

Safe Lookup

안전한 다운로드 프로그램

김주원
천윤수
백승규
이제민

INDEX

01

개요

목표, 기대효과

02

범위

핵심 기능

03

역할

세분화

04

내용

설계, 과정

05

일정

전반적인 계획

01 개요



목표

웹 사이트에 대한 안전성
검사와 해당 사이트에서 받는
파일 및 패킷에 취약점을
검사하는 프로그램 제작



기대 효과

- 웹 어플리케이션 보안 취약점과 대응 방안에 대한 학습
- 프로그램 구조에 대한 이해
- 프로그램 개발 역량 강화
- 팀 단위 개발 계획의 과정과 역할에 대한 이해

02 범위



추진 배경

- 피싱 사이트/ 악성 URL 증가
- 정교해진 악성 파일 배포
- 개인 정보 유출에 대한 불안
- 낮아진 사이버 공격의 진입장벽
- 보안 프로그램 개발에 대한 경험 필요

02 범위



핵심 기능

- 웹 페이지 보안 테스트
- 시스템 및 파일에 대한 로그 분석과 감사
- 파일에 대한 데이터 손실 방지 테스트
- 웹 페이지 및 파일 악성 프로그램 감지
- 보안 폴더를 활용한 파일 안정성 검사
- 편리한 GUI 제공

