


## 캡스톤디자인 4차 레포트

---



날짜	2024. 05. 07
학 과	인공지능사이버보안
과 목	캡스톤디자인
조	2 조


	캡스톤 디자인 4차 레포트			
	학 과	과 목	조	문서 최종 수정일
	인공지능사이버보안	캡스톤디자인	2 조	2024. 05. 07

목차

1. 서론 ..... 3

2. 피드백 ..... 3

3. 결론 ..... 7

	캡스톤 디자인 4차 레포트			
	학 과	과 목	조	문서 최종 수정일
	인공지능사이버보안	캡스톤디자인	2 조	2024. 05. 07

## 1. 서론

이번 레포트에서도 교수님과 멘토님과의 피드백을 반영한 프로젝트 전체 진행 상황에 대해 말씀드리겠습니다.

## 2. 피드백

4월 26일에 진행된 미팅을 통해서 받은 피드백에 대해 설명하겠습니다.

먼저 저희 팀에서 프로젝트의 업무 진행 상황을 파악하기 위해서 아래와 같은 TODO List를 사용하였습니다.

```
팀 TODO List

[TODO]

중간 발표 준비
-> 중간 발표 자료 작성

[DOING]

점검 모듈 개발
-> action 타입의 결과 성공/실패 여부 체크
-> 점검 스크립트 테스트
-> 결과 정리하여 InspectionItems 테이블에 데이터 Insert

SW 상세설계서 작성

[DONE]

스크립트 작성
-> 일부 윈도우 스크립트 작성

점검 모듈 개발
-> ssh, samba 연결 코드 작성, 예외처리, 동작 테스트
-> 선택된 규제 지침 xml 파일에서 필요한 데이터 파싱


UI, DB
-> DB 테이블 및 컬럼 설계 및 SQLite3로 구현
-> 입력 화면, 규제지침 선택 화면 구현
-> 입력값에 대한 정규표현식 적용
-> DB에 입력된 값을 기반으로 알맞는 규제 지침 선택 테이블 생성
-> 입력된 값과 선택된 규제 지침을 점검 모듈에 전달

[ISSUE]

없습니다.
```

[그림 1] 팀 TODO List

해당 TODO List는 팀 전체 뿐만 아니라 각 업무별로도 주간 마다 작성하여 업무 진행을 확인하고자 하였습니다. 하지만 이와 같은 방식은 전체적으로 업무가 어떻게 진행되고 있는지 한번에 파악하기 힘들다는 피드백을 받아서 멘토님께서 주신 아래 엑셀 양식을 이용하여 업무 진행 상황을 파악하고자 합니다.

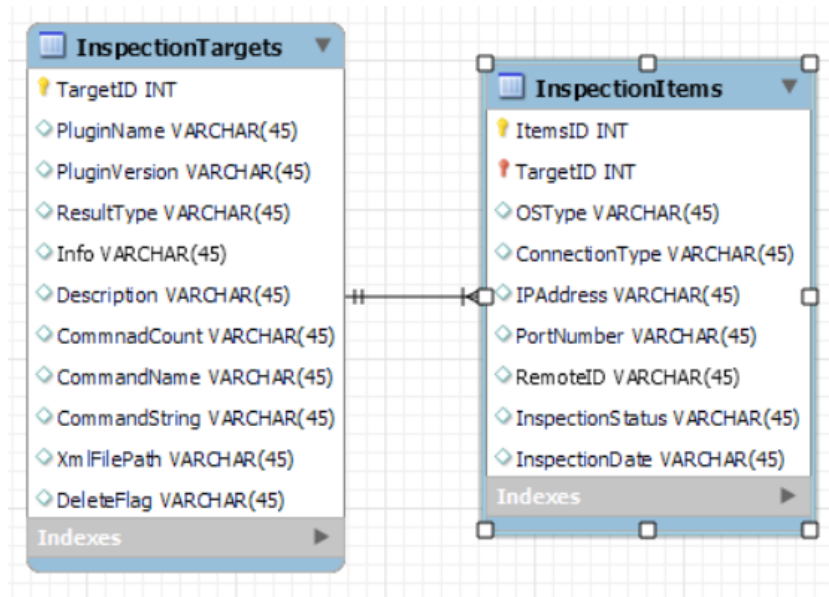
	캡스톤 디자인 4차 레포트			
	학 과	과 목	조	문서 최종 수정일
	인공지능사이버보안	캡스톤디자인	2 조	2024. 05. 07

