**캡스톤 디자인 I**

**종합설계 프로젝트**

|  |  |
| --- | --- |
| **프로젝트 명** | *마이아포(My Eye-tracking Portfolio)* |
| **팀 명** | *식스에이드(Six-Aid) 팀* |
| **문서 제목** | 계획서 |

|  |  |
| --- | --- |
| **Version** | 1.6 |
| **Date** | 2020-MAR-24 |

|  |  |
| --- | --- |
| **팀원** | 최 락준 (조장) |
| 우 승민 |
| 심 재욱 |
| 권 강민 |
| 윤 준호 |
| 황 채은 |

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

**문서 정보 / 수정 내역**

|  |  |
| --- | --- |
| **Filename** | 3조 계획서 - 마이아포.docx |
| **원안작성자** | 최락준, 우승민, 심재욱, 권강민, 윤준호, 황채은 |
| **수정작업자** | 최락준, 우승민, 심재욱, 권강민, 윤준호, 황채은 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 수정날짜 | 대표수정자 | Revision | 추가/수정 항목 | 내 용 |
| 2020-02-26 | 전원 | 1.0 | 최초 작성 | 계획서 개요 작성 및 역할분담 |
| 2020-02-28 | 전원 | 1.1 | 내용 추가 | 개발 목표 및 내용 추가 |
| 2020-03-01 | 권강민 | 1.2 | 내용 수정 | 개요 추가 및 개발 필요성 추가 |
| 2020-03-03 | 심재욱 ,최락준 | 1.3 | 내용 수정 | 개발 목표 및 내용 정리, 내용 추가 |
| 2020-03-11 | 황채은 | 1.4 | 내용 수정 | 배경 기술 작성, 역할분담 수정 및 비용 적시 |
| 2020-03-21 | 권강민 | 1.5 | 내용 추가 | 참고 문헌 정리 |
| 2020-03-24 | 전원 | 1.6 | 내용 정리 | 계획서 정리 및 내용 검토 |

**목 차**

[**1** **개요** 5](#_Toc35983237)

[1.1 프로젝트 개요 5](#_Toc35983238)

[1.2 추진 배경 및 필요성 5](#_Toc35983239)

[1.2.1 기술의 시장 현황 7](#_Toc35983240)

[1.2.2기 개발된 시스템 현황 10](#_Toc35983241)

[1.2.3 기 개발된 시스템의 문제점 11](#_Toc35983242)

[1.2.4 개발할 시스템의 필요성 12](#_Toc35983243)

[**2** **개발 목표 및 내용** 13](#_Toc35983244)

[2.1 목표 13](#_Toc35983245)

[2.2 연구/개발 내용 13](#_Toc35983246)

[2.3 개발 결과 16](#_Toc35983247)

[2.3.1 시스템 기능 요구사항 16](#_Toc35983248)

[2.3.2 시스템 비기능 요구사항 17](#_Toc35983249)

[2.3.3 시스템 구조 19](#_Toc35983250)

[2.3.4 결과물 목록 및 상세 사양 19](#_Toc35983251)

[2.4 기대효과 및 활용방안 20](#_Toc35983252)

[**3** **배경 기술** 21](#_Toc35983253)

[3.1 기술적 요구사항 21](#_Toc35983254)

[3.1.1 개발환경 21](#_Toc35983255)

[3.1.2 프로젝트 결과물 확인 21](#_Toc35983256)

[3.1.3 서버환경 22](#_Toc35983257)

[3.2 현실적 제한 요소 및 그 해결 방안 22](#_Toc35983258)

[3.2.1 소프트웨어 22](#_Toc35983259)

[3.2.2 기타 23](#_Toc35983260)

[**4** **프로젝트 팀 구성 및 역할 분담** 24](#_Toc35983261)

[**5** **프로젝트 비용** 25](#_Toc35983262)

[**6** **개발 일정 및 자원 관리** 25](#_Toc35983263)

[6.1 개발 일정 25](#_Toc35983264)

[6.2 일정별 주요 산출물 26](#_Toc35983265)

[6.3 인력자원 투입계획 28](#_Toc35983266)

[6.4 비 인적자원 투입계획 29](#_Toc35983267)

[**7** **참고 문헌** 29](#_Toc35983268)

# **개요**

## 프로젝트 개요

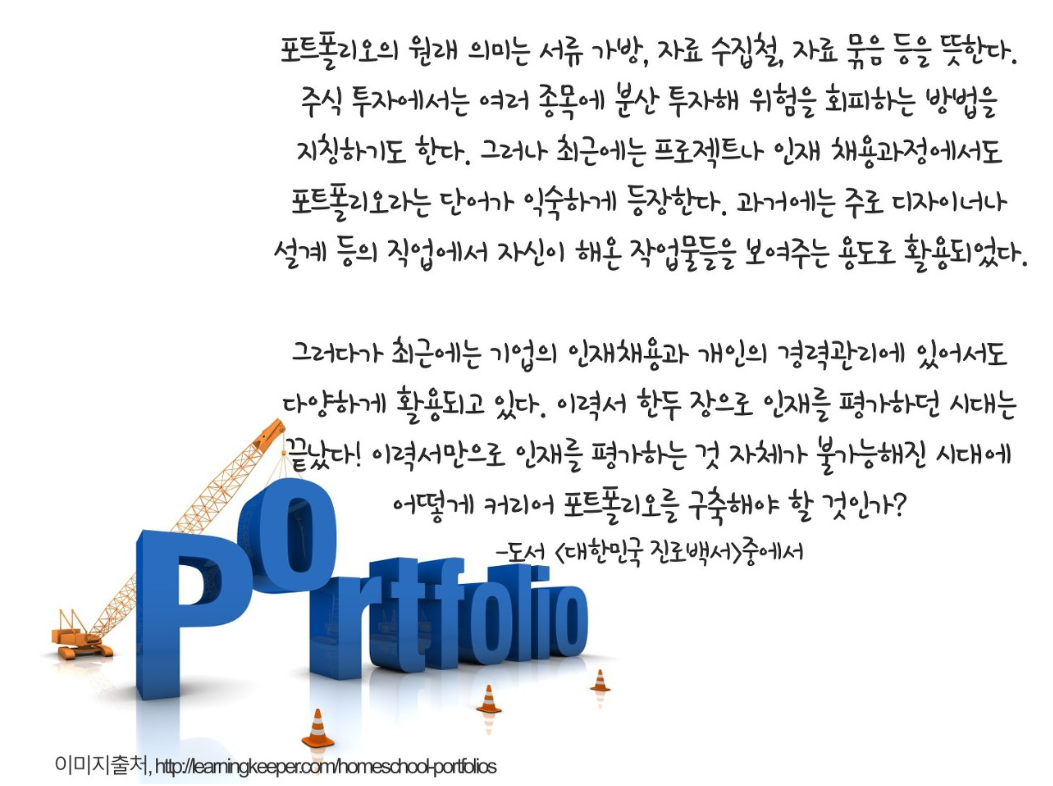
이 프로젝트는 Eye-tracking 기술을 이용하여 개발자(developer) 전용 포트폴리오를 생성 시켜주는 서비스이다. 상기한 기술을 이용하여 포트폴리오 열람자들의 시선 data를 수집하고, 수집한 데이터를 이용하여 작성한 포트폴리오의 구조 재구축(Layout 재배치, 컨텐츠 보완)을 동적 웹페이징 기술을 통하여 구현한다.

