|  |
| --- |
| **Smart Monitoring System**  **전공종합설계 - 2차 설계제안서** |

|  |  |  |  |
| --- | --- | --- | --- |
| **제출일** | 2017.4.27 | **소속학부** | 컴퓨터학부 |
| **과목명** | 전공종합설계1 | **조명** | Mr.ROBOT |
| **담당교수** | 이상준 | **조원** | 20112339 김성진  20112315 강준호  20112423 유상현 |

목 차

1. Project 개요 - 7 -

1.1. Project 제안 배경 - 7 -

1.2. Project 목적 - 8 -

1.3. Project 기대 효과 - 9 -

2. System 분석 - 10 -

2.1. 기술 분석 - 10 -

2.1.1. 요소 선택기 (Element Selector) - 10 -

2.1.2. RTCS (Real-Time Communication System) - 10 -

2.1.3. Script Injection - 10 -

2.2. 유사 서비스 분석 - 11 -

2.2.1. FEEDLY - 11 -

2.2.2. EverNote-WebClipper - 13 -

3. Design - 19 -

3.1. 예상시나리오 - 19 -

4. Project 수행 계획 - 34 -

4.1. Project 개발 자원 - 35 -

4.2. Project 일정표 - 36 -

# 개요

## 제안배경

RSS(Rich Site Summary)란?

뉴스나 블로그 사이트에서 주로 사용하는 콘텐츠 표현 방식으로 웹 사이트 관리자는 약속된 프로토콜을 따라 RSS Feed를 제공하면 사용자들은 RSS리더기를 통해서 웹 사이트 관리자가 의도하는 콘텐츠에 한하여 직접 사이트에 접근 하지 않고도 정보를 습득할 수 있게 해주는 방식이다.

현재 존재하는 RSS서비스들은 구독하고자 하는 웹 페이지의 관리자가 만들어준 피드를 사용하는 것이 대부분이다. 또한 RSS피드를 제공하지 않는 웹 페이지가 대다 수이다. 사용자 입장에서는 RSS서비스를 누리고 싶은데 누리지 못하는 경우가 태반 이며, 본인이 지정하지 않은 데이터도 불필요하게 받아야 하는 불합리함이 있다. 따라서 RSS피드를 제공하지 않는 웹 페이지의 피드를 제공하고, 사용자가 원하는 섹션의 정보만을 지정하여 받아볼 수 있게 하는 시스템이 필요하다.

더 나아가 사이트 관리자에 의해서 RSS가 제공되는 경우에 있어서도 사용자의 사용용도와 적합하지 않는 정보들도 의도치 않게 정보를 받아들여야하는 상황이 생길 수 있다.

예를 들어 ‘블로터’ “www.bloter.net”에서 제공하는 RSS를 chrome extension Feedly 리더기를 통하여 보면 다음과 같은 결과를 얻을 수 있다.



위의 그림은 블로터에서 제공하는 모든 카테고리의 구분없이 새롭게 추가되는 기사모두를 시간순으로 제공하고 있다. 블로터에서 제공하는 모든 기사를 구독하고 싶은 사용자도 있는 반면 블로터에서 제공하는 ‘IT열쇳말’ 카테고리만을 구독하고 싶은 사용자도 있을 수 있다.

우리의 프로젝트는 사용자의 사소한 요구사항에도 만족할 수 있는 RSS피드를 제공함과 동시에 축적된 피드를 바탕으로 Data Crawling을 위한 Engine을 만들고자 한다.

## 목적

우리 프로젝트는 웹 환경에서 RSS피드를 제공하지 않는 웹 페이지로부터 rss피드를 제작하여, 일반적인 사용자에게 RSS서비스를 제공하는 것에서 시작한다. 기존의 RSS서비스와 달리 사용자가 원하는 페이지의 원하는 섹션을 지정하면, 그 부분의 RSS피드를 우리가 제작하여 보다 사용자 욕구에 가까운 RSS서비스를 누릴 수 있게 하는 것이다. 그리고 궁극적으로는 우리 프로젝트가 만들어낸 RSS피드를 바탕으로 크롤러 엔진을 제작하여, 크롤러 제작을 용이하게 하는 것을 목표로 한다. 우리가 살고 있는 시대는 정보의 바다를 넘어 정보의 우주라고 할 정도로 데이터가 과다한 시대에 살고 있다. 일반적인 사용자는 사용해본 적도 없는 제타바이트(ZB) 단위의 데이터가 웹 상에서 돌아다니고 있다. 몇 년째, 언론에서는 빅데이터를 외치 며 데이터의 중요성을 강조하고 있다. 뿐만 아니라 지난 해 알파고가 초래한 인공 지능이라는 화두는 더욱 더 데이터의 중요성을 우리에게 상기시킨다. 이러한 현실 속에서 일반 사용자들에게는 RSS서비스를 응용하여 사용자 맞춤형 데이터 구독 서 비스를 제공하고, 동시에 데이터가 필요한 개발자들과 관계자들에게 RSS기반 크롤러 엔진을 제공하려 한다.

