

문제해결능력 향상을 위한
코드 채점 프로그램

소프트웨어 요구사항 명세서

2013년 10월 08일

인하대학교 컴퓨터정보공학부
캡스톤설계
교수 신병석

조장 12071092 전경원
조원 12070968 박진한
12071121 조성준

목 차

1. 서론 (Introduction)	3
가. 목적 (Purpose)	3
나. 범위 (scope)	3
다. 용어 및 약어 정리 (Definitions, Acronyms, and Abbreviations)	3
라. 개요 (Overview)	4
2. 종합기술 (Overall Description)	4
3. 상세 요구사항 (Specific Requirements)	7
가. 외적 인터페이스 요구사항 (External Interface Requirements)	7
나. 기능적 요구사항 (Functional Requirements)	10
다. 성능 요구사항 (Performance Requirements)	11
라. 논리적 데이터베이스 (Logical Databases)	11
마. 설계 제약조건 (Design Constraints)	13
바. 표준안 준수 (Standards Compliance)	14
사. 소프트웨어 시스템 특성	14
아. 기능별 구현 우선순위 (Functional Development Priority)	14
자. 기타 요구사항 (Other Requirements)	15
4. 추가 정보 (Supportiong Information)	15
가. 소프트웨어 요구사항 명세서	15
나. 문서 이력	16

1. 서론 (Introduction)

가. 목적 (Purpose)

본 문서는 문제해결능력 향상을 위한 코드 채점 프로그램 요구사항 명세서 (SRS)이다.

본 문서는 인하대학교 컴퓨터 정보 공학부 4학년 과목 "캡스톤 설계" 001분반 제 xx팀에서 설계 및 구현하기 위한 것으로 이를 위한 요구 사항을 정리, 분석하고, 기재된 내용을 바탕으로 시스템을 설계 및 구현한다.

본 문서는 인하대학교 "캡스톤 설계" 001분반 제 xx팀을 주요 독자로 한다. 부가적으로 담당 교수 신병석 교수님이 주요 독자가 될 수 있다.

추후 본 시스템을 상품으로 개발할 경우, 이와 관련된 모든 업체 직원들이 추가적인 독자가 될 수 있다.

나. 범위 (scope)

본 프로젝트의 개발 범위는 다음과 같다.

- 인증
학번 인증 또는 이메일 인증을 통한 회원 정보 저장 및 관리
처음 실행시 가입한다.
- 로그인/로그아웃
프로그램 이용을 위한 로그인/로그아웃 처리
- 코드채점 프로그램 가동
코드 채점 프로그램의 가동을 위한 접속
- 코드 채점 프로그램의 실행
메타 데이터 생성 및 관리
작성 코드 데이터 분류 및 정렬
- 그 외의 서비스
코드 채점 결과 알림에 대한 사용자들의 편의적인 댓글 기능
사용자 친화적 인터페이스

다. 용어 및 약어 정리 (Definitions, Acronyms, and Abbreviations)

- SRS : Software Requirement Specification

- GUI : Graphic User Interface
- DB : Database
- 모듈 : 프로그램의 자율적이고 교환 가능한 구성 단위

라. 개요 (Overview)

본 문서의 구성은 다음과 같다. 2장에서는 코드 채점 프로그램의 전반적인 개요를 알아본다. 이를 위해 결과물의 향후 전망과 결과물의 세부 기능, 사용자의 특성, 제약조건, 가정과 의존성 등의 각 절로 나누어 살펴본다. 3장에서는 코드 채점 프로그램의 상세 요구사항을 알아본다. 외적 인터페이스 요구사항과 기능적 요구사항, 성능 요구사항, 설계 제약조건, 소프트웨어 시스템 특성, 기타 요구사항을 각 절에서 살펴본다. 기능적 요구사항에는 데이터 흐름도와 프로세스 기술, 데이터 구조 명세, 데이터 사전을 포함한다.

