

창작물 거래 플랫폼

팀 명 : 팀제이
지도 교수 : 이병천 교수님
팀 장 : 이승규
팀 원 : 김경빈
오종찬
정윤조
이규한

2024. 11.

중부대학교 정보보호학과

목 차

1.서론

- 1.1 연구배경
- 1.2 연구 목적 및 필요성

2.관련연구

- 2.1 node js
- 2.2 vue.js
- 2.3 javascript
- 2.4 mysql
- 2.5 tosspayment
- 2.6 nas

3. 서비스소개

- 3.1 front
- 3.2 back

4.결론

- 4.1 결론,기대효과
- 4.2 보완사항

5.팀원소개 및 소스코드 공유

6.별첨 자료

1. 서론

1.1 연구 배경

무명 예술인들의 작품을 알리기 위한 서비스가 존재하지만, 접근성이 부족하고 실질적인 수입이 창출되지 않아서 정부차원에서 보조사업을 지원하고 있다. 그래서 예술인들의 자립을 위한 플랫폼 기획하였고, 더 나아가 자신 작품을 홍보하고, 포트폴리오로 활용할 수 있게 여러 서비스를 개발하려고 한다.

1.2 연구 목적 및 필요성

미술품 거래 사이트는 지역적 제약을 없애고 전 세계적인 시장 접근을 가능하게 합니다. 예술가나 갤러리뿐만 아니라 수집가도 특정 지역에 구애받지 않고 다양한 작품을 접하고 거래할 수 있습니다. 온라인 플랫폼은 작품의 가격, 작가 정보, 작품의 상태 등에 대한 명확한 정보를 제공하여 거래의 투명성을 높입니다. 또한 과거 거래 기록을 바탕으로 시세 분석이나 경향 파악이 가능해, 더 신중한 구매 결정을 돕습니다. 온라인 미술품 거래 사이트는 물리적인 공간에 가지 않아도 편리하게 거래할 수 있어, 바쁜 현대인들에게 시간과 비용을 절약해 줍니다. 또한 다양한 결제 방식과 배송 옵션을 제공해 사용자의 편의성을 극대화합니다. 전통적인 경매나 갤러리 시스템에서는 유명한 작가들이 주로 조명을 받는 경우가 많습니다. 그러나 온라인 플랫폼은 신진 작가들이 자신의 작품을 소개할 수 있는 기회를 제공하여, 그들이 더 넓은 시장에서 인지도를 높일 수 있도록 돕습니다. 미술품 거래 사이트는 미술품을 소유하고 싶은 일반 대중들에게도 더 쉽게 접근할 수 있는 통로를 제공합니다. 전통적인 갤러리나 경매에서는 높은 진입 장벽이 있지만, 온라인 거래 플랫폼에서는 다양한 가격대의 작품을 제공하여 대중들이 미술품 수집에 참여할 수 있는 기회를 확대합니다. 미술품 거래 사이트는 미술 시장을 더 투명하고 효율적으로 운영할 수 있게 하며, 다양한 참여자가 시장에 진입할 수 있는 기회를 제공합니다. 이는 전반적으로 미술 시장의 성장을 촉진하며, 더 많은 사람들이 예술 작품을 즐기고 거래할 수 있도록 돕습니다.

2. 관련 연구

2.1 Node JS

Node JS는 확장성 있는 네트워크 애플리케이션(특히 서버 사이드) 개발에 사용되는 소프트웨어 플랫폼으로, Chrome V8 JavaScript 엔진으로 빌드된 JavaScript 런타임 플랫폼이다. 작성 언어로 자바스크립트를 활용하며 논블로킹(Non-blocking) I/O와 단일 스레드 이벤트 루프를 통한 높은 처리 성능을 가지고 있다. 내장 HTTP 서버 라이브러리를 포함하고 있어 웹 서버에서 아파치 등의 별도의 소프트웨어 없이 동작하는 것이 가능하며 이를 통해 웹 서버의 동작에 있어 더 많은 통제를 가능케 한다. 블록 체인에 저장하는 거래 내역과 같은 데이터 이외의 채팅, 부가적인 사용자 정보 등을 Node JS의 프레임워크인 Express를 사용하여 만든 서버를 통해 저장하고 사용한다.

2.2 Vue

Vue.js는 자바스크립트 기반의 프론트 엔드 프레임워크로, 사용자 인터페이스를 구축하는 데 주로 사용된다. 매우 유연하고 반응성이 뛰어나며, SPA(싱글페이지어플리케이션)개발에 적합하다. 컴포넌트 기반 구조로써 작은 재사용 가능한 컴포넌트로 애플리케이션을 구축한다. 각각의 컴포넌트는 독립적으로 개발 및 유지보수할 수 있어 코드의 가독성과 재사용성을 높인다. 다른 프레임워크에 비해 매우 가볍고 성능이 우수하다. 특히 초기 로드 속도가 중요한 애플리케이션에 적합하다. 소규모 프로젝트에서 대규모 프로젝트에 이르기까지 쉽게 확장할 수 있다. 점진적인 도입이 가능해, 개발자의 필요에 맞춰 프레임워크를 확장할 수 있다. Vue.js는 학습 곡선이 낮고, 성능이 뛰어나며, 유연한 구조 덕분에 다양한 웹 애플리케이션에 적합한 프레임워크이다.