03 역할



김주원

PM & 디자인



천윤수

UI & 문서작성



백승규

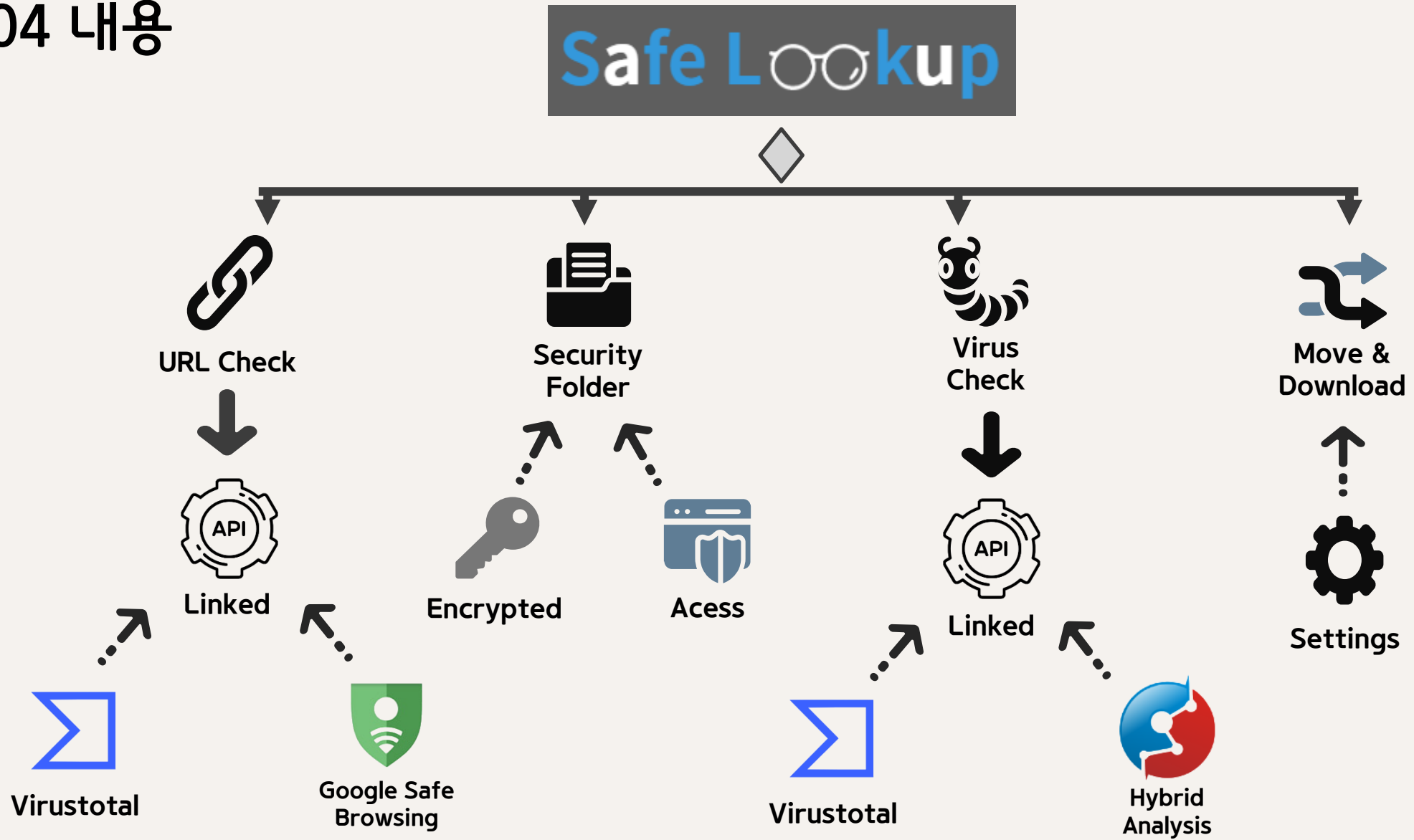
URL Check



이제민

Virus Check

04 내용



04 내용

URL Check

- URL 입력
- API 연동 및 안정성 검사 설정
- 결과 값 반환 및 분석
- 알림 기능 구현
- 재검사 및 차단기 기능 구현
- UI 구축

Security Folder

- 운영체제에서 권한 설정
- 프로그램에서 권한 설정
- 암호화 및 인증화 방법
- 파일 보호 여부
- 권한 및 파일 내부 확인
- 알림 기능 구현

Virus Check

- 보안폴더 파일 읽기 설정
- 소스코드 저장 경로 설정
- 라이브러리 설정
- API 연동 설정
- 결과값 반환 및 분석
- 재검사 및 차단기 기능 구현
- 알림 기능 구현
- 삭제/이동 경로 설정
- UI 구축

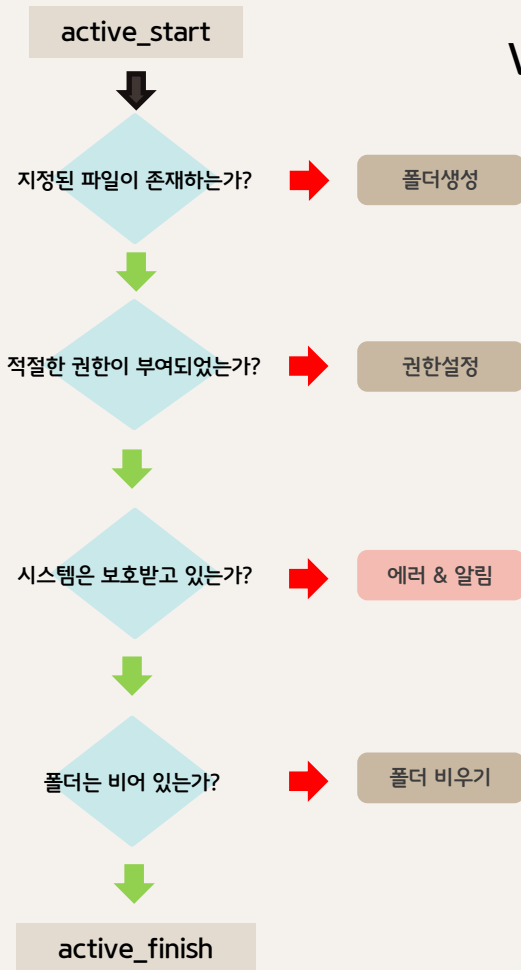
Move & Download

- 이동 권한 및 경로 설정
- 삭제 권한 및 경로 설정
- 이동 및 삭제 상황 설정
- 이동 및 삭제 알림 기능 구현
- 다운로드시 권한 및 경로 설정
- 다운로드시 알림 기능 구현