2. 종합기술 (Overall Description)

가. 프로젝트결과물전망 (Product Perspective)

1) 시스템인터페이스 (System Interfaces)

시스템은 Windows가 구동 가능한 컴퓨터의 단일 소프트웨어로 구성되며, 프로그램에서의 입력은 마우스와 정보 입력을 위한 키보드로 구성된다. 사용자 정보 입력과 출력을 위한 각각의 화면으로 구성되어 있고, 해당 버튼을 클릭함으로써 이동할 수 있다. 사용자 정보와 통보내용을 서버로 전송하기 위해 TCP/IP 프로토콜을 사용한다.

2) 사용자인터페이스 (User Interfaces)

소프트웨어 인터페이스는 로그인 화면과 사용자의 코드입력, 사용자 정보 입력, 정보 및 주의 사항 확인, 그리고 결과 화면으로 구성되며 각 화면은 알맞은 버튼을 이용해서 이동한다.

3) 하드웨어인터페이스 (Hardware Interfaces)

CPU: Intel Pentium3 1GHz 이상

Memory: 128MB 이상

4) 소프트웨어인터페이스 (Software Interfaces)

OS: Windows7

개발 환경: Microsoft Visual Studio 2010

개발 언어: C#

5) 통신인터페이스 (Communications Interfaces)

Modem: 56kbps 이상

Inha University 코드 채점 프로그램

Cabstone Design - 001 - Team xx

6) 운영 (Operations)

라이선스

본 프로그램은 영구 라이선스와 1인 1PC 라이선스 정책으로 운영한다.

업데이트 및 배포

프로그램을 업데이트해야 할 경우 패치파일을 통해 업데이트를 수행할 수 있도록 한다.

나. 결과물의기능 (Product Functions)

문제해결능력 향상을 위한 코드 채점 프로그램의 기능은 다음과 같은 기능을 수행한다.

- (기능 1) 사용자 친화적 인터페이스 : 사용자가 파악하기 쉽게 가장 직관적이며, 기초적인 GUI를 바탕으로 한 인터페이스 환경을 제공한다.

- (기능 2) 다양한 피드백 기능 : 문제에 대한 컴파일 결과를 DB에 저장한 것을 바탕으로 채점 결과, 오답 목록 등을 보여주어 사용자에게 피드백을 제공할 수 있고, 제출된 코드에 대해 사용자들끼리 의견을 주고 받을 수 있게 함으로써 더욱 빈번한 피드백이 이루어지게 한다.

다. 사용자특성 (User Characteristics)

문제해결능력 향상을 위한 코드 채점 프로그램의 주 사용자는 프로그래밍 언어의 입문단계나 아직 전문적인 수준을 갖추지 못한 학생들이 될 것이다.

라. 제약조건 (Constraints)

본 장의 가절에서 언급한 내용을 바탕으로 시스템을 설계 및 구현한다. 그 이외의 사항들은 개발자가 선호하는 방향을 선택하여 설계 및 구현하되, 다음의 사항을 준수한다.

- 사용되는 기술은 이미 널리 성능이 입증된 것을 사용한다.
- 별도의 라이선스가 필요하거나 저작권료 지불이 필요한 기술 또는 소프트웨어의 사용을 피한다. (단, 유익한 기술 또는 소프트웨어로 본 시스템의 필수적으로 필요한 경우는 제외한다.)
- 시스템 전반의 성능 향상을 모색하는 방향으로 결정한다.
- 사용자에게 보다 친숙하고 편리한 방향으로 결정한다.
- 가급적 오픈 소스 소프트웨어를 사용한다.
- 시스템의 추후 확장성과 가용성을 고려한다.
- 프로그램 제작 시 소스코드의 최적화를 통하여 시스템 자원의 낭비를 예방한다.
- 소스코드 작성 시 추후의 유지보수를 고려하여 충분한 주석을 추가한다.

