**캡스톤 디자인 I**

**종합설계 프로젝트**

|  |  |
| --- | --- |
| 프로젝트 명 | *IRMI(이르미)* |
| 팀 명 | *뉴로IRMI 팀* |
| 문서 제목 | 캡스톤 디자인 수행계획서 |

|  |  |
| --- | --- |
| **Version** |  |
| **Date** | 07 |

|  |  |
| --- | --- |
| **이름** | 최예인 |

|  |
| --- |
| **CONFIDENTIALITY/SECURITY WARNING**  이 문서에 포함되어 있는 정보는 국민대학교 전자정보통신대학 컴퓨터공학부 및 컴퓨터공학부 개설 교과목 캡스톤 디자인I 수강 학생 중 프로젝트 “IRMI(이르미)”를 수행하는 팀 “뉴로IRMI 팀”의 팀원들의 자산입니다. 국민대학교 컴퓨터공학부 및 팀 “뉴로IRMI 팀”의 팀원들의 서면 허락없이 사용되거나, 재가공 될 수 없습니다. |

**문서 정보 / 수정 내역**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 수정날짜 | 대표수정자 | Revision | 추가/수정 항목 | 내 용 |
| 2018-03-07 | 최예인 | 1.0 | 최초 작성 |  |
| 2018-04-09 | 최예인 | 1.1 | 내용 수정 |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**본 양식은 컴퓨터공학부 캡스톤 디자인 I 과목(산학분반)의 프로젝트 수행 계획서 작성을 위한 기본 양식입니다. 문서의 필수 항목을 제시하는 것이니 폰트, 문단 구조 등의 디자인 부분은 자유롭게 설정하기 바랍니다. 양식 내에 붉은 색으로 기술한 부분은 지우고 작성하기 바랍니다.**

**목 차**

[**1** **개요** 4](#_Toc347412182)

[1.1 프로젝트 개요 4](#_Toc347412183)

[1.2 추진 배경 및 필요성 4](#_Toc347412184)

[**2** **개발 목표 및 내용** 5](#_Toc347412185)

[2.1 목표 5](#_Toc347412186)

[2.2 연구/개발 내용 6](#_Toc347412187)

[2.3 개발 결과 7](#_Toc347412188)

[2.3.1 결과물 목록 및 상세 사양 7](#_Toc347412189)

[2.3.2 시스템 기능 및 구조 7](#_Toc347412190)

[2.4 기대효과 및 활용방안 7](#_Toc347412191)

[**3** **배경 기술** 8](#_Toc347412192)

[3.1 기술적 요구사항 8](#_Toc347412193)

[3.2 현실적 제한 요소 및 그 해결 방안 9](#_Toc347412194)

[3.2.1 하드웨어 9](#_Toc347412195)

[3.2.2 소프트웨어 9](#_Toc347412196)

[3.2.3 기타 9](#_Toc347412197)

[**4** **프로젝트 팀 구성 및 역할 분담** 10](#_Toc347412198)

[**5** **프로젝트 비용** 10](#_Toc347412199)

[**6** **개발 일정 및 자원 관리** 11](#_Toc347412200)

[6.1 개발 일정 11](#_Toc347412201)

[6.2 일정별 주요 산출물 12](#_Toc347412202)

[6.3 인력자원 투입계획 13](#_Toc347412203)

[6.4 비 인적자원 투입계획 14](#_Toc347412204)

[**7** **참고 문헌** 15](#_Toc347412205)

# **개요**

## 프로젝트 개요

IRMI는 Interactive Real-time Motion Infographics의 약자로 (주)뉴로어소시에이츠의 데이터 시각화를 위한 소프트웨어 기술 및 제품군을 지칭한다.

본 프로젝트에서 개발하려는 IRMI.js는 그 중에서 자바스크립트(JavaScript) 기술로 웹브라우저를 통해 표출 및 배포되는 데이터 시각화물을 개발, 운영하기 위한 웹 프레임워크 (Web Framework)이다.

