

오픈소스SW의 이해

CHAPTER 10

SEMESTER | 2024-2

박성미

목차

Table of Contents



01. 협업 문서

02. 기말 프로젝트

협업 문서

한림대학교

요구사항 명세서

- 개발자와 클라이언트 간의 손통을 돕기 위해 제작
- 프로젝트 업무 범위와 상세 기능 명시
- 비즈니스 요구사항: 서비스의 목표와 전략적 방향, 가치 제공 (왜 해야 하는지?)
- 사용자 요구사항: 최종 사용자의 관점에서 시스템이 제공해야 할 기능과 특성 (무엇을 할 수 있는지?)
- 기능적 요구사항: 시스템이 수행해야 하는 특정 기능이나 작업을 정의 (무엇을 해야 하는지?)
 - 예) 사용자 로그인 기능, 데이터 검색 기능, 주문 처리 기능
- 비기능적 요구사항: 시스템의 품질 속성을 정의 ➔ 시스템이 어떻게 작동해야 하는지에 대한 기대치를 설정
 - 예) 성능: 응답 시간, 처리량, 보안: 데이터 보호, 사용성: 사용 편의성...

요구사항 명세서

1.1 시스템 개요

오늘날 대부분의 기업에서는 정보 시스템 구축이 필수적인 기반으로 자리잡았다. 그에 따라 침해 사고보안에 대한 중요성은 날로 커지고 있다. 특히 2017년 6월 국내 웹 호스팅 기업 '나야나'에서 발생한 침해사고는 막대한 규모의 피해를 가져왔고, 비교적 안전하다고 여겨졌던 리눅스 서버 역시 침해사고의 안전지대가 아니라는 뼈아픈 교훈을 남겼다. 이런 실정을 개선하기 위해서는 사업체의 투자와 전문가 확보가 가장 우선적으로 이루어져야 하나, 많은 사업체들이 정보보호의 비용적인 부담감을 느끼고 있는 상황이다. 그럼에도 침해사고는 소수의 대기업에만 일어나는 사이버 공격이 아니라는 점을 감안하면 비용적 측면에서 도입 및 관리가 용이한 침해사고 대비책이 요구된다. 해당 서비스는 오픈 소스를 기반으로 하여 기업의 부담을 줄일 수 있으며, 특히 중소기업 대상으로 서비스를 제공할 수 있다.

아티팩트를 수집하고, 백업하며, 초기 대응에 필요한 아티팩트들은 애플리케이션을 통해 시각화하는 것이 최종 산출물이다. 애플리케이션은 사용자 ID와 사용자 IP 별로 서비스를 제공한다. 수집된 아티팩트들은 사용자가 원하는 시간, 필드를 선택해서 볼 수 있으며, 제공되는 그래프를 통해 의심가는 아티팩트를 추려서 볼 수 있다. 또한 침해사고에 미숙한 사용자의 의사 선택을 돕기 위해 침해사고의 상황에 따라 체크리스트를 제공한다.

2. 비즈니스 요구사항

비즈니스 요구사항	
Needs	중소기업에선 운영하는 서버에 해킹사고가 발생했을 때를 대비한 로그 수집 등의 준비가 미흡하다.
Goal	해킹사고 상황을 대비해 미리 로그를 수집, 백업, 시각화해주는 서비스를 제공한다.
Objective	지속적으로 로그를 수집해 저장해 사고발생시에 분석을 진행할 수 있는 데이터를 확보하도록 한다.
Operation Concept	서버관리자가 조치를 취하지 않아도 로그를 지속적으로 수집여 저장하도록 한다.
	수집된 로그에 대한 분석정보를 웹으로 볼 수 있도록 한다.

3. 사용자 요구사항

사용자 요구사항	반영여부
기능 요구사항	
지속적인 아티팩트 수집 기능	o
로그 백업 기능	o
비정형 로그 파싱 기능	o
로그 중복 처리 기능	o
로그 검색 기능	o
타임라인 기능	o
익스플로러 기능	o
의심행위 판단 기능	△
체크리스트 기능	△
사용자 웹셀 기능	x

4. 기능적 요구사항

요구사항 번호	기능-1	요구사항 명	지속적인 아티팩트 수집 기능
요구사항 설명	서버 환경에 맞는 아티팩트 저장 경로를 자동으로 수집하고 그에 맞는 아티팩트 리스트를 제공한다.		
	수집기는 데몬 프로세스로 작동해 지속적으로 아티팩트를 수집한다.		
	아티팩트는 시스템 로그 및 네트워크 패킷을 포함한 최대한의 로그들을 수집하며, 수집된 데이터는 로컬 저장공간에 저장한다.		
	로컬에 저장된 데이터는 네트워크 트래픽이 적은 시간대를 지정하여 원격 저장소에 백업한다.		
관련 요구 사항			
전제조건		우선순위	높음

