

Position: Quality Automation Engineer

Do you like to break things?

Do you obsess about quality?

Do you take pride in fanatical users?

Is automation the first thing you think of when you do something more than once?

Do repeatable algorithms allow you to sleep better at night?

We are looking for a passionate individual that enjoys using technology and programming to enhance and automate our existing QA processes. The goal is to improve overall efficiency and increase consistency of our daily testing activities. You will be responsible for general QA testing but will be focused on efficiency and automation. You will specialize in test automation using tools such as Selenium, script languages (Ruby, Python, Powershell, etc.) and whatever else you recommend. This is the opportunity to increase your non-functional testing skills, help grow the performance testing effort, work with a variety of people to implement solutions and have fun!

#### What you'll be doing

- Working in line with development and product management to produce a high quality product
- Staying up to date with best practices and tools to automate the testing and verification of our work
- Improving, updating, maintain and continually automating our UI testing strategy
- Using Python, Ruby, Powershell and any other scripting language that provides us the highest quality automated testing
- This is not an "ivory tower" role. You must be as credible and enthusiastic rolling up your sleeves and being hands on as you are about leading and setting strategy. Those who can do.
- Writing T-SQL MS SQL Server 2008

#### The skills that would add to the stack

- JavaScript
- Frameworks such as jQuery, ExtJS, Dojo, YUI, etc...
- CSS/HTML and cross browser compatibility (CSS3, HTML5 a plus), JSON and Ajax experience
- ASP.Net MVC
- C# with .NET 3.5 or 4

#### Our Joel Test

1. Do you use source control? Yes
2. Can you make a build in one step? Yes
3. Do you make daily builds? Automated and daily builds using TeamCity and MSBuilds that deploy to demo environments
4. Do you have a bug database? Yep
5. Do you fix bugs before writing new code? Depends. Is that what the customer wants? Or do they want more functionality and are willing to leave the bugfixes for now?
6. Do you have an up-to-date schedule? Yes. We plan each iteration and each feature and project as proactively as possible
7. Do you have a spec? Yes. We have requirements and user stories.
8. Do programmers have quiet working conditions? We work in a team environment, but are working to have a quiet solitary place when you need one.
9. Do you use the best tools money can buy? Since when has cost been the best indicator of value? There's this thing called Open Source that we're really into.
10. Do you have testers? Yes
11. Do new candidates write code during their interview? Yes
12. Do you do hallway usability testing? Yes. We demo each feature to anyone who cares or is a stakeholder.