2.3 JavaScript

자바스크립트(영어: JavaScript)는 객체 기반의 스크립트 프로그래밍 언어이다. 이 언어는 웹 브라우저 내에서 주로 사용되며, 다른 응용 프로그램의 내장 객체에도 접근할 수 있는 기능이 있다. 또한 Node.js와 같은 런타임 환경과 같이 서버 프로그래밍에도 사용되고 있다. 자바스크립트는 본래 넷스케이프 커뮤니케이션즈 코퍼레이션의 브렌던 아이크(Brendan Eich)가 처음에는 모카(Mocha)라는 이름으로, 나중에는 라이브스크립트(LiveScript)라는 이름으로 개발하였으며, 최종적으로 자바스크립트가 되었다. 자바스크립트가 썬 마이크로시스템즈의 자바와 구문이 유사한 점도 있지만, 이는 사실 두 언어 모두 C 언어의 기본 구문에 바탕을 뒀기 때문이고, 자바와 자바스크립트는 직접적인 연관성은 약하다. 이름과 구문 외에는 자바보다 셸프나 스킴과 유사성이 많다. 자바스크립트는 ECMA스크립트(ECMAScript)의 표준 사양을 가장 잘 구현한 언어로 인정받고 있으며 ECMAScript 5(ES5)까지는 대부분의 브라우저에서 기본적으로 지원되었으나 ECMAScript 6 이후부터는 브라우저 호환성을 위해 트랜스파일러로 컴파일된다.

2.4 MySQL

MySQL은 오픈 소스 관계형 데이터베이스 관리 시스템(RDBMS)으로, 많은 웹 애플리케이션과 시스템에서 데이터베이스로 널리 사용된다. 또한 성능, 확장성, 유연성을 갖춘 오픈 소스 관계형 데이터베이스로, 다양한 규모의 애플리케이션에서 사용될 수 있다. 특히 웹 애플리케이션과 같이 빠르고 안정적인 데이터 처리가 중요한 환경에 적합하다. 대규모 트랜잭션 처리나 고급 데이터베이스 기능이 필요하지 않은 경우, MySQL은 비용 효율적인 데이터베이스 솔루션으로서 뛰어난 선택이다.

2.5 toss payments

비바리퍼블리카에서 제공하는 간편 결제 및 결제 대행(PG, Payment Gateway) 서비스입니다. 토스페이먼츠는 주로 온라인 상거래 환경에서 결제 처리 및 관련 서비스를 제공하여, 기업과 소비자가 더욱 편리하고 안전하게 결제를 할 수 있도록 지원한다. 토스페이먼츠는 안전하고 신뢰성 높은 결제 환경을 제공하며, 특히 소규모 온라인 상점이나 스타트업에게 적합한 결제 대행 서비스이다. 다양한 결제 수단을 제공하고, 빠르고 간편하게 통합할 수 있는 점에서 많은 기업들에게 인기를 끌고 있다. 또한, 토스페이먼츠는 합리적인 수수료와 빠른 정산으로 사업자들에게 큰 장점을 제공한다.

2.6 NAS

NAS(Network Attached Storage)는 네트워크에 연결된 저장 장치로, 데이터를 중앙에서 저장하고 여러 사용자나 장치가 네트워크를 통해 접근할 수 있도록 해준다. NAS는 가정, 중소기업, 대기업 등 다양한 환경에서 데이터를 쉽게 관리하고 공유할 수 있는 유용한 솔루션이다. 데이터 관리와 공유를 위한 중앙화된 솔루션으로, 개인 사용자부터 기업까지 다양한 규모의 사용자에게 유용한 장비이다. NAS를 통해 데이터를 안전하게 저장하고, 필요에 따라 네트워크를 통해 다양한 장치에서 접근할 수 있으며, 백업 및 데이터 보호 기능도 제공받을 수 있다. 특히 대용량 데이터를 다루거나 데이터 접근성이 중요한 경우, NAS는 필수적인 장비로 자리 잡고 있다

