A complete and integrated platform, Microsoft SQL Server brings it all together to get more value out of existing IT skills and assets, increase the productivity and agility of IT departments, and quickly build flexible, innovative applications.

4.8SYSTEM DOCUMENTATION AND IMPLEMENTATION

4.8.1How the Program Will Be Installed

Before we start the installation of the HIMS software there are a few things we need to do. One of them is to get the Visual Studio 2012 application software installed on the system. The development environment on which this application is built as at the time this article is written already has the Visual Studio 2012 installed. The steps taken to install the HIMS software are on the ground that the Visual Studio 2012 has been installed.

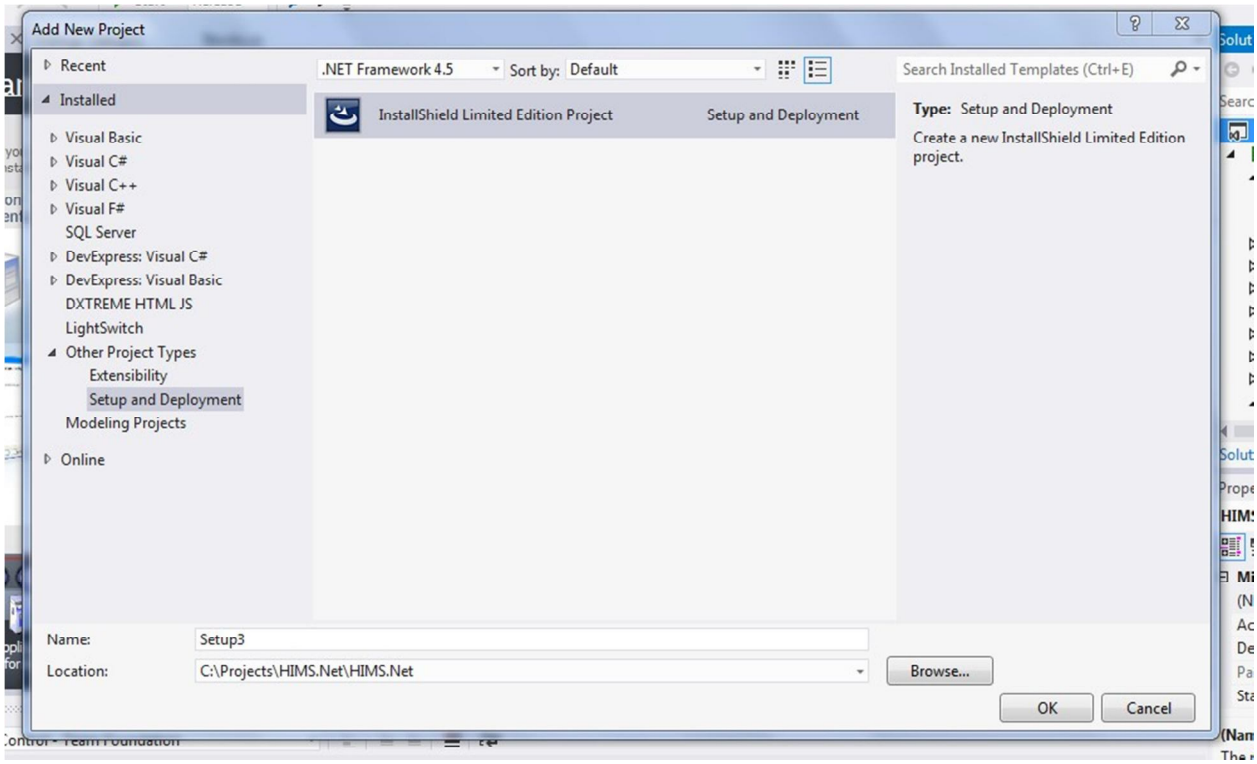
4.8.2 Pre-Installation

Step 1

Launch the Visual Studio Application

Step 2

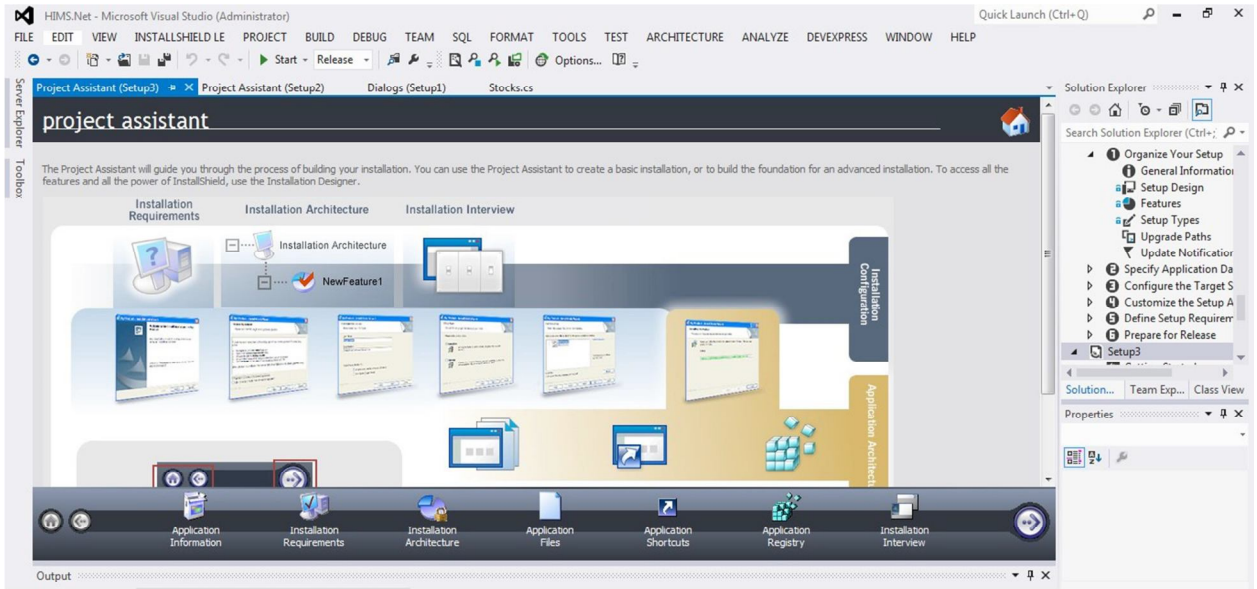
Navigate to the Solution Explorer pane and right click on Solution HIMS**.** Go to **―Add New‖**, Click on **NewProject.**



Step 3

Click on **Setup and Deployment**(on the above screen) and provide name in the **Name** textbox and click **Ok**.

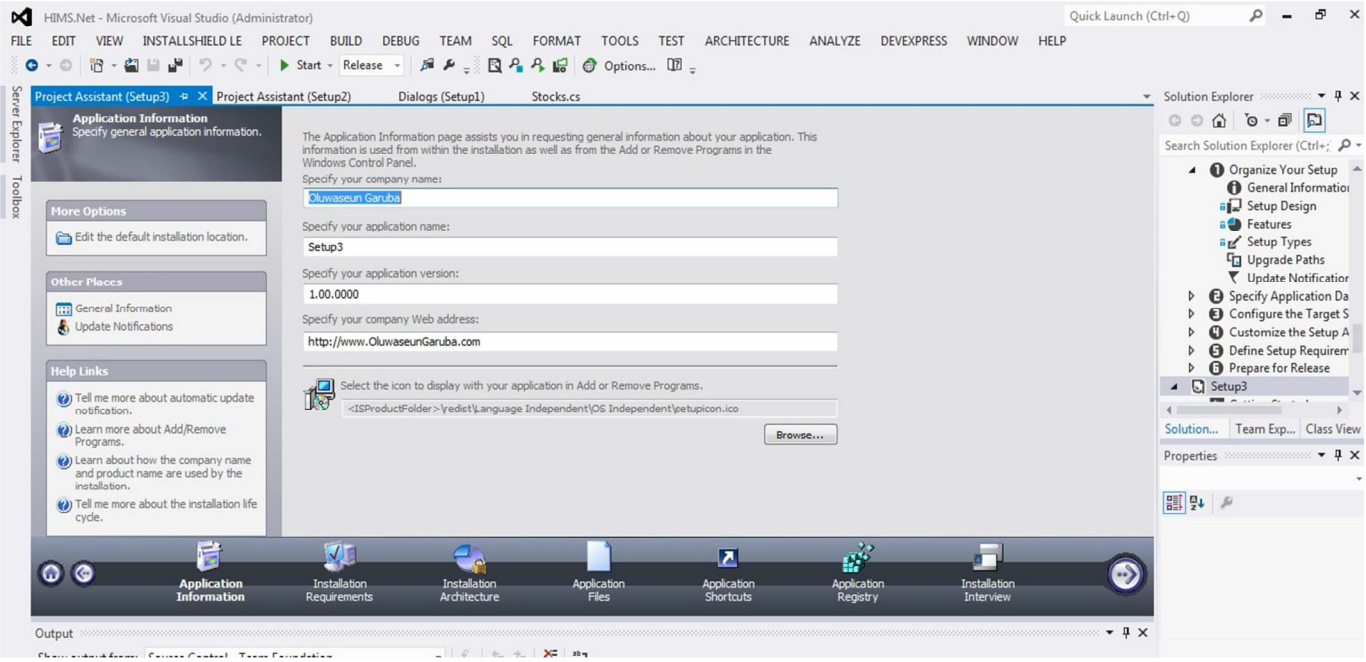
Step 4



On the screen that appears as the one above, click on

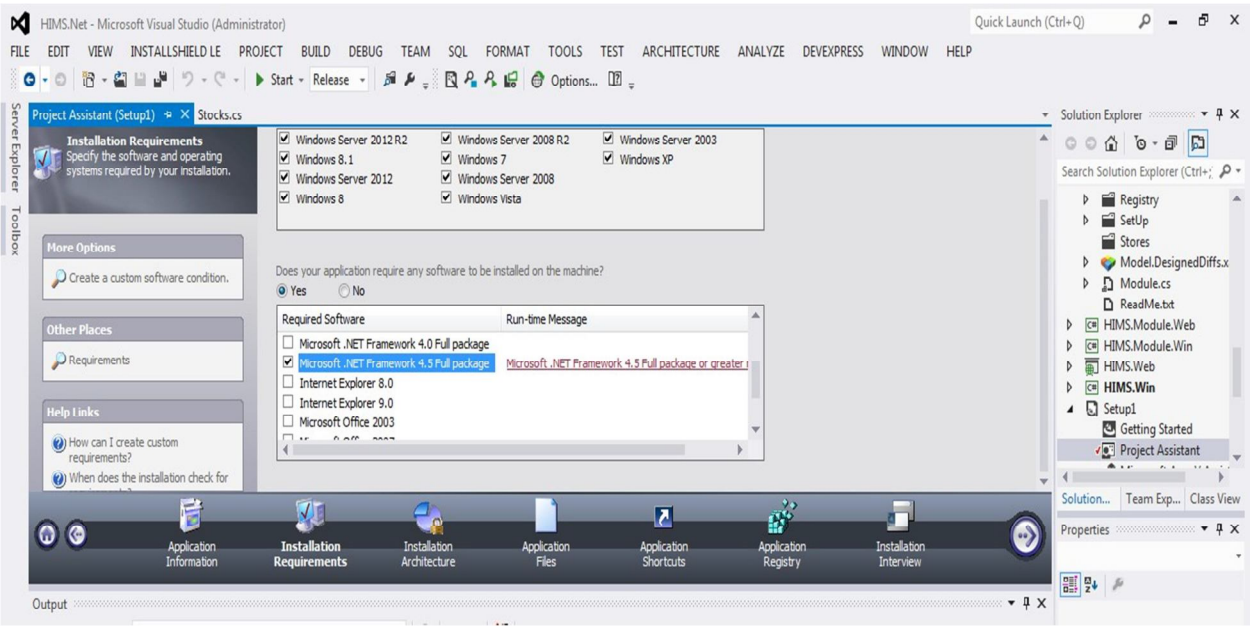
Step 5

On the **Application Information** page, specify your **Company Name**, **Application Name**, **Application Version** and **Company Web address** in the spaces provided. The screen that captures this is as shown below.



Step 6

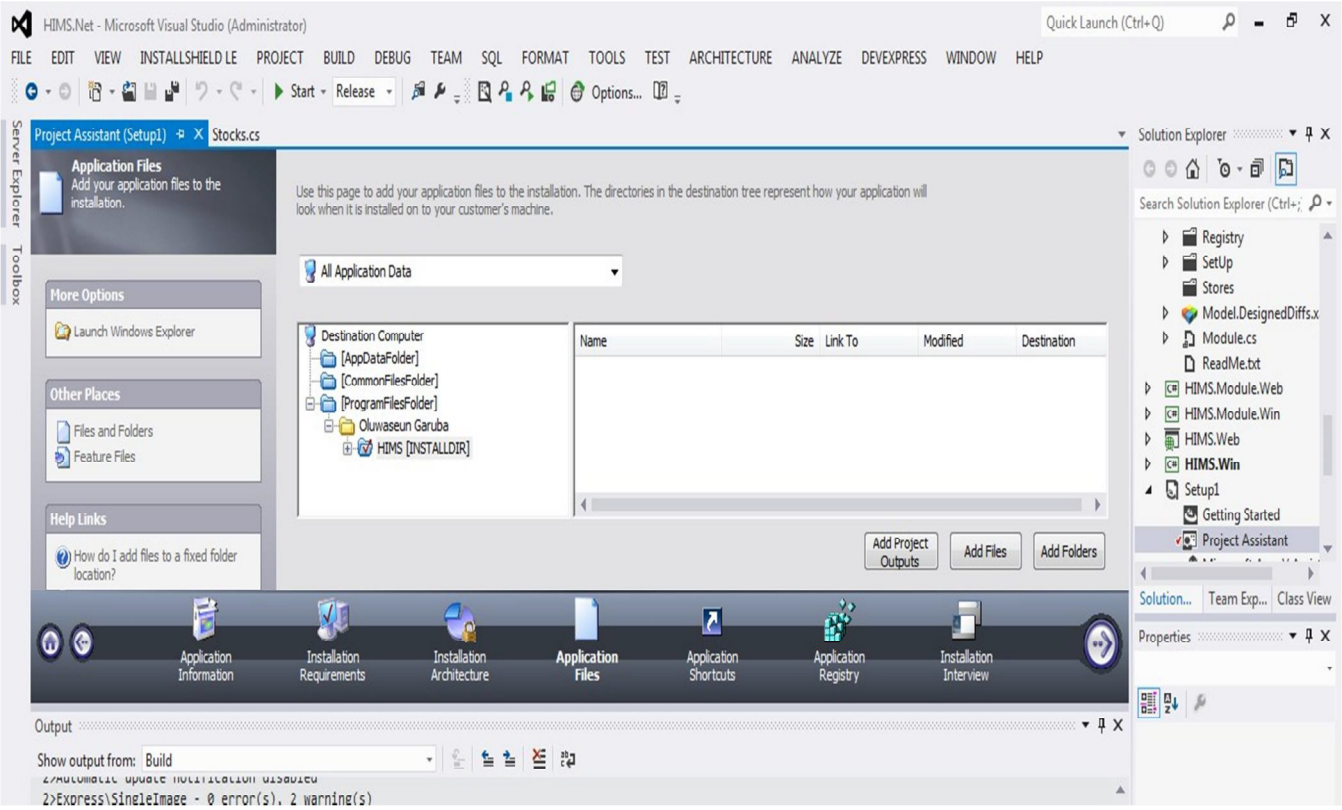
Click on **Installation Requirement.** On the **Software Requirement** area, click **Yes** radio button and tick the **Microsoft.Net Framework 4.5 Full Package** checkbox. An example is shown on



Step 7

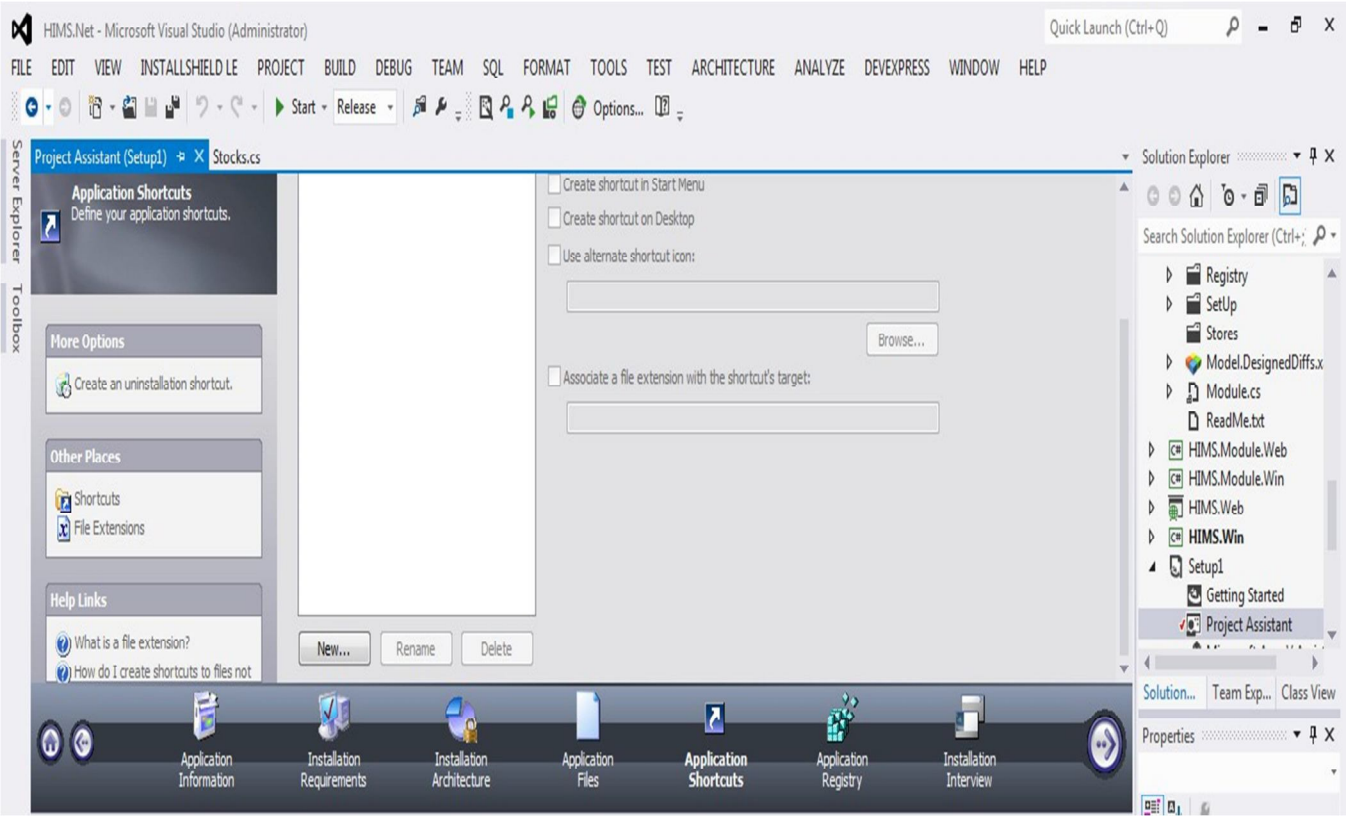
Move to **ApplicationFiles,** click **Add Projects** to specify the folder location for your application.