## 기대효과

- RSS 피드를 제공하지 않는 사이트에서도 RSS리더기를 통해서 마치 사이트에서 RSS피드를 제공하는 것처럼 사용할 수 있다.

– 사용자의 입맛에 맞게 제작된 각사이트들의 피드들의 사용량을 바탕으로 실제 사이트운영에 RSS피드 제공의 필요성을 제고할 수있다.

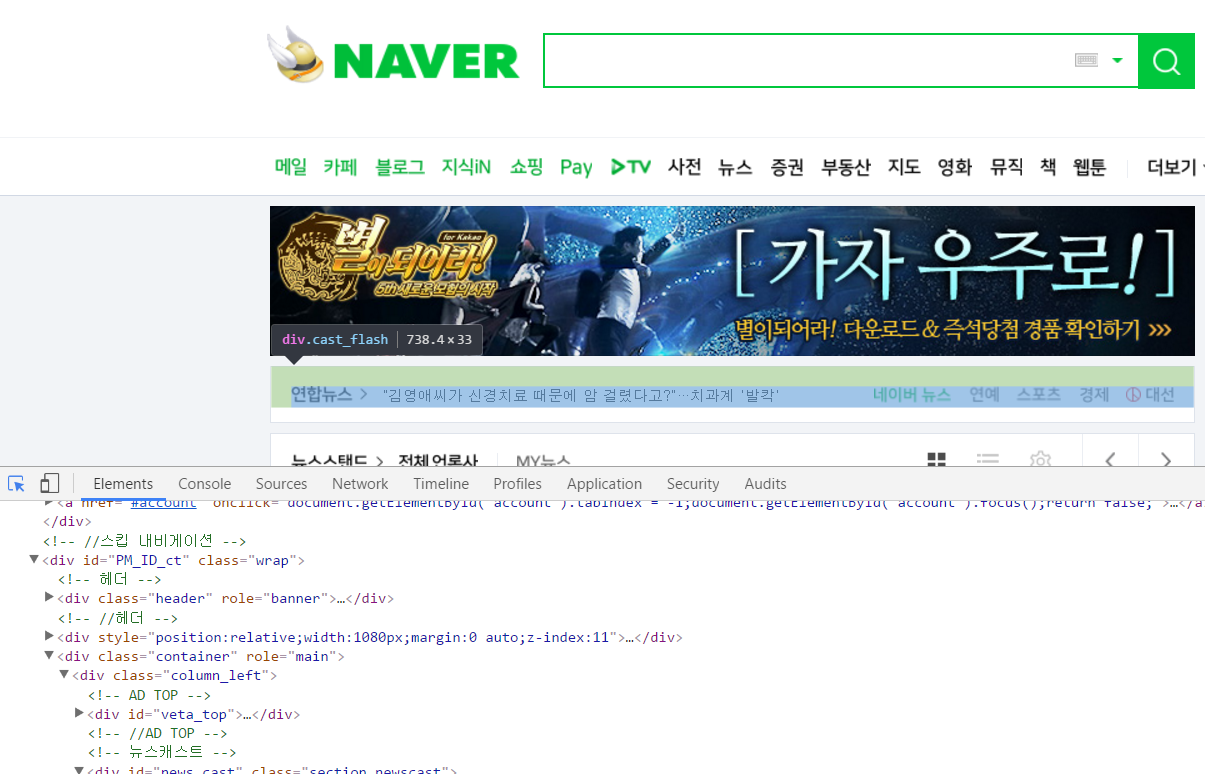
# SYSTEM 분석

## 핵심기술 분석

### 요소 선택기 (Element Selector)

**- Element Selector**

기존에 사이트에서 제공하는 Format 외에는 원하는 데이터가 있어도 RSS를 통해서 정보를 받아볼 수 가 없다. SMS는 사용자로부터 원하는 부분을 HTML 태그 단위로 쪼개어 선택할 수 있는 기능을 제공해야 한다.



