

# Secure Coding

유효곤

ugonfor@gmail.com

[외부 유출 금지]

# 강의자 소개

## 유효곤(ugonfor)

---

Researcher, KRAFTON AI (2025.04 예정)

KAIST AI대학원 석사 졸업(2025)

고려대학교 사이버국방학과 졸업(2023)

DEFCON(2020) Final, CODEGATE(2020, 2022) Final 등 CTF 다수 참여

Best of the Best 취약점 분석 9기(2021)

---

관심분야: AI (Efficient AI, Sustainable AI, AI Safety) | Computer Systems (Operation Systems, Reverse Engineering)

딥러닝, 컴퓨터 시스템 분야에 대해서 궁금한 것이 있다면 언제든지 편하게 디스코드 디엠주세요~

# 목차

---

## 1. 시큐어 코딩이란

- 의미
- 최근 IT분야, 시큐어 코딩의 중요성

## 2. 소프트웨어 개발 실습

- 목표
- 요구사항 도출
- 시스템 설계
- 시스템 구현
- 체크리스트 작성 및 테스트
- 유지보수

## 3. 과제

# 시큐어 코딩이란,

소프트웨어 프로그래밍 과정에서 안전한 코드를 만드는 것

: 소프트웨어 개발자가 선제적으로 보안 약점을 제거하는 것

: 정보보호에서 블루 팀\*의 역할

\*블루팀: 방어자(보통 소프트웨어 개발팀), 레드팀: 공격자(보통 화이트해커)

상위 개념: 시큐어 소프트웨어 공학

: 소프트웨어 개발 전 과정에 걸쳐 보안을 고려하여 안전한 소프트웨어를 만드는 것.

# 시큐어 코딩이란,

코딩 과정에서 참고 가능한 레퍼런스:

KISA 가이드라인:

- Python: <https://www.kisa.or.kr/2060204/form?postSeq=13&page=1>
- Java: <https://www.kisa.or.kr/2060204/form?postSeq=14&page=1>

C/C++ 시큐어 코딩 원서:

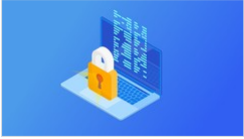





- <https://github.com/DarkCodeOrg/welcome-to-cybersecurity/>

시큐어 코딩 점검 툴:

- Polyspace Bug finder

...

# 시큐어 코딩이란,

	<b>Secure Coding - Secure application development</b> Methodologies and tools to develop <b>secure</b> applications G.L. Golinelli <b>4.0</b> ★★★★★ (2,720) 총 2.5시간 · 41개의 강의 · 중급자	₩25,000
	<b>Principles of Secure Coding</b> Mastering <b>Secure Coding</b> Practices for Robust Applications Chris B Behrens <b>4.4</b> ★★★★★ (6,607) 총 3.5시간 · 55개의 강의 · 중급자 <b>베스트셀러</b>	₩25,000
	<b>Cyber Secure Coder (CSC-110)</b> Certificate Exam Preparatory Course Stone River eLearning <b>4.4</b> ★★★★★ (6) 총 10시간 · 49개의 강의 · 모든 수준	₩25,000
<p>최고의 기업들이 수요가 많고 경력에 도움이 되는 능력 개발을 위해 <b>Udemy Business</b>를 신뢰합니다.</p> 		
	<b>Secure Coding &amp; Design Best Practices in Python</b> <b>Secure Coding</b> Best Practices, <b>Secure Coding</b> Principles, <b>Secure Coding</b> in Python Basics Strong <b>4.3</b> ★★★★★ (293) 총 3시간 · 51개의 강의 · 초급자	₩25,000
	<b>Secure Coding - Ensuring Safe Deployment of Code</b> Understanding the Significance of <b>Secure Coding</b> in DevOps Processes	₩44,000

Google secure coding

전체 이미지 동영상 쇼핑 뉴스 더보기 도구

Tool 가이드 이란 예제 Java PHP 점검 들 교육 솔루션

검색결과 약 406,000,000개 (0.26초)

**요즘IT**  
<https://yoym.wishket.com> · 개발 ·

**시큐어 코딩의 의미와 실천 방안: ①시큐어 코딩이란? | 요즘IT**  
 2022. 12. 12. — 시큐어 코딩은 무엇인가? 시큐어 코딩은 사이버 공격에 대한 방어뿐만 아니라, 개발자의 실수나 코드상의 논리적 오류로 인해 발생할 수 있는 문제점을 ...