The screen below shows a representation of what this means.



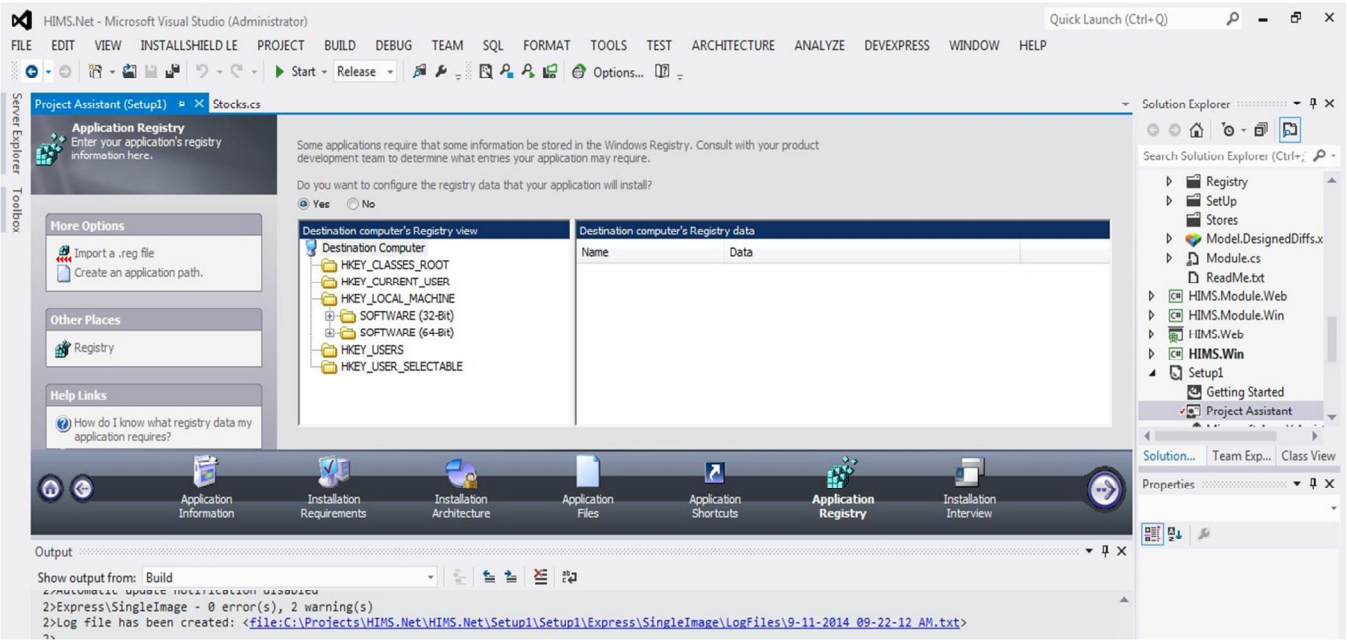
Step 8

Click **Application Shortcuts**



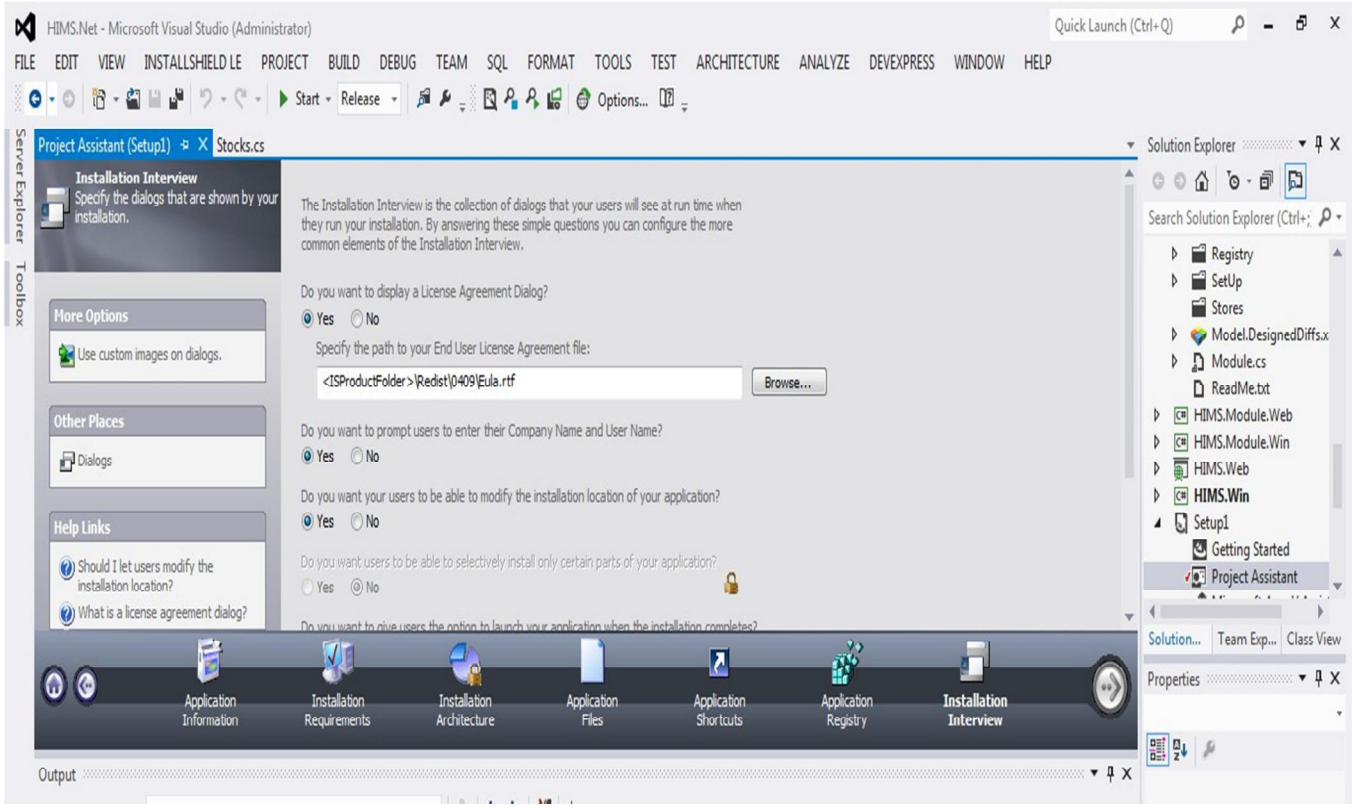
Step 9

Navigate to **Application Registry,** click on the **Yes** radio button to configure the registry data that your application will install. Below shows a screen explaining this.



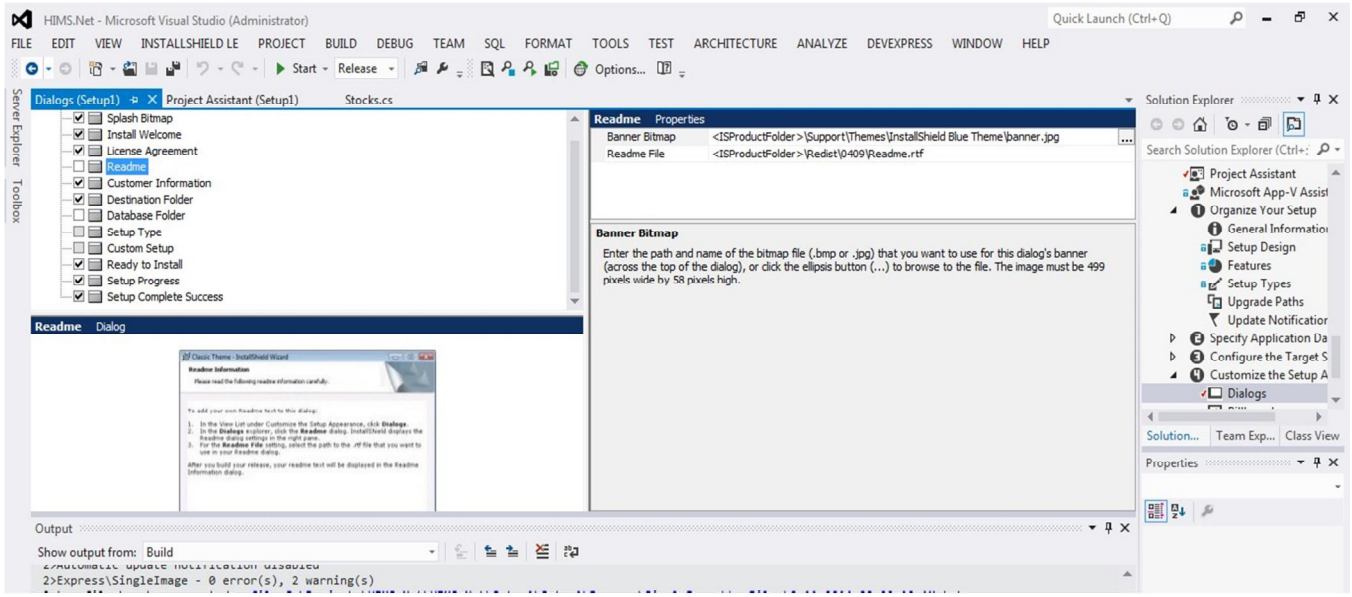
Step 10

Navigate to the **Interview** section, Click **Yes**to the License Agreement Dialog, prompt users to



Step 11

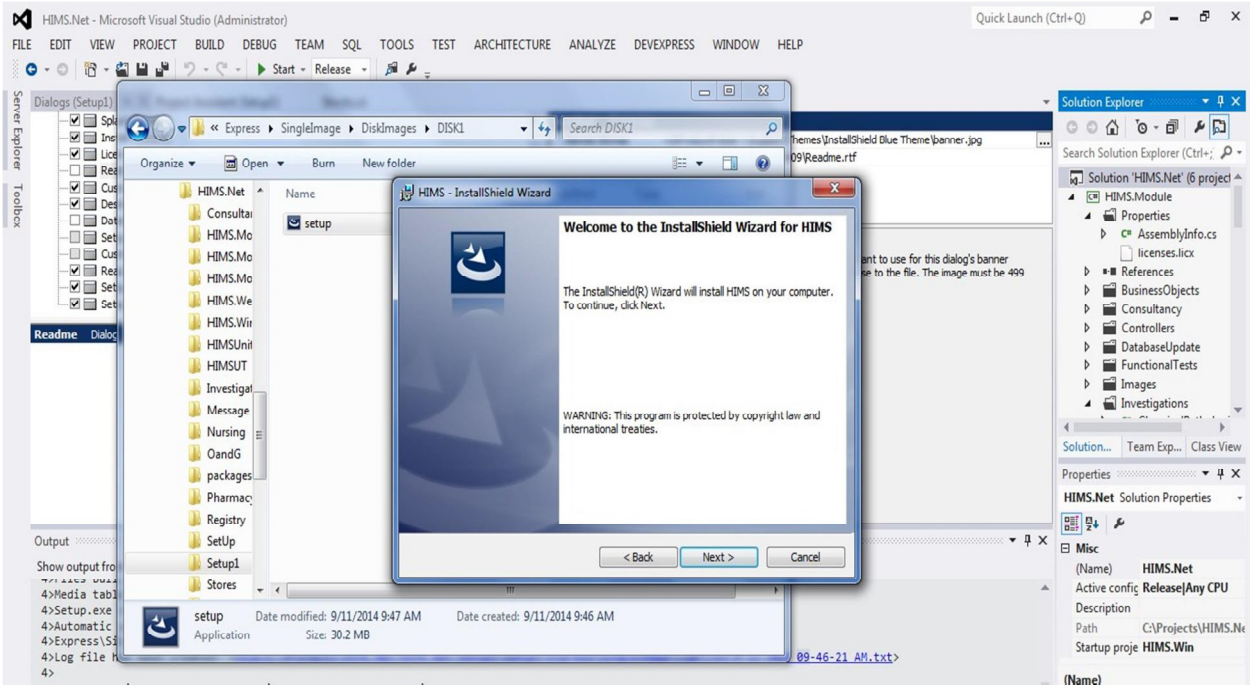
On the **Dialogs tab**, check the boxes of what you want to be a part of your program installation process.



4.8.3Program Installation

Step 1

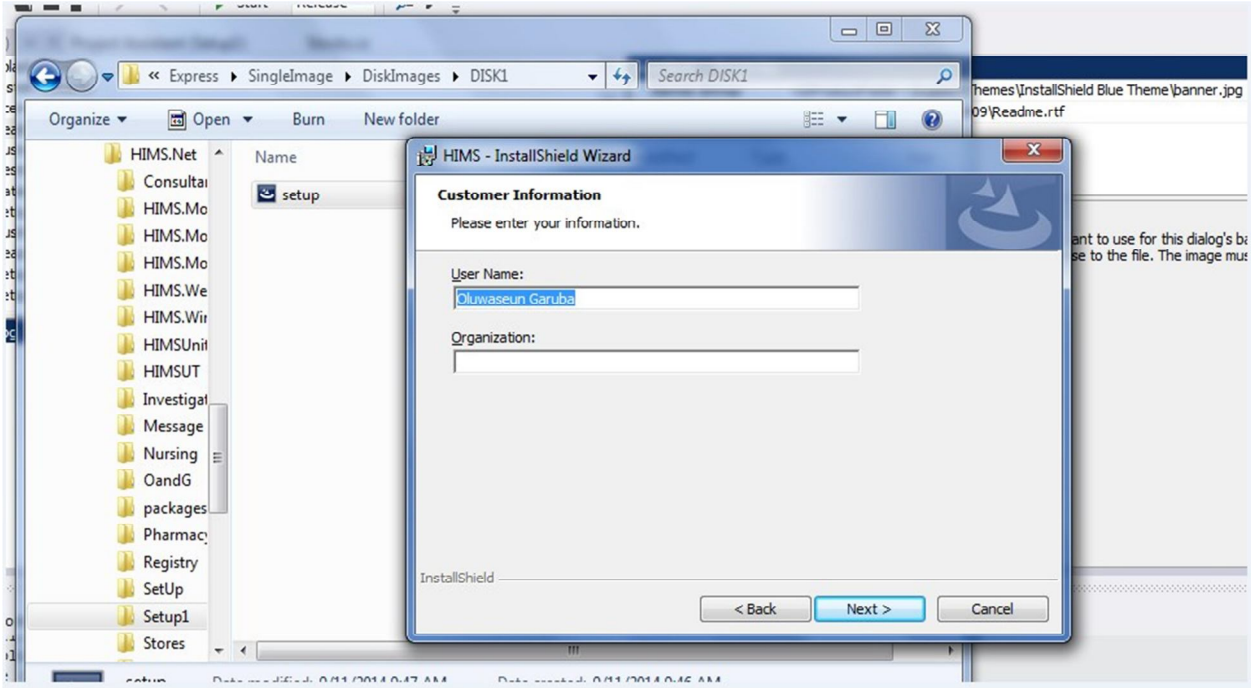
Locate the folder where the application resides and double click on the setup icon. The screen



Click **Next**

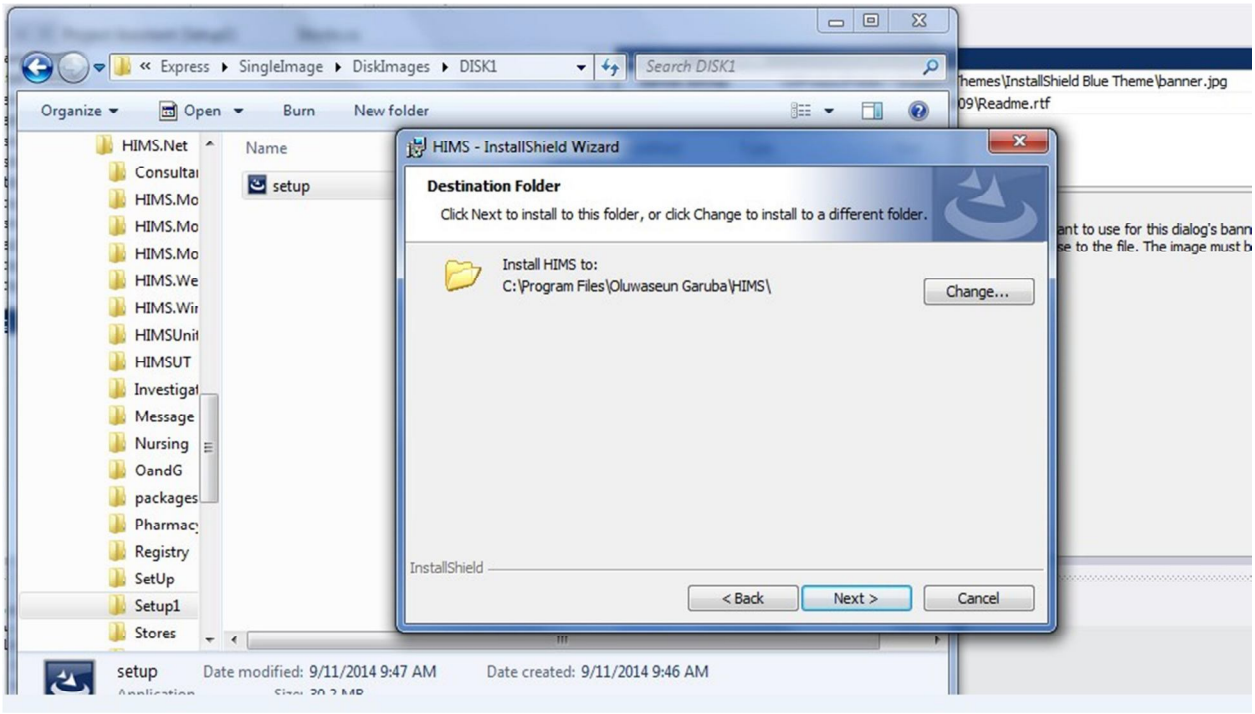
Step 2

Enter **User Name** and **Company Name**and click **Next**.



