

[ WebAssembly 기반 프로그래밍 교육 서비스 ]

---

## 요구사항 정의서

---

2024년 9월 13일

문서번호 : 2000-CoTA-Doc-001

소 속 : 충북대학교 소프트웨어학과

팀 명 : CoTA

팀 원 : 류정환, 김금영, 박용성

교 수 : 강재구 교수님

## 제/개정 이력

버전	날짜	작성자 성명	제/개정사항	비 고
v1.0	24.09.13	팀원 전원	초안 작성	

## <작성 가이드라인>

1. 주어진 양식에서 목차를 수정하지 않아야 합니다. 만약 임의 목차에 대하여 해당 사항이 존재하지 않는 경우는 해당 절의 목차를 삭제하지 않고, “해당 없음” 표시를 합니다.
2. 문서에는 반드시 페이지 번호를 기입합니다.
3. 주요 표와 그림은 해당 번호를 가져야 하며, 본문의 관련 내용에서 인용되어야 합니다.
4. 문서에 포함되는 모든 내용의 문구는 모호함이 없이 명확하게 표현되어야 합니다. ‘~ 할 수도 있다.’ 등의 표현은 사용하지 않도록 합니다.
5. 본문의 내용 작성에서 모든 문장은 주어, 동사, 목적어 등이 명확히 기술되어야 합니다. 단순히 ‘데이터를 입력 받아야 한다.’ 보다는 어떠한 데이터를 누가 입력받는지 명확하게 정의되어야 합니다.
6. 본문 내용은 짧게 끊어서 중요한 요점을 기술하는 개조식으로 작성하도록 합니다.
7. 폰트는 맑은고딕을 사용하고, 제목 1은 20pt, 제목 2는 14pt, 제목 3은 12pt, 제목 4는 11pt, 표준은 10pt, 줄간격은 130%로 작성합니다. 목차 페이지는 11pt에 줄간격 160%로 작성합니다.
8. 문서 작성을 위한 편집기는 공동 활용을 위하여 한글을 사용합니다.
9. 문서에 포함되는 그림은 고해상도의 이미지 형태로 삽입되어야 합니다.
10. 본 페이지는 삭제하고 작성합니다.

## 목 차

1. 서론	1
1.1 문서의 목적 및 범위	1
1.2 프로젝트 개요	1
1.3 용어 정의	1
1.3 참조 문서	1
2. 요구사항	2
2.1 기능적 요구사항	1
2.2 비기능적 요구사항	1
2.3 인터페이스 요구사항	1
3. 기타 제한사항	6
4. 참고문헌 및 부록	10

# 1. 서 론

## 1.1 문서 목적 및 범위

본 문서는 산학프로젝트 과목에서 사용자의 요구사항을 정의하고 분석하기 위한 문서로, 프로젝트 개발 계획부터 결과 도출, 관리까지, 사용자 요구사항의 전반적인 내용을 담고 있다.

## 1.2 프로젝트 개요

### 1.2.1 프로젝트 정의

웹 브라우저 상에서 작동하는 WebAssembly를 활용하여 C, Java와 같은 프로그래밍 언어작업 환경과 숙제 제출, 채점과 같은 프로그래밍 교육 환경을 복합적으로 제공함으로써 확장성이 높은 분산형 REPL 교육 프레임워크를 개발하고자 한다.

### 1.2.2 주요 기능 설명

#### 1) WebAssembly를 활용하여 C, Java 등의 프로그래밍 환경 제공

WebAssembly를 통해 브라우저에서 C, Java와 같은 언어를 실행할 수 있는 환경을 제공하며, 사용자는 설치 없이 다양한 운영체제에서 일관된 프로그래밍 경험을 할 수 있다.

#### 2) 숙제 제출 및 자동 채점 기능 제공

학생들은 웹에서 코드 과제를 제출하고, 시스템은 자동으로 채점하여 즉시 피드백을 제공한다. 채점자들은 이를 통해 학습자의 성취도를 평가하고 피드백을 줄 수 있다.

#### 3) 생성형 AI 언어 모델을 활용한 과제 힌트 제공

AI 모델을 활용해 학생들이 과제를 풀 때 힌트를 제공하며, 자연어 질문에도 응답할 수 있어 학습 과정에서의 어려움을 덜 수 있다.

