

# Check Mate SPMP



## - SW Project Management Plan -

팀명 : 체크 메이트

팀원 :

- 2017038063 박성진
- 2017038064 김동용
- 2017038069 이동우

# **- 목 차 -**

## **1. Introduction ----- 4**

- \* Purpose of this document**
- \* Project overview**
- \* Related documents, terms and so on**

## **2. Development Plan ----- 5**

- \* Resource : Staffing, Cost**
- \* WBS**
- \* Schedule (Gantt chart)**

## **3. Organization ----- 10**

- \* Team structure**
- \* Role and Responsibility**

## **4. Technical Management ----- 11**

- \* Change and Configuration management**
- \* Technology management**

<b>5. Quality control</b>	<b>14</b>
* Review method	
* Review periodic	
* Other quality control techniques	
<b>6. Development Environment</b>	<b>15</b>
* Required software and spec	
* Hardware spec	
* Space and security	
<b>7. Deliverables</b>	<b>16</b>
* Define the documents	
* Date and destination	
<b>8. Others</b>	<b>17</b>
* Considerable issues	
<b>9. References and Appendix</b>	<b>18</b>

# 1.Introduction

## 1.1 Purpose of this document

본 문서는 SW Project Management Plan 문서로, 프로젝트 관리를 위한 계획들을 수립하고 명세하고자 작성되었습니다. 프로젝트에 대한 개요를 시작으로 전반적인 프로젝트 기획부터 개발 계획까지 자세하게 기술하였습니다.

## 1.2 Project overview

체크 메이트는 기존 출석 방식의 문제점을 해결하고자 기획된 출결 관리 시스템입니다.

기존 출석 방식에는 다음과 같은 여러가지 문제점들이 존재합니다.

1. 대형 강의 내 소요되는 긴 출석 시간
2. 출석 후 이탈, 대리 출석 등과 같은 부정 출석 가능성
3. 수강생의 출결 확인, 교수자의 출결 관리 시 번거로움과 어려움

저희 서비스는 이러한 한계점을 다양한 부정 출석 방지 로직과 편리한 자동화 출석 관리 시스템을 통해서 해결하고자 합니다. 또한 사용자 편리성을 위해 PC, 모바일 환경 모두 지원합니다.

## 1.3 Related documents, terms and so on

\* (전자출결 시스템의 문제점과 해결방안에 대한 연구 - 사용자 인식을 중심으로 -, 이재학, 이희화, 한국디지털정책학회, 2019, 41 - 49)

\* (스마트기기를 이용한 상호협력기반 전자출결 시스템, 정필성, 조양현, 한국융합학회, 2021, 19 - 27)

## 2.Development Plan

### 2.1 Resource : Staffing, Cost

정확한 Function Point 산정을 위해 아래와 같은 단계에 맞춰 FP 를 계산하였습니다.

1. Determine the type of project:  
⇒ New project
2. Determine the scope of target system:  
⇒ Whole and In-house development
3. Identify data functinos & their complexity:

Internal Logical File		Data Element Type		
		1-19	20-50	>=51
Record Element Type	1	Low	Low	Mid
	2-5	Low	Mid	High
	>5	Mid	High	High

4. Identify transaction functions and their complexity

External Input		Data Element Type		
		1-4	5-15	>=16
Reference File Type	<2	Low	Low	Mid
	2	Low	Mid	High
	>2	Mid	High	High

5. Produce the unadjusted FP

Functions	Complexity Level (Weight)		
	Low	Mid	High
<b>ILF</b>	X 7	X 10	X 15
<b>EIF</b>	X 5	X 7	X 10
<b>EI</b>	X 3	X 4	X 6
<b>EO</b>	X 4	X 5	X 7
<b>EQ</b>	X 3	X 4	X 6

⇒ 따라서, UFP = (10 + 7 + 4 + 4 + 3) = 28

6. Determine the value adjustment factors(VAF)

- ⇒ 1. Data Communication: 4 points
- ⇒ 2. Distributed Data Processing: 3 points
- ⇒ 3. Performance: 3 points
- ⇒ 4. Heavily Used Configuration: 0 points
- ⇒ 5. Transaction Rate: 4 points
- ⇒ 6. Online Data Entry: 5 points
- ⇒ 7. End User Efficiency: 5 points
- ⇒ 8. Online Update: 5 points
- ⇒ 9. Complex Processing: 2 points
- ⇒ 10. Reusability: 4 points
- ⇒ 11. Installation Ease: 0 points

⇒ 12. Operational Ease: 4 points

⇒ 13. Multiple Sites: 0 points

⇒ 14. Facilitate Change: 5 points

그래서 TDI (Total Degree of Influence)는 44 points 입니다.

