

Dilan Wickramarathna

Supervisor: Mr. Mahen Jayalath

March 2014

**Analytical, Intelligent, Business and Inventory Automation Systems
(AIBIAS)**

A dissertation submitted in partial fulfilment of the University of Greenwich's

BSc (Hons) in Computer Science

Abstract

The target of the **Analytical, Intelligent, Business and Inventory Automation Systems (AIBIAS)** project are Visual Business Data Analyzing Dashboard, Usability of the voice command (self-assistant), re usability and multipurpose, mobility, safety, relevant, automotive intelligent sales and inventory management system. With all of these, the project was mainly directed to the internal and mobile Visual Data Analyzing and Automated dissection making based on the analyzed internal sales and product data, with avoiding the traditional development architectures, tools techniques.

By avoiding the traditional inventory control system, the proposed application is containing with Android and Windows Mobile application with windows application that doing the meager part of the mother application.

The proposed application was divided to three main sections such as Mother Application (Windows Application), Android Mobile Application, and Windows Mobile application. Mother Application is the main part of the application that controlling all the sales, stock and all the purchasing part, also including the analyzing part assigned to management side of the business aria as the Dashboard while The Point of sales Application was implemented to the Android mobile device platform and Windows mobile application is implemented to derive analytical decisions by using managerial reports.

Technologies such as cross platform technology, cloud based mobile development API, cloud computing and web services were used to implement this system. Mainly Visual Studio was used with C# and new developer tools such as Xamarin, Devexpress, and Telerik for the implementation.

Acknowledgements

I am consuming this opportunity to express my gratefulness to every person who help and guided me during the course of this BSc in Computer Science Final project. I am grateful for their ambitious supervision, invaluable helpful condemnation and approachable advice through the project effort. I am honestly thankful to them for sharing their honest and revealing opinions on an amount of productions interrelated to the project.

I express my warm thanks to Mr. Nandana Pathirage , Mr. Mahen Jayalath, Mr. Kaneeka Indupa and Mr. Pradeep Eranga for their provision and supervision at Victoria Hire education Campus and all the people who provided me with the facilities being obligatory and conductive conditions for my BSc in Computer Science Final project.

Table of Contents

1.0	Introduction and Background	14
1.1	Overview to the Project	14
1.1.1	Presenting Problem and its Origin	15
1.2	Significance of the Project	18
1.3	Project Aim and Objectives	19
1.4	Scope of the project.....	19
1.5	Constraints of the Project	20
1.6	Ethical Overview	20
1.6	Summary	21
2.0	Literature Review	22
2.1	Why Inventory Management is important?	22
2.2	Automated Inventory Management System	22
2.3	Desktop Application	23
3.4	Streamlined Functionality	23
2.5	Virus Protection	23
2.6	Accessibility.....	23
2.7	Client Request.....	23
2.8	Mobile application for inventory management.....	23
2.9	Similar Systems	24
2.10	Why Analytical, Intelligent, Business and Inventory Automation Systems (AIBIAS) is better than other similar systems?	25
2.11	Conclusion.....	25
3.0	System Analysis.....	26
3.1	Analyzing the Existing System	26
3.2	Data collection Protocol and Requirement Analysis Requirement	27
2.2.1	Data Collection Plan	27
2.2.2	Implementation of Research Technique: Questionnaires	29
2.2.3	Implementation of Research Technique: Interviews	36
3.3	Requirement specification of new system.....	37
3.3.2	Functional Requirement	37
3.3.3	Non- Functional Requirements.....	41
3.3.4	Technical Requirement	42

3.3.5 UML modules for the new system	43
3.4 Summary	45
4.0 System Design	46
4.1 Overall System Architecture	47
4.1.1.1 Owner.....	48
4.1.1.2 Employee with Windows Application	49
4.1.1.2 Employee Android Point of Sale application.....	50
4.2 Software Architecture.....	51
4.2.1 Overall Windows Application.....	51
4.2.2 Module Architecture.....	54
4.2.3 Overall Mobile Application	67
4.3 Data Design	68
4.3.1 Entity Relationship Diagram.....	68
4.3.2 Mapping of Logical Database to Relations.....	69
4.4 Interface Design of Windows Application.....	70
4.4.1 Application Menu.....	71
4.4.1 Login to the system.....	72
4.4.2 Point of Sales.....	72
4.4.3 Purchase Order	73
4.4.4 Goods Reserving Notes	73
4.4.5 Controles – Item Controle.....	74
4.4.5 Controles – Suplier Controle	74
4.4.6 Controles - Customer Controle	75
4.4.7 Item Master.....	75
4.4.8 Stock Transfer	76
4.4.9 Reports - Stock	76
4.4.10 Reports - Sales.....	77
4.4.11 Reports - Expiree.....	77
4.4.12 Reports – Reorder	78
4.4.13 Dashboard	78
4.4.14 Settings – User	79
4.4.15 Settings – User Level	79
4.4.16 Settings – Mobile Access.....	80

4.4.17 Exit conformation.....	80
4.5 Interface Design of Android Application.....	81
4.5.1 Login.....	81
4.5.2 Point of sales	81
4.6 Interface Design of Windows Mobile Application	82
4.6.1 Start Screen and Dashboard	82
4.6.2 Voice Recognition	84
4.6.3 Reorder report	86
4.6.4 Push Notification.....	88
4.7 Summary	89
5.0 Development.....	90
5.1 Coding Language and Database Selection	90
5.1.1 Selection of Coding Language	90
5.1.2 Database Selection.....	91
5.2 Technologies and Tools for Design and Implementation	92
5.2.1 Development Tools.....	92
5.3.2 Design Tools	97
5.3 Order of Development.....	99
5.3.1 Database Development.....	100
5.3.2 Common Layers libraries development.....	101
5.3.3 Windows Application Development	102
5.3.4 SOAP Web Service Development.....	102
5.3.5 Mobile Application Development	102
5.4 Common Development Architectures	103
5.4.1 Menu and login	103
5.4.2 Metro Menu Style	104
5.4.3 General Form Structure of Data Handling Forms	105
5.4.4 Customizable and Real-Time Data viewers/Grid view.....	106
5.5.5 Customizable Reports	111
5.5 layered application architecture.....	114
5.5.1 The User Layers/Libraries in the application	114
5.6 Coding Structures and Techniques	124
5.6.1 Notification API for Windows and Android Mobile application	124

5.6.1 App 42 API notification API for Windows Phone	127
5.7 Stored Procedures and Views	128
5.7.1 Stored Procedures.....	128
5.7.2 Views	135
5.7.3 Triggers.....	136
5.7.4 Indexes	137
5.8 Technical and significant software engineering techniques in each modules	138
5.8.2 Windows Application	138
5.8.3 SOAP Web Service (HummingBird.WebService).....	162
5.8.4 Android Application	164
5.8.5 Windows Phone Application.....	168
5.9 Summery.....	175
6.0 Testing.....	176
6.1 System Testing Objectives	176
6.2 Testing Strategy	177
6.3 Test Plan.....	178
6.3.1 Test Plan – Unit Testing.....	179
6.3.2 Test Plan – System Testing.....	185
7.0 Implementation	188
7.1 Major tasks on system implementation	188
7.2 Data Conversion / Loading.....	188
7.3 System Changeover.....	188
7.4 User Training.....	189
7.5 Summary	189
8.0 Evaluation	190
8.1 Evaluation of Project Practice	190
8.1.1 Analysis phase	190
8.1.2 Design phase	191
8.1.3 Development and testing and implementation phase	192
8.2 Evaluation of Product.....	193
8.2.1 Methods of Evaluation.....	193
8.2.3 Evaluation Criteria.....	193
8.3 Significance of the Solutions	195

8.4 Summary	197
9.0 Conclusion.....	198
9.1 Project Results Overview	198
9.2 Overview of Practice and Lessons Learned.....	198
9.3 Path to Future Enhancements	200
9.4 Conclusion to the Project.....	200
9.5 Summary	201
References	202
Appendix A - Project Proposal	205
1.0 Introduction	205
2.0 Presenting Problem.....	207
3.0 Application Proposal	208
3.1 Application Layout.	209
3.2 Functional requirement	212
3.3 Non- Functional Requirements.....	217
3.4 Technical Requirement	219
4.0 Significance of the project	221
4.1 Stocks.....	221
4.2 Sales	221
4.3 Managerial Support	221
5.0 Aim and Objectives	224
6.0 Scope of the project.....	225
7.0 Limitation of Project	226
8.0 Research Methodology	227
9.0 Ethical Overview	228
10 Project Plan	229
11 Summary	230
References	231
Appendix B - Testing	232
Appendix B: 1	232
Appendix B: 2	234
Appendix B: 3	235
Appendix B: 4	239

Appendix B: 5	241
Appendix B: 6	243
Appendix B: 7	246
Appendix B: 8	249
Appendix B: 9	251
Appendix C - Testing	255
2.2.2.3.2 Questionnaire 2 - Windows Application Design	257
2.2.2.3.3 Questionnaire 2 - Mobile Application Design	264
2.3.3.2 Analysis of gathered data	267
Interviews Question 1 – Analyze the current process Collect Inventory Data	267
5.2 Question.....	272
5.3 Question.....	272
6.1 Question.....	273
Question.....	273

List of Figures

Figure 1 Productivity.....	14
Figure 2 Business Process	14
Figure 3 Existing Problem.....	16
Figure 4 Mobile functions	48
Figure 5 User Functions	49
Figure 6 Layer Architecture.....	51
Figure 7 Module Architecture.....	52
Figure 8 Sample Dashboard	54
Figure 9 Purchase Orders.....	55
Figure 10 Goods Reserving Notes	56
Figure 11 Item Master.....	57
Figure 12 Controls Overview.....	58
Figure 13 Controls.....	59
Figure 14 User Management	60
Figure 15 Product Management	61
Figure 16 Point of Sales.....	62
Figure 17 Report Overview	63
Figure 18 Reports.....	64
Figure 19 Reports – Sales.....	65
Figure 20 Reports - Inventory	66
Figure 21 Sample Mobile Application Dashboard Design.....	67
Figure 22 Entity Relationship Diagram.....	68
Figure 23 Mapping of Logical Database to Relations.....	69
Figure 24 Interface Menu	71
Figure 25 Login screen	72
Figure 26 Point of Sales.....	72
Figure 27 Purchase Order	73
Figure 28 Goods Reserving Notes	73
Figure 29 Controls – Item controls.....	74
Figure 30 Controls – Supplier controls.....	74
Figure 31 Controls – User controls.....	75
Figure 32 Item Master.....	75
Figure 33 Stock Transfer	76
Figure 34 Reports - Inventory	76
Figure 35 Reports - Sales.....	77
Figure 36 Reports - Expiree.....	77
Figure 37 Reports – Reorder	78
Figure 38 Dashboard.....	78
Figure 39 Settings – User	79
Figure 40 Settings – User Level.....	79
Figure 41 Settings – Mobile Access.....	80
Figure 42 Exit conformation.....	80

Figure 43 Android Application Login.....	81
Figure 44 Android Application Point of sales window	81
Figure 45 Dashboard 2	82
Figure 46 Dashboard 1	82
Figure 47 Send Feedback	83
Figure 48 Menu.....	83
Figure 49 Voice recognition 2	84
Figure 50 Voice recognition 1	84
Figure 51 Date Picker	85
Figure 52 Expires.....	85
Figure 53 Reorder report 2	86
Figure 54 Reorder report 1	86
Figure 55 Reorder report 5	87
Figure 56 Reorder report 4	87
Figure 57 Reorder report 3	87
Figure 58 Windows Phone Push Notification	88
Figure 59 Cross Platform Technology	91
Figure 60 Development Tools	92
Figure 61 Visual Studio IDE	93
Figure 62 Xamarin Studio.....	94
Figure 63 SQL Server Management Studio	96
Figure 64 dbForge Studio for SQL Server.....	97
Figure 65 Order of Development.....	99
Figure 66 Create Tables	100
Figure 67 Menu and login	103
Figure 68 Metro Navigation.....	104
Figure 69 General Form Structure	106
Figure 70 Customizable and Real-Time Data viewers/Grid view' fetures	106
Figure 71 Ascending and descending.....	107
Figure 72 Data grouping and categorize.....	107
Figure 73 Multi Categorize.....	108
Figure 74 Filtering	108
Figure 75 custom filtering 1	109
Figure 76 custom filtering 2	109
Figure 77 Search.....	110
Figure 78 Multiple Search	110
Figure 79 make the grand total.....	112
Figure 80 filter by any columns.....	112
Figure 81 Print report.....	113
Figure 82 direct Export.....	113
Figure 83 Solution Explorer.....	114
In the Application as the image (Figure 84 Solution Explorer) the application has been divided in to separate tasks by allocating specific module to one library.....	114
Figure 85 HummingBird.DataLibrary View	114

Figure 86 Database Connection Class	115
Figure 87 Database Connection string in app.config	115
Figure 88 Created common properties to use in every classes	115
Figure 89 Sample Inherited class from database connection class	116
Figure 90 Referenced Layers.....	116
Figure 91 Usage of Referenced class	116
Figure 92 Sample Insert/Update Method.....	118
Figure 93 Sample Selection Class (Array List)	119
Figure 94 Sample Selection Class (Data Table)	119
Figure 95 Sample Selection Class (Single type).....	120
Figure 96 Sample Delete Class (Single type)	120
Figure 97 Accessors in the modeler layer	121
Figure 98 implemented layered that relations with each main application.....	123
Figure 99 Google Console Create Project	124
Figure 100 Google Console Create Project	124
Figure 101 Enable API	125
Figure 102 Enable API features	125
Figure 103 App 42 API Dashboard	127
Figure 104 Select library to download	127
Figure 105 Stored Procedures.....	128
Figure 106 Stored Procedures.....	129
Figure 107 Stored Procedure Architecture	130
From the architecture as present above (Figure 108 Stored Procedure Architecture) the procedure main task handler is the @RunType parameter when develop the application if developer need to achieve specific task from the stored procedure, the parameter should be supplied with the expectation of stored procedure with the used code.	130
Figure 109 C# Data Layer Code usage with the stored procedure	131
Figure 110 Significant Insert statement that used in the stored procedures.....	132
Figure 111 Sample Stored Procedure that used in the application	134
Figure 112 Views.....	135
Figure 113 Sample View that created in the database	136
Figure 114 Used triggers in the database.....	137
Figure 115 index.....	137
Figure 116 Methods that responsible for initialize the speech	138
Figure 117 Module Navigation function	139
Figure 118 Point of Sales tile.....	140
Figure 119 Supplier Tile.....	140
Figure 120 multiple slide frame for one module	140
Figure 121 Method that responsible to make the tiles analyzable	141
Figure 122 Responsible Stored Procedure to generate the analyzable data from tiles	142
Figure 123 Full of Data analyzing Menu.....	143
Figure 124 Accessing to another form's public properties	144
Figure 125 Mask Boxes	145
Figure 126 Mask Boxes Data Bind.....	145

Figure 127 Key Press Events in Point of Sales module.....	146
Figure 128 Send Notification button click.....	147
Figure 129 the method that responsible to handle and send notification to the Android application ...	148
Figure 130 App42 References.....	149
Figure 131 Notification server to Windows Phone application.....	149
Figure 132 Tab Loading Events	150
Figure 133 Database side selection for generate the Batch Number.....	151
Figure 134 User Levels in the Settings	152
Figure 135 applying the user privileges	152
Figure 136 user privileges	153
Figure 137 Authenticated user	154
Figure 138 dashboard designer	155
Figure 139 after design the dashboard.....	156
Figure 140 One of dashboard that using in the application	157
Figure 141 analyze manually.....	158
Figure 142 movable time range and changed cub data grid and charts according to the adjusted time range	158
Figure 143 dashboard output tootle sales.....	159
Figure 144 dashboard output for today sales.....	161
Figure 145 Today sales according to categories	161
Figure 146 web service explorer.....	162
Figure 147 sample web methods that created in Android web service	163
Figure 148 Asynchronous method	173
Figure 149 event hijacking to do custom tasks.....	174
Figure 150 Test Module: Product Controls.....	180
Figure 151 Test Module: Item Master	181
Figure 152 Test Module: Product Order	181
Figure 153 Test Module: Good reserving notes.....	182
Figure 154 Test Module: Payment schedule.....	182
Figure 155 Test Module: Dashboard.....	183
Figure 156 Test Module: Mobile Push Notification	183
Figure 157 Test Module: Mobile Push Notification	184

List of Tables

Table 1 Data Collection Plan	27
Table 2 Metro Navigation Coding	105
Table 3 Test Plan – Unit Testing.....	180
Table 4 Overview of Practice and Lessons Learned.....	199

1.0 Introduction and Background

This chapter is dedicated to present an introduction to the project and an outline of the problem which would function as the foundation of the entire project. Further, this chapter endeavors to identify the crux of the problem and align the direction of the research to remedy the issues that lie in the heart of the problem.

1.1 Overview to the Project

The business community plays an important role in the continued economic development and growth of the nation. Within this business community, various business professionals, or businessmen, play a key role in carrying out their duties that contribute to the development of the business.

Owners of business have a busy schedule since they has to deal with clients, investors as well as employees who work along. It is not always possible to physically attend in each these areas. Yet his ultimate goal of running a business is to gain profit. Therefore, it is very essential to keep an eye on the **inventory, stock and sales** that directly deals with profit.

A Business always depend on profit, it also main thing of a business, When specializing the profit,



Figure 1 Productivity



Figure 2 Business Process

Those factors are mainly contribute with organization or company. By using the **Technological business solution** to motivate and manage to manage those factors, Owners of business can reduce disequilibrium of the organization or company.

As such using **Technological business solution** with the business these areas of a business facilitates an effective management of,

- Flow of materials
- Utilizing people
- Utilization of equipment
- The Co-ordination of
 - Internal activities
 - Communication with
 - Owner
 - Customers

“Oriental Pharmacy” Limited, the first branded retail pharmaceutical chain in Gampaha, Sri Lanka, entered the market with a read of making a distinction within the retail pharmaceutical trade. “Oriental Pharmacy” has introduced associate innovative idea focused around superior client care, a good product assortment, and reasonable costs. By introducing those innovative ideas, management of “Oriental Pharmacy” has intended to increase their sales volumes and create a loyal client base via improving the quality of the service. But with the extant procedure, company has facing difficulties when carrying out such a services due to the manual nature of the business functions. For an example, managing the operational data, analyzing the business trends and patterns, quick decision making, easy access to the system, flexibility etc.

Therefore, to overcome such a situation and to interconnect the all its branches as a single operational unit, “Oriental Pharmacy” need to deploy a sales and inventory management system with smart features which addresses the above mentioned factors.

1.1.1 Presenting Problem and its Origin

Currently in “Oriental Pharmacy”, their using manual inventory system mostly based on document and log books, sometime the process is caring out with manual calculations, physical counting of the inventory and stock, there inventory system is simplified, conserved and controlled without engaging with a technical system. This involves the business updates the inventory by physically tally the inventory items on a frequent basis. There manual inventory systems are time overwhelming, because the business owner is taking the daily statues via analyzing the lots of notes graphs and temporary notes with unidentified errors that maintained in whole day.

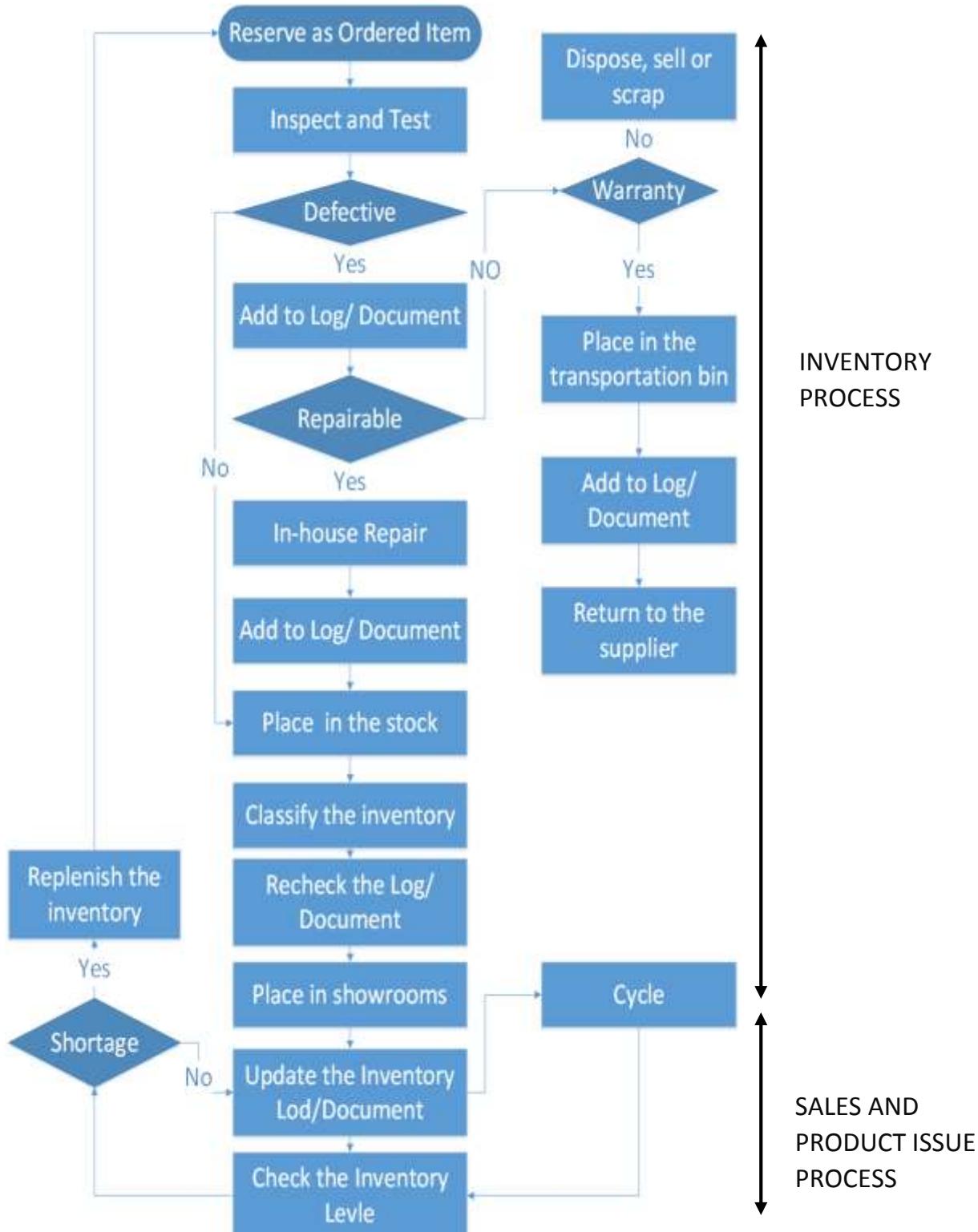


Figure 3 Existing Problem

In the current system, owner and employees are facing to the common, corresponding problems.

- Requirement of on time direct analyzing

In the current process of the subject there is no any direct accessible analyzed output of the business in that case, user or employees need to recount or calculate each time that they need to take some important information about or from the business.



Figure 4 Progress

- Every written data is not in a fast accessible format or analyzable index

In the current manual system and unplanned written reporting techniques are used, with those problems, any of calculation summary or any of financial information are not in a proper analyzable manner. According to the fact, the users of management level cannot be make proper action in immediate situation like immediate stock upgrades of stock count, financial calculation. With those situations, the employees are getting stocked with the process without proper data analyzing method. And working with lot of items in the inventory and listing down each items re calculation or exist inventory count, the process taking long time and wasting the money.

- Poor Communication

With their manual, inventory needs employee, managers to write down anytime an item is updating removing, or returning from the inventory. If one employee forgets to say that, the last low product has removed or about any changes from the inventory, a manager expects the item to offer for a client throughout a buying deal. Compared with a technical inventory system, their manual inventory system does not facilitate to communication within the geographical point. A technological inventory system updates mechanically once associate set of items has scanned out of the inventory, then all employee within the business recognize what offered in any respect times.

- Physical Counts

In the "Oriental Pharmacy" current manual inventory system doesn't give any assortment, as all numbers from the inventory are increased through physical inventory calculations. One in every of the problems of running a manual inventory system is that physical inventory counts should be performed often to regulate the things within the inventory. This is frequently wasting time and cost.

- Daily Purchases

In case, keeping track and making the report of daily purchases is another difficult foremost live with physical inventory systems. Whereas technological inventory systems scan the item and subtracts the item from the inventory, a manual inventory system needs the staff to write down the things oversubscribed throughout one workday. This will be a troublesome task, together worker might lose the list of things oversubscribed or another might forget to write down a buying deal.

- Re Ordering

With the current manual inventory, system does not update at the top of the day with updated inventory counts. This implies you need to undergo the inventory items at the needed time they wish to put order for brand spanking new materials, product or provides for the inventory. This will be a time overwhelming manner, as will bodily have to be compelled to suffer every product box and quick look thru items in the stick

- Managing the Expiries

After take the list and inventory information to document or logbook, owner getting big problem while take the expiry list of the particular day or between some time range. The process is getting hard while checking one by one in stock.

- Maintaining proper categorization

In the current system, they do not have appropriate categorization of the items, and they are practically getting lot of time and problems while searching a product to sale or other purpose, after that issue that affect to the customer satisfaction.

1.2 Significance of the Project

In Sri Lanka, few companies are fully automated. Others are semi-automated or still carrying on processing through manually. Manual processing system of stock and sales management are quite time consuming and tedious. In modern growing market, several businesses square measure, turning toward "**Analytical, Intelligent, Business and Inventory Automation Systems**" (**AIBIAS**) to perform their day-to-day tasks. The significance of the proposed system will mainly support in the areas of Stocks, Sales and Management, as per the requirement of the Pharmaceutical.

1.3 Project Aim and Objectives

Aim

The aim of this project is to develop an accurate, efficient and effective computerized Sales and Inventory Management System with smart features to Oriental Pharmacy which can overcome the problems and loose ends of the current system and provide maximum service to clients.

Objective

- To accomplish acceptable levels of customer service whereas observance of inventory costs inside realistic bounds.
- By interfusing the effective solution with new proposed application, mostly affect to the data analyzing and transparency of the business as well as the objective will be minimize the extra cost, time and man power.
- The proposed system can maximize the sales performance and minimize the time, from by achieving those targets business can make the customer satisfaction.
- Business environment need the free and clear background to run and transparency to the analyzing purposes, by providing the solution to automate the reporting and data process, business can achieve those objectives.

1.4 Scope of the project

AIBIAS will be developed to improve the efficiency of business tasks which undergo within the company environment. Since Oriental Pharmacy has a wide organizational operations such as Accounting and finance, Sales & Inventory, Administrative etc... with the time limitations for the development of project, system will be initially developed to automate the Sales & Inventory functions. This system is capable of covering the entire stock and sales handling process of the Sales & Marketing department. In addition to that, system is capable of providing necessary updates and notification to the owner of the company through windows mobile application. System is compatible with Windows based computers and a platform of .NET as well as the android application (Point of sale Application) that provided to do the sales.

1.5 Constraints of the Project

In the below it is explained about the constraints in the areas such as technical, budgetary, resources and time regarding to the development of new system

Budgetary

Since the project is self-funded, all the costs regarding to the project such as printing documents to has to be bared by the researcher itself. Therefore, there are certain limitations in the budgets. Although there are some advanced software which will provide some additional features in the development, difficulty in obtaining them due to unavailability of free versions.

Resources

Difficulties in obtaining some software resources for the development of the new system. Difficulties in meeting relevant parties for data gathering process, as they are in the top order of the hierarchy they have a busy work schedule.

Technical

Current technical knowledge of the researcher may not be adequate for the development of some modules in the new system.

Time

Difficulties in managing allocated time since the project work have to be done within given deadlines and along with the other academic.

1.6 Ethical Overview

AIBIAS has access to extremely sensitive information such as accounting information, Employee information, Customer information, Sales and Targets and etc. The system development team will identify this confidential information. And the system developers will ensure the privacy of the information by ensuring that information will be not in the hands of third party, which means the system has an obligation to maintain the confidentiality of an employee personal data.

Furthermore, this system will follow laws and regulatory framework that has standardized employee behaviors towards each other where employees are not differentiated on the basis of their race, gender, age, social status or other traits that make an individual unique. The only differentiating factor of this system will be performance. And also data will not be sold or, data will not be gathered by offering money. They will be only gathered using appropriate ways with the permission of relative parties that the data is belongs to.

1.6 Summary

The scope, depth and intricacies involved with the project and its goals and intentions are discussed in this chapter. In order to identify the intended effects of the project its target and significance has been explained. In order to guide the project to fruition, a set of objectives are discerned to act as a railing in the ascendance to its ultimate aim.

Hence, this chapter identifies the unique challenges confronted in the problem domain and Endeavour to align the direction of the project to remedy the issues that lie in the heart of the problem. The following chapter will elaborate on various facets of the research through a literature review.

2.0 Literature Review

The main aspect of doing a literature review is to find the strength and weakness of existing similar systems and researches. At the same time this helps to have wider knowledge on the activities carried by those systems and how far those activated related with the developing system. As mentioned before, the fact that this pharmacy has not gone through any automation in office activities led to analyze similar systems which perform similar tasks. Therefore literature review is a very important in such cases.

2.1 Why Inventory Management is important?

Managing inventory in a pharmacy is critical because not like other businesses sales of pharmaceutical products does not depend on the seasonal changes, market changes etc. Satisfying the needs of patients by managing inventory levels while achieving primary goals of the business is not an easy task. That why inventory management is come in to play. Inventory management should be done with careful planning and analysis. As an example ordering of the products must be depend on the inventory stock levels and sales levels. Therefore proper operational management is needed to plan those operational activities. Proper inventory management helps to lower their costs and increase their revenue. In order to minimize warehouse costs and to identify issues early, this inventory management is useful.

2.2 Automated Inventory Management System

Currently information technology plays a very big role in business. Information Technology and business are having a tight bond. In order to improve businesses IT is very much needed. By automating business process, businesses can gain competitive edge. Computer based inventory system will help to make the business operations quickly and accurately. Security issues in manual systems can also be overcome by having a computer based inventory management system. As a pharmaceutical company automation of inventory management is needed because to find out the patterns of sales and should be identified often. Through report generation feature these patters can be identified and generations of invoices will make business processes easy and faster. Some management decisions can be able to taken through the information given by the system. Such information are Sales improvements and sales pattern, identifying bestselling products, identifying loyal customers, identifying best suppliers etc. Through a system like this the information can be compared and categorized according to several parameters. Therefore through an automated system, number of employees can be reduced or employees can be allocated to some other tasks. All these benefits can be gained by automating an inventory system.

2.3 Desktop Application

Because of following reasons desktop application is better than web based system in implementing an inventory management system.

3.4 Streamlined Functionality

The standalone system operates significantly much faster than the web based system because its operations can be streamlined to the task at hand.

2.5 Virus Protection

This is not exposed to internet as the web based system does. So that it has considerably less possibility for virus infection, therefore no performance degradation will take place on the system. It does not require to use expensive anti-virus software as well.

2.6 Accessibility

User does not require internet connection to log on to the system and work on it. Therefore it is much accessible in contrast to web based system.

2.7 Client Request

Client has specifically requested for a Standalone system since they are familiar with MS (Microsoft) technologies.

2.8 Mobile application for inventory management

Mobile applications will allows the user to connect to the business processes wherever they are. It is not necessary to connect only with the mother application. Using the cross platform technology without considering the platform the device has, the application can be run. With this features the sales can be increased. As an example the business can have agents and those agents can go to the customer and sell their products as all the agents are given the access to this mobile application. Managers of company are having busy schedules. Through these mobile applications managers can log in to the system and can check the status of the business and get decision on the business wherever they are. Therefore by having a mobile application for inventory management system will improve the business processes.

2.9 Similar Systems

“TradeGecko” Software modules were used to study about how the transactions are done in a certain modules like inventory, purchase, sales, report generation and how the data manipulations are done to output particular results. “TradeGecko” is an inventory management software which stands for Comprehensive, Unified, Information System. It is software designed for growing businesses to measure performance, control costs, and increase profitability. This provides quality support where the software can be made to perform exactly as the business requires. “TradeGecko” has a wide variety of features suitable for managing businesses of many differing forms. From the face of a simple menu structure, complex business processes are easily accessible - such as inventory control, purchases, sales order control, accounting, order management and customer relationship management. This software generates intelligence stock reports and those reports as well explained summary of all transactions. This is also a cloud based software which leads the path to improvement of ecommerce.



Figure 5 TradeGecko

Furthermore in order to get an exact idea on Point of Sales (POS) functionality “Imonggo” point of sales software was reviewed. This software provides the point sales functionality to different products and POS for pharmaceutical products are reviewed. This software provides online web



Figure 6 Imonggo

based system plus the mobile system with easily understandable structure of a system. “Imonggo” offers very good understanding on the integration of sales with the stock levels and also how on the behaviors of the transactions. It also provides operational reports on sales and provides flexible way of presenting and generating sales invoices. This software can be downloaded and use as a standalone application and it provides online application as well. However standalone application is having some issues

with uninstalling.

“User Cake” software was used to study the functionalities of user management. This provides all the common and advanced functionalities of user management with security features. It also provides administrator privileges such as permission handling etc. One drawback of this software is the usage high memory capacity and taking backups is so slow.



Figure 7 User Cake

2.10 Why Analytical, Intelligent, Business and Inventory Automation Systems (AIBIAS) is better than other similar systems?

Humming bird system is better than the other similar systems because this system is implemented according to the customer need. Not like other software which are available to use for any system, this system contains special features which are unique to the “Oriental Pharmacy”.

This system is a standalone system backed by two mobile applications. Therefore security is higher than the systems that can be found in the internet. As it is standalone this will be more secured and as mobile applications are backed by the system, the system can be used anywhere they are.

As this is based on a cloud server, therefore it is more flexible, flexible and stable than that of in house IT staff.

2.11 Conclusion

Inventory management is an important factor for the success of a business. The need for an automated system for a pharmacy is vital because frequent changing demand. Therefore supply also must change accordingly with the demand for drugs in a pharmaceutical company. Successful inventory system will also help to gain competitive advantage over the other competitors by increasing sales and decreasing cost of sales. Mobile application supported desktop application will be the best solution for an automated inventory management system.

Most of the existing system are not customer based and they are not even freely available. Therefore having an inventory system with the customized features relevant for a given business will help to give the maximum advantage.

3.0 System Analysis

This document is planned as above:

- Description of the current system
- Data collection protocol and requirement analysis requirement
- Requirement specification of new system

3.1 Analyzing the Existing System

Currently In the “Oriental Pharmacy”, their using Manual inventory system mostly based on Document and log books, sometime the process is caring out with manual calculations, physical counting of the inventory and stock, there inventory system is simplified, conserved and controlled without engaging with a technical system. This involves the business updates the inventory by physically tally the inventory items on a frequent basis. There Manual inventory systems are time overwhelming, because the business owner is taking the daily statues via analyzing the lots of notes graphs and temporary notes with unidentified errors that maintained in whole day.

In the current system, owner and employees are facing to the common, corresponding problems.

3.2 Data collection Protocol and Requirement Analysis Requirement

2.2.1 Data Collection Plan

Topic	Stake Holder	Method	Applicable Date	Steps to Accomplish	Objective	Date & Time
			Variables			
Checking employee's IT knowledge	Owner	Questionnaire	Using full touch mobile device	Provide a printed question paper	Ability to using smart phone, device or tablet pc	2014/5/3
	Employee		Using Windows Application	Provide a printed question paper	Ability to use windows app, and metro based system with windows 8	2014/5/3
Analyze the current process Collect Inventory Data	Employee	Group Discussion	Collect information about current process of the inventory system	Ask Question according to the prepared interview Questionnaire	Process flow of the current inventory manage process including, Sales process Stock transfer and billing process.	2014/5/3
	Manager		Item Details, Dealers and all the resources, raw data that using to control inventory	Interview	Basic information about, Items, Prices, Unit details, Categories, Expires	2014/5/4
	Employee		Item Details, Dealers and all the resources, raw data that using to control inventory	Interview as Group		2014/5/4

Table 1 Data Collection Plan

Windows Application Design	Selected Employee from It Knowledge check and including owner	Questionnaire	Collect data to Design and implement the Application style, visible architecture, as user friendly.	Provide a printed question paper	Create user friendly application, as commonly friendly with employee	
Mobile Application Design	All Employee including owner	Questionnaire	Collect data to Design and implement the Mobile Application style, visible architecture, as user friendly.	Provide a printed question paper	Create user friendly Mobile application, as commonly friendly with employee	2014/5/5

2.2.2 Implementation of Research Technique: Questionnaires

2.2.2.1 Actual Plan

Topic	Stake Holder	Method	Applicable Date Variables	Steps to Accomplish	Objective	Date & Time
Checking employee's IT knowledge	Owner	Questionnaire	Using full touch mobile device	Provide a printed question paper	Ability to using smart phone, device or tablet pc	2014/5/10
	Employee		Using Windows Application	Provide a printed question paper	Ability to use windows app, and metro based system with windows 8	2014/5/12
Windows Application Design	Selected Employee from It Knowledge check and including owner	Questionnaire	Collect data to Design and implement the Application style, visible architecture, as user friendly.	Provide a printed question paper	Create user friendly application, as commonly friendly with employee	2014/5/12
Mobile Application Design	All Employee including owner	Questionnaire	Collect data to Design and implement the Mobile Application style, visible architecture, as user friendly.	Provide a printed question paper	Create user friendly Mobile application, as commonly friendly with employee	2014/5/15

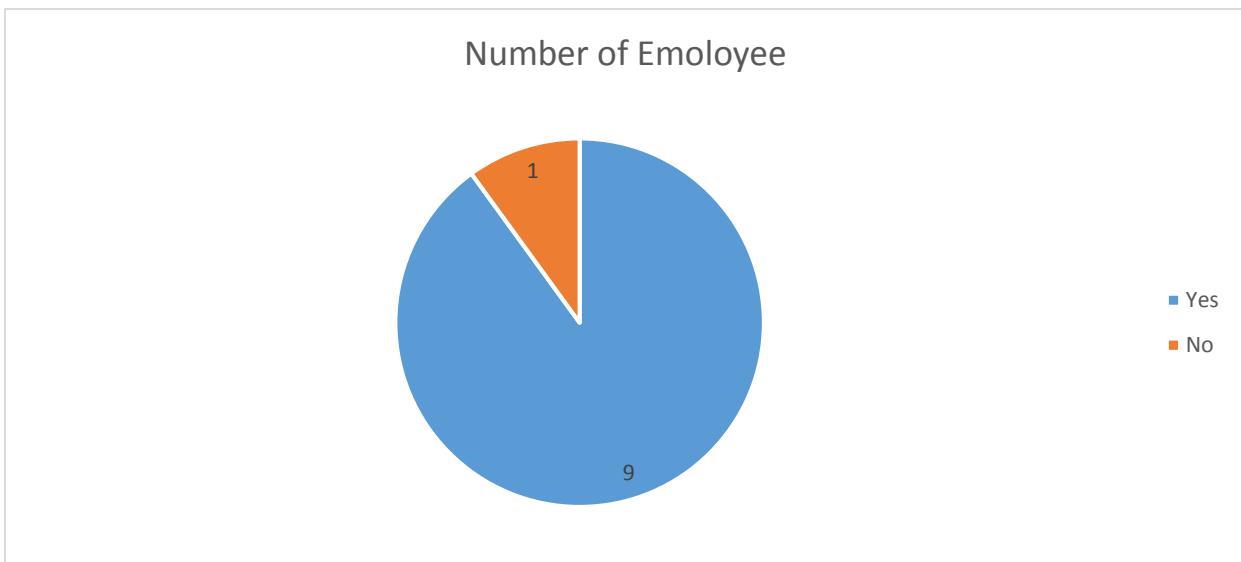
Table 2 Actual Plan

In the actual plane the questionnaire dates is changed from unexpected facts, and some employees are not attend to the questionnaire session.

2.2.2.3 Analysis of goatherd data

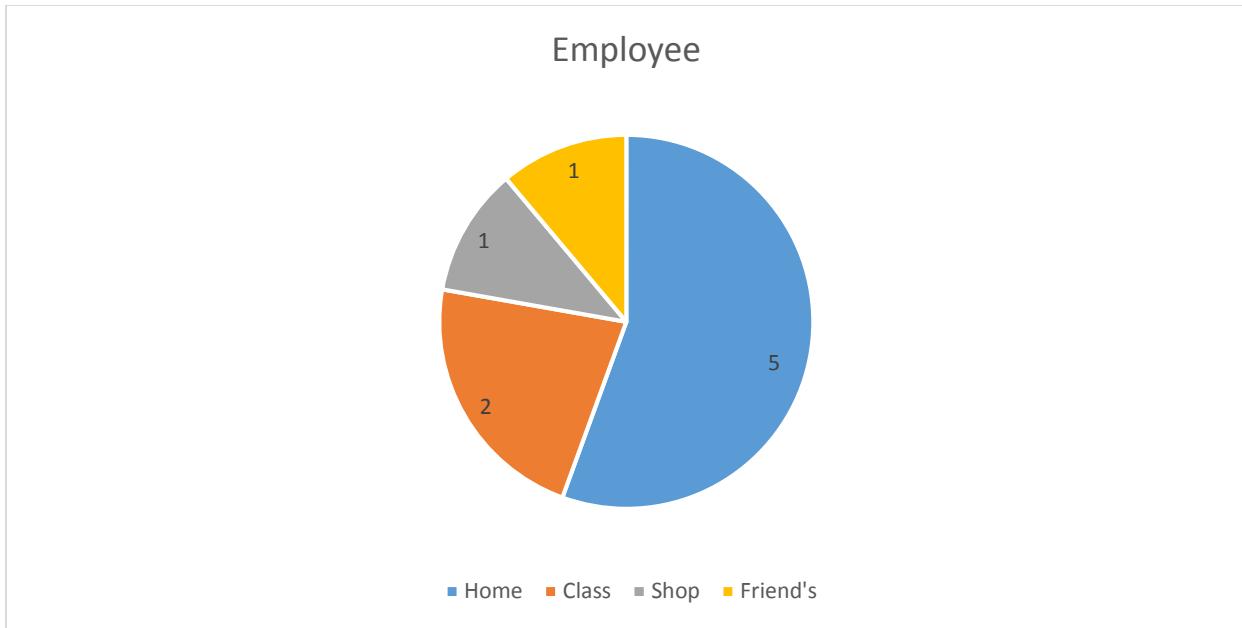
2.2.2.3.1 Questionnaire 1 - Employee - Check IT Knowledge

Question 1 - Have you ever used a computer before?



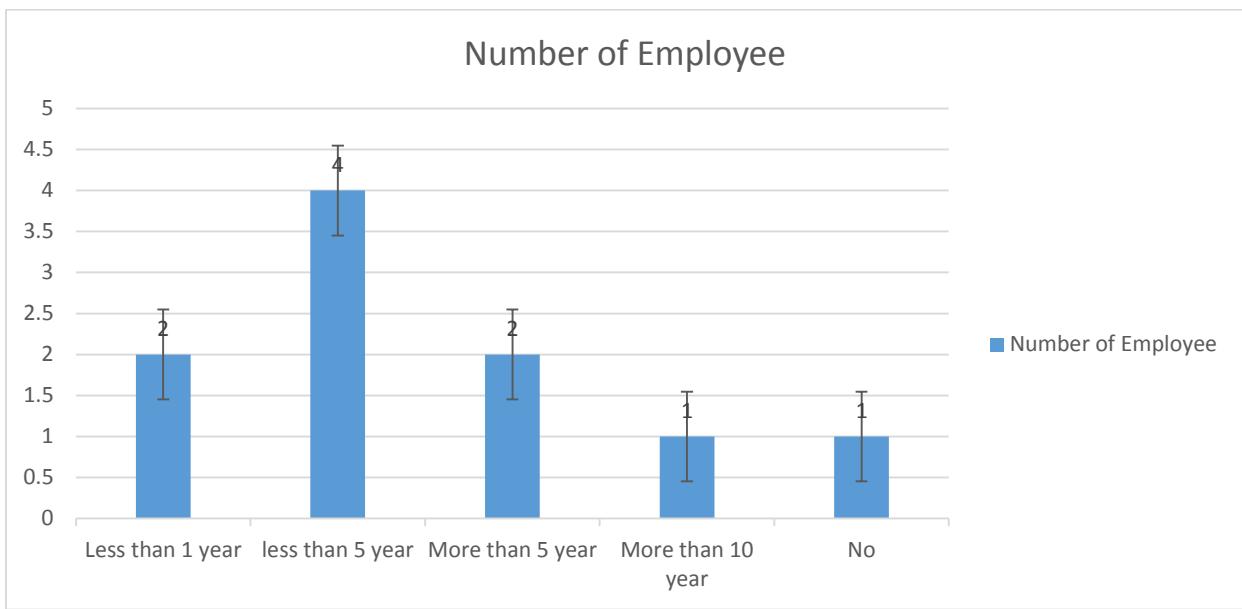
From 10 employees in the Pharmacy, his question is used to; take an idea about employees who can use the computer. As the result, the one employees is not used a computer before and others able to work with computer.

Question 1.1 - If "Yes" then where have you used it?



The question is used to; take an idea about the places that they are using the computer before, 9 employees are faced to the question without the not used employee.

Question 2 - How long have you been used a computer?

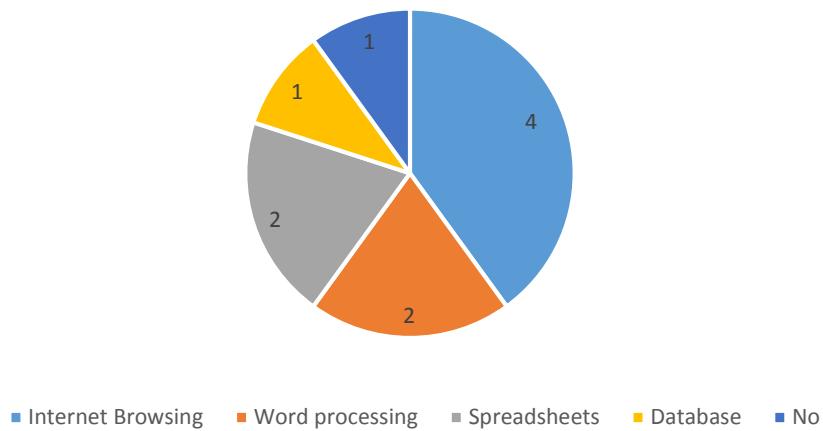


By the question, collecting the data about experience of using computer

Four of employee using computer less than 5 years and three of them are using the computer more than 5 year. As the decision, less than one-year employee need special training to use the computer with application.

Question 3 - For what have used the Computer?

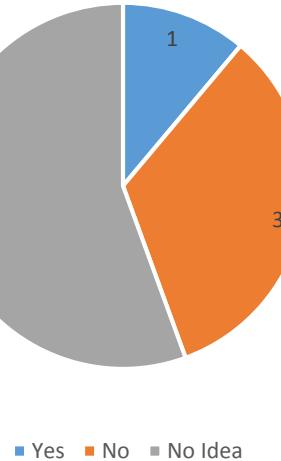
Number of Employee



Most of them are used a computer to browse the internet, and two of them can use Word-processing software and another two of them can work with spreadsheets. One of employee is aware with databases.

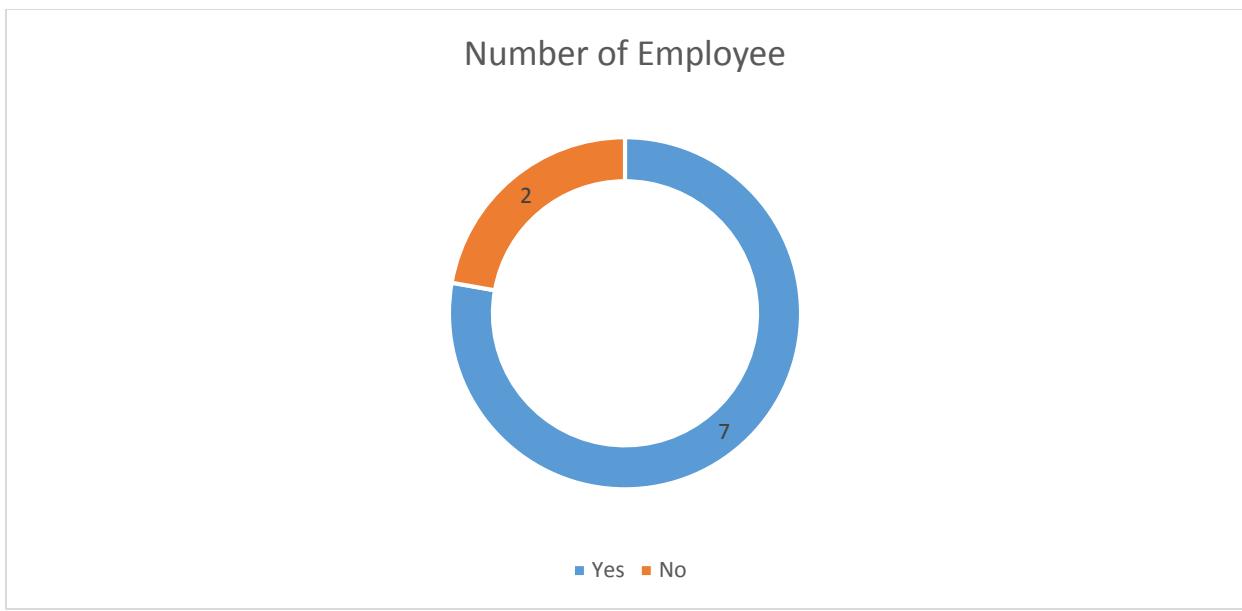
Question 4 - Have you used any automated computer system?

Number of employee



By asking the question, taking the idea about who is the employee failure with automated computer system. Five of them; have no idea about any automated computer system. In addition, three of them do not have used any automated system. One of them is used some automated computer system before.

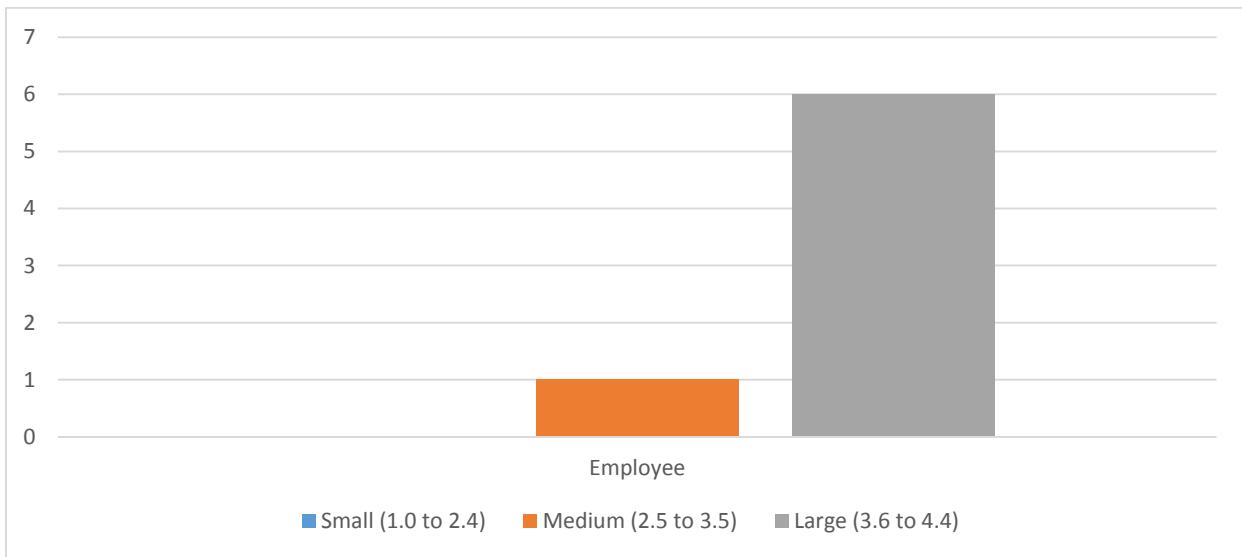
Question 5 - Have you attended any IT courses before?



Seven of the have attended to computer cause and the have some basic idea about the office tools.

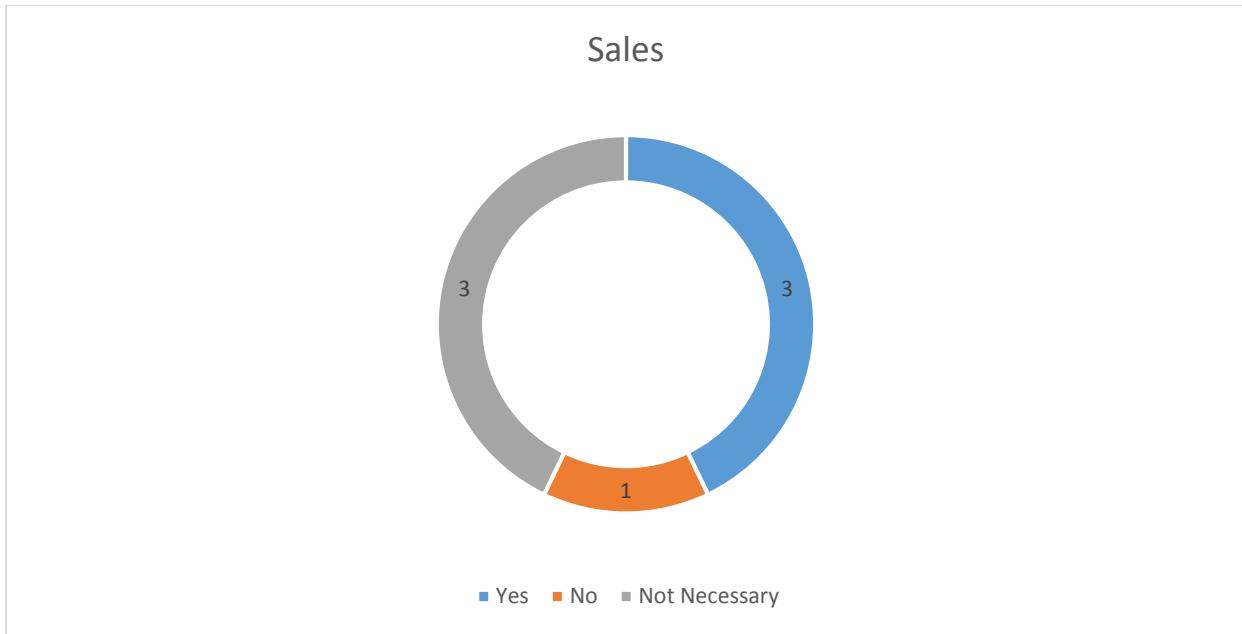
By this question, taking the idea to supply the application with proper device. As the result, the most chosen device is Tablet.

Question 5 - What size would you prefer your screen to be?



By the question taking an idea about the screen size to develop the mobile application. As the result, the popular screen size is large size.

Question 6 - Are interesting to include sounds with mobile application?



By the question checking user attraction with the sounds, in the mobile application, as the report, the Answer No and Answer Yes employees are same and Answer Not necessary employees are less.

Please refer (Appendix C: 2.3.3.1 Analysis of gathered data) for the analysis of gathered data from the interview sessions.

2.2.3 Implementation of Research Technique: Interviews

2.3.3.1 Actual Plan

Topic	Stake Holder	Method	Applicable Date Variables	Steps to Accomplish	Objective	Date & Time
Analyze the current process and Collect Inventory Data	Employee	Group Discussion	Collect information about current process of the inventory system	Ask Question according to the prepared interview Questionnaire	Process flow of the current inventory management process including, Sales process Stock transfer and billing process.	2014/5/20
	Manager	Interview	Item Details, Dealers and all the resources, raw data that using to control inventory	Interview	Basic information about, Items, Prices, Unit details, Categories, Expires	2014/5/20
	Employee		Item Details, Dealers and all the resources, raw data that using to control inventory	Interview as Group		2014/5/20

Please refer (Appendix C: 2.3.3.2 Analysis of gathered data) for the analysis of gathered data from the interview sessions.

3.3 Requirement specification of new system

Analytical, Intelligent, Business and Inventory Automation Systems (AIBIAS) is the application solution that proposed as required to the “Oriental Pharmacy”

With the Analytical, Intelligent, Business and Inventory Automation Systems (AIBIAS), there are two solutions available,

- Mobile Application
 - The Mobile application had mainly supplied for the owner of business to interact with mobile device.
- Windows Application
 - Windows application had provided to the employee, which is the base application for the Mobile application, while submitting some of important data that allowed communicating or pre assigned to notify to the owner via mobile application, windows application will synchronize to the mobile application via the web services through the internet.

Completely those two applications are **metro style** based application. Another very important features is “**Detailed Business Dashboard**” concept, those concepts are come up with the both Mobile and

“**Analytical, Intelligent, Business and Inventory Automation Systems (AIBIAS)**” is accessible from any location whether or not here in Haiti or the state or where around world. The Mobile application even permits to urge Updates concerning this satiation of the business likewise these days Sales, Stock and every one the notifications concerning the sales and inventory.

3.3.2 Functional Requirement

- Summarizing the whole business calculation to the graphs and new Gauge, Widgets are including in to the Dashboard to the application home.
- System should be able to Syncs with Mobile Application
 - Sends new invoices
 - Synchronizes the relevant Updates
 - Send invoice payments as notifications
- System should be able to use Mobile Application Integration for Inventory
 - Business owner able to login through the Mobile Application and see customers,

inventory, and everything that is accessible through the online.

- System should be able to process Price Lists

Multiple tariffs is related to one Central inventory, giving to business owner the liberty to supply completely different rating supported location, and/or client cluster. Tariffs may be allotted start/end date for promotional markdowns

- System should be able to Sales Promotions

From BOGO to specific off next purchase, Analytical, Intelligent, Business and Inventory Automation Systems (AIBIAS) offers you the liberty to form, run, and later read powerful reports on Promotions.

- System should be able to Multiple rate Support

Tax rates are assault the merchandise or location level. Set and manage completely different tax rates for all of your store locations!

- System should be able to Location Specific fast Tabs

Hot sellers could dissent by store location, thus why should not your fast buttons? Analytical, Intelligent, Business and Inventory Automation Systems (AIBIAS) supports store specific quick-tabs.

- System should be able to process Reorder and stock Levels

Automate your ordering method by process Reorder points and stock levels for product. Later use these rules as recommendation or rule for ordering.

- System should be able to Purchase Order Creation

Create Purchase orders to send dead set suppliers by email, or print out and send the hardcopy. Customize text, and add your emblem for that skilled feel!

- System should be able to Purchase Order Receiving

Track commercial instrument standing to visualize what has arrived and what is in transit. Write off broken product, and track partial orders.

- System should be able to make Stock renewal Report

One report is back to revolutionize the ordering method. One click of a button adds quantities supported stock level, another click to get commercial instrument documents to the multiple suppliers associated.

- System should be able to process Purchasing Report
Clean report by provider and/or product ordered in given date varies. Read and print from anywhere, at any time.
- System should be able to use with Cloud Hybrid System
Run your POS system each on-line and offline. System is freelance from net access and information is auto-synced upon re-connection.
- System should be able to check Suspend Sale
Place sales on hold and open up at later time for fast checkout. Add product. Found item is not within the system. No drawback, add or edit product on the fly right at POS.
- System should be able to process Void/Return
Create returns with ease, choice to print receipts with scalable barcode to hurry up the process!
- System should be able to make Speedy Product operation
Query product by name or code, scan barcodes, or choose things from inventory list to feature them to the group action.

By subscribing with the proposed system, owner and employees can revenue the solutions to the simply noticed problems as below mention.

According to the problems that found by the researching around pharmacy situation, decided solutions could be contemporary with those contains.

- **Problem**

Poor Communication

Since manual inventory, system does not facilitate the communication within the geographic point.

Solution

A business can avoid communication problems using the **Technological business solution**, by using special function of a system, network based communication functions based on intranet, LAN or Internet or automated other communication media. So also can save the time, Money with responsibility and quality of the company.

- **Problem**

Errors with Physical calculations

Physical inventory counts need to performer oftentimes to manage the things within the inventory.

Solution

A business can avoid Errors that come up with Physical calculations by using **Technological mathematical logics** by using automated calculation with inventory, Invoicing and financial calculations,

- **Problem**

Tracking Daily Purchases

Solution

A business can easily trace, make reports and get decision using report generation methods with **Technological business solution**, today world, can automatically generate any kind of reports, most automated systems are provided that type of function to customers.

- **Problem**

A manual inventory system does not update at the top of the day with updated inventory counts.

Solution

Using the automated alert system business owner, manager or responsible person can take alert about the current situation of the stock, sales daily sales, new updates and all. That path can be mobile alert and Mobile application that specialized for a business. Using that type of facility with the system can names most Transparency of a business.

- **Problem**

Businessperson does not have quick access to data concerning the Business

Solution

Today world portability technology can make up to date businessperson. Using portable child application with mobile device that made for the current organization or company, can take updates and alerts from anywhere in the world.

3.3.3 Non- Functional Requirements

- System should be able to provide accessibility irrespective of the location, Portability By giving the mobile application with the proposed application portability and access anywhere in the world, will be achieved.

- System should work in best Performance

Using the cloud data transferring methods and using the fastest devices as example fast computers and fast mobile devices with the new features, employee and owner can enjoy best performance experience with the proposed system,

- Accuracy and Precision

Using Digital calculations in the system, employee and owner can take most accurate and precise details, information and final calculations by the system to take decision.

- Reliability and Security

When doing a business reliability and security is most important thing. By giving, the login authentications and using virtual private (VPN) networks to communicate via the internet and using Security Socket Layer (SSL) with the mobile device data connections with encryptions, employee and owner can trust the application.

- Usability

Usability is most important thing while proposing and introducing new functions or new application to the society, in the proposed system, all of interfaces are designed for touch displays and, based on latest concept provided by Microsoft "Windows Metro Style App". Using more space in the desktop, owner or employee can avoid the lots of mistakes and

missing points while checking and entering data to the application. It is better than using mouse and keyboard to handle the application.

3.3.4 Technical Requirement

To get the best performance, accuracy and expected performance, system should run with minimal requirements as developer proposed.

In computer side, requirement should be

Processor Intel core i3 3.2 GHz
RAM 4 GB
HDD 500 GB
VGA 250 MHz
Monitor 15'inchs
Operating System Windows 8.1 Pro
Including Input, Output devices.

In Mobile Device side, requirement should be

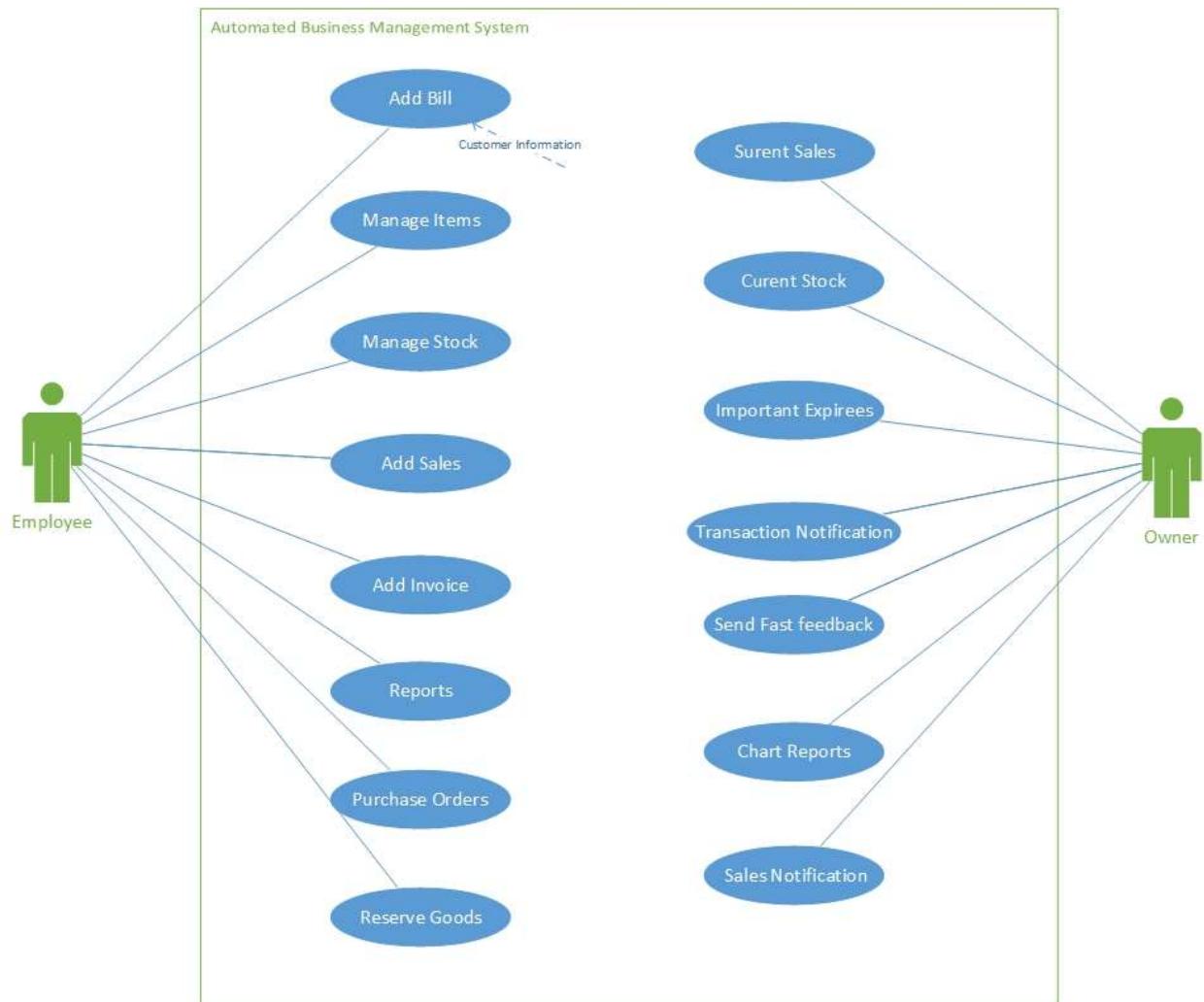
Processor dual core 2GHz
RAM 1 GB
HDD 500 GB
Graphic mobile graphic processor
Monitor 7.5'inchs
Operating System Windows Mobile 8.1 Pro

Including,

- Internet access with fast internet connection
- Good employee to handle the system

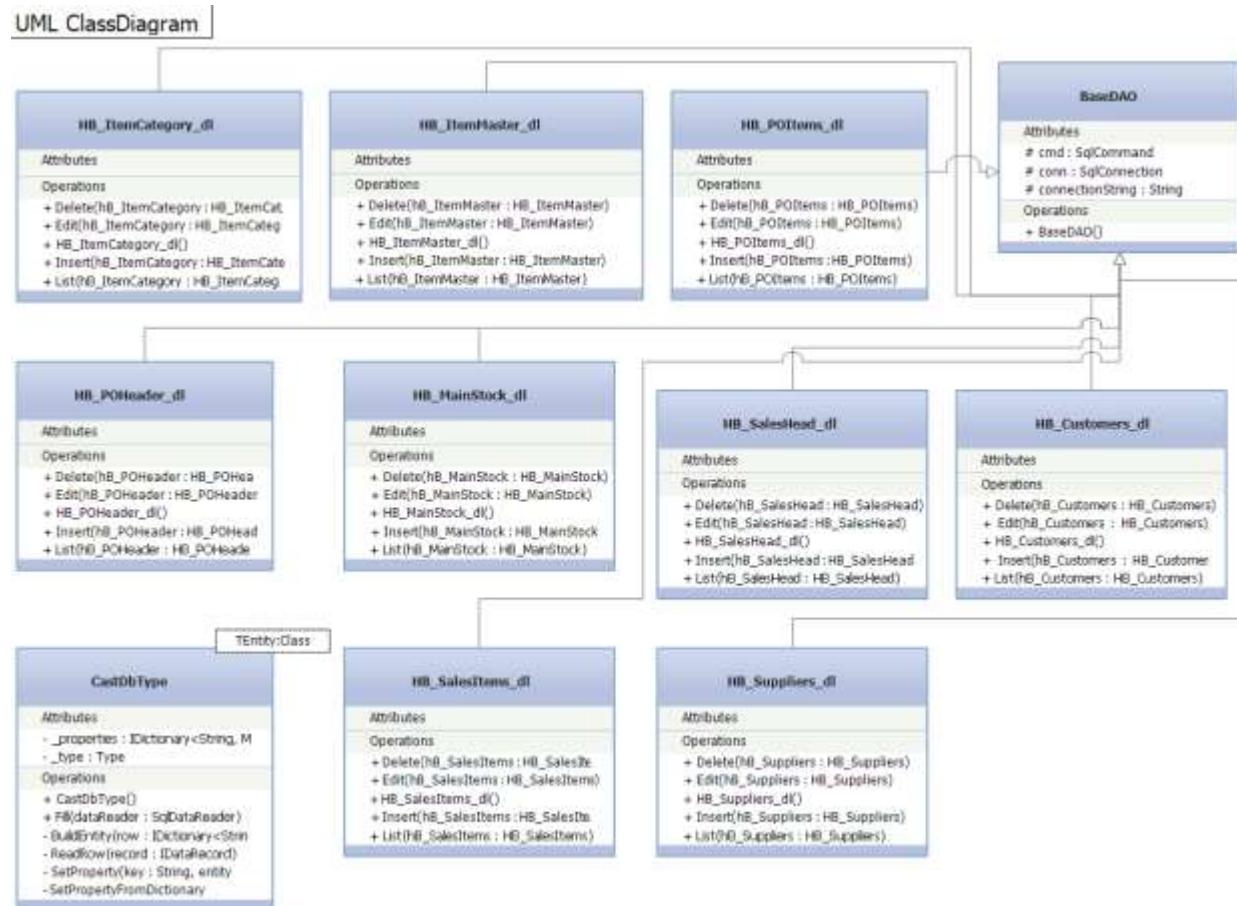
3.3.5 UML modules for the new system

3.3.5.1 Use case diagram for the new system



3.3.5.2 Class Diagram for the new system

Database Layer



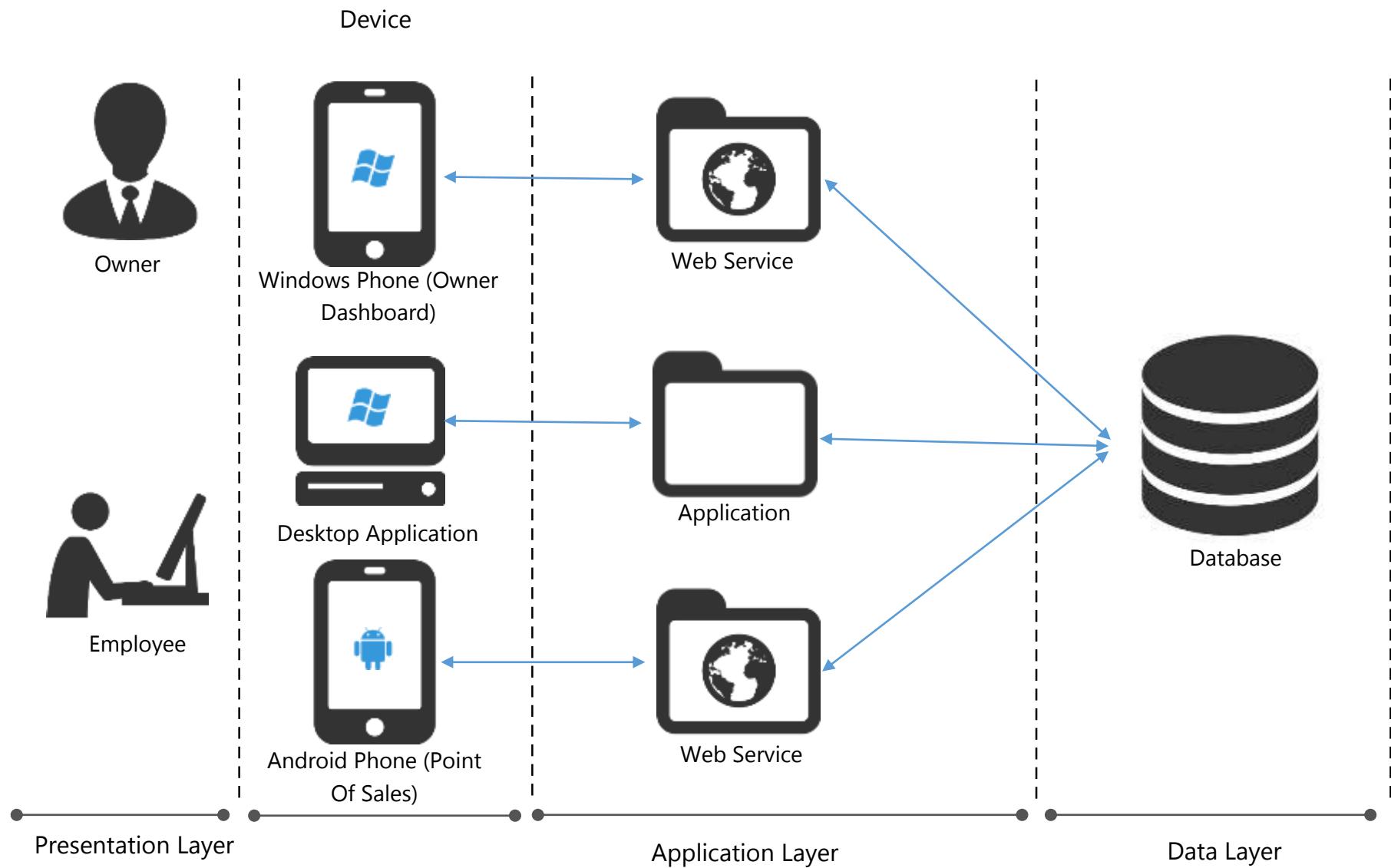
3.4 Summary

This chapter was dedicated to gather the data and analyses the existing system to make the proposed application's requirement and to analysis properly by evaluating exist problem and force to the design phrase properly with analyzed problems with solutions.

