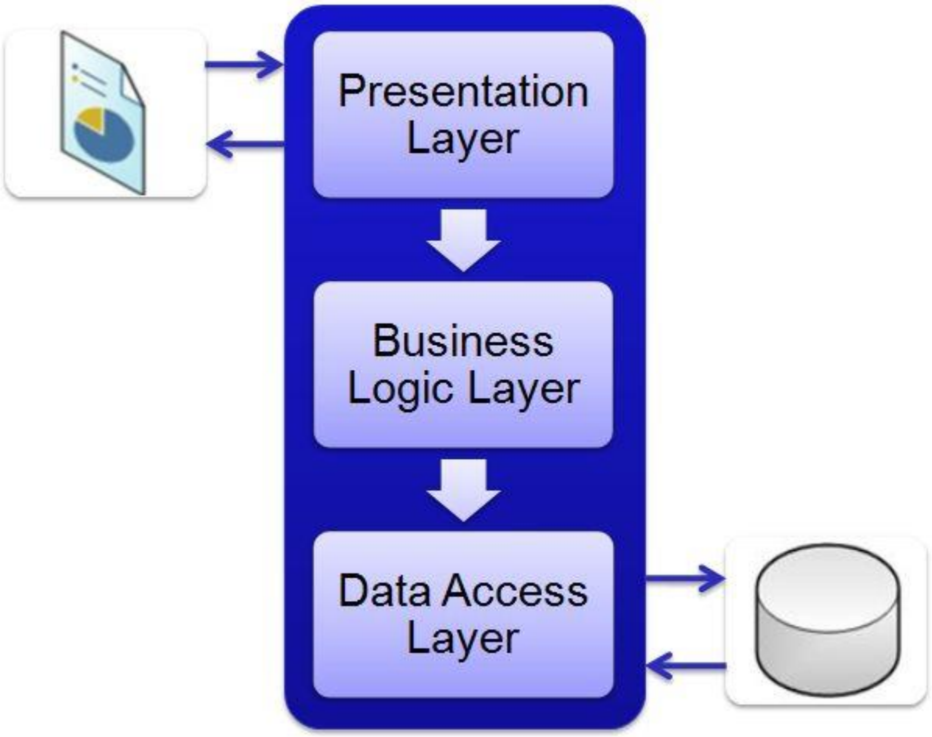


Figure 4.1 three layer architecture

4.8.1 Presentation Layer

Presentation Layer is the only layer which is directly connected with the user. So in this matter, it’s also a really important layer for marketing purposes. Presentation Layer is mainly used for getting user data and then passing it to Business Logic Layer for further procedure, and when data is received in Value Object then it’s responsible to represent value object in the appropriate form which user can understand [14].

4.8.2 Business Logic Layer

Business Logic Layer (BUS) works as a bridge between Presentation Layer and DAO. All the user values received from the presentation layer are being passed to BUS. The results received from the DAO are in row data in Data Table format but in BUS it's converting into Value Objects (VO). Business Logic Layer (BUS) is the most important class in the whole architecture because it mainly contains all the business logic of the program. Whenever a user wants to update the business logic of the program only need to update this class [14].

4.8.3 Database Access Layer

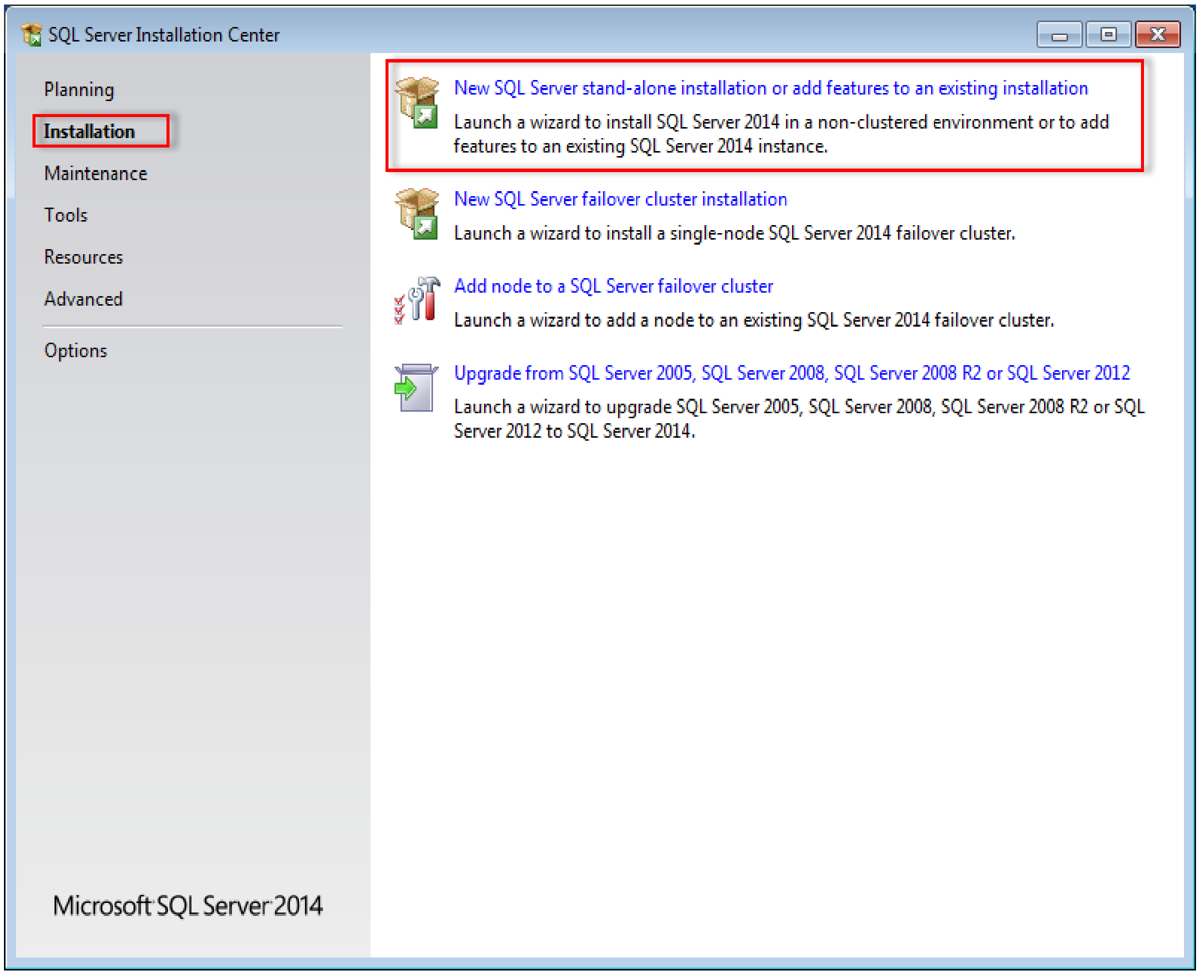
Database Access Layer (DAO) builds the query based on received parameters from the Business Logic Layer and passes it the dbConnection class for execution. And simple return results from the dbConnection class to Business Logic Layer [14].

CHAPTER 5

SOFTWARE INSTALLATION PROCESS

5.1 SQL Server Installation

1. Log in as an administrator on the server (of a workstation in case of a standalone installation).
2. Go to **Inventory & POS** folder then **SQL Server 2014** and start the installation by clicking **Setup.exe**.
3. The **SQL Server Installation Center** screen will be displayed.



New SQL Server stand-alone installation or add features to an existing installation
Launch a wizard to install SQL Server 2014 in a non-clustered environment or to add features to an existing SQL Server 2014 instance.

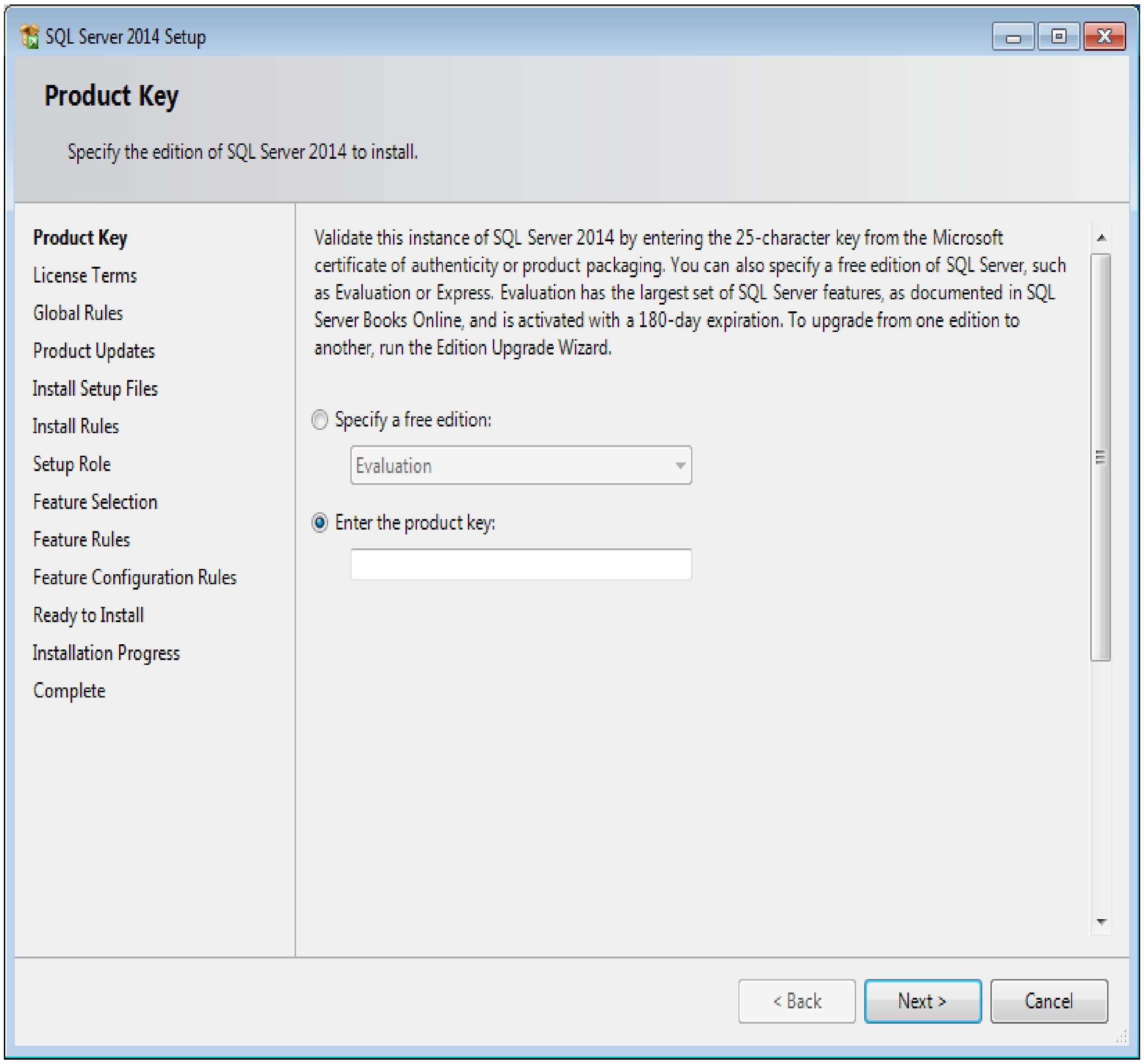
New SQL Server failover cluster installation
Launch a wizard to install a single-node SQL Server 2014 failover cluster.

Add node to a SQL Server failover cluster
Launch a wizard to add a node to an existing SQL Server 2014 failover cluster.

Upgrade from SQL Server 2005, SQL Server 2008, SQL Server 2008 R2 or SQL Server 2012
Launch a wizard to upgrade SQL Server 2005, SQL Server 2008, SQL Server 2008 R2 or SQL Server 2012 to SQL Server 2014.

Figure 5.1 SQL server installation center

4. Click **Installation** on the left panel. Next, click **New SQL Server Stand-alone installation or add features to an existing installation**.



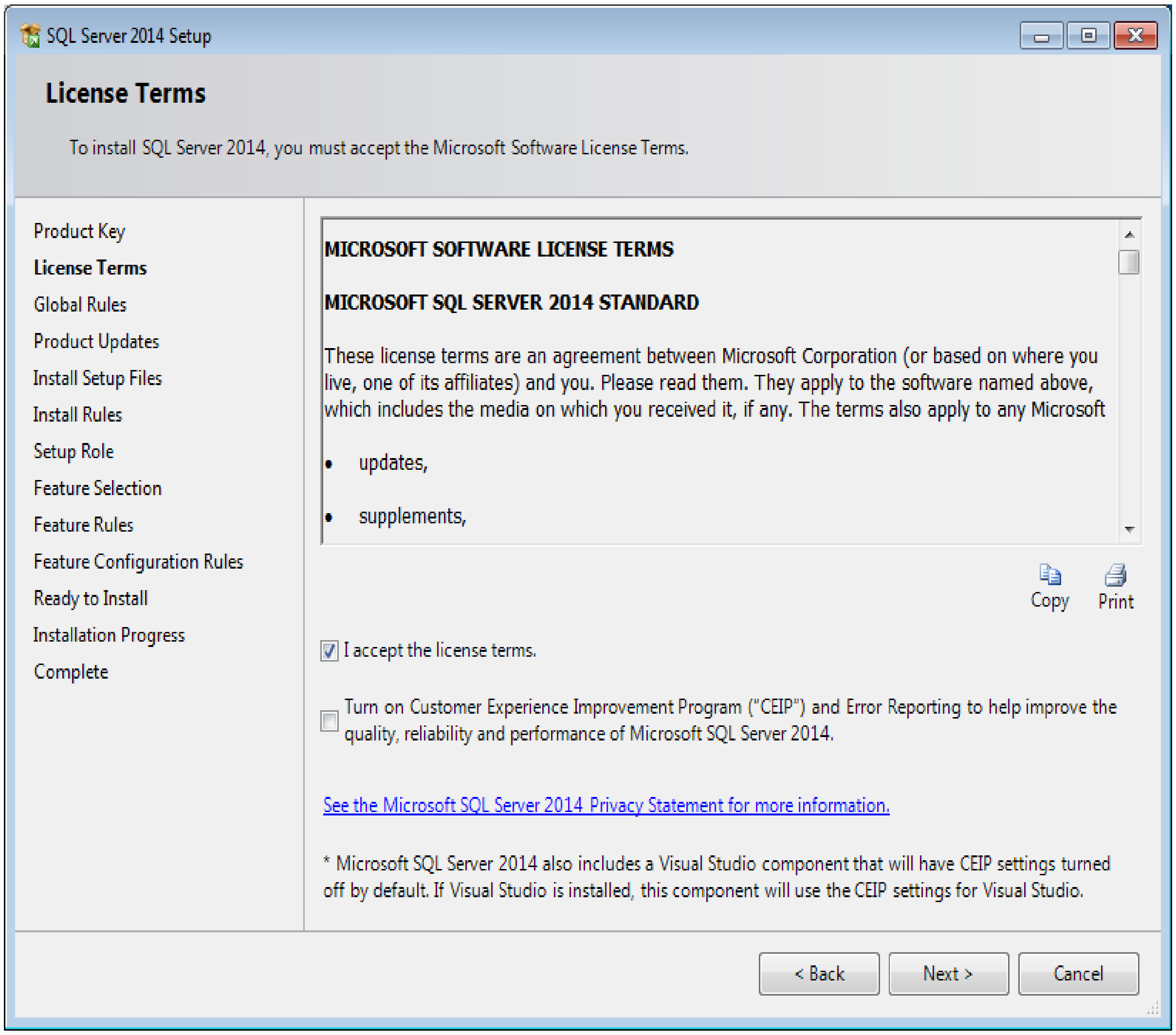
Validate this instance of SQL Server 2014 by entering the 25-character key from the Microsoft certificate of authenticity or product packaging. You can also specify a free edition of SQL Server, such as Evaluation or Express. Evaluation has the largest set of SQL Server features, as documented in SQL Server Books Online, and is activated with a 180-day expiration. To upgrade from one edition to another, run the Edition Upgrade Wizard.

○ Specify a free edition: Evaluation

● Enter the product key:

Figure 5.2 SQL server Product Key Entry

5. In the **Product Key** screen, type the product key at **Enter the product key** if the product key is not filled automatically. Click **Next** to continue. The screen with the license terms will be displayed.



SQL Server 2014 Setup

License Terms

To install SQL Server 2014, you must accept the Microsoft Software License Terms.

MICROSOFT SOFTWARE LICENSE TERMS

MICROSOFT SQL SERVER 2014 STANDARD

These license terms are an agreement between Microsoft Corporation (or based on where you live, one of its affiliates) and you. Please read them. They apply to the software named above, which includes the media on which you received it, if any. The terms also apply to any Microsoft

- updates,
- supplements,

[✓] I accept the license terms.