올해 12월 말까지 개발 예정인 프로젝트이며, 산학 프로젝트 기간인 3개월 동안은 다음과 같이 프로젝트가 진행 될 예정이다.

회사에서는 IRMI.js 개발을 진행하는 동시에, 나는 IRMI.js 개발에 도움이 될 수 있도록 기존에 존재하는 여러 데이터 시각화 라이브러리들을 이용해 차트등의 데이터 시각화물을 만들고, 그 경험을 바탕으로 각 방법의 장단점들을 비교, 분석하여 보고서를 작성 할 예정이다. 그를 통해 데이터 시각화의 경험을 쌓고, 이미 존재하는 여러 라이브러리의 한계를 분석함으로써 IRMI.js를 보완하는데에 도움이 되고자 한다. 또한 대략적인 IRMI.js의 엔진이 만들어진 뒤, 그것을 이용해 앞서 제작한 것과 동일한 결과의 시각화 물을 만들고, 기존의 것들에 비해 어느 부분이 개선되었는지 분석한다. 그 후, IRMI.js를 더 보완하는 과정에 참여하여 추가적인 차트 기능 구현 등을 할 예정이다.

## 추진 배경 및 필요성

날이 갈수록 데이터는 방대해져간다. 하지만 데이터가 주는 결과를 이해하지 못한다면, 데이터의 유용성은 떨어진다.

정부가 운영하는 공공데이터포털(www.data.go.kr)의 데이터를 종류별로 구분한 결과, 전체 1만6880개 데이터 중 88.5%(1만4939개)가 재가공이 필요한 파일데이터로 확인됐다. 이처럼 국내에서는 많은 공공데이터가 쌓이고 있지만, 이를 활용하지 못하고 그대로 방치되는 경우가 많다.

따라서 데이터의 분석 결과를 쉽게 이해할 수 있도록 시각적으로 표현해주는 데이터 시각화가 필요하다. 단순한 데이터의 나열보다 한눈에 알아볼 수 있는 사진 한 장이 더 효과적이기 때문이다.

하지만 그런 방대한 양의 데이터를 한눈에 알기 쉽게 시각화하는 데에 있어 많은 시간과 노력이 필요한 것이 현재 데이터 시각화의 현실이다.

- 지금까지의 데이터 시각화 구현방법

1. 직접 Javascript 웹 라이브러리를 이용한 코딩
   1. D3.js (장기간 전문적인 학습이 필요한 웹 기술로 개발 인력 및 기간 자원 확보 부담)
   2. 자유도 높은 결과물(단, 라이브러리에서 지원하는 한계 내에서만)
   3. 데이터를 연동하는 코드는 별도의 라이브러리로 구현해야 함.
   4. Public 영역에 내놓을 결과물이 되려면 매우 많은 코딩량 필요.
2. 차트 라이브러리 이용
   1. chart.js , Google Chart
   2. 자유도가 떨어지는 결과물(원하는 디자인이 나오기엔 약간의 무리가 있음)
   3. 데이터를 연동하는 코드는 별도의 라이브러리로 구현해야 함.
   4. Public 영역에 내놓을 결과물이 되려면 여전히 많은 코딩량 필요.
3. 솔루션 이용
   1. Tableau, Power BI
   2. 코딩없이 빠른 속도로 결과물 배포 가능
   3. 데이터 연동이 가능하지만, 해당 도구만을 위한 학습 필요
   4. 기업 내부 같은 Private 영역에서 해당 데이터를 이해하는 사람에게만 도움이 되는 결과물

따라서 위의 단점들을 보완해줄 수 있는

# **개발 목표 및 내용**

## 목표

데이터 시각화물을 개발 및 배포하기 위해 최소의 웹 개발 지식만으로도 구현 가능한 웹 프레임워크를 개발한다.

IRMI.js를 사용해 빠른 속도로 자유도가 높은 웹용 데이터 시각화 결과물을 구축하게 도와준다.