5. 비기능적 요구사항

5.1. 하드웨어 요구사항

요구사항 번호	비기능-HW-1	요구사항 명	작동 플랫폼
요구사항 설명	본 시스템은 리눅스 서버 에서 작동 되도록 한다.		
	애플리케이션은 Ubuntu 및 CentOS에 Python 언어를 기반으로 설치해 테스트한다.		
관련 요구 사항			
전제조건		우선순위	

5.2. 성능 요구사항

요구사항 번호	비기능-성능-1	요구사항 명	에이전트 서버 리소스 최소 사용 기능
요구사항 설명	에이전트에는 수집기와 전송 도구만을 운용한다.		
	수집된 아티팩트들은 전송 도구를 통해 데이터를 압축되고 저장 및 분석 서버로 전송된다.		
	수집기와 전송 도구는 사용하지 않을 때에는 리소스를 사용하지 않는다.		
관련 요구 사항			
전제조건		우선순위	높음

요구사항 명세서

1.1 시스템 개요

오늘날 대부분의 기업에서는 정보 시스템 구축이 필수적인 기반으로 자리잡았다. 그에 따라 침해 사고보안에 대한 중요성은 날로 커지고 있다. 특히 2017년 6월 국내 웹 호스팅 기업 '나야나'에서 발생한 침해사고는 막대한 규모의 피해를 가져왔고, 비교적 안전하다고 여겨졌던 리눅스 서버 역시 침해사고의 안전지대가 아니라는 뼈아픈 교훈을 남겼다. 이런 실정을 개선하기 위해서는 사업체의 투자와 전문가 확보가 가장 우선적으로 이루어져야 하나, 많은 사업체들이 정보보호의 비용적인 부담감을 느끼고 있는 상황이다. 그럼에도 침해사고는 소수의 대기업에만 일어나는 사이버 공격이 아니라는 점을 감안하면 비용적 측면에서 도입 및 관리가 용이한 침해사고 대비책이 요구된다. 해당 서비스는 오픈 소스를 기반으로 하여 기업의 부담을 줄일 수 있으며, 특히 중소기업 대상으로 서비스를 제공할 수 있다.

아티팩트를 수집하고, 백업하며, 초기 대응에 필요한 아티팩트들은 애플리케이션을 통해 시각화하는 것이 최종 산출물이다. 애플리케이션은 사용자 ID와 사용자 IP 별로 서비스를 제공한다. 수집된 아티팩트들은 사용자가 원하는 시간, 필드를 선택해서 볼 수 있으며, 제공되는 그래프를 통해 의심가는 아티팩트를 추려서 볼 수 있다. 또한 침해사고에 미숙한 사용자의 의사 선택을 돕기 위해 침해사고의 상황에 따라 체크리스트를 제공한다.

2. 비즈니스 요구사항

비즈니스 요구사항	
Needs	중소기업에선 운영하는 서버에 해킹사고가 발생했을 때를 대비한 로그 수집 등의 준비가 미흡하다.
Goal	해킹사고 상황을 대비해 미리 로그를 수집, 백업, 시각화해주는 서비스를 제공한다.
Objective	지속적으로 로그를 수집해 저장해 사고발생시에 분석을 진행할 수 있는 데이터를 확보하도록 한다.
Operation Concept	서버관리자가 조치를 취하지 않아도 로그를 지속적으로 수집여 저장하도록 한다.
	수집된 로그에 대한 분석정보를 웹으로 볼 수 있도록 한다.

3. 사용자 요구사항

사용자 요구사항	반영여부
기능 요구사항	
지속적인 아티팩트 수집 기능	o
로그 백업 기능	o
비정형 로그 파싱 기능	o
로그 중복 처리 기능	o
로그 검색 기능	o
타임라인 기능	o
익스플로러 기능	o
의심행위 판단 기능	△
체크리스트 기능	△
사용자 웹셀 기능	x

4. 기능적 요구사항

요구사항 번호	기능-1	요구사항 명	지속적인 아티팩트 수집 기능
요구사항 설명	서버 환경에 맞는 아티팩트 저장 경로를 자동으로 수집하고 그에 맞는 아티팩트 리스트를 제공한다.		
	수집기는 데몬 프로세스로 작동해 지속적으로 아티팩트를 수집한다.		
	아티팩트는 시스템 로그 및 네트워크 패킷을 포함한 최대한의 로그들을 수집하며, 수집된 데이터는 로컬 저장공간에 저장한다.		
	로컬에 저장된 데이터는 네트워크 트래픽이 적은 시간대를 지정하여 원격 저장소에 백업한다.		
관련 요구 사항			
전제조건		우선순위	높음

5. 비기능적 요구사항

5.1. 하드웨어 요구사항

요구사항 번호	비기능-HW-1	요구사항 명	작동 플랫폼
요구사항 설명	본 시스템은 리눅스 서버 에서 작동 되도록 한다.		
	애플리케이션은 Ubuntu 및 CentOS에 Python 언어를 기반으로 설치해 테스트한다.		
관련 요구 사항			
전제조건		우선순위	

