Kayla Verheyen

Prework 1:

So you want to be a web developer:

Web developer:

Indeed search:

1. Django Python Web Developer:
   1. “integration with **SOAP APIs**”
2. **SOAP** (originally **Simple Object Access Protocol**) is a messaging [protocol](https://en.wikipedia.org/wiki/Protocol_(computing)) specification for exchanging structured information in the implementation of [web services](https://en.wikipedia.org/wiki/Web_service) in [computer networks](https://en.wikipedia.org/wiki/Computer_network). Its purpose is to induce [extensibility](https://en.wikipedia.org/wiki/Extensibility), [neutrality](https://en.wikipedia.org/wiki/Neutrality_(philosophy)) and independence.
3. In [computer programming](https://en.wikipedia.org/wiki/Programming_language), an **application programming interface** (**API**) is a set of subroutine definitions, [communication protocols](https://en.wikipedia.org/wiki/Communication_protocols), and tools for building software. In general terms, it is a set of clearly defined methods of **communication** between various components. A good API makes it easier to develop a [computer program](https://en.wikipedia.org/wiki/Computer_program) by providing all the building blocks, which are then put together by the [programmer](https://en.wikipedia.org/wiki/Programmer). An API may be for a web-based system, [operating system](https://en.wikipedia.org/wiki/Operating_system), [database system](https://en.wikipedia.org/wiki/Database_system), computer hardware, or [software library](https://en.wikipedia.org/wiki/Library_(computing)). An API specification can take many forms, but often includes specifications for [routines](https://en.wikipedia.org/wiki/Subroutine), [data structures](https://en.wikipedia.org/wiki/Data_structure), [object classes](https://en.wikipedia.org/wiki/Class_(computer_programming)), [variables](https://en.wikipedia.org/wiki/Variable_(computer_science)), or [remote calls](https://en.wikipedia.org/wiki/Remote_procedure_call). [POSIX](https://en.wikipedia.org/wiki/POSIX), [Windows API](https://en.wikipedia.org/wiki/Windows_API) and [ASPI](https://en.wikipedia.org/wiki/Advanced_SCSI_programming_interface) are examples of different forms of APIs. Documentation for the API is usually provided to facilitate usage and implementation.

Source: Wikipedia:

1. Java full stack web developer:
   1. “Experience with **Spring MVC and Spring Batch frameworks**”
      1. The Spring Web model-view-controller (MVC) framework is designed around a dispatcher servlet that dispatches requests to handlers, with configurable handler mappings, view resolution, locale and theme resolution as well as support for uploading files. The default handler is based on the @controller and the @request mapping annotations offering a wide range of flexible handling methods.
      2. Spring Web Flow (SWF) aims to be the best solution for the management of web application page flow. SWF integrates with existing frameworks like Spring MVC, Struts, and JSF, in both servlet and portlet environments. If you have a business process (or processes) that would benefit from a conversational model as opposed to a purely request model, then SWF may be the solution.
      3. Spring batch frameworks: Spring Batch is a lightweight, comprehensive batch framework designed to enable the development of robust batch applications vital for the daily operations of enterprise systems. Spring Batch builds upon the productivity, POJO-based development approach, and general ease of use capabilities people have come to know from the Spring Framework, while making it easy for developers to access and leverage more advance enterprise services when necessary. Spring Batch is not a scheduling framework. There are many good enterprise schedulers available in both the commercial and open source spaces such as Quartz, Tivoli, Control-M, etc. It is intended to work in conjunction with a scheduler, not replace a scheduler.

Source: <https://docs.spring.io/spring-framework/docs/3.2.x/spring-framework-reference/html/mvc.html>; <https://docs.spring.io/spring-batch/3.0.x/reference/html/spring-batch-intro.html>

* 1. “Experience with unit test frameworks including **Junit, Jasmine, Karma, Mockito and PowerMock”**
     1. Junit: **JUnit** is a [unit testing](https://en.wikipedia.org/wiki/Unit_testing) [framework](https://en.wikipedia.org/wiki/Software_framework) for the [Java programming language](https://en.wikipedia.org/wiki/Java_(programming_language)). JUnit has been important in the development of [test-driven development](https://en.wikipedia.org/wiki/Test-driven_development), and is one of a family of [unit testing](https://en.wikipedia.org/wiki/Unit_testing) frameworks which is collectively known as [xUnit](https://en.wikipedia.org/wiki/XUnit" \o "XUnit) that originated with [SUnit](https://en.wikipedia.org/wiki/SUnit" \o "SUnit)
     2. Mockito: **Mockito** is an [open source](https://en.wikipedia.org/wiki/Open_source) [testing framework](https://en.wikipedia.org/wiki/List_of_unit_testing_frameworks) for [Java](https://en.wikipedia.org/wiki/Java_(software_platform)) released under the [MIT License](https://en.wikipedia.org/wiki/MIT_License).[[1]](https://en.wikipedia.org/wiki/Mockito#cite_note-gojko-1)[[2]](https://en.wikipedia.org/wiki/Mockito#cite_note-stackoverflow-2) The framework allows the creation of [test double](https://en.wikipedia.org/wiki/Test_double) objects ([mock objects](https://en.wikipedia.org/wiki/Mock_object)) in [automated unit tests](https://en.wikipedia.org/wiki/Test_automation) for the purpose of [test-driven development](https://en.wikipedia.org/wiki/Test-driven_development) (TDD) or [behavior-driven development](https://en.wikipedia.org/wiki/Behavior-driven_development) (BDD).
     3. Jasmine: **Jasmine** is an [open source](https://en.wikipedia.org/wiki/Open_source) testing framework for [JavaScript](https://en.wikipedia.org/wiki/JavaScript).[[4]](https://en.wikipedia.org/wiki/Jasmine_(JavaScript_testing_framework)#cite_note-4) It aims to run on any JavaScript-enabled platform, to not intrude on the application nor the [IDE](https://en.wikipedia.org/wiki/Integrated_development_environment), and to have easy-to-read syntax. It is heavily influenced by other unit testing frameworks, such as ScrewUnit, JSSpec, JSpec, and [RSpec](https://en.wikipedia.org/wiki/RSpec).[[5]](https://en.wikipedia.org/wiki/Jasmine_(JavaScript_testing_framework)#cite_note-5)
     4. **PowerMock** is a Java framework that allows you to unit test code normally regarded as untestable.
     5. **Karma:** The main goal for Karma is to bring a productive testing environment to developers. The environment being one where they don't have to set up loads of configurations, but rather a place where developers can just write the code and get instant feedback from their tests. Because getting quick feedback is what makes you productive and [creative](http://vimeo.com/36579366).

