System Tests

Sprint 1:

A. User story 1 from sprint 1: As a user I want a nice Landing Page because I want to immediately see meaningful information presented in a neat manner.
B. User story 3 from sprint 1: As a user I want the information to load quickly.

1. Go to jobstats.net in a desktop web browser.
2. The resulting page should have black navbar at the top and display job data.

A. User story 2 from sprint 1: As a user I want data from Indeed because it is a popular job site and will help me stay informed.

1. Run the scraping script from the command line.
2. There should be command line output which confirms you are scraping job listings.

Sprint 2:

A. User story 1 from sprint 2: As a user I want to view meaningful trends in job data because it will help me learn valuable skills so that I am competitive in my job market.
B. User story 2 from sprint 2: As a user I want clear visuals that represent job data because I want to easily analyse the relationships between skill sets in the tech industry.

1. Go to jobstats.net
2. There should be a graph displaying how many listings which included the keywords Java, Python, Ruby, PHP, iOS, Android were posted on Indeed each day from November 3 onward.
3. In the url enter http://jobstats.net/?keywords=mysql.
4. The graph should now display how many mysql listings were posted.

5. In the url enter http://jobstats.net/?keywords=mysql&filters=agile.
6. The graph should still display how many listings were posted each day which included mysql, but now filtered to only consider those which also included agile.

Sprint 3:

A. User story 1 from sprint 3: As a user I want a pleasant user experience because I want to use the site easily.
B. User story 2 from sprint 3: As a user I want the site to present consistent and accurate data

1. Go to jobstats.net
2. There should be a graph displaying the percentage of job postings which included the keywords Java, Python, Ruby, PHP, iOS, Android were posted on Indeed each day from November 3 onward.
3. There should be text fields above the graph which let the user add items to the graph and filter results.

C. User story 2 from sprint 3: As a user I want the site to present consistent and accurate data

1. Every day at 11:30 PM UTC, new data should be scraped from and indeed and stored in the database. A log file should be create and the last message in it should be 'done! total jobs scraped: ___'

Sprint 4:

A. User story 1 from sprint 4: As a user I want an easy to use graphical user interface

1. Go to jobstats.net
2. There should be text fields above the graph which let the user add items to the graph and filter results, those text fields should save user input after a reload.
3. Clicking the 'jobstats' text at the top right should reset graph.
4. Entering both keywords and filters and the submitting should submit both

B. User story 2 from sprint 4: As a user I want a bug free experience

1. Go to http://jobstats.net/?keywords=c%23%2Cc%2B%2B%2Cc&filters=.
2. The three c languages displayed should all have distinct graphs
3. Go to http://jobstats.net/?keywords=blablablabla&filters=.

4. The page should load properly and not crash.
5. Enter ‘data science, project management’ into the keywords textbox and submit
6. After a reload, the graph should display correctly and the keywords text box should save the full user input.
7. Enter ‘java’ into both the keywords textbox and the filters textbox, submit.
8. After a reload, the page should display one trace labeled java for which all data points are at 100%