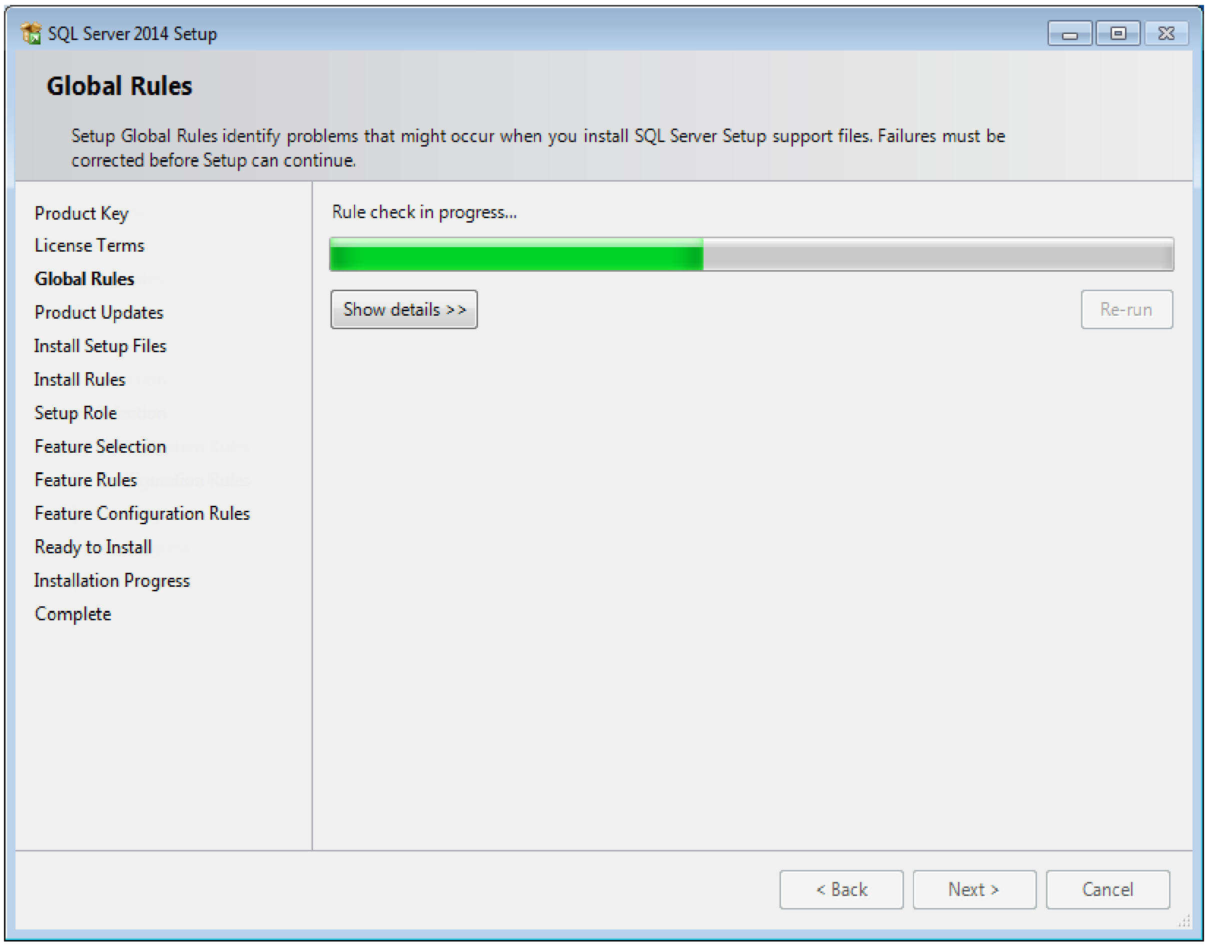
[ ] Turn on Customer Experience Improvement Program ("CEIP") and Error Reporting to help improve the quality, reliability and performance of Microsoft SQL Server 2014.

See the Microsoft SQL Server 2014 Privacy Statement for more information.

* Microsoft SQL Server 2014 also includes a Visual Studio component that will have CEIP settings turned off by default. If Visual Studio is installed, this component will use the CEIP settings for Visual Studio.

Figure 5.3 SQL server License Terms

6. To continue, select **I accept the license terms** and click **Next**.


SQL Server 2014 Setup

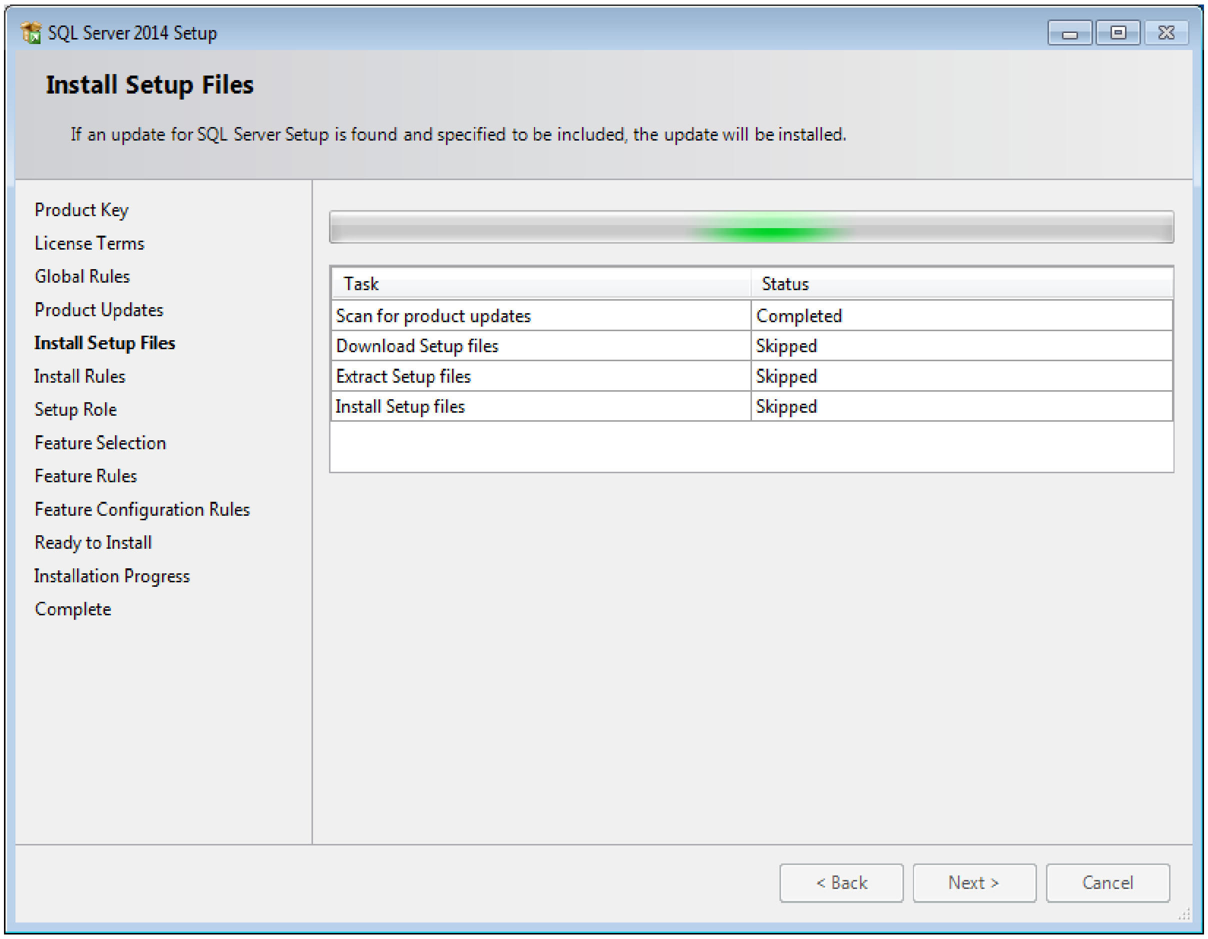
Global Rules

Setup Global Rules identify problems that might occur when you install SQL Server Setup support files. Failures must be corrected before Setup can continue.

Rule check in progress...

Figure 5.4 SQL server setup global Rules

7. Ecks will be performed to identify possible problems when installing the SQL Server Setup support files. The SQL Server Setup support files are needed for the installation of MS SQL Server 2014 and are automatically installed via this installation wizard. Click **Show details** >> to view the details of these checks. When all checks are passed, the results are displayed in this screen. When a check has failed, a red icon will be displayed. You can click the link in the **Status** column for more information about the check and the error. After solving the issue, click **Re-run** to perform the checks again.

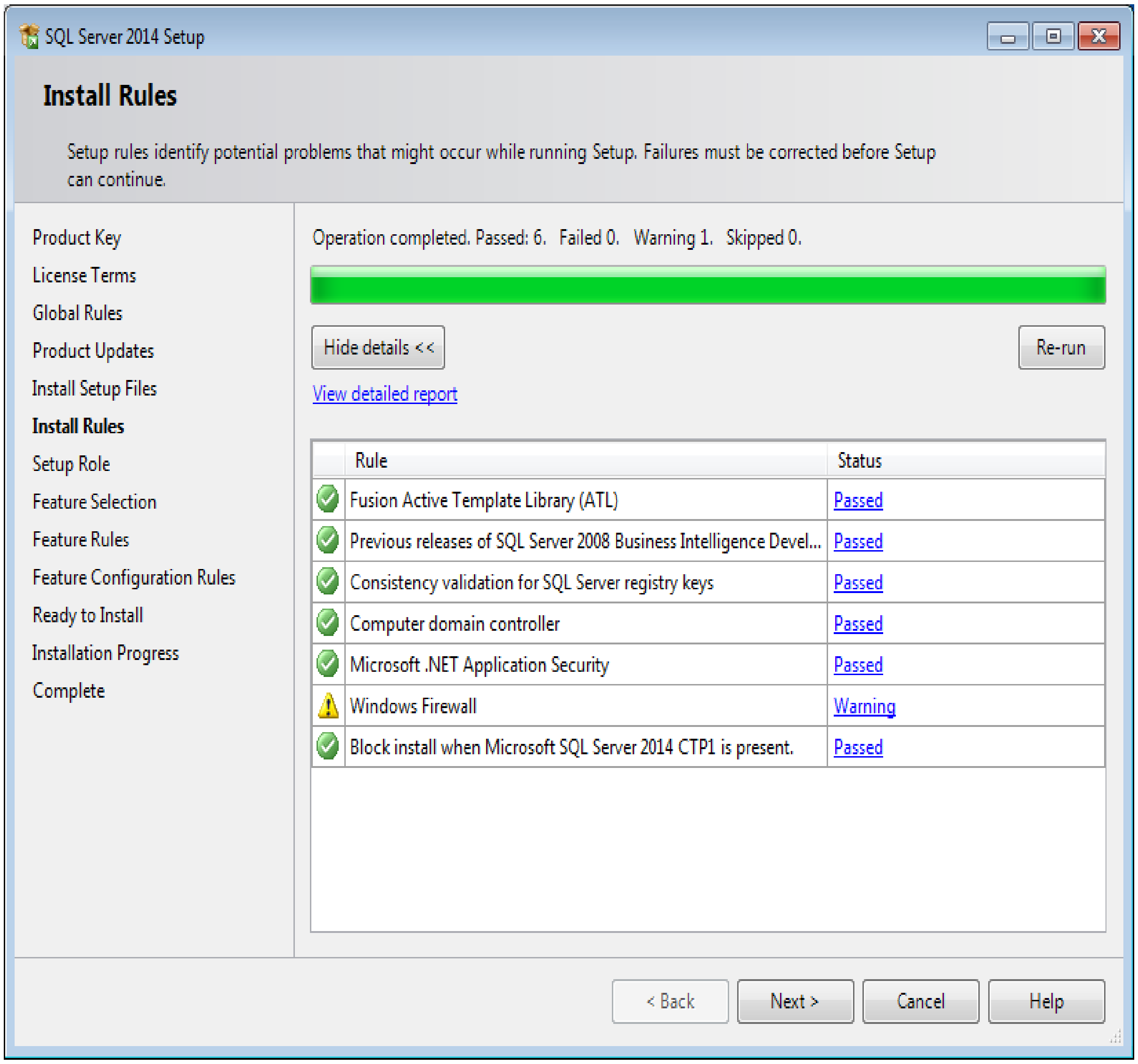


If an update for SQL Server Setup is found and specified to be included, the update will be installed.

Task	Status
Scan for product updates	Completed
Download Setup files	Skipped
Extract Setup files	Skipped
Install Setup files	Skipped

Figure 5.5 SQL server Install setup files

8. The installation wizard checks for updates and installs the available update files. As soon as this is done, the screen will close automatically and the next screen will be displayed:



Install Rules

Setup rules identify potential problems that might occur while running Setup. Failures must be corrected before Setup can continue.

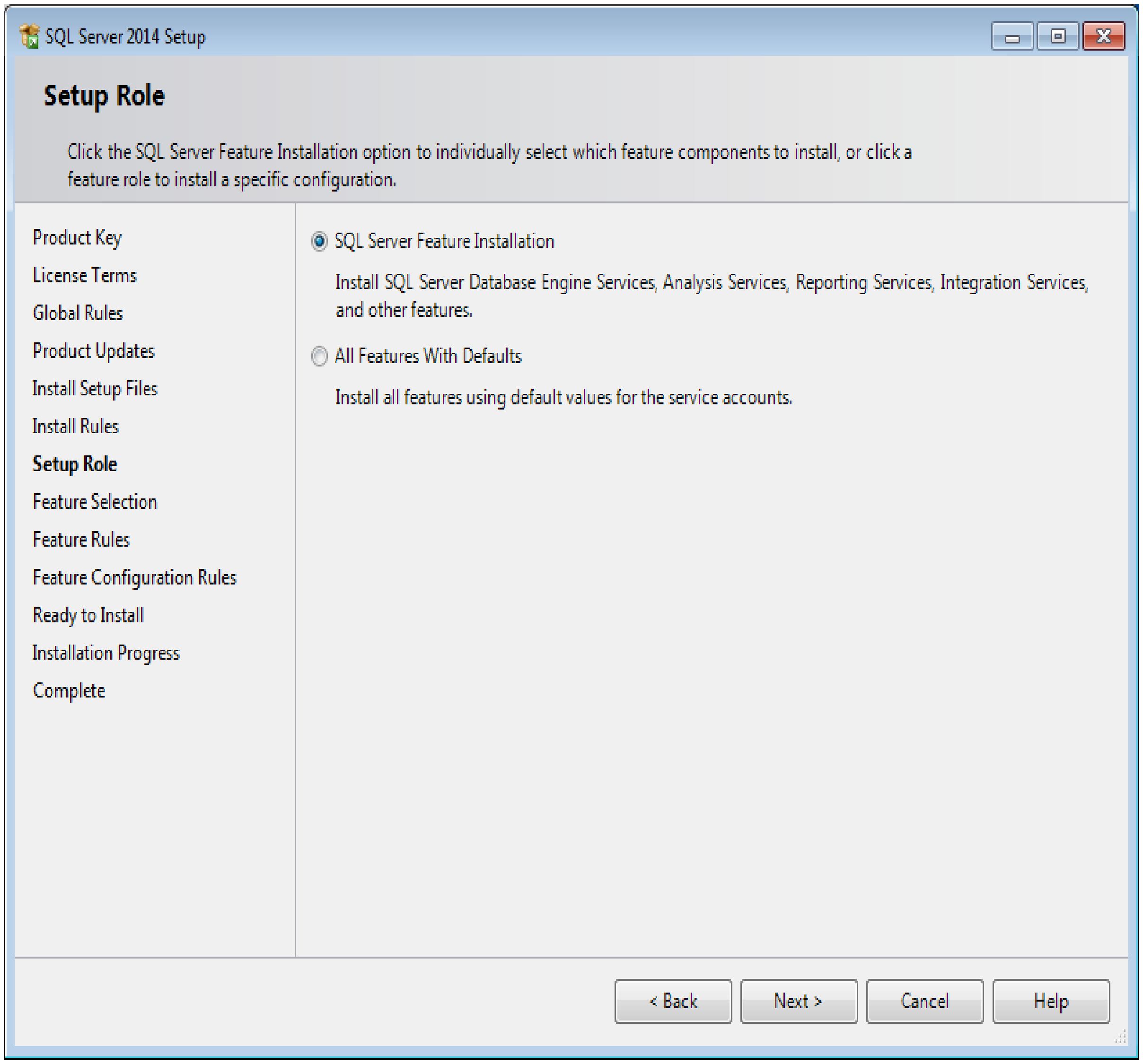
Operation completed. Passed: 6. Failed 0. Warning 1. Skipped 0.

	Rule	Status
✓	Fusion Active Template Library (ATL)	Passed
✓	Previous releases of SQL Server 2008 Business Intelligence Devel...	Passed
✓	Consistency validation for SQL Server registry keys	Passed
✓	Computer domain controller	Passed
✓	Microsoft .NET Application Security	Passed
⚠	Windows Firewall	Warning
✓	Block install when Microsoft SQL Server 2014 CTP1 is present.	Passed

Figure 5.6 SQL server Install Rules

9. Checks will be performed to identify potential problems that might occur during the installation. Click **Show details** >> to view the details of these checks. When a check has failed, a red icon will be displayed. You can click the link in the **Status** column for more information about the check and the error. After solving the issue, click **Re-run**.

10. Click **Next** to continue. However, the **Next** button is available only when no checks have failed.



Click the SQL Server Feature Installation option to individually select which feature components to install, or click a feature role to install a specific configuration.

