PC 간 자동 폴더 동기화 어플리케이션

*김정호*

목차

[PC 간 자동 폴더 동기화 어플리케이션 1](#_Toc145145367)

[개요 및 배경 설명 4](#_Toc145145368)

[해당 과제를 선택한 이유 4](#_Toc145145369)

[실제 업무 경험 4](#_Toc145145370)

[회사의 요구를 충족 4](#_Toc145145371)

[배경 설명 4](#_Toc145145372)

[사용할 요소 기술 5](#_Toc145145373)

[프로그래밍 언어: Go, Typescript, React 5](#_Toc145145374)

[플랫폼: Wails 5](#_Toc145145375)

[기타 사항 5](#_Toc145145376)

[TCP/IP 연결 5](#_Toc145145377)

[파일 압축 기법 5](#_Toc145145378)

[PC 간 서버 역할 5](#_Toc145145379)

[네트워크 공부 6](#_Toc145145380)

[사용자 인터페이스 (UI) 6](#_Toc145145381)

[목표 및 주요 기능 7](#_Toc145145382)

[간편한 설정 및 사용 7](#_Toc145145383)

[자동 동기화 7](#_Toc145145384)

[효율적인 대역폭 관리 7](#_Toc145145385)

[다중 플랫폼 지원 7](#_Toc145145386)

[실시간 동기화 7](#_Toc145145387)

[로그 및 이력 기록 7](#_Toc145145388)

[동기화의 의미 8](#_Toc145145389)

[파일 동기화 8](#_Toc145145390)

[데이터 일관성 8](#_Toc145145391)

[실시간 동기화 8](#_Toc145145392)

[프로젝트의 중요성 9](#_Toc145145393)

[데이터 관리 편의성 9](#_Toc145145394)

[시간 절약 및 효율성 9](#_Toc145145395)

[데이터 손실 방지 9](#_Toc145145396)

[다중 플랫폼 지원 9](#_Toc145145397)

# 개요 및 배경 설명

## 해당 과제를 선택한 이유

### 실제 업무 경험

이 프로젝트 주제를 선택한 주요 이유는 회사에서 요구하는 솔루션을 개발하는 경험을 쌓기 위해서 입니다.

### 회사의 요구를 충족

이 프로젝트가 현재 회사에서 진행하고 있는 프로젝트의 요구 사항을 해결하는 데 도움이 되고 실무에서 필요한 기술과 역량을 향상시킬 수 있기 때문입니다.

## 배경 설명

데이터 관리와 파일 공유는 매우 중요한 요소입니다. 많은 사용자들이 여러 개의 컴퓨터나 기기를 사용하며, 파일 및 데이터를 다양한 위치에서 액세스해야 합니다. 그 결과, 데이터를 효과적으로 관리하고 동기화하는 필요성이 증가하고 있습니다.

기존의 동기화 도구는 사용자가 파일 및 폴더를 수동으로 선택하고 동기화해야 하는 경우가 많아 불편할 수 있습니다. 또한, 두 개 이상의 로컬 폴더 간에 변경 사항을 자동으로 동기화하는 옵션이 부족합니다.

이러한 문제를 해결하고자 **PC 간 자동 폴더 동기화 어플리케이션**을 개발하려는 것입니다. 이 어플리케이션은 사용자가 선택한 로컬 폴더들 간에 변경된 파일 및 폴더를 자동으로 감지하고 동기화하는 기능을 제공합니다.

# 사용할 요소 기술

## 프로그래밍 언어: Go, Typescript, React

Go 언어는 간결하면서도 효율적인 언어로 알려져 있습니다. 병행 처리, 메모리 관리 등의 기능을 갖추고 있어서 파일 동기화와 관련된 작업을 효과적으로 처리하기에 적합합니다.

TypeScript를 사용하면 코드의 타입 안정성을 높일 수 있으며,

React는 동적이고 반응적인 사용자 인터페이스를 쉽게 개발할 수 있게 해줍니다.

## 플랫폼: Wails

Wails는 Go 언어로 웹 기반 GUI 애플리케이션을 개발할 수 있도록 하는 프레임워크입니다. 이를 사용하여 사용자 친화적인 그래픽 사용자 인터페이스를 구축할 수 있으며, 이미 알고 있는 기술로 멋진 애플리케이션을 만들고 다양한 플랫폼에서 실행 가능한 애플리케이션을 개발할 수 있습니다.

Go 언어와 Wails 프레임워크를 사용하여 백엔드를 개발하고, 프론트엔드는 React와 TypeScript를 사용하여 효율적이고 사용자 친화적인 애플리케이션을 개발할 수 있습니다.

## 기타 사항

### TCP/IP 연결

프로젝트의 주요 목표는 두 개의 PC 간에 안전하게 파일을 전송하는 것입니다. 이를 위해 TCP/IP 프로토콜을 사용하여 안정적인 데이터 통신을 구현해야 합니다.