4.0 System Design

This document provides a comprehensive architectural overview of the system, using a number of different architectural views to depict different aspects of the system. It is intended to capture and convey the significant architectural decisions that have been made on the system.

4.1 Overall System Architecture



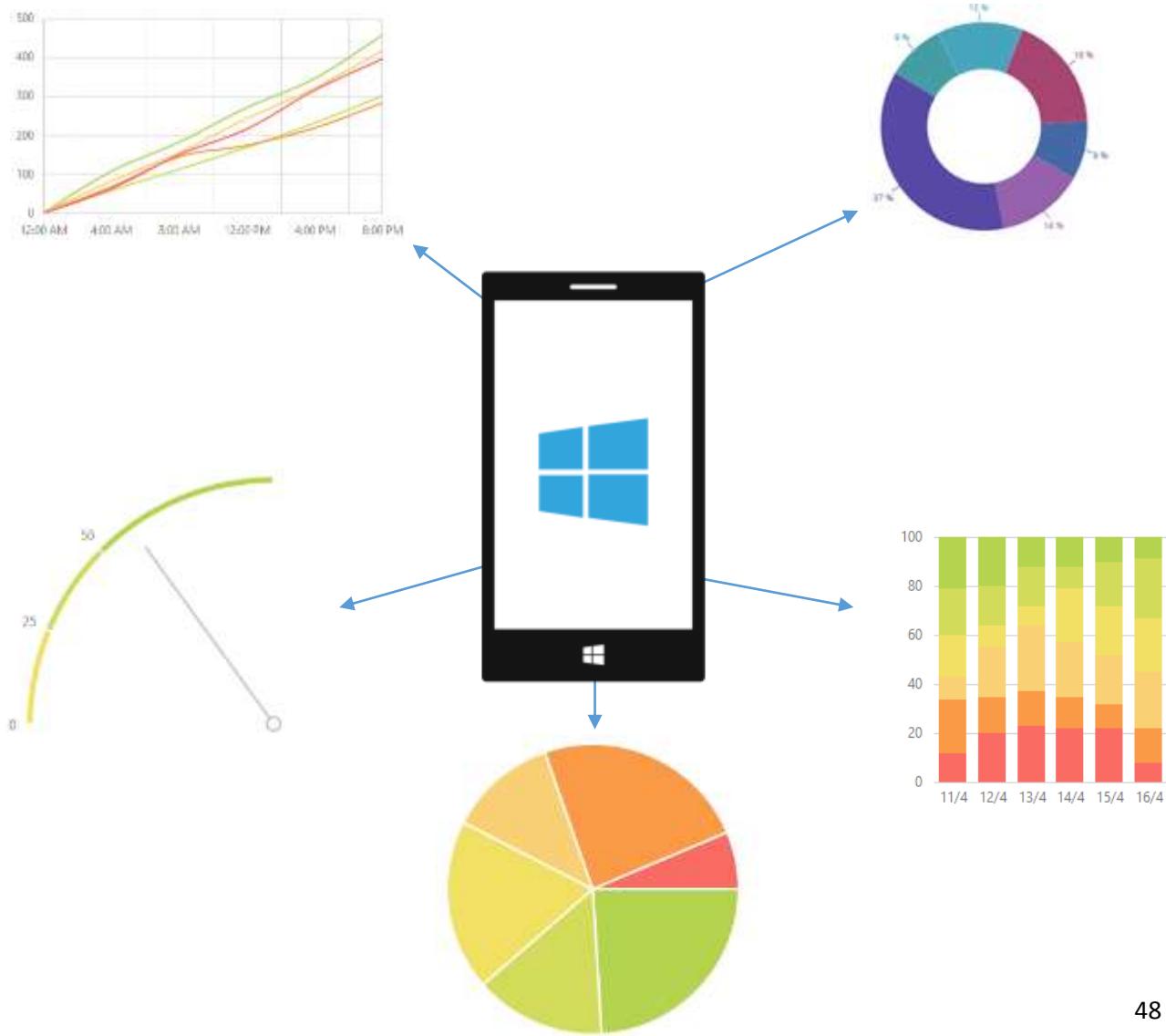
4.1.1.1 Owner – Windows Phone Application



Owner can use the mobile device with the application, by connecting to the internet owner can view the summarized and detailed business dashboard on the mobile screen.

The dashboard is contain with important, pre produced notifications and many information that need to highlight daily business deals the.

- Sales forecast / Chart
- Purchase forecast / Chart
- Movement of stock (In Chart form or Numerical Fig)
- Stock Value
- Loss & Profit



4.1.1.2 Employee - Windows Application



Employee can access to the Main windows application of the Analytical, Intelligent, Business and Inventory Automation Systems (AIBIAS). Employee is the data entry member, updating invoices, Stock details. Good Reserving notes, Purchase Invoices, and full of handling part are going with windows application by the employee.

As the most attractive and important component of the Analytical, Intelligent, Business and Inventory Automation Systems (AIBIAS) windows application, the detailed **Dashboard is included to the application** to quick review of the business,

- Purchase order request.
- Movement of stocks
- Availability of stocks
- Re-Order levels
- Stock in demand
- Expiring stock in hand
- Generating Reports (Automated)

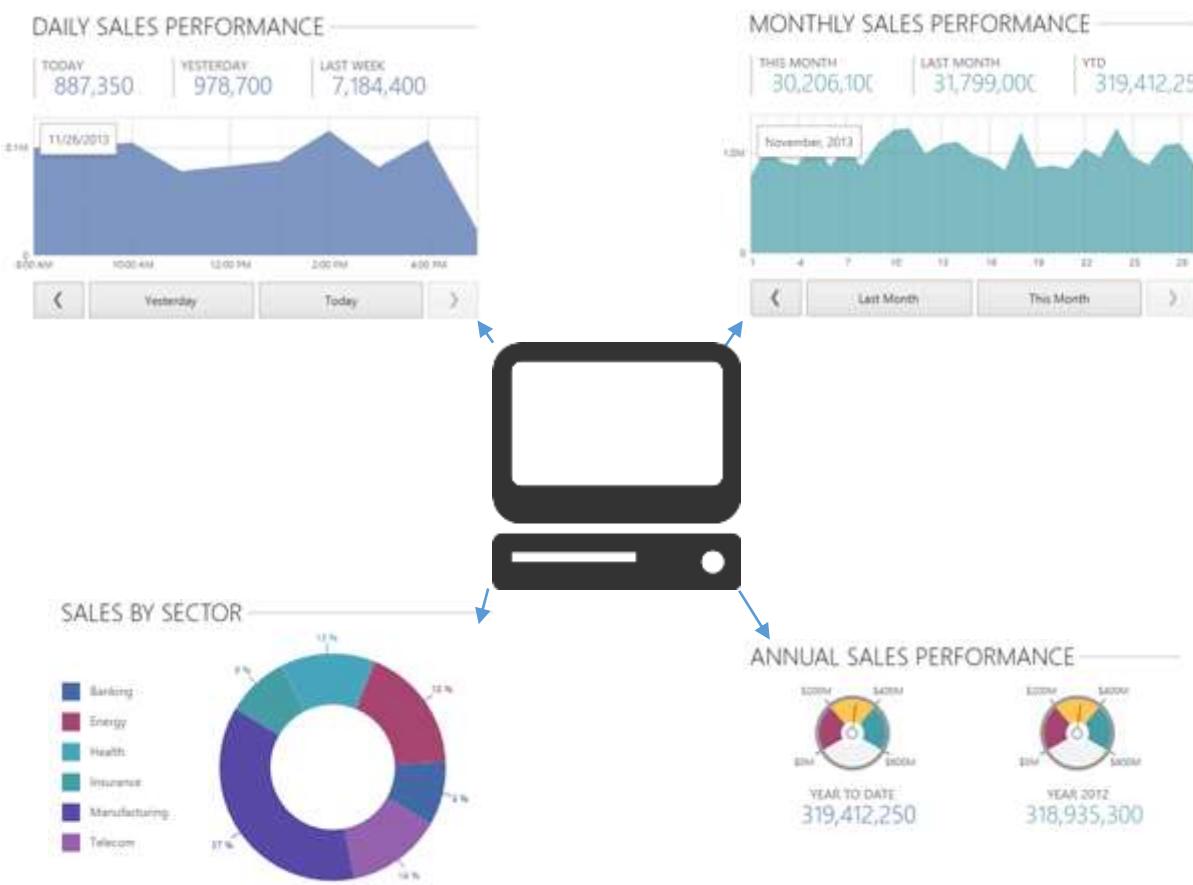


Figure 9 User Functions

4.1.1.2 Employee - Android Point of Sale application



Employee can access to the android application of the Analytical, Intelligent, Business and Inventory Automation Systems (AIBIAS). Employee can do the invoices without accessing the main application. Employees able to work any place in the business area. The application has been designed for the local network that connects to the main database via the WIFI network.

Employee able to view the live and updated data from main live database.

4.2 Software Architecture

4.2.1 Overall Windows Application

Default Layer architecture in the proposed system

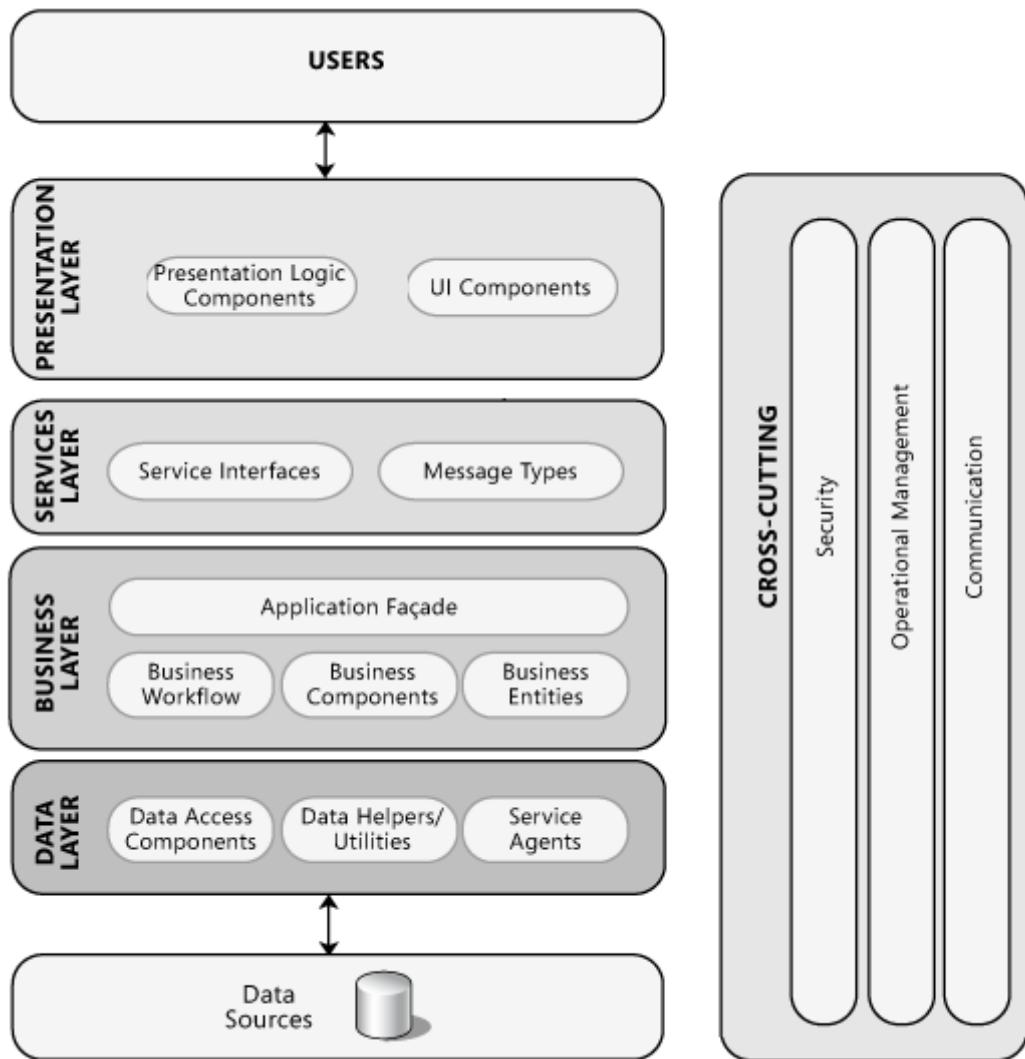


Figure 10 Layer Architecture

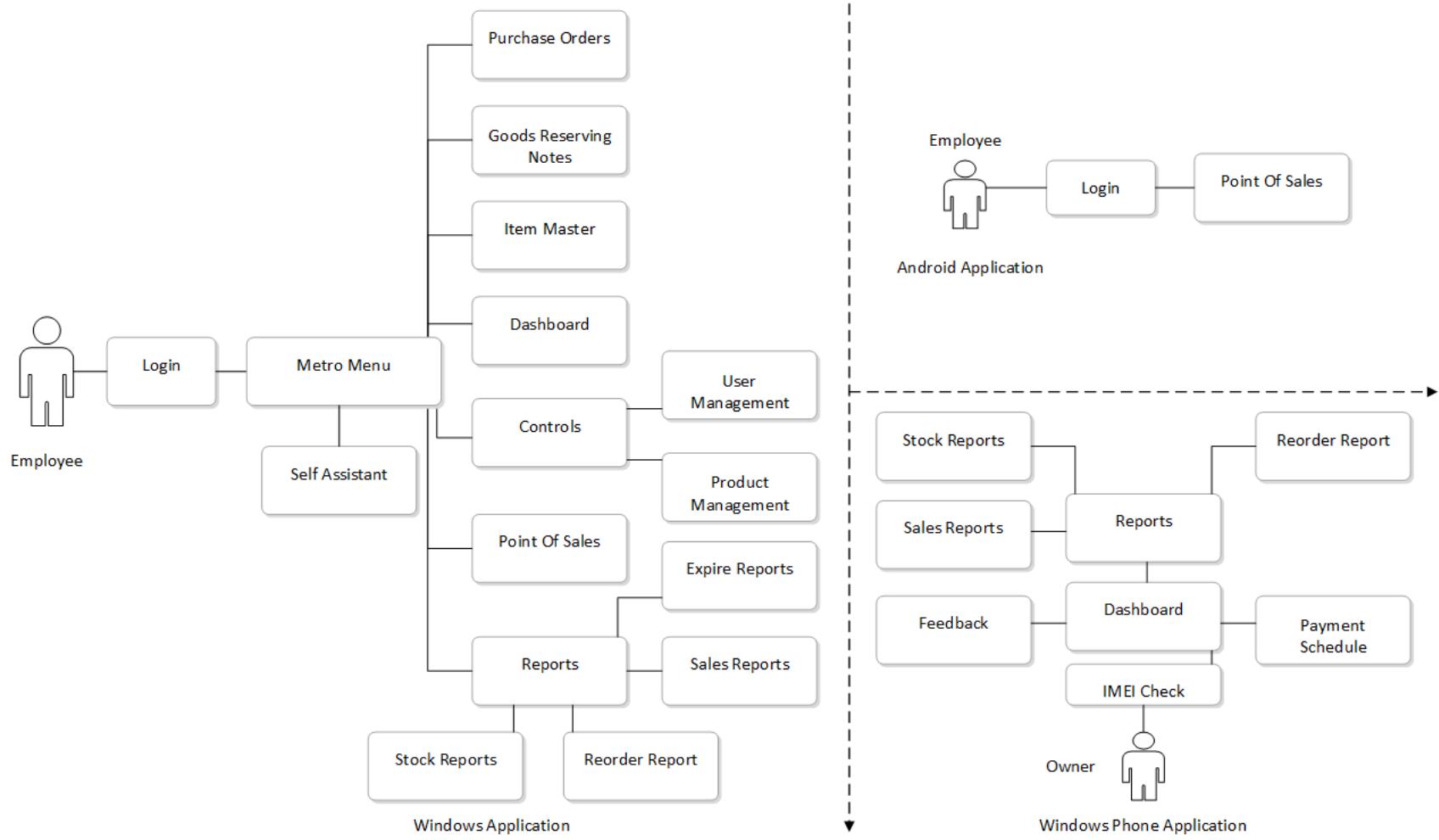


Figure 11 Module Architecture

4.2.1.1 Windows Application

As the System Architecture, the Employee can log in to the system by confirming the Username and password, after the login to system, user can begin with Dashboard, the dashboard is representing whole summary current situation of the business, as the system user can explore the sales, inventory, stock, expiries etc...

The menu tiles also included in to the dashboard and users can navigate from detailed tile buttons. By clicking in the tile button user not be able to open the expected window separately and user only can navigate from the main dashboard to targeted window.

In the dashboard screen user can see the sub modules (the forms connected from detailed tiles)

General Forms

- Purchase Order
- Point of Sales
- Item Master
- Stock Transfer
- Controls
 - User Controls
 - Product management

Reports

- Stock Report
- Item Report
- Expiree Report
- Re Order Report

4.2.1.2 Mobile Application

In the mobile application is user need to verify the IMEI before log in to the system. The process is automated. Windows application is able to manage the IMEI number that can connect with the server.

Basically owners and responsible persons only log in to the system and owner in the person who can visible all the features as the admin privileges. Other mobile users are can be the admins or some special user. Those all privileges are assigned to the IMEI number.

In mobile application, users are able to view the graphical views of the situation of the business and users are able to feedback according to the situation to responsible persons, or selected user groups.

Mainly all graphical contexts, are shown as the Graphs, charts, and tables, to better analyze.

4.2.2 Module Architecture

4.2.2.1 Module 1 – Dashboard

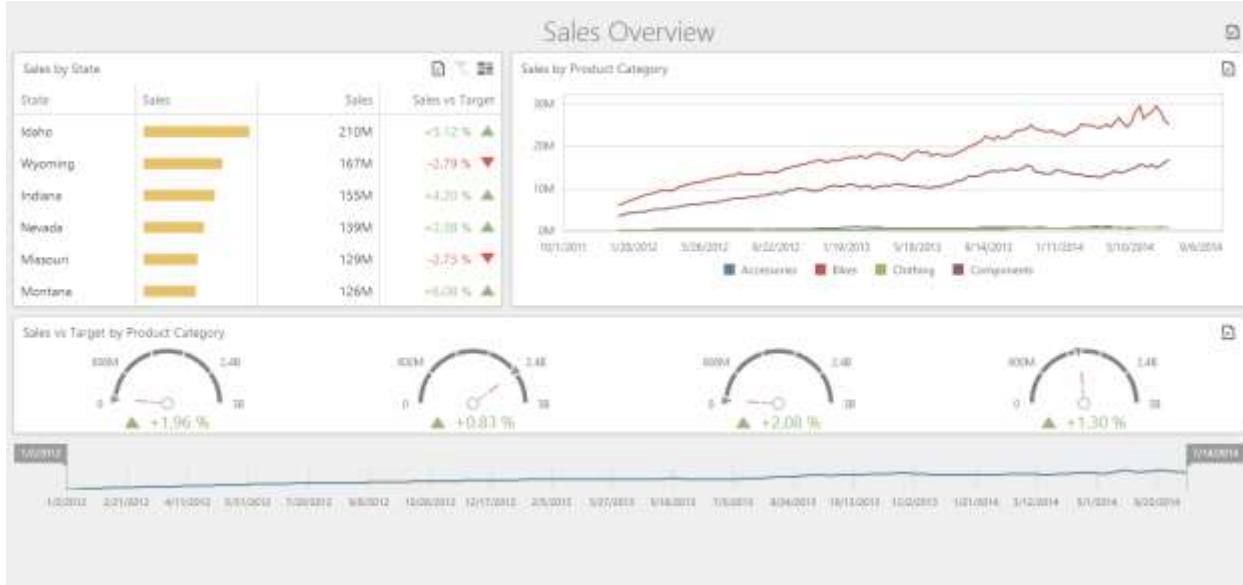


Figure 12 Sample Dashboard

The Dashboard is the most important and intelligent part of the application. The dashboard is representing the Analytics data of the current situation with sales and stock.

User can overview without the reviving the whole report and easy to get decision about the business.

The best Enterprise way to the analyses the statuses is dashboard

By the Dashboard is insightful and information rich decision support systems for executives and business users across platforms and application is a simple matter of selecting the appropriate UI representing element like Chart, Pivot Table, Data Card, Gauge, Map or Grid.

4.2.1.1 Self-Assistant in the Dashboard

In the application, by turning on the Self-Assistant user can ask the basic situation from the voice command from the assistant without any review of the report or dashboard elements.

4.2.2.2 Module 2 - Purchase Orders

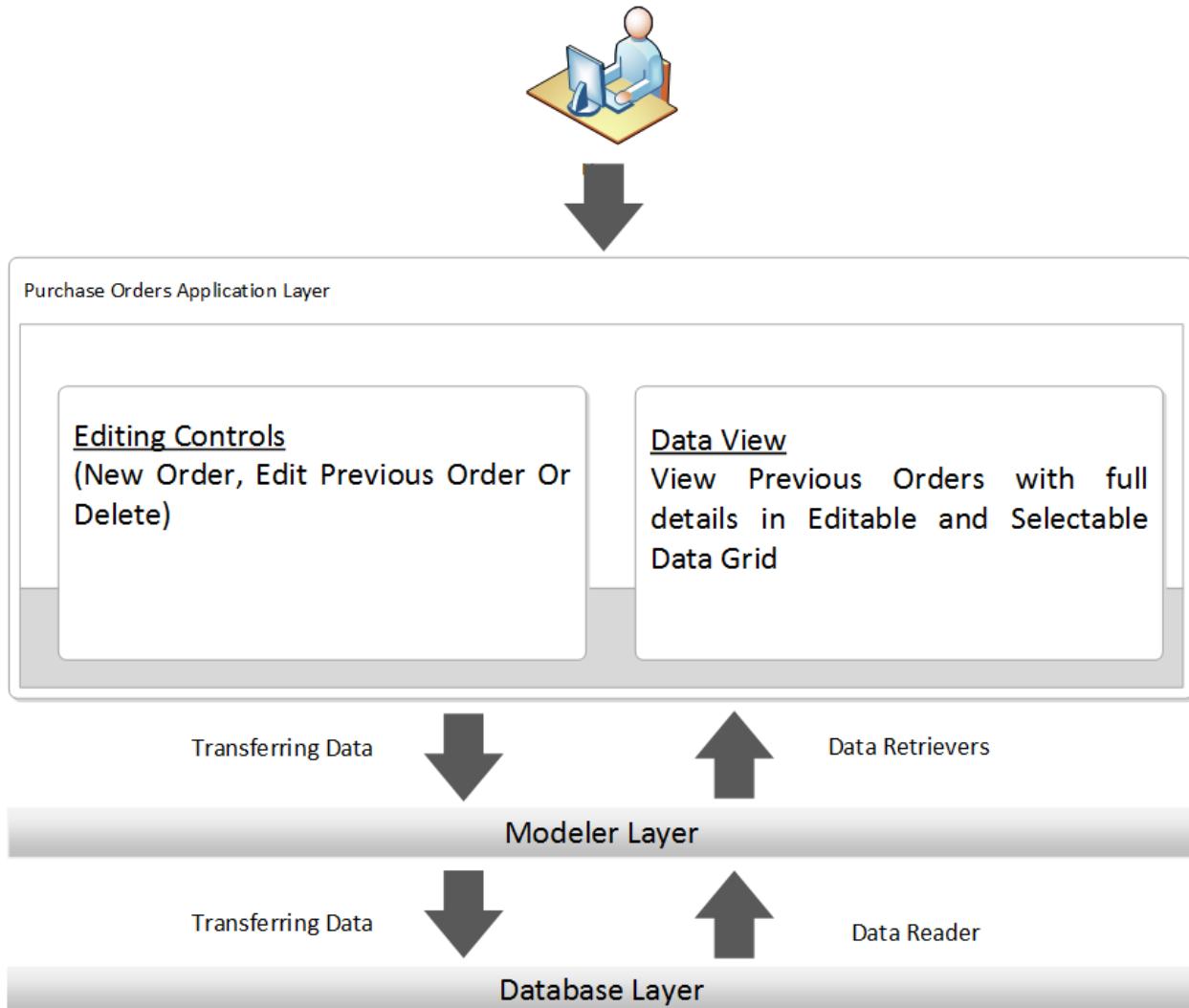


Figure 13 Purchase Orders

The module of Purchase Order is the responsible module to manage purchases, mainly user can make the purchase order and print the order to issue. User can add exist product to the system and justify the cost of the issued invoice.

4.2.2.3 Module 3- Goods Reserving Notes

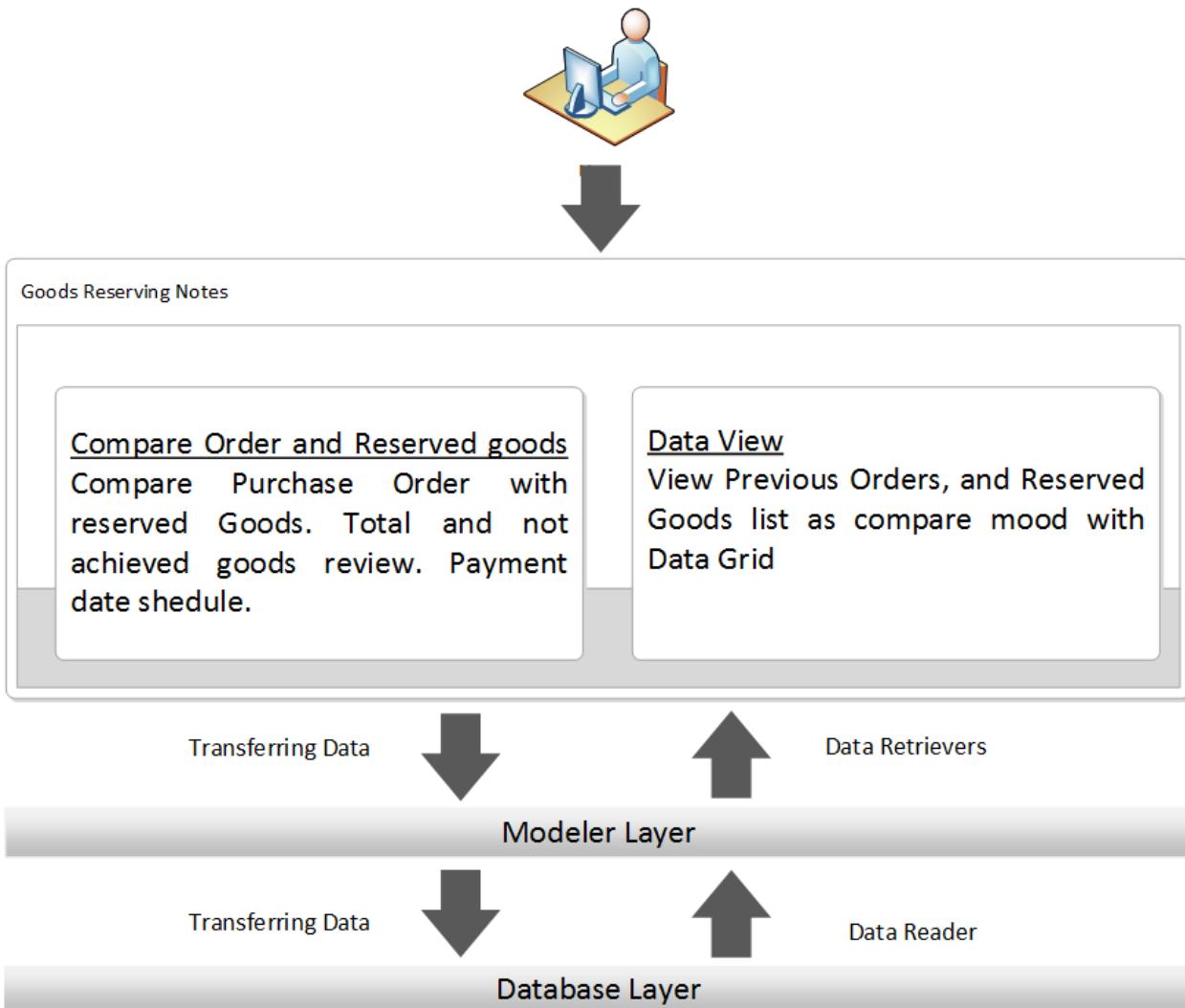


Figure 14 Goods Reserving Notes

The Goods reserving notes is part of a purchase order, after reserve the goods according to the issued purchase order, the goods reserving module is using to compare the invoice and reserved goods according to the saved records. In this user can generate the sample report of the reserved goods,

User can directly transfer the reserved goods to the inventor and update stock the update stock process is automated.

4.2.2.4 Module 4 - Item Master

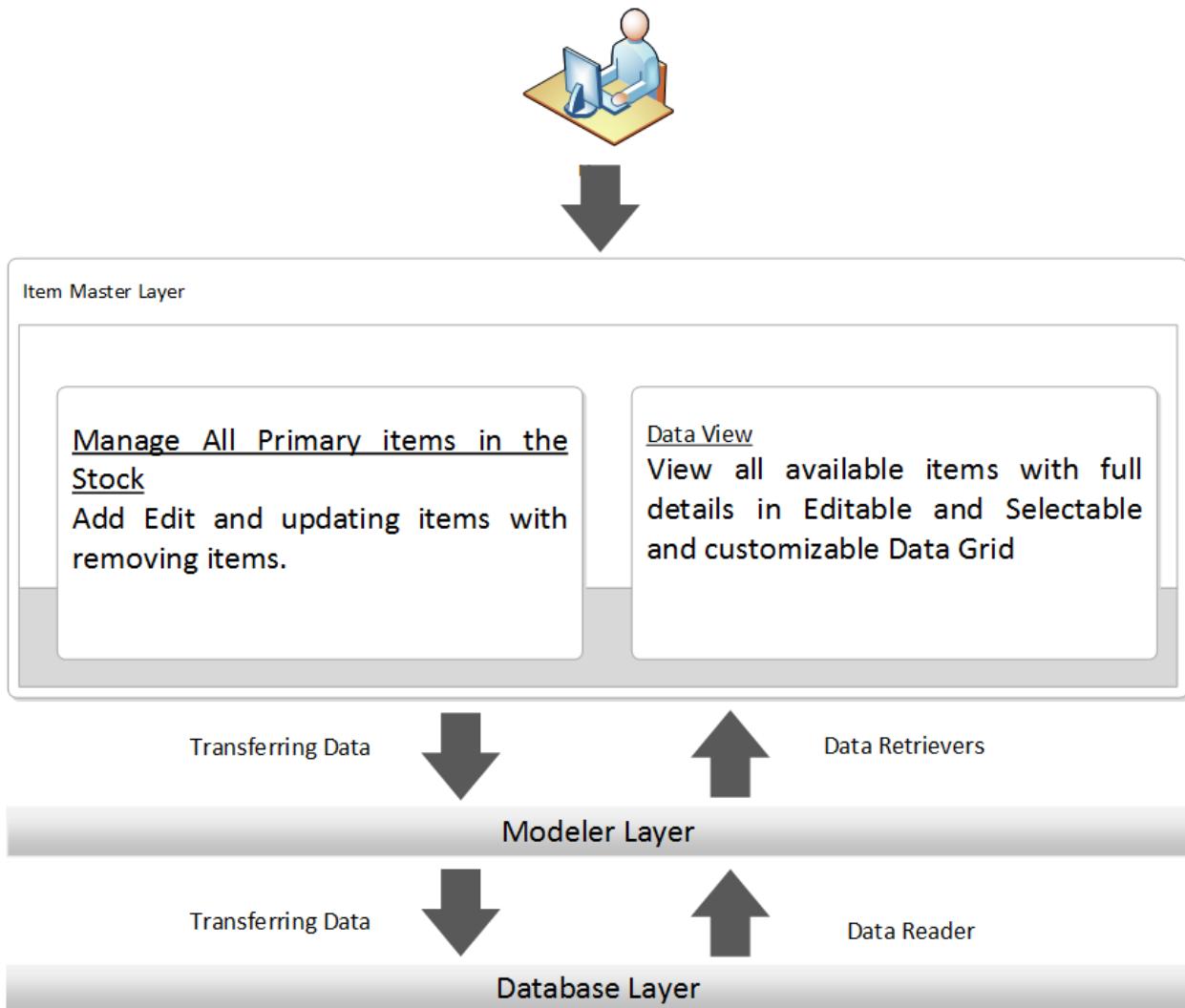


Figure 15 Item Master

Item master is managing item records that contain with inventory, user can overview all the available items and change the item primary details. In the module user can expand all the items by customizable grid view from dragging and dropping the columns to group border and user can filter and group according to chosen column in the data viewer (grid view).

4.2.2.5 Module 5 - Controls Overview

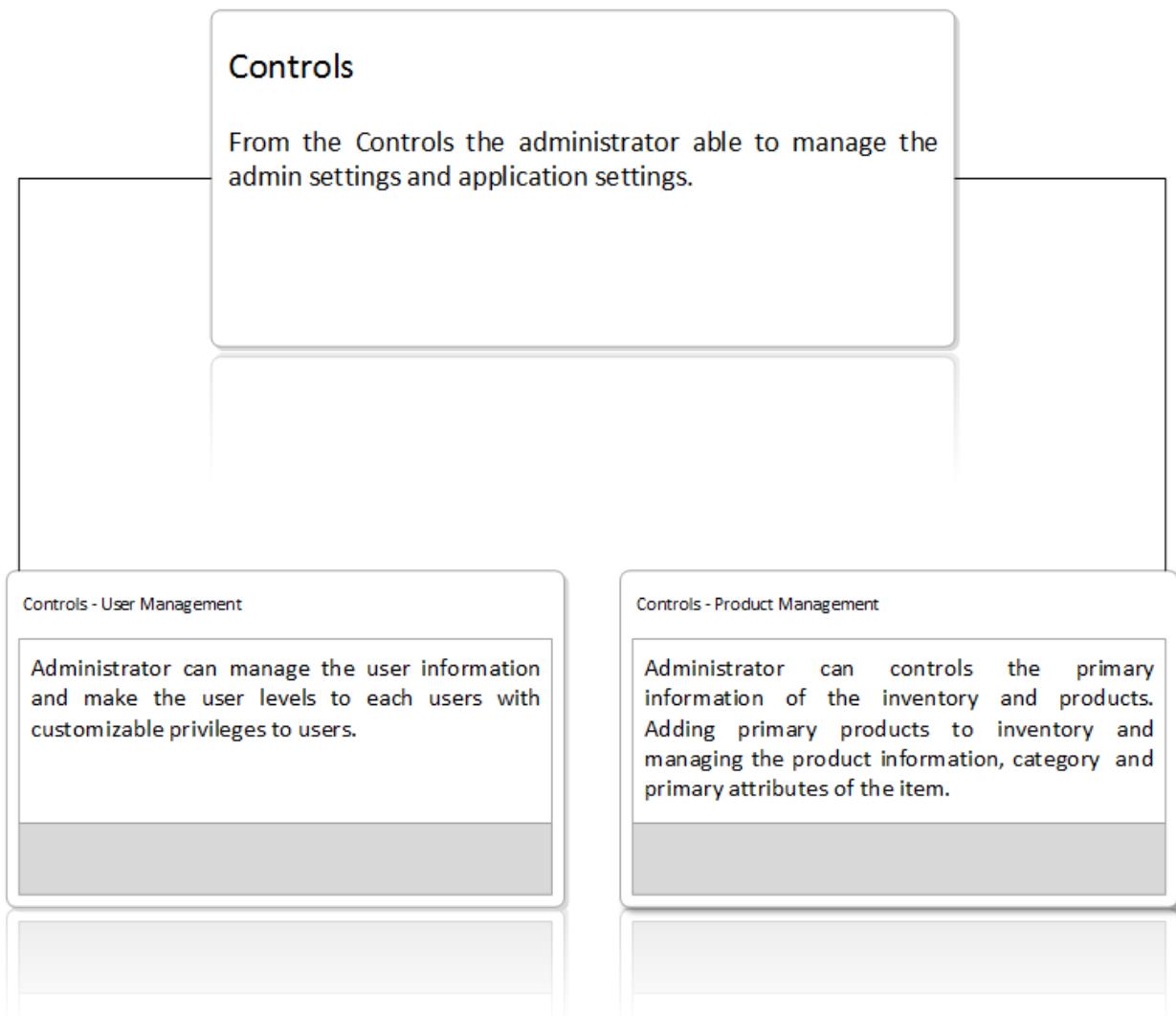


Figure 16 Controls Overview

The controls is dividing to two part of the application,

1. User Management – manage user information and control the user levels.
2. Product Management – manage product details like product categories and settings regarding the product.

4.2.2.6 Module 6 - Controls

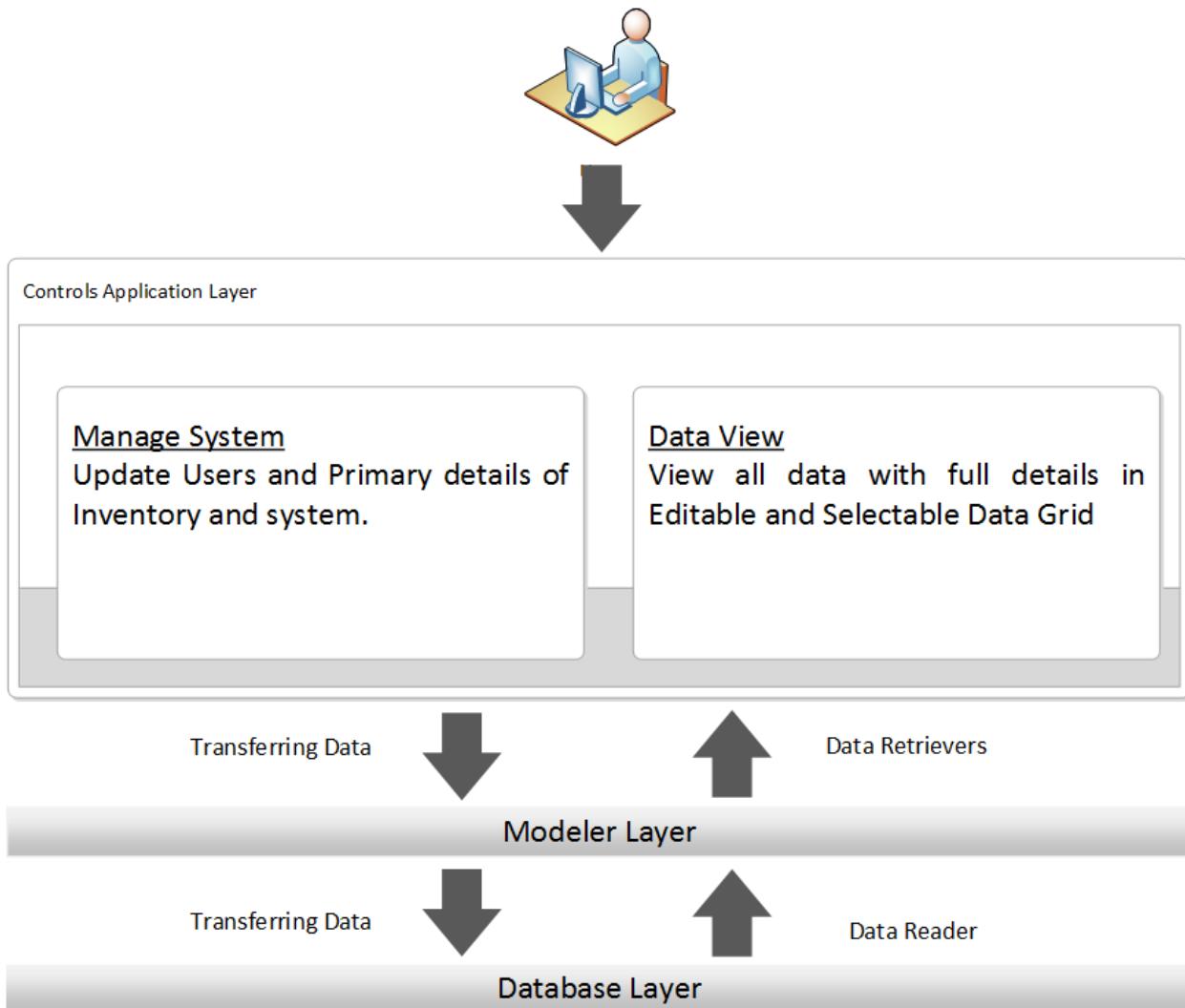


Figure 17 Controls

User can manage basic information of the system and manage product information. The controls is dividing to two part of the application,

1. User Management – manage user information and control the user levels.
2. Product Management – manage product details like product categories and settings regarding the product.

4.2.2.7 Module 7 - Controls - User Management

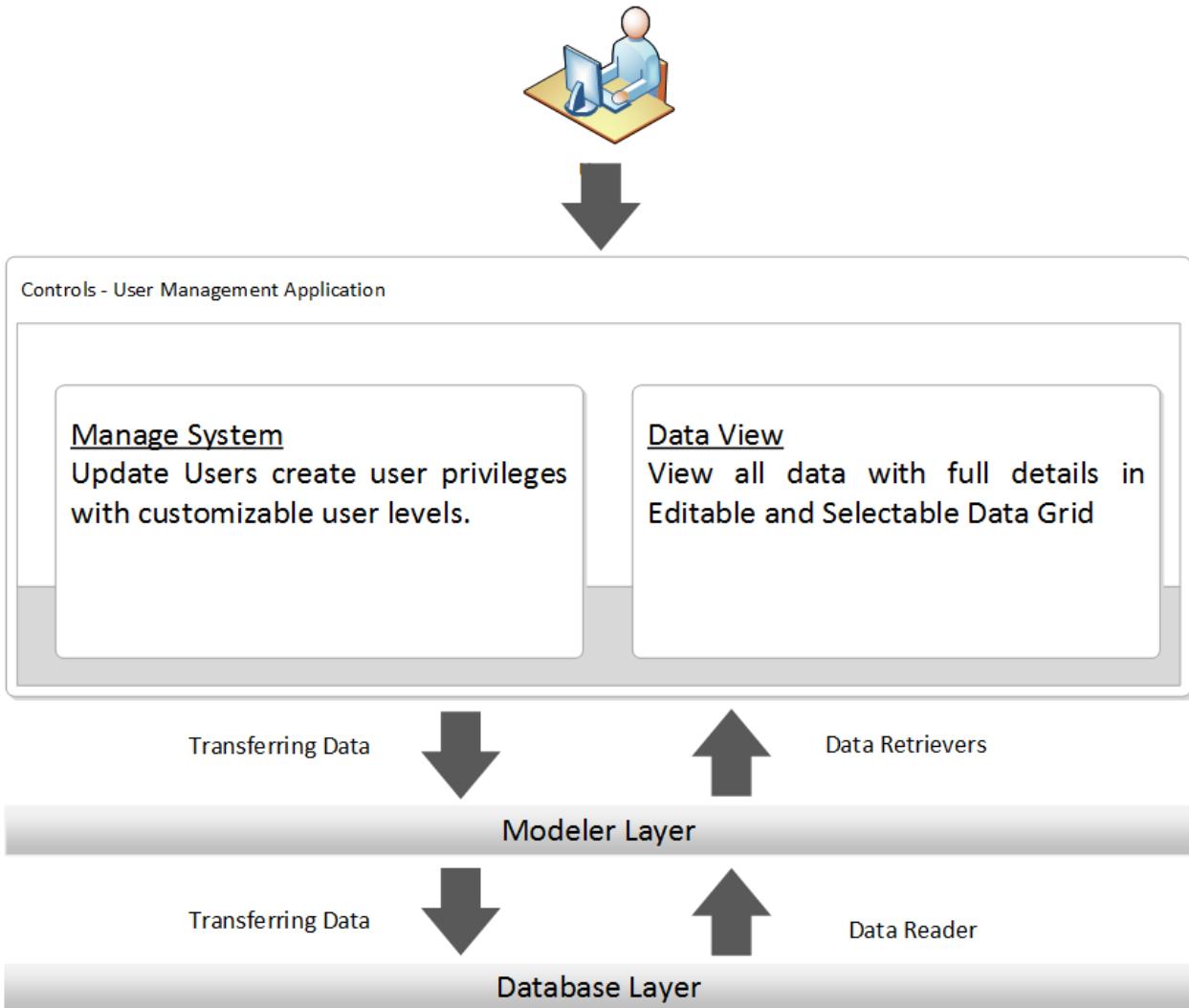


Figure 18 User Management

The user management can change the user information and user levels with privileges. This is an administrator function. And only administrators can access to the User Management.

4.2.2.8 Module 8 - Controls - Product Management

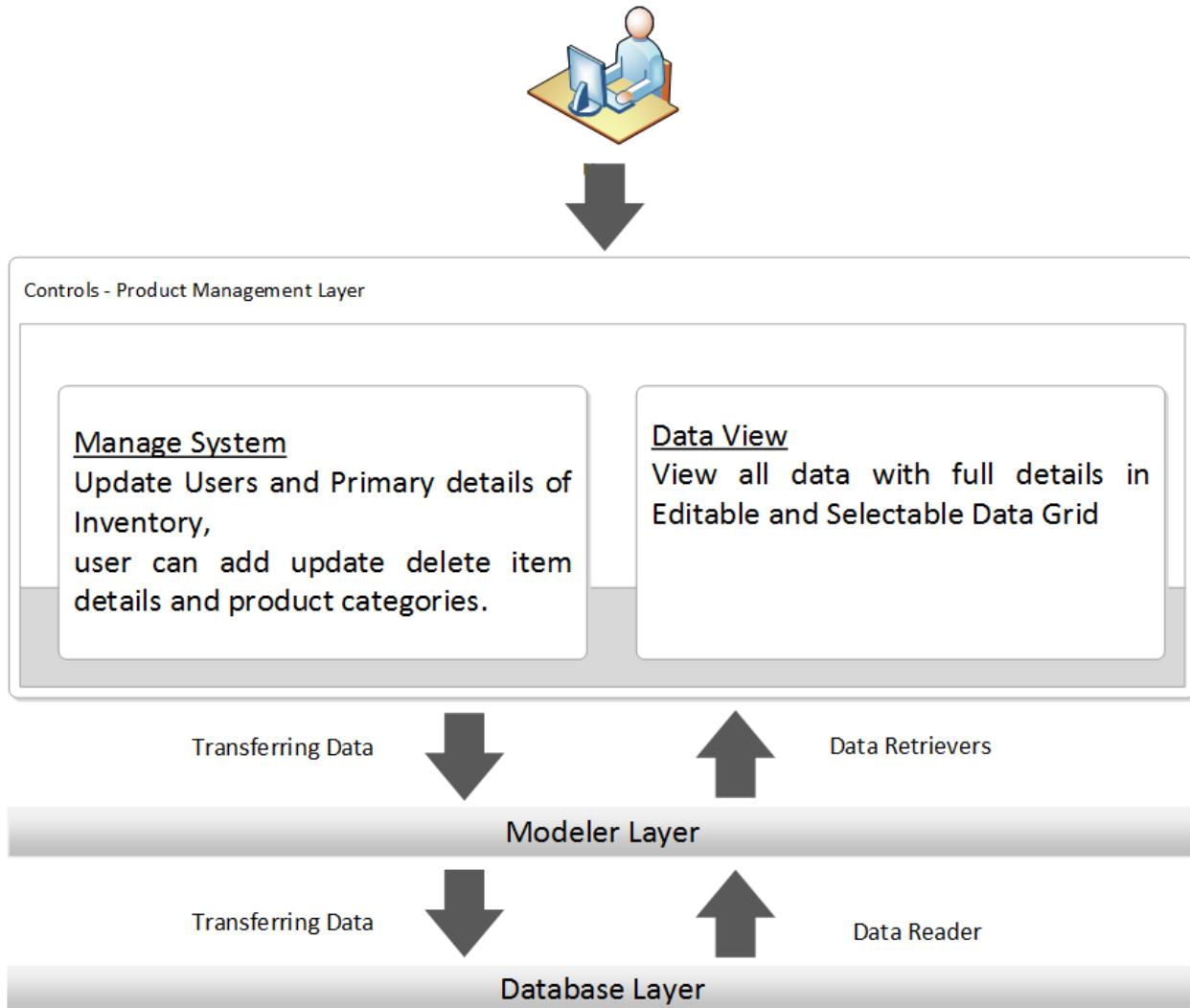


Figure 19 Product Management

Product management also administrator function and only administrator can change the product primary details like product categories and product prices and all primary details about product.

4.2.2.9 Module 9 - Point of Sales

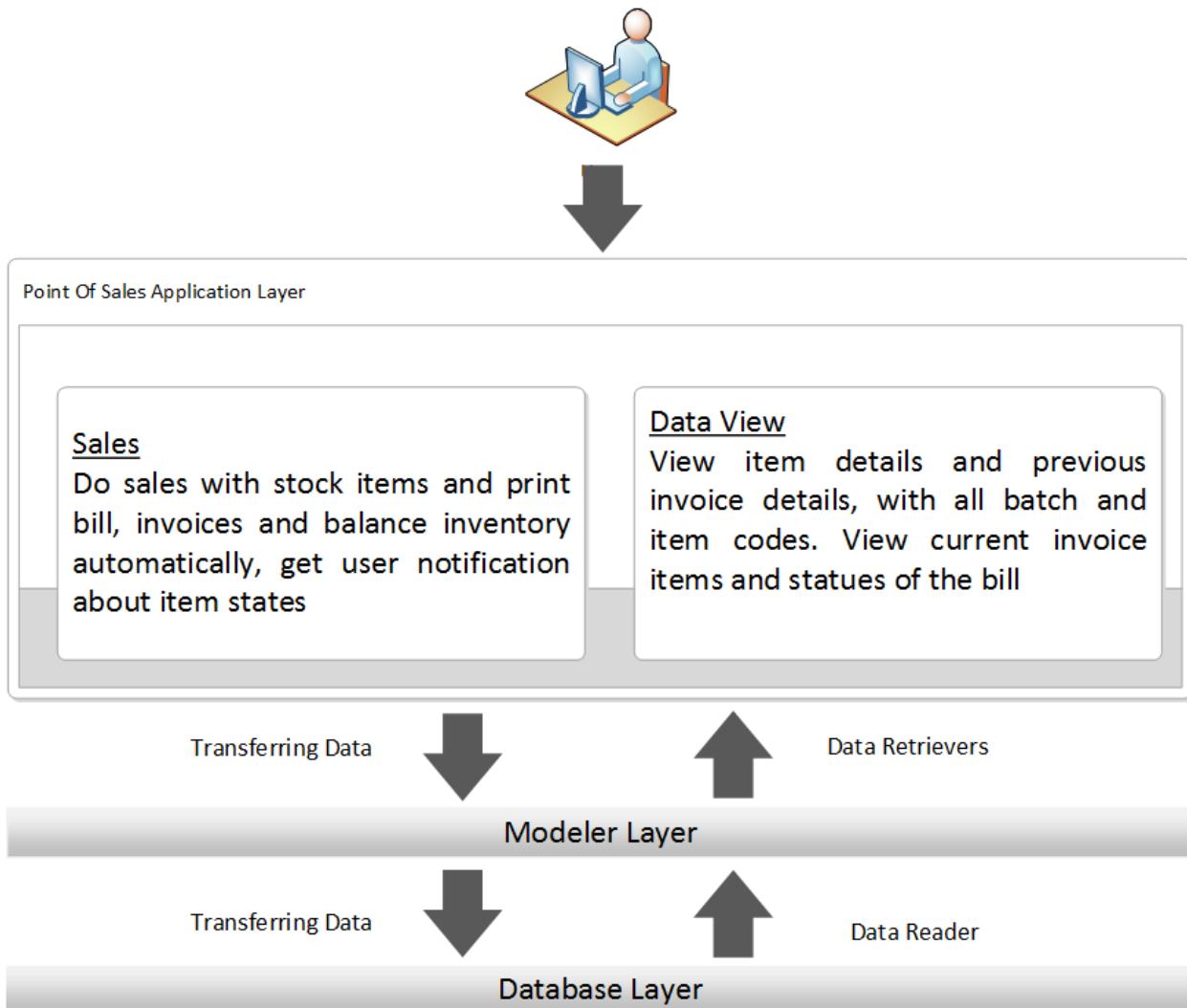


Figure 20 Point of Sales

Point of sales is main part of the application, the point of sale is managing the sales part of the application and user can add items to an invoice and proceed the invoice and user can print the invoice as the requirement.

From the module user can view all primary details of the entered batch number or product id or product name, user can use any of these attributes to make an invoice as a key for the item details.

After the invoice the inventory balance and report and notification are automated,

4.2.2.10 Module 10 - Report Overview

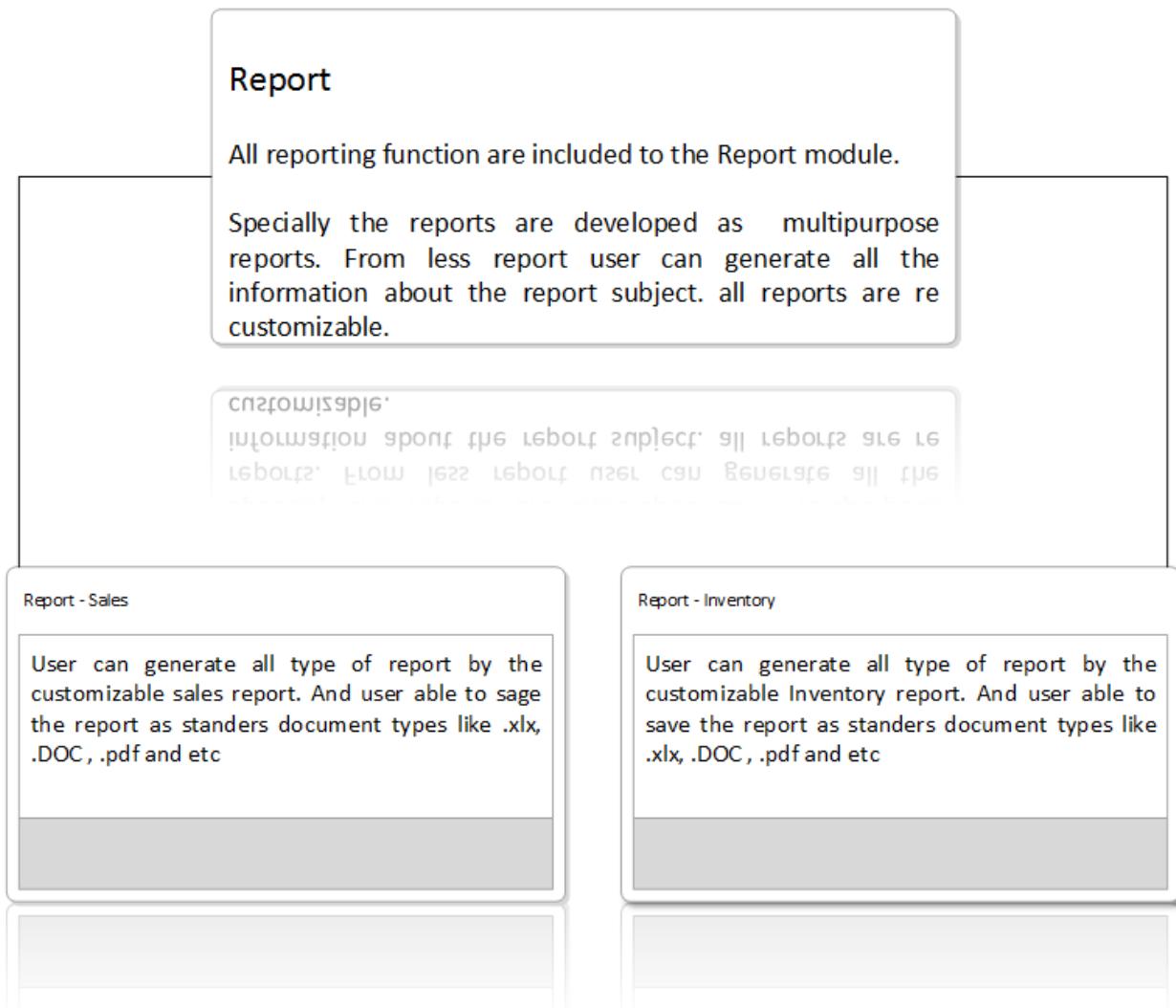


Figure 21 Report Overview

4.2.2.11 Module 11 - Reports

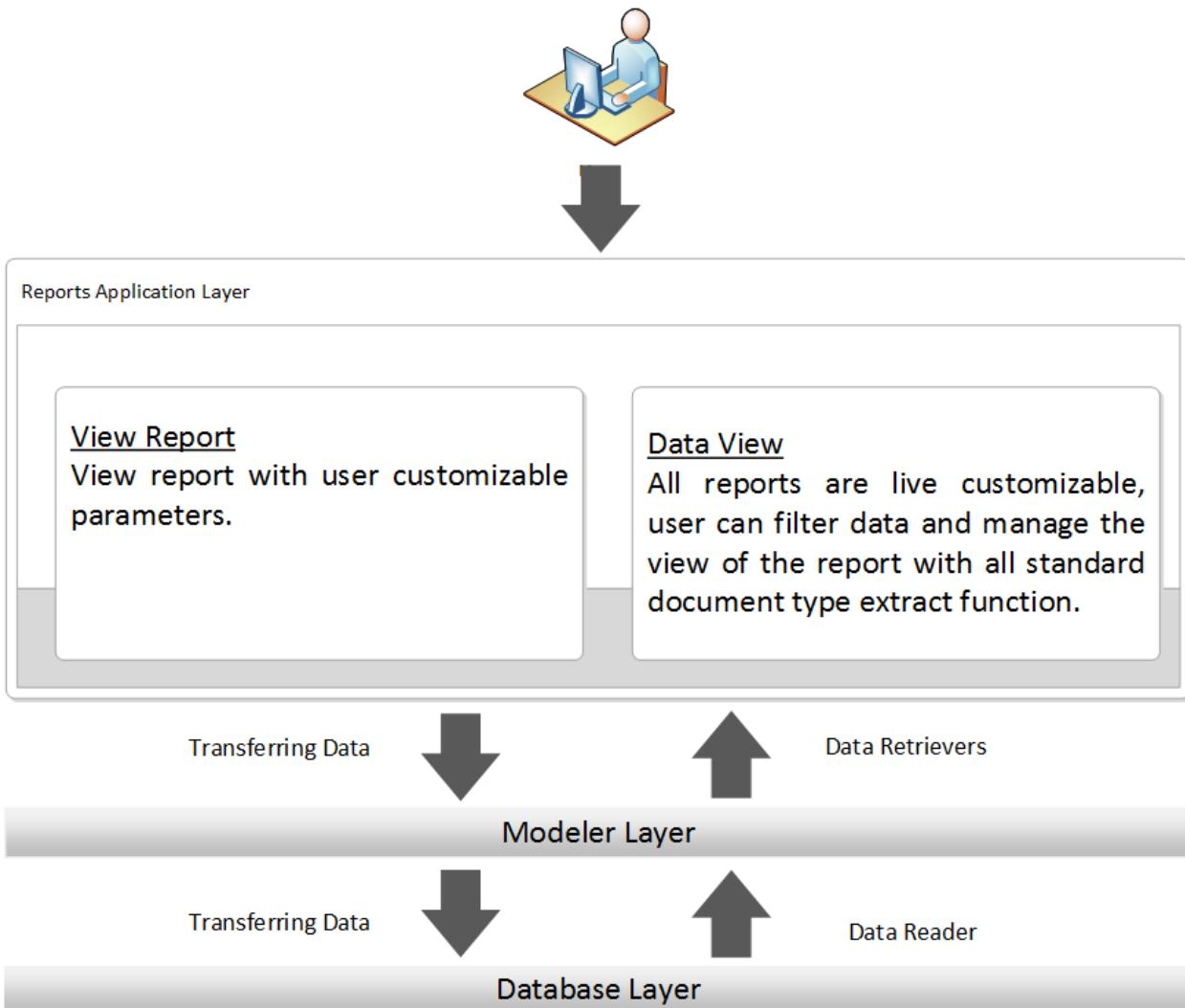


Figure 22 Reports

User can view the reports about inventory and sales. All function are automated and user can view the live details about the sales and inventory.

All reports are managed by the user levels and specific user levels only can access to the report.

Reports are less but, one report module can make lot of report outputs by customizing the report.

4.2.2.12 Module 12 - Reports - Sales

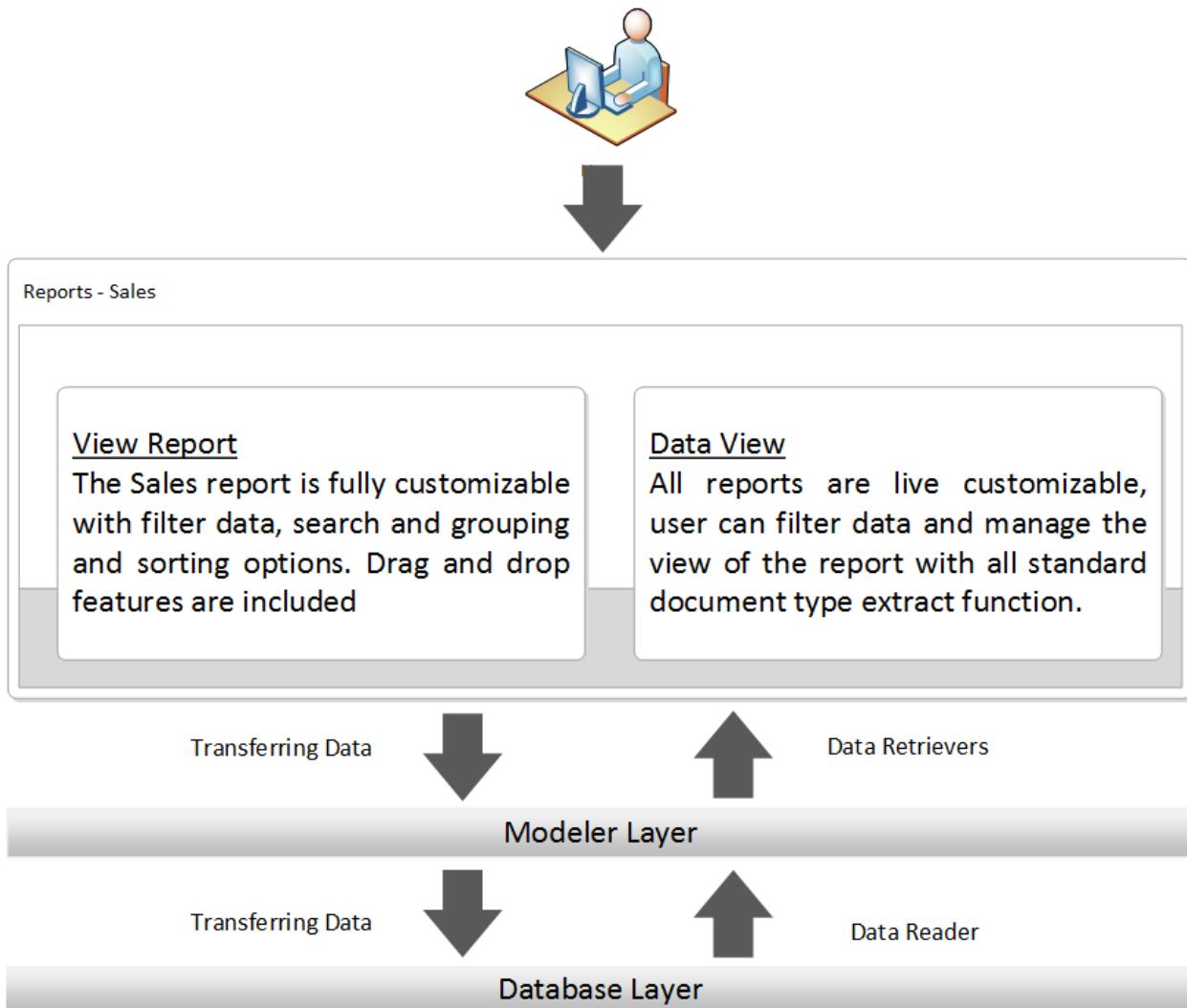


Figure 23 Reports – Sales

Authorized user can view the sales report and the user can generate all sales report types.

As example –

1. Today Sales
2. Monthly Sales
3. Weekly sales
4. Yearly sales
5. Profit of the current sales criteria

4.2.2.13 Module 13 - Reports - Inventory

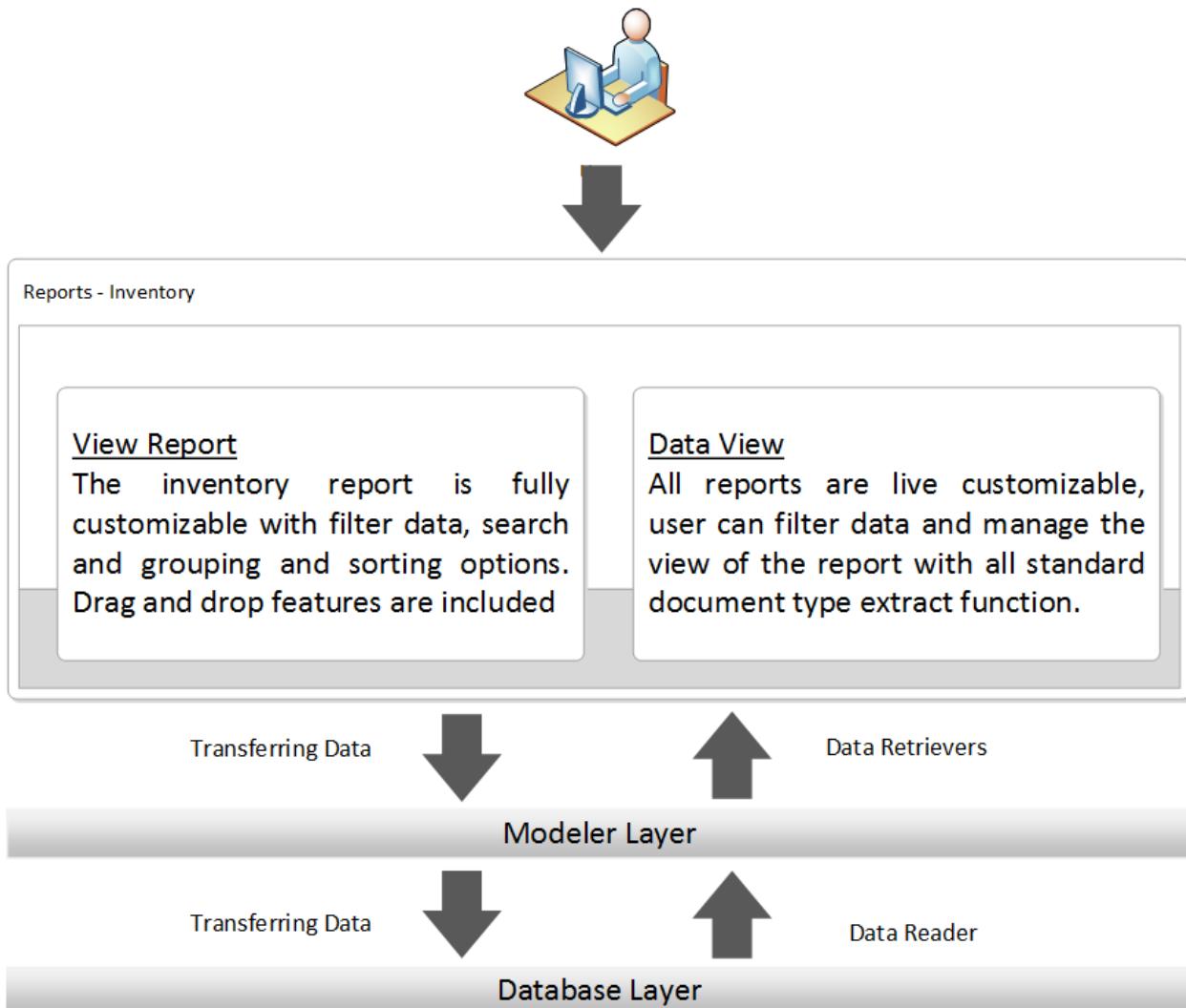


Figure 24 Reports - Inventory

Authorized user can view the inventory report and the user can generate all sales report types.

As example –

1. All items
2. Stock details
3. Quantities
4. Quantity ranges
5. Expiries
6. Items in Re order level

4.2.3 Overall Mobile Application



Figure 25 Sample Mobile Application Dashboard Design

By the Mobile application user can review the business statuses from the mobile application dashboard the application.

The mobile application is contain with only one dashboard and the design layout is flat metro stile, the dashboard is similar to the windows application dashboard.

