

# System Test Countermeasure for Coffee Machine System

- System Test Result Review
- Solution

## Project Team

### Team 1

201211333 김영호

201211347 박성근

201211364 이경민

201211376 임제현

Latest update on:

**2016-12-04**

---

## Team Information

## Table of Contents

1	Introduction.....	3
1.1	Objectives.....	3
1.2	References.....	3
2	Test testing result .....	3
2.1	System testing result.....	3
3	System test analysis.....	4
3.1	Problems .....	4
4	Solutions .....	4
5	후기 .....	5

## 1 Introduction

### 1.1 Objectives

본 문서는 2016 년 건국대학교의 소프트웨어공학 개론 강의의 실습과제를 설명한다. 실습 과제는 coffee machine 을 소프트웨어로 구성된 가상의 시스템으로 구현하는 것이다.

### 1.2 References

[2016SE\_B][T1]STP 2.0

[2016SE\_B][T1]STR 1.0

[2016SE\_B][T1]SRA 4..0

T6\_TO\_T1\_STR.pdf

## 2 Test testing result

### 2.1 System testing result

Test case identifier	Input Specification	Output Specification	Pass / Fail
TEAM1_CM_001_001	Bean = 100 Water = 1000 Command Extract	Coffee Used powder data Spending time	Spending time Fail
TEAM1_CM_001_003	Coffee Powder = on Water = 1000 Command Extract	Coffee Used powder data Spending time	Spending time Fail
TEAM1_CM_002	Command Clean	Water – 500 Spending Time	Spending time Fail
TEAM1_CM_008	Exit	Exit	Exit Fail

### 3 System test analysis

#### 3.1 Problems

- (1) 커피를 추출, 분쇄, 커피머신 청소시 시간이 흐르지 않는다.
- (2) 항시 종료 버튼이 없다.
- (3) 대기중 시간이 변하지 않는다.

#### 4 Soutlions

- (1) 홈페이지에 있는 파일을 다운로드 받아 실행시 SRA에 명시된 사항과 동일한 시간이 흘러감. 이상없음
- (2) 종료 버튼은 [2016SE\_B][T1]SRA 4.0에 아래와 같이 명시되어 있다.

- Cygwin 타이핑을 위해 다음과 같은 명령어 리스트 출력

Display example (예시)	
0	커피 추출
1	청소 버튼
2	농도 설정 버튼
3	재료 보충 버튼
4	찌꺼기 제거 버튼
5	예약 버튼
6	온도 설정 버튼
9	커피 머신 종료 버튼

### 3.4 Other requirements

- 원두 분쇄에 필요한 시간: 5 초
- 커피 추출에 필요한 시간: 10 초
- 청소에 필요한 시간: 10 초
- 가열에 필요한 시간: 10 초
- 재료 보충은 즉시
- 각 작업의 진행 시간 동안 다른 작업은 수행 할 수 없다.
- 커피 추출 후 남은 커피 가루는 사용자에게 의해 직접 제거된다. (cygwin 의 타이핑으로 이를 대체한다.)
- Stop Button 에서 받아온 데이터를 통해 사용자가 원할 때 종료한다.(커피 추출, 청소 중에는 작업을 완료 후 종료하고, 예약이 있는 상태에서는 그 예약 행동을 완료하고 종료한다.)

어떤 작업을 실행중에 종료 버튼을 누르면 그 작업을 끝낸 뒤에 종료를 실행한다. 이상없음.

(3) 대기중 시간이 변하지 않는 것은 구현의 한계이므로 수정할 수 없다.

## 5 후기

(1) 김영호 : 지금 생각해보면 첫주차때가 제일 힘들었던 것 같다. 이제껏 설계는 하지않고 무조건 구현부터 했었는데 막상 설계를 하려고 하니 SA 문서를 어떻게 만들어야 할 지 몰랐다. 1주일동안 문서를 만들려고 팀원들과 매일 늦게까지 했다. 문서를 만들고 나니 그럴듯하게 보여서 잘 만든 것 같다고 생각을 했는데 아이고... 나중에 2주차를 하고, 3주차를 하고 생각이 바뀌고, 각 기능들을 더 상세하게 만들려다보니 문서가 계속 수정되어 힘들었다. 그래도 3주차 이후 부터는 앞주차 보다는 무난하게 진행했던 것 같다. 수업을 듣고 과제를 하던 도중 모르는 부분은 조교님께 질문을 했는데 이해가 잘 되게 답변해주셔서 수월하게 진행되었고 전반적으로 한학기동안 좋은 수업이었다.

