

ReadyToShip

By

Fernando C. Cremer

Submitted to
the Faculty of the Information Technology Program
in Partial Fulfillment of the Requirements for
the Degree of Bachelor of Science
in Information Engineering Technology

University of Cincinnati
College of Applied Science

May 2007

ReadyToShip

By

Fernando C. Cremer

Submitted to
the Faculty of the Information Technology Program
in Partial Fulfillment of the Requirements
for
the Degree of Bachelor of Science
in Information Engineering Technology

© Copyright 2007 Fernando C. Cremer

The author grants to the Information Technology Program permission to reproduce and distribute copies of this document in whole or in part.

Fernando C. Cremer

Date

Robert Schlemmer, Faculty Advisor

Date

Hazem Said, Ph.D. Department Head

Date

Table of Contents

Section	Page
Table of Contents	i
List of Illustrations	iii
Abstract	v
Introduction	1
1. Problem Description	3
2. Solution Description	4
2.1 User Profiles	5
2.1.1 Administrator	5
2.1.2 Project Manager	6
2.1.3 User	6
2.2 Design Protocol	6
2.2.1 Application Structure	6
2.2.2 Database Structure	7
2.2.3 Graphics and Visual Style	8
2.2.4 Application Design and Navigation	8
2.3 Other Options	9
3. Deliverables	9
3.1 RTS Admin	9
3.2 ReadyToShip	9
3.2.1 TestTrack Import	10
3.2.2 Simple Machine Software Import	10
3.2.3 E-mail Import	10
3.2.4 Manual Incident Creation	11
3.2.5 Send Mail Capabilities	11
3.2.6 Project Administration Tools	12
4. Design and Development	12
4.1 Budget	12
4.2 Timeline	13
4.2.1 Fall Quarter Accomplishments	13
4.2.2 Winter Quarter Accomplishments	13
4.2.3 Spring Quarter Accomplishments	14
4.3 Software	14
4.4 Hardware	15

5.	Proof of Design	15
5.1	Database Configuration	15
5.2	RTS ADMIN	18
5.2.1	User Management	18
5.2.2	Project Management	20
5.2.3	User Documentation	24
5.3	ReadyToShip	24
5.3.1	Log-In Process	24
5.3.2	Tester Management	25
5.3.3	Incident Management	28
5.3.4	E-mail Management and Tracking	35
5.3.5	TestTrack Import	38
5.3.6	Simple Machine Forum Import	41
5.3.7	Reports	42
5.3.8	User Documentation	44
6.	Conclusion and Recommendations	46
6.1	Conclusion	46
6.2	Recommendations	47
7.	References	48

List of Illustrations

Figure Number	Page
Figure 1. Report an Issue Page on Seapine's website	2
Figure 2. Current Scenario	3
Figure 3. Proposed Solution	5
Figure 4. RTS Admin database	7
Figure 5. ReadyToShip Client database	8
Figure 6. Actual Budget	12
Figure 7. Proposed Timeline	13
Figure 8. Database Connection Configuration Window	17
Figure 9. Available SQL Servers in the Seapine Network	17
Figure 10. Available Databases in the Specified SQL Server	18
Figure 11. Manage User Menu Option and Toolbar Button	19
Figure 12. User List Window	19
Figure 13. Add User Window	20
Figure 14. Manage Projects menu and toolbar button	20
Figure 15. Project List Window	21
Figure 16. Add Project Window General Tab	22
Figure 17. Add Project Window Database Tab	23
Figure 18. Add Project Window Project Users Tab	23
Figure 19. RTS Admin Help Topics	24
Figure 20. Log in Window	25
Figure 21. View Testers toolbar button and menu option	25

Figure 22. Tester List Window	26
Figure 23. Details Tab in the View Tester Window	27
Figure 24. E-mails Tab in the View Tester Window	28
Figure 25. View Incident Menu Option and Toolbar Button	29
Figure 26. Incident List Window	29
Figure 27. Incident Display Filter Options	30
Figure 28. Incident Detail Window	32
Figure 29. Status Options	33
Figure 30. Enter Signature Button and Sample Signature	33
Figure 31. E-mails Tab on the Incident Detail Window	34
Figure 32. History Tab on the Incident Detail Window	35
Figure 33. Manage E-mails Window	36
Figure 34. Filter Options in the Manage E-mails Window	37
Figure 35. SQL Server in the Seapine Network	38
Figure 36. TestTrack Import Options Window	39
Figure 37. Import From TestTrack Menu Option and Toolbar Button	39
Figure 38. Import From TestTrack Progress Window	41
Figure 39. Report Options	42
Figure 40. Incident Grouped By Tester Report (Top)	43
Figure 41. Incident Grouped By Tester Report (Bottom)	44
Figure 42. ReadyToShip Index Page	45
Figure 43. Incident Help Page	46

Abstract

ReadyToShip is a windows application aimed at helping the Seapine employee that moderates Seapine's beta sites to better track interactions with testers. Seapine Software is the leading provider of quality-centric application lifecycle management (ALM) solutions for product development and IT organizations. The application is written in C# using Microsoft Visual Studio 2005 and it uses Microsoft SQL 2005 Server to store its database. The application helps the employee track of all tester interaction trough incidents. The application can import beta site tester defects and feature requests posted on the beta site web page as well as e-mails sent to a POP3 mail account. The application takes these various tester contacts and converts them to incidents. Incidents can also be manually created, to account for those incidents that may originate through a phone call. The application also provides some basic reporting tools to get some metrics on the information gathered from the testers.

ReadyToShip

Introduction

Seapine software is “a global provider of innovative application lifecycle management (ALM) solutions that manage process, change, and quality in product development and IT organizations.” (11) Some of Seapine’s product brands include:

- **QA Wizard** - Automated functional and regression testing.
- **Surround SCM** - Software configuration management.
- **TestTrack Pro** - Issue and development process management.
- **TestTrack TCM** - Test case planning and tracking.

In the fall of 2006 Seapine launched for the first time TestTrack TCM, and also released two new major upgrades for TestTrack Pro (7.6) and Surround SCM (5.0). A beta program was created for these three releases, and a new approach was taken in managing the beta site for these products. The previous approach was to contact certain customers to recruit as beta testers. The testers would agree to test the beta software and would send some feedback on the product. The problem with the old approach is that the beta testers were few and the feedback provided by them was even less (9).

For this beta program, Seapine instead launched a Web site to manage the beta sites and incorporated it to its new Web site, labs.seapine.com. Seapine sent an e-mail to all of its customers, and made an announcement on the CM Crossroads website to invite anybody to sign up and join the beta program. According to Rick Riccetti, Seapine’s

CEO and founder, this site was created to increase beta site members and also give users more ways to give Seapine valuable feedback(9). Some of the features of this site were the ability to download the latest beta release candidate, access to user forums to discuss defects, feature requests and best practices and the ability to enter a request for a feature, and to report a defect. Figure 1 below displays a similar page found on Seapine’s web site to the one used on the beta program web site where users can report an issue with any product.

Seapine Software Search

SOLUTIONS PRODUCTS SERVICES SUPPORT PARTNERS NEWS & EVENTS

Customer Support

Report an Issue

Please complete the following form to report a bug or other issue in Seapine products.

Issue Information
* Indicates required information

Product:*

Version:

Operating System:

Serial Number:

Issue Description:*

Contact Information

Name:*

Company:*

Email:*

Phone:

Overview
Upgrades
Documentation
Knowledgebase
User Forums
Request Feature
Report Issue
Look Up Issue

"I sent the request via email; I was very impressed with the speed of a response with a solution to my problem. Based on other email support, yours rocks."
Michael Hartary
Kinex Technologies Inc.

Figure 1 – Report an Issue Page on Seapine’s website

1. Problem Description

The information gathered from the beta program web site was not being stored in one single place. The list of users that register were being stored in a MySQL database, and the feature requests and defects were automatically imported to a TestTrack Pro database. Also, some beta users were still sending e-mails to the support inbox and call the support department. E-mails that came to the support inbox create a case in Seapine's Customer Relationship Management (CRM) tool. Under this scenario there are three different places where information relevant to the beta members was being stored. Figure 2 below illustrates this current scenario.

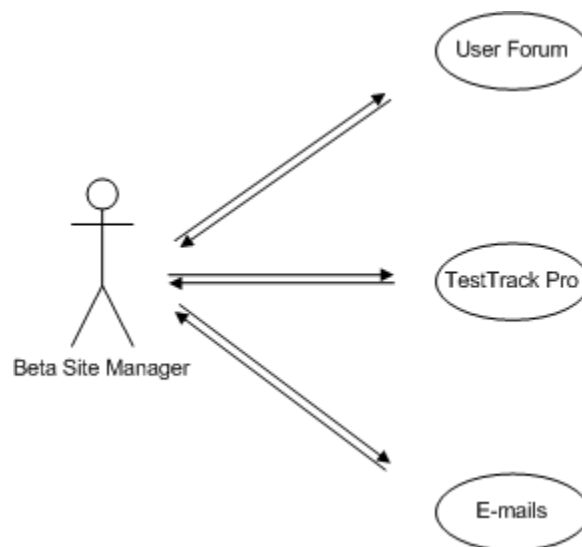


Figure 2 – Current Scenario

The purpose of every beta program is to get outside validation to build confidence, determine product readiness to ship, early learning and feedback, testimonials, quotes and success stories and to build ongoing relationships with a friendly set of customers (5). A Beta program is one of the most critical factors in the success or failure of a product (6).

Since the data was being stored in three different places, it was very difficult for the Seapine employee in charge of managing the beta tester feedback to track the various feedback submitted by the testers. The minimum number of serious testers in an exclusive beta program is about 100 (12), and the lack of a unified interface and database makes it difficult to be able to identify users that Seapine may want to contact for more exclusive beta programs. An exclusive beta program is something that Seapine is considering for the next beta release for any of its products.

The setup inhibited the ability to see the big picture and be able to determine the state of the product (8). Without a reliable way to view all information submitted by beta site members, it was difficult to be able to accurately get useful data from the beta testers. If a company does not understand the needs of its customers, how can it possibly deliver a product? (3)

2. Solution Description

A tool that integrates and brings together all data coming from these different sources into one database and one interface solves this problem. This tool imports the data from all sources, filters it and saves it to a Microsoft SQL 2005 database. This tool offers a simple to use interface that a beta program manager is be able to use to extract accurate information from the beta program.

This tool is called ReadyToShip and is similar to a light weight CRM tool that allows the beta program manager to manage all beta site tester feedback, and be able to

easily retrieve their e-mails submitted, phone calls, as well as defects and feature requests submitted via the site. Figure 3 illustrates the proposed scenario that the tool delivers.

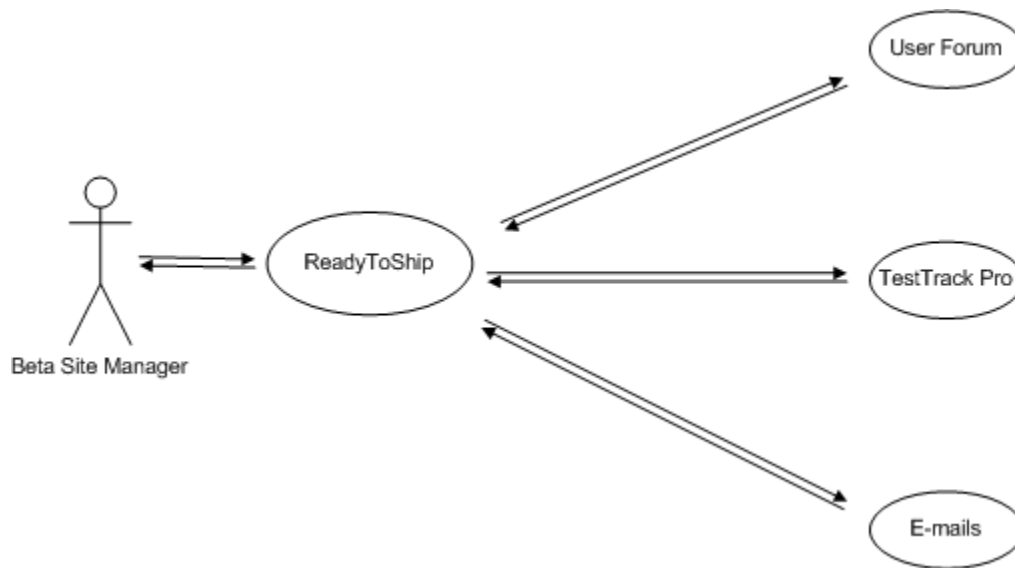


Figure 3 – Proposed Solution

2.1 User Profiles

In order to minimize a learning curve, user profiles are managed by security groups as it is done with Seapine products. There are three security groups that have access. All intended users have a medium to high IT literacy.