4.3 Data Design

4.3.1 Entity Relationship Diagram

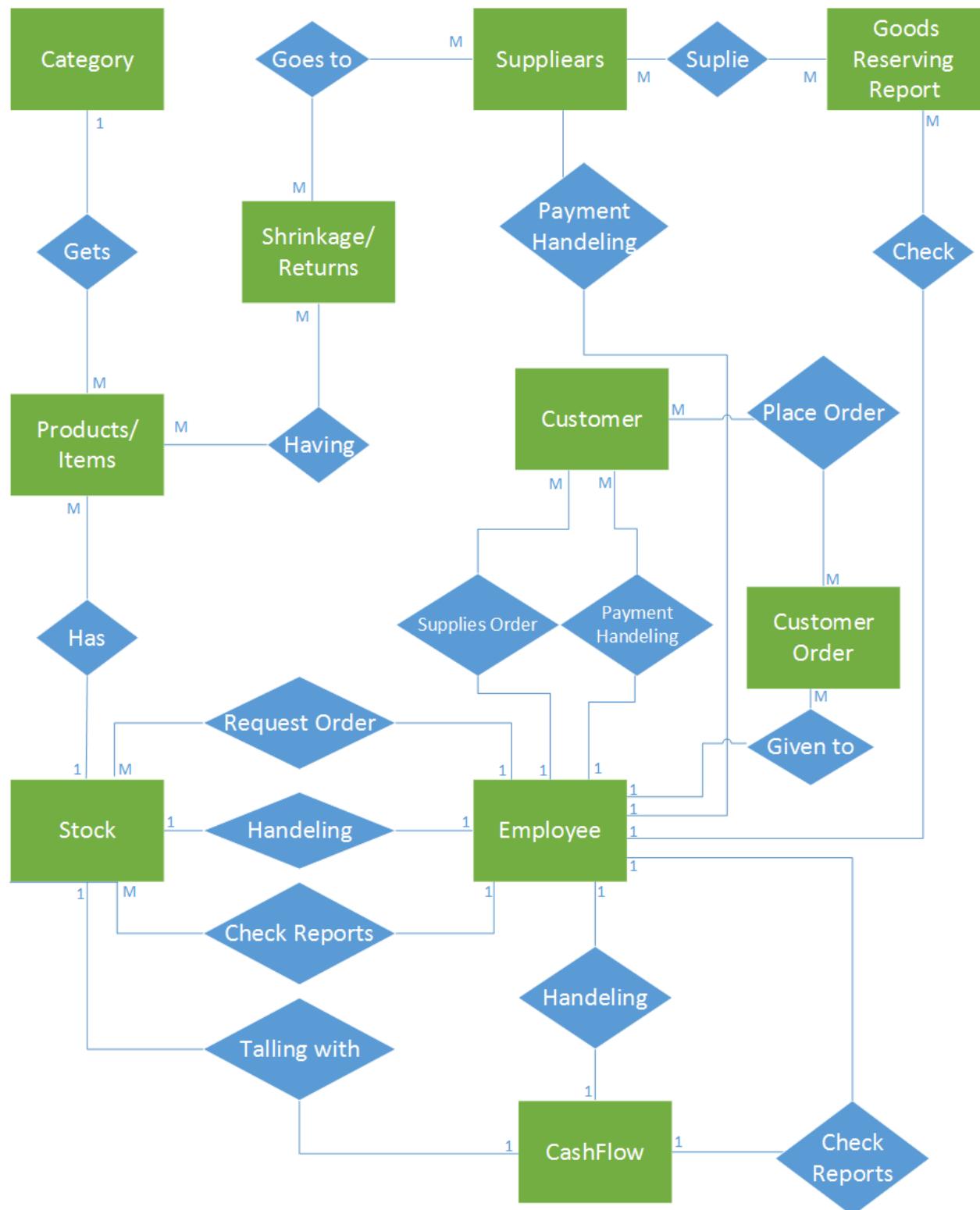


Figure 26 Entity Relationship Diagram

4.3.2 Mapping of Logical Database to Relations

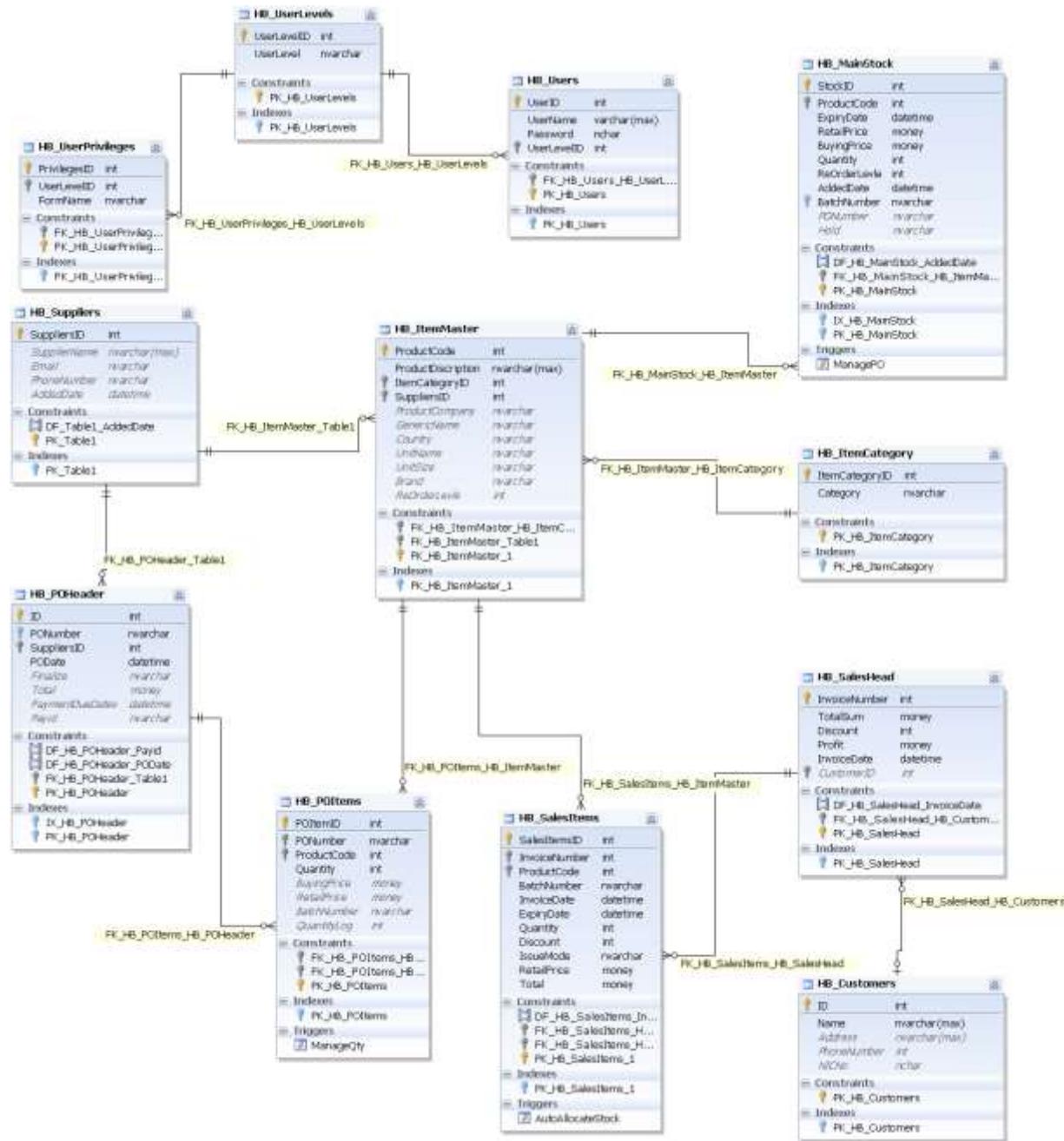


Figure 27 Mapping of Logical Database to Relations

4.4 Interface Design of Windows Application

Introduction

The main layout is designed with Microsoft Metro Style, and the architecture is mainly focused to flat user interfaces, it mostly increase the application performance and alleviative of the application.

The architecture is using in Windows 8 and upward with Windows Apps, and the application style is maintaining the same interface, to handle all the sub interfaces that mapped to main menu.

In the main interface the menus are created by the tiles, each tiles are programmed to access some interface and that interface is not appearing as new window, the main window is navigating to side and the required interface is presenting as a navigated interface.

Menu tiles can represent by deferent sizes and groups with many colors to user attraction and data represent purposes. The tiles sizes can be changed like



The interface is focused to touch features and design architecture is based on touch systems.

In each menu tile can be present some special data and some important notification to user.

As an example the image is explaining the whole story about metro application menu layout and all the features that can present via the metro menu. The menu layout is designed as the Dashboard concept and lots of important data retrieving methods are created.

The menu is mapping all modules with the dashboard. Users can easily navigate true the tiles to each module. After the main menu the sub interfaces are designed, to accomplish deferent tasks.

Menu is design for the easy access to the modules. Via the big tiles with customizable tile sizes user can easily find the needed module to access. And user can view current statues of the module true the tile view.

4.4.1 Application Menu

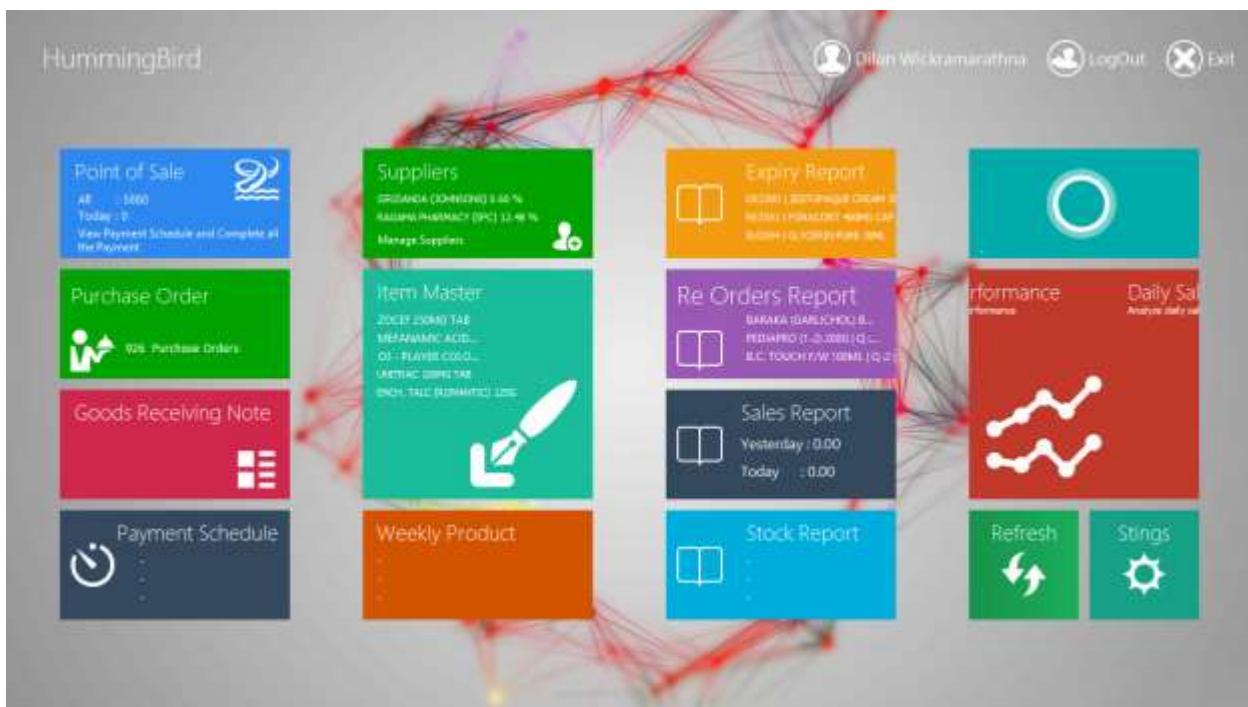


Figure 28 Interface Menu

As above, the application menu is created to the access to each items that In all sub interfaces, user can see the back to Home button to return menu and close button, those are common function and default to all interfaces.

4.4.1 Login to the system

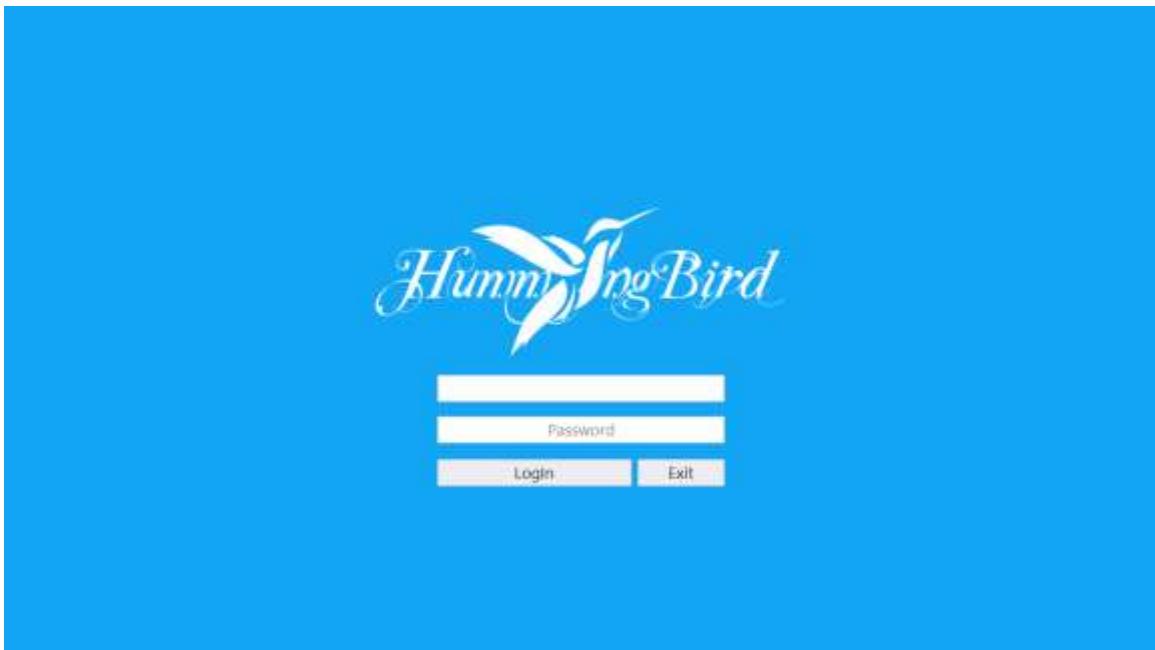


Figure 29 Login screen

444.2 Point of Sales

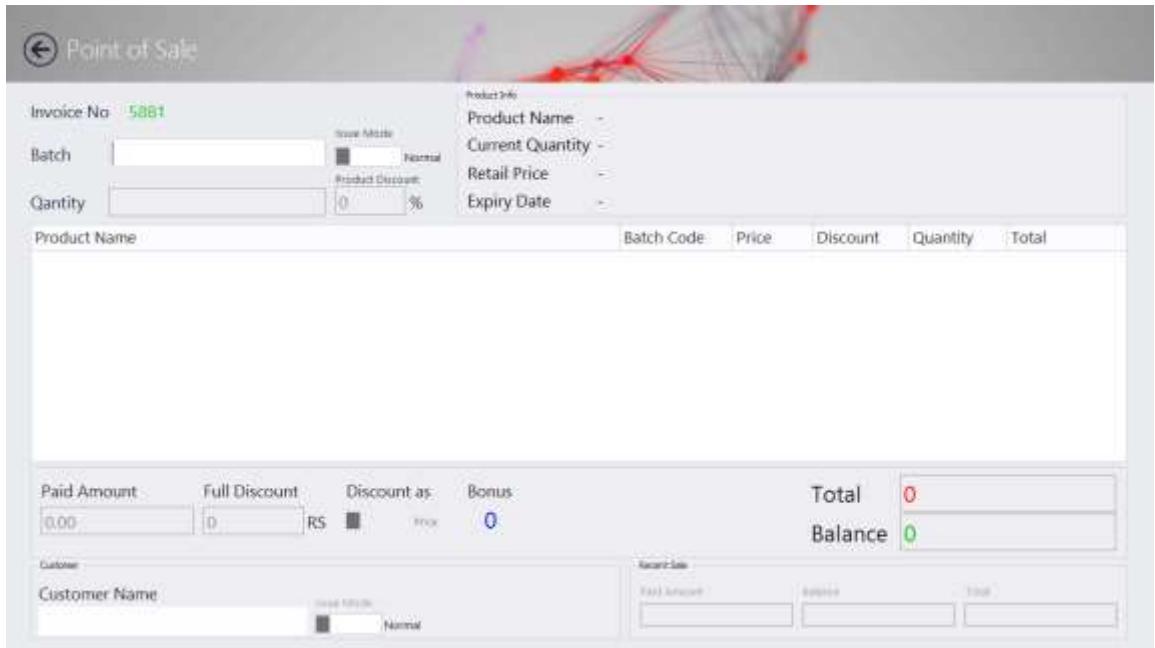
The image shows the Point of Sales (POS) interface. The title bar says "Point of Sale". The main area has several sections: "Invoice No." with value "5881", "Batch" dropdown, "Quantity" input field with value "0", and "Product Info" dropdowns for "Product Name", "Current Quantity", "Retail Price", and "Expiry Date". Below this is a grid table with columns "Product Name", "Batch Code", "Price", "Discount", "Quantity", and "Total". Further down are sections for "Paid Amount" (0.00), "Full Discount" (0 RS), "Discount as" (0%), "Bonus" (0), "Total" (0), and "Balance" (0). At the bottom, there are sections for "Customer" and "Sales Tax".

Figure 30 Point of Sales

3.4.3 Purchase Order

Purchase Order

Suppliers	Order Date	Product Description	Quantity	PO Date	Supplier Name
	11/02/2014	N.S. MULTI VIT. BODY LOTI. PACK 50ML	1	6/17/2014	NATURE SECRETS (MANJULA D...
Product		BECLOMATE 400MCG CAP	400	6/28/2014	RAGAMA PHARMACY (SPC)
Quantity		ENCH. COLO (GLAMOUR) 120ML	2	7/4/2014	ENCHANTUR
PO Number	PO-3...	AMITONE SYRUP - 100ML	1		
		FREESTYLE STRIP	6	7/9/2014	ISHARA PHARMACY
		ENSURE VANILLA 400G	1		
		PO Number: PO-4129			
		CELOX R. 200MG CAP	50	7/9/2014	ISHARA PHARMACY
		BETNOVATE CREAM 10G	3	7/14/2014	RAGAMA PHARMACY (SPC)
		SATHUTA CONDOM	40	7/18/2014	MEDKIN - PANADURA
		VIT B-12 INJECTION 10ML	999	7/25/2014	RAGAMA PHARMACY (SPC)
		EHAMONIN 375MG TAB	3	7/26/2014	MEDIHOUSE - PANNIPITIYA
		PO Number: PO-4135			
		N.S. UNDER EYE CREAM 15ML			
		N.S. (TURMERIC GOLD) CREAM 50ML			
		N.S. MULTI VIT. FACE CREAM 50ML			
		PO Number: PO-4191			

New Save Record 0 of 0

Figure 31 Purchase Order

4.4.4 Goods Reserving Notes

Goods Receiving Note

PO ID	Suppliers	Order Date	Product Description	Quantity	PO Date	Supplier Name
			N.S. MULTI VIT. BODY LOTI. PACK 50ML	1	6/17/2014	NATURE SECRETS (MANJULA D...
			BECLOMATE 400MCG CAP	400	6/28/2014	RAGAMA PHARMACY (SPC)
			ENCH. COLO (GLAMOUR) 120ML	2	7/4/2014	ENCHANTUR
			AMITONE SYRUP - 100ML	1		
			FREESTYLE STRIP	6	7/9/2014	ISHARA PHARMACY
			ENSURE VANILLA 400G	1		
			PO Number: PO-4129			
			CELOX R. 200MG CAP	50	7/9/2014	ISHARA PHARMACY
			BETNOVATE CREAM 10G	3	7/14/2014	RAGAMA PHARMACY (SPC)
			SATHUTA CONDOM	40	7/18/2014	MEDKIN - PANADURA
			VIT B-12 INJECTION 10ML	999	7/25/2014	RAGAMA PHARMACY (SPC)
			EHAMONIN 375MG TAB	3	7/26/2014	MEDIHOUSE - PANNIPITIYA
			PO Number: PO-4135			
			N.S. UNDER EYE CREAM 15ML			
			N.S. (TURMERIC GOLD) CREAM 50ML			
			N.S. MULTI VIT. FACE CREAM 50ML			
			PO Number: PO-4191			

Add Cancel Record 0 of 0

Product Name Batch Expiry Date Re Order Quantity Retail Price Buying Price Hold

Batch Number Expiry Date Retail Price Buying Price Hold

Quantity Reorder Level

Item Close Item Credit Note Total Amount 0 RS Cancel Proceed

Figure 32 Goods Reserving Notes

4.4.5 Controles – Item Controle

Item Category - Category Name :

	Save	Clear	Delete																								
<input type="checkbox"/> Suppliers																											
<input type="checkbox"/> Customers																											
<div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> <input style="width: 100%; height: 25px; border: none; outline: none; font-size: 10pt; padding: 2px; margin-bottom: 2px;" type="text"/> Find Clear </div> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="padding: 2px;">Category</th> </tr> </thead> <tbody> <tr><td style="padding: 2px;">ANTHELMINTICS</td></tr> <tr><td style="padding: 2px;">ANTIBIOTICS</td></tr> <tr><td style="padding: 2px;">BABY CARE</td></tr> <tr><td style="padding: 2px;">CNS</td></tr> <tr><td style="padding: 2px;">CONTRACEPTIVE</td></tr> <tr><td style="padding: 2px;">COSMETICS</td></tr> <tr><td style="padding: 2px;">CREAM & OINMENT</td></tr> <tr><td style="padding: 2px;">DIABETICS</td></tr> <tr><td style="padding: 2px;">EYE/EAR/NEASOL DROPS</td></tr> <tr><td style="padding: 2px;">FUNGUS</td></tr> <tr><td style="padding: 2px;">GLT</td></tr> <tr><td style="padding: 2px;">GASTRO</td></tr> <tr><td style="padding: 2px;">GENARAL</td></tr> <tr><td style="padding: 2px;">HEALTH CARE</td></tr> <tr><td style="padding: 2px;">HOMONS</td></tr> <tr><td style="padding: 2px;">HYPERTENTION DRUGS</td></tr> <tr><td style="padding: 2px;">INHALERS</td></tr> <tr><td style="padding: 2px;">INJECTION & INPURIM</td></tr> <tr><td style="padding: 2px;">LIPID LOWERING</td></tr> <tr><td style="padding: 2px;">MILK POWDER</td></tr> <tr><td style="padding: 2px;">MOUTH WASH</td></tr> <tr><td style="padding: 2px;">NUTRITION SUPPLEMENT</td></tr> <tr><td style="padding: 2px;">PAIN KILLERS</td></tr> </tbody> </table>				Category	ANTHELMINTICS	ANTIBIOTICS	BABY CARE	CNS	CONTRACEPTIVE	COSMETICS	CREAM & OINMENT	DIABETICS	EYE/EAR/NEASOL DROPS	FUNGUS	GLT	GASTRO	GENARAL	HEALTH CARE	HOMONS	HYPERTENTION DRUGS	INHALERS	INJECTION & INPURIM	LIPID LOWERING	MILK POWDER	MOUTH WASH	NUTRITION SUPPLEMENT	PAIN KILLERS
Category																											
ANTHELMINTICS																											
ANTIBIOTICS																											
BABY CARE																											
CNS																											
CONTRACEPTIVE																											
COSMETICS																											
CREAM & OINMENT																											
DIABETICS																											
EYE/EAR/NEASOL DROPS																											
FUNGUS																											
GLT																											
GASTRO																											
GENARAL																											
HEALTH CARE																											
HOMONS																											
HYPERTENTION DRUGS																											
INHALERS																											
INJECTION & INPURIM																											
LIPID LOWERING																											
MILK POWDER																											
MOUTH WASH																											
NUTRITION SUPPLEMENT																											
PAIN KILLERS																											

Figure 33 Controls – Item controls

4.4.5 Controles – Suplier Controle

Figure 34 Controls – Supplier controls

4.4.6 Controles - Customer Controle

The screenshot shows a user interface titled "Product Controls". On the left, there is a sidebar with sections for "Item Category", "Customer Name", "Suppliers", "Address", "Customers", "Phone Number", and "NIC Number". Below these are "Save", "Clear", and "Delete" buttons. On the right, there is a search bar with placeholder text "Enter what to search...". Below the search bar is a table with columns: "Name", "Address", "Phone Number", and "NIC Number". A row in the table has the text "[Open Customer]".

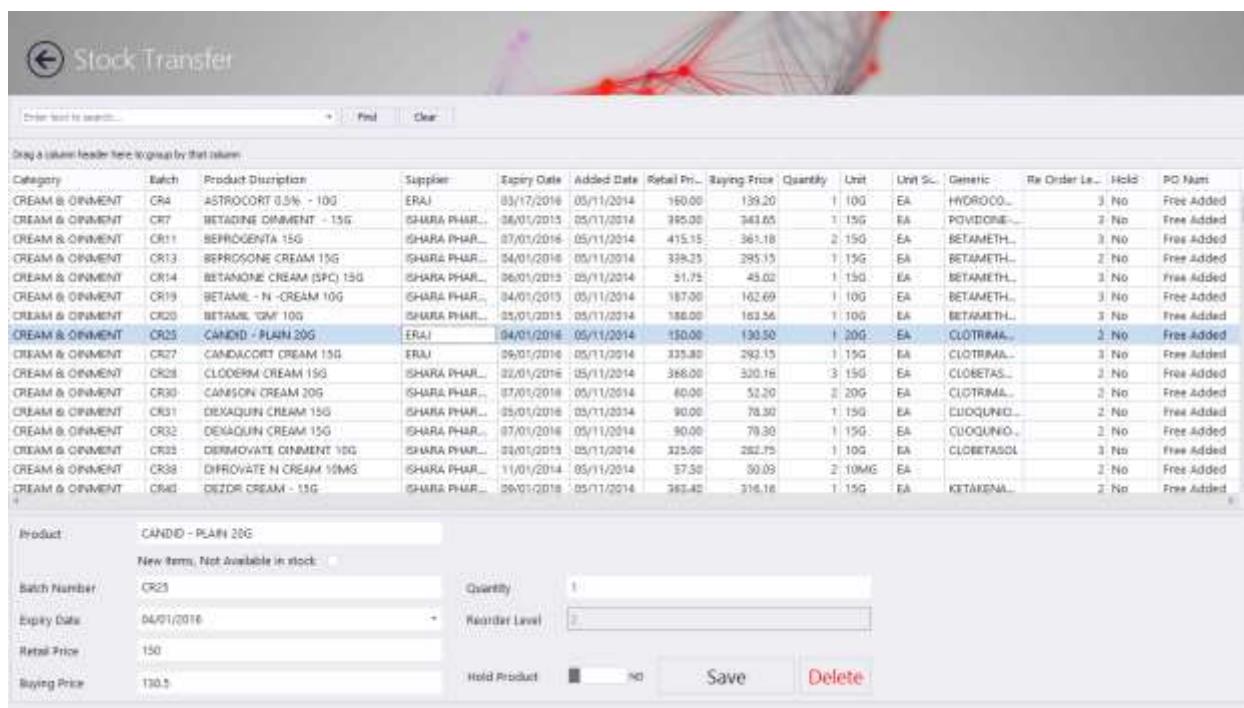
Figure 35 Controls – User controls

4.4.7 Item Master

The screenshot shows a user interface titled "Item Master". On the left, there are several dropdown menus and input fields: "Prod. Description" (set to "4EVER (TEA TREE OIL) F/W 50ML"), "Supplier" (set to "4-REVER SIN NATURALS"), "Company" (empty), "Generic Name" (empty), "Country" (empty), "Unit Name" (set to "50ML"), "Unit Size" (set to "EA"), "Brand" (empty), "Item Category" (set to "COSMETICS"), and "Reorder Level" (set to "3"). Below these is a table with columns: "Product Description", "Category", "Supplier", "Company", "Generic", "Country", "Unit", "Unit Size", and "Brand". The table lists various cosmetic items such as "ALMOND NAIL LACQUER REMOVER", "TEA TREE FACE WASH 50ML", "VENIVNAL FACE WASH 100ML", etc. At the bottom of the table are "New", "Save", and "Delete" buttons.

Figure 36 Item Master

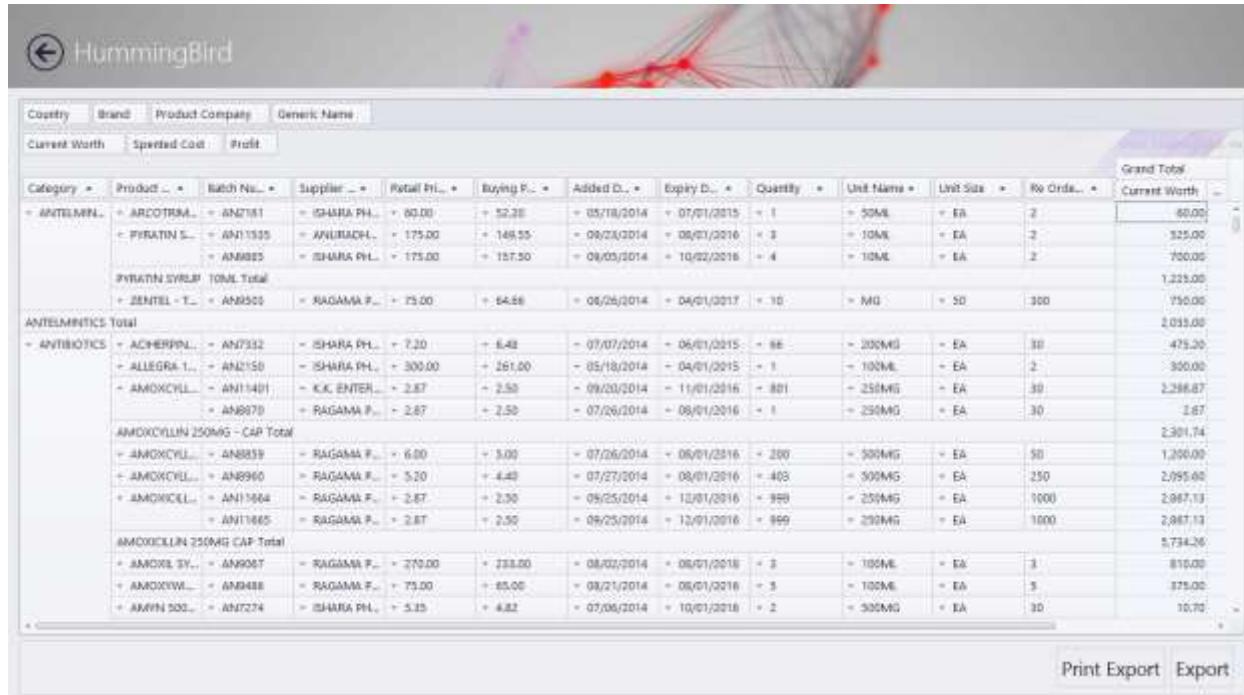
4.4.8 Stock Transfer

A screenshot of a software application titled "Stock Transfer". The interface includes a search bar at the top with fields for "Enter Text to search...", "Find", and "Clear". Below the search bar is a table header with columns: Category, Batch, Product Description, Supplier, Expiry Date, Added Date, Retail Pr..., Buying Price, Quantity, Unit, Unit Size, Generic, Re Order Le..., Hold, and PO Num. The main body of the table lists various pharmaceutical products with their details. At the bottom of the form, there is a section for entering new stock transfer details: Product (CANDID - PLAIN 20G), Quantity (1), Expiry Date (04/01/2016), and a placeholder for Reorder Level. There are also fields for Retail Price (150), Buying Price (130.5), Hold Product (checkbox), and buttons for Save and Delete.

Category	Batch	Product Description	Supplier	Expiry Date	Added Date	Retail Pr...	Buying Price	Quantity	Unit	Unit Size	Generic	Re Order Le...	Hold	PO Num
CREAM & OINTMENT	CR4	ASTROCORT 0.5% - 10G	ERAJI	03/17/2016	05/11/2014	160.00	139.20	1	100	EA	HYDROCORTISONE	3	No	Free Added
CREAM & OINTMENT	CR7	BETADINE OINTMENT - 15G	ISHARA PHAR...	06/01/2013	05/11/2014	395.00	343.65	1	15G	EA	POVIDONE-IODINE	2	No	Free Added
CREAM & OINTMENT	CR11	SEPROGENATE 15G	ISHARA PHAR...	07/01/2016	05/11/2014	415.15	361.18	2	15G	EA	BETAMETHASONE	3	No	Free Added
CREAM & OINTMENT	CR13	BEPROSODINE CREAM 15G	ISHARA PHAR...	04/01/2016	05/11/2014	338.25	295.15	1	15G	EA	BEPROSODINE	2	No	Free Added
CREAM & OINTMENT	CR14	BETAMONE CREAM (SPC) 15G	ISHARA PHAR...	06/01/2013	05/11/2014	31.75	28.02	1	15G	EA	BETAMETHASONE	3	No	Free Added
CREAM & OINTMENT	CR19	BETAMOL - N-CREAM 10G	ISHARA PHAR...	04/01/2013	05/11/2014	187.00	162.69	1	100	EA	BETAMETHASONE	3	No	Free Added
CREAM & OINTMENT	CR20	BETAMOL GM 10G	ISHARA PHAR...	04/01/2013	05/11/2014	186.00	162.56	1	100	EA	BETAMETHASONE	3	No	Free Added
CREAM & OINTMENT	CR25	CANDID - PLAIN 20G	ERAJI	04/01/2016	05/11/2014	150.00	130.50	1	200	EA	CLOTRIMAZOLE	2	No	Free Added
CREAM & OINTMENT	CR27	CANDICARE CREAM 15G	ERAJI	09/01/2016	05/11/2014	335.80	292.15	1	15G	EA	CLOTRIMAZOLE	3	No	Free Added
CREAM & OINTMENT	CR28	CLODERM CREAM 15G	ISHARA PHAR...	02/01/2016	05/11/2014	368.00	320.16	3	15G	EA	CLORETASAL	2	No	Free Added
CREAM & OINTMENT	CR30	CANSON CREAM 20G	ISHARA PHAR...	07/01/2016	05/11/2014	60.00	52.20	2	20G	EA	CLOTRIMAZOLE	2	No	Free Added
CREAM & OINTMENT	CR31	DEXAQUIN CREAM 15G	ISHARA PHAR...	05/01/2016	05/11/2014	90.00	78.30	1	15G	EA	CLOQUINOL	2	No	Free Added
CREAM & OINTMENT	CR32	DEXAQUIN CREAM 15G	ISHARA PHAR...	07/01/2016	05/11/2014	90.00	78.30	1	15G	EA	CLOQUINOL	2	No	Free Added
CREAM & OINTMENT	CR33	DERMOVATE OINTMENT 10G	ISHARA PHAR...	04/01/2013	05/11/2014	325.00	282.75	1	100	EA	CLOBETASOL	3	No	Free Added
CREAM & OINTMENT	CR38	DIPROVATE N CREAM 10MG	ISHARA PHAR...	11/01/2014	05/11/2014	57.50	50.09	2	10MG	EA	DIPROVATE	2	No	Free Added
CREAM & OINTMENT	CR40	DEZOR CREAM - 15G	ISHARA PHAR...	06/01/2016	05/11/2014	363.40	316.18	1	15G	EA	KETAMENOL	2	No	Free Added

Figure 37 Stock Transfer

4.4.9 Reports - Stock

A screenshot of a software application titled "HummingBird". The interface includes a search bar at the top with fields for "Country", "Brand", "Product Company", and "Generic Name". Below the search bar is a table showing inventory details. The table has columns: Category, Product, Batch No., Supplier, Retail Pr..., Buying Pr..., Added D..., Expiry D..., Quantity, Unit Name, Unit Size, Re Order L..., Current Worth, and Grand Total. The table lists various pharmaceutical products with their details. At the bottom of the form, there are buttons for Print, Export, and Export.

Category	Product	Batch No.	Supplier	Retail Pr...	Buying Pr...	Added D...	Expiry D...	Quantity	Unit Name	Unit Size	Re Order L...	Current Worth	Grand Total
- ANTELMINTS	+ ARCOFRM...	- AN2181	- ISHARA PH...	+ 60.00	+ 52.3E	+ 05/18/2014	+ 07/01/2015	+ 1	50ML	+ EA	2	60.00	
	+ PIRATIN S...	- AN1335	- ANIRUOHL...	+ 175.00	+ 148.55	+ 09/23/2014	+ 08/01/2016	+ 3	10ML	+ EA	2	525.00	
	+ AM8835	- ISHARA PH...	+ 175.00	+ 157.50	+ 08/03/2014	+ 10/22/2016	+ 4	10ML	+ EA	2	700.00		
	PIRATIN SYRUP TOTAL											1225.00	
	+ ZENTEL T...	- AN9500	- RAGAMA P...	+ 75.00	+ 64.6E	+ 06/26/2014	+ 04/01/2017	+ 10	MD	+ 10	300	750.00	
ANTELMINTICS TOTAL												2015.00	
- ANTI-BIOTICS	+ ACHEPPNL...	- AN1732	- ISHARA PH...	+ 7.20	+ 6.4E	+ 07/07/2014	+ 06/01/2015	+ 66	200MG	+ EA	30	475.20	
	+ ALLEGRA 1...	- AN1215B	- ISHARA PH...	+ 300.00	+ 261.00	+ 05/18/2014	+ 04/01/2015	+ 1	100ML	+ EA	2	300.00	
	+ AMOXICYL...	- AN1140H	- KIC ENTER...	+ 2.67	+ 2.50	+ 09/20/2014	+ 11/01/2016	+ 801	250MG	+ EA	30	2,266.67	
	+ AN8870	- RAGAMA P...	+ 2.67	+ 2.50	+ 07/26/2014	+ 08/01/2016	+ 1	200MG	+ EA	30	1.67		
	AMOXICILLIN 250MG CAP Total											2,301.74	
	+ AMOXCS...	- AN8859	- RAGAMA P...	+ 6.00	+ 5.00	+ 07/26/2014	+ 06/01/2016	+ 200	200MG	+ EA	50	1,200.00	
	+ AMOXCYL...	- AN8960	- RAGAMA P...	+ 5.20	+ 4.4E	+ 07/27/2014	+ 06/01/2016	+ 403	300MG	+ EA	250	2,095.00	
	+ AMOXICELL...	- AN11664	- RAGAMA P...	+ 2.67	+ 2.30	+ 09/25/2014	+ 12/01/2016	+ 959	250MG	+ EA	1000	2,987.13	
	+ AN11665	- RAGAMA P...	+ 2.67	+ 2.30	+ 09/25/2014	+ 12/01/2016	+ 959	200MG	+ EA	1000	2,867.13		
	AMOXICILLIN 250MG CAP Total											5,734.26	
	+ AMOXIV...	- AN9087	- RAGAMA P...	+ 270.00	+ 233.00	+ 08/02/2014	+ 06/01/2016	+ 3	100ML	+ EA	3	810.00	
	+ AMOXIVM...	- AN9488	- RAGAMA P...	+ 75.00	+ 65.00	+ 03/21/2014	+ 06/01/2016	+ 3	100ML	+ EA	3	225.00	
	+ AMIN 500...	- AN7274	- ISHARA PH...	+ 5.25	+ 4.8E	+ 07/06/2014	+ 10/01/2016	+ 2	300MG	+ EA	30	10.75	

Figure 38 Reports - Inventory

4.4.10 Reports - Sales

Sales Report Filter View											Print Export	
Date	Total Sum	Invoice Number	Product Description	Batch No.	Issue Month	Quantity	Discount	Retail Price	Final Invoice Discount	Total Total		
- 7/2/2014	+ 20.66	+ 676	+ POLYBON INJECTION 2ML	-	+ N	+ 1	-	+ 30.66		30.66		
+ 51.17	+ 680	+ GENTALAB EYE/EAR DROPS	-	+ N	+ 1	-	-	+ 51.17		51.17		
+ 59.00	+ 677	+ VICKS - INHALER (0.5)	-	+ N	+ 1	-	-	+ 59.00		59.00		
+ 85.00	+ 659	+ CREAM BANDAGE	-	+ N	+ 1	-	-	+ 85.00		85.00		
		+ Dettol Plaster	-	+ N	+ 2	-	-	+ 5.00		10.00		
		659 Total								95.00		
+ 96.00	+ 670	+ SOFRAMYCIN CREAM 20G	-	+ N	+ 1	-	-	+ 96.00		96.00		
+ 120.00	+ 668	+ CHANDANA LEERA 12G	-	+ N	+ 1	-	-	+ 120.00		120.00		
+ 138.60	+ 664	+ BECLOMIN OINTMENT 15G	-	+ N	+ 1	-	-	+ 138.60		138.60		
+ 150.00	+ 653	+ J.H. BABY COLO (MOR. DEW.)	-	+ N	+ 1	-	-	+ 150.00		150.00		
+ 160.00	+ 674	+ MODOS (SUPREME DOT)	-	+ N	+ 1	-	-	+ 160.00		160.00		
+ 179.10	+ 678	+ FERROUS SULFATE	-	+ N	+ 10	-	-	+ 0.86		8.60		
		+ FOLIC ACID - 1MG	-	+ N	+ 10	-	-	+ 0.33		2.30		
		+ KALZANIA TAB 500MG	-	+ N	+ 10	-	-	+ 9.50		95.00		
		+ SILVER CREAM 25G	-	+ N	+ 1	-	-	+ 65.00		65.00		
		+ VITAMIN C - (SP) 100MG	-	+ N	+ 10	-	-	+ 0.82		8.20		
		678 Total								179.10		
+ 256.89	+ 672	+ ASTHALIN TDP 400MG CAP	-	+ N	+ 20	-	-	+ 8.25		127.00		
		+ POLYBON TAB	-	+ N	+ 21	-	-	+ 5.19		129.09		
		672 Total								256.89		

Figure 39 Reports - Sales

4.4.11 Reports - Expiree

Country	Unit Name	Unit Size	Brand	Product Company	Generic Name	Expiry Date	Added Date	Retail Price	Billing Price	Quantity	Re Order	Supplier Name
+ 2014	+ December	+ ANTIBIOTICS	+ PARACETAMOL 500MG TAB	+ AN1294	+ 12/12/2014	+ 09/09/2014	+ 9.26	+ 9.22	+ 10	+ 300	+ ISHARA PHARMACY	
		+ COSMETICS	+ CLE (APPLE) PFM 30ML	+ CD1591	+ 12/12/2014	+ 06/06/2014	+ 255.00	+ 246.85	+ 3	+ 3	+ SRISANDA (DHEIGSON)	
		+ CLE (BERRY) PFM 30ML	+ CD1598	+ 12/12/2014	+ 06/06/2014	+ 355.00	+ 246.85	+ 1	+ 2	+ SRISANDA (DHEIGSON)		
		+ ENOL STICK OEO (KULLURU) 25ML	+ CD7117	+ 12/12/2014	+ 07/04/2014	+ 295.00	+ 200.75	+ 1	+ 2	+ ENCHANTLUR		
		+ ENOL STICK OEO (CHARMING) 25ML	+ CD7112	+ 12/12/2014	+ 07/04/2014	+ 295.00	+ 208.75	+ 1	+ 2	+ ENCHANTLUR		
		+ ENOL STICK OEO (CHARMINTO) 25ML	+ CD7113	+ 12/12/2014	+ 07/04/2014	+ 295.00	+ 238.75	+ 1	+ 2	+ ENCHANTLUR		
		+ ZESTOPAGUE CREAM 20G	+ CR3300	+ 12/12/2014	+ 06/05/2014	+ 322.00	+ 266.00	+ 2	+ 3	+ ISHARA PHARMACY		
		+ EYE CARE	+ REFRESH LIQUID 15ML	+ SY3338	+ 12/12/2014	+ 07/25/2014	+ 557.00	+ 464.59	+ 1	+ 2	+ ISHARA PHARMACY	
		+ FACE POWDER	+ HELOLAH LIQUID 20ML	+ H69157	+ 12/12/2014	+ 08/06/2014	+ 40.00	+ 36.40	+ 7	+ 10	+ NESTLE (HARSHAD DEF.)	
		+ PAIN KILLERS	+ CELECOXIB 200MG CAP (SPC)	+ PA7385	+ 12/12/2014	+ 07/07/2014	+ 6.66	+ 4.14	+ 52	+ 10	+ ISHARA PHARMACY	
		+ RESPIRATORI.	+ CEPTAL 10MG TAB	+ RE7228	+ 12/12/2014	+ 07/06/2014	+ 9.66	+ 8.89	+ 20	+ 30	+ ISHARA PHARMACY	
		+ CLAVAMOX 250MG TAB	+ RE7299	+ 12/12/2014	+ 07/06/2014	+ 41.00	+ 36.90	+ 3	+ 30	+ ISHARA PHARMACY		
		+ POMACORT 400MG CAP	+ RE3251	+ 12/12/2014	+ 03/06/2014	+ 18.70	+ 16.27	+ 38	+ 50	+ ISHARA PHARMACY		
		+ HEXOLON 1MG TAB	+ RE7188	+ 12/12/2014	+ 07/03/2014	+ 18.70	+ 16.34	+ 38	+ 50	+ CITY PHARMA - HILIGODDA		
		+ RE7268	+ 12/12/2014	+ 07/06/2014	+ 27.13	+ 23.42	+ 38	+ 30	+ ISHARA PHARMACY			
		+ GLYCERIN FINE 30ML	+ BU3394	+ 12/12/2014	+ 03/06/2014	+ 40.00	+ 32.30	+ 4	+ 10	+ ISHARA PHARMACY		
+ 2015	+ April	+ ANTIBIOTICS	+ ALLEGRA 100ML	+ AN1291	+ 04/12/2015	+ 03/16/2014	+ 300.00	+ 261.00	+ 1	+ 2	+ ISHARA PHARMACY	
		+ ANICE STRIP 100ML	+ AN1240	+ 04/12/2015	+ 03/26/2014	+ 220.00	+ 211.40	+ 1	+ 3	+ ISHARA PHARMACY		
		+ BLUCOP P 225MG TAB	+ AN7280	+ 04/12/2015	+ 07/06/2014	+ 5.00	+ 3.22	+ 44	+ 20	+ ISHARA PHARMACY		
		+ CLE	+ TRIFLUOPERAZINE 5MG TAB	+ CTN291	+ 04/12/2015	+ 07/07/2014	+ 0.85	+ 0.77	+ 140	+ 30	+ ISHARA PHARMACY	
		+ COSMETICS	+ EYE PARASIS CREAM 50G	+ CD1145	+ 04/12/2015	+ 07/07/2014	+ 900.00	+ 232.00	+ 1	+ 2	+ NEW GLOBAL DES - BATT...	
		+ CREAM & OIL	+ HB (COOL MENTHOL) SHAMPO. 70ML	+ CD4838	+ 04/12/2015	+ 06/15/2014	+ 170.00	+ 135.03	+ 1	+ 2	+ MAYLEYS (S.L. LALI)	
		+ ISATAME - H-CREAM 10G	+ CR18	+ 04/12/2015	+ 03/21/2014	+ 187.00	+ 163.69	+ 1	+ 2	+ ISHARA PHARMACY		
		+ BETHOMATE CREAM 50	+ CR496	+ 04/12/2015	+ 08/06/2014	+ 135.00	+ 114.00	+ 2	+ 1	+ RADAMA PHARMACY (SPC)		
		+ DERMAKLE "H" CREAM 10G	+ CR4493	+ 04/12/2015	+ 04/01/2014	+ 219.00	+ 200.00	+ 1	+ 2	+ LIMA PHARMACY (SPR ...)		

Print Export Export

Figure 40 Reports - Expiree

4.4.12 Reports – Reorder

Reorder Point:	Category	Product Description	Quantity	Re-order Level	Supplier Name
0.00	- ANTIBIOTICS	- AUGMENTIN 625MG TAB	- 0	- 50	RAGAMA PHARMACY (SPC)
		- AZASIL 500MG TAB	- 0	- 30	AJABRS & CO
		- CEF - 150 200MG TAB	- 0	- 20	ISHARA PHARMACY
		- CEFALEXIN (RAHBARY) SYRUP 100ML	- 0	- 2	RAGAMA PHARMACY (SPC)
		- CLAVAC SYRUP	- 0	- 2	ISHARA PHARMACY
		- ERYTHRO 100ML SYRUP	- 0	- 2	ISHARA PHARMACY
		- FEEDING BOTTLE (U)	- 0	- 50	J. M. GRIFFIN & SONS
		- LECIF SYRUP 100ML	- 0	- 2	ISHARA PHARMACY
		- RANDOX 100ML SYRUP	- 0	- 2	ISHARA PHARMACY
		- ZITHRIN 15ML SYRUP	- 0	- 3	LANKA PHARMACY (EMERCHEM)
	- BABY CARE	- B.C. BABY OIL 100ML	- 0	- 2	BABY CHERAMY
		- B.C. BABY TALC 100G	- 0	- 3	BABY CHERAMY
		- B.C. COLO (ANGEL FAIRY) 100ML	- 0	- 2	BABY CHERAMY
		- B.C. COLO (FLORAL) 100ML	- 0	- 2	BABY CHERAMY
		- B.C. COLO (HERBAL) 100ML	- 0	- 2	BABY CHERAMY
		- B.C. COLO (HERBAL) 50ML	- 0	- 3	BABY CHERAMY
		- B.C. COLO (BITTY) 50ML	- 0	- 3	BABY CHERAMY
		- B.C. COLO (LUCKY) 100ML	- 0	- 3	BABY CHERAMY
		- B.C. COLO (LUCKY) 50ML	- 0	- 2	BABY CHERAMY
		- B.C. COLO (NORMAL) 50ML	- 0	- 3	BABY CHERAMY
		- B.C. CREAM 50ML	- 0	- 3	BABY CHERAMY
		- B.C. MED ST. LOTION 100ML	- 0	- 3	BABY CHERAMY
		- CLODARD MOUTH WASH 200ML	- 0	- 2	BABY CHERAMY

Print Export Export

Figure 41 Reports – Reorder

4.4.13 Dashboard



Figure 42 Dashboard

4.4.14 Settings – User

The screenshot shows the 'Settings – User' interface. On the left, there is a form for creating a new user:

Name:	Account owner's name:
User Name:	User Name
Password:	User Password
Conform Password	
User Level:	Select the User Level:

Below the form is a grid view of existing users:

Name	User Name	User Level
Dilan Wickramaratne	Dilan	Administrator
User	STDUser	User

Buttons at the bottom of the form include Save, Clear, and Delete.

Figure 43 Settings – User

4.4.15 Settings – User Level

The screenshot shows the 'Settings – User Level' interface. On the left, there is a form for assigning modules to a user level:

User Level:	Available Modules and Reports
<ul style="list-style-type: none">Point of SalePurchase OrderGoods Receiving NoteItem MasterStock TransferProduct ControlsPayment ScheduleDashboardStringStock ReportReOrders ReportExpiry ReportSales Report	

Below the form is a grid view of user levels and their assigned modules:

Level Name	Form
Administrator	<ul style="list-style-type: none">PointofSalePurchaseOrderGoodsReceivingNoteItemMasterStockTransferProductControlsPaymentScheduleDashboardStringStockReportReOrdersReportExpiryReportSalesReport
User	<ul style="list-style-type: none">PointofSalePurchaseOrderGoodsReceivingNoteItemMasterStockTransferPaymentScheduleDashboard

Buttons at the bottom of the form include Save, Clear, and Delete.

Figure 44 Settings – User Level

4.4.16 Settings – Mobile Access

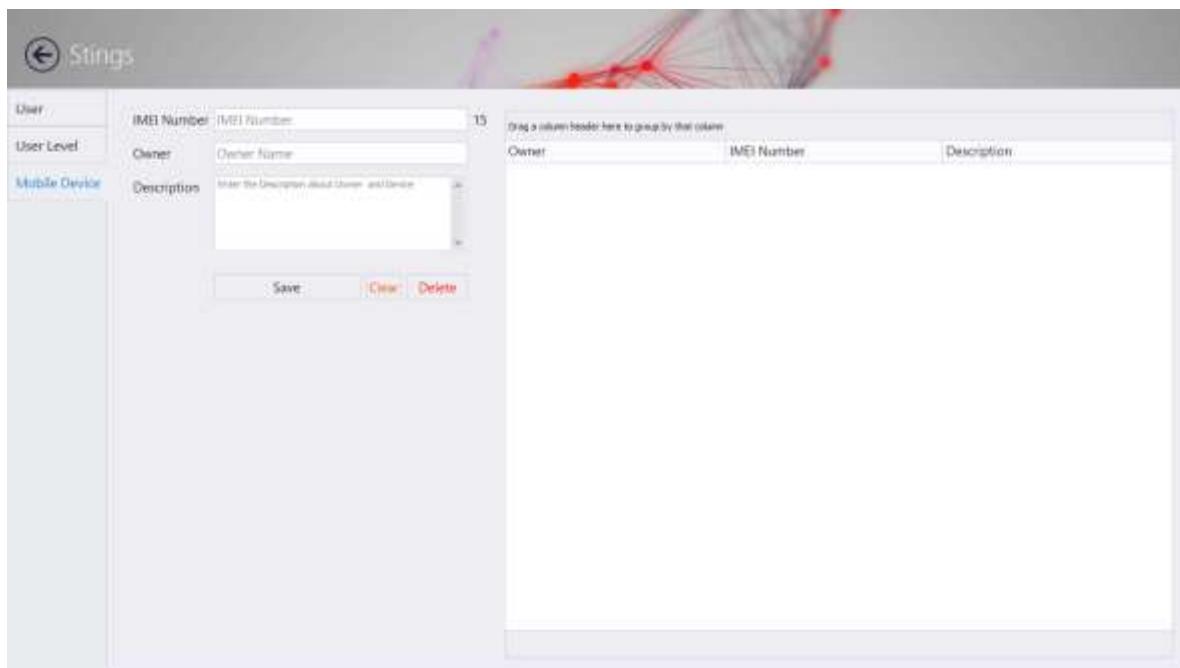


Figure 45 Settings – Mobile Access

4.4.17 Exit conformation

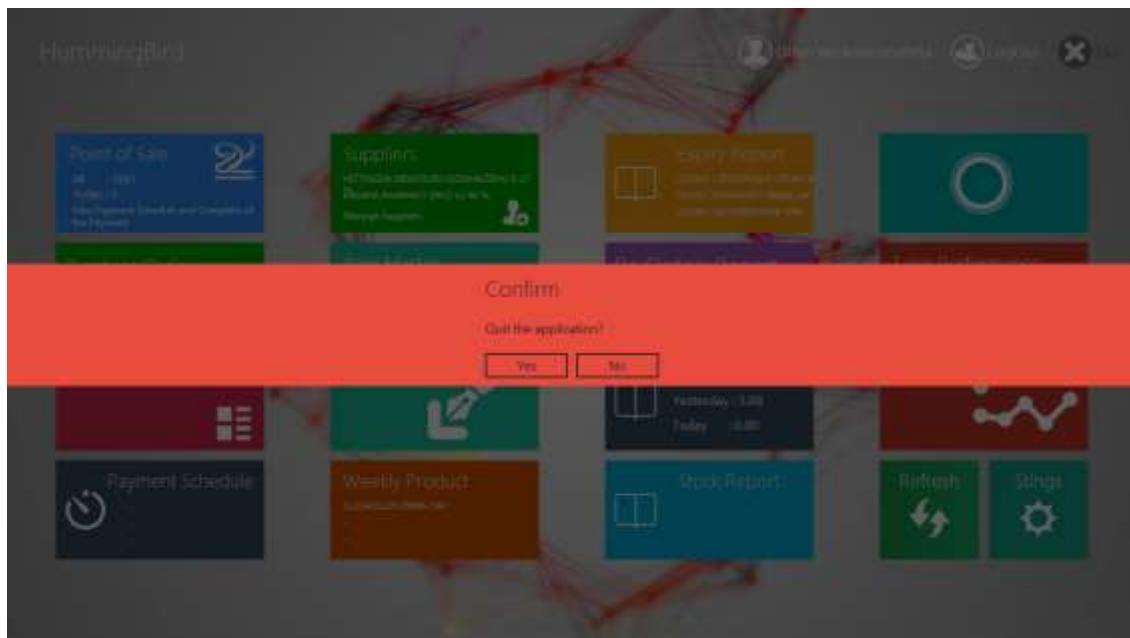


Figure 46 Exit conformation

4.5 Interface Design of Android Application

4.5.1 Login

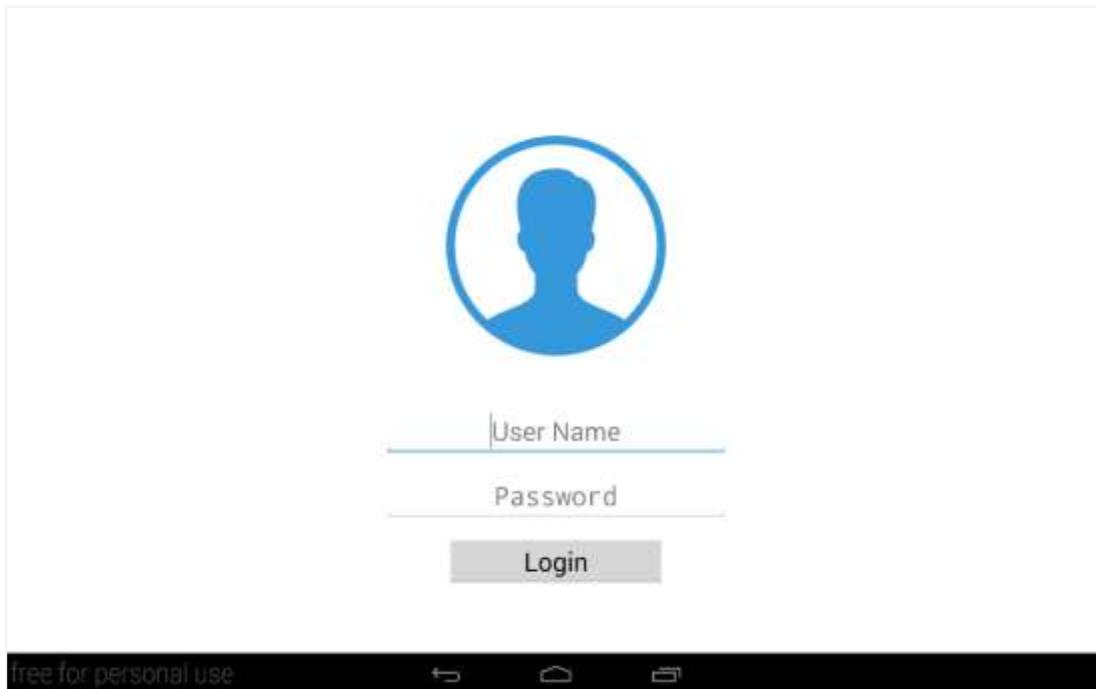


Figure 47 Android Application Login

4.5.2 Point of sales

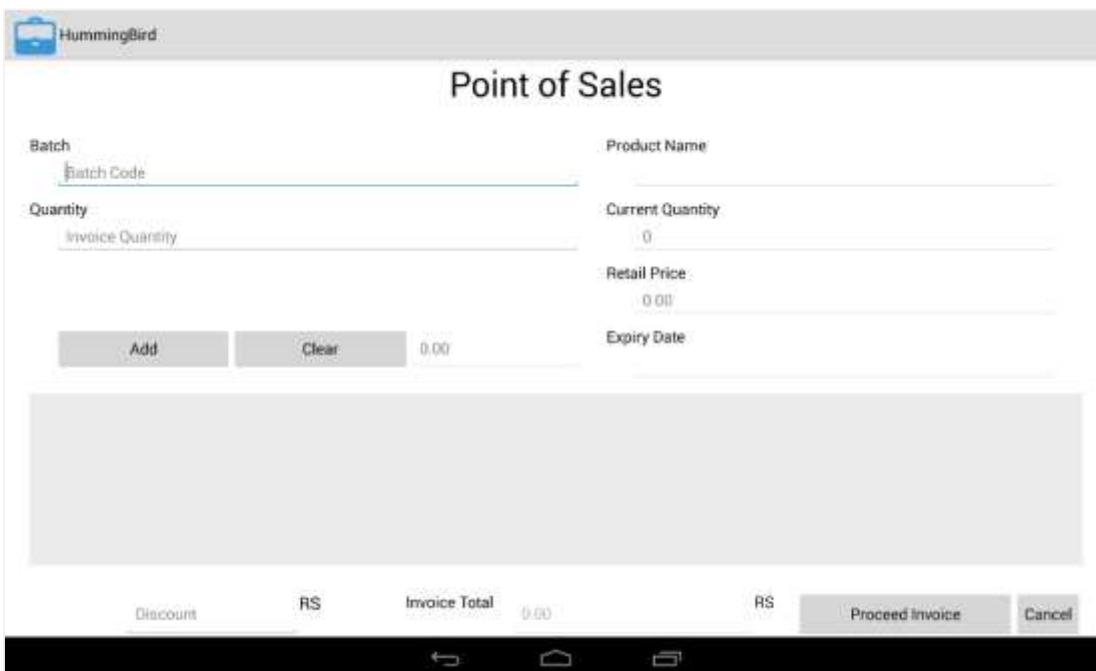


Figure 48 Android Application Point of sales window

4.6 Interface Design of Windows Mobile Application

4.6.1 Start Screen and Dashboard



Figure 50 Dashboard 1



Figure 49 Dashboard 2

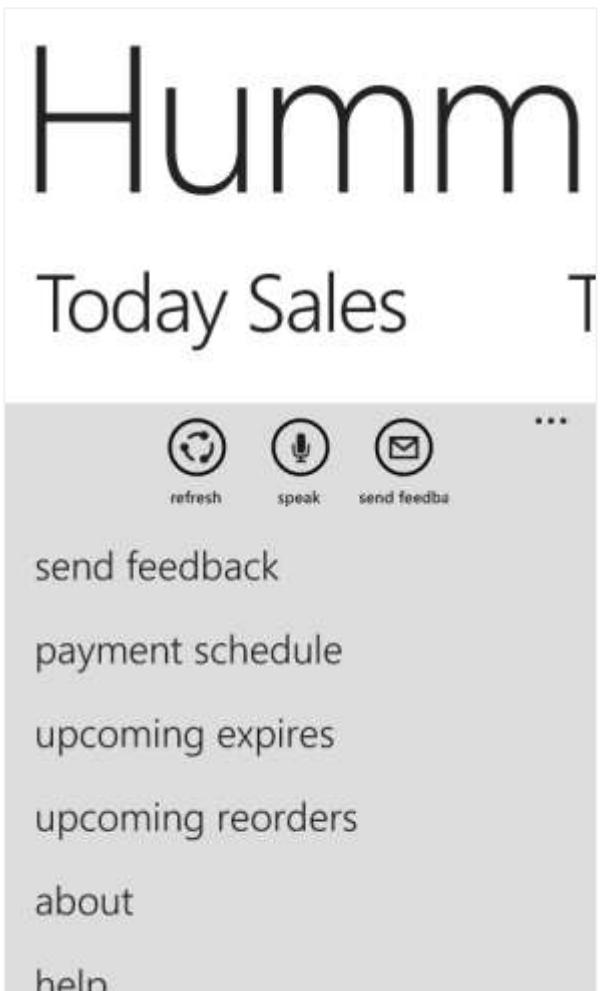


Figure 52 Menu

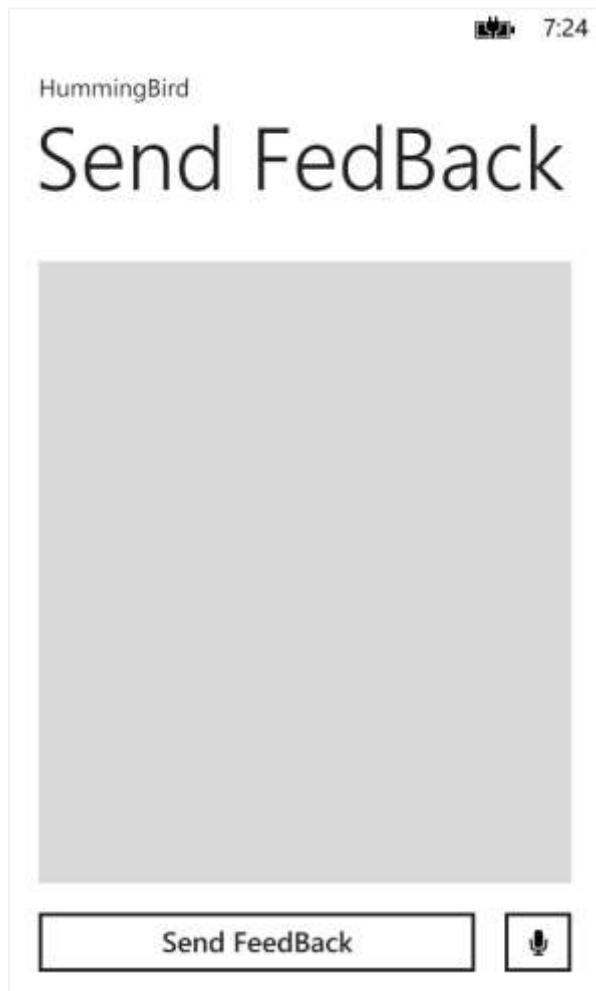


Figure 51 Send Feedback

4.6.2 Voice Recognition

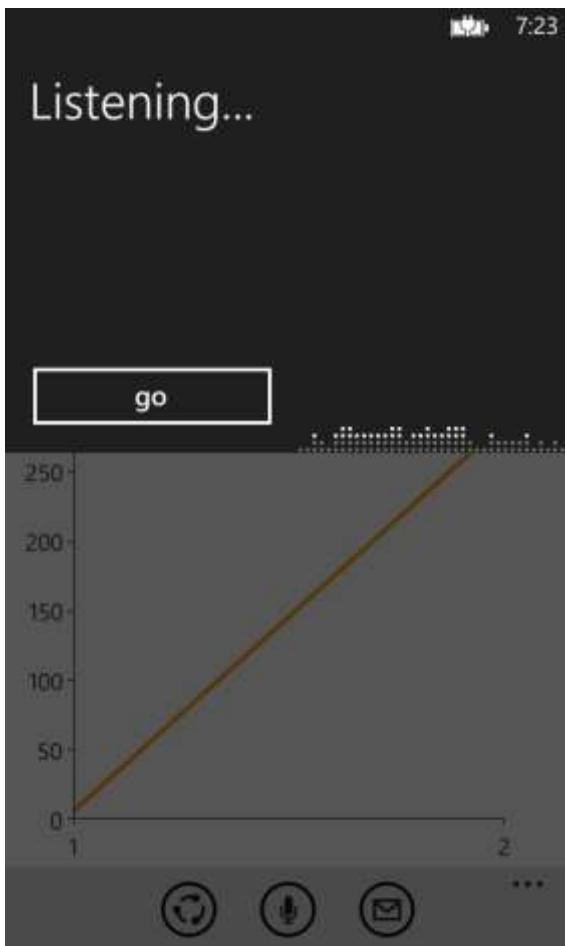


Figure 54 Voice recognition 1

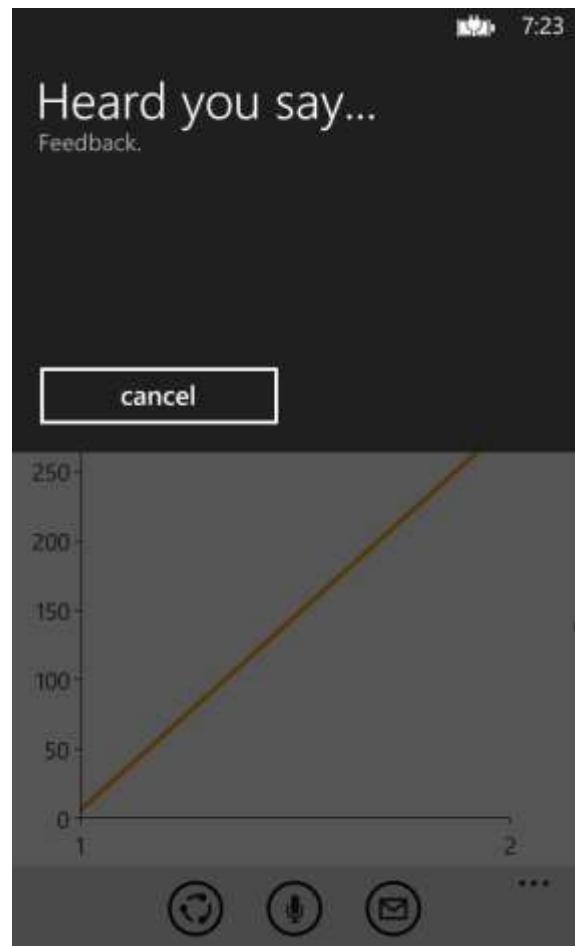


Figure 53 Voice recognition 2



Figure 56 Expires



Figure 55 Date Picker

4.6.3 Reorder report

Upcoming ReOrders		
CloseUp Zone	InLevel Zone	Next
ABSOLUTE RED - 22M	4	
On Hand : 3 Order Levle : 2 COSMETICS ISHARA PHARMACY	On On CC 4-I	On On CC 4-I
ACTIFAST PANADOL 5	4	
On Hand : 70 Order Levle : 50 PAIN KILLERS ISHARA PHARMACY	On On CC 4-I	On On CC 4-I
ADVANT 8MG TAB	4	
On Hand : 42 Order Levle : 30 HYPERTENTION DRUGS ISHARA PHARMACY	On On CC 4-I	On On CC 4-I
ALTIVA 180MG TAB	4	
On Hand : 40 Order Levle : 30 RESPIRATORY SYSTEM	On On CC 4-I	On On CC 4-I

Figure 58 Reorder report 1

Upcoming ReOrders		
CloseUp Zone	InLevel Zone	Next
4EVER (TEA TREE OIL)	4	
On Hand : 3 Order Levle : 3 COSMETICS 4-REVER SKIN NATURALS	On On CC 4-I	On On CC 4-I
4EVER (TEA TREE) F/W	4	
On Hand : 3 Order Levle : 3 COSMETICS 4-REVER SKIN NATURALS	On On CC 4-I	On On CC 4-I
4EVER (VANIVEL) F/W	4	
On Hand : 3 Order Levle : 3 COSMETICS 4-REVER SKIN NATURALS	On On CC 4-I	On On CC 4-I
4EVER BODY MILK (VA	4	
On Hand : 2 Order Levle : 2 COSMETICS 4-REVER SKIN NATURALS	On On CC 4-I	On On CC 4-I

Figure 57 Reorder report 2

Orders OutOf Stock

4EVER (GOLD) F/W 100g A
On Hand : 0
Order Level: 5
COSMETICS
4-REVER SKIN NATURALS

4EVER (KOHO & KAHA) A
On Hand : 0
Order Level: 2
COSMETICS
4-REVER SKIN NATURALS

4EVER SUN BLOCK CR A
On Hand : 0
Order Level: 3
COSMETICS
4-REVER SKIN NATURALS

AGICLOB - GM
On Hand : 0
Order Level: 2
COSMETIC, DERMATOLOGY

ReOrders HighRisk Zone

4EVER (ALMOND) FAIF 4
On Hand : 1
Order Level: 3
COSMETICS
4-REVER SKIN NATURALS

4EVER (KOHO & KAHA) 4
On Hand : 1
Order Level: 3
COSMETICS
4-REVER SKIN NATURALS

4EVER (KOHO & KAHA) 4
On Hand : 2
Order Level: 5
COSMETICS
4-REVER SKIN NATURALS

4EVER (ROSE ANTI AC A
On Hand : 1
Order Level: 3
COSMETICS

hing ReOrders MediumRisk Zo H

4EVER (KOHO & KAHA) 4
On Hand : 2
Order Level: 3
COSMETICS
4-REVER SKIN NATURALS

4EVER (ROSE) F/W 50ml 4
On Hand : 2
Order Level: 3
COSMETICS
4-REVER SKIN NATURALS

4EVER (SEAWEED) F/M 4
On Hand : 2
Order Level: 3
COSMETICS
4-REVER SKIN NATURALS

4EVER (VANIVEL) F/W 4
On Hand : 2
Order Level: 3
COSMETICS

Figure 61 Reorder report 3

Figure 60 Reorder report 4

Figure 59 Reorder report 5

4.6.4 Push Notification



Figure 62 Windows Phone Push Notification

4.7 Summary

5.0 Development

This chapter discuss development process of the entire application under the areas such as working flow of the application development, process flow of the execution, development strategies, database tools, applied programming concepts, and tools.

5.1 Coding Language and Database Selection

5.1.1 Selection of Coding Language

As the proposed application, the language is chosen according the running platform, the application is running on Microsoft windows platform and the application is created as windows application, with metro technology (flat stile) the application is compatible with thatch based users and normal IO (input and output) devices like keyboard and mouse.

As the proposed application the planned mobile application is focused the windows Mobile device. Because of the windows wide platform most suitable and compatible language is Microsoft C sharp and as the interface markup language is XML.

With the proposed system to communicate with mobile application and club database most suitable communication option is Web services, as Web services must appropriate and platform support web service type is selected as SOAP Web service.

Development and database languages that used in the application

- C#
- XML
- SQL

As the best practices in programming language the object oriented architecture, MVC architecture or layered architecture, easily can be handled by the C sharp, or .Net based programming language. According to the targeted platform and use experiences that collected from the analyzing part of the project, the windows based programming language selected.

They XML language used for the development purpose of the mobile application and dashboard design of the dashboard in the windows application.

As the Microsoft platform the best language is C# (c Sharp) and accordingly the database engine is chosen as Microsoft SQL Server.

From the C sharp language and Microsoft.net platform, most user-friendly and customizable third-party plug-ins can use to make their occasion in good standard and user-friendly.

Windows based applications are running one .Net framework, with the .Net framework, most compatibility languages C sharp more than visual basic or C Languages.

