

# 강의 03<sub>/16</sub>:

## 오픈소스 기본 Tech stack

---

신정규  
2017년 3월 16일



# 기술 스택

---

내가 뭘 할 수 있는가?

# 클래스 목표

---

- Full-stack developer의 어려움
- Full-stack을 갖춰 보는 것의 필요성
  - 내가 무엇을 할 수 있는가?
  - 내가 하고 싶은 것을 만들기 위해서 무엇이 필요한가?
  - 내가 다른 사람에게 어떤 것을 요구해야 하는가?
- End-to-end 개발 프로세스
  - 간단하지만 혼자 다 할 수 있는 자신만의 도구모음

# Tech stacks

---

- 기술 스택
  - 내가 다룰 수 있는 기술들의 모음
  - '연장통'
  - 무엇인가를 하고 싶다. 어떻게?

# Tech stacks

---

- 프론트엔드 개발자
  - 백엔드 / 서버 개발자
  - 앱 개발자
    - Android / iOS / ETC
  - 연구자
  - 디자이너
- 풀스택 개발자?



# 프론트엔드 개발 (1)

---

- 눈에 보이는 부분 개발
- 디자인을 알아야 함
- 플랫폼 의존성
  - “웹”이냐 “앱”이냐?
  - 어떤 기기에서 돌아가는가?
- 서버사이드 동시 개발?
  - APM / NPM 을 사용한 프로젝트의 경우
  - Npm: node.js 패키지 매니저.



# 프론트엔드 개발 (2)

- 프론트엔드 개발 언어
  - 서버사이드와 점차 가까워지고 있음
  - 자바스크립트 언어의 변종들: Node.js / TypeScript (MS) / CoffeeScript / Dart (구글)
- 웹 (HTML5 / ES6)
  - 고전: jQuery (MS?) / Bootstrap (트위터)
  - Angular.js (구글) / backbone.js (단일 페이지 앱의 경우)
  - React / React Native (페이스북) / Polymer (구글)
- 컨테이너
  - 데스크탑 컨테이너 : electron (github)



# 백엔드 / 서버 개발 (1)

---

- 오픈소스 기반의 서비스를 만들때 가장 중요함
- 프론트엔드 개발자가 백엔드/서버도 동시 개발할 경우 (여러분):
  - 스케일링 이전
  - 프로토타이핑 때 유리
  - 리팩토링이 필요해짐
- 데이터베이스 마스터
  - 서버 사이드 개발할 때 독립적인 포지션임



# 백엔드 / 서버 개발 (2)

## ■ 서버사이드 / 백엔드 개발 언어

- 한국: Java+Spring

- 외국

- Node.js : V8 engine에 기반한 서버사이드 자바스크립트 프레임워크

- Python 3

- 다양한 프레임워크 존재. 웹 용으로 flask / Django 가 많이 사용됨

- Go

- C를 대체하는 용도로 각광받고 있음

- RoR (SNS) / Scala / Spark (머신러닝의 경우)



# 데이터베이스

---

- 무언가 “저장” 하려면 반드시 필요한 요소
- 오픈소스 데이터베이스들
  - Flat file
    - CSV
  - SQLite, IndexedDB
  - RDBMS
    - PostgreSQL, MariaDB, MySQL
  - NoSQL
    - MongoDB, Redis, Memcache (for volatile storage)

# 하드웨어

---

- 물리적 상호작용을 위한 부분
- 오픈소스 하드웨어
  - Arduino: 다양한 보드 및 규격 지원
    - Uno, Due, Mega, Leonardo...
  - Raspberry Pi: 컴퓨팅 파워를 필요로 하는 경우
    - Raspberry Zero / Raspberry Pi 3
  - ESP8266 / ESP32 기반 플랫폼: 싸고 강력하면서 현대 언어를 사용하고 싶은 경우
    - NodeMCU, IBM Bluemix

# 앱 개발 (1)

---

## ■ 모바일 프론트엔드 개발

- 프론트엔드 개발자와 협업할 수 있음
- 프레임워크에 따라 다름
  - React Native / Polymer / Cordova 등의 경우 웹 프론트엔드를 앱으로 사용할 수 있게 해 줌

## ■ 고려사항

- 반응 속도가 엄청 빨라야 하는 앱들?
- 실시간 커뮤니케이션이 필요한 앱들?



# 앱 개발 (2)

---

- 앱 플랫폼의 네이티브 언어에 대한 이해가 필요함
  - Android: Java+Android framework
  - iOS: Objective-C + Swift
  - Windows / macOS / Linux
    - Various languages
- 크로스 플랫폼
  - Java
  - 바인딩 개념 : QT / GTK / Mono?



# Problem 1:

---

데스크탑 노트 앱 만들기

- 이번 수업에서 배운 내용 및 아이디어등을 기록하는 자신만의 노트앱을 만듭니다.
- 최소 요구 사항
  - 웹 인터페이스를 사용합니다.
  - 데이터가 휘발성이면 안됩니다. (저장이 되어야 합니다)
  - 저장한 데이터를 검색할 수 있어야 합니다.
  - 적은 날짜를 기록할 수 있어야 합니다.
- 추가적으로
  - 기능이 추가될수록 가산점을 얻습니다.

# Assignment #1

---

## ■ 노트앱 기획하기

- 어떤 종류의 노트 앱을 만들어 볼지 생각해 봅니다.
- 대략적인 디자인을 스케치해 봅니다.
  - 메뉴 / 색상 / 글씨 등
- 스토리보드를 짜 봅니다.
  - 첫 화면은 어떻게 보여줄 것인지 생각해 봅니다.
  - 어떤 과정을 거쳐 어떤 식으로 페이지를 이동하는지 사용자 동선을 그려봅니다.
- 제출할 것: 기획서 및 스토리보드 (다음주 수요일 0시까지)





# Next is...

## 4/16: OSS Web Frontends – ES6-based frameworks

@inureyes

Questions? [inureyes@gmail.com](mailto:inureyes@gmail.com)