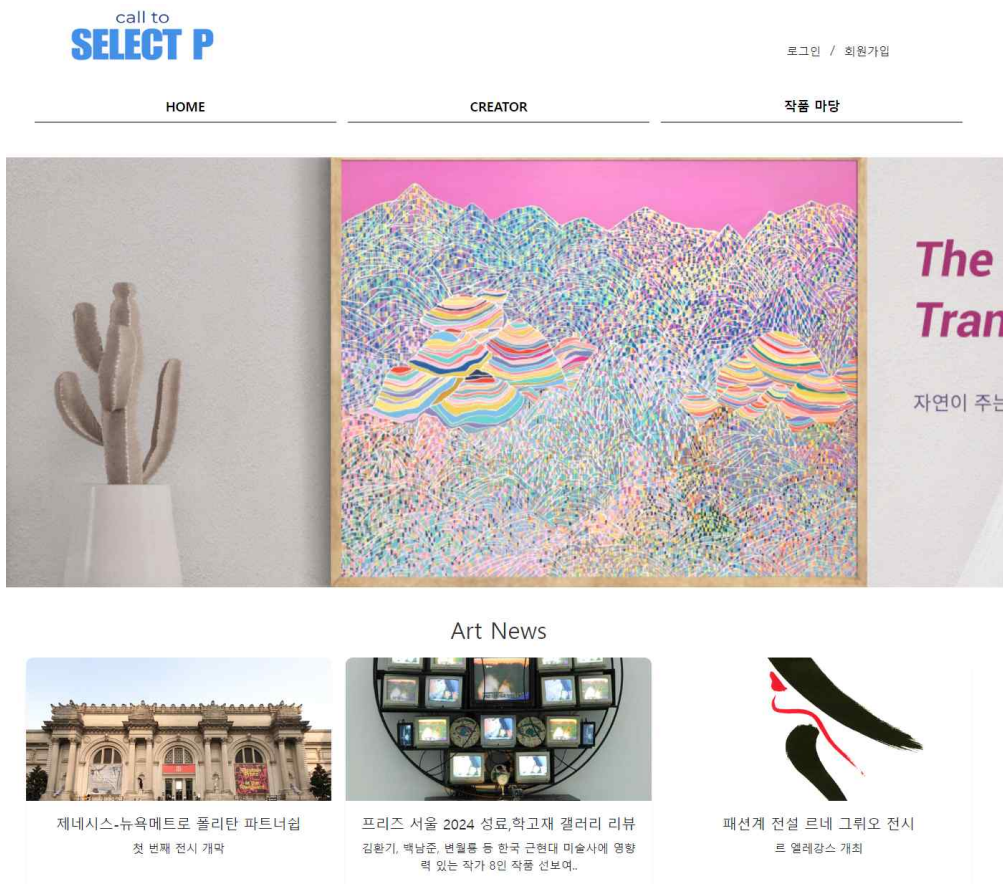
2.6.1 Synology NAS

DSM(디스크스테이션 매니저, DiskStation Manager): 시놀로지 NAS에서 가장 큰 강점 중 하나는 DSM이라는 자체 운영체제이다. DSM은 웹 기반의 직관적이고 사용하기 쉬운 관리 인터페이스를 제공하여, NAS를 설정하고 운영하는 과정을 매우 간편하게 만든다. NAS는 파일 서버 기능 외에도 다양한 애플리케이션을 통해 미디어 서버, 클라우드 스토리지, 백업 서버, 웹 서버, 이메일 서버 등으로 활용할 수 있다. NAS는 RAID 0, 1, 5, 6, 10 등 다양한 RAID 구성을 지원하여, 데이터 보호 및 성능 최적화를 위한 옵션을 제

공한다. 강력한 보안 옵션을 제공하여, 네트워크 접근 제어, 방화벽, VPN, 2단계 인증 등의 기능을 통해 NAS를 안전하게 보호한다.

3. 본론

3.1 서비스소개



메인페이지

메인페이지는 사용자들이 웹사이트에 처음 접속했을 때 마주하는 첫 화면으로, 전체적인 서비스의 소개 및 작품들을 보여준다. 하단에는 최신 아트 뉴스 섹션이 배치되어 있다. 이 페이지는 전체 서비스의 첫인상을 제공하는 만큼, 사용자 경험을 고려하여 시각적으로 깔끔하고 직관적인 구조를 유지하고 있다.

기능설명

로그인/회원가입 페이지는 사용자가 웹 애플리케이션에 접근하기 위해 로그인할 수 있는 인터페이스를 제공하는 Vue 컴포넌트이다. 이 페이지는 사용자 인증을 처리하고, 성공적

으로 인증된 사용자가 애플리케이션에 접근할 수 있도록 한다.

회원가입/로그인 기능구현

1. 이메일과 비밀번호를 입력할 수 있는 두 개의 입력 필드를 제공합니다. v-model을 사용하여 사용자가 입력한 값을 컴포넌트의 데이터 속성과 바인딩합니다.
2. 이메일 형식이 맞는지 확인하는 기본적인 유효성 검사를 수행합니다. 사용자가 필수 입력 사항을 입력하지 않으면 에러 메시지를 표시합니다. 그 후 회원가입이 완료되어 db에 회원정보가 입력되면서 회원가입이 완료된다.
3. 로그인 요청시 사용자 입력값을 auth.js 모듈로 전달하여 로그인 API를 호출하여 인증 성공시 Vuex(세션)에 저장하며, 이후 페이지를 리다이렉트하여 애플리케이션 내부 페이지로 이동합니다. 실패 시 에러 메시지를 표시하여 잘못된 로그인 정보임을 사용자에게 알립니다.

프론트에서 사용자로부터 로그인 정보를 입력받고, 서버 모듈과 상호작용하여 사용자 인증을 처리한다. 백엔드 서버와의 통신을 통해 사용자 인증을 관리하고, 애플리케이션의 세션을 유지하며 보호된 페이지에 대한 접근을 제어한다.









일반 회원가입



간편 회원가입








크리에이터 목록

 <p>asdf 고전 명화 주로 다룹니다 프로필 보기</p>	 <p>망그러진 곰 안녕하십니까! 프로필 보기</p>	 <p>최고심 이쁜 사진 조아해여 프로필 보기</p>	 <p>햄깁 반가워요~! 프로필 보기</p>
 <p>9omdory 구미고미 프로필 보기</p>	 <p>주르르 버튜버예요 프로필 보기</p>	 <p>BBP bbp 프로필 보기</p>	 <p>Siu siu 프로필 보기</p>

크리에이터 페이지

크리에이터 페이지는 등록된 크리에이터들의 목록을 확인하고 각 크리에이터의 프로필을 상세히 볼 수 있는 기능을 제공한다. 이 페이지에서는 크리에이터가 등록한 작품 리스트와 프로필 이미지를 포함한 개인 정보가 표시된다. 각 크리에이터 프로필을 통해 사용자는 해당 크리에이터의 작품 활동 확인할 수 있으며, 크리에이터의 상세 페이지로 이동하여 더 많은 정보를 얻을 수 있다.