VAF(Value Adjustment Factors)는  $(TDI * 0.01) + 0.65 = 1.09$  입니다.

7. Produce the adjusted FP

따라서 저희 프로젝트의 AFP(Adjusted Function Point)는  $UAF * VAF = 28 * 1.09$

$= 30.52$  points 입니다.

이를 LOC(Lines Of Codes)로 환산하면, JAVA 기준으로  $30.52 * 53 = 1617$  Lines 입니다.

## 2.2 WBS

단계 구분	주요 업무	세부 업무	선행 작업	상태	진척율	시작일	종료일	작업기간
1. 분석	1.1 고객 요구 분석			작업 전	0%	2021-10-01	2021-11-01	32
		1.1.1 요구사항 정의		작업 전	0%	2021-10-01	2021-10-07	7
		1.1.2 요구사항 수집		작업 전	0%	2021-10-08	2021-10-21	14
		1.1.3 요구사항 분석		작업 전	0%	2021-10-22	2021-10-29	8
		1.1.4 시스템 제안서 작성		작업 전	0%	2021-10-30	2021-11-01	3
2. 설계	2.1 기획 및 화면 설계			작업 전	0%	2021-11-02	2021-11-15	14
		2.1.1 프로젝트 관리 계획서 작성		작업 전	0%	2021-11-02	2021-11-08	7
		2.1.2 세부기능 개발 정의		작업 전	0%	2021-11-02	2021-11-08	7
		2.1.3 프로젝트 화면 설계		작업 전	0%	2021-11-09	2021-11-15	7
	2.2 개발 프로세스 설계			작업 전	0%	2021-11-02	2021-11-29	28
		2.2.1 프로젝트 흐름도 작성		작업 전	0%	2021-11-02	2021-11-08	7
		2.2.2 기능적 모델링		작업 전	0%	2021-11-09	2021-11-15	7
		2.2.3 Activity, Use-Case Diagram		작업 전	0%	2021-11-09	2021-11-15	7
		2.2.4 구조적 모델링		작업 전	0%	2021-11-16	2021-11-22	7
		2.2.5 행위적 모델링		작업 전	0%	2021-11-23	2021-11-29	7
		2.2.6 모델링 밸런싱		작업 전	0%	2021-11-23	2021-11-29	7
	2.3 시스템 구조 설계			작업 전	0%	2021-11-16	2021-11-23	8
		2.3.1 물리적 구조 선택		작업 전	0%	2021-11-16	2021-11-18	3
		2.3.2 구조별 기술 스택 선정		작업 전	0%	2021-11-18	2021-11-21	4
		2.3.3 시스템 구조 모델링		작업 전	0%	2021-11-21	2021-11-23	3
	2.4 DB 설계			작업 전	0%	2021-11-23	2021-11-30	8
		2.4.1 테이블 설계		작업 전	0%	2021-11-23	2021-11-27	5
		2.4.2 테이블 간 관계성 설계		작업 전	0%	2021-11-23	2021-11-27	5
		2.4.3 스키마 설계		작업 전	0%	2021-11-26	2021-11-30	5
3. 구현	3.1 디자인			작업 전	0%	2021-12-01	2021-12-14	14
		3.1.1 UI/UX 디자인		작업 전	0%	2021-12-01	2021-12-14	14
	3.2 퍼블리싱			작업 전	0%	2021-12-08	2021-12-21	14
		3.2.1 마크업 개발		작업 전	0%	2021-12-08	2021-12-14	7
		3.2.2 마크업 UI/UX 디자인 적용		작업 전	0%	2021-12-15	2021-12-21	7
	3.3 개발			작업 전	0%	2021-12-22	2022-04-03	103
		3.3.1 웹 컴포넌트, 페이지 개발		작업 전	0%	2021-12-22	2022-02-15	56
		3.3.2 컴포넌트 간 상태관리		작업 전	0%	2021-12-22	2022-02-15	56
		3.3.3 DB 연동 및 API 개발		작업 전	0%	2021-12-22	2022-01-24	35
		3.3.4 클라우드 환경 구축		작업 전	0%	2022-01-25	2022-02-13	21
		3.3.5 사용자 정보 암호화 로직 구축		작업 전	0%	2022-02-14	2022-02-27	14
		3.3.6 클라이언트, 서버 연결		작업 전	0%	2021-12-22	2022-01-24	35
		3.3.7 분산 처리 시스템 개발		작업 전	0%	2022-02-28	2022-04-03	35
4. 테스트	4.1 내부검수			작업 전	0%	2022-04-04	2022-05-08	35
		4.1.1 테스트용 데이터 구축		작업 전	0%	2022-04-04	2022-04-17	14
		4.1.2 CI/CD 구축		작업 전	0%	2022-04-04	2022-04-24	21
		4.1.3 유닛 테스트		작업 전	0%	2022-04-18	2022-04-24	7
		4.1.4 통합 테스트		작업 전	0%	2022-04-25	2022-05-01	7
		4.1.5 시스템 테스트		작업 전	0%	2022-05-02	2022-05-08	7
	4.2 외부검수			작업 전	0%	2022-05-09	2022-06-21	45
		4.2.1 알파 테스트		작업 전	0%	2022-05-09	2022-05-15	7
		4.2.2 베타 테스트		작업 전	0%	2022-05-16	2022-06-14	30
		4.2.3 인수 테스트		작업 전	0%	2022-06-15	2022-06-21	7
5. 배포 및 운영	5.1 배포			작업 전	0%	2022-04-03	~	기간없음
		5.1.1 Dockerize		작업 전	0%	2022-04-03	2022-04-09	7
		5.1.2 웹사이트 및 앱스토어 배포		작업 전	0%	2022-06-15	~	기간없음
	5.2 운영			작업 전	0%	2022-06-15	~	기간없음
		5.1.1 사용자 불편사항, 오류 수집		작업 전	0%	2022-06-15	~	기간없음
		5.1.2 오류 및 불편사항 개선		작업 전	0%	2022-06-15	~	기간없음
		5.1.3 사용자 정보 암호화		작업 전	0%	2022-06-15	~	기간없음
		5.1.4 리그레션 테스트		작업 전	0%	2022-06-15	~	기간없음