**TISTORY**  
<https://codelib.tistory.com> · ... ·

**01 시큐어코딩(secure coding) 이란? - CODELIB - 티스토리**  
 2018. 8. 11. — 1. 시큐어코딩(**secure coding**) 이란? 소프트웨어(SW)를 개발함에 있어 개발자의 실수, 논리적 오류 등으로 인해 SW에 내포될 수 있는.

**KISA 한국인터넷진흥원**  
<https://www.kisa.or.kr> · form ·

**JavaScript 시큐어코딩 가이드**  
 보안취약점 및 침해사고 대응, 인쇄하기 공유하기, 닫기, 트위터, 페이스북, JavaScript 시큐어코딩 가이드. 담당자: 디지털정부보호팀 이수원. 전화: 061-820-1429.

관련 질문 :

- What is the meaning of secure code? ▾
- What is a secure coding technique? ▾
- What are the principles of secure coding? ▾
- Why is secure coding important? ▾

## 시큐어 코딩이란,

<https://github.com/basicsstrong/secure-coding-practices-python>

<https://github.com/OWASP/SecureCodingDojo>

<https://github.com/OWASP/Go-SCP>

...

- 입력값 검증, 보안기능 활성화, 시간 및 상태처리, 에러 처리, 코드 오류 처리, 캡슐화, API 오용방지 등
- 개발자들이 이런 원칙을 모를 까? No.
- 문제는 한정된 시간.
- 개발을 할 때는 기능 구현에 집중하다 보니, 보안 기능에 대해서 미처 생각하지 못한 경우가 많음.

## 시큐어 코딩이란,

---

시큐어 코딩은 주어진 예제에 적용하는 것은 쉽지만,

실제로 개발을 하면서 시큐어 코딩을 신경 쓰는 것은 굉장히 어렵습니다.

→ 여러분이 길러야 하는 능력은, 개발을 하면서도 무의식적으로 안전한 소프트웨어를 만드는 능력



# 최근 IT분야, 시큐어 코딩의 중요성

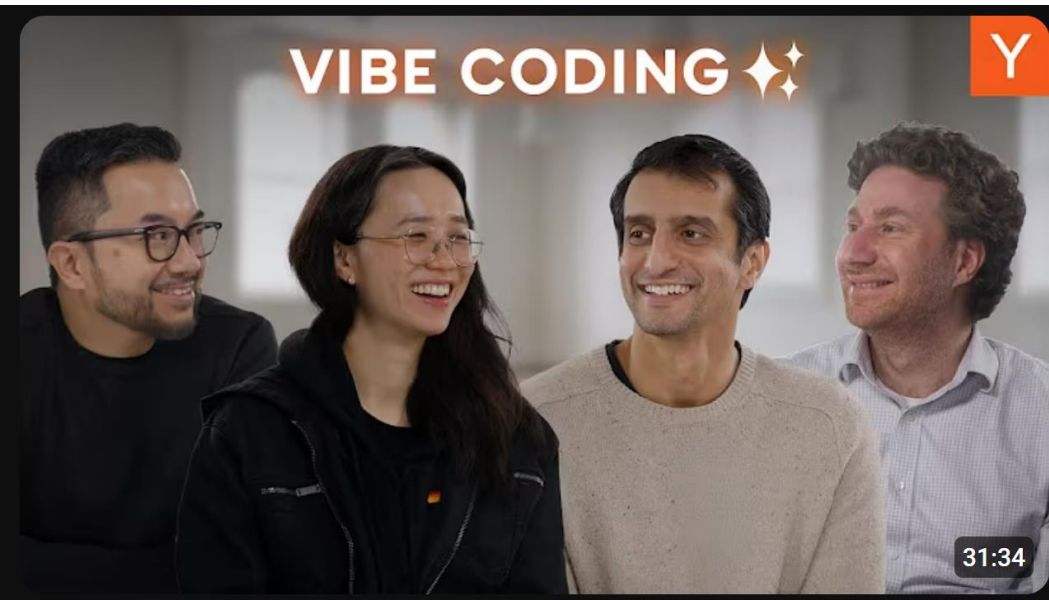
최근 AI의 발달로, 코드의 상당부분을 AI가 작성해주고 있음.