### 파일 압축 기법

파일 전송을 최적화하고 대역폭을 절약하기 위해 파일 압축 기법을 구현해야 합니다. 압축 알고리즘을 선택하고 파일을 효과적으로 압축 및 압축 해제해야 합니다.

### PC 간 서버 역할

두 개의 PC가 서로 서버 역할을 수행해야 하는 경우, 네트워크 통신 및 데이터 교환을 위한 서버 및 클라이언트 모드를 지원해야 합니다. 이렇게 함으로써 양쪽 PC가 파일을 주고받을 수 있습니다.

### 네트워크 공부

TCP/IP 프로토콜 및 네트워크 통신에 대한 기본적인 이해가 필요합니다. 네트워크 구성, 라우팅, 포트 관리 등에 대한 공부와 구현이 프로젝트의 핵심 부분일 것입니다.

### 사용자 인터페이스 (UI)

Wails 프레임워크를 사용하여 사용자 친화적인 UI를 개발해야 합니다. 파일 선택, 전송 진행 상황 표시, 오류 처리 등을 고려해야 합니다.

# 목표 및 주요 기능

이 프로젝트를 통해 사용자들은 데이터 동기화를 더 효율적으로 관리하고 파일 공유를 더욱 용이하게 할 수 있을 것입니다.

## 간편한 설정 및 사용

사용자가 쉽게 폴더를 추가하고 동기화 설정을 구성할 수 있어야 합니다.

## 자동 동기화

변경된 파일 및 폴더를 자동으로 감지하고, 이를 선택한 로컬 폴더들 간에 자동으로 동기화해야 합니다.

## 효율적인 대역폭 관리

네트워크 대역폭을 효율적으로 활용하여 사용자 경험을 최적화해야 합니다.

## 다중 플랫폼 지원

다양한 운영체제(Windows, macOS, Linux)에서 사용할 수 있어야 합니다.

## 실시간 동기화

사용자가 선택한 방식으로 데이터를 동기화할 수 있어야 합니다.

## 로그 및 이력 기록

동기화 활동에 대한 로그를 기록하고, 문제 해결을 위해 필요한 정보를 제공해야 합니다.

# 동기화의 의미

**동기화**는 데이터나 정보의 일관성을 유지하고 다른 위치나 장치 간에 변경 사항을 반영하는 과정을 나타냅니다. 이는 다음과 같은 의미를 포함합니다.

**동기화**는 데이터의 일관성 및 업데이트를 관리하고 다양한 위치 또는 장치 간에 정보를 공유하는 데 필수적인 작업으로 프로젝트의 핵심 기능 중 하나입니다.

## 파일 동기화

파일 및 폴더의 변경 사항을 감지하고, 이를 다른 위치나 장치와 동일하게 유지하도록 하는 프로세스입니다. 이를 통해 변경된 파일이나 폴더를 자동으로 복사, 이동 또는 업데이트할 수 있습니다.

## 데이터 일관성

데이터베이스나 서버와 같은 다양한 시스템 간에 일관성을 유지하기 위해 데이터를 업데이트하고 동기화합니다. 이를 통해 여러 사용자가 동일한 정보를 수정할 수 있습니다.

## 실시간 동기화

데이터 변경 사항을 실시간으로 반영하거나, 사용자가 설정한 시간에 동기화 작업을 수행하는 방식을 포함합니다.

# 프로젝트의 중요성

**PC 간 자동 폴더 동기화 어플리케이션** 프로젝트는 데이터 관리와 파일 공유를 위한 중요한 도구로서 다음과 같은 이유로 중요성을 가집니다.

## 데이터 관리 편의성

프로젝트를 통해 사용자는 다수의 컴퓨터나 기기 간에 데이터를 효율적으로 관리할 수 있습니다. 이로써 파일 및 폴더를 별도의 작업 없이 자동으로 동기화하면서 일관성을 유지할 수 있습니다.

## 시간 절약 및 효율성

수동으로 파일을 복사하거나 이동하는 번거로운 작업을 줄여 줍니다. 자동 동기화를 통해 변경된 내용이 즉시 반영 되므로 작업 효율성이 향상됩니다. 수동으로 파일을 복사하거나 이동하는 번거로운 작업을 줄여줍니다. 자동 동기화를 통해 변경된 내용이 즉시 반영 되므로 작업 효율성이 향상됩니다.

## 데이터 손실 방지

중요한 데이터를 여러 기기 간에 일관되게 유지하면서 데이터 손실을 방지합니다. 이로써 실수나 장치 고장으로 인한 데이터 손실을 최소화합니다.

## 다중 플랫폼 지원

여러 운영체제에서 작동하는 어플리케이션으로, 다양한 사용자에게 도움을 제공할 수 있습니다.

다양한 운영체제(Windows, macOS, Linux)에서 사용할 수 있어야 합니다.