**Micah Connelly  
WEB231  
Professor Krasso  
February 6, 2020**

**Assignment 9.4 – Client-Side Debugging**

All popular browsers today include a host of developer tools that allow for a range of benefits, including the inspection of HTML and CSS, finding errors in JavaScript functions, memory usage, and more (Mozilla). With these tools developers are able to make significant or minor changes to their web applications with minimal effort, yet enjoy significant results in stylistic comparisons, debugging minor or major problems, or simply even manipulating the DOM for immediate validation.

The Developer console allows you to do many things (Google) to improve your web development work, but three tools in particular are useful to me as a full stack developer that I do not believe any developer can easily work without:

1. The Elements tab
2. The Console tab
3. The Network tab

With the Elements tab, you can easily see and manipulate the HTML and CSS for real-time changes, which is a fast and convenient way to make styling or layout adjustments without having to touch your main code and save the changes to then refresh the page. This quick alteration allows you to easily compare and contrast different style choices or ideas to try and finalize your preference and narrow down the scope of possibilities. Then, when you’ve reached a decision, you can simply copy and paste the final code and use it to make the change in your main file(s).  
  
The Console tab is every web developer’s best friend. It’s *the* feature you really can’t work without, in this developer’s opinion. With the immediate changes you can make to the DOM, checking for errors with faulty code, or simply reading the output of objects with console.log(); to ensure your interpretation of complex objects and references is correct, all make the Console tab masterfully versatile, simple to use, and offering of insight that you simply cannot get easily from other tools.

The Network tab can be essential to comparing your APIs GET and POST calls to ensure front and back-end information is aligning how you intended. Being able to so easily review the data sent and received in its raw form can help in a process of elimination to better understanding where a breakdown in a problem can be when debugging code that doesn’t work as anticipated.   
  
For example, I inadvertently had a JavaScript model with a datatype as “String,” whereas my C# model had “Int32.” I quickly saw this difference when the POST request has quotation marks around the data that should’ve been an integer datatype, telling me that it was being treated as a string and therefore my problem was on the front-end code. I was able to update my model to reflect to match the expected datatype and resolve my bug. So, while perhaps not as versatile as the other tabs, the Network tab has its usefulness.

While there are other tabs within the Console tools that certainly serve a purpose at some time and place for all developers, these three tabs server the largest and most common needs that most developers will encounter. And therefore, they’re the three I would recommend to every new developer to add to their arsenal of skills.

**Citations**:  
  
**Mozilla**:  
<https://developer.mozilla.org/en-US/docs/Learn/Common_questions/What_are_browser_developer_tools>

**Google**:  
<https://developers.google.com/web/tools/chrome-devtools>