5.2. 성능 요구사항

요구사항 번호	비기능-성능-1	요구사항 명	에이전트 서버 리소스 최소 사용 기능
요구사항 설명	에이전트에는 수집기와 전송 도구만을 운용한다.		
	수집된 아티팩트들은 전송 도구를 통해 데이터를 압축되고 저장 및 분석 서버로 전송된다.		
	수집기와 전송 도구는 사용하지 않을 때에는 리소스를 사용하지 않는다.		
관련 요구 사항			
전제조건		우선순위	높음

연결

R&R (Role and Responsibility)

- 팀 내 구성원의 역할과 책임은 명확히 정의 ➔ 업무 분담의 효율성 보장
- 업무의 중복을 피하고 효율성 향상이 주목적

- 역할: 팀원에게 주어진 활동/책임 명시
- 책임: 역할에 따라 수행해야 하는 의무/업무 범위 명시, 결과물 기준 제시

R&R (Role and Responsibility)

Role	Name	Responsibilities	Deliverables
[INSERT JOB TITLE]	[INSERT NAME]	EXAMPLE: Research skill market	EXAMPLE: Training Recommendations
[INSERT JOB TITLE]	[INSERT NAME]	EXAMPLE: Oversee course creation	EXAMPLE: 12-video modules
[INSERT JOB TITLE]	[INSERT NAME]	EXAMPLE: Drive employee participation	EXAMPLE: Staff announcement emails

<https://wellhub.com/en-us/blog/employee-development/roles-and-responsibilities-template/>

나의 주 역할은?

주요기능/문서 관리/플랫폼 중심 역할 분배 ➔ 해당 역할의 의무 정리

ID	업무	담당자	SubID	세부사항
1	Github 데이터 구축 및 관리 (Github 데이터 관리자)	홍길동	1-1	데이터 구조 설계
			1-2	데이터 관리 방안

결국, 업무 분담!!!

화면 설계서 & IA (Information Architecture)

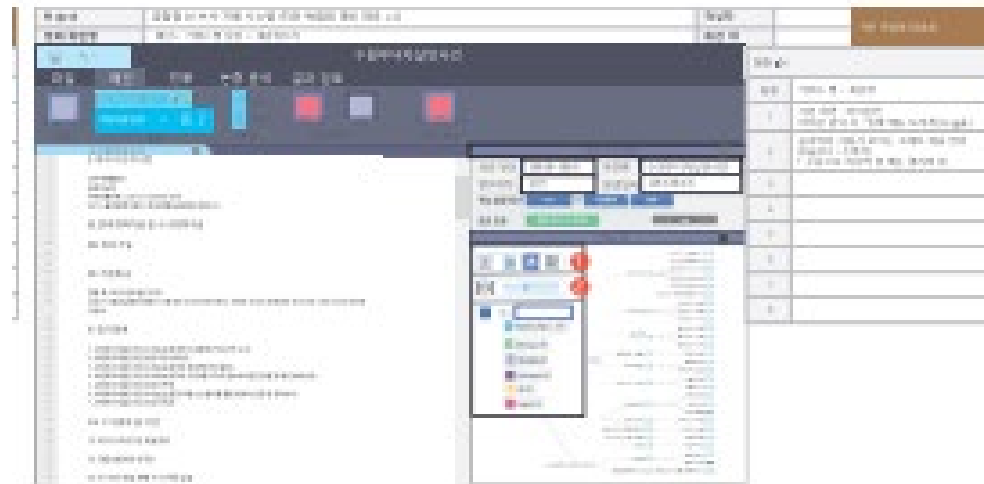
❖ 화면설계서

- 프로젝트 요구사항 및 기능 정의를 바탕으로 목표 서비스 또는 시스템 화면을 설계
- 화면의 구성요소, 기능, 정책 등을 자세히 정의
- 화면, 기능 단위