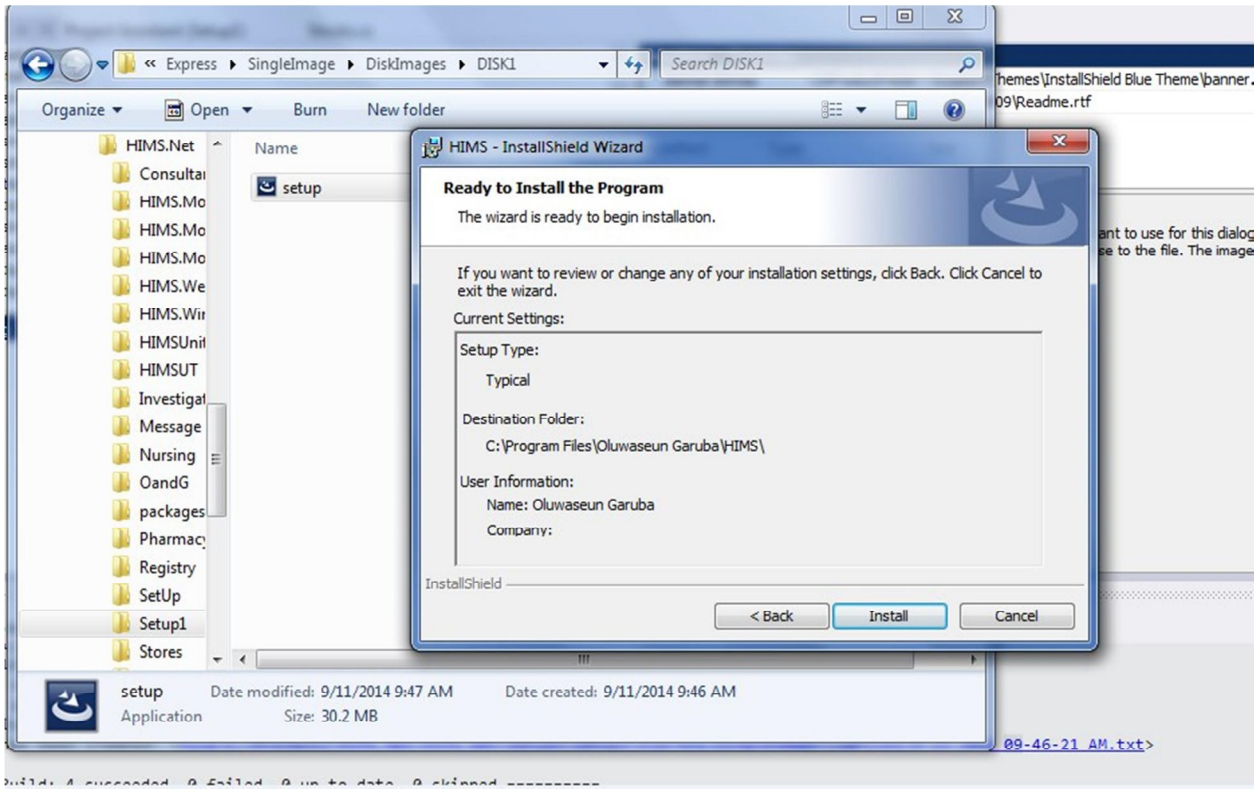
Step 3

Specify the program destination folder and click



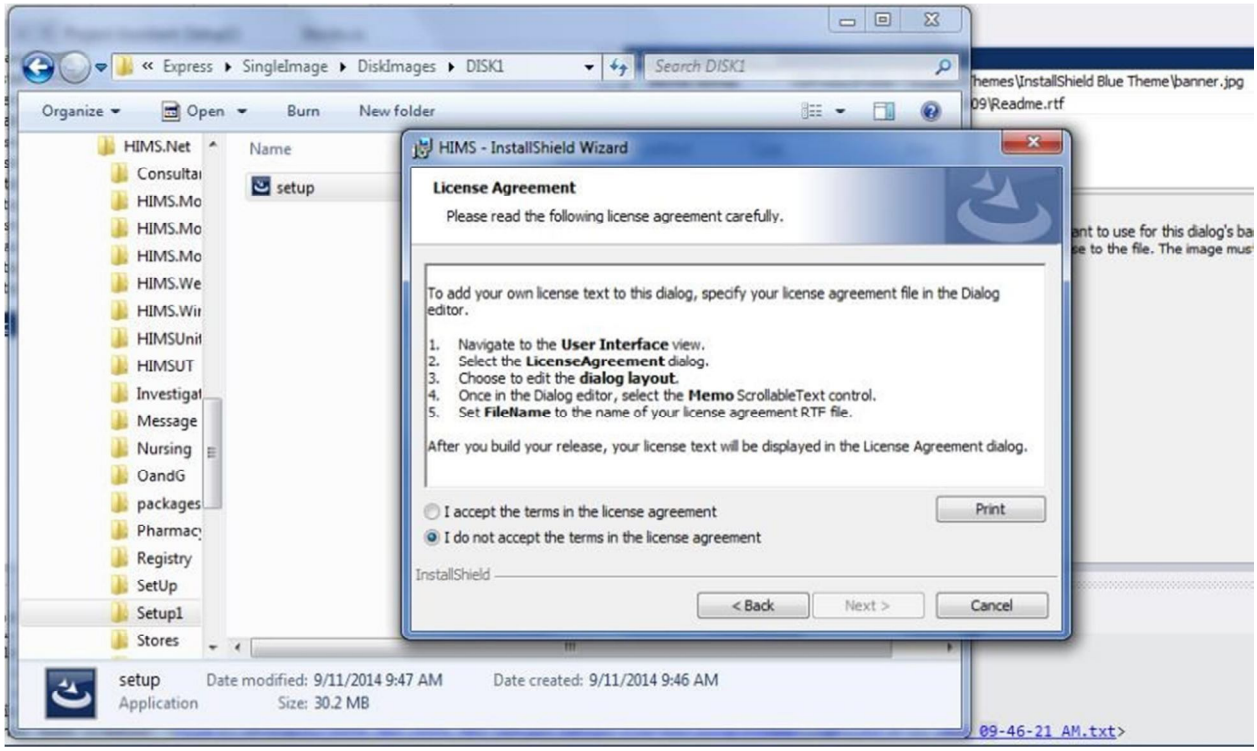
Step 4

Click on **Install** to install the HIMS program.



Step 5

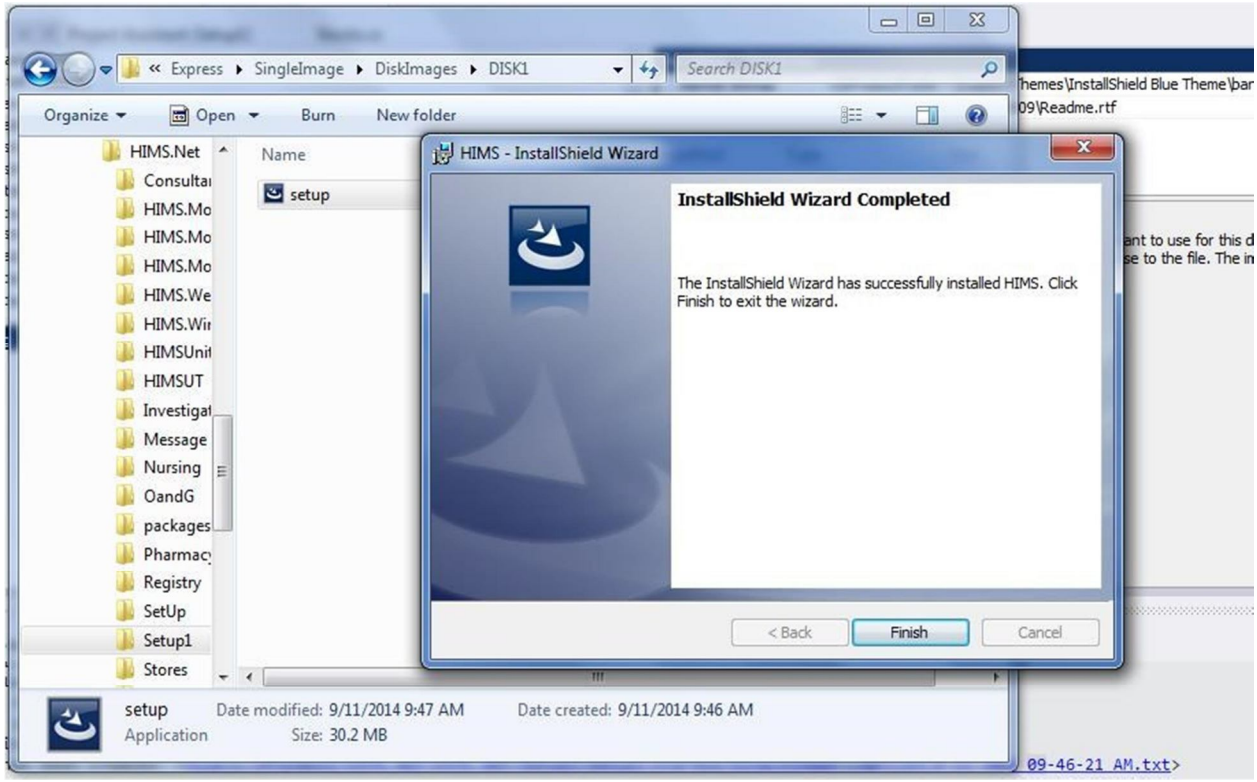
Click on the **I accept the terms in the License Agreement** radio button to accept the terms and conditions and click **Next**



The installation begins

Step 6

Click **Finish** and the program is installed on the system.



Microsoft SQL Server 2012 Installation:

Step by Step Instructions

As per Microsoft, SQL Server 2012 can be installed on Windows 7, Windows Server 2008 R2, Windows Server 2008 Service Pack 2, Windows Vista Service Pack 2.

Note: SQL Server 2012 installation on Windows 7 required me to install Windows 7 SP1. It is recommended to perform Windows update before beginning SQL Server 2012 installation.

Microsoft SQL Server 2012 on 32-bit and 64-bit processors

32-bit systems – Computer with Intel or compatible 1GHz or faster processor (2 GHz or faster is recommended.)

64-bit systems – 1.4 GHz or faster processor
What are memory requirements for SQL Server 2012?

Minimum of 1 GB of RAM (2 GB or more is recommended.) **What**

are hard disc requirements for SQL Server 2012?

2.2 GB of available hard disk space **SQL**

Server 2012 Installation:

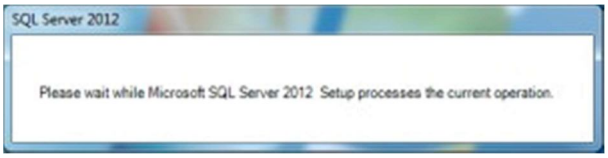
Step 1

Insert SQL Server 2012 installation media into your DVD drive/virtual disc drive.

At this point, you will be prompted (based on your Windows settings) with AutoPlay dialog as

Click on Run SETUP.EXE (Alternatively you can launch set up by locating and double clicking on SETUP.EXE file on the media drive)

At this time, SQL Server 2012 install will run background processes and you should see the following dialog.



STEP 2:

If your OS is compatible, at this point the install should show you the SQL Server Installation Center.

Click on _Installation‘ link (2nd in the list)

STEP 3:

Click on _New SQL Server stand-alone installation or add features to an existing installation‘ (See the image below). All the options on this screen are self-explanatory and I recommend you

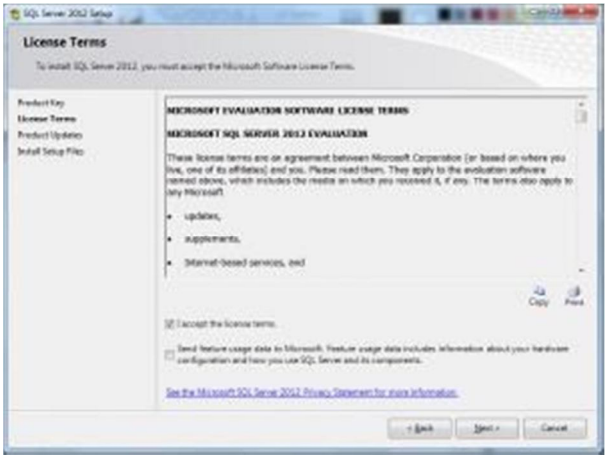
At this point, the install wizard will check for various setup rules and present you with any issues that needs to be resolved for set up to continue (See image below).

STEP 4:

In this step the install will ask for product key. A default option of _Specify a free edition‘ should be already selected and _Evaluation‘ is selected from the drop-down list (See image below).

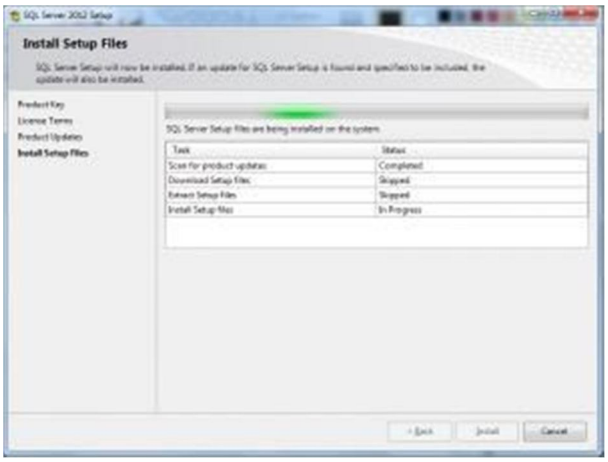
STEP 5:

Accept SQL Server 2012 Evaluation license terms and conditions (See image below).

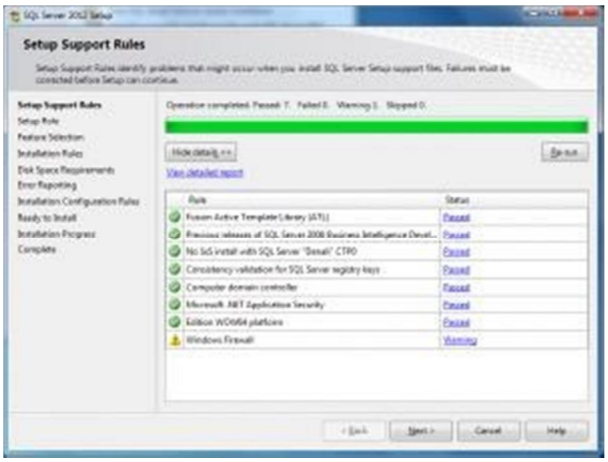


STEP 6:

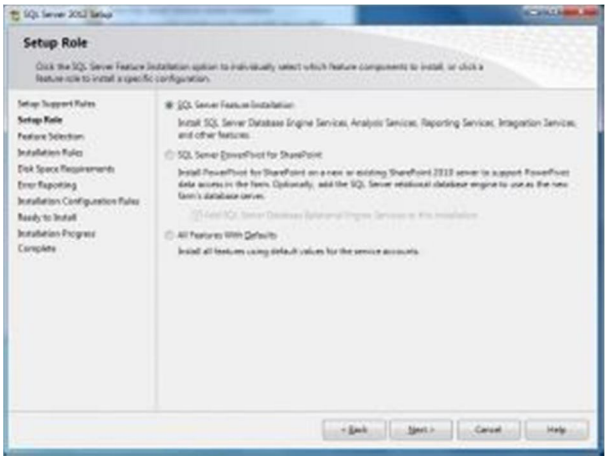
In the next screen, SQL Server 2012 install wizard will ask you to install setup files needed. Click on _Install‘ button (see image below).



Once the setup files are installed, you should see the install complete screen (see image below).Click Next.



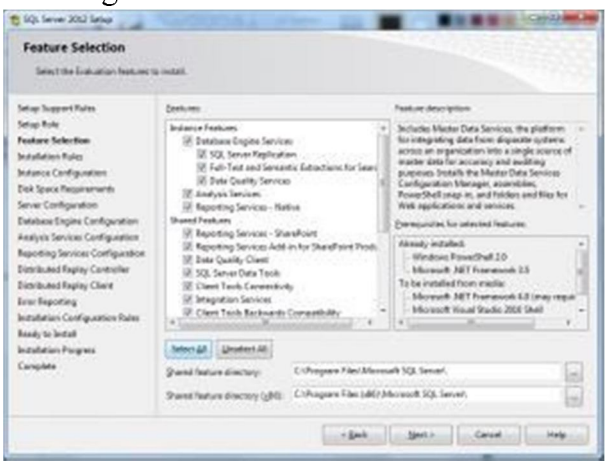
In this step make sure _SQL Server Feature Installation‘ is selected and click Next to continue (see image below).



Step 8:

At this time you should see _Feature Selection‘ screen. I recommend you to click on _Select All‘ if you are new to installing SQL Server (which is why I guess you are using these instructions).

