Bugtracker project spec/ plans

Contents

Bugtracker project spec/ plans	1
Bug tracker spec	2
Jser stories	
Database layout	3
Bugs:	3
User:	3
Posts:	
Follow:	3
Previous Bugtracker	4
Plans	

Bug tracker spec

The plan is to make an app which can log bugs and allow a user to post steps taken to fix them.

I have made an early version of this as a website but I want this to use windows forms.

The user stories assume many users can interact with the app, which I believe would be more impressive.

We are not using local .mdf files as we are using a cloud-hosted DB called ClearDb. This is being hosted on Heroku and I will walk you through the steps required to connect to this.

User stories

- 1. I want users to be able to register and log in, so a user can track bug's they've logged/worked on/ fixed
- 2. A user should be able to store details of a bug they have found, such as a title, a description, a priority etc
- 3. Users should be able to see a list of all logged bugs
- 4. Users should be able to store updates on the steps they have taken to fix a bug
- 5. Bugs should be able to have their status updated once they are fixed
- 6. A user should be able to communicate with a fix posted on their bug (in case the fix doesn't work)
- 7. Bugs should be organised into 'projects' to keep them grouped together (?)
- 8. A user should be able to take 'ownership' of a bug to prevent multiple people from working on it (? Can do away with 'solver' idea and allow multiple users to post fixes if they want)
- 9. Users contributions should be tracked (how many fixes posted etc)

Database layout

Bugs:

- Name
- Description
- Priority
- Screenshot?
- Location
- Status (unclaimed, claimed, solved, fix posted)
- timestamp

User:

- Username
- -Password
- bugs posted / fixed

Posts:

- Bug ID
- Text body
- userID
- timestamp

-

Follow:

Will allow a user to follow a bug

BugID

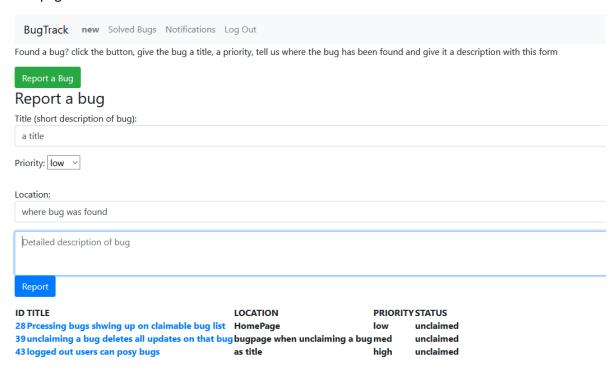
userID

I believe this will do for a basics of the DB, we can add things as neded.

Previous Bugtracker

As mentioned, I have made a version of this as a webapp, however it is only hosted locally. I pretty much finished it and it only needed some UI work to finish but I hate UI and it makes me tired just thinking about doing it

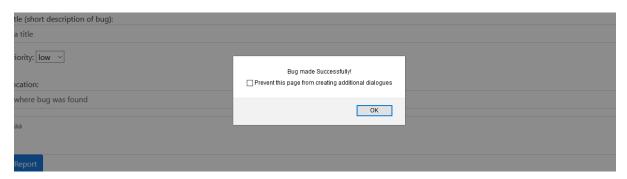
Homepage



After a user registered and logged in (features that come with the Django framework this is based on) they would see this screen. A user is able to enter details about a bug and post them.

A user can also view bugs that have not been claimed

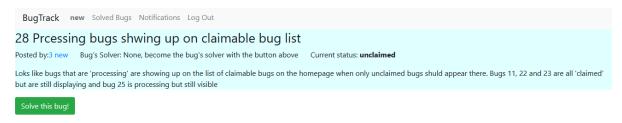
I tried to implement some kind of ownership of bugs such that a user could become a bug's 'solver' However I do not intend to keep this feature for bugtrackerv2 as it's complex and sites like stackoverflow do not use this ideology. We will discuss it.