- Cursor, Copilot, ChatGPT, Claude, ...

다양한 고성능의 Coding이 가능한 AI들이 공개되었고, 활용되고 있음.

# 최근 IT분야, 시큐어 코딩의 중요성

Y Combinator(미국 최대 규모의 스타트업 액셀러레이터)에서도 “Vibe Coding이 미래다” 언급



**VIBE CODING ✨**

**Y** Vibe Coding Is The Future  
조회수 18만회 · 2주 전

**Y** Y Combinator

Andrej Karpathy recently coined the term “vibe coding” to describe how LLMs are getting so good that devs can simply “give in to ...

챕터 11 Intro | What is vibe coding? | What founders in the current YC batch are saying | Debugging and building...

31:34

## 최근 IT분야, 시큐어 코딩의 중요성

AI 활용하여 코딩을 하는 경우, 특히 보안 관련하여 구멍이 생길 수 있음.

MVP를 만드는 데 치중하며, 일반적으로 요구사항을 자연어로 설명할 때 보안에 관한 고려가 전혀 없음.

→ 궁극적으로 보안의 경우, 사람이 직접 고려하는 분야로 남아 있을 것.

## 최근 IT분야, 시큐어 코딩의 중요성

AI 활용하여 코딩을 하는 경우, 특히 보안 관련하여 구멍이 생길 수 있음.

MVP를 만드는 데 치중하며, 일반적으로 요구사항을 자연어로 설명할 때 보안에 관한 고려가 전혀 없음.

→ 궁극적으로 보안의 경우, 사람이 직접 고려하는 분야로 남아 있을 것.

## 소프트웨어 개발 실습

실제로 소프트웨어를 개발해보며, 시큐어 코딩을 적용해보자.

# 소프트웨어 개발 실습

## 환경 세팅

- 리눅스 환경 세팅 필요 (WSL / VMWare / VirtualBox 등을 통하여 Ubuntu 설치)
- Github 계정 생성 및 git 설치
- ngrok 설치 (<https://ngrok.com/downloads/linux?tab=snap>)
- miniconda 설치 (<https://www.anaconda.com/docs/getting-started/miniconda/install>)
- <https://github.com/ugonfor/secure-coding> repository git clone 후 README를 따라 환경 세팅

## 소프트웨어 개발 실습

---

실제로 소프트웨어를 개발해보며, 시큐어 코딩을 적용해보자.

- 시큐어 코딩을 적용하기가 얼마나 어려운 지
- 어떤 것을 간과할 수 있는 지

# 소프트웨어 개발 실습

---

간단한 중고거래 플랫폼 구현

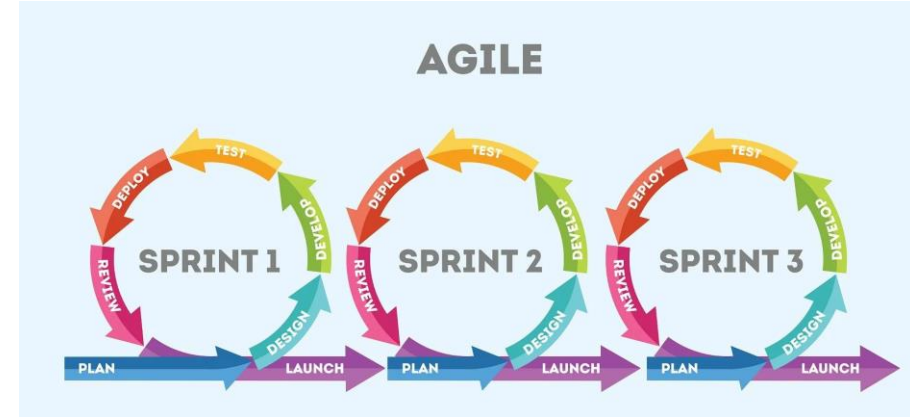
Tiny Second-hand Shopping Platform.



