

팀 구성 및 명단

교과목명	팀 과제명	성 명	학년	학 번	연락처 e_mail
			2	23101201	panpan2872002@gmial.com

Project 계획서

팀장	서명
팀원	서명
팀원	서명

팀 Project 명	
-------------	--

Project 개요 및 예상결과물	<div><div>◦ 개요 :</div><div>" " 가 . . .</div></div> <div><div>◦ 예상결과물 :</div><div>. CRUD 가, . .</div></div>
-----------------------	--

Project 진행방법 및 절차(내용 및 차별성, 업무분담, 일정, 진행방안, 전략)

웹 프로그래밍 프로젝트 계획서

1. 프로젝트 목표

"사진 추억 갤러리 웹 프로젝트"는 사용자가 자신의 소중한 추억을 사진과 간단한 설명과 함께 기록하고, 추억을 웹에서 간편하게 관리할 수 있도록 돕는 웹 애플리케이션입니다. 이 프로젝트의 목표는 사용자에게 직관적인 사진 갤러리 기능을 제공하여, 사진을 통해 개인의 이야기를 시각적으로 저장하고 공유할 수 있는 공간을 구현하는 것입니다.

2. 구현 내용

- **사진 업로드 및 설명 추가:** 사용자가 로컬 저장소에서 사진을 업로드하고, 각 사진에 대해 제목과 간단한 설명을 입력할 수 있는 기능.
- **슬라이드쇼 기능:** 사용자가 업로드한 사진을 자동으로 순차적으로 보여주는 슬라이드쇼 기능.
- **갤러리 보기:** 업로드 된 사진들을 그리드 형태로 배열하여 갤러리 형식으로 볼 수 있는 기능.
- **사진 편집 및 삭제 기능:** 사진과 설명을 수정하거나 삭제할 수 있는 기능.
- **공유 링크 생성:** 각 사진 또는 전체 갤러리에 대한 고유의 공유 링크를 생성하여, 사용자들이 다른 사람과 추억을 공유할 수 있는 기능.
- **프로필 비공개 추가 설정:** 공유된 링크를 통해 사진을 본 사람들은 사진과 설명만 볼 수 있고, **프로필 정보는 비공개로** 유지됩니다. 이 경우, 개인 정보를 보호할 수 있어 안전한 선택이 됩니다.

3. 구현 방안

1. 프론트 엔드 (HTML, CSS, JavaScript):

- **HTML/CSS:** 전체적인 레이아웃과 스타일링을 HTML과 CSS로 구성하며, 직관적인 사용자 인터페이스를 설계합니다.
- **JavaScript:** 사진 업로드, 슬라이드쇼 기능, CRUD (생성, 읽기, 업데이트, 삭제) 기능을 JavaScript로 구현하여, 사용자 경험을 동적으로 만듭니다.

2. 데이터 처리 (JavaScript 및 로컬 스토리지):

- 사진 데이터와 설명을 로컬 스토리지에 저장하여 새로고침 시에도 데이터가 유지되도록 구현합니다.
- Ajax를 이용해 데이터의 비동기 처리를 구현하여 빠르고 끊김 없는 사용자 경험을 제공합니다.

3. 웹 API 활용:

- 공유 링크 생성 및 SNS 공유 기능을 위해 Web API를 활용하여 외부 플랫폼과의 연결을 도모합니다.

4. 프로젝트 일정

- **11월 4일 ~ 11월 6일:** 프로젝트 계획서 작성 및 UI/UX 설계
- **11월 7일 ~ 11월 26일:** 기본 기능 구현
 - 사진 업로드 및 설명 추가 기능 구현
 - CRUD 기능 및 로컬 스토리지 저장 기능 구현
- **11월 27일 ~ 12월 9일:** 추가 기능 구현
 - 슬라이드쇼 기능 구현
 - 공유 링크 생성 기능 및 공유 인터페이스 구현
- **12월 10일 ~ 12월 16일:** 최종 점검 및 완료 보고서 작성
 - 프로젝트 전체 기능 점검 및 오류 수정
 - UI/UX 최종 조정
 - 완료 보고서 작성 및 프로젝트 마무리

5. 팀원 구성 및 임무 분담

- **팀원 구성:** 팀은 1인(이맛판_23101201)로 구성됩니다.
- **팀원 임무 분담:**
 - **프로젝트 기획 및 설계:** 프로젝트 목표와 구현 내용을 정의하고, 일정에 맞춰 설계 및 진행.
 - **프론트 엔드 개발:** HTML, CSS, JavaScript를 사용하여 사용자 인터페이스와 기능을 개발.
 - **기능 구현:** 사진 업로드, 슬라이드쇼, CRUD, 공유 링크 생성 기능을 구현.
 - **UI/UX 최적화:** 사용자 경험을 고려한 디자인 및 기능 최적화.

- **테스트 및 디버깅:** 각 기능이 올바르게 작동하는지 테스트하고, 발견된 오류를 수정.
- **보고서 작성:** 프로젝트 완료 후 완료 보고서 작성 및 프로젝트 결과 정리.

6. 설계 시 제약사항 및 고려사항

1. 제약사항

- **저장 용량 제한:** 로컬 스토리지 용량 제한으로 인해, 많은 사진을 저장할 경우 저장 공간이 부족할 수 있습니다.
- **구현방안:** 사진 파일 크기를 제한하고, 업로드 시 자동으로 압축하여 용량을 줄이는 기능을 추가합니다. 더 많은 사진을 저장하고 싶을 경우 클라우드 스토리지와 연동하여 외부 서버에 사진을 저장할 수 있는 옵션도 고려합니다.

2. 고려사항

- **개인정보 보호:** 사용자의 프로필 정보와 업로드 된 사진의 메타 정보가 외부에 노출되지 않도록 주의해야 합니다.
- **구현방안:** 기본적으로 프로필 정보는 비공개로 설정하고, 사용자가 공유할 때 공개 여부를 선택할 수 있는 옵션을 제공합니다. 또한, 사진 업로드 시 위치 데이터와 같은 메타 정보를 자동으로 삭제하는 기능을 추가합니다.