

This assignment will help you to start you on the road to using a *Bug Reporting* system as part of your everyday development activities ...

Using an On-Line Bug Tracker

*Learning to report issues as a **User***

SEF – Fall 2014

Contents

The Back-Story 2

Pre-Assignment Activities..... 2

How Best to Proceed through this Assignment 2

Tenant (v1.01) ... The Program 4

FogBugz ... the On-Line Reporting Tool 5

Assignment Deliverables 9

The Back-Story ...

In this assignment, you play the role of a Software Engineering student who has hired themselves out as a **tester**. Specifically, you have been hired by a collection of landlords to test a program for them and review it. The landlords have hired a Software Development company known as **BuggySoft Inc.** to create this program, so that they can easily manage all of the tenants in their buildings. So in reality, what you are doing is performing User Acceptance Testing (UAT) on behalf of the landlords!

"This should be an easy task" you think to yourself ...

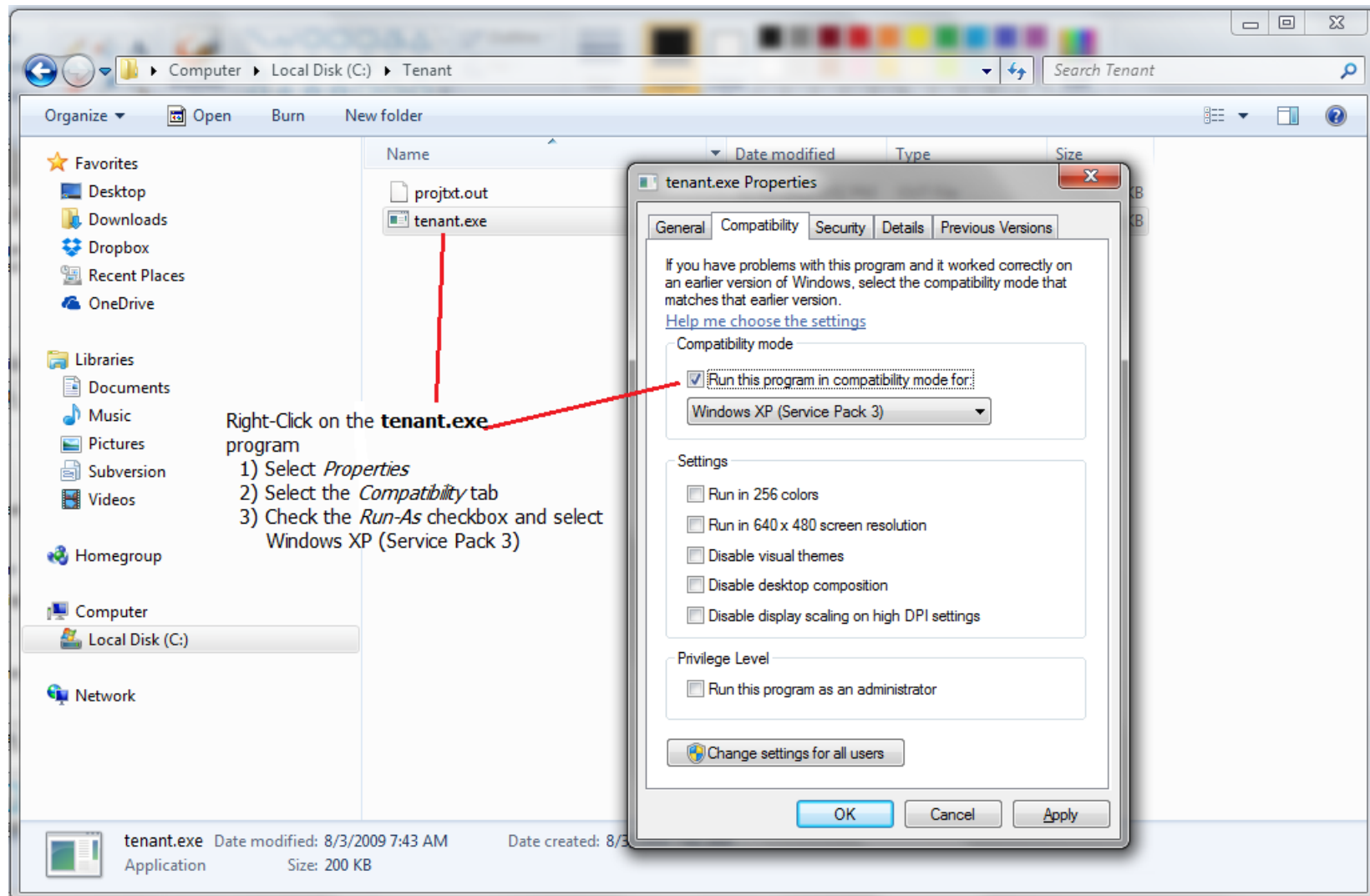
To help you complete your job, BuggySoft Inc. has set-up and configured an on-line bug / issue / enhancement / question tracking system for you to use. For the purposes of this exercise, let's simply call the bugs / issues / enhancements / questions that you may enter INCIDENTS. This on-line system is hosted at a company known as FogBugz and your login entry point is <https://sef-cc.fogbugz.com>. You can access your account by logging in using your Conestoga email address as your loginID and your student number as your password.