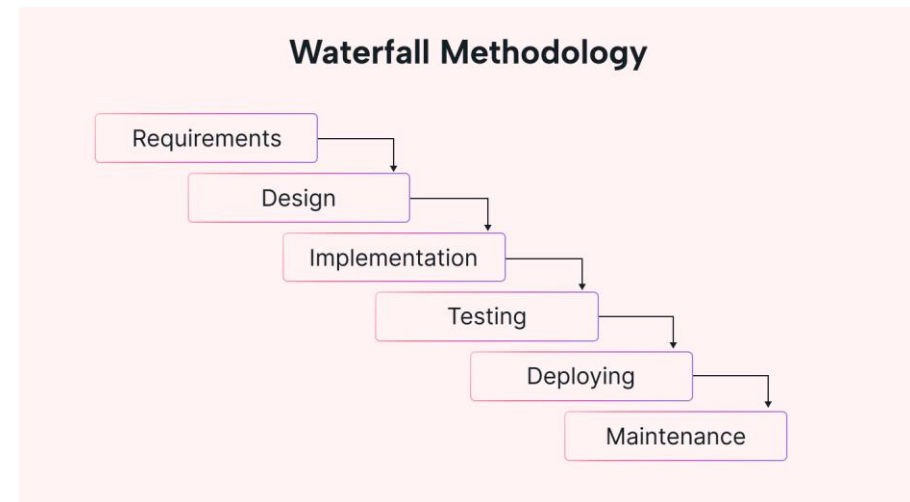
# 소프트웨어 개발 실습

## 소프트웨어 개발 주기

1. 요구사항 도출
2. 시스템 설계
3. 시스템 구현
4. 테스트
5. 유지 보수



Agile 방법론



Waterfall 방법론

# 소프트웨어 개발 실습

## 소프트웨어 개발 주기

1. **요구사항 도출** : 목표를 달성하기 위해, 필요한 요소가 무엇이 있는 지 도출하는 과정
2. **시스템 설계** : 도출된 요구사항을 기반으로, 시스템을 구체적으로 설계하는 과정
3. **시스템 구현** : 실제로 코딩을 하는 과정
4. **테스팅** : 구현된 소프트웨어가 요구사항을 제대로 충족하는 지 검증하는 과정
5. **유지 보수** : 실제로 소프트웨어를 배포하고, 지속적으로 유지 보수 하는 것

## 목표

---

“중고거래가 가능한 플랫폼”

## 요구사항 도출

### 중고거래 플랫폼에 있어야 하는 기능:

- 사람들이 플랫폼에 가입할 수 있어야 함
- 상품들을 올리고 볼 수 있어야 함.
- 플랫폼 사용자들끼리 소통이 가능해야함.
- 악성 유저나 상품을 차단 해야 함.
- 유저들 간의 송금이 가능해야함
- 상품의 검색할 수 있어야 함.
- 관리자가 플랫폼의 모든 요소를 관리할 수 있어야 함.
- ...

### 필요한 비기능적 요소:

- 보안
- 디자인
- 서버 속도
- ...

## 요구사항 도출

### 중고거래 플랫폼에 있어야 하는 기능:

- 사람들이 플랫폼에 가입할 수 있어야 함
- 상품들을 올리고 볼 수 있어야 함.
- 플랫폼 사용자들끼리 소통이 가능해야함.
- 악성 유저나 상품을 차단 해야 함.
- 유저들 간의 송금이 가능해야함
- 상품의 검색할 수 있어야 함.
- 관리자가 플랫폼의 모든 요소를 관리할 수 있어야 함.
- ...