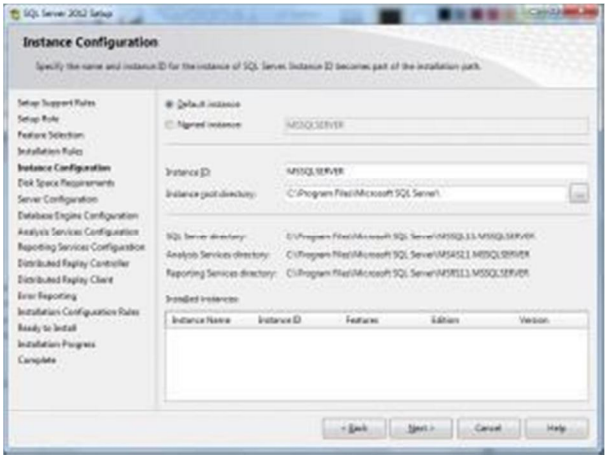
See image below.



At this point, the wizard will again check your system for any blocking issues (see image below).

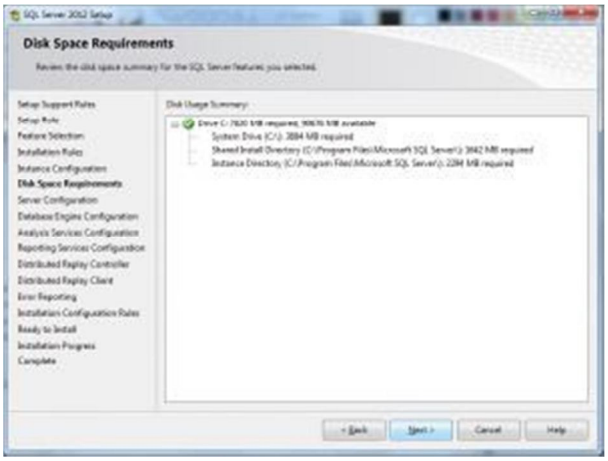
Upon successful completion click _Next‘ **STEP 9:**

In this step you will be able to create a new SQL Server Instance. However, I recommend to go with default settings i.e, default instance (see image below).



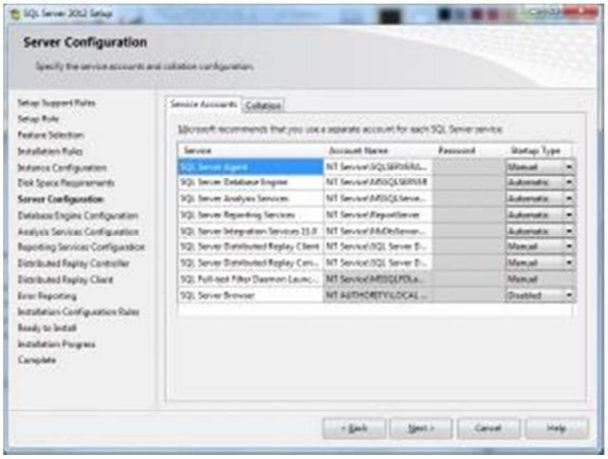
STEP 10:

Installer will check your system for required disk space and provide you a summary (see image below). Click Next.

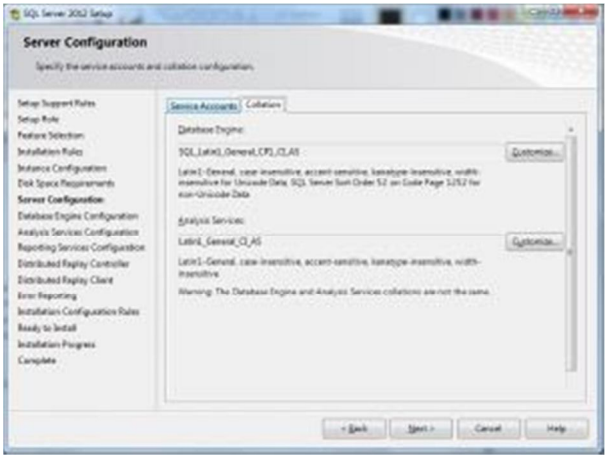


STEP 11:

Unlike previous versions of SQL Server, SQL Server 2012 uses an already created individual and independent service accounts for running various SQL Server services. Review and leave



Before clicking on Next to continue, click on collation tab and read through descriptions for your learning. Leave default selections as-is (see image below).



Click Next to continue.

STEP 12:

In this step, please follow the instructions carefully!!!

As you can see in the image below, you can have either windows authentication mode or mixed mode. Windows authentication mode lets you add only windows users while mixed mode lets you create users with a unique login and password.

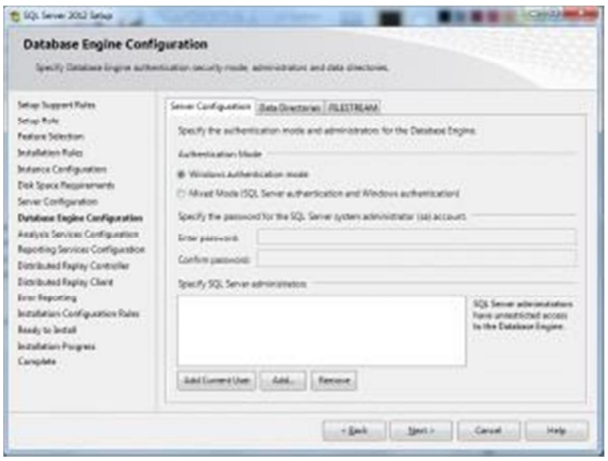
Make sure you select Mixed Mode (converse to what is shown in the image below).

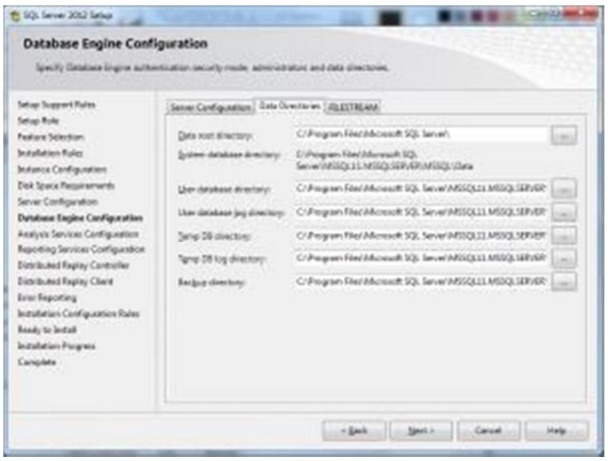
Enter a password for _sa‘ account.

Also make sure to click on _Add Current User _ which will add the account with which you are

Note: Image below does not reflect the instructions above. Follow the instructions above carefully.

Click on Data Directories and FILESTREAM tabs and review for your learning (See images below).



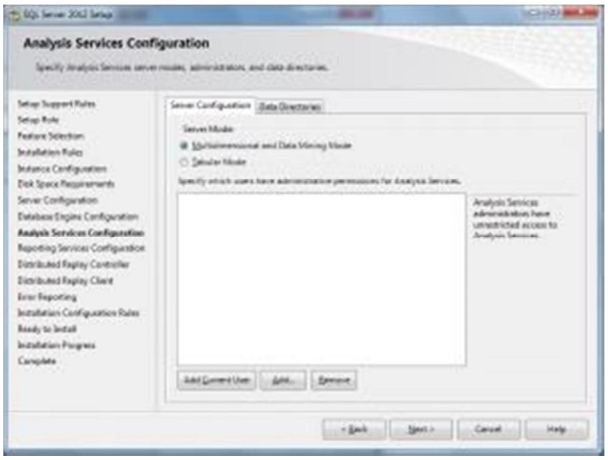


Click Next to continue.

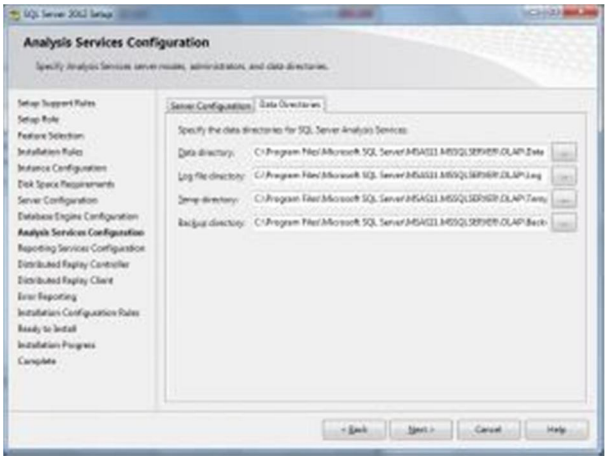
STEP 13:

Leave default selections as-is.

Make sure to click on _Add Current User‘ which will add the account with which you are



Click on Data Directories tab to review (see image below).

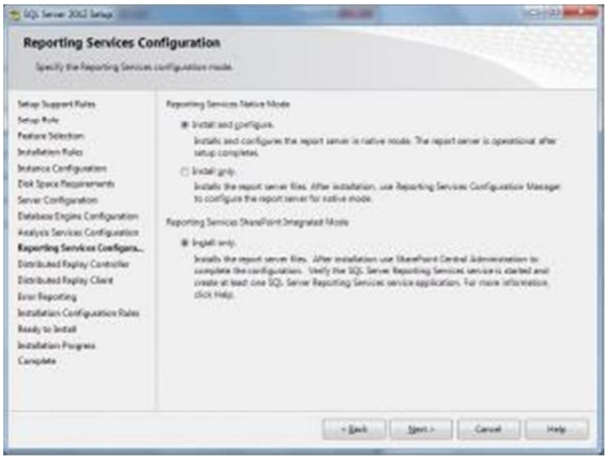


Click Next to continue.

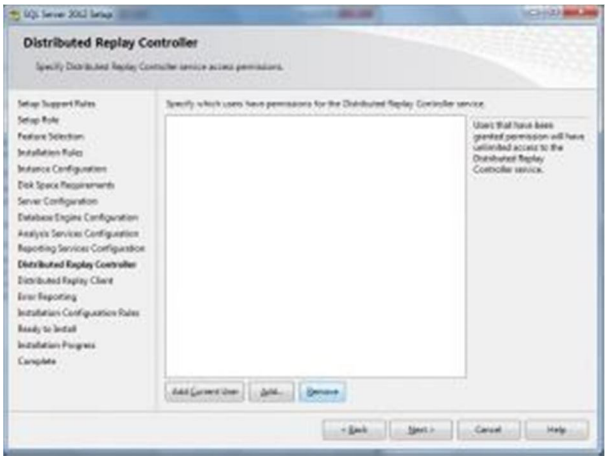
STEP 14:

In this step leave the default options as-is.

Click Next to continue.



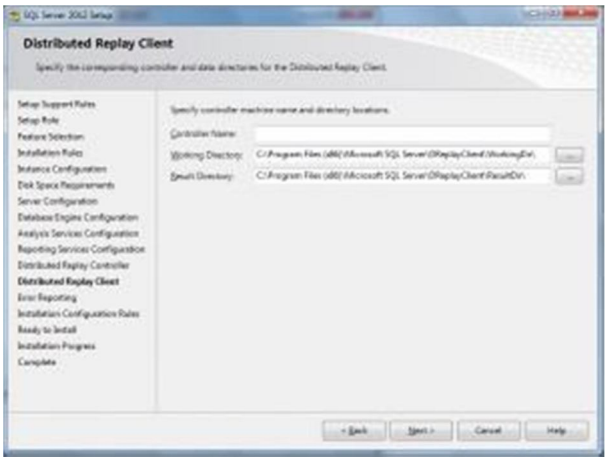
In this step Click Add Current User to add your windows account as an administrator account for Distributed Replay Controller.



Click Next to continue.

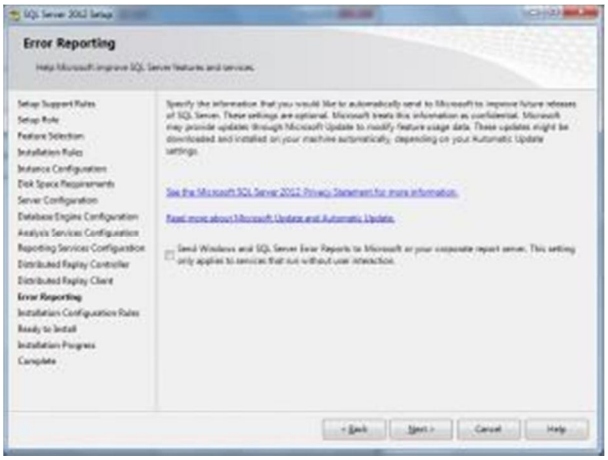
STEP 16:

Leave the Controller Name blank and Click Next to Continue (see image below).



STEP 17:

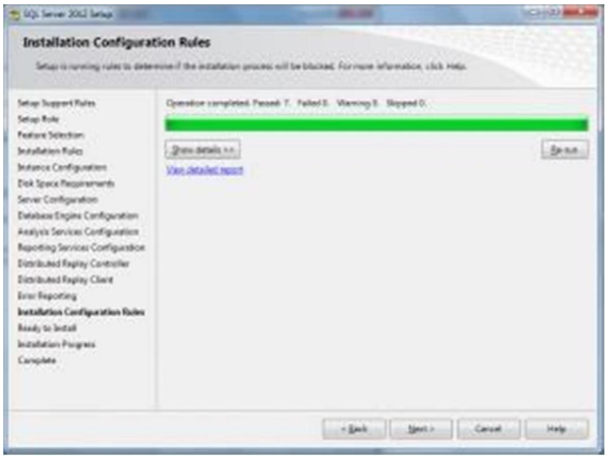
Check box if you would like to send error reports to Microsoft. I have left it as-is (see image



Click Next to Continue.

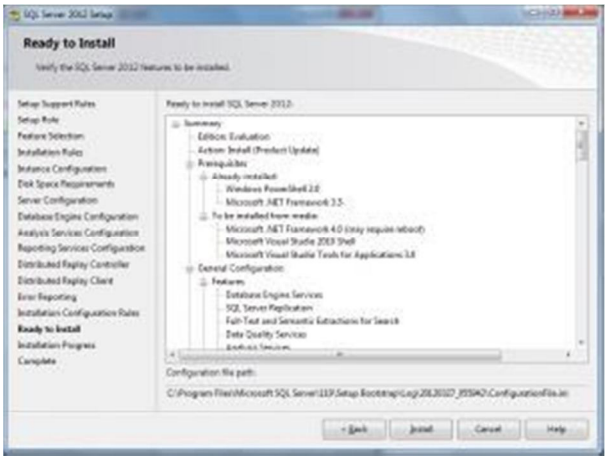
At this point, set up will again verify your system for any potential blocking issues.

Once the operation is completed, click Next to continue (see image below).

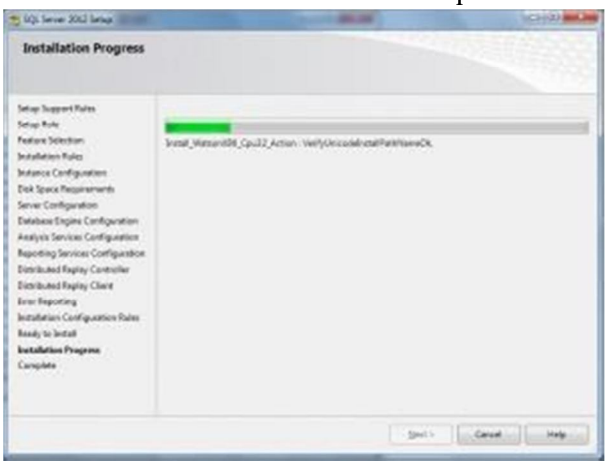


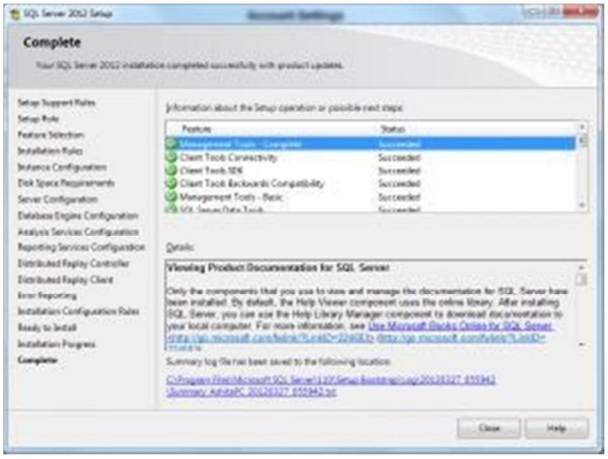
STEP 18:

Review the options you selected for the install and verify the path to which configuration file will be stored. You can take advantage of this configuration file for future installs.



Installer will take a while in this step before the installation is complete.





4.8.4:Hardware and Operating Systems Requirements

Hardware Requirement

SERVER

Business use- The Power® 740 Express® server offers large memory capacity, outstanding performance, and industrial-strength virtualization for smallto-midsize database servers and server consolidation.

UPS (UNINTERRUPTIBLE POWER SYSTEM)

HP T750J Uninterruptible Power System (AF456A)

Electrical input connection:	NEMA 5-15P
Electrical output connections:	(6) NEMA 5-15R
Communications serial ports:	Standard DB-9 port
Load segments:	1
Battery type:	Maintenance-free, sealed, valve-regulated acid (VRLA)
Hot-swap batteries:	Yes
Communications software:	HP Power Protector
Operating voltage:	100V, 110V

Software Requirement

Processor (Max) -	Sixteen 64-bit 3.55 GHz POWER7 cores
Memory (Min/Max) -	8 GB to 512 GB DDR3
Internal storage (Max) -	Up to 4.8 TB

4.8.5:How the Software will be used

The software will be installed on the hospital **server**and individual clients systems (User computers) will be connected to it. This method is adopted for centralized management, security reasons and for the sharing of common resources.

4.8.6:Training Requirements

The training requirements of the staff for the use of the software include:

1. A basic knowledge of Ms Office.
2. Knowledge of Ms Server 2012 administration for the safe installation and running of

3. Must have worked with the hospital for a minimum of 3 years to be sure the staff has basic knowledge of the workflow in the hospital environment.

4.8.7: Conversion Procedure

There are basically 3 types of conversion procedure (implementation). Below is a list of them:

1. Direct Changeover
2. Parallel running
3. Phased Implementation

Direct Changeover

This is one of the methods to change from an existing system to a new one. Although this is a simple method, it takes a lot of planning. All the files have to be ready to load all the files have to be ready to load, all the users of the system need to have been trained in advance and the system must have been properly tested. If the system doesn't work when switched on, the system will have to be shutdown the users can't go back to the old system.

