|  |  |  |
| --- | --- | --- |
| Starter Kit Topics and Resources | August 8  2016 | |
| This document is a “rough draft” listing of topics and resources that will inform the POC self-starter kits to facilitate accelerated understanding , adoption and use of Nodejs and open source tools for API development | | For POC Research and Discussion |

**Framework and Curriculum for Self-Starter Learning Kit(s) - (My Wish-List)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
| # | **Subject**  (Focus Areas / Modules) | **Topics**  (Focus Concepts)  (1-2 pages) | **Objectives**  (Learning Goals) | **Learning Resources**  (Links) |
|  |  |  |  |  |
| 1 | Self)- Starters Kit Modules | Learning Prerequisites,  APIs, Node. JS and Open Source Development Process and Tools  **Concepts (examples)**   1. **Npm** – Stands for node package manager. This will help us install the latest modules directly from the servers instead of checking for the latest version, downloading and copying to the appropriate directory. It also has other uses. 2. **Express**– Express is a highly recommended web application framework for node.js. It takes care of the low level services for us. 3. **Module**– A module is an independent, reusable piece of code. you can download them and use them directly in your project. ex: express, http, https etc. are some of the modules for node.js. | Self-Starter Kit Framework  Prerequisites  Overview of API Economy, Node. JS Concepts, Open Source Development Process and Tools |  |
| 2. | APIs | 1. API Economy 2. Restful APIs 3. Market Place 4. DevOps 5. JSON |  |  |
| 3 | Environment Setup  Basics | 1. Node.JS Server 2. Node Package Manager  * Important Modules  1. Node Application Project - Anatomy of File structure   - Key files   1. Frameworks   - Express  - Creating A Web Server With Express.js  - Istanbul.JS | 1. Environment building blocks 2. Application file structure 3. Dependencies |  |
| 4 | Tools | 1. Git / GitHub 2. DockerHub 3. Travis 4. CI 5. Logging Tools 6. Debug Tools 7. Testing Frameworks |  |  |
| 5 | Advanced Nodejs Topics | 1. Prototype 2. Callbacks 3. Inheritance 4. Synch 5. Event Loop 6. Clusters 7. Workshop / Exercises / Code Samples   **Asynchronous–** Asynchronous means an event which is happening independently of other events. The main advantage of the asynchronous approach is scalability. | Basic definitions and understanding of core concepts | Reuse selected Resources and GitHub materials as much as possible |
| 6 | Working with Databases | 1. Mongo DB 2. MySQL | **Mongoose–** Mongoose is a Mongo DB object modeling tool for node.js. It becomes easy to interact with mongo dB database server using Mongoose module.  **Db-MySQL–** It is a node.js module to interact with MySQL database. |  |
| 7 | RESTful APIs | 1. JSON 2. APIs |  |  |
| 8 | Testing Automations | 1. Jasmine.js 2. Istanbul.js |  |  |
| 9 | API DevOps Cycle |  |  |  |
|  | **Other Subjects** |  |  |  |
|  |  |  |  |  |
|  | The Front-end Web App | 1. Angular.js 2. Ember.js |  |  |
|  | The Front-end Web App |  |  |  |

**Learning Resources** (in order of preference)

**Reference Sites (**Reliable / independent Nodejs and JavaScript reference materials)

1. Node Source Of Truth - [Nodejs.Org](https://nodejs.org/en/docs/)
2. MDN - <https://developer.mozilla.org/en-US/docs/Web/JavaScript>
3. ECMAScript6 -<http://es6-features.org/#Constants>
4. Google - TBD (V8?)

**Recommended Books?**

1. From Amazon - TBD
2. [www.nodebeginner.org/](http://www.nodebeginner.org/)

**Training and Tutorials**

1. Udemmy \_reasonably priced training (<$30)
2. Express - <http://www.tutorialindustry.com/node-js-express-tutorial>
3. <https://www.airpair.com/javascript/node-js-tutorial>
4. [Nodeschool](http://nodeschool.io/index.html%20)
5. Git - [Git-it Guide](file:///C:\Users\Safee\AppData\Roaming\npm\node_modules\git-it\guide\challenges\githubbin.html)
6. [Node Hero - Node.js Project Structure Tutorial | @RisingStack](https://blog.risingstack.com/node-hero-node-js-project-structure-tutorial/)

**Blogs**

1. [Why Use Node.js? A Comprehensive Tutorial with Examples | Toptal](file:///C:\___%20%20%202016%20Projects\EA%20Initiatives\Integration%20Architecture\POC\Self%20Stater%20Kit\Why%20Use%20Node.js%3f%20A%20Comprehensive%20Tutorial%20with%20Examples%20|%20Toptal)
2. [Node.js Best Practices](http://www.codemag.com/Article/1407041)
3. <https://blog.risingstack.com/node-hero-node-js-project-structure-tutorial/>
4. [How I Structure RESTful APIs using Express 4](http://www.codekitchen.ca/guide-to-structuring-and-building-a-restful-api-using-express-4/)

**Vendor-Platform Based Learning Resources**

**Microsoft**

* Learn how to create a simple Node.js web application and deploy it to an Azure cloud service - [Build and deploy a Node.js application to an Azure Cloud Service](https://azure.microsoft.com/en-us/documentation/articles/cloud-services-nodejs-develop-deploy-app/)

A tutorial that builds on the cloud service tutorial, and demonstrates how to use the Express module. - [Build a Node.js web application using Express on an Azure Cloud Service](https://azure.microsoft.com/en-us/documentation/articles/cloud-services-nodejs-develop-deploy-express-app/)

**IBM**

* Learn how to create a simple Node.js web application and deploy it to an Azure cloud service - [Build and deploy a Node.js application to an Azure Cloud Service](https://azure.microsoft.com/en-us/documentation/articles/cloud-services-nodejs-develop-deploy-app/)
* A tutorial that builds on the cloud service tutorial, and demonstrates how to use the Express module. - [Build a Node.js web application using Express on an Azure Cloud Service](https://azure.microsoft.com/en-us/documentation/articles/cloud-services-nodejs-develop-deploy-express-app/)

**Oracle**

* Learn how to create a simple Node.js web application and deploy it to an Azure cloud service - [Build and deploy a Node.js application to an Azure Cloud Service](https://azure.microsoft.com/en-us/documentation/articles/cloud-services-nodejs-develop-deploy-app/)

A tutorial that builds on the cloud service tutorial, and demonstrates how to use the Express module. - [Build a Node.js web application using Express on an Azure Cloud Service](https://azure.microsoft.com/en-us/documentation/articles/cloud-services-nodejs-develop-deploy-express-app/)

**Amazon**

* Learn how to create a simple Node.js web application and deploy it to an Azure cloud service - [Build and deploy a Node.js application to an Azure Cloud Service](https://azure.microsoft.com/en-us/documentation/articles/cloud-services-nodejs-develop-deploy-app/)
* A tutorial that builds on the cloud service tutorial, and demonstrates how to use the Express module. - [Build a Node.js web application using Express on an Azure Cloud Service](https://azure.microsoft.com/en-us/documentation/articles/cloud-services-nodejs-develop-deploy-express-app/)