

산학협력프로젝트 수행계획서

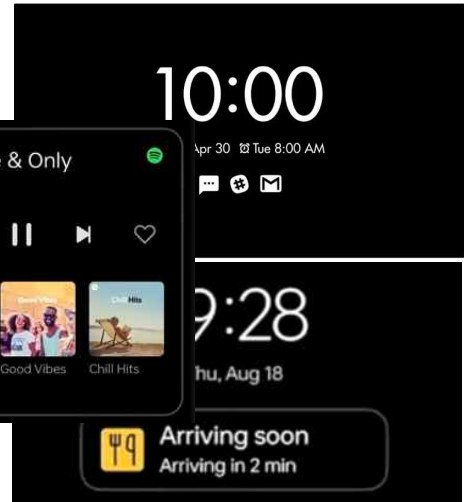
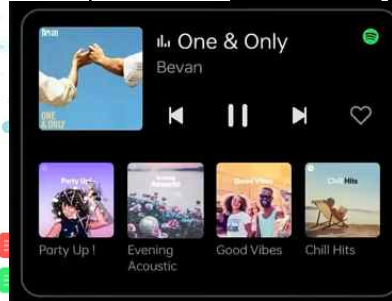
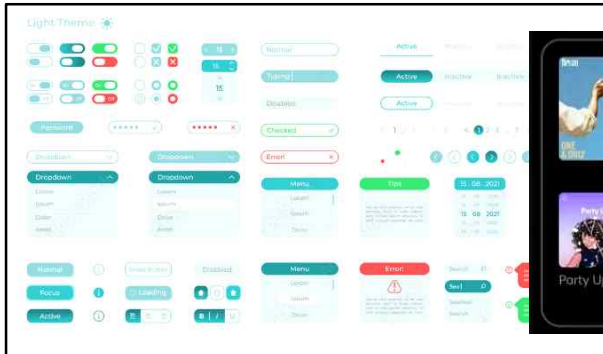
과제명	개인 맞춤형 IdleView 앱/위젯 개발		
협력기관명	LG전자	과제멘토	이동훈 책임
책임교수	정설영	소속	IT대학 컴퓨터학부
참여인원	(총 5명) 기업체 1명, 참여교수 1명, 학부과정 3명		
수행기간	2024.09.01.~12.31.(4개월)	유형	중기
추진배경	<p>○ 디스플레이가 대기 중 상태인 경우 사용자가 의미 있는 정보를 전달 받을 수 있는 앱 개발이 필요</p> <p>○ 배터리 사용량을 최소화한 상태에서 다양한 정보를 제공하는 기능을 구현함으로써, 대기 중인 상태에서도 실시간 혹은 지속적으로 정보를 얻을 수 있는 앱 개발을 목표로 함</p>		
목표 및 내용	<p>○ IdleView 앱</p> <ul style="list-style-type: none"> - 기기를 사용하지 않을 때 최소한의 전력으로 사용자 맞춤형 화면을 지속적으로 제공하는 IdleView 앱 개발 <p>○ 웹 앱 기능</p> <ul style="list-style-type: none"> - 사용자 인증 시스템 <ul style="list-style-type: none"> • 로그인 기능 구현 • 개별 사용자 휴면 설정 불러오기 <ul style="list-style-type: none"> ✓ 로그인 사용자의 IdleView 화면 표시 ✓ 로그인된 사용자 맞춤형 대시보드 및 인터페이스 컨트롤러 제공 - 화면 및 위젯 설정 관리 <ul style="list-style-type: none"> • 대시보드 <ul style="list-style-type: none"> ✓ 사용자 대시보드에서 IdleView 화면에 표시할 기능 선택 및 저장 ✓ 설정 화면 사전 보기 제공 • 인터페이스 컨트롤러 <ul style="list-style-type: none"> ✓ 웹 앱에서 기능을 실시간으로 선택하여 화면에 표시 - IdleView 앱 맞춤형 기능 <ul style="list-style-type: none"> • 기기가 설정된 IdleView 화면 표시 <ul style="list-style-type: none"> ✓ 이미지 슬라이드 쇼 ✓ 날씨 정보 표시 (온도/강수/미세먼지 등) ✓ 일정 목록(캘린더, 할 일 등)ü 미디어 재생 ✓ +@ (IdleView에 표시할 수 있는 기능 추가 개발) ✓ IdleView 실행 20초 후 Dimming 상태로 전환(배터리 소모 최소화) <p>○ JS Service 기능</p> <ul style="list-style-type: none"> - 미디어 플레이 <ul style="list-style-type: none"> • IdleView 앱 실행 시 미디어 목록의 콘텐츠를 연속 재생 • 화면 밝기 감소 상태에서도 미디어 플레이 유지 - 정보 모니터링 & 분석 <ul style="list-style-type: none"> • 실시간 데이터 모니터링 및 기존 데이터 분석을 통해 사용자에게 필요한 정보 제공 		