#### 4) WebAssembly 환경의 신뢰성 확보

WebAssembly는 클라이언트 측에서 실행되므로 서버 성능에 의존하지 않고도 일관된 성능을 보장하며, 학습자는 빠르고 신뢰성 높은 코딩 환경을 경험할 수 있다.

## 1.3 용어 정의

본 문서의 이해를 돕기 위해 사용된 모든 용어 및 약어를 설명하고 정의합니다.

용어	설명

## 1.4 참조 문서

해당사항 없음.

# 2. 요구사항

## 2.1 기능적 요구사항

### 2-1-1) 사용자 관리 기능 - UF

#### UF1. 사용자 계정 관리

##### UF1.1. 회원가입 승인

UF1.1.1. 관리자는 신규 회원가입 요청을 승인 또는 거부할 수 있어야 한다.

UF1.1.2. 관리자는 사용자 정보를 조회하고 수정할 수 있어야 한다.

##### UF1.2. 사용자 권한 관리

UF1.2.1. 관리자는 사용자 계정을 활성화 또는 비활성화할 수 있어야 한다.

UF1.2.2. 관리자는 사용자 권한(일반 사용자, 관리자 등)을 부여하거나 변경할 수 있어야 한다.

##### UF1.3. 사용자 계정 삭제

UF1.3.1. 관리자는 필요에 따라 사용자 계정을 삭제할 수 있어야 한다.

### 2-1-2) 시스템 모니터링 및 유지보수 기능 - SMF

#### SMF1. 시스템 모니터링

### SMF1.1. 실시간 모니터링

SMF1.1.1. 관리자는 서버의 실시간 상태(트래픽, CPU 사용량, 메모리 사용량 등)를 확인할 수 있어야 한다.

### SMF1.2. 로그 관리

SMF1.2.1. 시스템은 사용자 활동 및 시스템 오류 로그를 기록하고 저장해야 한다.

SMF1.2.2. 관리자는 로그를 조회, 검색, 분석할 수 있어야 한다.

## SMF2. 시스템 유지보수

### SMF2.1. 시스템 업데이트

SMF2.1.1. 관리자는 시스템의 업데이트를 계획하고 실행할 수 있어야 한다.

SMF2.1.2. 시스템은 업데이트 시 서비스 중단을 최소화해야 한다.

### SMF2.2. 백업 및 복구

SMF2.2.1. 시스템은 정기적으로 데이터 백업을 수행해야 한다.

SMF2.2.2. 관리자는 필요 시 백업 데이터를 이용하여 시스템을 복구할 수 있어야 한다.

## 2-1-3) 보안 관리 기능 - SF

### SF1. 보안 설정 관리

#### SF1.1. 접근 제어

SF1.1.1. 관리자는 시스템의 접근 권한을 설정하고 관리할 수 있어야 한다.

SF1.1.2. 관리자는 비정상적인 접근 시도를 감지하고 대응할 수 있어야 한다.

#### SF1.2. 데이터 보안

SF1.2.1. 시스템은 사용자 개인정보를 암호화하여 저장해야 한다.

SF1.2.2. 시스템은 데이터 전송 시 SSL/TLS 프로토콜을 사용해야 한다.

## 2-1-4) 시스템 성능 및 안정성 관리 기능 - PMF

### PMF1. 성능 모니터링

#### PMF1.1. 자원 사용량 모니터링

PMF1.1.1. 관리자는 시스템의 CPU, 메모리, 네트워크 사용량 등을 모니터링할 수 있어야 한다.

#### PMF1.2. 성능 이슈 감지

PMF1.2.1. 시스템은 성능 저하 발생 시 관리자에게 알림을 제공해야 한다.

## PMF2. 안정성 관리

### PMF2.1. 장애 대응

PMF2.1.1. 시스템은 장애 발생 시 자동으로 복구 절차를 수행해야 한다.

PMF2.1.2. 관리자는 장애 원인을 분석하고 재발 방지 대책을 수립할 수 있어야 한다.

## 2-1-5) 클래스 생성 및 관리 기능 - CF

### CF1. 클래스 생성 및 설정

#### CF1.1 클래스 생성

- CF1.1.1 사용자는 클래스 이름, 설명, 공개 여부를 설정하여 클래스를 생성할 수 있어야 한다.
- CF1.1.2 사용자는 클래스에 참가자를 링크로 초대하거나 특정 이메일 주소로 초대할 수 있어야 한다.