마. 가정과 의존성 (Assumptions and Dependencies)

본 문서의 모든 시스템은 Windows 운영체제를 기반으로 설계 및 구현한 것

을 가정하여 작성되었다. 따라서 모든 내용은 Windows 운영체제를 기준으로 작성되었으며, 타 운영체제에서 동일하게 적용되지 않을 수 있다. 이후 타 운영체제로 버전을 고려 할 수 있다.

3. 상세 요구사항 (Specific Requirements)

가. 외적 인터페이스 요구사항 (External Interface Requirements)

메인 화면

Do Coding!!

회원가입

로그인

[문제풀기](#)

[질문과 답변](#)

[자유게시판](#)

• 이번 주 가장 많이 푼 문제

[가장 많이 푼 문제](#)[도전하기](#)

• 이번 주 가장 많이 시도 된 문제

[가장 많이 시도 된 문제](#)[도전하기](#)

금주의 코딩 왕 : 문제를 가장 많이 푼 사람 ID

금주의 노력 왕 : 문제풀이 시도가 가장 많은 사람 ID

- 7 -

회원 관련 GUI

회원 가입

User ID	
E-mail	
비밀번호	
비밀번호 확인	
주 사용 언어	<input type="checkbox"/> C 언어 <input type="checkbox"/> JAVA

