

캡스톤 디자인 I 종합설계 프로젝트

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

Version	1.0
Date	2018-04-12

팀원	이진식 (조장)
	강원지
	권유상
	김주용
지도교수	황선태 교수님

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	SSD Test 자동화 시스템	
	팀 명	1g's dormitory	
	Confidential Restricted	Version 1.0	2018-04-10

CONFIDENTIALITY/SECURITY WARNING

이 문서에 포함되어 있는 정보는 국민대학교 전자정보통신대학 컴퓨터공학부 및 컴퓨터공학부 개설 교과목 캡스톤 디자인Ⅰ 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

문서 정보 / 수정 내역


Filename	중간보고서_산학분반_권유상.doc
원안작성자	권유상
수정작업자	

수정날짜	대표수정자	Revision	추가/수정 항목	내 용
2009-04-12	권유상	1.0	최초 작성	

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	SSD Test 자동화 시스템	
	팀 명	1g's dormitory	
	Confidential Restricted	Version 1.0	2018-04-10

목 차

1	프로젝트 목표	4
2	수행 내용 및 중간결과	5
2.1	계획서 상의 연구내용	5
2.2	수행내용	5
3	수정된 연구내용 및 추진 방향	8
3.1	수정사항	8
4	향후 추진계획	9
4.1	향후 계획의 세부 내용	9
5	고충 및 건의사항	10


 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	SSD Test 자동화 시스템	
	팀 명	1g's dormitory	
	Confidential Restricted	Version 1.0	2018-04-10

1 프로젝트 목표

이 프로젝트는 SSD Test Tool을 Web에서 제어하여 실행하고, 그 결과를 visualize 해서 보여주는 시스템을 만드는 것이다.

현재 여러 SSD Test Tool(fio, nbio, oakgate, sanblaze 등)이 존재하는데 이 Tool들을 실행하기 위해서는 테스트할 SSD가 장착되어 있는 PC(또는 장비)에서만 가능하고, 테스트에 소요되는 시간 또한 길다. 이 때문에 한 사람이 직접 PC로 가서 테스트를 실행하고 완료된 후 결과값을 받아 보기까지 기다려야 한다. 한 번의 테스트를 실행하기 위해서도 이러한 비용(시간과 사람의 노력)이 소비되는데, Fadu와 같이 SSD를 전문적으로 다루는 회사에서는 많은 양의 SSD를 테스트하기 위해 좀 더 편하게 할 수 있는 방법을 생각하게 되었다. 그 결과, 장착할 수 있는 SSD의 개수가 제한되어 있는 각각의 PC를 하나의 서버로 제어하여 각각 테스트를 실행하고, 결과 로그들을 Database에 저장하여 Web에서 사용자가 원하는 테스트 결과를 볼 수 있게 하는 시스템을 만들어 보려고 한다.

또한 원격으로 테스트 실행과 결과 로그들을 받아보는 작업이 완료된다면, 그 후에는 결과 로그를 활용하여 그래프나 테이블과 같이 사용자가 보기 편한 데이터로 만들어서 각각의 SSD에 대한 테스트 결과를 비교/분석 하기 쉽게 Web에 디자인 하고자 한다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	SSD Test 자동화 시스템	
	팀 명	1g's dormitory	
	Confidential Restricted	Version 1.0	2018-04-10

2 수행 내용 및 중간결과

2.1 계획서 상의 연구내용

테스트 자동화 시스템은 Web에서 SSD Test Tool을 제어하여 실행하고, 결과를 시간과 장소에 상관 없이 받아 볼 수 있는 시스템이다. 다양한 SSD Test Tool의 Test script 정보들을 Web에서 설정하여 중앙 서버(Host Server)로 전달하고 그 내용을 database에 저장한 후, SSD가 장착된 각 장비(Test Server)에 전송한다. Test script에 대한 정보를 넘겨받은 각 장비의 Test Server는 Test script를 만들어 실행하고 그 결과를 다시 중앙 서버로 전송한다. 결과값을 받은 중앙서버는 Database에 테스트 로그들을 저장하고 Web으로 전송하여 보여준다. Web에서는 서버에서 받은 데이터를 바탕으로 사용자가 비교/분석 하기 편한 형태로 가공하여 보여준다. 이러한 과정을 통해 인터넷이 연결된 모든 곳에서는 테스트를 실행하고 결과를 받아볼 수 있게 된다.