Chrome 요소검사의 Element Selector

Chrome 브라우저에서 제공하는 요소검사 기능 중 HTML태그를 선택하는 기능에 착안하여 Element seletor를 구현하고자 한다. 수시로 업데이트 되는 부분을 선택해야 하기 때문에 그림에서 보이는 것처럼 마우스로 클릭한 부분만을 선택하는 것이 아닌 반복되는 Tag를 통해 출력 되고 있는 모든 요소들을 한꺼번에 선택 할 수 있도록 구현할 예정이다.

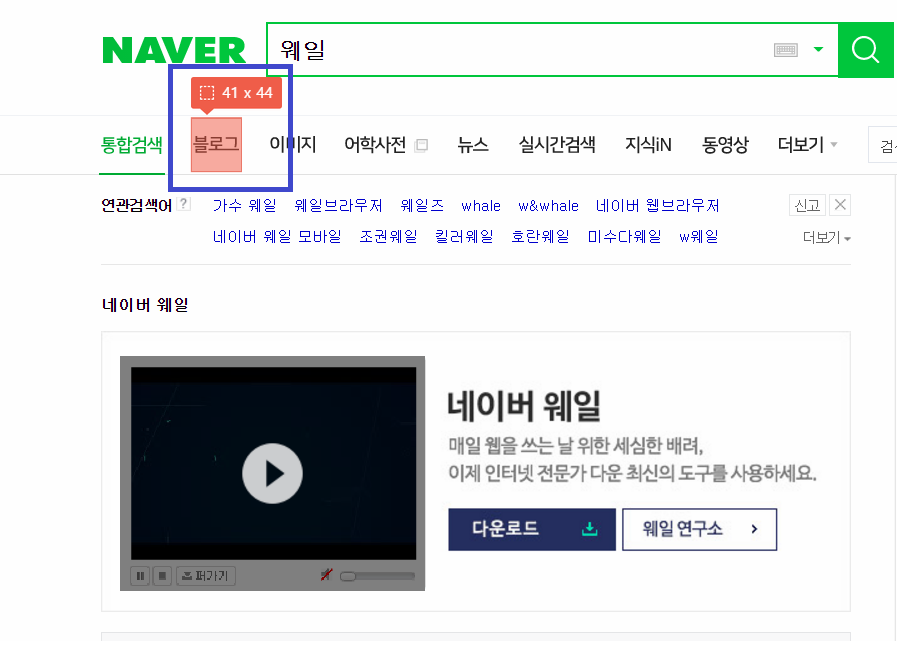
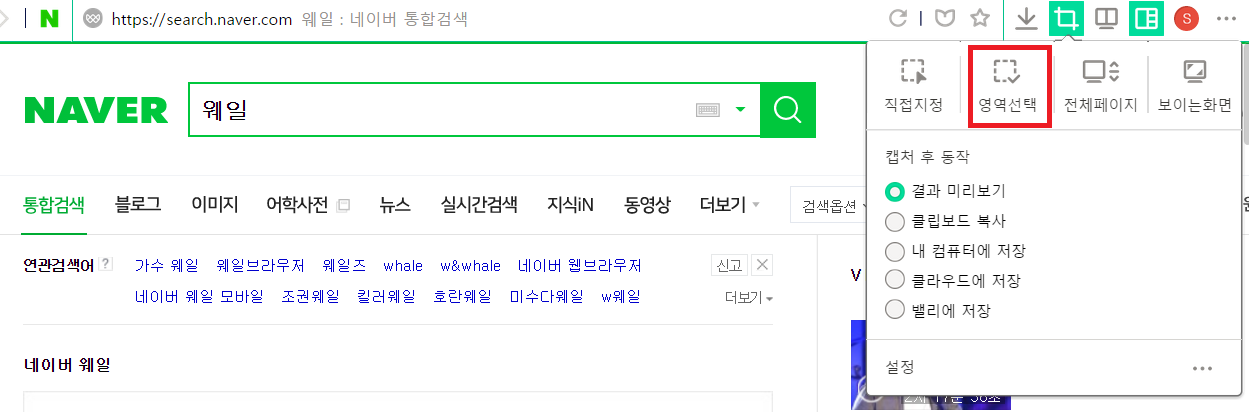


태그 선택의 예시화면

예시화면에서 보이는 것처럼 “한국 경제”부분을 클릭하면 같은 구조로 이루어져 있는 모든 태그를 선택할 수 있도록 하는 기능을 구현할 예정이다.