등록한 작품

 <p>이메일: 1@1 소개: 소개 정보가 없습니다.</p>	 <p>Landscape 에시1 5000원</p>	 <p>멋들어진 포스터 간지나요 4000원</p>	 <p>태스크보 그냥 멋있어서 소장중입니다 12000원</p>
	 <p>호랑이 조선시대에 그려진 것으로 추정 15000원</p>		

크리에이터 프로필 페이지

기능설명

크리에이터 목록을 사용자에게 표시하는 Vue 컴포넌트이다. 이 컴포넌트는 크리에이터 데이터를 백엔드에서 가져와 사용자가 확인할 수 있다. 주요 기능은 크리에이터 목록을 불러 오고, 각 크리에이터의 프로필로 이동할 수 있는 기능을 구현했다.

백엔드에서 제공된 creator.js API를 통해 서버로부터 크리에이터 목록을 불러옵니다. 이 목록은 v-for 디렉티브를 통해 페이지에 렌더링된다.

각 크리에이터는 이미지, 이름, 간단한 소개 등의 정보를 포함하며, 이 정보들은 axios를 통해 서버에서 동적으로 가져옵니다.

각 크리에이터 항목에는 "프로필 보기" 버튼이 제공되며, 사용자가 버튼을 클릭하면 해당 크리에이터의 id를 기반으로 개별 프로필 페이지로 리다이렉션한다. 프로필로 이동하면, 개별 크리에이터의 상세 정보를 제공하고, 등록된 작품 목록을 보여준다.

The screenshot shows the 'call to SELECT P' website interface. At the top left is the logo 'call to SELECT P'. On the right, there is a '로그아웃' (Logout) button with a user icon. Below the logo, there are navigation tabs: 'HOME', 'CREATOR', and '작품 마당' (Marketplace). A blue button labeled '작품 등록' (Register Work) is located in the top right corner. The main content area is titled '작품 목록' (Work List) and displays a grid of six artwork items, each with a thumbnail image, title, price, and a '작품 보기' (View Work) button.

작품 이미지	작품 제목	가격	작품 보기
	Landscape	5000원	작품 보기
	태스크보	12000원	작품 보기
	망그리진 만화2	2000원	작품 보기
	꽃	1000원	작품 보기
	기하학컬렉션	100000원	작품 보기
	호랑이	15000원	작품 보기

At the bottom of the page, there is a pagination control with buttons for '이전' (Previous), '1', '2', and '다음' (Next).

작품 페이지

작품페이지는 등록된 상품들의 목록을 사용자에게 보여주는 Vue 컴포넌트이다. 데이터베이스에 등록된 상품들의 목록을 불러와, 이를 그리드 형식으로 페이지에 렌더링한다. 각 상품은 이미지, 제목, 가격 등의 기본 정보를 포함하며, 사용자가 상품을 클릭하면 해당 상품의 상세 페이지로 이동할 수 있다.

상품 목록이 길어지면 페이지네이션 기능을 통해 한 페이지에 일정 수의 상품만 보여준다. 페이지 번호를 클릭하면 해당 페이지의 상품 목록을 서버에서 다시 요청하여 표시한다.

The screenshot shows the '작품 등록' (Product Registration) page. At the top, there is a navigation bar with 'HOME', 'CREATOR', and '작품 마당' (Product Market) links, along with a '로그아웃' (Logout) button and a user profile icon. The main heading is '작품 등록'. The form includes the following fields:

- 상품 제목** (Product Title): A text input field with the placeholder '작품 제목'.
- 상품 설명** (Product Description): A text area with the placeholder '간단한 상품설명'.
- 가격** (Price): A text input field with the value '1000000'.
- 카테고리 태그** (Category Tag): A text input field with the placeholder '태그 입력 후 Enter'. Below it, there are four tags: '태그는 ×', '커스텀 ×', '어무렇게 ×', and '입력이 가능합니다 ×'.
- 이미지 파일 업로드** (Image File Upload): A section with a '파일 선택' (File Select) button and a grid of 20 sports-related icons. Below the grid is a '상품 등록' (Product Register) button.

At the bottom of the page, there is a footer with links: 'Home', 'Features', 'Pricing', 'FAQs', and 'About'.

작품 등록 페이지

새로운 작품을 등록할 수 있도록 지원하는 Vue 컴포넌트이다. 사용자로부터 작품의 제목, 설명, 가격, 카테고리 태그 및 이미지 파일을 입력받아, 이를 서버로 전송하여 데이터베이스에 저장하는 기능을 수행합니다.