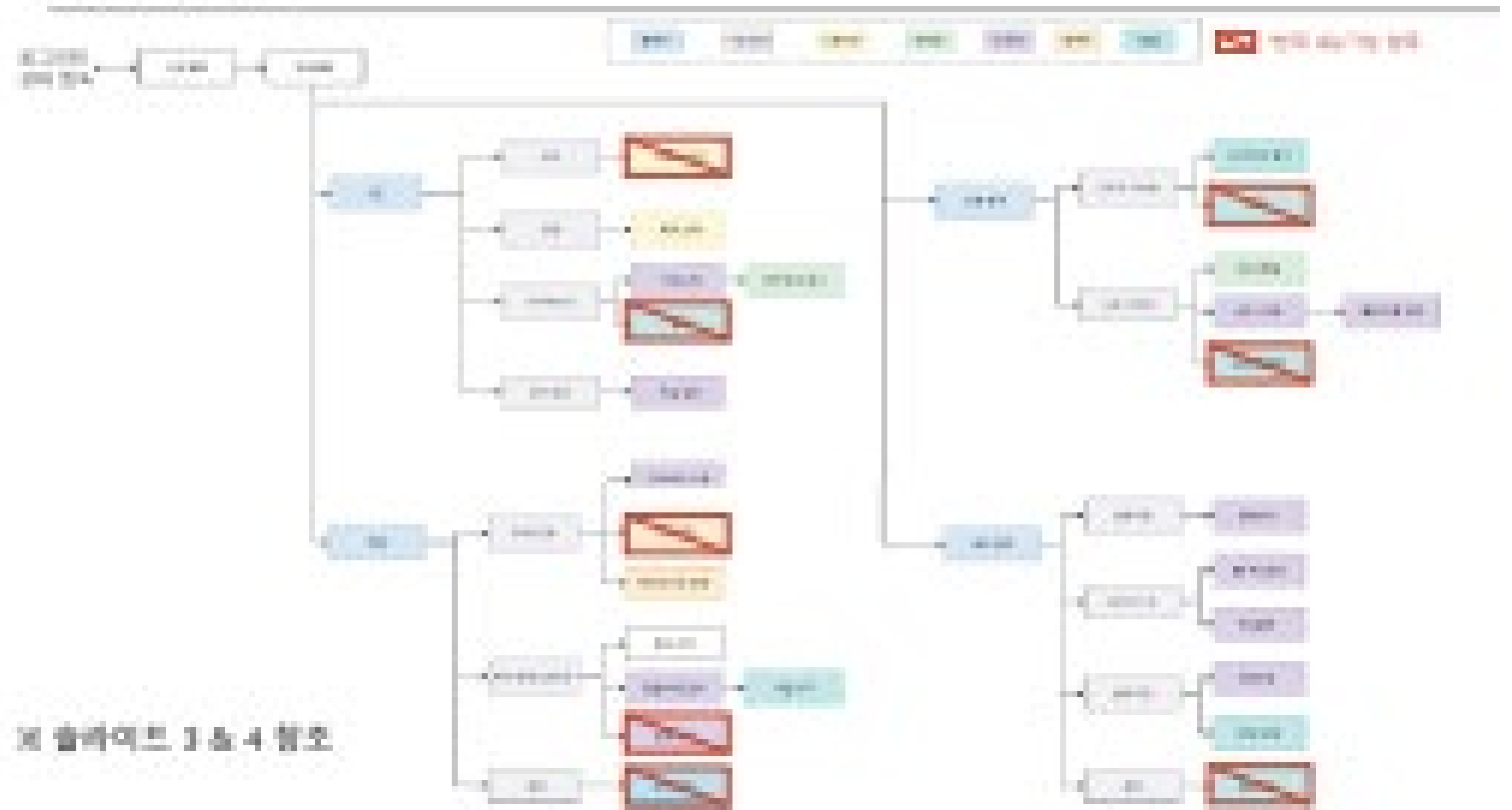
❖ IA

- 메뉴를 분류 그룹화하여 Depth 구조로 설계
- 정보를 다양한 목적, 기능에 따라 분류해서 정리

화면 설계서 & IA (Information Architecture)



