**프로젝트 기술서**

프로젝트 명: STIRRING

목차

[**1.** **프로젝트 개요** 2](#_Toc536407918)

[1.1 프로젝트 목적 2](#_Toc536407919)

[1.2 서비스 비전 2](#_Toc536407920)

[1.3 기술 스택 2](#_Toc536407921)

[**2.** **설계** 3](#_Toc536407922)

[2.1 데이터 오브젝트 3](#_Toc536407923)

[2.2 Simple Usecase 4](#_Toc536407924)

[2.3 페이지 6](#_Toc536407925)

[2.4 기타 6](#_Toc536407926)

[**3.** **구현** 7](#_Toc536407927)

[3.1 프론트엔드 7](#_Toc536407928)

[3.2 백엔드 7](#_Toc536407929)

[3.3 데이터베이스 7](#_Toc536407930)

[**4.** **결과** 8](#_Toc536407931)

[4.1 프레젠테이션 8](#_Toc536407932)

[4.2 Github 8](#_Toc536407933)

[4.3 접속 링크 8](#_Toc536407934)

# **프로젝트 개요**

## 프로젝트 목적

1. 본 프로젝트는 웹 서비스에 대한 이해와 개발역량 향상을 목적으로 하는 개인 프로젝트이다.
2. Vue.js / Node.js 등 대표적인 웹 프레임워크를 이용하여 웹 서비스를 구현한다.

## 서비스 비전

1. STIRRING은 사진 데이터를 자유롭게 공유하는 웹 서비스이다.
2. 개발 참조 모델: [Unsplash.com](https://unsplash.com), [Pixabay.com](https://pixabay.com)

## 기술 스택

1. Vue.js
2. Node.js + Express
3. MongoDB + Mongoose
4. JsonWebToken
5. REST API
6. Git

# **설계**

## 데이터 오브젝트

1. User: 사용자 정보

|  |  |  |
| --- | --- | --- |
| 속성 | 데이터 타입 | 설명 |
| \_id | string | User 구별을 위한 임의의 ID값, 자동생성, Unique |
| email | string(email) | 유저 이메일, 회원가입 시 입력, 로그인, Unique |
| password | string(password) | 유저 패스워드, 회원가입 시 입력, 로그인 |
| name | string | 유저 네임, 회원가입 시 입력 |
| image | number | 유저 프로필용 이미지 |
| posts | [string] | 유저가 게시한 post 리스트 |
| followings | [string] | 유저가 follow하는 유저 수, 리스트 |
| followers | [string] | 유저를 follow하는 유저 수, 리스트 |
| collections | [string] | 유저가 만든 collection 리스트 |
| interests | [string] | 유저가 관심가지는 tag 리스트, 최대 5개 |
| likes | [string] | 유저가 like 표시한 post 리스트 |
| createdAt | date | 유저가 회원가입한 일시 |
| updatedAt | date | 유저가 정보를 수정한 일시 |
|  |  |  |

1. Post: 게시물 정보

|  |  |  |
| --- | --- | --- |
| 속성 | 데이터 타입 | 설명 |
| \_id | string | Post 구별을 위한 임의의 ID값, 자동생성, Unique |
| author | string(id), string(name) | 이 post를 작성한 User(id, name) 데이터 |
| data | data | 업로드된 사진(동영상)의 정보를 담은 객체 |
| tags | [string] | post가 담은 정보를 나타내는 tag, 최대 5개 |
| likes | number | post에 like 표시한 user의 수 |
| downloads | number | post의 data를 내려 받기 한 수 |
| views | number | post를 자세히 보기한 user의 수 |
| createdAt | date | post가 게시된 일시 |
| updatedAt | date | post가 수정된 일시 |
|  |  |  |

1. Data: 게시된 사진, 동영상의 정보

|  |  |  |
| --- | --- | --- |
| 속성 | 데이터 타입 | 설명 |
| \_id | string | data 구별을 위한 임의의 ID값, 자동생성, Unique |
| filename | string | 사진(동영상)이 업로드 될 때 주어지는 이름, 자동생성 |
| size | number | 사진(동영상)의 용량 |
| resolution | json | 사진의 해상도, width \* height |
| format | string | 사진(동영상)의 데이터 포맷, 확장자 |
| location | location | 사진(동영상)에 저장된 위치정보, 옵션 |
|  |  |  |

