

LOCAESCAPE

플러터를 이용한 나만의 캘린더 앱 만들기

01	프로젝트 주제 및 설명
02	요구사항 수립
03	계획일정 수립
04	주요 기능 설명
05	시행 착오
06	느낀점
07	데모 시현 영상

프로젝트 주제 및 소개



이름: 로케이스케이프(LOCATION + ESCAPE)

**주 기능: 위치 기반 API를 활용하여
내가 설정한 위치에서 벗어날 시 추후 안내를 알림으로 발송**

**알림 간격(강함: 15분 중간: 30분 약함: 1시간)
을 내가 직접 설정하여 계획의 강제성을 높임**

요구사항 수립

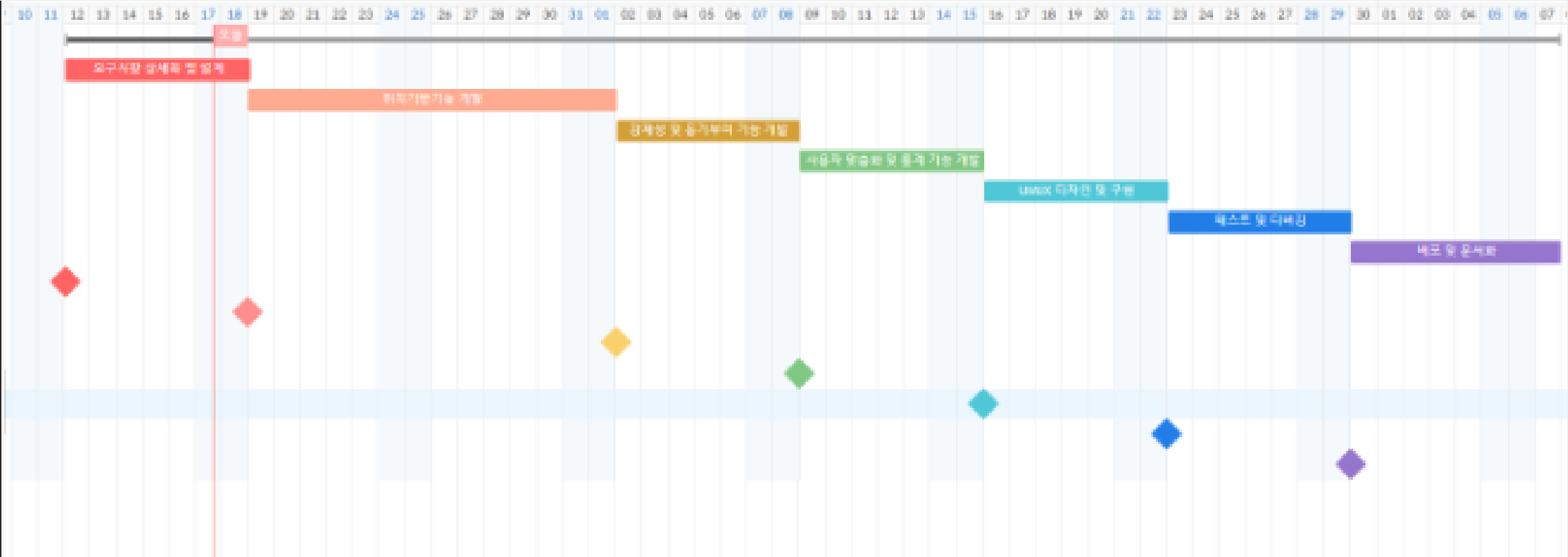
캘린더앱을 많이 사용해봤지만 알림이 잘 오지 않아 나도 모르는 사이에
“미룬이” 가 되었었다.

너무 복잡한 인터페이스로 인해 필요한 기능을 찾기가 어려움

주로 아침에 캘린더를 설정하는데 날씨를 같이 띄워주어 날씨에 대비했으면 좋겠다.

내 위치를 실시간으로 파악하여 그에 맞는알림을 발송해주는 어플이 필요하다고 느꼈다.

