#6 Final Presentation

Software Verification

Software Verification Team 4

강 정 모
송 상 연
신 승 화

INDEX

1 CTIP

- Jenkins
- Redmine
- SonarQube
- TestLink

2 V&V

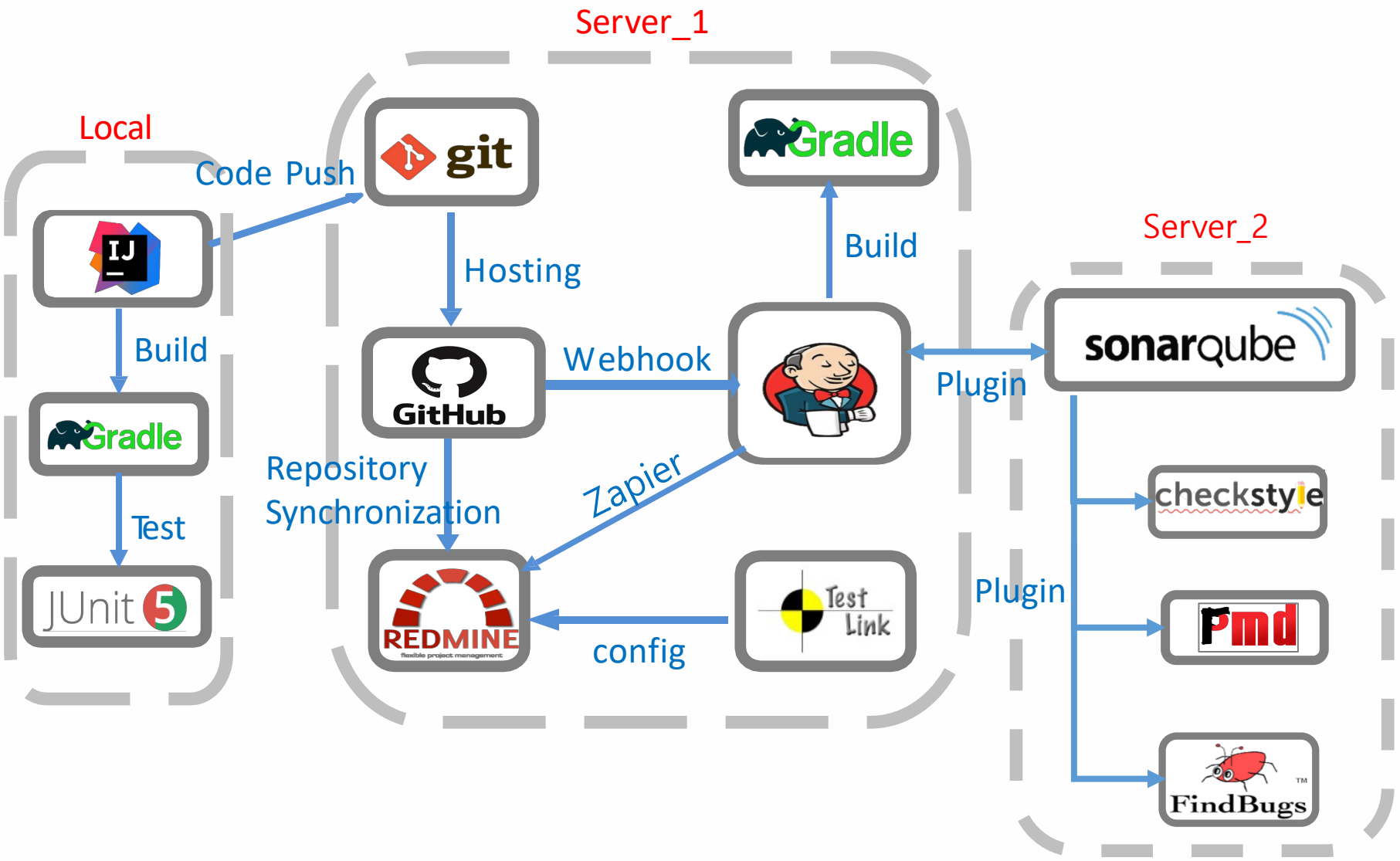
3 Overall

1

CTIP

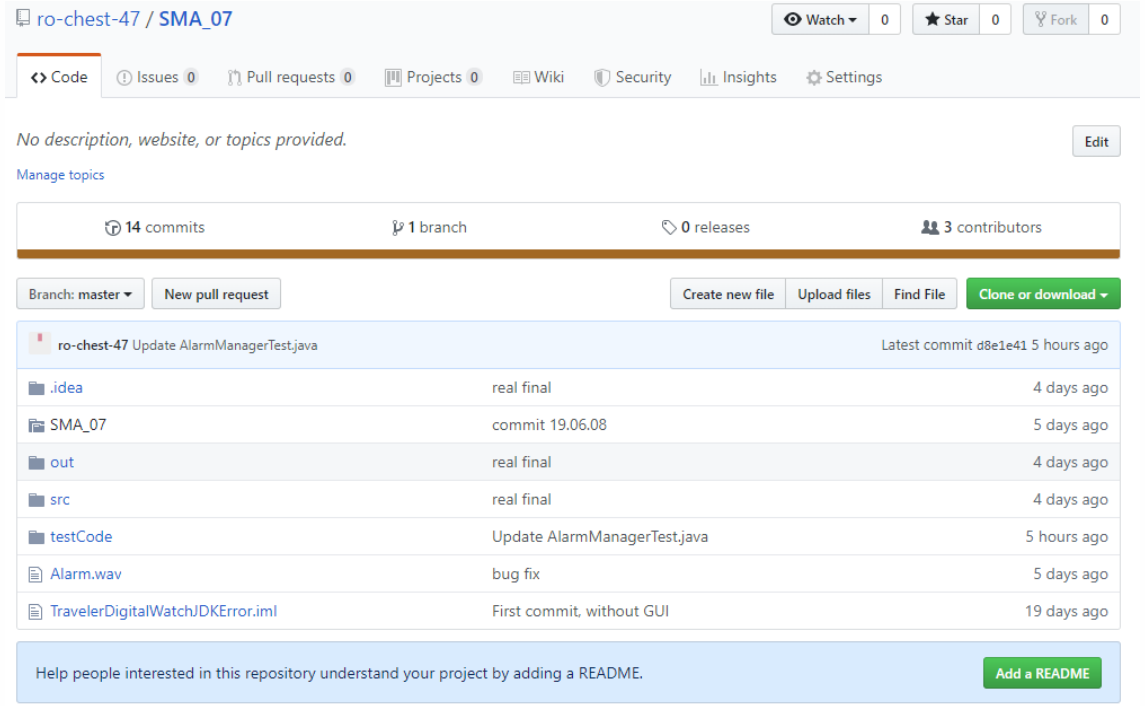
- Jenkins
- Redmine
- SonarQube
- TestLink

