

Redmine - Feature #3595

Evidence Based Scheduling

2009-07-07 21:00 - George Montana Harkin

Status:	New	Start date:	2009-07-07
Priority:	Normal	Due date:	
Assignee:		% Done:	0%
Category:	Issues planning	Estimated time:	40.00 hours
Target version:			
Resolution:			
Description			
It would be great if Redmine could provide Evidence Based Scheduling similar to how FogBugz does.			
See http://www.joelonsoftware.com/items/2007/10/26.html for a description.			
Comments? Worth implementing? Problem areas that might come up?			
I might end up implementing this for fun if anyone feels it would be useful.			

History

#1 - 2009-07-09 14:06 - Ludovic Aelbrecht

This would be an extremely interesting feature - I'd personally love it if you could implement this.

#2 - 2009-07-10 06:25 - Eric Davis

This would be a good feature for a plugin. I believe the existing Plugin API should provide everything needed for this.

#3 - 2009-07-10 07:54 - George Montana Harkin

Eric Davis wrote:

This would be a good feature for a plugin. I believe the existing Plugin API should provide everything needed for this.

I'll focus my efforts towards a plugin then. Thanks.

#4 - 2009-07-10 15:17 - Hans Kazan

George Montana Harkin wrote:

I'll focus my efforts towards a plugin then. Thanks.

+1

That would be great, the ideas and concepts are something we also favor.

His idea is implemented in the Fogbugz tool, I do not know if the concept is copyrighted?
I can imagine that a look a like is feasible however.

#5 - 2010-12-01 16:56 - Jesse Katzman

It doesn't look like this is being worked on... correct?

#6 - 2010-12-01 17:06 - George Montana Harkin

This is not being worked on at the moment. Feel free.

#7 - 2010-12-01 17:55 - Jesse Katzman

I'm about to make some acronyms to reduce writing, but let's see if this is correct. Break it down!

To calculate Velocities (how accurate developers are in estimating their time):

1. Give developers without history a fake, wildly inconsistent history.
2. Keep track of Estimated Time (ET). (ET must be estimated by developer.)

3. Keep track of Actual Time (AT). (AT must include time allocated to doing task, not time actually spent on task.)
4. If a task is completed, then AT/ET is a Velocity.
5. If developer has completed a few tasks, delete fake history.

Now we need to arrive at an Evidence-Based Schedule. We'll calculate 100 possible futures for each developer.

1. Multiply each of their upcoming ETs by a randomly selected Velocity.
2. Add up total to arrive at an estimated time for finishing tasks.
3. Repeat 100 times.
4. Lay down these estimated times on a calendar that takes into account work schedule, vacations, holidays, etc.

We now have an Evidence-Based Schedule. If 95 of these estimated times takes place before a certain date, then we can say there is a 95% probability the tasks will be completed before then.

I think the hardest part for projects that don't work a 9-5 would be to lay this down on a calendar. A possible solution is to calculate an Evidence-Based Schedule using *estimated/actual completion dates* instead of *estimated/actual hours spent working* when necessary.

#8 - 2011-05-19 15:44 - Etienne Massip

- *Category set to Issues planning*

#9 - 2012-12-23 05:19 - Patrick Cummins

Does the backlogs plugin provide this?

George, I see your git repo here: <https://github.com/harking/Redmine-Evidence-Based-Scheduling>, but github is offline at the moment. Have you made any progress on this?

Thanks

#10 - 2012-12-25 01:42 - Terence Mill

It's not offline by github, its just no existing (any more).
I wrote Goerge a mail too, to asks for restore.

As far as i know the part with the past, means calculuating velocity is done by [redmine_charts2 plugin](#) well.

Missing is still the future prediction via monte carlo method. This part is more difficult, because you have to think about howto do auto selection of developers historic velocities(which projects data to take). And which user to take, the one assigned or just (a manual selected) team velocity average. There are many way to do it, which shall be supported and configurable.

#11 - 2013-07-15 10:02 - Peter Volkov

howto do auto selection of developers historic velocities (which projects data to take).

I'm not Monte Karlo method expert but as far as I know you just have to take these developer's historic data randmoly (as random as possible :). If redmine was written on python I'd delved in this task but now I have to wait for the hero!