가 입

취 소

문제 관련 GUI

▷ 문제 출제하기

문제ID	시간제한	메모리제한	제출 횟수	정답횟수
출제자		출처		분류



관리자 로그인
중

- ▷ 문제 출제하기
- ▷ 문제 목록보기
- ▷ 문제 풀기
- ▷ 채점 기록보기

제 출 GUI

제출 된 답안

문제 정보

문제	제출자	제출 시간	채점 결과

소스 코드

댓글

ID ☆☆☆ comment

채점 관련 GUI

▷ 채점 기록보기

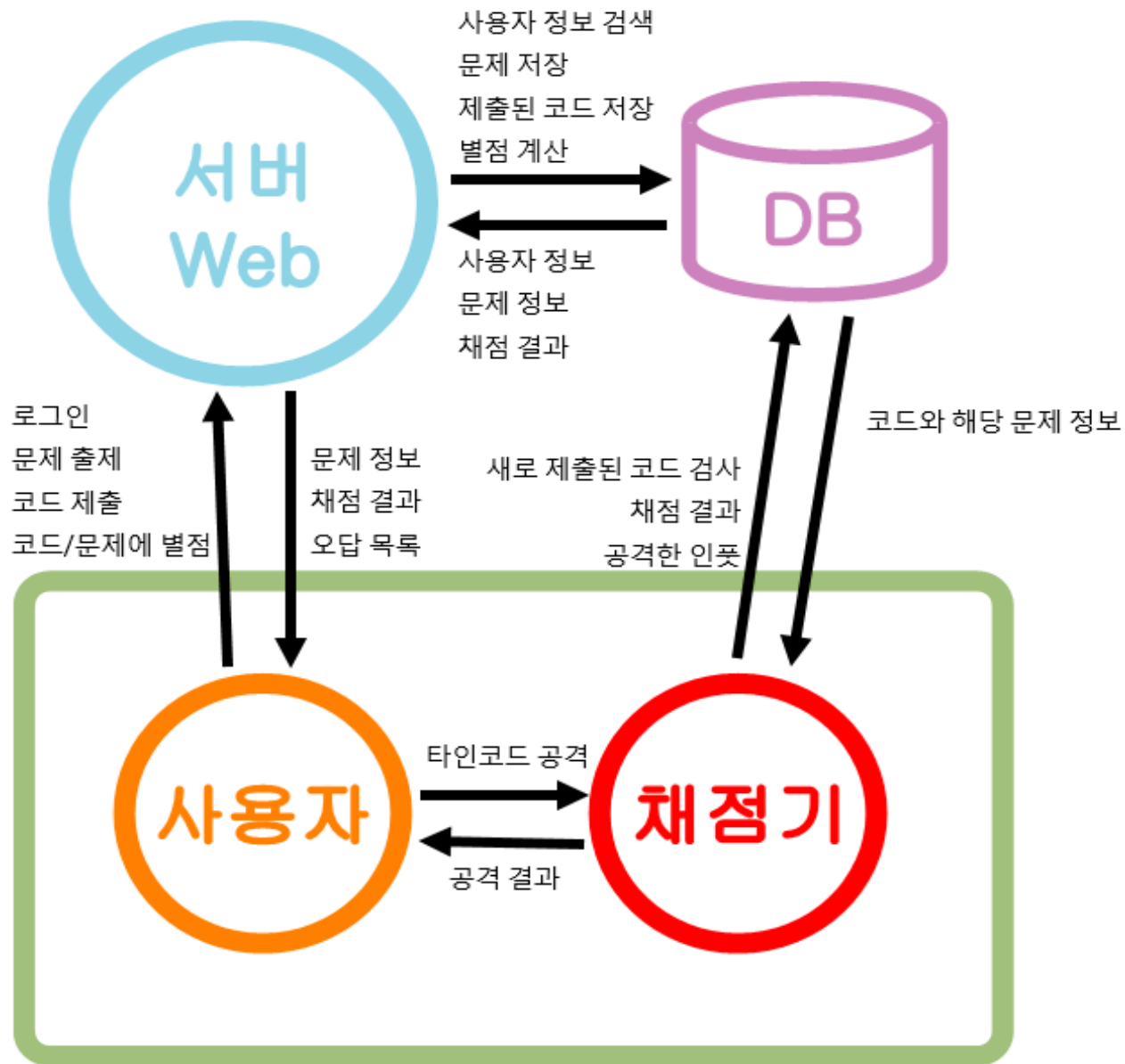


관리자 로그인
중

- ▷ 문제 출제하기
- ▷ 문제 목록보기
- ▷ 문제 풀기
- ▷ 채점 기록보기

No.	문제	제출자	언어	길이	결과	수행시간	제출시간

나. 기능적 요구사항 (Functional Requirements)



제목	설명
로그인	사용자가 입력한 사용자 정보(ID, PW)를 서버로 보낸다.
사용자 정보 검색	사용자의 로그인 정보에 대한 질의.
문제 출제/ 문제저장	문제 정보를 DB에 저장한다.
코드 제출 제출된 코드 저장	사용자의 코드를 DB에 저장한다.
코드/문제에 별점 별점 계산	코드에 별점을 부과한다. 문제에 별점을 부과한다.
새로 제출된 코드	채점기는 일정 시간을 주기로 로그인된 사용자에게 대한 새로운 코드를 검사

검사	한다.
채점 (채점 결과)	채점기는 코드를 컴파일 후 테스트 하여 결과를 DB에 저장한다.
오답 목록	채점시 오답이 나올 경우 어떤 인풋에 대해서 오답인지 사용자에게 보낸다.
타인 코드 공격	정답을 맞은 타인의 코드를 보고 오답이 예상되는 코드를 넣어본다. 채점기는 이를 테스트 한다.
공격 결과	테스트 한 결과를 사용자에게 알려준다.
공격한 인풋	공격에 성공했을 경우 해당 인풋을 DB에 저장한다.

다. 성능 요구사항 (Performance Requirements)

본 시스템은 보편적으로 공공기관에서(학교 등) 사용되는 PC의 사양에서 안정적인 구동되어야 한다. 서버는 apache, mysql 을 실행시키기에 적합한 환경이어야 하며, 클라이언트 PC 는 C++ 과 JAVA 가 컴파일이 가능한 환경이어야 한다. (사용자의 언어선택에 따라 둘 중 하나만 요구될 수 있다.) 채점을 위해서 안정적인 통신환경이 보장되어야 한다. 문제의 인풋이 매우 클 수 있으므로 100 MB 이상의 메모리를 확보할 수 있어야 한다.