Advantages

- Takes the minimal time and effort
- The new system is up and running immediately

Disadvantages

- If the new system fails, there is no back-up system, so data can be lost

Parallel Running

In this method of implementation, a new system and an existing system run side by side. To input the same data and perform the same processes, compare their output and prove the reliability of the new system if the new system is accepted, the existing system will

Advantages

- If anything goes wrong with the new system, the old system will act as a back-up.
- The outputs from the old and new systems can be compared to check that the new system is running correctly

Disadvantages

- Entering data into two systems, and running two systems together, takes a lot of extra time and effort

Phased Implementation

The phased implementation is a changeover that takes place in stages.

Advantages

- Allows users to gradually get used to the new system
- Staff training can be done in stages

Disadvantage

- If a part of the new system fails, there is no back-up system, so data can be lost

This project adopts the Phased Implementation for conversion form the manual to the new software. The phased implementation is done such that the each module in the new system is run and if it works well, the next module is run and so on. In the case where one of the module doesn‘t run properly, the module is taken out and the needed change effected and re-introduced to the system. This is done until the entire module of the new system is implemented.

4.8.8:Adoption Date

The new software will be fully adopted and commissioned, when the programmed is installed,

the staff who will use the system are all trained, and then the new software will be adopted by the hospital.

CHAPTER FIVE

5.0 SUMMARY, CONCLUSION AND RECOMMENDATION

5.1 SUMMARY

The project work was embarked upon with the major aim of reducing the administrative difficulties and difficulties staff and patients face in the General Hospital, Asokoro. The current manual system which involved a lot of stress, a labor intensive and time consuming process were a patient moves from one office to another for registration and where staffs(depending on the one handling an issue) will have to sit all day attending to patients one after the other was analyzed carefully and the problems were highlighted.

REFERENCES

Hern-Underwood M. J. and Workman(1993) *Group climate: a significant retention factor for pediatric nurse managers.*

O'Brien, J (1999). *Management Information Systems – Managing Information Technology in the Internetworked Enterprise*. Boston: Irwin McGraw-Hill.

Cholewka, P. A. (2006). *Implementation of a health care information system in Lithuania*

Brailer, D. J. (2005). *Economic perspective on health information technology*.

Wilcke, B. (2008). *Finding the knowledge in Information.*

Paul R. Vegoda(1987).

Dujat C, Haux R, Schmucker P, Winter A.(1996). *Digital Optical Archiving of Medical Records in Hospital Information Systems- a practical approach towards the computer-based patient record*

Onwujekwe, O. (2005). *Inequities in healthcare seeking in treatment of communicable endemic diseases in Southeast Nigeria.*

Ouma, S., & Herselman, M. E. (2008). *E-health in rural areas: Case of developing countries.*

Pant, S., Hsu, C., (1995), *Strategic Information Systems Planning: A Review, Information Resources Management Association International Conference*, May 21–24, Atlanta.

Serena, (2007), *An Introduction to Agile Software Development.*

Keenan, C. R., Nguyen, H. H., & Srinivasan, M. (2006). Electronic medical records and their impact on residents and medical student education.

R. Goe and M. Kenney. 1990. *"Should Information Be a Social or Economic Good in the Emerging Information Society." In Sven Lundstedt (ed.) Telecommunications, Values, and the Public Interest*

Namita, K. Mansi, C., Avinash, C. *Project Report on Hospital Management System.*

Williams Laurie, (2007), *A Survey of Agile Development Methodologies.*

Nabimara Charles, (2007), *Computerized Health Records Management System*

Idowu, B., Adagunodo, R., & Adedoyin, R. (2006), *Information technology infusion model for health sector in developing country: Nigeria as a case.*

Grimm, N., & Shaw, N. (2007), *Using participatory action research to evaluate and improve change: Pilot implementation of electronic health records in rural Tanzania.*

Krishna, R., Kelleher, K., & Stahlberg, E. (2007). *Patient confidentiality in the research use of clinical medical databases.*

Gillies, A., & Howard, J. (2005). An international comparison of information in adverse events

Sisniega, L. C. (2009). *Barriers to electronic government use as perceived by citizens at the municipal level in Mexico.*

Svensson, P. (2002).

Stone, T., Patrick, T. B., & Brown, G. D. (2005). *Strategic Management of Information Systems in Healthcare*.

Morath, J. M., & Turnbull, J. E. (2005). *To do no harm: Ensuring patient safety in health care organizations.*

Cohan, P. S. (2005). *CFO's to tech: I'll spend for the right technology.*

Furukawa, M. F., Raghu, T. S., Spaulding, T. J., & Vinze, A. (2008). *Adoption of health information technology for medication safety in U.S. hospitals.*

Fuji, K. T., & Galt, K. A. (2008). *Pharmacists and health information technology: Emerging issues in patient safety.*

Harrison, J., & McDowell, G. (2008). *The role of laboratory information systems in healthcare quality improvement.*

Woodside, J. M. (2007). *EDI and ERP: A real-time framework for healthcare data exchange.*

Phillips, J. T. (2009). *Selecting software for managing physical & electronic records.*

Crane, J., & Crane, F. G. (2006). *Preventing medication errors in hospitals through a systems approach and technological innovation: A prescription for 2010.*

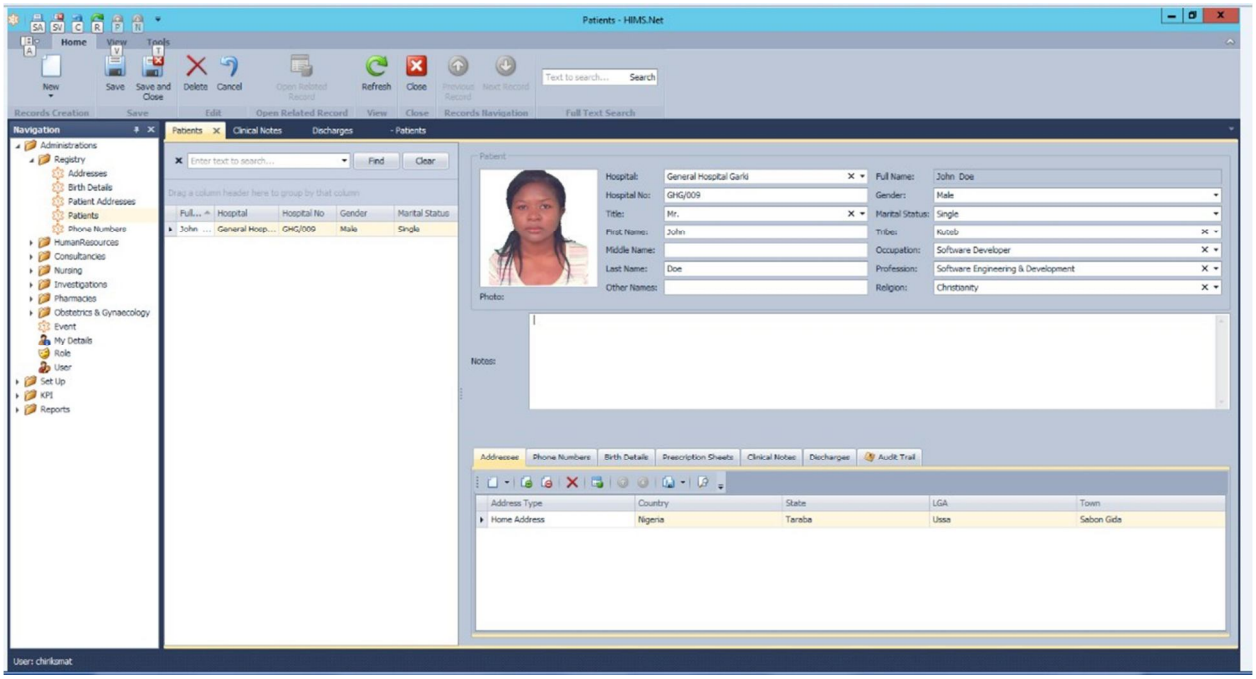
Rennolls, K., & AL-Shawabkeh, A. (2008). *Formal structures for data mining, knowledge discovery and communication in a knowledge management environment.*

Linz, A. J., & Fallon L.F Jr.(2008). *Public perception regading the impact of electronic medical records on health care quality and medical errors. Journal of Controversial Medical Claims.*

Pineault et al.(1989).

Appendix A

Sample Outputs Login Screen



Investigations Histopathology Screen

Navigation
- Administration
  - My Details
  - Role
  - User
- Registry
- Consultancies
- Nursing
- Investigations
  - Chemical Pathologies
  - Haematologies
  - Histo Pathologies
  - Microbiologies
  - Profile
  - Radiologies
  - Serologies
- Pharmacies
- Obstetrics & Gynaecology
- Set Up
- Reports

Histo Pathologies | Bilikisu Garuba - Histo Pathc | Histo Pathologies

Histo Pathologies

Patient Name:		State No:	
Lab No:		Date Recieved:	
Lab Scientist:		Date Issued:	
Chief Lab Scientist:		No Of Pieces:	
Diagnosis:		No Of Blocks:	
Required Exam:		Type:	
Any Previous:		Consultant:	
Clinical Details:			

Notes:

Audit Trail

User Name	Modified On	Operation Type	Property Name

Home | View | Tools

Navigation
- Administration
  - My Details
  - Role
  - User
- Registry
- Consultancies
- Nursing
- Investigations
  - Chemical Pathologies
  - Haematologies
  - Histo Pathologies
  - Microbiologies
  - Profile
  - Radiologies
  - Serologies
- Pharmacies
- Obstetrics & Gynaecology
- Set Up
- Reports

Histo Pathologies | Bilikisu Garuba - Histo Pathc | Histo Pathologies | Microbiologies | Bilikisu Garuba - Microbiolog

Microbiologies

Patient Name:	Bilikisu Garuba	Isolate Stage:	Antiology
Unit Code:	MB101	Diagnosis:	Typhoid
Lab No:	101	Lab Scientist:	Helen Turner
Type:		Date:	2/25/2014
Antibiogram:	Antiology	Chief Lab Scientist:	Aperun Aisha

Notes:

Audit Trail

User Name	Modified On	Operation Type	Property Name
seun	3/11/2014	InitialValueAssigned	LabScientist
seun	3/11/2014	InitialValueAssigned	LabNo
seun	3/11/2014	InitialValueAssigned	Date
seun	3/11/2014	InitialValueAssigned	ChiefLabScientist

Chemical Pathology

Home View Tools

Navigation

- Administration
  - My Details
  - Role
  - User
- Registry
- Consultancies
- Nursing
- Investigations
  - Chemical Pathologies
  - Haematologies
  - Histo Pathologies
  - Microbiologies
  - Profile
  - Radiologies
  - Serologies
- Pharmacies
- Obstetrics & Gynaecology
- Set Up
- Reports

Chemical Pathologies | seun - User | Bilikisu Garuba - Chemical Pa

Chemical Pathologies

Patient Name:	Bilikisu Garuba	Clinical Details:	
Specimen Date:	5/16/2014	Specimen:	
Specimen Time:	1:48pm	Consultant:	Dr. A.R Garuba
Recieved Date:	5/16/2014	Pathologist:	Kunle
Recieved Time:	2:00pm	Lab Scientist:	Helen Turner
Specimen Condition:	ok	Chief Lab Scientist:	Aperun Aisha
Diagnosis:	Typhoid	Lab No:	101

Notes:

Audit Trail

User Name	Modified On	Operation Type	Property Name

Navigation

- Administration
  - My Details
  - Role
  - User
- Registry
- Consultancies
- Nursing
- Investigations
  - Chemical Pathologies
  - Haematologies
  - Histo Pathologies
  - Microbiologies
  - Profile
  - Radiologies
  - Serologies
- Pharmacies
- Obstetrics & Gynaecology
- Set Up
- Reports

Chemical Pathologies | seun - User | Bilikisu Garuba - Chemical Pa | Haematologies | Bilikisu Garuba - Haematolo

Haematologies

Patient Name:	Bilikisu Garuba	Speciemen Collection:	
Lab No:	101	Speciemen Collection Date:	5/16/2014
Recieved Date:	5/16/2014	Diagnosis:	Malaria
Reported Date:	5/16/2014	Haematologist:	Michela Udioh
Reported Time:	1:48pm	Lab Scientist:	Helen Turner
Specimen Nature:		Chief Lab Scientist:	Aperun Aisha
Required Examination:	Malaria Parasite		

Notes:

Audit Trail

User Name	Modified On	Operation Type	Property Name

Nursing Treatments

Home View Tools

Navigation

- Administration
  - My Details
  - Role
  - User
- Registry
- Consultancies
- Nursing
  - Administrations
  - Admissions
  - Consultants
  - Diet Restrictions
  - Fluid Chart
  - Intravenous Therapy
  - Take Home
  - Treatments
  - Vital Signs
- Investigations
- Pharmacies
- Obstetrics & Gynaecology
- Set Up
- Reports

seun - User | Micheal Randy - Treatments

Treatments

Patient Name:	Micheal Randy	Instruction:	Drugs to be taken twice daily
Drug Type:	Capsules	Pharmacist:	Bola Are
Drug Description:	Capsules	Attendant Nurse:	Nwanneka Iwu
Dosage:	3 capsules twice daily	Frequency:	8 hourly
Route:	NA	Duration:	5 days
Physician:	Dr. A.R Garuba		

Notes:

Audit Trail

User Name	Modified On	Operation Type	Property Name
seun	3/11/2014	InitialValueAssigned	PatientName
seun	3/11/2014	InitialValueAssigned	Dosage
seun	3/11/2014	InitialValueAssigned	Route
seun	3/11/2014	InitialValueAssigned	Frequency
seun	3/11/2014	InitialValueAssigned	Pharmacist

Navigation | Appearance | Windows | Actions

Navigation

- Administration
  - My Details
  - Role
  - User
- Registry
- Consultancies
- Nursing
  - Administrations
  - Admissions
  - Consultants
  - Diet Restrictions
  - Fluid Chart
  - Intravenous Therapy
  - Take Home
  - Treatments
  - Vital Signs
- Investigations
- Pharmacies
- Obstetrics & Gynaecology
- Set Up
- Reports

seun - User | Discharges | Administrations | Administrations

Administrations

Timing:	Bisola Olaitan	Physician:	Dr. Sabo Vinishe
Duration:	1 month	Pharmacist:	Micheala Udioh
Attendant Nurse:	Jeremiah Lan	Date:	2/10/2014

Notes:

Audit Trail

User Name	Modified On	Operation Type	Property Name
seun	3/11/2014	InitialValueAssigned	Physician
seun	3/11/2014	ObjectCreated	
seun	3/11/2014	InitialValueAssigned	Pharmacist
seun	3/11/2014	InitialValueAssigned	Date

Home View Tools

Navigation

- Administration
  - My Details
  - Role
  - User
- Registry
- Consultancies
- Nursing
- Investigations
- Pharmacies
  - Locations
  - Prescriptions
  - Schedules
  - Sheets
  - Stocks
- Obstetrics & Gynaecology
- Set Up
- Reports

Bilikisu Garuba - Microbiolog | Prescriptions | Bilikisu Garuba - Prescriptior | Stocks | Septrin - Stocks

Stocks

Drug Name:	Septrin	Pharmacy Code:	PH102
Drug Category:	Tablets	Expiry Date:	9/26/2014
Cost Price:	20 Naira	Collected By:	Francis
Quantity:	20	Supplier Code:	SC 101
Selling Price:	30	Manufacture Date:	10/6/2013
Discounted Amount:	NA	Dosage:	3 tablets twice daily
Discount Percent:	NA	Direction Of Use:	NA
Discount Computed:	NA	Status:	Available
Date Of Collection:	2/10/2014	Constituents:	Adenotil, propanol
Batch No:	ST 102	Stock Date:	1/5/2014
NAFDACReg No:	DE10034		

Notes:

Audit Trail

User Name	Modified On	Operation Type	Property Name
seun	3/11/2014	InitialValueAssigned	SupplierCode
seun	3/11/2014	InitialValueAssigned	StockDate

Prescriptions

Navigation
- Administration
  - My Details
  - Role
  - User
- Registry
- Consultancies
- Nursing
- Investigations
- Pharmacies
  - Locations
  - Prescriptions
  - Schedules
  - Sheets
  - Stocks
- Obstetrics & Gynaecology
- Set Up
- Reports

Bilikisu Garuba - Microbiolog | Prescriptions | Bilikisu Garuba - Prescriptior

Prescriptions

Patient Name:	Bilikisu Garuba	Duration:	3 days
Drug Category:	Tablets	Amount:	50 Naira
Pharmacist:	Micheala Udioh	Quantity:	20
Description:		Total Amount:	1000
Dosage:	3 tablets twice daily	Sheet Code:	24432
Doctor:	Paul White		

Notes:

Audit Trail

User Name	Modified On	Operation Type	Property Name
seun	3/11/2014	InitialValueAssigned	Duration
seun	3/11/2014	InitialValueAssigned	Doctor
seun	3/11/2014	ObjectCreated	
seun	3/11/2014	InitialValueAssigned	DrugCategory

Navigation
- Administration
  - My Details
  - Role
  - User
- Registry
- Consultancies
- Nursing
- Investigations
- Pharmacies
- Obstetrics & Gynaecology
  - Antenatals
  - APGARScore
  - Deliveries
  - Schedules
- Set Up
- Reports

Sunday Umana - Antenatals

Antenatals

Patient Name:	Sunday Umana	Feet:	2
Last Menstrual Period:	9/8/2013	Sickle:	
Gravida:		General:	
Blood Group:	AB+	Attendant Nurse:	Chioma Ekebi
Height:	5.8'	Physician:	Paul White
Weight:	52kg	Expected Date Of Delivery:	5/13/2014
Pelvis:	4.5	Date:	3/11/2014
Nipples:			

Notes:

Audit Trail

User Name	Modified On	Operation Type	Property Name
seun	3/11/2014	InitialValueAssigned	LastMenstrualPeriod
seun	3/11/2014	InitialValueAssigned	Physician
seun	3/11/2014	InitialValueAssigned	ExpectedDateOfDelivery