2.1.1 Administrator

This user has full permissions and is in charge of installing the application, sets which users have access, and establishes the SQL server database to be used.

2.1.2 Project Manager

This is the beta site manager and has full permissions within the scope of a project. This user is able to set up connections to the TestTrack and MySQL databases for importing information as well as all other project options. This security group contains most users. Beta sites are usually managed by a single member of the Technical Support team. These employees have a high level of IT literacy.

2.1.3 User

This is equivalent to a “guest” user account. This account mainly has read only permissions and is to be used only by individuals who may want to review feedback from testers. A likely candidate for this type of user would be a product manager.

2.2 Design Protocol

2.2.1 Application Structure

The tool consists of two applications. One application is RTS Admin and is used to manage the application on a global scope. This application is used to manage users that have access to the application (Beta Site Managers) and the SQL Server connection information. Projects are created using this application by providing information required to connect to the designated SQL Server and create the database. This application is used by the System Administrator, and has to be a member of the “Administrator” security group in the application in order to be able to access it.

The other application is the “ReadyToShip” client. This is the application that most users use. This is the application that actually is used to manage the incidents. Incidents are any phone calls, e-mails, or web site posts from the testers.

2.2.2 Database Structure

There are two databases. One is be the back end database for the admin tool, RTA Admin. The other database stores the actual project, including beta site user list and incidents using the ReadyToShip client. Figure 4 below represents the database for the RTS Admin application.

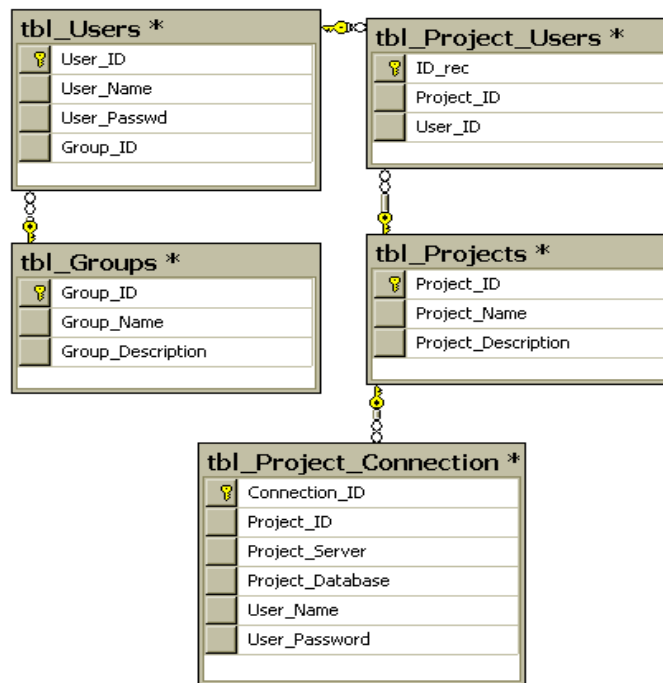


Figure 4 – RTS Admin database

Figure 5, located on the next page, shows the database structure for the ReadyToShip Client.

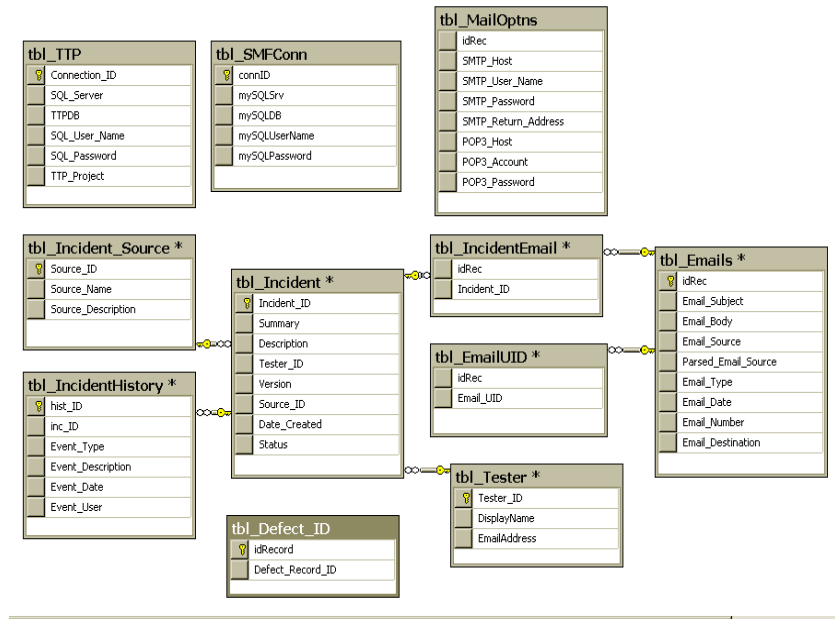


Figure 5 – ReadyToShip Client database

2.2.3 Graphics and Visual Style

To ease the learning curve for Seapine employees, ReadyToShip uses the same graphics for its buttons as the Seapine products. The applications themselves are designed using Seapine’s TestTrack Studio as a model. The colors used in the application are the standard windows application foreground and background colors.

2.2.4 Application Design and Navigation

The applications’ design and navigation are consistent with Seapine’s TestTrack Studio application. Both applications consist of a Multi Document Interface (MDI) window. The applications show data in list windows. All functions are available through menu options, and the most used functions also have buttons on the toolbar.

2.3 Other Options

There are few software packages in the market that are able to deliver this solution. However, these software packages can not import data from TestTrack Pro unless professional services are purchased or if it is customized by the software provider. One of the solutions in the market is from Centercode Solutions and the tool is called Centercode Connect. This is a web based interface and it is hosted by Centercode Solutions. The cost per project is \$3,000 (1). Creating a custom-built tool ensures that all of the needs are met and since all tools needed to create this solution are already owned by Seapine, the cost was non-existing.

3. Deliverables

There are several functions that ReadyToShip and RTS Admin provide.

3.1 RTS Admin

System administrators setup users to have access, create projects and determine the SQL server to be used. The administrator must use an SQL user account that has enough security permissions to create a database. This user account is used for all communications between the ReadyToShip client and its database.

3.2 ReadyToShip

The user tracks and manages all beta user interaction with the ReadyToShip client. ReadyToShip manages beta tester communications with incidents. An incident is a communication between a beta tester and Seapine. The application consists of modules, each specialized for each functionality to track these incidents.

3.2.1 TestTrack Import

ReadyToShip imports defects and feature requests submitted by beta testers. These are entered in the web pages and are automatically imported into the TestTrack database, which is stored in Microsoft SQL. The application accounts for previously imported records and only imports those records that are new. The application also accounts for the beta tester not existing in the ReadyToShip project database. When importing new records, it checks to see if the tester exists or not. If the tester is not found, the tester is added.

3.2.2 Simple Machine Software Import

ReadyToShip imports the list of registered testers from the MySQL database that the user forum software, Simple Machine Software (SMS), uses to store its information. The application accounts for previously imported testers and subsequent imports only import new testers.

3.2.3 E-mail Import

ReadyToShip imports e-mails from a designated POP3 account. Each e-mail object is saved to the ReadyToShip project database. The e-mails in the POP3 account are only read. Once the e-mail is imported into the database, the user can then delete e-mails, create an incident from an e-mail, or link an e-mail with an existing incident. This import accounts for previously imported e-mails. Any e-mails already imported are not re-imported even if they are deleted from the ReadyToShip project database. The import

module also accounts for a tester that may have sent an e-mail but does not exist in the database. If the e-mail address is not found, a tester is created.

3.2.4 Manual Incident Creation

ReadyToShip gives the user the ability to also enter new incidents manually. These are for those incidents that originate by a non import method. The most likely scenario is when a beta tester calls Seapine to report an issue.

3.2.5 Send Mail Capabilities

ReadyToShip is able to act like a mail client and connect to a Simple Mail Transfer Protocol (SMTP) server to send e-mails. This allows for the user to reply, forward and send e-mails to testers. ReadyToShip keeps a track of the e-mail and if the reply is to an e-mail that is associated with an incident, it links the reply to this incident. This allows the user to keep track of all e-mails associated with the incident.

3.2.6 Project Administration Tools

The user has the ability to specify a POP3 account to import e-mails from and is able to specify an SMTP mail server to deliver e-mails. The administrator is also able to specify the TestTrack and SMF databases to import data from.

4. Design and Development

4.1 Budget

The project did not incur any costs. All applications used for the development of this tool were already owned by Seapine and the project did not require a dedicated computer or special hardware. Figure 6 below displays how much money this project would cost if Seapine did not have any of the components mentioned above (Total Retail Cost).

Item	Description	Retail Cost	Seapine Cost	Student Cost
Visual Studio Professional 2005 (7)	Already owned by Seapine	\$799.00	0.00	\$0.00
Microsoft® SQL Server Standard Edition 2005 (7)	Already owned by Seapine	1,849.00	0.00	0.00
Server Machine DELL OptiPlex GX620 DT(2)	Comparable machine already owned by Seapine	\$1,785	0.00	\$0.00
Total Retail Cost		\$4,433.00		
Total Seapine Cost			\$0.00	
Total Student Cost				\$0.00

Figure 6 – Actual Budget

4.2 Timeline

The timeline shown in figure 7 on the next page is accurate as of the time of the writing of this document. No major incidents occurred that would have changed the timeline.

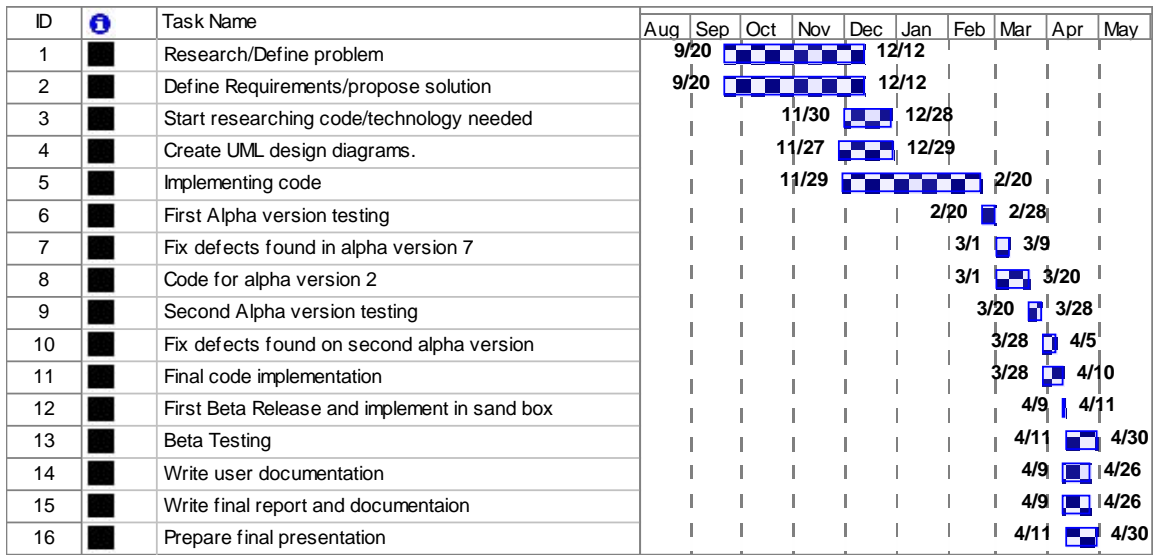


Figure 7 – Proposed Timeline

4.2.1 Fall Quarter Accomplishments

- Definition of the problem.
- Research of possible solutions.
- Solution identified.
- Proposal presentation made to faculty.
- Proposal document submitted.