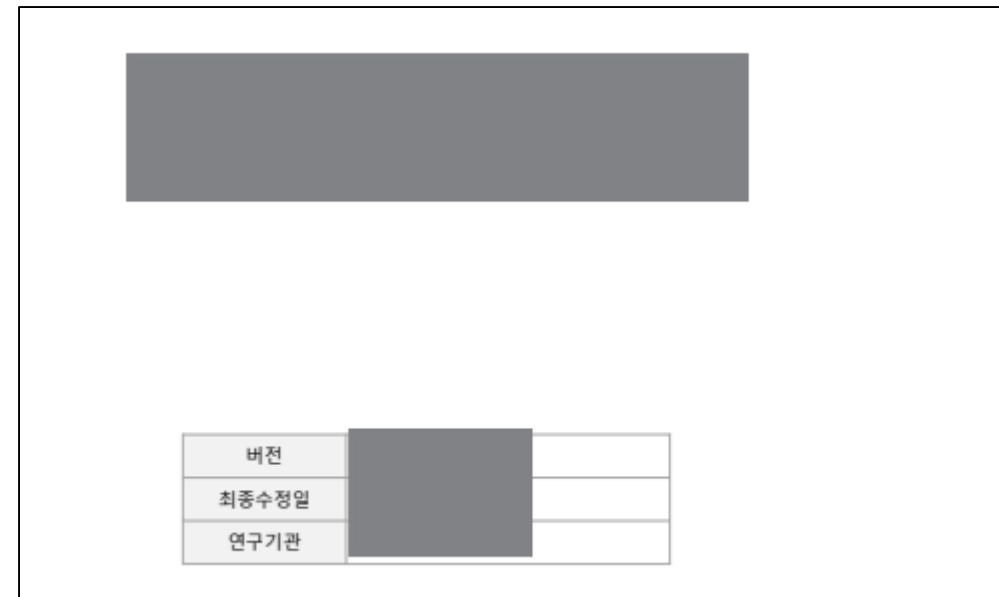
IA 및 화면흐름도



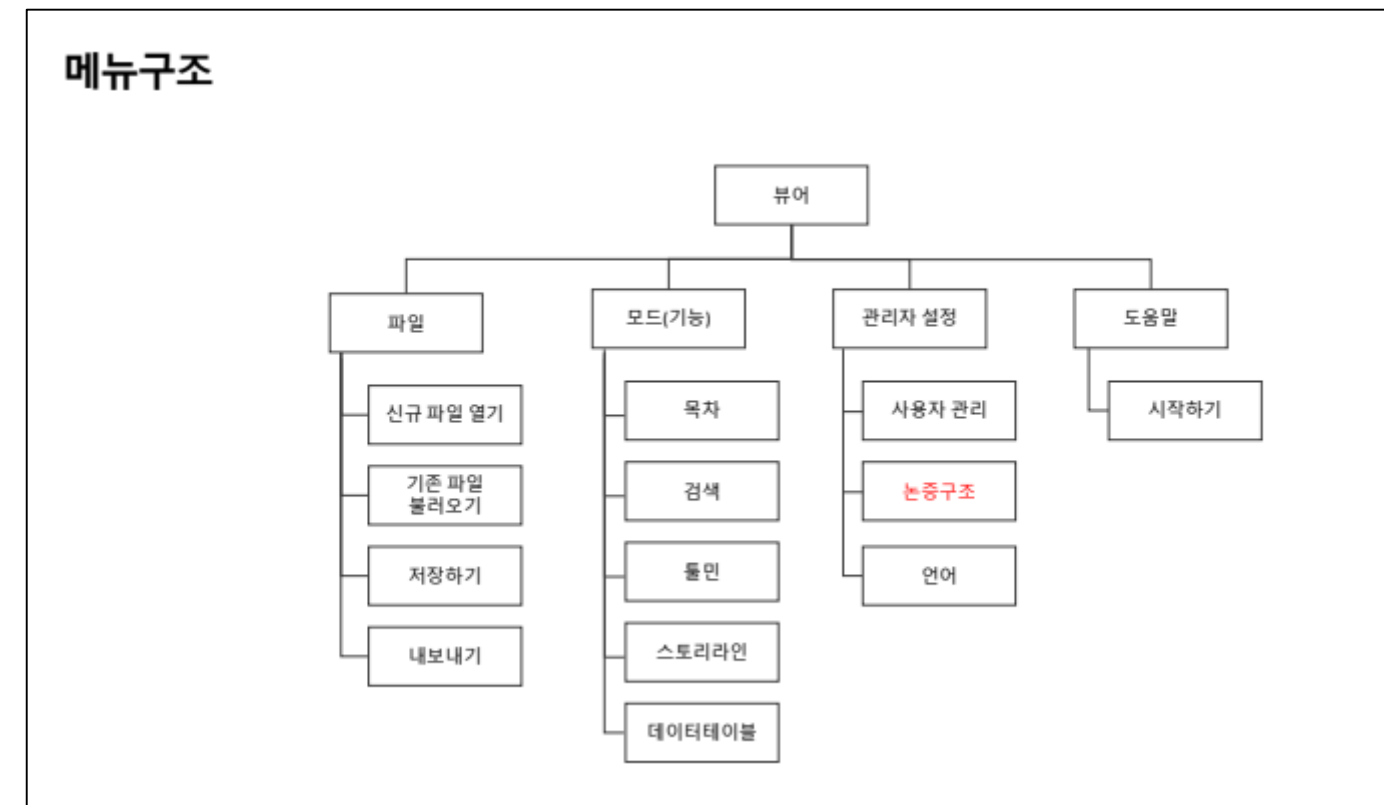
※ 슬라이드 3 & 4 참조

!!! 참조 자료: <https://yslab.kr/148>

1. 표지
2. 문서변경이력
3. **플로우차트**
4. 메뉴구조
5. 화면설계서



날짜	버전	변경사항
2022. 02. 24.	0.1	- 최초 작성
2022. 03. 04	0.2	-
2022. 03. 07	0.3	-
2022. 03. 08	0.31	-
2022. 03. 10	0.4	-
2022. 04. 02	0.5	-
2022. 04. 11	0.6	-
2022. 04. 12	0.7	-
2022. 04. 12	0.71	-
2022. 04. 13	0.8	-
2022. 05. 09	0.9	-



Page Title	수정 보완 검토사항	Page ID	P04	Description	Page	6
Path	[메인화면] > [수정/보완]					