계획일정 수립



<div>요구사항 상세화 및 설계</div> <div>↑중간</div> <div>12/05/2025 - 19/05/2025</div>
<div>위치기반기능 개발</div> <div>↑중간</div> <div>19/05/2025 - 02/06/2025</div>
<div>강제성 및 동기부여 기능 개발</div> <div>↑중간</div> <div>02/06/2025 - 09/06/2025</div>
<div>사용자 맞춤화 및 통계 기능 개발</div> <div>↑중간</div> <div>09/06/2025 - 16/06/2025</div>
<div>UI/UX 디자인 및 구현</div> <div>↑중간</div> <div>16/06/2025 - 23/06/2025</div>
<div>테스트 및 디버깅</div> <div>↑중간</div> <div>23/06/2025 - 30/06/2025</div>
<div>배포 및 문서화</div> <div>↑중간</div> <div>30/06/2025 - 07/07/2025</div>

주요기능 설명

SCHEDULE_INPUT_DIALOG.DART

- 위치 권한 설정 코드

먼저 위치 권한을 요청 후 일정 추가 버튼
에서 지도 버튼을 누르면 실시간으로 사
용자의 위치를 파악하여 지도에 반영

```
// 위치 권한 확인
+ :: LocationPermission permission = await Geolocator.checkPermission();
    if (permission == LocationPermission.denied) {
      permission = await Geolocator.requestPermission();
      if (permission != LocationPermission.whileInUse &&
          permission != LocationPermission.always) {
        if (!mounted) return;
        ScaffoldMessenger.of(context).showSnackBar(
          const SnackBar(content: Text('위치 권한이 필요합니다')),
        );
        return;
      }
    }

// 현재 위치 가져오기
Position position = await Geolocator.getCurrentPosition(
  desiredAccuracy: LocationAccuracy.bestForNavigation,
);
if (!mounted) return;

// 구글맵 위치 선택 화면으로 이동
final selectedAddress = await Navigator.of(context).push(
  MaterialPageRoute(
    builder: (context) => GoogleMapLocationPicker(
      initialPosition: LatLng(position.latitude, position.longitude),
    ),
  ),
);

// 선택된 주소 업데이트
if (selectedAddress != null && mounted) {
  _locationController.text = selectedAddress;
}
} catch (e) {
  if (!mounted) return;
  ScaffoldMessenger.of(context).showSnackBar(
    SnackBar(content: Text('위치 정보를 가져오는데 실패했습니다: $e')),
  );
}
```

주요기능 설명

SCHEDULE_INPUT_DIALOG.DART

- 위치 이탈 시 문구 출력

설정해둔 위치와 100m 이상 차이가 나면 자동으로 알림 문구를 출력

```
// 현재 위치와 일정 위치의 거리 계산 후 문구 출력
FutureBuilder<double>(  
  future: _getDistanceFromCurrentLocation(schedule.locationLatLng),  
  builder: (context, snapshot) {  
    if (snapshot.connectionState == ConnectionState.done && snapshot.hasData) {  
      if (snapshot.data! > 100) { // 100m 이상 벗어났을 때  
        return Padding(  
          padding: const EdgeInsets.only(top: 8),  
          child: Text(  
            '할일 잘 마무리 하셨나요? 고생하셨습니다!',  
            style: TextStyle(  
              color: Colors.<i>blueGrey</i>,&br/>              fontStyle: FontStyle.italic,  
            ),  
          ),  
        );  
      }  
    }  
    return const SizedBox.shrink();  
  },  
)
```

