기존 앱 비교 분석-SNS

| 앱 이름 | 카카오톡 | 인스타그램 |
| --- | --- | --- |
| 단체 채팅방이 있는가? | O | O |
| 일정 관리가 가능한가? | O | X |
| 그룹 멤버들과 목표를 세우고 실천할 수 있는가? | O | X |
| 디데이가 표시되는가? | O | X |
| 다른 사용자들과 함께 개인 일정을 공유할 수 있는가? | X | X |
| 진행 과정, 일상을 공유할 수 있는 피드가 있는가? | X | O |
| 일정 관리 대시보드(간트차트)가 있는가? | X | X |
| 장점 | 프라이빗한 공간(채팅)에서 그룹별로 이미지, 동영상, 파일 등 공유할 수 있음 | 1:1 또는 그룹으로 채팅방에서 정보 교환이 가능함 |
| 단점 | 개개인의 일정을 공유할 수 없으며 오직 채팅방 안에서만 정보를 교환할 수 있음 | 일정을 관리하는 기능 자체가 없으며 정보를 교환하는 피드가 공개적이므로 상당히 제한적임 |

기존 앱 비교 분석-프로젝트 관리 앱

| 앱 이름 | Notion | Microsoft Teams | Trello |
| --- | --- | --- | --- |
| 단체 채팅방이 있는가? | X | O | X |
| 일정 관리가 가능한가? | O | O | O |
| 그룹 멤버들과 목표를 세우고 실천할 수 있는가? | O | O | O |
| 디데이가 표시되는가? | O | O | O |
| 다른 사용자들과 함께 개인 일정을 공유할 수 있는가? | O | O | O |
| 진행 과정, 일상을 공유할 수 있는 피드가 있는가? | X | X | O |
| 일정 관리 대시보드(간트차트)가 있는가? | O | O | O |
| 장점 | 일정 관리가 용이하며 다른 사용자들의 템플릿 양식 공유 기능이 있음 | 단체 일정 관리가 중심인 앱으로 누구나 쉽게 사용할 수 있음 | 직관적이고 사용하기 쉬운 인터페이스를 제공 |
| 단점 | 진행 과정을 세세히 공유할 피드가 없음 | 정보를 교환하는 매개체가 오직 채팅방으로 제한적임 | 본적으로 무료이지만, 추가적인 고급 기능을 위해서는 프리미엄 구독을 해야 함, 피드가 공개적임, 채팅방이 없어 개인적인 정보 교환이 용이하지 않음 |

1. JavaScript

JavaScript는 다양한 기능을 수행할 수 있는 다목적 프로그래밍 언어입니다

이 언어는 시간과 날짜를 처리하는 다양한 함수를 제공하여 실시간 업데이트, 스케줄링, 타이머 등을 구현할 수 있으며, 배열, 객체, 맵, 세트 등 다양한 데이터 구조를 사용하여 데이터를 구성하고 처리할 수 있습니다.

더불어 JavaScript는 Node.js를 활용하여 서버 측 개발에도 사용됩니다. 이를 통해 서버에서 데이터베이스와 상호 작용하고 API를 제공하는 등의 작업을 수행할 수 있습니다. JavaScript는 웹 브라우저 뿐만 아니라 Node.js를 통해 서버 측 애플리케이션 개발에도 활용됩니다. 이로써 웹 프런트엔드와 백엔드를 동일한 언어로 개발할 수 있는 이점을 제공합니다.

또한, JavaScript 코드는 거의 모든 주요 플랫폼에서 동작하며, 동적 타입 언어로서 개발자에게 유연성을 제공하고 코드를 신속하게 작성하고 변경할 수 있는 장점을 가지고 있습니다.

JavaScript는 웹 개발에서 핵심 역할을 하며, 현대적인 웹 애플리케이션과 웹 사이트를 개발하는 데 필수적인 기술로 사용됩니다.

1. React-native - jsx

React Native는 JavaScript와 JSX(JavaScript XML)를 사용하여 모바일 앱을 개발하는 데 사용되는 프레임워크입니다.

