



국민대학교  
전자정보통신대학  
컴퓨터공학부

**CONFIDENTIALITY/SECURITY WARNING**


이 문서에 포함되어 있는 정보는 국민대학교 전자정보통신대학 컴퓨터공학부 및 컴퓨터공학부 개 설 교과목 캡스톤 디자인 I 수강 학생 중 프로젝트 “SSD Test 자동화 시스템”을 수행하는 팀 “1g's dormitory”의 팀원들의 자산입니다. 국민대학교 컴퓨터공학부 및 팀 “1g's dormitory”의 팀원들의 서면 허락없이 사용되거나, 재가공 될 수 없습니다.

# 캡스톤 디자인 I

## 종합설계 프로젝트

프로젝트 명	SSD Test 자동화 시스템
팀 명	1g's dormitory
문서 제목	계획서

Version	1.1
Date	2018-04-11

 <b>국민대학교</b> <b>컴퓨터공학부</b> <b>캡스톤 디자인 I</b>	<b>계획서</b>		
	<b>프로젝트 명</b>	SSD Test 자동화 시스템	
	<b>팀 명</b>	1g's dormitory	
	Confidential Restricted	Version 1.1	2018-04-11

---

<b>이름</b>	<b>강원지</b>
-----------	------------

Page 1 of 9

All rights are reserved. Reproduction in whole or in parts is prohibited without the written consent of the copyright owner.


## 문서 정보 / 수정 내역

수정날짜	대표수정 자	Revision	추가/수정 항 목	내 용
2018-03-07	강원지	1.0	최초 작성	최초 작성
2018-04-11	강원지	1.1	계획서 수정	계획서 내용 보완

 <b>국민대학교</b> <b>컴퓨터공학부</b> <b>캡스톤 디자인 I</b>	<b>계획서</b>		
	<b>프로젝트 명</b>	SSD Test 자동화 시스템	
	<b>팀 명</b>	1g's dormitory	
	Confidential Restricted	Version 1.1	2018-04-11

## 목 차

1 개요 .....	4
1.1 프로젝트 개요 .....	4
1.2 추진 배경 및 필요성 .....	4
2 개발 목표 및 내용 .....	5
2.1 목표 .....	5
2.2 연구/개발 내용 .....	5
2.3 개발 결과 .....	6
2.3.1 개발 결과 시나리오 .....	6
2.3.2 시스템 설계도 .....	오류! 책갈피가 정의되어 있지 않습니다.
2.3.3 비기능 요구사항 .....	7
2.4 기대효과 및 활용방안 .....	8
3 배경 기술 .....	8
3.1 기술적 요구사항 .....	8
3.2 현실적 제한 요소 및 그 해결 방안 .....	9
4 개발 일정 및 자원 관리 .....	9
4.1 개발 일정 .....	9
4.2 일정 별 주요 산출물 .....	10

 <b>국민대학교</b> <b>컴퓨터공학부</b> <b>캡스톤 디자인 I</b>	<b>계획서</b>		
	<b>프로젝트 명</b>	SSD Test 자동화 시스템	
	<b>팀 명</b>	1g's dormitory	
	Confidential Restricted	Version 1.1	2018-04-11

# 1 개요

## 1.1 프로젝트 개요

최근 SSD의 몸값이 치솟고 있다. 전력 소모가 적고, 빠른 정보 처리 속도를 가진다는 강점이 기존의 하드디스크를 대체하는 저장장치로써 SSD가 대두되는 이유로 보인다. SSD 개발을 위해서는 개발된 SSD의 성능을 테스트하는 작업이 필수적이다. 그러한 이유로 SSD 시장이 활성화됨에 따라서 SSD의 성능을 테스트하는 tool의 중요성이 높아지고 있다. 기존에 상용화되어 있는 Test Tool들은 한번에 적은 수의 SSD만을 테스트할 수 있게 되어있고 또한, 원격에서 테스트를 할 수 없게 되어있는 것이 일반적이다. FADU에서는 이러한 한계점을 보완하기 위해서 사용자가 한 번의 동작으로 수십 대의 SSD를 테스트할 수 있고 직접 tool이 있는 곳에서만 테스트할 수 있는 것이 아닌 원하는 곳 어디에서든지 웹을 통해서 테스트할 수 있는 시스템을 만들기로 하였다.