Sources: Wikipedia: <https://en.wikipedia.org/wiki/JUnit>; <https://en.wikipedia.org/wiki/Mockito>; <https://karma-runner.github.io/2.0/index.html>

Front end developer

Indeed search

1. Front-end Developer: Chicago IL
   1. “Knowledge of JavaScript frameworks like React.js or Node.js.”
      1. React.js: react java script library: In [computing](https://en.wikipedia.org/wiki/Computing), **React** (also known as **React.js** or **ReactJS**) is a [JavaScript library](https://en.wikipedia.org/wiki/JavaScript_library)[[2]](https://en.wikipedia.org/wiki/React_(JavaScript_library)#cite_note-react-2) for building [user interfaces](https://en.wikipedia.org/wiki/User_interfaces). It is maintained by [Facebook](https://en.wikipedia.org/wiki/Facebook) and a community of individual developers and companies
      2. Node.js: **Node.js** is an [open-source](https://en.wikipedia.org/wiki/Open-source_software), [cross-platform](https://en.wikipedia.org/wiki/Cross-platform) [JavaScript](https://en.wikipedia.org/wiki/JavaScript) [run-time environment](https://en.wikipedia.org/wiki/Runtime_system) that executes JavaScript code outside the browser. Historically, JavaScript was used primarily for [client-side scripting](https://en.wikipedia.org/wiki/Client-side_scripting), in which scripts written in JavaScript are embedded in a webpage's HTML and run client-side by a JavaScript engine in the user's web browser. Node.js lets developers use JavaScript to write Command Line tools and for [server-side scripting](https://en.wikipedia.org/wiki/Server-side_scripting)—running scripts server-side to produce [dynamic web page](https://en.wikipedia.org/wiki/Dynamic_web_page) content *before*the page is sent to the user's web browser. Consequently, Node.js represents a "JavaScript everywhere" paradigm,[[5]](https://en.wikipedia.org/wiki/Node.js#cite_note-5) unifying [web application](https://en.wikipedia.org/wiki/Web_application) development around a single programming language, rather than different languages for server side and client side scripts.
2. **Jr- Front End WEB Developer ( Entry Level):**
   1. **“**Proficient understanding of code versioning tools, such as GIT/SVN”
      1. **GIT/svn:** is a conduit for bidirectional operation between a Subversion repository and Git. It supports Subversion branches and tags, importing multiple Subversion repositories into a single Git repository, and incrementally updating the Git repository with changes from the Subversion parent. It is designed for developers who wish to contribute to projects that use Subversion, but would rather be using Git.

Source: <https://git.wiki.kernel.org/index.php/Interfaces,_frontends,_and_tools#Subversion>;

* 1. “Good understanding of SEO principles and ensuring that the site will adhere to them”
     1. **Search engine optimization is not a one-time deal.**  
        Your site should never be stagnant for long and neither should you. To keep rankings high, your content should be fresh and your site should be updated regularly. Don’t be afraid to delete things, either. If you have blog posts from two years ago that no one ever visits, delete them and fill that spot with something new and interesting.
     2. **SEO is better explained as “making it easier for search engines to find your site.”**  
        The acronym “SEO” is deceptively simple. You’re not in any way affecting the search engine itself. You’re affecting your site, which Google will then take notice of. It’s important to remember that you play by Google’s rules, not the other way around.
     3. **SEO should never be focused on getting traffic. It should be focused on getting the *right*traffic.**  
        If your SEO strategy involves getting your site in front of as many pairs of eyes as possible, you’re doing it wrong. You’ll end up wasting a lot of time and money reaching out to people that aren’t right for you. Always work hard to target the correct audience. “Anybody that uses Google” is not an acceptable target audience.
     4. **Content is king.**  
        While links will always be an essential part of your SEO, [Google is looking for relevant content](https://www.wsoaonline.com/how-content-affects-website-rankings-in-a-post-panda-environment/). You’ll want to keep your content fresh so search engines will have something new to look at. Make your content relevant and interesting to your audience.
     5. **Black hat SEO is on its way out.**  
        It’s been prevalent for a long time, but thankfully it looks like the end is coming for black hat SEO tactics. Cheap tricks and shady practices may have seemingly nice results early on, but they rarely last. Plus, it’s been proven time and time again that it is much better to be friends with Google than enemies. Google is cracking down harder than ever to try and stamp out sites that are ranking in a way they deem unethical or unnecessary.

Source: <https://www.wsoaonline.com/5-seo-principles-to-remember>;

5.