라. 논리적 데이터베이스 (Logical Databases)

1) 테이블명 : user

설 명 : 사용자 정보

필드명	타입	설명	Key
userNo	int	사용자 일련번호	PK
userId	varchar	사용자 아이디	
password	char	사용자 비밀번호	
email	varchar	사용자 이메일	
language	int	사용자 주력 사용 언어	

2) 테이블명 : problem

설 명 : 문제 정보

필드명	타입	설명	Key
problemNo	int	문제 일련번호	PK
author	int	출제자	FK (user - userNo)
timeLimit	int	시간제한	
title	varchar	제목	
description	text	문제 설명	

inputFormat	text	입력 서식	
outputFormat	text	출력 서식	
rightCode	varchar	정답 코드	
rightCodeLang	int	정답 코드 언어	
starPoint	int	별점 합계	
starCnt	int	별점 준 사람 수	

3) 테이블명 : testData

설 명 : 문제 입출력 데이터 정보

필드명	타입	설명	Key
dataNo	int	데이터 일련번호	PK
problemNo	int	문제 일련번호	FK (problem - problemNo)
isSample	boolean	예제로 보여줄 것인지?	
input	text	테스트 인풋	
output	text	테스트 아웃풋	

4) 테이블명 : submit

설 명 : 사용자 코드 제출 / 채점

필드명	타입	설명	Key
submitNo	int	제출 일련 번호	PK
userNo	int	제출 사용자 일련번호	FK (user - userNo)
problemNo	int	문제 일련번호	FK (problem - problemNo)
sourceCode	text	소스코드	
sourceLang	int	소스코드 언어	
submitTime	datetime	제출 시간	
status	int	상태 (대기중, 컴파일중, 테스트중, 채점 끝)	
result	int	결과 (정답, 오답, 시간초과, 컴파일에러, 런타임에러)	
runningTime	int	실행시간	
judgeLog	text	채점 로그	
starPoint	int	별점 합계	
starCnt	int	별점 준 사람 수	

5) 테이블명 : submitComment

설 명 : 제출된 코드에 대한 댓글

필드명	타입	설명	Key
commentNo	int	댓글 번호	PK
submitNo	int	제출 일련번호	FK (problem - problemNo)

userNo	int	사용자 일련번호	FK (user - userNo)
commentTime	datetime	댓글 쓴 시간	
commentStr	varchar	댓글 내용	

6) 테이블명 : problemComment

설 명 : 문제에 대한 댓글

필드명	타입	설명	Key
commentNo	int	댓글 번호	PK
problemNo	int	문제 일련번호	FK (problem - problemNo)
userNo	int	사용자 일련번호	FK (user - userNo)
commentTime	datetime	댓글 쓴 시간	
commentStr	varchar	댓글 내용	

7) 테이블명 : submitStar

설 명 : 제출된 코드에 대한 별점 중복 부여 방지 테이블

필드명	타입	설명	Key
submitNo	int	제출 일련번호	PK, FK (problem - problemNo)
userNo	int	사용자 일련번호	PK, FK (user - userNo)
point	int	별점 점수	

8) 테이블명 : problemStar

설 명 : 문제에 대한 별점 중복 부여 방지 테이블

필드명	타입	설명	Key
problemNo	int	문제 일련번호	PK, FK (problem - problemNo)
userNo	int	사용자 일련번호	PK, FK (user - userNo)
point	int	별점 점수	

마. 설계 제약조건 (Design Constraints)

본 장의 가절에서 언급한 내용을 바탕으로 시스템을 설계 및 구현한다. 그 외의 사항들은 개발자가 선호하는 방향을 선택하여 설계 및 구현하되, 단 다음의 사항을 준수한다. 클라이언트 PC 는 사용자가 원하는 언어에 대한 컴파일 이 가능하다고 가정한다. C, C++ 사용자 PC 엔 Visual Studio 2010 버전