According to the selected language, the whole application including Mobile application, is developed by using C# language, Specially the Android application, the **Android application also Developed by the C#, Using Cross Platform** technology

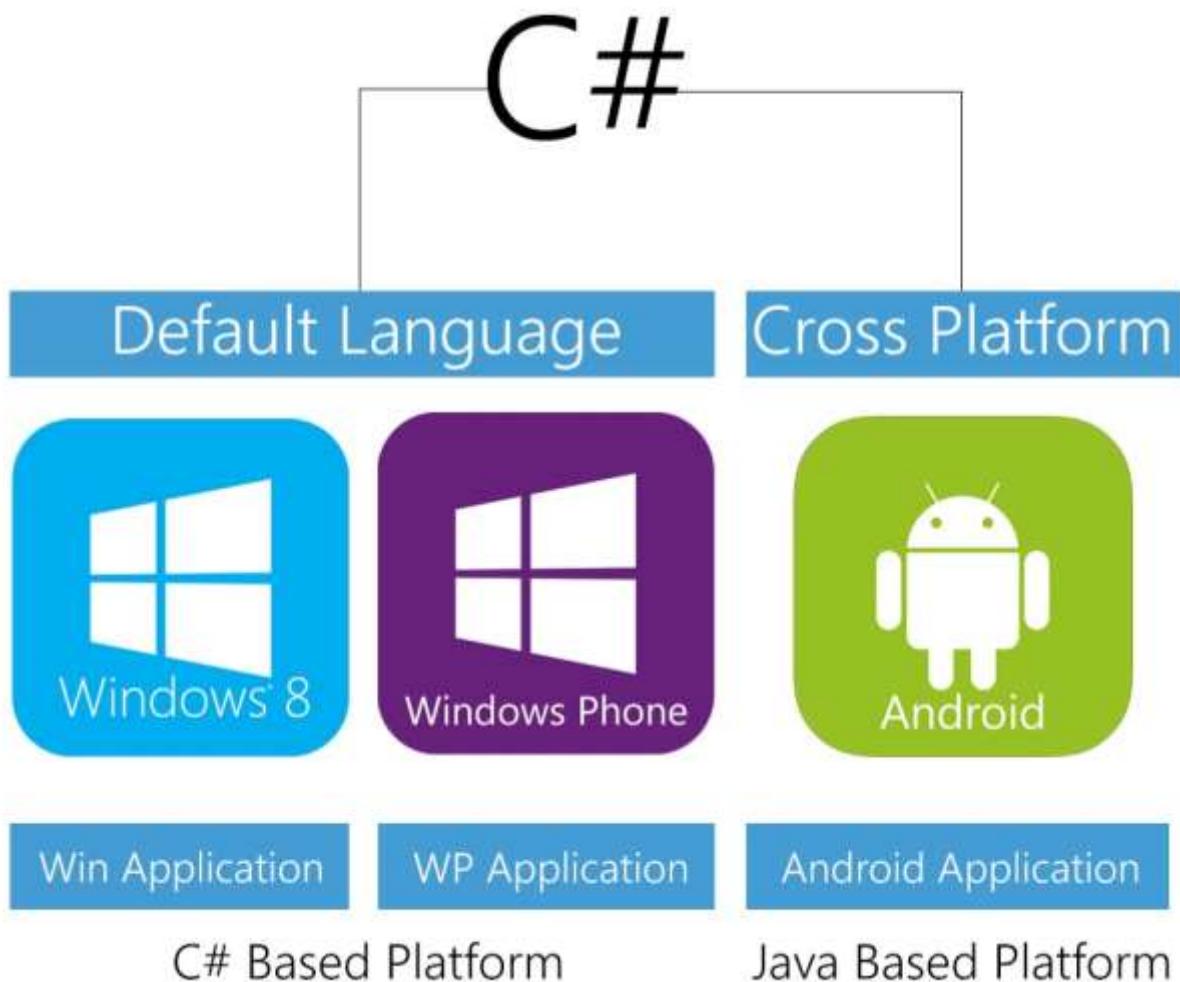


Figure 63 Cross Platform Technology

5.1.2 Database Selection

According to the application platform, the database engine is selected as Microsoft SQL server. The Microsoft SQL Server is windows based and .Net environment friendly Database server. The database format is Microsoft, the database format for .net developers.

5.2 Technologies and Tools for Design and Implementation

5.2.1 Development Tools

After the Language selection for the project, the tools selection was the next important part, a project romance and successful outcomes are depend on the tools that used to develop the application as well as coding language.

In the proposed application the development area is wide and usage of tools and IDEs also high. As well as plugins and devices that used to test and development are important.

In the process of development of the proposed system, several tools were required to support various activities such as creating models or other components required in the development process. Most of the tools have been designed and developed with several specific functions to make easy the system developers role in the system. Some of the tools that may be required are described below.



Figure 64 Development Tools

5.2.1.1 Development IDE

An integrated development environment (IDE) is a programming environment that has been packaged as an application program, typically consisting of a code editor, a compiler, a debugger, and a graphical user interface (GUI) builder. Some IDE has facility to generate the source codes as well. These tools are specially designed to help the software developers to ease the development process interactively. Adobe dream viewer can be taken as an example for such an IDE which is used for development of web pages. It has the code generating feature for html elements.

Mainly the IDE that used to development are

- Microsoft Visual Studio 2013 Ultimate (www.visualstudio.com)



Microsoft Visual Studio is the main Development IDE that used to develop the Windows application and Windows Phone Application.

No day the world biggest .Net based development IDE is the Visual Studio, with hires capability with chosen language (C#) to develop the Project.

The Visual Studio is capable of developing the Windows Mobile application. In the project the Windows Phone application also developed using the Microsoft Visual Studio and using in build Windows Phone 8 Emulator.

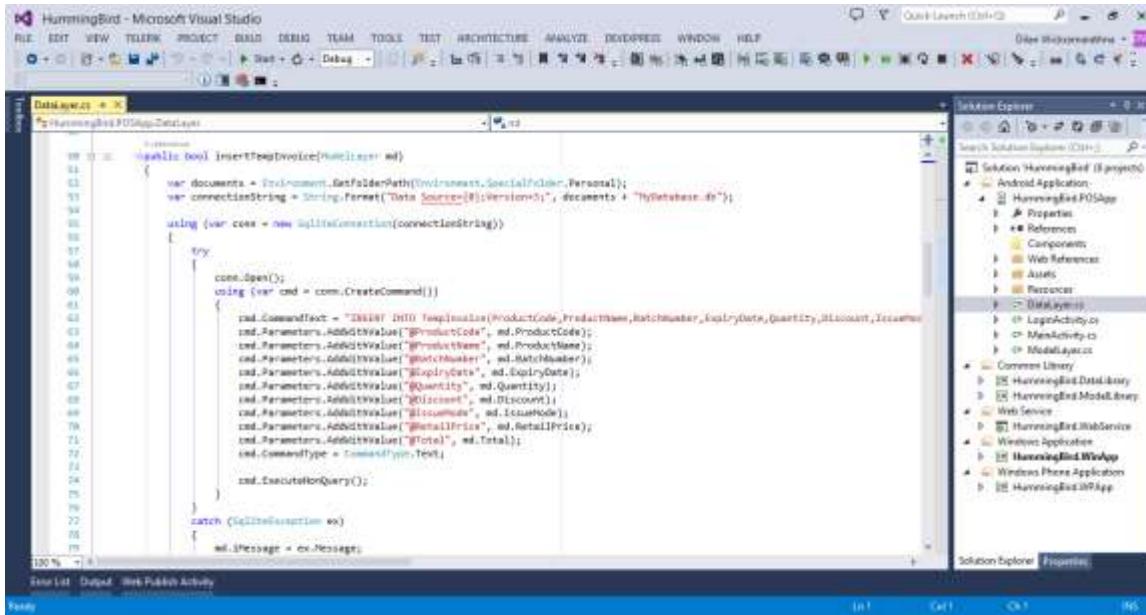


Figure 65 Visual Studio IDE

- **Xamarin Studio 5.5** (www.xamarin.com)



Xamarin is a software product company that interfused real-world cross platform independent mobile and Mac based libraries that exposes a set of libraries APIs for achieve shared mobile device functions across IOS, Android and Windows Mobile, to make and develop re usably mobile application libraries and Mobile application faster and easier than traditional mobile application development technology.

To select the Xamarin as the android application development that proposed with the project, the Xamarin platform is based on the Microsoft .Net framework and C# language.

Xamarin Studio is used to develop the Android application that comes with the proposed application, the Xamarin Studio is based on **mono cross platform technology**.

And the Xamarin function are working with the Visual studio, and android application can be developed in the visual studio using Xamarin features.



Figure 66 Xamarin Studio

5.2.1.2 Helping Tools and Plugging

In the development, to make the Windows and Windows Phone application user friendly and most attraction, and add mode features to the application with new technology, most new localized plugins are used with the application.

- **Telerik** (www.telerik.com)



Telerik is new revolutionary product that present Windows UI controls and Mobile UI Controls also ASP.Net UI controls.

By using the Telerik, the windows application user interfaces are developed to make better user experience with new technologies apart from traditional and classic windows application user interfaces.

From the Telerik UI controls, the user interfaces are developed, besides that, the controls are mostly used to make the application useful and do the multiple tasks from one controls that included in the interface. As the example, Data grids, from the new data grids, users can easily categorize and edit select filter and extract the data in single click and user can real time customize the controls to make the analytical interface.

From the development side, by using the custom controls with the application, can develop the application much faster than traditional controls and can lowdown the complexity of the unwanted codes, and algorithms.

- **DevExpress** (www.devexpress.com)



DevExpress was another external plugging that used to develop the application and Data Analytical Dashboard, DevExpress is a software product company that providing the UI controls to the Borland Delphi, and Active X controls to the Visual Studio to mane easy the development and used attraction and high quality user experience with increased coding profane.

5.2.1.3 Database Management System (DBMS) and Database Tools

The proposed software is essentially developed as a database product, therefore the system needs to interact with the database and the DBMS in order to query, store the data from and to the database. MSSQL is an example for a Database Management System (DBMS) which is the currently using .net based enterprise level Web-based applications, Web Services and virtually all the .Net framework target Web properties.

In the enterprise development the proper database engine and database design development client is the most important thing when data management. To accomplish the database development the used Database Platform was Microsoft SQL Server and primary database client was the Microsoft SQL Server Management Studio, for the development purpose the Express Edition and Local Database version are used.

- **Microsoft SQL Server** (www.microsoft.com/en-us/server-cloud/products/sql-server/)



In the Enterprise industrial level with the .Net develop framework based development and most recommended database engine is Microsoft SQL Server. As the current requirement to develop the application, .Net framework has been used. According to that, as better database environment, SQL Server has been selected.

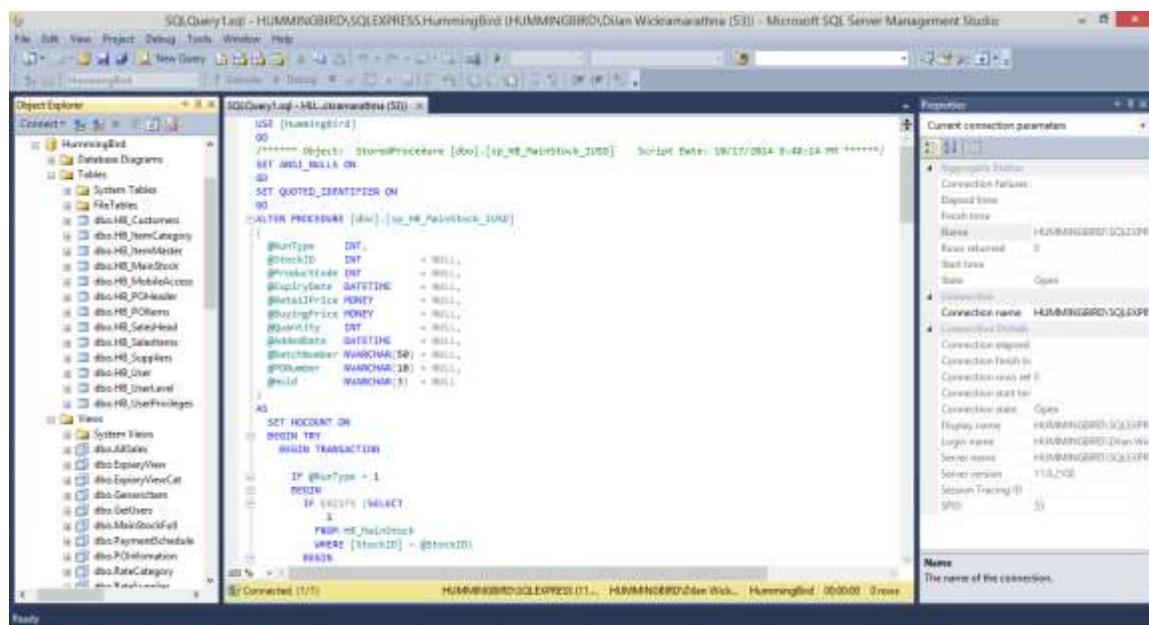


Figure 67 SQL Server Management Studio

- **dbForge Studio for SQL Server** (www.devart.com)

The dbForge Studio for SQL Server is used as Database Management tool to manage and format and optimize the SQL queries and as the extra tool, dbForge Studio for SQL Server is good at schema comparing and database administration.

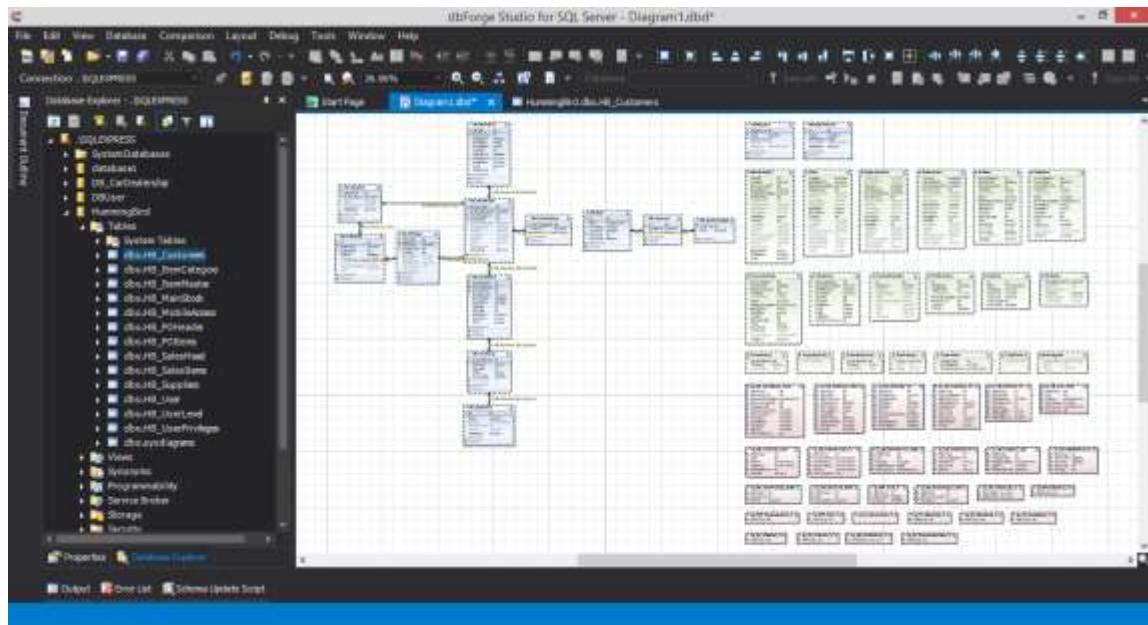


Figure 68 dbForge Studio for SQL Server

5.3.2 Design Tools

Data designing tools has been used to design diagram, UML designs, relational schema and graph while designing the project.

- **Microsoft Visio 2013**

Microsoft Visio 2013 is used to data modeling, develop the diagrammatical designs and data analyzing designs to the project. Mostly used in the designing stage of the project as well as system analyzing stage.

- **Microsoft Project 2013**

The Microsoft project is used to develop the gantt chart, time frame for the project.

- **EdroMax 8**

EdroMax also used to develop the diagrammatical designs and data analyzing designs to the project. Mostly used in the designing stage of the project as well as system analyzing stage.

- **Coggle**

Coggle is used to design the tree chart to the project documentation, mostly in the analyzing stage.

5.3 Order of Development

It is very important to decide about the order which the components would be developed. Around the development order was selected which begins with interface designing to develop the proposed **IBAIS**. The primary advantage with the top-down development order is that there is always a working version. Once the interface modules are completed development moves down to the next set of components of the software design as shown below. And the development order has been lined up as discussed below according to the tree chart.

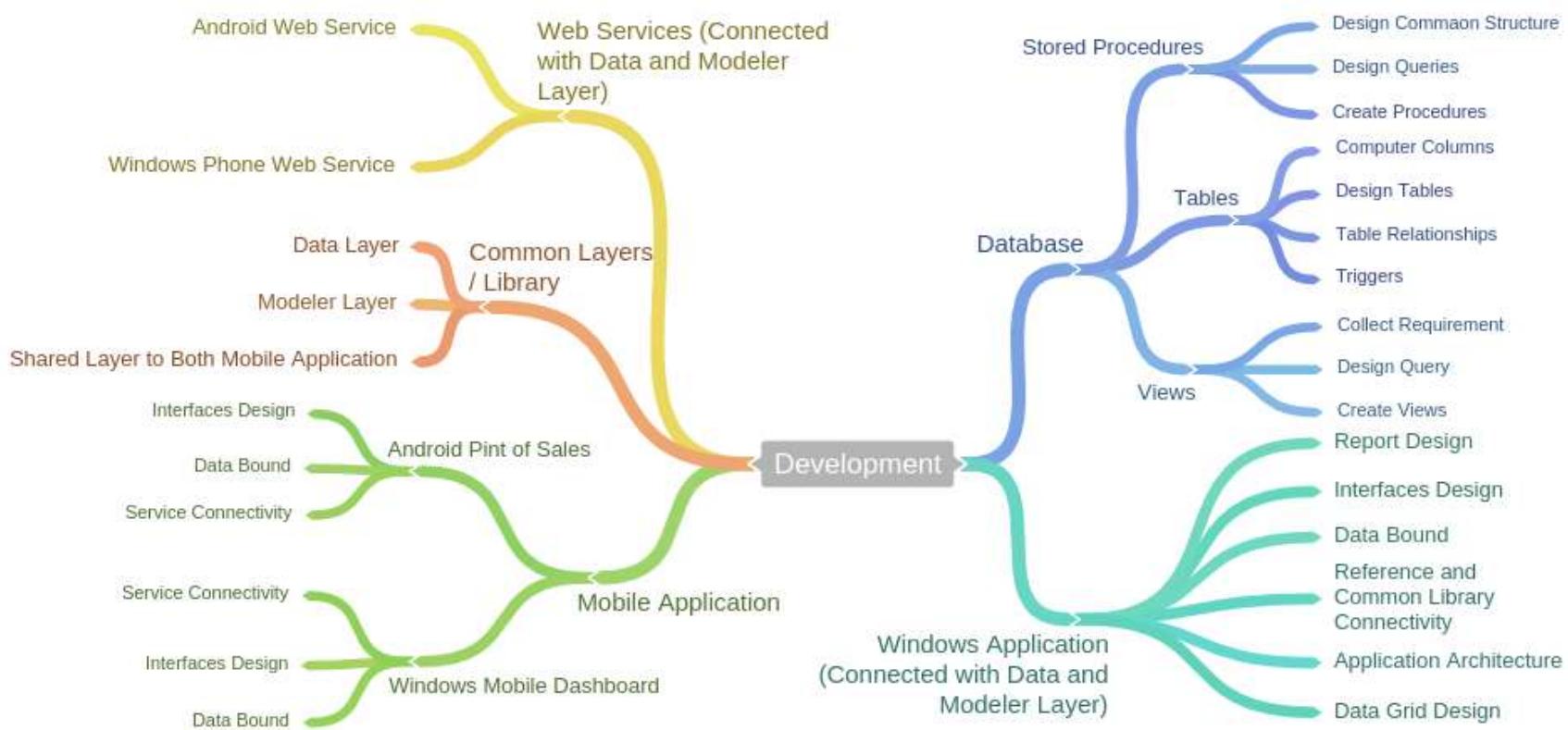


Figure 69 Order of Development

As above, the development part is divided in to main five sectors, as plane also lined up as below.

1. Database Development
2. Common Layers libraries development
3. Windows Application Development
4. Web Service Development
5. Mobile Application Development

5.3.1 Database Development

Database development is the main part of the process that chosen, according to the designed database structure in the Design phase, the database was created. In the database development part the development was separated in to main there part as below.

5.3.1.1 Create Tables

In the table creation stage all tables are created with the pre design diagrams, to design the database, main tool was the SQL Server 2012 SP2 Design Diagramed was used to automate the design and development process of the table, the main reason to the tool and feature that used to develop the tables in SQL Server, by using the database Design Tables, to create computed columns, table relationships can be done with wide analysis with the explained concept.

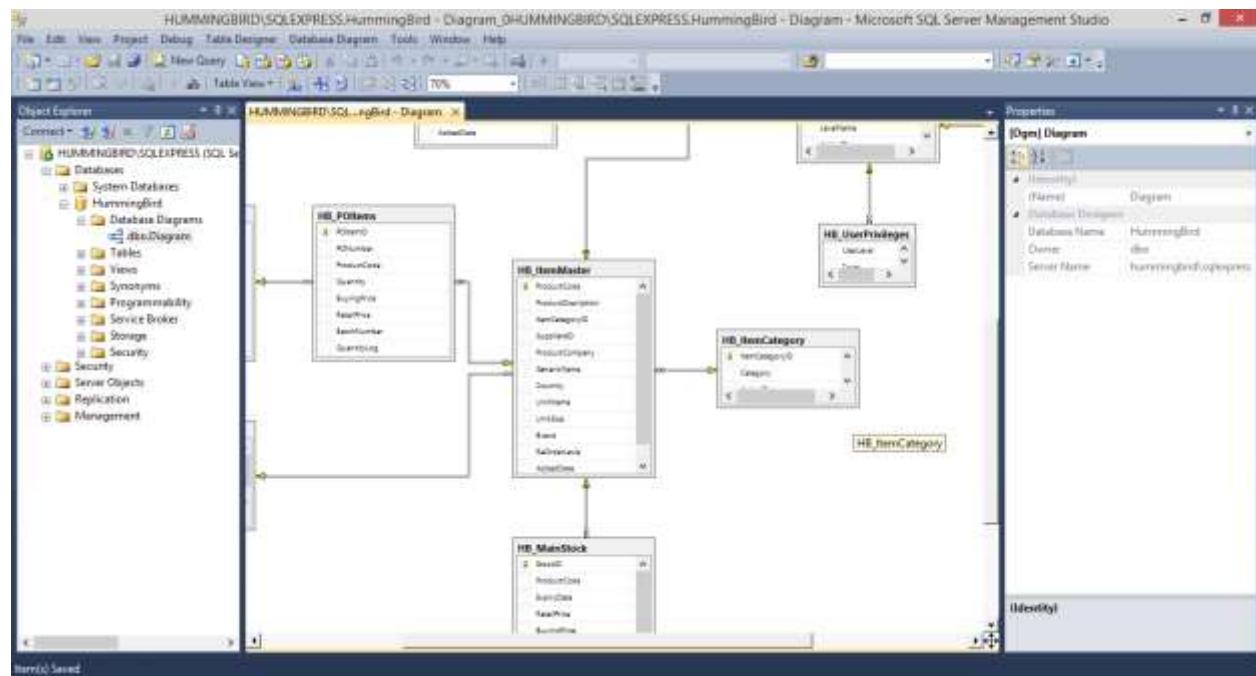


Figure 70 Create Tables

5.3.1.1 Create Views

The views are created to avoid the complexity of the query and optimize the performance of the queries and selection statements, and re use the queries. And to avoid the query duplication.

After collecting requirement about the required view in the database, all views are generated after design the optimized query.

*Creation of the Views has been discussed under the **Common Development Architectures***

5.3.1.1 Create Stored Procedures

Stored Procedures, in the enterprise database development, the Stored Procedures are used to lot of statements as a single execution plane. The Stored Procedures is group of transact SQL statement that complied with a single execution plane to take the return data as Output Parameter, Return Codes, Result Set that retrieving from Select statement or Cursors.

Using the Stored Procedures,

- Application can increase the data transaction performance.
- Stored Procedures are more productive and easy to use.
- Enterprise level concept.
- More scalable.
- Fast and easy maintainability.
- Interoperability.
- Security.
- Reliability.

As the reasons of above, the stored procedures are chosen as the database construction and database transaction method.

*(Creation and design common structure, design queries create stored procedures has been discussed under the **Common Development Architectures**)*

5.3.2 Common Layers libraries development

After finishing the database development, next step was making of the data and modeler layers to the application, by the section, all data transferring processors are handled via the Class libraries. From the section, mainly, the Data Layer and Modeler layers are developed. Those layers are used with the proposed Web Service and Windows Application to data transportation in batwing Application and Database.

By using the separately created layers, the re usability is achieved. And tasks are divided in to the separate responsible classes.

5.3.3 Windows Application Development

The Windows application was the next section that developed according to the order, the process is started from creating the Interfaces, and created interfaces are designed by using User Controls to achieve the re usability.

After creating the interfaces with specific architecture that important at **exclusive** data handling, the data binding process is attached by importing the common data and modeler layer as references to the application via the inbuilt classes that created in the referred layers.

5.3.4 SOAP Web Service Development

After the Windows application development, the next stage was the Web service, to the web services the common layers that created in the previous stage are used. As the lined up next stage, Mobile application is depend on the Web Services.

5.3.5 Mobile Application Development

The mobile application development is the next stage as the lined up plane. From the stage, two smart mobile application are developed. To achieve two deferent aspects.

Android Application – The Android application is used to do the Point of sales and developed to the android tablet devices. From the application employees can do the sales without accessing the main application the installed in the computer. Employees can use the business place internal Wi-Fi network to the data transferring process.

Windows Phone application – The Windows Phone application is used to accomplish the Data Analyzing purpose of business owner. From the application, owner can graphically view the situation of the business. Application is work with any internet source and can access anywhere in the world.

5.3.5.1 Support Mobile Application Programming Interface (API)

To all the mobile application the notification services has been used, for the both android and windows phone services, specific Application Programing Interface (API) used.

For the Android application, to handle the push notification, the **Google Developer Console** and using **Google Cloud Messaging for Android Service** has been used

(<https://console.developers.google.com>)

For the Windows phone application, to handle the push notification, the **APP42 Management Console** has been used (<https://apphq.shephertz.com>)

*(More information about API Usage has been discussed under the **Coding Structures and Techniques, Notification API for Windows and Android Mobile application**)*

5.4 Common Development Architectures

In the caption discussing about the commonly used techniques to develop all the modules in the application and used new and technologies that used in commonly all modules.

5.4.1 Menu and login

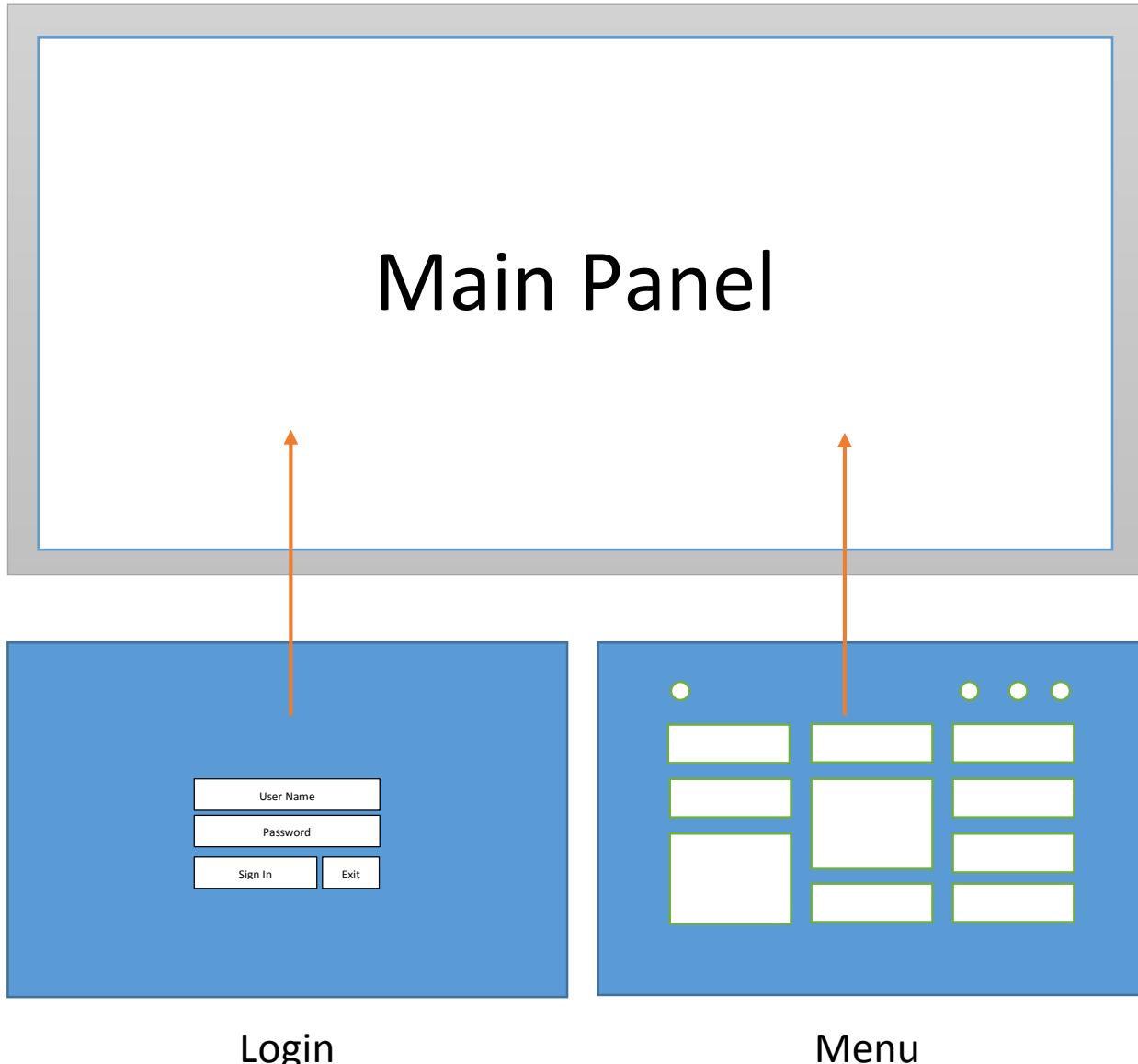


Figure 71 Menu and login

The login and main menu is included on to one windows form that named as Home as the **User Controls**, and transformation between the menu and login (login and logout), is handling from

the based windows from, and all the click events that firing from controls (Buttons, tile window buttons) that access level maids as public in the user controls are detecting by the Home windows form and managing events according to the commands that coded in the C# class file.

Please refer (Appendix B: 1) for the coding that used in this phase

5.4.2 Metro Menu Style

In the menu, users can **navigate** in to the required module and re trace from the main menu by clicking on the metro tile and back button in any module.

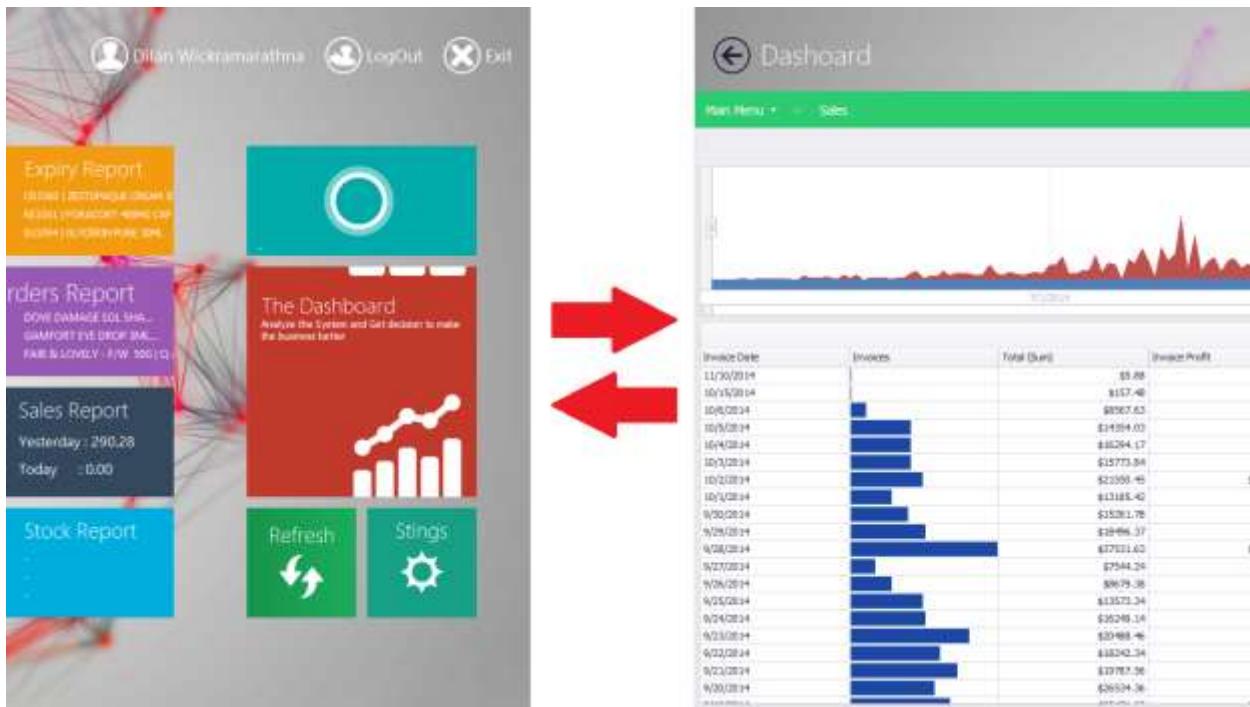


Figure 72 Metro Navigation

The structure of the metro module hierarchy is has been handled by the document DevExpress manager.

By adding the Windows UI components to the user controller developers able to customize the tiles that need to be displayed. And by handling the Query controller in each metro tile click, can handle the required module that need to be display.

The query control has been handled by the following code.

Please refer (Appendix B: 2) for the coding that used in this phase

```

SetUserPrivileges Set User Allowed Tiles
1 reference
private void SetUserPrivileges()
{
    mdUp.UserLevel = iGlobal.UserLevel;
    DataTable dt = dlUs.SelectUserPrivileges(mdUp);

    PointofSaleTile.Visible = false;
    PurchaseOrderTile.Visible = false;
    GoodsReceivingNoteTile.Visible = false;
    ItemMasterTile.Visible = false;
    StockTransferTile.Visible = false;
    ProductControlsTile.Visible = false;
    PaymentScheduleTile.Visible = false;
    DashboardTile.Visible = false;
    StingsTile.Visible = false;
    StockReportTile.Visible = false;
    ReOrdersReportTile.Visible = false;
    ExpiryReportTile.Visible = false;
    SalesReportTile.Visible = false;

    foreach (DataRow dtr in dt.Rows)
    {
        switch (dtr["Form"].ToString())
        {
            case "PointofSale":
                PointofSaleTile.Visible = true;
                break; ↴
            case "PurchaseOrder":
                PurchaseOrderTile.Visible = true;

```

Table 2 Metro Navigation Coding

5.4.3 General Form Structure of Data Handling Forms

In the general form structure all the controls are created to maintain easy access to user and user friendly.

The main purpose of creating common structured windows is fast data transferring with edit update delete and add new records in to window getting fast and easy.

As the sample, in the Stock Transfer module, the main grid view is holding data and user can click each record and take the all sub information to the textboxes and can force to edit or delete.

To manage the special features on the grid view the Devexpress XtraGris has been used and customized as the current requirement.

Stock Transfer

Click

Category	Batch	Product Description	Supplier	Expiry Date	Added Date	Retail Pr...	Buying Price	Quantity	Unit	Unit Sl.	Generic	Re Order Le...	Hold	PO Num
CREAM & OINTMENT	CR4	ASTROCORT 0.5% - 10G	ISHARA PHAR...	03/17/2016	09/11/2014	160.00	139.20	1 10G	EA	HYDROCORTISONE		3 No	Free Added	
CREAM & OINTMENT	CR7	BETADINE OINTMENT - 15G	ISHARA PHAR...	09/01/2015	05/11/2014	793.00	343.65	1 15G	EA	POVIDONE-IODINE		3 No	Free Added	
CREAM & OINTMENT	CR11	SEPROGENTA 15G	ISHARA PHAR...	07/01/2016	05/11/2014	415.15	361.18	2 15G	EA	BETAMETHASONE		3 No	Free Added	
CREAM & OINTMENT	CR13	SEPROSONE CREAM 15G	ISHARA PHAR...	04/01/2016	05/15/2014	339.25	295.15	1 15G	EA	BETAMETHASONE		3 No	Free Added	
CREAM & OINTMENT	CR14	BETANONE CREAM (SPC) 15G	ISHARA PHAR...	08/01/2013	05/11/2014	31.75	45.02	1 15G	EA	BETAMETHASONE		3 No	Free Added	
CREAM & OINTMENT	CR19	BETAMIN - N - CREAM 10G	ISHARA PHAR...	04/01/2016	05/11/2014	107.00	102.69	1 10G	EA	BETAMETHASONE		3 No	Free Added	
CREAM & OINTMENT	CR20	BETAMIN - GM 10G	ISHARA PHAR...	03/01/2015	05/11/2014	108.00	103.58	1 10G	EA	BETAMETHASONE		3 No	Free Added	
CREAM & OINTMENT	CR25	CANDID - PLAIN 20G	ERAJ	04/01/2016	05/11/2014	130.00	130.50	1 20G	EA	CLOTRIMAZOLE		2 No	Free Added	
CREAM & OINTMENT	CR27	CANDACORT CREAM 15G	ERAJ	09/01/2016	05/17/2014	333.80	292.15	1 15G	EA	CLOTRIMAZOLE		3 No	Free Added	
CREAM & OINTMENT	CR28	CLODERM CREAM 15G	ISHARA PHAR...	02/01/2016	05/11/2014	369.00	320.16	3 15G	EA	CLOBETASOL		2 No	Free Added	
CREAM & OINTMENT	CR30	CANIVON CREAM 20G	ISHARA PHAR...	07/01/2016	05/11/2014	60.00	52.20	2 20G	EA	CLOTRIMAZOLE		2 No	Free Added	
CREAM & OINTMENT	CR31	DEKAQUIN CREAM 15G	ISHARA PHAR...	03/01/2016	05/11/2014	90.00	78.39	1 15G	EA	CLOQUINOL		2 No	Free Added	
CREAM & OINTMENT	CR32	DEKAQUIN CREAM 15G	ISHARA PHAR...	07/01/2016	05/11/2014	90.00	78.39	1 15G	EA	CLOQUINOL		2 No	Free Added	
CREAM & OINTMENT	CR33	DERMOVATE OINTMENT 10G	ISHARA PHAR...	03/01/2015	05/11/2014	323.00	283.75	1 10G	EA	CLORETASOL		3 No	Free Added	
CREAM & OINTMENT	CR38	DIPROVATE IN CREAM 10MG	ISHARA PHAR...	11/01/2014	05/11/2014	57.50	50.03	2 10MG	EA	DIPOTASSUM		2 No	Free Added	
CREAM & OINTMENT	CR46	DEZOR CREAM - 15G	ISHARA PHAR...	09/01/2016	05/13/2014	363.40	316.18	1 15G	EA	KETAKENAL		2 No	Free Added	

Product: ASTROCORT 0.5% - 10G

New items. Not Available in stock.

Batch Number: CR4

Expiry Date: 03/17/2016

Retail Price: 160

Buying Price: 139.20

Quantity: 1

Reorder Level: 1

Hold Product: No

Save Delete

Figure 73 General Form Structure

5.4.4 Customizable and Real-Time Data viewers/Grid view

In the application the grid views are created to make users work fast and make work easy.

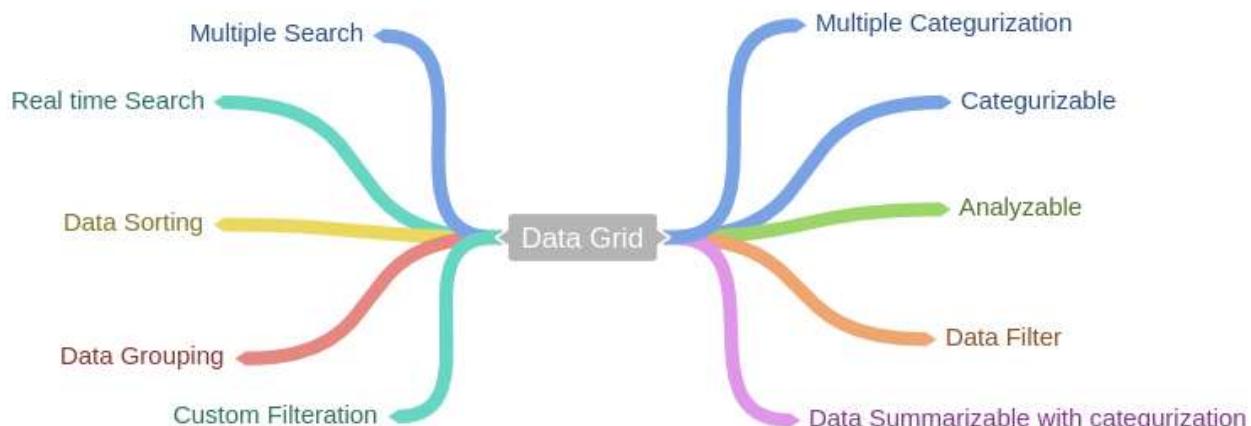


Figure 74 Customizable and Real-Time Data viewers/Grid view' fatures

Special features of the Data viewers/Grid view

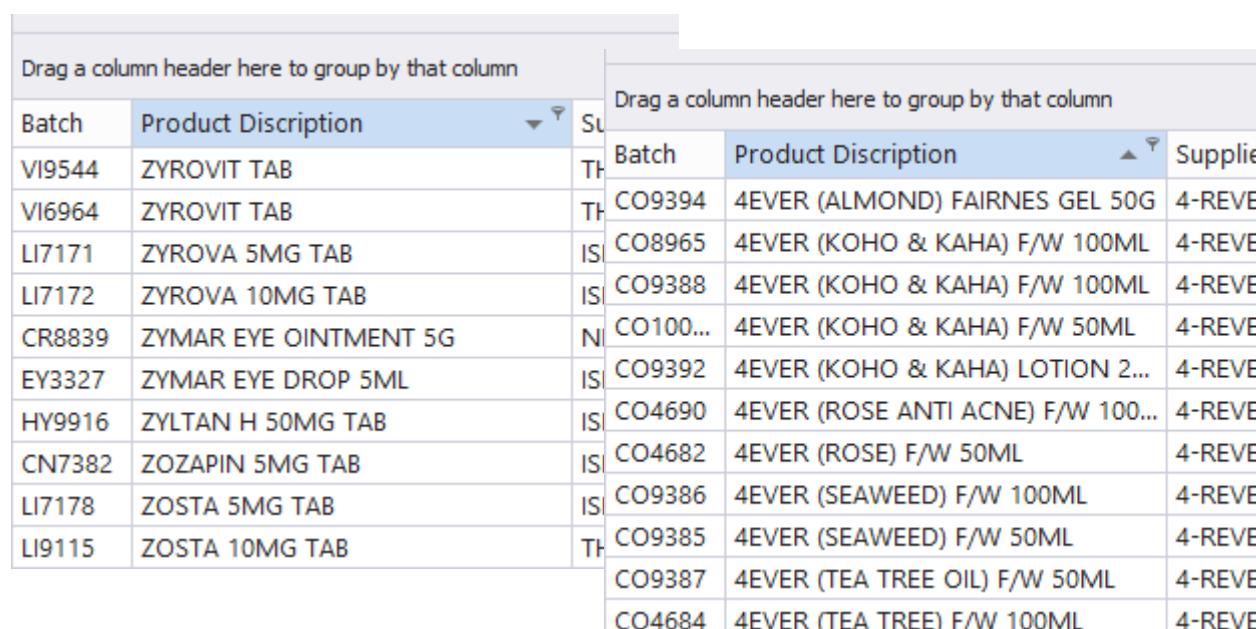
- Data sorting
- Group

- Categorize
- Multi Categorize
- Filtering
- Custom filtering
- Search

All the functions are included in every module that required the Data viewers/Grid view, and the main purpose of having those function is **Real-time Data Analyzing**.

5.4.4.1 Data sorting

Ascending and descending sorting features on any column

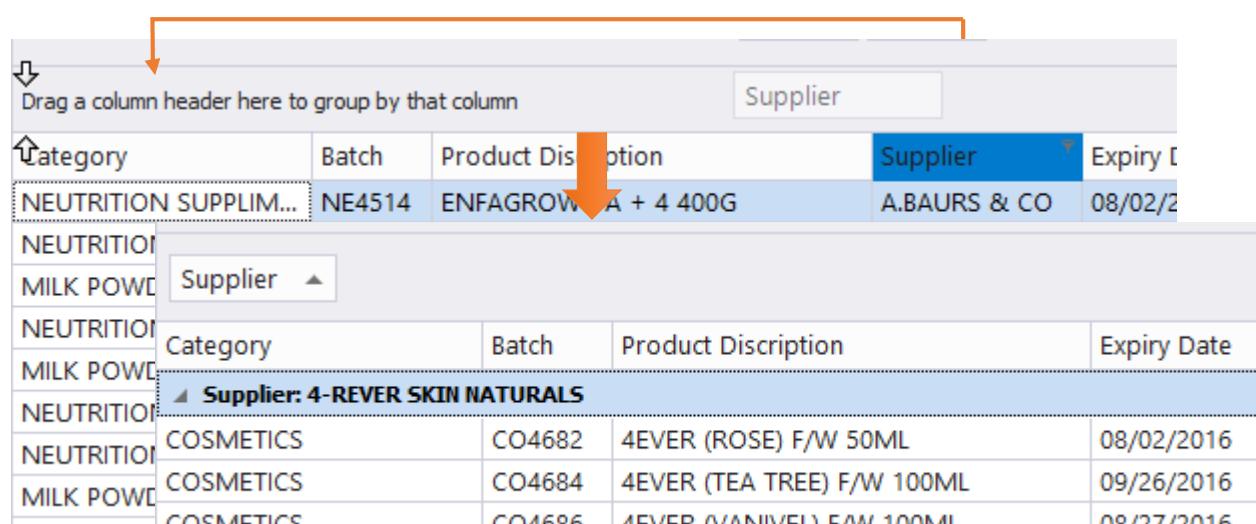


Batch		Product Description		Supplier	
VI9544	ZYROVIT TAB	TH	CO9394	4EVER (ALMOND) FAIRNES GEL 50G	4-REVER
VI6964	ZYROVIT TAB	TH	CO8965	4EVER (KOHO & KAHA) F/W 100ML	4-REVER
LI7171	ZYROVA 5MG TAB	IS	CO9388	4EVER (KOHO & KAHA) F/W 100ML	4-REVER
LI7172	ZYROVA 10MG TAB	IS	CO100...	4EVER (KOHO & KAHA) F/W 50ML	4-REVER
CR8839	ZYMAR EYE OINTMENT 5G	NI	CO9392	4EVER (KOHO & KAHA) LOTION 2...	4-REVER
EY3327	ZYMAR EYE DROP 5ML	IS	CO4690	4EVER (ROSE ANTI ACNE) F/W 100...	4-REVER
HY9916	ZYTAN H 50MG TAB	IS	CO4682	4EVER (ROSE) F/W 50ML	4-REVER
CN7382	ZOZAPIN 5MG TAB	IS	CO9386	4EVER (SEAWEED) F/W 100ML	4-REVER
LI7178	ZOSTA 5MG TAB	IS	CO9385	4EVER (SEAWEED) F/W 50ML	4-REVER
LI9115	ZOSTA 10MG TAB	TH	CO9387	4EVER (TEA TREE OIL) F/W 50ML	4-REVER
			CO4684	4EVER (TEA TREE) F/W 100ML	4-REVER

Figure 75 Ascending and descending

5.4.4.2 Data grouping and categorize

By dragging the needed column to grout aria user can group full of grid in to grouped grid according to the selected column.



Drag a column header here to group by that column					
Category	Batch	Product Description	Supplier	Expiry Date	
NEUTRITION SUPPLIM...	NE4514	ENFAGROW A + 4 400G	A.BAURS & CO	08/02/2016	
NEUTRITION					
MILK POWD	Supplier ▲				
NEUTRITION	Category		Batch	Product Description	Expiry Date
MILK POWD					
NEUTRITION	▲ Supplier: 4-REVER SKIN NATURALS				
NEUTRITION	COSMETICS		CO4682	4EVER (ROSE) F/W 50ML	08/02/2016
MILK POWD	COSMETICS		CO4684	4EVER (TEA TREE) F/W 100ML	09/26/2016
MILK POWD	COSMETICS		CO4686	4EVER (VANILLA) F/W 100ML	08/27/2016

5.4.4.3 Multi Categorize

User are able to do multi categorization by dragging multiple columns in to grouping area.

		Supplier	Category						
Batch	Product Description			Expiry Date	Added Date	Retail Price			
Supplier: A.BAURS & CO									
Category: MILK POWDER									
MI4523	ENFAMIL A+ (2) 400G			10/23/2015	06/08/2014	1,016.0			
MI4525	ENFAGROW A+ (3) 400G			01/10/2016	06/08/2014	1,290.0			
MI4535	ENFAMIL A+ (01) 400G			07/17/2015	06/08/2014	1,275.0			
MI119...	ENFAGROW A+ (3) 400G			02/01/2016	10/03/2014	1,290.0			
Category: NEUTRITION SUPPLEMENT									
Supplier: A.C.S. DISTRIBUTORS									
Category: BABY CARE									
BA9237	BAMBI PAMPERS (S)			05/01/2016	08/12/2014	150.0			
Category: COSMETICS									
CO99...	OPTIMUM COLO 50ML			06/01/2016	09/06/2014	490.0			
CO99...	O3 - COLONGE SPRAY 50ML			06/01/2016	09/06/2014	335.0			

Figure 77 Multi Categorize

5.4.4.4 Filtering

User are able to filter by any column as the need and customize the filtration by using custom filter option.

Drag a column header here to group by that column							
Batch	Product Description	Supplier	Category (Custom)	Expiry Date	Added Date	Retail ...	
GE9629	LUMBAR SUPPORT - L	RAGAMA PHA...	4-REVER SKIN NATURALS	7 08/31/2014	4,773.0		
GE9632	LUMBAR SUPPORT - M	RAGAMA PHA...	A.BAURS & CO	9 08/31/2014	4,243.0		
GE9626	LUMBAR SUPPORT - XL	RAGAMA PHA...	A.C.S. DISTRIBUTORS	9 08/31/2014	4,243.0		
NE9931	APPETON WEIGHT GAINER (ADULT...)	ARPICO PHAR...	A.K. DRUGS	5 09/06/2014	3,750.0		
NE9932	APPETON WEIGHT GAINER (CHILD)...)	ARPICO PHAR...	ALARIS LANKA (OSUL)	5 09/06/2014	3,350.0		
NE8702	APPETON WEIGHT GAINER (CHILD)...)	ARPICO PHAR...	AMEDCO PVT. LTD	5 07/15/2014	3,350.0		
GE11805	CERVICAL COLLAR (L)	RAGAMA PHA...	ANURADHA SERVICES - (ASTRON)	9 09/28/2014	2,970.0		
HE11804	ABDOMINAL SUPPORT (M)	RAGAMA PHA...	ARPICO PHARMA - MAHARAGAMA	9 09/28/2014	2,970.0		
GE9634	ABDOMINAL SUPPORT (L)	RAGAMA PHA...	ASIA VISION - VEYANGODA	9 08/31/2014	2,970.0		
NE9466	ENFAGROW A+ (3) 800G	A.BAURS & CO	BABY CHERAMY	5 08/20/2014	2,430.0		
NE8970	ENFAGROW A+ (3) 800G	A.BAURS & CO	BIO LABS	5 07/27/2014	2,430.0		
NE8654	ENSURE VANILLA 400G	RAGAMA PHA...	CBL	7 07/11/2014	2,220.0		
NE9432	ADVANCED WEIGHT GAINER 500G	MEDIHOUSE -	CEYOKA (J.N. DIST)	5 08/20/2014	2,200.0		
DI10006	RESOURCE MILK P. 400G	RAGAMA PHA...	CHANDANALEEPAA (SANJEEWAKA AUR.)	5 09/07/2014	2,190.0		
NE8653	PEDIA SURE VANILA 400G	RAGAMA PHA...	CITY PHARMA - NUPEGODA	5 07/11/2014	1,990.0		
GE0627	ANKLE SUPPORT - I	RAGAMA PHA...	COLOMBO	5 01/01/2010	09/21/2014	1,000.0	
			D.M. DISTRIBUTORS				
			DARLOY BUTLOR (COW & GATE)				
			DETOL (POLHENA TR)				

Figure 78 Filtering

5.4.4.5 Custom filtering

Category	Expiry Date	Added Date	Retail ...	Buying Price	Quantity	Unit
GENARAL	01/01/2017	08/31/2014	4,773.00	4,152.27	1	L
GENARAL	01/01/2019	08/31/2014	4,243.00	3,690.90	1	M
GENARAL					1	XL
NEUTRITION SU					3	450G
NEUTRITION SU					1	450G
NEUTRITION SU					1	450G
GENARAL					1	EA
HEALTH CARE					1	MEDIU
GENARAL					1	L
NEUTRITION SU					1	800G
NEUTRITION SU					2	800G
NEUTRITION SU					2	400G
NEUTRITION SU					1	500G
DIABETICS	05/31/2010	05/07/2014	2,150.00	1,990.51	1	400G
NEUTRITION SUPPLIM...	02/01/2015	07/11/2014	1,990.00	1,791.19	1	400G

Custom AutoFilter

Show rows where:

Category

Is like

And Or

Figure 79 custom filtering 1

...	GENARAL	01/01/2017	08/31/2014	4,773.00	4,152.27	1	L
...	GENARAL	01/01/2019	08/31/2014	4,243.00	3,690.90	1	M
...	GENARAL					1	XL
...	NEUTRITION SU					3	450G
...	NEUTRITION SU					1	450G
...	NEUTRITION SU					1	450G
...	GENARAL					1	EA
...	HEALTH CARE					1	MEDIU
...	GENARAL					1	L
O	NEUTRITION SU					1	800G
O	NEUTRITION SU					2	800G
O	NEUTRITION SU					2	400G
-	NEUTRITION SU					1	500G
-	NEUTRITION SU					1	400G
-	DIABETICS					1	400G
...	NEUTRITION SUPPLIM...					1	EA
...	GENARAL						

Custom AutoFilter

Show rows where:

Category

Is like

And Or

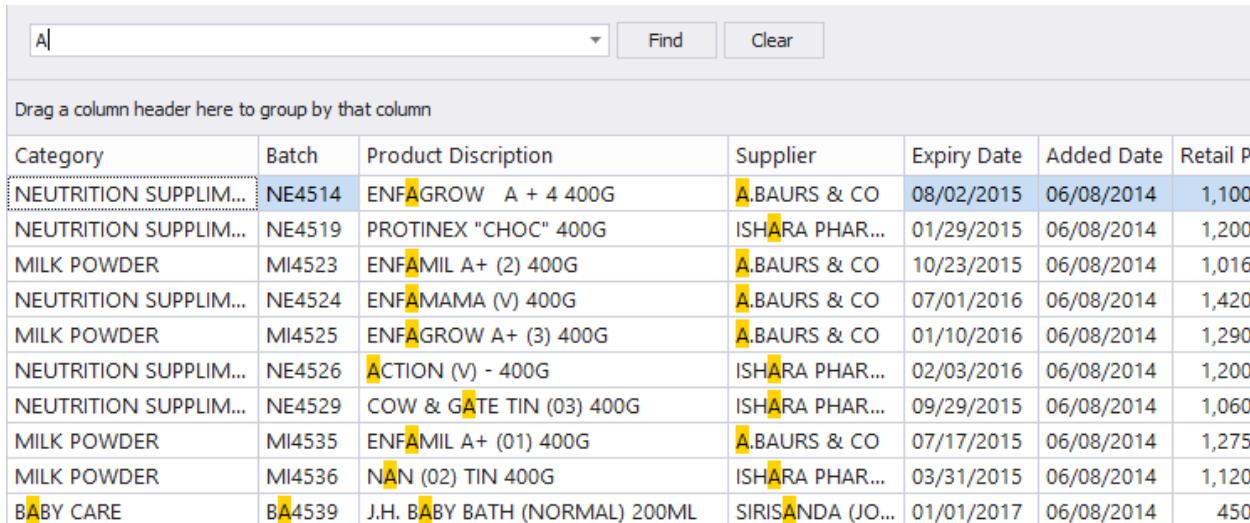
Quantity

- Equals
- Does not equal
- Is like
- Is not like
- Is greater than
- Is greater than or equal to
- Is less than
- Is less than or equal to
- Is null
- Is not null
- Is blank
- Is not blank

Figure 80 custom filtering 2

5.4.4.6 Multiple Search

Searching is handled in the all grid view, the searching field is not created to each column, the search is created to search any column in the grid and search by any single character or number in the grid

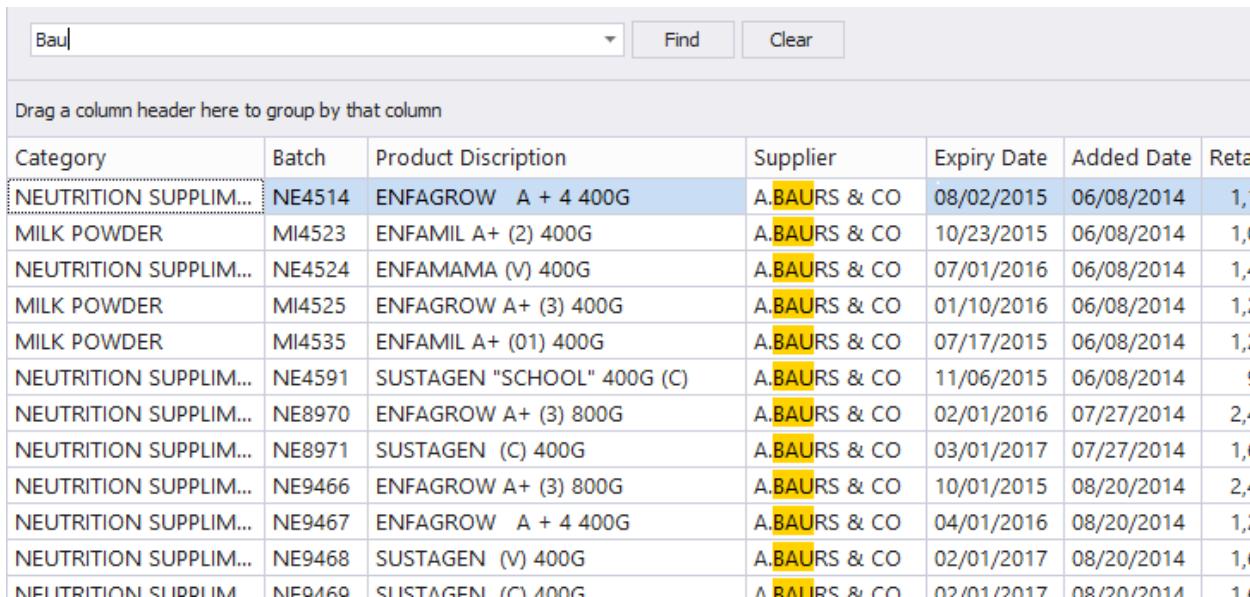


The screenshot shows a search interface with a search bar containing 'A'. Below the search bar is a message: 'Drag a column header here to group by that column'. The main area is a grid table with columns: Category, Batch, Product Description, Supplier, Expiry Date, Added Date, and Retail P. The grid contains 12 rows of product information.

Category	Batch	Product Description	Supplier	Expiry Date	Added Date	Retail P
NEUTRITION SUPPLIM...	NE4514	ENFAGROW A + 4 400G	A.BAURS & CO	08/02/2015	06/08/2014	1,100
NEUTRITION SUPPLIM...	NE4519	PROTINEX "CHOC" 400G	ISHARA PHAR...	01/29/2015	06/08/2014	1,200
MILK POWDER	MI4523	ENFAMIL A+ (2) 400G	A.BAURS & CO	10/23/2015	06/08/2014	1,016
NEUTRITION SUPPLIM...	NE4524	ENFAMAMA (V) 400G	A.BAURS & CO	07/01/2016	06/08/2014	1,420
MILK POWDER	MI4525	ENFAGROW A+ (3) 400G	A.BAURS & CO	01/10/2016	06/08/2014	1,290
NEUTRITION SUPPLIM...	NE4526	ACTION (V) - 400G	ISHARA PHAR...	02/03/2016	06/08/2014	1,200
NEUTRITION SUPPLIM...	NE4529	COW & GATE TIN (03) 400G	ISHARA PHAR...	09/29/2015	06/08/2014	1,060
MILK POWDER	MI4535	ENFAMIL A+ (01) 400G	A.BAURS & CO	07/17/2015	06/08/2014	1,275
MILK POWDER	MI4536	NAN (02) TIN 400G	ISHARA PHAR...	03/31/2015	06/08/2014	1,120
BABY CARE	BA4539	J.H. BABY BATH (NORMAL) 200ML	SIRISANDA (JO...	01/01/2017	06/08/2014	450

Figure 81 Search

Users able to search by any character and user can sees the highlighted output.



The screenshot shows a search interface with a search bar containing 'Bau'. Below the search bar is a message: 'Drag a column header here to group by that column'. The main area is a grid table with columns: Category, Batch, Product Description, Supplier, Expiry Date, Added Date, and Retail P. The grid contains 16 rows of product information.

Category	Batch	Product Description	Supplier	Expiry Date	Added Date	Reta
NEUTRITION SUPPLIM...	NE4514	ENFAGROW A + 4 400G	A.BAURS & CO	08/02/2015	06/08/2014	1,1
MILK POWDER	MI4523	ENFAMIL A+ (2) 400G	A.BAURS & CO	10/23/2015	06/08/2014	1,0
NEUTRITION SUPPLIM...	NE4524	ENFAMAMA (V) 400G	A.BAURS & CO	07/01/2016	06/08/2014	1,4
MILK POWDER	MI4525	ENFAGROW A+ (3) 400G	A.BAURS & CO	01/10/2016	06/08/2014	1,2
MILK POWDER	MI4535	ENFAMIL A+ (01) 400G	A.BAURS & CO	07/17/2015	06/08/2014	1,2
NEUTRITION SUPPLIM...	NE4591	SUSTAGEN "SCHOOL" 400G (C)	A.BAURS & CO	11/06/2015	06/08/2014	9
NEUTRITION SUPPLIM...	NE8970	ENFAGROW A+ (3) 800G	A.BAURS & CO	02/01/2016	07/27/2014	2,4
NEUTRITION SUPPLIM...	NE8971	SUSTAGEN (C) 400G	A.BAURS & CO	03/01/2017	07/27/2014	1,6
NEUTRITION SUPPLIM...	NE9466	ENFAGROW A+ (3) 800G	A.BAURS & CO	10/01/2015	08/20/2014	2,4
NEUTRITION SUPPLIM...	NE9467	ENFAGROW A + 4 400G	A.BAURS & CO	04/01/2016	08/20/2014	1,2
NEUTRITION SUPPLIM...	NE9468	SUSTAGEN (V) 400G	A.BAURS & CO	02/01/2017	08/20/2014	1,6
NEUTRITION SUPPLIM...	NE9469	SUSTAGEN (C) 400G	A.BAURS & CO	02/01/2017	08/20/2014	1,6

Figure 82 Multiple Search

5.5.5 Customizable Reports

In the proposed application, reports are main requirement as they requested, in the data gathering phase. To achieve the data analyzing and to check real time business statuses the reports are used.

In the application major function and important is, using one table, users can generate many reports at once, and can generate multiple outputs. And users can export the report in popular stander document format like Microsoft Word, Excel, PDF and HTML.

Users can customize the report as their requirement and take the printable output from the report.

To customize the report users can drag and drop the column to make grand totals, and that unwanted columns to exclude from the report to remove aria.

Important aria of the Reports



- Customizable
- Categorizable
- Multiple reports at once at one place
- Summerizable
- Minimizable to definition of the main details
- Filters and Customizable Filter
- Customizable Print Option
- Exportable (PDF, Word, Excel, HTML)

In the report all columns are categorizable, by making the needed column as primary (by dragging column to front) user can make the report as categorized report in real time.

The screenshot shows a report interface with a header row containing 'Country', 'Brand', 'Product Company', 'Generic Name', 'Added Date', and 'Unit Size'. Below this is a section titled 'Grand Totals' with a blue arrow pointing to it. To the right of this section is a table titled 'Group Column Header Rows' with a blue arrow pointing to its title. The main body of the report displays data grouped by category, with a summary table at the bottom showing totals for various categories like ANTELMINTICS, ANTIBIOTICS, etc.

Country	Brand	Product Company	Generic Name	Added Date	Unit Size	Grand Totals			Group Column Header Rows									
Current Worth	Spent Cost	Profit				Category	Product	Batch No.	Supplier	Retail Pr.	Buying F...	Expiry D...	Quantity	Unit Name	Re Ord...	Current Worth	Spent Cost	Profit
ANTELMINTICS	PYRATIN S...	AN11535	ANURADH...	175.00	149.33	08/01/2016	- 3	- 10ML	Z	525.00	448.65	76.35						
		AN8603	ISHARA PH...	175.00	157.50	10/02/2016	- 4	- 10ML	Z	700.00	680.00	70.00						
	PYRATIN SYRUP 10ML Total									1,225.00	1,079.65	146.35						
	ZENTEL - T...	AN8503	RAGAMA P...	75.00	64.66	04/01/2017	+ 10	- MG	100	750.00	646.60	103.40						
ANTELMINTICS Total										2,035.00	1,777.45	257.55						
- ANTIBIOTICS										190,318.57	163,876.58	24,441.99						
- BABY CARE										26,827.16	23,120.58	3,706.58						
- CNS										60,192.45	50,194.60	9,937.85						
- CONTRACEPTIVE										1,330.00	1,080.00	270.00						
- COSMETICS										178,238.00	154,238.42	23,999.56						

Figure 83 make the grand total

Users can filter by any columns and generate the report as required.

The screenshot shows a report interface with a sidebar on the left containing a 'Filter' section. The sidebar lists various product categories with checkboxes next to them, such as ANTELMINTICS, ANTIBIOTICS, ANTIDIARRHOEIC, BABY CARE, CNS, CONTRACEPTIVE, COSMETICS, CREAM & LOTION, DIABETICS, and MEDICINE. A blue arrow points to this sidebar. The main body of the report shows data grouped by category, with a summary table at the bottom showing totals for various categories like ANTELMINTICS, ANTIBIOTICS, etc.

Country	Brand	Product Company	Generic Name	Added Date	Unit Size	Grand Totals			Group Column Header Rows									
Current Worth	Spent Cost	Profit				Category	Product	Batch No.	Supplier	Retail Pr...	Buying F...	Expiry D...	Quantity	Unit Name	Re Ord...	Current Worth	Spent Cost	Profit
ANTELMINTICS	PYRATIN S...	AN11535	ANURADH...	175.00	149.33	08/01/2016	- 3	- 10ML	Z	525.00	448.65	76.35						
		AN8603	ISHARA PH...	175.00	157.50	10/02/2016	- 4	- 10ML	Z	700.00	680.00	70.00						
	PYRATIN SYRUP 10ML Total									1,225.00	1,079.65	146.35						
	ZENTEL - T...	AN8503	RAGAMA P...	75.00	64.66	04/01/2017	+ 10	- MG	100	750.00	646.60	103.40						
ANTELMINTICS Total										2,035.00	1,777.45	257.55						
- ANTIBIOTICS										190,318.57	163,876.58	24,441.99						
- BABY CARE										26,827.16	23,120.58	3,706.58						
- CNS										60,192.45	50,194.60	9,937.85						
- CONTRACEPTIVE										1,330.00	1,080.00	270.00						
- COSMETICS										178,238.00	154,238.42	23,999.56						
- CREAM & LOTION										125,200.39	108,005.00	17,195.36						
- DIABETICS																		

Figure 84 filter by any columns

All generated and customized reports can be printed at the time, and by the customize print option users can make the page customization as normal word document.

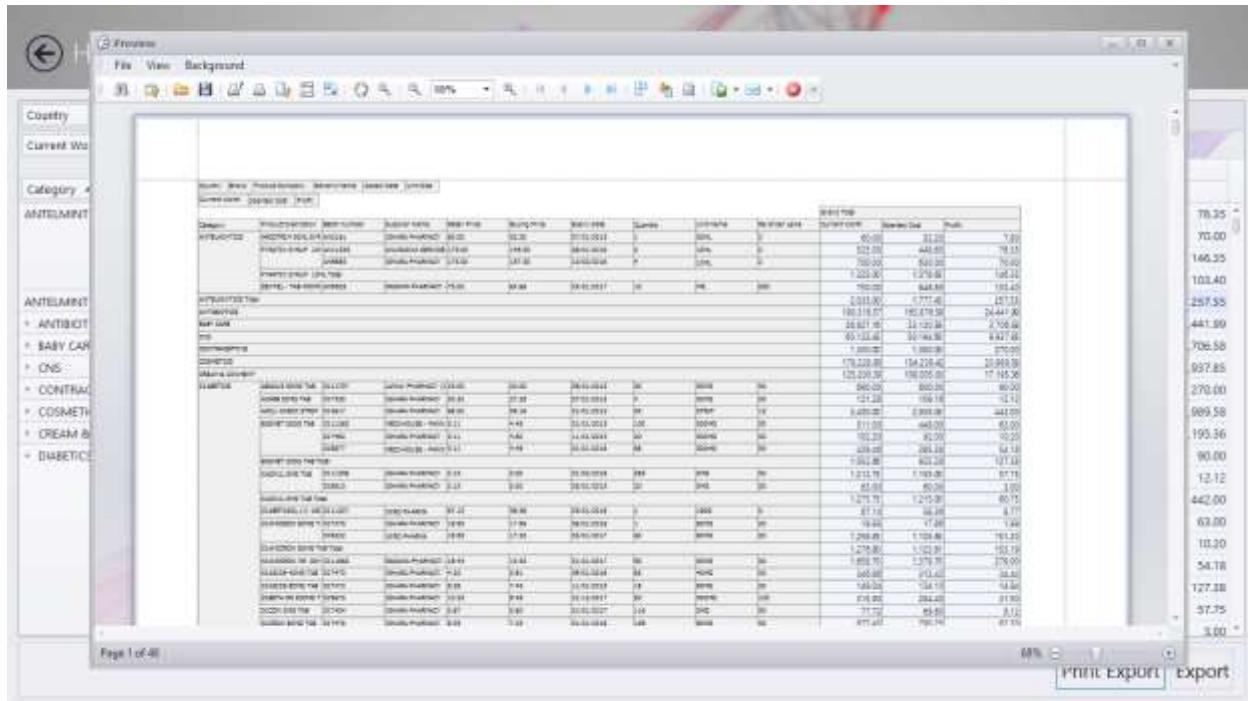


Figure 85 Print report

All the reports can export in several standard document formats, like PDF, MS Word, and MS Excel.

