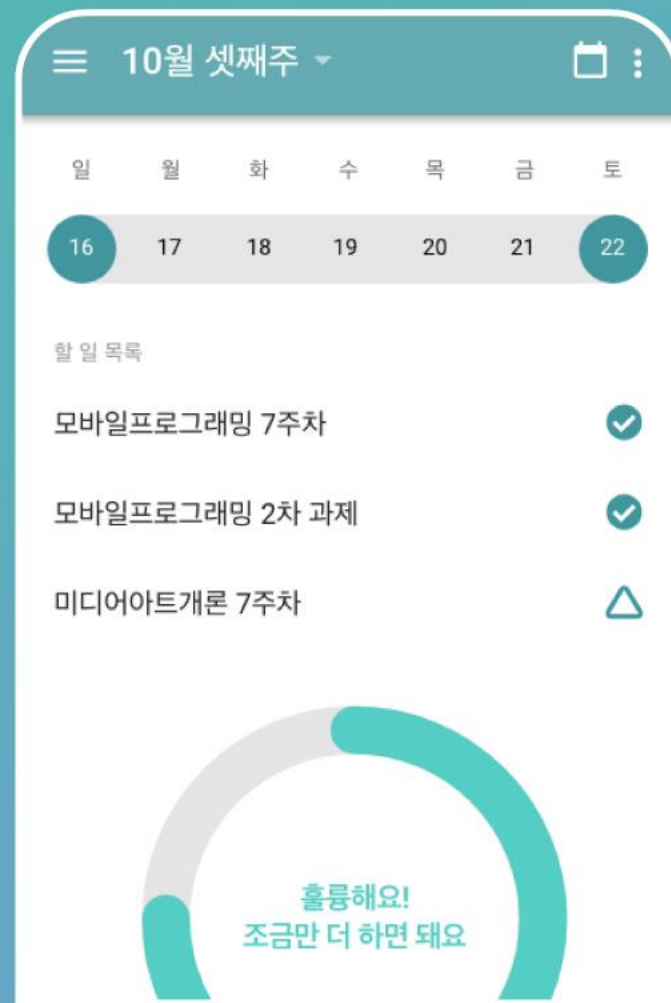


비대면 수업에 최적화된 스마트 스케줄러 앱 개발 제안

20162626 현지예 / 20192909 신지원



Aim & Background

온라인 수업, 과제, 할일
스페셜 스케줄러로 간편하게 관리하자!

시간표만 입력하면 간편하게 일정을 관리할 수 있어요



CORVID-19 이슈로 비대면 수업이 전면화된 현 시점, 학생들에게는 시간과 공간에 구애받지 않고 주도적으로 효율적인 시간관리를 할 필요와 기회가 주어졌다.

그러나 변화한 환경에 맞춤형된 일정관리 앱은 시중에 배포되어 있지 않다. 시중에 배포되어 있는 일정관리, 시간관리 앱으로는 매 주 반복되는 온라인 수업 일정을 등록하는 데에 번거로움이 따른다. 또한 스마트캠퍼스에도 기한 별로 과목별 진행 상태를 한번에 직관적으로 확인할 수 있는 기능이 없다.

따라서 **수업 시간표를 한 번만 등록하면 자동으로 매주 일정에 추가되며, 다른 일정들과 동시에 관리가 가능하여 학생들에게 최적화된 일정관리 어플리케이션을 개발하고자 한다.**