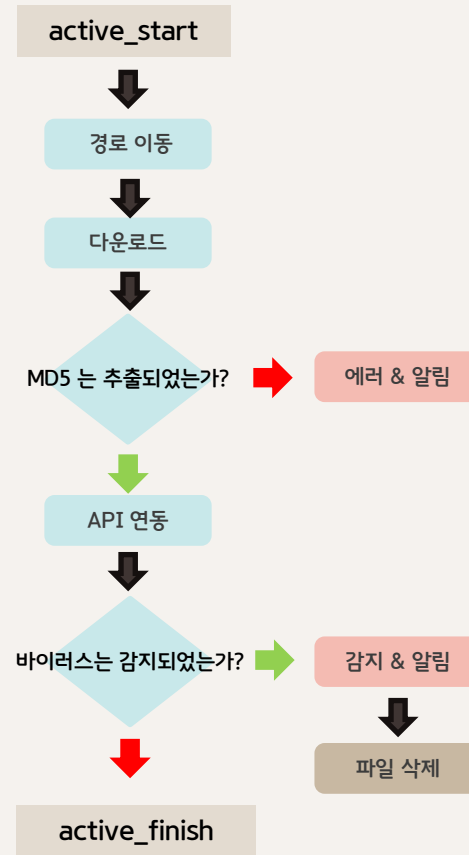
04 내용

Flow Chart

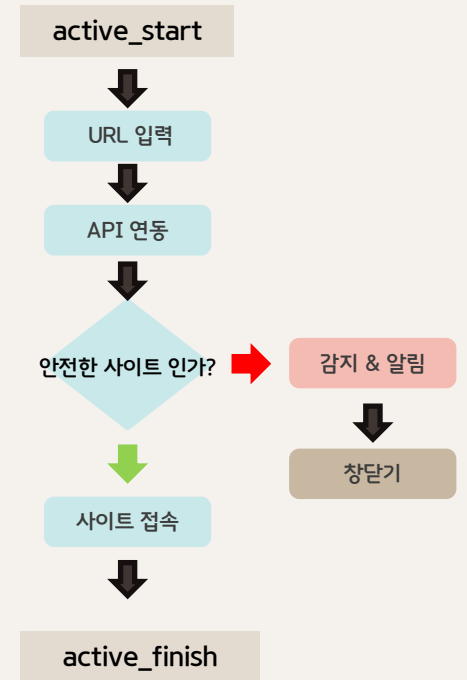
Security Folder



Virus Check



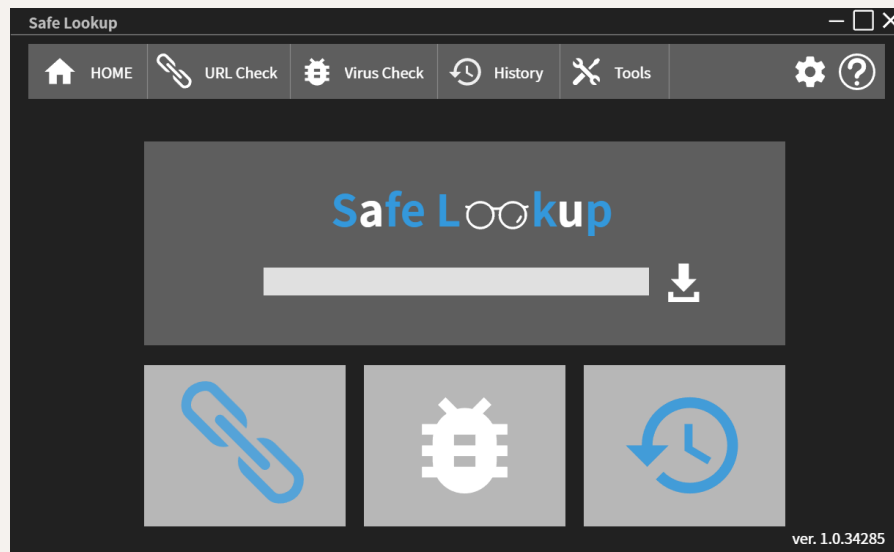
URL Check



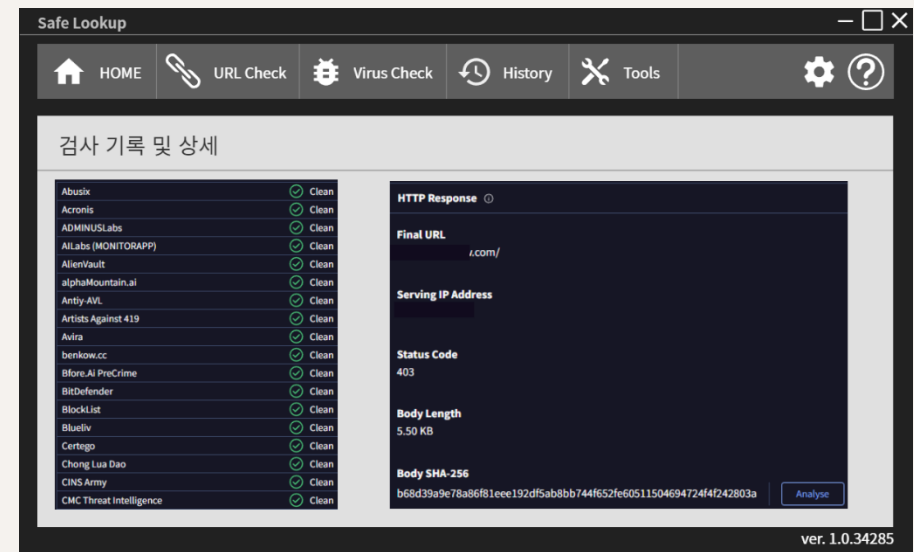
04 내용

초기 디자인

메인화면 UI



URL 검사 결과 UI



04 내용

Virus Check

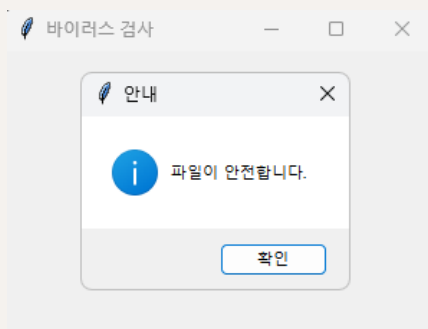
진행 과정

구현 내용

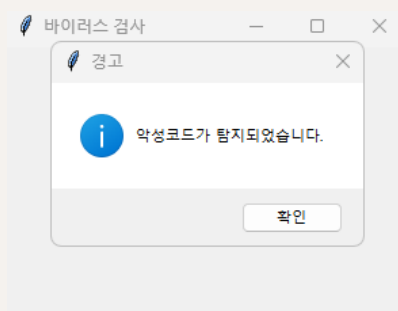
- OS, hashlib 모듈 & Virustotal API 사용
- 파일 무결성 검사 진행

정상 파일 --- 0 의심파일 --- 1
리턴값 == 1 → 파일 삭제

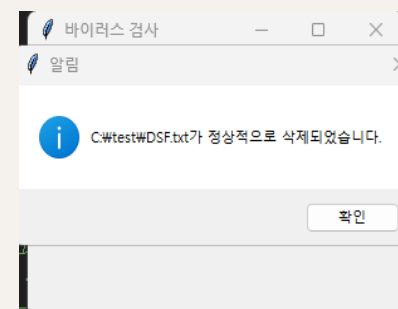
정상 파일 검사 결과



의심 파일 검사 결과



의심 파일 삭제 결과



03 내용

Virus Check

진행 과정

오류 수정

- 프로그램 실행 후 디렉터리에 들어온 파일을 인식 불가

```
# 디렉토리 내의 파일을 리스트업
files = [os.path.join(directory_path, file) for file in os.listdir(directory_path) if os.path.isfile(os.path.join(directory_path, file))]
download_times = {file: os.path.getctime(file) for file in files} # 파일 다운로드 시간 기록
files.sort(key=lambda x: download_times[x], reverse=False) # 다운로드 시간 기준으로 파일 정렬
```