#### CF1.2 클래스 관리

- CF1.2.1 생성자는 등록된 참가자 목록을 확인하고, 참가자의 이름, 이메일 등을 조회할 수 있어야 한다.
- CF1.2.2 생성자는 참가자를 수동으로 추가하거나 참가자의 등록을 승인 또는 거부할 수 있어야 한다.

## 2-1-6) 문제 출제 기능 - PF

### PF1. 문제 작성 및 설정

- PF1.1 출제자는 문제 설명, 입출력 예, 테스트 케이스를 제공하여 문제를 작성할 수 있어야 한다.
- PF1.2 테스트 케이스는 정답 코드를 제공할 경우 자동으로 생성될 수 있어야 한다.
- PF1.3 출제자는 문제에서 사용할 프로그래밍 언어를 추가하거나 제거할 수 있어야 한다.
- PF1.4 출제자는 과제 제출 기한을 설정할 수 있어야 한다.

## 2-1-7) 문제 배포 및 관리 기능 - DMF

### DMF1. 문제 배포 및 분류

- DMF1.1 출제자는 특정 클래스의 참가자에게 문제를 배포할 수 있어야 한다.
- DMF1.2 출제자는 문제를 카테고리 또는 태그로 분류하여 관리하고 검색할 수 있어야 한다.

## 2-1-8) 문제 통계 및 피드백 기능 - SF

### SF1. 문제 통계 조회

- SF1.1 출제자는 문제 해결 여부, 참가자의 성공률 등을 조회할 수 있어야 한다.



**SF2. 피드백 제공**

- SF2.1 출제자는 제출된 코드에 대해 참가자에게 피드백을 제공할 수 있어야 한다.

**2-1-9) 다양한 환경 제공 기능 - EF****EF1. 프로그래밍 언어의 확장 가능성**

- EF1.1 사용자는 C, Java, Python, JavaScript 등 다양한 프로그래밍 언어로 문제를 풀 수 있어야 한다.

**EF2. 온라인/오프라인 연속성**

- EF2.1 사용자는 오프라인에서도 코드 실행 및 디버깅이 가능해야 하며, 인터넷 연결 없이도 학습을 지속할 수 있어야 한다.

**2-1-10) 사용자 맞춤형 학습 환경 제공 기능 - PLF****PLF1. 코드 힌트 및 문제 설명**

- PLF1.1 사용자는 AI가 제공하는 코드 힌트나 문제 해결 전략을 요청할 수 있어야 한다.
- PLF1.2 사용자는 학습 중 문제가 쉽게 이해될 수 있도록 상세한 문제 설명을 받을 수 있어야 한다.

**PLF2. 맞춤형 단계적 학습 관리**

- PLF2.1 사용자는 난이도가 점진적으로 상승하는 문제를 제공받아 자신의 학습 상태를 관리할 수 있어야 한다.

**2-1-11) 과제 관리 및 추적 기능 - TMF****TMF1. 과제 제출 이력 및 성과 분석**

- TMF1.1 사용자는 자신이 제출한 과제의 이력을 확인하고, 성과에 대한 통계를 제공받을 수 있어야 한다.

**TMF2. 성과 기반 피드백**

- TMF2.1 사용자는 특정 문제나 언어에서의 성과를 분석하여 자기 개선에 활용할 수 있어야 한다.

## 2-1-12) 팀 협업 및 공유 기능 - CCF

### CCF1. 실시간 협업 기능

- CCF1.1 사용자는 팀 프로젝트를 수행할 때 실시간 코드 편집 및 리뷰 기능을 통해 다른 팀원과 협업할 수 있어야 한다.

## 2-1-13) 보안 기능 - SF

### SF1. 과제 및 코드 보안

- SF1.1 사용자는 자신의 과제 및 코드에 대해 보안 조치를 기대할 수 있어야 한다.

## 2-1-14) 다양한 문제 유형 제공 기능 - PT

### PT1. 문제 유형

- PT1.1 사용자는 알고리즘 문제 등 다양한 문제 유형을 제공받아야 한다.
- PT1.2 사용자는 문제에 대한 실시간 피드백을 받을 수 있어야 한다.

## 2-1-15) 도전 과제 및 성과 시스템 - CS

### CS1. 도전 과제 및 보상

- CS1.1 사용자는 학습 목표를 달성할 때 점수 체계를 이용하여 점수를 부여받아야 한다.
- CS1.2 점수 체계를 토대로 순위에 따라 배지 및 보상을 받을 수 있어야 한다.