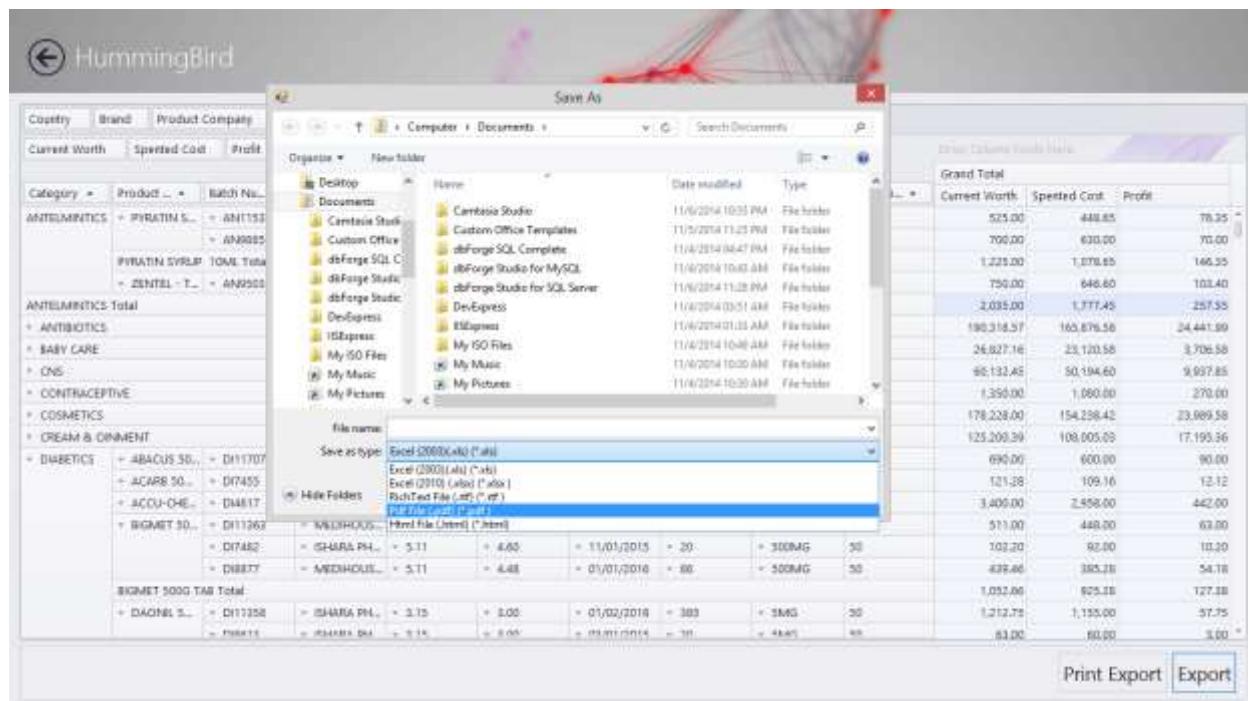


Figure 86 direct Export

5.5 layered application architecture

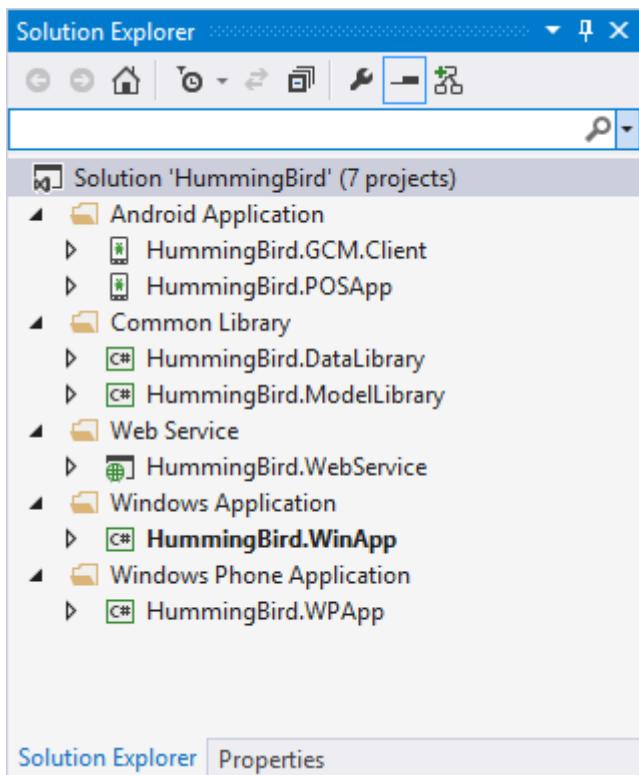


Figure 87 Solution Explorer

5.5.1 The User Layers/Libraries in the application

5.5.1.1 *HummingBird.DataLibrary*

The Main **Data Layer** in the application. All Data transaction and all methods that responsible to the Data base has been developed in the HummingBird.DataLibrary

The all classed extended by the BaseConnection.cs class that related to Database Connection.

And all classes that responsible to specific task has been developed with all the methods that responsible to the current assign work and the body of specific class has been developed with commonly make a transaction with database using **Stored Procedure** technology and **ADO.Net**.

Special Development technologies in the **HummingBird.DataLibrary** (Data Layer)

The application has been developed by using **Layered architecture**. Mobile Application, Web Service and all the main application are developed by using **Layered architecture**.

In the Application as the image (Figure 88 Solution Explorer) the application has been divided in to separate tasks by allocating specific module to one library.

Basically to each main parts of the application related with

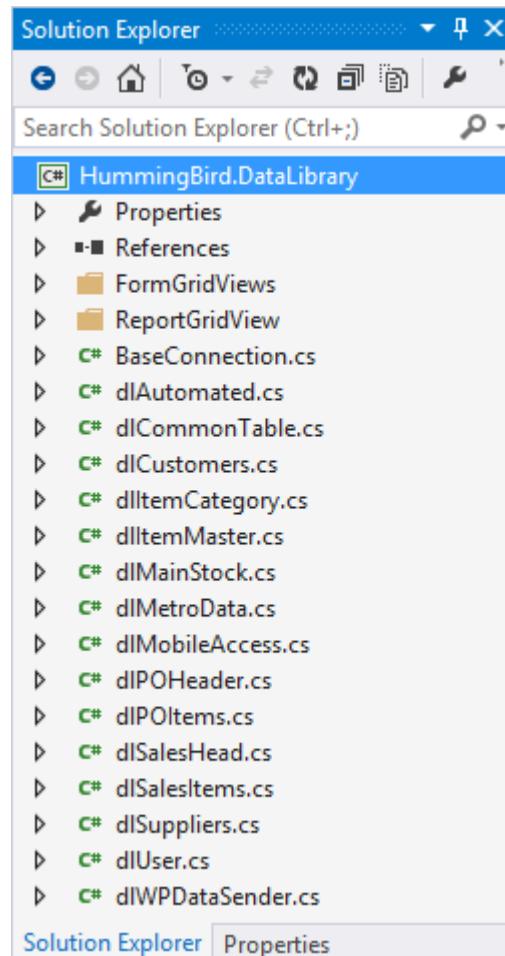
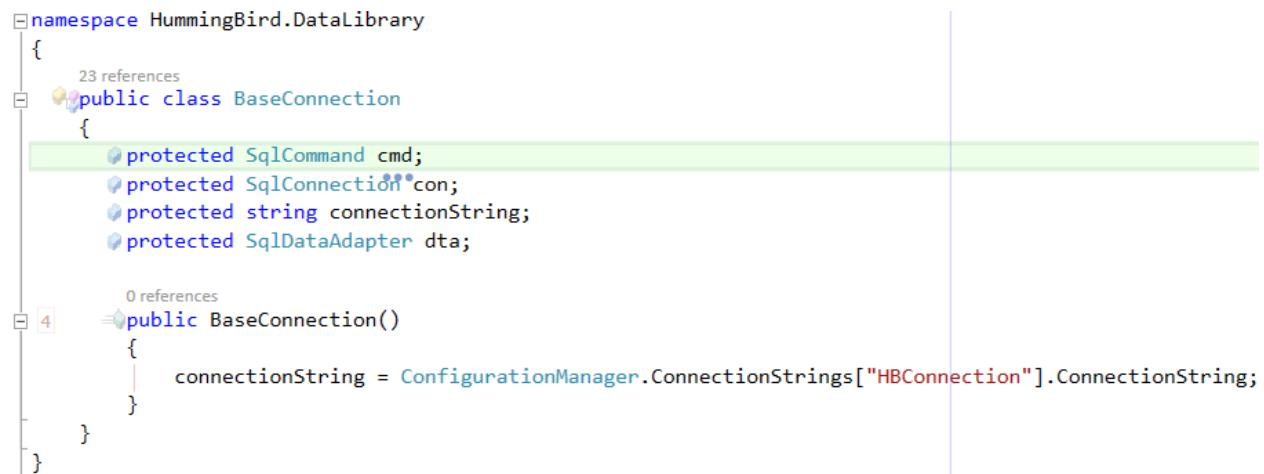


Figure 89 HummingBird.DataLibrary View

BaseConnection.cs (Database Connection)

The class is the responsible class for the Database connection. As the Database Connection technology the ADO.Net technology has been used.



```
namespace HummingBird.DataLibrary
{
    public class BaseConnection
    {
        protected SqlCommand cmd;
        protected SqlConnection con;
        protected string connectionString;
        protected SqlDataAdapter dta;

        public BaseConnection()
        {
            connectionString = ConfigurationManager.ConnectionStrings["HBConnection"].ConnectionString;
        }
    }
}
```

Figure 90 Database Connection Class

The Database connection is using the **app.config** configuration file to read the Database Connection, and the Database connection string has been created in the **app.config** file.

By creating the **app.config** file to store the database connection developer or administrator can be able to change, manage the connection string without changing the Application or source codes.

```
<connectionStrings>
    <add name="HBConnection"
        connectionString="Data Source=.\SQLEXPRESS;Initial Catalog=HummingBird;Integrated Security=True"
        providerName="System.Data.SqlClient" />
</connectionStrings>
```

Figure 91 Database Connection string in app.config

And from the base connection, main SQL server related properties has been provided to re-use and minimize the code usage.

```
protected SqlCommand cmd;
protected SqlConnection con;
protected string connectionString;
protected SqlDataAdapter dta;
```

Figure 92 Created common properties to use in every classes

In the Data Layer all other classes has been **inherited** by the BaseConnection.cs (Database Connection)

```

dlItemCategory    HB_ItemCategory's data access class
4 references
public class dlItemCategory : BaseConnection
{
    InsertItemCategory    Method responsible to insert HB_ItemCategory objects into the database
    1 reference
    public bool InsertItemCategory(mdItemCategory ct)
    {
        con = new SqlConnection(connectionString);
        try
        {
            con.Open();
            cmd = new SqlCommand("sp_HB_ItemCategory_IUD", con);
            cmd.CommandType = CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("@RunType", 1);
            cmd.Parameters.AddWithValue("@ItemCategoryID", ct.ItemCategoryID);
            cmd.Parameters.AddWithValue("@Category", ct.Category);
            cmd.ExecuteNonQuery();
        }
        catch (SqlException)
        {
            return false;
        }
        finally
        {
            con.Close();
        }
        return true;
    }
}

```

Figure 93 Sample Inherited class from database connection class

`using HummingBird.ModelLibrary;` As well as the HummingBird.DataLibrary (Data Layer) is
`using System.Collections;` using the HummingBird.ModelLibrary as a reference.
`using System.Data;`
`using System.Data.SqlClient;` And using the specific created object class to use the all
 attributes the created in the **HummingBird.ModelLibrary**,

```

public bool InsertItemCategory(mdItemCategory ct)
{
    con = new SqlConnection(con
    try
    {
        con.Open();
        cmd = new SqlCommand("sp_HB_ItemCategory_IUD", con);
        cmd.CommandType = CommandType.StoredProcedure;
        cmd.Parameters.AddWithValue("@RunType", 1);
        cmd.Parameters.AddWithValue("@ItemCategoryID", ct.ItemCategoryID);
        cmd.Parameters.AddWithValue("@Category", ct.Category);
        cmd.ExecuteNonQuery();
    }
}

```

Figure 95 Usage of Referenced class

Common Usage of the Classes in the HummingBird.DataLibrary (Data Layer)

The Data Layer is using by the two main application in the project

- HummingBird.WebService (Web Service)
- HummingBird.WinApp (Windows Application)

And the two of Android and Windows Phone mobile applications are using Data layer true the HummingBird.WebService (Web Service).

For the all aspect of connection and transaction with the database, by all the application that represent the presentation layer (Windows application, Mobile application).

In the Data Layer all the classes has been assigned to do a specific task of the main applications and web service.

Sample Tasks that created in the Data Layer

In the application the **Insert and Update** task has been assign to one method in a specific class to re-usability and minimize the code usability. Other delete and select processes handled by each methods and methods has been changed according to the requirement. The all database transaction are handled by the **Stored Procedures** (please refer the 4.8 Stored Procedures and Views) and all the update and insert processes are hdeled by one condition in each **Stored Procedures** in database. All the **Stored Procedures** that used in the development has been developed by using **specific and unique architecture for the application**.

Insert/ Update Method

```
public bool InsertItemMaster(mdItemMaster mdl)
{
    con = new SqlConnection(connectionString);
    try
    {
        con.Open();
        cmd = new SqlCommand("sp_HB_ItemMaster_IUSD", con) { CommandType = CommandType.StoredProcedure };
        cmd.Parameters.AddWithValue("@RunType", 1);
        cmd.Parameters.AddWithValue("@ProductCode", mdl.ProductCode);
        cmd.Parameters.AddWithValue("@ProductDiscription", mdl.ProductDiscription);
        cmd.Parameters.AddWithValue("@ItemCategoryID", mdl.ItemCategoryID);
        cmd.Parameters.AddWithValue("@SuppliersID", mdl.SuppliersID);
        cmd.Parameters.AddWithValue("@ProductCompany", mdl.ProductCompany);
        cmd.Parameters.AddWithValue("@GenericName", mdl.GenericName);
        cmd.Parameters.AddWithValue("@Country", mdl.Country);
        cmd.Parameters.AddWithValue("@UnitName", mdl.UnitName);
        cmd.Parameters.AddWithValue("@UnitSize", mdl.UnitSize);
        cmd.Parameters.AddWithValue("@Brand", mdl.Brand);
        cmd.Parameters.AddWithValue("@ReOrderLevle", mdl.ReOrderLevle);
        cmd.ExecuteNonQuery();
        return true;
    }
    catch (SqlException ex)
    {
        mdl.iMessage = ex.Message;
        return false;
    }
    finally
    {
        con.Close();
    }
}
```

Figure 96 Sample Insert/Update Method

In the method as the main properties like SqlCommand, SqlConnection, connectionString, SqlDataAdapter has been used from the inherited BaseConnection.cs class and the code re usage has been represented.

As well and the Modeler Layer HummingBird.ModelLibrary has been used in the each method the required to use the parameters and as the data communications.

Select and Delete Methods

```
public ArrayList SelectItemDescriptionList()
{
    var ItemList = new ArrayList();
    try
    {
        con = new SqlConnection(connectionString);
        cmd = new SqlCommand("sp_GV_ItemMaster_S", con) { CommandType = CommandType.StoredProcedure };
        cmd.Parameters.AddWithValue("@RunType", 1);
        con.Open();
        var dr = cmd.ExecuteReader();

        while (dr.Read())
        {
            ItemList.Add(dr[0]);
            ItemList.Add(dr[1]);
        }
    }
    catch (SqlException)
    {
        throw; ⏪
    }
    finally
    {
        con.Close();
    }
    return ItemList; ↵
}
```

Figure 97 Sample Selection Class (Array List)

```
public DataTable SelectItemGenericInfo()
{
    var dts = new DataTable();
    try
    {
        con = new SqlConnection(connectionString);
        dta = new SqlDataAdapter("sp_HB_ItemMaster_IUSD", con);
        dta.SelectCommand.CommandType = CommandType.StoredProcedure;
        dta.SelectCommand.Parameters.AddWithValue("@RunType", 7);

        con.Open();
        dta.Fill(dts);
    }
    catch (SqlException)
    {
        throw; ⏪
    }
    finally
    {
        con.Close();
        dta.Dispose();
    }
    return dts; ↵
}
```

Figure 98 Sample Selection Class (Data Table)

```

public string GetProductCode(mdItemMaster mdl)
{
    con = new SqlConnection(connectionString);
    string PC = "";
    try
    {
        con.Open();
        cmd = new SqlCommand("sp_HB_ItemMaster_IUSD", con) { CommandType = CommandType.StoredProcedure };
        cmd.Parameters.AddWithValue("@RunType", 8);
        cmd.Parameters.AddWithValue("@ProductDiscription", mdl.ProductDiscription);
        var sdr = cmd.ExecuteReader();

        while (sdr.Read())
        {
            PC = sdr["ProductCode"].ToString();
        }
    }
    catch (SqlException)
    {
        throw; ⏪
    }
    finally
    {
        con.Close();
    }
    return PC; ↵
}

```

Figure 99 Sample Selection Class (Single type)

 DeleteItemMaster Method responsible to delete HB_ItemMaster objects from a database

1 reference

```

public bool DeleteItemMaster(mdItemMaster mdl)
{
    con = new SqlConnection(connectionString);
    try
    {
        con.Open();
        cmd = new SqlCommand("sp_HB_ItemMaster_IUSD", con) { CommandType = CommandType.StoredProcedure };
        cmd.Parameters.AddWithValue("@RunType", 4);
        cmd.Parameters.AddWithValue("@ProductCode", mdl.ProductCode);
        cmd.ExecuteNonQuery();
        return true; ↵
    }
    catch (SqlException ex)
    {
        mdl.iMessage = ex.Message;
        return false; ↵
    }
    finally
    {
        con.Close();
    }
}

```

Figure 100 Sample Delete Class (Single type)

5.5.1.2 HummingBird.ModelLibrary

The Main **Modeler Layer** in the application. All modelers that responsible to the data layer and Application has been developed in the HummingBird.ModelLibrary

In the modeler library all the **Accessors** of the specific modeler class.



The screenshot shows a code editor window with the following content:

```
mdPOHeader Data model for HB_POHeader table
14 references
public class mdPOHeader
{
    0 references
    public int RunType { get; set; }
    0 references
    public int ID { get; set; }

    7 references
    public string PONumber { get; set; }

    2 references
    public int SuppliersID { get; set; }

    0 references
    public DateTime PODate { get; set; }

    4 references
    public string Finalize { get; set; }

    3 references
    public decimal Total { get; set; }

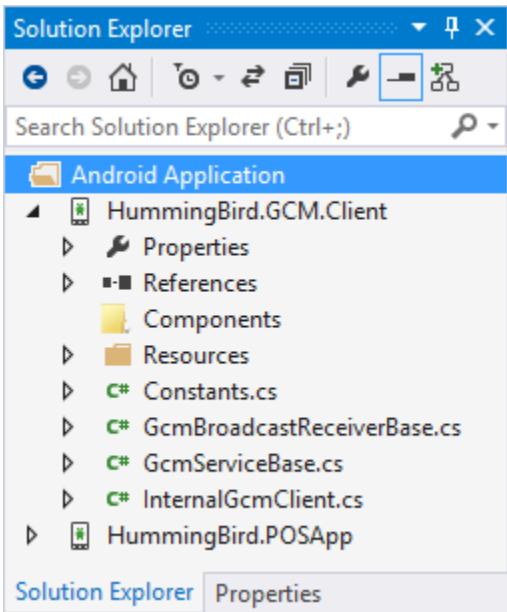
    2 references
    public DateTime PaymentDueDates { get; set; }

    3 references
    public string Payid { get; set; }

    7 references
    public string iMessage { get; set; }
}
```

Figure 101 Accessors in the modeler layer

5.5.1.3 HummingBird.GCM.Client



The Layer using to Connect with **the Google Cloud Messaging API (GCM) Helper Library** related to the Android application

The HummingBird.GCM.Client is specific to the Android application. And created to use the android push notification. The class library has been **Re-created** according to the java based library (gcm.jar) that using java android development. To use with the C# and Cross platform technology.

The library is providing the server side logics the using to handle the Google Cloud Messaging services.

The library is providing the connectivity with **Google Play Service** to the developed application as the live

service that running on the application installed android device.

Implementation of the HummingBird.GCM.Client has been included in to the Appendices of the project under the Development capture and HummingBird.GCM.Client please refer for the mode information.

5.5.4 All implemented layered that relations with each main application.

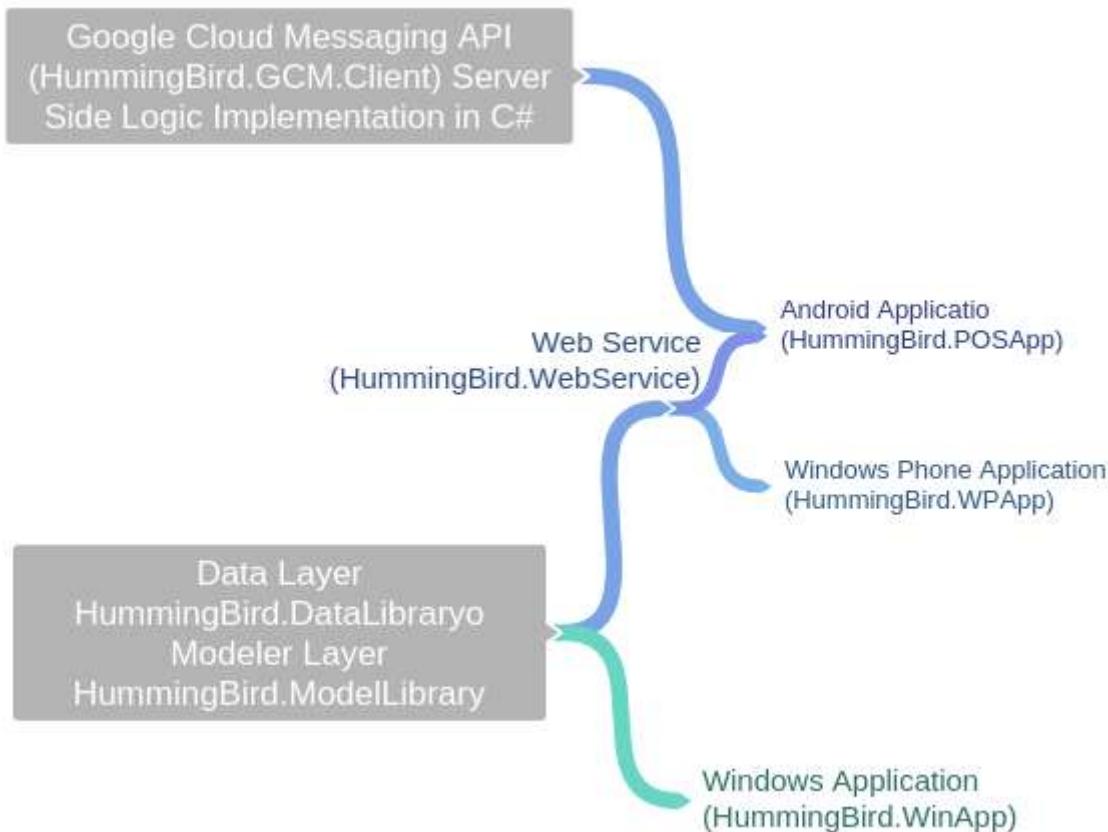


Figure 102 implemented layered that relations with each main application

- Windows Application (HummingBird.WinApp)
 - HummingBird.DataLibrary
 - HummingBird.ModelLibrary
- Web Service
 - HummingBird.DataLibrary
 - HummingBird.ModelLibrary
- Mobile Application (Android)
 - HummingBird.GCM.Client
 - And true the Web Service the HummingBird.DataLibrary and HummingBird.ModelLibrary has been referenced virtually.

5.6 Coding Structures and Techniques

5.6.1 Notification API for Windows and Android Mobile application

5.6.1.2 Create the API Project

By accessing the <https://cloud.google.com/console> web page and sign in with the Google ID after the process user can be create a project on the Google

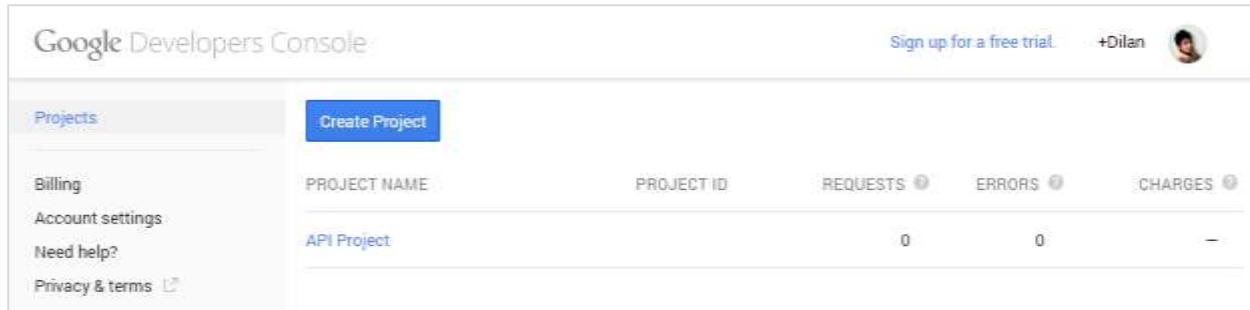


Figure 103 Google Console Create Project

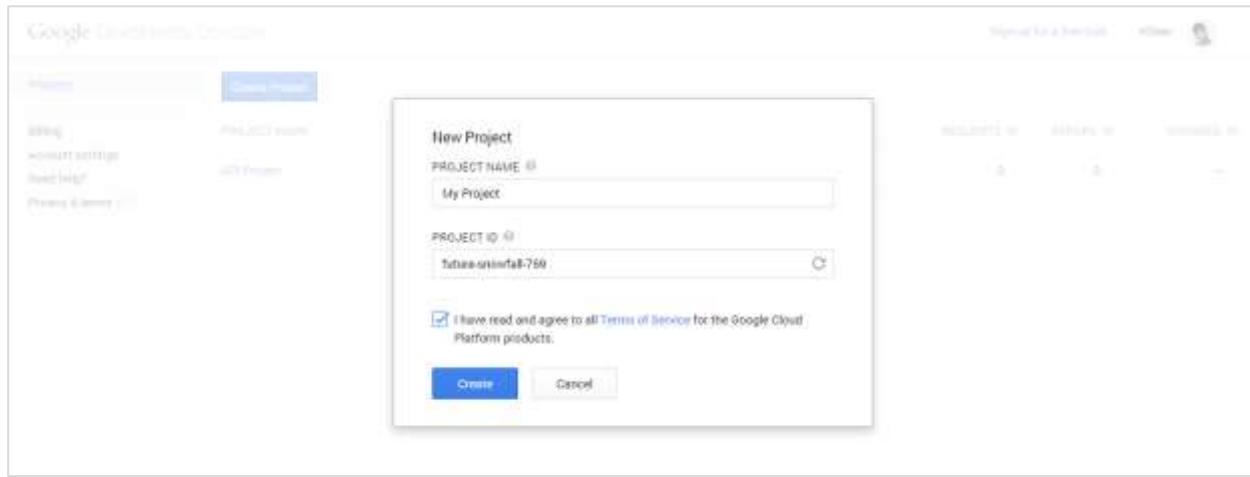


Figure 104 Google Console Create Project

In the Dashboard user can be enable the required project types and features the required to the API by clicking on the Enable API

Figure 105 Enable API

By clicking on the On/Off Button users can enable the required features

NAME	QUOTA	STATUS
Google Cloud Messaging for Android		ON

NAME	QUOTA	STATUS
Ad Exchange Buyer API	1,000 requests/day	OFF
Ad Exchange Seller API	10,000 requests/day	OFF
Admin SDK	150,000 requests/day	OFF
AdSense Host API	100,000 requests/day	OFF

Figure 106 Enable API features

As the requirement of the sending notification the Google Cloud Messaging for Android API has been enabled.

To ovation as API Key user need to access to the **APIs & auth and Credentials**, and by clicking the **Create New Key** under the **Public API Access** users can generate the Key

After Creating the API we can make the application with the notification (as we created API and Key)

Decide that Google-provided GCM association server that we need to use (HTTP) hypertext transfer protocol or XMPP (CCS). GCM association servers take messages from a 3rd-party application server (written by you) and send them to a GCM-enabled golem application (the "client app," additionally written by you) running on a tool.

Implement AN application server (the "3rd-party application server") to move together with your chosen GCM association server. The app server sends information to a GCM-enabled golem consumer application via the GCM association server. For a lot of data concerning implementing the server facet, see Implementing GCM Server. Make our consumer app. this can be the GCM-enabled golem application that runs on a tool.

The **Google Cloud Messaging API (GCM) Helper Library** that re-created for the cross platform technology, we have to use the API key to send and reserve the notifications.

```
namespace HummingBird.POSApp
{
    [BroadcastReceiver(Permission = Constants.PERMISSION_GCM_INTENTS)]
    [IntentFilter(new string[] { Constants.INTENT_FROM_GCM_MESSAGE }, Categories = new string[] { "@PACKAGE_NAME@" })]
    [IntentFilter(new string[] { Constants.INTENT_FROM_GCM_REGISTRATION_CALLBACK }, Categories = new string[] { "@PACKAGE_NAME@" })]
    [IntentFilter(new string[] { Constants.INTENT_FROM_GCM_LIBRARY_RETRY }, Categories = new string[] { "@PACKAGE_NAME@" })]
    2 references
    public class GcmBroadcastReceiver : GcmBroadcastReceiverBase<GcmService>
    {
        IMPORTANT: Change this to your own Sender ID!
        The SENDER_ID is your Google API Console App Project Number
        public static string[] SENDER_IDS = new string[] { "650463209487" };
    }
}
```



5.6.1 App 42 API notification API for Windows Phone

The API provider is www.apphq.shephertz.com

The screenshot shows the App 42 API Management Console dashboard. The left sidebar lists various services: AppHQ Home, App Manager (selected), Create App, App Dashboard, View App, Application Keys, Key Management, Method Authorization, Technical Service Manager, Business Service Manager, Push Notification, Custom Code, API Testing, Achievement Service, User Acquisition, Analytics Cloud, Appmark - Project Management, Billing & Invoicing. The main area is titled 'App List' with filters for 'App Type (All)' and 'Owner (All)'. It displays a table for 'My Apps' with one entry: 'WP42APP' (Status: Online, ACL: DISABLED, Owner: Me). Columns include App, ACL, Status, App State, Owner, Permission, Collaborators, Replay Attack Protect, and Analytics State. Below this is a section for 'Other Apps' which says 'No Apps.' At the bottom, there's a copyright notice: 'Copyright © 2014 Shephertz Technologies Pvt Ltd. All rights reserved. Terms & Conditions | Privacy Policy'.

Figure 107 App 42 API Dashboard

From the API that used to send the push notification with the development we have to use specific Library that the provided with the API, according to the technology.

Download SDKs in Single Step



Figure 108 Select library to download

As the requirement the client application need the reserving features and the server application need a broadcasting library by connecting the API Keys that provided by the Service.

5.7 Stored Procedures and Views

5.7.1 Stored Procedures

In the application all the database transactions has been handled with the Stored Procedures and views, the data layer of the application is connected to the stored procedures.

Using Stored Procedure benefits can achieve as explained below Inactive

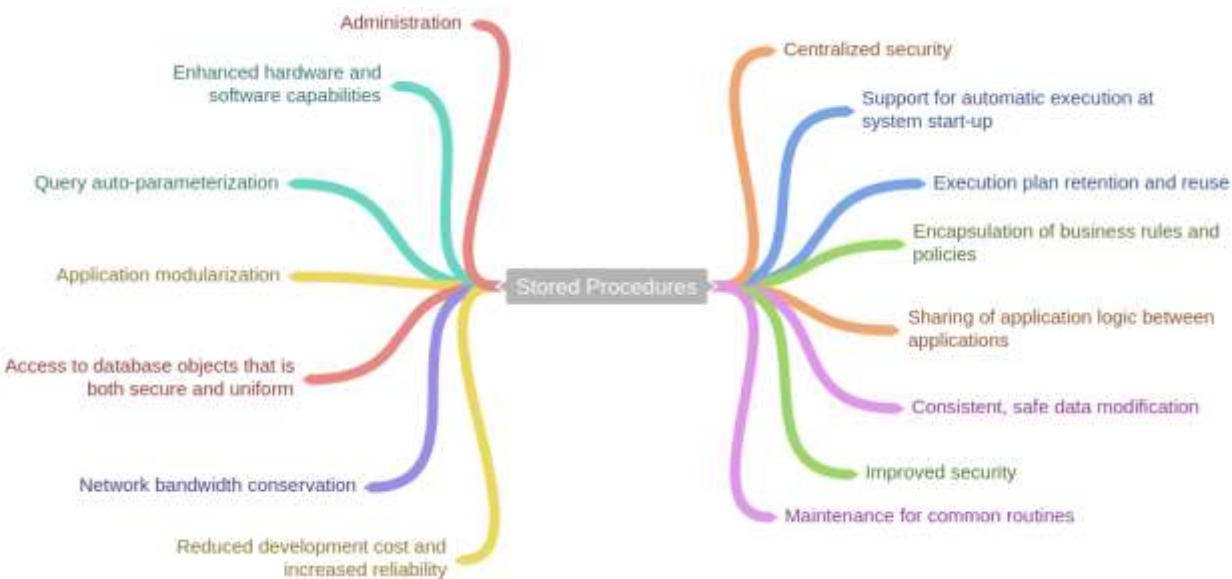


Figure 109 Stored Procedures

In the application, the database technology has been specialized with the stored procedures. By using the stored procedure the visible application layers has been Significant to some special task as well as the database. The database side also playing a big task regarding the data transferring and doing the application side tasks faster than application process.

To accomplish the tasks using the stored procedures, a specific and unique methodology has been used starting from designing stored procedures and development of the stored procedure. According to the developed stored procedures, the database side also a playing an application level task. In the application from the specialized stored procedure architecture, re usability has been achieved.

5.7.1.1 Development of Stored Procedures

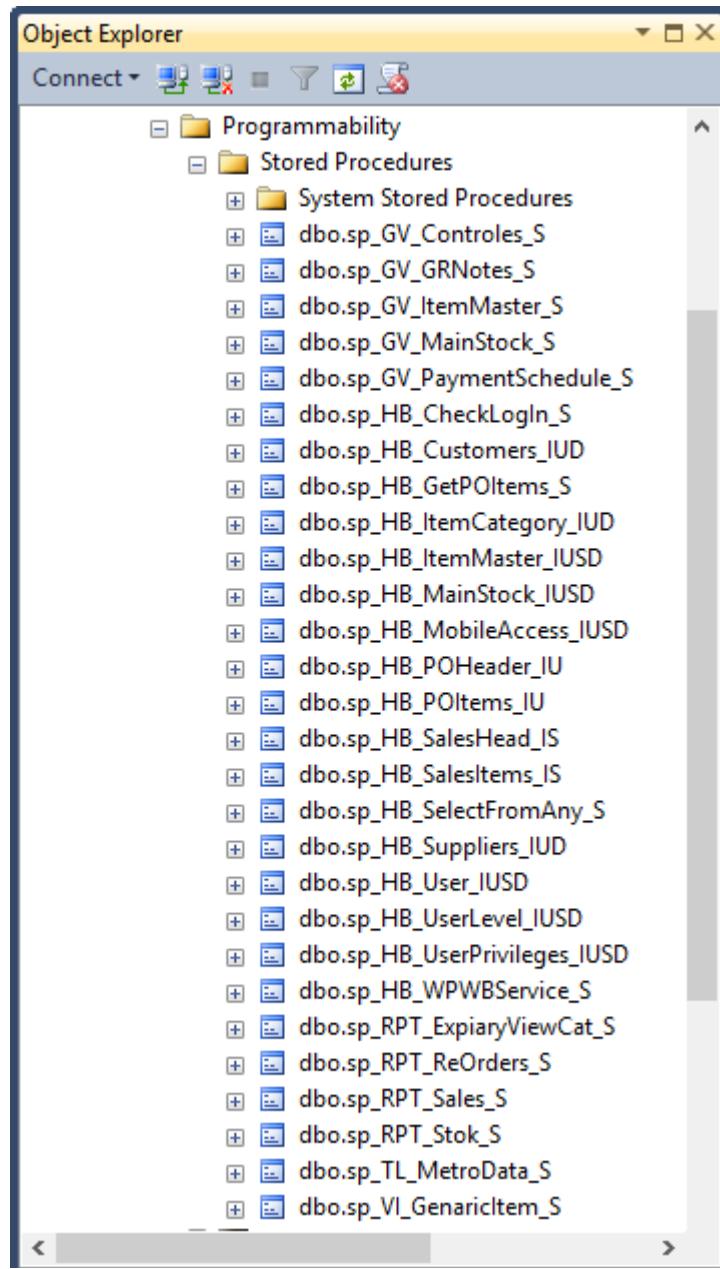


Figure 110 Stored Procedures

The stored procedures are modularized to do a specific task range, as an example, the Supplier table's **insert, update, select** tasks are given to one specific Stored Procedure in the database.

As the Naming of the stored procedures of the application, the specific prefix has been used to identify the stored procedure's task.

- **_I** - Representing the Stored procedure belongs to Insert statements

- **U** - Representing the Stored procedure belongs to Update statements
- **S** – Representing the Stored procedure belongs to Select statements
- **D** - Representing the Stored procedure belongs to Delete statements

According to the prefix usage,

_IUSD prefixed stored procedure is handling all the statements of the specified table

5.7.1.2 Common Stored Procedure architecture

```

CREATE PROCEDURE dbo.sp_HB_PROCEDURE_IUSD
(
    @RunType      INT,
    -- Parameters
)
AS
    SET NOCOUNT ON
    BEGIN TRY
        BEGIN TRANSACTION

        IF @RunType = 1
            BEGIN
                -- Task Under the @RunType = 1
            END

        IF @RunType = 2
            -- Task Under the @RunType = 2
        END

        IF @RunType = 3
            BEGIN
                -- Task Under the @RunType = 3
            END
        COMMIT TRANSACTION
    END TRY

    BEGIN CATCH
        ROLLBACK TRANSACTION;
        THROW;
    END CATCH
GO

```

Figure 111 Stored Procedure Architecture

From the architecture as present above (Figure 112 Stored Procedure Architecture) the procedure main task handler is the **@RunType** parameter when develop the application if developer need to achieve specific task from the stored procedure, the parameter should be supplied with the expectation of stored procedure with the used code.

5.7.1.3 C# Data Layer Code usage with the stored procedure

As the significant stored procedure architecture the coding part has been coated with the task handler parameter (**@RunType**) that created in the stored procedure to execute specific task.

The diagram illustrates the relationship between a C# method and its corresponding SQL stored procedure. A red bracket on the left side of the C# code maps to the parameter definition in the stored procedure's script. Another red bracket on the right side of the C# code maps to the conditional logic within the stored procedure's body.

```
public bool InsertCustomer(mdCustomers cu)
{
    con = new SqlConnection(connectionString);
    try
    {
        con.Open();
        cmd = new SqlCommand("sp_HB_PROCEDURE_IUSD", con) { CommandType = CommandType.StoredProcedure };
        cmd.Parameters.AddWithValue("@RunType", 1);
        cmd.ExecuteNonQuery();
    }
    catch (SqlException ex)
    {
        cu.iMessage = ex.Message;
        return false;
    }
    finally
    {
        con.Close();
    }
    return true;
}
```

Stored Procedure

```
CREATE PROCEDURE dbo.sp_HB_PROCEDURE_IUSD
(
    @RunType     INT,
    -- Parameters
    AS
    SET NOCOUNT ON
    BEGIN TRY
        BEGIN TRANSACTION

        IF @RunType = 1
            BEGIN
                -- Task Under the @RunType = 1
            END
        COMMIT TRANSACTION
    END TRY

    BEGIN CATCH
        ROLLBACK TRANSACTION;
        THROW;
    END CATCH
    GO
```

Figure 113 C# Data Layer Code usage with the stored procedure

By calling the stored procedure with specific task handler parameter (**@RunType**) the targeted statement can be executed.

5.7.1.4 The Significant Insert statement that used in the stored procedures.

In the application data layer insert and update methods are working as a same function, to manage the deferens between the insert and update functions, the stored procedure technology has been used. From the provided stored procedure with significant inset and update mode, the Update button requirement has been ignored, and system getting easier than traditional applications. As the problem statement and all the development, the re-usability, minimum work space to do lot of works and malty task in user side and development engineering aspects has been archived from using that kind of technologies.

Insert statement in the stored procedure.

```
CREATE PROCEDURE dbo.sp_HB_PROCEDURE_IUSD
(
    @RunType    INT,
    @ID         INT
)
AS
    SET NOCOUNT ON
    BEGIN TRY
        BEGIN TRANSACTION

        IF @RunType = 1
        BEGIN
            IF EXISTS (SELECT
                1 AS Exist
                FROM   Table
                WHERE  [ID] = @ID)
                BEGIN
                    UPDATE   Table
                    SET     Field ans Value
                    WHERE  [ID] = @ID
                END
            ELSE
                BEGIN
                    INSERT INTO   Table
                    VALUES      VALUES
                END
        END
        COMMIT TRANSACTION
    END TRY
    BEGIN CATCH
        ROLLBACK TRANSACTION;
        THROW;
    END CATCH
GO
```

The code snippet shows a T-SQL stored procedure named sp_HB_PROCEDURE_IUSD. It takes two parameters: @RunType (INT) and @ID (INT). The procedure starts by setting NOCOUNT ON. It then begins a TRY block, which contains a BEGIN TRANSACTION statement. Inside this block, there is a conditional statement: IF @RunType = 1. If true, it checks if the ID exists in the Table using an IF EXISTS query. If it does, it performs an UPDATE operation; otherwise, it performs an INSERT operation. Both the UPDATE and INSERT statements set the Field to its Value where the ID matches the parameter @ID. After the IF block, the transaction is committed. Finally, the CATCH block handles any errors that occurred during the transaction, and the procedure ends with GO.

Checking whether the ID exist or not

If Exist then Updating the Recoard

If not Exist the Record is Inserting

Figure 114 Significant Insert statement that used in the stored procedures.

In the stored procedure from the statement, first checking whether any record exist according to the passed parameter ID and if exist any record according to the passed ID parameter, then statement is continuing with updating part, if not exist any record according to the passed ID parameter, the statement is connecting with Inserting the record to table.

In the application the reusable insert update method has been developed according to the special stored procedure architecture, the ID is not visible or editable interfaces in the application, because of that the parameter ID is not duplicating or getting wrong, only the IS using when editing the record in the Application.

The discussed significant inserting updating methods is handling the one button to do two tasks at once. And minimize the unwanted code usage in the classes. As well as debugging purposes also minimizing from the above technology discussed above.

Please refer (Appendix B: 3) for the coding that used in this phase

5.7.1.5 Sample Stored Procedure that used in the application

```

ALTER PROCEDURE [dbo].[sp_HB_MainStock_IUSD]
(
    @RunType      INT,
    @StockID      INT          = NULL,
    @ProductCode  INT          = NULL,
    @ExpiryDate   DATETIME    = NULL,
    @RetailPrice   MONEY        = NULL,
    @BuyingPrice  MONEY        = NULL,
    @Quantity     INT          = NULL,
    @AddedDate    DATETIME    = NULL,
    @BatchNumber  NVARCHAR(50) = NULL,
    @PONumber     NVARCHAR(10) = NULL,
    @Hold         NVARCHAR(3)  = NULL
)
AS
    SET NOCOUNT ON
    BEGIN TRY
        BEGIN TRANSACTION

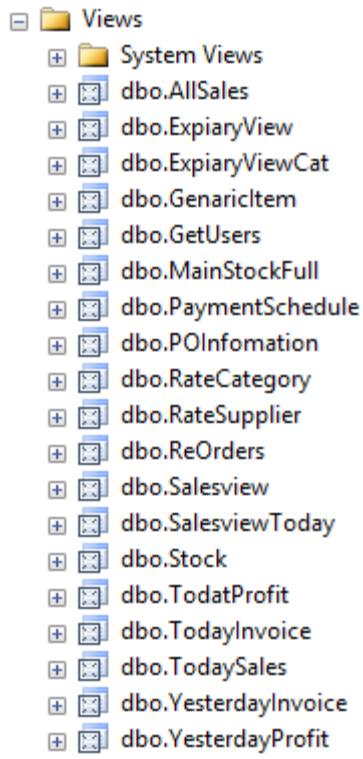
        IF @RunType = 1
        BEGIN
            IF EXISTS (SELECT
                1
            FROM HB_MainStock
            WHERE [StockID] = @StockID)
            BEGIN
                UPDATE HB_MainStock
                SET [ProductCode] = COALESCE(@ProductCode, ProductCode),
                    [ExpiryDate] = COALESCE(@ExpiryDate, ExpiryDate),
                    [RetailPrice] = COALESCE(@RetailPrice, RetailPrice),
                    [BuyingPrice] = COALESCE(@BuyingPrice, BuyingPrice),
                    [Quantity] = COALESCE(@Quantity, Quantity),
                    [BatchNumber] = COALESCE(@BatchNumber, BatchNumber),
                    [Hold] = COALESCE(@Hold, Hold)
                WHERE [StockID] = @StockID
            END

            ELSE
            BEGIN
                INSERT INTO HB_MainStock ([ProductCode]
                , [ExpiryDate]
                , [RetailPrice]
                , [BuyingPrice]
                , [Quantity]
                , [BatchNumber]
                , [PONumber]
                , [Hold])
                VALUES (@ProductCode, @ExpiryDate, @RetailPrice, @BuyingPrice, @Quantity,
                    @BatchNumber, NULLIF(@PONumber, ''), NULLIF(@Hold, ''))
            END
        END
        ELSE
        BEGIN
            SELECT
                [StockID],
                [ProductCode],
                [ExpiryDate],
                [RetailPrice],
                [BuyingPrice],
                [Quantity],
                [AddedDate],
                [BatchNumber],
                [Hold]
            FROM HB_MainStock
        END
    END TRY
    BEGIN CATCH
        ROLLBACK TRANSACTION
    END CATCH

```

Figure 115 Sample Stored Procedure that used in the application

5.7.2 Views



In the application to reduce the query usage and re use the queries and minimize the complex queries in every selection, the Views has been created in the database.

In the application mainly views has been used for

- To analyzing purposes
- Make reports
- Re usable Queries
- Minimize the query length in stored procedure's select statements
- Generate the optimized queries to multiple usage

Figure 116 Views

5.7.2.1 Sample View that created in the database

Please refer (Appendix B: 4) for the coding that used in this phase

```

SELECT DISTINCT
    im.ProductDescription,
    SUM(ms.Quantity) AS Quantity,
    im.ReOrderLevle,
    ct.Category,
    ISNULL
    ((SELECT
        sp.SupplierName
    FROM dbo.HB_POHeader AS ph
    INNER JOIN dbo.HB_Suppliers AS sp
        ON ph.SuppliersID = sp.SuppliersID
    WHERE (ph.PONumber = MAX(ms.PONumber))), (SELECT
        sp.SupplierName
    FROM dbo.HB_ItemMaster AS im
    INNER JOIN dbo.HB_Suppliers AS sp
        ON im.SuppliersID = sp.SuppliersID
    WHERE (im.ProductCode = MAX(ms.ProductCode)))
    ) AS SupplierName,
    CONVERT(DECIMAL(32, 2), CONVERT(DECIMAL, SUM(ms.Quantity)) / CONVERT(DECIMAL, im.ReOrderLevle) * 100)
    AS OrderRange
FROM dbo.HB_MainStock AS ms
INNER JOIN dbo.HB_ItemMaster AS im
    ON ms.ProductCode = im.ProductCode
INNER JOIN dbo.HB_ItemCategory AS ct
    ON im.ItemCategoryID = ct.ItemCategoryID
GROUP BY    im.ProductDescription,
            im.ReOrderLevle,
            ct.Category,
            im.ReOrderLevle

```

Figure 117 Sample View that created in the database

5.7.3 Triggers

In the tables, after inserting records or after update records, the triggers has been created to do a specific task according to the inserted or updated record. The triggers are doing a role more than computer columns in the tables, and triggers are automated.

In the developed database, the triggers used in some tables to accomplished the automated tasks.

Used triggers in the database.

Sample trigger created in HB_MainStock to handle the product quantities

```
USE [HummingBird]
GO
***** Object: Trigger [dbo].[EndProduct]    Script Date: 11/20/2014 10:21:24 PM *****
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
ALTER TRIGGER [dbo].[EndProduct]
ON [dbo].[HB_MainStock]
AFTER UPDATE
AS
DECLARE @Quantity AS INT
DECLARE @BatchNumber NVARCHAR(50)

SELECT
    @Quantity = Quantity,
    @BatchNumber = BatchNumber
FROM INSERTED

IF @Quantity <= 0
BEGIN
    UPDATE HB_MainStock
    SET EndProduct = 1
    WHERE BatchNumber = @BatchNumber
END
ELSE
IF @Quantity >= 1
BEGIN
    UPDATE HB_MainStock
    SET EndProduct = 0
    WHERE BatchNumber = @BatchNumber
END
```

Figure 118 Used triggers in the database.

Stored procedure that made for check the user login

Please refer (Appendix B: 5) for the coding that used in this phase

5.7.4 Indexes

In the developer database Indexes has been used to speed up the performance of queries, and selections. By indexing the tables, users can take high performance from database, and fast data search, sort and views.

```
CREATE UNIQUE INDEX IX_HB_MainStock
ON HummingBird.dbo.HB_MainStock (BatchNumber)
ON [PRIMARY]
GO
```

Figure 119 index

5.8 Technical and significant software engineering techniques in each modules

5.8.2 Windows Application

5.8.2.1 Voice Commands/Self Assistant

```
public void initializeSpeech()
{
    const string cmdFile = @"commands.txt";
    try
    {
        gbuilder.Append(new Choices(File.ReadAllLines(cmdFile)));
    }
    catch (Exception ex)
    {
        MessageBox.Show(
            String.Format(
                "The 'Commands' file must not contain empty lines.\n{0}", ex.Message
            ), "Error", MessageBoxButtons.OK, MessageBoxIcon.Error);
        pr.StartInfo.FileName = cmdFile;
        pr.Start();
        Application.Exit();
        return; ↵
    }
    var gr = new Grammar(gbuilder);
    try
    {
        sRecognize.UnloadAllGrammars();
        sRecognize.RecognizeAsyncCancel();
        sRecognize.RequestRecognizerUpdate();
        sRecognize.LoadGrammar(gr);
        sRecognize.SpeechRecognized += sRecognize_SpeechRecognized;

        sRecognize.SetInputToDefaultAudioDevice();
        sRecognize.RecognizeAsync(RecognizeMode.Multiple);
    }

    catch (Exception ex)
    {
        MessageBox.Show("Grammar Builder Error" + ex.Message);
        return; ↵
    }
}
```

Figure 120 Methods that responsible for initialize the speech

The Windows Application has been voice command enables by using the Windows default speech recognition library and inbuilt hardcoded voice commands and events.

The Voice Commands has been developed to control all metro UI navigation buttons, by starting the self-assistant, users can navigate to each modules by pronouncing the module name. The

developed Self Assistant is limited service and cannot recognize all the human voice commands. And not responding to all commands, the specific commands are hardcoded in the text document inside of the script directory and application is reading the command file and if matching command found in the list then only the specific methods is firing according to the voice command.

Please refer (Appendix B: 6) for the coding that used in this phase

```

=◆ public void speakText(string textSpeak)
{
    sRecognize.RecognizeAsyncCancel();
    sRecognize.RecognizeAsyncStop();
    pBuilder.ClearContent();
    pBuilder.AppendText(textSpeak.ToString());
    sSynth.SelectVoice(Assistant);
    sSynth.SpeakAsync(pBuilder);
    sRecognize.RecognizeAsyncCancel();
    sRecognize.RecognizeAsyncStop();
    sRecognize.RecognizeAsync(RecognizeMode.Multiple);
}

1 reference
=◆ private void sRecognize_SpeechRecognized(object sender, SpeechRecognizedEventArgs e)
{
    switch (e.Result.Text)
    {
        case "start working":
            AssistanTile.Elements[0].Text = e.Result.Text.ToString();
            Swork = true;
            if (!started)
            {
                speakText(String.Format("Hello {0}. This is Hummingbird control center", iGlobal.Name));
                started = true;
            }
            break; ↴
    }

    if (Swork)
    {
        case "go to the point of sales":
        case "point of sales":
            AssistanTile.Elements[0].Text = e.Result.Text.ToString();
            windowsUIView.ActivateDocument(PointofSaleDocument);
            exitCondition = true;
            break; ↴

        case "go to the purchase order":
        case "purchase order":
            AssistanTile.Elements[0].Text = e.Result.Text.ToString();
            windowsUIView.ActivateDocument(PurchaseOrderDocument);
            break; ↴
    }
}

```

Figure 121 Module Navigation function

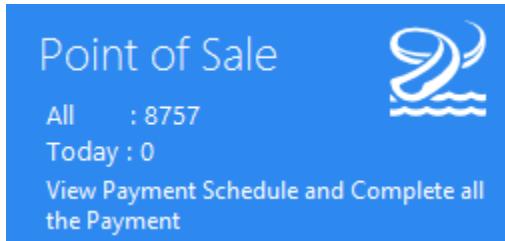
All the navigation handling using the switch, by pronouncing the proper words, users can take the correct outcome from the speech recognition module.

5.8.2.2 Tile information analyzing component

In the windows application, the Windows UI / Metro style menu containing the **Tiles** (as Microsoft Windows 8 start menu) and all the menus are containing the specific tasks and modules. In each tiles, employee can see specific information about the module contain.

By giving this features, the data analyzing part that main aim of the application that developing, has been accomplished as expected. Data analyzing part is the one of main and major scope of the project.

As the example for the tile data analyzing features



As you can see in the screenshot of the Point of Sales tile the today invoice count and tootle invoice count has been previewed.

Figure 122 Point of Sales tile



As well as single tile, the multiple tile frames are included to make the information more, according to the module, in the tile the best supplier has been appeared.

Figure 123 Supplier Tile



Figure 124 multiple slide frame for one module

As above in the multiple tile frame firsts aspiring Supplies and from the second tile, Item category's main information that representing the best sales item category.

For make the files as analyzing part of the application, a complex method has been created to get the information and set to the tiles' faces as meaningful and analyzable tile. As well as a specific Stored Procedure also created to accomplish the task.

Method that responsible to make the tiles analyzable.

Please refer (Appendix B: 6) for the coding that used in this phase

```
private void ITileInfo()
{
    var POSShe = dlMd.POSStileInfo();

    for (int i = 0; i < POSShe.Rows.Count; i++)
    {
        DataRow dr = POSShe.Rows[i];
        PointofSaleTile.Frames[0].Elements[2].Text = "All &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;" + dr["AllInvoice"];
        PointofSaleTile.Frames[0].Elements[3].Text = "Today : " + dr["TodayInvoice"];
    }

    var PORder = dlMd.PUOTileInfo();

    for (int i = 0; i < PORder.Rows.Count; i++)
    {
        DataRow dr = PORder.Rows[i];
        PurchaseOrderTile.Elements[i + 1].Text = dr["POOnWay"].ToString();
    }

    var PaymentShe = dlMd.PaymentScheduleTileInfo();

    for (int i = 0; i < PaymentShe.Rows.Count; i++)
    {
        DataRow dr = PaymentShe.Rows[i];
        PaymentScheduleTile.Elements[i + 1].Text = dr["Payment"].ToString();
    }

    var itemMas = dlMd.ItemMasterTileInfo();

    for (int i = 0; i < itemMas.Rows.Count; i++)
    {
        DataRow dr = itemMas.Rows[i];
        ItemMasterTile.Elements[i + 1].Text = dr["ProductDiscription"].ToString();
    }

    var StockRpt = dlMd.MainStockTileInfo();
```

Figure 125 Method that responsible to make the tiles analyzable

From the method taking data from the database true the data layer and presenting true the tiles face.

Responsible Stored Procedure to generate the analyzable data from tiles.

```

ALTER PROCEDURE [dbo].[sp_TL_MetroData_S]
(
    @RunType INT
)
AS
    -- Point Of Sales Invoices
    IF @RunType = 1
    BEGIN
        SELECT (SELECT
                    COUNT(InvoiceNumber)
                FROM HB_SalesHead)
                AS AllInvoice,
                (SELECT
                    COUNT(InvoiceNumber)
                FROM HB_SalesHead
                WHERE CONVERT(VARCHAR, InvoiceDate, 101) = CONVERT(VARCHAR, GETDATE(), 101))
                AS TodayInvoice
    END

    -- Purchase Order Count
    IF @RunType = 2
    BEGIN
        SELECT
            CONCAT(COUNT(hp.PONumber), ' Purchase Orders') AS POOnWay
        FROM HB_POHeader hp
        WHERE hp.Finalize = 'No'
        AND hp.Payid = 'No'
    END

    -- UpComing Payments
    IF @RunType = 3
    BEGIN
        SELECT TOP (3)
            CONCAT([month], ' ', [day], ' | ', [PONumber], ' | ', [Total]) AS Payment
        FROM [PaymentSchedule]
        ORDER BY CONVERT(DATE, PaymentDueDates) ASC
    END

    --Random Products
    IF @RunType = 4
    BEGIN
        SELECT TOP (5)

```

Figure 126 Responsible Stored Procedure to generate the analyzable data from tiles

Please refer (Appendix B: 7) for the coding that used in this phase

As the result of the tile analyzing component

<p>Point of Sale</p> <p>All : 8757 Today : 0</p> <p>View Payment Schedule and Complete all the Payment</p> 	<p>Item Categories</p> <p>VITAMIN & MINARALS 53.20 % HYPERTENTION DRUGS 41.00 %</p> <p>Manage Item Categories</p> 	<p>Expiry Report</p> <p>GA12504 DOLIUM SUSP. 30ML CR12505 DAIVOBET 15G OINTME CR3300 ZESTOPAQUE CREAM 30</p> 
<p>Purchase Order</p> <p> 1046 Purchase Orders</p>	<p>Item Master</p> <p>PROPRANOLOL 40... VITMIN E - (SPC) 4... HERBESSER 200MG... ATV-10 10MG TAB H&S (MEN HAIR RETAIN) SHAMP. 70ML</p> 	<p>Re Orders Report</p> <p>GLUCOLIN 100G PKT Q:... POLYTAR LIQ 65ML Q:1 ... TREVIA 100MG TAB Q:28 R:3</p> 
<p>Goods Receiving Note</p> 		<p>Sales Report</p> <p>Yesterday : 0.00 Today : 0.00</p> 
<p>Payment Schedule</p> <p> - - -</p>	<p>Stock Transfer</p> <p>Manage Stock and make immediate changes in Stock</p> 	<p>Stock Report</p> <p>ENFAMAMA (V) 40... HUMALOG MIX 50 ... JONAC GEL 30G</p> 

Figure 127 Full of Data analyzing Menu

5.8.2.3 Accessing to another form's public properties

Whole application is running under the main form, other controls embodied in to the user controls and all the user controls navigating inside the main form true the panel controls. And main form is reading some events of the public control in the user controls that running on the main form (Home).

By changing the access level of any control like button textbox or any control in a user control, can make the control to access from another form.

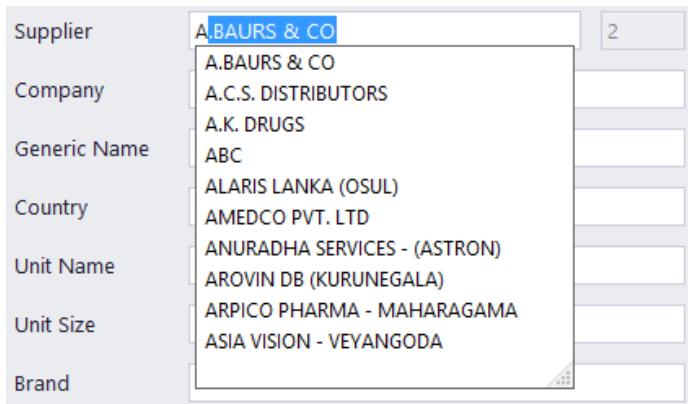
```
public partial class Home : XtraForm
{
    // objects created to access in to usercontrolse
    private ucLogin ucLog = new ucLogin();
    private dlUser dlUs = new dlUser();
    private mdUser mlUs = new mdUser();

    1 reference
    public Home()
    {
        InitializeComponent();
        panelControl.Controls.Add(ucLog);
        ucLog.Dock = DockStyle.Fill;
        ucLog.btLogin.Click += btLogin_Click;
        ucLog.btExit.Click += btExit_Click;
        ucLog.tbPassword.KeyDown += tbPassword_KeyDown;
    }

    // password key down in login
    1 reference
    void tbPassword_KeyDown(object sender, KeyEventArgs e)
    {
        if (e.KeyCode == Keys.Enter)
            CheckLogin();
    }
}
```

Figure 128 Accessing to another form's public properties

5.8.2.4 Mask Boxes



Mask boxes has been created for the auto load/ filter in any textbox, by typing the first letters in the text box users can take the subjected output from the textbox. Then users can select needed value from the textbox.

Figure 129 Mask Boxes

```
var txBatchNumber = tbBatchNumber.MaskBox;
txBatchNumber.AutoCompleteSource = AutoCompleteSource.CustomSource;
txBatchNumber.AutoCompleteMode = AutoCompleteMode.SuggestAppend;
var collectionBatch = new AutoCompleteStringCollection();
var bList = dlMstk.SelectBatch();
foreach (string lst in bList)
{
    collectionBatch.Add(lst);
}

txBatchNumber.AutoCompleteCustomSource = collectionBatch;
```

Figure 130 Mask Boxes Data Bind

By taking the wanted data from data layer to bind in to a textbox and storing and binding the list to the text box as the Mask box data source. Users can see the output from textbox.

5.8.2.5 Key down events in Point of Sales module

```
private void TB_Quantity_KeyDown(object sender, KeyEventArgs e)
{
    if (e.KeyCode == Keys.Enter && TB_Quantity.Text != "0" && TB_Quantity.Text != string.Empty
        && TB_Discount.Text != string.Empty &&
        ((Convert.ToInt32(TB_Quantity.Text)) <= (Convert.ToInt32(lB_Quantity.Text))))
    {
        AddToGrid();
    }
    else if (e.KeyCode == Keys.Right)
    {
        TB_Discount.Enabled = true;
        TB_Discount.Focus();
    }
    else if (e.KeyCode == Keys.Down)
    {
        TB_PaidAmount.Enabled = true;
        TB_PaidAmount.Focus();
    }
    else if (e.KeyCode == Keys.Escape)
    {
        EscFire();
    }
    else
    {
        TB_Quantity.Focus();
    }
}
```

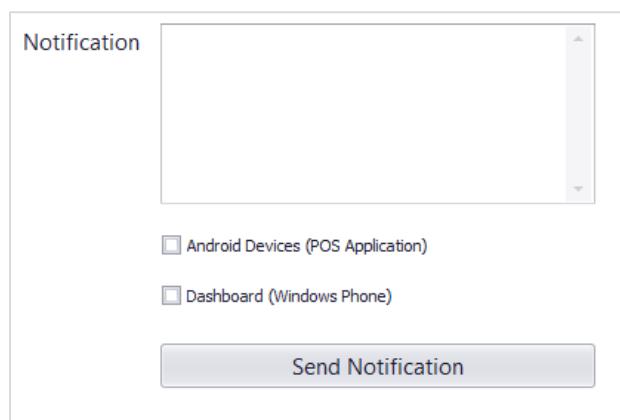
Figure 131 Key Press Events in Point of Sales module

In the point if sales and most if required other interfaces are contain with the key down event, for make the process faster. By using the key down event and identifying the pressed key, the multiple works can handled according to the key in single textbox.

As the example:

If user need to do the sales in the Point of Sales without touching the mouse, and only using the key board and only specific keys, the requirement can handled by the key down event and using other events as well.

5.8.2.6 Notification Server



Notification services has been used to transfer the notification from windows application as the server to the mobile application (Android device and Windows Mobile device)

After typing the message on the message box and by checking the specific device to send the notification, user can sent the notification to the mobile as explained below.

Send Notification from the Windows application

```
private void btSend_Click(object sender, EventArgs e)
{
    if (!string.IsNullOrEmpty(meNotification.Text))
    {
        var message = meNotification.Text;
        try
        {
            if (chAndroid.Checked)
            {
                var dt = dlMob.SelectMobileAccessAndroid();
                if (dt.Rows.Count != 0)
                {
                    foreach (DataRow dr in dt.Rows)
                    {
                        SendNotification(dr["DeviceID"].ToString(), message);
                    }
                }
            }
            if (chWinPhone.Checked)
            {
                pushObj = sp.BuildPushNotificationService();
                pushObj.SendPushMessageToUser(userId, message);
            }
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.Message);
        }
    }
}
```

Figure 132 Send Notification button click

Notification server to Android Application (Reflected from the Google Cloud Messaging API)

All the register android devices need a key that providing by the Google cloud messaging API and the key need be stored in the database and the key need to be update if the application has been re installed or modified. The key storing process accomplished via the web service. And when sending the notifications from the windows application to the android devices the key is required ad the device address to send the notification message.

The method that responsible to handle and send notification to the Android application

```
public string SendNotificationAndroidApplication(string deviceId, string message)
{
    var GoogleAppID = "AIzaSyCE8CQeWBKhGCYnk8Y00V5btuU3p-TcS10";
    var SENDER_ID = "650463209487";

    var value = message;

    WebRequest webRequest = WebRequest.Create("https://android.googleapis.com/gcm/send");
    webRequest.Method = "post";
    webRequest.ContentType = "application/x-www-form-urlencoded; charset=UTF-8";
    webRequest.Headers.Add(string.Format("Authorization: key={0}", GoogleAppID));

    webRequest.Headers.Add(string.Format("Sender: id={0}", SENDER_ID));

    var postData = String.Format
        ("collapse_key=score_update&time_to_live=108&delay_while_idle=1&data.message={0}&registration_id={1}",
        value, deviceId);

    Console.WriteLine(postData);
    Byte[] byteArray = Encoding.UTF8.GetBytes(postData);
    webRequest.ContentLength = byteArray.Length;

    Stream dataStream = webRequest.GetRequestStream();
    dataStream.Write(byteArray, 0, byteArray.Length);
    dataStream.Close();

   WebResponse webResponse = webRequest.GetResponse();
    dataStream = webResponse.GetResponseStream();
    StreamReader tReader = new StreamReader(dataStream);
    String sResponseFromServer = tReader.ReadToEnd();

    tReader.Close();
    dataStream.Close();
    webResponse.Close();
    return sResponseFromServer;
}
```

Figure 133 the method that responsible to handle and send notification to the Android application

The method is making the web request with the Google API Key and sender id (ID is created in the Google application console under the Google Cloud Messaging API) parameters and message and send the request as POST method via the web request.

Notification server to Windows Phone application (Reflected from the App42 API)

The App42 API need to be referenced to the application, from the downloaded libraries.

```
using com.shephertz.app42.paas.sdk.csharp;
using com.shephertz.app42.paas.sdk.csharp.pushNotification;
```

Figure 134 App42 References

After the referencing the library, the app Kept and security key need to be include to the method to send as the parameters with the message. The API detecting the user from the user id.

```
⌚ ServiceAPI sp = new ServiceAPI(
    "4cb05677663fe95f19a8c54e94e2ab8b9b9c21cf34f3ea91b1847e0843156d94",
    "a6a0eefafa51fa74c02c5ffa9652272054b71474b28b987e2eef061e17d8e397f4");

⌚ PushNotificationService pushObj = null;
⌚ const String userId = "3gbzone@gmail.com";
    1 reference
≡⌚ private void SendNotificationWindowsPhone(string message)
{
    pushObj = sp.BuildPushNotificationService();
    pushObj.SendPushMessageToUser(userId, message);
}
```

Figure 135 Notification server to Windows Phone application

5.8.2.7 Tab Loading Events

In the application, some of forms and user controls containing the multiple tasks divided from the tabs, and the added controls, grid views and mask boxes are loading long data begin from the form load, that task making the application slower, to avoid the slowness of the application from that situation. The tab page loading event has been used to control the data loading methods. Data only loading to the page when accessing to the tab page.

```
----- Tab Paint Events -----
1 reference
≡◆ private void tpUsers_Paint(object sender, PaintEventArgs e)
{
    gcUser.DataSource = dlUs.SelectUser();
}

1 reference
≡◆ private void tpUserLevel_Paint(object sender, PaintEventArgs e)
{
    gcUserLevel.DataSource = dlUs.SelectUserLevelWithPrivileges();
}

1 reference
≡◆ private void tpMobile_Paint(object sender, PaintEventArgs e)
{
    gcMobile.DataSource = dlMob.SelectMobileAccess();
}
```

Figure 136 Tab Loading Events

5.8.2.8 Batch Number generate algorithm in Good Reserving Notes module

In the inventory system as the main identifier of the item, the batch number has been decided to use, and creation of the specific and unique bath to each and every item, was a logical algorithm.

```
IF @RunType = 5
BEGIN
    SELECT
        SUBSTRING(dt.Category, 1, 2) AS Category,
        dt2.StockID,
        (SUBSTRING(dt.Category, 1, 2) + dt2.StockID) AS Batch
    FROM (SELECT
        HB_ItemCategory.Category
    FROM HB_ItemMaster
    INNER JOIN HB_ItemCategory
        ON HB_ItemMaster.ItemCategoryID = HB_ItemCategory.ItemCategoryID
    WHERE HB_ItemMaster.ProductCode = @ProductCode) dt
    LEFT OUTER JOIN (SELECT
        CAST((CASE
            WHEN MAX(StockID) IS NULL THEN 1
            ELSE MAX(StockID) + 1
        END) AS NVARCHAR(10)) AS StockID
    FROM HB_MainStock) dt2
        ON dt.Category != dt2.StockID
END
```

Figure 137 Database side selection for generate the Batch Number

The selected Batch number has been generated according to the category name and existing product count of the stock and the generated Batch Number is not getting equal with any batch number in future.

5.8.2.9 User authentication and application settings

In the application as the user requirements in the problem analyzing phase, the user privileges and assigning authentication to each users has been covered.

The main handler of the user control developed in the settings module of the windows application.

From the Setting Module

- Handling the Windows Phone application IMEI verification.
- Assigning the users to the system
- Manage users
- Managing user levels and authentication
- Manage /send the Mobile Notification

Manage and assign user levels and authentications to each user level

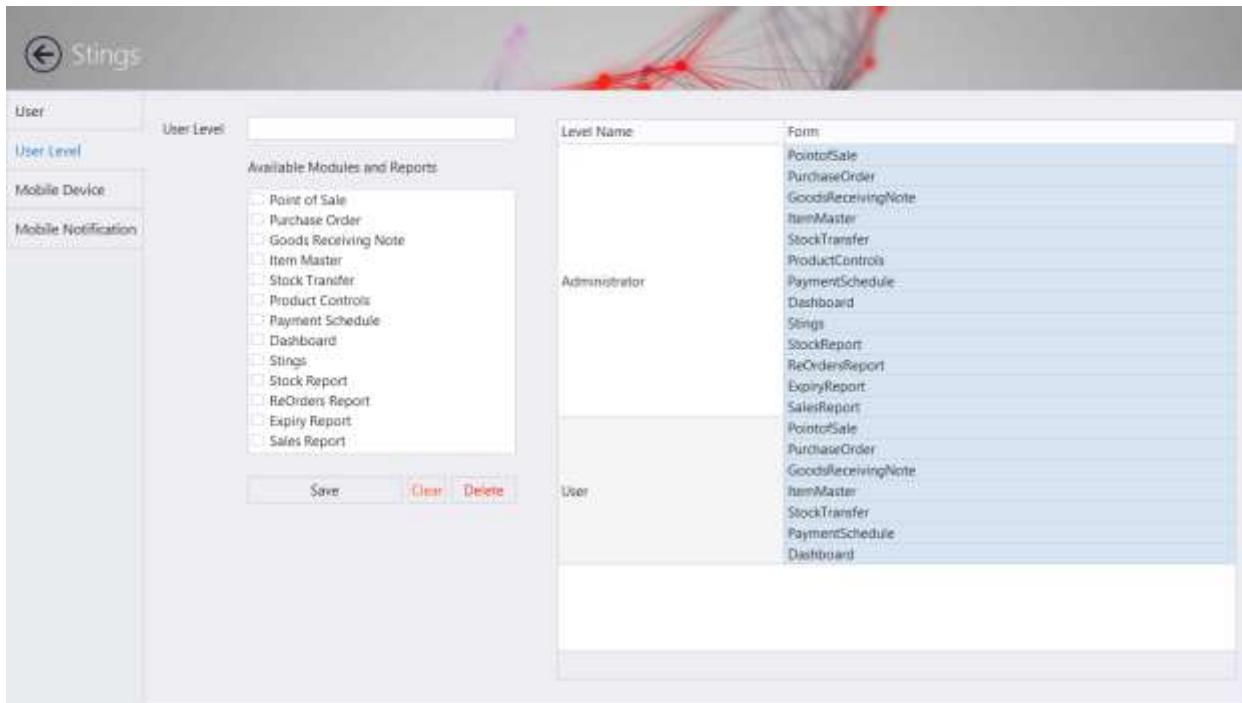


Figure 138 User Levels in the Settings

Authorized employee or administrator can create the users and assign the user levels to created user. According to the situation if user need to assign with new customized user level, administrator or authorized user can change the user levels and custom privileges to the created user level.