4.2.2 Winter Quarter Accomplishments

- Application and database design started.
- Initial Implementation of code.
- Deliverables defined.
- User protocol defined.
- Design protocol defined.

- Design Freeze presentation made to faculty.
- Design Freeze document submitted.

4.2.3 Spring Quarter Accomplishments

- Continued and finalized implementation of code.
- Continued testing of implemented code.
- Help document finished.
- Applications finished.
- Presented project at Tech Expo.
- Final presentation made to faculty.
- Final project report submitted.

4.3 Software

The following lists the applications used in the development design and deployment of the application in order of most importance:

- Visual Studio 2005 – Application was programmed using C#. This language was chosen because of its ease of use and familiarity the author had with this language.
- SQL Server 2005 Management Studio – Used to manage test databases. Microsoft SQL server was chosen because of its availability under Seapine's current license.

- Surround SCM – Stored and managed the source code. Source Code Management (SCM) software allows to version each change and makes it easier to manage these changes.
- TestTrack Studio – Entered and tracked defects of applications and also defined Test Cases and performed Test Runs.
- Microsoft Project 2000 – Created project timeline and timeline figure used in this report.
- Microsoft Visio 2000 – Created UML diagrams used in presentations and reports, including this one.

4.4 Hardware:

No special hardware is required for this project. There were already numerous SQL 2005 servers installed on several machines. The user interface runs on any workstation that was imaged by Seapine's IT department.

5. Proof of Design

The following illustrates the different functionality offered by both applications.

5.1 Database Configuration

Both the RTS Admin and ReadyToShip applications need to connect to the RTS Admin database on startup. The user or administrator can either connect to an existing database or the application creates a new database.

The administrator does not need to first create the database in Microsoft SQL. The application creates the database, ensuring consistency and also preventing any errors caused by the administration creating the database incorrectly.

When either application starts, it looks for a configuration file to get the connection information to the RTS Admin database. If it does not find the configuration file, the configuration file has incorrect information, or if the application can not connect with the information in the configuration file, the administrator is prompted to configure the connection. This is accomplished by giving the user a window on which to enter the configuration information. RTA Admin shows a different window than ReadyToShip. The main difference is that the window displayed by RTS Admin also gives the administrator the ability to tell the application to create a new database. Figure 8 below shows the window that is displayed by RTS Admin to allow the administrator to configure the database connection or to create a new one.

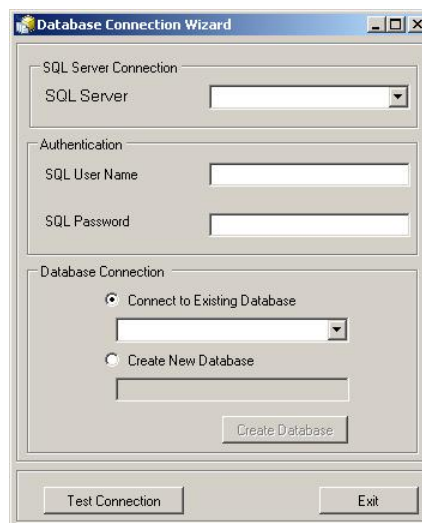


Figure 8 - Database Connection Configuration Window

The application allows the user to browse the network for available SQL servers on the network. Figure 9 below shows the available SQL servers found on the Seapine network using the window that is displayed by ReadyToShip.

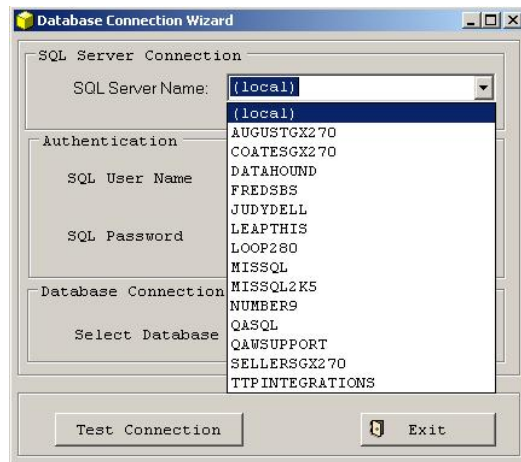


Figure 9 – Available SQL Servers in the Seapine Network

Once a server has been selected and valid credentials entered, the administrator is able to browse databases in the specified SQL server. Figure 10, located on the next page, displays the databases available in the selected Microsoft SQL Server.

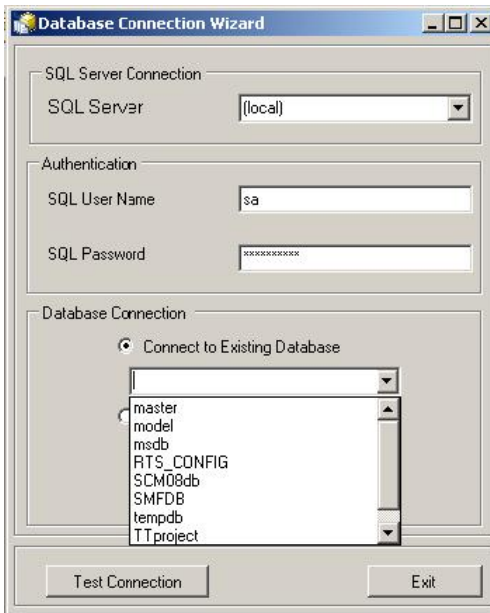


Figure 10 – Available Databases in the Specified SQL Server

Once the information has been entered, the application tests the validity of this information and if it is valid, it saves it to the configuration file.

5.2 RTS ADMIN

RTS Admin provides the administrator two main functions. To create users that use the ReadyToShip application and to set up the connectivity to the projects these users use with the ReadyToShip application.

5.2.1 User Management

The administrator must create the users that use the ReadyToShip application. To create a user, the administrator selects “Manage”>”Users” from the menu bar, or clicks on the “Manage Users” button on the toolbar. Figure 11, located on the next page, shows both options outlined.

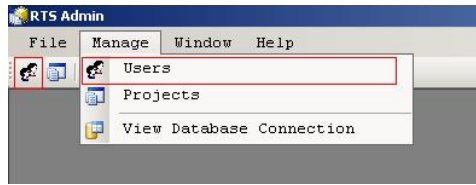


Figure 11 – Manage User Menu Option and Toolbar Button

This displays the User List window, which displays any existing users. The administrator can create, edit, view or delete a user from this window. Figure 12 below displays the User List window.

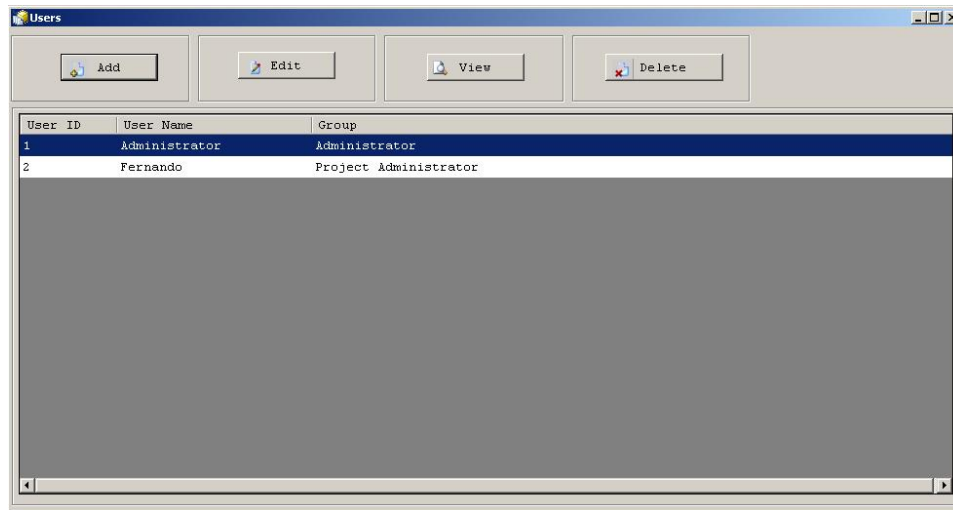


Figure 12 – User List Window

To create a new user, the administrator clicks on the “Add” button. This brings up the “Add User” window. To add a user, only a user name and the role need to be provided. It is not required to enter a password. Figure 13, located on the next page, shows the “Add User” window.



Figure 13 – Add User Window

5.2.2 Project Management

The administrator must create the projects that the users use to track beta tester communications. To create a project, the administrator selects “Manage”>”Projects” from the menu bar or clicks on the “Manage Projects” button on the toolbar. Figure 14 below shows both options outlined.

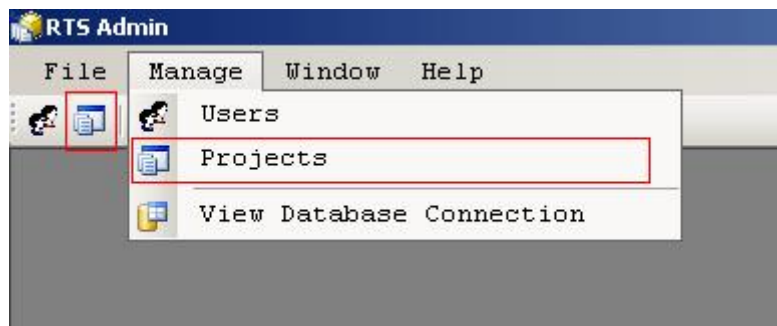


Figure 14 - Manage Projects menu and toolbar button

This displays the project list window. The administrator can add, edit, view or delete a project from this window. Figure 15 below displays the project list window.

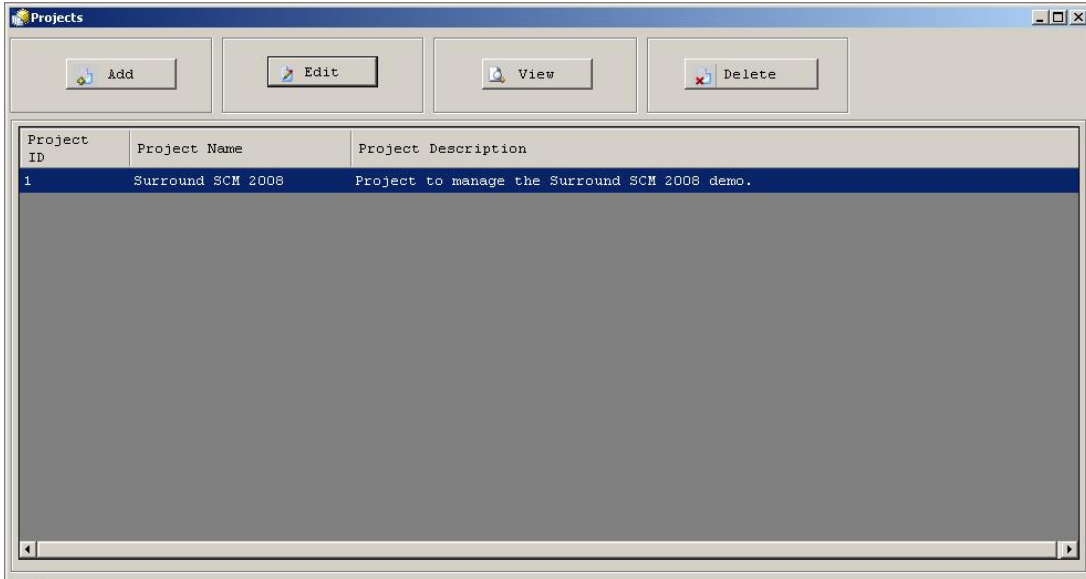


Figure 15 - Project List Window

To create a new project, the administrator clicks on the “Add” button. This brings up the “Add Project” window. To add a project, the administrator must provide three different types of information. The “Add Project” window has three tabs, each tab for each type of information. The first tab is the “General” tab. There, the user enters the project name and the project description. Figure 16, located on the next page, displays the “General” tab in the “Add Project” window.