[✓ SQL Server Feature Installation]

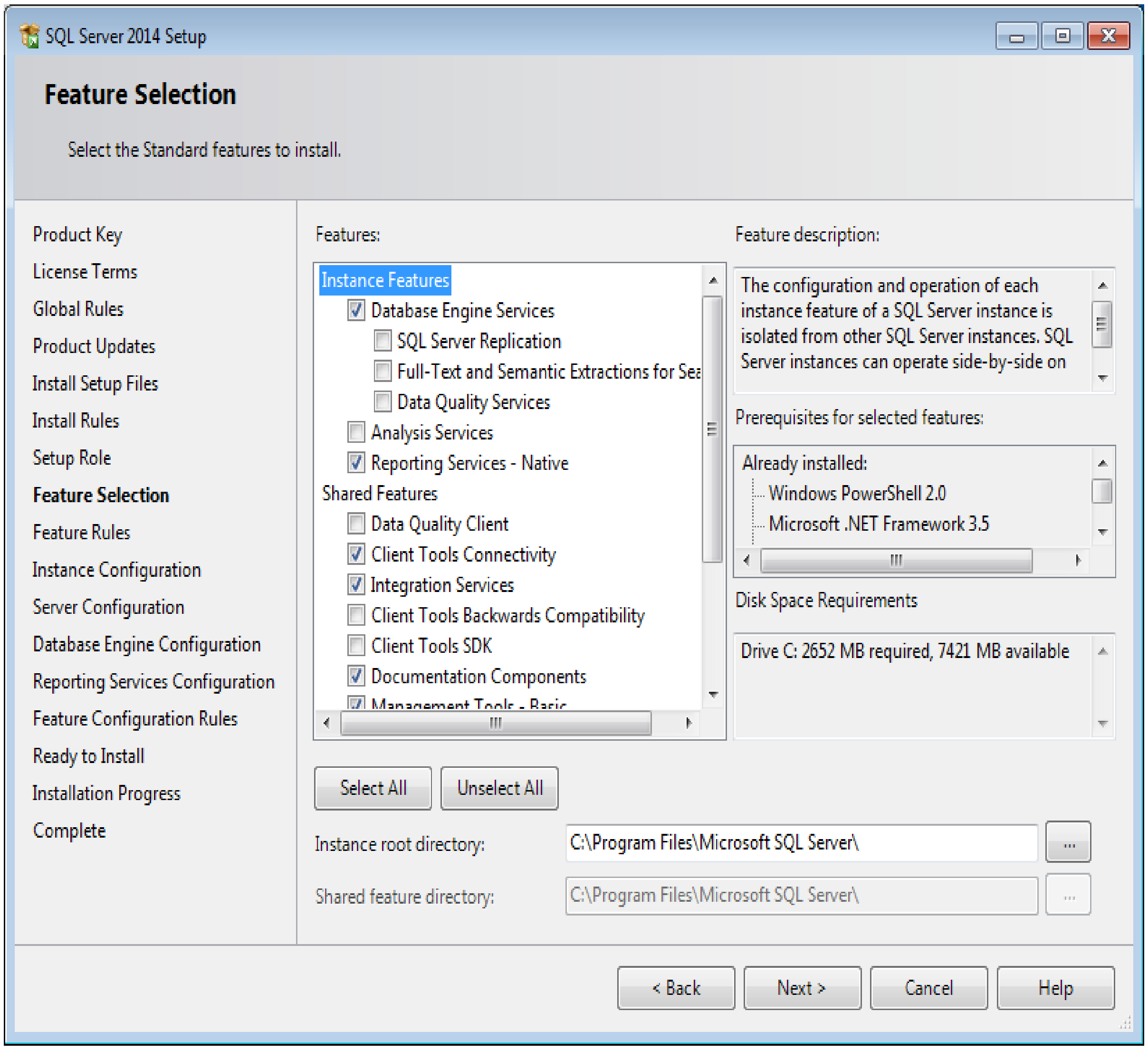
Install SQL Server Database Engine Services, Analysis Services, Reporting Services, Integration Services, and other features.

All Features With Defaults

Install all features using default values for the service accounts.

Figure 5.7 SQL server setup role

11. In the **Setup Role** screen, define if you want to install all components of the SQL Server with default values. Select **SQL Server Feature Installation** and click **Next** to continue.


SQL Server 2014 Setup

Feature Selection

Select the Standard features to install.

Features:

Instance Features
- [✓] Database Engine Services
  - [ ] SQL Server Replication
  - [ ] Full-Text and Semantic Extractions for Sea
  - [ ] Data Quality Services
- [ ] Analysis Services
- [✓] Reporting Services - Native

Shared Features
- [ ] Data Quality Client
- [✓] Client Tools Connectivity
- [✓] Integration Services
- [ ] Client Tools Backwards Compatibility
- [ ] Client Tools SDK
- [✓] Documentation Components
- [✓] Management Tools - Basic

Feature description:

The configuration and operation of each instance feature of a SQL Server instance is isolated from other SQL Server instances. SQL Server instances can operate side-by-side on

Prerequisites for selected features:

Already installed:
- Windows PowerShell 2.0
- Microsoft .NET Framework 3.5

Disk Space Requirements

Drive C: 2652 MB required, 7421 MB available

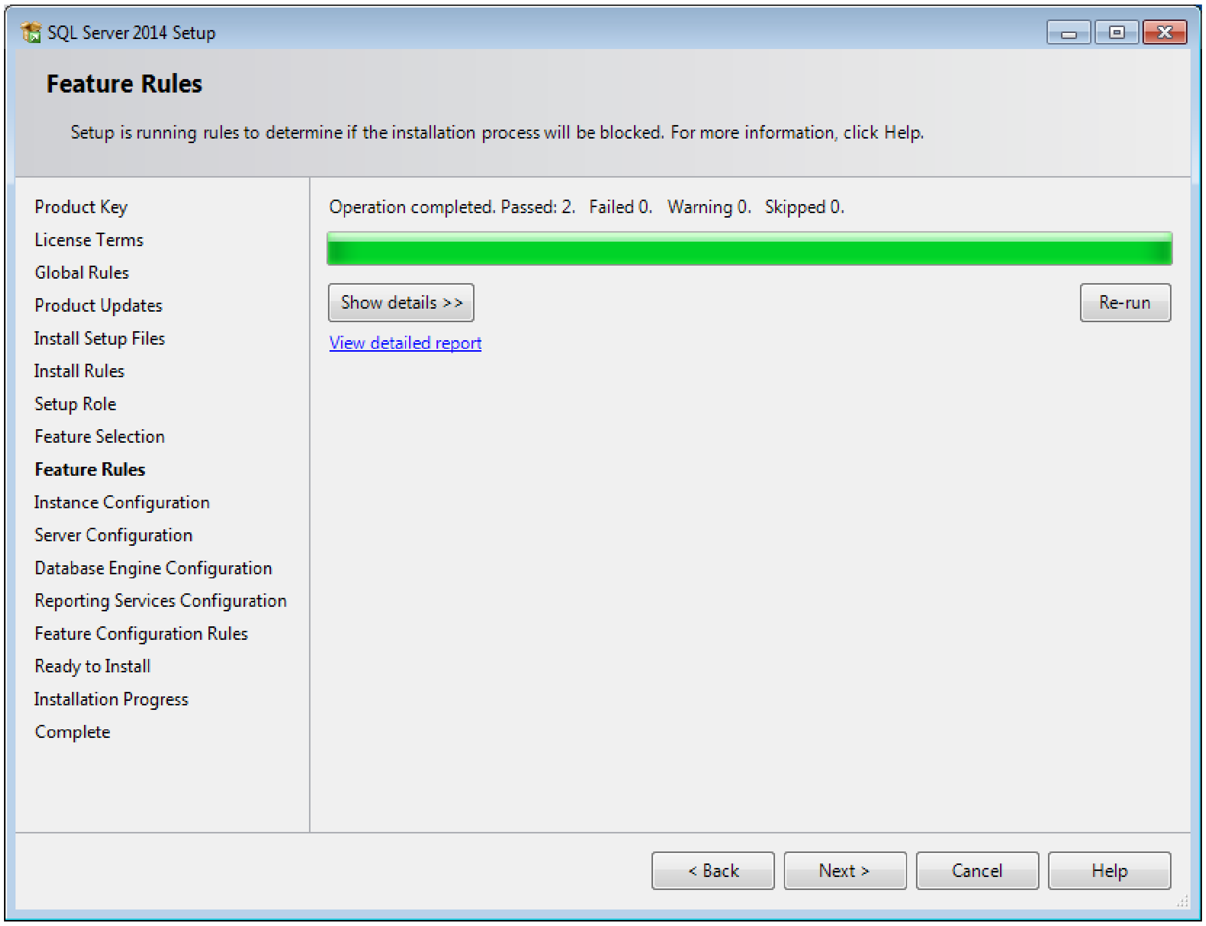
Instance root directory: C:\Program Files\Microsoft SQL Server\

Shared feature directory: C:\Program Files\Microsoft SQL Server\

Figure 5.8 SQL server feature selection

12. Select the features to install. When you select a feature, the corresponding information about this feature will be displayed on the right. Select the following features:
    - Database Engine Services
    - Reporting Services - Native (applicable only when you are using SQL Server Reporting services and/or Exact Globe Next with extended functionality)
    - Client Tools Connectivity
    - Integration Services
    - Documentation Components
    - Management Tools - Basic
    - Management Tools - Complete

    Besides selecting the features, you do not have to change anything else. Click **Next** to continue.


SQL Server 2014 Setup

Feature Rules

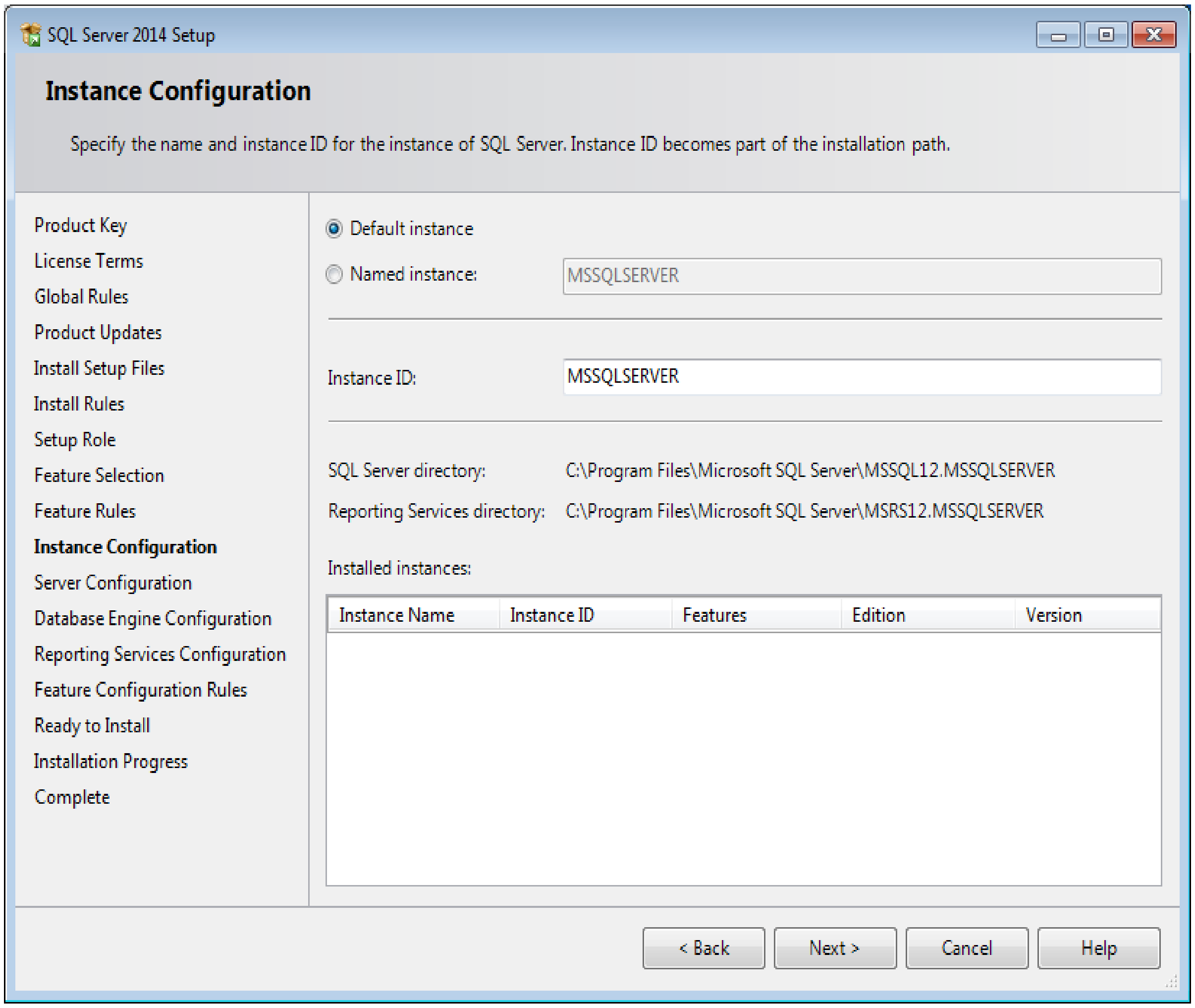
Setup is running rules to determine if the installation process will be blocked. For more information, click Help.

Operation completed. Passed: 2. Failed 0. Warning 0. Skipped 0.

Figure 5.9 SQL server Feature Rules

13. Checks will be performed to identify issues that might block the installation. Click **Show details** >> to view the details of these checks. When a check has failed, a red icon will be displayed. You can click the link in the **Status** column for more information about the check and the error. After solving the issue, click **Re-run.**

14. Click **Next** to continue. However, the **Next** button is available only when no checks have failed.



SQL Server 2014 Setup

Instance Configuration

Specify the name and instance ID for the instance of SQL Server. Instance ID becomes part of the installation path.

[✓ Default instance]

Named instance: MSSQLSERVER

Instance ID: MSSQLSERVER

SQL Server directory: C:\Program Files\Microsoft SQL Server\MSSQL12.MSSQLSERVER

Reporting Services directory: C:\Program Files\Microsoft SQL Server\MSRS12.MSSQLSERVER

Installed instances:

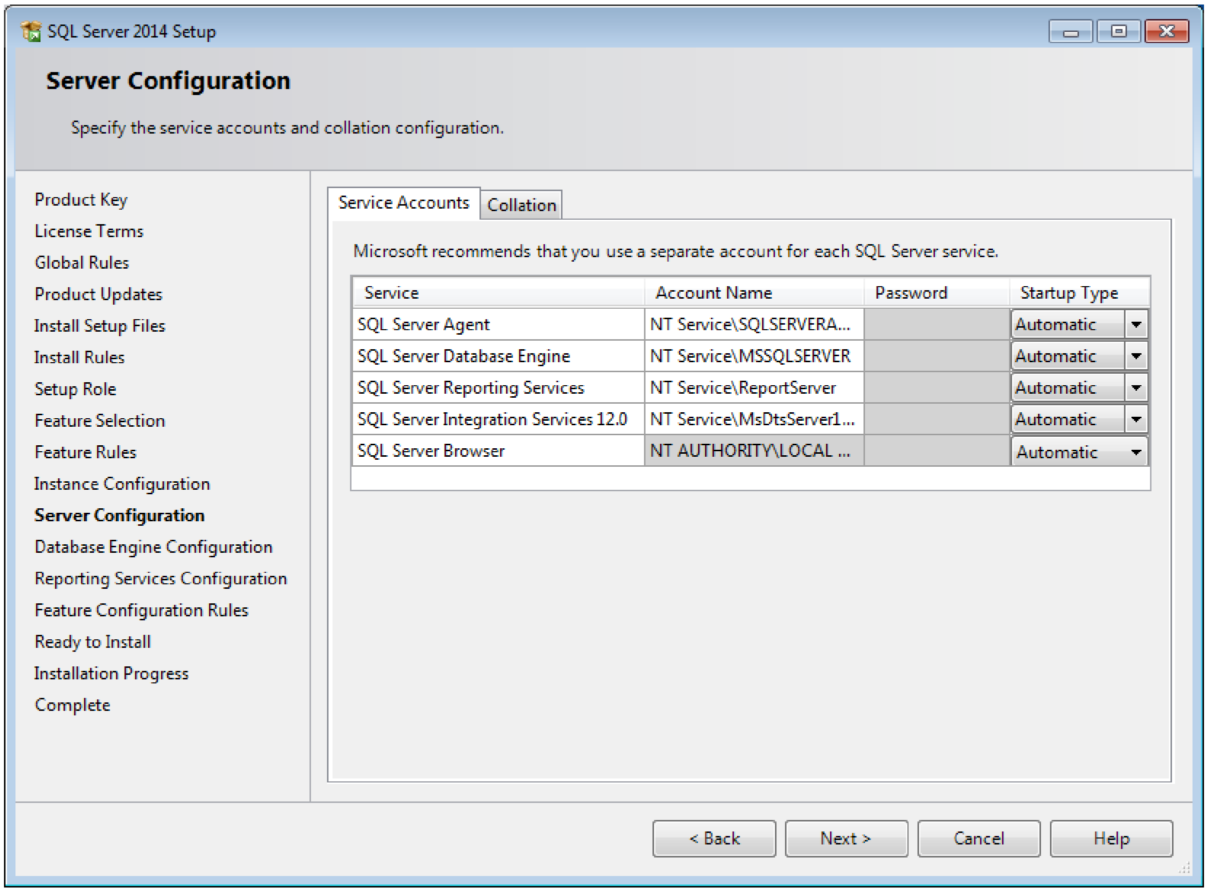
Instance Name	Instance ID	Features	Edition	Version

Figure 5.10 SQL server instance configuration

15. When the actions in the previous screen are completed, the screen will close automatically and the next screen will be displayed. In the **Instance Configuration** screen, define a name for the instance. This is the name for your SQL Server. Click **Next**.

    New installation – When it is a new installation, there is no SQL Server present in your system yet. In this case, select **Default instance**. There is no need to change anything else in the screen. At the bottom of the screen, you can see if there are already other SQL Server instances present. When this is a new installation in a system where no SQL Server is present yet, this part of the screen is empty.

    Installation new instance – When there are already existing SQL Server installations present, you have to select **Named instance** and define a name in the box next to it, for example "SQL2014". The name defined here has to be unique and cannot be the same as the name of the other instances that are already present on your system. The field **Instance ID** will be filled automatically and you do not have to change this manually. At the bottom of the screen, you can see if any other instances are present.


SQL Server 2014 Setup

Server Configuration

Specify the service accounts and collation configuration.