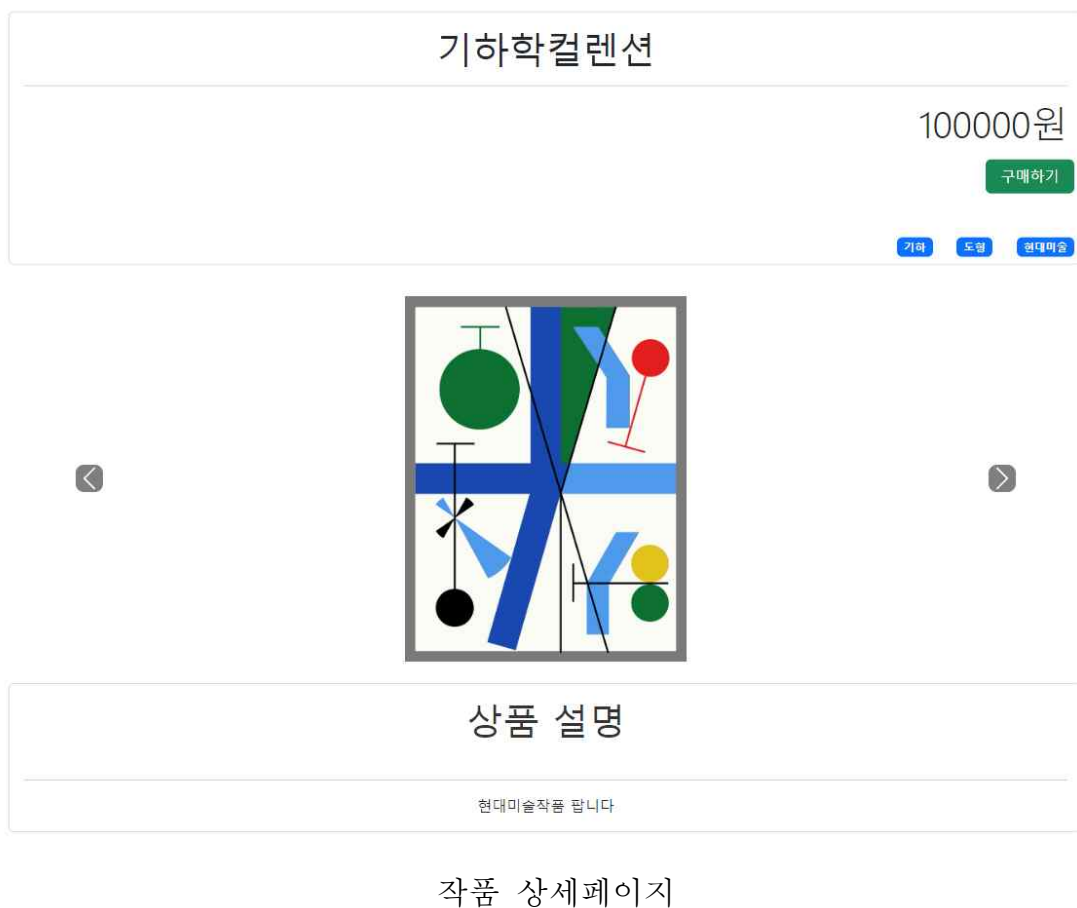
기능설명

사용자가 작품의 제목, 설명, 가격을 입력할 수 있는 입력 필드가 제공된다. 각 필드는 v-model을 통해 Vue의 양방향 데이터 바인딩을 사용하여 입력 데이터를 컴포넌트의

상태에 즉시 반영한다.

카테고리 태그는 사용자가 텍스트 필드를 통해 입력하고, Enter 키를 누르면 태그 목록에 추가되는 방식이다. 사용자는 여러 개의 이미지를 선택하여 업로드할 수 있으며, 선택한 이미지들은 미리보기로 사용자에게 즉시 보여진다. FileReader API를 사용하여 이미지 파일을 base64 형식으로 변환하여 미리보기를 제공한다.

사용자가 모든 정보를 입력한 후 "작품 등록" 버튼을 클릭하면 axios를 통해 서버에 상품 데이터를 전송한다. 이때 FormData 객체를 사용하여 이미지 파일과 작품 정보를 함께 서버에 전송하여 데이터베이스에 저장한다.



작품의 상세 정보를 보여주는 Vue 컴포넌트이다. 사용자는 이 페이지를 통해 선택한 상품의 제목, 가격, 이미지, 설명, 태그 등을 확인할 수 있으며, 구매하기 버튼을 통해 상품 구매 요청을 보낼 수 있다.

기능설명

axios를 통해 서버로부터 특정 상품의 상세 정보를 가져와, 화면에 렌더링한다. 상품의

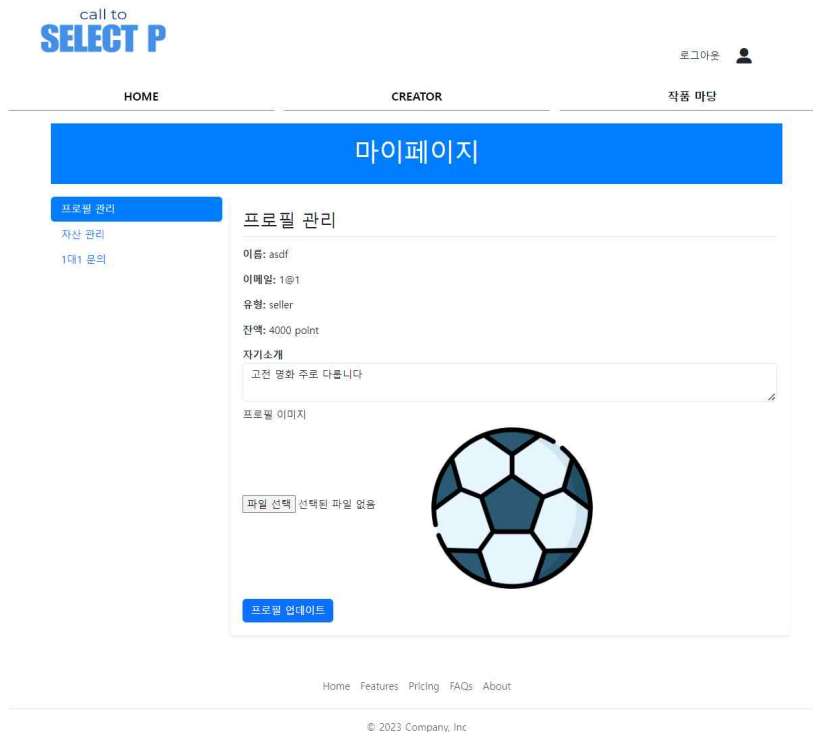
제목, 가격, 설명, 이미지 배열, 태그 등이 화면에 표시되며, 태그는 v-for 디렉티브를 사용해 각각의 태그가 배지 형식으로 나타납니다. 이미지들은 캐러셀 컴포넌트를 사용해 슬라이드 방식으로 보여준다. 각 이미지에 대한 v-for 반복문을 사용해 동적으로 상품 이미지를 표시한다.