Deliveries

Navigation
- Administration
  - My Details
  - Role
  - User
- Registry
- Consultancies
- Nursing
- Investigations
- Pharmacies
- Obstetrics & Gynaecology
  - Antenatals
  - APGARScore
  - Deliveries
  - Schedules
- Set Up
- Reports

Sunday Umana - Antenatal | Deliveries | Schedules | Schedules | Deliveries

Deliveries

Blood Loss:		Birth Weight:	
Physician:		Perinum State:	
Booked:		Placental Membrane:	
Where Booked:		Placental Weight:	
Parity:		Cord Length:	
Children Alive:		Foetal Abnormality:	
Children Dead:		Fresh Still Birth:	
Pregnancy Age:		Macerated Still Birth:	
Rupture Of Membrane:		Treatment Code:	
Diagnosis Code:		Attendant Nurse:	
Delivery Time:		Booking Type:	
Delivery Type:		Episiotomy:	
Child Sex:		Complications:	
Circumference Head:		Chest Circumference:	
Birth Length:		Delivery Date:	

Notes:

Audit Trail

User Name	Modified On	Operation Type	Property Name

Home | View | Tools

Navigation
- Administration
  - My Details
  - Role
  - User
- Registry
- Consultancies
- Nursing
- Investigations
- Pharmacies
- Obstetrics & Gynaecology
  - Antenatals
  - APGARScore
  - Deliveries
  - Schedules
- Set Up
- Reports

Sunday Umana - Antenatal | Deliveries | Schedules | Schedules | Deliveries

Schedules

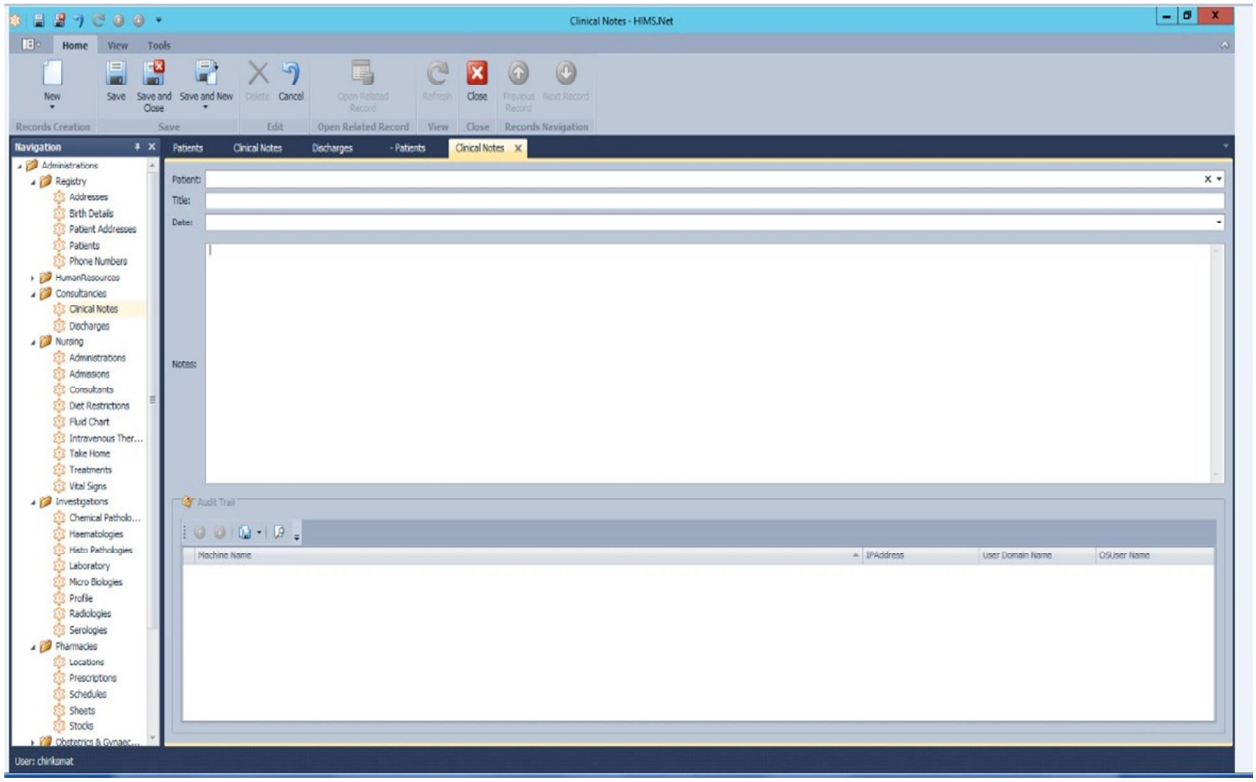
Patient Name:		Weight:	
Fondus Height:		Heamoglobin:	
Presentation And Positioning:		Blood Pressure:	
Foetal Heart:		Treatment Code:	
Oedema:		Date Visited:	
Urine:			

Notes:

Audit Trail

User Name	Modified On	Operation Type	Property Name

Consultancy Clinical Notes



Appendix B

Program Codes

Consultancy Clinical Notes

```
using System; using System.Linq; using
System.Text; using DevExpress.Xpo;
using DevExpress.ExpressApp; using
System.ComponentModel; using
DevExpress.ExpressApp.DC; using
DevExpress.Data.Filtering; using
DevExpress.Persistent.Base; using
System.Collections.Generic; using
DevExpress.ExpressApp.Model; using
DevExpress.Persistent.BaseImpl; using
DevExpress.Persistent.Validation;
using HIMS.Module.SetUp; using
HIMS.Module.Registry;

namespace HIMS.Module.Consultancy
{

    [DefaultClassOptions
```

```
    [NavigationItem("Consultancies")]
public class ClinicalNotes : BaseObject
    {
        public ClinicalNotes(Session session)
            : base(session)
        {
        }
        public override void AfterConstruction()
        {
            base.AfterConstruction();
        }

        // Fields...
        private Patients _Patient;
private DateTime _Date;         private
string _Title;                 private
string _Notes;
         private XPCollection<AuditDataItemPersistent> _auditTrail =
null;
        [Association("Patients-ClinicalNotes")]
public Patients  Patient
        {
get
{
                return _Patient;
            }
            set
            {
                SetPropertyValue("Patient", ref _Patient, value);

}
        public string  Title
        {
get
{
                return _Title;
            }
set
            {
                SetPropertyValue("Title", ref _Title, value);

        }
        [Size(4026)]
public string Notes
        {
get
{
                return _Notes;
            }
set
            {
                SetPropertyValue("Notes", ref _Notes, value);

        }
        public DateTime Date
        {
get
{
                return _Date;
            }
set
```

```
            {
                SetPropertyValue("Date", ref _Date, value);

}
        public XPCollection<AuditDataItemPersistent> AuditTrail
        {
get
{
                if (_auditTrail == null)
                    _auditTrail = AuditedObjectWeakReference.GetAuditTrail(Session, this);
                return _auditTrail;



    } }
```

Discharges

```
using System; using System.Linq; using
System.Text; using DevExpress.Xpo;
using DevExpress.ExpressApp; using
System.ComponentModel; using
DevExpress.ExpressApp.DC; using
DevExpress.Data.Filtering; using
DevExpress.Persistent.Base; using
System.Collections.Generic; using
DevExpress.ExpressApp.Model; using
DevExpress.Persistent.BaseImpl; using
DevExpress.Persistent.Validation;
using HIMS.Module.SetUp; using
HIMS.Module.Registry;

namespace HIMS.Module.Consultancy
{
    [DefaultClassOptions]
    [NavigationItem("Consultancies")]
public class Discharges : BaseObject
    {
        public Discharges(Session session)
            : base(session)
        {
}
        public override void AfterConstruction()
        {
            base.AfterConstruction();
        }         // Fields...         private
DischargeType _DischargeType;         private
Patients _Patient;         private DateTime
_Date;         private Doctor _DischargeDoctor;
private string _Witness;         private
ConsultantNurse _DischargeNurse;         private
string _Notes;

        private XPCollection<AuditDataItemPersistent> _auditTrail = null;
        [Association("Patients-Discharges")]
public Patients Patient
        {
get
```

```
                return _Patient;
            }
set
            {
                SetPropertyValue("Patient", ref _Patient, value);

        }



        public DischargeType DischargeType
        {
            get
            {
                return _DischargeType;
            }
set
            {
                SetPropertyValue("DischargeType", ref _DischargeType, value);

}
        public ConsultantNurse  DischargeNurse
        {
get
{
                return _DischargeNurse;
            }
set
            {
                SetPropertyValue("DischargeNurse", ref _DischargeNurse, value);

        }
        public string  Witness
        {
get
{
                return _Witness;
            }
set
            {
                SetPropertyValue("Witness", ref _Witness, value);

        }
        public Doctor  DischargeDoctor
        {
get
{
                return _DischargeDoctor;
            }
set
            {
                SetPropertyValue("DischargeDoctor", ref _DischargeDoctor, value);

        }

        public DateTime Date
        {
get
{
                return _Date;
            }
set
```

```
            {
                SetPropertyValue("Date", ref _Date, value);

        }
        [Size(500)]
        public string Notes
        {
get
{
                return _Notes;
            }
set
            {
                SetPropertyValue("Notes", ref _Notes, value);

}
        public XPCollection<AuditDataItemPersistent> AuditTrail
        {
get
{
                if (_auditTrail == null)
                    _auditTrail = AuditedObjectWeakReference.GetAuditTrail(Session, this);
                return _auditTrail;
            }
        }
    } }
```

Investigations

```
using System; using System.Linq; using
System.Text; using DevExpress.Xpo;
using DevExpress.ExpressApp; using
System.ComponentModel; using
DevExpress.ExpressApp.DC; using
DevExpress.Data.Filtering; using
DevExpress.Persistent.Base; using
System.Collections.Generic; using
DevExpress.ExpressApp.Model; using
DevExpress.Persistent.BaseImpl; using
DevExpress.Persistent.Validation;
using HIMS.Module.SetUp;

namespace HIMS.Module.Investigations
{
    [NavigationItem("Investigations")]
    [DefaultClassOptions]
    public class ChemicalPathologies : BaseObject
    {
        public ChemicalPathologies(Session session)
            : base(session)
        {

        }
 public override void AfterConstruction() {
            base.AfterConstruction();
```

```
        // Fields...         private
Diagnosis _Diagnosis;         private
PatientName _PatientName;         private
DateTime _SpecimenDate;         private
DateTime _RecievedDate;         private
LabNo _LabNo;
        private ChiefLabScientist _ChiefLabScientist;
private LabScientist _LabScientist;         private
Pathologist _Pathologist;         private Doctor
_Consultant;         private string _Specimen;
private string _ClinicalDetails;         private
string _SpecimenCondition;         private string
_RecievedTime;         private string _SpecimenTime;
private string _Notes;

        private XPCollection<AuditDataItemPersistent> _auditTrail = null;

        public PatientName PatientName
        {
get
{
                return _PatientName;
            }
set
            {
                SetPropertyValue("PatientName", ref _PatientName, value);

        }
        public DateTime SpecimenDate
        {
get
{
                return _SpecimenDate;
            }
set
            {
                SetPropertyValue("SpecimenDate", ref _SpecimenDate, value);

        }
        public string  SpecimenTime
        {
get
{
                return _SpecimenTime;
            }
set
            {
                SetPropertyValue("SpecimenTime", ref _SpecimenTime, value);

        }

        public DateTime  RecievedDate
        {
get
{
                return _RecievedDate;
            }
set
            {
                SetPropertyValue("RecievedDate", ref _RecievedDate, value);
```

```
        }
        public string  RecievedTime
        {
get
{
                return _RecievedTime;
            }
set
            {
                SetPropertyValue("RecievedTime", ref _RecievedTime, value);

        public string  SpecimenCondition
        {
get
{
                return _SpecimenCondition;
            }
set
            {
                SetPropertyValue("SpecimenCondition", ref _SpecimenCondition, value);

        public Diagnosis Diagnosis
        {
get
{
                return _Diagnosis;
            }
set
            {
                SetPropertyValue("Diagnosis", ref _Diagnosis, value);

            public string
ClinicalDetails
        {
get
{
                return _ClinicalDetails;
            }
set {
                SetPropertyValue("ClinicalDetails", ref _ClinicalDetails, value);
            }
        }
        public string  Specimen
        {
get
{
                return _Specimen;
            }
set
            {
                SetPropertyValue("Specimen", ref _Specimen, value);
            }
        }
        public Doctor Consultant
        {
get
{
                return
```

```
            }
set
            {
                SetPropertyValue("Consultant", ref _Consultant, value);

        ''
        public Pathologist Pathologist
        {
get
{
                return _Pathologist;
            }
set
            {
                SetPropertyValue("Pathologist", ref _Pathologist, value);
            }
        }          public LabScientist
LabScientist
        {
get
{
                return _LabScientist;
            }
set
            {
                SetPropertyValue("LabScientist", ref _LabScientist, value);

        ''
        public ChiefLabScientist ChiefLabScientist
        {
get
{
                return _ChiefLabScientist;
            }
            set
            {
                SetPropertyValue("ChiefLabScientist", ref _ChiefLabScientist, value);

        ''
        public LabNo LabNo
        {
get
{
                return _LabNo;
            }
set
            {
                SetPropertyValue("LabNo", ref _LabNo, value);

        ''
[Size(500)]         public
string Notes
        {
get
{
                return _Notes;
            }
set
            {
                SetPropertyValue("Notes", ref _Notes, value);
```

```
        }
        public XPCollection<AuditDataItemPersistent> AuditTrail
        {
get
{
                if (_auditTrail == null)
                    _auditTrail = AuditedObjectWeakReference.GetAuditTrail(Session, this);
                return _auditTrail;
            }

        }

    } }
```

Heamatology

```
using System; using System.Linq;
using System.Text; using
DevExpress.Xpo; using
DevExpress.ExpressApp; using
System.ComponentModel; using
DevExpress.ExpressApp.DC; using
DevExpress.Data.Filtering; using
DevExpress.Persistent.Base;
using
System.Collections.Generic;
using
DevExpress.ExpressApp.Model;
using
DevExpress.Persistent.BaseImpl;
using
DevExpress.Persistent.Validation
; using HIMS.Module.SetUp;

namespace HIMS.Module.Investigations
{
    [NavigationItem("Investigations")]
    [DefaultClassOptions]
    public class Haematologies : BaseObject
    {
        public Haematologies(Session session)
            : base(session)
        {

        }
        public override void AfterConstruction()
        {
            base.AfterConstruction();

        }

        // Fields...         private Diagnosis
_Diagnosis;         private PatientName _PatientName;
private DateTime _RecievedDate;         private DateTime
_ReportedDate;         private DateTime
_SpeciemenCollectionDate;         private
ChiefLabScientist _ChiefLabScientist;         private
LabScientist _LabScientist;         private Haematologist
_Haematologist;         private string
```

```
_SpeciemenCollection;         private RequiredExamination
_RequiredExamination;         private string
_SpecimenNature;         private string _ReportedTime;
private LabNo _LabNo;         private string _Notes;
         private XPCollection<AuditDataItemPersistent> _auditTrail =
null;

        public PatientName PatientName
        {
get
{
                return _PatientName;
            }
set
            {
                SetPropertyValue("PatientName", ref _PatientName, value);

            public
LabNo LabNo
        {
            get
            {
                return _LabNo;
            }
set
            {
                SetPropertyValue("LabNo", ref _LabNo, value);
            }
        }

        public DateTime RecievedDate
        {
get
{
                return _RecievedDate;
            }
set
            {
                SetPropertyValue("RecievedDate", ref _RecievedDate, value);

        }

        public DateTime ReportedDate
        {
get
{
                return _ReportedDate;
            }
set
            {
                SetPropertyValue("ReportedDate", ref _ReportedDate, value);

        }
        public string ReportedTime
        {
get
{
                return _ReportedTime;
            }
set
```

```
                SetPropertyValue("ReportedTime", ref _ReportedTime, value);

        }          public string
SpecimenNature
        {
get
{
                return _SpecimenNature;
            }
set
            {
                SetPropertyValue("SpecimenNature", ref _SpecimenNature, value);

        }
        public RequiredExamination RequiredExamination
        {
get
{
                return _RequiredExamination;
            }
set
            {
                SetPropertyValue("RequiredExamination", ref _RequiredExamination, value);

        }
        public string SpeciemenCollection
        {
get
{
                return _SpeciemenCollection;
            }
set
            {
                SetPropertyValue("SpeciemenCollection", ref _SpeciemenCollection, value);

        }

        public DateTime SpeciemenCollectionDate
        {
get
{
                return _SpeciemenCollectionDate;
            }
set
            {
                SetPropertyValue("SpeciemenCollectionDate", ref _SpeciemenCollectionDate,
value);             }
        }
          public Diagnosis
Diagnosis
        {
get
{
                return _Diagnosis;
            }
set
            {
                SetPropertyValue("Diagnosis", ref _Diagnosis, value);

        }
        public Haematologist
```

```
        {
get
{
                return _Haematologist;
            }
            set
            {
                SetPropertyValue("Haematologist", ref _Haematologist, value);

        ''
        public LabScientist LabScientist
        {
get
{
                return _LabScientist;
            }
set
            {
                SetPropertyValue("LabScientist", ref _LabScientist, value);

        ''
        public ChiefLabScientist ChiefLabScientist
        {
get
{
                return _ChiefLabScientist;
            }
set
            {
                SetPropertyValue("ChiefLabScientist", ref _ChiefLabScientist, value);

        ''
[Size(500)]         public
string Notes
        {
get
{
                return _Notes;
            }
set
            {
                SetPropertyValue("Notes", ref _Notes, value);

        ''          public XPCollection<AuditDataItemPersistent>
AuditTrail
        {
get
{
                if (_auditTrail == null)
                    _auditTrail = AuditedObjectWeakReference.GetAuditTrail(Session, this);
                return _auditTrail;


        ''
```

