Desarrollo Web

Sin Importar Qué Camino Tomes
- Uso Básico de la Terminal
- SSH Básico
- Git Basico (github)
- Comunicación entre Cliente <-> Servidor
- RESTful Web Services / GET POST PUT DELETE requests
- Publicar tu sitio
- Dominio, DNS

Front End Básico

Fundamentos de Javascript
- Basicos
- Manipulación del DOM (queries, selectors)
- The weird parts: scope, closure, first-class functions
- JS Modular (Clases reusables)
- jQuery (Ya no es necesario)

HTML

CSS
- Basicos
- Css responsivo (Media Queries)
- Imágenes responsivas
- Layouts y Grids (Grid CSS, Flexbox)
- Animaciones Css

Front End Developer

Herramientas CSS
- Precompiladores: SASS^ / LESS / Stylus / Post CSS
- Frameworks CSS: CSS Grid / Bootstrap / Materialize
- Guias de estilo: BEM / Atomic Design / SMACSS
- Animation

Frond End Build Tools
- Webpack
- NPM Scripts
- Gulp / Grunt

JS Skills
- Type checker
  - Typescript ^
  - Flow
- ES6 (2015,2016,2017) (Babel)
- Programación reactiva: RxJS
- PWA: Service Workers, Push Notifications,
- SSR (Sever Side Rendering)

Javascript Frameworks
- Angular
  - ngrx para state management
  - rxjs
  - material
  - ivy para SSR
- React
  - Relay o Apollo para GraphQL
  - Next.js o After.js para SSR
  - CSS: Glamorous / Emotion / Styled-Components
- Vue.js
  - Nuxt para SSR
  - Vuex para state management
- Ember.js

State Management
- Flux
- MobX
- Redux
- ngrx

Unit Testing
- Jest
- Mocha
- Jasmine
- Karma
- enzyme (testing react)

Back End

Scripting Languages
- Node.js (express^, hapi^)
- Python (django, flask^)
- Ruby(Ruby on Rails, Sinatra^)
- PHP(Laravel, Lumen^)

Functional Languages
- Haskell
- Scala
- Clojure
- Elixir

High-Performance/Compiled Languages
- Go / GoLang
- Rust
- Java
- C#

Data
- Relational: MySQL, PostgreSQL
- Redis / sessions / caching
- Document: MongoDB^, DynamoDB, Couchbase, RethinkDB
- Graph: Neo4j / OrientDB
- Search Engine: ElasticSearch^, Solr
- Message Brokers (RabbitMQ^, ZeroMQ, Kafka)

Cosas a aprender en el lenguaje elegido:
- Unit/Functional Testing
- API's: RESTful / GraphQL
- Mejores prácticas de seguridad
- Authorization/Authentication: OAUTH2 / JWT
- Deploy
- Websockets
- ORM/Data management

DevOps

Server Management (balancers, instances, routers, auto scalable)
- Linux
- Ansible (Orquestration)
- Terraform ^
- Salt
- Chef
- Puppet
- IAC (Infrastructure as Code)

Cloud Providers
- Digital Ocean
- Amazon Web Services (AWS)
- Azure
- Google Cloud

Continuous Integration
- CircleCI
- Codeship
- Jenkins ^
- Spinnaker^ by Netflix
- DroneCI
- TeamCity (by Intellij) ^ *$ small project
- SemaphoreCI
- Bamboo

Docker (containerized applications and databases)
- Big Scale: Kubernetes, Mesos
- Small Scale: Docker Swarm, Docker Cloud
- Medium Scale: Rancher, Docker Datacenter/Universal Control Plane

Containers
- Docker
- Kubernetes
- Titus by Netflix*
- Monitoring