✓ 외출 시 준비물 알림, 약 복용 시간 알림 등

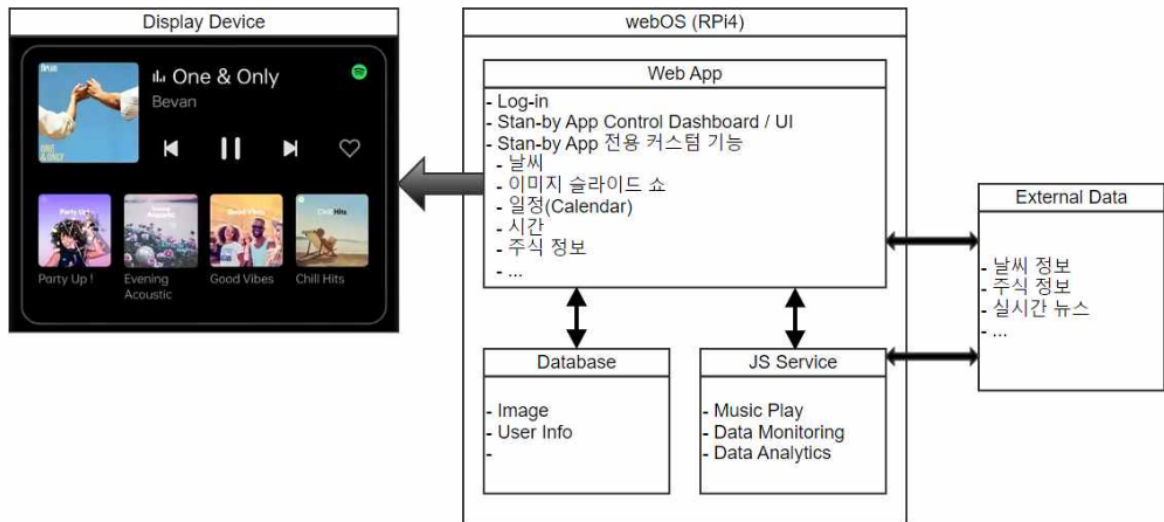
○ 주요 언어

- HTML5, CSS, Node.js, TypeScript

- Dashboard를 통한 스탠바이 앱 화면 설정 및 관리



- 시스템 다이어그램



기대효과

- 기기가 대기 상태에서도 최소한의 배터리 사용으로 사용자가 원하는 정보나 기능을 표기할 수 있는 IdleView 앱 기능 확보

1. 과제 목적 및 필요성

1.1. 과제 목적

- 1.1.1. 본 과제는 대기 중인 디스플레이에서 사용자 맞춤형 정보를 효율적으로 제공하는 IdleView 앱의 중요성을 인식하고, 이를 구현하기 위한 첨단 기술 개발에 중점을 둡니다.
- 1.1.2. 배터리 소모를 최소화한 상태에서 사용자 맞춤형 대시보드와 실시간 정보 제공 기능을 개발하고, 웹OS 플랫폼에서 동작하는 IdleView 앱을 통해 디스플레이가 대기 중일 때도 유용한 정보를 제공하는 것을 목표로 합니다.
- 1.1.3. 본 과제의 결과는 사용자 경험 개선, 전력 효율성 향상, 그리고 스마트 디스플레이의 정보 전달 역량을 크게 확대할 것으로 기대됩니다..