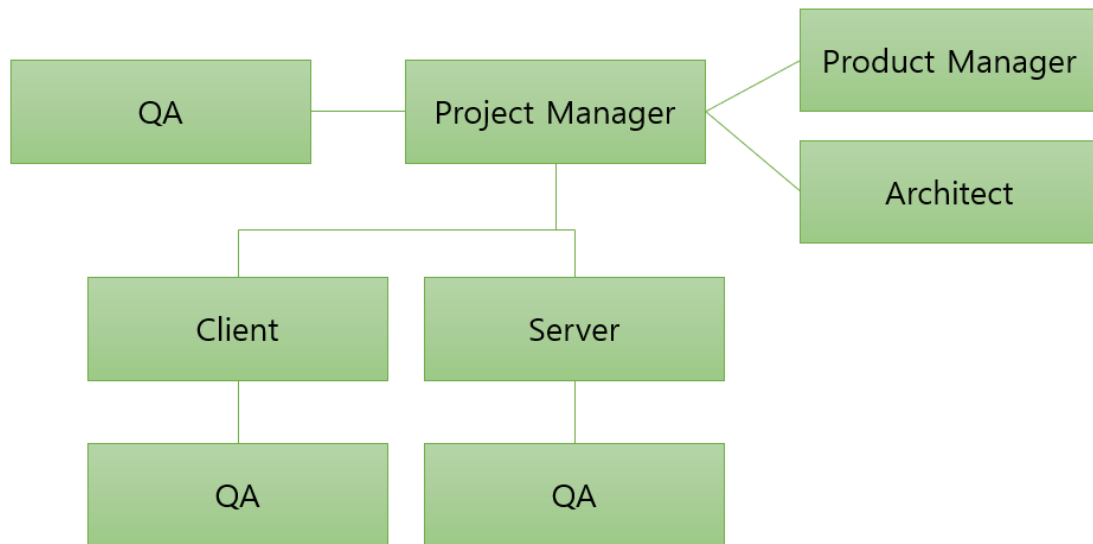


## 2.3 Schedule (Gantt Chart)

[illegible][illegible]