프로젝트 일정표										
현재 날짜	2024년 5월 1일 수요일	시작일	2024-03-18	신규						
		종료일	2024-06-03	금주 진행						
		주	11	일정 지연(일정보다 진도를 적음)						
		일	0	일정이후 (100%이후)						
				일정내						
중분류	세부항목	시작일	종료일	진도율	기간	경과일	잔여일	담당자	비고	
분석/설계	팀 발당 및 프로젝트 상세 파악	2024-03-18	2024-03-24	100%	7	7	7	팀 전체		
	요구사항 설계 및 목표, 업무 분장 설정	2024-03-18	2024-03-31	100%	14	14	14	팀 전체		
	점검 모듈 설계	2024-03-23	2024-03-31	100%	9	9	9	오병윤, 김연수		
	UI 및 DB 설계	2024-03-28	2024-04-06	100%	10	10	10	김건희, 조유빈		
스크립트 개발	SW 상세설계서 작성	2024-03-18	2024-04-30	80%	35.2	44	44	팀 전체		
	Windows 규제지침 선별	2024-03-23	2024-04-14	100%	23	23	23	이도현, 최정민		
	Windows 점검 항목 스크립트 작성	2024-03-23	2024-05-18	50%	28.5	57	39	18	이도현, 최정민	
	Linux 규제지침 선별	2024-03-23	2024-05-18	0%	0	57	39	18	이도현, 최정민	
점검 모듈 개발	Linux 점검 항목 스크립트 작성	2024-03-23	2024-05-18	0%	0	57	39	18	이도현, 최정민	
	서동, 웹캠 및 주변 어댑터 문서화	2024-04-01	2024-05-19	0%	0	49	30	19	이도현, 최정민	
	점검 대상 접속 모듈 개발	2024-03-31	2024-04-14	100%	15	15	15	김연수		
	XML 파일 데이터 파싱 모듈 개발	2024-03-31	2024-04-14	100%	15	15	15	오병윤		
UI & DB 개발	자동 점검 모듈 개발	2024-03-31	2024-05-20	50%	25.5	51	31	20	오병윤, 김연수	
	점검 모듈 UI 연결	2024-04-09	2024-05-23	20%	9	45	22	23	오병윤, 김건희	
	DB 테이블 및 필드 구현	2024-04-02	2024-04-05	100%	4	4	4	김건희		
	점검 대상 정보 입력 UI	2024-04-01	2024-04-10	100%	10	10	10			
테스트	규제지침 선택 UI	2024-04-01	2024-04-10	80%	8	10	10			
	점검 진행 현황 및 결과 UI			0%	0	1	1			
	점검 이력 상세 조회 UI			0%	0	1	1			
	규제지침 등록 UI			0%	0	1	1			
보고서	점검 항목 세부 내용 UI			0%	0	1	1			
	점검 모듈 테스트 및 최적화			0%	0	1	1			
	UI 테스트			0%	0	1	1			
보고서	최종 보고서	2024-05-27	2024-06-02	0%	7		7			


[그림 2] 프로젝트 일정표

위 진행 상황을 매주 수정하여 전체적으로 업무가 어떻게 진행되고 있는지 와 어떤 업무가 지연되고 있는지 파악하고자 합니다.