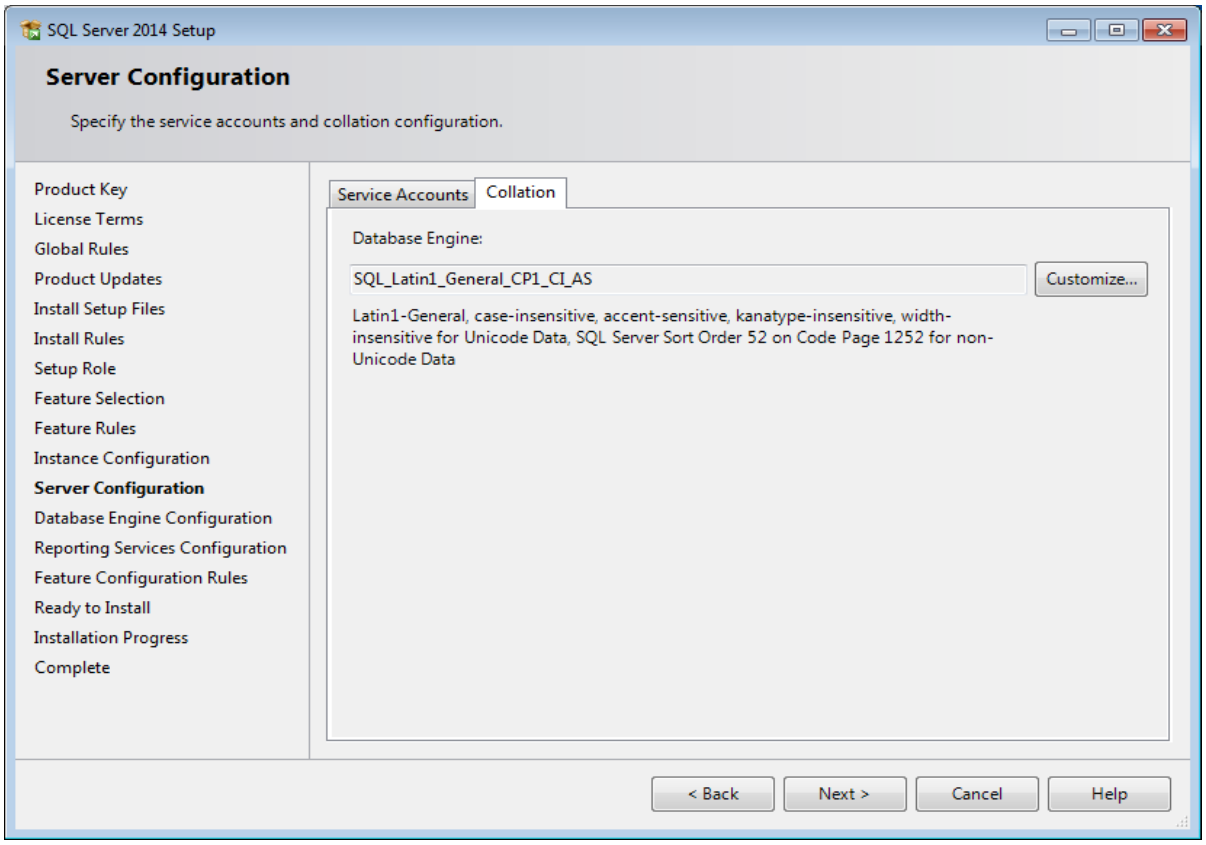
Microsoft recommends that you use a separate account for each SQL Server service.

Service	Account Name	Password	Startup Type
SQL Server Agent	NT Service\SQLSERVERA...		Automatic
SQL Server Database Engine	NT Service\MSSQLSERVER		Automatic
SQL Server Reporting Services	NT Service\ReportServer		Automatic
SQL Server Integration Services 12.0	NT Service\MsDtsServer1...		Automatic
SQL Server Browser	NT AUTHORITY\LOCAL ...		Automatic

Figure 5.11 SQL server configuration (Service Accounts)

16. In this screen, under the **Service Accounts** tab, you can define which account you want to use for starting the MS SQL Server 2014 services. For a standard installation, accept the data as suggested in the screen. Keep in mind that when you select a user for which the password regularly changes, you will also have to change the password in the services every time. If you forget to change the password in the services, the services cannot be started and you will not be able to use the SQL Server until you have updated the password in the services, and have started the services. Depending on the account selected at **Account Name**, you have to enter the corresponding password at **Password**.

17. In the **Startup Type** column, you can define that the service has to start automatically when the system starts. The **SQL Server Database Engine** is the SQL Server itself. This service should always be started in order to use the SQL Server. The **SQL Server Browser** service is a service that is used for identifying the ports that the installation of SQL Server listens to. This service needs to be started to enable network access. The **Startup Type** option of this service should be **Automatic**. The account name of this service cannot be changed.


SQL Server 2014 Setup

Server Configuration

Specify the service accounts and collation configuration.

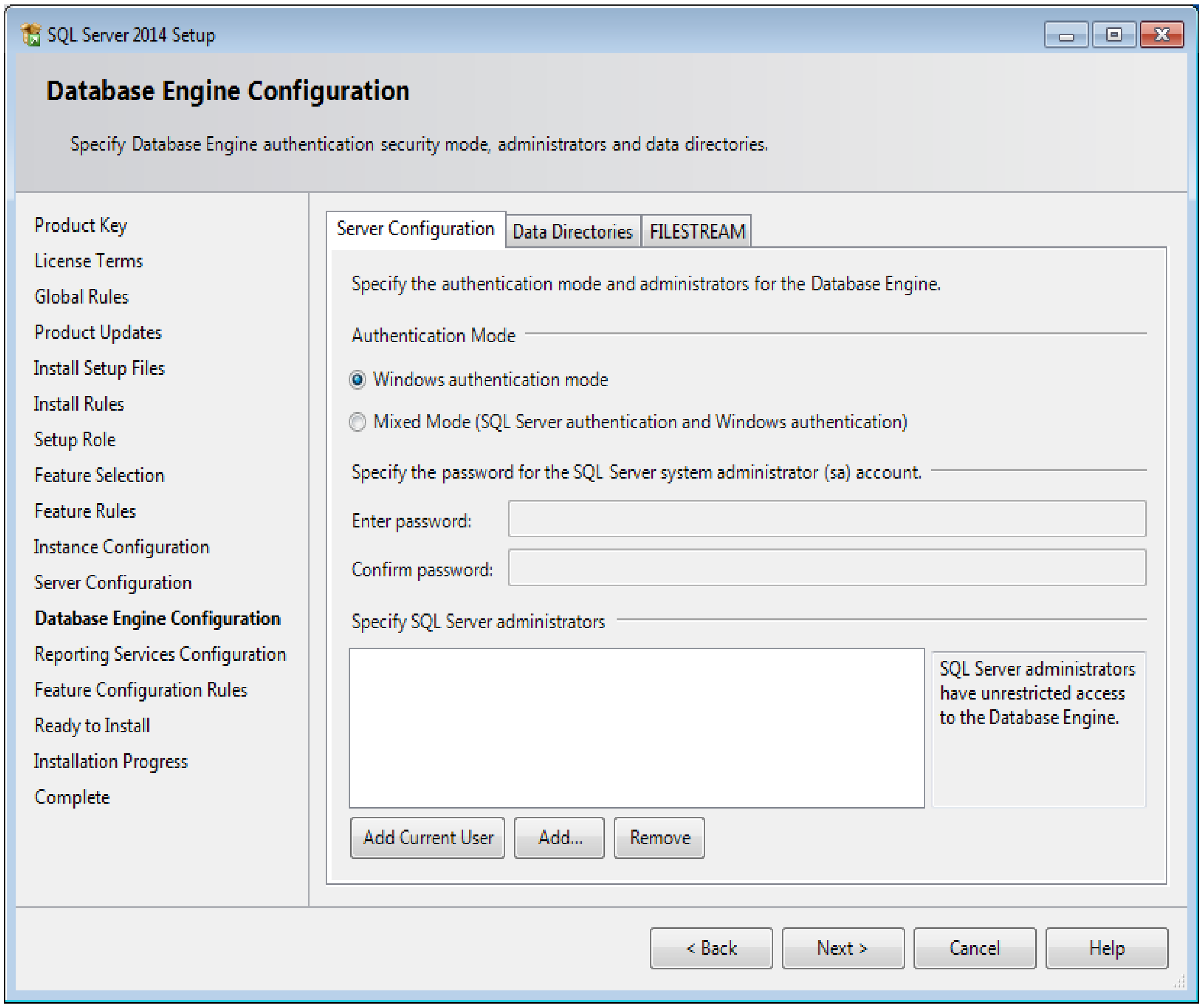
Database Engine:

SQL_Latin1_General_CP1_CI_AS

Latin1-General, case-insensitive, accent-sensitive, kanatype-insensitive, width-insensitive for Unicode Data, SQL Server Sort Order 52 on Code Page 1252 for non-Unicode Data

Figure 5.12 SQL server configuration (Collation)

18. Under the **Collation** tab, you can define the collation (character set) settings. For Western European countries, you can accept the collation as displayed in the screen. Click **Next** to continue. The following screen will be displayed:



SQL Server 2014 Setup

Database Engine Configuration

Specify Database Engine authentication security mode, administrators and data directories.

Specify the authentication mode and administrators for the Database Engine.

Authentication Mode

[✓ Windows authentication mode]

Mixed Mode (SQL Server authentication and Windows authentication)

Specify the password for the SQL Server system administrator (sa) account.

Enter password:

Confirm password:

Specify SQL Server administrators

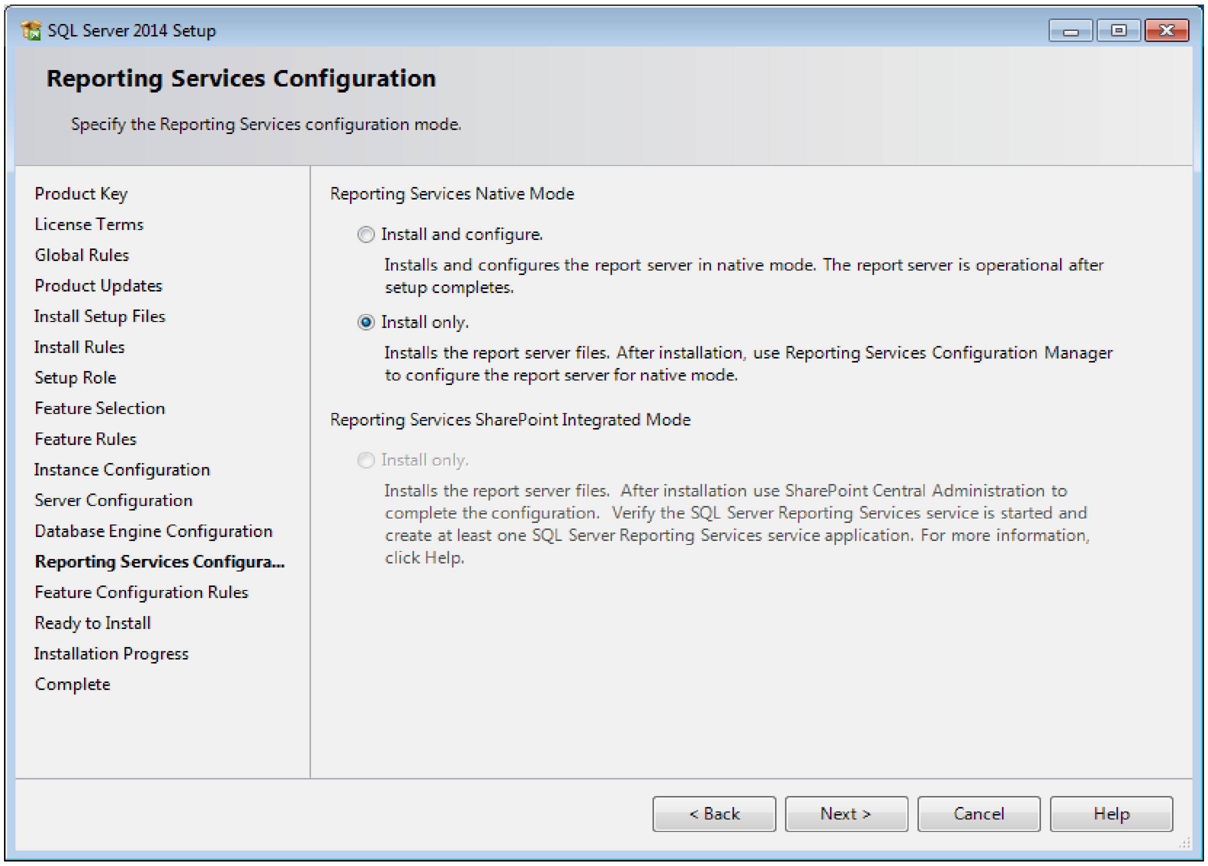
SQL Server administrators have unrestricted access to the Database Engine.

Figure 5.13 SQL server Database Engine configuration

19. Under the **Server Configuration** tab, define how you would like to log on to the SQL Server. When you are working with Exact Synergy Enterprise and/or Exact Globe Next with extended functionality, select **Mixed Mode (SQL Server authentication and Windows authentication)**. Type your password at **Enter password** and **Confirm password**. This password is a password that will be used for the SA user; this default user is the System Administrator within the SQL Server. If you select **Windows authentication**.

20. Click **Add Current User** to add the users that are currently performing the installation as an administrator. Click **Add** to add other users or **Remove** to remove selected users.
21. Click **Next** to continue.

    Note: The following screen will be displayed only if you have selected the **Reporting Services – Native** feature.



Reporting Services Configuration

Specify the Reporting Services configuration mode.

Reporting Services Native Mode

( ) Install and configure.

Installs and configures the report server in native mode. The report server is operational after setup completes.

[✓ Install only.]

Installs the report server files. After installation, use Reporting Services Configuration Manager to configure the report server for native mode.

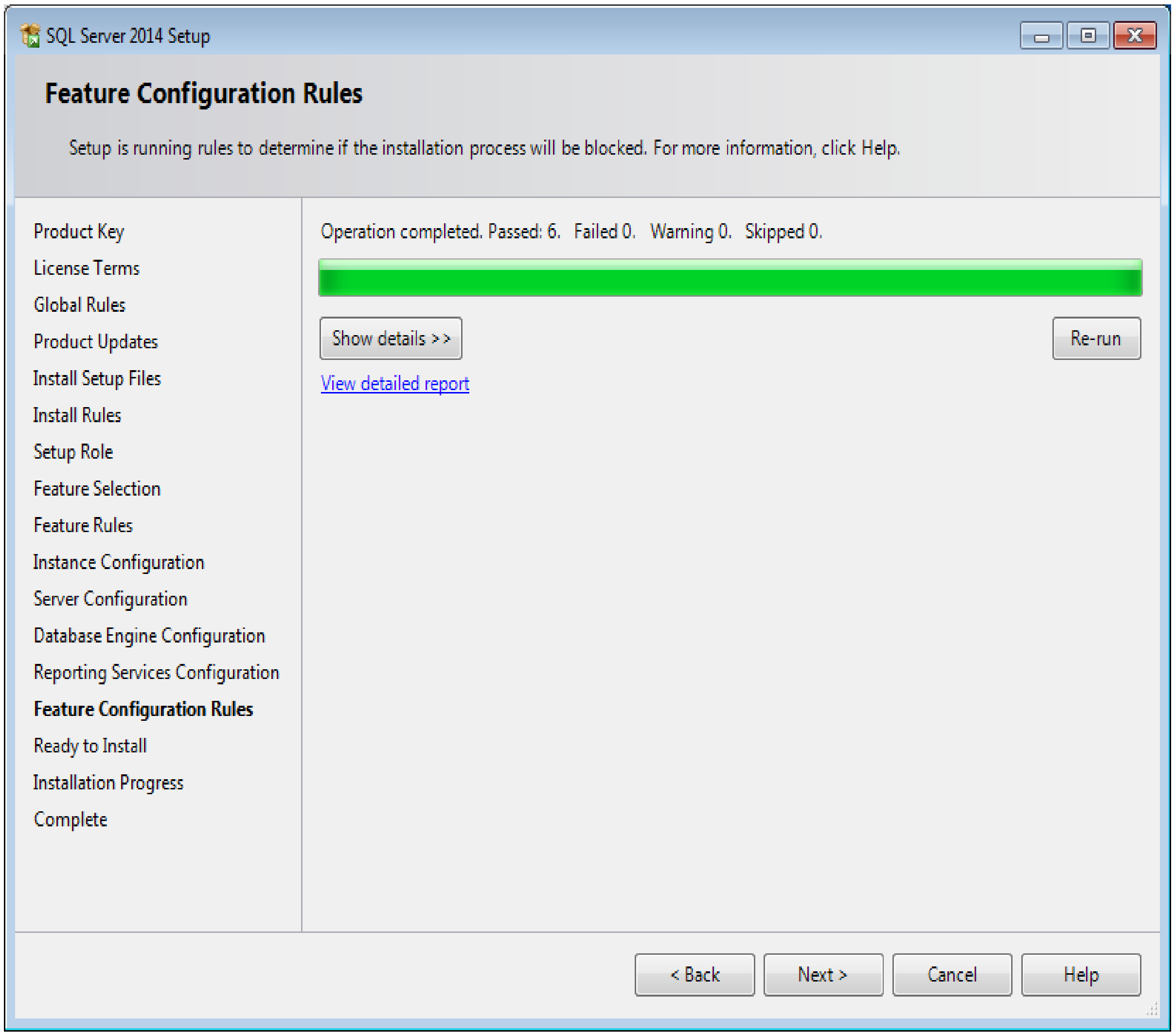
Reporting Services SharePoint Integrated Mode

( ) Install only.