## 1.2 추진 배경 및 필요성

### 1.1.1 기존에 개발된 시스템의 한계점

현재 FADU에서 사용하고 있는 SSD 테스트 시스템은 한 번에 테스트할 수 있는 SSD의 개수가 한정되어 있다. 이런 상황에서 대량의 SSD를 테스트하기 위해서는 매우 많은 시간과 인력이 필요하다. 또한, 테스트를 회사 내부망에서만 실행이 가능한 상황이며, 테스트의 결과를 필요한 정보만 찾아서 열람하기가 불편한 형태로 결과가 저장되는 시스템으로 이루어져 있다.

### 1.1.2 개발할 시스템의 필요성

위에서 언급한 한계점들을 보완하기 위해서 대량 SSD의 테스트를 한 번에 실행할 수 있고 외부에서도 테스트를 control할 수 있는 시스템이 필요하다. 그리고 테스트의 결과를 한눈에 파악하기 쉽도록 정리하여 보여 줄 필요가 있다. 더불어 firmware update 기능, 테스트 이력관리 기능 등이 있다면 더 체계적으로 테스트를 관리할 수 있을 것이다.

 <b>국민대학교</b> <b>컴퓨터공학부</b> <b>캡스톤 디자인 I</b>	<b>계획서</b>		
	<b>프로젝트 명</b>	SSD Test 자동화 시스템	
	<b>팀 명</b>	1g's dormitory	
	Confidential Restricted	Version 1.1	2018-04-11

## 2 개발 목표 및 내용

### 2.1 목표

기존에 회사에서 사용하고 있는 SSD Test System 의 한계점을 보완할 수 있는 새로운 Test System 을 개발한다. 새로운 test system 개발에는 아래와 같은 기능을 가지는 것을 목표로 한다

- Remote test 기능 : test 를 회사 내에서 뿐만 아니라 어디에서도 실행하고 결과를 볼 수 있도록 한다.
- Multiple test 기능 : 적은 수의 SSD 만을 테스트할 수 있는 것이 아니라 다수의 SSD 를 테스트 할 수 있도록 한다.
- Visualizing 기능 : 테스트 결과를 간결하게 요약하고 중요한 결과들을 그래프로 볼 수 있도록 하여 SSD 분석을 쉽게 할 수 있도록 한다.
- Firmware update 기능 : 기존에는 각 test PC 에서 firmware update 를 해야 했으나 새로운 system 을 통해서는 한 번에 여러 대의 SSD 를 firmware update 를 할 수 있도록 한다.

### 2.2 연구/개발 내용

- Remote test 기능 : remote test 를 가능하게 하도록 사용자가 웹에서 DB 에 쌓여 있는 test script 중 실행할 script 를 선택하면 front-end 에서 test script 의 id 를 host server 로 전달하고 host server 는 그 id 와 매치되는 script 를 test server 로 전달하여 test 를 실행한다.
- Multiple test 기능 : multiple test 가 가능하기 위해서 회사에서는 한 PC 에 4 개의 SSD 를 제어할 수 있는 하드웨어를 제작 중이며, 사용자는 많은 SSD 중에서 테스트를 실행할 SSD 의 id 를 선택하여 host server 로 전송하고 host server 에서는 그 id 에 맞는 SSD 가 있는 랙의 test server 로 script 를 전송하여 test 를 실행한다.
- Firmware update 기능 : 현재 firmware update 를 하기 위해서 freedom studio 라는 tool 을 사용하고 있으며 원격에서 firmware 를 update 하기 위해서 test server 와 연동하여 실행할 것이다.
- Visualizing 기능 : 테스트 결과는 json 형식으로 host server 에 전달된다. Host server 에서는 그 테스트 결과를 DB 에 저장한 후에 front-end 로 전달한다. Front-

 <b>국민대학교</b> <b>컴퓨터공학부</b> <b>캡스톤 디자인 I</b>	<b>계획서</b>		
	<b>프로젝트 명</b>	SSD Test 자동화 시스템	
	<b>팀 명</b>	1g's dormitory	
	Confidential Restricted	Version 1.1	2018-04-11

end에서는 전달된 결과를 가지고 테이블을 생성하거나 그래프를 생성하여 웹화면에 띄운다.