예시 코드에서 다루는 부분

### 필요한 비기능적 요소:

- 보안
- 디자인
- 서버 속도
- ...

## 시스템 설계

요구사항을 실제로 어떻게 구현할 것인지 구체화 하고 그에 따라 시스템을 디자인 하는 것

- 도식으로 그리는 것이 일반적이나, 글로 작성해도 무방

# 시스템 설계

- 사람들이 플랫폼에 가입할 수 있어야 함
  - 로그인 페이지가 있어야함
  - 회원가입 페이지가 필요함
  - 사용자 프로필을 확인할 수 있음 좋겠음.
  - 마이페이지 기능이 필요함 (소개글 및 비밀번호 업데이트)
  - 아이디는 중복되지 않아야 함.
  - 유저 정보는 데이터베이스로 관리해야함.
- 상품들을 올리고 볼 수 있어야 함.
  - 상품 등록이 가능한 페이지를 만들어야 함.
  - 내가 등록한 상품들을 확인 및 관리가 가능해야 함.
  - 각 상품은 상품명, 가격, 사진을 보여주어야 함.
  - 등록된 상품은 누구나 볼 수 있어야 함.
  - 상품 정보는 데이터베이스로 관리해야함.
  - 상품을 볼 때에는 이름만 보여주고, 각 상품을 클릭하면 자세한 페이지를 볼 수 있도록!
- 플랫폼 사용자들끼리 소통이 가능해야함.
  - 사람들끼리 의사소통하기 위해 채팅을 할 수 있어야 함.
  - 전체 유저가 소통할 수 있는 채팅이 있어야 함.
  - 각각의 유저가 1대1로 소통할 수 있는 채팅이 있어야함.
- 악성 유저나 상품을 차단 해야 함
  - 불량 상품 혹은 사용자를 신고할 수 있어야 함.
  - 신고 사유를 신고시 작성하게 만들자.
  - 일정 횟수 이상 신고된 상품은 차단됨.
  - 일정 횟수 이상 신고된 유저는 휴면계정 전환

# 시스템 설계

- 유저 관리
  - 회원 가입 기능
  - 로그인 기능
  - 사용자 조회 기능
  - 마이페이지 기능 (소개글 및 비밀번호 업데이트)
- 상품 관리
  - 상품 등록 기능
  - 등록된 상품 관리 기능
  - 상품 조회 기능
  - 상품 상세 페이지 기능
- 유저 소통 기능
  - 실시간 전체 채팅 기능
  - 1대1 채팅 기능
- 악성 유저 필터링
  - 불량 유저 및 상품 신고 기능
  - 불량 상품 삭제 기능
  - 불량 유저 휴면 기능



# 시스템 설계

- 웹페이지 설계:
  - 기본 페이지
  - 회원 가입 페이지
  - 로그인 페이지
  - 전체 상품 조회 및 채팅 페이지
  - 새 상품 등록 페이지
  - 상품 상세 조회 페이지
  - 신고 페이지

## 시스템 설계

---

- 데이터베이스 설계:
  - 사용자 정보(사용자 아이디, 사용자 계정명, 비밀번호, 소개글)
  - 상품 정보 (상품 아이디, 상품명, 상품설명, 가격, 판매자 아이디)
  - 신고 정보 (신고 아이디, 신고자 아이디, 타겟 아이디, 신고 사유)

# 시스템 구현

## 코드 확인!

- 유저 관리
  - 회원 가입 기능
  - 로그인 기능
  - 사용자 조회 기능
  - 마이페이지 기능 (소개글 및 비밀번호 업데이트)
- 상품 관리
  - 상품 등록 기능
  - 등록된 상품 관리 기능
  - 상품 조회 기능
  - 상품 상세 페이지 기능
- 유저 소통 기능
  - 실시간 전체 채팅 기능
  - 1대1 채팅 기능
- 악성 유저 필터링
  - 불량 유저 및 상품 신고 기능
  - 불량 상품 삭제 기능
  - 불량 유저 휴면 기능

# 테스트

## 1. 체크리스트 작성 및 확인

- 요구사항을 모두 만족하는가?
- 기능은 정상적으로 작동하는 가?
- 보안 요소들이 제대로 구현되어 있는가?