"구매하기" 버튼을 클릭하면 axios를 사용해 서버로 구매 요청을 보냅니다. 이때 구매자의 ID와 상품 ID가 함께 전송되며, 성공적으로 구매가 완료되면 사용자에게 알림 메시지가 표시된다.

구매 관련 트랜잭션 기능은 상품을 구매하는 과정에서 구매자와 판매자 간의 거래가 이루어지고, 이 거래 과정에서 트랜잭션을 기록하여 구매 내역을 관리한다. 해당 로직은 주로 백엔드의 Products.js 모듈에서 처리한다.

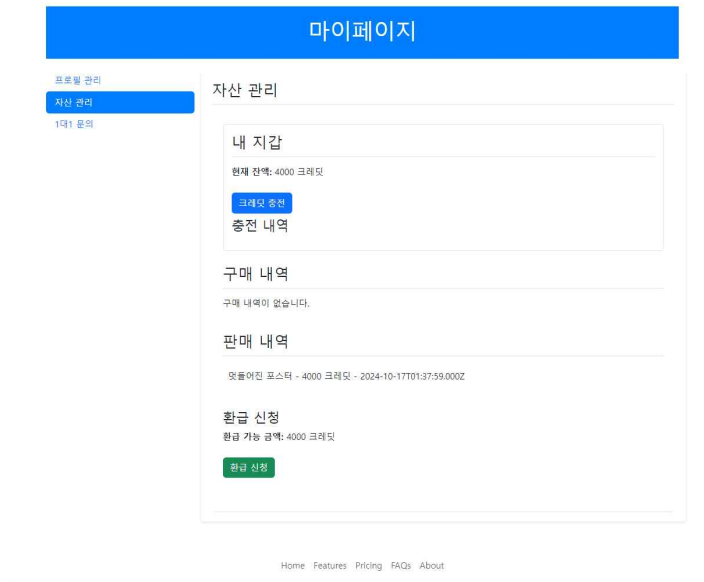
구매요청 > 구매자의 잔액확인 > 트랜잭션 생성 > 구매자의 잔액차감 > 판매자의 잔액증가

구매자와 판매자 간의 금전적 거래를 안전하고 효율적으로 처리하는 중요한 역할을 한다. 트랜잭션을 통해 모든 거래가 기록되고, 구매자의 잔액 관리, 판매자의 수익 관리가 이루어진다. 이를 통해 이중 거래 방지, 정확한 금액 처리, 거래의 추적 가능성을 보장하며, 구매 내역과 판매 내역도 체계적으로 관리할 수 있다.