## 2-1-16) 자동 저장 및 복구 기능 - AR

### AR1. 코드 자동 저장 및 복구

- AR1.1 사용자는 작업 중인 코드가 실시간으로 저장되며, 예기치 않은 오류 발생 시 자동 복구 기능을

이용할 수 있어야 한다.

## 2-1-17) 다양한 문제 풀이 환경 제공 기능 - PECF

### PECF1. 문제 풀이 환경 커스터마이징

- PECF1.1 사용자는 문제 풀이 시 편리하게 사용할 수 있도록 코드 편집기의 테마(밝은 모드, 어두운 모드 등)를 설정할 수 있어야 한다.

### PECF2. 문제 유형에 따른 인터페이스

- PECF2.1 사용자는 문제의 유형(알고리즘 문제, 디버깅 문제 등)에 따라 다른 인터페이스를 제공받아, 문제 풀이 시 적절한 도구를 활용할 수 있어야 한다.

## 2-1-18) 실습 모드 및 시험 모드 제공 기능 - PSM

### PSM1. 실습 모드

- PSM1.1 사용자는 다양한 실습 문제를 풀면서 코드를 실행해 보고 디버깅할 수 있는 자유로운 환경에서 학습할 수 있어야 한다.

### PSM2. 시험 모드

- PSM2.1 사용자는 시간 제한이 있는 시험 모드를 통해 자신의 실력을 테스트할 수 있어야 하며, 즉각적인 피드백을 받을 수 있어야 한다.

## 2-1-19) 학습 기록 및 이력 관리 기능 - LTM

### LTM1. 학습 이력 조회

- LTM1.1 사용자는 자신의 학습 이력(문제 풀이 기록, 제출 기록, 성과)을 조회하고 분석할 수 있어야 한다.

### LTM2. 학습 기록 연동

- LTM2.1 사용자의 학습 기록은 GitHub 및 코딩 이력 관리 시스템과 연동될 수 있어야 한다.

## 2.2 비기능적 요구사항

### 2-2-1) 운영 환경에 대한 요구사항 - OR

- OR1. 시스템은 다양한 웹 브라우저 및 운영체제에서 안정적으로 동작해야 한다.  
OR2. 서버 인프라는 확장성이 있어야 하며, 트래픽 증가에 대응할 수 있어야 한다.

### 2-2-2) 성능 요구사항 - PR

- PR1. 시스템은 동시 접속자 1,000명 이상을 안정적으로 지원해야 한다.  
PR2. 시스템의 응답 시간은 2초 이내여야 한다.

### 2-2-3) 보안 요구사항 - SR

- SR1. 시스템은 관리자 계정에 대해 이중 인증을 제공해야 한다.  
SR2. 시스템은 모든 데이터 전송을 암호화해야 한다.  
SR3. 시스템은 정기적인 보안 점검을 수행해야 한다.

### 2-2-4) 문화 및 정책적 요구사항 - CR

- CR1. 시스템은 관리자에게 제공되는 모든 정보를 한국어로 출력해야 한다.  
CR2. 시스템 운영은 개인정보 보호법 및 관련 법규를 준수해야 한다.

### 2-2-5) 품질 요구사항 - QR

- QR1. 시스템은 연중무휴 99.9% 이상의 가용성을 유지해야 한다.  
QR2. 장애 발생 시 1시간 이내에 복구되어야 한다.  
QR3. 시스템은 관리자 인터페이스의 사용성을 높이기 위해 직관적인 UI/UX를 제공해야 한다.

## 2.3 인터페이스 요구사항

### 2-3-1) 관리자 인터페이스 요구사항 - UIF

- UIF1. 시스템은 웹 기반의 관리자 전용 인터페이스를 제공해야 한다.
- UIF2. 관리자 인터페이스는 직관적이고 사용하기 쉬워야 한다.
- UIF3. 관리자 인터페이스는 모바일 기기에서도 사용할 수 있어야 한다.

### 2-3-2) 내부 시스템 인터페이스 요구사항 - IIF

- IIF1. 시스템은 모듈화되어 유지보수가 용이해야 한다.
- IIF2. 시스템의 각 모듈은 표준화된 API를 통해 통신해야 한다.
- IIF3. 로그 및 모니터링 데이터는 중앙에서 관리되어야 한다.