D3.js 같은 경우는 해당 라이브러리를 제대로 학습하고 익힐때까지 오랜 시간이 걸리기 때문에, 코드로 하나하나 구현해야 하는 부분을 각각의 함수로 만들어 사용자가 선언을 해주는 것만으로도 해당 차트 등을 구현할 수 있게 하는 것이 목표이다.

여기서의 개발 목표는

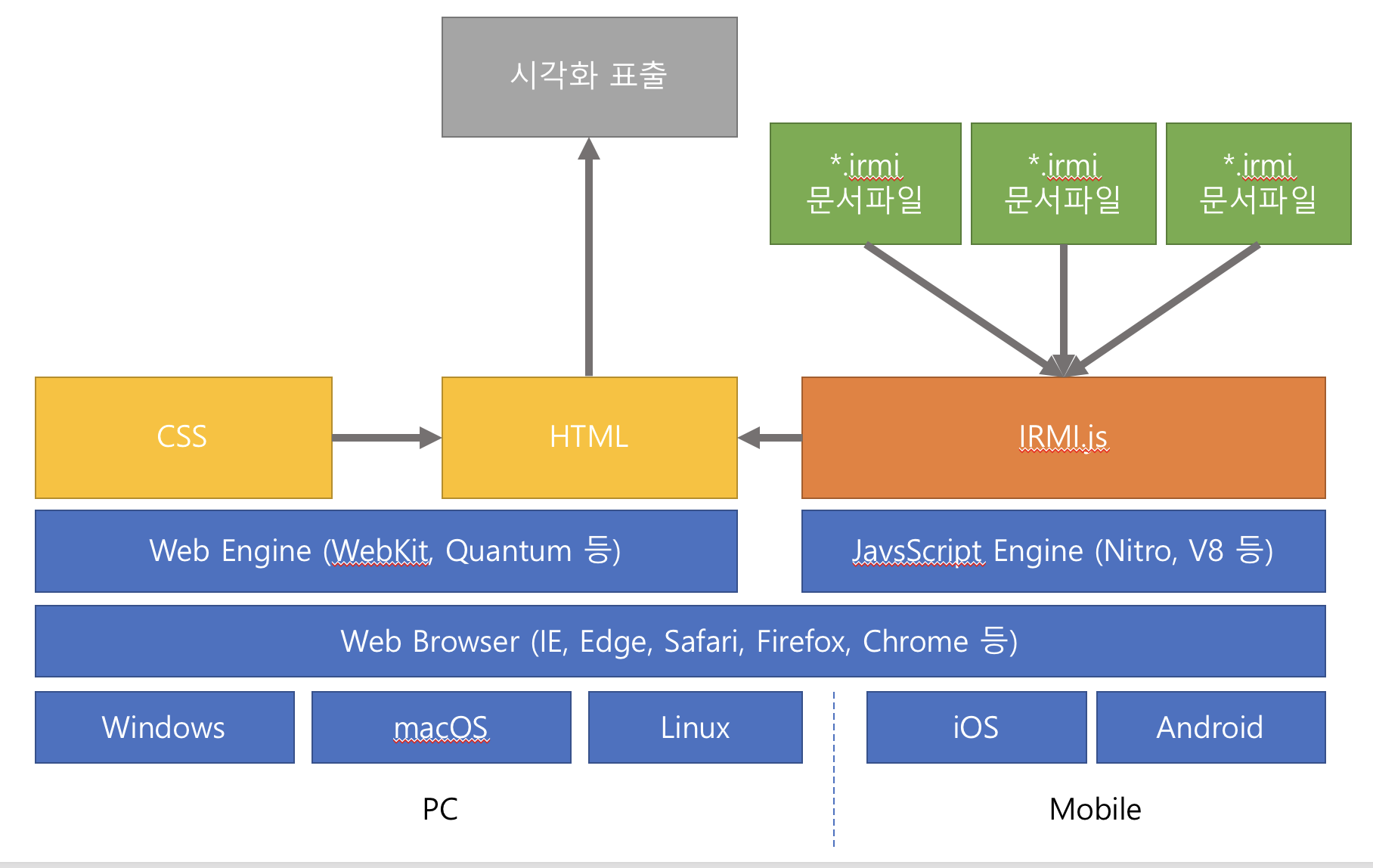
## 연구/개발 내용

앞선 목표를 구체화시켜 다음의 구체적인 내용이 가능하도록 한다.