또한 포트폴리오 작성자가 포트폴리오를 작성할 때, 여러가지 포트폴리오 예시를 보여주어 작성자의 시선이 오래 머무른 레이아웃 디자인을 조합하여 URL 형태로 제공해준다.

## 추진 배경 및 필요성

현대 사회에서의 대부분의 성인들은 입사하거나 이직을 할 때 자기소개서 혹은 포트폴리오를 작성하여 제출해야한다. <그림 1>을 보면 포트폴리오의 중요성을 판단할 수 있다. 이 때 자신이 작성한 포트폴리오가 잘 작성된 것인지 객관적으로 판단하기가 어렵기 때문에 많은 시간과 막대한 재화를 투자하여 전문가에게 조언을 받기도 한다. 우리는 이러한 개인의 자원낭비를 Eye-Tracking 기술을 통해 해결하고자 한다.

Eye-Tracking(시선 추적) 기술은 시선의 지점(사람이 어디를 응시하고 있는지) 또는 머리에 대한 눈의 상대적인 움직임을 측정하는 과정이다. 영상 시스템, 심리학, 인지 언어학, 제품 디자인 등에 대한 리서치에 사용하고 있다. 즉, 눈은 마음의 창이라는 문장을 기술적으로 실현시킨 것이다. 이 기술을 이용한다면 좋은 포트폴리오에는 시선이 오래 머물 것이고, 이를 통해 자신의 포트폴리오의 객관성을 확보할 수 있는 밑바탕이 된다.



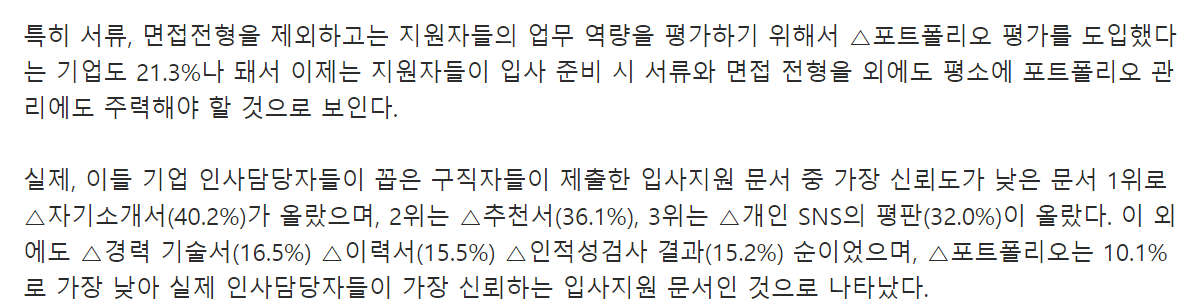
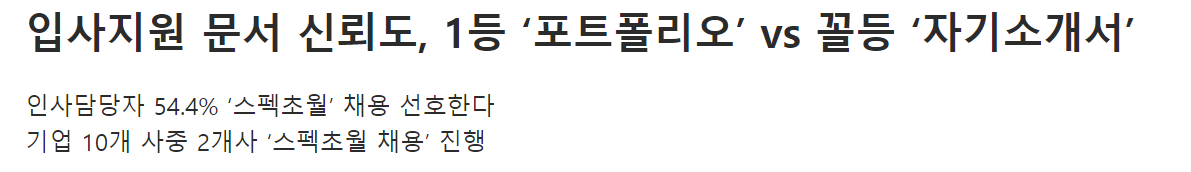
<그림 1>

### **1.2.1 기술의 시장 현황**

**1) 포트폴리오 시장 현황**

포트폴리오 플랫폼이 있긴하지만 외국 사이트가 많아 사용하기에 불편하고 디자이너 포트폴리오에 편중된 경향이 있다. 개발자들에게 특화된 포트폴리오 시장은 없다고 봐도 무방할 정도로 블루오션이라고 말할 수 있다.

<그림 2>를 보면 요즘 기업에서는 이력서와 같이 개인의 신상정보 문서 보다는 개인의 성향 및 참여한 프로젝트들을 나열하여 직군에 알맞은 인재임을 스스로 표현하는 포트폴리오를 더욱 선호하는 추세이다. 이러한 추세는 특히 IT기업에서 두드러진다.