## 3.Organization

### 3.1 Team structure



### 3.2 Role and Responsibility

Role	Responsibility	Name	Report to
Project Manager	박성진	박성진	-
Product Manager	-	-	-
Client	김동용	김동용	-
Server	박성진, 이동우	박성진, 이동우	-
Client_QA	김동용	김동용	-
Server_QA	박성진, 이동우	박성진, 이동우	-
Entire_QA	-	-	-
Architect	-	-	-

## 4. Technical Management

### 4.1 Change and Configuration management

시스템 및 서비스에 대한 최신의 형상을 꾸준히 관리하고 업데이트하기에는 많은 노력이 듭니다. 형상 관리와 관련된 아래와 같은 자동화 도구를 사용합니다.

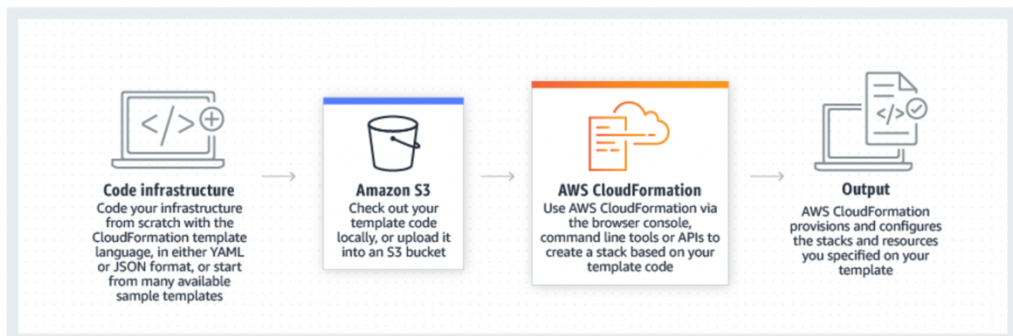
#### 1. 구성 관리 도구 - Chef



Chef는 물리적 서버, 가상 머신 및 클라우드에서 머신 설정을 처리하기 위한 구성 관리 도구입니다. 개발자와 IT 운영 전문가가 협력하여 IT 인프라에 애플리케이션을 배포할 수 있도록 도와줍니다.

#### 2. 코드로서의 인프라 도구 - CloudFormation

AWS CloudFormation을 사용하면 인프라를 코드로 처리하여 AWS 및 서드 파티 리소스를 모델링, 프로비저닝 및 관리할 수 있습니다.



#### 3. 파이프라인 도구 - Jenkins



[웹사이트 방문](#)

Jenkins는 Java로 작성된 오픈 소스 자동화 서버입니다. 개발자가 CI/CD 환경을 설정할 수 있도록 소프트웨어 프로젝트를 지속적으로 빌드 및 테스트하는 데 사용됩니다. 또한 Subversion, Git, Mercurial 및 Maven과 같은 버전 제어 도구를 지원합니다.

Jenkins는 풍부한 플러그인 및 통합 에코시스템으로 문서화되어 있으며 확장성이 뛰어납니다. 이러한 이유로 많은 사용자 지정이 필요한 대규모 프로젝트에서 사용할 때 탁월합니다.

Jenkins 자체는 무료이지만 주의, 업데이트 및 유지 관리가 필요한 서버에서 실행되어야 합니다.

