**Prueba programada QA con automatización.**

*2013. Growth Acceleration Partners*

En nuestra empresa se realizó la implementación de una aplicación para el manejo interno de Vacaciones para los empleados (<http://rocky-escarpment-5701.herokuapp.com>), sin embargo, en dicho desarrollo no se contó con la suficiente revisión de calidad, lo cual garantizara que el sitio se encontrara libre de problemas.

Debido a lo anterior, el equipo de desarrollo consideró que era necesario agregar pruebas automatizadas a la implementación actual, con el fin de lograr que las nuevas características de la aplicación estuvieran respaldadas por las pruebas y en caso de fallos, se lograra detectar los problemas a tiempo.

Ante ello, la prueba programada consistirá en la automatización de al menos los siguientes escenarios:

* Ingreso a la página y acción de entrar a la cuenta con usuario/password (sin comillas): "tester@growthaccelerationpartners.com"/"asdf1234"
* Creación de un nuevo usuario llamado "James Smith", con identificación '1234567890', fecha de ingreso '21-01-2010' y siendo el nombre del líder igual al suyo (para identificar su empleado).
* Agregar una solicitud de vacaciones al empleado James Smith, dicha solicitud será de 3 días, sin importar la fecha.
* Verificar en la parte pública que al utilizar la identificación '1234567890', se retorne el resultado correcto.

**Nota**: En todos los casos anteriores, solo se mencionaron los caminos "positivos", sin embargo, sería bueno que para todos los escenarios se valide al menos un escenario "negativo".

Para la implementación de dichos escenarios, usted puede elegir la tecnología con la que más se sienta cómodo, sin embargo, en todos los casos, se requiere que las pruebas verifiquen tanto como se pueda la información de la página y no consistan únicamente en utilizar el Selenium IDE para grabar el proceso de los clicks y presentar dicho código sin validadores adicionales. Añadido a esto, se recomienda (no es requerido, aunque sí es un "plus") la utilización de Capybara (<https://github.com/jnicklas/capybara>) con rspec, pues esta sería la herramienta con la cual se desarrollaran los casos de prueba de las aplicaciones, una vez que ingrese a la compañía.

Añadido a lo anterior, algunos recursos que podrían ayudarle a realizar la prueba eficientemente son:

* Repositorio del código fuente: <https://github.com/rrodriguez-gap/vacations_management>
* Información sobre Selenium: <http://thinkinginsoftware.blogspot.com/2010/11/acceptance-test-driven-development-atdd.html>
* Información sobre Capybara: [https://github.com/jnicklas/capybara](https://github.com/jnicklas/capybara%20) ó <http://railscasts.com/episodes/257-request-specs-and-capybara?view=similar>
* Información sobre como generar datos de prueba (capybara): <https://github.com/thoughtbot/factory_girl>

Para cerrar, para esta prueba se requerirá como entregables:  
- El código de las pruebas automatizadas (en la tecnología elegida por usted)  
- Escritura en un documento de algunos de los escenarios de prueba (en formato de Test Cases manuales).  
- Reporte de al menos un defecto que considere importante resaltar (con un formato que considere adecuado). En caso de encontrar más de uno, puede describir aquel que considere más importante y los demás los puede poner como anotaciones sencillas de una sola oración. Por el contrario, en caso de no encontrar ning{un defecto, favor escribir un reporte de alguna posible mejora por realizar a la aplicación.  
  
Cualquier consulta puede enviar un correo a [rrodriguez@growthaccelerationpartners.com](mailto:rrodriguez@growthaccelerationpartners.com) o contactarme por skype a **rarodriguezr**

**Proceso para ejecutar la aplicación en local (Windows)**

* Instalar la última versión de Ruby (http://rubyforge.org/frs/download.php/76642/rubyinstaller-1.9.3-p362.exe)
* Instalar el devkit, cuyos pasos se encuentran en: https://github.com/oneclick/rubyinstaller/wiki/Development-Kit
* Descargar el código de la aplicación desde el repositorio: https://github.com/rrodriguez-gap/vacations\_management
* En la carpeta del proyecto, editar el archivo "config/database.yml", para que coincida con la información de su base de datos local (Mysql)
* Desde una consola, entrar a la carpeta contenedora de la aplicación y digitar (líneas separadas):

"bundle install"

"rake db:create"

"rake db:migrate"

"rake db:seed"

* Luego para correr la aplicación, digitar: "rails s"
* En caso que esté usando capybara, para correr las pruebas debe digitar en la consola "rake spec"