* 1. JavaScript로 개발하는 코드량을 줄이도록 다기능의 라이브러리 제공.
  2. 직접적인 JavaScript 코딩 없이도 구축 가능.
  3. 데이터 시각화를 정의하는 별도의 파일 포맷(.irmi)를 규정.
  4. 데이터 디자이너가 irmi 파일에 데이터 소스(URL, API)와 차트 정의만 하면 작동하도록 함.
  5. Chart
     + 1. bar, line, area를 포함해 일반적으로 많이 활용되는 차트 유형들
       2. 디자이너가 Adobe Illustrator 등에서 디자인 후 생성 가능한 SVG 이미지 표출 내용과 연동시켜 데이터 시각화하도록 함
       3. 각 차트 내에서 커스텀 디자인을 위한 조건 유형들을 서술하고 나열.

1. Sample Code
   1. 각 차트 및 데이터 유형에 따라 참고할 수 있는 데이터 결과물 제공.
2. 소개 웹사이트
   1. GitHub Page를 사용해 본 결과물을 공개하고 소개하는 웹사이트 제공..

위의 내용의 결과물을 응용되는 데이터 시각화 시스템은 다음과 같은 구조로 작동함.

****

개발을 위한 방법은 세부적으로 Agile 방법론을 최대한 따르며, 매주 1회 이상 기업과 학생 간 오프라인 협의 및 짝(pair)코딩을 진행할 예정이다.

개발의 효율성을 높이기 위해 업무할당(Trello), 코드 형상관리 도구(GitHub, BitBucket)와 해당 서비스가 제공하는 이슈 트래커, 그리고 이들를 연동해 온라인 협의를 할 수 있는 메신저 도구(Slack)를 적극적으로 활용할 계획이다.

## 개발 결과

**웹 프레임워크(Web Framework)**

1. TypeScript로 작성된 IRMI.js 원 소스코드
2. Javascript로 작성된 IRMI.js 배포용 코드 및 압축형(min.js) 라이브러리 코드
3. 활용 예제 프로젝트
4. 활용을 위한 기술 문서
5. 소개 웹사이트

**대조군 비교 보고서**

동일 데이터 시각화물을 만들기 위해 기존 기술 대비 구축 단계의 편의성을 비교할 수 있는 객관적 지표를 제시한다. 해당 보고서는 결과물을 소개하는 별도의 웹사이트를 통해 공개할 예정이다. 비교 대상은 D3.js와 Chart.js를 설정하였다.