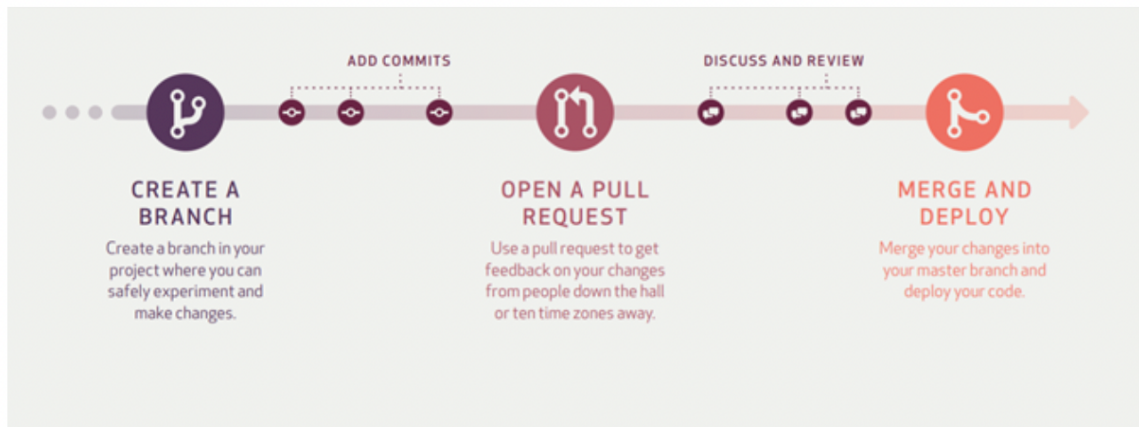
#### 4. 소스 제어 도구 - Github



GitHub는 소프트웨어 개발자를 위한 웹 기반 버전 제어 및 협업 플랫폼입니다. Git의 코드 리포지토리, 협업 도구, Wiki 및 프로젝트용 작업 관리 도구에 대한 웹 인터페이스를 제공하여 소셜 코딩을 용이하게 합니다.

추가적으로 Github 내에서 브랜칭 전략으로 아래와 같은 흐름의 GitHub Flow를 사용합니다.

#### Github-flow



코드 컨벤션 및 Github 메세지 컨벤션은 아래와 같이 정의합니다.

1. 변수명: Camel case

2. 클래스명: Pascal case

3. 브랜치명: ke-bab

4. 깃 허브 메세지 컨벤션

ADD : 코드에 추가사항이 생겼을 때 쓴다

ex) git commit -m "[ADD] #(이슈번호) - add login page"

FIX : 버그 또는 오류 해결 시에 사용한다.

DEL : 기존 코드를 삭제했을 시에 쓴다

RFR (REFACTOR) : 코드 리팩토링 시에 사용한다.

CHO (CHORE) : 잡일 예를 들어서 파일 구조 변경, 파일 이동할 때 쓴다

DOC : ReadMe 등등 문서 작업 시에 쓴다

## 4.2 Technology management

체크 메이트에서 활용하는 기술들은 아래와 같습니다.

1. QR code Generator and reading
2. OCR (Optical Character Recognition)
3. 부정 출석 방지를 위한 클라이언트와 서버간 유효성 검사
4. 부정 출석 방지를 위한 철저한 본인 인증
5. Docker 기반의 Cloud 환경 배포
6. React, Redux, Styled-component, AntD, Material-UI, Bootstrap, yup 등과 같은 FrontEnd 기술
7. Node, Express, Java, Spring, MySQL, Sequelize, Docker, Naver SENS 등과 같은 Backend 기술

# 5.Quality control

## 5.1 Review method

CheckList 를 사용합니다.

Defect type/event Occurrence	Dates							Total
	Sunday	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	
Description 1		9		8		3		20
Description 2		1		4				5
Description 3							10	10
Description 4			3		6			9
Description 5						4		8
Description 6				6			1	7
Description 7		3	3					6
Description 8	2			2				4
Description 9	5							5
Total	0	13	6	20	11	4	0	74

## 5.2 Review periodic

	정기 회의	비정기 회의
시간	매주 목요일 오후 10 시	필요 시에 진행
토의 내용	프로젝트 진행사항, 추가 이슈사항 논의	이슈 중심으로 토의
참석자	전원	이슈 관련자

## 5.3 Other quality control techniques

Pareto Chart	확률 밀도와 분배 정도를 한 눈에 볼 수 있는 차트
The Cause and Effect Diagram	잠재적인, 현재 존재하고 있는 문제들의 원인을 파악하기 위해 사용하는 diagram
Histrogram	어떤 이벤트의 빈도에 대해서 한 눈에 볼 수 있는 diagram
Scatter Diagram	어떤 두 가지 parameter 의 관계를 볼 수 있는 diagram
Graphs	process 의 시간 순서에 따른 변화를 확인할 수 있는 diagram

## 6.Development Environment

### 6.1 Required software and spec

Software Environment	VS Code, MySQL, IntelliJ
----------------------	--------------------------

### 6.2 Hardware spec

없습니다.