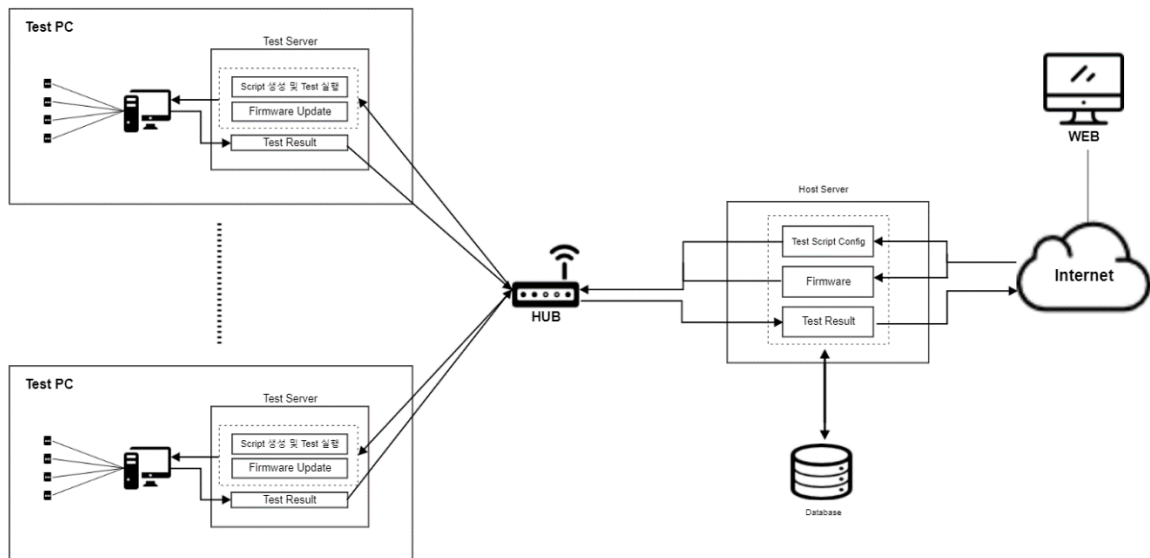
## 2.3 개발 결과

### 2.3.1 개발 결과 시나리오

- 1) 사용자는 Test Script 를 선택할 수 있다.
  - Preset script 중에서 선택할 수 있다.
  - Test 에 필요한 개별 속성들을 직접 선택하거나 입력하여 script 를 생성할 수 있다.
  - Test script 여러 개를 한 번에 실행할 시나리오를 선택할 수 있다.
- 2) 사용자는 Test 를 실행할 수 있다.
  - 원격에서 웹을 통해 test 를 실행할 수 있다.
  - Test 의 종류를 선택할 수 있다. (fio, tNVMme 등 ..)
- 3) 사용자는 test 결과를 열람할 수 있다
  - Test 결과가 요약된 내용과 결과를 가지고 생성한 그래프가 포함된 엑셀 파일을 다운로드할 수 있다.
- 4) SSD 의 firmware 가 변경되었을 시 웹을 통해서 firmware 를 업데이트할 수 있다.

 <b>국민대학교</b> <b>컴퓨터공학부</b> <b>캡스톤 디자인 I</b>	<b>계획서</b>		
	<b>프로젝트 명</b>	SSD Test 자동화 시스템	
	<b>팀 명</b>	1g's dormitory	
	Confidential Restricted	Version 1.1	2018-04-11

### 2.3.2 시스템 설계도



### 2.3.3 비기능 요구사항

#### 1. 속도

- SSD Test 를 할 때, 사용하는 리소스가 크기 때문에, Test 외의 performance 에 영향이 최소화 되어야 한다. 따라서 performance 에 영향이 최소화 되도록 가벼운 언어인 C 언어 등으로 서버를 구현하여 Overhead 를 최소화 하는 방안으로 작업한다.

#### 2. 안정성

- 동시에 여러 Test 장비를 가동했을 때, 그 메시지들을 처리하는 서버가 부하에 견딜 수 있어야 한다. 또한 서비스 중 서버가 다운되면 로그에 대한 신뢰도가 떨어진다. 따라서 PM2 등 프로세스 매니저를 도입한다.