(2) 이경민 : 한번도 생각해보지 않았던 부분을 배울 수 있었던 수업이었다. 지금까지 배운 내용중엔 매니저, 기술자에 대한 부분을 배우지 않았는데 프로그래밍에 있어 사람까지 생각해보는 좋은 수업이었다. 실습은 뭔가 비효율 적이라는 생각을 많이 한 동시에 별거 아닌것 같은 커피머신에 이만한 공을 들인다고 생각하니 대단하다고 느꼈다 비효율 적이라고 느낀 부분은 수업에서 내용을 배울 때마다 SRA를 바꿔야 했던점.. 그리고 초기에 개발을 고려하지 않아 바뀐점도 몇 개 있었다. 작은 프로젝트이니 이런 것 보단 다른 개

발 방법이 더 어울릴 거라고 생각도 했다. 각자 자신의 작업을 하고 합쳐져서 하나가 된 것을보니 뿌듯하긴 했다.

- (3) 박성근 : 이론시간엔 전반적인 소프트웨어 공학에 대한 설명을 들었고, 그것을 직접 해보는 것이 실습시간이었습니다. 처음에는 이론과 실습을 서로 연관 지을 수가 없어서 실습을 하면서도 이 과정이 왜 필요한지, 무엇을 해야 하는지 알 수 없었습니다. 학기가 끝나가는 지금 예서야 이때까지 해온 과정들이 연결되는 느낌이 듭니다. 처음에는 지난 년도에 한 것들을 보고 참조하면서 대강 이렇게 하면 되겠구나, 했었는데 교수님께서 기본 원리가 divide & conquer 라는 것을 설명 해주신 것을 듣고서 그제서야 처음 저희가 작작 한 것의 잘못된 점들이 보였고, 수정을 했습니다. 또 구현을 생각하면서 이런 부분은 좀 더 구체화 되어야 된다는 점들이 보였고, 몇 개의 process 들이 합쳐지고 삭제되고, 추가되어 가면서 현재의 SASD가 완성 되었습니다. 개인적으로는 항상 완벽하게 설계했다고 생각했지만, 단계가 지나가면 갈수록 설계의 허점들이 보이고, 설계를 수정하고, 또 수정하고, 계속 수정해나간 학기가 된 것 같습니다.

수업은 확실히 매력적 이었습니다. 그냥 단순히 입력은 무엇으로 하고 출력은 어떻게 나오는 '코드를 짜는 것'에 중점을 둔게 아니라 '코드를 짜기 위해 무엇을 해야하는가'에 중점을 둔 수업은 처음 이였기에 좋았습니다. 또한, 테스트 부분도 상당히 흥미로웠습니다. 항상 테스트가 중요하다고 책에서만 배웠지, 실제로는 항상 코드를 조금 만들고 돌려보고 내가 생각한대로 안되면 조금 바꾸고 그런식으로 구현과 테스트가 섞어서 항상 만들고 제대로 된 테스트는 진행하지도 않았었는데, 유닛으로 잘라서 테스트 하고, 또 시스템 전체에서 요구사항대로 잘 만들어졌는지 확인 해보고 하는 과정을 거침으로서, 좀더 완벽한 시스템을 만들 수 있었습니다. 그리고 다른조가 저희조를 테스트 해주는 것도 좋았습니다. 저희는 저희가 만들어서 시스템상의 오류를 찾지 못할 수도 있지만, 코드의 내용을 모르는 다른조가 매뉴얼만 보고 저희 시스템을 테스트 해주어서 좀더 객관적으로 저희 시스템을 볼 수 있었습니다.

조금 아쉬웠던 점으로는 좀 더 날카롭게 지적해주시고, 정확하게 피드백을 해주셨으면 좀 더 멘탈이 흔들리고 프로젝트의 완성도도 높아지지 않았을까 합니다. 그이외에는 좋은 경험이 되었던 것 같습니다.

- (4) 임제현 : 처음이 중요하다'라는 생각이 들게 한 수업이었다. 지금까지 무작정 구현만 해오다가 이렇게 세부적으로 기능을 나누어 설계를 하고 그것에 맞추어 구현을 하려고 보니 너무 힘들었다. 설계가 미흡하게 되어있으면 다시 수정을 반복해서 해야하는 일이 배가 되었다. 하지만 확실히 설계를 어느정도 잘 했다면, 이해하기도 쉽고, 구현하기도 수월했다. 따지고 보니 이전처럼 무작정 구현을 한 것과, 이렇게 세부적으로 나누어서 설계하

고 그에 맞춰 구현을 했을 때의 걸린 시간과 비용을 생각해보면 후자가 더 효율적인 것 같다. 아무래도 이런 수업을 처음 들어서 그런지, 팀프로젝트 중반까지는 매우 힘들었다. 중반이 넘어가서는 수정해야할 부분이 줄어들고 익숙해져서 괜찮았다. 부족한 부분이 있던 결과물이었지만, 많은 것을 배워 팀원들과 만든 커피 머신 시스템이 완성되어 만족스럽다. 단순한 '코더'가 아닌, 개발자가 되기 위해 필요한 것들을 많이 알고, 배운 수업이었다.