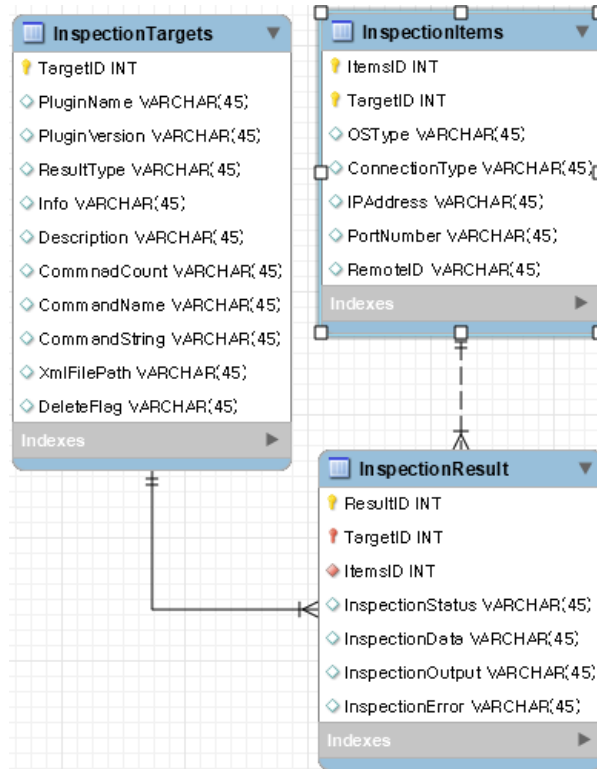
기술적인 내용에서는 먼저 DB에 관한 이야기가 나왔습니다. 기존에 저희가 설계하였던 DB 관계도는 아래와 같습니다.



[그림 3] 이전 DB 관계도

	캡스톤 디자인 4차 레포트			
	학 과	과 목	조	문서 최종 수정일
	인공지능사이버보안	캡스톤디자인	2 조	2024. 05. 07

이 관계도를 보시고 멘토님께서 'InspectionItems 테이블의 점검 대상을 나타내는 컬럼과 점검 결과를 나타내는 테이블을 분리하여 관리하는 것이 더욱 효율적이다' 라고 말씀을 주셔서 해당 내용을 반영하여 아래와 같이 DB를 분할하였습니다.



[그림 4] 수정된 DB 관계도

결과만 따로 저장하고 InspectionItems와 InspectionTargets의 primary key를 외래키로 두어 관계를 설정하였습니다. 해당 방식을 사용했을 경우 점검 내용 상세 보기 등의 기능을 수행할 때 지속적으로 많은 데이터를 조회하지 않고 필요한 데이터만 조회하여 DB 접속에 대한 부담을 줄여 줄 수 있을 것이라 기대합니다.

그 다음에는 점검 항목 선택 페이지에 관한 피드백이 있습니다. 저희가 처음 구현한 점검 항목 선택 화면은 선택한 운영체제에 따라 해당하는 점검 항목 목록을 보여주어 선택할 수 있도록 구현하였습니다.

	캡스톤 디자인 4차 레포트			
	학 과	과 목	조	문서 최종 수정일
	인공지능사이버보안	캡스톤디자인	2 조	2024. 05. 07

취약점 점검 시스템

선택	이름	설명	실행 방식	결과 방식	삭제
<input type="checkbox"/>	백신 프로그램 업데이트	Windows Defender 백신...	Powershell	action	삭제
<input type="checkbox"/>	계정 잠금 임계값 변경	계정 잠금 임계값을 5로 ...	Powershell	action	삭제
<input type="checkbox"/>	Administrator 계정 이름 ...	Administrator 계정 이름...	Powershell	action	삭제
<input type="checkbox"/>	Guest 계정 사용 안함 설정	Guest 계정을 ...	Powershell	action	삭제
<input type="checkbox"/>	컴퓨터 계정 암호 최대 ...	컴퓨터 계정 암호 최대 ...	Powershell	action	삭제
<input type="checkbox"/>	원격으로 액세스 할 수 ...	원격 레지스트리 서비스...	Powershell	action	삭제
<input type="checkbox"/>	원격에서 이벤트 로그파...	원격에서 로그 파일을 ...	Powershell	action	삭제
<input type="checkbox"/>	D8 로그인 시 Windows ...	적절한 Windows 인증 ...	Powershell	action	삭제

<

점검 실행

[그림 5] Windows 대상 점검 항목 선택 화면

취약점 점검 시스템


선택	이름	설명	실행 방식	결과 방식	삭제
<input type="checkbox"/>	계정 잠금 임계값 변경	계정 잠금 임계값을 5회...	Bash	action	삭제