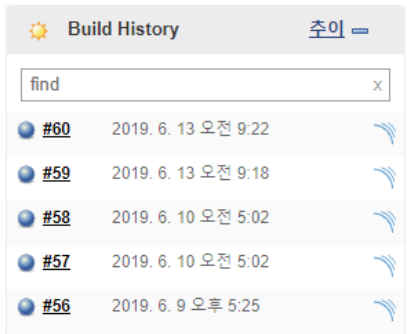
CTIP Overall Structure



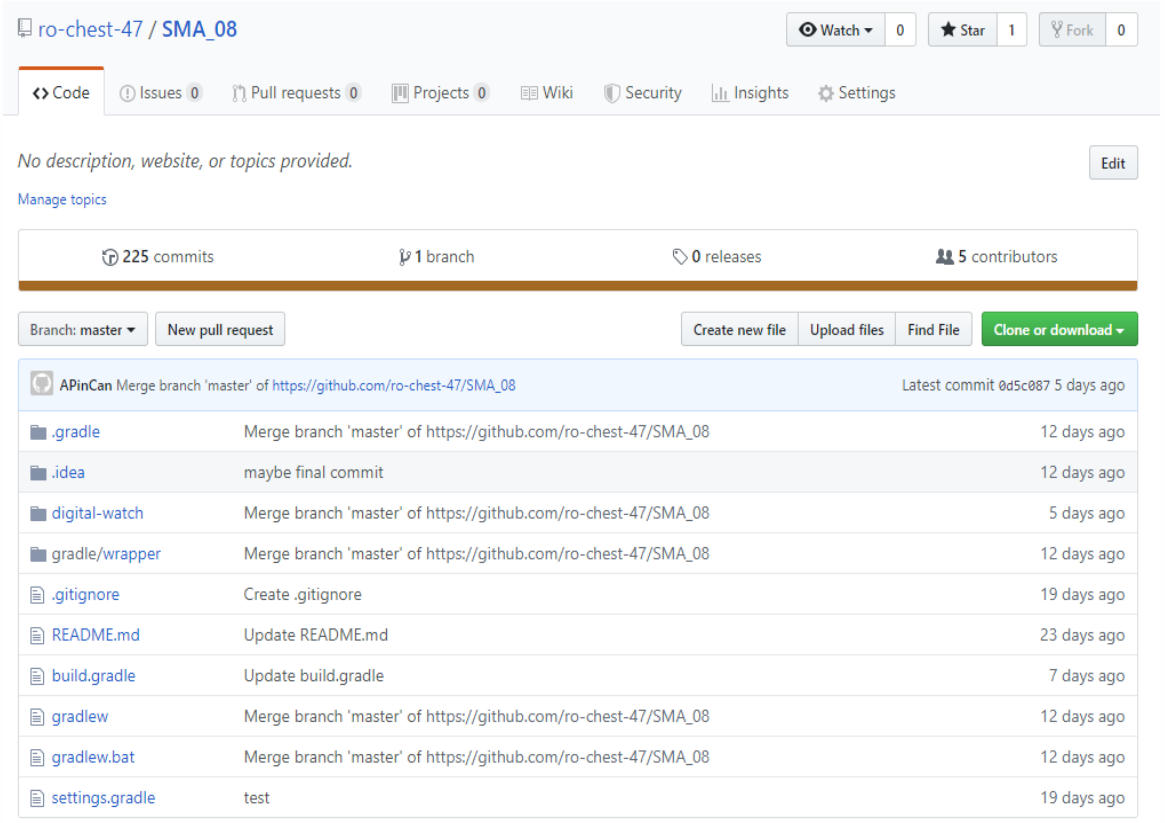
Jenkins

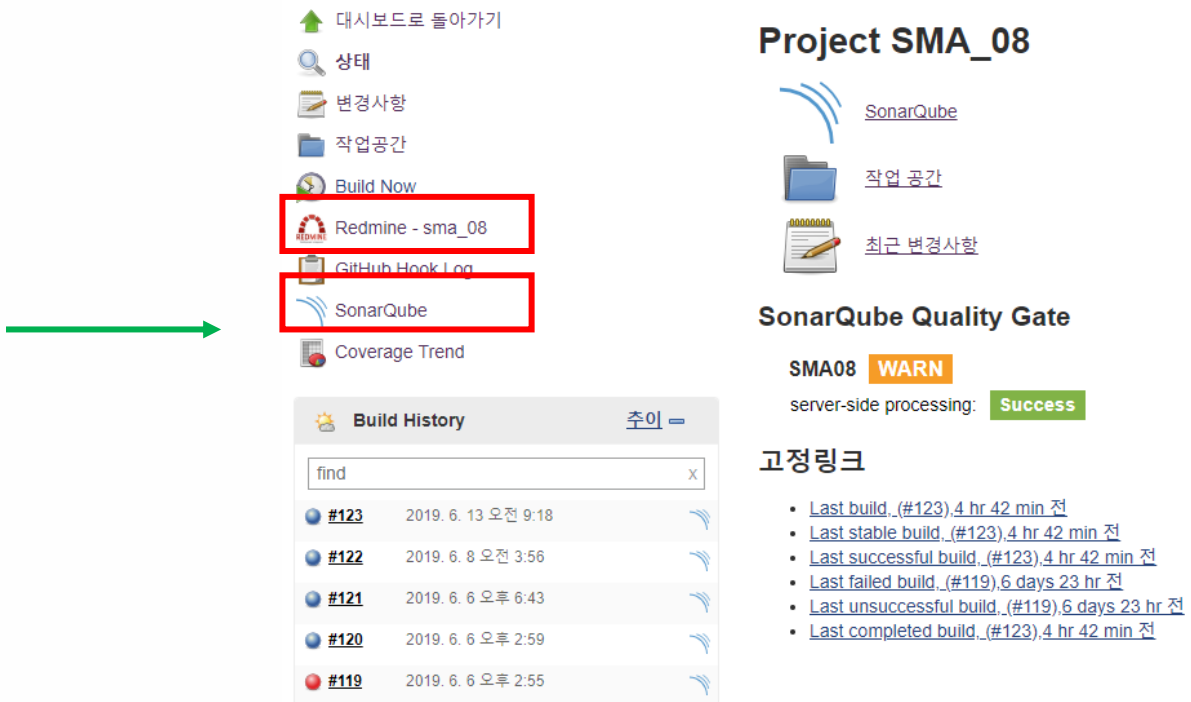
GitHub & Jenkins – 7조





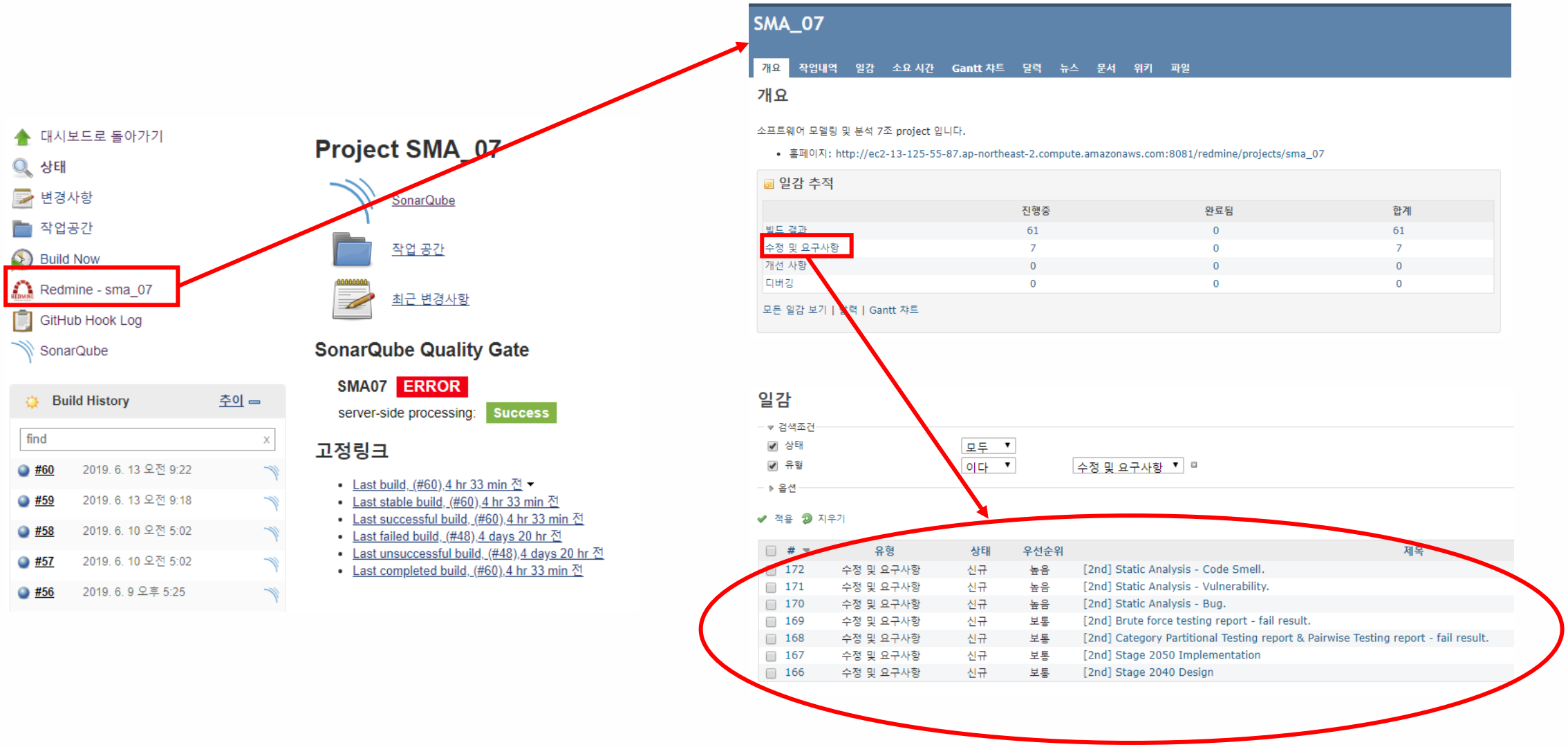
GitHub & Jenkins – 8조



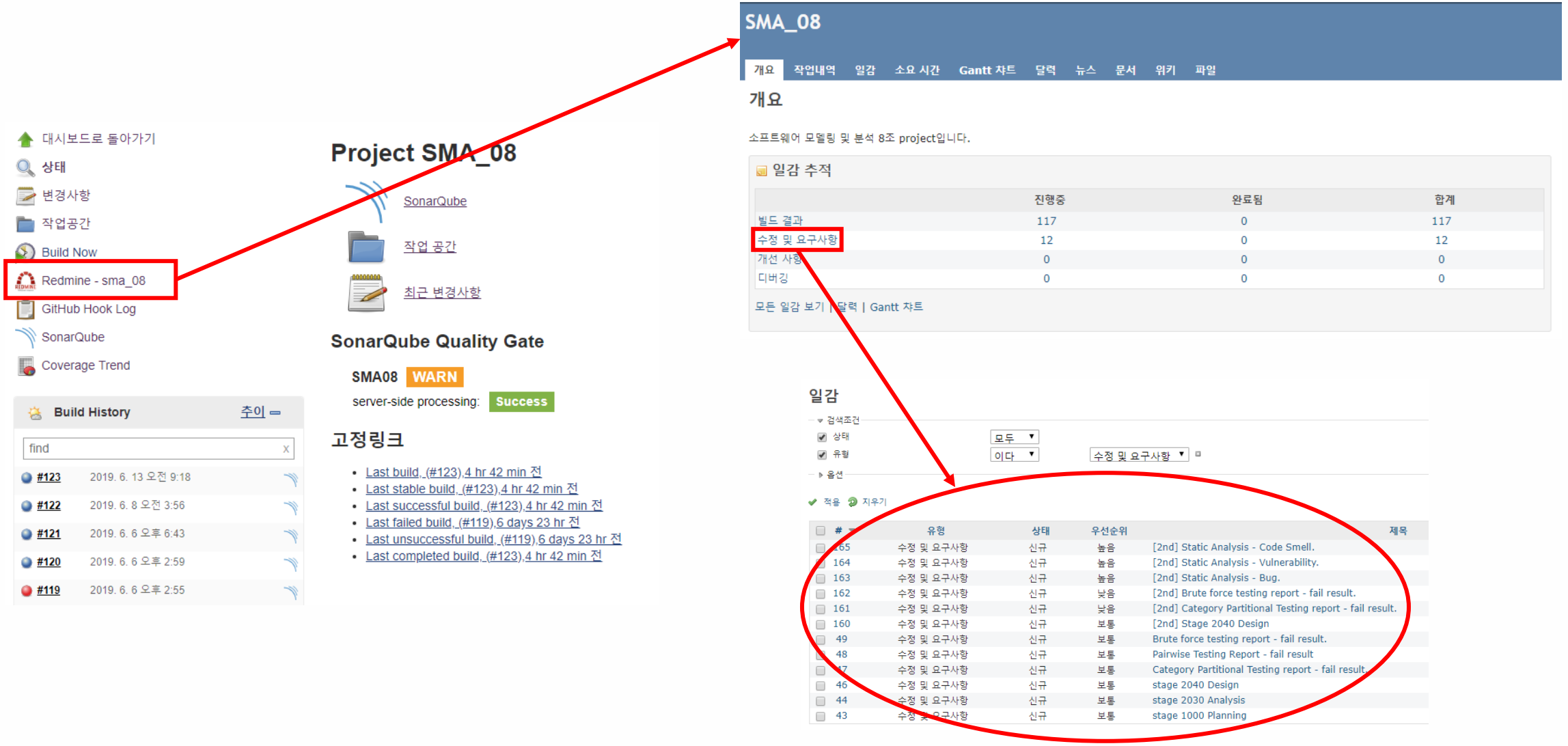


Redmine

Redmine & Jenkins – 7조

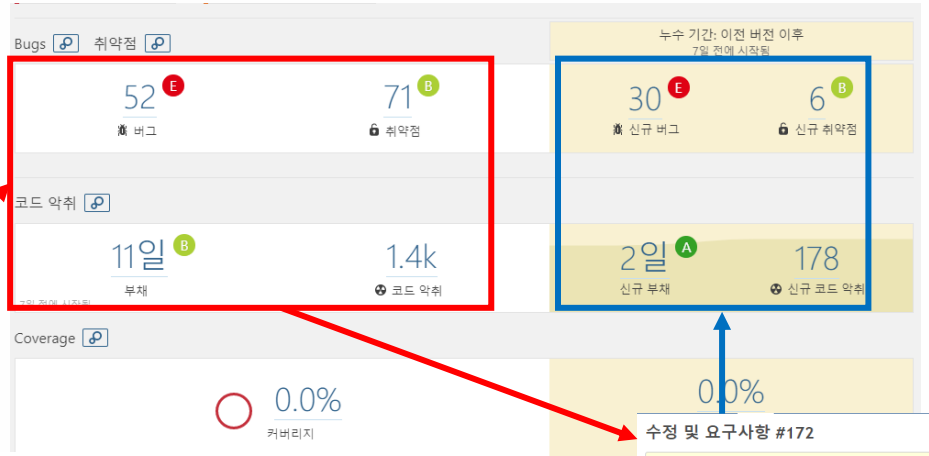


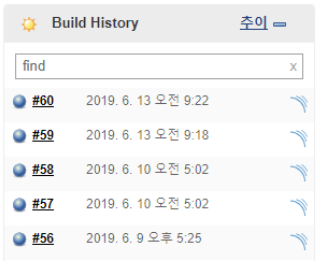
Redmine & Jenkins – 8조