### 6.3 Space and security

Cloud	AWS, GCP, Naver cloud
-------	-----------------------

# 7.Deliverables

## 7.1 Define the documents

### 5. Deliverable Description and Requirements

[Provide a brief overview of the purpose of the deliverable and how it fits within the overall completion of the project (including the deliverable's objectives and scope). Reference the name, version, and date of any document applicable to this deliverable. List the specific requirements for this deliverable from the procurement, Statement of Work, and/or contract. Provide the specific acceptance criteria for the deliverable. This information may be provided in text format or in the table format as demonstrated below.]

Deliverable: <sup>4.2</sup>	<Deliverable Name> <sup>4.2</sup>	
Deliverable Description: <sup>4.2</sup>		
Deliverable Reference: <sup>4.2</sup>	<i>[This may include a Schedule IV-B, statute, rule, contract, Statement of Work, etc.]<sup>4.2</sup></i>	
Deliverable Requirements: <sup>4.2</sup>		
Format: <sup>4.2</sup>	<i>[Indicate the format of the document and any associated diagrams, spreadsheets (e.g., MS Word, MS Visio, MS Project, Adobe PDF, etc. Specify any specific format requirements such as text, point size, indentation, line spacing, or layout requirements such as page size, margin size, one/two sided, binding, etc. Also indicate any specific submission requirements (hard copy, email, certified mail, etc.)]<sup>4.2</sup></i>	
	<sup>4.2</sup> Requirement <sup>4.2</sup>	Acceptance Criteria <sup>4.2</sup>
Required Outline of Deliverable: <sup>4.2</sup>	<i>[Title Page]<sup>4.2</sup></i>	<i>[Title page will include:<sup>4.2</sup></i> <ul style="list-style-type: none"><li><i>• Project Name<sup>4.2</sup></i></li><li><i>• Deliverable Number<sup>4.2</sup></i></li><li><i>• Deliverable Name<sup>4.2</sup></i></li><li><i>• Date Submitted]<sup>4.2</sup></i></li></ul>
	<i>[Table of Contents]<sup>4.2</sup></i>	<i>[The Table of Contents will identify the document's main sections and corresponding page numbers and hyperlink directly to those pages]<sup>4.2</sup></i>
	<i>[Glossary of Terms / Abbreviations]<sup>4.2</sup></i>	<i>[The Glossary of Terms / Abbreviations will list the acronyms, technical terms, and abbreviations used in the deliverable and provide their corresponding meanings.]<sup>4.2</sup></i>
	<i>[Introduction]<sup>4.2</sup></i>	<i>[The introduction will include:<sup>4.2</sup></i> <ul style="list-style-type: none"><li><i>• Executive Summary<sup>4.2</sup></i></li><li><i>• Business Objectives]<sup>4.2</sup></i></li></ul>
	<i>[Specific Content]<sup>4.2</sup></i>	<sup>4.2</sup>
	<i>[Specific Content]<sup>4.2</sup></i>	<sup>4.2</sup>
	<i>[Specific Content]<sup>4.2</sup></i>	<sup>4.2</sup>
	<i>[Next Steps]<sup>4.2</sup></i>	<sup>4.2</sup>
	<i>[Reference Materials]<sup>4.2</sup></i>	<i>[The Reference Materials section will provide a listing of supporting documents, if any, for the deliverable. This will include a table with these columns: Document Name, Format, Purpose or Description of the Document.]<sup>4.2</sup></i>
	<i>[Signature and Acceptance Page]<sup>4.2</sup></i>	<i>[A Signature and Acceptance page will be included for signatures of submitter and approval authorities.]<sup>4.2</sup></i>

## 7.2 Date and destination

디자인: 2021-12-01 ~ 2021-12-14

퍼블리싱: 2021-12-08 ~ 2021-12-21

개발: 2021-12-22 ~ 2022-04-03

배포: 웹사이트 배포, 앱스토어 및 구글 플레이 스토어



# 8.Others

## 8.1 Considerable issues

### 개인정보 보호법

판례 연혁 위임행정규칙 규제 생활법령

#### 개인정보 보호법

[시행 2020. 8. 5.] [법률 제16930호, 2020. 2. 4., 일부개정]