Pre-Assignment Activities

1. Before beginning this assignment, it is recommended that you review and remind yourself of the reason why we using bug-tracking software, what we typically want to capture in our bug-reports and why. This information is found in the second half of Module-10 (slides 15 through 31)

How Best to Proceed through this Assignment

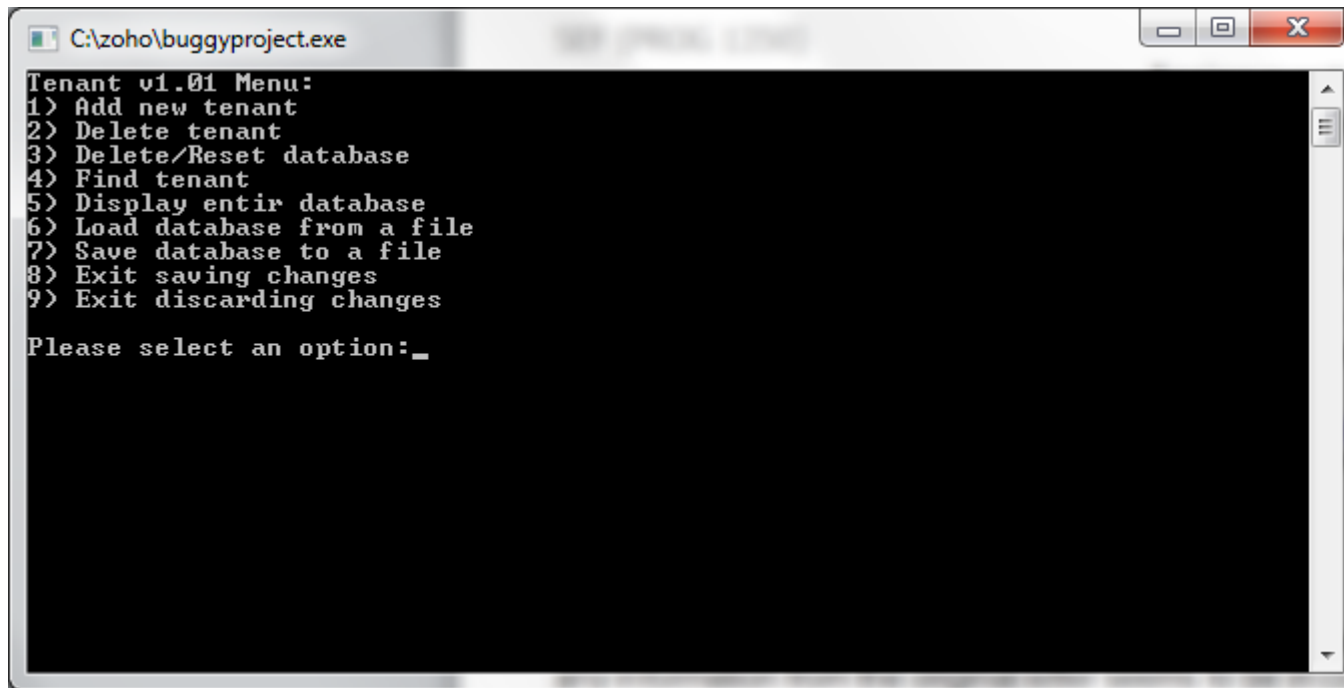
1. Before logging on to FogBugz and beginning the actual filing of INCIDENT reports, it is **highly recommended** that you spend some time **playing around with and running** the BuggySoft deliverable – a program called **Tenant (v1.01)**
 - a. You'll need to do this to better understand the *domain* and *expected behaviour* of this program because the landlords hired you to complete this task and never told you what the program you'd be testing was supposed to do ...
2. For some reason, BuggySoft Inc. developed this program to run on the Windows XP operating system!
 - a. At first you panic – but then you realize you can make this work for you on your own test machine by simply:
 - Downloading the Tenant v1.01 program and placing it in a directory on your test machine (perhaps called C:\tenant). The program to download is called BuggyProject.zip and is found in the Assign-03 area of eConestoga.
 - The ZIP file contains the program (called `tenant.exe`) as well as a sample database file call `projtxt.out`
 - And configuring the program to run in *Windows XP (Service Pack 3) compatibility mode* – as shown in the following screen capture