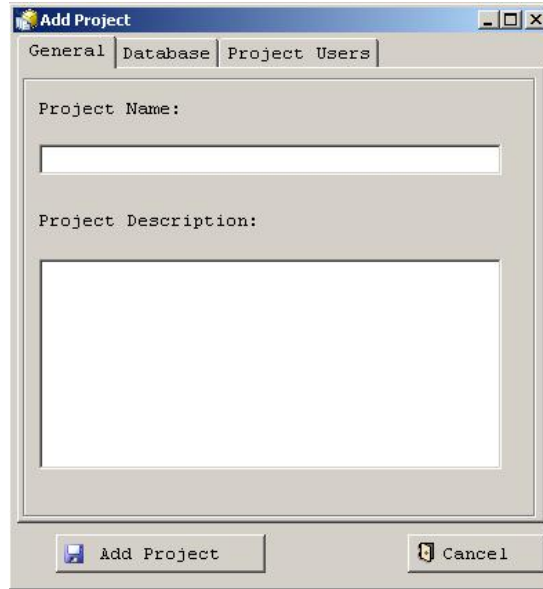


Figure 16 - Add Project Window General Tab

The second type of information is the connection information to the SQL server database for the project. The administrator must provide the SQL server, valid SQL credentials and a name of the database to create in the SQL server. The SQL user becomes the database owner because the database is created under this SQL user. Figure 17, located on the next page, displays the “Database” tab in the “Add Project” window.

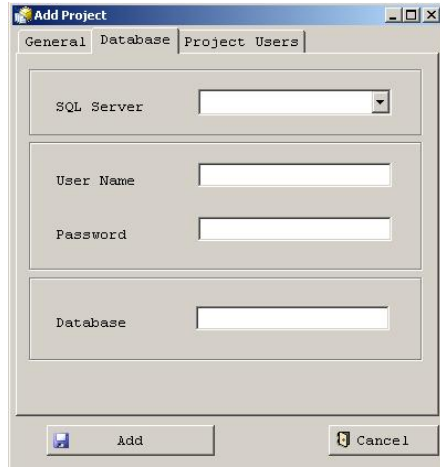


Figure 17 - Add Project Window Database Tab

The third type of information is the user or users that have access to this project. In most cases there is only one user associated with a project, since only one employee manages the beta program for a product. Figure 18 below displays the “Project Users” tab.

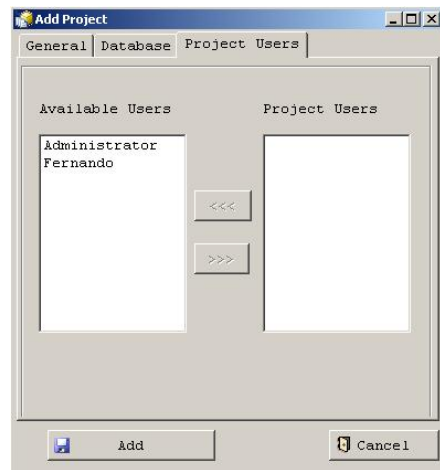


Figure 18 - Add Project Window Project Users Tab

5.2.3 User Documentation

The administrator is able to access documentation to learn how to use the application. To access the user documentation, the administrator selects “Help”>”RTS Admin Help”. The help documentation in RTS Admin is a static web site. Figure 19 below shows the various sections available in the user documentation.

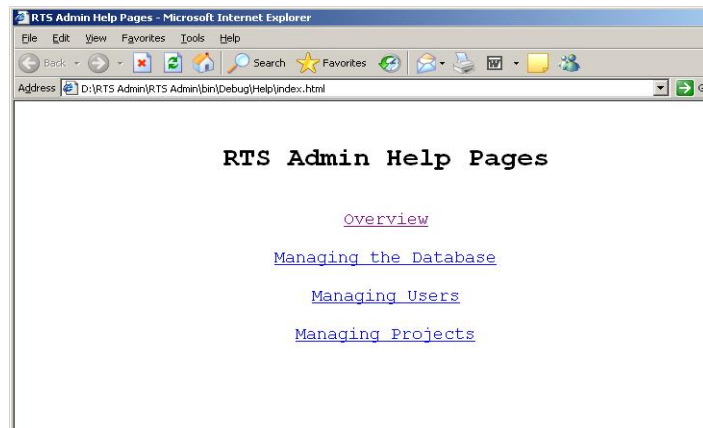


Figure 19 - RTS Admin Help Topics

5.3 ReadyToShip

The ReadyToShip client application offers several functionalities to aid the user with tracking beta tester communications.

5.3.1 Log-In Process

When ReadyToShip starts and finds a valid a valid connection to the RTS Admin database, it displays the user window to enter a user name and password. Once the information is entered, the user clicks on the “Get Projects” button to get a list of the

projects that are available to this user. Figure 20, located below, displays the log in window.



Figure 20 - Log in Window

5.3.2 Tester Management

To manage beta testers, the user selects “View”>”Testers” from the menu bar or clicks on the “View Testers” button. Figure 21 below show these options outlined.



Figure 21 - View Testers toolbar button and menu option

The tester list window appears. From this window, the user can select to add, edit, view or delete a beta tester. Figure 22 below displays the Tester list window.

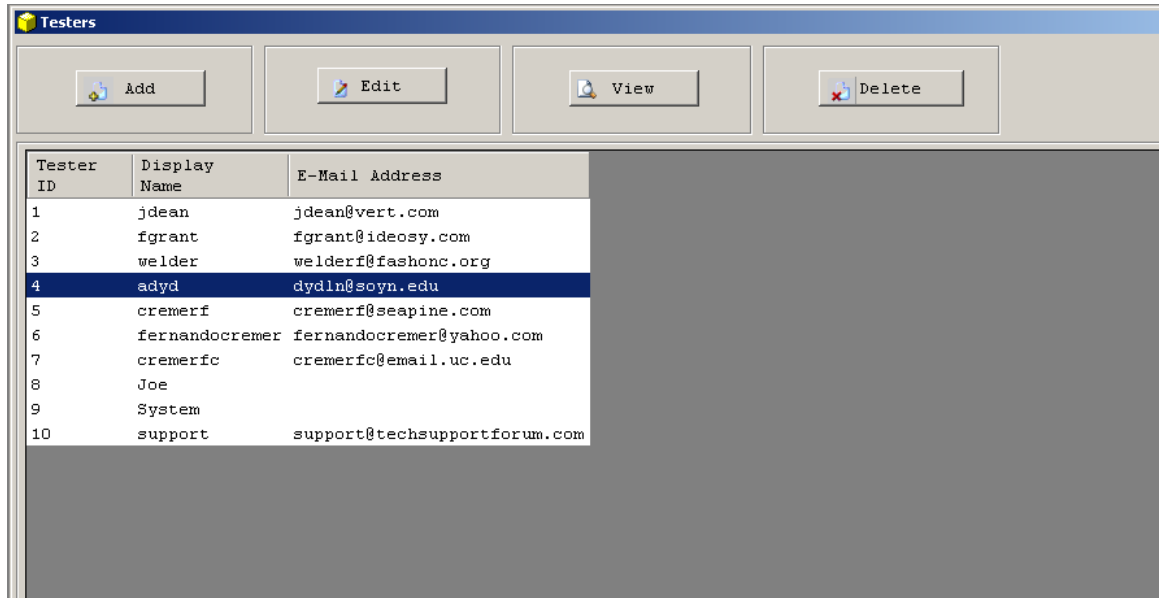


Figure 22 - Tester List Window

In the current structure of the beta program web site, most users are imported and the “Add” button should be rarely used. When beta testers register for the beta program, they only have to provide a “Display Name”, a password and an e-mail address. ReadyToShip only imports the “Display Name” and the e-mail address only. To view information about the tester, the user selects a tester and clicks on the “View” button. This brings up the “View Tester” window. This window consists of two tabs, the first one displaying the display name used in the user forum and the e-mail address provided in the registration process. Figure 23, located here below, shows the “Detail” tab in the “View Tester” window.

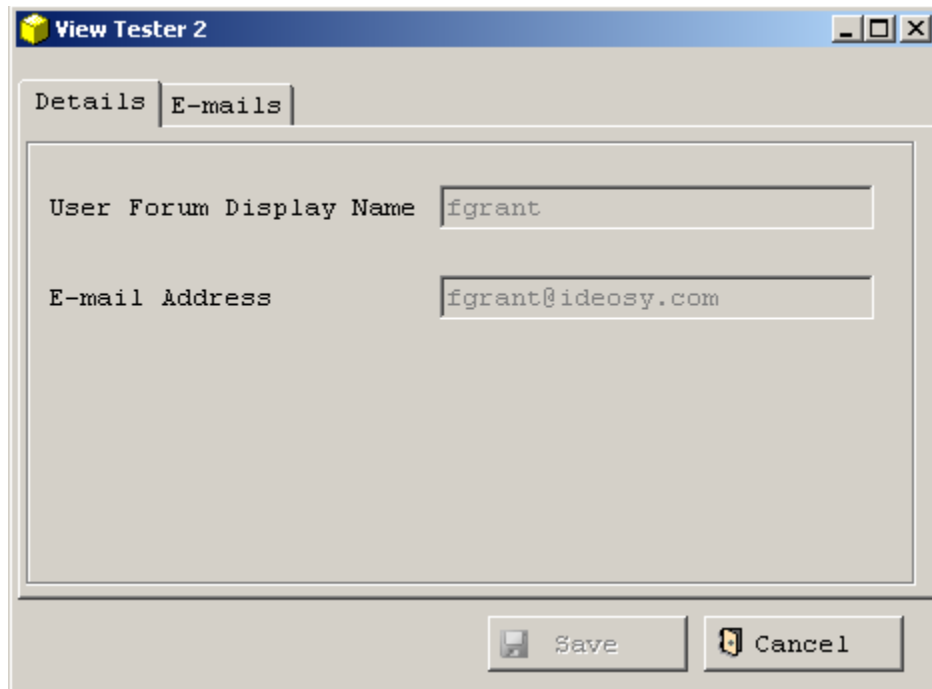


Figure 23 - Details Tab in the View Tester Window

The second tab of the “View Tester” window is used to view the e-mails received from this tester. The user can only view e-mails from this tab. To reply or forward any e-mail from this window, the user selects “edit” from the Tester list window. The “E-mails” tab in the “Edit User” window has the “reply” and “forward” buttons enabled. Please refer to section 5.3.3 for more details on the e-mail functionality. Figure 24, located on the next page, displays the “E-mail tab” of the “Edit Tester” window.