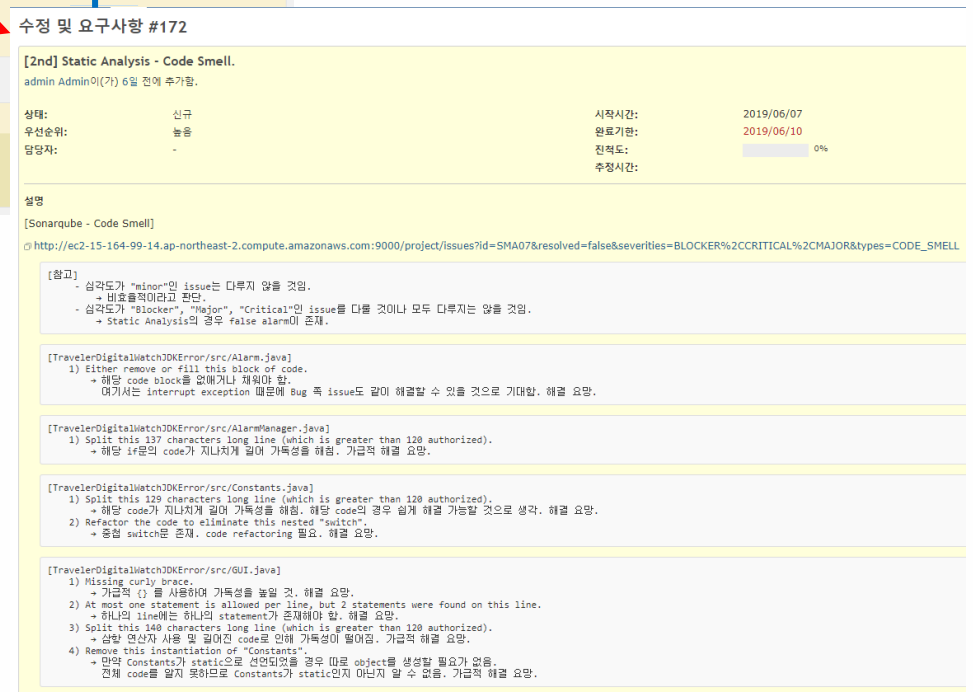


SonarQube

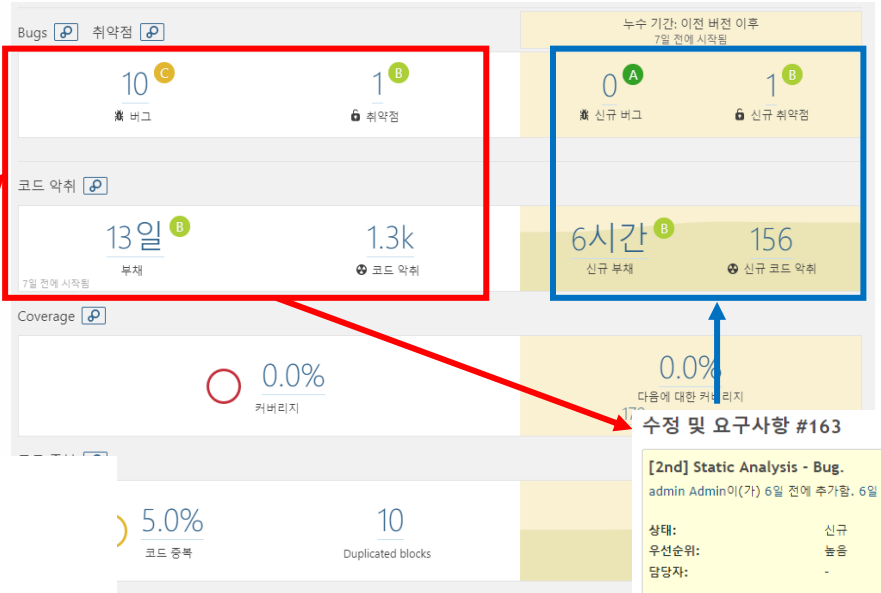
Sonarqube & Jenkins – 7조

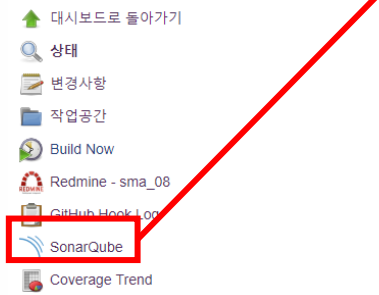


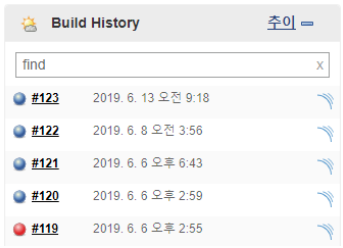




Sonarqube & Jenkins – 8조







TestLink

TestLink & Redmine – 7조

테스트 케이스	요약	상태	중요도	실행타입
sma7-1 : R.2.1 시간을 00:00:00으로 세팅하고 타이머 시작	시간을 00:00:00으로 세팅하고 타이머 시작	Draft	보통	수동
sma7-2 : R.3.2 00:00:00인 상태에서 stopwatch 리셋 후 시작	00:00:00인 상태에서 stopwatch 리셋 후 시작	Draft	보통	수동