### 2-3-3) 외부 시스템 인터페이스 요구사항 - EIF

- EIF1. 시스템은 외부 모니터링 도구와 연동할 수 있어야 한다.
- EIF2. 시스템은 알림 시스템(SMS, 이메일 등)과 연동하여 관리자에게 알림을 제공할 수 있어야 한다.

### 2-3-4) 클래스 생성 인터페이스 요구사항 - CIF

- CIF1. 시스템은 사용자가 클래스를 생성할 수 있도록 생성 버튼을 제공해야 한다.
- CIF2. 사용자는 클래스 생성 시 정보를 입력할 수 있는 폼을 제공받아야 한다.

### 2-3-5) 출제자 인터페이스 요구사항 - QIF

- QIF1. 출제자는 문제 설명, 제한 사항, 입출력 예를 작성할 수 있는 텍스트 편집기를 제공받아야 한다.
- QIF2. 코드 블록과 마크다운 형식을 지원하여 다양한 서식을 사용할 수 있어야 한다.
- QIF3. 출제자는 제출 가능 언어를 선택할 수 있는 인터페이스를 제공받아야 한다.
- QIF4. 작성한 문제의 미리보기를 통해 참가자에게 어떻게 보일지 확인할 수 있어야 한다.

### 2-3-6) 테스트 케이스 및 채점 설정 인터페이스 요구사항 - TIF

- TIF1. 출제자는 테스트 케이스를 추가할 수 있는 직관적인 입력 폼을 제공받아야 한다.
- TIF2. 출제자는 시간과 메모리 제한을 설정할 수 있어야 하며, 이를 테스트 케이스별로 조정할 수 있어야 한다.
- TIF3. 출제자는 정답 코드를 작성할 수 있는 에디터와 언어 선택 기능을 제공받아야 한다.
- TIF4. 정답 코드 실행 결과를 실시간으로 확인할 수 있어야 하며 자동 생성된 테스트 케이스가 통과하는지 여부를 확인할 수 있어야 한다.

## 3. 기타 요구사항

### 3-1) 법적 및 규제 준수 요구사항 - LR

#### LR1. 법적 준수

LR1.1. 시스템은 개인정보 보호법 등 관련 법률을 준수해야 한다.

LR1.2. 저작권 및 라이선스 조건을 준수하여 소프트웨어 및 콘텐츠를 사용해야 한다.

### 3-2) 문서화 및 교육 요구사항 - DR

#### DR1. 시스템 문서화

DR1.1. 시스템의 아키텍처, 설계, 운영 절차에 대한 기술 문서를 작성해야 한다.

DR1.2. 관리자 매뉴얼과 사용자 가이드를 제공해야 한다.

#### DR2. 교육 및 지원

DR2.1. 관리자는 시스템 운영 및 유지보수에 대한 교육을 받아야 한다.

DR2.2. 시스템 사용자를 위한 지원 채널(FAQ, 고객 지원 등)을 마련해야 한다.

### 3-3) 서비스 수준 협약(SLA) 요구사항 - SLA

#### SLA1. 서비스 가용성

SLA1.1. 시스템은 연중무휴 99.9% 이상의 가용성을 유지해야 한다.

SLA1.2. 장애 발생 시 복구 시간을 1시간 이내로 유지해야 한다.

#### SLA2. 지원 대응

SLA2.1. 관리자는 사용자 문의에 대해 24시간 이내에 응답해야 한다.

SLA2.2. 긴급 상황 발생 시 즉각적인 대응 절차를 마련해야 한다.

## 4. 참고문헌 및 부록

본 문서와 관련하여 기타 참고할 문서나 문서 작성에 참고한 문헌이 있다면 일반적인 참고문헌 작성 방법에 따라 목록화 합니다.

또한 부록으로 첨부되어야 할 개발 환경 관련 하드웨어 스펙이나 소프트웨어 도구에 대한 정보 등이 있다면 부록으로 넣을 수 있습니다.

<끝>

<https://www.cisp.or.kr/wp-content/uploads/2021/03/2.%EA%B3%B5%EA%B3%B5SW%EC%82%AC%EC%97%85-%EC%A0%9C%EC%95%88%EC%9A%94%EC%B2%AD%EC%84%9C-%EC%9E%91%EC%84%B1%EC%9D%84-%EC%9C%84%ED%95%9C-%EC%9A%94%EA%B5%AC%EC%82%AC%ED%95%AD-%EA%B0%80%EC%9D%B4%EB%93%9C-20210219.pdf>