1. 데이터 시각화 결과물 구축 과정(단계)
   1. 기존 기술은 HTML, CSS, JavsScript 파일을 직접 생성하고 필요한 내용을 모두 코딩하는 개별 단계가 필요함.
   2. 본 결과물은 기본으로 제공하는 HTML, CSS, JavsScript 조합물(Skeleton, Templete)에 JSON 형식의 IRMI 포맷 파일에 요구사항을 명시하는 최소의 단계로 제작이 가능함. 비교 결과는 각 단계의 절대적인 수와 반복적인 테스트를 통한 각 단계별 제작 시간의 통계를 제시함.
2. 구축에 필요한 사전 배경 지식
   1. 타 기술은 HTML, CSS, JavsScript를 모두 알고 해당 라이브러리 사용법을 학습해야 사용 가능함.
   2. 본 결과물은 약간의 HTML에 대한 이해와 JSON 포맷을 읽을 수 있다면 바로 활용 가능함.
3. 기본 제공 차트 종류
   1. D3.js는 기본 제공 차트 없이 직접 코드로 구현 또는 복잡한 타 응용 결과물을 학습 후 사용이 가능하다. Chart.js는 기본 8종의 차트를 제공한다.
   2. 본 결과물은 보다 나은 디자인 완성도의 10종 이상 차트를 기본으로 제공한다.
4. 커스텀 자유도 및 난이도
   1. D3.js는 자유도가 매우 높으나 해당 코딩을 위한 자체적인 제공 함수 학습이 필요하고, 복잡성 높은 코드를 직접 작성해야 함.
   2. Chart.js는 기본 제공 차트의 정해진 요소를 수정해 사용 가능하도록 제약이 되어있고, 해당 코딩은 JavaScript를 이해하는 수준에서 난이도가 낮은 편임.
   3. 본 결과물은 JavaScript 코딩 없이 디자이너가 제공하는 SVG파일을 대상으로 JSON 규칙을 따르는 IRMI 자체 문서 포맷에 명시하는 것만으로 커스텀 시각화 결과물 제작이 가능함. 디자이너의 의도를 온전히 살리고, 시각화 제작자의 코딩 업무를 최소로 억제할 수 있음. 비교 결과는 코드의 양으로 제시 가능.
5. 동일 데이터 시각화를 대량으로 업데이트할 경우 소요 시간 및 CPU 부하
   1. Virtual DOM을 활용해 더 빠른 속도로 업데이트하는 결과를 제시함.

## 기대효과 및 활용방안

(주)뉴로어소시에이츠 명의의 계정으로 GitHub 웹사이트에 오픈소스로 공개함으로써 다양한 데이터 시각화 필요가 있는 곳에서 활용 가능하도록 한다.

IRMI Studio 등 이후 진행할 프로젝트에 기반 기술로 활용될 수 있다.

(주)뉴로어소시에이츠의 데이터 시각화 구축 프로젝트 사업에 솔루션으로 제안 및 사업 진행한다.

# **배경 기술**

## 기술적 요구사항

프로젝트를 개발하는 데 필요한 환경

* macOS 또는 Windows 10의 운영체제
* Visual Studio Communition Edition 또는 Visual Studio Code의 개발도구
* TypeScript Compiler의 컴파일 환경
* TypeScript 개발 언어 (최신 버전 2.7)
  + 모든 개발 코드는 TypeScript를 사용함.
  + 기존 JavaScript로 배포되는 타 라이브러리와 연동시 TypeScript에 연결 할 수 있는 정의(Definition) 파일(.d.ts)를 작성해 사용한다.
  + 그밖의 문법적 내용은 다음의 주소를 참고한다.
    - http://www.typescriptlang.org
* Chrome, Firefox Developer Edition (디버깅용 웹브라우저 도구)

프로젝트 결과물을 확인할 수 있는 환경

* macOS 또는 Windows 10, Linux의 PC용 운영체제 또는 iOS, Android의 모바일용 운영체제
* 해당 운영체제에서 구동가능한 웹브라우저
  + Chrome, Safari, Firefox, Internet Explorer, Edge
* 필요시 데이터가 제공되는 오픈API에 접속하기 위한 인터넷 연결 환경
* 필요시 자체적으로 구현한 데이터 제공 오픈API 웹 서버

## 현실적 제한 요소 및 그 해결 방법

### 하드웨어

웹 브라우저를 통해 표출되는 데이터 시각화물을 위한 기술이기 때문에, 각종 웹 브라우저(Internet Explorer, Edge, Chrome, Safari, Firefox)의 모든 구동 환경에서 문제 없이 작동해야한다

.

따라서 이들 웹 브라우저 테스트 환경을 최대로 확보하기 위해 windows PC와 애플의 macOS, 리눅스 구동 시스템 뿐만 아니라 iOS와 안드로이드의 모바일 환경 기기들도 구비되어야 한다.

현재 본사에서 제공하는 기기를 활용해 문제를 해결하고자 한다.