1.2. 과제 필요성

1.2.1. IdleView 시스템의 중요성: 스마트 디바이스에서 대기 화면의 효율적인 활용은 사용자 경험을 크게 향상시킬 수 있으며, 특히 사용자가 필요로 하는 정보를 맞춤형으로 제공하는 IdleView 시스템은 공공 장소, 상업 시설, 가정에서 모두 중요한 역할을 할 수 있습니다.

1.3. 결론

사용자의 요구를 반영한 맞춤형 IdleView 시스템을 제안하고, 실시간 정보 제공 기능을 통합하여 배터리 소모를 최소화하는 효율적인 솔루션을 개발하고자 합니다. 이를 통해 스마트 디바이스의 대기 화면이 단순한 대기 상태를 넘어, 실제로 사용자에게 유용한 정보를 제공하는 플랫폼으로 기능할 수 있을 것으로 기대됩니다.

2. 과제 내용 및 추진 방법

본 과제의 목표는 WebOS 플랫폼에서 동작하는 Idle View 시스템을 신규로 개발하는 것입니다. Idle View 시스템은 기기 유휴 상태에서 사용자 맞춤형 정보를 화면에 표시하고, 다양한 컨트롤 기능을 제공하여 사용자 경험을 향상시키는 역할을 합니다.

2.1. 개발 목표

2.1.1. 유저 맞춤형 정보 표시

유저가 설정한 대시보드에서 Idle View 화면에 표시할 정보를 선택하고, 다양한 위젯을 구성할 수 있는 기능을 개발합니다. 기기 유휴 상태에서 유저가 선호하는 정보(날씨, 뉴스, 일정 등)를 실시간으로 제공할 수 있도록 할 예정입니다.

2.1.2. 에너지 효율

기기가 Idle View 화면에서 장시간 구동될 경우 배터리 소모를 최소화하기 위한 에너지 절약 기능을 개발합니다. 예를 들어, 유휴 상태 20초 후 화면이 Dimming 상태로 전환되도록 구현하여 전력 소모를 줄일 수 있습니다.

2.1.3. 인터페이스 컨트롤러

Idle View 화면에서 표시되는 정보와 인터페이스를 제어할 수 있는 유연한 컨트롤러를 제공하며, 유저가 설정한 시간 또는 조건에 따라 화면을 전환하거나 업데이트할 수 있습니다.

2.2. 추진 방법

2.2.1. 프론트엔드 개발: React 및 Enact 라이브러리를 사용하여 사용자 대시보드와 Idle View의 UI를 개발합니다. 특히, WebOS에서 효율적으로 동작할 수 있도록 최적화된 컴포넌트를 사용하며, 사용자 맞춤형 대시보드 및 위젯 관리 기능을 구현합니다.

- Node.js: 프론트엔드와 백엔드 모두 JavaScript 기반의 Node.js를 사용하여 일관된 실행 환경을 제공할 수 있음.
- TypeScript: 자바스크립트의 확장 언어로, 타입 시스템을 통해 코드의 안정성을 보장하고 오류를 사전에 방지할 수 있음.
- Vite: HMR을 지원하는 빌드 도구이자 매우 빠른 빌드 속도가 장점, 개발 속도를 크게 향상시킬 수 있음.
- Enact: Enact는 WebOS에 최적화된 React 기반 프레임워크로, WebOS 플랫폼에서 필요한 기능을 쉽게 구현할 수 있음.
- React-Router-Dom: 단일 페이지 애플리케이션(SPA)의 라우팅을 담당하여, 사용자 경험을 자연스럽게

전환할 수 있음.