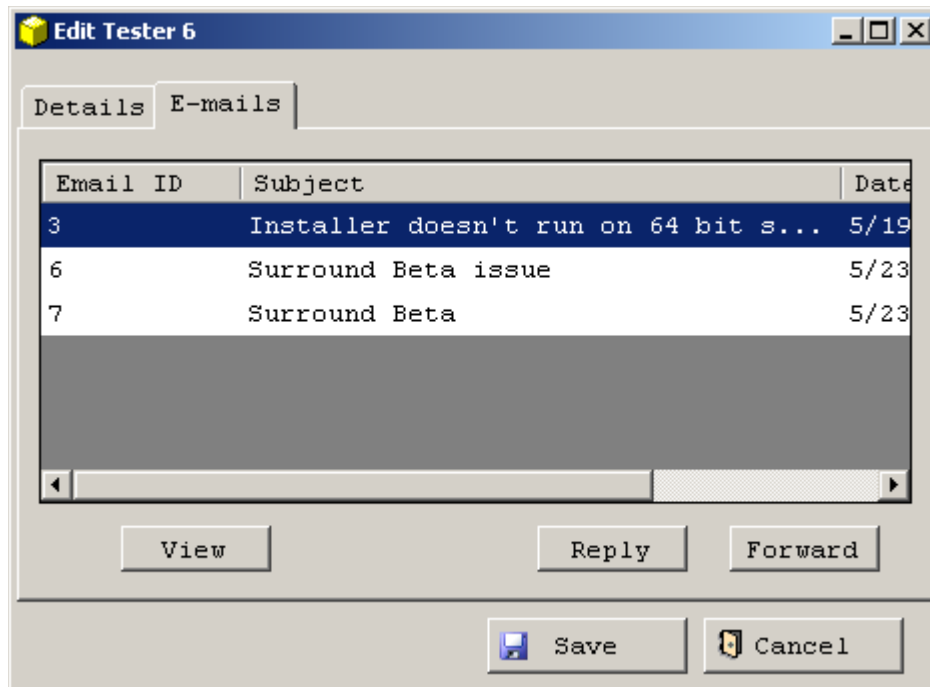


Figure 24 - E-mails Tab in the View Tester Window

5.3.3 Incident Management

ReadyToShip manages incidents. An incident is any contact a beta site user makes. An e-mail, a phone call, a user forum post, feature request, defect report, all are referred to as incidents. Incidents are associated with the beta site user. Each incident has a source, which indicates the incidents' origin (phone call, e-mail, user forum, etc...). Incidents also have a status, which help the user manage the incidents. The user is able to create incidents manually for those initiated through a phone call.

To manage incidents, the user uses the Incident list window. To access the Incident list window, the user clicks on the "View">"Incidents" option from the menu bar or clicks on the "View Incidents" button on the toolbar. Figure 25, located on the next page, displays both options outlined.

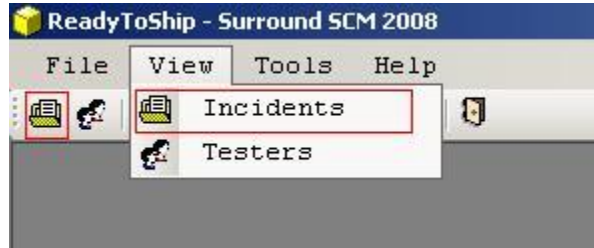


Figure 25 – View Incident Menu Option and Toolbar Button

The Incident list window displays all of the incidents. On the upper right corner, the window has a drop down menu with options to filter which incidents to display.

Figure 26 below displays the Incident list window.

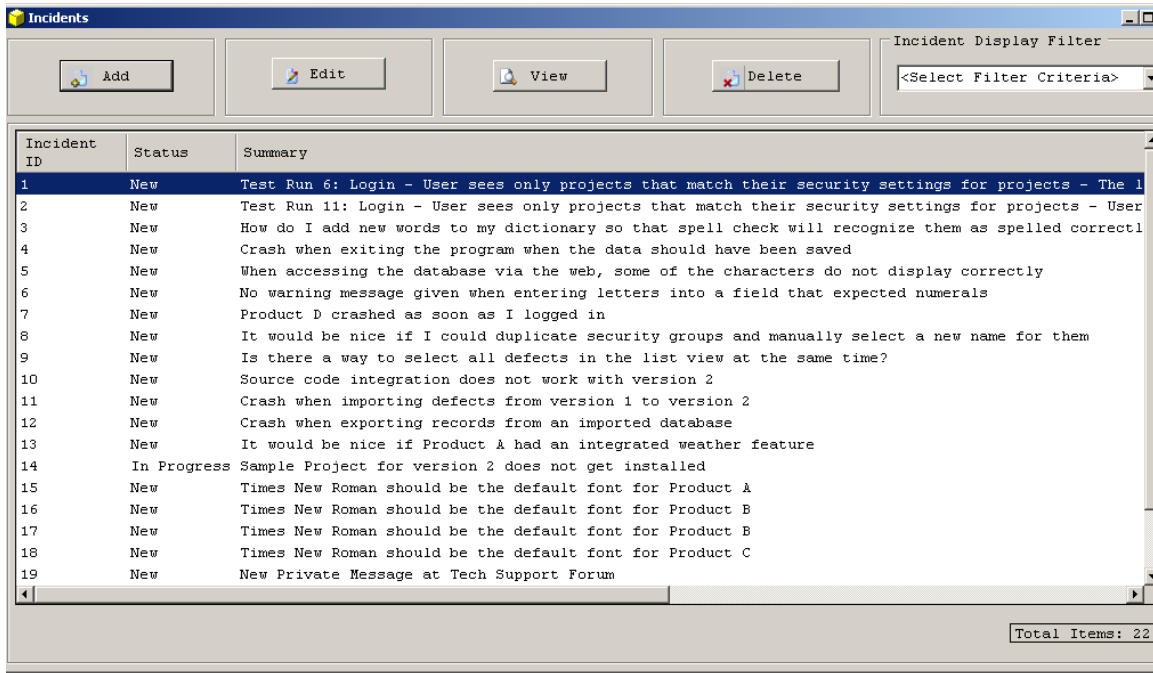


Figure 26 - Incident List Window

On the upper right hand corner the user can use the Incident Display Filter to filter the Incidents that are displayed. The user clicks on the drop down menu and is given five options that the user can filter the display by. Figure 27 below displays the filter options given to the user.

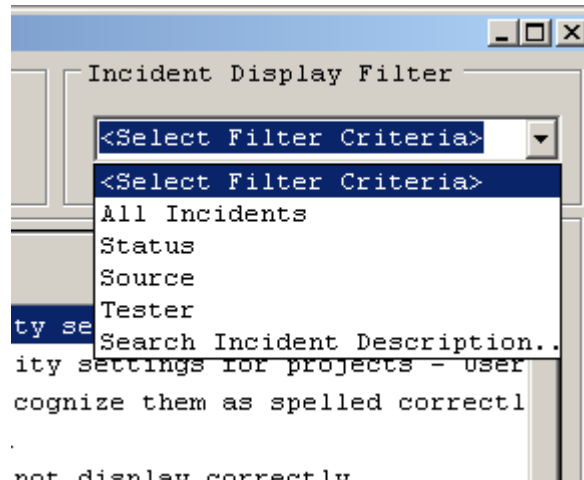


Figure 27 - Incident Display Filter Options

The first option displays all of the incidents in the project that user is logged into. The remaining options display a second window from which to further filter the display of incidents. Options two through four display a window with a drop down menu. For example, if the user selects the “Tester” option, the second window displays a drop down menu with a list of all the testers. The user selects a tester to only have displayed the Incidents reported by the selected tester. The last option displays a window with a text box to enter a string to search the incident description and only display those incidents whose description matches this string.

On the bottom right corner displays a total of all incidents displayed. This adds power to the filter options. In the example in the paragraph above, not only can the user filter the display of incidents by a tester, the user can also see a total of the incidents reported by this tester.

In the current beta program structure, more incidents are imported by other events. However, since a beta tester may still contact Seapine by phone, the “Add” button gives the user the ability to also track those communications.

The user can add, edit view or delete any incident by selecting the appropriate button at the top of the incident list window. All these buttons except for the delete window displays the Incident detail window. Depending on the button selected, the “Add Incident”, “Edit Incident” or “View Incident” detail window displays. Figure 28, located next page, displays the “Edit Incident” detail window.

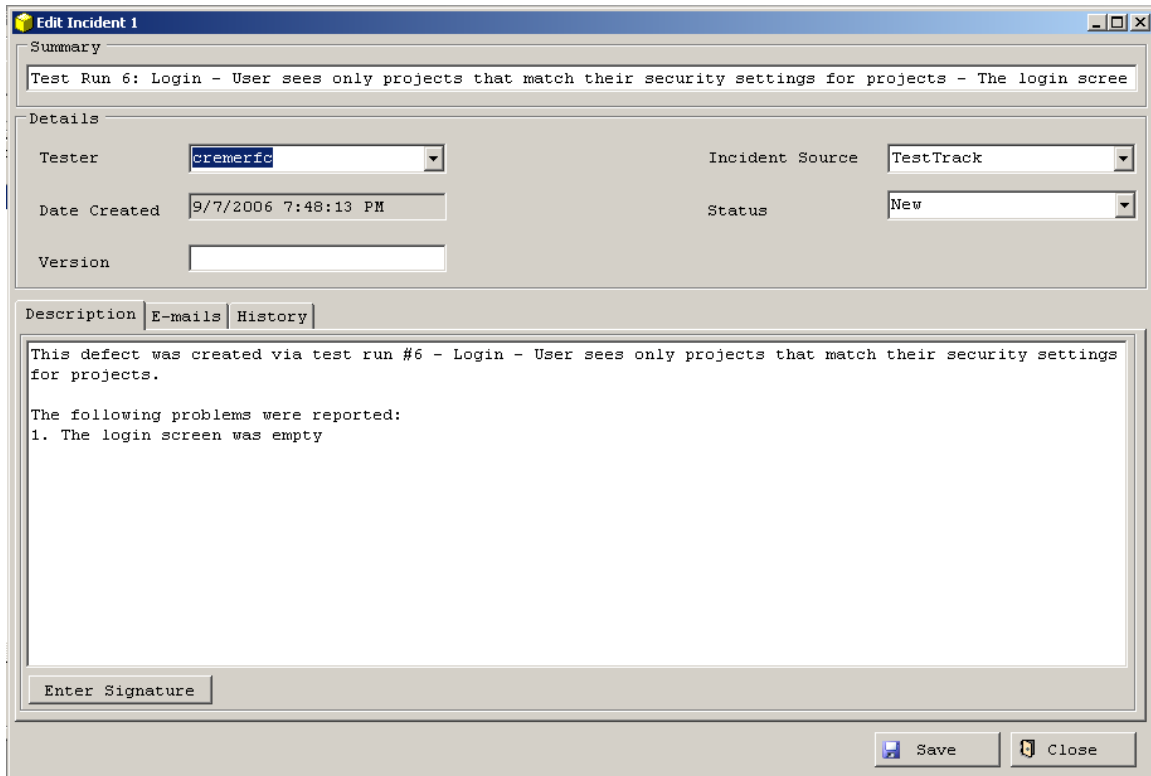


Figure 28 – Incident Detail Window

The Incident detail window consists of an upper and lower section. The upper section of the window contains fields with details about this incident. It contains the tester that reported the incident, a text field to enter the beta release version involved in the incident, the date when the incident is reported, the source of the incident and finally, the status of the incident. The status is used by the user to better manage the Incidents by being able to identify which incidents need the user’s attention. There are four status options for incidents. Figure 29, located on the next page, displays the status options displayed in the “Status” drop down menu.

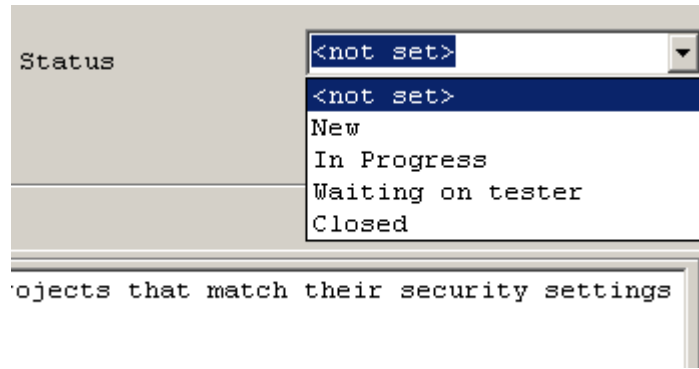


Figure 29 - Status Options

The lower section of the Incident detail window consists of three tabs. First is the “Description” tab, which contains a rich text box for a more detailed description of the incident. The tab also contains the “Enter Signature” button, which appends the description with the current system time and date, as well as the currently logged in user. Figure 30 below displays the signature button and how the signature is displayed outlined.