개인정보보호위원회(개인정보보호정책과), 02-2100-3043

#### 제1장 총칙

**제1조(목적)** 이 법은 개인정보의 처리 및 보호에 관한 사항을 정함으로써 개인의 자유와 권리를 보호하고, 나아가 개인의 존엄과 가치를 구현함을 목적으로 한다. <개정 2014. 3. 24.>

**제2조(정의)** 이 법에서 사용하는 용어의 뜻은 다음과 같다. <개정 2014. 3. 24., 2020. 2. 4.>

보

1. "개인정보"란 살아 있는 개인에 관한 정보로서 다음 각 목의 어느 하나에 해당하는 정보를 말한다.
  - 가. 성명, 주민등록번호 및 영상 등을 통하여 개인을 알아볼 수 있는 정보
  - 나. 해당 정보만으로는 특정 개인을 알아볼 수 없더라도 다른 정보와 쉽게 결합하여 알아볼 수 있는 정보. 이 경우 쉽게 결합할 수 있는지 여부는 다른 정보의 입수 가능성 등 개인을 알아보는 데 소요되는 시간, 비용, 기술 등을 합리적으로 고려하여야 한다.
  - 다. 가목 또는 나목을 제1호의2에 따라 가명처리함으로써 원래의 상태로 복원하기 위한 추가 정보의 사용·결합 없이는 특정 개인을 알아볼 수 없는 정보(이하 "가명정보"라 한다)
- 1의2. "가명처리"란 개인정보의 일부를 삭제하거나 일부 또는 전부를 대체하는 등의 방법으로 추가 정보가 없는 특정 개인을 알아볼 수 없도록 처리하는 것을 말한다.
2. "처리"란 개인정보의 수집, 생성, 연계, 연동, 기록, 저장, 보유, 가공, 편집, 검색, 출력, 정정(訂正), 복구, 이용, 제공, 공개, 파기(破棄), 그 밖에 이와 유사한 행위를 말한다.
3. "정보주체"란 처리되는 정보에 의하여 알아볼 수 있는 사람으로서 그 정보의 주체가 되는 사람을 말한다.
4. "개인정보파일"이란 개인정보를 쉽게 검색할 수 있도록 일정한 규칙에 따라 체계적으로 배열하거나 구성한 개인정보의 집합물(集合物)을 말한다.
5. "개인정보처리자"란 업무를 목적으로 개인정보파일을 운영하기 위하여 스스로 또는 다른 사람을 통하여 개인정보를 처리하는 공공기관, 법인, 단체 및 개인 등을 말한다.
6. "공공기관"이란 다음 각 목의 기관을 말한다.
  - 가. 국회, 법원, 헌법재판소, 중앙선거관리위원회의 행정사무를 처리하는 기관, 중앙행정기관(대통령 소속 기관과 국무총리 소속 기관을 포함한다) 및 그 소속 기관, 지방자치단체
  - 나. 그 밖의 국가기관 및 공공단체 중 **대통령령**으로 정하는 기관
7. "영상정보처리기기"란 일정한 공간에 지속적으로 설치되어 사람 또는 사물의 영상 등을 촬영하거나 이를 유·무선망을 통하여 전송하는 장치로서 **대통령령**으로 정하는 장치를 말한다.
8. "과학적 연구"란 기술의 개발과 실증, 기초연구, 응용연구 및 민간 투자 연구 등 과학적 방법을 적용하는 연구를 말한다.

## 9.References and Appendix

- \* (전자출결 시스템의 문제점과 해결방안에 대한 연구 - 사용자 인식을 중심으로 -, 이재학, 이희화, 한국디지털정책학회, 2019, 41 - 49)
- \* (스마트기기를 이용한 상호협력기반 전자출결 시스템, 정필성, 조양현, 한국융합학회, 2021, 19 - 27)
- \* <https://www.plutora.com/blog/configuration-management>
- \* <https://eehoeskrap.tistory.com/14>
- \* <https://blog.naver.com/teamlegend/30023388172>
- \* <https://make-some-wave.tistory.com/entry/wbs>
- \* <https://www.brighthubpm.com/certification/72854-a-roundup-of-quality-control-tools-and-techniques/>
- \* <https://blog.asiaqualityfocus.com/qc-checklist-importers/>
- \* [https://mpra.ub.uni-muenchen.de/77941/1/MPRA\\_paper\\_77941.pdf](https://mpra.ub.uni-muenchen.de/77941/1/MPRA_paper_77941.pdf)
- \* <https://bcho.tistory.com/813>