- Styled-Components: CSS-in-JS 방식으로 컴포넌트 기반의 스타일링이 가능하며, 유지보수성과 가독성이 높아짐.
- Axios: HTTP 요청 처리를 위한 라이브러리로, API 호출과 비동기 작업을 쉽게 구현할 수 있음.

2.2.2. 백엔드 개발: Node.js 및 Express를 기반으로 사용자 설정 저장 및 데이터를 제공하는 서버를 구축합니다. Prisma ORM을 활용하여 MySQL 데이터베이스와 연동하며, 사용자 선호 데이터를 저장하고 관리하는 기능을 개발합니다.

- Express: 백엔드에서 가장 널리 사용되는 경량 웹 프레임워크로, 빠르고 간단하게 서버 구축이 가능함.
- Prisma: 데이터베이스와의 상호작용을 쉽게 할 수 있는 ORM 도구로, TypeScript와의 호환성이 뛰어나고, MySQL과 같은 관계형 데이터베이스를 쉽게 관리할 수 있음.

2.2.3. 에너지 절감 알고리즘: Idle View 실행 시 일정 시간이 지나면 화면 밝기를 조절하거나, 배경 데이터를 처리하는 빈도를 낮추는 에너지 절감 알고리즘을 구현합니다. 이를 통해 전력 소모를 최소화하고, 사용자 편의를 고려한 기능을 추가할 예정입니다. 리소스를 절약하기 위해 React의 useCallback혹을 사용하여 불필요한 렌더링을 줄일 수 있으며, 애니메이션의 경우 requestAnimationFrame을 활용하여 성능을 최적화할 계획입니다. 또한, Enact에서도 유사한 기능이 있는지 확인할 예정입니다.

2.3. 유닛 테스트 및 최적화: 개발된 시스템의 기능 테스트를 통해 발생할 수 있는 문제를 조기에 발견하고, 성능 최적화를 위해 WebOS 플랫폼에 특화된 유닛 테스트를 진행합니다.

- 유닛 테스트 (Unit Testing): 각 기능(로그인, 위젯 관리, 에너지 절약 모드)에 대해 유닛 테스트를 수행하여 개별 컴포넌트가 올바르게 작동하는지 확인합니다.
- 통합 테스트 (Integration Testing): 개별 기능 테스트가 완료된 후, 통합 테스트를 통해 시스템의 모든 구성 요소(프론트엔드, 백엔드, 데이터베이스)가 잘 작동하는지 검증합니다.
- 사용자 테스트 (User Testing): 몇 명의 사용자에게 앱을 사용해 보도록 요청하여 기능, 사용 용이성, 성능에 대한 피드백을 제공하는 간단한 사용성 테스트를 진행합니다.
- 성능 테스트 (Performance Testing): 앱이 성능 요구 사항을 충족하는지 확인하기 위해, 유휴 및 활성 상태에서 배터리 사용량을 모니터링합니다.
- 회귀 테스트 (Regression Testing): 업데이트나 변경 후, 새로운 코드가 기존 기능을 손상시키지 않는지 확인하기 위해 회귀 테스트를 수행합니다.

2.4. 소통 계획

프로젝트의 성공적인 진행을 위해 팀원 간의 원활한 소통이 중요합니다. 이를 위해 다음과 같은 커뮤니케이션 계획을 수립합니다.

● 정기 회의:

매주 금요일 오후 6시에 정기 회의를 진행합니다. 이 회의에서는 현재 진행 중인 작업, 문제점, 다음 주 목표 등을 논의합니다. 회의는 Zoom 또는 Google Meet과 같은 화상 회의 플랫폼을 통해 진행됩니다.