파일을 검사하는 함수 내에 파일을 리스트업 해주는 코드를 추가해 오류 수정

03 내용

Virus Check

진행 과정

기능 보완

- 보고서 출력 기능 추가
- 파일 선택 검사 기능 추가

- 보고서 **텍스트 위젯** 변경을 통해 가독성 높임
- 직접 검사할 파일 선택 가능
- **GUI** 수정

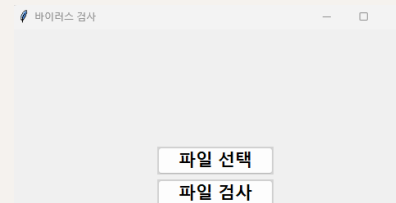
보고서 출력



텍스트 위젯 기능



파일 선택 검사 기능



03 내용

URL Check 진행 과정

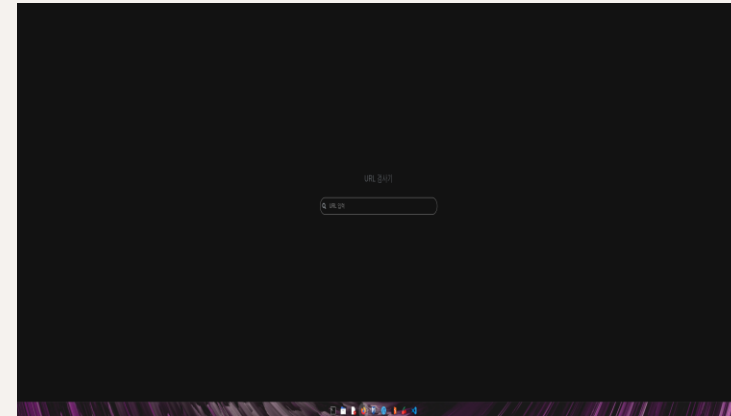
구현 내용

- API 연동 및 URL 스캔
- URL Check UI 설계

API 연동 및 URL 스캔 코드

```
1 // URL 체크 API
2 // URL 체크 API
3 // URL 체크 API
4 // URL 체크 API
5 // URL 체크 API
6 // URL 체크 API
7 // URL 체크 API
8 // URL 체크 API
9 // URL 체크 API
10 // URL 체크 API
11 // URL 체크 API
12 // URL 체크 API
13 // URL 체크 API
14 // URL 체크 API
15 // URL 체크 API
16 // URL 체크 API
17 // URL 체크 API
18 // URL 체크 API
19 // URL 체크 API
20 // URL 체크 API
21 // URL 체크 API
22 // URL 체크 API
23 // URL 체크 API
24 // URL 체크 API
25 // URL 체크 API
26 // URL 체크 API
27 // URL 체크 API
28 // URL 체크 API
29 // URL 체크 API
30 // URL 체크 API
31 // URL 체크 API
32 // URL 체크 API
33 // URL 체크 API
34 // URL 체크 API
35 // URL 체크 API
36 // URL 체크 API
37 // URL 체크 API
38 // URL 체크 API
39 // URL 체크 API
40 // URL 체크 API
41 // URL 체크 API
42 // URL 체크 API
43 // URL 체크 API
44 // URL 체크 API
45 // URL 체크 API
46 // URL 체크 API
47 // URL 체크 API
48 // URL 체크 API
49 // URL 체크 API
50 // URL 체크 API
51 // URL 체크 API
52 // URL 체크 API
53 // URL 체크 API
54 // URL 체크 API
55 // URL 체크 API
56 // URL 체크 API
57 // URL 체크 API
58 // URL 체크 API
59 // URL 체크 API
60 // URL 체크 API
61 // URL 체크 API
62 // URL 체크 API
63 // URL 체크 API
64 // URL 체크 API
65 // URL 체크 API
66 // URL 체크 API
67 // URL 체크 API
68 // URL 체크 API
69 // URL 체크 API
70 // URL 체크 API
71 // URL 체크 API
72 // URL 체크 API
73 // URL 체크 API
74 // URL 체크 API
75 // URL 체크 API
76 // URL 체크 API
77 // URL 체크 API
78 // URL 체크 API
79 // URL 체크 API
80 // URL 체크 API
81 // URL 체크 API
82 // URL 체크 API
83 // URL 체크 API
84 // URL 체크 API
85 // URL 체크 API
86 // URL 체크 API
87 // URL 체크 API
88 // URL 체크 API
89 // URL 체크 API
90 // URL 체크 API
91 // URL 체크 API
92 // URL 체크 API
93 // URL 체크 API
94 // URL 체크 API
95 // URL 체크 API
96 // URL 체크 API
97 // URL 체크 API
98 // URL 체크 API
99 // URL 체크 API
100 // URL 체크 API
```

URL Check UI



03 내용

URL Check

진행 과정

기능 보완

- 검사 결과 및 상세 정보 출력
- 통계 데이터 차트로 표시

기능 보완 구현 코드

[illegible]

03 내용

개발 이슈

보안 폴더 파트

- 이슈 분야: 암호화
- 이슈 내용: Windows 내부 생성 보안 폴더 암호화
- 이슈 세부:
 - Windows OS → 암호화 기능 제공 X
(낮은 버전은 제공)
 - 개발 언어 & 도구 → 암호화 기능 추가 X