Applying the user privileges to the application modules

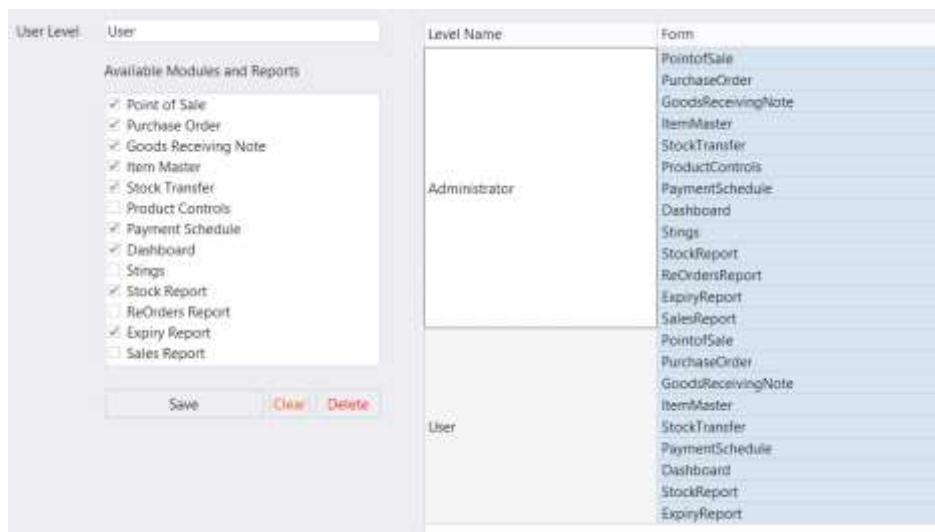


Figure 139 applying the user privileges

In the above screenshot the user privileges applied to the general user level. And the process of checking and applying the user privileges to each module handled by the main startup event.

When login user in to the system, the user privileges and major information about the user has been stored in to the application's static variables, and when accessing to the Windows UI Menu, the stored information are reading and considering the authentications and privileges to each modules from the database. After that, making the tiles in the Windows UI visible, enable or hide and disable. The below according is explaining the procedure that making the user privileges to each module according to the authentication level.

```
private void SetUserPrivileges()
{
    mdUp.UserLevel = iGlobal.UserLevel;
    DataTable dt = dlUs.SelectUserPrivileges(mdUp);

    PointofSaleTile.Visible = false;
    PurchaseOrderTile.Visible = false;
    GoodsReceivingNoteTile.Visible = false;
    ItemMasterTile.Visible = false;
    StockTransferTile.Visible = false;
    ProductControlsTile.Visible = false;
    PaymentScheduleTile.Visible = false;
    DashboardTile.Visible = false;
    StingsTile.Visible = false;
    StockReportTile.Visible = false;
    ReOrdersReportTile.Visible = false;
    ExpiryReportTile.Visible = false;
    SalesReportTile.Visible = false;

    foreach (DataRow dtr in dt.Rows)
    {
        switch (dtr["Form"].ToString())
        {
            case "PointofSale":
                PointofSaleTile.Visible = true;
                break; ↴
            case "PurchaseOrder":
                PurchaseOrderTile.Visible = true;
                break; ↴
            case "GoodsReceivingNote":
                GoodsReceivingNoteTile.Visible = true;
                break; ↴
            case "ItemMaster":
                ItemMasterTile.Visible = true;
                break; ↴
            case "StockTransfer":
                StockTransferTile.Visible = true;
                break; ↴
        }
    }
}
```

Figure 140 user privileges

Please refer (Appendix B: 7) for the coding that used in this phase

After applying the user privileges, the user can only access to some specific module as below.



Figure 141 Authenticated user

5.8.2.10 Dashboard Module Most Significant part of the Project

As the proposed application and analyzed information in the data gathering phase, mainly a fully functional data analyzing module is discovered as a very important and significant module. As a result of that, a Dashboard module has been introduced to the application to do the data analyzing part of the application. The module in the dashboard is analyzing the whole sales and all the information of the business and representing those analyzed data as the summarized data to the user by displaying true the graphical view.

The graphical view is containing with

Time range changer

- Chart
- Graph
- Pie charts
- Tables
- Pivot grid.

Development of the Dashboard

The dashboard is created as the **XML Document** the XML document is containing the data analyzing queries and structure of the graphical objects like charts graphs.

The Devexpress Dashboard Designer has been used as the tool to develop the XML document.

To create the dashboard, SQL Views are required, views are better than table because tables cannot represent the relational data without using the query customized query, but Views are created with pre created and already connected with all required tables as complete data table. As well as the views are live updating.

The Main requirement to create the dashboard is properly well formatted and optimized SQL Query or View

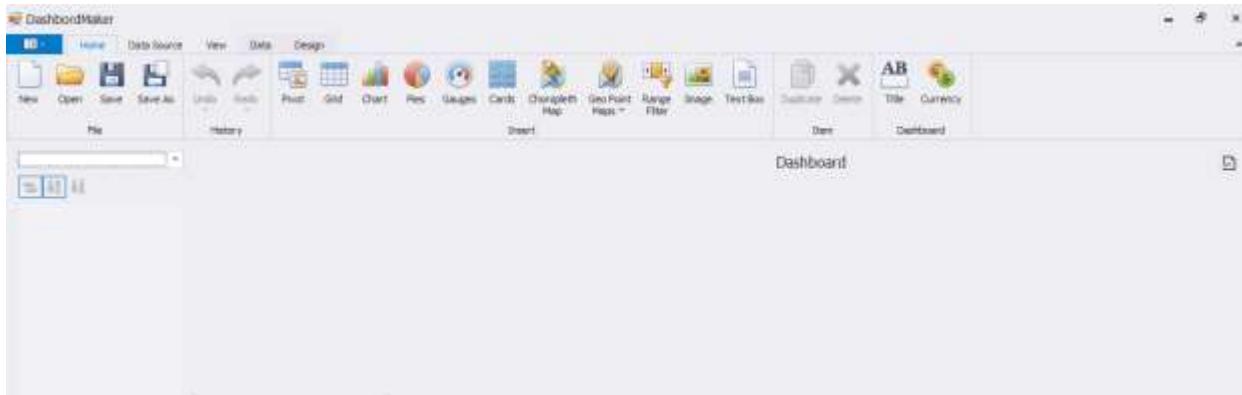


Figure 142 dashboard designer

From the dashboard designer first need to create the Data source and connect the required data source and view that required to make the data analyzes.

By clicking on the new button in the reborn controller developer can start create the dashboard by creating the data source ad selecting the view that needed to design.

After creating the data source and selecting the view that needed to make the analyzing dashboard. Developer can select the controller that need to make the graphical view, from the menu that provided that can be graph, date range, widgets, polite grid or chart.

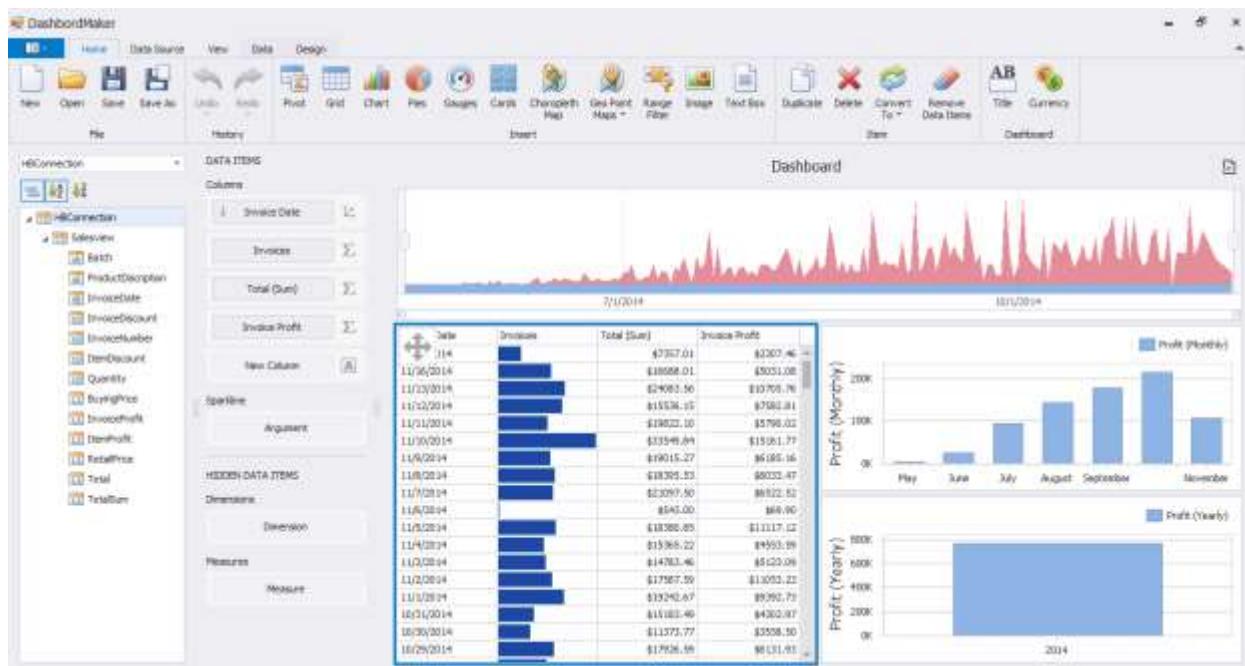


Figure 143 after design the dashboard

After designing the dashboard, can export the dashboard as the XML document.

The dashboard is embedding from the Devexpress Dashboard viewer too. And need hire bandwidth to analyze from the database if database captaining with long data capacity.

One of dashboard that using in the application

Please refer (Appendix B: 7) for the coding that used in this phase

```

<?xml version="1.0" encoding="utf-8"?>
<Dashboard CurrencyCulture="en-US">
    <title Text="Dashboard" />
    <DataConnections>
        <DataConnection Name="HBConnection" ProviderKey="MSSqlServer">
            <Parameters>
                <Parameter Name="server" Value=".\\SQLEXPRESS" />
                <Parameter Name="database" Value="HummingBird" />
                <Parameter Name="useIntegratedSecurity" Value="True" />
                <Parameter Name="read only" Value="1" />
                <Parameter Name="generateConnectionStringHelper" Value="false" />
            </Parameters>
        </DataConnection>
    </DataConnections>
    <DataSources>
        <DataSource ComponentName="HBConnection" Name="HBConnection">
            <DataProvider DataConnection="HBConnection" SupportSql="true">
                <Selection>
                    <Table Name="Salesview">
                        <Columns>
                            <Column Name="InvoiceDate" Alias="InvoiceDate" />
                            <Column Name="InvoiceNumber" Alias="InvoiceNumber" />
                            <Column Name="ProductDescription" Alias="ProductDescription" />
                            <Column Name="Batch" Alias="Batch" />
                            <Column Name="BuyingPrice" Alias="BuyingPrice" />
                            <Column Name="RetailPrice" Alias="RetailPrice" />
                            <Column Name="Quantity" Alias="Quantity" />
                            <Column Name="ItemDiscount" Alias="ItemDiscount" />
                            <Column Name="InvoiceDiscount" Alias="InvoiceDiscount" />
                            <Column Name="Total" Alias="Total" />
                            <Column Name="TotalSum" Alias="TotalSum" />
                            <Column Name="InvoiceProfit" Alias="InvoiceProfit" />
                            <Column Name="ItemProfit" Alias="ItemProfit" />
                        </Columns>
                    </Table>
                </Selection>
            </DataProvider>
        </DataSource>
    </DataSources>
    <Items>
        <RangeFilter ComponentName="rangeFilterDashboardItem1" Name="Range Filter 1" DataSource="HBConnection">
            <DataItems>

```

Figure 144 One of dashboard that using in the application

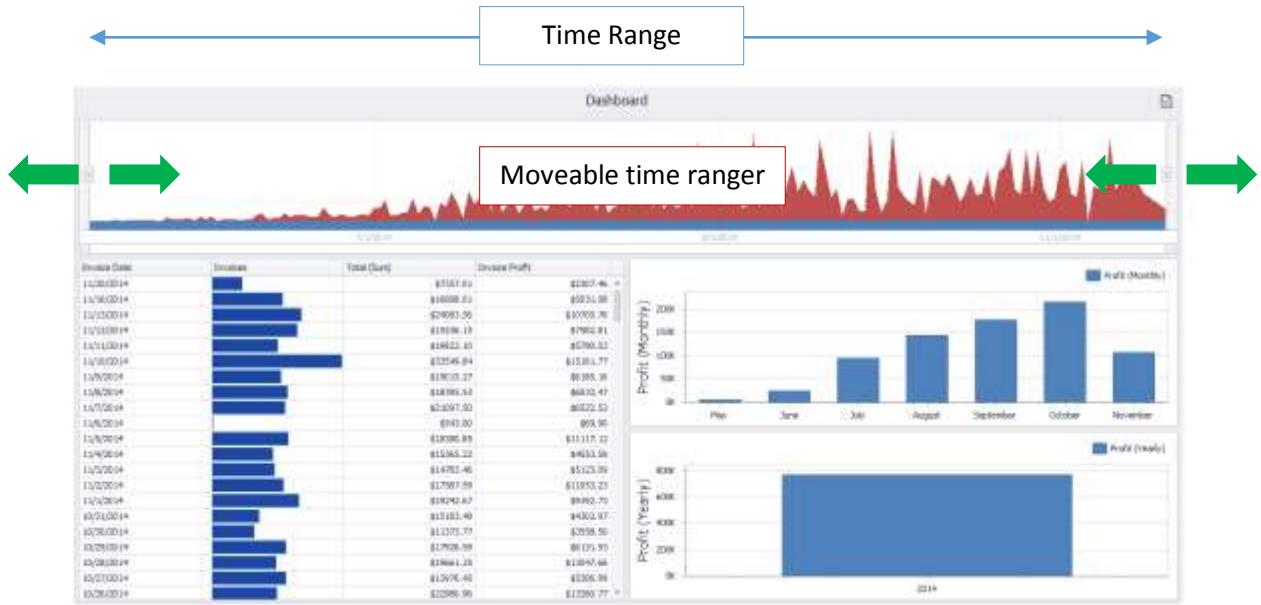


Figure 145 analyze manually

When moving the time range, the all data has been changing according to the time adjusted time range as represented below.

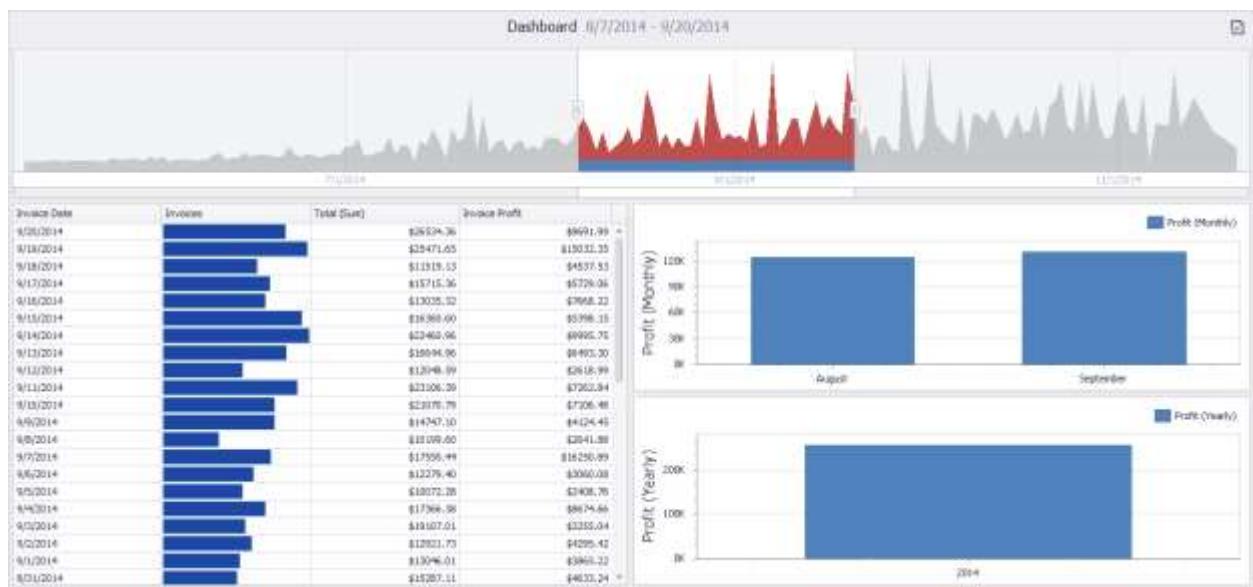


Figure 146 movable time range and changed cub data grid and charts according to the adjusted time range

The Completed dashboard

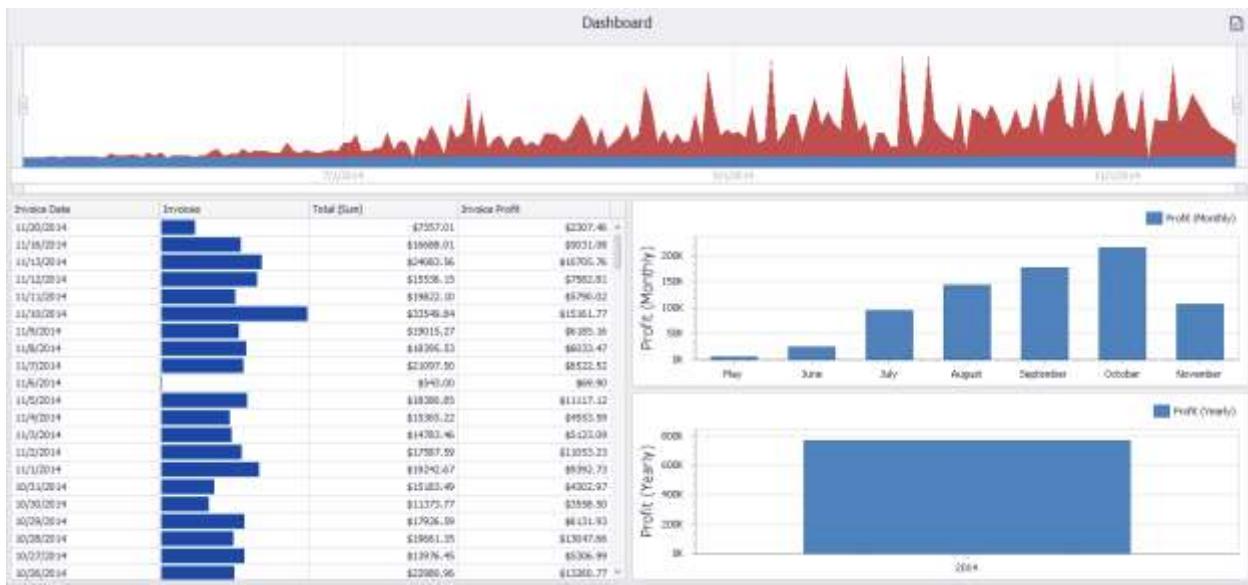


Figure 147 dashboard output tootle sales

The View that used to represent and analyze the tootle sales

SELECT

```

sh.InvoiceNumber,
sh.InvoiceDate,
sh.Discount AS InvoiceDiscount,
sh.TotalSum,
(SELECT
    SUM((pms.RetailPrice - pms.BuyingPrice) * psi.Quantity - psi.Discount) -
psh.Discount AS ItemProfit
    FROM dbo.HB_SalesItems AS psi
    INNER JOIN dbo.HB_SalesHead AS psh
        ON psi.InvoiceNumber = psh.InvoiceNumber
    INNER JOIN dbo.HB_MainStock AS pms
        ON psi.BatchNumber = pms.BatchNumber
    WHERE (psi.InvoiceNumber = sh.InvoiceNumber)
    GROUP BY      psh.InvoiceNumber,
                    psh.Discount)
AS InvoiceProfit,
si.BatchNumber AS Batch,
im.ProductDiscription,
ms.BuyingPrice,
ms.RetailPrice,
si.Quantity,
si.Discount AS ItemDiscount,
```

```

si.Total,
si.IssueMode,
(CASE
    WHEN si.IssueMode = 'F' THEN 0
    ELSE (((ms.RetailPrice - ms.BuyingPrice) * si.Quantity) - si.Discount)
END) AS ItemProfit,
ISNULL
((SELECT
    sp.SupplierName
FROM dbo.HB_POHeader AS ph
INNER JOIN dbo.HB_Suppliers AS sp
    ON ph.SuppliersID = sp.SuppliersID
WHERE (ph.PONumber = ms.PONumber)), (SELECT
    sp.SupplierName
FROM dbo.HB_ItemMaster AS im
INNER JOIN dbo.HB_Suppliers AS sp
    ON im.SuppliersID = sp.SuppliersID
WHERE (im.ProductCode = ms.ProductCode))
) AS SupplierName,
ct.Category
FROM dbo.HB_SalesHead AS sh
INNER JOIN dbo.HB_SalesItems AS si
    ON sh.InvoiceNumber = si.InvoiceNumber
INNER JOIN dbo.HB_MainStock AS ms
    ON si.BatchNumber = ms.BatchNumber
INNER JOIN dbo.HB_ItemMaster AS im
    ON ms.ProductCode = im.ProductCode
INNER JOIN dbo.HB_ItemCategory AS ct
    ON im.ItemCategoryID = ct.ItemCategoryID

```



Figure 148 dashboard output for today sales

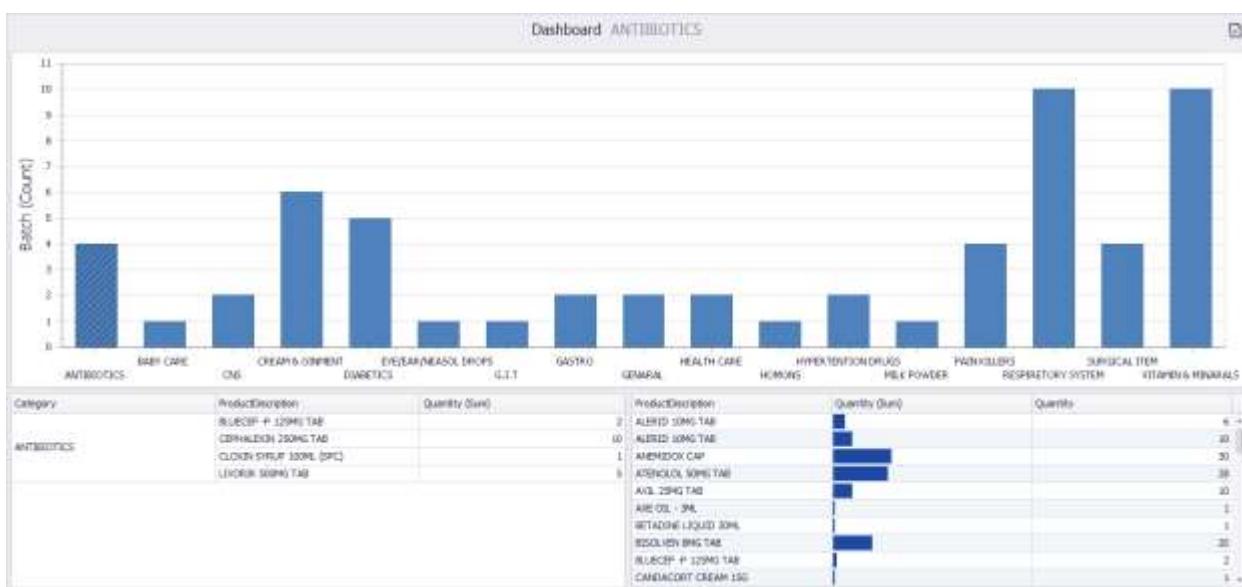


Figure 149 Today sales according to categories

5.8.3 SOAP Web Service (HummingBird.WebService)

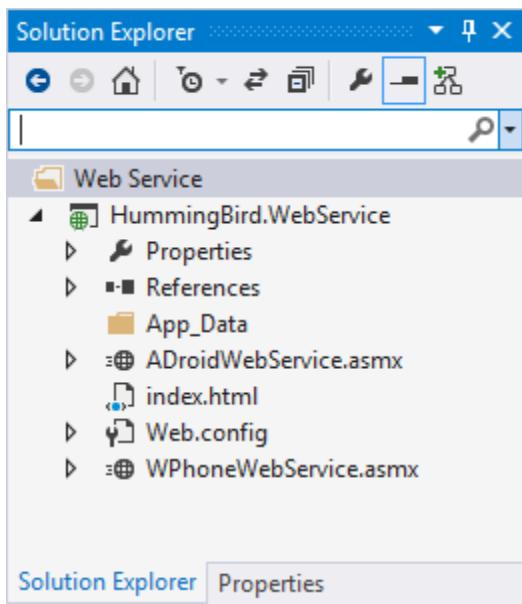


Figure 150 web service explorer

In the application web services has been created to accomplish mobile application's data transferring purposes. In the created web service application two web services has been developed

1. ADroidWebService.asmx

The web service is containing all the web methods that responsible to handle tasks in Android mobile application.

2. WPhoneWebService.asmx

The web service is containing all the web methods that responsible to handle tasks in Windows phone mobile application.

To the web service, the common layers that we created, to do the database transactions and modeler layer of the application also included as the references.

In all two web services, the class objects are created from the data layer and related modeler layer class object also created.

In each methods that created in the web services, deferring the data layer and taking the data from the database true the data layer and returning true the web service.

The web services contain has been developed as below explained from the screen shot.

```

    ◑ private dlSalesHead dlSLhd = new dlSalesHead();
    ◑ private mdSalesHead mdSLhd = new mdSalesHead();

    ◑ private dlSalesItems dlSLi = new dlSalesItems();
    ◑ private mdSalesItems mdSLi = new mdSalesItems();

    ◑ private dlUser dlUs = new dlUser();
    ◑ private mdUser mlUs = new mdUser();

    ◑ private dlMainStock dlMstk = new dlMainStock();
    ◑ private mdMainStock mdMstk = new mdMainStock();

    ◑ private dlMobileAccess dlMob = new dlMobileAccess();
    ◑ private mdMobileAccess mdMob = new mdMobileAccess();

    [WebMethod]
    0 references
    ◑ public ArrayList GetBatch()
    {
        return dlMstk.SelectBatch(); ↵
    }

    [WebMethod]
    0 references
    ◑ public DataTable GetFromBatch(string Batch)
    {
        mdMstk.BatchNumber = Batch;
        return dlMstk.SelectFromBatch(mdMstk); ↵
    }

    [WebMethod]
    0 references
    ◑ public String GetInvoiceNumber()
    {
        return dlSLhd.GetNextInvoiceNumber(); ↵
    }

```

Figure 151 sample web methods that created in Android web service

5.8.4 Android Application

The android application has been created using the **Cross platform** technology by avoiding the traditional android development technologies. In the development as the development tool newest **Xamarin Android** has been used.

Based on that the development processes and technologies has been had to be re-create and use the new technologies came with the Cross platform technology

5.8.4.1 Inbuilt Self Creation Database

Creating the self-creatable database, has required to the android application in **SQLite** format as the potable and inbuilt database.

The requirement of the Potable database. The proposed Point Of Sales android application need to hold the invoice that going to proceed, until submit to the database true the web service. As the result of the requirement the database has been provided to create it automatically by application itself.

The method that created the database in first run of the android application.

```
public void createDB()
{
    var documents = Environment.GetFolderPath(Environment.SpecialFolder.Personal);
    try
    {
        var connectionString = String.Format("Data Source={0};Version=3;", documents +
"MyDatabase.db");
    }
    catch (Exception)
    {
        SqliteConnection.CreateFile(documents + "//MyDatabase.db");
    }
    finally
    {
        var connectionString = String.Format("Data Source={0};Version=3;", documents +
"MyDatabase.db");
        using (var conn = new SqliteConnection(connectionString))
        {
            conn.Open();
            using (var cmd = conn.CreateCommand())
            {
                cmd.CommandText = "CREATE TABLE IF NOT EXISTS TemplInvoice (ID INTEGER NOT
NULL PRIMARY KEY AUTOINCREMENT,InvoiceNumber INTEGER,ProductCode
```

```

        INTEGER,ProductName NVARCHAR(1000),BatchNumber NVARCHAR(15),ExpiryDate
        DATE,Quantity INTEGER,Discount INTEGER,IssueMode NVARCHAR(1),RetailPrice
        NUMERIC,Total NUMERIC");
    cmd.CommandType = CommandType.Text;
    cmd.ExecuteNonQuery();
    conn.Close();
}
}
}
}

```

The method is firing when first run of the application and creating the database itself as the requirement

5.8.4.2 Android Service and Google Cloud Message Broadcast Receiver

By referencing the re customized android Google cloud messaging service client library (HummingBird.GCM.Client) to the android application, the notification reserve service has been created.

The service is installing on the android device in first run, and the service is running until application is getting stop or getting uninstall from the device. The service is responsible for reserve notifications.

Android Service classes and Google Cloud Message Broadcast Receiver

Android service

```

[Service]
public class GcmService : GcmServiceBase
{
    public GcmService() : base(GcmBroadcastReceiver.SENDER_IDS) { }
    public ADroidWebService ws = new ADroidWebService();

    protected override void OnRegistered(Context context, string registrationId)
    {
        Console.WriteLine("Device Id:" + registrationId);

        var telephonyManager = (TelephonyManager)GetSystemService(TelephonyService);

        var IMEI = telephonyManager.DeviceId;
        var regid = registrationId;
        var User = "Android";
        var typ = "Android";
    }
}

```

```

var res = ws.insertMobileAccessFrmMobile(IMEI, regid, User, typ);

var preferences = GetSharedPreferences("AppData", FileCreationMode.Private);
var deviceId = preferences.GetString("DeviceId", "");

if (string.IsNullOrEmpty(deviceId))
{
    var editor = preferences.Edit();
    editor.PutString("DeviceId", registrationId);
    editor.Commit();
}
}

protected override void OnMessage(Context context, Intent intent)
{
    if (intent != null && intent.Extras != null)
    {
        var message = intent.Extras.GetString("message");
        createNotification("Push Sample", message);
    }
}

protected override void OnUnRegistered(Context context, string registrationId)
{
    //Receive notice that the app no longer wants notifications
}

protected override void OnError(Context context, string errorId)
{
    //Some more serious error happened
}

private void createNotification(string title, string desc)
{
    var notificationManager = GetSystemService(Context.NotificationService) as
NotificationManager;
    var uIntent = new Intent(this, typeof(MainActivity));
    var notification = new Notification(Resource.Drawable.Icon, title);
    notification.Flags = NotificationFlags.AutoCancel;
    notification.Defaults = NotificationDefaults.Sound;
    notification.SetLatestEventInfo(this, title, desc, PendingIntent.GetActivity(this, 0, uIntent, 0));
    notificationManager.Notify(1, notification);
}

```

Android Google Cloud Message Broadcast Receiver

```
[BroadcastReceiver(Permission = Constants.PERMISSION_GCM_INTENTS)]
[IntentFilter(new string[] { Constants.INTENT_FROM_GCM_MESSAGE }, Categories = new
string[] { "@PACKAGE_NAME@" })]
[IntentFilter(new string[] { Constants.INTENT_FROM_GCM_REGISTRATION_CALLBACK },
Categories = new string[] { "@PACKAGE_NAME@" })]
[IntentFilter(new string[] { Constants.INTENT_FROM_GCM_LIBRARY_RETRY }, Categories =
new string[] { "@PACKAGE_NAME@" })]
public class GcmBroadcastReceiver : GcmBroadcastReceiverBase<GcmService>
{
    //The SENDER_ID is your Google API Console App Project Number

    public static string[] SENDER_IDS = new string[] { "650463209487" };
}
```

5.8.5 Windows Phone Application

The windows application has been created for the owner of the business. And owner can analyze the business true the windows phone application and make the decisions according to the results.

5.8.5.1 Voice Recognition

The voice recognition services has been developed in the application and using windows phone operating system's inbuilt voice recognition libraries.

The voice recognition has been created for only move in to the each pages. And user need to provide the proper name of the page that need to access.

```
this.recoWithUI = new SpeechRecognizerUI();

SpeechRecognitionUIResult recoResult = await recoWithUI.RecognizeWithUIAsync();

if (recoResult.RecognitionResult != null)
{
    var text = recoResult.RecognitionResult.Text;

    switch (text.ToLower())
    {
        case "send feedback":
        case "send feedback.":
        case "feedback":
        case "feedback.":
            NavigationService.Navigate(new Uri("/FeedBack.xaml", UriKind.RelativeOrAbsolute));
            break;

        case "go to the payment schedule":
        case "go to the payment schedule.":
        case "payment schedule":
        case "payment schedule.":
        case "schedule":
        case "schedule.":
        case "payment":
        case "payment.":
            NavigationService.Navigate(new Uri("/PaymentSchedule.xaml",
UriKind.RelativeOrAbsolute));
            break;

        case "go to the upcoming expires":
```

```

        case "go to the upcoming expires.":
        case "expires":
        case "expires.":
        case "go to upcoming expires":
        case "go to upcoming expires.":
        case "upcoming expires":
        case "upcoming expires.":
            NavigationService.Navigate(new Uri("/UpcomingExpires.xaml",
UriKind.RelativeOrAbsolute));
            break;

        case "go to the upcoming reorder":
        case "go to the upcoming reorder.":
        case "reorder":
        case "reorder.":
        case "order":
        case "order.":
        case "upcoming reorder":
        case "upcoming reorder.":
        case "go to upcoming reorder":
        case "go to upcoming reorder.":
            NavigationService.Navigate(new Uri("/UpcomingReOrders.xaml",
UriKind.RelativeOrAbsolute));
            break;
    }
}

```

5.8.5.2 Notification service

Then notification service has been started within the application main interface start. And on tile installing the notification service in the phone and lessening to the notification server by providing the device id. And pre created the API key and security key that given from the APP42 notification service, inbuilt I the application.

Method that initializing the notification service in the windows phone

```

public SpeechSynthesizer synth = new SpeechSynthesizer();
ServiceAPI sp = new
ServiceAPI("4cb05677663fe95f19a8c54e94e2ab8b9b9c21cf34f3ea91b1847e0843156d94",
"a6a0eefa51fa74c02c5ffa9652272054b71474b28b987e2eef061e17d8e397f4");
PushNotificationService pushObj = null;

```

```

static String userId = "3gbzone@gmail.com";
NotificationCallBack callback;

public MainPage()
{
    HttpNotificationChannel channel;
    SystemTray.SetProgressIndicator(this, indicator);
    pushObj = sp.BuildPushNotificationService();

    this.expenseDiff = specialValue - this.range1.MinValue;
    this.profitDiff = this.range2.MaxValue - this.range2.MinValue;

    String channelName = "App42PushNotificationTilest";
    channel = HttpNotificationChannel.Find(channelName);

    callback = new NotificationCallBack();

    if (channel == null)
    {
        channel = new HttpNotificationChannel(channelName);

        channel.ChannelUriUpdated += new
EventHandler<NotificationChannelUriEventArgs>(PushChannel_ChannelUriUpdated);
        channel.ErrorOccurred += new
EventHandler<NotificationChannelErrorEventArgs>(PushChannel_ErrorOccurred);
        channel.ShellToastNotificationReceived += new
EventHandler<NotificationEventArgs>(PushChannel_ShellToastNotificationReceived);

        channel.Open();
        Collection<Uri> TileLocations = new Collection<Uri>();
        // remote images in the tile
        TileLocations.Add(new Uri("http://api.shephertz.com/"));
        channel.BindToShellTile(TileLocations);
        channel.BindToShellToast();
    }
    else
    {
        channel.ChannelUriUpdated += new
EventHandler<NotificationChannelUriEventArgs>(PushChannel_ChannelUriUpdated);
        channel.ErrorOccurred += new
EventHandler<NotificationChannelErrorEventArgs>(PushChannel_ErrorOccurred);
        channel.ShellToastNotificationReceived += new
EventHandler<NotificationEventArgs>(PushChannel_ShellToastNotificationReceived);
    }
}

```

```

        StoreURIWithApp42(channel.ChannelUri.ToString());

    }

}

void PushChannel_ChannelUriUpdated(object sender, NotificationChannelUriEventArgs e)
{
    StoreURIWithApp42(e.ChannelUri.ToString());
}

void PushChannel_ErrorOccurred(object sender, NotificationChannelErrorEventArgs e)
{
    Dispatcher.BeginInvoke(() =>
        MessageBox.Show(String.Format("error occurred.",
            e.ErrorType, e.Message, e.ErrorCode, e.ErrorAdditionalData))
    );
}

void PushChannel_ShellToastNotificationReceived(object sender, NotificationEventArgs e)
{
    StringBuilder message = new StringBuilder();
    string relativeUri = string.Empty;

    message.AppendFormat("App42 Notification {0}:\n",
        DateTime.Now.ToShortTimeString());

    foreach (string key in e.Collection.Keys)
    {
        message.AppendFormat("{0}: {1}\n", key, e.Collection[key]);

        if (string.Compare(
            key,
            "wp:Param",
            System.Globalization.CultureInfo.InvariantCulture,
            System.Globalization.CompareOptions.IgnoreCase) == 0)
        {
            relativeUri = e.Collection[key];
        }
    }

    Dispatcher.BeginInvoke(() => MessageBox.Show(message.ToString()));
}

```

```
void StoreURIWithApp42(String ChannelUri)
{
    pushObj.StoreDeviceToken(userId, ChannelUri, this);
}

void App42Callback.OnException(App42Exception exception)
{
    Deployment.Current.Dispatcher.BeginInvoke(() =>
    {
        indicator.isVisible = false;
    });
    Console.WriteLine(exception.ToString());
}

void App42Callback.OnSuccess(object response)
{
    Deployment.Current.Dispatcher.BeginInvoke(() =>
    {
        indicator.isVisible = false;
    });
    Console.WriteLine(response.ToString());
}
```

5.8.5.3 Using asynchronous method to read the SOAP web service

In the windows phone applications to read the SOAP web services asynchronous methods has been used. In the developed application also providing the SOAP web service. According to the situation from the web methods, data retrieved from the asynchronous methods.

Sample asynchronous that used in the windows phone application

```
private void Reload()
{
    try
    {
        WPhoneWebServiceSoapClient ws = new WPhoneWebServiceSoapClient();
        ws.getTodayInvoiceCompleted += ws_getTodayInvoiceCompleted;
        ws.getTodayInvoiceAsync();

        ws.getYesterdayInvoiceCompleted += ws_getYesterdayInvoiceCompleted;
        ws.getYesterdayInvoiceAsync();
    }
    catch (Exception ex)
    {
        MessageBoxResult result = MessageBox.Show(ex.Message, "Error", MessageBoxButton.OK);

        if (result == MessageBoxResult.OK)
        {
            Application.Current.Terminate();
        }
    }
}

1 reference
void ws_getYesterdayInvoiceCompleted(object sender, iWebService.getYesterdayInvoiceCompletedEventArgs e)
{
    object data = e.Result;
    SetSeries(data, 0);
}

1 reference
void ws_getTodayInvoiceCompleted(object sender, iWebService.getTodayInvoiceCompletedEventArgs e)
{
    object data = e.Result;

    SetSeries(data, 1);
}
```

Figure 152 Asynchronous method

4.9.5.4 Usage if hijacked events to accomplish the custom task

In the method below an asynchronous methods used to bind date to the list boxes, in the developed windows mobile interface, there is four lists created to deferent approach but same data formats and same web service with deferent parameters.

As the worst case scenario, in the application need to be created separate methods to take the data to the list boxes

But in the development below, the asynchronous event has been hijacked to accomplish to do same task within defiant List boxes and deferent parameters.

```
private void OrderZone(int From, int To, ListBox listBox)
{
    WPhoneWebServiceSoapClient ws = new WPhoneWebServiceSoapClient();
    ws.GetReorderListCompleted += (sender, e) => ws_GetReorderListCompleted(sender, e, listBox);
    ws.GetReorderListAsync(From, To);
}

1 reference
void ws_GetReorderListCompleted(object sender, GetReorderListCompletedEventArgs e, [ListBox] listBox)
{
    var result = e.Result;
    XElement doc = result.Any1;

    List<Reorder> clist = new List<Reorder>();

    foreach (XElement docElement in doc.Elements())
    {
        foreach (XElement countryelement in docElement.Elements("Table"))
        {
            clist.Add(new Reorder()
            {
                Description = countryelement.Element("Description").Value.ToString(),
                OnHand = countryelement.Element("OnHand").Value.ToString(),
                OrderLevle = countryelement.Element("OrderLevle").Value.ToString(),
                Category = countryelement.Element("Category").Value.ToString(),
                SupplierName = countryelement.Element("SupplierName").Value.ToString()
            });
        }
    }
    listBox.ItemsSource = clist;
}
```

Figure 153 event hijacking to do custom tasks

5.9 Summery

This chapter was dedicated to provide a discussion about development and development techniques as well as the development process of the application by following best practices of the software development. And tools techniques, coding imparments, architectures that used in the development of the application.

6.0 Testing

Intelligent, Business and Automated Inventory Systems is critical for the delivery of a successful project. The key purpose of testing is to check whether that the developed system is fulfilled the user requirements and confirms to the test results expected under a wide range of conditions which could have been tested. To accomplish a successful system implementation, all aspects of the testing segment should be reviewed with the contribution of the developer as well as users.

6.1 System Testing Objectives

Minor objectives for action software system Testing are to make sure that the answer meets the business and user requirements, to catch errors that may be bugs or faults, to making sure that a system is prepared for use, to determinative user acceptability, to gaining confidence that it works, evaluating the capabilities of a system to point out that a system performs as intended, and to corroborative documentation of the application.

System testing facilitates the developed Intelligent, Business and Automated Inventory Systems (IBAIS) to be analyzed completely in order to confirm that the specifications and business functions which it was proposed are being encountered. The mechanisms of the system are being tested at the development of those components. But it is required to test those mechanisms to confirm the combinations of the system components were properly done.

System testing is conducted by checking the fallowing checkpoints are tailored with the developed system.

Performance

Performance testing is performed to verify the system operate under peak and continuous loads of processes at a significant speed.

Accuracy

Inaccurate information leads the whole system failure if the information is not stable. The test is performed to verify the outputs of the system are accurate in various operational environments.

Functionality

Test is performed to make sure the system meet requirement specifications and hence supports business requirements of the company. Functionalities should address the problems existed in current system and its processes.

Interfaces

New system is developed to provide more information using less number of interfaces and users should be able identify them easily and separately. Testing is performed to make sure the interfaces are done according to the specifications and the interfaces are linked each other in well-organized manner.

Volume

Because the new Intelligent, Business and Automated Inventory Systems (IBAIS) allows more information about product, invoices, user and sales details to be stored, this testing is identified as important to verify the system is capable of monitoring a large no of tasks while handle large volume of input data.

Security

Since the system handles sensitive information of the organization's and it's users security of the system should be well-ensured, this testing is performed to verify the feature of the system ensure access, integrity and recovery features operate as expected.

6.2 Testing Strategy

Testing strategy ensures that all the tests which are to be conducted have been identified and the test will cover each and every part of the system according to the scope of the system. Testing strategy concern project scenario as well as the strategy has been used for system development. Following tests will be carried out to ensure the delivery of high quality product.

Unit Testing

Unit testing is intended to verify that a module of the system has been developed according to the specifications. It also ensures the modules communicate with each other properly.

Unit testing will only tests the components of the Intelligent, Business and Automated Inventory Systems (IBAIS) themselves. It does not help to identify each and every error in the system such as performance problems, integration errors. Therefore unit testing will be helpful only when it used in combination with other testing techniques.

Integration testing

Integration testing will identify the errors that were not or couldn't identify in the previous testing. Purpose of conducting the integration testing is verifying the functional and performance requirements defined on project design specifications.

There are different types of integration techniques. Integration testing will take the modules of the system which are tested under unit testing, as inputs for testing and group them in better

collection. Then it will apply tests defined in test plan to those grouped units to verify system integration and delivers the output as integrated Intelligent, Business and Automated Inventory Systems (IBAIS) which is ready for the next testing which is known as system testing.

System Testing

Overall functionalities of the system including the modules integrated modules and interfaces will be tested out of the system testing. System testing will not receive many errors since the input which is taken to the system testing is the system and modules which are tested under unit testing and integration testing. But the errors which identify during this testing process will be critical since it will be affected to the integration of the Intelligent, Business and Automated Inventory Systems (IBAIS)'s modules and will have to do changes of the system.

Acceptance Testing

Acceptance testing will be the next testing to be conducted. It is designed to ensure that all the changes made come across with the original system specifications and user requirements of the Intelligent, Business and Automated Inventory Systems (IBAIS) during the design, development and other initial stages. Basically this will decide how the system has completed the user requirements.

6.3 Test Plan

Test plan defines a systematic approach to test a system. This will discuss the process of conducting the above mentioned testing strategies. Under test plan, test strategies which are identified as to be carried, will be separated into testing plans as unit testing and system testing. Components of the system and integration of components will be carried under unit testing. System testing and acceptance testing will be carried under System testing. Concerns made when designing the test plan are,

- Identifying the components and features to be tested and not tested
- Ensuring all required elements are in place for testing
- Who conducts the testing for particular component or feature
- Plan for make necessary changes for issues arise on testing

6.3.1 Test Plan – Unit Testing

Unit testing is performed before the implementation and integration of system components to ensure the functionality. Main components of the system will be tested during the unit testing.

Module	Description
Point of Sales	Main functions of the point of sales, make the sales, view the selected batch numbers details are carried out before integration to make sure they are error free.
Product Controls	Basic functions of the item categories module, customer module, and supplier module such as create, delete, update and etc. are carried out before integration to make sure they are error free.
Item Master	Basic functions such as create, delete, update and etc. are carried out before integration to make sure they are error free.
Product Order	Basic functions of the product order such as get suppliers get product list and do save the order, select multiple items and etc... are carried out before integration to make sure they are error free.
Good reserving notes	View orders, accepting goods reserving notes are carried out before integration to make sure they are error free.
Stock Controls	Basic functions such as create, delete, update and etc. are carried out before integration to make sure they are error free.
Report Generation Module	Report module is tested by producing various types of pdf reports. The objective of this testing is to minimize the errors that can be occurred in this module and to provide smooth functioning.
Payment schedule	Check the schedule after approving a goods reserving note.
Dashboard	Checking the accuracy of analyzed data in the dashboard.
Mobile Push Notification	This component is tested to make sure it is sending proper alert to the client application

User Authentication module	This component is tested by entering various user names and passwords to make sure it validates the authorized employee names and passwords. And the application menu getting changing according to the assigned user privileges.
-----------------------------------	---

Table 3 Test Plan – Unit Testing

Unit Testing

Test Module: Product Controls

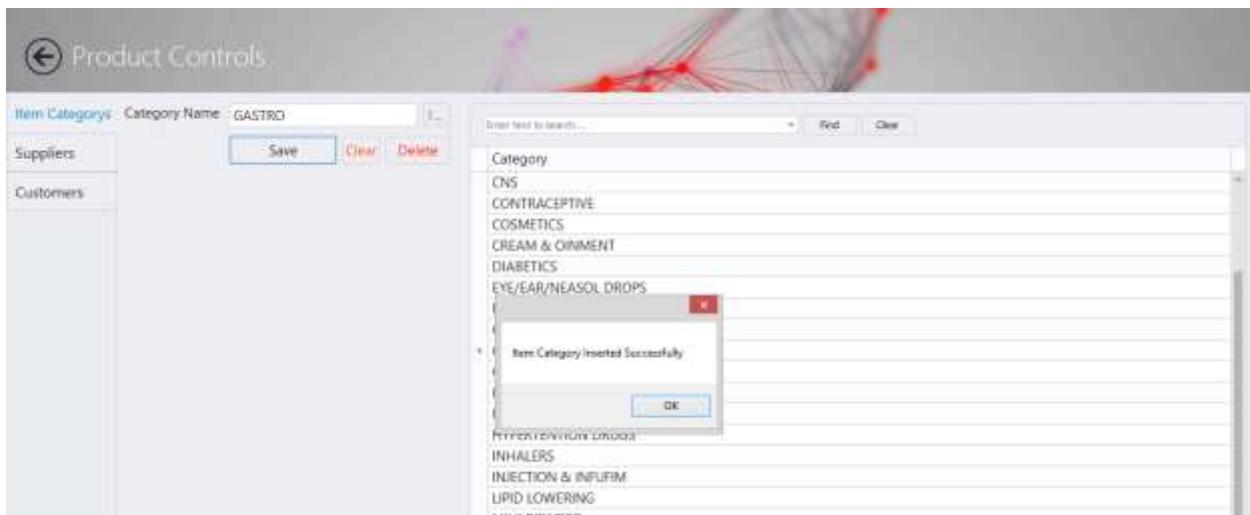


Figure 154 Test Module: Product Controls

Sample Data: “GASTRO”

Statuses: Success

Test Module: Item Master

The screenshot shows the 'Item Master' interface. On the left, there are several input fields: Product Description ('FAIR & HANDSOME CREAM'), Supplier ('COLOMBO'), Company ('ABC COMPANY'), Generic Name ('AMOXICILLIN + CLAVULANATE'), Country (''), Unit Name ('MG'), Unit Size ('375'), Brand (''), Item Category ('COSMETICS'), and Reorder Level ('20'). Below these are buttons for 'New', 'Save', and 'Delete'. The main area is a grid table with columns: Product Description, Category, Supplier, Company, Generic, Country, Unit, Unit Size, and Brand. A modal dialog box in the center says 'Product Inserted Successfully' with an 'OK' button. The grid contains rows for various items like 'BABY TEET HANDLE', 'FEEDING BOTTLE (L)', etc.

Figure 155 Test Module: Item Master

Sample Data: "ABC Company"

Statuses: Success

Test Module: Product Order

The screenshot shows the 'Purchase Order' interface. It includes fields for Suppliers ('SINGHE HOLDINGS'), Order Date ('11/20/2014'), Product (''), Quantity ('0'), and PO Number ('PO-7...'). Below is a table with columns: Product Code, Product Name, and Quantity. Two rows are listed: 'B188 SAFE GUARD (10) - M 12' and '4023 DAKTACORT 14G CR 12'. To the right is a search bar and a grid of purchase orders. A modal dialog box in the center says 'Order Added Successfully' with an 'OK' button. The grid shows various items with their respective PO numbers, quantities, PO dates, and supplier names.

Figure 156 Test Module: Product Order

Sample Data: "SAFE Guard (10)-M"
"Daktacora 14G CR"

Statuses: Success

Test Module: Good reserving notes

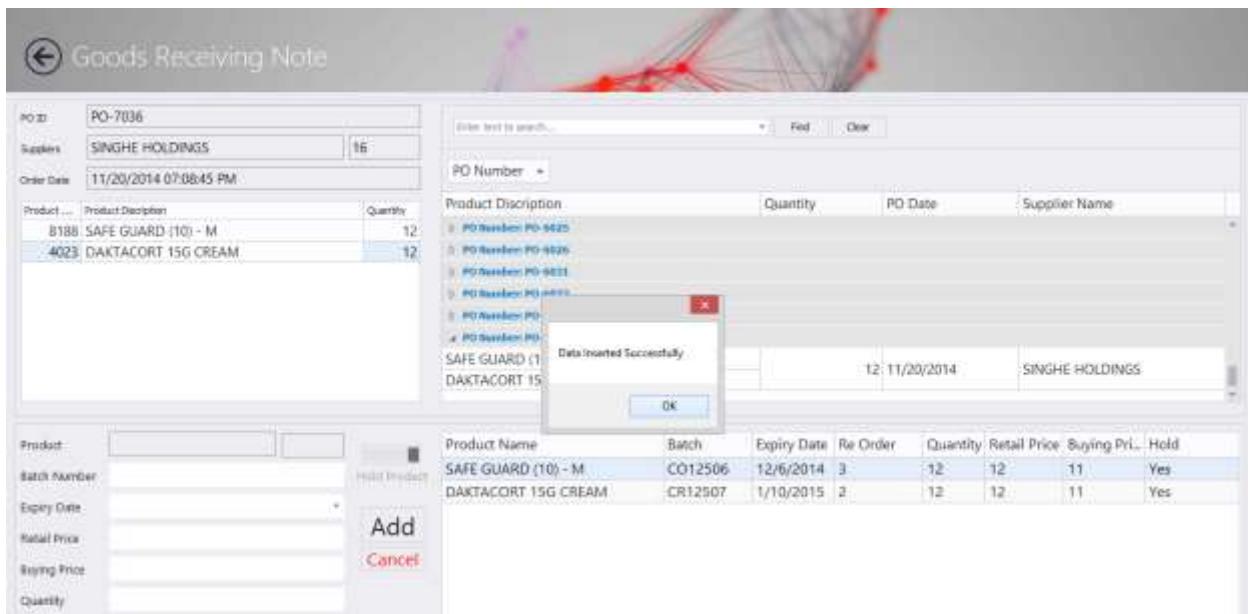


Figure 157 Test Module: Good reserving notes

Sample Data: "PO-7036"

Statuses: Success

Test Module: Payment schedule

Payment Schedule						
Year	Month	Day	PO Number	Supplier Name	Total	PO Date
			PO-7036	SINGHE HOLDINGS	264,0000	11/20/2014

Figure 158 Test Module: Payment schedule

Sample Data: Last approved order ID (PO-7036) has been arrived in the Payment schedule

Statuses: Success

Test Module: Dashboard



Figure 159 Test Module: Dashboard

No sample data used to test the dashboard checking the accuracy of the visible data.

Test Module: Mobile Push Notification

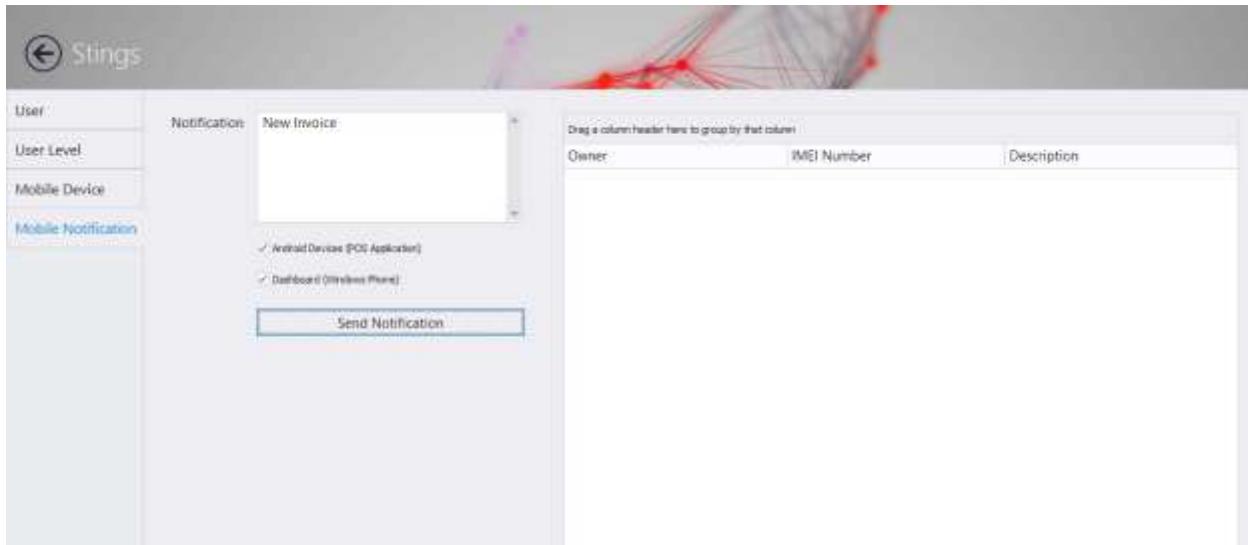


Figure 160 Test Module: Mobile Push Notification

Sample Data: “New Invoice” A notification has appeared in the mobile application

Statuses: Success

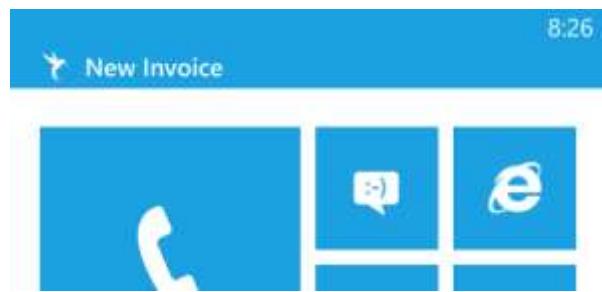


Figure 161 Test Module: Mobile Push Notification

6.3.2 Test Plan – System Testing

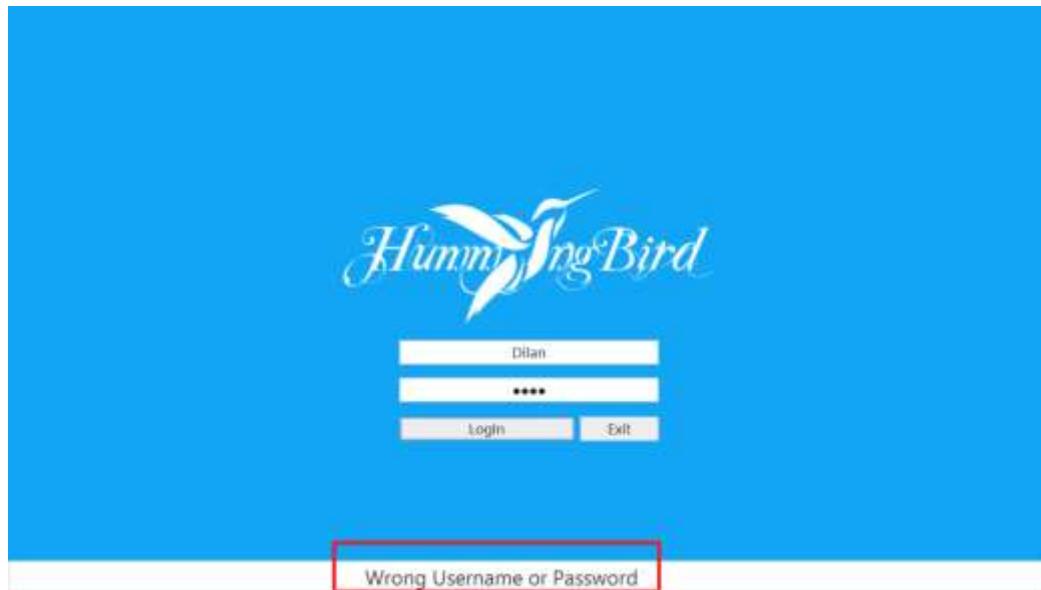
System testing will be carried under main three areas in the system, Item Master, and report generation in order to test the functions of the overall Analytical, Intelligent, Business and Inventory Automation Systems.

First system test will be carried to test the functionalities of the monitoring system. These tests consist of testing, monitoring activities, status display, system maintenance and employee login activities.

Test Scenario	Description
Testing and validating login menu	Test user login for valid user name and password. User login should identify the user's privileges, name. Only the users which are registered users should be allowed to login.
Main Menu with deferent user privileges	Test whether the metro style menu has been correctly updated according to the logged user level.
Tile information in the metro stile tiles	Test whether the metro style tile menu has been properly updating according to contain of module.
Add/update/delete item in the item master	Test item master details modifications module. This should allow user to add new item to the system by clicking "New button". Existing item details should retrieve upon selecting the particular item raw in the data grid view and then updating the details in order to edit the item details. After adding or updating the employee or editing the item details save option should save the item details to the database.
Add/update/delete item in the Suppliers	Test the supplier module by adding/ updating/ deleting a supplier.
Check the reports	Checking the reports and checking the accuracy of the reports
Check the batch number auto generation	In the good reserving notes checking weather correctly appearing the correct batch number

System Testing

Scenario 1: Testing and validating login menu

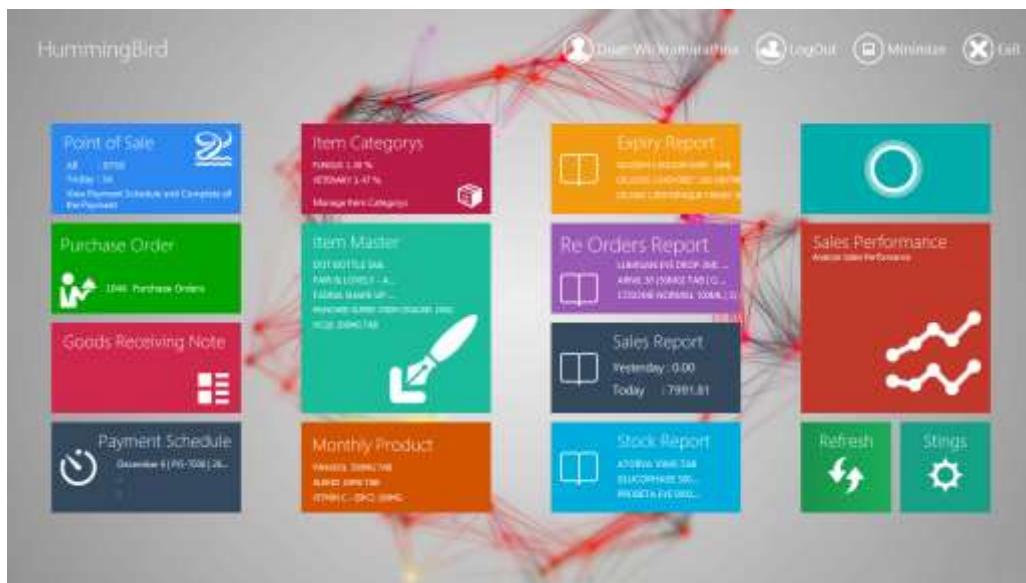


Sample Data: Wrong Password

A Wrong Password message has been aspired

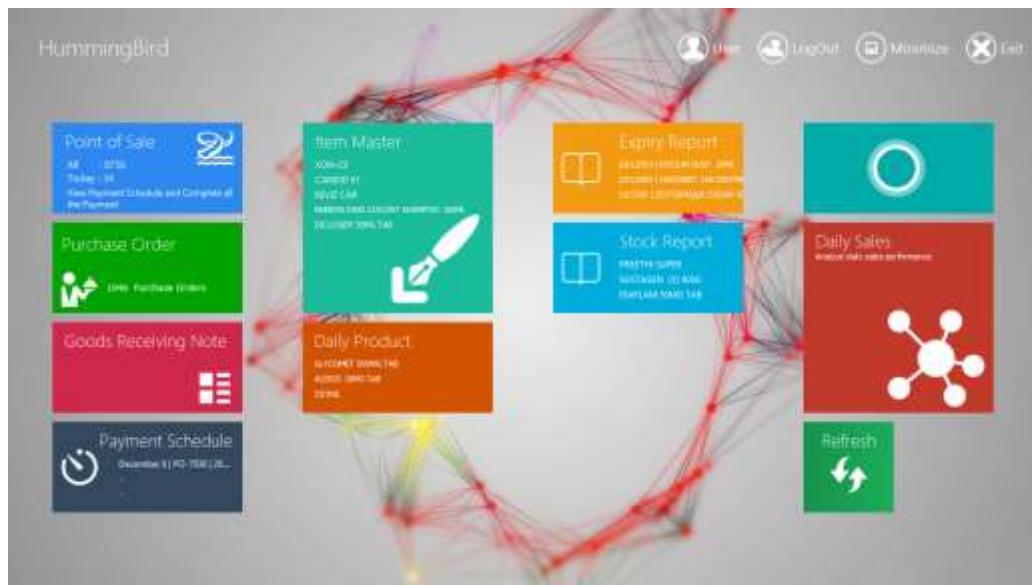
Statuses: Success

Scenario 2: Main Menu with deferent user privileges



Sample Data 1: Administrator Username and password (Dilan, 123)

From the admin user level all modules has been enabled, all menus are visible in the menu.



Sample Data 2: General User and password (User, 123)

From the admin user level all modules has been enabled, all menus are visible in the menu.

Statues: Success

(All other test cases are attached to the appendix.)

7.0 Implementation

Implementation Plan is designed to identify how the developed system will be deployed and transitioned into an operational system. Plan will provide an overview of the system, major tasks which will be involved during the implementation and other requirements which supports to the implementation process.

7.1 Major tasks on system implementation

Some of the major task on the implementation process of the developed system is listed in the below.

- Planning overall implementation and co-ordination of the system implementation
- Testing the server for internet connection
- Identifying resource requirements for the implementation
- Providing training sessions for the users
- Ensure all the required hardware and software are available

7.2 Data Conversion / Loading

When an information system is installed and introduced it requires the data entered from the existing system. The existing system could be a manual system or computerized system or both of the systems. The data from those systems must be collected, converted into necessary formats and entered to the new system. This data conversion and loading is essential to smooth process of the new implemented system.

When considering the developed Analytical, Intelligent, Business and Inventory Automation Systems (AIBIAS), it has been developed for an alternative for the existing manual task allocating procedure. Details such as employee records and task details have to be freshly and manually entered into the database through the system interfaces.

7.3 System Changeover

The Parallel Change Over methodology is recommended for the implementation of the Analytical, Intelligent, Business and Inventory Automation Systems (AIBIAS) in the Oriental Pharmacy environment. In parallel change over, the new developed system will be put into use while using the current task allocation procedure also for a certain period of time. During that period the Analytical, Intelligent, Business and Inventory Automation Systems (AIBIAS) is tested for the system testing and acceptance test with its users of the environment where it is installed. The existing system continues operating until the new Analytical, Intelligent, Business and Inventory Automation Systems (AIBIAS) is tested thoroughly and determines to be error free and ready to operate independently.

Advantage of using parallel changeover is the new system will get ample of time to be observed for testing while minimizing the risks can be occurred by system failure. It will also enable users to get trained while using the old system.

7.4 User Training

The users are a main part of the developed system. It is essential for users to know about the available functions and how to use them properly to take the maximum benefits of using the system after implementation.

There are two types of user roles defined in the system namely General Users and Administrators. General users are users who use the Analytical, Intelligent, Business and Inventory Automation Systems (AIBIAS) for day today business management purposes and administrators are responsible for administrative functions and maintaining the system to keep it operational. The recommended training methods are as follows.

General Users	System Administrators
Training Sessions for hands-on experiences	Training Sessions for hands-on experiences
Practical exercises	Practical exercises
One-on-one tutorials	Questions and answers
Group training sessions	Group training sessions

7.5 Summary

The system development activities that occur after the system design stage and the activities that occur before the delivery of the product called implementation were expressed. The program development part described the order of implementation, language selection and tools. Testing is the independent from the system development and the testing has done in several test runs to determine all the defects. Implementation plan has the sequence order of set of activities such as installation, data conversion, and change over and user training.

8.0 Evaluation

This chapter will present the reader a detailed evaluation on project and the developed product while evaluating the objectives were achieved, how the project deviated from its original specifications and the problems aroused during the lifetime of the project. Furthermore it will also discuss the remedies that have been taken to overcome the aroused problems and knowledge which have been gathered by providing solutions for such issues. On the other hand as product evaluation, it will further discuss about the product while evaluating how far it met the functionally requirements set by users during the initial stages of the project.

8.1 Evaluation of Project Practice

Evaluation of project practice is done to evaluate the project plan and the deviation occurred during the phases of project. In this sub section it is also discussed about the successfullness of the project and the obstacles faced during the project and how they have being tackled. Phases of the project identified as to carry the evaluation are analysis, design and implementation.

8.1.1 Analysis phase

Analysis phase is responsible on identifying the requirements of the users for the development of the new system. It is done using fact-finding techniques such as interviews, observations, questionnaires and other. This is a vital process since successful identification of user requirements will help to complete the project successfully on time with less number of modifications after implementation.

Key process of Analysis phase

- Fact finding
- Identifying problems and limitations of the current system
- Modeling current system
- Identifying functional and non-functional requirements for the system
- Modeling the proposed system

8.1.1.1 Evaluating Analysis phase

As the system scope defines the end users of the developing system are in higher levels of the Oriental Pharmacy organizational hierarchy, they have a very busy work schedule. Due to that fact difficulties were aroused in conducting interviews with accordance with the planned schedule. But other methodologies such as questionnaires, observations and document reviews were greatly helpful in the data gathering process. Identifying different users of the system and involving them in the data gathering phase helped to look at the system from different perspectives. Although the busy schedules of the participants in the data gathering phase made

the analysis process to drag over the scheduled deadlines, as an overall the process of requirement collection and analysis was completed successfully with great help of staff members and management of Oriental Pharmacy.

8.1.1.2 Learning Experiences

Since the analysis phase is carried with the assist and guidance of higher rank official in the organizational hierarchy, it helped to learn about the how to analyze the situation critically by looking at the core of the problem scenario. Along with that, since most of the personnel's who involved in the process has high academic qualifications and experience, always they gave guidance and advises to make this project successful with accordance with their experience. Gathering information from different kind of staff members enabled to look at a problem in different views and have a clear picture about it.

In addition to the project analysis processes, it helped to understand some of the official procedures which are maintained in the daily routines inside the Oriental Pharmacy.

8.1.2 Design phase

This is a vital phase on the project because design can be named as the blue print of the system which is going to be developed. Designing of system components, databases, windows application and mobile applications interfaces and overall system architecture will be carried out during this phase.

Key Processes of Design Phase

- Designing system, software and module architecture
- Designing database
- Designing interfaces
 - Windows Application
 - Android Application
 - Windows Phone Application
- Integration of modules

8.1.2.1 Evaluating Design phase

With the successful identifications of user requirements after the completion of the analysis phase of the project, design phase was conducted without many difficulties. Overall system architecture was designed with the major functions and components identified on the proposed system. Software architecture has been designed to give a clear picture about the components

of the system and communication between them. Database was initially logically designed using ER diagrams and then converted into set of relations to make the proper database. Interfaces were designed successfully along with ideas from users to make it as much as user friendly and it also were helped to develop the prototype of the system.

8.1.2.2 Learning Experiences

Design phase allowed gaining experience on deriving important data which needed to the product development from the data gathered during analysis stage. It also helped to identify how the design theories can be applied in real time scenarios. For example in the case of designing the dataset, theories for mapping a logical database to tables were greatly helpful in identifying the attributes, primary keys and foreign keys.

8.1.3 Development and testing and implementation phase

This phase is responsible for all the development and construction, testing and implementation of the system. The coding of the system is done and several testing strategies are used to test the system before implementation to provide an error and bug free system.

Key processes of Development phase

- Develop the databases
 - Tables, Views, Stored Procedures and SQL Jobs
- Develop the common layers
 - Data layer, Modeler layers, Web services
 - API Layers
- Develop the Interfaces
 - Mobile (Android, Windows Phone)
 - Windows
- Make the combination and relation between interfaces and layers

Key processes of Testing phase

- Identifying testing strategies
- Design of Test plan
- Identifying major tasks on implementation

Key processes of Testing phase

- Planning implementation
- Make the implementation document

8.2 Evaluation of Product

Evaluation of the project is done to make sure the newly developed system has met the functional and nonfunctional requirements requested by the clients. Each component of the system is evaluated using several methods along with getting feedback from the users.

8.2.1 Methods of Evaluation

In order to complete this phase successfully several evaluation methods were planned considering different users of the system. Methods which were used are,

- Interviews

After the system has been implemented and the user training programs have been conducted, several employees of the institute were interviewed. In the interviews they were given the functional requirement which were obtained in the data analysis phase and evaluated according to them how far the system has achieved them.

- Questionnaire

In order to identify the failures of the system and future enhancements to the system, a suitable questionnaire was distributed.

- Observation

Since the parallel changeover was adopted in the implementation phase, the use of new system and the old system was observed to check whether the new system is functioning properly and to determine whether the user training has been successful.

8.2.3 Evaluation Criteria

The evaluation of product was done by examining the problem arisen and the solution provided by the developed system, comparing the functional requirements with the components developed in the system to provide them. Apart from them system is also evaluated using the user friendliness of the system. System is evaluated considering following criteria,

- Achieving specification requested by users
- Smooth functionality of the system
- User friendliness
- Efficiency of the system

8.2.3.1 Evaluating Functional Requirements

Functional requirement specified by the user at the analysis phase was compares with the components developed to address them.

Functional Requirements	User Type	Module
Ability to manage users	Administrator	Application general settings
Ability to manage user levels and access permissions	Administrator	Application general settings
Ability to manage mobile devices access (Android, Windows Phone)	Administrator	Application general settings
Ability to send mobile notification	Administrator, Manager and Automated process	Application general settings
Ability to access dashboard	Administrator, Manager and authorized users	Main Menu
Ability to assess reports	Administrator, Manager and authorized users	Reports
Ability to access Windows Phone application	Authorized IEMI numbers of the mobile devices	Windows Phone and web service
Ability to access Android Point of Sales application	Authorized employees	Android tablet device and web service
Ability to do a Sales	Authorized employees, any authorized user	Point of sales in main application and, android application

Ability to manage the general products details	Authorized employees, any authorized user	Main application, Product controls
Add a product to the inventory, edit, change	Authorized employees, any authorized user	Item Master
Ability to do a product order	Authorized employees, any authorized user	Product Order
Ability to add products via good reserving	Authorized employees, any authorized user	Good reserving notes
Ability to change the products in stock, manage product, edit, change	Authorized employees, any authorized user	Stock control
Ability to view payment schedule	Authorized employees, any authorized user	Payment schedule

8.3 Significance of the Solutions

The following table shows the key problems found in the manual procedure and the solutions provided by the newly developed **Analytical, Intelligent, Business and Inventory Automation Systems (AIBIAS)** to address those problems.