3. Once you have played around with the program for a while and think that you're ready to begin logging INCIDENTS against it – then you can log into <https://sef-cc.fogbugz.com> and begin
 - a. But you have been assured by the landlords that you won't find many issues with the program. BuggySoft Inc. has assured them that the program is stable and working properly [famous last words ...]
 - b. Keep in mind that the landlords are also looking to you in your Software Engineering experience of best practices to make some recommendations for enhancements of features, etc. on their behalf

Tenant (v1.01) ... The Program

1. When you first run the program you are presented with the following user interface:



```
Tenant v1.01 Menu:
1> Add new tenant
2> Delete tenant
3> Delete/Reset database
4> Find tenant
5> Display entire database
6> Load database from a file
7> Save database to a file
8> Exit saving changes
9> Exit discarding changes

Please select an option: _
```

2. You know that this program's purpose is to allow the landlord of a building to keep track of their tenants within a database file.
3. Finally one of the landlords contacts you to tell you what they remember the purpose of the program to be. The program that they originally asked BuggySoft Inc. to write was supposed to have the following features:
 - The landlords wanted a good ol' console window program to be menu driven
 - They asked for the ability to add your tenants to the database. For each tenant, they wanted to store their :
 - Full name (first and last)
 - Phone number
 - Apartment number
 - Amount of rent that they pay
 - As you read this list you're asking yourself – did they ever tell BuggySoft Inc. what format you wanted this information in?
 - Meaning for the name – are you supposed to enter *lastName*, *firstName* or *firstName lastName*?
 - For the phone number do you enter it in the format XXX-XXX-XXXX or (XXX)XXX-XXXX?
 - For the rent – do you enter the amount as \$YYY.YY or simply YYY.YY ?
 - Oh well you think – I'm sure that BuggySoft Inc. followed good practices and asked the customers what they wanted ...

- The landlords also wanted the ability to delete a specific tenant from the database (if they moved out) and/or find a specific tenant in the database (if you couldn't remember what their rent was or their phone number was and you needed to see it)
 - So in your mind, you are thinking that you would be able to use this program to find a tenant, bring up their information – be able to change it (increase their rent, change the phone number, etc.) and resave the database
 - But then again, the landlords didn't specifically ask BuggySoft Inc, for this feature – but hey – it's a logical requirement – BuggySoft Inc. should have thought of it themselves and programmed it – right ?!?! (Famous last words ...)
 - The landlords also wanted the ability to delete (or reset) the entire database to be blank – if you ever decided to kick out all of your tenants ...
 - The landlords will be viewing the tenants in the database quite a log – so logically you expect this output to be a nicely formatted table's worth of information displaying all of your tenants
 - The landlords wanted the ability to LOAD and SAVE the database file
 - And lastly – they needed the option to exit the program as well ...
4. So by now, you have played around with the program a little, you know what functionality the program was to have (thank goodness that the landlord remembered and told you) and you are ready to start formally testing it ... you are ready to start your UAT !