1. Tag: 게시물에 표시될 #태그, Type: Set(String)
2. Collection

|  |  |  |
| --- | --- | --- |
| 속성 | 데이터 타입 | 설명 |
| \_id | string | collection 구별을 위한 임의의 ID값, 자동생성, Unique |
| title | string | collection 제목 |
| posts | [string] | collection에 등록된 post 리스트 |
| author | user | collection을 생성한 user 정보 |
| createdAt | date | collection을 생성한 일시 |
| updatedAt | date | collection을 수정한 일시 |
|  |  |  |

## Simple Usecase

1. User

|  |  |  |
| --- | --- | --- |
| UC\_ID | UC\_Name | UC\_Description |
| U01 | 회원가입 | 회원가입을 할 수 있다. (local, google, facebook)  email: email 형식, 6~254, 중복체크, 가입 후 email 인증  password: 6~20자, 공백 불가, 비밀번호 확인  name: 2~20자, 특수문자 불가, 끝 공백 불가 |
| U02 | 로그인 | 로그인 할 수 있다. (local, google, facebook) |
| U03 | 로그아웃 | 로그아웃 |
| U04 | 비밀번호 찾기 | 이메일 인증 후 비밀번호 변경 2단계로 연결 |
| U05 | 비밀번호 변경 | 1단계: 이전 비밀번호 확인  2단계: 새 비밀번호 입력(비밀번호 확인 입력) |
| U06 | 회원 탈퇴 | 비밀번호로 인증 후 탈퇴  게시물 삭제 또는 남기기 |
| U07 | 회원 정보 조회 | 회원정보 조회 |
| U08 | 프로필 이미지 등록 | 프로필 이미지 등록  미 등록 시 기본 이미지 |
| U09 | 프로필 이미지 삭제 |  |
| U10 | 프로필 이미지 수정 | 등록을 덮어쓰기는 방식으로 구현 |
| U11 | 팔로잉 | A follows B: A Folloings에 B 추가, B Followers에 A추가 |
| U12 | 언팔로잉 |  |
| U13 | 팔로잉 리스트 조회 |  |
| U14 | 팔로워 리스트 조회 |  |
| U15 | 관심태그 조회 |  |
| U16 | 관심태그 추가 | 최대 5개 |
| U17 | 관심태그 삭제 |  |
| U18 | 유저 검색 | 이름으로 검색  이메일로 검색 |
|  |  |  |

1. Post

|  |  |  |
| --- | --- | --- |
| UC\_ID | UC\_Name | UC\_Description |
| P01 | 게시하기 | data 등록 후 게시물 생성  Tags: 최대 8개 |
| P02 | 수정하기 | Tags 수정, 최대 8개 |
| P03 | 삭제하기 |  |
| P04 | 자세히보기 | views 증가 |
| P05 | 리스트보기 | 자세히보기로 연결 |
| P06 | 검색하기 | 태그로 검색  유저로 검색 |
| P07 | Author 연결 | 유저 정보 조회로 연결 |
| P08 | like | 토글, User like Post: user likes에 post 추가, post likes에 user 추가 |
| P09 | unlike | 토글, User like Post: user likes에 post 삭제, post likes에 user 삭제 |
| P10 | 파일 다운로드 | Public 링크로 연결, downloads 증가 |
| P11 | collection 추가 | 컬렉션 리스트 모달로 연결 |
| P12 | collection 삭제 | 컬렉션 리스트 모달로 연결 |
| P13 | views 증가 | 자세히 보기 시 증가 |
| P14 | 공유하기 | 링크 복사 |
| P15 |  |  |

1. Collection

|  |  |  |
| --- | --- | --- |
| UC\_ID | UC\_Name | UC\_Description |
| C01 | 리스트 보기 | 컬렉션 생성, 삭제 / 포스트 추가, 제거 로 연결 |
| C02 | 컬렉션 생성 | 타이틀: String, 최대 30자 |
| C03 | 컬렉션 삭제 |  |
| C04 | 포스트 추가 | 컬렉션에 추가, 컬렉션에서 삭제로 토글 |
| C05 | 포스트 삭제 | 컬렉션에서 삭제, 컬렉션에 추가로 토글 |
| C06 |  |  |

1. 기타 미분류

|  |  |  |
| --- | --- | --- |
| UC\_ID | UC\_Name | UC\_Description |
| E01 | 언어 변경 | 다국어 지원: English, 한국어 |
| E02 | 반응형 페이지 | 뷰 크기에 맞는 레이아웃 지원(Mobile, PC) |
| E03 |  |  |