Solutions to product domain			
Problem to Address	Solution Provided	Justification	Remarks
No proper way to make the product in well formatted document	Application user can add products to database and retrieve printable documentation from the on time customizable reports	In early stages, when studying the problems and limitations of the system this was identified as a major issue. In the requirement specification for the new system this function was given a high priority.	Successful
No specific task for any employee all are	By assigning specific user level and managing the	This was also a common issue mentioned by users	Successful

mixed and doing what they want	access to each employee to specific works by the system from user controls and privileges	and management in the data gathering process. In the user and administration module, this feature was included.	
No proper categorization to any products in the inventory	In the product controls, the product categorization has been included and from the report and on time categorization to real-time decision on the product grid views.	This was also a common issue mentioned by users in the data gathering process. In the product controls, stock control, and item master, module, this feature was included.	Successful
No supplier contacts to specific product	When adding product to the inventory, the supplier requirement has been included and the supplier monument has been controlled with the goods reserving notes	To order the product, there was not a proper documentation and no proper contacts with them, the supplier to specific product or category has been included as a module in product controls and employees can view in reports or in the stock control as well as item master. It also managed by the good reserving notes	Successful
Data analyzing	As real-time and intelligent function the Dashboard has been included as the solution to analyze the sales and business as well as the productivity and growth and all of business processes to take decisions and make changes of the business way and identify the process of business and how the business and sales has been lined up as the time period.	As the pre requirement that mentioned by users in the data gathering process, the data analyzing module The Dashboard included to fix the problem.	Successful
On-time communication between employee	The notification API services that provided with the Mobile applications (Android and windows	To decision making process the fast notifications are required as a smart features.	Successful

and Management as well as Owners	phone) has been used to notify some information to the users and owners.		
---	--	--	--

8.4 Summary

This chapter was dedicated to provide a detailed evaluation on the project practices and the product developed to the reader. Process evaluation was done by evaluating phase by phase while product was evaluated discussing the problems of the current system and functional requirements of the new system.

9.0 Conclusion

The aim of this chapter is to conclude the project through discussing the overview of the achieved project result and project achievements along with lessons learnt. Furthermore it is discussed about the future enhancements which can be used to improve the productivity of the system.

9.1 Project Results Overview

The aim of this system development project was to build an **Analytical, Intelligent, Business and Inventory Automation Systems (AIBIAS)** that improves the flexibility of the internal and external sales, report generation, business analyzing communication regarding to various responsibilities achieved in Oriental Pharmacy by employees. In the early stages of the research problems and limitations were found out and data gathering and analysis was done to identify user requirements to overcome those problems. In the end of the project a working prototype of a computerized Analytical, Intelligent, Business and Inventory Automation Systems was developed to address those issues along with several value added services over the process.

9.2 Overview of Practice and Lessons Learned

Starting from March 2014, this system development project was carried until the end of year 2013 which means a total of 8 months. In these 8 months researchers had to engage in several phases relevant to the System Development Life Cycle. Each engaged phase had its own set of tasks to do and objectives to achieve. In this long process several lessons were learned and experience was gained. In the following table it is discussed about the Overview of Practice and lessons learnt in each phase.

Phase	Objective	Overview of Practice and Lessons Learned
Literature Review	To identify what is a computerized Analytical, Intelligent, Business and Inventory Automation Systems and to study similar existing system	A good literature review on the project base idea will set a good foundation for the project. Several resources from the internet were used to complete this chapter successfully. During this chapter a thorough knowledge within project scope as well as outside the scope about Analytical, Intelligent, Business and Inventory Automation Systems have been gained. It includes areas such as functions, suitable technologies, advantages/disadvantages have been studied.
Data Gathering and Analysis	To produce the functional and non-functional	Using several techniques data has been gathered and after analyzing the data set of requirements has been produced.

	requirement specification for the system	In this phase experience and knowledge gained on how to apply data finding and analyzing theories in real system development environment. Detailed explanation is provided on chapter Evaluation section 8.0
Design Phase	To design the overall system architecture, module architecture, database and interfaces.	Using diagrams architecture was clearly presented and using proper programming languages the interfaces were designed. In this phase several new frameworks consisting rich inbuilt functionalities based on .Net and Cross platform language which can be used to develop applications easily and efficiently were discovered and learnt. For more details on this phase refer to chapter Evaluation section 8.0
Testing and implementation	To testing the developed solution and implement the system in real environment	The system was completely developed and tested for errors and bugs, testing strategies have been used to improve the yield of testing. Implementation was carried out adapting to the parallel changeover method. Experience on use of testing strategies in the real environment and method of implementing the system according a plan was gained through this phase. For more details on this phase refer to chapter Evaluation section 8.0.
Evaluation Phase	To evaluate process and product to ensure whether it has achieved its aim and objectives.	In this phase process of the project and product has been evaluated separately along with the feedbacks from the users through interviews, observations and questioners. This phase helped to gain knowledge on how to evaluate a process of a project from different perspectives. Furthermore it assisted in identifying the fields that need to be evaluated when evaluating a software project

Table 4 Overview of Practice and Lessons Learned

Learning experiences gained throughout the project are much important to face the challenges in future project developments as well as day to day tasks. Some key things gained from the project throughout the entire project life are

- Helped to improve the business and IT project management skills.

- Working on given time scheduled.
- New programming languages and tools available.
- Improved writing skills by preparing a comprehensive project report

Hence, it is the belief of researchers that the lessons learnt and experience gained on this project can be successfully incorporated in conducting researches in different context so that the researchers can produce quality outcomes.

9.3 Path to Future Enhancements

As this project was carried out as 8 month project while engaging with the other academic studies researchers had to limit the scope which is stated in the introduction chapter. Stated scope defines the system for a part of organizational hierarchy only. As in the future, the scope of the use of this product can be widened to the entire organizational hierarchy.

In addition to widening the scope, several new functionalities can be added in the long run. Some of desired examples for new functions are document uploading function relating to tasks, provide ability to analyze business, enable exchange of notification through the system directly can alerts to mobile devices (Windows or Android). Furthermore, the producing reports function and dashboard module can be enhanced by introducing new set of customizable reports.

Furthermore for since the developed IBAIS is a windows application, for window mobile device and android tablet users an application on famous platform such as Apple and blackberry can be developed so that user can retrieve the updates to their deferent mobile application.

9.4 Conclusion to the Project

The project was started on March 2014 and continued throughout the year. Project consisted of main phases and each phase was aligned with a set of goals to drive the project to the success. Initially a literature review was done to gain a firm foundation on the project theme.

As the next step data gathering was conducted using suitable techniques to gather data from user and the gathered data have being analyzed to produce the requirement specification for the new system. Then according to the specification produced the architecture of the system, data base design and interface designing were conducted in the designing phase. Afterwards the software was coded and tested using apposite testing strategies to find errors and bugs before the implementation to provide a reliable software solution.

After the implementation users feedbacks were taken during the evaluation process using several techniques and it was used to evaluate the product while evaluating the project process also. To represent the project a comprehensive project report was prepared.

From this research study, it has proven that there was an timely need of an computerized Analytical, Intelligent, Business and Inventory Automation Systems which will provide utter advantages over the manual task allocating procedure and by implementing the developed IBAIS that need has being fulfilled lucratively up to a certain scope.

9.5 Summary

As a summary in this chapter, in the first section contains an overview of the project result. Then it was discussed about the project process and learning outcomes during the project life time. Furthermore as it is very vital discuss the future enhancements for the project, next section was dedicated to suggest some developments as future enhancements. Finally this chapter ends with a brief conclusion about the activities done from start to the end of the project.

References

- Bright, P. (2014) Visual Studio goes cross platform with Cordova integration from Microsoft, *Ars Technica*, [online] Available at: <http://arstechnica.com/information-technology/2014/05/visual-studio-goes-cross-platform-with-cordova-integration-from-microsoft/> (Accessed 20 November 2014).
- Blogs.msdn.com, (2014) Opening up Visual Studio and .NET to Every Developer, Any Application: .NET Server Core open source and cross platform, Visual Studio Community 2013 and preview of Visual Studio 2015 and .NET 2015 - Somasegar's blog - Site Home - MSDN Blogs, [online] Available at: <http://blogs.msdn.com/b/somasegar/archive/2014/11/12/opening-up-visual-studio-and-net-to-every-developer-any-application-net-server-core-open-source-and-cross-platform-visual-studio-community-2013-and-preview-of-visual-studio-2015-and-net-2015.aspx> (Accessed 20 November 2014).
- Client, I. (2014) Implementing GCM Client | Android Developers, *Developer.android.com*, [online] Available at: <https://developer.android.com/google/gcm/client.html> (Accessed 20 November 2014).
- Codemag.com, (2014) Layered Architecture, Dependency Injection, and Dependency Inversion, [online] Available at: <http://www.codemag.com/Article/0705071> (Accessed 20 November 2014).
- Developer.xamarin.com, (2014) Notifications | Xamarin, [online] Available at: http://developer.xamarin.com/guides/cross-platform/application_fundamentals/notifications/ (Accessed 20 November 2014).
- Developer.xamarin.com, (2014) Walkthrough - Using Local Notifications in Xamarin.Android | Xamarin, [online] Available at: http://developer.xamarin.com/guides/cross-platform/application_fundamentals/notifications/android/local_notifications_in_android_walkthrough/ (Accessed 20 November 2014).
- Developer.nokia.com, (2012) SOAP web service request with credentials for Windows Phone 7, [online] Available at: <http://developer.nokia.com/community/discussion/showthread.php/234133-SOAP-web-service-request-with-credentials-for-Windows-Phone-7> (Accessed 20 November 2014).
- Documentation.devexpress.com, (2014) Online Documentation - Developer Express Inc., [online] Available at:

- <https://documentation.devexpress.com/#windowsforms/clsDevExpressXtraGridGridControltopic> (Accessed 20 November 2014).
- Dogi, G. (2014) Step by step 3 tier architecture in asp.net using c# example, *Onlinebuff.com*, [online] Available at: http://www.onlinebuff.com/article_step-by-step-3-tier-architecture-in-aspnet-using-c-example_45.html (Accessed 20 November 2014).
- Hadyelsahar Web Blog, (2011) Building and Consuming Web services in Windows phone 7, [online] Available at: <http://hadyelsahar.wordpress.com/2011/10/08/webservices-in-wp7/> (Accessed 20 November 2014).
- Hmkcode.com, (2014) Android Google Cloud Messaging Tutorial | HMKCode, [online] Available at: <http://hmkcode.com/android-google-cloud-messaging-tutorial/> (Accessed 20 November 2014).
- Krill, P. (2014) Xamarin teams with Microsoft on cross-platform mobile development, *InfoWorld*, [online] Available at: <http://www.infoworld.com/article/2609715/microsoft-net/xamarin-teams-with-microsoft-on-cross-platform-mobile-development.html> (Accessed 20 November 2014).
- Kumar, D. (2012) Consuming WCF SOAP Service in Windows 8 Metro Application, *Debug Mode*, [online] Available at: <http://debugmode.net/2012/06/27/consuming-wcf-soap-service-in-windows-8-metro-application/> (Accessed 20 November 2014).
- Mono-project.com, (2014) Documentation | Mono, [online] Available at: <http://www.mono-project.com/docs/> (Accessed 20 November 2014).
- Msdn.microsoft.com, (2014) Chapter 5: Layered Application Guidelines, [online] Available at: <http://msdn.microsoft.com/en-us/library/ee658109.aspx> (Accessed 20 November 2014).
- Msdn.microsoft.com, (2014) Asynchronous Programming with Async and Await (C# and Visual Basic), [online] Available at: <http://msdn.microsoft.com/en-us/library/hh191443.aspx> (Accessed 20 November 2014).
- Msdn.microsoft.com, (2014) Cross-Platform Development with the Portable Class Library, [online] Available at: [http://msdn.microsoft.com/en-us/library/gg597391\(v=vs.110\).aspx](http://msdn.microsoft.com/en-us/library/gg597391(v=vs.110).aspx) (Accessed 20 November 2014).
- nokia, D. (2014) Using Telerik RadCartesianChart to create a game scoreboard - Wiki, *Developer.nokia.com*, [online] Available at:

http://developer.nokia.com/community/wiki/Using_Telerik_RadCartesianChart_to_create_a_game_scoreboard (Accessed 20 November 2014).

Schouler, P. and anzeigen, M. (2010) .NET - Red zone : Best practices and latest stuff: RIA Services: Windows Phone 7 and SOAP endpoint, *Dotnet-redzone.blogspot.com*, [online] Available at: <http://dotnet-redzone.blogspot.com/2010/07/ria-services-windows-phone-7-and-soap.html> (Accessed 20 November 2014).

Support.microsoft.com, (2014) How to call a Visual C# method asynchronously, [online] Available at: <http://support.microsoft.com/kb/315582> (Accessed 20 November 2014).

Telerik.com, (2014) Overview | UI for Windows Phone 7 Documentation, [online] Available at: <http://www.telerik.com/help/windows-phone/radchart-overview.html> (Accessed 20 November 2014).

upadhyay, j. and profile, V. (2012) Windows Phone Application Development: The use of SOAP web services in windows phone applications, *Upadhyayjitesh.blogspot.com*, [online] Available at: <http://upadhyayjitesh.blogspot.com/2012/12/the-use-of-soap-web-services-in-windows.html> (Accessed 20 November 2014).

Appendix A - Project Proposal

1.0 Introduction

The business community plays an important role in the continued economic development and growth of the nation. Within this business community, various business professionals, or businessmen, play a key role in carrying out their duties that contribute to the development of the business.

Owners of business have a busy schedule since they has to deal with clients, investors as well as employees who work along. It is not always possible to physically attend in each these areas. Yet his ultimate goal of running a business is to gain profit. Therefore, it is very essential to keep an eye on the **inventory, stock and sales** that directly deals with profit.

A Business always depend on profit, it also main thing of a business, When specializing the profit,

Those factors are mainly contribute with organization or company. By using the **Technological business solution** to motivate and manage to manage those factors, Owners of business can reduce disequilibrium of the organization or company.

As such using **Technological business solution** with the business these areas of a business facilitates an effective management of

- Flow of materials
- Utilizing people
- Utilization of equipment
- The Co-ordination of
 - Internal activities
 - Communication with
 - Owner
 - Customers

“ISANKA Pharmacy” Limited, the first branded retail Pharmaceutical Chain in Gampaha entered the market with a read of making a distinction within the retail pharmaceutical trade. Headed by a team of pros, “ISANKA Pharmacy” has introduced associate innovative idea focused around superior client care, a good product assortment, reasonable costs and a number valuable additions. Introduction of those innovative ideas has resulted in positive increase in sales volumes and a loyal client base. But currently ISANKA Pharmacy in a problem with data management,

At the discussion, it was revealed that the system which is in operation at the other branch of the “ISANKA Pharmacy” is a manual system. The major issues that are been dealing with the currents system are as follows

2.0 Presenting Problem

- Poor Communication, since manual inventory system doesn't facilitate the communication within the geographic point.
- Errors with Physical calculations, Physical inventory counts should be performed oftentimes to manage the things within the inventory.
- Lack of a system to properly track daily purchases
- Ordering provides, a manual inventory system doesn't update at the top of the day with updated inventory counts.
- Businessman do not have quick access to data concerning the Business

3.0 Application Proposal

Hummingbird is the application solution that proposed as required to the “ISANKA Pharmacy”

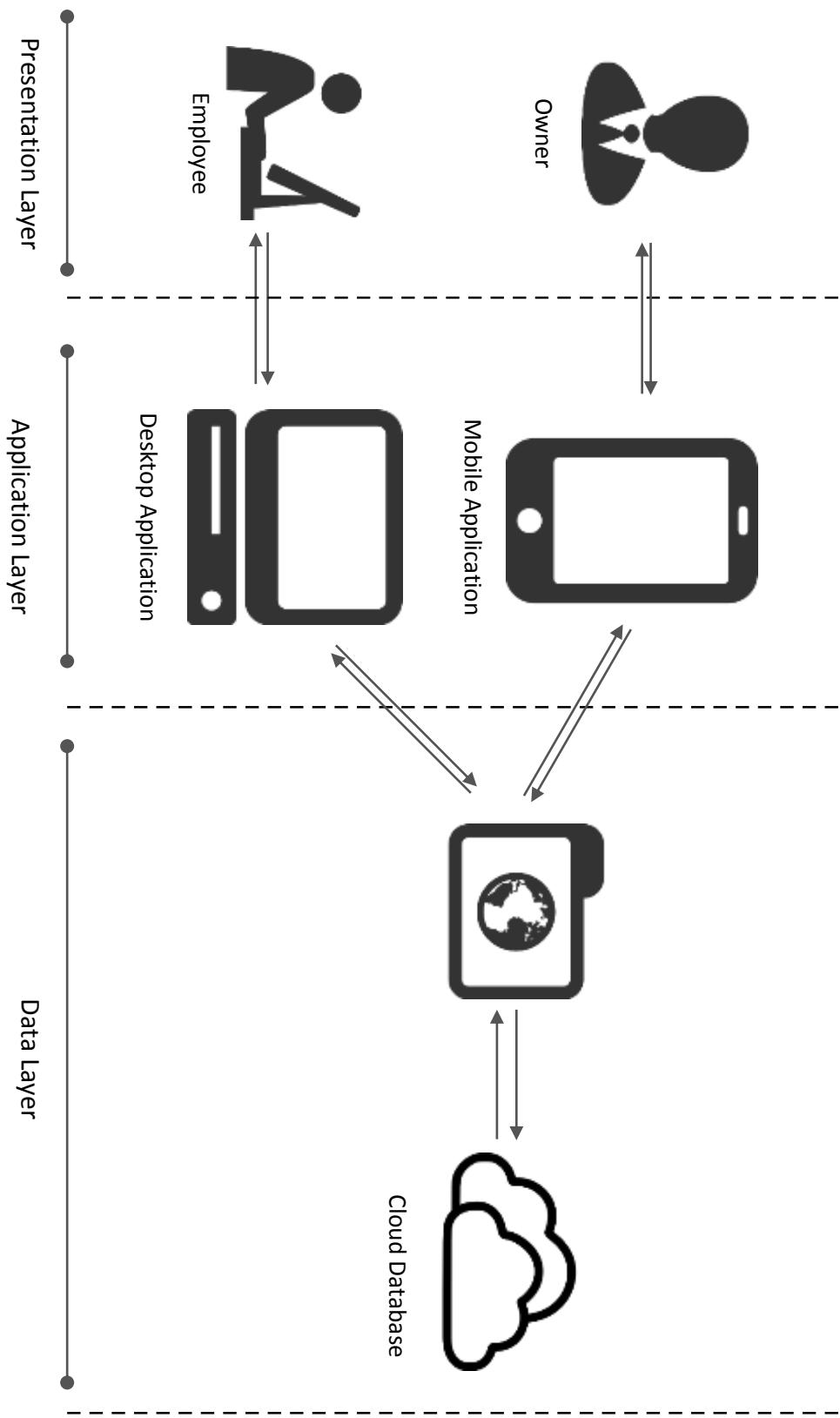
With the **Hummingbird** there are two solutions available,

- Mobile Application
 - The Mobile application is mainly supplied for the owner of business to interact with mobile device.
- Windows Application
 - Windows application is provided to the employee, which is the base application for the Mobile application, while submitting some of important data that allowed to communicate or pre assigned to notify to the owner via mobile application, windows application will sync to the mobile application via the web services through the internet.

Completely those two applications are **metro style** based application. Another very important features is “**Detailed Business Dashboard**” concept, those concepts are come up with the both Mobile and

“**Hummingbird**” is accessible from any location whether or not here in Haiti or the state or where around world. The Mobile application even permits to urge Updates concerning this satiation of the business likewise these days Sales, Stock and every one the notifications concerning the sales and inventory.

3.1 Application Layout.





Owner can use the mobile device with the application, by connecting to the internet owner can view the summarized and detailed business dashboard on the mobile screen.

The dashboard is contain with important, pre produced notifications and many information that need to highlight daily business deals the.

- Sales forecast / Chart
- Purchase forecast / Chart
- Movement of stock (In Chart form or Numerical Fig)
- Stock Value
- Loss & Profit

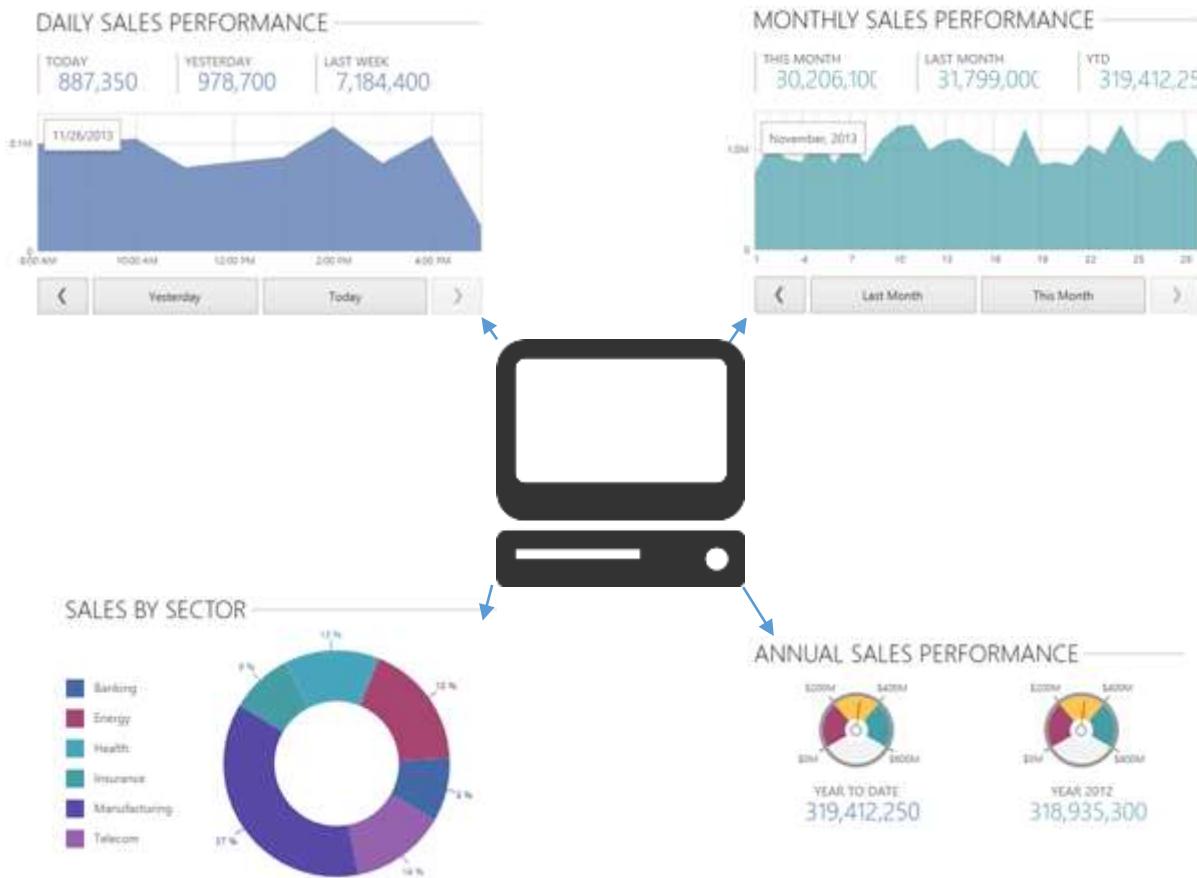




Employee can access to the Main windows application of the HummingBird. Employee is the data entry member, updating invoices, Stock details. Good Reserving notes, Purchase Invoices, and full of handling part is going with windows application by the employee.

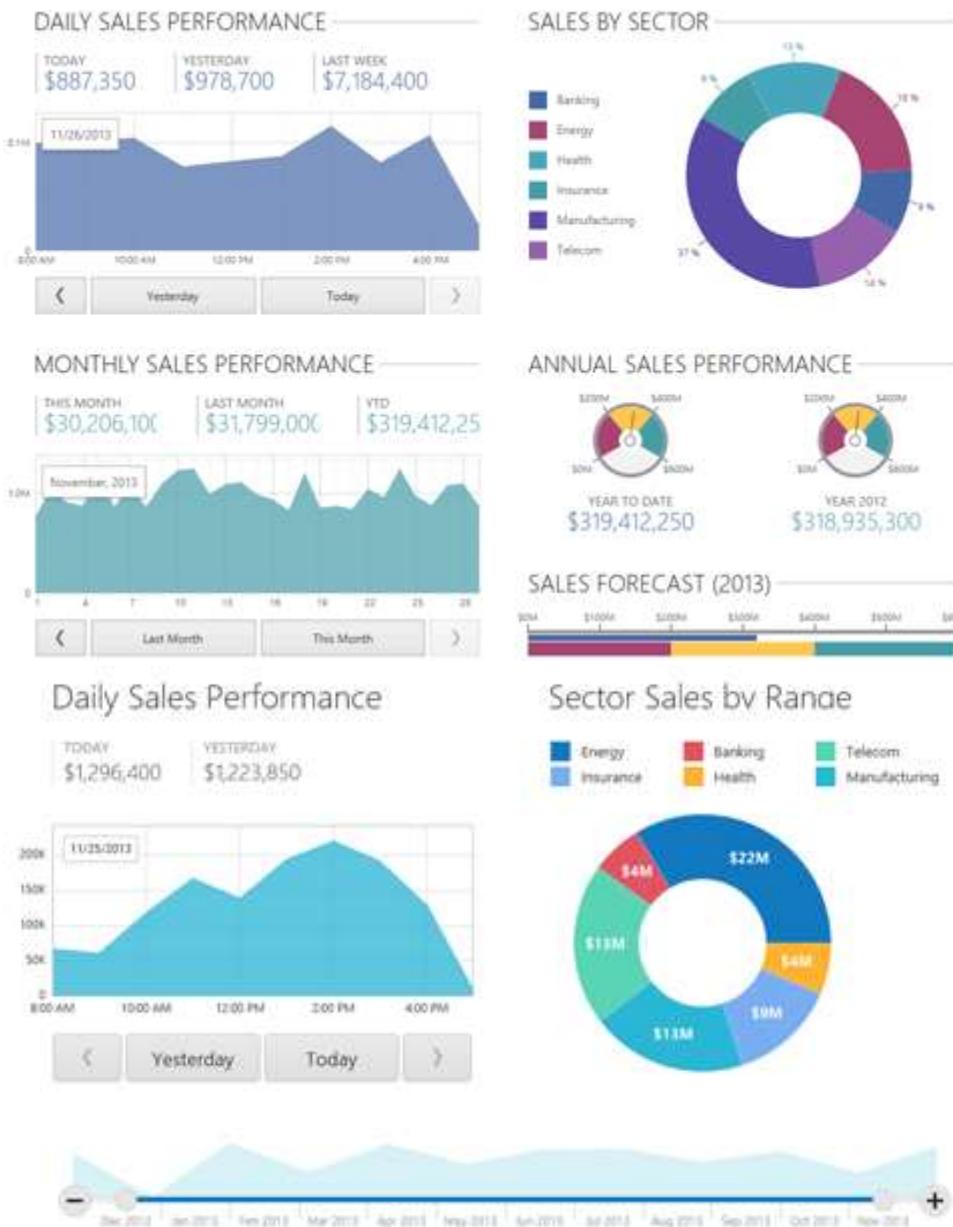
As the most attractive and important component of the HummingBird windows application, the detailed **Dashboard is included to the application** to quick review of the business,

- Purchase order request.
- Movement of stocks.
- Availability of stocks.
- Re-Order levels.
- Stock in demand.
- Expiring stock in hand
- Generating Reports (Automated)



3.2 Functional requirement

- Summarizing the whole business calculation to the graphs and new Gauge Widgets including Dashboard to the application home.



- System should be able to Syncs with Mobile Application
 - Sends new invoices
 - Synchronizes the relevant Updates
 - Send invoice payments as notifications



- System should be able to use Mobile Application Integration for Inventory
 - Business owner able to login through the Mobile Application and see customers, inventory, and everything that's accessible through the online.
- System should be able to process Price Lists

Multiple tariffs is related to one Central inventory, giving to business owner the liberty to supply completely different rating supported location, and/or client cluster. Tariffs may be allotted start/end date for promotional markdowns
- System should be able to Sales Promotions

Freedom to form rules and implement your selection of ads. From BOGO to specific off next purchase, HummingBird offers you the liberty to form, run, and later read powerful reports on Promotions.
- System should be able to Multiple rate Support

Have locations/products with completely different tax rates? No problem! Tax rates is assault the merchandise or location level. Set and manage completely different tax rates for all of your store locations!
- System should be able to Location Specific fast Tabs

Hot sellers could dissent by store location, thus why shouldn't your fast buttons? HummingBird supports store specific quick-tabs.

- System should be able to process Reorder and stock Levels
Automate your ordering method by process Reorder points and stock levels for product.
Later use these rules as recommendation or rule for ordering.
- System should be able to Purchase Order Creation
Create Purchase orders to be sent dead set suppliers by email, or print out and send the hardcopy. Customize text, and add your emblem for that skilled feel!
- System should be able to Purchase Order Receiving
Track commercial instrument standing to visualize what has arrived and what's in transit.
Write off broken product, and track partial orders.
- System should be able to make Stock renewal Report
One report back to actually revolutionize the ordering method. One click of a button adds quantities supported stock level, another click to get commercial instrument documents to the multiple suppliers associated.
- System should be able to process Purchasing Report
Clean report by provider and/or product ordered in given date vary. Read and print from anywhere, at any time.
- System should be able to use with Cloud Hybrid System
Run your POS system each on-line and offline. System is freelance from net access and information is auto-synced upon re-connection.
- System should be able to check Suspend Sale
Place sales on hold and open up at later time for fast checkout. Add product. Found item not within the system? No drawback, add or edit product on the fly right at POS.
- System should be able to process Void/Return
Create returns with ease, choice to print receipts with scalable barcode to hurry up the process!

- System should be able to make Speedy Product operation
Query product by name or code, scan barcodes, or choose things from inventory list to feature them to the group action.

By subscribing with the proposed system, owner and employees can revenue the solutions to the simply noticed problems as below mention.

According to the problems that found by the researching around pharmacy situation, decided solutions can be contemporary with those contains.

Problem

Poor Communication

Since manual inventory system doesn't facilitate the communication within the geographic point.

Solution

A business can avoid communication problems using the **Technological business solution**, by using special function of a system, network based communication functions based on intranet, LAN or Internet or automated other communication media. So also can save the time, Money with responsibility and quality of the company.

Problem

Errors with Physical calculations

Physical inventory counts should be performed oftentimes to manage the things within the inventory.

Solution

A business can avoid Errors that come up with Physical calculations by using **Technological mathematical logics** by using automated calculation with inventory, Invoicing and financial calculations,

Problem

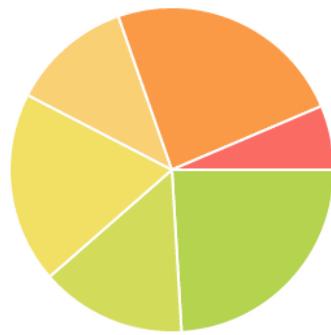
Tracking Daily Purchases

Solution

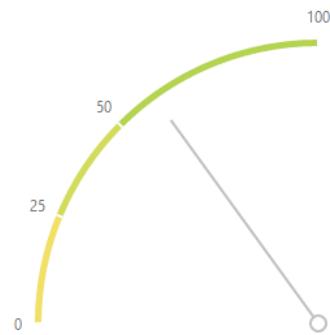
A business can easily trace, make reports and get decision using report generation methods with **Technological business solution**, today world, can automatically generate

any kind of reports, most automated systems are provided that type of function to customers.

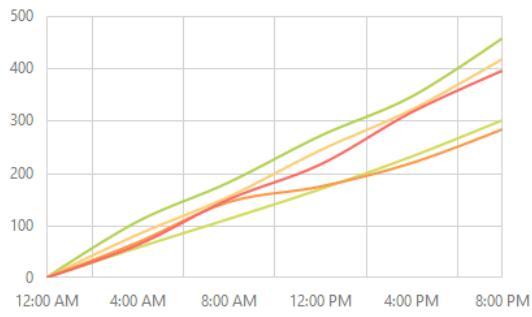
Traffic Distribution (Ad Campaign Summary) —



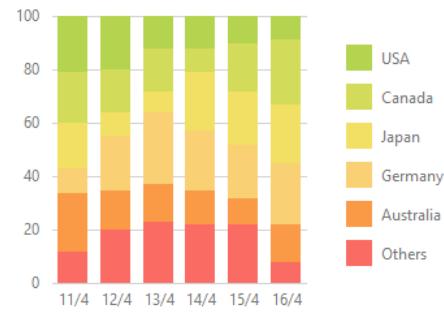
Advertising Efficiency —



Traffic by Hour (Yesterday) —



Traffic by Day (Last Six Days) —



Problem

A manual inventory system doesn't update at the top of the day with updated inventory counts.

Solution

Using the automated alert system business owner, manager or responsible person can take alert about the current situation of the stock, sales daily sales, new updates and all. That path can be mobile alert and Mobile application that specialized for a business. Using that type of facility with the system can name most Transparency of a business.

Problem

Businessman do not have quick access to data concerning the Business

Solution

Today world portability technology can make up to date businessman. Using portable child application with mobile device that made for the current organization or company, can take updates and alerts from anywhere in the world.

3.3 Non- Functional Requirements

- System should be able to provide accessibility irrespective of the location, Portability By giving the mobile application with the proposed application portability and access anywhere in the world, will be achieved.
- System should work in best Performance
Using the cloud data transferring methods and using the fastest devices as example fast computers and fast mobile devices with the new features, employee and owner can enjoy best performance experience with the proposed system,
- Accuracy and Precision
Using Digital calculations in the system, employee and owner can take most accurate and precise details, information and final calculations by the system to take decision.

- Reliability and Security

When doing a business reliability and security is most important thing. By giving the login authentications and using virtual privet (VPN) networks to communicate via the internet and using Security Socket Layer (SSL) with the mobile device data connections with encryptions, employee and owner can trust the application.

- Usability

Usability is most important thing while proposing and introducing new functions or new application to the society, in the proposed system, all of interfaces are design for touch displays and, based on latest concept provided by Microsoft "Windows Metro Style App". Using more space in the desktop, owner or employee can avoid the lots of mistakes and missing points while checking and entering data to the application. It is better than using mouse and keyboard to handle the application.

Metro Style Based System

"Most of you know Metro very well, so here is only basic information. Metro is the name of the design language created for the WP7 interface. Sources of this design included Swiss influenced print and packaging with its emphasis on simplicity, way-finding graphics found in transportation hubs and other Microsoft software with a strong focus on motion and content. Metro has also revitalized third party applications. The standards that have been developed for Metro provide a great baseline, for designers and developers alike. Those standards help them to create successful gesture-driven WP 7 experiences" (Copus, L. Metro style – (dis)advantage of Windows Phone 7 – CodeProject In-text: (Copus, 2012))



3.4 Technical Requirement

To get the best performance, accuracy and expected performance, system should be run with minimal requirements as developer proposed.

In computer side, requirement should be

Processor	Intel core i3 3.2 GHz
RAM	4 GB
HDD	500 GB
VGA	250 MHz
Monitor	15'inchs
Operating System	Windows 8.1 Pro
	Including IO devices.

In Mobile Device side, requirement should be

Processor	dual core 2GHz
RAM	1 GB
HDD	500 GB
Graphic	mobile graphic processor
Monitor	7.5'inchs
Operating System	Windows Mobile 8.1 Pro

Including,

- Internet access with fast internet connection
- Good employee to handle the system

4.0 Significance of the project

In Sri Lanka, few companies are fully automated. Others are semi-automated or still carrying on processing through manually. Manual processing system of stock and sales management are quite time consuming and tedious.

In modern growing market, several businesses square measure, turning toward Automated Intelligent Systems (AIS) to perform their day to day tasks.

The Significance of the proposed system will mainly support in the areas of Stocks, Sales and Management, as per the requirement of the Pharmaceutical.

4.1 Stocks

The main function of this system in the area of stock, will identify the stock mix in a clear pattern to understand the different demands of the stock. The demands are influenced by both External and Internal factors and are balanced by the creation of Purchase Order Request (POR) to keep supplies at a reasonable or prescribed level.

This also will highlight the following;

- ✓ Purchase order request.
- ✓ Movement of stocks.
- ✓ Availability of stocks.
- ✓ Re-Order levels.
- ✓ Stock in demand.
- ✓ Expiring stock in hand.

4.2 Sales

Since the stock been maintained through the system, the sales side will require few more facilities such as

- ✓ Sale Price.
- ✓ Available Stock.
- ✓ Available discounts.

4.3 Managerial Support

While supporting the Sales & Stocks, the same will be assisting the management for improvement, management and information. In additions to the above

- ✓ Sales forecast / Chart

- ✓ Purchase forecast / Chart
- ✓ Movement of stock (In Chart form or Numerical Fig)
- ✓ Stock Value
- ✓ Loss & Profit

Further the system will also give Mobile Notification as per the requested areas of information, especially the Purchase Order and the other importance aspects.

about pc,,, environment. Sys requirnment..
technologies/.... Languages ... and others,,

5.0 Aim and Objectives

Aims is to develop an advanced dynamic **Business Process Automation System** to equip Sales & Marketing of the ISANKA Pharmacy.

The proposed system is the long term target which I am trying to achieve. The Objectives of the proposed system are

- Creating a user friendly environment for the users of the system
- Increase the efficiency of the whole process by 60% with immediate effect on completion of the proposed system by end of July, 2014.
- Minimize paper work.
- Expedite the Stock and Sales handling process by 40% to 60%.

6.0 Scope of the project

ISANKA Pharmacy consists of Accounts department, Sales & Marketing department, Administrative department etc.

This project caters for the requirements of the Sales & Marketing department. This system is capable of covering the entire stock and sales handling process of the Sales & Marketing department.

This system is capable to run on a single computer or a LAN (Local Area Network).

In addition to that this is capable of providing necessary updates and notification to the owner of the company through mobile application.

System is compatible with Windows based computers and a platform of .NET.

7.0 Limitation of Project

When considering the whole project there are some limitation factors such as

- Time
- Knowledge
- Poor user support

In considering the duration of the project it was given short period of time to complete the whole project successfully. Due to the limited time only most important and useful features can be added to the system. Since I was at the early stage of my degree program as well as the development deals with lot of new technologies knowledge has become a limitation on developing the system with diverse advanced technologies.

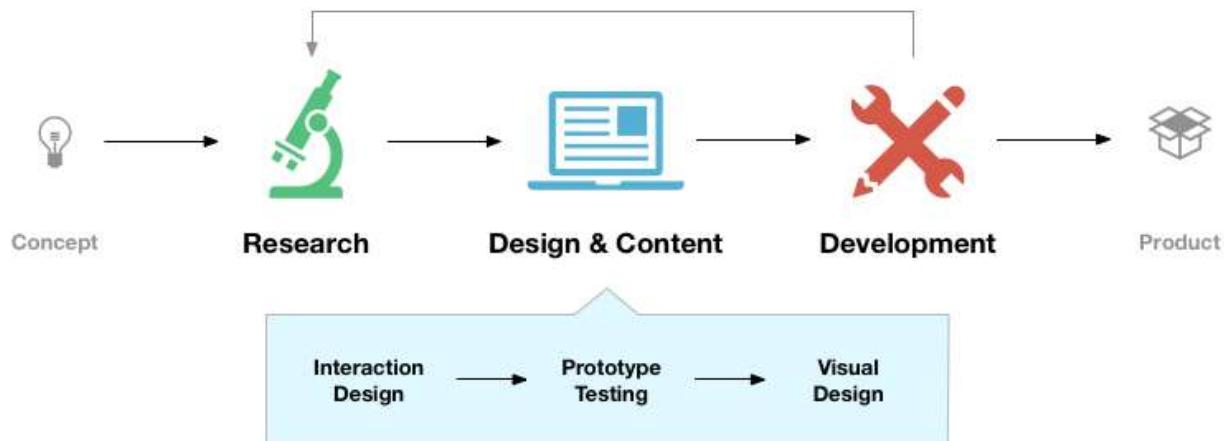
As the end users of this system acquire a busy schedule, allocating time to get the required information for the development and feedback on prototypes is very difficult. Also they have not much explosion of Information Technology it is very difficult in obtaining technical background of the current system. Since development should done with cash in hand adding additional features that consumes finances will be limited.

8.0 Research Methodology

When gathering data to analyze the proposed requirement the Questionnaires and Interview was the best way to understand and make the good plane to achieve targets of the project. The following research methodologies will be used:

- Questionnaire consisting of both open ended and closed ended questions
- Interviews conducted as formal and informal
- Observations will be conducted overtly to understand the work flow
- Document reviews will be used to understand the reporting

Research plan



9.0 Ethical Overview

The Project is designed for the use of ISANKA Pharmacy and its results will only be applicable to its own use. Confidentiality of all data gathered for the research will be safeguarded as a high priority. In acquiring the data, too, all the ethical standards of research will be adhered to. The participant will be informed of their rights and research implications in the initiation of the data gathering process and they are given the leeway to opt out from participation whenever they wish. Without the informed consent from the authorities ISANKA Pharmacy none of the data collection procedures will be conducted.

10 Project Plan

	Task Name	Duration	Start	Finish	Predes	10	Nov 1, '14	Jan 5, '14	Mar 9, '14	May 13, '14	Jul 12, '14	Sep 14, '14	Nov 20
1	Proposal report (Draft)	23 days	Mon 1/20/14	Wed 3/19/14									
2	Proposal Presentation & Report	20 days	Wed 2/19/14	Tue 3/18/14									
3	Manually Scheduled	50 days	Wed 3/19/14	Tue 5/27/14	2								
4	Design: Presentation & Report	48 days	Mon 3/31/14	Wed 6/4/14									
5	Implementation & Development (Prototype): Presentation & Report	178 days	Wed 3/12/14	Fri 11/14/14									
6	Final Presentation	1 day	Sat 11/15/14	Sat 11/15/14	5								

11 Summary

Using a Technological business solution is facilitate an effective management to the business. Owners of business have a busy schedule since they has to deal with clients, investors as well as employees who work along. It is not always possible to physically attend in each these areas. The intelligent business system is related to the “pharmacy with associating with an innovative idea. It is identified the problem where it has a poor communication and facilitate the communication by using a manual inventory system. There daily tracking purchase does not update at the top of the day and they do not have any quick access to their data. The application proposal describe the way of how the applications, data layer and the user can involve in the system. The system is accessible from any location with the cloud based network.

With the integration of a mobile application the Business owner able to login through the Mobile Application and see customers, inventory, and everything that's accessible through the online. The system facilitates by syncing and Sends new invoices from HummingBird to Mobile Application Synchronizes Updates between HummingBird and Mobile Application and it send Special invoice payments from HummingBird to Mobile Application as a notification.

By using these special functions it become an advanced solution for the manual based system with the replacement of using special function of a system, network based communication functions based on intranet, LAN or Internet or automated other communication media. So also can save the time, Money with responsibility and quality of the company.

It aims to develop an advanced dynamic **Business Process Automation System** to equip Sales & Marketing of the ISANKA Pharmacy. With objectives of the proposed system the pharmacy can fulfill the requirements of the Sales & Marketing department. This system is capable of covering the entire stock and sales handling process of the Sales & Marketing department.

References

Bibliography: Copus, L. 2012. Metro style – (dis)advantage of Windows Phone 7 - CodeProject. [online] Available at: <http://www.codeproject.com/Tips/377869/Metro-style-dis-advantage-of-Windows-Phone> [Accessed: 7 Mar 2014].

Bibliography: Microsoft.com. 2014. Modern Design at Microsoft. [online] Available at: <http://www.microsoft.com/en-us/news/stories/design/> [Accessed: 7 Mar 2014].

Bibliography: Msdn.microsoft.com. 2014. Index of UX guidelines for Windows Store apps (Windows). [online] Available at: <http://msdn.microsoft.com/en-us/library/windows/apps/hh465424.aspx> [Accessed: 7 Mar 2014].

Bibliography: Weedmark, D. 2014. Developer's Guide to Windows Metro Style Apps. [online] Available at: <http://jobsearchtech.about.com/od/skills/a/Metro-Style-Apps-Overview.htm> [Accessed: 7 Mar 2014].

Appendix B - Testing

Please refer (2.3.3.2 Analysis of gathered data) for the analysis of gathered data from the interview sessions.

Appendix B: 1

```
// objects created to access in to user controls
private ucLogin ucLog = new ucLogin();
private dlUser dlUs = new dlUser();
private mdUser mlUs = new mdUser();

public Home()
{
    InitializeComponent();
    // set default panel as login
    panelControl.Controls.Add(ucLog);
    ucLog.Dock = DockStyle.Fill;
    // access to button click on logon user control
    ucLog.btLogin.Click += btLogin_Click;
    ucLog.btExit.Click += btExit_Click;
    ucLog.tbPassword.KeyDown += tbPassword_KeyDown;
}

// password key down in login
void tbPassword_KeyDown(object sender, KeyEventArgs e)
{
    if (e.KeyCode == Keys.Enter)
        CheckLogin();
}

bool loginExit = false;

// Exit button click on login
private void btExit_Click(object sender, EventArgs e)
{
    loginExit = true;
    Application.Exit();
}

// login button
private void btLogin_Click(object sender, EventArgs e)
{
    CheckLogin();
}
// login method
```

```

private void CheckLogin()
{
    mlUs.UserName = ucLog.tbUserName.Text;
    mlUs.Password = ucLog.tbPassword.Text;

    var CheckUser = dlUs.CheckUser(mlUs);

    iGlobal.LoginID = mlUs.uID;
    iGlobal.Name = mlUs.Name;
    iGlobal.LevelName = mlUs.LevelName;
    iGlobal.UserName = CheckUser;
    iGlobal.UserLevel = mlUs.UserLevel;

    if (CheckUser == ucLog.tbUserName.Text)
    {
        // set default panel as menu after login
        var ucApp = new AppHome(true);

        panelControl.Controls.Clear();

        panelControl.Controls.Add(ucApp);
        panelControl.Refresh();
        ucApp.Dock = DockStyle.Fill;
        // access to logout button click In tile controller
        ucApp.tileContainer.ButtonClick += tileContainer_ButtonClick;
    }
    else
    {
        lbStatues.Text = String.Format("Wrong Username or Password {0}", mlUs.iMessage);
        flyoutPanel1.ShowPopup();
    }
}

// tile container logout button click
private void tileContainer_ButtonClick(object sender,
DevExpress.XtraBars.Docking2010.ButtonEventArgs e)
{
    var naeme = e.Button.Properties.Caption;

    if (naeme == "LogOut")
    {
        panelControl.Controls.Clear();
        panelControl.Controls.Add(ucLog);
        ucLog.Dock = DockStyle.Fill;
        panelControl.Refresh();
    }
}

```

```

        ucLog.tbPassword.Text = string.Empty;
        ucLog.tbPassword.Focus();
    }
}

private void Home_FormClosed(object sender, FormClosedEventArgs e)
{
    Application.Exit();
}

```

Appendix B: 2

```

private void windowsUIView_QueryControl(object sender,
DevExpress.XtraBars.Docking2010.Views.QueryControlEventArgs e)
{
    e.Control = GetControl(e.Document.ControlTypeName);
}

// get the proper control according to the tile document
protected virtual Control GetControl(string name)
{
    switch (name)
    {
        case "PointofSale":
            return new PointofSale();
        case "PurchaseOrder":
            return new PurchaseOrder();
        case "GoodsReceivingNote":
            return new GoodsReceivingNote();
        case "ItemMaster":
            return new ItemMaster();
        case "StockTransfer":
            return new StockTransfer();
        case "ProductControls":
            return new ProductControls();
        case "PaymentSchedule":
            return new PaymentSchedule();
        case "StockReport ":
            return new StockReport();
        case "ReOrdersReport":
            return new ReorderReport();
        case "ExpiryReport":
            return new ExpiryReport();
        case "SalesReport":
            return new SalesReport();
    }
}

```

```

        case "StockReport":
            return new StockReport();
        case "Information":
            return null;
        case "Stings":
            return new Setings();
        case "Dashboard":
            return new ucDashboard();
    }
    return null;
}

```

Appendix B: 3

```

USE [HummingBird]
GO
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
ALTER PROCEDURE [dbo].[sp_HB_ItemMaster_IUSD]
(
    @RunType      INT,
    @ProductCode   INT      = NULL,
    @ProductDiscription NVARCHAR(MAX) = NULL,
    @ItemCategoryID  INT      = NULL,
    @SuppliersID    INT      = NULL,
    @ProductCompany  NVARCHAR(250) = NULL,
    @GenericName    NVARCHAR(250) = NULL,
    @Country       NVARCHAR(250) = NULL,
    @UnitName       NVARCHAR(5)  = NULL,
    @UnitSize       NVARCHAR(10) = NULL,
    @Brand          NVARCHAR(250) = NULL,
    @ReOrderLevle   INT      = NULL
)
AS
SET NOCOUNT ON
BEGIN TRY
    BEGIN TRANSACTION
        IF @RunType = 1
        BEGIN
            IF EXISTS (SELECT
                1
                FROM HB_ItemMaster
                WHERE [ProductCode] = @ProductCode)

```

```

BEGIN
    UPDATE HB_ItemMaster
    SET [ProductDiscription] = COALESCE(@ProductDiscription, ProductDiscription),
        [ItemCategoryID] = COALESCE(@ItemCategoryID, ItemCategoryID),
        [SuppliersID] = COALESCE(@SuppliersID, SuppliersID),
        [ProductCompany] = COALESCE(@ProductCompany, ProductCompany),
        [GenericName] = COALESCE(@GenericName, GenericName),
        [Country] = COALESCE(@Country, Country),
        [UnitName] = COALESCE(@UnitName, UnitName),
        [UnitSize] = COALESCE(@UnitSize, UnitSize),
        [Brand] = COALESCE(@Brand, Brand),
        [ReOrderLevle] = COALESCE(@ReOrderLevle, ReOrderLevle),
        [AddedDate] = GETDATE()
    WHERE [ProductCode] = COALESCE(@ProductCode, ProductCode)
END

ELSE
BEGIN
    INSERT INTO HB_ItemMaster ([ProductDiscription]
    , [ItemCategoryID]
    , [SuppliersID]
    , [ProductCompany]
    , [GenericName]
    , [Country]
    , [UnitName]
    , [UnitSize]
    , [Brand]
    , [ReOrderLevle])
    VALUES (@ProductDiscription, @ItemCategoryID, @SuppliersID,
    NULLIF(@ProductCompany, ""), NULLIF(@GenericName, ""), NULLIF(@Country, ""),
    NULLIF(@UnitName, ""), NULLIF(@UnitSize, ""), NULLIF(@Brand, ""), @ReOrderLevle)
END
END
IF @RunType = 2
BEGIN
    SELECT
        [ProductCode],
        [ProductDiscription],
        [ItemCategoryID],
        [SuppliersID],
        [ProductCompany],
        [GenericName],
        [Country],

```

```

[UnitName],
[UnitSize],
[Brand],
[ReOrderLevle]

FROM HB_ItemMaster
END

IF @RunType = 3
BEGIN
SELECT
[ProductCode],
[ProductDiscription],
[ItemCategoryID],
[SuppliersID],
[ProductCompany],
[GenericName],
[Country],
[UnitName],
[UnitSize],
[Brand],
[ReOrderLevle]

FROM HB_ItemMaster
WHERE [ProductCode] = COALESCE(@ProductCode, ProductCode)
AND [ProductDiscription] = COALESCE(@ProductDiscription, ProductDiscription)
END

IF @RunType = 4
BEGIN
DELETE HB_ItemMaster
WHERE [ProductCode] = @ProductCode

END

IF @RunType = 5
BEGIN
SELECT
SUBSTRING(dt.Category, 1, 2) AS Category,
dt2.StockID,
(SUBSTRING(dt.Category, 1, 2) + dt2.StockID) AS Batch
FROM (SELECT
HB_ItemCategory.Category
FROM HB_ItemMaster

```

```

    INNER JOIN HB_ItemCategory
        ON HB_ItemMaster.ItemCategoryID = HB_ItemCategory.ItemCategoryID
    WHERE HB_ItemMaster.ProductCode = @ProductCode) dt
LEFT OUTER JOIN (SELECT
    CAST((CASE
        WHEN MAX(StockID) IS NULL THEN 1
        ELSE MAX(StockID + 1)
    END) AS NVARCHAR(10)) AS StockID
FROM HB_MainStock) dt2
    ON dt.Category != dt2.StockID
END

IF @RunType = 6
BEGIN
    SELECT
        IDENT_CURRENT('HB_ItemMaster') + 1 AS ProductCode
END

--Select Generic data
IF @RunType = 7
BEGIN
    SELECT DISTINCT
        ProductCompany,
        Country,
        UnitName,
        Brand,
        GenericName
    FROM HB_ItemMaster
END

--Select ProductCode From Name
IF @RunType = 8
BEGIN
    SELECT
        ProductCode
    FROM HB_ItemMaster
    WHERE ProductDiscription = @ProductDiscription
END

IF @RunType = 9 -- Not Exist Product in Stock
BEGIN
    SELECT
        ProductDiscription
    FROM HB_ItemMaster AS T1

```

```

    WHERE NOT EXISTS (SELECT
        ProductCode
        FROM HB_MainStock AS T2
        WHERE (ProductCode = T1.ProductCode))
        ORDER BY ProductCode
    END

    COMMIT TRANSACTION
END TRY

BEGIN CATCH
    ROLLBACK TRANSACTION;
    THROW;
END CATCH

```

Appendix B: 4

```

CREATE VIEW dbo.ExpiryView
AS
SELECT
    ct.Category,
    ms.BatchNumber,
    im.ProductDiscription,
    ms.ProductCode,
    ISNULL((SELECT
        sp.SupplierName
    FROM HB_POHeader ph
    INNER JOIN dbo.HB_Suppliers sp
        ON ph.SuppliersID = sp.SuppliersID
    WHERE PONumber = ms.PONumber), (SELECT
        sp.SupplierName
    FROM HB_ItemMaster im
    INNER JOIN HB_Suppliers sp
        ON im.SuppliersID = sp.SuppliersID
    WHERE im.ProductCode = ms.ProductCode)
)
    AS SupplierName,
    im.Country,
    CONVERT(VARCHAR(11), ms.ExpiryDate, 101) AS ExpiryDate,
    'This Week' AS ExpiryPeriod,
    CONVERT(VARCHAR(11), ms.AddedDate, 101) AS AddedDate,
    ms.RetailPrice,
    ms.BuyingPrice,

```

```

ms.Quantity,
im.UnitName,
im.UnitSize,
im.Brand,
im.ProductCompany,
im.ReOrderLevle,
im.GenericName

FROM HB_MainStock ms
INNER JOIN HB_ItemMaster im
ON ms.ProductCode = im.ProductCode
INNER JOIN HB_ItemCategory ct
ON im.ItemCategoryID = ct.ItemCategoryID

WHERE DATEPART(WEEK, ms.ExpiryDate) = DATEPART(WEEK, GETDATE())

UNION
SELECT
ct.Category,
ms.BatchNumber,
im.ProductDiscription,
ms.ProductCode,
ISNULL((SELECT
sp.SupplierName
FROM HB_POHeader ph
INNER JOIN dbo.HB_Suppliers sp
ON ph.SuppliersID = sp.SuppliersID
WHERE PONumber = ms.PONumber), (SELECT
sp.SupplierName
FROM HB_ItemMaster im
INNER JOIN HB_Suppliers sp
ON im.SuppliersID = sp.SuppliersID
WHERE im.ProductCode = ms.ProductCode)
)
AS SupplierName,
im.Country,
CONVERT(VARCHAR(11), ms.ExpiryDate, 101) AS ExpiryDate,
'This Month' AS ExpiryPeriod,
CONVERT(VARCHAR(11), ms.AddedDate, 101) AS AddedDate,
ms.RetailPrice,
ms.BuyingPrice,
ms.Quantity,
im.UnitName,
im.UnitSize,

```

```

im.Brand,
im.ProductCompany,
im.ReOrderLevle,
im.GenericName
FROM HB_MainStock ms
INNER JOIN HB_ItemMaster im
ON ms.ProductCode = im.ProductCode
INNER JOIN HB_ItemCategory ct
ON im.ItemCategoryID = ct.ItemCategoryID

WHERE DATEPART(MONTH, ms.ExpiryDate) = DATEPART(MONTH, GETDATE())
GO

```

Appendix B: 5

```

CREATE PROCEDURE [dbo].[sp_HB_CheckLogin_S]
(
    @UserName NVARCHAR(250),
    @Password NVARCHAR(250)
)
AS
IF EXISTS (SELECT
    1
    FROM GetUsers
    WHERE GetUsers.UserName = @UserName COLLATE SQL_Latin1_General_CP1_CS_AS
    AND GetUsers.password = @Password COLLATE SQL_Latin1_General_CP1_CS_AS)
BEGIN
    SELECT DISTINCT
        (CASE
            WHEN (SELECT
                    GetUsers.password
                    FROM GetUsers
                    WHERE GetUsers.UserName = @UserName COLLATE SQL_Latin1_General_CP1_CS_AS
                    AND GetUsers.password = @Password COLLATE SQL_Latin1_General_CP1_CS_AS)
                = @Password THEN @UserName
            ELSE 'Reject'
        END) AS CheckUser,
CASE
    WHEN (SELECT
            id
            FROM GetUsers
            WHERE UserName = @UserName COLLATE SQL_Latin1_General_CP1_CS_AS

```

```

        AND GetUsers.password = @Password COLLATE SQL_Latin1_General_CP1_CS_AS)
    IS NULL THEN 0
ELSE (SELECT
    id
    FROM GetUsers
    WHERE GetUsers.UserName = @UserName COLLATE SQL_Latin1_General_CP1_CS_AS
    AND GetUsers.password = @Password COLLATE SQL_Latin1_General_CP1_CS_AS)
END AS LoginID,

CASE
WHEN (SELECT
    Name
    FROM GetUsers
    WHERE GetUsers.UserName = @UserName COLLATE SQL_Latin1_General_CP1_CS_AS
    AND GetUsers.password = @Password COLLATE SQL_Latin1_General_CP1_CS_AS)
    IS NULL THEN 'Null'
ELSE (SELECT
    Name
    FROM GetUsers
    WHERE GetUsers.UserName = @UserName COLLATE SQL_Latin1_General_CP1_CS_AS
    AND GetUsers.password = @Password COLLATE SQL_Latin1_General_CP1_CS_AS)
END AS Name,

CASE
WHEN (SELECT
    UserLevel
    FROM GetUsers
    WHERE GetUsers.UserName = @UserName COLLATE SQL_Latin1_General_CP1_CS_AS
    AND GetUsers.password = @Password COLLATE SQL_Latin1_General_CP1_CS_AS)
    IS NULL THEN 'Null'
ELSE (SELECT
    UserLevel
    FROM GetUsers
    WHERE GetUsers.UserName = @UserName COLLATE SQL_Latin1_General_CP1_CS_AS
    AND GetUsers.password = @Password COLLATE SQL_Latin1_General_CP1_CS_AS)
END AS UserLevel,

CASE
WHEN (SELECT
    UserLevelID
    FROM GetUsers
    WHERE GetUsers.UserName = @UserName COLLATE SQL_Latin1_General_CP1_CS_AS
    AND GetUsers.password = @Password COLLATE SQL_Latin1_General_CP1_CS_AS)
    IS NULL THEN 0

```

```

ELSE (SELECT
      UserLevelID
      FROM GetUsers
      WHERE GetUsers.UserName = @UserName COLLATE SQL_Latin1_General_CI_AS
      AND GetUsers.password = @Password COLLATE SQL_Latin1_General_CI_AS)
END AS UserLevelID

FROM GetUsers
END
ELSE
BEGIN
  SELECT
    ('WrongUser') AS CheckUser,
    CAST(0 AS INT) AS LoginID,
    ('Null') AS Name,
    ('Null') AS UserLevel,
    CAST(0 AS INT) AS UserLevelID
  END

```

Appendix B: 6

```

private void ITileInfo()
{
  var POSShe = dIMd.POSTileInfo();

  for (int i = 0; i < POSShe.Rows.Count; i++)
  {
    DataRow dr = POSShe.Rows[i];
    PointofSaleTile.Frames[0].Elements[2].Text =
    "All &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;:" + dr["AllInvoice"];
    PointofSaleTile.Frames[0].Elements[3].Text = "Today :&nbsp;" + dr["TodayInvoice"];
  }
  var PORder = dIMd.PUOTileInfo();

  for (int i = 0; i < PORder.Rows.Count; i++)
  {
    DataRow dr = PORder.Rows[i];
    PurchaseOrderTile.Elements[i + 1].Text = dr["POOnWay"].ToString();
  }

  var PaymentShe = dIMd.PaymentScheduleTileInfo();

  for (int i = 0; i < PaymentShe.Rows.Count; i++)
  {

```

```

    DataRow dr = PaymentShe.Rows[i];
    PaymentScheduleTile.Elements[i + 1].Text = dr["Payment"].ToString();
}

var itemMas = dIMd.ItemMasterTileInfo();

for (int i = 0; i < itemMas.Rows.Count; i++)
{
    DataRow dr = itemMas.Rows[i];
    ItemMasterTile.Elements[i + 1].Text = dr["ProductDiscription"].ToString();
}

var StockRpt = dIMd.MainStockTileInfo();

for (int i = 0; i < StockRpt.Rows.Count; i++)
{
    DataRow dr = StockRpt.Rows[i];
    StockReportTile.Frames[0].Elements[i + 1].Text = dr["ProductDiscription"].ToString();
}

var ExpiRpt = dIMd.ExpiaryReportTileInfo();

for (int i = 0; i < ExpiRpt.Rows.Count; i++)
{
    DataRow dr = ExpiRpt.Rows[i];
    ExpiryReportTile.Elements[i + 1].Text = dr["Product"].ToString();
}

var ReOrd = dIMd.ReOrderTileInfo();

for (int i = 0; i < ReOrd.Rows.Count; i++)
{
    DataRow dr = ReOrd.Rows[i];
    ReOrdersReportTile.Frames[0].Elements[i + 1].Text = dr["Info"].ToString();
}

var Sale = dIMd.SalesTileInfo();

for (int i = 0; i < Sale.Rows.Count; i++)
{
    DataRow dr = Sale.Rows[i];
    SalesReportTile.Frames[0].Elements[1].Text = dr["YesterdayProfit"].ToString();
    SalesReportTile.Frames[0].Elements[2].Text = dr["TodayProfit"].ToString();
}

```

```

}

var StockMo = dIMd.StockMonthlyProduct();

for (int i = 0; i < StockMo.Rows.Count; i++)
{
    DataRow dr = StockMo.Rows[i];
    StockTransferTile.Frames[1].Elements[i + 1].Text = dr["MonthlyProduct"].ToString();
}

var StockWe = dIMd.StockWeeklyProduct();

for (int i = 0; i < StockWe.Rows.Count; i++)
{
    DataRow dr = StockWe.Rows[i];
    StockTransferTile.Frames[2].Elements[i + 1].Text = dr["WeeklyProduct"].ToString();
}

var StockDa = dIMd.StockDailyProduct();

for (int i = 0; i < StockDa.Rows.Count; i++)
{
    DataRow dr = StockDa.Rows[i];
    StockTransferTile.Frames[3].Elements[i + 1].Text = dr["DailyProduct"].ToString();
}

var CatRt = dIMd.CategoryRate();

for (int i = 0; i < CatRt.Rows.Count; i++)
{
    DataRow dr = CatRt.Rows[i];
    ProductControlsTile.Frames[1].Elements[i + 1].Text = dr["Category"].ToString();
}

var SupRt = dIMd.SupplierRate();

for (int i = 0; i < SupRt.Rows.Count; i++)
{
    DataRow dr = SupRt.Rows[i];
    ProductControlsTile.Frames[2].Elements[i + 1].Text = dr["Supplier"].ToString();
}

```

Appendix B: 7

```
USE [HummingBird]
GO
***** Object: StoredProcedure [dbo].[sp_TL_MetroData_S]      Script Date: 11/20/2014
10:55:44 PM *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
ALTER PROCEDURE [dbo].[sp_TL_MetroData_S]
(
    @RunType INT
)
AS
    -- Point Of Sales Invoices
    IF @RunType = 1
    BEGIN
        SELECT (SELECT
            COUNT(InvoiceNumber)
        FROM HB_SalesHead)
        AS AllInvoice,
        (SELECT
            COUNT(InvoiceNumber)
        FROM HB_SalesHead
        WHERE CONVERT(VARCHAR, InvoiceDate, 101) = CONVERT(VARCHAR, GETDATE(), 101))
        AS TodayInvoice
    END

    -- Purchase Order Count
    IF @RunType = 2
    BEGIN
        SELECT
            CONCAT(COUNT(hp.PONumber), ' Purchase Orders') AS POOnWay
        FROM HB_POHeader hp
        WHERE hp.Finalize = 'No'
        AND hp.Payid = 'No'
    END

    -- UpComing Payments
    IF @RunType = 3
    BEGIN
        SELECT TOP (3)
            CONCAT([month], ' ', [day], ' | ', [PONumber], ' | ', [Total]) AS Payment
        FROM [PaymentSchedule]
        ORDER BY CONVERT(DATE, PaymentDueDates) ASC
    END

    --Random Products
    IF @RunType = 4
    BEGIN
        SELECT TOP (5)
            ProductDescription
        FROM HB_ItemMaster
        ORDER BY NEWID()
    END

    --Random New Products to Stock
```

```

IF @RunType = 5
BEGIN
    SELECT TOP (3)
        msf.ProductDiscription
    FROM MainStockFull msf
    WHERE CONVERT(DATE, msf.AddedDate)
        BETWEEN
        CONVERT(DATE, DATEADD(WEEK, -1, GETDATE()))
    AND CONVERT(DATE, GETDATE())
        ORDER BY NEWID()
END

--Upcomming Expiarys
IF @RunType = 6
BEGIN
    SELECT
        TOP (3)
        CONCAT(BatchNumber, ' | ', ProductDiscription) AS Product
    FROM ExpiaryViewCat
    ORDER BY CONVERT(DATE, ExpiryDate) ASC
END

--Reorders as Range
IF @RunType = 7
BEGIN
    SELECT TOP (3)
        CONCAT(ProductDiscription, ' | Q : ', Quantity, ' | R : ', ReOrderLevle) AS Info
    FROM ReOrders
    WHERE OrderRange BETWEEN 0 AND 100
        ORDER BY NEWID()
END

--Sales Compair Today and Yesterday
IF @RunType = 8
BEGIN
    SELECT
        CONCAT('Yesterday : ', (SELECT
            Profit
        FROM YesterdayProfit)
    ) AS YesterdayProfit,
        CONCAT('Today      : ', (SELECT
            Profit
        FROM TodatProfit)
    ) AS TodayProfit
END

--Stock Monthly Product
IF @RunType = 9
BEGIN
    SELECT DISTINCT TOP (3)
        ProductDiscription AS MonthlyProduct,
        COUNT(ProductDiscription) AS count
    FROM Salesview
    WHERE CONVERT(DATE, InvoiceDate) BETWEEN CONVERT(DATE, DATEADD(MONTH, -1, GETDATE()))
    AND CONVERT(DATE, GETDATE())
        GROUP BY ProductDiscription
        ORDER BY count DESC
END

```

```

--Stock Weekly Product
IF @RunType = 10
BEGIN
    SELECT DISTINCT TOP (3)
        ProductDescription AS WeeklyProduct,
        COUNT(ProductDescription) AS count
    FROM Salesview
    WHERE CONVERT(DATE, InvoiceDate) BETWEEN CONVERT(DATE, DATEADD(WEEK, -1, GETDATE()))
AND CONVERT(DATE, GETDATE())
        GROUP BY ProductDescription
        ORDER BY count DESC
END

--Stock Daily Product
IF @RunType = 11
BEGIN
    SELECT DISTINCT TOP (3)
        ProductDescription AS DailyProduct,
        COUNT(ProductDescription) AS count
    FROM Salesview
    WHERE CONVERT(DATE, InvoiceDate) = CONVERT(DATE, GETDATE())
        GROUP BY ProductDescription
        ORDER BY count DESC
END

--Category Rate
IF @RunType = 12
BEGIN
    SELECT TOP (2)
        CONCAT(Category, ' ', [Range], ' %') AS Category
    FROM RateCategory
    ORDER BY NEWID()
END

-- Supplier Rate
IF @RunType = 13
BEGIN
    SELECT TOP (2)
        CONCAT(SupplierName, ' ', [Range], ' %') AS Supplier
    FROM RateSupplier
    ORDER BY NEWID()
END

```

Appendix B: 8

```
private void SetUserPrivileges()
{
    mdUp.UserLevel = iGlobal.UserLevel;
    DataTable dt = dIUs.SelectUserPrivileges(mdUp);

    PointofSaleTile.Visible = false;
    PurchaseOrderTile.Visible = false;
    GoodsReceivingNoteTile.Visible = false;
    ItemMasterTile.Visible = false;
    StockTransferTile.Visible = false;
    ProductControlsTile.Visible = false;
    PaymentScheduleTile.Visible = false;
    DashboardTile.Visible = false;
    StingsTile.Visible = false;
    StockReportTile.Visible = false;
    ReOrdersReportTile.Visible = false;
    ExpiryReportTile.Visible = false;
    SalesReportTile.Visible = false;

    foreach (DataRow dtr in dt.Rows)
    {
        switch (dtr["Form"].ToString())
        {
            case "PointofSale":
                PointofSaleTile.Visible = true;
                break;
            case "PurchaseOrder":
                PurchaseOrderTile.Visible = true;
                break;
            case "GoodsReceivingNote":
                GoodsReceivingNoteTile.Visible = true;
                break;
            case "ItemMaster":
                ItemMasterTile.Visible = true;
                break;
            case "StockTransfer":
                StockTransferTile.Visible = true;
                break;
            case "ProductControls":
                ProductControlsTile.Visible = true;
                break;
            case "PaymentSchedule":
                PaymentScheduleTile.Visible = true;
                break;
        }
    }
}
```

```
        break;
    case "Dashboard":
        DashboardTile.Visible = true;
        break;
    case "Stings":
        StingsTile.Visible = true;
        break;
    case "StockReport":
        StockReportTile.Visible = true;
        break;
    case "ReOrdersReport":
        ReOrdersReportTile.Visible = true;
        break;
    case "ExpiryReport":
        ExpiryReportTile.Visible = true;
        break;
    case "SalesReport":
        SalesReportTile.Visible = true;
        break;
    }
}
}
```

Appendix B: 9

```
<?xml version="1.0" encoding="utf-8"?>
<Dashboard CurrencyCulture="en-US">
    <Title Text="Dashboard" />
    <DataConnections>
        <DataConnection Name="HBConnection" ProviderKey="MSSqlServer">
            <Parameters>
                <Parameter Name="server" Value=".\\SQLEXPRESS" />
                <Parameter Name="database" Value="HummingBird" />
                <Parameter Name="useIntegratedSecurity" Value="True" />
                <Parameter Name="read only" Value="1" />
                <Parameter Name="generateConnectionHelper" Value="false" />
            </Parameters>
        </DataConnection>
    </DataConnections>
    <DataSources>
        <DataSource ComponentName="HBConnection" Name="HBConnection">
            <DataProvider DataConnection="HBConnection" SupportSql="true">
                <Selection>
                    <Table Name="Salesview">
                        <Columns>
                            <Column Name="InvoiceDate" Alias="InvoiceDate" />
                            <Column Name="InvoiceNumber" Alias="InvoiceNumber" />
                            <Column Name="ProductDiscription" Alias="ProductDiscription" />
                            <Column Name="Batch" Alias="Batch" />
                            <Column Name="BuyingPrice" Alias="BuyingPrice" />
                            <Column Name="RetailPrice" Alias="RetailPrice" />
                            <Column Name="Quantity" Alias="Quantity" />
                            <Column Name="ItemDiscount" Alias="ItemDiscount" />
                            <Column Name="InvoiceDiscount" Alias="InvoiceDiscount" />
                            <Column Name="Total" Alias="Total" />
                            <Column Name="TotalSum" Alias="TotalSum" />
                            <Column Name="InvoiceProfit" Alias="InvoiceProfit" />
                            <Column Name="ItemProfit" Alias="ItemProfit" />
                        </Columns>
                    </Table>
                </Selection>
            </DataProvider>
        </DataSource>
    </DataSources>
    <Items>
```

```

<RangeFilter ComponentName="rangeFilterDashboardItem1" Name="Range Filter 1"
DataSource="HBConnection">
  <DataItems>
    <Measure DataMember="InvoiceNumber" SummaryType="Count"
UniqueName="DataItem0" />
    <Dimension DataMember="InvoiceDate" DateTimeGroupInterval="DayMonthYear"
UniqueName="DataItem1" />
    <Measure DataMember="InvoiceProfit" UniqueName="DataItem2" />
  </DataItems>
  <Argument UniqueName="DataItem1" />
  <Series>
    <Simple SeriesType="Area">
      <Value UniqueName="DataItem0" />
    </Simple>
    <Simple SeriesType="StackedArea">
      <Value UniqueName="DataItem2" />
    </Simple>
  </Series>
</RangeFilter>
<Grid ComponentName="gridDashboardItem1" Name="Grid 1" ShowCaption="false"
DataSource="HBConnection">
  <DataItems>
    <Measure DataMember="InvoiceNumber" SummaryType="Count"
UniqueName="DataItem0" Name="Invoices" />
    <Dimension DataMember="InvoiceDate" SortOrder="Descending"
DateTimeGroupInterval="DayMonthYear" UniqueName="DataItem2" Name="Invoice Date" />
    <Measure DataMember="InvoiceProfit" UniqueName="DataItem3" Name="Invoice
Profit">
      <NumericFormat FormatType="Currency" Unit="Ones" />
    </Measure>
    <Measure DataMember="Total" UniqueName="DataItem4">
      <NumericFormat FormatType="Currency" Unit="Ones" />
    </Measure>
  </DataItems>
  <GridColumns>
    <GridDimensionColumn>
      <Dimension UniqueName="DataItem2" />
    </GridDimensionColumn>
    <GridMeasureColumn DisplayMode="Bar" AlwaysShowZeroLevel="true">
      <Measure UniqueName="DataItem0" />
    </GridMeasureColumn>
    <GridMeasureColumn>
      <Measure UniqueName="DataItem4" />
    </GridMeasureColumn>
  </GridColumns>
</Grid>

```

```

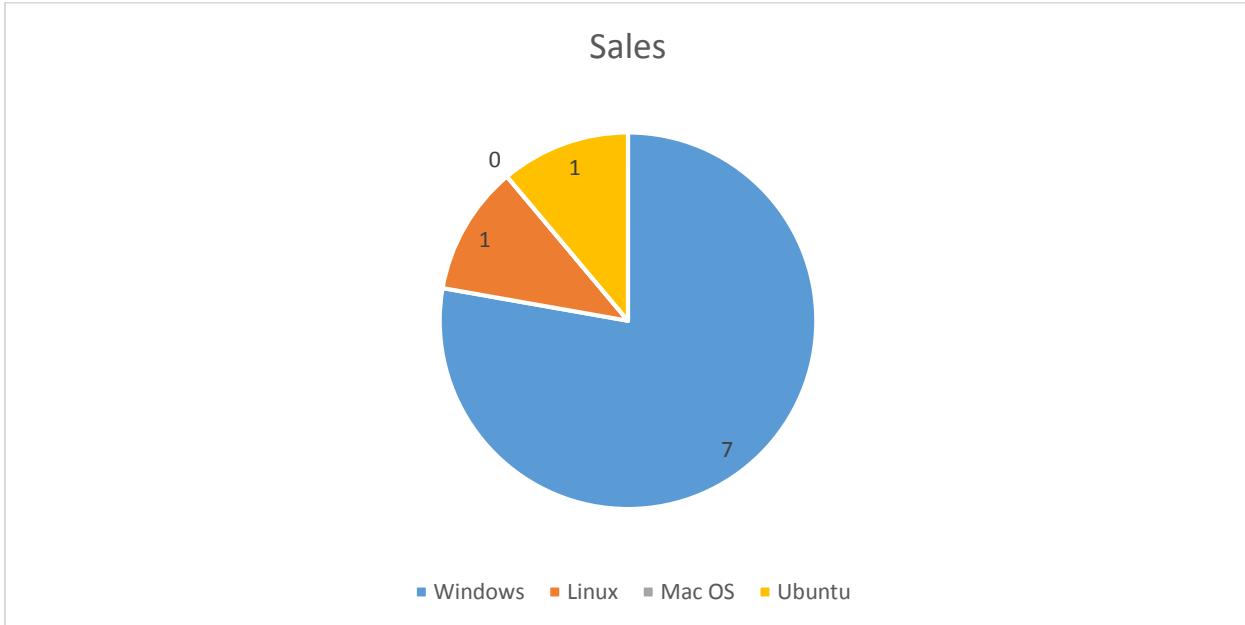
<GridMeasureColumn>
  <Measure UniqueName="DataItem3" />
</GridMeasureColumn>
</GridColumns>
<GridOptions />
</Grid>
<Chart ComponentName="chartDashboardItem1" Name="Chart 1" ShowCaption="false"
DataSource="HBConnection">
  <DataItems>
    <Dimension DataMember="InvoiceDate" DateTimeGroupInterval="Month"
UniqueName="DataItem0" />
    <Measure DataMember="InvoiceProfit" UniqueName="DataItem1" Name="Profit
(Monthly)">
      <NumericFormat FormatType="Currency" Unit="Ones" />
    </Measure>
  </DataItems>
  <Arguments>
    <Argument UniqueName="DataItem0" />
  </Arguments>
  <Panes>
    <Pane Name="Pane 1">
      <Series>
        <Simple>
          <Value UniqueName="DataItem1" />
        </Simple>
      </Series>
    </Pane>
  </Panes>
</Chart>
<Chart ComponentName="chartDashboardItem2" Name="Chart 1" ShowCaption="false"
DataSource="HBConnection">
  <DataItems>
    <Dimension DataMember="InvoiceDate" UniqueName="DataItem0" />
    <Measure DataMember="InvoiceProfit" UniqueName="DataItem1" Name="Profit
(Yearly)">
      <NumericFormat FormatType="Currency" Unit="Ones" />
    </Measure>
  </DataItems>
  <Arguments>
    <Argument UniqueName="DataItem0" />
  </Arguments>
  <Panes>
    <Pane Name="Pane 1">
      <Series>

```

```
<Simple>
  <Value UniqueName="DataItem1" />
</Simple>
</Series>
</Pane>
</Panes>
</Chart>
</Items>
<LayoutTree>
  <LayoutGroup Orientation="Vertical">
    <LayoutItem Weight="28.571428571428573" DashboardItem="rangeFilterDashboardItem1"
/>
    <LayoutGroup Weight="71.428571428571431">
      <LayoutItem Weight="49.9457111834962" DashboardItem="gridDashboardItem1" />
      <LayoutGroup Weight="50.0542888165038" Orientation="Vertical">
        <LayoutItem Weight="49.866666666666667" DashboardItem="chartDashboardItem1" />
        <LayoutItem Weight="50.13333333333333" DashboardItem="chartDashboardItem2" />
      </LayoutGroup>
    </LayoutGroup>
  </LayoutGroup>
</LayoutTree>
</Dashboard>
```

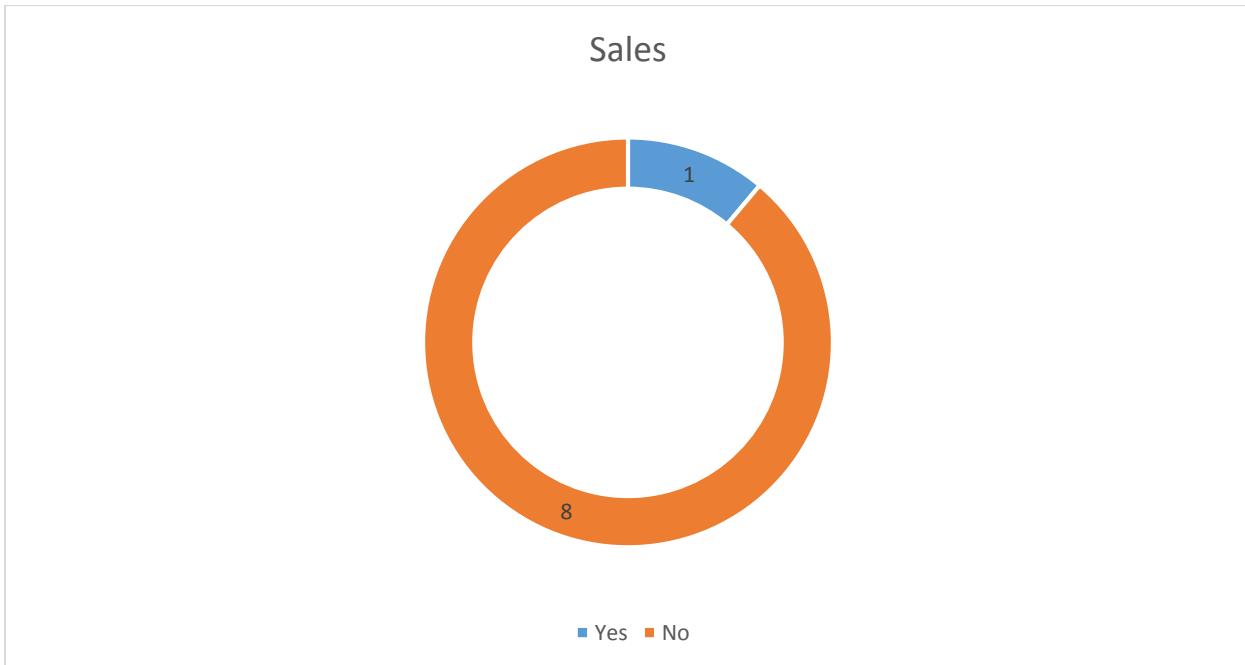
Appendix C - Testing

Question 6 - What type of Operating systems you have been work with?