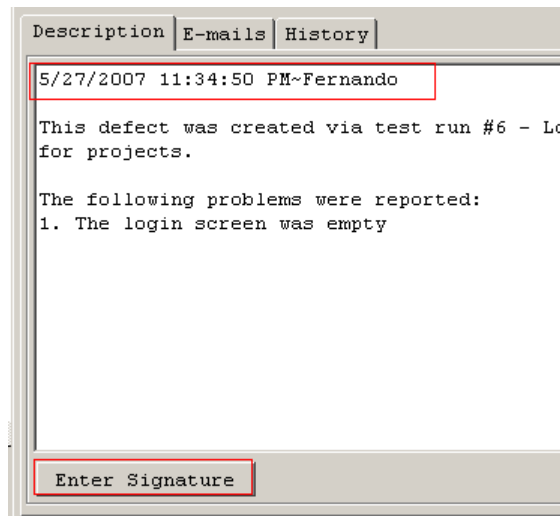


Figure 30 – Enter Signature Button and Sample Signature

The second tab is for tracking e-mails specific to this incident. This tab shows the e-mails received and sent related to this incident. The user can only reply, forward or send an e-mail related to this incident if the user edits the incident. Please refer to section 5.3.3 for more details on the e-mail functionality. Figure 31 below shows the e-mail tab.

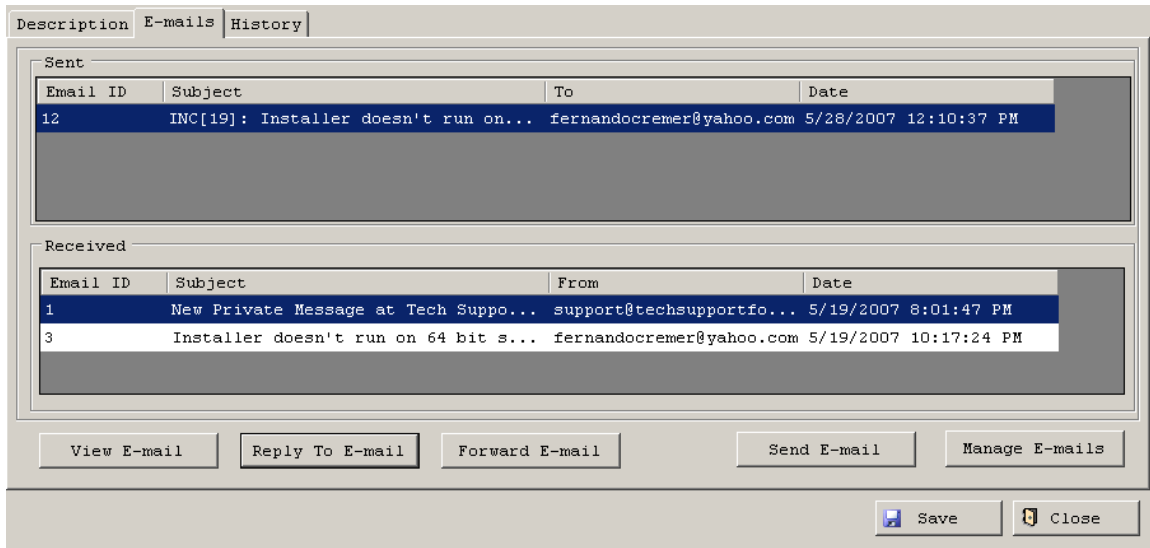


Figure 31 – E-mails Tab on the Incident Detail Window

The third and last tab is for viewing the history of the specified incident. Every event that somehow modifies the incident is recorded. From a change in the status to an e-mail being replied to, all events are recorded. Figure 32, located on the next page, displays the “History” tab.

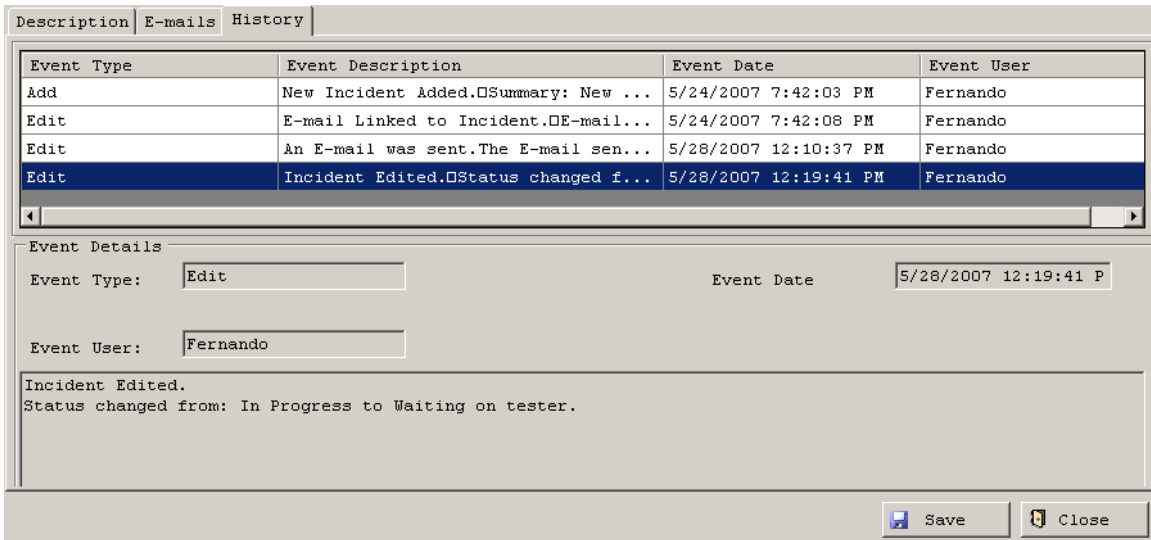


Figure 32 – History Tab on the Incident Detail Window Figure

5.3.4 E-mail Management and Tracking

There are several aspects to the e-mail functionality provided in the ReadyToShip application. ReadyToShip imports e-mails from a specified POP3 e-mail account, sends e-mails through a specified SMTP server, and tracks the e-mails through various options.

Before e-mails can be managed, the e-mail configuration must be set. The e-mail configuration includes the POP3 account settings as well as the SMTP server to send the e-mails through. To set these settings, the user selects “Tools”>”Options”>”E-mails” from the menu bar or selects the “Manage E-mails” button on the toolbar.

The manage e-mails window appears. From this window the user can get e-mails from the POP3 account, filter the view of e-mails, search the e-mails, view an e-mail, reply to an e-mail, forward an e-mail, create an incident from an e-mail, link an e-mail to

an existing incident and delete any e-mail. Figure 33 below displays the “Manage E-mails” window.

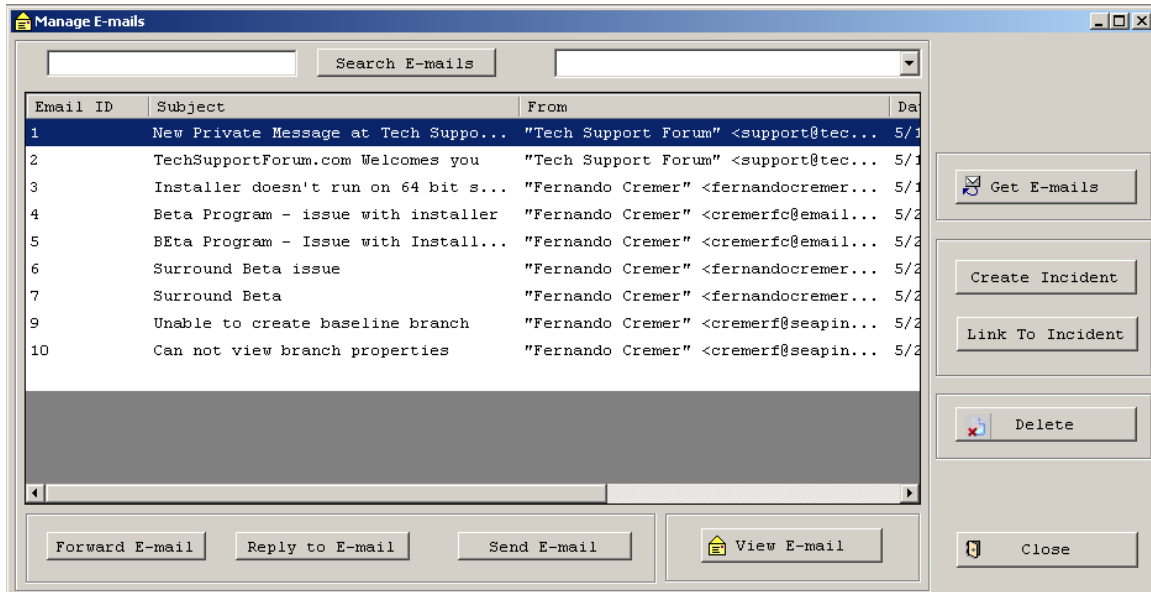


Figure 33 – Manage E-mails Window

When E-mails are imported, ReadyToShip reads the e-mails from the POP3 account and copies the e-mail information to its database. The e-mail Unique Identifier (UID) is saved on a separate table for each e-mail imported. When an e-mail is deleted from the Manage E-mails window, it is only deleted from the ReadyToShip database not the POP3 account. The corresponding record on the table that contains the UID, however is not deleted. This is because when ReadyToShip imports e-mails, it first checks the e-mail UID for every e-mail already imported. This prevents re-importing old e-mails.

When an incident is created from an e-mail, the incident’s source is set to “E-mail” and the incident status is set to “new”. The add history event associated with the

incident documents the e-mail that created it. Also, ReadyToShip uses the e-mail address that sent the e-mail and searches the tester table for a match. If it finds a match, ReadyToShip associates the incident with this tester. If it does not find a match, ReadyToShip creates a new tester entry. It strips the domain portion of the e-mail address (everything starting from the “@”) and uses that as the Display Name. When an e-mail is linked to an existing incident, an edit history event is recorded and it documents the e-mail.

The “Manage E-mails” window provides some filter options to limit the display of the e-mails shown. Figure 34 below displays the available filter options.

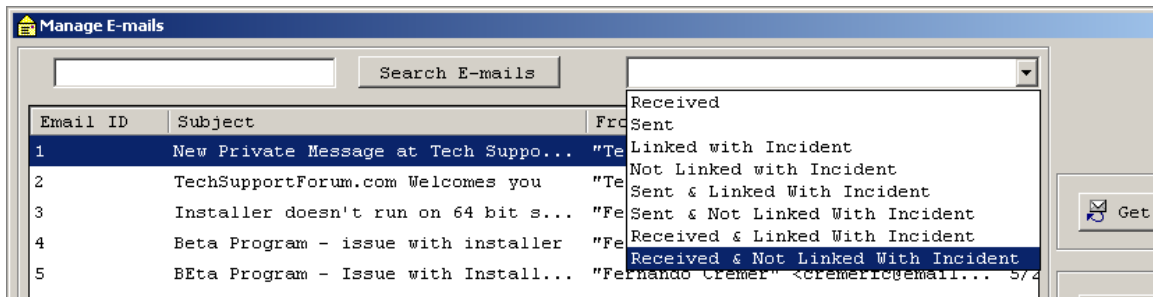


Figure 34 – Filter Options in the Manage E-mails Window

With the use of these filters, the user is able differentiate those e-mails that have been sent versus those received, linked with incident versus those that are not, as well as other combinations. The “Search E-mails” button searches all the e-mails for the specified string entered in the text box next to the button. It searches the e-mail subject, body and e-mail address.

5.3.5 TestTrack Import:

ReadyToShip imports the records from the TestTrack project database. The user must first set the connection settings to be able to retrieve the records from the TestTrack database. To enter these settings, the user selects the “Tools”>”Options”>”TestTrack”. This opens the window where the user enters the connection information. The window provides a drop down menu that allows browsing the network for available SQL servers. Figure 35 below illustrates the dropdown menu with the available SQL servers in the Seapine network.

