

Gen Z 세대를 위한 공유 캘린더

|| [간신히 GEN Z]
프로젝트 제안서
2024-09-22

2271246 김소진, 2071111 이준용, 2091019 채희석, 2071012 이연준

1. 프로젝트 배경 및 설명

밀레니얼 Z 세대를 일컫는 MZ 세대를 넘어서 Generate Z 세대를 부르는 Gen Z 세대라는 말이 최근들어 유행하고 있다. 이는 2000 년생부터 시작하는 Z 세대를 의미하며 우리가 미디어에서 보는 MZ 세대 특징과 매우 유사하다. 스마트폰 보급 발달로 인터넷 사용에 익숙하며, 대부분의 상황에서 많은 귀찮음을 느낀다. 이러한 젊은 Gen Z 세대는 학교와 사회에 적응함에 있어서 그룹을 형성하고 이런 그룹들과의 상호작용을 필요로 한다. 이 과정에서 서로의 일정을 맞추고, 약속을 잡으며, 바쁜 와중에도 놀거리 및 취미생활에 시간을 할애한다. 이를 기반으로 본 프로젝트를 기획해 낼 수 있었다. 각자의 일정을 정리할 수 있을 뿐만 아니라 그룹을 형성하여 서로의 일정을 확인하고, 더 나아가서는 그룹내의 일정 및 약속을 자동으로 잡아주는 공유 캘린더를 기획하였다. 귀찮음에 의한 편리함을 추구하는 Gen Z 세대를 겨냥한 어플리케이션이며 프로젝트 참여자들도 이 세대에 해당하여 보다 더 사용자의 입장에서 다가갈 수 있다고 생각한다.

2. 프로젝트 목표 (목적)

사용자들로 하여금 캘린더 즉 달력을 활용하여 일정관리를 한눈에 할 수 있도록하고, 그룹을 형성하여 이 그룹내에 서로 일정을 공유하는 공유 캘린더의 역할을 하는 어플리케이션을 개발하는것이 프로젝트의 목표이다. 여기에 더해 공유그룹에 포함된 인원들의 일정을 자동으로 종합해 최적의 날짜, 최적의 시간대에 모임시간 및 약속 일정을 잡아주는 기능을 추가한 다목적용 공유 캘린더를 구축할 것이다.

3. 고수준 요구 사항

- 서버의 사용자 정보 등록 및 송수신
- 일정 공유에 대한 서버 통신
- 그룹 추가 및 삭제에 대한 서버 송수신
- 일정 공개여부 및 타임라인 설정 동기화
- 직관적으로 일정을 한눈에 볼 수 있는 UI 구축
- 편의성 및 사용자의 편리성 반영

4. 주요 기능

- 캘린더 (달력)의 날짜 및 시간을 볼 수 있는 기능
- 캘린더에 일정을 추가하고 삭제하는 기능
- 로그인 및 회원가입 기능
- 일정을 공유하는 그룹을 생성 및 삭제하는 기능
- 그룹에 인원을 추가하고 삭제하는 기능
- 일정에 메모기능 - 일정이 다가왔을 시 알람기능
- 기간 및 일정의 시간을 설정하는 기능
- 그룹에 속해있는 사람에게 일정을 공개 및 비공개 하는 기능
- 그룹에 속해있는 사람들의 약속 및 미팅을 최적의 날짜 및 시간에 잡아주는 기능

5. 기대효과

바쁜 일정들을 일괄적으로 추가하고 공유함으로써 일정을 놓치지 않고 효율적으로 관리할 수 있으며, 팀 프로젝트, 약속, 소모임 활동 등 서로의 시간을 조율하여 만남을 가져야함에 있어서 보다 더 쉽게 일정 및 약속을 잡을 수 있게 된다. 이로 인해 간단한 기능들로 약속을 잡는 시간을 줄여주고 이에 따라서 캘린더를 활용해 하루, 한주, 한달의 시간을 알차게 분배하여 사용할 수 있는 기대를 한다. 더 나아가 그룹을 공유하는 사람들끼리의 커뮤니티 형성에도 간접적으로 도움을 줄 수 있고, 기존의 사용자들이 자주 사용했던 캘린더와의 연동을 통해 편의성도 증대해볼 수 있다.