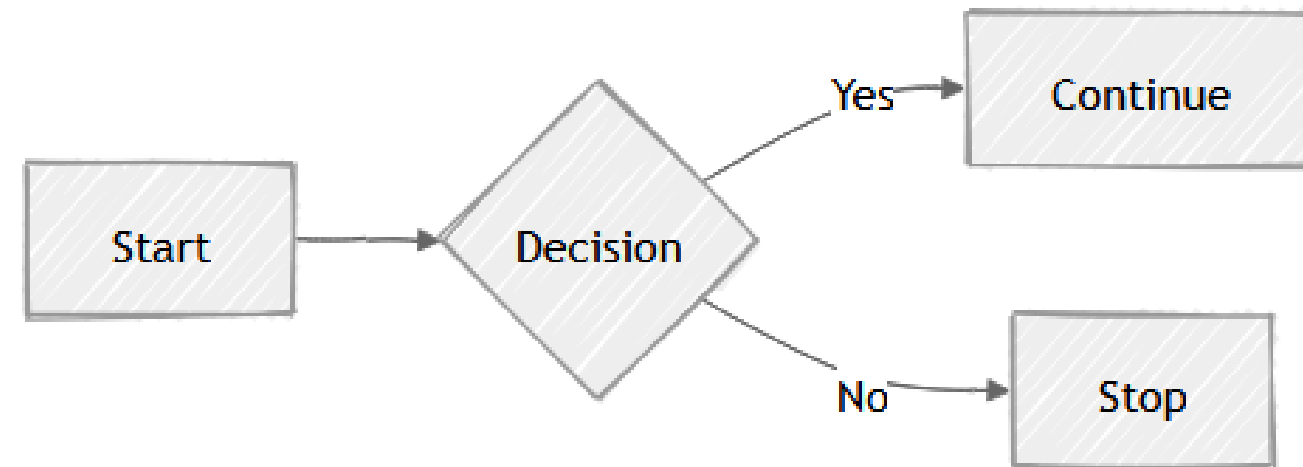
1	<p>사용자 텍스트와 관련된 법령 용어 추출 (수정 가능)</p> <p>※ <u>전체 텍스트</u>를 대상으로 할 것인지 <u>문맥</u>에 따라 일부 텍스트를 대상으로 할 것인지, 사용자가 <u>선택한</u> 일부 문단을 대상으로 할 것인지 등의 필요. 영문법 교정기 Grammarly의 경우 전체 텍스트를 대상으로 교정할 문장을 탑재함</p>
2	사용자 텍스트와 관련된 참조 조문 추출
3	마우스 hover 시 해당 조문 표시
4	<p>수정 보완해야할 검토사항 유형별 탭</p> <ul style="list-style-type: none"> - 클릭 시 각 유형별 검토사항 표시 - 유형 별 검토사항 건수 표시
5	해결되지 않은 검토사항 진행 바(bar)
6	<p>클릭 시 모든 수정사항 표시 (문서 위치 순)</p> <p>- [POS]로 이동</p>

화면 설계서 & IA (Information Architecture)

<https://www.mermaidchart.com/>

❖ mermaid

- 텍스트와 코드로 다이어그램 생성
- Javascript 기반 다이어그램 / 차트 제작 도구
- Markdown 과 함께 사용 가능



```
---config:
  look: handDrawn theme: neutral
---
flowchart LR
  A[Start] --> B{Decision}
  B --> |Yes| C[Continue]
  B --> |No| D[Stop]
```

화면 설계서 & IA (Information Architecture)

플로우 차트 (Flow Chart)

- 특정 대상의 정보/사건 등 흐름도 표현

노드 (상자) 하나 그리기

```
---
title: Node
---
flowchart LR
  id
```

다이어그램 유형 선언하고 시작!

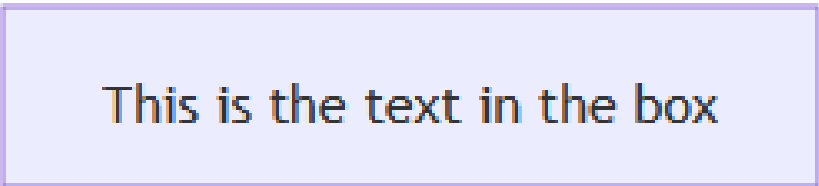
Node



노드에 글 쓰기 (인코딩 문제 발생시 “ 사용)

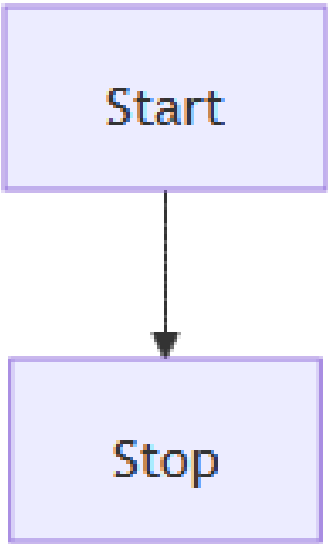
```
---
title: Node with text
---
flowchart LR
  id1[This is the text in the box]
```