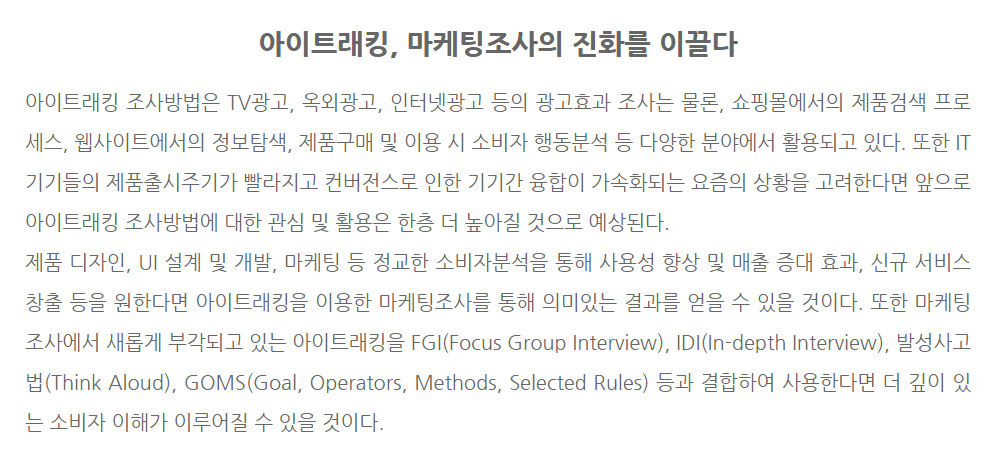


<그림 2>

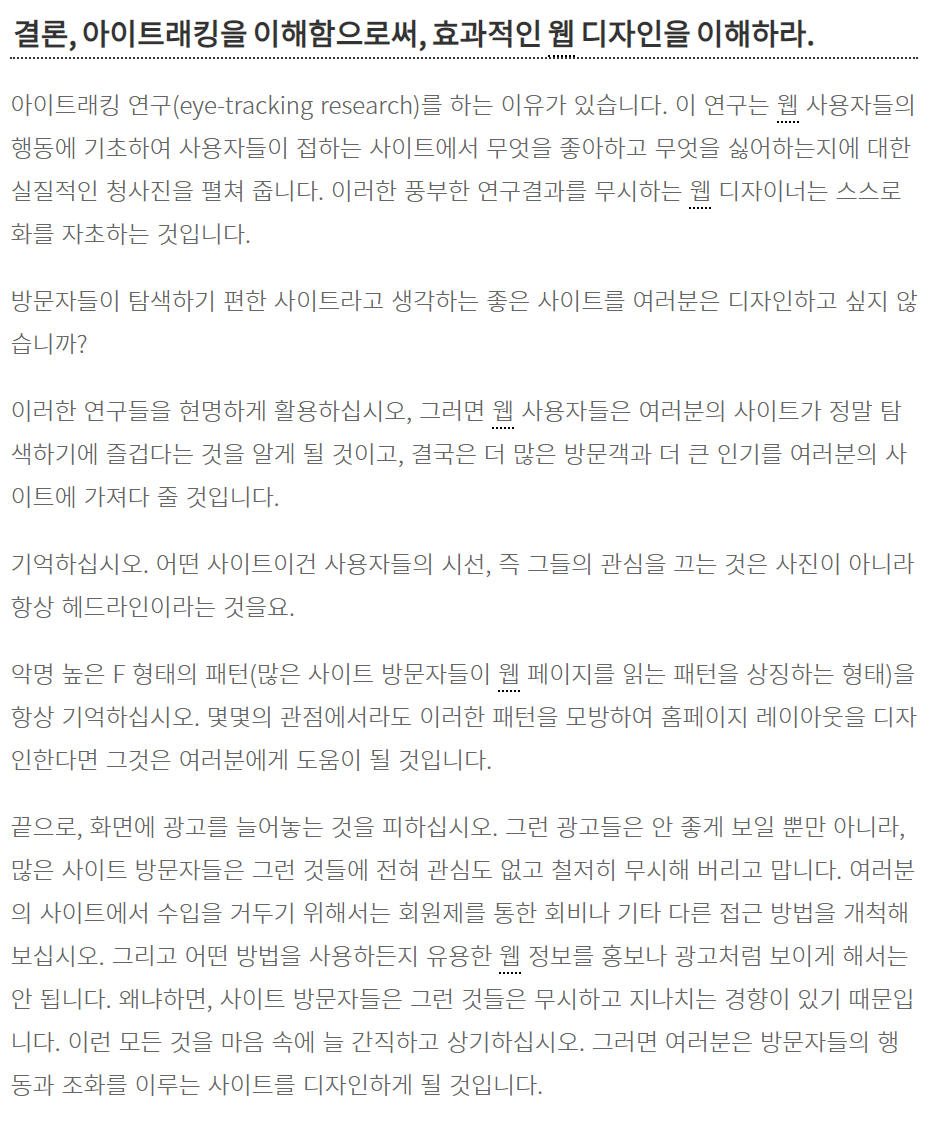
**2) Eye-Tracking기술 시장 현황**

Eye-Tracking 시장은 현재 굉장히 많이 활성화 되어있는 시장이다. 시선처리의 수치화를 높은 수준으로 달성하여 상용화 되고 있는 서비스들이 많이 존재한다. 특히 고객들의 니즈(needs)를 파악해야하는 마케팅 부분에서는 그 활용도가 매우 높다. 이익의 창출은 고객들의 주머니로부터 시작이된다. 주머니를 열게하려면 고객들의 니즈(needs) 파악은 필수이다.

<그림 3>과 <그림 4>는 아이트래킹이 마케팅에 미치는 영향과 이를 효과적으로 사용할 수 있는 주제가 무궁무진함을 보여준다.