이 프레임워크는 모바일 애플리케이션을 효율적으로 빠르게 개발할 수 있도록 도와주며, JSX를 활용하여 React Native 애플리케이션의 사용자 인터페이스(UI)를 정의하고 렌더링합니다. JSX는 JavaScript 코드 내부에 통합되어 있어 JavaScript 코드와 자연스럽게 결합되며, 앱의 UI를 구성하고 렌더링하는 데 사용됩니다.

React 및 React Native는 가상 DOM을 활용하여 성능을 최적화하며, 효율적인 UI 업데이트와 화면 갱신을 실현하여 빠른 앱 실행을 가능하게 합니다.

React Native는 iOS와 Android 플랫폼에서 동일한 코드베이스를 공유할 수 있도록 지원하므로, 두 플랫폼에서 애플리케이션을 쉽게 배포하고 유지 보수할 수 있습니다. 이를 통해 개발자는 효율적으로 크로스 플랫폼 모바일 앱을 개발할 수 있습니다.

1. Firebase - NoSQL

Firebase는 Google에서 제공하는 클라우드 기반 백엔드 서비스 및 개발 플랫폼으로, NoSQL 데이터베이스를 사용하여 데이터를 저장하고 관리할 수 있습니다.

NoSQL 데이터베이스는 데이터의 크기와 부하에 따라 관리가 용이합니다. 더 많은 서버 노드를 추가하여 처리 능력을 쉽게 확장할 수 있으므로 대규모 애플리케이션에서도 좋은 성능을 유지할 수 있으며 데이터베이스에 대한 엑세스 권한을 간편하게 설정하고 관리하여 데이터 보안과 사용자 권한을 효과적으로 관리할 수 있습니다.

뿐만 아니라, Firebase의 Realtime Database는 실시간 데이터 동기화를 지원하여 데이터 변경 사항을 실시간으로 클라이언트에 전달하고 업데이트된 내용을 즉시 표시할 수 있게 합니다.

Firebase는 데이터베이스 외에도 웹 호스팅 및 클라우드 함수와 같은 다른 서비스를 제공하여 애플리케이션을 손쉽게 호스팅하고 서버리스 기능을 통합할 수 있습니다. 이를 통해 개발자들은 애플리케이션의 백엔드 인프라를 효율적으로 구축하고 관리할 수 있습니다.

1. Android Studio

Android Studio는 안드로이드 앱 개발에 사용되는 통합 개발 환경(IDE)입니다.

Android Studio의 레이아웃 에디터는 사용자 인터페이스(UI)를 시각적으로 디자인하고 레이아웃을 조정하는 기술을 사용하기에 용이합니다.

안드로이드 기기의 시뮬레이터로 사용할 수 있는 가상 디바이스(에뮬레이터)를 제공하여 이를 통해 다양한 안드로이드 버전 및 화면 크기에서 앱을 테스트할 수 있습니다.

1. Webpack

Webpack은 모듈 번들러로서, 웹 개발에서 자바스크립트, 스타일시트, 이미지 및 다른 리소스와 같은 다양한 파일을 하나로 묶고 최적화하는데 사용됩니다. 복잡한 웹 애플리케이션의 코드를 구성 요소로 분리하고 관리하기 쉽게 만들어 코드의 가독성을 향상시키고 개발자가 모듈 단위로 작업할 수 있습니다.

주요 목적은 모듈화된 자원을 효율적으로 번들링하고 최적화하여 웹 애플리케이션의 성능을 향상시키고 개발자의 생산성을 높이는 것입니다. 이로써 복잡한 웹 프로젝트를 효과적으로 관리하고 최신 웹 개발 도구와 트렌드에 부합하는 애플리케이션을 구축할 수 있습니다.

1. Xcode

Xcode는 Apple의 통합 개발 환경(IDE)으로, Apple의 공식 언어인 Swift를 지원하므로 더욱 안전하고 효율적인 코드를 작성할 수 있습니다.