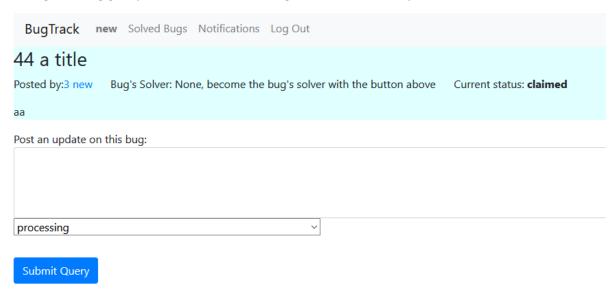
Reporting a bug gets you a success message

ID TITLE	LOCATION	PRIORITY STATUS		
28 Prcessing bugs shwing up on claimable bug list	HomePage	low	unclaimed	
39 unclaiming a bug deletes all updates on that bug bugpage when unclaiming a bug med unclaimed				
43 logged out users can posy bugs	as title	high	unclaimed	
44 a title	where bug was found	low	unclaimed	

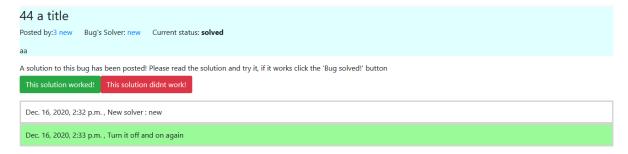
Bug now shows in the list



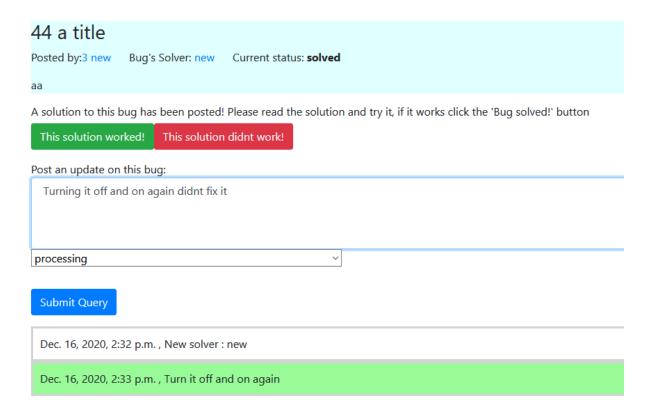
Clicking on a bug gets you the details and (if bug is unclaimed) the option to claim it



If you claim a bug then you get the option to post updates on it, also giving you the option of setting a new status for the bug



As the update's status was 'solved' The poster has the option of selecting if the 'solution' worked or not



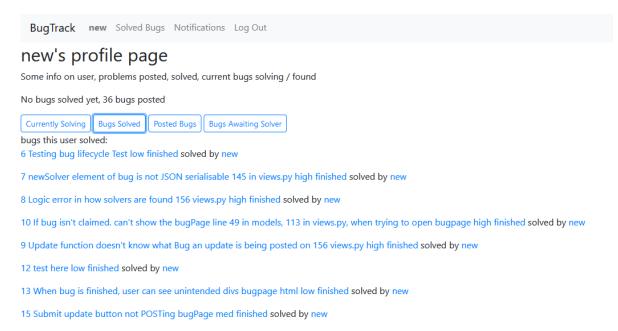
Selecting 'no' prompts a user to advise why the solution didn't work and allows the user to change the bug's status

This used to display a box underneath the 'solved' update, however I must have broken this functionality Imao

The idea was to allow some kind of communication between the bug's solver and the user who posted the bug, which is fairly important.

Clicking 'yes the solution worked' sets the bug's status to solved and takes it out of the list of bugs to be solved on the front page.

Users also have profile pages where they can see details of bugs they have posted / bugs they have solved etc



Plans

So the plan is to essentially simplify and remake the bugtracker as a windows forms application. If we can implement the functionality from the old app and make it look good then I think it would be a good project to have on your portfolio.