FogBugz ... the On-Line Reporting Tool

1. As mentioned previously, BuggySoft Inc. has created a login for you to use in the <https://sef-cc.fogbugz.com/> space
 - a. You can use your Conestoga email address as your loginID and your student number is your password
 - b. When you log into the system you should see something like the following screen capture (see Figure 1)

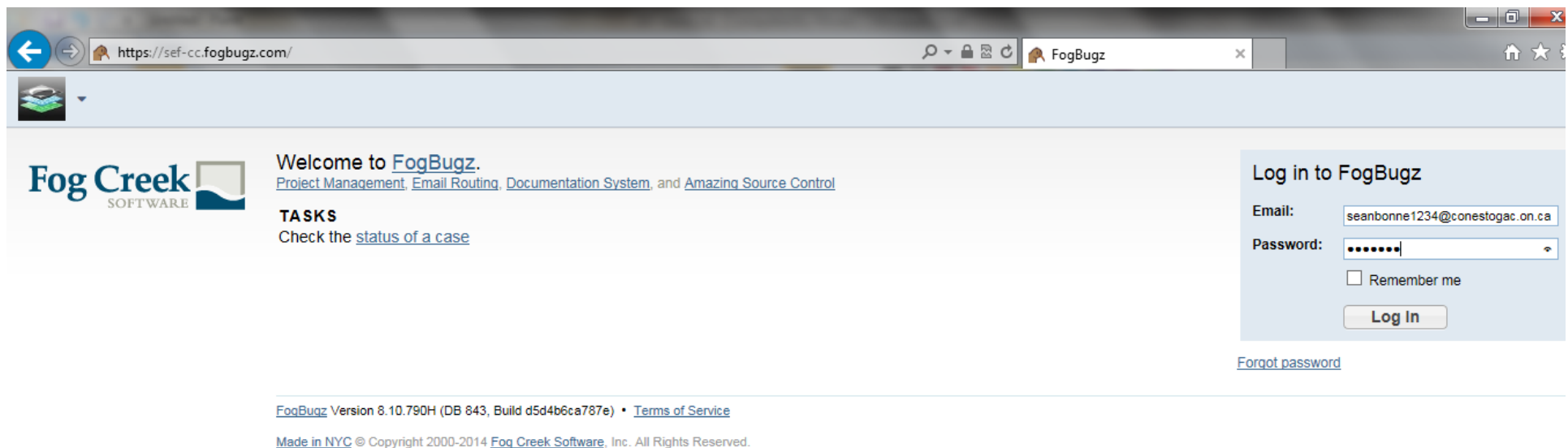


Figure 1: Logging into FogBugz

2. When you initially login, you will find yourself in the main incident summary window of FogBugz
 - a. From here you can easily see all of the INCIDENTS that you've previously logged
 - b. From here you can quickly log a new INCIDENT if you wish
 - c. See Figure 2 (below) for ideas and recommendations