Installs the report server files. After installation use SharePoint Central Administration to complete the configuration. Verify the SQL Server Reporting Services service is started and create at least one SQL Server Reporting Services service application. For more information, click Help.

Figure 5.14 SQL server reporting server configuration

22. In this screen, define if you want to configure the reporting services feature now or if you want to configure it later. Select **Install only** to install the feature and configure it later. Click **Next** to continue.


SQL Server 2014 Setup

Feature Configuration Rules

Setup is running rules to determine if the installation process will be blocked. For more information, click Help.

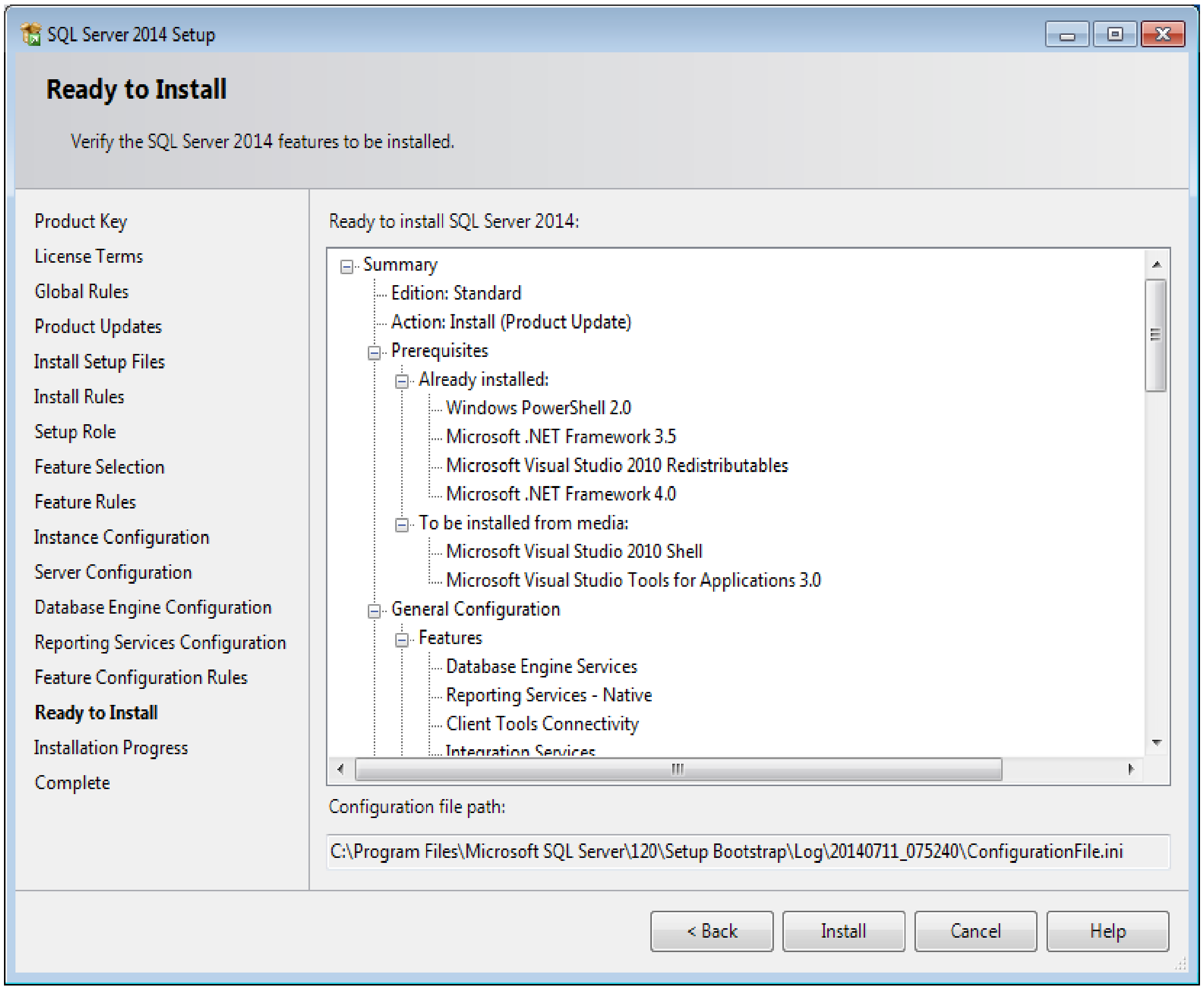
Product Key
License Terms
Global Rules
Product Updates
Install Setup Files
Install Rules
Setup Role
Feature Selection
Feature Rules
Instance Configuration
Server Configuration
Database Engine Configuration
Reporting Services Configuration
Feature Configuration Rules
Ready to Install
Installation Progress
Complete

Operation completed. Passed: 6. Failed 0. Warning 0. Skipped 0.

Figure 5.15 SQL server Feature Configuration rules

23. In the **Feature Configuration Rules** screen, checks will be performed for problems that might occur during the installation. Click **Show details** >> to view the details of these checks. When a check has failed, a red icon will be displayed. You can click the link in the **Status** column for more information about the check and the error. Click **Re-run**.

24. Click **Next** to continue. However, this button is available only when no checks have failed.



Ready to Install

Verify the SQL Server 2014 features to be installed.

Ready to install SQL Server 2014:

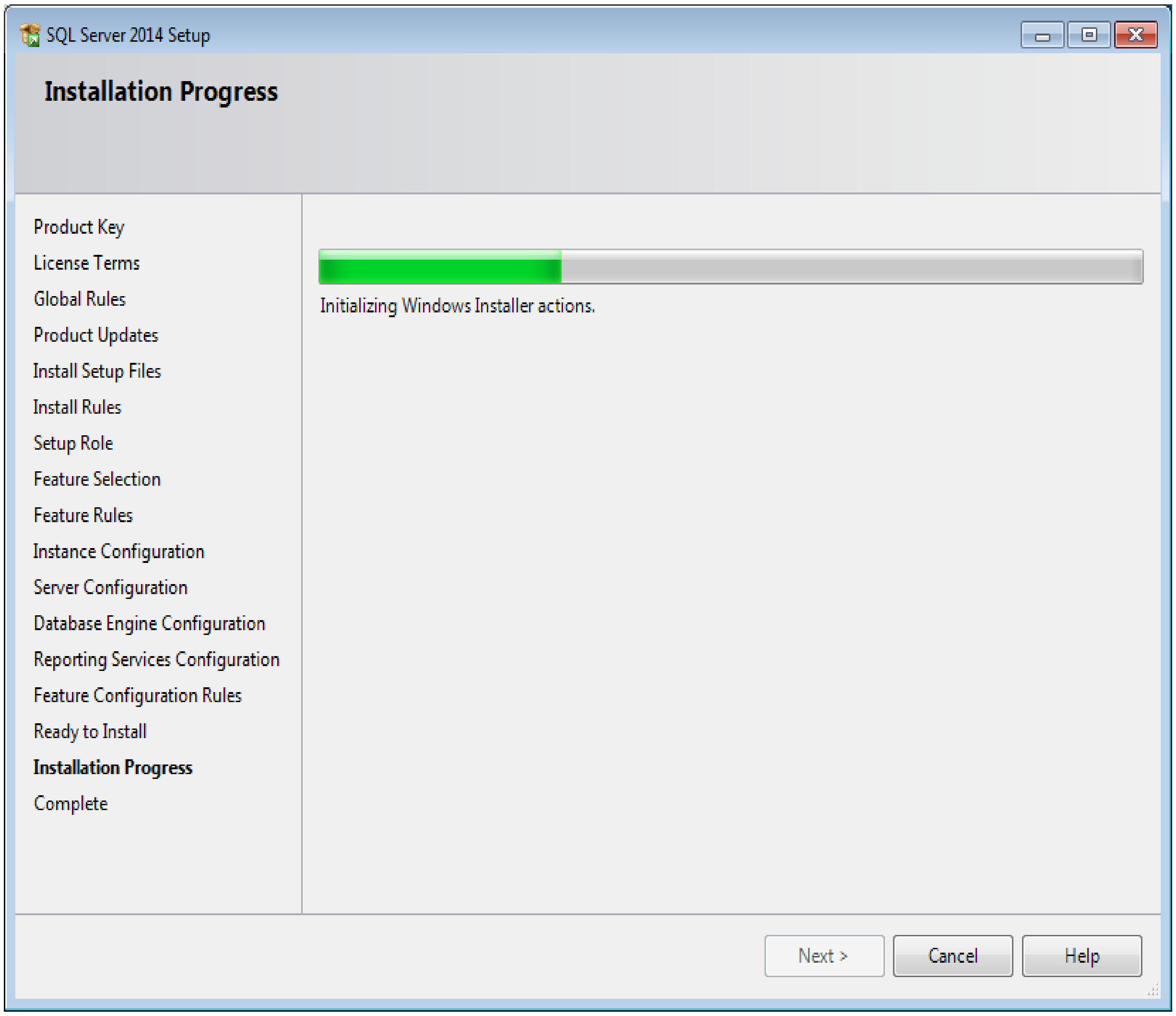
- Summary
  - Edition: Standard
  - Action: Install (Product Update)
  - Prerequisites
    - Already installed:
      - Windows PowerShell 2.0
      - Microsoft .NET Framework 3.5
      - Microsoft Visual Studio 2010 Redistributables
      - Microsoft .NET Framework 4.0
    - To be installed from media:
      - Microsoft Visual Studio 2010 Shell
      - Microsoft Visual Studio Tools for Applications 3.0
  - General Configuration
    - Features
      - Database Engine Services
      - Reporting Services - Native
      - Client Tools Connectivity
      - Integration Services

Configuration file path:

C:\Program Files\Microsoft SQL Server\120\Setup Bootstrap\Log\20140711_075240\ConfigurationFile.ini

Figure 5.16 SQL server ready to install

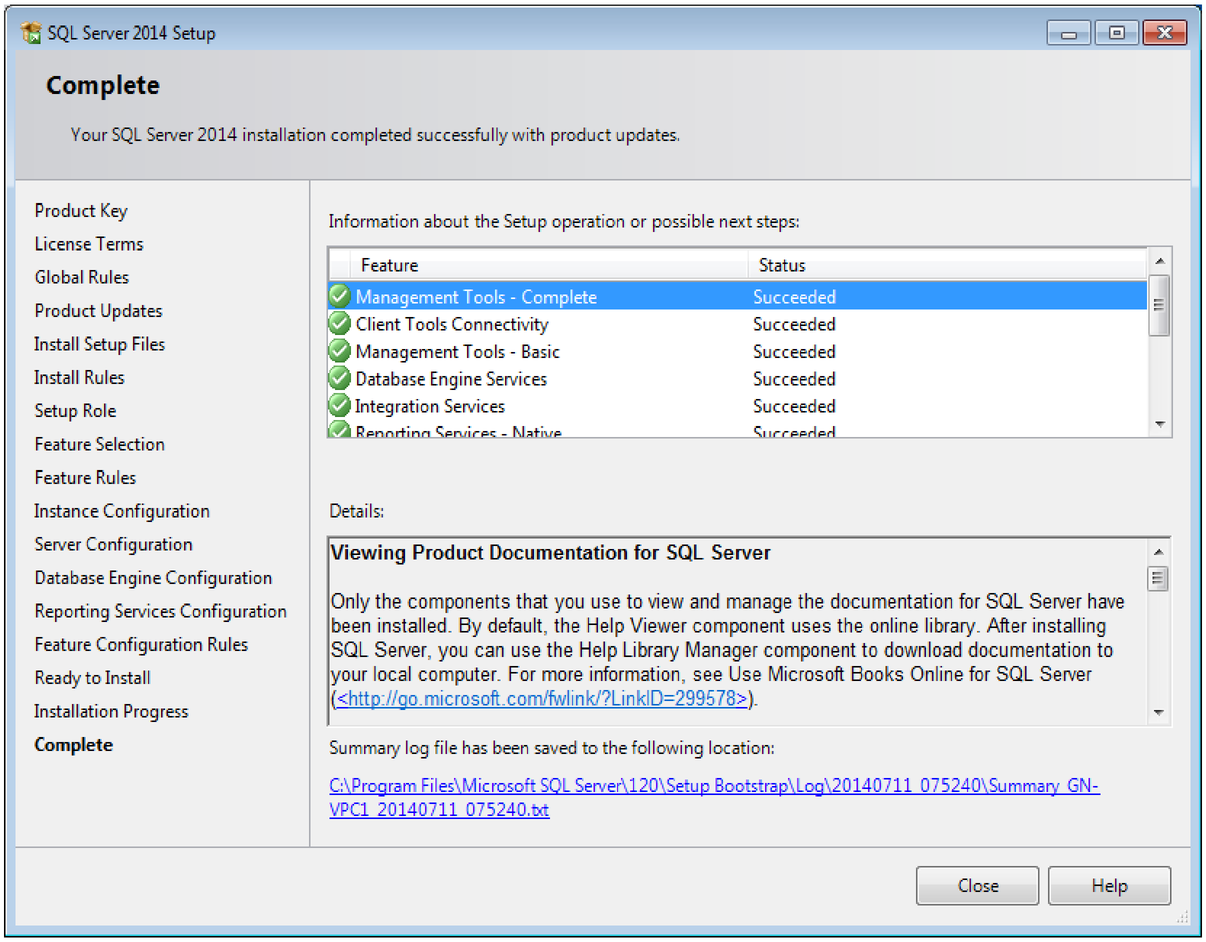
25. In the **Ready to Install** screen, a summary of the defined installation criteria will be displayed. Click **Install** to start the installation.



Initializing Windows Installer actions.

Figure 5.17 SQL server Installation Progress

26. The screen displaying the progress of the installation will displayed. The installation might take some time.


SQL Server 2014 Setup

Complete

Your SQL Server 2014 installation completed successfully with product updates.

Information about the Setup operation or possible next steps:

Feature	Status
✓ Management Tools - Complete	Succeeded
✓ Client Tools Connectivity	Succeeded
✓ Management Tools - Basic	Succeeded
✓ Database Engine Services	Succeeded
✓ Integration Services	Succeeded
✓ Reporting Services - Native	Succeeded

Details:

Viewing Product Documentation for SQL Server

Only the components that you use to view and manage the documentation for SQL Server have been installed. By default, the Help Viewer component uses the online library. After installing SQL Server, you can use the Help Library Manager component to download documentation to your local computer. For more information, see Use Microsoft Books Online for SQL Server (<http://go.microsoft.com/fwlink/?LinkID=299578>).

Summary log file has been saved to the following location:

C:\Program Files\Microsoft SQL Server\120\Setup Bootstrap\Log\20140711_075240\Summary_GN-VPC1_20140711_075240.txt

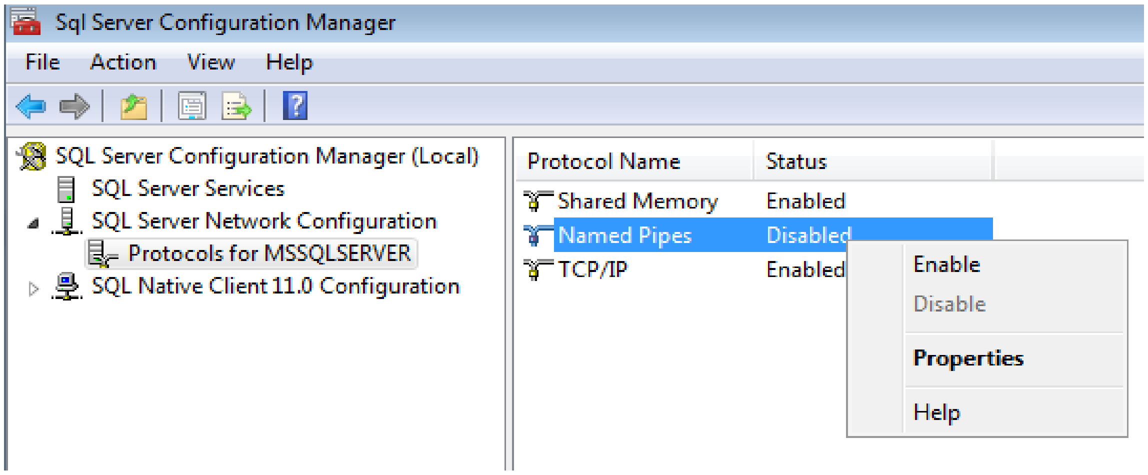
Figure 5.18 SQL server Complete

27. As soon as the installation has completed successfully, the installed components are displayed with a green icon. Click **Close** to close the wizard.
28. To be able to access the SQL Server via a network, you have to activate some protocols. When working with a network environment, you have to check if the protocols are activated. If required, activate the protocols.

1. Start (on the server where MS SQL Server 2014 is installed) **SQL Server Configuration Manager**. In Windows, go to **Start**, select **All Programs**, followed by **Microsoft SQL Server 2014**. Click **Configuration Tools**, and then **SQL Server Configuration_Manager.**

   a. Click **SQL Server Network Configuration**.

b. Select **Protocols for xxxx**. ("xxxx" has to be replaced by the name you have defined for the SQL server during installation.)
c. Right-click the protocols **Named Pipes** and **TCP/IP** and click **Enable** to enable these protocols.