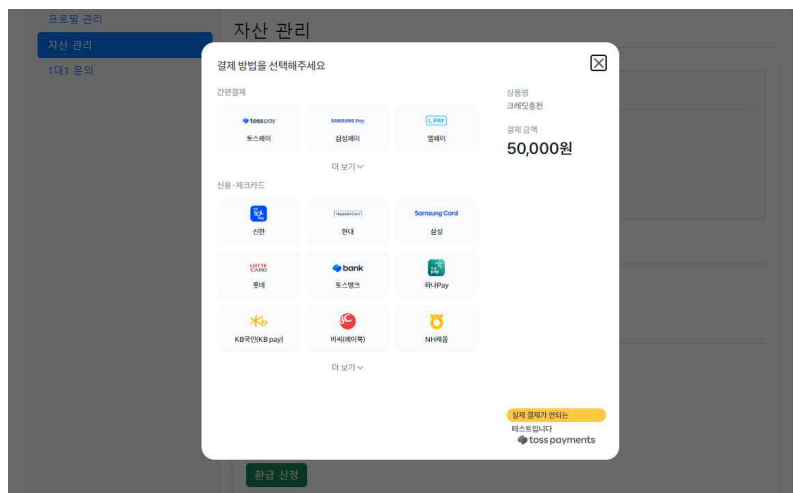


마이페이지

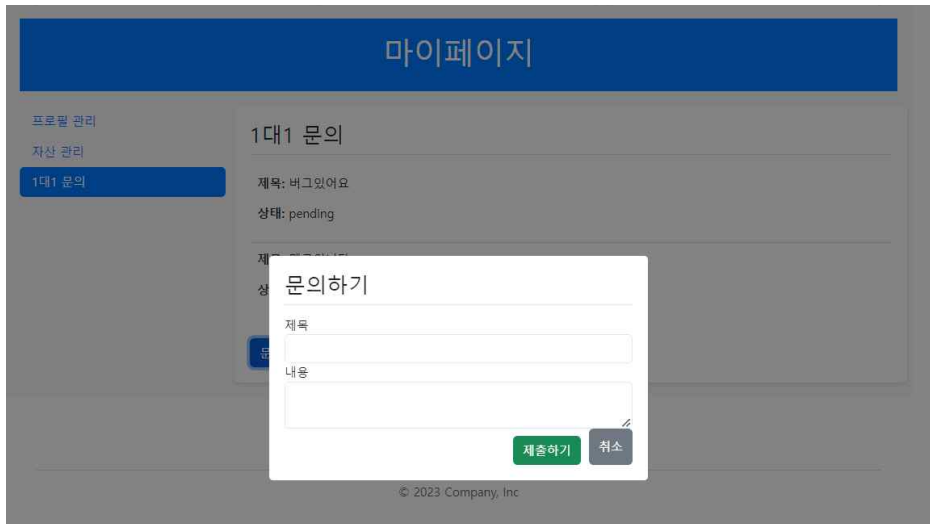
사용자의 프로필 정보를 관리할 수 있는 인터페이스를 제공한다. 사용자는 자신의 이름, 이메일, 잔액을 확인할 수 있으며 자기소개 를 작성하거나 프로필 사진을 업로드하는 기능을 사용할 수 있다. 이 기능은 v-model을 사용하여 양방향 데이터 바인딩을 구현하며, 변경된 정보를 서버로 전송해 업데이트할 수 있는 메커니즘을 제공한다.



사용자의 자산 현황을 관리하는 기능을 포함하고 있다. 사용자는 자신의 잔액을 확인하고, 구매 내역, 판매 내역, 그리고 환급 신청 내역을 조회할 수 있다. 이러한 기능은 자산 관리 탭에서 구현되며, 사용자가 이전에 수행한 거래 내역을 불러오고, 이를 리스트 형식으로 표시한다.



내 지갑 섹션에서는 사용자가 현재 보유 중인 크레딧 잔액을 확인할 수 있으며, 크레딧 충전 기능을 통해 추가 크레딧을 구매할 수 있다. pay 함수는 충전 방식을 선택하고, 이를 토스페이먼츠 API 등을 활용하여 실제 결제를 처리한다.



사용자가 1대1 문의를 통해 관리자에게 문의를 할 수 있는 기능을 제공한다. 사용자는 문의의 제목과 내용을 입력할 수 있으며, 이를 서버로 전송하여 저장한다. 문의는 modal 형태의 팝업 창을 통해 작성된다. 사용자는 자신이 작성한 문의 내역을 확인할 수 있으며, 각 문의의 답변상태를 볼 수 있다.



MySQL을 기반으로 설계되었으며, 사용자 관리, 제품 관리, 거래 관리 등 주요 기능을 지원하기 위해 여러 개의 테이블로 구성되어 있다. 각 테이블은 특정 기능을 담당하며, 사용자, 제품, 거래 관련 데이터를 효과적으로 저장하고 관리하기 위한 구조로 설계했다.

users 테이블은 사용자 정보를 저장하는 핵심 테이블이다. 사용자는 고유한 ID를 부여받으며, 이 테이블은 사용자 유형, 이메일, 비밀번호, 잔액 등의 정보를 포함한다. 이 테이블의 필드를 통해 사용자 프로필 관리, 로그인 및 인증 절차, 구매 및 판매 활동이 이루어진다.

products 테이블은 사용자가 등록한 상품 정보를 저장한다. 이 테이블은 판매자가 등록한 상품의 제목, 설명, 가격, 이미지, 태그 등을 관리하며, 구매자는 이 정보를 기반으로 상품을 선택하여 구매할 수 있다. products 테이블은 users 테이블과 연결되어 있어 각 상품이 어떤 사용자가 등록했는지를 추적할 수 있다.

transactions 테이블은 사용자 간의 거래 내역을 기록하며, 구매와 판매 정보를 관리하는 중요한 역할을 한다. 거래가 이루어질 때 구매자와 판매자의 정보, 거래된 상품, 거래 금액, 거래 상태(진행 중, 완료, 실패)를 기록한다. 이 테이블을 통해 구매 내역과 판매 내역을 구분하고, 거래와 관련된 재정적 변화를 추적할 수 있다.

inquiries 테이블은 1대1 문의 내역을 저장하며, 사용자와 관리자가 소통할 수 있는 기능을 지원한다. 사용자는 이 테이블을 통해 문의를 생성할 수 있으며, 관리자는 문의 상태를 확인하고 답변을 제공할 수 있다.

credits 테이블은 사용자가 크레딧을 충전할 때 그 내역을 저장한다. 사용자는 이 테이블을 통해 충전 내역을 확인하고, 자신의 크레딧 사용 이력을 관리할 수 있다.

withdrawals 테이블은 사용자가 환급 요청을 할 때 그 내역을 기록한다. 관리자는 이 요청을 검토하고 처리 상태를 업데이트할 수 있으며, 사용자는 자신의 환급 상태를 확인할 수 있다.