## 페이지 구성

1. 메인
   1. 검색 링크
   2. 로그인 링크 (비 로그인 시)
   3. 회원가입 링크 (비 로그인 시)
   4. 로그아웃 링크 (로그인 시)
   5. 유저 프로필 자세히 보기 링크(로그인 시)
   6. 언어 선택 링크
   7. Post 게시 링크
   8. 검색 영역
   9. Post 리스트 컨포넌트(최신순, 인기순)
2. 회원가입
   1. Email, Password, Name 입력, 검증 -> 이메일 인증 -> 로그인 페이지
3. 로그인
   1. 이메일 입력, 패스워드 입력
   2. 로그인 버튼 -> 유저 데이터 검증 ->메인 페이지
   3. 회원가입 버튼(링크)
   4. 비밀번호 찾기 버튼(링크) -> 이메일 입력 -> 이메일 인증 -> 비밀번호 변경 2단계
   5. Google, Facebook 로그인 링크
4. 유저 프로필 자세히 보기
   1. 유저 개인정보 자세히 보기/수정 링크
   2. 프로필 이미지
   3. 관심 태그
   4. Posts
   5. Likes
   6. Collections
5. 유저 개인정보 자세히 보기/수정
   1. 프로필 이미지 변경
   2. 이메일
   3. 네임
   4. 비밀번호 변경 링크
6. 검색(Tags, User.name)
   1. 검색어 입력 -> 검색
   2. 검색 결과 리스트 탭(Posts, Collections, Users/최신순, 인기순)  
      -> 간단히 보기 리스트
7. Post 게시
   1. 사진 올리기 영역
   2. 태그 입력 영역
8. Post 자세히 보기
   1. Author User 정보 -> 회원 정보 페이지
   2. Like 추가/삭제
   3. Collection 추가/삭제 버튼 -> 모달
   4. 다운로드 버튼 -> 링크 복사, 파일 다운로드 선택
   5. 사진 영역
9. Post 간단히 보기
   1. 사진 영역
   2. Author User 정보
   3. Like 추가/삭제
   4. Collection 추가/삭제 버튼 -> 모달
10. Collections 리스트 모달(추가/삭제 용)
    1. Collection 생성 버튼
    2. Collection 리스트 -> 추가/삭제
11. Collections 자세히 보기
    1. Collections(title, posts) 리스트 보기
    2. Collection 자세히 보기 -> Title, Posts

## 기타

1. 다국어 처리

* 언어 설정 변수
* CSS 다중 폰트 설정

1. 반응형 웹 페이지

# **구현**

## 프론트엔드

1. /main (/ 및 정의되지 않은 라우터 포함)
2. /user/login
   1. Email 폼 체크: email 형식, 6~254자
   2. Password 폼 체크: 6~20자, 공백 체크
   3. Login 버튼: (Post)/api/users/login  
      -> 성공: 메인 페이지로 연결   
      //TODO\_HERE: 반환된 토큰으로 인증을 유지하는 부분 이해하고 구현하기  
      -> 실패: 에러 메시지 출력
   4. OAuth 로그인 버튼: OAuth로 연결
3. /user/signup
   1. Email 폼 체크: email 형식, 6~254자
   2. Password 폼 체크: 6~20자, 공백 체크, PasswordConfirm 체크
   3. Name 폼 체크: 1~20자, 끝 공백 불가, 특수문자 불가
   4. Signup 버튼: (Post)/api/users  
      -> 성공: /signup/confirm 페이지로 연결 -> 이메일 인증 -> 로그인 페이지로 연결  
      -> 실패: 에러 메시지 출력
4. /user/password/forget
   1. 이메일 폼 체크
   2. 이메일 인증
   3. /user/password/new 페이지로 연결
5. /user/password/old
   1. 현재 비밀번호 폼 체크
   2. 현재 비밀번호 검증 체크
   3. /user/password/new 페이지로 연결
6. /user/password/new
   1. 새 비밀번호 입력
   2. 새 비밀번호 확인 입력
   3. 콜 스택으로 연결
7. /user/drop
   1. 현재 비밀번호 체크
   2. 메인 페이지로 연결
8. /user/mypage
9. /user/details

## 백엔드

## 데이터베이스

# **결과**

## 프레젠테이션

## Github

## 접속 링크