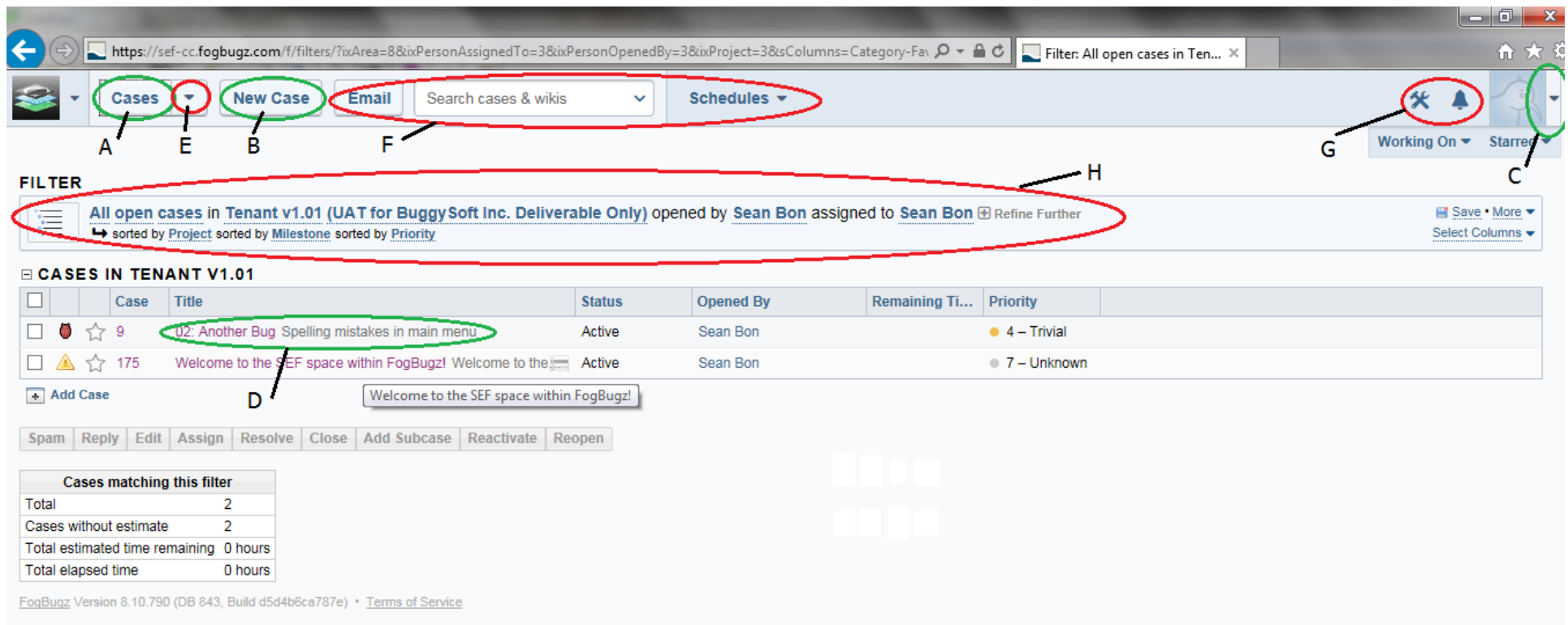


Figure 2: FogBugz Navigation

AREAS YOU CAN ACCESS

- A – The Cases button – click on this button to return to this main menu area that displays all of your current list of INCIDENTS
- B – The New Case button – click on this button when you are ready to log a new INCIDENT – it will take you to the main incident entry window (described below)
- C – The User Actions dropdown – select this dropdown menu when you are ready to log out of FogBugz
- D – Hyperlinks to existing INCIDENTS – you can click on any of the hyperlinks present in your existing set of INCIDENTS to be able to go in and read / review the information

AREAS YOU NEED TO STAY AWAY FROM

- E – You can easily *mess-up* your view of FogBugz by clicking on any of the options in this dropdown – stay away from it

- F – FogBugz offers many other integrated features that we are not using in this exercise – so please stay away from (and don't click) on any of Email, Search or Schedule buttons
- G – The tools and notifications areas – Since BuggySoft Inc. set up and configured your account – there is no need for you to go into these areas – steer clear of them – because again, they can quickly *mess-up* your view of the system
- H – Your Incident Filter settings – again since BuggySoft Inc. configured your login for you – you need to leave this filter alone. It ensures that you can see your incidents and navigate easily through this exercise

3. One of the items you'll notice in your INCIDENT list when you initially log into FogBugz is a *Welcome incident*

- If you click on the incident hyperlink, you'll be brought to the incident's expanded information – this information is a good review for you as to how you should be entering and capturing your INCIDENTS along the way (Figure 3)
- You'll want to follow these instructions and guidelines ...