Synology Nas 웹서버 구축하기



Web Station 을 사용하여 웹사이트 호스팅

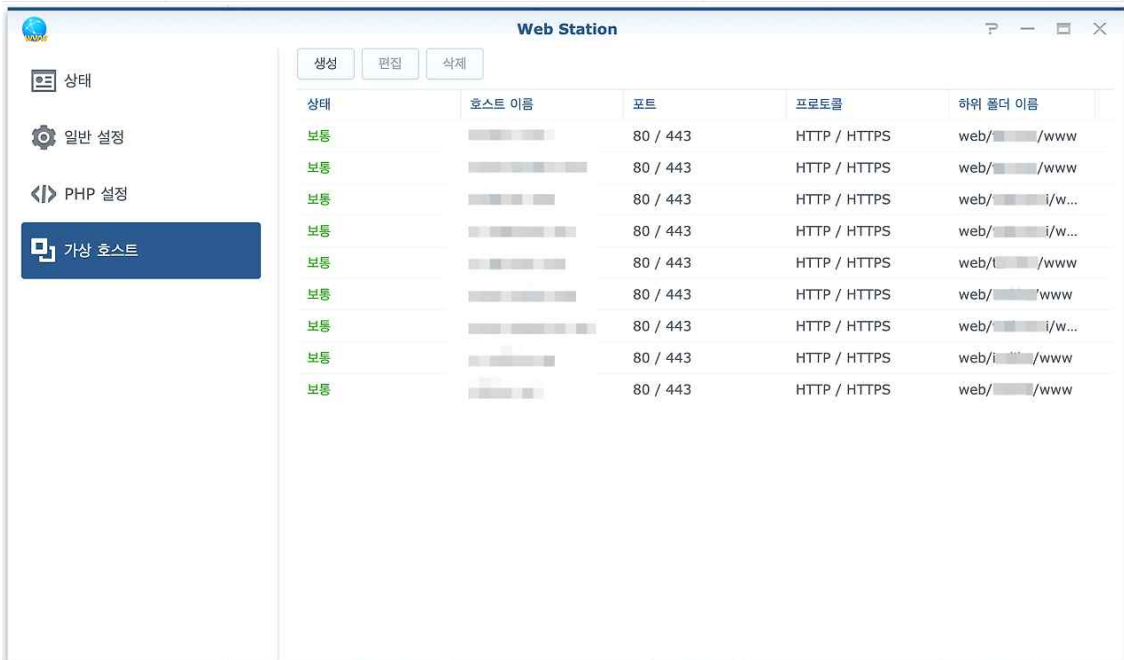
1. 페이지 파일을 Synology NAS 의 web 공유 폴더에 넣는다. File Station , S MB 등을 통해 이 공유 폴더에 액세스할 수 있다.

2. 웹사이트의 백엔드 설정을 구성한다

시놀로지에서는 'Let's Encrypt' 인증서 발급을 해주는데, 가상 호스트를 추가해야



	URL	예
HTTP	http:// NAS의 IP 주소 : 포트 번호	http://xxx.1.1.1:80
	http:// NAS의 도메인 이름 : 포트 번호	http://synorocky.com:80
HTTPS ³	https:// NAS의 IP 주소 : 포트 번호	https://xxx.1.1.1:443
	https:// NAS의 도메인 이름 : 포트 번호	https://synorocky.com:443



'보안' 인증서 구성에서 사이트 추가한게 보여진다. 호스트들은 권한을 부여받아야야 웹서비스에 접속이 가능하다.

4.1 결론 및 기대효과

작품을 만드는 이용자들에게는 작품들을 전시하는 진열장 또는 포트폴리오로 활용하여 작품 활동을 전반적으로 관리할 수 있게 하는 매니지먼트 서비스로써 성장이 기대되고, 작품을 구매하는 이용자 측면에서는 거래 할 때 발생하는 저작권 문제 및 분쟁을 플랫폼을 통해서 원활하게 해결하여 안전한 거래를 할 수 있고, 거래내역을 토대로 자신의 컬렉션을 구성하는 서비스 개발을 기획중이다. UI/UX 개선, 결제 시스템 고도화, 보안 강화, 알림 시스템 도입 등으로 사용자 경험을 높일 수 있으며,

4.2 보완사항

현재 크레딧 충전 및 결제 기능이 준비 중이므로, 외부 결제 API나 서비스(예: 결제 대행 서비스)를 연동하여 사용자의 실제 결제 및 환급 흐름을 완성하는 것이 중요합니다. 결제와 같은 민감한 부분에서 SSL 적용 및 데이터 암호화가 필요하다. 거래 내역, 구매 및 판매 활동에 대한 실시간 알림을 제공하면 사용자 경험이 더 향상될 수 있다. 예를 들어, 채팅 메시지, 구매 완료 알림, 환급 진행 상태 알림 등을 실시간으로 제공할 수 있으면 서비스의 질이 높아질 것으로 예상된다. 또한 NAS와 클라우드 서비스를 결합하여 더 안정적인 데이터 보관 및 관리가 가능하도록 개선할 필요가 있다. 또한, 다양한 결제 수단의 추가와 사용자 피드백 시스템 도입을 통해 서비스의 완성도를 높일 수 있을 것이다.

5. 팀원 소개

- 이승규 팀장
역할: 백엔드, db설계 프론트엔드 개발
- 김경빈
역할: 백엔드, db설계 프론트엔드 개발
- 이규한
역할: 프론트엔드 개발, UI 디자인 더미데이터 생성
- 정윤조
역할: 프론트엔드 개발, UI 디자인 더미데이터 생성
- 오종찬
역할: 프론트엔드 개발, UI 디자인 더미데이터 생성

소스 코드

팀프로젝트 - <https://github.com/sincereperson/education>

6. 발표자료 별첨