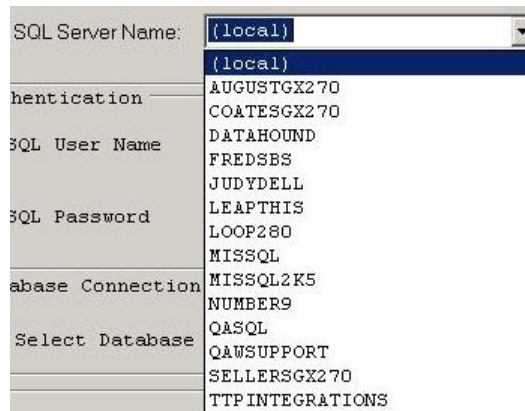


Figure 35 – SQL Server in the Seapine Network

The user login used to connect must have permissions to be able to browse the database tables and retrieve the records. Figure 36, located on the next page, illustrates the TestTrack Import Options window.

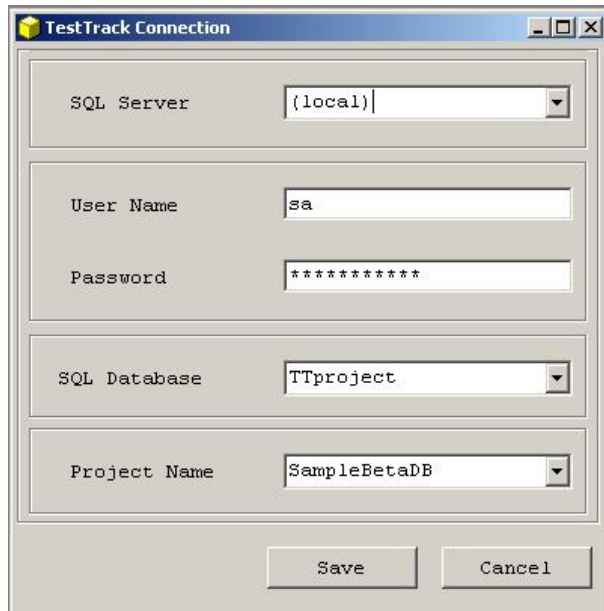


Figure 36 – TestTrack Import Options Window

To import the records from the TestTrack Database the user selects the “Tools”>”Import From TestTrack” from the menu bar or clicks on the “Import From TestTrack” button from the toolbar. Figure 37 below displays the options outlined.

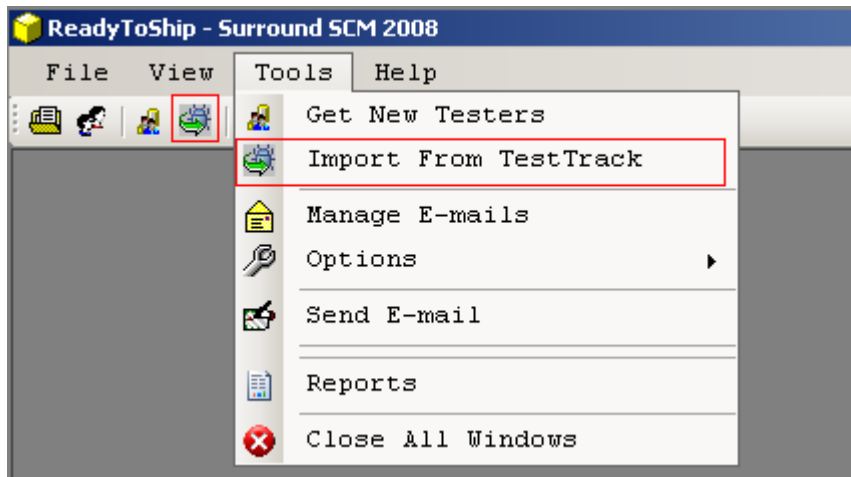


Figure 37 - Import From TestTrack Menu Option and Toolbar Button

When ReadyToShip imports records from the TestTrack project database, it reads the records that it needs and copies them to the table where the incidents are stored. The incident's source is set to "TestTrack" and the incident's status is set to "New". ReadyToShip also looks at the user record in the TestTrack database and it looks at the tester table in the ReadyToShip project database. If it finds the tester, it associates the incident with that tester. If it doesn't find the tester, it adds the tester to the ReadyToShip project database and associates the incident with the tester.

ReadyToShip also saves the last imported record ID from the TestTrack database. The record ID is a number that increments each time a record is added in TestTrack, so ReadyToShip only needs to know the ID of the last imported record, since that is the greatest number at that time. Next time the user imports from the TestTrack project database, it only looks for those records with a record ID greater than the one saved in the ReadyToShip project database. This prevents old records from being imported again.

ReadyToShip displays a window with a progress bar and a label to displays messages to the user. If records are imported, the label displays the total number of imported defects, if no new records are found, the label displays a message stating so, and if an error occurs, the label displays the error. Figure 38, located on the next page, displays the TestTrack Import status window.

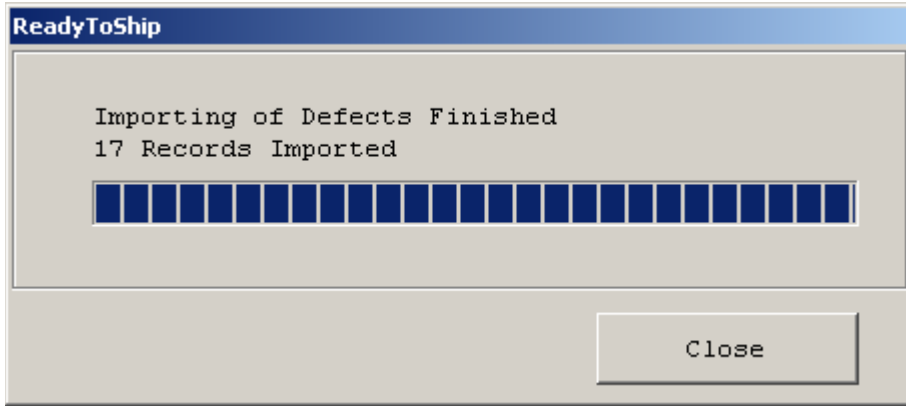


Figure 38 - Import From TestTrack Progress Window

5.3.6 Simple Machine Forum Import

ReadyToShip imports records from the Simple Machine Forum (SMF) database. The user must first set the connection settings to this database. To set these options, the user selects “Tools”>”Options”>”Simple Machine Forum”. The user must provide the host name of the MySQL database, valid credentials, and the database name.

To import the records, the user selects “Tools”>”Get New Testers” option from the menu bar or clicks on the “Get New Tester” button on the toolbar. The “memberID”, which is the associated ID of the tester in the SMF database, is also imported. When ReadyToShip imports the records from the SMF database, it checks the “memberID” and only imports those that do not match any “memberID” stored in the ReadyToShip database.

5.3.7 Reports

ReadyToShip provides reports to give the user the ability to get some metrics on the data in the ReadyToShip project. To run a report, the user selects “Tools”>”Reports” from the menu bar or clicks on the “Reports” button on the toolbar.

The Reports window displays with a dropdown menu and a label that tells the user that depending on the size of the ReadyToShip project database, the application may appear to hang while the report is generated. The drop down menu has all of the report options that ReadyToShip offers. Figure 39 below displays the report options that ReadyToShip offers.

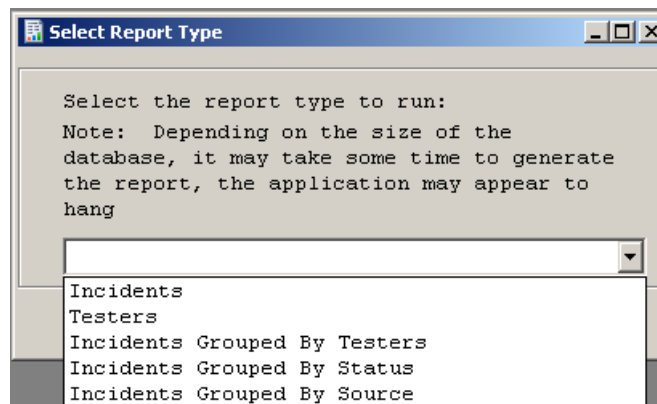


Figure 39 – Report Options

Reports in ReadyToShip are in HTML format. ReadyToShip dynamically creates the HTML page when the report is generated. The style used in the HTML pages is very minimal. The first two report options display the entire records and it gives a total at the bottom of the report. The remaining three options display the subtotal for each grouping

and also displays to total of all the records. Figures 40 (below) and 41 (located on the next page) display the “Incidents Grouped By Testers” report.

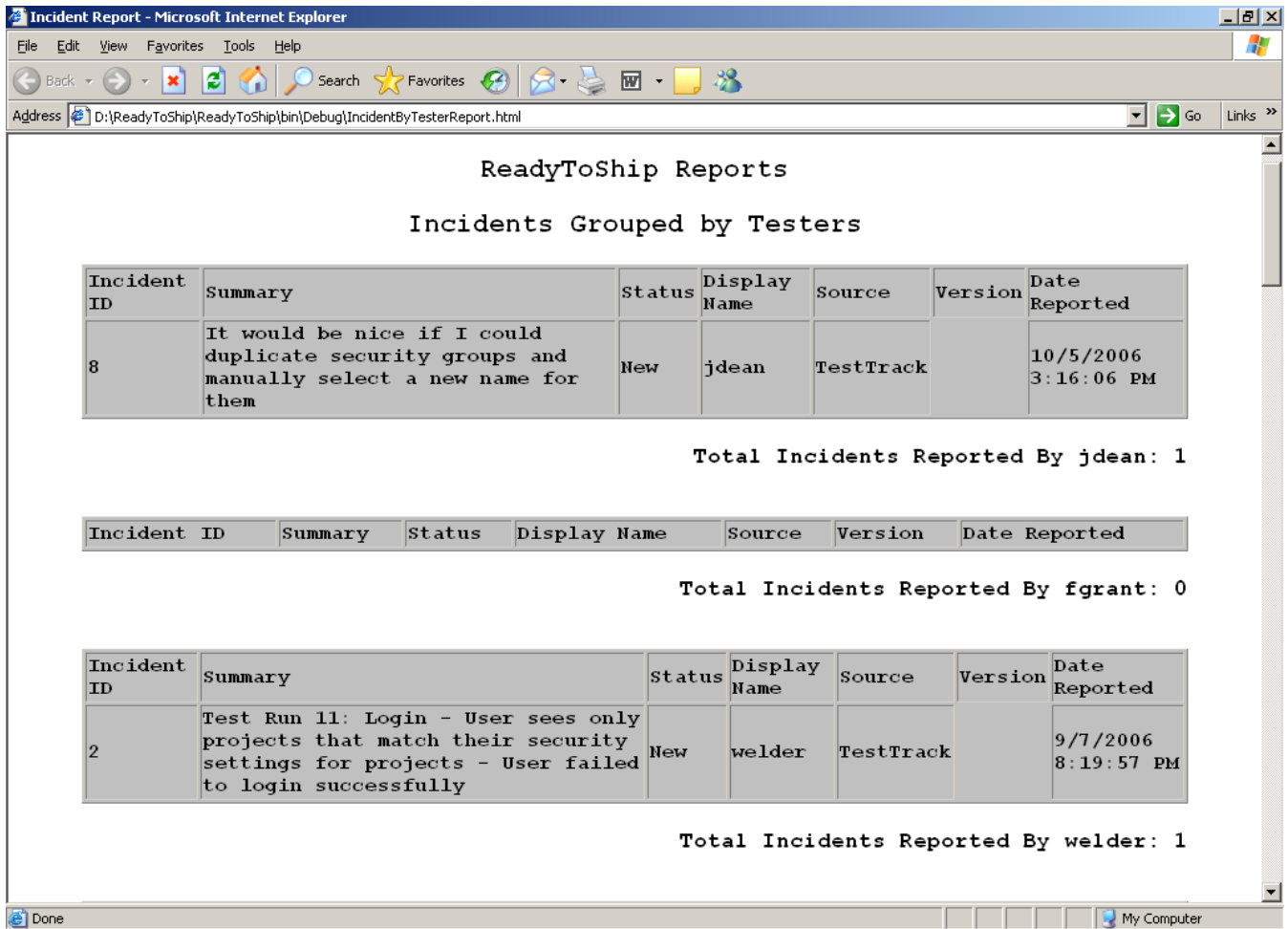


Figure 40 – Incident Grouped By Tester Report (Top)