→ 구현된 각 부분에 대해서, 위 사항들이 제대로 만족하는 지 점검하는 체크리스트를 작성

→ 이후, 실제로 코드를 확인해보며 만족하는 지 점검

## 1. 체크리스트 작성 및 확인

### - 체크리스트 예시(보안 요소 들이 제대로 구현되었는가?)

1	Section	Checklist Item	Description
2	회원가입 및 프로필 관리	서버측 입력 검증	사용자명(username)과 비밀번호(password)에 대해 길이, 허용 문자 집합, 형식 등 서버측 검증 수행. XSS 공격 방지를 위해 입력값 필터링 및 인코딩 적용 여부 확인
3		CSRF 보호	회원가입, 로그인, 프로필 수정 등 모든 폼에 대해 CSRF 토큰 사용 여부를 확인하여 요청 위조 공격 방지
4		비밀번호 보안	비밀번호를 평문으로 저장하지 않고 bcrypt, Argon2 등 강력한 해시 알고리즘과 고유 salt를 적용하여 암호화 저장하는지 확인
5		세션 쿠키 설정	세션 쿠키에 HttpOnly 및 HTTPS 환경에서 Secure 플래그가 적용되어 있는지 확인
6		세션 만료 및 재인증	일정 시간 이후 세션 만료 및 민감 작업 시 재인증 로직이 구현되어 있는지 확인
7		실패 로그인 방어	로그인 실패 횟수에 따른 계정 잠금 혹은 지연(time-out) 메커니즘 적용 여부 확인
8		오류 메시지	오류 발생 시 내부 정보(스택 트레이스, DB 정보 등)가 노출되지 않도록 처리되어 있는지 확인
9		상품 등록 및 관리	폼 입력 검증
10		XSS 방어	사용자 입력(상품 설명 등)에 대해 HTML 태그 및 스크립트 코드 이스케이프 또는 필터링 적용 여부 확인
11		인증된 사용자만 등록	상품 등록, 수정, 삭제 기능이 로그인한 사용자에게만 허용되도록 접근 제어가 구현되어 있는지 확인
12		소유자 확인	상품 수정 및 삭제 시, 요청한 사용자가 해당 상품의 소유자인지 검증하는 로직이 구현되어 있는지 확인
13		데이터 무결성	데이터베이스에 저장되기 전 모든 필수 항목 및 형식이 올바른지 검증하는 로직이 있는지 확인
14	실시간 채팅 및 메시징	메시지 내용 검증	채팅 메시지에 대해 길이 제한, 허용 문자 집합, XSS 이스케이프 처리 여부 등 서버측 검증 수행 여부 확인
15		사용자 인증	Socket 연결 시 사용자가 인증된 상태인지 확인하는 로직(예: 로그인 상태 확인)이 적용되어 있는지 확인
16		메시지 검증	클라이언트에서 수신한 메시지 데이터의 형식 및 내용에 대해 서버측 검증 로직이 존재하는지 확인
17		Rate Limiting	동일 사용자가 단기간에 과도한 메시지를 보내지 않도록 제한하는 기능(스팸 방지)이 구현되어 있는지 확인
18		연결 암호화	운영 환경에서 WSS(SSL/TLS 암호화된 웹소켓)를 사용하여 데이터 전송의 기밀성이 보장되는지 확인
19	안전 거래 및 신고	폼 입력 검증	신고 대상(target_id) 및 신고 사유(reason)에 대해 서버측 입력 검증, 길이 제한, XSS 방어 적용 여부 확인
20		인증된 사용자 접근	신고 기능은 반드시 로그인한 사용자만 접근 가능하도록 제어되어 있는지 확인
21		데이터 무결성 및 로그 관리	신고 접수 시 올바른 형식의 데이터가 저장되고, 신고 활동이 감사 로그로 기록되는지 확인
22		신고 남용 방지	동일 사용자의 반복 신고 제한, 신고 건수 제한 및 관리자 검토 프로세스 등 신고 기능 남용 방지 로직이 구현되어 있는지 확인