sma7-3 : R.2.1 Timer의 올바른 시간표시	Timer가 흐를 때, 올바른 시간 단위로 흐르는가를 확인한다. (ex. 1분일때 59초로 제대로 넘어가는가)	Draft	보통	수동
sma7-4 : R.3.0 Stopwatch의 올바른 시간표시	Stopwatch가 흐를 때, 올바른 시간 단위로 흐르는가를 확인한다. (ex. 분은 60분까지, 초는 60초까지/ 59분일 때 1시간으로 제대로 넘어가는가)	Draft	보통	수동
sma7-5 : R.2.1 Timer 실행중 시간 변경	Timer 실행중 시간 변경이 가능하지 않은지 확인한다.	Draft	보통	수동
sma7-6 : R.3.0 Stopwatch 실행중 시간 변경	Stopwatch 실행중 시간 변경이 가능하지 않은지 확인한다.	Draft	보통	수동
sma7-7 : R.10.0 중복된 Alarm	Alarm을 2개이상 중복되게 설정한 경우 알람이 제대로 울리는지 확인한다.	Draft	보통	수동
sma7-8 : R.10.0 Time 모드일 때 알람	Time 모드 화면일 때, 알람이 제때 울리는지 확인한다.	Draft	보통	수동
sma7-9 : R.10.0 Timer 모드일 때 알람	Timer 모드 화면일 때, 알람이 제때 울리는지 확인한다.	Draft	보통	수동
sma7-10 : R.10.0 Stopwatch 모드일 때 알람	Stopwatch 모드 화면일 때, 알람이 제때 울리는지 확인한다.	Draft	보통	수동
sma7-11 : R.10.0 Global time 모드일 때 알람	Global time 모드 화면일 때, 알람이 제때 울리는지 확인한다.	Draft	보통	수동
sma7-12 : R.10.0 Time 모드일 때 타이머	Time 모드 화면일 때, 타이머가 제때 울리는지 확인한다.	Draft	보통	수동