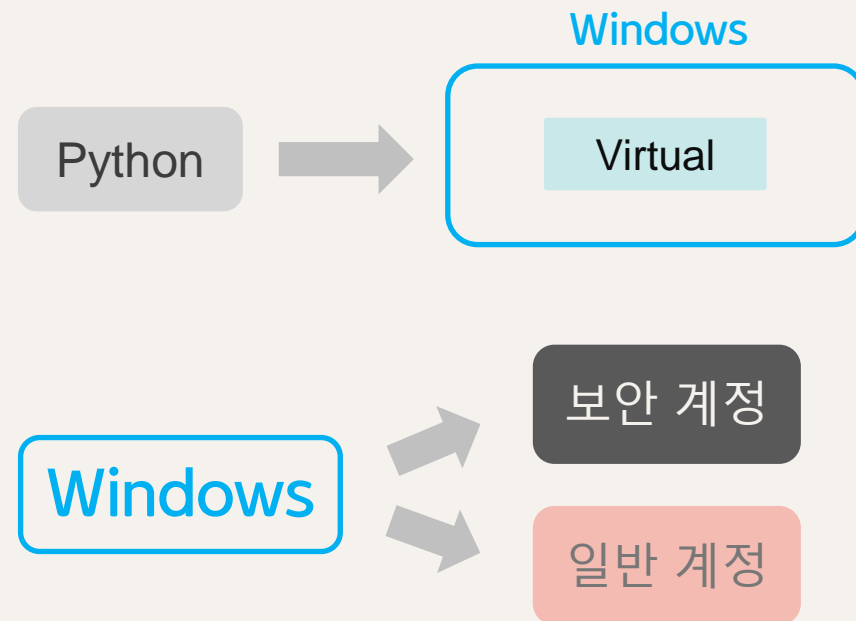


03 내용

개발 이슈

대안 탐색

- 대안 1 : 가상환경 내부에 포안폴더 생성
한계: - 개발 언어&도구 → 다른 OS 구현 어려움
- 무거워지는 프로그램
- 대안 2 : Windows 내부에 보안폴더를 위한 계정 생성
한계: - 불필요한 계정 생성
- 부팅시 계정 선택 화면 생성
- 프로그램 실행시 계정 전환의 문제



최종 결정

- 다운로드 파일 검사 시점 변경: 다운로드 시 → 다운로드 완료된 파일
- 역할 변경:

PA

천윤수

보안 폴더 → UI & 문서작성

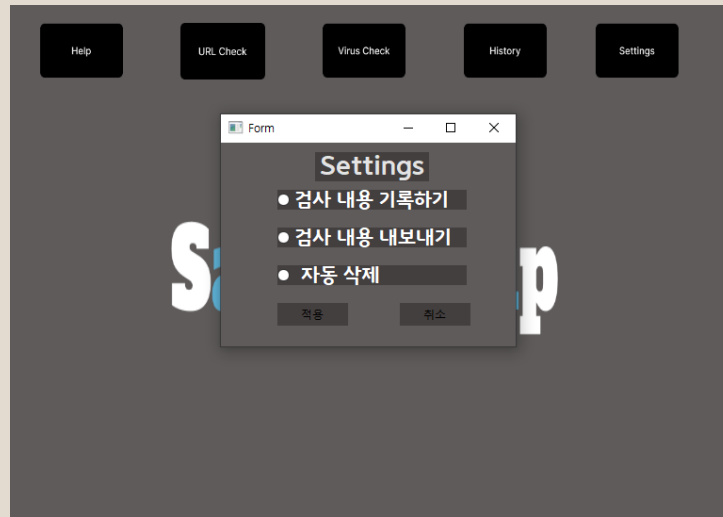
03 내용

UI / UX

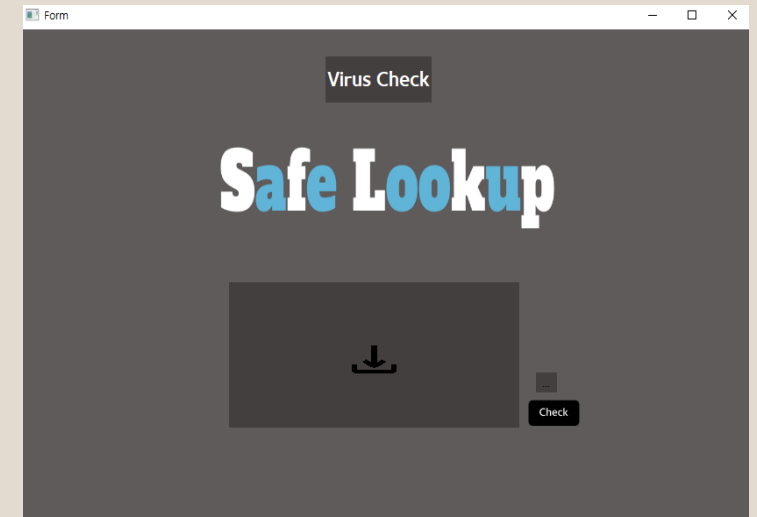
메인 화면



설정 화면



기능 화면



03 내용

UI / UX

[illegible]

```
# help widget func
def button1Function(self) :
    self.help_widget = QWidget()
    self.ui_help = Ui_help_widget()
    self.ui_help.setupUi_help(self.help_widget)
    self.help_widget.show()

# virus widget func
def button2Function(self) :
    self.virus_widget = QWidget()
    self.ui_virus = Ui_virus_widget()
    self.ui_virus.setupUi_virus(self.virus_widget)
    self.virus_widget.show()

# history widget func
def button3Function(self) :
    self.history_widget = QWidget()
    self.ui_history = Ui_history_widget()
    self.ui_history.setupUi_history(self.history_widget)
    self.history_widget.show()

# setting widget func
def button4Function(self) :
    self.settings = QWidget()
    self.ui_settings = Ui_settings()
    self.ui_settings.setupUi_settings(self.settings)
    self.settings.show()

# url widget func
def button5Function(self) :
    self.url_widget = QWidget()
    self.ui_url = Ui_url_check_widget()
    self.ui_url.setupUi_url(self.url_widget)
    self.url_widget.show()
```