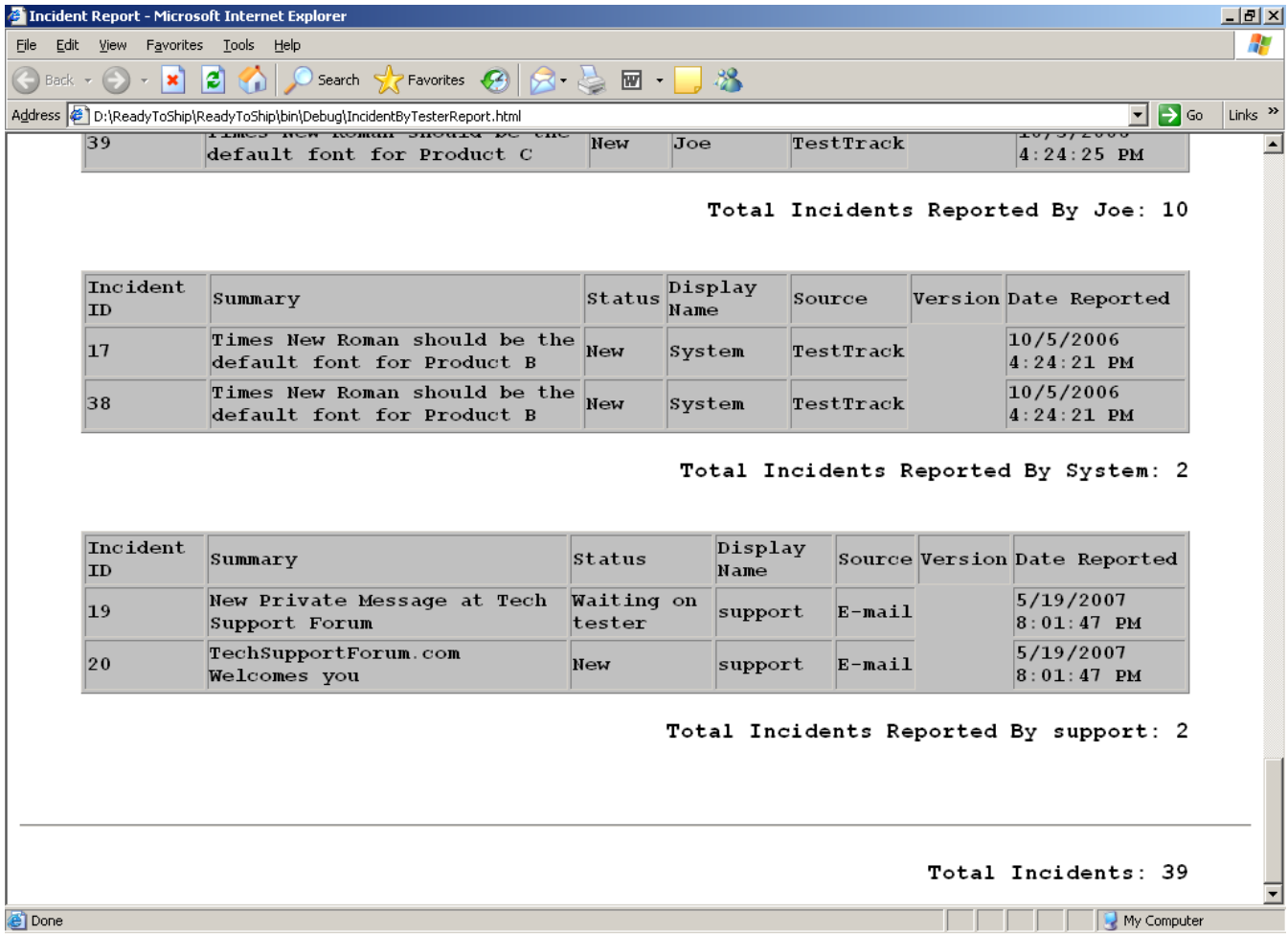


Figure 41 – Incident Grouped By Tester Report (Bottom)

5.3.8 User Documentation

The user can access documentation to learn how to use the application. To access the user documentation, the user selects “Help”>”ReadyToShip” option from the menu bar, or clicks on the “ReadyToShip” button on the toolbar.

The documentation is presented as a static web site that consists of various HTML files. The user can use the browser features to search, print and export the user

documentation. The first help page shown is the index page which shows the various help topics available. Figure 42 below displays the index page.

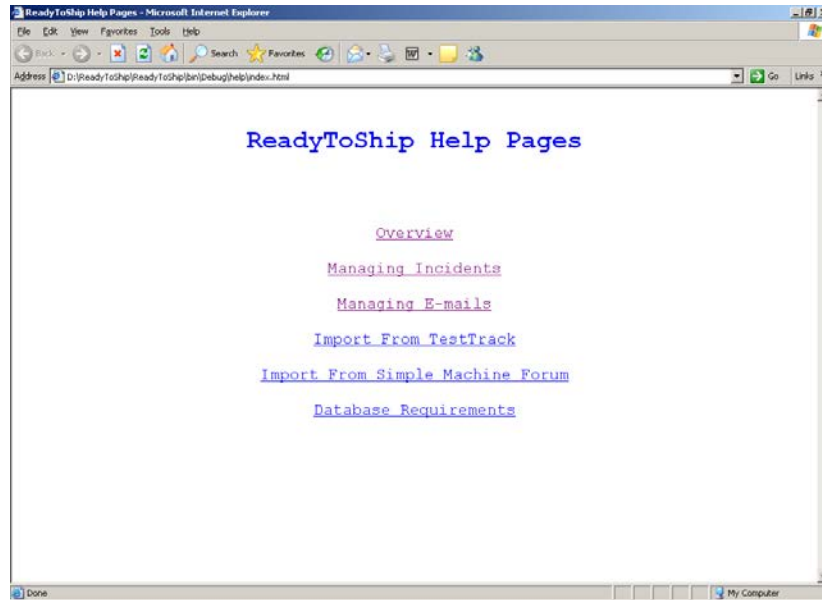


Figure 42 – ReadyToShip Index Page

From the index page, the user is able to select any category to view. Once the user is on that page, there is a link at the top and bottom of the page that the user can click on to return to the index page. Figure 43, located on the next page, displays the top of the “Managing Incidents” page.

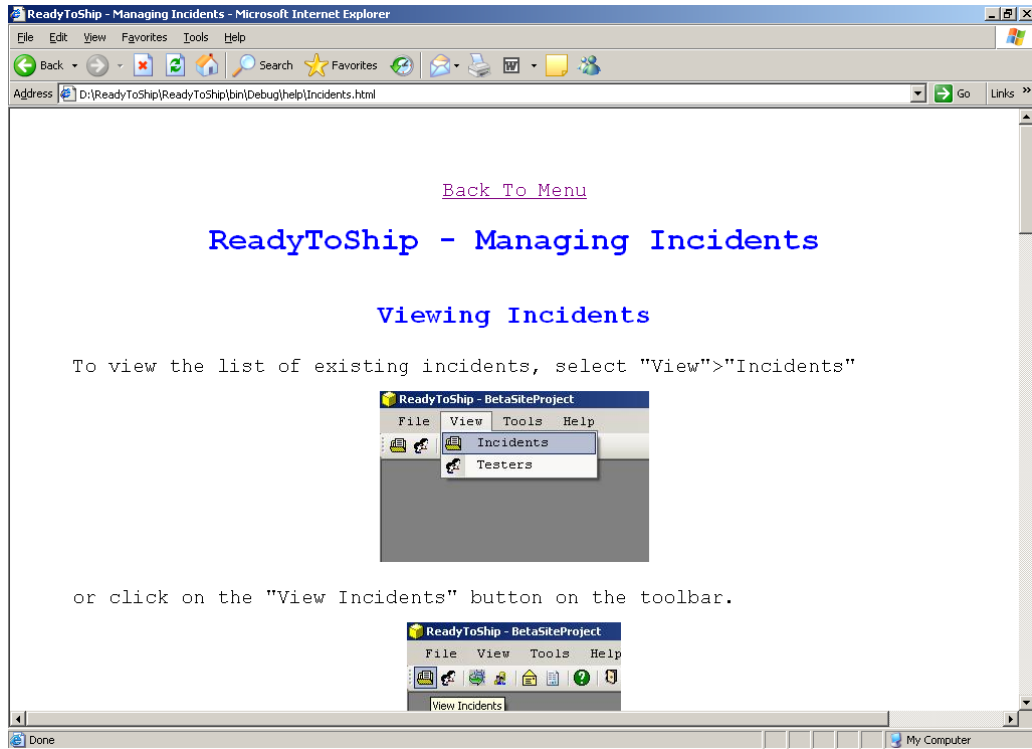


Figure 43 – Incident Help Page

6. Conclusion and Recommendations

6.1 Conclusion

ReadyToShip solves the problem posed by the current Beta Program structure. It provides an easy to use interface and it requires minimal configuration. The project allowed the author to create a complete, sophisticated program for the first time. The project also allowed him to experience the complete software development.

The ReadyToShip application and documentation will be given to Seapine for optional use in the management of future Beta Programs. The source code will be placed in the Seapine's Source Code Management solution.

6.2 Recommendations

The version of ReadyToShip in this document is 1.0, the very first release. Since the source code will be available on the Seapine Source Control server, Seapine employees may add features as they see the need.

There are several improvements that the author plans to make within the next year. Tentatively, these features would be available in version 2.0 of the application. Some of the features that may be added may include but are not limited to the ability to create custom filters, the ability to create custom reports, a stronger search feature, integration with Seapine's Active Directory Server and the concept of user settings to customize the application.

References

1. CenterCode Software. 20 Nov. 2006.
<<http://www.centercode.com/solutions/connect/>>
2. DELL computers
<http://configure.us.dell.com/dellstore/config.aspx?cs=555&oc=MLB1557&m_1=1=XPP2E&m_1=628DD&c=us&l=en>
3. Fine, Michael. Beta Testing for Better Software. New York: John Wiley & Sons, 2002.
4. Larman, Craig. Applying UML and Patterns. Upper Saddle River, NJ: Prentice Hall PTR, 2005.
5. Lawley, Brian. "How to Run an Effective Beta Program." Online whitepaper. 280 group, 2006 <http://www.280group.com/whitepapers.htm#beta>
6. Lawley, Brian. "How to Run a successful Beta Program." Online webinar. www.FeaturePlan.com. <http://www.featureplan.com/forms/brian2/brian2.html>
7. Microsoft Corporation.
<<http://www.microsoft.com/products/info/default.aspx?view=22&pcid=515c9859-958b-4433-b4f9-91f37258ca2f>>
8. Questa, Jennifer. Personal interview. 30 Oct. 2006.
9. Riccetti, Rick. Personal interview. 22 Sept. 2006.
10. Sceppa, David. Microsoft ADO.NET. Redmond, WA: Microsoft, 2002.
11. Seapine Software. 15 Nov. 2006. <<http://www.seapine.com/company.html>>
12. Spolsky, Joel. "Top Twelve Tips for Running a Beta Test", Online posting. 2 March 2004. <http://www.joelonsoftware.com/articles/BetaTest.html>.

13. Vireira, Robert. Beginning SQL 2005 programming. Indianapolis: Wiley, 2006.
14. Watson, Karli, et al. Beginning Visual C# 2005. Indianapolis: Wiley, 2006.