<

점검 실행

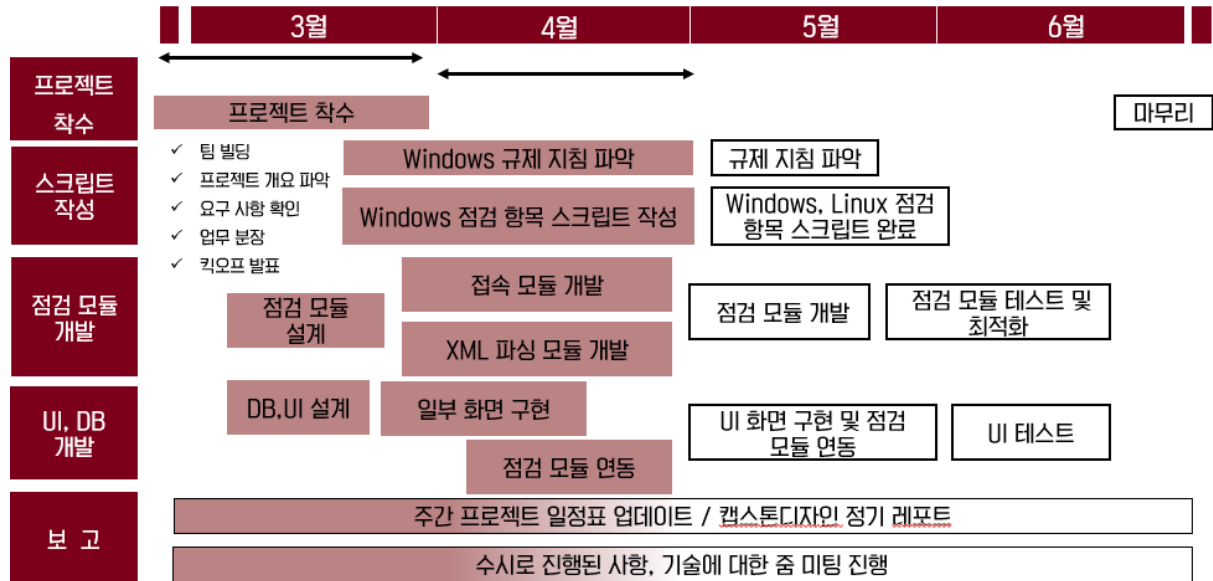
[그림 6] Linux 대상 점검 항목 선택 화면

이를 확인하신 멘토님께서서는 일부 점검 항목들은 운영체제에 종속되지 않고 동작하는 항목이 있을 수 있어 이러한 상황을 고려해야 한다고 말씀해 주셨습니다. 따라서 실제 그러한 항목이 있는지 주요정보통신기반시설 기술적 취약점 분석 평가 상세 가이드에서 확인 후 그러한 항목들은 어떻게 처리할지 논의하여 구현할 예정입니다.

	캡스톤 디자인 4차 레포트			
	학 과	과 목	조	문서 최종 수정일
	인공지능사이버보안	캡스톤디자인	2 조	2024. 05. 07

### 3. 결론

이번 레포트에서는 원전에 대한 보안 취약점 자동화 점검 도구는 어떻게 작성되었는지 살펴보았고, 여태 진행되었던 사항을 교수님과 멘토님에게 의논하여 얻은 피드백을 어떻게 반영했는지에 대해 보여드렸습니다. 이제까지 진행했던 진행 사항을 일정표로 나타내면 아래와 같습니다.



[그림 7] 팀 실제 수행 일정

중간 발표 이후에는 Linux와 Windows에 대한 규제 지침 선별 및 스크립트 작성을 완성하고 이에 따라 점검 모듈에서 점검 스크립트를 사용하였을 때 결과 타입에 따른 점검 여부 설정 알고리즘을 완성할 예정입니다. UI 또한 나머지 설계하였던 화면을 모두 구현하여 점검 모듈, DB와 연동한 다음 전반적으로 코드에 대한 리뷰 및 리팩토링을 진행할 예정입니다. 마지막으로요구사항대로 프로그램이 정상적으로 동작하는지 간단하게 테스트를 진행하여 프로젝트를 마칠 예정입니다