Sql Server Configuration Manager
File
Action
View
Help
SQL Server Configuration Manager (Local)
SQL Server Services
SQL Server Network Configuration
Protocols for MSSQLSERVER
SQL Native Client 11.0 Configuration
Enable
Disable
Properties
Help

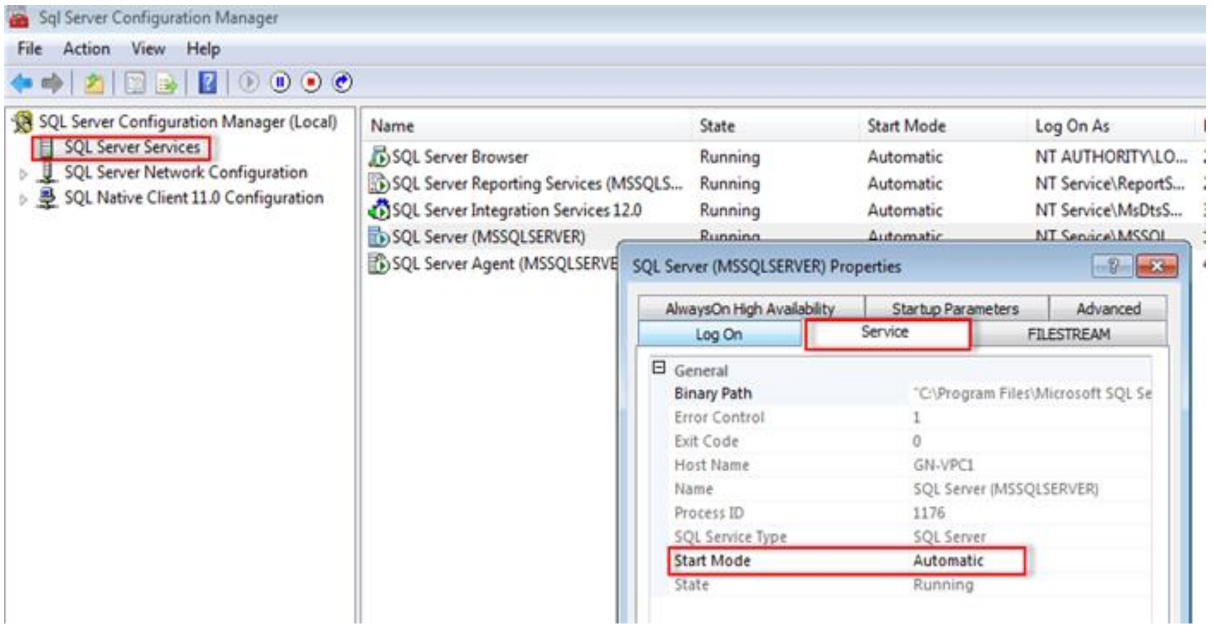
Protocol Name	Status
Shared Memory	Enabled
Named Pipes	Disabled
TCP/IP	Enabled

Figure 5.19 SQL server configuration manager

2. You might also need to start the **SQL Server Browser** service. This service is used for identifying the ports that the installation of SQL Server listens to. When this service is not started, you can use the SQL Server only locally. To check if this service is started, follow these steps:
3. Start (on the server where MS SQL Server 2014 is installed) **SQL Server Configuration Manager**. In Windows, got to **Start**, select **All Programs**, followed by **SQL Server 2014**. Click **Configuration Tools**, and then **SQL Server Configuration Manager**.
   In Windows 8, press the Windows key and the Q key on your keyboard. On the right at **Search** in the **Apps** field, type "SQL Server Configuration Manager". Next, click **SQL Server Configuration Manager** on the left side of the screen.

   a. Click **SQL Server Services**.
   b. On the right side, the **SQL Server Browser** service is displayed. **Note:** When it is not started, the icon is red. When it is started, the icon is green. Ensure that the service starts automatically so that every time the system is restarted, the service will also start.
   c. Right-click **SQL Server Browser** and select **Properties**.
   d. In the **Properties** screen, click the **Service** tab.

e. At **Start Mode**, select **Automatic**, as displayed in the following screen:


Sql Server Configuration Manager
File Action View Help
SQL Server Configuration Manager (Local)
SQL Server Services
SQL Server Network Configuration
SQL Native Client 11.0 Configuration
SQL Server (MSSQLSERVER) Properties
AlwaysOn High Availability
Startup Parameters
Advanced
Log On
Service
FILESTREAM

Name	State	Start Mode	Log On As
SQL Server Browser	Running	Automatic	NT AUTHORITY\LO...
SQL Server Reporting Services (MSSQLS...	Running	Automatic	NT Service\ReportS...
SQL Server Integration Services 12.0	Running	Automatic	NT Service\MsDtsS...
SQL Server (MSSQLSERVER)	Running	Automatic	NT Service\MSSQL...
SQL Server Agent (MSSQLSERVE			

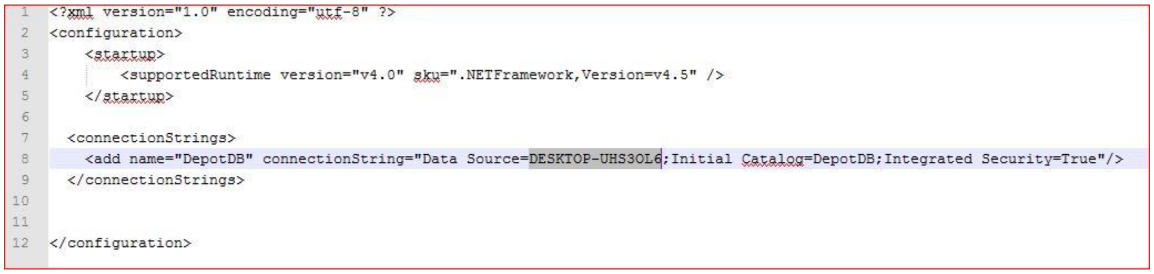
General	
Binary Path	"C:\Program Files\Microsoft SQL Se
Error Control	1
Exit Code	0
Host Name	GN-VPC1
Name	SQL Server (MSSQLSERVER)
Process ID	1176
SQL Service Type	SQL Server
Start Mode	Automatic
State	Running

Figure 5.20 SQL server configuration manager (MSSQLSERVER) properties

f. Click **Apply** to apply the changes. The SQL Server browser will now start automatically every time the system is started.
g. Next, start **SQL Server Browser**.
h. In the **Properties** screen, click the **Logon** tab.
i. Click **Start** to start the service. When the **Start** button is inactive, the **SQL Server Browser** service is already started and you do not have to start it.

5.2 SQL Server Database Connection

1. Go to **Inventory & POS** folder
2. Open **DEPOTManagementAndPOS**
3. Open **DepotSetup**
4. Then open **DepotSetup**
5. Open **Express**
6. Open **DVD-5**
7. Open **DiskImages**
8. Open **DISK1**
9. Open **program files**
10. Open **Depot**
11. Open **My Product Name**
12. Then open **DEPOTManagementAndPOS.exe.config** file with notepad++
13. Replace the **Data Source** portion with **SQL Server Name**   highlighted below



```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
    <startup>
        <supportedRuntime version="v4.0" sku=".NETFramework,Version=v4.5" />
    </startup>

  <connectionStrings>
    <add name="DepotDB" connectionString="Data Source=DESKTOP-UHS3OL6;Initial Catalog=DepotDB;Integrated Security=True"/>
  </connectionStrings>


</configuration>
```

Figure 5.21 SQL server database connection

14. Save it

5.3 Software Installation

1. Go to **Inventory & POS** folder
2. Open **DEPOTManagementAndPOS**
3. Open **DepotSetup**
4. Then open **DepotSetup**
5. Open **Express**
6. Open **DVD-5**
7. Open **DiskImages**
8. Open **DISK1**
9. Click **setup.exe** file
10. After setup a shortcut will create on desktop named **DepotShop**
11. Click **DepotShop** to open the software

CAHPTER 6

USER MANUAL

6.1 Login Panel

- This is the opening page of the software
- Manager have to login to inter the software

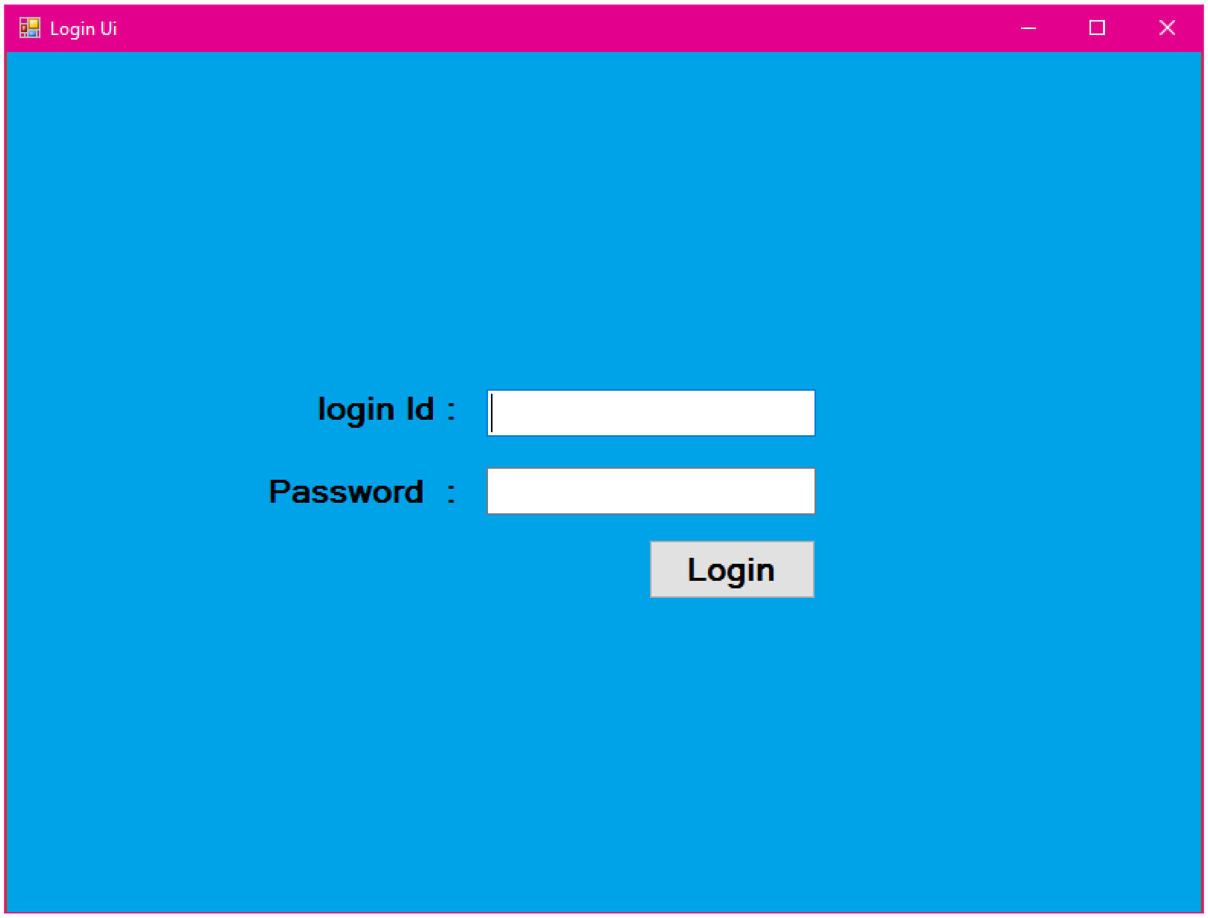
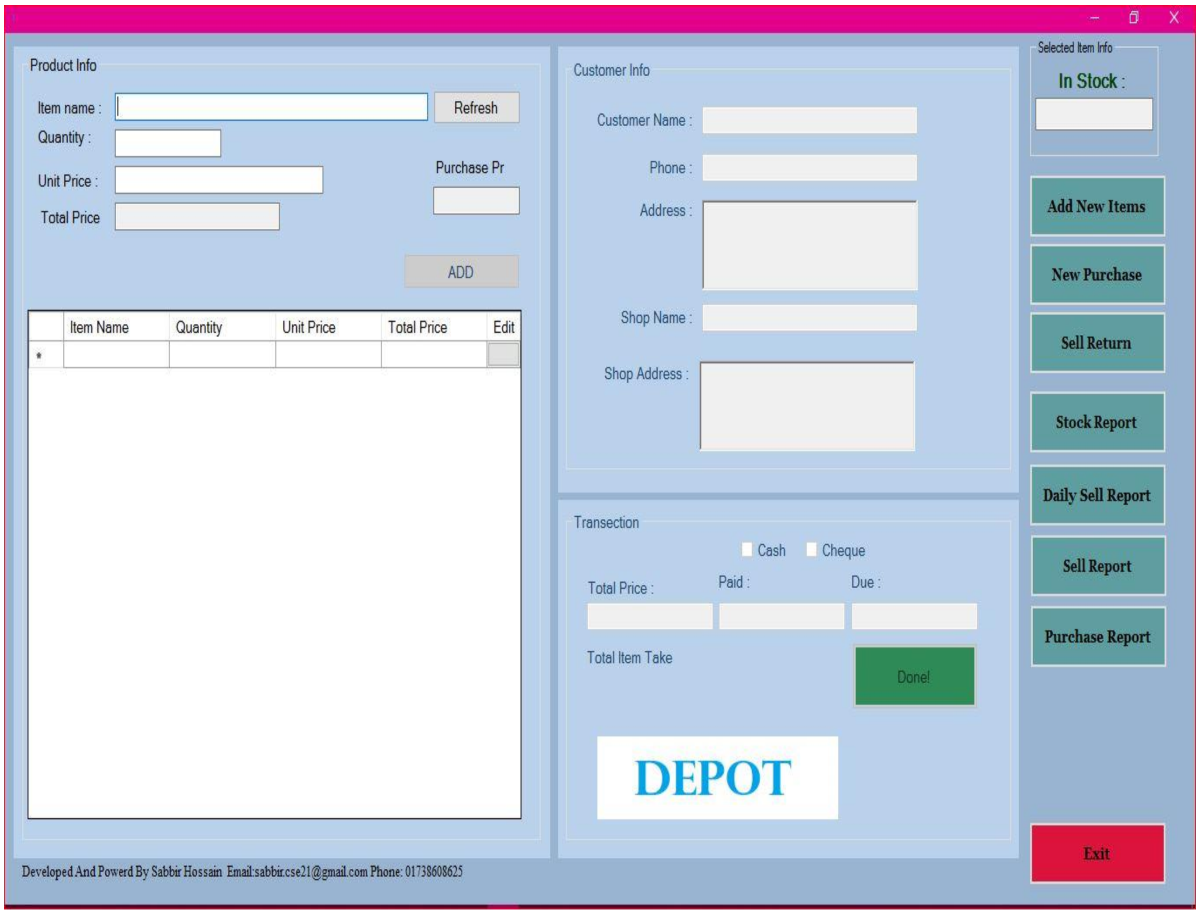


Figure 6.1. Admin Panel

6.2 Sell Product Entry

- Here user can input product info, customer info, shop name and address
- Transaction method should be selected
- Total price will update automatically
- Item list table will update automatically by adding multiple products
- Stock report will show for every product
- User can buy product and add to software by clicking 'New Purchase' button
- By clicking 'Add New Items' user can add a new product
- User can check stock report , Daily Sell Report, and have sell return option


Product Info

Item name : Refresh

Quantity :

Unit Price : Purchase Pr

Total Price

ADD

	Item Name	Quantity	Unit Price	Total Price	Edit
*					

Customer Info

Customer Name :

Phone :

Address :

Shop Name :

Shop Address :

Transection

Cash Cheque

Total Price : Paid : Due :

Total Item Take

Done!