2.2 수행내용

1. Server

서버는 Web과 Database, 각각의 테스트 PC를 연결해 주는 중앙 서버(Host Server)와, 테스트할 SSD가 장착된 PC에서 중앙서버와 통신해주는 테스트 서버(Test Server)로 나뉜다.

a. Host Server (node.js의 express 사용)

- Test Server에 json형식의 데이터를 Http method인 post 요청을 통해 주고 받는다.
- Test Server에서 받은 테스트 결과 로그들을 json 형식으로 Web에 전달한다.
- Web에서 요청한 데이터나, Test Server로 보낼 데이터를 Database에서 불러와 json 방식으로 전달한다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	SSD Test 자동화 시스템	
	팀 명	1g's dormitory	
	Confidential Restricted	Version 1.0	2018-04-10

b. Test Server(node.js의 express 사용)

- Host Server에서 받은 Test script의 설정 값으로 Test script를 작성한다. Test script의 설정 값은 Database에 미리 정의해 놓은 스키마에 의해 저장된 값으로 json 형식으로 받아 script 파일을 생성한다. Test script에 들어가는 설정 값이 많다보니, 모든 설정 값들을 받아서 만드는 것이 아니라 일정부분 고정된 설정 값들은 정적으로 입력해 놓는다.
- 작성된 Test script를 사용하여 테스트를 실행한다. 테스트 실행은 명령어를 통해서 이루어 지며 node.js의 내장 모듈인 child process를 통해 실행하였다.
- 실행한 테스트에 대한 로그 값이 저장된 directory에서 로그 파일들을 json 형태로 만들어 Host Server로 보낸다. Test script에 들어가는 설정 값 중에 원하는 로그 파일과 로그파일이 저장될 위치를 설정할 수 있는 항목이 있다. 이 항목을 통해 로그 디렉토리를 만들어 그 안에 로그파일들을 저장하고 해당 디렉토리를 읽어서 안에 있는 모든 로그 파일들을 각각 json 형식의 데이터로 만들어 전송한다.
- 위 세 가지 과정은 순차적으로 진행되어야 하므로 동기식으로 구현 하였다.

2. Database (mongo db 사용)

- Test script에 들어가는 각 설정 값(config)들을 하나의 document에 저장한다.
- Test script의 설정 값을 저장한 document에는 id 값이 있는데 이 값을 통해 해당 script의 결과 로그 목록이 저장된 document와 연결 된다.
- 결과 document에는 여러 로그 항목에 대한 배열이 저장되어 있어 그 index에 따라 해당 로그의 값들이 저장되어 있다.
- Host Server와 연동해 Web에서 요청한 자료를 전달해 줄 수 있다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	SSD Test 자동화 시스템	
	팀 명	1g's dormitory	
	Confidential Restricted	Version 1.0	2018-04-10

3. Web (node.js의 react 사용)

- Host Server와 연동하여, 원하는 페이지를 띄울 수 있다.
- Database에 있는 Test script 목록을 불러와 테스트 가능한 스크립트들을 보여줄 수 있다.
- 저장된 Test script 목록에 있는 것들 중 하나를 선택하여 테스트를 실행할 수 있고, 테스트가 끝나면 그 결과값을 받아 볼 수 있다.


 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	SSD Test 자동화 시스템	
	팀 명	1g's dormitory	
	Confidential Restricted	Version 1.0	2018-04-10

3 수정된 연구내용 및 추진 방향

3.1 수정사항

- **Server**

서버 측을 모두 node.js의 express를 사용하여 구현 하였는데, 이를 python으로 수정할 것으로 예상된다. 여기서 “예상된다” 라고 하는 이유는 아직 회사측에서 정식으로 확정된 것이 아니기 때문이다. 현재 이 시스템의 작동을 위해 fio라는 Test tool만을 사용하고 있는데 나중에 다른 Test tool까지 시스템에 적용하려면 python으로 서버를 구성하는 편이 더 편하다는 것이 회사측 입장이다. 그 이유는 fio가 아닌 다른 Test tool에서 제공하는 api들이 모두 python으로 구성되어 있고 그것들을 서버에서 사용하려면 같은 python을 사용하는 것이 더 편하기 때문이다. 만약 바뀌게 된다면 일주일 정도를 투자하여 node.js 코드를 python으로 바꾸는 작업을 할 예정이다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	SSD Test 자동화 시스템	
	팀 명	1g's dormitory	
	Confidential Restricted	Version 1.0	2018-04-10

4 향후 추진계획

4.1 향후 계획의 세부 내용

1. 지금까지는 하나의 Test tool에 대해서만 자동화 작업을 진행하였는데 다른 Test tool들에 대해서도 같은 작업을 하도록 추가한다.
2. 테스트 결과를 Web에서 보여줄 때 그래프나 테이블 같은 방식으로, 사용자가 더 편하게 볼 수 있도록 구현한다.
3. Web에서 Test script에 있는 설정값들을 직접 바꾸거나 추가하여 script를 추가, 삭제, 변경할 수 있도록 한다.
4. 이 항목들 외의 추가적인 요구사항은 사용자에게 직접 받아서 추가한다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	SSD Test 자동화 시스템	
	팀 명	1g's dormitory	
	Confidential Restricted	Version 1.0	2018-04-10

5 고충 및 건의사항

아직까지는 없습니다.