# 테스트

---

## 2. 실제 테스트

- 테스트 케이스들을 생성 및 입력
- 각 테스트 케이스에 대해서 “정상적으로 작동”하는 지 확인

## 유지보수

---

실제로 사용해보며 정상작동 하더라도, “불편한 부분”은 없는 지 확인

요구사항이 잘못된 것은 없는지? 각 단계에서 잘못된 것은 없는지? 등을 확인.

→ 이후 적절한 단계로 돌아가서 다시 수행.

# 마무리 - 과제

## Tiny Second-hand Shopping Platform

- 플랫폼의 요구사항을 만족하도록 개발 (뒷 페이지 확인)
- 플랫폼 개발 전 과정(요구사항 분석, 시스템 설계, 구현, 체크리스트 작성 및 테스트, 유지보수)에 대한 보고서 작성(양식은 자유)
- 개발 과정에서 확인한 보안 약점들이 무엇이고, 어떻게 변경하였는지 작성
- 완성한 프로그램은 github에 public으로 올리고, README.md에 환경 설정 및 실행방법 명시
- 다음 페이지에서 언급한 최소 요구사항을 모두 만족한다면, 이외 어떤 기능이든 추가 가능하며, 다른 모든 것이 자유.
- 단, 최대한 보안 약점이 없도록 할 것.



# 마무리 - 과제

## 플랫폼 요구사항:

### 중고거래 플랫폼에 있어야 하는 기능:

- 사람들이 플랫폼에 가입할 수 있어야 함
- 상품들을 올리고 볼 수 있어야 함.
- 플랫폼 사용자들끼리 소통이 가능해야함.
- 악성 유저나 상품을 차단 해야 함.
- 유저들 간의 송금이 가능해야함
- 상품의 검색할 수 있어야 함.
- 관리자가 플랫폼의 모든 요소를 관리할 수 있어야 함.

해당 부분에 대한  
시스템 설계는, 24page에서  
명세한 것을 모두 포함할 것  
(필요에 따라 기능추가 할 것)

해당 부분은  
시스템 분석부터 진행할 것

논리적으로  
타당하다면,  
요구사항 변경 가능  
(요구사항 추가는  
어떤 것이든 가능)

## 마무리 - 과제

---

제출 기한: 2025년 4월 25일(금) 23:59:59

제출 내용: 개발 전 과정에 대한 설명이 포함된 보고서(pdf파일, github repository 링크 포함하여 제출)

제출 형식: [WHS][secure-coding][XX반]이름(전화번호뒷자리).pdf (공백 전혀 없음)

- 예시: [WHS][secure-coding][01반]홍길동(1234).pdf

제출처: LMS

질문: Discord DM(유효곤)

## 마무리 - 프로젝트

화이트햇 2단계에서는 프로젝트를 진행하게 됩니다.

주제는 멘토가 제안하는 형식이지만, 저와 함께 수행하고 싶은 주제가 있다면 아래 설문을 작성해주세요.

주제 제안 시 참고할 예정입니다.

설문 링크는 다음 링크에서 확인 가능합니다 : <https://ugonfor.kr/lecture/whs3>

아래 내용에 대해서는 깊이 있는 조언이 가능합니다.

- AI, 리버스 엔지니어링, 네트워크, 컴퓨터 시스템

아래 내용에 대해서는 다른 더 우수한 멘토님들이 더 잘 조언해줄 것 같습니다.

- 정책, 컨설팅, IoT, 포렌식, 클라우드, ...