Selected Item Info

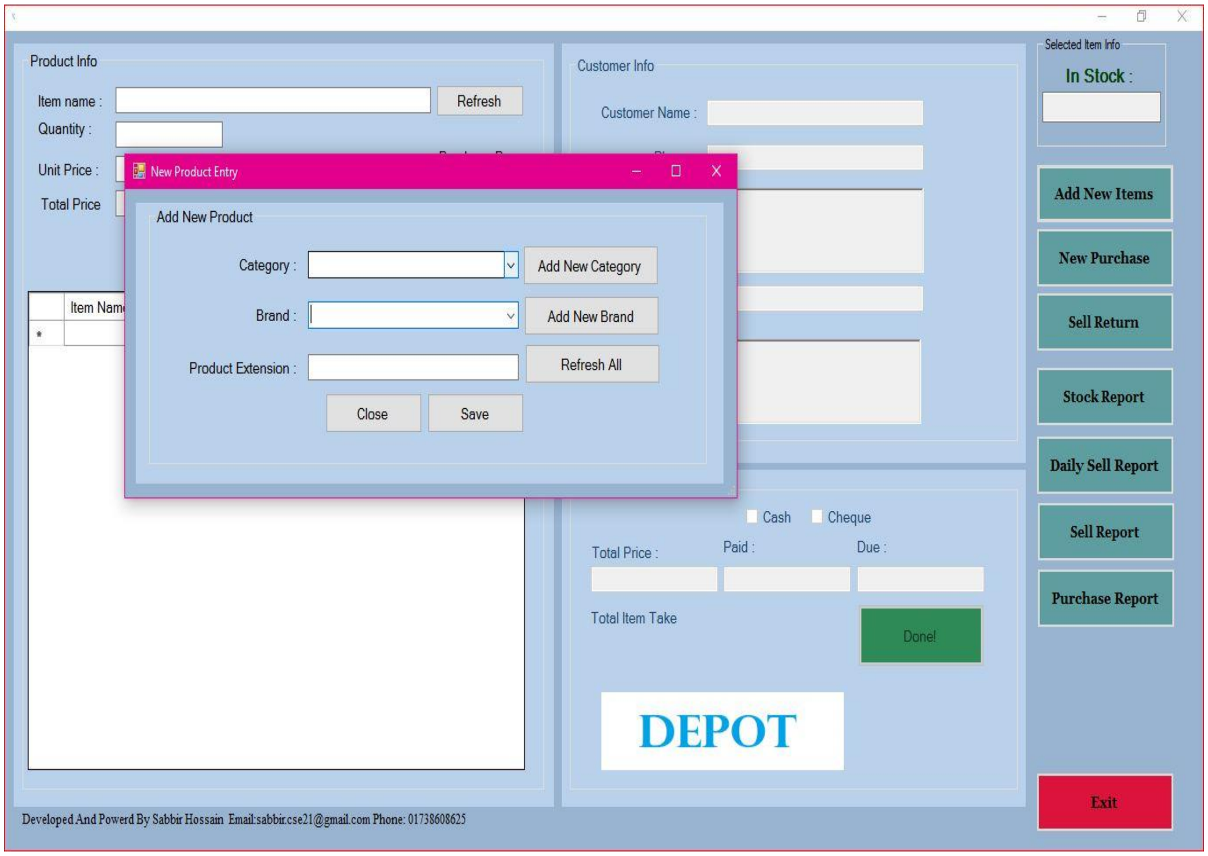
In Stock :

Developed And Powerd By Sabbir Hossain Email:sabbir.cse21@gmail.com Phone: 01738608625

Figure 6.2. Sell Product Entry

6.3 Add New Items

- By clicking ‘Add New Items’ button new window will appear
- In New Product Entry window user can select category and brand by dropdown
- User can add product name by add ‘Product Extension’ and save the new product
- User can add new category and brand by clicking ‘Add New Category’ and ‘Add New Brand’
- User can add new category and use existing brand and vise-versa



Developed And Powerd By Sabbir Hossain Email:sabbir.cse21@gmail.com Phone: 01738608625

Figure 6.3. New Product Entry

6.3.1 Add New Category

- By clicking ‘Add New Category’ button Add New Category window will appear
- Input New Category Name and save it

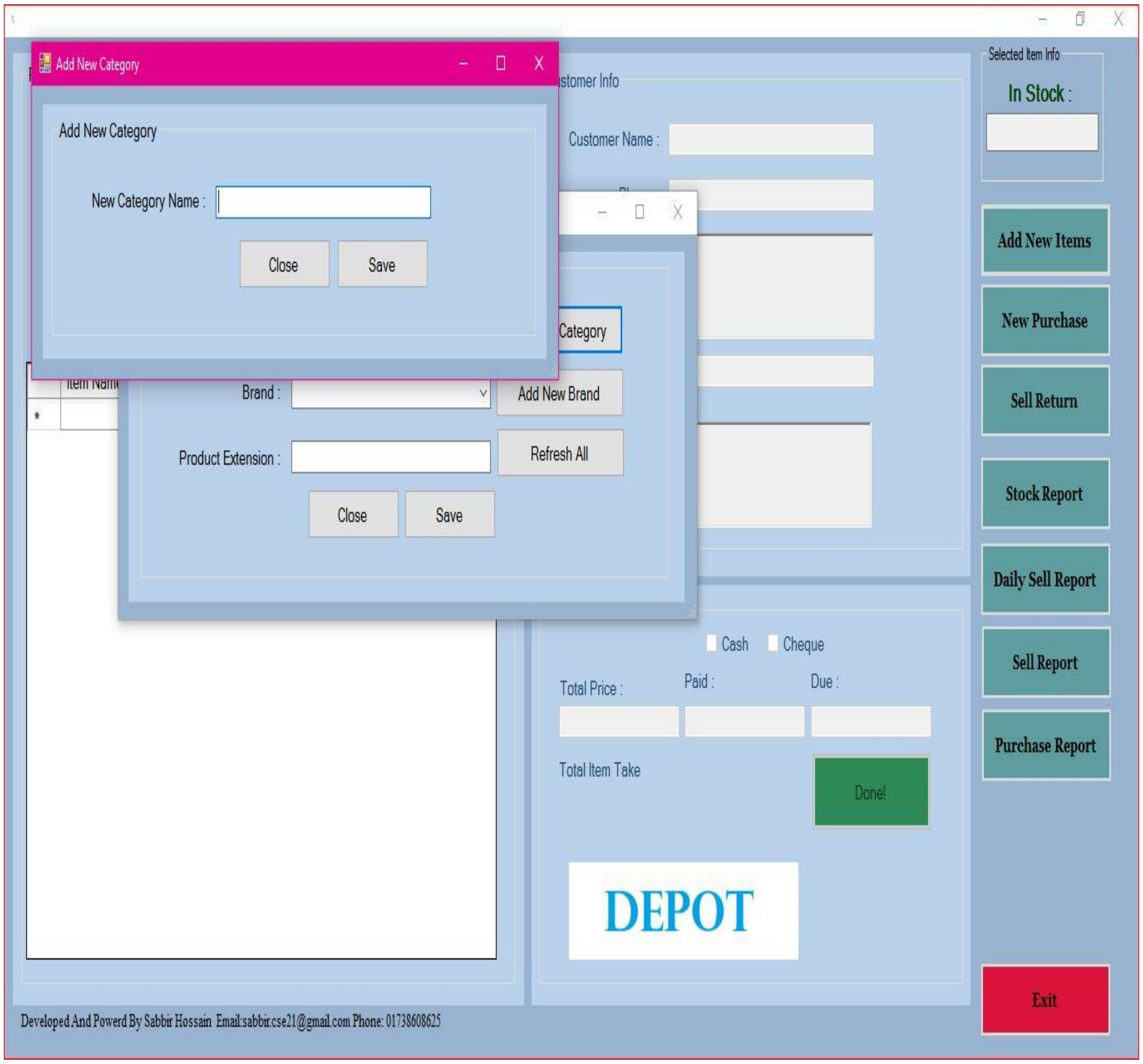


Figure 6.4. Add New Category

6.3.2 Add New Brand

- By clicking 'Add New Brand' button Create New Brand window will appear
- Input New Brand Name and select the category from dropdown and save it

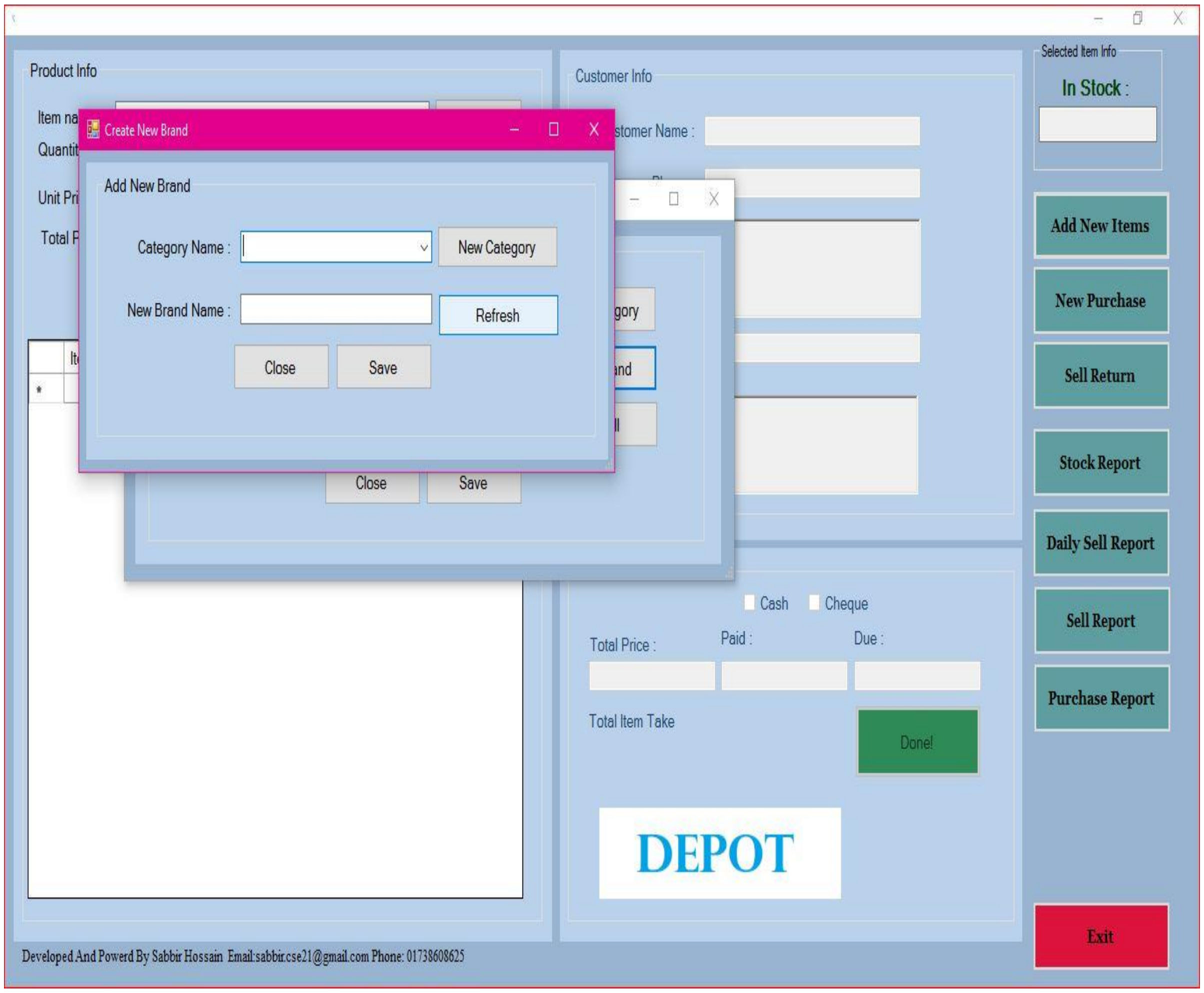


Figure 6.5 Add New Brand

6.4 New Purchase

- User can update new product info by clicking this button
- He has to input software name, unit price, and quantity of product
- Total price will update automatically
- User has to select the date
- User can view Depot (stock product)

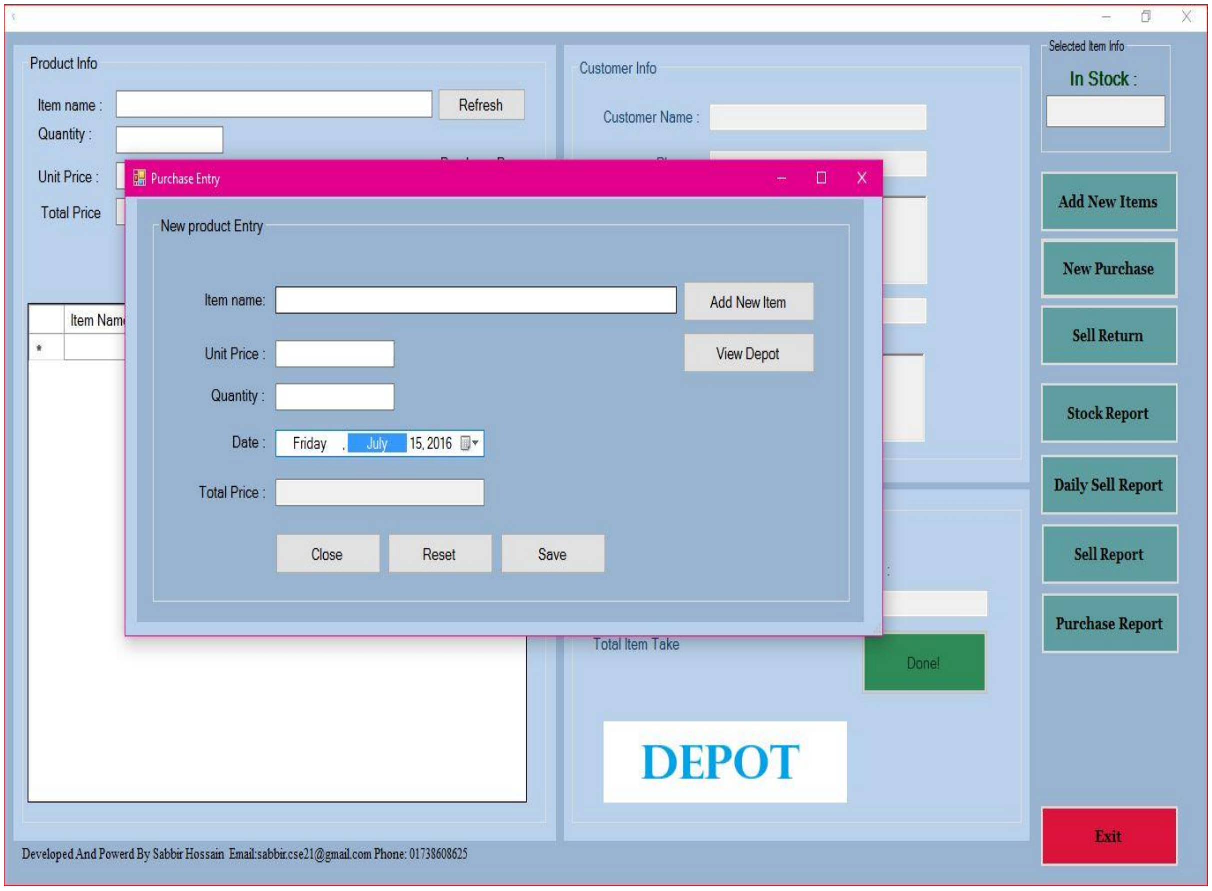
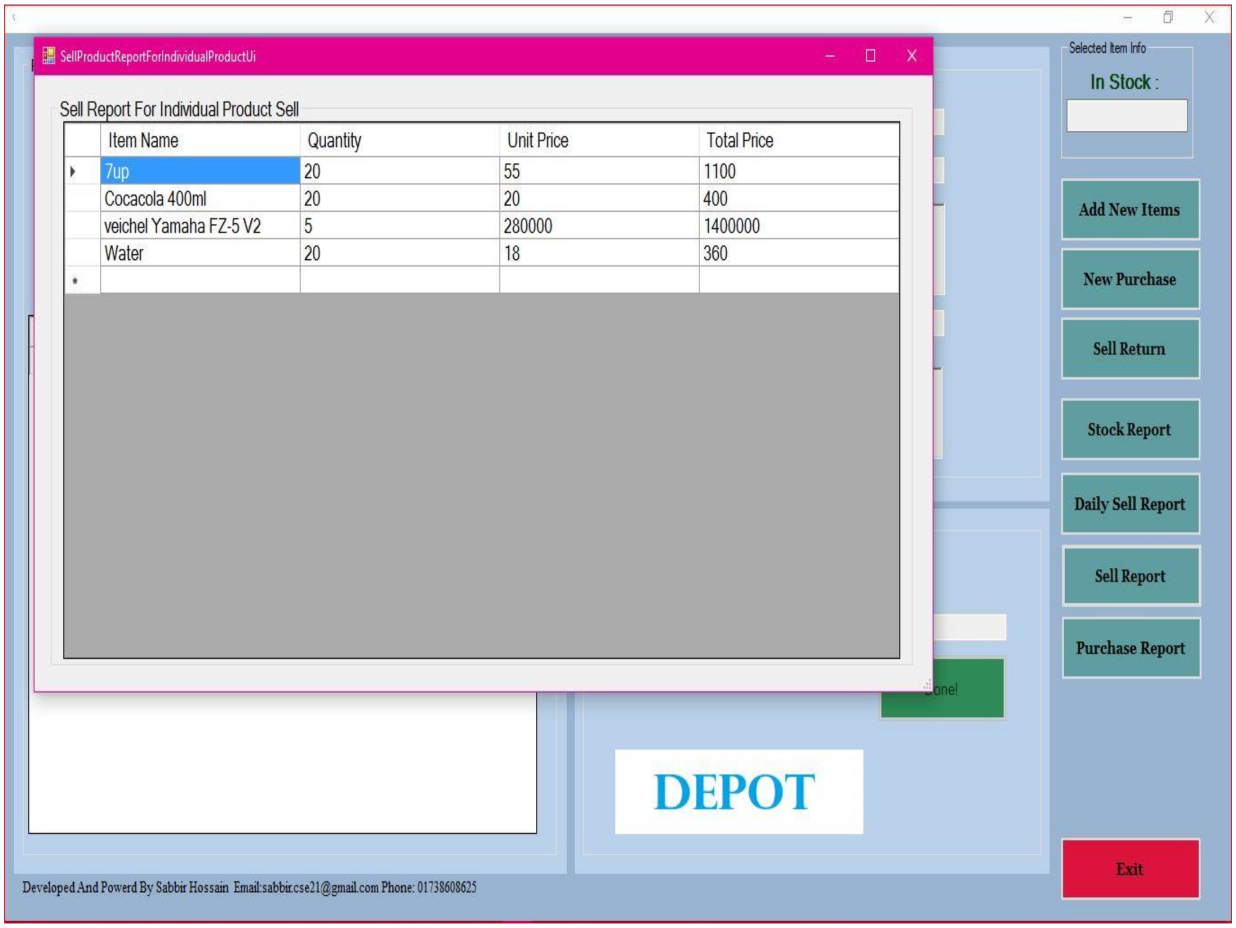


Figure 6.6 Purchase Entry

6.5 Stock Report

- By clicking ‘Stock Report’ button Stock Report window will appear
- Here user can see the product total sell, total stock, and last entry of this product by entering ProductID or Name.
- User can add new item clicking ‘Add New Item’ and also delete an item by clicking ‘Delete Item’
- By ‘View Depot’ button user can see the full stock report of the depot (warehouse / shop)