Histopathology

```
using System; using System.Linq; using
System.Text; using DevExpress.Xpo;
using DevExpress.ExpressApp; using
System.ComponentModel; using
DevExpress.ExpressApp.DC; using
DevExpress.Data.Filtering; using
DevExpress.Persistent.Base; using
System.Collections.Generic; using
DevExpress.ExpressApp.Model; using
DevExpress.Persistent.BaseImpl; using
DevExpress.Persistent.Validation;
using HIMS.Module.SetUp;

namespace HIMS.Module.Investigations
{
    [NavigationItem("Investigations")]
    [DefaultClassOptions]
    public class HistoPathologies : BaseObject
    {
        public HistoPathologies(Session session)
            : base(session)
        {

        }
        public override void AfterConstruction()
        {
            base.AfterConstruction();

        }

        // Fields...         private
ChiefLabScientist _ChiefLabScientist;         private
LabScientist _LabScientist;         private LabNo
_LabNo;         private PatientName _PatientName;
private DateTime _DateRecieved;         private
DateTime _DateIssued;         private Doctor
_Consultant;         private string _Type;
private string _NoOfBlocks;         private string
_NoOfPieces;         private string _StateNo;
private string _ClinicalDetails;         private
string _AnyPrevious;         private
RequiredExamination _RequiredExam;         private
Diagnosis _Diagnosis;         private string _Notes;
         private XPCollection<AuditDataItemPersistent> _auditTrail =
null;

        public PatientName PatientName
        {
get
{
                return _PatientName;
            }
set
            {
                SetPropertyValue("PatientName", ref _PatientName, value);

        }
```

```
        public LabNo LabNo
        {
get
{
                return _LabNo;
            }
set
            {
                SetPropertyValue("LabNo", ref _LabNo, value);

        ''

        public LabScientist LabScientist
        {
get
{
                return _LabScientist;
            }
set
            {
                SetPropertyValue("LabScientist", ref _LabScientist, value);

        ''

        public ChiefLabScientist ChiefLabScientist
        {
get
{
                return _ChiefLabScientist;
            }
set
            {
                SetPropertyValue("ChiefLabScientist", ref _ChiefLabScientist, value);

        ''          public
Diagnosis Diagnosis
        {
get
{
                return _Diagnosis;
            }
set             {
SetPropertyValue(
"Diagnosis", ref
_Diagnosis,
value);

        ''
        public RequiredExamination RequiredExam
        {
get
{
                return _RequiredExam;
            }
set
            {
                SetPropertyValue("RequiredExam", ref _RequiredExam, value);
            }
        }
        public string
```

```
        {
get
{
                return _AnyPrevious;
            }
set
            {
                SetPropertyValue("AnyPrevious", ref _AnyPrevious, value);

        ''
        public string ClinicalDetails
        {
get
{
                return _ClinicalDetails;
            }
set
            {
                SetPropertyValue("ClinicalDetails", ref _ClinicalDetails, value);

        ''
        public string StateNo
        {
get
{
                return _StateNo;
            }
set
            {
                SetPropertyValue("StateNo", ref _StateNo, value);

        ''

        public DateTime DateRecieved
        {
get
{
                return _DateRecieved;
            }
set
            {
                SetPropertyValue("DateRecieved", ref _DateRecieved, value);

        ''

        public DateTime DateIssued
        {
get
{
                return _DateIssued;
            }
set
            {
                SetPropertyValue("DateIssued", ref _DateIssued, value);

        ''
        public string  NoOfPieces
        {
get
{
                return
```

```
            }
set
            {
                SetPropertyValue("NoOfPieces", ref _NoOfPieces, value);

        ''
        ''
        public string  NoOfBlocks
        {
get
{
                return _NoOfBlocks;
            }
set
            {
                SetPropertyValue("NoOfBlocks", ref _NoOfBlocks, value);

        ''
        ''
        public string  Type
        {
get
{
                return _Type;
            }
set
            {
                SetPropertyValue("Type", ref _Type, value);

        ''
        ''
        public Doctor  Consultant
        {
            get
{
                return _Consultant;
            }
set
            {
                SetPropertyValue("Consultant", ref _Consultant, value);

        ''
        ''
[Size(500)]         public
string Notes
        {
get
{
                return _Notes;
            }
set
            {
                SetPropertyValue("Notes", ref _Notes, value);

        ''
        ''
        public XPCollection<AuditDataItemPersistent> AuditTrail
        {
get
{
                if (_auditTrail == null)
                    _auditTrail = AuditedObjectWeakReference.GetAuditTrail(Session, this);
                return _auditTrail;


        ''
        ''
```

```
    } }
```

Microbiology

```
using System; using System.Linq; using
System.Text; using DevExpress.Xpo;
using DevExpress.ExpressApp; using
System.ComponentModel; using
DevExpress.ExpressApp.DC; using
DevExpress.Data.Filtering; using
DevExpress.Persistent.Base; using
System.Collections.Generic; using
DevExpress.ExpressApp.Model; using
DevExpress.Persistent.BaseImpl; using
DevExpress.Persistent.Validation;
using HIMS.Module.SetUp;

namespace HIMS.Module.Investigations {
    [NavigationItem("Investigations")]
    [DefaultClassOptions]
    public class Microbiologies : BaseObject
    {
        public Microbiologies(Session session)
            : base(session)
        {

        }
        public override void AfterConstruction()
        {
            base.AfterConstruction();

        }

        // Fields...
        private Diagnosis _Diagnosis;
private PatientName _PatientName;
private DateTime _Date;
        private ChiefLabScientist _ChiefLabScientist;
private LabScientist _LabScientist;         private
string _IsolateStage;         private string
_Antibiogram;         private string _Type;
private LabNo _LabNo;         private string
_UnitCode;         private string _Notes;

        private XPCollection<AuditDataItemPersistent> _auditTrail = null;

        public PatientName PatientName
        {
get
{
                return _PatientName;
            }
set
            {
                SetPropertyValue("PatientName", ref _PatientName, value);

        }
        public string
```

```
        {
get
{
                return _UnitCode;
            }
set
            {
                SetPropertyValue("UnitCode", ref _UnitCode, value);

        ''
        public LabNo LabNo
        {
get
{
                return _LabNo;
            }
set
            {
                SetPropertyValue("LabNo", ref _LabNo, value);

        ''
        public string  Type
        {
get
{
                return _Type;
            }
set
            {
                SetPropertyValue("Type", ref _Type, value);

        ''
        public string  Antibiogram
        {
get
{
                return _Antibiogram;
            }
set
            {
                SetPropertyValue("Antibiogram", ref _Antibiogram, value);

        ''
        public string  IsolateStage
        {
get
{
                return _IsolateStage;
            }
set
            {
                SetPropertyValue("IsolateStage", ref _IsolateStage, value);
}
        }

        public Diagnosis Diagnosis
        {
get
{
                return
```

```
            }
set
            {
                SetPropertyValue("Diagnosis", ref _Diagnosis, value);

        ''

        public LabScientist LabScientist
        {
get
{
                return _LabScientist;
            }
set
            {
                SetPropertyValue("LabScientist", ref _LabScientist, value);

        ''

        public DateTime  Date
        {
get
{
                return _Date;
            }
set
            {
                SetPropertyValue("Date", ref _Date, value);

        ''
        public ChiefLabScientist ChiefLabScientist
        {
get
{
                return _ChiefLabScientist;
            }
set
            {
                SetPropertyValue("ChiefLabScientist", ref _ChiefLabScientist, value);

        ''
[Size(500)]         public
string Notes
        {
get
{
                return _Notes;
            }
set
            {
                SetPropertyValue("Notes", ref _Notes, value);

        ''          public XPCollection<AuditDataItemPersistent>
AuditTrail
        {
get
{
                if (_auditTrail == null)
                    _auditTrail = AuditedObjectWeakReference.GetAuditTrail(Session, this);
                return _auditTrail;
```

```
        }
    } }
```

Administrations

```
using System; using System.Linq; using
System.Text; using DevExpress.Xpo;
using DevExpress.ExpressApp; using
System.ComponentModel; using
DevExpress.ExpressApp.DC; using
DevExpress.Data.Filtering; using
DevExpress.Persistent.Base; using
System.Collections.Generic; using
DevExpress.ExpressApp.Model; using
DevExpress.Persistent.BaseImpl; using
DevExpress.Persistent.Validation;
using HIMS.Module.SetUp;

namespace HIMS.Module.Nursing
{
    [NavigationItem("Nursing")]
[DefaultClassOptions]
    public class Administrations : BaseObject
    {
        public Administrations(Session session)
            : base(session)
        {

        }
        public override void AfterConstruction()
        {
            base.AfterConstruction();

        }

        // Fields...         private DateTime
_Date;         private Pharmacist _Pharmacist;
private Doctor _Physician;         private
ConsultantNurse _AttendantNurse;         private
string _Duration;         private Timing
_Timing;         private string _Notes;

        private XPCollection<AuditDataItemPersistent> _auditTrail = null;
         public Timing
Timing
        {
get
{
return _Timing;
            }
set
            {
                SetPropertyValue("Timing", ref _Timing, value);

        }
        public string
```

```
        {
get
{
                return _Duration;
            }
set
            {
                SetPropertyValue("Duration", ref _Duration, value);

        ///
        public ConsultantNurse AttendantNurse
        {
get
{
                return _AttendantNurse;
            }
set
            {
                SetPropertyValue("AttendantNurse", ref _AttendantNurse, value);

        ///
        public Doctor Physician
        {
get
{
                return _Physician;
            }
set
            {
                SetPropertyValue("Physician", ref _Physician, value);
            }
        }
        public Pharmacist Pharmacist
        {
get
{
                return _Pharmacist;
            }
set
            {
                SetPropertyValue("Pharmacist", ref _Pharmacist, value);

        ///

        [Size(500)]
public string Notes
{             get
{
                return _Notes;
            }
set
            {
                SetPropertyValue("Notes", ref _Notes, value);

        ///

        public DateTime  Date
        {
get
{
                return
```

```
            }
set
            {
                SetPropertyValue("Date", ref _Date, value);

        ;
        public XPCollection<AuditDataItemPersistent> AuditTrail
        {
get
{
                if (_auditTrail == null)
                    _auditTrail = AuditedObjectWeakReference.GetAuditTrail(Session, this);
                return _auditTrail;
            }

        }

    } }
```

Admissions

```
using System; using System.Linq; using
System.Text; using DevExpress.Xpo;
using DevExpress.ExpressApp; using
System.ComponentModel; using
DevExpress.ExpressApp.DC; using
DevExpress.Data.Filtering; using
DevExpress.Persistent.Base; using
System.Collections.Generic; using
DevExpress.ExpressApp.Model; using
DevExpress.Persistent.BaseImpl; using
DevExpress.Persistent.Validation;
using HIMS.Module.SetUp;



namespace HIMS.Module.Nursing
{

    [DefaultClassOptions]
[NavigationItem("Nursing")]     public
class Admissions : BaseObject
    {
        public Admissions(Session session)
            : base(session)
        {

        }
        public override void AfterConstruction()
        {
            base.AfterConstruction();

        }
        // Fields...
private DateTime _Date;
private string _Type;
private string _Notes;
private string _BedNo;
private Doctor
```

```
        private ConsultantNurse _AttendantNurse;

        private XPCollection<AuditDataItemPersistent> _auditTrail = null;

        public ConsultantNurse AttendantNurse
        {
get
{
                return _AttendantNurse;
            }
set
            {
                SetPropertyValue("AttendantNurse", ref _AttendantNurse, value);

          ;           public Doctor
Physician
        {
get
{
                return _Physician;
            }
set
            {
                SetPropertyValue("Physician", ref _Physician, value);

        ;
        public string  BedNo
        {
get
{
                return _BedNo;
            }
set
            {
                SetPropertyValue("BedNo", ref _BedNo, value);

        ;
        public string Type
        {
get
{
                return _Type;
            }
set
            {
                SetPropertyValue("Type", ref _Type, value);

        ;
[Size(500)]         public
string  Notes
        {
get
{
                return _Notes;
            }
set
            {
                SetPropertyValue("Notes", ref _Notes, value);

        ;
```

```
        public DateTime  Date
        {
get
{
                return _Date;
            }
set
            {
                SetPropertyValue("Date", ref _Date, value);


        public XPCollection<AuditDataItemPersistent> AuditTrail
        {
get
{
                if (_auditTrail == null)
                    _auditTrail = AuditedObjectWeakReference.GetAuditTrail(Session, this);
                return _auditTrail;



    } }
```

Treatments

```
using System; using System.Linq; using
System.Text; using DevExpress.Xpo;
using DevExpress.ExpressApp; using
System.ComponentModel; using
DevExpress.ExpressApp.DC; using
DevExpress.Data.Filtering; using
DevExpress.Persistent.Base; using
System.Collections.Generic; using
DevExpress.ExpressApp.Model; using
DevExpress.Persistent.BaseImpl; using
DevExpress.Persistent.Validation;

namespace HIMS.Module.Nursing
{

    [DefaultClassOptions
|NavigationItem("Nursing")]     public
class Treatments : BaseObject
    {
        public Treatments(Session session)
            : base(session)
        {

        }
        public override void AfterConstruction()
        {
            base.AfterConstruction();

        }

        // Fields...         private
string _PatientName;         private
string _Duration;         private string
```

```
_Frequency;         private string
_AttendantNurse;         private string
_Pharmacist;         private string
_Instruction;         private string
_Physician;         private string
_Route;         private string _Dosage;
private string _DrugDescription;
private string _DrugType;
private string _Notes;

        private XPCollection<AuditDataItemPersistent> _auditTrail = null;

        public string  PatientName
        {
get
{
                return _PatientName;
            }
set
            {
                SetPropertyValue("PatientName", ref _PatientName, value);

        public string  DrugType
        {
get
{
                return _DrugType;
            }
set
            {
                SetPropertyValue("DrugType", ref _DrugType, value);

        public string  DrugDescription
        {
get
{
                return _DrugDescription;
            }
set
            {
                SetPropertyValue("DrugDescription", ref _DrugDescription, value);

        public string  Dosage
        {
get
{
                return _Dosage;
            }
set
            {
                SetPropertyValue("Dosage", ref _Dosage, value);

           public
string  Route
        {
get
{
                return
```

```
            }
set
            {
                SetPropertyValue("Route", ref _Route, value);

        ''

        public string  Physician
        {
get
{
                return _Physician;
            }
set
            {
                SetPropertyValue("Physician", ref _Physician, value);

        ''
        public string  Instruction
        {
get
{
                return _Instruction;
            }
set
            {
                SetPropertyValue("Instruction", ref _Instruction, value);

        ''
        public string  Pharmacist
        {
get
{
                return _Pharmacist;
            }
set
            {
                SetPropertyValue("Pharmacist", ref _Pharmacist, value);

        ''
        public string  AttendantNurse
        {
get
{
                return _AttendantNurse;
            }
set
            {
                SetPropertyValue("AttendantNurse", ref _AttendantNurse, value);

        ''          public string
Frequency
        {
get
{
                return _Frequency;
            }
set
            {
                SetPropertyValue("Frequency", ref _Frequency, value);
```

```
        }
        public string  Duration
        {
get
{
                return _Duration;
            }
set
            {
                SetPropertyValue("Duration", ref _Duration, value);


[Size(500)]         public
string Notes
        {
get
{
                return _Notes;
            }
set
            {
                SetPropertyValue("Notes", ref _Notes, value);

}
        public XPCollection<AuditDataItemPersistent> AuditTrail
        {
get
{
                if (_auditTrail == null)
                    _auditTrail = AuditedObjectWeakReference.GetAuditTrail(Session, this);
                return _auditTrail;
            }

        }
    } }
```

Obstetrics and Gynecology Antenatals

```
using System; using System.Linq; using
System.Text; using DevExpress.Xpo;
using DevExpress.ExpressApp; using
System.ComponentModel; using
DevExpress.ExpressApp.DC; using
DevExpress.Data.Filtering; using
DevExpress.Persistent.Base; using
System.Collections.Generic; using
DevExpress.ExpressApp.Model; using
DevExpress.Persistent.BaseImpl; using
DevExpress.Persistent.Validation;

namespace HIMS.Module.OandG
{

    [DefaultClassOptions
[NavigationItem("OandG")]     public
class Antenatals : BaseObject
    {
        public Antenatals(Session session)
            : base
```

```
        {

        }
        public override void AfterConstruction()
        {
            base.AfterConstruction();

        }

        // Fields...         private string
_PatientName;         private DateTime
_LastMenstrualPeriod;         private DateTime
_ExpectedDateOfDelivery;         private DateTime
_Date;         private string _Physician;
private string _AttendantNurse;         private
string _General;         private string _Sickle;
private string _Feet;         private string
_Nipples;         private string _Pelvis;
private string _Weight;         private string
_Height;         private string _BloodGroup;
private string _Gravida;         private string
_Notes;

        private XPCollection<AuditDataItemPersistent> _auditTrail = null;

        public string  PatientName
        {
get
{
                return _PatientName;
            }
set
            {
                SetPropertyValue("PatientName", ref _PatientName, value);

        }           public DateTime
LastMenstrualPeriod
        {
get
{
                return _LastMenstrualPeriod;
            }
set
            {
                SetPropertyValue("LastMenstrualPeriod", ref _LastMenstrualPeriod, value);

        }
        public string Gravida
        {
get
{
                return _Gravida;
            }
set
            {
                SetPropertyValue("Gravida", ref _Gravida, value);

        }
        public string
```

```
        {
get
{
                return _BloodGroup;
            }
set
            {
                SetPropertyValue("BloodGroup", ref _BloodGroup, value);

        ///
        public string Height
        {
get
{
                return _Height;
            }
set
            {
                SetPropertyValue("Height", ref _Height, value);

        ///
        public string Weight
        {
get
{
                return _Weight;
            }
set
            {
                SetPropertyValue("Weight", ref _Weight, value);

        ///           public
string Pelvis
        {
get
{
                return _Pelvis;
            }
            set
            {
                SetPropertyValue("Pelvis", ref _Pelvis, value);

        ///
        public string Nipples
        {
get
{
                return _Nipples;
            }
set
            {
                SetPropertyValue("Nipples", ref _Nipples, value);

        ///
        public string Feet
        {
get
{
                return _Feet;
            }
set
```

```
            {
                SetPropertyValue("Feet", ref _Feet, value);

        public string Sickle
        {
get
{
                return _Sickle;
            }
set
            {
                SetPropertyValue("Sickle", ref _Sickle, value);

        public string General
        {
get
{
                return _General;
            }
set
            {
                SetPropertyValue("General", ref _General, value);

                   public string
AttendantNurse
        {
get
            {
                return _AttendantNurse;
            }
set
            {
                SetPropertyValue("AttendantNurse", ref _AttendantNurse, value);

        public string Physician
        {
get
{
                return _Physician;
            }
set
            {
                SetPropertyValue("Physician", ref _Physician, value);
            }
        }

        public DateTime ExpectedDateOfDelivery
        {
get
{
                return _ExpectedDateOfDelivery;
            }
set
            {
                SetPropertyValue("ExpectedDateOfDelivery", ref _ExpectedDateOfDelivery,
value);             }
```

```
        [Size(500)]
public string Notes
        {
get
{
                return _Notes;
            }
set
            {
                SetPropertyValue("Notes", ref _Notes, value);
}
        }
        public DateTime  Date
        {
get
{
                return _Date;
            }
set
            {
                SetPropertyValue("Date", ref _Date, value);
}
        }
        public XPCollection<AuditDataItemPersistent> AuditTrail
        {
get
{
                if (_auditTrail == null)
                    _auditTrail = AuditedObjectWeakReference.GetAuditTrail(Session, this);
                return _auditTrail;
            }

        }
    } }
```