6. 관련 비즈니스 프로세스 또는 시스템

이 프로젝트에서 기획한 공유 캘린더는 현재 존재하는 여러 일정 관리 및 그룹 커뮤니케이션 시스템과 유사하면서도 차별화된 기능을 제공한다. 일정을 최적화 시켜 자동적으로 일정을 만들어주는것이 주요 요인이며 주요 비즈니스 프로세스는 다음과 같다.

- 사용자 등록 및 그룹 형성 : 사용자는 회원가입 후 자신만의 일정을 관리하고, 같은 그룹에 속한 사람들과 일정을 공유할 수 있다.
- 일정 추가 및 공유 : 사용자는 자신의 일정을 추가하고, 그룹 구성원과 해당 일정을 공유할 수 있다. 공유 여부는 사용자가 선택할 수 있다.
- 자동 일정 조율 및 약속 주선 : 그룹 내 사용자들이 일정을 기획하거나 팀 미팅 등 공동활동의 필요성이 있다면 시스템은 그룹 구성원의 일정 데이터를 기반으로 최적의 시간과 날짜를 추천한다.
- 알림 및 동기화 : 사용자는 앱을 통해 일정에 가까워지면 알림으로 일정에 대해 다시 한번 상기하고, 일정에 대해 변동 사항이 생겼을 경우 즉시 동기화 하여 업데이트 된다.

7. 예상되는 기술적 어려움

- 일정 최적화 알고리즘 선정 : 그룹 구성원의 일정 데이터를 기반으로 최적의 일정을 자동으로 추천하는 알고리즘을 구축하는것이 주된 과제이며 모든 일정을 반영한 최적화된 일정을 추천하기 위한 시간 복잡도를 해결하는것이 우선 과제이다.
- 캘린더 시스템 연동 : 기존에 사용자가 사용하던 캘린더 등과의 연동을 구현할때 데이터 호환성과 동기화 문제가 발생할 수 있다.
- 사용자 데이터 및 일정 데이터 서버 연동 문제 : 서버 통신을 통해 사용자 정보 및 일정 데이터가 정상적으로 전달되고 이에 대해서 실시간 동기화 및 공개여부에 따른 일정반영에 대한 섬세한 작업이 요해진다.
- 다양한 플랫폼 지원 : 직관적인 UI 및 사용하는 기능에 있어서 여러플랫폼에서도 동일한 성능을 보여줄 수 있게 UI 가 일관적이고 다양한 플랫폼에 적용될 수 있도록 하여야 한다.

8. 구현 계획

1. 요구사항 분석 및 설계 (프로젝트 세부 요구사항 분석 및 시스템구조, 데이터베이스 설계)
2. 백엔드 서버 개발 (서버 및 데이터베이스 구축)
3. 프론트엔드 개발 (UI 및 디자인 구축)
4. 자동 일정 최적화 기능 개발 (일정 최적화 알고리즘 구축 및 반영)
5. 서버 연동 및 테스트 (시스템 전반에 대한 디버깅 및 피드백 반영)
6. 최종 마무리 단계 (완성본에 기반한 정상적인 구동 총 확인 후 배포)

9. 고수준 일정

~ 9/27 프로젝트 제안서 제출

9.28 ~ 10/4 요구사항 분석 및 설계 (요구사항 분석서 제출)

10/5 ~ 10/18 백엔드 서버 개발 및 Github 와 같은 공동 개발환경 구축

10/19 ~ 10/25 UI 설계 및 디자인 (UI 설계 보고서 제출)

10/26 ~ 11/1 UI 및 디자인 구축

11/2 ~ 11/8 자동 일정 최적화 기능 개발 및 적용 (구현 기술 확정 보고서 제출)

11/9 ~ 11/15 최종 기능이 반영된 프로토타입 완성 (서버 연동 포함)

11/16 ~ 11/22 테스트 및 디버깅 과정 (크로스 플랫폼 여부 적용 토의도 진행)

11/23 ~ 11/29 최종 테스트 및 디버깅 (최종 프로젝트 완성)

11/30 ~ 12/6 프로젝트에 전반적인 점검 및 발표 (발표자료, 발표시연영상, 최종 보고서 제출)

12/7 ~ 12/13 프로젝트 상호 평가 및 후기 작성

- 매주 금요일 마감 기준으로 일정을 작성했으며 세부 일정은 변동될 수 있습니다. -