Node with text



방향

```
flowchart TD
  Start --> Stop
```



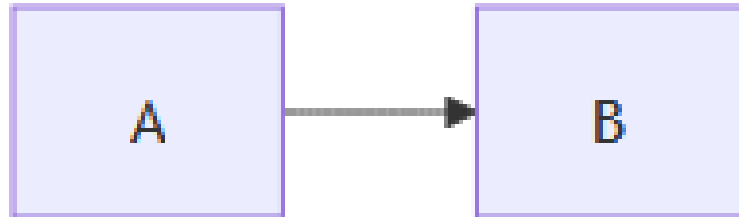
- TB - Top to bottom
- TD - Top-down/ same as top to bottom
- BT - Bottom to top
- RL - Right to left
- LR - Left to right

화면 설계서 & IA (Information Architecture)

플로우 차트 (Flow Chart)

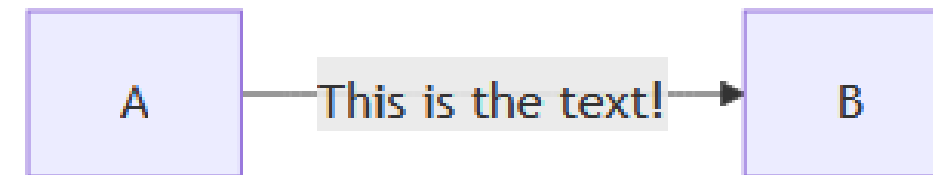
선

```
flowchart LR
  A-->B
```



```
flowchart LR
```