03 내용

통합 이슈

전체 파트

· 이슈 분야: 기능 구현

· 이슈 내용:

개별 프로그램 통합 및 연동 이슈

- 서로 다른 프로그램 설계
- 통합 과정에서 예상하지 못한 오류



지연

소통

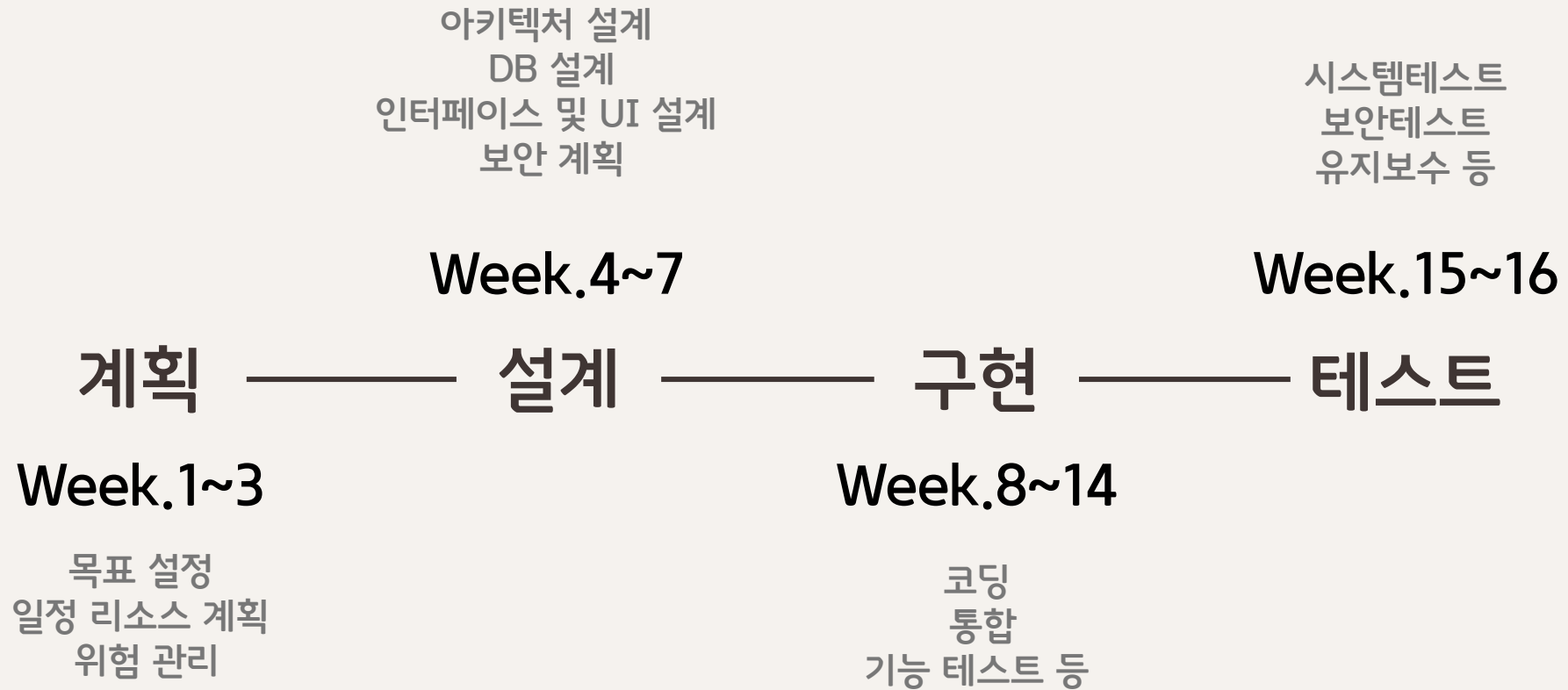
개발

자기 학습



경험

05 일정



05 일정

의사소통 방법

구분	구분	내용	시기	대상
보고	착수보고	<ul style="list-style-type: none"> · 프로젝트 개요 및 수행계획 수립 · 추진 기능 및 WBS 협의 	3월 말	전원
	중간보고	<ul style="list-style-type: none"> · 계획 대비 진척도 점검 · 주요 이슈 및 해결 방안 공유 · 기능 연동 방법 및 향후 진행 보고 	5월 초	전원
	완료보고	<ul style="list-style-type: none"> · 산출물 테스트 보고 · 프로젝트 수행 결과 보고 	6월 중	전원
정기회의	주간회의	<ul style="list-style-type: none"> · 금주 작업 및 차주 계획 보고 · 이슈 논의 및 해결 	매주	전원
비정기회의	이슈회의	<ul style="list-style-type: none"> · 긴급 이슈상황 보고 · 해결 방안 논의 	수시	PM/담당파트