sma7-13 : R.10.0 Timer 모드일 때 타이머	Timer 모드 화면일 때, 타이머가 제때 울리는지 확인한다.	Draft	보통	수동
sma7-14 : R.10.0 Stopwatch 모드일 때 타이머	Stopwatch 모드 화면일 때, 타이머가 제때 울리는지 확인한다.	Draft	보통	수동
sma7-15 : R.10.0 Global Time모드일 때 타이머	Global Time 모드 화면일 때, 타이머가 제때 울리는지 확인한다.	Draft	보통	수동
sma7-81 : R.4.0 Alarm 비활성화 시 알람이 울리지 않는지 확인	Alarm 비활성화 시 알람이 울리지 않는지 확인한다.	Draft	보통	수동
sma7-82 : R.6.0 Global Time이 제대로 설정되는지 확인	Global Time이 설정하고 싶은 시간으로 제대로 설정되는지 확인	Draft	보통	수동
sma7-83 : R.8.0 4가지 모드 설정 가능한지 확인	시간, 타이머, 스탑워치, 알람, 여행지 시간, 난수 생성 중 4가지를 선택하여 모드 변경으로 선택이 가능한지 확인한다.	Draft	보통	수동
sma7-84 : R.5.2 난수 숫자 범위 설정이 가능한지 확인	난수를 생성할 숫자 범위 설정이 가능한지 확인한다.	Draft	보통	수동
sma7-85 : R.5.1 난수 생성이 가능한지 확인	User가 난수를 요청하면 R.5.2에서 설정한 숫자 범위내의 수를 복원 추출하여 보여주는지 확인	Draft	보통	수동
sma7-86 : R.1.0 요일이 제대로 변경되는지 확인	연,월 ,일 을 조정할 때 날짜에 맞게 요일이 제대로 변하는지 확인	Draft	보통	수동
sma7-87 : R.2.0 타이머의 최대 시간	타이머의 최대 시간이 일정한지 확인	Draft	보통	수동
sma7-88 : R.1.0 시간 설정	내가 설정하고 싶은 시간대로 시간을 설정할 수 있는가를 확인	Draft	보통	수동