네이버에서 최근 공개한 Whale 브라우저의 경우를 살펴보면 Element Selector의 기능을 잘 보여주고 있다.

웨일 브라우저에서는 페이지를 캡처하는 기능을 제공하는데 여기서 영역을 선택하여 캡처하는 부분은 Element Selector 부분을 잘활용하고 있다.



위의 그림에서처럼 HTML태그별로 영역을 선택할 수 있다. 우리의 프로젝트도 HTML 태그별로 공통된 부분을 추출하는 기능을 핵심기술로 이용해야하기 때문에 이기능을 적극 활용할 예정이다.

### RTCS(Real-Time Communication System)

**-RealTime Communication System(RTCS)**

사용자에 의해서 선택된 부분에 대해서 데이터를 지속적으로 갱신해주는 것이 무엇보다도 우선시 되어야 할 기술이다. 기존에 사이트에서 RSS를 제공해주었다면 사이트에서 RSS 피드를 갱신을 해주어야만 업데이트된 정보를 받아볼 수가 있다. 하지만 SMS은 RSS 피드를 독점적으로 생산, 관리를 하기 때문에 실제 사이트에서 갱신된 정보를 받아오는 과정과 피드를 갱신시켜주는 과정을 구현해야 한다.

웹은 일반적으로 비동기방식 통신을 기반으로 하기 때문에 정보를 RealTime으로 보이기 위해서는 동기방식 기술구현이 필요하다. 현재 웹에서 사용되는 실시간 통신 방식은 몇가지가 있다.

가장기본적인 통신 방식은 Polling 방식인데 서버에 주기적으로 요청하는 방식이기 때문에 서버에 오버헤드가 발생할 수 있어 SMS에서는 WebSocket을 기반으로하는

클라이언트의 한번의 요청만으로 지속적으로 통신이 가능한 방법으로 구현하고자 한다.