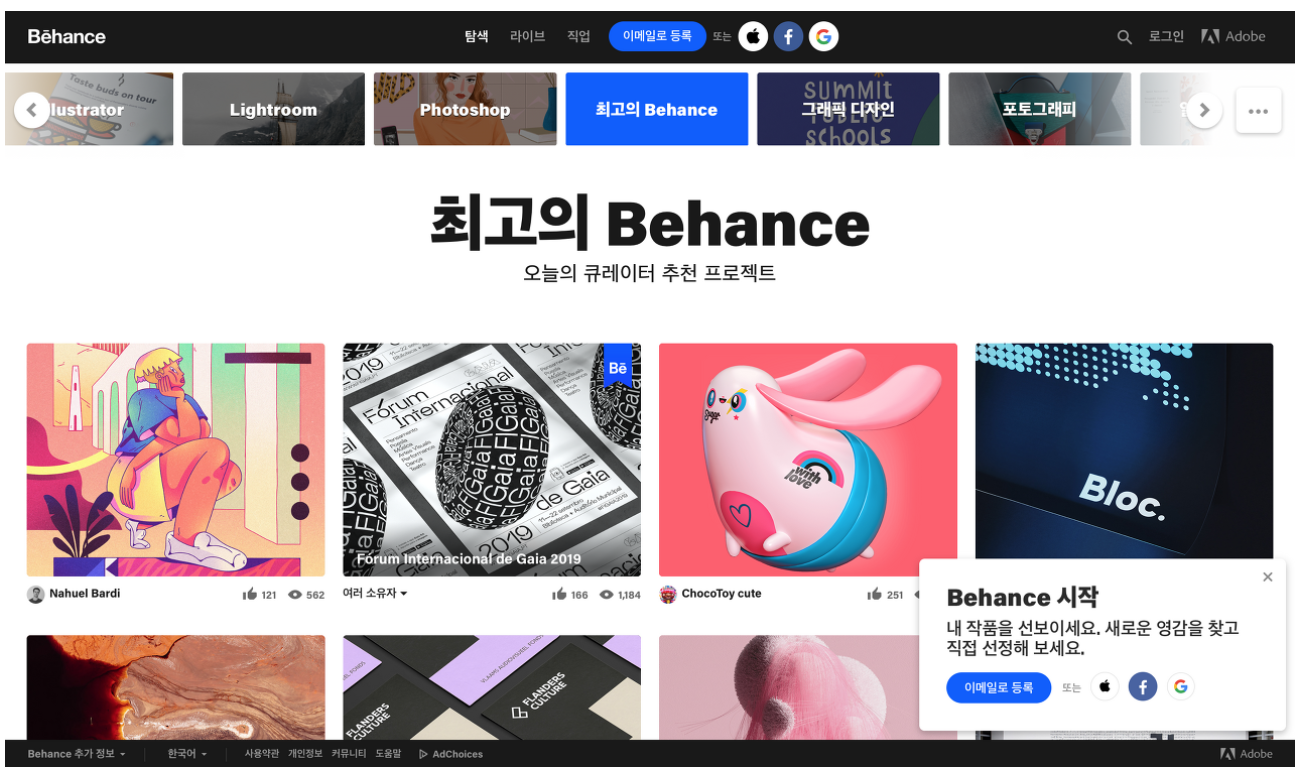
<그림 3>



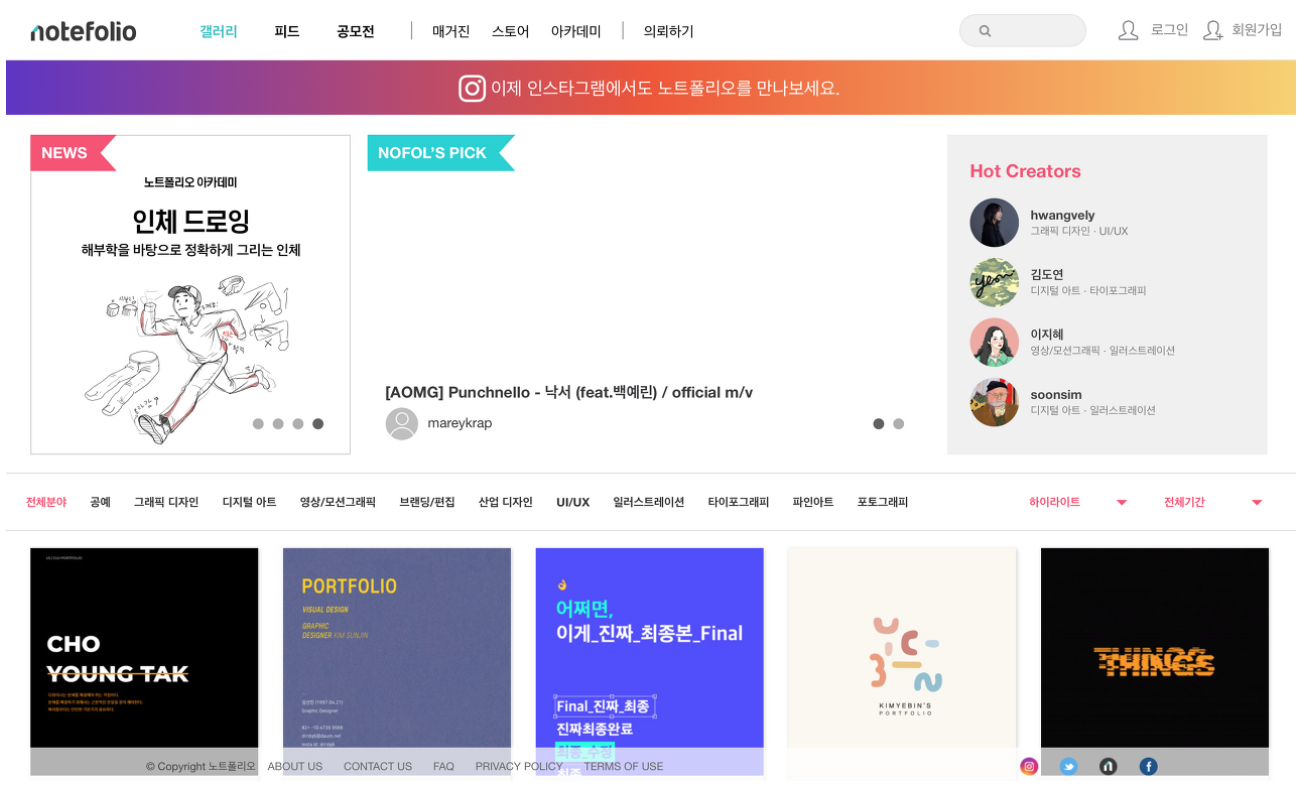
<그림 4>

### **1.2.2기 개발된 시스템 현황**

현재 서비스되고 있는 사이트로는 notefolio, pinterest, Behance, 라우드소싱 등이 있다. 이 사이트들 모두의 공통적인 특징으로는 디자이너들의 포트폴리오 허브라는 것이다. 특히 Behance는 Adobe사가 운영하는 사이트로 세계적인 디자이너의 포트폴리오를 감상할 수 있다. 라우드소싱 같은 경우는 국내에만 서비스 중인 사이트로써 모든 프로젝트나 디자이너가 한국인이라는 특징이 있다. 대기업에서 진행한 프로젝트들도 굉장히 많이 있으며, ‘디자인 콘테스트’ 방식으로 수많은 디자이너들에게 의뢰도 맡길 수 있다.



<그림 5. Behance site>



<그림 6. notefolio site>

### **1.2.3 기 개발된 시스템의 문제점**

1.2.2에서 언급했다시피 기 개발된 시스템에서는 오직 디자이너들을 위한 공간으로 활용되는 경우가 대부분이다. 포트폴리오의 보편적인 의미는 “자신의 이력이나 경력 또는 실력 등을 알아볼 수 있도록 자신이 과거에 만든 작품이나 관련 내용 등을 모아 놓은 자료철 또는 자료 묶음, 작품집으로, 실기와 관련된 경력증명서”로 정의할 수 있다. 요즘 포트폴리오가 사용되는 분야는 디자인 뿐만 아니라 여러 카테고리에서 고루 사용되는 추세인데, 이 의미를 디자이너들에게만 한정적으로 사용했다. 다른 문제점으로는 자신의 포트폴리오를 만들 당시에 객관적으로 잘 만들고 있는지를 확인할 방법이 없다는 것이다. 포트폴리오를 완성시킨 후에 다른사람들의 댓글 혹은 좋아요의 개수로만 판단이 가능하다. 즉, 피드백을 사후에 받을 수 있기 때문에 사전에 좋은 포트폴리오를 만들고 있는지를 파악할 수가 없다. 게다가 하나의 포트폴리오를 만드는데 오랜 시간이 소요되는데, 이는 포트폴리오도 어느 정도의 형식이 있기에 이를 구성할 때, 일일이 같은 작업을 반복적으로 해야할 때가 있으며, 동시에 다른 사람들과는 차별화되는 구성을 생각해내야 하기 때문이다.

### **1.2.4 개발할 시스템의 필요성**

위에 언급된 문제점들을 해결하기 위해서는 디자인 카테고리 뿐만 아니라 다양한 카테고리를 품을 수 있는 플랫폼이 필요하다.