 <b>국민대학교</b> <b>컴퓨터공학부</b> <b>캡스톤 디자인 I</b>	<b>계획서</b>		
	<b>프로젝트 명</b>	SSD Test 자동화 시스템	
	<b>팀 명</b>	1g's dormitory	
	Confidential Restricted	Version 1.1	2018-04-11

## 2.4 기대효과 및 활용방안

새로운 시스템을 사용하여 test 를 수행할 경우에 회사외부에서도 테스트를 실행 하고, 그 결 과를 열람할 수 있기 때문에 개발에 어떤 문제가 생겼을 때 파악을 하기에 편리하다. 또한 한번 의 동작으로 많은 SSD test 를 실행할 수 있기 때문에 인력 소모와 시간의 소모를 줄일 수 있다.

## 3 배경 기술

### 3.1 기술적 요구사항

프로젝트 개발환경

1) Client(SSD test) server

OS : Centos 6.9 server

Server : C 기반 자체 개발 서버

2) Host server

OS : Centos 6.7 server

Server : node.js(^v6.9.0 LTS)

Framework : express(^v4.0.0 LTS)

Database : Mongodb

Proxy : Nginx

3) Client

Browser : Chrome

Language : HTML + JavaScript(React.js)

프로젝트 버전관리

1) GitHub(public)

2) Slack



 <b>국민대학교</b> <b>컴퓨터공학부</b> <b>캡스톤 디자인 I</b>	<b>계획서</b>		
	<b>프로젝트 명</b>	SSD Test 자동화 시스템	
	<b>팀 명</b>	1g's dormitory	
	Confidential Restricted	Version 1.1	2018-04-11

## 3.2 현실적 제한 요소 및 그 해결 방안

### 1. 소프트웨어

- 기존에 FADU 에서 테스트할 때 사용하는 program 을 활용하여 시스템 개발을 진행하고 있으나 그 program 들 중 사내에서 자체개발한 program 을 license 문제로 사용을 하지 못한다.


### 2. 하드웨어

- PC 한 대로 SSD 여러 개를 연결하여 control 할 수 있는 하드웨어가 ( test server ) 있어야 하나 아직 하드웨어가 제작 중인 상황이어서 PC 한 대에 SSD 한 개를 연결하여 시스템 개발을 진행하고 있다.

## 4 개발 일정 및 자원 관리

### 4.1 개발 일정

항목	세부내용	3 월	4 월	5 월	6 월	비고
요구사항분석	요구 분석					
	SRS 작성					
관련분야연구	주요 기술 연구					
	관련 시스템 분석					
설계	시스템 설계					
구현	코딩 및 모듈 테스트					
테스트	시스템 테스트					

 <b>국민대학교</b> <b>컴퓨터공학부</b> <b>캡스톤 디자인 I</b>	<b>계획서</b>		
	<b>프로젝트 명</b>	SSD Test 자동화 시스템	
	<b>팀 명</b>	1g's dormitory	
	Confidential Restricted	Version 1.1	2018-04-11

## 4.2 일정 별 주요 산출물

마일스톤	개요	시작일	종료일
계획서 발표	SSD Test PC 및 Host server PC 정하기 개발 환경 완성 (개발 PC OS 설치) 산출물 : 1. 프로젝트 수행 계획서 2. 프로젝트 기능 일람표	~	2018-03-08
설계 완료	시스템 설계 완료 산출물 : 1. 시스템 설계 사양서	2018-03-09	2018-03-23
1 차 중간 보고	개발/연구내용 1~6 구현 완료 산출물 : 1. 프로젝트 1 차 중간 보고서 1 차분 구현 소스 코드	2018-03-24	2018-04-12
2 차 중간 보고	개발/연구내용 6~14 구현 완료 산출물 : 1. 프로젝트 2 차 중간 보고서 2 차분 구현 소스 코드	2018-04-13	2018-05-18
테스트	시스템 통합 테스트 산출물: 시스템 테스트 결과 보고서	2018-05-18	2018-05-29
최종 보고서	최종 보고		2018-06-01