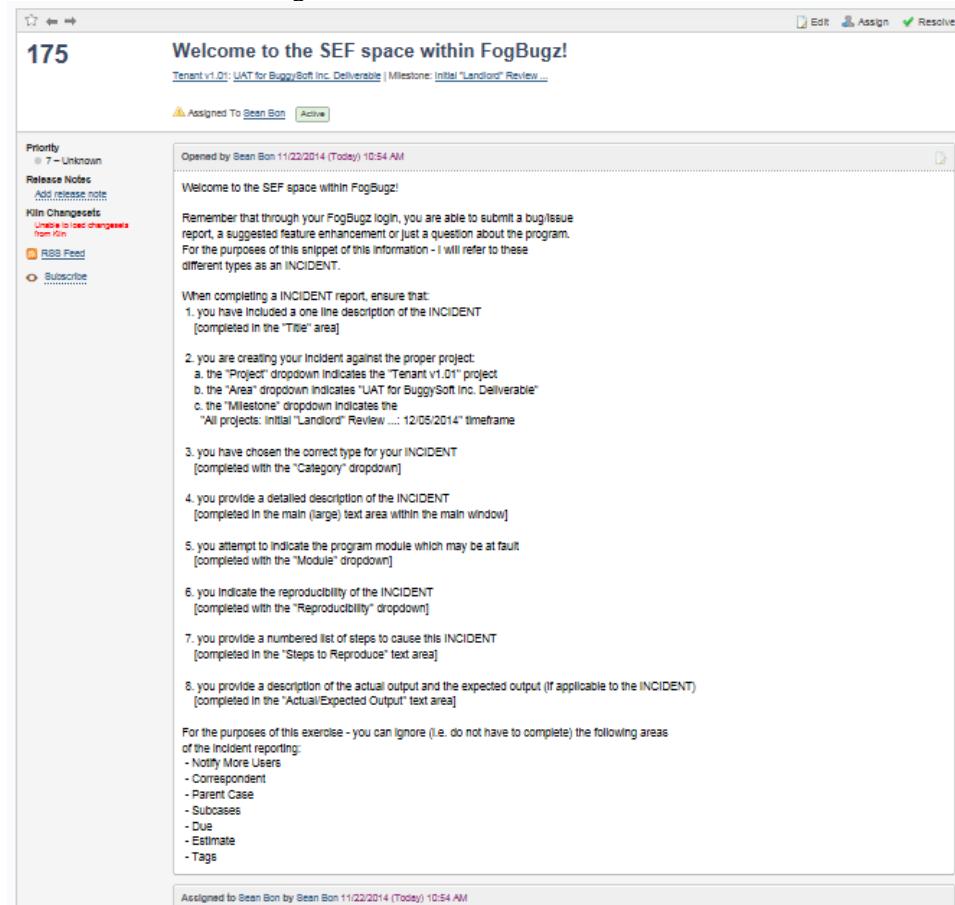


Figure 3: "Welcome" INCIDENT

4. When you are ready to log your first (or any other subsequent) INCIDENT – you can click on the **NEW CASE** button across the top navigation panel
- This will bring you to the main INCIDENT reporting window (Figure 4 below)
 - Make sure that you follow best practices (as outlined in Module-10) and enter clear, concise and detailed information about the INCIDENT at hand
 - Remember that BuggySoft Inc. will be using the information that you log to help them troubleshoot and debug (using the Scientific Method of Debugging) the issue
 - The more clear, complete and detailed information you can provide them the better chance they will have at stabilizing (recreating) the incident and determining what needs to be corrected
 - Entering an INCIDENT against the Tenant v1.01 program into FogBugz is as easy as A, B, C, D, E, F, G, (maybe) H, I, J, K and SAVE (again see capture below)
 - A – Title area – this is where you enter a one-line informative description of the INCIDENT – try to sum up the issue in a single line
 - B – The Project / Area / Milestone of the INCIDENT – more than likely, these dropdowns will be defaulted to the proper values for your INCIDENTS. If they are not, ensure that they contain the values:
 - Project – “Tenant v1.01”
 - Area – “UAT for BuggySoft Inc. Deliverable”
 - Milestone – “Initial landlord review ...”
 - C – INCIDENT Category – depending on the type of information you are capturing – use this dropdown to tell whether your information is for a
 - BUG (default value) – the program is misbehaving or not functioning correctly
 - ENHANCEMENT – you have a suggestion for a new feature or modified feature that BuggySoft should include
 - INQUIRY – a general question that you want to pose to BuggySoft about something within the program
 - OTHER – a general purpose (“catch-all”) other classification
 - D – Steps to Reproduce – in this area you want to capture (in a numbered list style) the steps that BuggySoft needs to follow in order to reproduce the INCIDENT – to be able to see what you are seeing and reporting about
 - E – Actual Output – in this area capture information about what it is that you are seeing in the program
 - If you use screen captures (see H below) to gather your output – then make a note of the screen capture attached and what it is showing here in the Actual Output area
 - F – Expected Output – in this area you capture and give details about what you actually expected the Tenant program to output in this case – make sure to give good details
 - G – INCIDENT Detailed Description – use this main text area input to capture, clear, detailed and informative information about the INCIDENT. Use this area to expand upon the one-line summary you captured in area A above
 - H – Attach Files button – if you want to take screen captures of the program as run it and come across INCIDENTS – feel free to and use this button to attach files
 - Again – just remember to make note of the screen capture(s) and what they are showing in area E above
 - I – INCIDENT Priority – use this dropdown to suggest a priority to BuggySoft Inc. for the issue being reported. As mentioned in Module 10, we need to be realistic in terms of prioritizing
 - If the program is unusable as a result of the INCIDENT, then a “critical” priority may be in order
 - If the program is usable but performance / usability is greatly reduced, then a “major” priority may be in order
 - ...