Eye-Tracking기술이 마케팅 부분에서 사람의 관심과 흥미를 포착해낼 수 있다는 점에 착안하여 포트폴리오를 구성하는데도 이 기술의 도입은 상당한 도움이 될 것임을 느꼈다. 사람의 시선 데이터를 추출하여, 포트폴리오를 작성한 후가 아니라, 작성할 당시에 이를 재가공하여 여러 형태의 정보로 포트폴리오를 구성하는데 도움을 주는 기능이 있다면, 그리고 이를 어느정도 자동으로 해주는 기능이라면 매우 편리할 것임은 자명한 사실이다.

# **개발 목표 및 내용**

## 목표

본 프로젝트는 사용자에게 아이트래킹을 통한 최적의 포트폴리오 웹 플랫폼 서비스 개발을 목표로 한다. 플랫폼의 이용 주체에 따라 세부 목표 기능은 아래와 같다.

**1. 포트폴리오 등록자**

1) 등록자의 정보를 넣으면 등록자의 취향을 분석하여 자동 포트폴리오 생성.

2) 등록자에게 포트폴리오 관람자들의 아이트래킹 로그 분석 및 시각화 정보 제공.

3) 아이트래킹 정보를 통해 refactoring된 포트폴리오 추천.

**2. 포트폴리오 열람자**

1) 포트폴리오의 outline을 보는 동안 열람자의 관심사 파악.

2) 열람자가 가장 관심있는 컨텐츠를 우선적으로 열람 가능.

## 연구/개발 내용

**2.2.1 개발 세부 목표**

***Front-End:***

사용자에게 플랫폼 서비스의 인터페이스를 제공한다.

사용자가 직접 경험하는 부분과 경험의 아키텍처를 담당한다.

백그라운드에서 서버 데이터를 가져와 페이지를 동적으로 만든다.

***Back-End:***

사용자의 요청에 다른 데이터를 검색, 저장 또는 변경한다.

REST API를 사용하여 웹 응용 프로그램을 개발한다.

프론트엔드에서 전달하는 아이트래킹 데이터를 유의미한 정보로 변환해 의사결정을 돕는다.

**2.2.2 연구/개발 방법**

***Front-End:***

1. 사용자의 아이트래킹 결과인 좌표와 실제 웹 구성 요소를 Mapping한다. (HTML,CSS,REACT.JS)

2. Mapping 완료 된 웹 구성요소를 이용하여 포트폴리오 레이아웃, 디자인 요소를 구성한다. (HTML,CSS,REACT.JS)

3. 플랫폼의 인터페이스를 구성한다. (HTML,CSS, Angular)

4. 사용자의 요청을 백엔드에 전달한다.

***Back-End:***

1. '**NodeJS**'를 이용하여 API 기술 문서 작성한다.