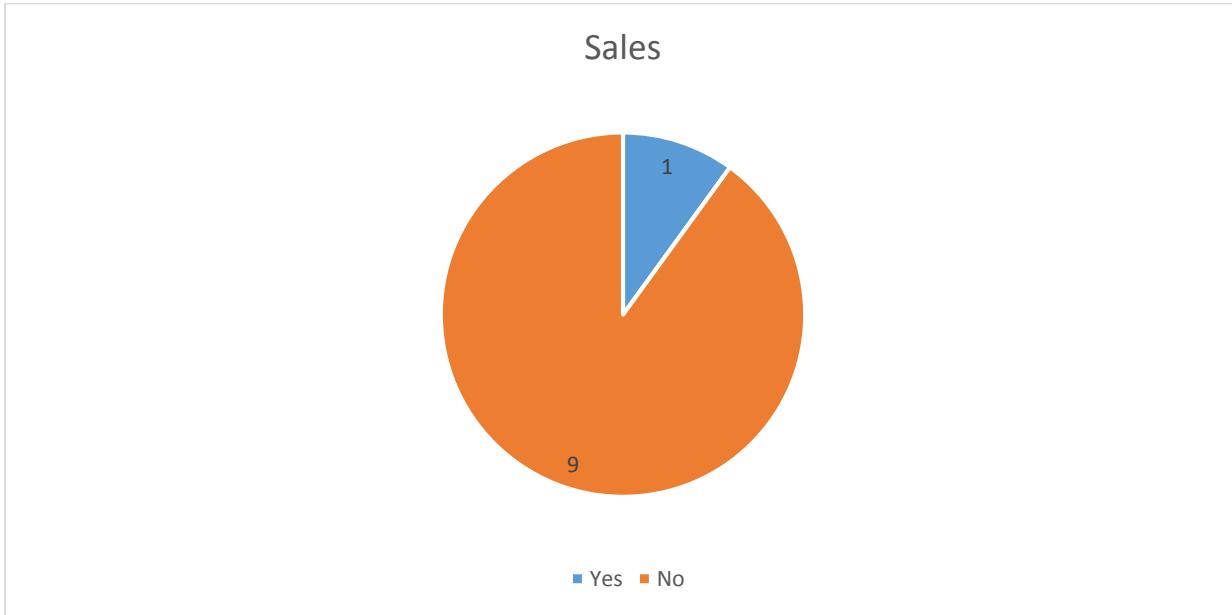
By asking this question, checking the operating system who failure with, by the result six of them are failure with windows based operation system as expected and as required.

Question 7 - Have you ever work with Windows 8 with Windows Apps



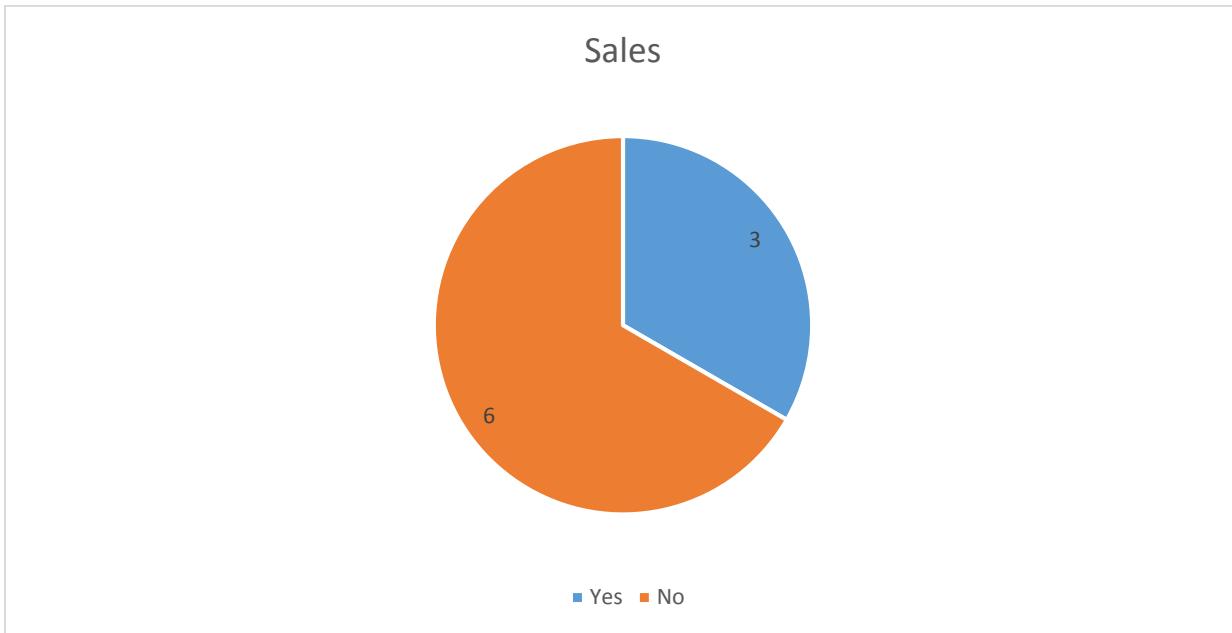
By asking the question, taking as idea about who can manage the new features in windows 8 app and, who can work with proposed system and widows app. Form the result, only one of employee can work with windows app and others need to complete some tanning.

Question 8 - Have you ever used a touch computer system?



By asking that question we can take an idea about who used touch computer, by the answer, only one person sued a touch computer and others never used touch computer.

Question 9 - Have you ever work with Point of Sales (POS) system?



By the question checking the usability of Point of Sales (POS) system, from the answer, three of employees aware to use the POS system. In addition, six of them not aware to use the POS system. As the result, need to give some special training about the POS system.

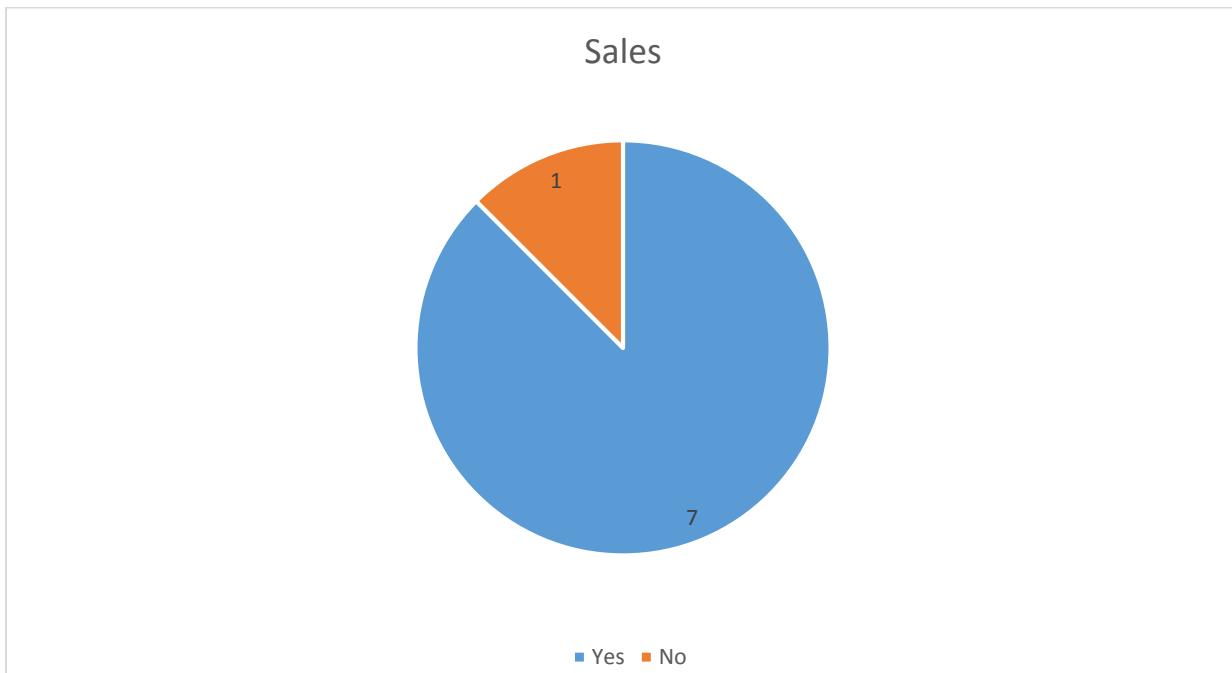
2.2.2.3.2 Questionnaire 2 - Windows Application Design

The Questionnaire is provided to employees who specially chosen from the first IT knowledge questionnaire. From 10 employees, seven employees war chosen to give the questionnaire. The questionnaire and analyzing report is mainly focus in to Windows based application.

The analyzing report focused on

- To analyses the Designing technics
- Implement the interfaces
- Report generating methods
- Grid views and functions that proposed to use with application
- Data input dialogs design

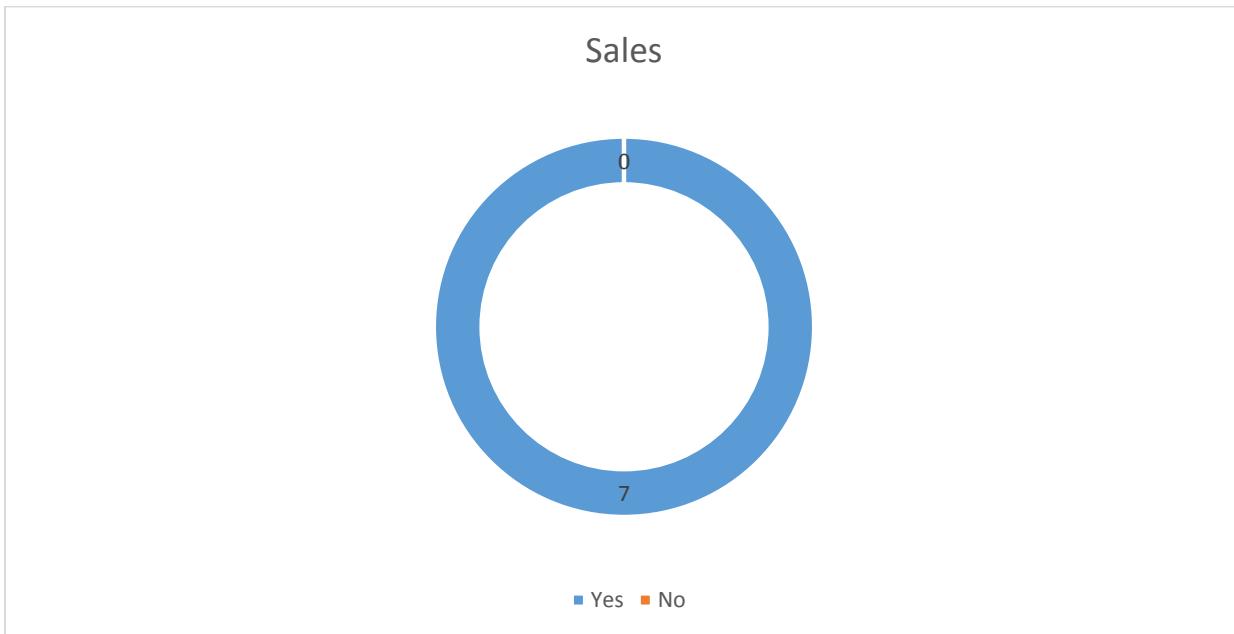
Question 1 - Have you familiar with windows based operating system?



From chose seven employee, most of employees are aware with windows based operating system. In addition, one employee is not familiar with windows based operating system. With the answers, then the platform is directly focus to the windows based application.

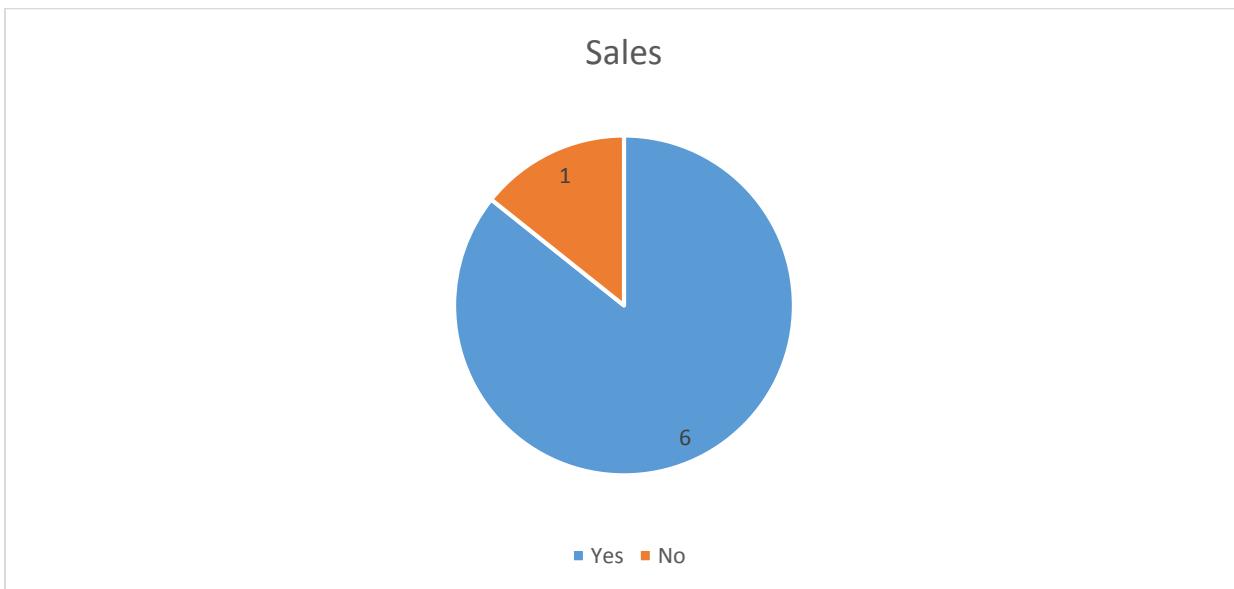
As the result, all are familiar with windows based application interfaces.

Question 2 - Do you believe you can work with Microsoft Windows platform?



By the question, checking the ability to use windows based application for the employee, as the answer all employee are believe they can work with Microsoft Windows platform.

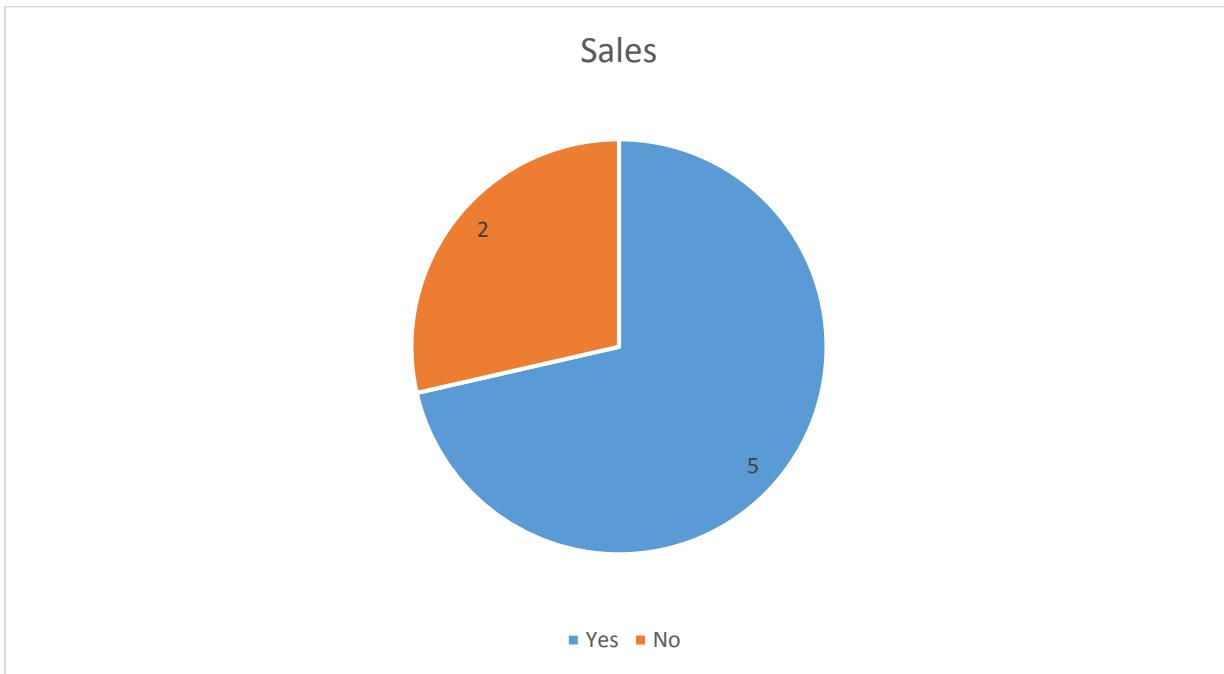
Question 3 - Have you worked with any Windows based application?



By the question, six of employees were worked with windows application; one of employee is not familiar with windows based application.

As the second question the all employee think they can work with windows based application.

Question 4 - Can you install an application in windows?



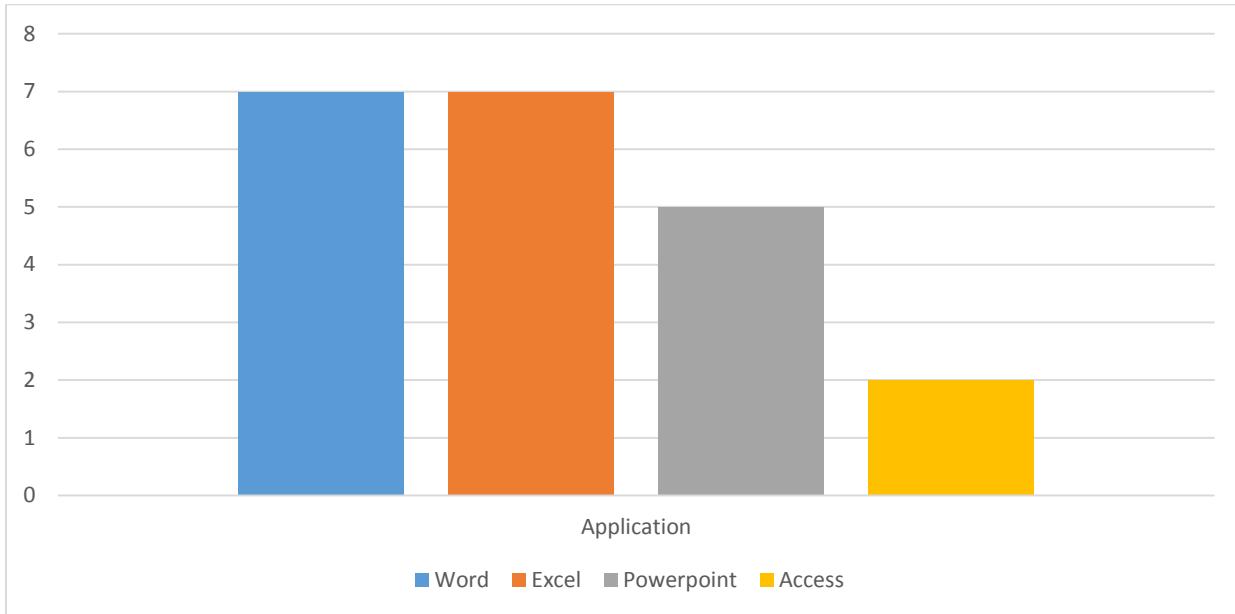
As the question, checking the ability to install windows application in the windows operating system. As the result, five of employees are able to install application in windows and, two of employees are not think they can install the application on windows.

Question 5 - Have you any experience with Data Entering?



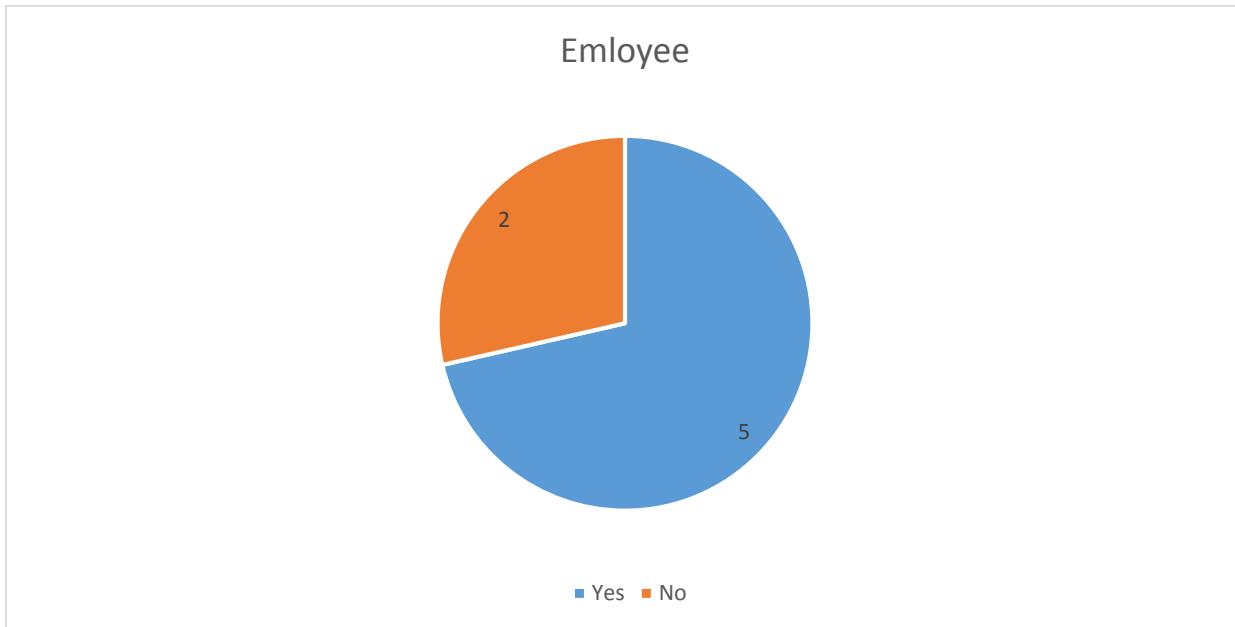
As the result, all employees are familiar with data entering, likewise word or some data entering method.

Question 6 – If you have you worked with Microsoft or other Office package select bellow



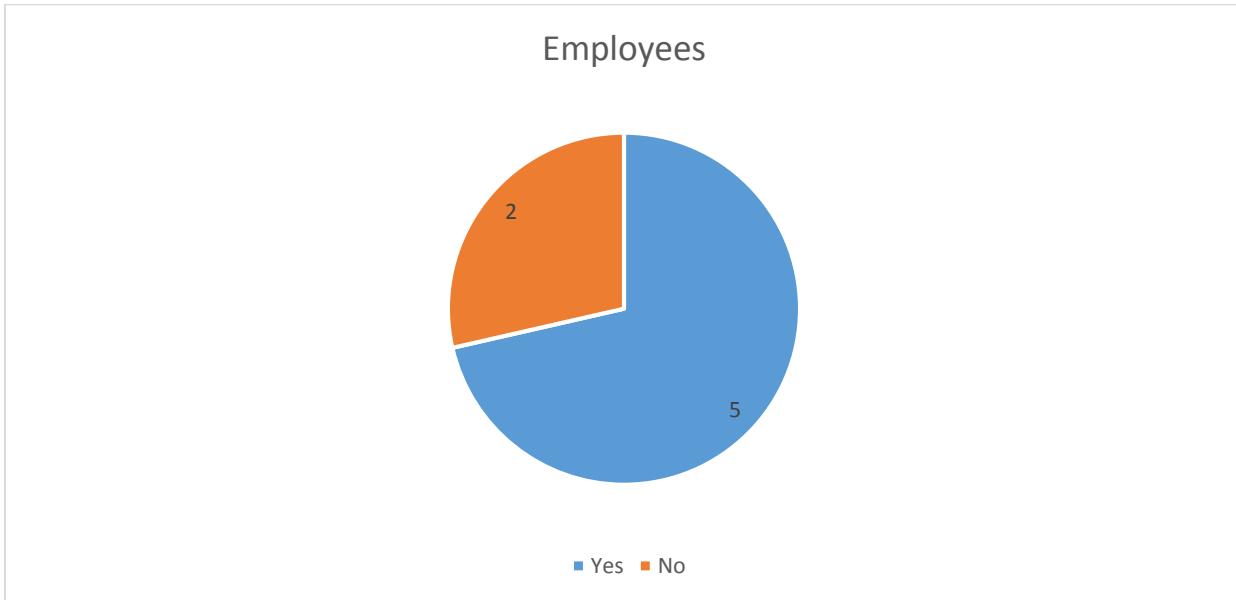
As the question, word and Excel is used with all selected employee, but access and PowerPoint is taking less than word and excel. As the result decided to use more word, excel based interfaces and data interface styles, with functions.

Question 7 - Have you documented any Excel sheet?



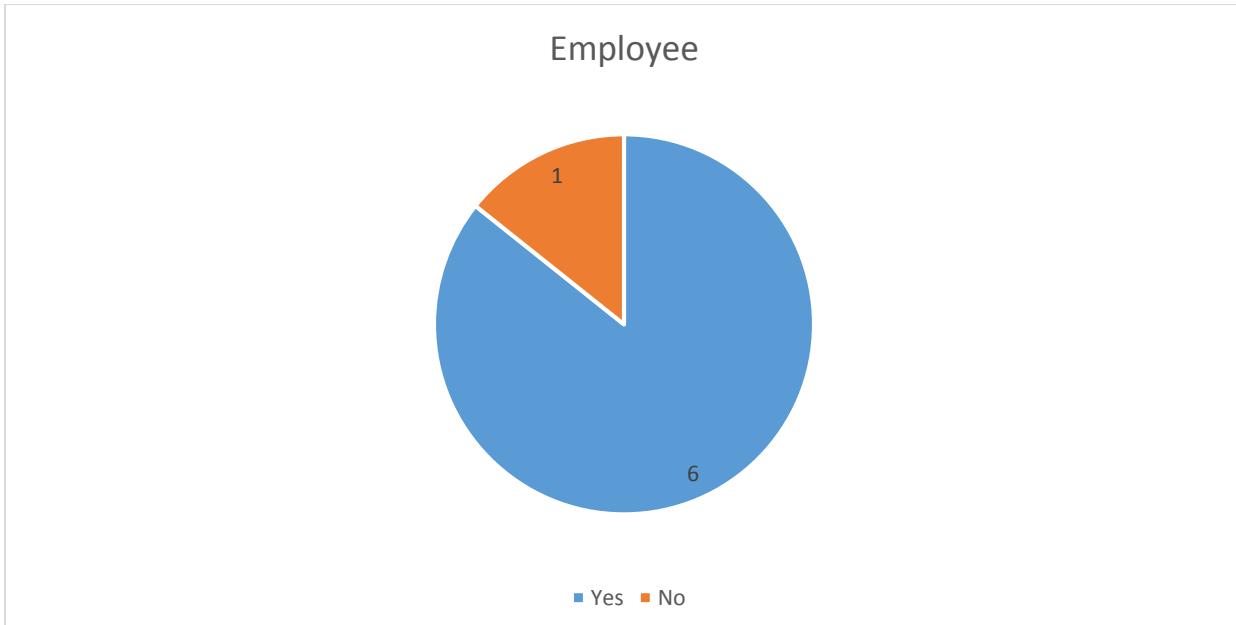
As the question, and result, most of them are aware with Microsoft Excel. The question is based with application data grid design.

Question 8 - Are you familiar with Data sorting and filtering in Excel?



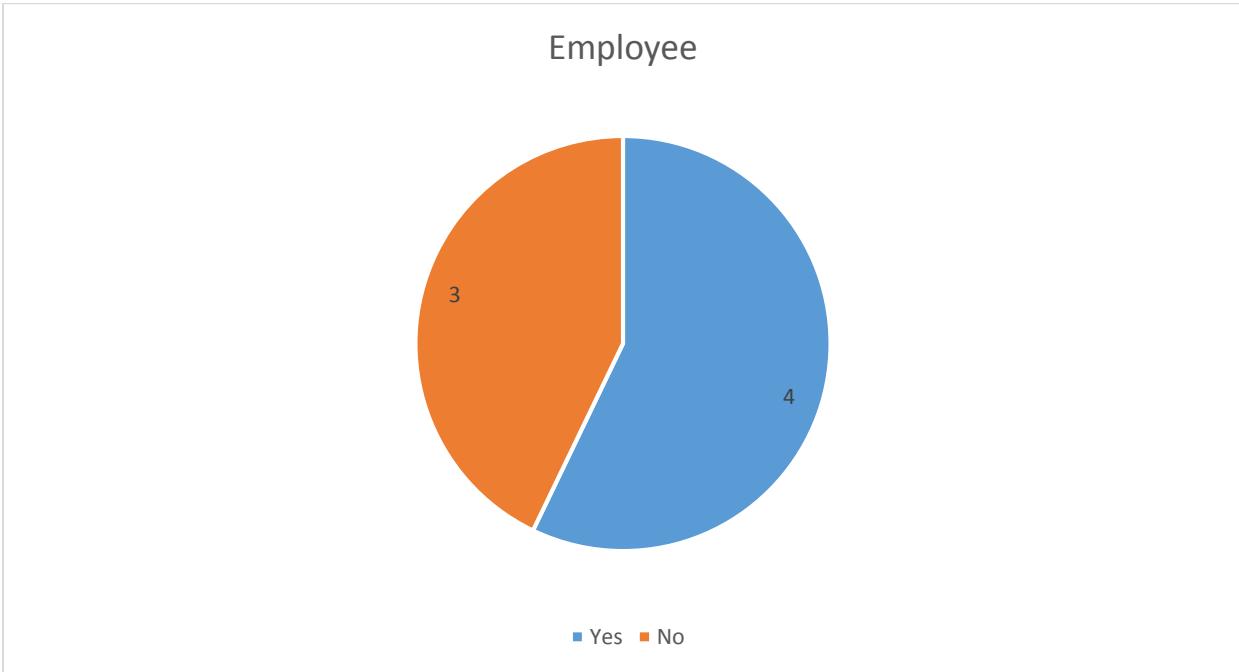
By the question checking whether user can or cannot process the output file formats from the report in windows application, as the result currently 5 employees able to the operation.

Question 9 - Are you familiar with printing documents?



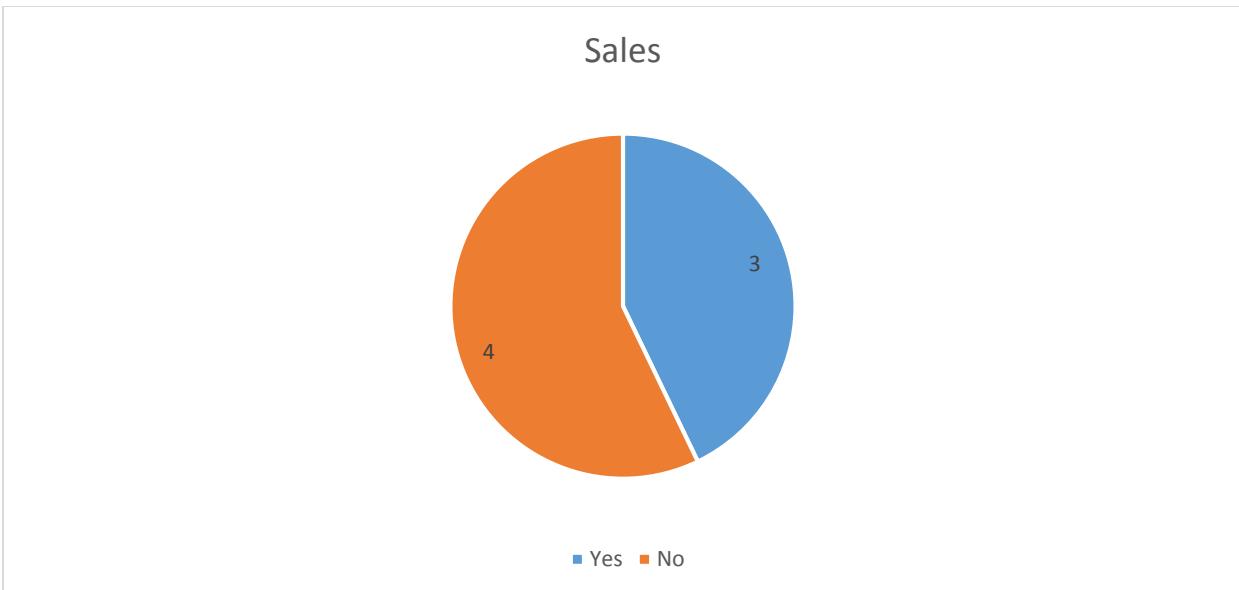
Taking idea about printing skills of the employee. Reports need to be print, as they wanted. Then the requirement is need to be proceed. As the result, six of employee can print a document.

Question 10 - Can you make calculations via Excel?



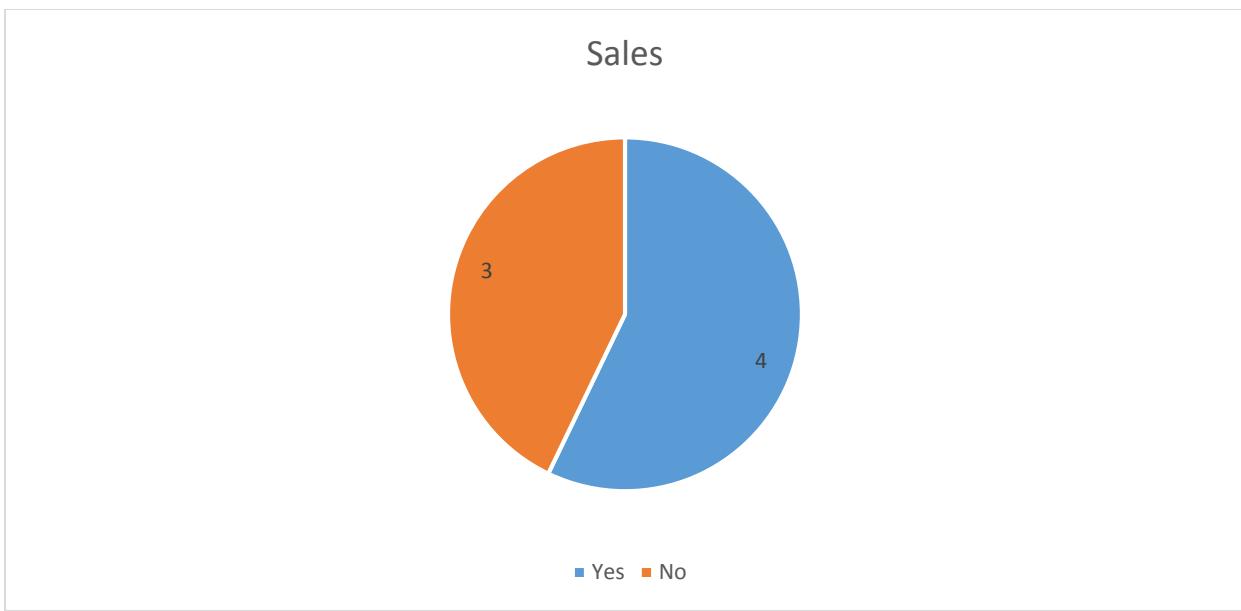
As the question, checking the skill of calculation in Excel. As the result four of employee can make calculations in excel and three of them cannot do the calculation.

Question 11 – Have you worked with stock reports?



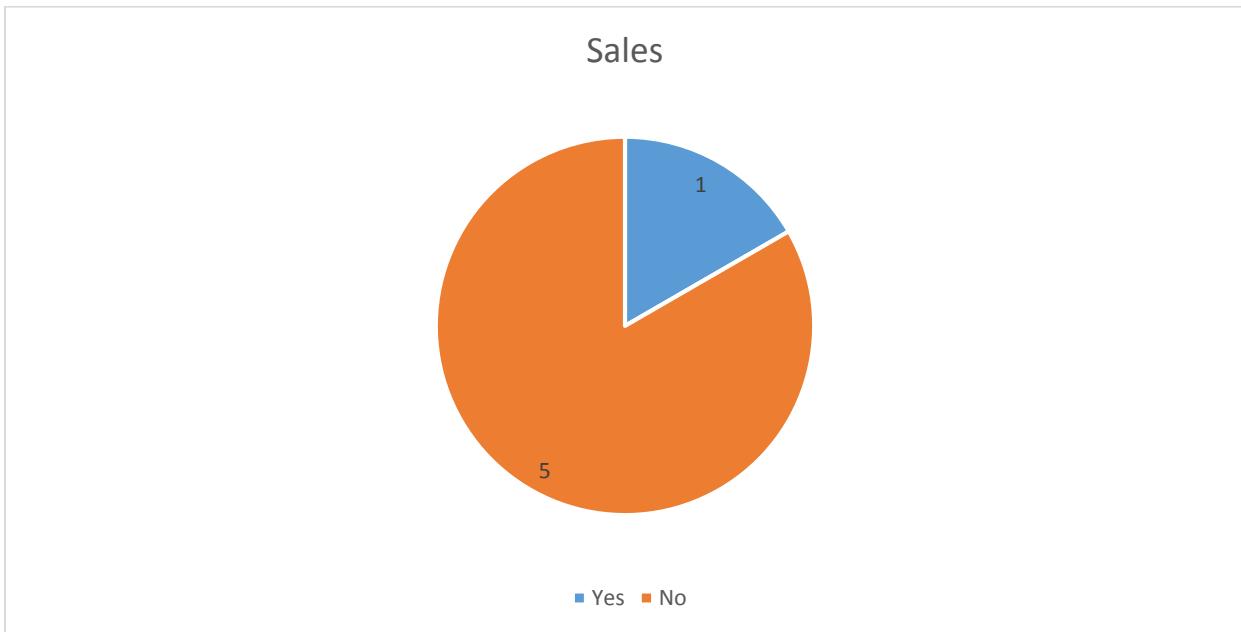
As the question checking the skill of working reports, by the result, three of employees are able to work with report and another four employee is not worked with reports.

Question 12 - Do you think, you can make a report from given details?



As the question, checking whether employee can make a report as given some details, as the report three of them cannot make reports as expected and four of employee can make reports.

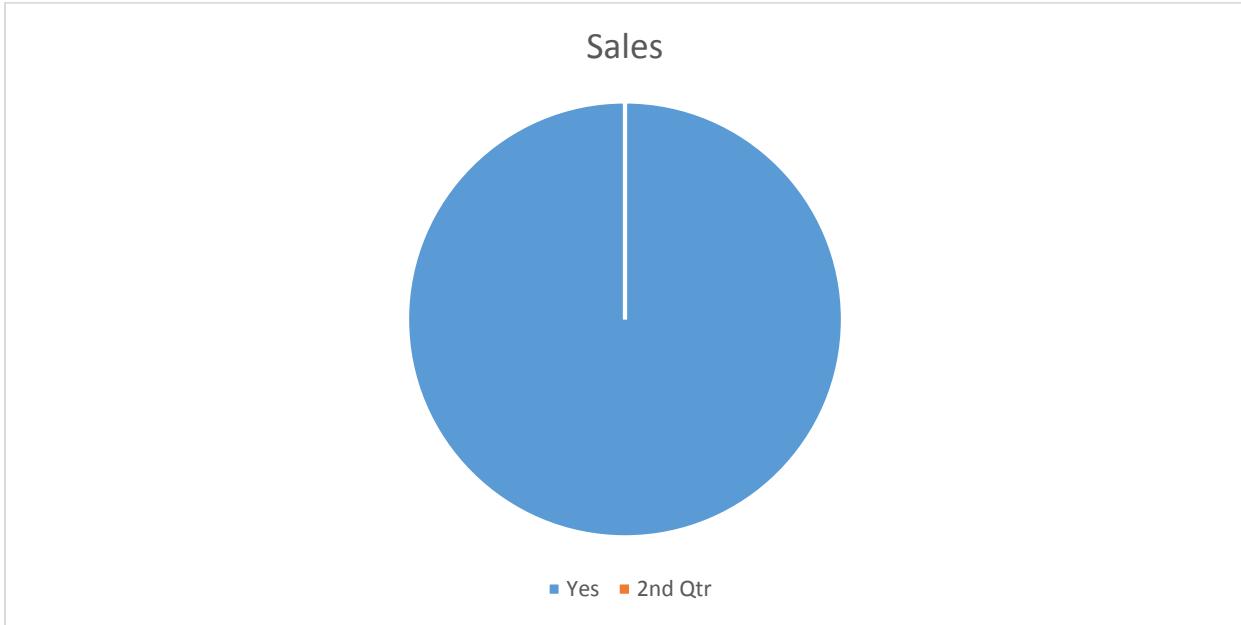
Question 13 - Are aware with any Databases system



As the question checking whether user employee can handle the databases, as the result in the pharmacy only one employee can handle a database.

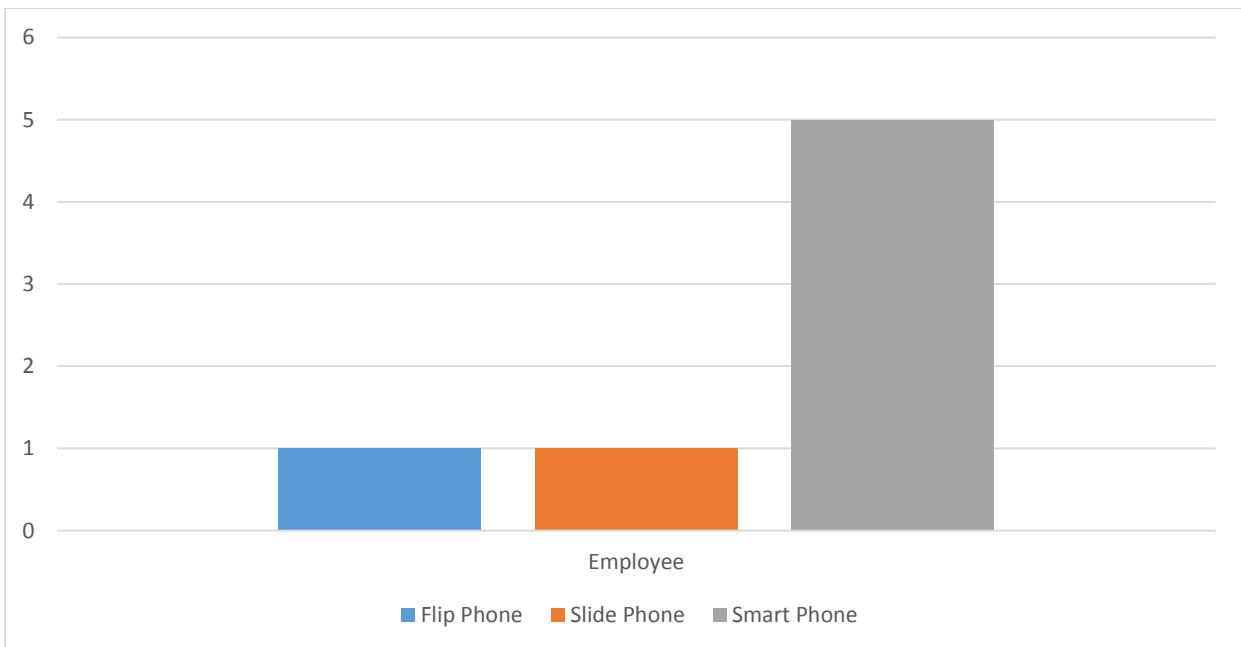
2.2.2.3.3 Questionnaire 2 - Mobile Application Design

Question 1 - Have you a smart mobile device?



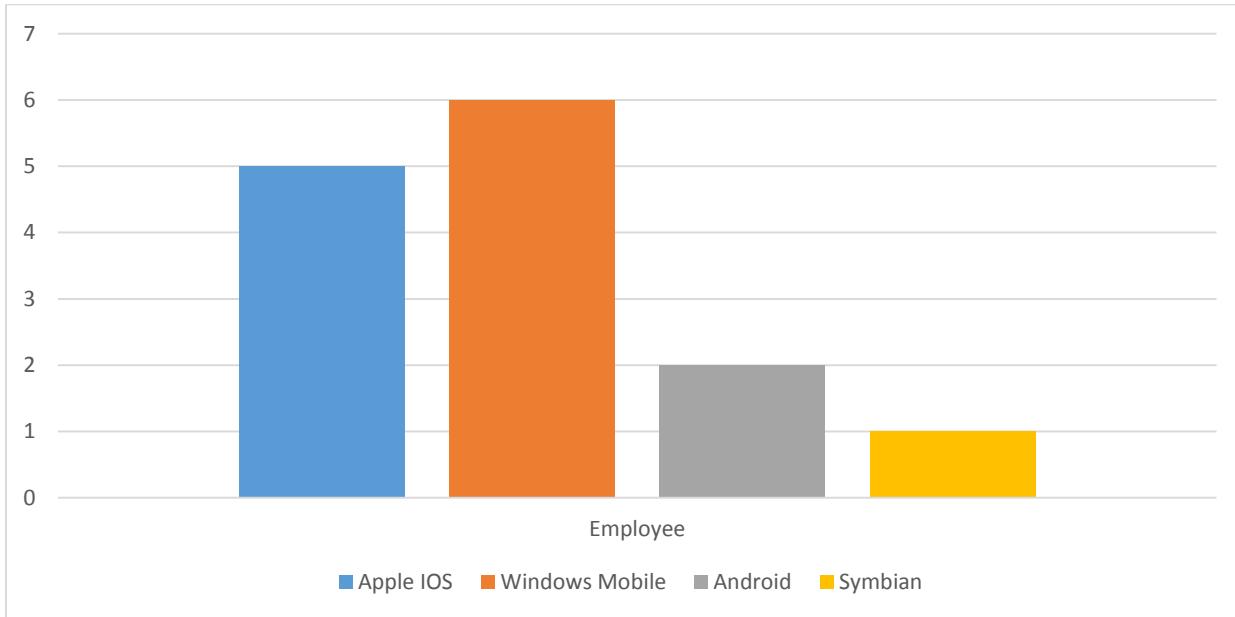
By this question checking whether employee can have a mobile device with the experience. As the result all employee have a mobile device,

Question 2 - What type of mobile device would you like to have?



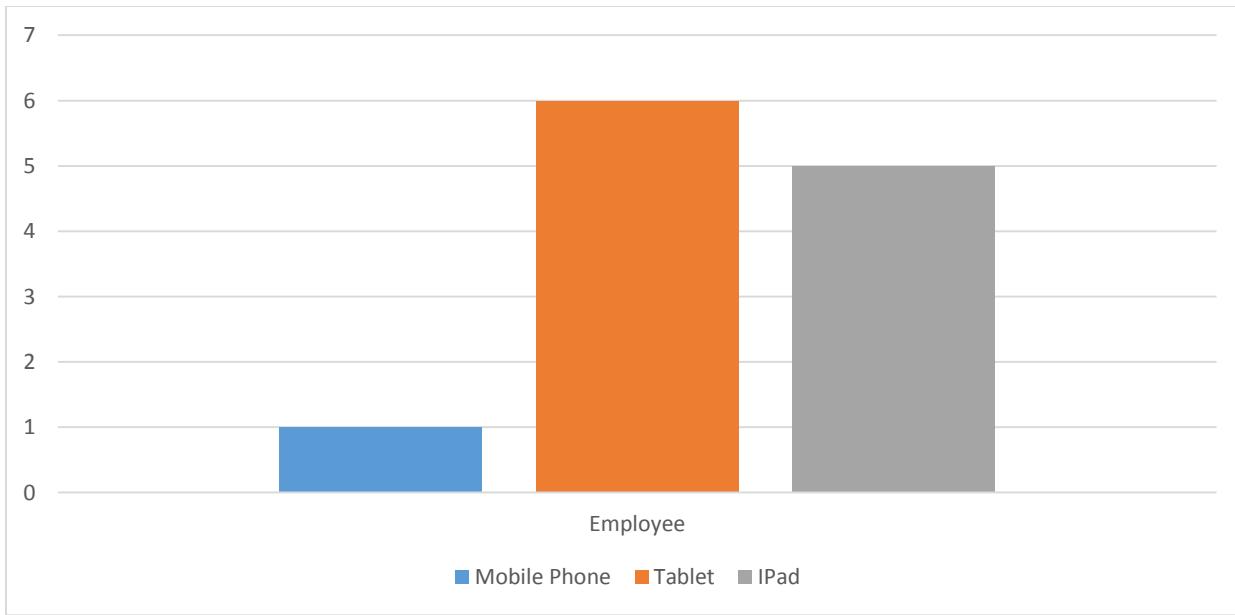
This question directly focusing to the targeted mobile device, as the result from seven employee, five employees are using Smart mobile devices.

Question 3 - What type of Mobile Operating system device you like to use?



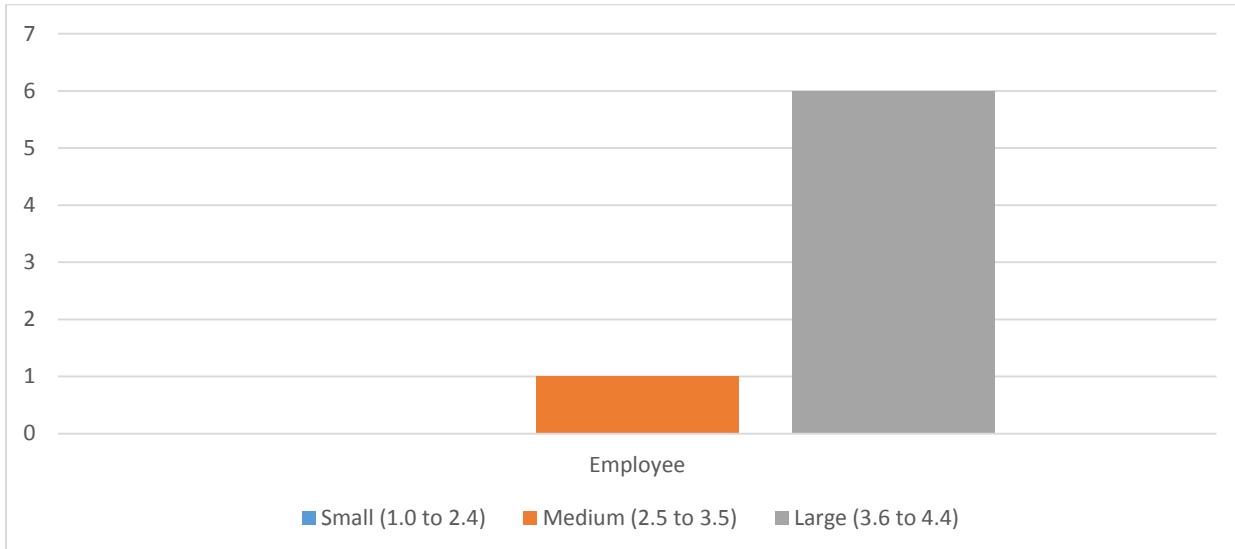
By this question, taking the information to select the system platform to develop the mobile application. As the result, the most used mobile platform was Windows mobile.

Question 4 - What type of Device you like to use?



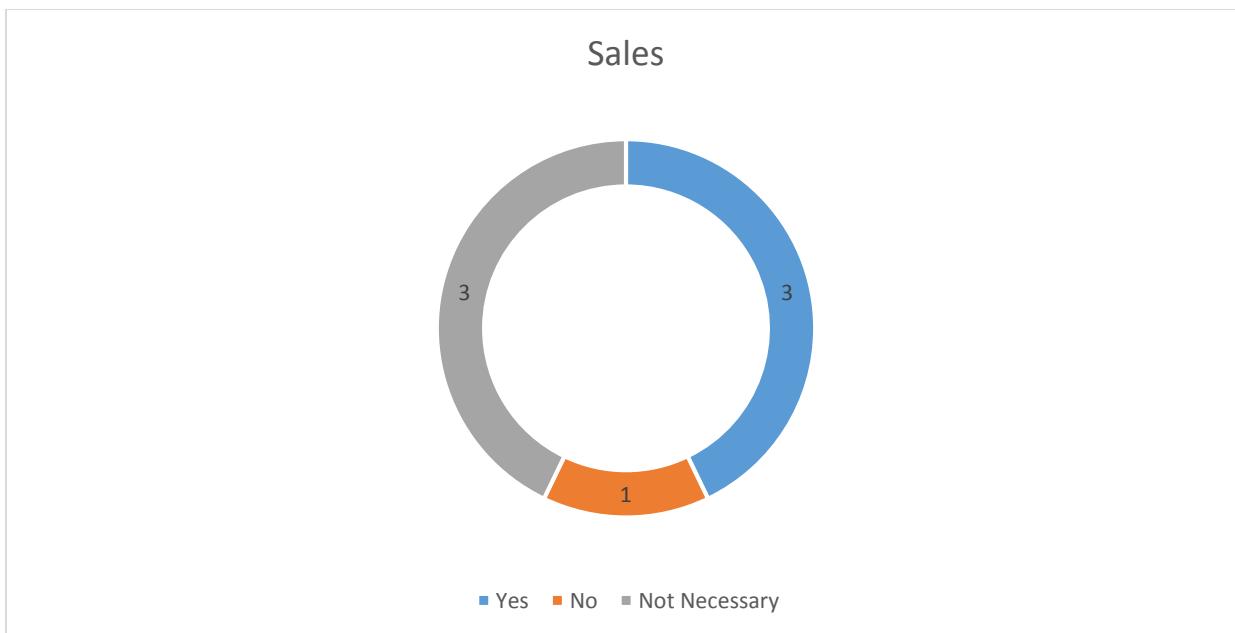
By this question, taking the idea to supply the application with proper device. As the result, the most chosen device is Tablet.

Question 5 - What size would you prefer your screen to be?



By the question taking an idea about the screen size to develop the mobile application. As the result, the popular screen size is large size.

Question 6 - Are interesting to include sounds with mobile application?



By the question checking user attraction with the sounds, in the mobile application, as the report, the Answer No and Answer Yes employees are same and Answer Not necessary employees are less.

2.3.3.2 Analysis of gathered data

Interviews Question 1 – Analyze the current process Collect Inventory Data

1. Taking Information about items in the inventory

1.1 Question - Approximately, how many inventory items do you have?

Answer – more than five thousand items.

By the answer, they are maintaining big amount of inventory data as same, and they have to have lot of records with the inventory, with implementing the application, developer have to maintain the proper database with indexing techniques and proper relations.

1.2 Question - Describe how you categorize items

Answer – currently they are maintaining the physical labeling system inside the stock. In addition, they are maintaining the simple records according to the items; as usual, they are manually categorizing the items by experience, without proper technique.

Analyzed Outcome – By the answer, developer needs to implement the proper categorization method.

1.3 Question - What basic information do you record about items?

Unit of measure

Normally they are using Kilogram, Milligram, ML, L and as bottles, packets etc.

Category

Currently they are maintaining physical Category with labels inside the stock, and most of categories are deciding by experience.

Usage about Serial number

They are not maintaining the Serial numbers.

Receiving location and Dealers, Suppliers

They are maintaining the document to manage Dealers, Suppliers, mainly the document is containing about the Phone number only.

Shipping and order entry information

Most of orders are not properly managed; they are taking temporary list and maintaining a document.

Purchasing information

They are maintaining document to manage the purchasing information.

Analyzed Outcome –

1.4 Question - Is there anything you track about items that consider out of the ordinary?

Answer – they are tracking items with Quantity and expiries, but most of information documented and find the item in document is hard.

Analyzed Outcome – As the discussion, need to implement proper report to generate the expiries and re orders with quantities.

Question - Describe how you record and track item revisions (change control)

Answer – they are maintaining a document to each months, and they are placing a record while introducing the items to the stock, about Quantities and specially expiries, after that they are checking the records in period.

Analyzed Outcome –

1.5 Question - Describe how you record cost information on items:

Answer – they are maintaining the separate documents, when they need to calculate some information according to inventory, they have to rearrange stock and read current stock.

They are maintaining the document to

- Unburdened standard cost
- Average cost
- Material burden rate
- Burdened standard cost
- Estimated cost
- Alternate costs
- Pending costs

Analyzed Outcome –

1.6 Question - How do you specify the cost of sales account for inventory transactions involving sales orders?

Answer – they are normally maintaining a printed data sheet, and they are updating the data sheet day by day and calculating cost and needed information.

Analyzed Outcome –

1.7 Question - Do you use cross-references with items?

Currently they are not maintaining any cross references with items. However, they are maintaining the batch number to each items. The batch number is created with category and last record number of the stock data sheet.

Analyzed Outcome –

2. Transactions

2.1 Question - What types of inventory transactions do you perform?

Receive material from a vendor

They are reserving material from vendor, and they are maintaining separate data sheet to each vendor with transactions.

Return material to a vendor

Return material transactions and logs are maintaining with document to each vendor.

Receive customer returns (return material authorizations)

When reserving the returns, they are maintain the return log and balancing the returns with transaction.

Transfer material between warehouses

They are not maintaining separate warehouses, they are using only one centralize warehouses and all the items are stored in the single warehouses.

Transfer material between inventory subdivisions

They are maintaining sub divisions in the inventory to separate the categories and divide the reacting items each other to separate divisions.

View on-hand and available-to-transact quantities for a given item during a transaction

They are holding the items using a red removable label and separating in same item group at same place.

Analyzed Outcome –

2.2 Question - How do you check availability of any content at any future point?

Answer – when they need to find the current stock and products availability without physical calculation, they are checking the reports created with the inventory details.

Analyzed Outcome –

3 .Replacement planning

3.1 Question - Explain what item planning strategy you using with the inventory?

As the answers, they are maintaining two types of main planning methods

Inventory Planning methods.

1. Master production schedule (MPS) planning
2. Material requirements planning (MRP)

Content minimum- maximum planning (minimum quantity to reorder content- maximum content to manage the inventory)

Reorder level planning, is running with the inventory documentation and data sheets.

Analyzed Outcome –

3.2 Question - Do you use safety on stock calculating planning?

Answer – they are using the stock calculation plan in each every month and they are updating the inventory data sheets with expiries and re orders.

3.3 Question - How do you track time that consume for manufacturing goods?

Answer – in the inventory log, they have the reserve date and expiree time of the goods, and they are maintaining the monthly expiree report, that they update monthly, with expiree date end, change the stock

Analyzed Outcome –

3.4 Question - Do you use a system like time barrier, to stop bring order before that they can produce (manufacture)?

Answer - they are maintaining the time barrier but the time cap is mainly depend with the requirement and expiries and re order level.

Analyzed Outcome –

3.5 Question - Do you track costs of order and cost for the transport?

Answer – normal they are maintaining the cost for orders and transport.

Analyzed Outcome – Transport handling option need to be include to the include with the Order management.

4. Forecasting

4.1 Question - Explain how you analyze future demand for items?

Answer - they are maintaining sales report manually, after reviewing the report deciding the fast moving items and demanded items of the period.

Analyzed Outcome – generate easy access report for fast moving items.

How do you estimate future demand history based on past transactions?

Answer – when they found some fast moving item around some area or specific period, they are preparing the extra stock and releasing when needed.

Analyzed Outcome – need to manage fast moving items, as an auto-generated report.

5. Accuracy

5.1 Question - Do you use physical inventory tags?

Answer – they are using physical inventory tags to each items and each categories.

Their dividing the items by using tags to separate, expiries, damages and returns.

Analyzed Outcome - need to implement the tag (Batch Number) auto generate option with the application.

5.2 Question - Do you use sequential inventory item count?

How do decide when should count?

Answer – they are using manual count for the inventory, and they are reporting the process in every month. After the item count, they are collecting the data and updating the inventory data sheets in manual way.

Analyzed Outcome – need to auto generate data report based on inventory to return the inventory item count and needed data.

5.3 Question - Do you assign activity based costing (ABC) classes to inventory items?

If, how do you rank items (e.g., based on current on-hand rate, transaction history - usage or value)?

Answer –

Analyzed Outcome –

5.4 Question

Explain how modifications on quantity and values are made as the result of sequent count? Do you use counting system to decide when to make a count and modification?

Answer –

Analyzed Outcome –

6. Warehouse

6.1 Question

Explain the structure of your organization, number of warehouses, accepting docks, distribution centers, and item manufacturing industry

Answer –

Analyzed Outcome –

7. Inquiries

7.1 Question - What type of online inquiries method does u prefer?

Answer – there maintaining Transaction inquiries to view transaction history made on item inventory, by item type, location, usability and Item inquiries to calculate the on-hand, planning, status, and sequence and Control inquiries to inventory valuation and cost info On hand inquiries to view item quantities in sub inventories, locations, in-transit and receiving

Analyzed Outcome – As the answer need to implement the tracking data and history of data and transactions.

8. Reports

8.1 Question - What type of report generate do essential?

Answer – they are maintaining the manual and documented reports as bellow.

- Transaction reports
- Management reports (on-hand, planning, status)
- Control report (inventory history and cost adjustment)
- Expires
- Sales
- Returns
- Re Orders

Analyzed Outcome – As the answer needs to implement the general reports, those are need to be automate and live updated with all transactions.

Question - Any other report generate requirement.

Answer – They are requesting the Sales reports with between selected days as parameters.

In addition, especially they need to notifications as the must know facts like short expiries, payments, special transactions, and current stock statues.

Analyzed Outcome – as the answer they need some automated inform system like notifications.

9. Volumes

9.1 Question - How many inventory items do you hold?

Answer – they are holding more than 5000 items inventory items at a time.

Analyzed Outcome – database and system need to be implement the manage data storage and data loading.

9.2 Question - How many receiving transactions do you perform on average monthly?

Answer – Normally they are perform more than 10000 transactions per month.

Analyzed Outcome – report generation and data storage need to be implement with good transparency with quality transactions and calculations.

9.3 Question - How many issue transactions do you perform on average monthly?

Answer – as the answer in documented system, they perform issues in average 10% with some error in data gathering and calculations.

Analyzed Outcome – Accuracy of the system must be in better way with implementation.

9.4 Question - How many inventory organizations and warehouses do you have?

Answer – they are having one warehouses and all controls are in a single organization. However, they have an idea to publish business to another

Analyzed Outcome – as the answer need to implement centerline database to connect from another location or another branches with sub division of the organization.

In addition, need to be ruddy and implement the deferent user account supports with sub divisions with centralized database.