SellProductReportForIndividualProductUi

Sell Report For Individual Product Sell

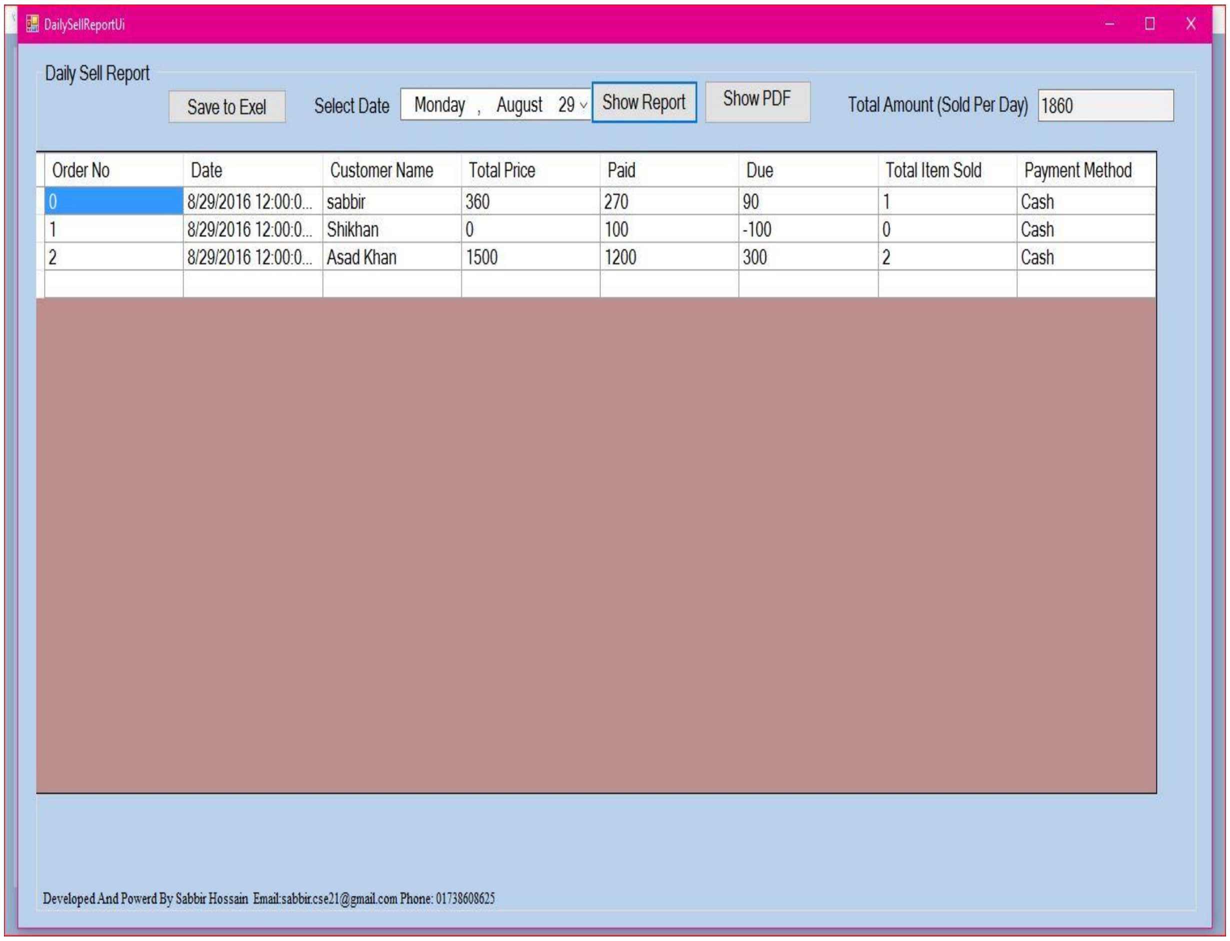
Item Name	Quantity	Unit Price	Total Price
7up	20	55	1100
Cocacola 400ml	20	20	400
veichel Yamaha FZ-5 V2	5	280000	1400000
Water	20	18	360

Developed And Powerd By Sabbir Hossain Email:sabbir.cse21@gmail.com Phone: 01738608625

Figure 6.7 Stock Report

6.6 Daily Sell Report

- By daily sell report option user can see a specific days sell from the database
- User should select the date from the top of the window
- User can show daily sell report on this table or show in PDF format
- Daily total sell will show in the right corner
- Daily sell report window will show all the details of that day like Figure 6.7



Total Amount (Sold Per Day) 1860

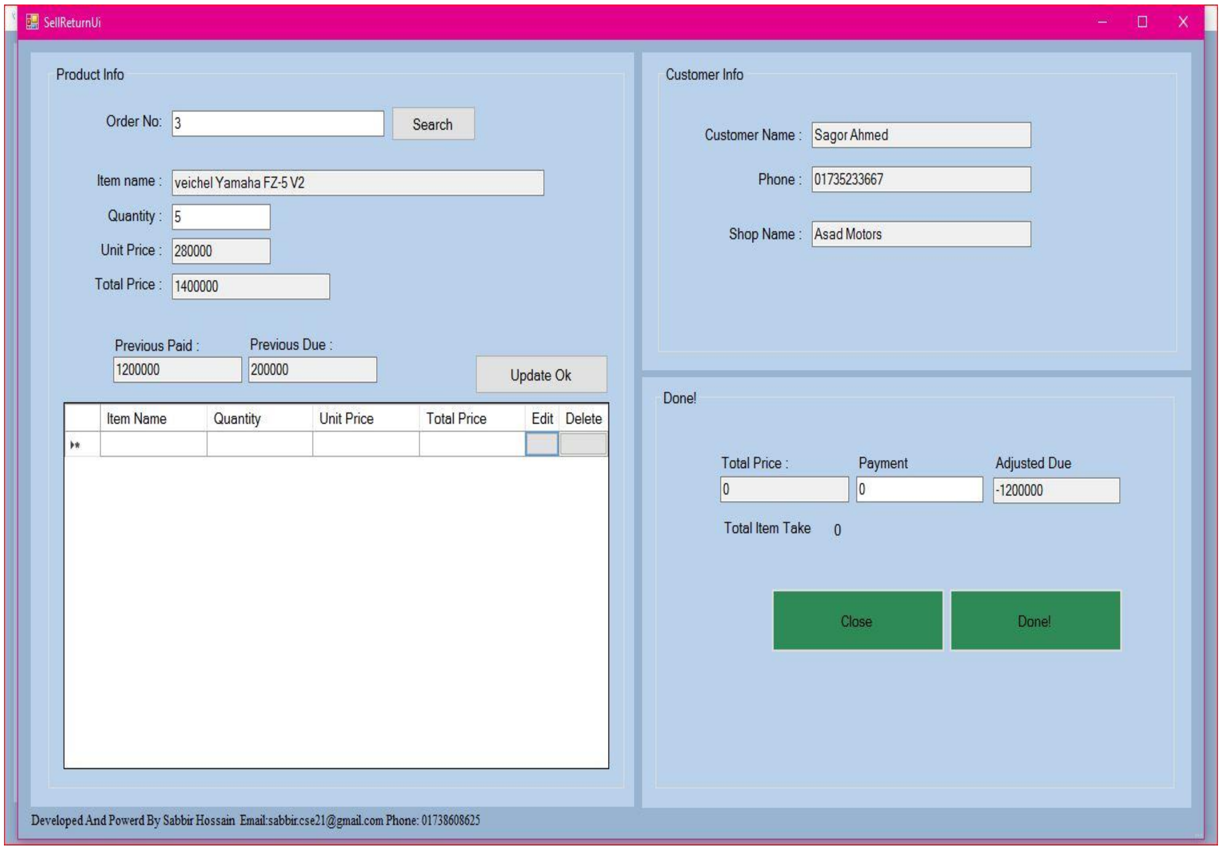
Order No	Date	Customer Name	Total Price	Paid	Due	Total Item Sold	Payment Method
0	8/29/2016 12:00:0...	sabbir	360	270	90	1	Cash
1	8/29/2016 12:00:0...	Shikhan	0	100	-100	0	Cash
2	8/29/2016 12:00:0...	Asad Khan	1500	1200	300	2	Cash

Developed And Powerd By Sabbir Hossain Email:sabbir.cse21@gmail.com Phone: 01738608625

Figure 6.8 Daily Sell Report

6.7 Sell Return

- Customer have great opportunity to return the sell by Sell Return option
- User can easily return the sell by our software
- User have to search the order no in the Order No field then all the details of that order will be shown
- Here user can update, edit or delete the order
- Software will auto update the payment
- See sell return graphical interface on Figure 6.8



Product Info

Order No: 3
Search

Item name : veichel Yamaha FZ-5 V2
Quantity : 5
Unit Price : 280000
Total Price : 1400000

Previous Paid : 1200000
Previous Due : 200000
Update Ok

	Item Name	Quantity	Unit Price	Total Price	Edit	Delete
▸*						

Customer Info

Customer Name : Sagor Ahmed
Phone : 01735233667
Shop Name : Asad Motors

Done!

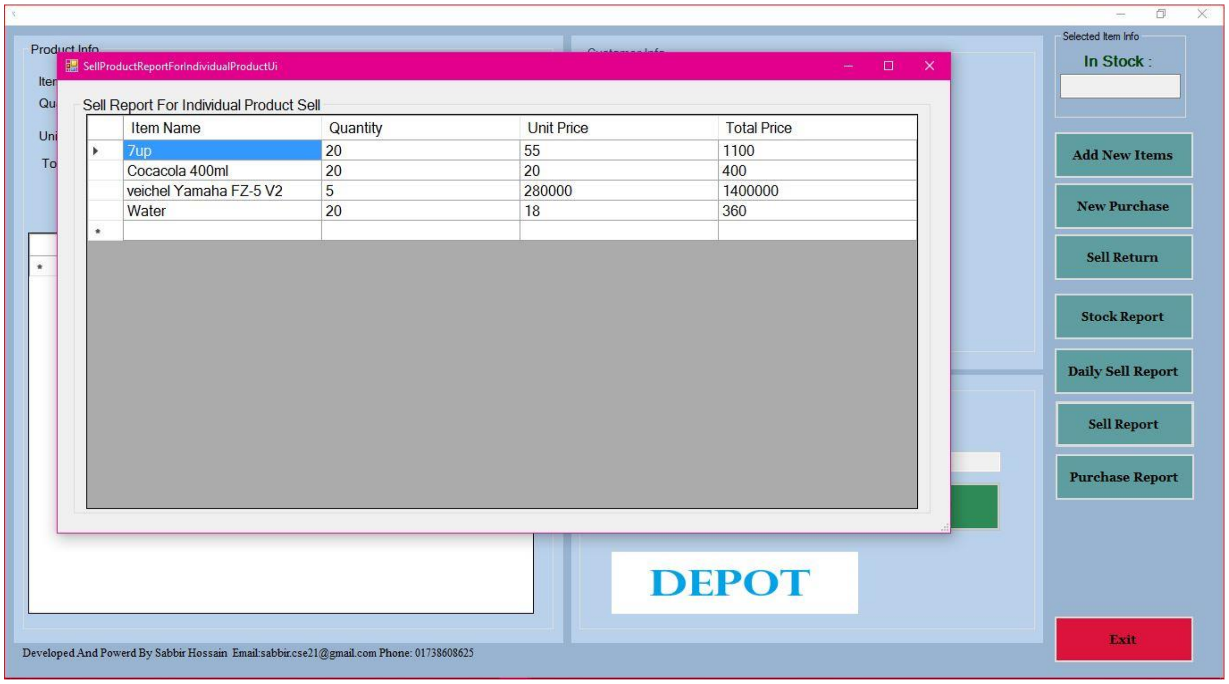
Total Price : 0
Payment 0
Adjusted Due -1200000
Total Item Take 0

Developed And Powerd By Sabbir Hossain Email:sabbir.cse21@gmail.com Phone: 01738608625

Figure 6.9. Sell Return

6.8 Sell Report

- User can see total sell report here



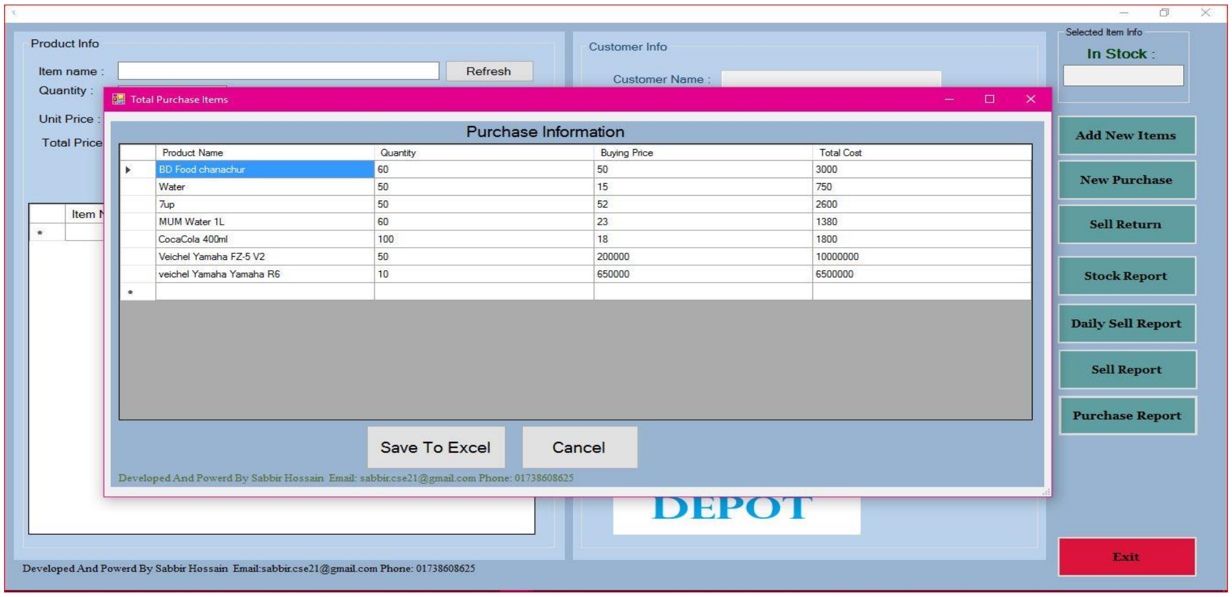
Item Name	Quantity	Unit Price	Total Price
7up	20	55	1100
Cocacola 400ml	20	20	400
veichel Yamaha FZ-5 V2	5	280000	1400000
Water	20	18	360

Developed And Powerd By Sabbir Hossain Email:sabbir.cse21@gmail.com Phone: 01738608625

Figure 6.10 Sell Report

6.9 Purchase Report

- User can see total purchase report here



Purchase Information

Product Name	Quantity	Buying Price	Total Cost
BD Food chanachur	60	50	3000
Water	50	15	750
7up	50	52	2600
MUM Water 1L	60	23	1380
CocaCola 400ml	100	18	1800
Veichel Yamaha FZ-5 V2	50	200000	10000000
veichel Yamaha Yamaha R6	10	650000	6500000

Developed And Powerd By Sabbir Hossain Email: sabbir.cse21@gmail.com Phone: 01738608625

Developed And Powerd By Sabbir Hossain Email:sabbir.cse21@gmail.com Phone: 01738608625

Figure 6.11 Purchase Report

CHAPTER 7

CONCLUSION AND FUTURE WORK

7.1 Limitations

For future enhancement there are few suggestion to improve the system abilities-

- There is no barcode reader, so user have to input the product manually
- No Direct Invoice Printing Option

7.2 Future Plans

- Include barcode reader
- Include direct invoice printing option

7.3 Conclusion

This Desktop Application provide facilities to the store manager and the customers. It saves time as it allow auto calculation of multiple purchase and store in the database for future concern. Administrator has a privilege to create, modify, and delete the particular product info. Customer can return the purchase and can take invoice from the shop. In this computerized system, it can observe that the product purchase information and calculation obtained with ease and accuracy. The user with minimum knowledge with computer can be able to operate the software easily.

References:

[1] “Point of Sale” Available: (Viewed July 12 2016) https://en.wikipedia.org/wiki/Point_of_sale

[2] Thomas Bronack 2012, “Inventory Management System”, Viewed July 12 2016, http://dcag.com/images/INVENT01.pdf

[3] “Software Development Life Cycle” Available: (Viewed July 13 2016) http://www.tutorialspoint.com/sdlc/sdlc_overview.htm

[4] “Software Development Models” Available: (Viewed July 13 2016) http://istqbexamcertification.com/what-are-the-software-development-models/

[5] “Agile Model” Available: (Viewed July 14 2016) http://www.tutorialspoint.com/sdlc/sdlc_agile_model.htm

[6]”Agile model Advantage, Disadvantage” Available: (Viewed July 14 2016) http://istqbexamcertification.com/what-is-agile-model-advantages-disadvantages-and-when-to-use-it/

[7] “Agile Mythologies” Available: (Viewed July 14 2016) https://www.versionone.com/agile-101/agile-methodologies/

[8] Karl Wiegers and Joy Beatty, “Software Requirements” Edition-3

[9] “What is C# Language” Available: (Viewed July 15 2016) https://msdn.microsoft.com/en-us/library/z1zx9t92.aspx

[10] “Why use C#” Available: (Viewed July 15 2016) https://msdn.microsoft.com/en-us/library/aa664274(v=vs.71).aspx

[11] “C# WinForms” Available: (Viewed July 15 2016) http://zetcode.com/gui/csharpwinforms/introduction/

[12] “Why use WinForms” Available: (Viewed July 15 2016) http://stackoverflow.com/questions/6564795/wpf-vs-winforms-for-net-newbies

[13] “What is SQL server, Why SQL server” Available: (Viewed July 16 2016) http://www.databasejournal.com/features/mssql/article.php/3769211/What-is-SQL-Server.htm

[14] “Three Layer Architecture” Available: (Viewed July 16 2016) http://www.codeproject.com/Articles/36847/Three-Layer-Architecture-in-C-NET