● 소통 도구:

- ✓ 카카오톡: 팀원 간의 빠른 소통을 위해 카카오톡 그룹 채팅을 활용합니다. 긴급한 문제나 빠른 질문은 카카오톡을 통해 실시간으로 해결할 수 있습니다.

- ✓ 디스코드: 프로젝트 관련 자료 공유 및 비공식적인 소통을 위해 디스코드 서버를 운영합니다. 특정 주제에 대한 채널을 만들어 팀원들이 자유롭게 의견을 나눌 수 있도록 합니다.

- 회의록 작성:

각 회의 후, 회의록을 작성하여 팀원들에게 공유합니다. 회의록에는 논의된 내용, 결정사항, 할당된 작업, 다음 회의 일정 등을 포함합니다. 이를 통해 모든 팀원이 동일한 정보를 공유하고, 후속 작업을 명확히 할 수 있도록 합니다.

- 작업 조정 및 이정표 관리:

각 팀원은 자신의 작업 상태를 정기적으로 업데이트하고, 진행 중인 작업과 이정표에 대한 피드백을 제공합니다. 이를 통해 팀 전체의 진행 상황을 투명하게 관리할 수 있습니다.

3. 과제 추진 일정 및 예산 활용 계획

세부 추진 내역	기술 개발 기간														
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
계획 수립															
요구사항 분석															
api 명세 작성															
UI 디자인															
로그인 기능 구현															
대시보드 기능 구현															
IdleView 앱 맞춤형 기능 구현															
JS service 기능 구현															
최종 테스트 및 피드백															
github 기술 문서 작성															
최종 발표 준비 및 보고서 작성															

(단위 : 천원)

구 분			현 금	현 물	소 계
2024. 09. 01. ~ 2024. 12. 31.	직접비	내부인건비			
		외부인건비			
		학생인건비			
		연구장비.재료비	200		
		연구활동비			
		연구과제추진비			
		연구수당 (해당시)			
		소 계			
합 계			200		

4. 기대효과 및 활용방안

4.1. 기술적 관점

Idle View 시스템은 전력 소모를 최소화하면서도 유저 맞춤형 정보를 제공합니다. 이를 통해 WebOS 플랫폼에서 더욱 최적화된 사용자 경험을 제공할 수 있습니다.

4.2. 경제적 관점

배터리 소모를 줄이는 Idle View 시스템은 장기적으로 전력 비용을 절감할 수 있습니다.

4.3. 사용자 편의성

Idle View 시스템은 유휴 상태에서도 필요한 정보를 사용자에게 즉시 제공하여 편의성을 높이며, 사용자 맞춤형 대시보드와 같은 기능으로 개인화된 경험을 제공합니다.

4.4. 다양한 적용 가능성

본 시스템은 여러 환경에서 활용 가능하며, 스마트 디스플레이 및 정보 제공 시스템의 발전에 기여할 수 있습니다.

5. 예상되는 주요 과제성과

본 프로젝트는 개발 과정을 GitHub에 공개할 계획입니다. 이를 통해 다양한 개발자들이 프로젝트에 참여하거나 활용할 수 있도록 하는 데 중점을 둡니다. 또한, 기술 문서를 작성하여 향후 프로젝트 확장 및 유지보수에 도움을 줄 수 있도록 하며, 이를 통해 개발 과정의 투명성을 높이고, 커뮤니티와의 협업을 촉진할 예정입니다.

6. 참여인력(세부)

지도교수	소속	경북대학교 컴퓨터학부		성명	정설영
참여인력 (산업체)	기업명	성명	직위	전화	Email
	LG전자	이동훈	책임		
과제 참여 학생	소속(학과)	학위과정 (성별)	학번	성명	담당업무
	컴퓨터학부	학사과정 (여)	2022117062	게니시 톿쿠	프론트엔드 개발
	컴퓨터학부	학사과정 (남)	2022116301	김승진	데이터베이스 개발
	컴퓨터학부	학사과정 (남)	2022111441	이강인	백엔드 개발 (팀장)