핵심 기능 및 기대효과

시간표 연동

매 주 할 일 목록에
온라인 수업이
시간표에 맞추어
자동으로 추가

자유자재 To-do list

주간동안 해야할 일
목록 중 선택하여
일간 할일 목록에
추가하거나 서로
이동할 수 있게 함

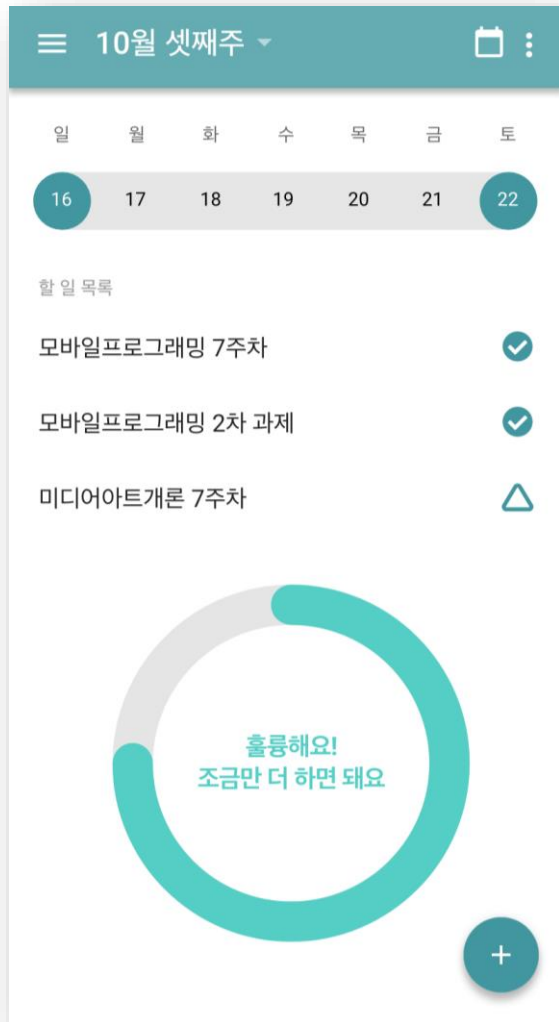
직관적인 완료도 체크

수행도 그래프로
주간 일정의 완료도를
직관적으로 확인할
수 있게 함

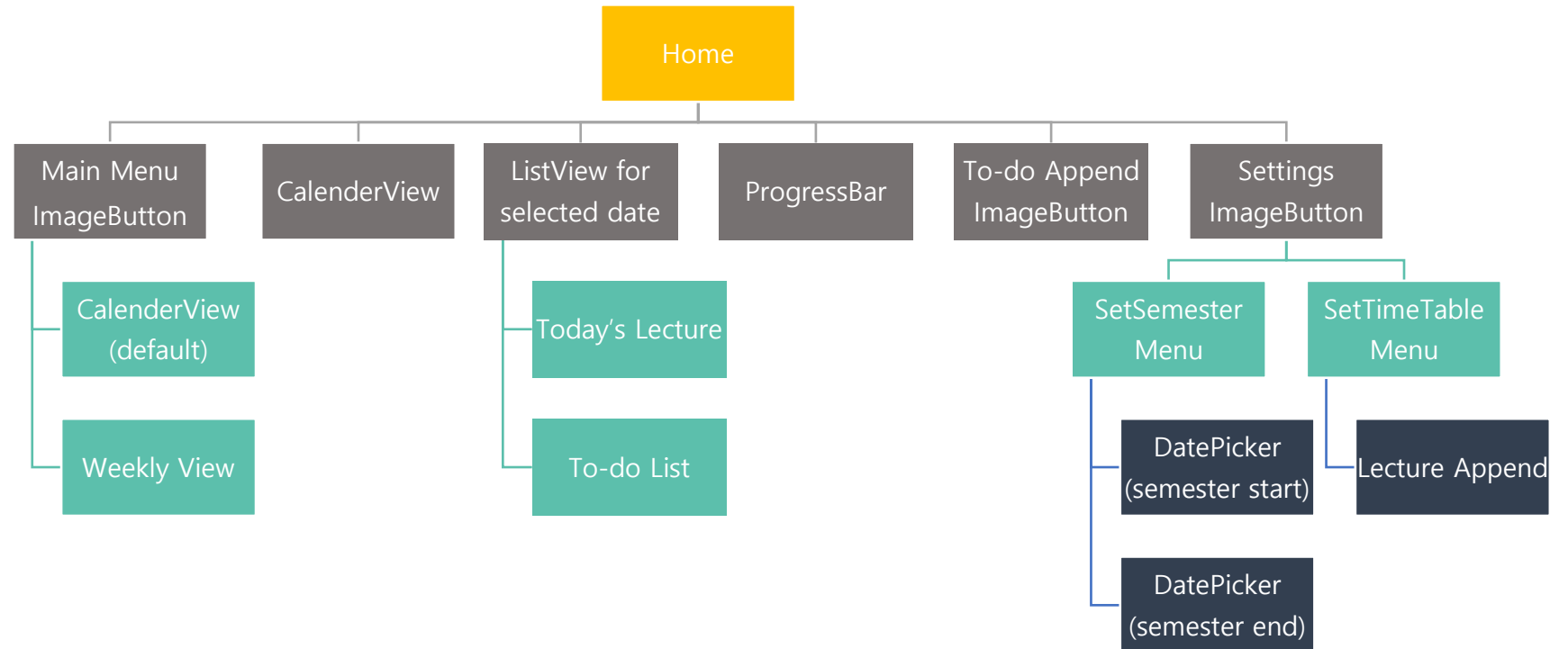
기대효과

- 프리미엄 모델로 앱 내
부분적인 기능을 유료화
- 추후 각 대학의 스마트
캠퍼스 데이터와 직접
연동 가능성

Contents



앱 구조도



Contents

I. Home(메인 화면)

- 날짜 표시

월간 캘린더는 CalendarView로 표시하고(default), 주간 캘린더는 별도로 구현하여 option에서 선택할 수 있게끔 하였다. 할일을 조회할 날짜를 선택할 수 있으며, default는 현재 날짜이다.

- To-do List 표시

ListView를 통해 캘린더에서 선택된 날짜에 맞는 할 일 목록(들어야 할 강의를 포함함)을 출력한다. 등록된 시간표에 따라 강의는 자동으로 To-do List에 표시되며, 사용자가 추가한 일정(과제, 시험 등)도 함께 표시한다.

- Progress Bar

원형 ProgressBar로 선택된 날짜의 할 일 완료도를 표시한다.

- Plus Button

FloatingActionButton을 클릭하면 새로운 일정을 추가하는 Dialog가 나타나며, To-do list에 추가할 수 있다.

II. Toolbar(상단 메뉴바)

Toolbar 위젯을 사용하여 커스텀 툴바를 구현하였다.

- View Option