Antenatals

```
using System; using System.Linq; using
System.Text; using DevExpress.Xpo;
using DevExpress.ExpressApp; using
System.ComponentModel; using
DevExpress.ExpressApp.DC; using
DevExpress.Data.Filtering; using
DevExpress.Persistent.Base; using
System.Collections.Generic; using
DevExpress.ExpressApp.Model; using
DevExpress.Persistent.BaseImpl; using
DevExpress.Persistent.Validation;

namespace HIMS.Module.OandG
{

    [DefaultClassOptions
|NavigationItem("OandG")]
    public class Deliveries : BaseObject
    {
        public Deliveries(Session session)
            : base(session)
```

```
        }          public override void
AfterConstruction()
        {
            base.AfterConstruction();

        }

        // Fields...
        private DateTime _DeliveryDate;
private string _ChestCircumference;
private string _Complications;
private string _Episiotomy;         private
string _BookingType;
        private string _AttendantNurse;
private string _TreatmentCode;
private string _MaceratedStillBirth;
private string _FreshStillBirth;
private string _FoetalAbnormality;
private string _CordLength;         private
string _PlacentalWeight;         private
string _PlacentalMembrane;         private
string _PerinumState;         private string
_BirthWeight;         private string
_BirthLength;         private string
_CircumferenceHead;         private string
_ChildSex;         private string
_DeliveryType;         private string
_DeliveryTime;         private string
_DiagnosisCode;         private string
_RuptureOfMembrane;         private string
_PregnancyAge;         private string
_ChildrenDead;         private string
_ChildrenAlive;         private string
_Parity;         private string
_WhereBooked;         private string
_Booked;         private string _Physician;
private string _BloodLoss;         private
string _Notes;

        private XPCollection<AuditDataItemPersistent> _auditTrail = null;

        public string  BloodLoss
        {
get
{
                return _BloodLoss;
            }
set
            {
                SetPropertyValue("BloodLoss", ref _BloodLoss, value);

            }          public string
Physician
        {
get
{
                return _Physician;
            }
set
```

```
            {
                SetPropertyValue("Physician", ref _Physician, value);

        ˜
        public string  Booked
        {
get
{
                return _Booked;
            }
set
            {
                SetPropertyValue("Booked", ref _Booked, value);

        ˜
        public string  WhereBooked
        {
get
{
                return _WhereBooked;
            }
set
            {
                SetPropertyValue("WhereBooked", ref _WhereBooked, value);

        ˜
        public string  Parity
        {
get
{
                return _Parity;
            }
set
            {
                SetPropertyValue("Parity", ref _Parity, value);

        ˜
        public string  ChildrenAlive
        {
get
{
                return _ChildrenAlive;
            }
set
            {
                SetPropertyValue("ChildrenAlive", ref _ChildrenAlive, value);

        ˜
        public string  ChildrenDead
        {
get
{
                return _ChildrenDead;
            }
set
            {
                SetPropertyValue("ChildrenDead", ref _ChildrenDead, value);

        ˜
        public string
```

```
        {
get
{
                return _PregnancyAge;
            }
set
            {
                SetPropertyValue("PregnancyAge", ref _PregnancyAge, value);

        ///
        public string  RuptureOfMembrane
        {
get
{
                return _RuptureOfMembrane;
            }
set
            {
                SetPropertyValue("RuptureOfMembrane", ref _RuptureOfMembrane, value);

        ///
        public string  DiagnosisCode
        {
get
{
                return _DiagnosisCode;
            }
set
            {
                SetPropertyValue("DiagnosisCode", ref _DiagnosisCode, value);

        ///
        public string  DeliveryTime
        {
get
{
                return _DeliveryTime;
            }
set
            {
                SetPropertyValue("DeliveryTime", ref _DeliveryTime, value);

        ///
        public string  DeliveryType
        {
get
{
                return _DeliveryType;
            }
set
            {
                SetPropertyValue("DeliveryType", ref _DeliveryType, value);

        ///
        public string  ChildSex
        {
get
{
                return _ChildSex;
            }
set
```

```
            {
                SetPropertyValue("ChildSex", ref _ChildSex, value);

        ''
        public string  CircumferenceHead
        {
get
{
                return _CircumferenceHead;
            }
set
            {
                SetPropertyValue("CircumferenceHead", ref _CircumferenceHead, value);

        ''
        public string  BirthLength
        {
get
{
                return _BirthLength;
            }
set
            {
                SetPropertyValue("BirthLength", ref _BirthLength, value);

        ''
        public string  BirthWeight
        {
get
{
                return _BirthWeight;
            }
set
            {
                SetPropertyValue("BirthWeight", ref _BirthWeight, value);

        ''
        public string  PerinumState
        {
get
{
                return _PerinumState;
            }
set
            {
                SetPropertyValue("PerinumState", ref _PerinumState, value);

        ''


        public string  PlacentalMembrane
        {
get
{
                return _PlacentalMembrane;
            }
set
            {
                SetPropertyValue("PlacentalMembrane", ref _PlacentalMembrane, value);

        ''
        public string
```

```
        {
get
{
                return _PlacentalWeight;
            }
set
            {
                SetPropertyValue("PlacentalWeight", ref _PlacentalWeight, value);

        ///
        public string  CordLength
        {
get
{
                return _CordLength;
            }
set
            {
                SetPropertyValue("CordLength", ref _CordLength, value);

        ///
        public string  FoetalAbnormality
        {
get
{
                return _FoetalAbnormality;
            }
set
            {
                SetPropertyValue("FoetalAbnormality", ref _FoetalAbnormality, value);
            }
        }          public string
FreshStillBirth
        {
get
{
                return _FreshStillBirth;
            }
set
            {
                SetPropertyValue("FreshStillBirth", ref _FreshStillBirth, value);

        ///
        public string  MaceratedStillBirth
        {
get
{
                return _MaceratedStillBirth;
            }
set
            {
                SetPropertyValue("MaceratedStillBirth", ref _MaceratedStillBirth, value);

        ///
        public string  TreatmentCode
        {
get
{
                return _TreatmentCode;
            }
set
```

```
            {
                SetPropertyValue("TreatmentCode", ref _TreatmentCode, value);

        ///



        public string  AttendantNurse
        {
get
{
                return _AttendantNurse;
            }
set
            {
                SetPropertyValue("AttendantNurse", ref _AttendantNurse, value);

        ///           public string
BookingType
        {
get
{
                return _BookingType;
            }
set
            {
                SetPropertyValue("BookingType", ref _BookingType, value);

        ///
        public string  Episiotomy
        {
get
{
                return _Episiotomy;
            }
            set
            {
                SetPropertyValue("Episiotomy", ref _Episiotomy, value);

        ///
        public string  Complications
        {
get
{
                return _Complications;
            }
set
            {
                SetPropertyValue("Complications", ref _Complications, value);

        ///
        public string  ChestCircumference
        {
get
{
                return _ChestCircumference;
            }
set
            {
                SetPropertyValue("ChestCircumference", ref _ChestCircumference, value);

        ///
```

```
        public DateTime  DeliveryDate
        {
get
{
                return _DeliveryDate;
            }
set
            {
                SetPropertyValue("DeliveryDate", ref _DeliveryDate, value);


        [Size(500)]
public string Notes
        {
get
{
                return _Notes;
            }
set
            {
                SetPropertyValue("Notes", ref _Notes, value);
}         }
        public XPCollection<AuditDataItemPersistent> AuditTrail
        {
get {
                if (_auditTrail == null)
                    _auditTrail = AuditedObjectWeakReference.GetAuditTrail(Session, this);
                return _auditTrail;



    } }
```

Pharmacy Prescriptions

```
using System; using System.Linq; using
System.Text; using DevExpress.Xpo;
using DevExpress.ExpressApp; using
System.ComponentModel; using
DevExpress.ExpressApp.DC; using
DevExpress.Data.Filtering; using
DevExpress.Persistent.Base; using
System.Collections.Generic; using
DevExpress.ExpressApp.Model; using
DevExpress.Persistent.BaseImpl; using
DevExpress.Persistent.Validation;
using HIMS.Module.SetUp;

namespace HIMS.Module.Pharmacies
{

    [DefaultClassOptions
[NavigationItem("Pharmacies")]     public
class Prescriptions : BaseObject
    {
        public Prescriptions(Session session)
            : base(session)
```

```
        }          public override void
AfterConstruction()
        {
            base.AfterConstruction();

        }

        // Fields...
        private Pharmacist _Pharmacist;
private PatientName _PatientName;
private string _SheetCode;
private string _TotalAmount;
private string _Quantity;         private
string _Amount;
        private Duration _Duration;
        private Doctor _Doctor;
private Dosage _Dosage;         private
string _Description;         private
DrugCategory _DrugCategory;         private
string _Notes;

        private XPCollection<AuditDataItemPersistent> _auditTrail = null;

        public PatientName PatientName
        {
get
{
                return _PatientName;
            }
set
            {
                SetPropertyValue("PatientName", ref _PatientName, value);

        ⋮
        public DrugCategory DrugCategory
        {
get
{
                return _DrugCategory;
            }
set
            {
                SetPropertyValue("DrugCategory", ref _DrugCategory, value);
            }
        }

        public Pharmacist  Pharmacist
        {
get
{
                return _Pharmacist;
            }
set
            {
                SetPropertyValue("Pharmacist", ref _Pharmacist, value);

        ⋮
        public string
```

```
        {
get
{
                return _Description;
            }
set
            {
                SetPropertyValue("Description", ref _Description, value);

        ''

        public Dosage  Dosage
        {
get
{
                return _Dosage;
            }
set
            {
                SetPropertyValue("Dosage", ref _Dosage, value);

        ''
        public Doctor  Doctor
        {
get
{
                return _Doctor;
            }
set
            {
                SetPropertyValue("Doctor", ref _Doctor, value);
            }
        }
        public Duration  Duration
        {
get
{
                return _Duration;
            }
set
            {
                SetPropertyValue("Duration", ref _Duration, value);

        ''
        public string  Amount
        {
get
{
                return _Amount;
            }
set
            {
                SetPropertyValue("Amount", ref _Amount, value);

        ''
        public string  Quantity
        {
get
{
                return
```

```
            }
set
            {
                SetPropertyValue("Quantity", ref _Quantity, value);

        ''
        public string  TotalAmount
        {
get
{
                return _TotalAmount;
            }
set
            {
                SetPropertyValue("TotalAmount", ref _TotalAmount, value);

        ''
        public string  SheetCode
        {
get
{
                return _SheetCode;
            }
set
            {
                SetPropertyValue("SheetCode", ref _SheetCode, value);

        ''
[Size(500)]         public
string Notes
        {
get
{
                return _Notes;
            }
set
            {
                SetPropertyValue("Notes", ref _Notes, value);

}
        public XPCollection<AuditDataItemPersistent> AuditTrail
        {
get
{
                if (_auditTrail == null)
                    _auditTrail = AuditedObjectWeakReference.GetAuditTrail(Session, this);
                return _auditTrail;
            }

        }

    } }
```

Stocks

```
using System; using
System.Linq; using
System.Text; using
```

```
using
DevExpress.ExpressA
pp; using
System.ComponentMod
el; using
DevExpress.ExpressA
pp.DC; using
DevExpress.Data.Fil
tering; using
DevExpress.Persiste
nt.Base; using
System.Collections.
Generic; using
DevExpress.ExpressA
pp.Model; using
DevExpress.Persiste
nt.BaseImpl; using
DevExpress.Persiste
nt.Validation;
using
HIMS.Module.SetUp;

namespace HIMS.Module.Pharmacies
{

    [DefaultClassOptions]
[NavigationItem("Pharmacies")]     public
class Stocks : BaseObject
    {
        public Stocks(Session session)
            : base(session)
        {

        }
        public override void AfterConstruction()
        {
            base.AfterConstruction();

        }

        // Fields...         private
DateTime _DateOfCollection;         private
DateTime _ExpiryDate;         private
DateTime _ManufactureDate;         private
DateTime _StockDate;         private Status
_Status;         private string
_DirectionOfUse;         private string
_Constituents;         private Dosage
_Dosage;         private SupplierCode
_SupplierCode;         private string
_CollectedBy;         private string
_PharmacyCode;         private string
_NAFDACRegNo;         private string
_BatchNo;         private string
_DiscountComputed;         private string
_DiscountPercent;         private string
_DiscountedAmount;         private string
_SellingPrice;         private string
_Quantity;         private string
_CostPrice;         private string
```

```
private DrugCategory _DrugCategory;
private DrugName _DrugName;

        private XPCollection<AuditDataItemPersistent> _auditTrail = null;
public DrugName DrugName
        {
get
{
                return _DrugName;
            }
set
            {
                SetPropertyValue("DrugName", ref _DrugName, value);
            }
        }
        public DrugCategory DrugCategory
        {
get
{
                return _DrugCategory;
            }
set
            {
                SetPropertyValue("DrugCategory", ref _DrugCategory, value);

        ''
[Size(500)]         public
string  Notes
        {
get
{
                return _Notes;
            }
set
            {
                SetPropertyValue("Notes", ref _Notes, value);

        ''
        public string  CostPrice
        {
get
{
                return _CostPrice;
            }
set
            {
                SetPropertyValue("CostPrice", ref _CostPrice, value);

        ''
        public string  Quantity
        {
get
{
                return _Quantity;
            }
set
            {
                SetPropertyValue("Quantity", ref _Quantity, value);

        ''
        public string
```

```
        {
get
{
                return _SellingPrice;
            }
set
            {
                SetPropertyValue("SellingPrice", ref _SellingPrice, value);

        ''
        public string  DiscountedAmount
        {
get
{
                return _DiscountedAmount;
            }
set
            {
                SetPropertyValue("DiscountedAmount", ref _DiscountedAmount, value);

        ''
        public string  DiscountPercent
        {
get
{
                return _DiscountPercent;
            }
set
            {
                SetPropertyValue("DiscountPercent", ref _DiscountPercent, value);

        ''
        public string  DiscountComputed
        {
get
{
                return _DiscountComputed;
            }
set
            {
                SetPropertyValue("DiscountComputed", ref _DiscountComputed, value);

        ''

        public DateTime  DateOfCollection
        {
              get
       {
                     return _DateOfCollection;
              }
              set
              {
                SetPropertyValue("DateOfCollection", ref _DateOfCollection, value);
              }
        }
        public string  BatchNo
        {
get
{
                return
```

```
            }
set
            {
                SetPropertyValue("BatchNo", ref _BatchNo, value);

        ///
        public string  NAFDACRegNo
        {
get
{
                return _NAFDACRegNo;
            }
set
            {
                SetPropertyValue("NAFDACRegNo", ref _NAFDACRegNo, value);

        ///
        public string  PharmacyCode
        {
get
{
                return _PharmacyCode;
            }
set
            {
                SetPropertyValue("PharmacyCode", ref _PharmacyCode, value);
            }
        }

        public DateTime  ExpiryDate
        {
               get           {
        return _ExpiryDate;
               }
 set
              {
                SetPropertyValue("ExpiryDate", ref _ExpiryDate, value);

        ///
        public string CollectedBy
        {
get
{
                return _CollectedBy;
            }
set
            {
                SetPropertyValue("CollectedBy", ref _CollectedBy, value);

        ///
        public SupplierCode  SupplierCode
        {
get
{
                return _SupplierCode;
            }
set
            {
                SetPropertyValue("SupplierCode", ref _SupplierCode, value);
            }
```

```
        public DateTime  ManufactureDate
        {
               get
 {
                     return _ManufactureDate;
               }
 set
              {
                SetPropertyValue("ManufactureDate", ref _ManufactureDate, value);

        ''
        public Dosage  Dosage
        {
get
{
                return _Dosage;
            }
set
            {
                SetPropertyValue("Dosage", ref _Dosage, value);

        ''
        public string  DirectionOfUse
        {
get
{
                return _DirectionOfUse;
            }
set
            {
                SetPropertyValue("DirectionOfUse", ref _DirectionOfUse, value);

        ''

        public Status Status
        {
get
{
                return _Status;
            }
set
            {
                SetPropertyValue("Status", ref _Status, value);

        ''
        public string  Constituents
        {
get
{
                return _Constituents;
            }
set
            {
                SetPropertyValue("Constituents", ref _Constituents, value);

        ''

        public DateTime  StockDate
```

```
                get
 {
                      return _StockDate;
                }
 set
               {
                 SetPropertyValue("StockDate", ref _StockDate, value);


        public XPCollection<AuditDataItemPersistent> AuditTrail
        {
get
{
                if (_auditTrail == null)
                    _auditTrail = AuditedObjectWeakReference.GetAuditTrail(Session, this);
                return _auditTrail;
            }

        }
        }

    }
```

An efficient model for healthcare management a case study of Asokoro General Hospital, FCT, Abuja. By **Garuba O.** is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License