### Script Injection

프로젝트의 최초 시작은 Chrome Extension에서 동작하는 형태로 시작할 예정이기 때문에

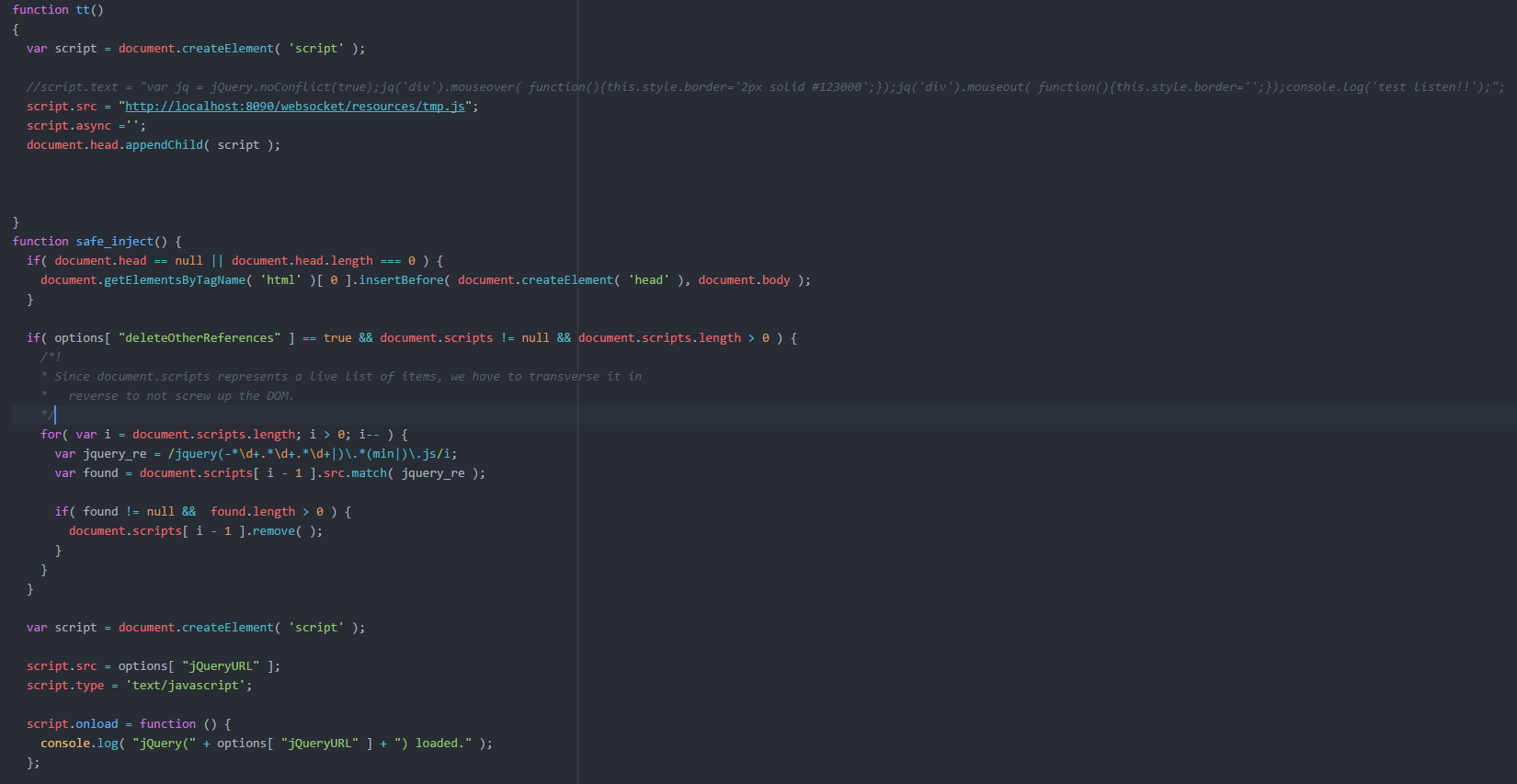
Chrome Extension에서 제공하는 Script Injection을 통해서 상용화된 페이지에서 원하는 Script를 실행할 수 있도록 하여 Element Selector를 구현한다.

위의 그림은 Extension을 통해 네이버 뉴스 페이지에 태그를 선택하는 Script를 Injection한 예시이다.



브라우저에 보여지는 페이지를 DOM객체로 접근하여 기능을 구현할 것이므로 jQuery를 먼저 페이지에 삽입 후 JQuery를 이용하여 Injection 할 기능을 개발한다.

다음은 Jquery를 삽입 후 Sample Source를 삽입한 예제 그림이다.



## 유사 서비스 분석

우리가 진행하는 프로젝트는 현재 서비스 하고 있는 RSS리더기와의 호환성을 토대로 우리의 서비스에 대한 수요군을 폭넓게 확장하고자 한다. 즉 사용자들이 RSS피드를 한눈에 보기 쉽도록 UI를 구성하는 작업에 큰 비중을 두지 않는다. 하지만 서비스 되고 있는 리더기들과의 호환을 위해 그들이 추구하는 UI의 형태로 표출할 수 있도록 피드를 제작하고자 한다.

현재 서비스 하고있는 대표적인 리더기들은 다음과 같다.

### FEEDLY

기능, 디자인, UI 면에서 세련된 점이 특징이다.