2. '**Passport.JS**'를 이용하여 SNS 로그인 및 세션 정보를 관리한다.

3. AWS '**Lambda**'를 이용하여 사용자의 아이트래킹 데이터를 가공한다.

4. AWS **EC2()**를 이용하여 웹 서버를 호스팅 한다.

5. AWS **API GateWay()**를 이용하여 API 개발을 간소화하고 SDK를 생성한다.

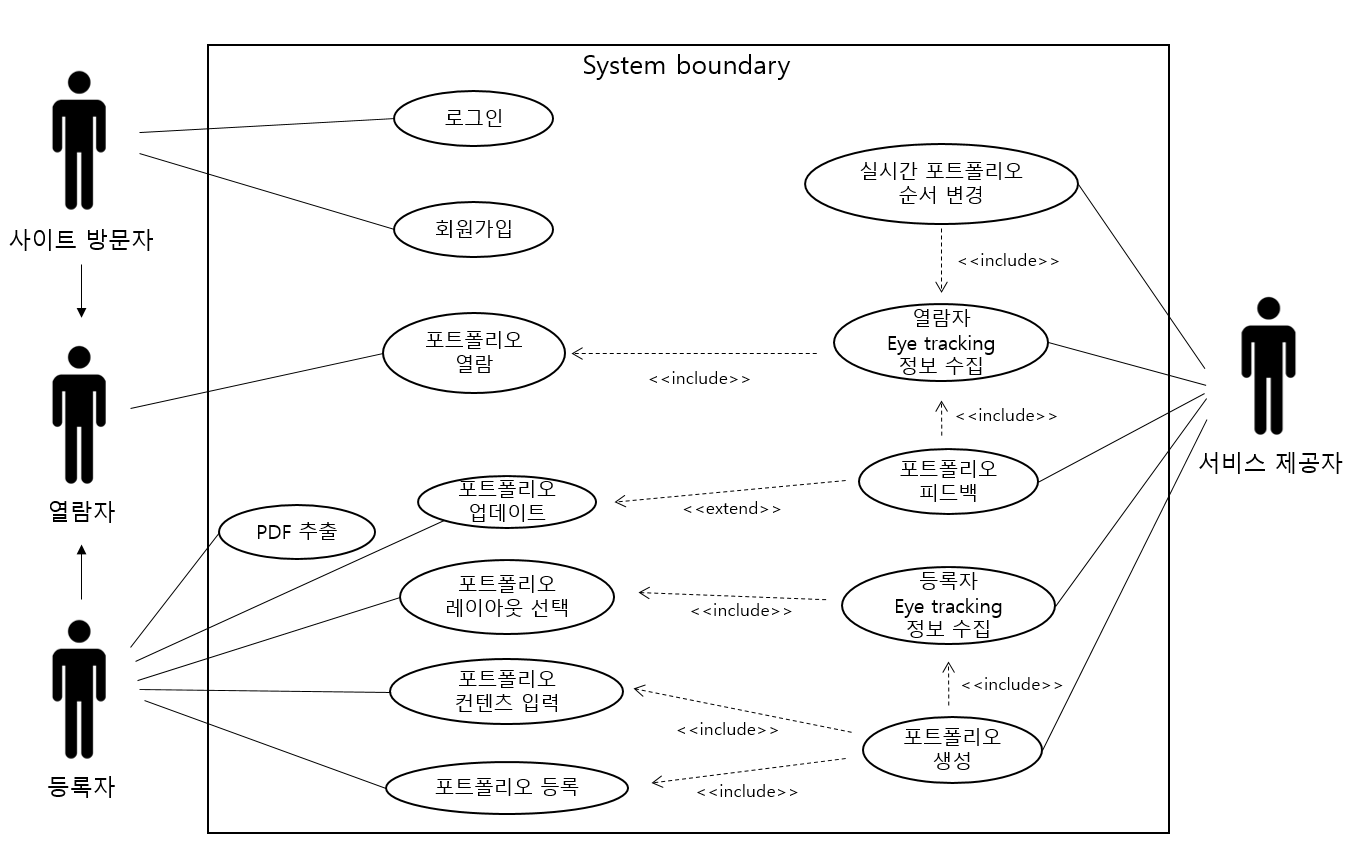
6. AWS **RDS()** ,**DynamoDB()**를 이용하여 데이터를 저장 및 관리한다.

7. AWS **S3** 를 이용하여 등록된 포트폴리오를 저장한다.

8. AWS **Kinesis**를 이용하여 대용량의 실시간 데이터를 안정적으로 수집 및 처리 할 수 있도록 한다.

## 개발 결과

### **2.3.1 시스템 기능 요구사항**

****

|  |  |  |
| --- | --- | --- |
| **Functional Requirement** | **내용** | **중요도** |
| FR1 | 포트폴리오 레이아웃을 선택할 수 있다. | 중 |
| FR2 | 시스템에 로그인 할 수 있다. | 상 |
| FR3 | 시스템에 회원가입 할 수 있다. | 상 |
| FR4 | 포트폴리오를 열람할 수 있다. | 상 |
| FR5 | 포트폴리오를 PDF로 추출 할 수 있다. | 하 |
| FR6 | 포트폴리오를 업데이트 할 수 있다. | 중 |
| FR7 | 포트폴리오의 컨텐츠를 입력 할 수 있다. | 상 |
| FR8 | 포트폴리오를 등록할 수 있다. | 상 |
| FR9 | 실시간으로 포트폴리오의 순서를 변경 할 수 있다. | 상 |
| FR10 | 열람자의 Eye Tracking 정보를 수집할 수 있다. | 상 |
| FR11 | 포트폴리오에 대해 피드백을 줄 수 있다. | 상 |
| FR12 | 등록자의 Eye Tracking 정보를 수집 할 수 있다. | 중 |
| FR13 | 포트폴리오를 생성할 수 있다. | 상 |