주요기능 설명

CALENDAR_WIDGET.DART

- 캘린더 UI 생성 코드

TableCalendar 위젯을 사용하여 캘린더 생성 후 필수 파라미터 3개 (오늘, 선택날짜, 보여줄 마지막 날짜)를 선언하고 각각 파라미터에 기능 추가

```
Padding(  
  padding: const EdgeInsets.all(16.0),  
  child: TableCalendar<Schedule>(  
    focusedDay: _focusedDay,  
    firstDay: DateTime(2020),  
    lastDay: DateTime(2030),  
    locale: 'ko_KR',  
    calendarFormat: CalendarFormat.month,  
    headerStyle: HeaderStyle(  
      formatButtonVisible: false,  
      titleCentered: true,  
      leftChevronIcon: Icon(Icons.chevron_left, color: const Color(0xFF457B9D)),  
      rightChevronIcon: Icon(Icons.chevron_right, color: const Color(0xFF457B9D)),  
      titleTextStyle: GoogleFonts.comfortaa(  
        fontSize: 20,  
        fontWeight: FontWeight.bold,  
        color: const Color(0xFF457B9D),  
      ),  
    ),  
    calendarStyle: CalendarStyle(  
      defaultTextStyle: GoogleFonts.nunito(color: Colors.blueGrey[900]),  
      weekendTextStyle: GoogleFonts.nunito(color: Colors.redAccent),  
      todayDecoration: BoxDecoration(  
        color: const Color(0xFFA8DADC).withOpacity(0.5),  
        shape: BoxShape.circle,  
      ),  
      selectedDecoration: BoxDecoration(  
        color: const Color(0xFF457B9D),  
        shape: BoxShape.circle,  
      ),  
      markerDecoration: BoxDecoration(  
        color: Colors.orangeAccent,  
        shape: BoxShape.circle,  
      ),  
      markerSize: 8,  
      markerMargin: const EdgeInsets.symmetric(horizontal: 1),  
    ),  
  ),  
),
```


주요기능 설명

CALENDAR_WIDGET.DART - 캘린더 UI 생성 코드

각각의 파라미터에 이벤트 핸들러를 추가하여 기능 구현

```
selectedDayPredicate: (day) => isSameDay(day, _selectedDay),
eventLoader: (day) {
  final dateKey = DateTime(day.year, day.month, day.day);
  return _schedules[dateKey] ?? [];
},
onDaySelected: (selectedDay, focusedDay) {
  setState(() {
    _selectedDay = selectedDay;
    _focusedDay = focusedDay;
  });
  _showConditionalAddDialog(selectedDay);
},
calendarBuilders: CalendarBuilders(
  markerBuilder: (context, date, events) {
    if (events.isNotEmpty) {
      return Positioned(
        right: 1,
        bottom: 1,
        child: Container(
          padding: const EdgeInsets.all(6),
          decoration: const BoxDecoration(
            color: Colors.<i>orangeAccent</i>,
            shape: BoxShape.circle,
          ),
          child: Text(
            '${events.length}',
            style: GoogleFonts.<i>nunito</i>(
              color: Colors.<i>white</i>,
              fontWeight: FontWeight.<i>bold</i>,
              fontSize: 12,
            ),
          ),
        ),
      );
    }
    return null;
  },
),
),
),
),
),
```

시행착오

PUBSPEC.YAML에 의존성 추가 충돌 문제

여러개의 SDK를 사용하다보니 각가의 의존성이 충돌하여 FLUTTER PUB GET 명령어가 계속 실행되지 않음 => 각각의 SDK의 DEV를 일일이 찾아 들어가서 최적의 조합을 찾음

구글API 키 생성 및 추가 충돌 문제

구글 API를 사용하기 위해 결제수단 등록 및 어떤 API끼리 써야 호환성이 좋은지를 몰라 실행하지 못함
=> 각각의 API의 장단점을 분석 및 비교하여 최적의 조합을 구성

LLM 초반 딥러닝 미흡

PERPLEXITY PRO를 사용하였는데 하나의 오류를 가지고 계속 질문해도 오류사항을 계속 지적해주며 추가를 요청하였지만 어째서인지 같은말을 다섯번정도 반복 => 협박아닌 협박을 가하니 정신을 차림.

느낀점

처음 시도해보는 개발 툴이다 보니 많이 어색하기도 하고 어려운점도 많았다.

그래도 열심히 노력한 것 같아 뿌듯하기도 하다.

LLM이 없었더라면 성공할 수 없었을 것 같다.

이번 기회에 LLM을 효율적으로 사용하는 방법을 배운것 같다.

깃허브의 잔디도 앞으로 꾸준히 심어서 자기계발에 힘써야겠다.

식목일에 나무도 심고 깃허브 잔디도 심는 사람이 되어야겠다고 다짐했다.

데모 시현 영상 시청

감사합니다