### 소프트웨어

웹 브라우저는 Just-in-Time 컴파일러를 활용해 스크립트 언어인 Javascript 코드를 번역해 사용하고, HTML 코드에 의해 나타나는 모든 요소들을 대량의 Tree 구조 내에서 변경하는 접근해서 느린 성능 및 모바일 환경의 배터리 효율을 떨어뜨릴 수 있다.

이를 극복하기 위해 React와 Vue.js 웹 프레임워크들에서 선보인 Virtual DOM 개념을 본 프로젝트에서도 구현시켜 웹 페이지의 SVG 영역 안에 나타나는 요소들을 갱신시킬 때 효율성을 높이고자 한다.

JavaScript는 낙관적 사고방식에 근거해 메모리를 비대하게 사용하고 객체지향의 Class 개념을 지원하기 보다는 Instance에 요소들을 자유롭게 정의하고 추가할 수 있는 방식으로 개발된 코드의 안정성을 확보하기 위한 추가적인 코딩에 의하여 복잡성이 올라간다.

본 프로젝트에는 생산성 및 무결성 환경을 위해 TypeScript 기술을 최대로 활용하고자 한다.

이를 사용하는 데는 TypeScript 개발 환경을 기본 지원하는 Microsoft의 Visual Studio 상용 버전을 사용해 비용이 발생할 수 있다.

대안으로 오픈소스 개발에 사용 가능한 라이센스를 제공하는 무상의 Visual Studio Community Edition을 사용하거나 TypeScript 개발도구를 개발로 설치하고 무료 오픈소스 에디터 도구인 Visual Studio Code을 사용해 문제를 해결하고자 한다.

기존 jQuery 등의 기술 사용을 배제하고 JavaScript 표준(ECMA Script 2015)에서 제공하는 기술들을 직접 활용해 구현함으로써 오픈소스 사용에 따른 라이센스 문제를 해결하고자 한다.

또한 본 프로젝트 결과물(IRMI.js)은 오픈소스로 공개함을 목표로 함으로써, 만약 타 오픈소스 활용에 의한 라이센스 제약이 발생하더라도 명시 후 제약에 의한 공개로 문제를 해결하고자 한다.

### 기타

# **개발 일정 및 자원 관리**

## 개발 일정

****

****

## 일정별 주요 산출물

|  |  |  |  |
| --- | --- | --- | --- |
| 마일스톤 | 개요 | 시작일 | 종료일 |
| 계획서 발표 | 개발 환경 완성 (visual studio code 설치), Javascript 및 TypeScript 학습  **산출물 :**   1. 프로젝트 수행 계획서 2. 프로젝트 기능 일람표 | ~ | 2018-03-09 |
| 경쟁제품 비교분석 | 경쟁제품을 사용하여 시각화 데모 개발  **산출물 :**   1. 경쟁제품 비교분석 보고서 2. 시각화 데모 | 2018-03-09 | 2018-04-12 |
| 1차 중간 보고 | 경쟁제품을 사용하여 시각화 데모 개발  **산출물 :**   1. 프로젝트 1차 중간 보고서 2. 프로젝트 진도 점검표 3. 1차분 구현 소스 코드 | 2018-04-12 | 2018-05-17 |
| 2차 중간보고 | 초기 IRMI.js 웹 프레임워크를 이용한 데이터 시각화 결과물 제작  **산출물:**   1. 프로젝트 2차 중간 보고서 2. 프로젝트 진도 점검표 3. 2차분 구현 소스 코드 | 2018-05-17 | 2018-05-29 |
| 최종 보고서 | 최종 보고  **산출물:**   1. 경쟁제품 비교분석 보고서 2. 시각화 데모 3. 초기 IRMI.js 웹 프레임워크를 이용한 데이터 시각화 결과물 4. 버그 확인 및 개선점에 관한 보고서 5. 개발자용 문서 및 소개 웹사이트 | 2018-05-29 |  |