### **2.3.2 시스템 비기능 요구사항**

|  |  |  |  |
| --- | --- | --- | --- |
| 번호 | 내용 | 품질 속성 | 중요도 |
| 1 | 검색 데이터 무결성 유지 | 삭제, 수정된 포트폴리오는 검색 했을 때 최신 상태를 유지한다. | 10 |
| 2 | 스토리지 신뢰도 | 스토리지 내구성은 99.9% 이상을 유지한다. | 8 |
| 3 | 개인정보 보호 | 고객의 개인정보는 암호화하여 관리하고, 개인정보 관련 창에서는 https를 지원한다. | 10 |
| 4 | 24시간 사이트 유지 | 사이트는 24시간 온라인이며, 점검으로 이용이 제한될 경우, 점검 화면을 띄운다. | 8 |
| 5 | 스토리지 비용 최소화 | 아이트래킹 정보는 처음 인덱싱을 위해 사용되고 매달 말에 포트폴리오 갱신을 위해 즉시 접근할 수 있어야 한다. | 6 |
| 7 | UI | 사이트 UI는 복잡하지 않고 포트폴리오의 썸내일 위주로 사용자에게 보여진다. | 7 |
| 8 | 적은 서버 지연 시간 | 모든 페이지 이동 및 리다이랙션은 3초 이내로 진행되어야 한다. | 8 |

### **2.3.3 시스템 구조**

### **2.3.4 결과물 목록 및 상세 사양**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **대분류** | **소분류** | **기능** | **형식** | **비고** |
| 웹 페이지 | 사용자 인터페이스 | 포트폴리오를 열람 및 등록 할 수 있다. | WEB |  |
| 입력 | 웹캠 | 사용자의 시선을 추적한다 | HW |  |
| 출력 | PDF | WEB으로 구성되어있는 포트폴리오를 파일로 출력가능하다. | PDF |  |
| 알고리즘 | *rating* | Eye Tracking 데이터를 이용해 사용자의 시선 추적 데이터와 웹 요소 간 우선 순위 파악 | 모듈 |  |

## 기대효과 및 활용방안

1. 포트폴리오에서 자신의 관심사에 맞는 정보를 우선적으로 볼 수 있다.

2. 자신의 정보를 입력하는 것만으로 간단하게 포트폴리오를 만들 수 있다.

3. 아이트래킹을 통한 정보로 자신의 포트폴리오에 사람들이 관심을 많이 가지는 부분을 알 수 있다.

4. 개선된 자신의 포트폴리오를 자동으로 만들 수 있다.

5. 포트폴리오를 간편하게 PDF로 변환하여 종이 문서화가 가능하다.

# **배경 기술**

## 기술적 요구사항

### **3.1.1 개발환경**

1. 작업 기기

- 노트북6대

2. 운영체제

- Winsdows 10, Linux(Ubuntu 18.04)

3. 프로그래밍 언어

- Python, Javascript

4. 사용 오픈소스

- AWS Lambda, AWS EC2, AWS API gateway, AWS RDS, DynamoDB, AWS Kinesis, AWS Amplify, AWS S3

### **3.1.2 프로젝트 결과물 확인**

**1) 데스크탑 버전 웹페이지**

OS : Windows 10

Browser : Chrome

**2) 모바일 버전 웹페이지**

OS : Android, iOS

Browser : Chrome

### **3.1.3 서버환경**

1. 'NodeJS'를 이용하여 Rest API를 구현한다.

2. 'Passport.JS'를 이용하여 SNS 로그인 및 세션 정보를 관리한다.

3. AWS 'Lambda'를 이용하여 사용자의 아이트래킹 데이터를 유의미한 정보로 가공한다.

4. AWS EC2를 이용하여 웹 서버를 호스팅 한다.

5. AWS API GateWay를 이용하여 API 개발을 간소화하고 SDK를 생성한다.

6. AWS RDS, DynamoDB를 이용하여 데이터를 저장 및 관리한다.

8. AWS Kinesis를 이용하여 대용량의 실시간 데이터를 안정적으로 수집 및 처리 할 수 있도록 한다.

9. AWS Amplify를 AWS 기반의 확장 가능한 모바일 및 웹 앱을 손쉽게 생성, 구성 및 구현 가능하게 한다. (HTTPS 및 인증)

## 현실적 제한 요소 및 그 해결 방안

### **3.2.1 소프트웨어**

**1) 실시간 포트폴리오 처리 문제**

열람자의 관심도 순으로 나열하여 실시간으로 포트폴리오를 보여줄 때 열람자의 속도 맞춰 페이지 로딩이 오래 걸리지 않게 처리해야 한다. 이것은 열람자의 관심도 순으로 포트폴리오를 재배치하는 것이 아닌 순위에 해당하는 페이지를 보여주는 것을 해결한다.

**2) 아이트래킹 정확성 문제**

아이트래킹을 이용해 동공의 좌표를 받아 올 때 정확성이 떨어질 수 있다. 정확성을 높이기 위해 페이지에 몇 개의 구역을 나눠 보다 정확한 아이트래킹 좌표를 얻을 수 있도록 해결한다.

### **3.2.2 기타**

**1) 서버 유지 문제**

수용 가능한 사용자의 수를 넘어 서버에 문제가 생길 수 있다. 이때 AWS S3서비스로 서버를 유지 하도록 해결한다.