ImageButton을 ContextMenu로 register하여, 메뉴에서 주간 뷰를 선택할 경우 CalendarView 대신 Weekly Calender를 표시하게 한다. Weekly Calendar의 경우 기존 Android Package가 존재하지 않으므로, 별도로 코딩하여 구현하였다.

- Settings

ImageButton을 ContextMenu로 register하여, 메뉴에서 학기 설정을 선택할 경우 개강일과 종강일을 DatePicker로 선택하는 Dialog가, 시간표 설정을 선택할 경우 강의를 추가할 수 있는 Dialog가 나타난다. 시간표 설정의 경우 강의명을 EditText, 요일을 RadioButton, 시작-종료시간을 TimePicker로 사용자가 입력할 수 있게 하였다.

Reference

- 참고 문헌: "모바일 애플리케이션의 수익화 전략 5가지", 위시켓 블로그,
<http://blog.wishket.com/%EB%AA%A8%EB%B0%94%EC%9D%BC-%EC%95%B1-%EC%88%98%EC%9D%B5-%EB%AA%A8%EB%8D%B8-%EC%A0%95%EB%A6%AC>
- 참고 서적: 『Android Studio를 활용한 안드로이드 프로그래밍(5판)』, 우재남, 박길식, 한빛아카데미(2020)
- 구현 참조: "주단위 달력 만들기", TaT Coding Blog,
http://kittenjunkwon.blogspot.com/2016/12/android_29.html
- 구현 참조: "플로팅 작업 버튼 추가", 안드로이드 개발자 문서,
<https://developer.android.com/guide/topics/ui/floating-action-button?hl=ko>

Appendix

- Full Code: <https://github.com/jiyehyeon/project>