05 일정

WBS

단계	Mar W.1-2	Mar W.3-4	Apr W.1-2	Apr W.3-4	May W.1-2	May W.3-4	Jun W.1-2	Jun W.3-4
기획	100%							
분석 및 설계			100%					
웹 페이지 검사								90%
다운로드 검사								90%
보안폴더				40%				
UI/UX								80%
테스트								0%

05 일정

WBS

작성자	전윤수	작성일	2023년 03월 18일						
시작일	2023년 03월 06일	최종 수정일	2023년 03월 24일						
종료일	2023년 07월 21일	버전	WBS v1.0						
단계 구분	주요 업무	세부 업무	작업자	상태	진척율	시작일	종료일	작업기간	M W D
1. 기획	1.1 계획 수립								
		1.1.1 보안 이슈 검토 및 목표 선정	김주원/전윤수/이재민/백승규	완료	100%	2023-03-06	2023-03-11	15	
		1.1.2 정보 수집 및 개요서 작성	김주원/전윤수/이재민/백승규	완료	100%	2023-03-10	2023-03-15	28	
2. 분석 및 설계	2.1 수행 계획서 작성								
		2.1.1 정보 수집 및 범위 선정	김주원/전윤수/이재민/백승규	완료	100%	2023-03-16	2023-03-18	12	
		2.1.2 아키텍처/UI 구성	김주원/전윤수/이재민/백승규	완료	100%	2023-03-17	2023-03-20	8	
		2.1.2 블루우 차트 설계	김주원/전윤수/이재민/백승규	완료	100%	2023-03-18	2023-03-22	20	
3. 구현	3.1 웹 페이지 검사 개발								
		3.1.1 API 연동	김주원/백승규	완료	100%	2023-03-25	2023-05-20	30	
		3.1.2 작성코드 검사 조건부 구현	김주원/백승규	진행중	90%	2023-03-25		23	
		3.1.3 알림 및 검사 기록 출력 구현	김주원/백승규	진행중	90%	2023-04-01		28	
		3.1.3 다운로드 파일 이름 및 삭제 구현	김주원/백승규	중단	0%	*****	*****	***	
	3.2 다운로드 파일 검사 개발								
		3.2.1 다운로드 파일 검사 구현	김주원/이재민	중단	0%	*****	*****	***	
		3.2.2 API 연동	김주원/이재민	완료	100%	2023-03-25	2023-05-20	29	
		3.2.3 바이러스 검사 조건부 구현	김주원/이재민	진행중	90%	2023-03-25		41	
		3.2.4 알림 및 검사 기록 출력 구현	김주원/이재민	진행중	90%	2023-04-01		25	
	3.3 보안 모듈 구축								
		3.3.1 권한 설정	김주원/전윤수	중단	40%	2023-03-25	2023-05-13	35	
		3.3.2 보안 모듈 무결성 검사	김주원/전윤수	중단	40%	2023-03-25	2023-05-13	27	
		3.3.3 암호화 및 인출	김주원/전윤수	중단	40%	2023-03-25	2023-05-13	35	
	3.4 UI/UX 구축								
		3.4.1 UI 디자인	김주원/전윤수	완료	100%	2023-03-17	2023-06-17	32	
		3.4.2 UI 설계	김주원/전윤수	진행중	90%	2023-04-22		46	
		3.4.2 UI 연동 및 기능 구현	김주원/전윤수/이재민/백승규	진행중	80%	2023-05-20		40	
4. 테스트	4.1 테스트 및 리뷰								
		4.1.1 기능 테스트	김주원/전윤수/이재민/백승규	진행중	0%	2023-06-20			
		4.2 성능 테스트	김주원/전윤수/이재민/백승규	진행중	0%	2023-06-20			
		4.3 신뢰성 테스트 및 최종 검수	김주원/전윤수/이재민/백승규	진행중	0%	2023-06-20			
5. 완료	5.1 배포								