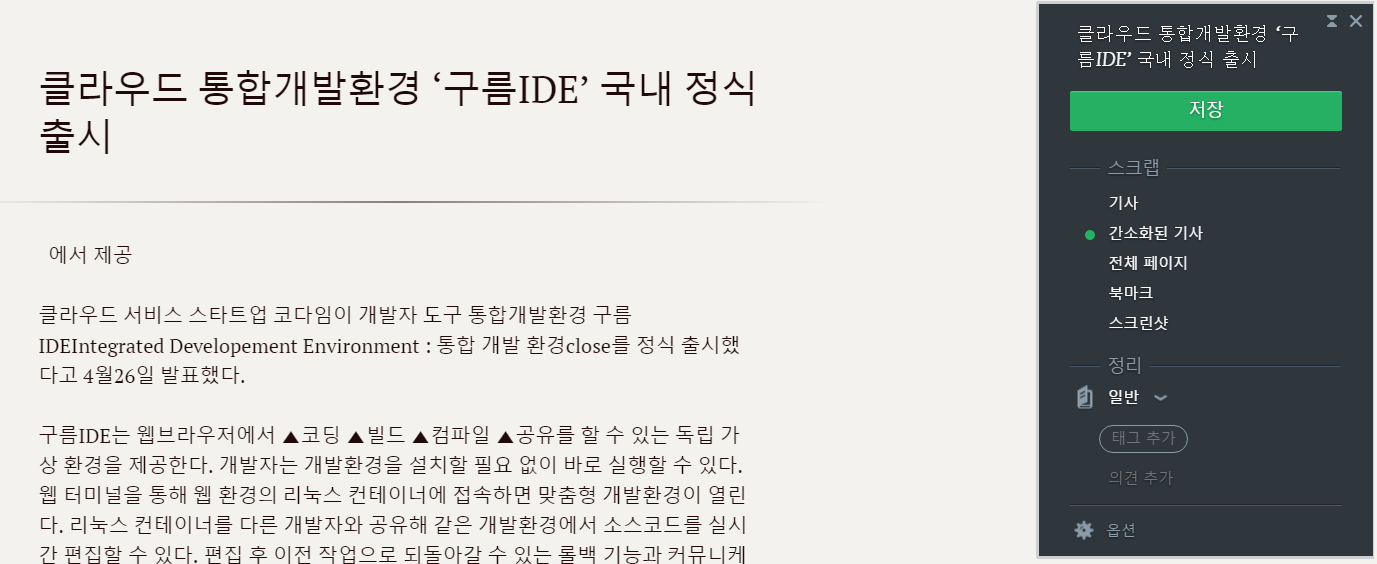


블로터 사이트에서 제공하는 RSS피드를 바탕으로 FEEDLY에서 구독하는 경우 위와 같은 UI로 제공받을 수 있다. UI를 구성하는데 사용되는 RSS피드 태그는 다음과 같다.



### EverNote-WebClipper

Web-Clipper란  텍스트, 이미지, 링크 등을 포함해 웹 페이지를 스크랩할 수 있는 기능이다. EverNote에서 지원하는 웹클리핑 기능은 현재 서비스되고 있는 웹클리퍼중에 가장 뛰어난 성능을 보여준다.



위의 그림처럼 EverNote의 웹클리핑 기능은 뉴스기사 와 같은 페이지를 자동으로 복사하여 저장할 수 있는 기능을 제공한다.

우리의 프로젝트는 웹 클리핑 기능에서 더 나아가 웹클리핑을 하기전 사전에 반복되는 구조를 서비스에 인식을 시켜둠으로써 새롭게 추가되는 정보를 자동으로 클리핑(HTML태그를 그대로 복사한다면 웹클리핑이 되고 RSS format이나 ATOM format으로 제공한다면 RSS FEED가 될것이다)할 수 있다는 점에서 차이가 있다.

# DESIGN

## 예상 시나리오

예상 시나리오와 오픈소스 리소스의 경우 팀원들간 협력을 위해 구글 스프레드시트를 사용하고있다.

MR.ROBOT 기능명세서 링크

<https://docs.google.com/spreadsheets/d/1v-TmOXf6Pf4ruMLK1TrmkFX24jVkVs-eBNfDzxkKR50/edit#gid=0>

