**Sails.JS Reference**

작성자 : 양동호 (Rex)

<http://sailsjs.org/>

**<Sails.JS 란 무엇인가? >**

sails: 돛, sail: 항해하다

- 엔터프라이즈급 NodeJS 응용프로그램을 구축하기 쉽도록 MVC패턴을 모방하도록 설계되어있는 웹 개발 프레임워크!

- JSON을 사용하여 모든 DB(mySQL, MongoDB 등)들을 완전히 동일한 API를 활용해서 쉽게 접근 (Waterline을 사용)

- 기본적으로 ExpressJS 와 Socket.io 가 내장되어있다고 한다.

- ExpressJS 를 더 MVC답게, 더 쉽게 사용하도록 개발되었다. 다르게말하면, JAVA의 Servlet에 대한 프레임워크가 Spring이라면, 이것은 ExpressJS 의 프레임워크라고 할 수도 있을 것 같다. 개발이 상당히 쉬워진다고 한다.  
그러니 결국 ExpressJS 에 종속적인것이다. 하지만 NodeJS기반의 Server layer Framework는 ExpressJS 가 점령한것처럼 보여지니, 이를 더 쉽게 만들고자한 Sails.JS는 꾸준히 발전할 수 있으리라고 본다.

- Ruby on Rails 가 있다면, Node on Sails 가 있다고 말하기도 한다. Rails의 JS버전이라고도 하며, 그만큼 Ruby on Rails 의 분위기를 느낄수 있다고 한다.

**< 설치 >**

1. NodeJS 가 설치된 상태에서 아래의 명령어를 입력한다.   
npm –g install sails

2. Sails의 새로운 프로젝트를 생성한다. (testProject)  
 sails new testProject

3. Rest API 를 Generate 하기 (user). 아래의 명령어를 실행하면 User.js(model) 와 UserController.js 파일이 생성된다.  
 sails generate api user

4. 서버를 lift 한다. (Start Up Server)  
 cd testProject  
 sails lift

5. “localhost:1337/user”로 접속하면 웹소켓 호환 user의 API를 볼수 있다.

**< Sails 앱의 해부학 >**

<http://sailsjs.org/documentation/anatomy/my-app/api/models>

sails new 프로젝트명 명령어를 통해 프로젝트를 생성하면, 아래와 같은 폴더구조가 생성된다.

myApp

- api  
 - controller 🡪 Model과 상호작용하고 클라이언트에 적절한 뷰를 렌더링하기 위한 Back-end logic이 들어있는 곳  
 - models 🡪 Sails App에 대한 Data를 포함하는 구조. 컨트롤러와 Mapping이 된다.  
 - policies 🡪 App의 Controller에 대한 액세스를 인증하기위해 거쳐야하는 미들웨어 js파일  
 - response 🡪 클라이언트에게 서버 응답을 보내주기위한 logic을 담고있다. 우리들 필요에맞게 편집하자.  
 - services 🡪 서비스는 컨트롤러와 유사하지만, 일반적으로 req ~ res 사이에 일어날수 없는 일에 사용된다.  
 (예를들면, 이메일보내기, 문자보내기, 타사API에서 데이터를 불러온뒤 클라이언트에 pusing 등…)

- asset 🡪 호스팅할때 사용하는 정적 파일을 저장하는 공간. myApp/assets/newFolder/data.txt 파일은,  
 Sails lift 를 하고나면 <http://localhost:1337/newFolder/data.txt> url로 파일에 액세스가 가능해진다.  
 - images 🡪 호스팅에 사용할 정적 img 파일을 넣어두는 곳. localhost:1337/images/파일명.jpg 로 접근가능하다.  
 - js 🡪 호스팅할때 사용할 정적 .js파일을 넣어두는 곳. Socket.io 를 통해 통신한다고 하는 것 같다.  
 - styles 🡪 호스팅에 사용할 정적 .css 파일을 넣어두는 곳  
 - templates 🡪 Front-end 쪽의 html템플릿을 보관하는 곳이라고 한다. Angular 등에서 사용하는…? Grunt와 관련.

- config 🡪 Sails App을 구성하는 파일들이 있으며, Customizing 또한 가능하다.  
 - env 🡪 API 키 또는 원격DB 암호화 같은 다양한 환경설정이 포함되어있다. sails.config.enviroment 전역객체  
 - locales 🡪 다양하게 방문하는(다국적) 클라이언트를 지원할 수 있는 정보가 담겨있다.

- node\_modules

- task 🡪 Sails는 Grunt를 통해 asset을 관리한다. Grunt task에 대한 정보가 포함되어있다. Customizing도 가능하다.  
 - config 🡪 실제적으로 어떤 파일들을 어떻게 관리할 것인지 task에 대한 설정이 담겨있는 .js 파일들이 들어있다.  
 - register 🡪 어떤 task모듈을 사용할 것인지 grunt.registerTask() 메소드를 호출하는 .js 파일들이 들어있다.

- views 🡪 custom views가 모두 들어있는 디렉토리이다. myApp/config/routes.js 파일에서 경로설정을 하거나,  
또는 Controller의 logic에서 res.view() method 를 사용한다.

[Sails.js] Ruby on Rails 가 있다면 Node on Sails 도 있다 - 윤영식 블로그  
<http://mobicon.tistory.com/321>

벨리데이션에 관하여. type선정  
<https://github.com/balderdashy/waterline-docs/blob/master/models/validations.md>

워터라인과 몽고디비 결합하기  
<https://gist.github.com/wangyangkobe/70933684f2de6e24e037>