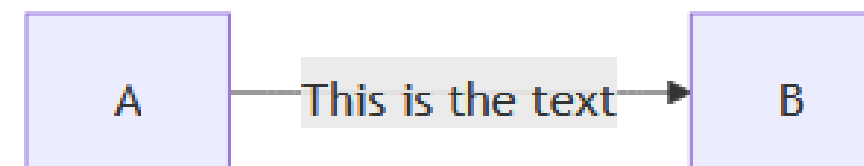
```
A-- This is the text! ---B
```

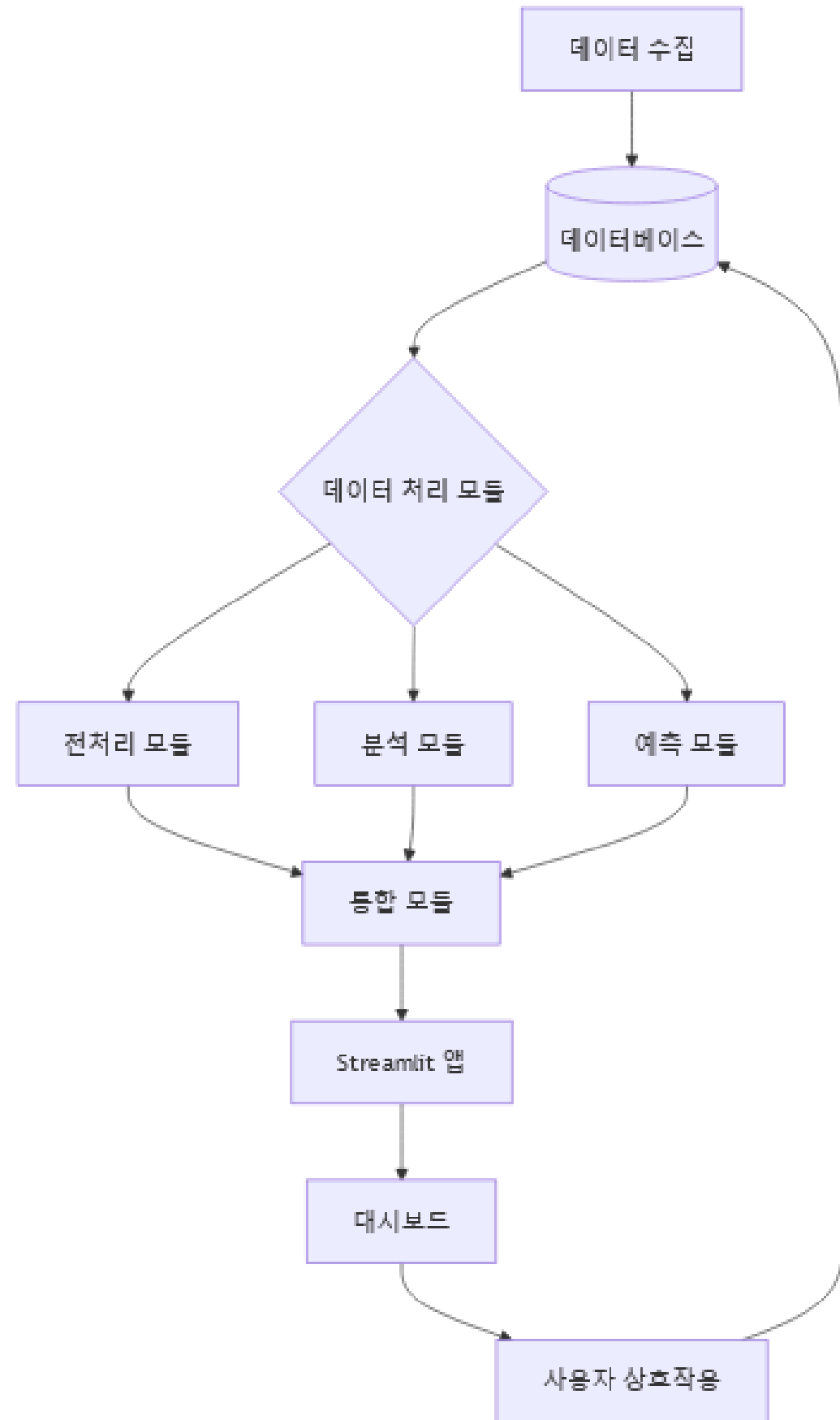


둘 중 하나!

```
flowchart LR
```

```
A---|This is the text|B
```





Mermaid로 만들어보기

<https://mermaid.live>

기말 프로젝트

한림대학교

기말프로젝트

1.프로젝트 계획

- 주제 및 목표 선정
- 프로젝트 작업 단위로 분해하여 WBS 작성
- 팀원별 역할 정의 및 분담 (RnR)

OUT:

- WBS 문서 및 역할 분담(R&R)표 (엑셀, PDF)
- 발표자료 (프로젝트 계획 및 목표)

2.기능 구현

- 개인 핵심 기능 2건 이상 (Fork해서 -> main으로 merge) * RnR에서 할당
- 팀 공동 개발 기능 최소 1건 이상 (동일 파일 수정 및 기능 추가 -> 충돌 해결, 여러 브랜치 머지 -> 그 후 main)
- 그 외 여러 기능 들 각자 분담, 이슈 & PR 진행

※ 조건: Python 기반, Streamlit 사용, 가상환경 사용, GPT 사용 시 대화기록 부록으로 첨부, 주석은 본인이!!!

OUT:

- 화면설계서/기능명세서 등 기능 설명 자료,
- 발표 자료 (구현 영상)

기말프로젝트

3. 협업 및 문제 해결

- 프로젝트 README 작성
 - 프로젝트 개요, 설치방법, 사용방법, 주요 기능 설명 등
- Github branch, PR, **Issue**, Milestone, Project, 라벨, assign, **코드 리뷰** 운영
- **회의록 작성 (4회) - 논의사항/진행상황 + 회의 결과 정리 (과제!!!)**
- 클라썸 외 디스코드 등 사용 가능 (단, 이 경우 대화 첨부. 개인/업무 대화 분리!!!)

OUT: 회의록, 커밋 및 코드 기여, PR 및 Issue 해결, 발표 자료 (활동 기록/문제 해결)

제출 자료

- ✓ 보고서 1부: (팀 & 개인)
- ✓ 발표자료 1부: (팀)
- ✓ 최종 레포 1: (팀)

[illegible]

12월 2024				
월요일	화요일	수요일	목요일	금요일
2	3  팀플	4	5  팀플	6
9  1차 제출	10  발표	11	12  발표	13  발표
16	17	18	19  최종 제출	20
23	24	25	26	27
30	31			

보충기간

기말프로젝트

- 오늘 할일 - Brainstorming
 - 주제 확정하기
 - 코어 기능 정하기
 - 업무 분담하기
 - 일정 정하기

평가 항목

구분	항목	세부 평가 기준	
화면 설계서 등 협업 문서	화면 설계서	직관성과 완성도, 시스템과의 일관성	+ 외부평가
	기타 협업문서	구조의 명확성, 업무 효율성	
시스템 개발 및 디플로이	기능 구현 완성도	요구된 기능 완성도, 추가 기능의 혁신성/유용성	
	디플로이 안정성	시스템 안정성, 배포 시 발생 이슈 해결력	
라이선스 선택 및 적용	라이선스 선택 및 적용	코드와 문서에 라이선스 명확히 표시	+ 내부평가
팀 공동 개발	충돌 해결	동일 파일 수정 후 충돌 해결	
	브랜치 병합	여러 브랜치 작업 후 main 병합	
	브랜칭 전략 활용도	브랜치 관리 및 활용력	
개별 기능 및 R&R	기술 난이도 및 기여도	난이도, 기여도 등 추가점수	
	핵심 기능	Fork 후 main으로 병합한 기능 2건 이상	
	추가 기능 분담 및 구현	배정받은 추가 기능 구현	
명세서 작성	명세서 완성도	기능 및 모듈 설명의 명확성과 구체성	
	문서 구조 및 가독성	논리적 구성과 가독성	
PR 및 코드 리뷰	PR 작성 능력	작업 내용을 명확히 설명한 PR 작성	
	코드 리뷰 참여	팀원 PR 리뷰와 자신의 PR 대응	

수업시간에 배운 것 최대한 활용!!!

오늘은 여기까지.

Thank you for listening.