# **프로젝트 팀 구성 및 역할 분담**

|  |  |
| --- | --- |
| **이름** | **역할** |
| 최락준 | * Backend 서버 구축 * 아이트래킹 DB연결 |
| 권강민 | * Backend DB 연결 및 구축 * Documentation 정리 |
| 심재욱 | * Backend 서버 구축 * 웹페이지 재구조화 |
| 우승민 | * Frontend 아이트래킹 웹페이지 연동 * 아이트래킹 알고리즘 최적화 |
| 윤준호 | * Frontend 웹페이지 구성 * Sub Backender |
| 황채은 | * Frontend 웹페이지 구성 * Unit testing |

# **프로젝트 비용**

|  |  |
| --- | --- |
| **항목** | **예상치 (MD)** |
| 아이디어 구상 | 40MD |
| 아이트래킹 연구 및 기술 테스트 | 50MD |
| 클라우드 서비스 구축 | 60MD |
| 포트폴리오 데이터셋 구축 | 20MD |
| 웹페이지 및 UI 제작 | 20MD |
| 데모 생성 및 테스트 | 10MD |
| 합 | 200MD |

# **개발 일정 및 자원 관리**

## 개발 일정

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **항목** | **세부내용** | **1월** | **2월** | **3월** | **4월** | **5월** | **6월** | **비고** |
| 요구사항분석 | 요구 분석 |  |  |  |  |  |  |  |
| SRS 작성 |  |  |  |  |  |  |  |
| 관련분야연구 | 주요 기술 연구 |  |  |  |  |  |  |  |
| 관련 시스템 분석 |  |  |  |  |  |  |  |
| 설계 | 시스템 설계 |  |  |  |  |  |  |  |
| 구현 | 코딩 및 모듈 테스트 |  |  |  |  |  |  |  |
| 테스트 | 시스템 테스트 |  |  |  |  |  |  |  |

## 일정별 주요 산출물

|  |  |  |  |
| --- | --- | --- | --- |
| **마일스톤** | **개요** | **시작일** | **종료일** |
| 계획서 발표 | * 개발 환경 테스트 (아이트래킹 모듈 테스트) * 아이디어 구체화 및 실현가능성 토의   **산출물 :**   1. 프로젝트 수행 계획서.docx 2. 프로젝트 ppt slid 3. 프로젝트 발표 동영상 | 2020-02-26 | 2020-03-27 |
| 설계 완료 | 시스템 설계 완료  **산출물 :**   1. 시스템 설계 사양서 | 2020-03-07 | 2020-03-24 |
| 중간 보고 | * 아이트래킹 모듈 구현 완료 * 데이터베이스 구축 * 웹페이지 구축   **산출물 :**   1. 프로젝트 중간 보고서 2. 프로젝트 진도 점검표 3. 프로젝트 중간 소스 코드 | 2020-03-27 | 2020-04-24 |
| 구현 완료 | 시스템 구현 완료  산출물:  1. AWS 연동  2. 동적 웹페이지 구조화  3. 아이트래킹 모듈 접합 | 2020-04-24 | 2020-06-02 |
| 테스트 | 시스템 통합 테스트  **산출물:**  1. 안정화 version 배포  2. 유스케이스에 맞는 시나리오 적용 및 테스트 | 2020-04-24 | 2020-06-02 |
| 최종 보고서 | 최종 보고  **산출물:**  1. 최종 보고서 작성  2. 시연 영상 업로드 | 2020-06-02 | 2020-06-12 |

## 인력자원 투입계획

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **이름** | **개발항목** | **시작일** | **종료일** | **총개발일(MD)** |
| 최락준 | *Backend)*   * *서버 구축 및 아이트래킹 DB 연결* * *rest api 개발, aws 연결* | 2020-04-01 | 2020-06-02 | 63 |
| 권강민 | *Backend)*   * *DB 연결 및 구축* * *문서 정리* | 2020-04-01 | 2020-06-02 | 63 |
| 심재욱 | *Backend)*   * *aws 서버 구축* * *웹페이지 재구조화* | 2020-04-01 | 2020-06-02 | 63 |
| 윤준호 | *Frontend)*   * *웹페이지 구성* * *유닛테스트* | 2020-04-01 | 2020-06-02 | 63 |
| 황채은 | *Frontend)*   * *웹페이지 구성* * *아이트래킹 성능개선* | 2020-04-01 | 2020-06-02 | 63 |
| 우승민 | *Frontend)*   * *아이트래킹 웹페이지 연동* * *디자인 개발* | 2020-04-01 | 2020-06-02 | 63 |

## 비 인적자원 투입계획

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **항목** | **Provider** | **시작일** | **종료일** | **수량** |
| 개발용 PC 6대 | LG Gram | 2020-02-26 | 2020-06-12 | 2 |
| lenovo | 2020-02-26 | 2020-06-12 | 1 |
| 한성컴퓨터 | 2020-02-26 | 2020-06-12 | 1 |
| ASUS | 2020-02-26 | 2020-06-12 | 2 |
| Android 이동통신기 | Samsung | 2020-02-26 | 2020-06-12 | 2 |
| ios 이동통신기 | Apple | 2020-02-26 | 2020-06-12 | 4 |

# **참고 문헌**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **번호** | **종류** | **제목** | **출처** | **발행년도** | **저자** | **기타** |
| 1 | 웹페이지 | 포트폴리오의 시대, 이력서 한두 장의 시대는 끝났다! | brunch | 2019 | 정철상 |  |
| 2 | 기사 | 입사지원 문서 신뢰도, 1등 ‘포트폴리오’ vs 꼴등 ‘자기소개서’ | 잡코리아 | 2014 | 변지성 |  |
| 3 | 웹페이지 | 심리생리학을 품은 마케팅조사  아이트래킹(Eye-Tracking) | 블로그 | 2012 | 오이모 |  |
| 4 | 기사 | 아이트래킹 연구가 알려준 사용자 경험 디자인 4가지 필수 규칙 | 디자인로그 | 2013 | MARC SCHENKER |  |
| 5 | 웹페이지 | 디자인사이트 | 디자인킷 | 2020 | - |  |
| 6 | 웹페이지 | WebGazer.js | WebGazer.js | - | - | 배경기술 |
| 7 | 깃허브 | 배경 기술에 웹캠 아이트레킹 오픈소스 | WebGazer | - | - | 배경기술 |
| 8 | 웹페이지 | AWS 클라우드 서비스 | Amazon | - | - | 배경기술 |