Xcode의 주요 목적은 Apple의 다양한 플랫폼 및 기기에서 동작하는 애플리케이션을 개발하고 배포하는 것입니다. 이를 통해 개발자는 iOS, macOS, watchOS 및 tvOS에서 동작하는 고품질의 앱을 개발할 수 있으며, 다양한 기능 및 서비스를 활용하여 사용자의 질을 향상시킬 수 있습니다. Xcode는 개발 생산성을 높이고 개발자가 앱을 디자인, 개발, 테스트 및 배포하는 모든 단계를 지원하는 도구입니다.

1. Gradle

Gradle은 빌드 자동화 도구로서, 주요 목적은 소프트웨어 프로젝트를 효율적으로 빌드하고 관리하는 것입니다.

Gradle을 사용하여 프로젝트를 자동으로 빌드하고, 코드 컴파일, 테스트 실행, 라이브러리 의존성 관리 등을 자동화하며 프로젝트를 모듈화하고, 각 모듈을 독립적으로 관리하여 복잡한 프로젝트를 보다 구조화된 형태로 관리합니다.

또한, Gradle은 필요한 외부 라이브러리를 쉽게 가져와 사용할 수 있어 개발자가 다양한 리소스를 효율적으로 활용할 수 있으며, 병렬 빌드와 캐싱을 통해 빌드 성능을 향상시킵니다. 이로써 개발자는 빠른 빌드 프로세스와 효율적인 프로젝트 관리를 실현할 수 있습니다.

1. Npm, Yarn

npm (Node Package Manager)은 JavaScript 및 Node.js 프로젝트에서 패키지와 라이브러리를 설치, 관리, 공유하기 위한 공식 패키지 매니저입니다. 이는 JavaScript 개발자들에게 프로젝트 의존성을 효과적으로 관리하고 크로스 플랫폼 환경에서 작업할 수 있는 도구로서 Node.js 환경에서 실행되며, Windows, macOS, Linux 등 다양한 플랫폼에서 작동합니다.

프로젝트의 package.json 파일을 통해 의존성을 정의하고, npm install 명령을 사용하여 필요한 패키지를 손쉽게 설치할 수 있습니다. 또한, package.json에서 정의한 스크립트를 실행하여 빌드, 테스트, 배포와 같은 작업을 자동화하는 데 활용됩니다. 주된 목적은 Node.js 환경에서 JavaScript 다양한 프로젝트에서 필요한 패키지를 효율적으로 관리 및 모듈을 관리하고 개발 프로세스를 단순화하는 것 입니다.

Yarn은 JavaScript 패키지 매니저로, npm과 유사한 목적을 가지고 있습니다. Yarn은 패키지 다운로드와 설치를 빠르게 처리하여 개발자의 작업 속도를 향상시키고, 병렬 설치 및 패키지 캐싱을 통해 빌드 시간을 최소화합니다.

Yarn의 주요 목적은 패키지 관리 시스템을 향상시켜 JavaScript 패키지의 더 빠른 다운로드와 설치, 보안 및 성능 향상을 제공하는 것입니다. Yarn은 npm의 단점을 보완하면서 JavaScript 개발자들에게 더 나은 패키지 관리 기능을 제공합니다.

참고문헌-<https://blog.naver.com/wandoo13/223196047546> (노션 기능)

-<https://thetigerhouse.tistory.com/entry/%EC%B9%B4%EC%B9%B4%EC%98%A4%ED%86%A1-%EC%9D%BC%EC%A0%95%EA%B4%80%EB%A6%AC-%EA%B8%B0%EB%8A%A5> (카톡 기능)

-<https://support.microsoft.com/ko-kr/office/microsoft-teams%EC%9D%98-%EC%83%88%EB%A1%9C%EC%9A%B4-%EA%B8%B0%EB%8A%A5-d7092a6d-c896-424c-b362-a472d5f105de> (Ms Teams 기능)

-<https://trello.com/guide/tips-tricks#stickers> (Trello 기능)