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **대분류** | **1차분류** | **2차분류** | **시나리오** | **처리** | **비고** | **최종수정날짜** |
| 클라이언트 | 1.회원관리 | 회원가입 | 소셜계정을 이용하여 회원가입 | 소셜계정 연동 모듈로 처리 |  | 2017-04-18 |
| 로그인 | 로그인 성공시 메인페이지로 이동 | URL을 입력할 수 있는 페이지 |  | 2017-04-18 |
| 2.리드 형식선택 | . | 현재 서비스되고 있는 리더기중 본인이 사용하기 원하는 리더기가 지원하는 리드형식가 유사한 템플릿을 선택 | 현재 서비스되는 리더기들의 리드형식 자료조사가 관건! |  | 2017-04-19 |
| 3.피드생성 | 범주 선택 | 피드를 생성하고자하는 정보가 뉴스인지, 채용정보인지 등 범주를 선택하게함 |  |  | 2017-04-18 |
| 타겟 페이지 로드 | 요소 추출을 원하는 페이지 URL입력 | 렌더링 엔진을 이용하여 페이지 내부에서 자체적으로 렌더링 | 유상현 : 렌더링 엔진이 아니라 iframe형식으로 내부에서 페이지를 부르는것이 효율적이라고함 | 2017-04-19 |
| 요소 선택 | 선택된 요소와 비슷한 요소들을 선택하여 보여줌 |  |  | 2017-04-18 |
| 피드 등록 | RSS 피드XML형식으로 변환 후 서버에 등록 후 사용자에게 XML 경로를 제공 | 서버측에서 확인하기 위한 별도의 MetaData 파일 생성후 저장 | 서버에서 실제 타겟 페이지에 수정된 내용이 있는지 지속적으로 확인 하기 위해서는 RSS 피드만으로는 부족함 | 2017-04-18 |
| 4.피드편집 | 요소 변경 | 피드를 수집하는 규칙을 페이지 내의 타겟 요소를 변경하여 수정 |  |  | 2017-04-18 |
| 피드 삭제 | 수집하고 있는 XML,metaData파일을 삭제함 | 축적된 실제 데이터를 삭제하는 것은 추후에 고려 |  | 2017-04-18 |
| 피드 재등록 | RSS 피드XML형식으로 변환 후 서버에 등록 후 사용자에게 XML 경로를 제공 |  |  | 2017-04-18 |
| 5.RSS리더기와 연동 |  | 현재 서비스되는 RSS 피드와 같은 Format으로 XML파일을 제공하여 기존의 RSS리더기에 피드추가 |  |  | 2017-04-18 |
| 6.피드관리 | 리드 호출 통계 | RSS를 지원하지않는 사이트에서도 통계치를 보고 RSS지원 여부를 고려해볼 수 있음 |  |  |  |
| 데이터 추출 | 등록된 규칙에 의해 수집된 정보를 XML,JSON 등의 포멧으로 제공받음 |  |  | 2017-04-18 |
| 서버 | 피드갱신 | 타겟 서버와비교 | 등록된 페이지와 기등록된 데이터와 비교하여 변경여부가 있는지 확인 | 등록된 페이지의 평균 갱신속도를 계산하여 적절한시간으로 반복요청하게 구현할 예정 |  | 2017-04-18 |
| 추가된 데이터 반영 | 변경된 사항이 있을 시에 관리하고 있는 데이터에 반영 |  |  | 2017-04-18 |
| 데이터 축적 |  | 사용자가 선택한 카테고리별로 데이터를 축적하여 통계치 제공 |  |  |  |

# PROJECT 수행 계획

## PROJECT 개발자원

|  |  |
| --- | --- |
| **책임 및 역할** | **책임자** |
| 사용하는 오픈소스 최적화 | 강준호 |
| Server 개발 | 김성진 |
| Front 개발 | 유상현 |

오픈소스 리소스

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **분류** | **오픈소스** | **설명** | **비고** | **추가날짜** |
| 화면개발 | Bootstrap | 자바스크립트 및 CSS 오픈소스로 웹페이지를 반응형으로 쉽게 개발 할 수 있도록 함 |  | 2017-04-18 |
| jquery | 자바스크립트 오픈소스로 웹 요소(엘리먼트)를 효율적으로 찾고 수정할 수 있음 |  | 2017-04-18 |
| datatable | 수집된 정보를 웹화면서에 쉽게 개발하기 위해 사용 |  | 2017-04-18 |
| 서버측 개발 | sockjs | 실제 페이지에서 수정이 일어난 다음 빠른 시간안에 피드를 수정해주어야 하기때문에 동기방식을 이용한 통신을 하기위해 사용 | <https://github.com/sockjs/sockjs-client> | 2017-04-18 |
| gson | 서버측 구현을 JAVA를 통해 개발할 계획이므로 자바 객체를 JSON화 시키기위해사용 | <https://github.com/google/gson> | 2017-04-18 |
| mongoDB | 크로스 플랫폼 도큐먼트 지향 데이터베이스 시스템으로 각 웹사이트별로 상이한 데이터들을 정형화시키기에는 무리가있음으로NOSQL기반 몽고디비를 사용한다. | <https://www.mongodb.com/> | 2017-04-18 |

## 협업 Tool

소스코드와 문서 작업은 GitHub을 이용해 형상관리를 한다.

<https://github.com/sjkim2322/Mr_ROBOT>

## PROJECT 일정표