이상이 설치되어 있다고 가정한다. Java 사용자 PC 엔 JDK 최신버전이 설치되어 있다고 가정한다. 사용되는 기술은 이미 널리 성능이 입증된 것을 사용한다. 별도의 라이선스가 필요하거나 저작권료 지불이 필요한 기술 또는 소프트웨어의 사용을 피한다. 사용자에게 보다 친숙하고 편리한 방향으로 결정한다. 시스템 전반의 성능 향상을 모색하는 방향으로 결정한다. 가급적 오픈 소스 소프트웨어를 사용한다. 시스템의 추후 확장성과 가용성을 고려한다. 프로그램 제작 시 소스코드의 최적화를 통하여 시스템의 자원의 낭비를 예방한다. 소스코드 작성 시 추후의 유지보수를 고려하여 충분한 주석을 추가한다.

바. 표준안 준수 (Standards Compliance)

시스템 내의 모든 프로그램은 관용적인 프로그래밍 기법을 따른다. 프로그램 내의 함수 명과 변수 명은 헝가리안 표기법을 따르며, 그 외의 사항은 ISO, IEEE의 표준안을 따른다.

사. 소프트웨어 시스템 특성

1) 신뢰도 (Reliability)

사용자의 정보 입력에 맞는 데이터 제공

2) 가용성 (Availability)

사용자의 요청에 언제든지 허용 응답시간 내에 요청

3) 보안 (Security)

사용자의 개인 정보의 암호화 및 데이터의 암호화

4) 유지가능성 (Maintainability)

관리자의 서버 관리 및 유지 보수 용이성 극대화

5) 호환성 (Portability)

Windows 개별을 이용하여 타 시스템으로의 호환성과 이식성 향상 제고

아. 기능별 구현 우선순위 (Functional Development Priority)

1) 우선 구현 사항

- 클라이언트

사용자 친화적 인터페이스를 구현
입력 받은 사용자 정보를 객체화 하는 모듈
객체화된 사용자 정보를 서버로 송신

- 서버

객체화된 사용자 정보를 서버에서 수신하여 그룹화하는 알고리즘
그룹화된 사용자들에게 개별적으로 결과를 송신하는 모듈

- DB

객체화된 사용자 정보를 저장할 수 있는 모듈
문제 정보를 저장할 수 있는 모듈

- 채점기

코드 컴파일 후 테스트한 결과를 DB에 저장하는 모듈
맞은 정답을 제출한 타인의 코드를 넣어보고 오답이 예상되는 코드를 넣는 능력

2) 선택 구현 사항

일정 시간을 주기로 로그인된 사용자에게 관한 새로운 코드 검사 실행 모듈
사용자의 코드에 댓글이 생성될 때 이를 알리는 모듈

자. 기타 요구사항 (Other Requirements)

없음

4. 추가 정보 (Supporting Information)

가. 소프트웨어 요구사항 명세서

본 소프트웨어 요구사항 명세서는 IEEE의 권고안 (IEEE Recommended Practice for Software Requirements Specifications)을 따라 작성되었음.

나. 문서 이력

날 짜	버 전	설 명	작 성 자
2013.10.3.목	Ver.1.10	기획안 아이디어 점검, 보충	박진한, 전경원, 조성준
2013.10.4.금	Ver.1.11	스타일 작업 및 개요	전경원
2013.10.5.토	Ver.1.12	SRS 1장 초안작성	전경원, 박진한
2013.10.6.일	Ver.1.13	SRS 2장, 3장 초안작성, SRS 내용 검토 및 추가	전경원, 박진한, 조성준
2013.10.7.월	Ver.1.14	SRS 3장 내용 추가, 내용 검토	전경원, 박진한, 조성준
2013.10.8.화	Ver.1.15	통합, 서식정리	조성준