- J – Module at Fault – use this dropdown to help categorize the INCIDENT being reported in terms of what functional area (or module) of the program that this INCIDENT should be targeted against
- K – Reproducibility – use this dropdown to tell BuggySoft about your experience in trying to reproduce this INCIDENT
- SAVE – The Open button – when you've entered all of the information that you want / need to for a good, informative INCIDENT – then click the OPEN button and it will add it to your list in the main "Cases" area of FogBugz
 - If you omit something or want to change something in an INCIDENT after opening it, you can always click on the INCIDENT hyperlink in the "Cases" area to edit the incident

5. Round and round the testing wheel goes ...

a. How many bugs should you enter you ask? How many new features should you request?

- Good questions! **You have until Sunday November 30, 2014 at 11:59pm** to enter all of the bugs and features you can
- At a minimum – I would think that **entering 10 bugs and 2 requested new features (enhancement)** would be a **minimum** that you should be entering and reporting on behalf of your clients the landlords ...
 - I would expect that as a maximum you can probably stop at 20 bugs and 5 enhancements

Assignment Deliverables

Since you are using an online bug reporting tool – you have technically nothing to submit to the eConestoga dropbox for this assignment.

1. Just know that on after the assigned deadline, BuggySoft Inc. will be going into the FogBugz system and extracting all of the issues and features that you and each of the other landlords on your team have entered

2. You will be judged (and marked) based upon the :

- Number of issues that you find and the number of new features that you request (remember – there are minimum numbers)
- More importantly – you will be judged on the quality and detail of the bugs that you report

3. To do a good job on this assignment – you need to :

- **Think like a Tester** - when coming up with your set of tests to run
- **Think like a Developer** - when you come across an issue and try to determine the bug that you are seeing – what was their logic? How did they code this?
 - Being able to try and determine how they coded something will help you be able to record the issue with good detail and also be able to replicate the issue
- **Think like a Business Analyst** - when you are assessing the purpose of this program – seeing what features it does have and coming up with other, better, more useful features ...
- **Think like a Customer** - when you are expecting that the program behaves the way you want it to ...
- In Short – you need to be able to wear all of the hats that a good Software Engineer must wear ...

A Title

B Project: Tenant v1.01 Area: UAT for BuggySoft Inc. Deliverable Milestone: All projects: Initial "Landlord" Review ...: 12/05/2014

C Category: Bug Assigned To: Case Opener (Sean Bon) Status: Active

Notify More Users

D Steps to Reproduce

E Actual Output / Results

F Expected Output / Results

Correspondent

Priority: 7 - Unknown

Parent Case

Subcases

Due: mm/dd/yyyy hh:mm AM

Estimate current:

Tags

Module: --

Reproducibility: --

Opened by Sean Bon 11/23/2014 (Today) 8:51 AM

Plain text Rich text

Use Ctrl+Enter to quickly submit any case.

I Open **SAVE** **G** Cancel **H** Attach files

Figure 4 : INCIDENT Reporting Area