수정 및 요구사항 #169

[2nd] Brute force testing report - fail result.

admin Admin이(가) 6일 전에 추가함.

상태:	신규	시작시간:	2019/06/07
우선순위:	보통	완료기한:	2019/06/10
담당자:	-	진척도:	0%
		추정시간:	

설명

1. timer 모드에서 시간을 00:00:00으로 세팅하고 시작 버튼을 눌러 timer를 실행한다.
   => 프로그램이 동작하지 않음.
2. stopwatch 모드에서 시간이 00:00:00일 때 리셋 버튼을 눌러 리셋 후, 시작 버튼을 눌러 stopwatch를 실행한다.
   => 프로그램이 동작하지 않음.
3. stopwatch 모드에서 59분 59초에서 1시간 00분 00초로 넘어가는지 확인한다.
   => 59분에서 1시간으로 넘어가지 않음.
4. 동일한 alarm을 2개 이상 중복하여 설정한 후, 설정한 시간이 되었을 때 alarm이 제대로 울리는지 확인한다.
   => 동일한 alarm을 2개 이상 중복하여 설정할 경우, 설정한 시간이 되었어도 alarm이 제대로 울리지 않음.
5. alarm을 1개 설정한 후에, time 모드 화면으로 전환하고 기다렸다가 설정한 시간에 alarm이 제대로 울리는지 확인한다.
   => alarm이 제 시간에 제대로 울리지 않고, 설정한 시간에서 6초 정도 시간이 더 흐른 후에 울림.
6. alarm을 1개 설정한 후에, timer 모드 화면으로 전환하고 기다렸다가 설정한 시간에 alarm이 제대로 울리는지 확인한다.
   => alarm이 제 시간에 제대로 울리지 않고, 설정한 시간에서 6초 정도 시간이 더 흐른 후에 울림.
7. alarm을 1개 설정한 후에, stopwatch 모드 화면으로 전환하고 기다렸다가 설정한 시간에 alarm이 제대로 울리는지 확인한다.
   => alarm이 제 시간에 제대로 울리지 않고, 설정한 시간에서 6초 정도 시간이 더 흐른 후에 울림.
8. alarm을 1개 설정한 후에, global time 모드 화면으로 전환하고 기다렸다가 설정한 시간에 alarm이 제대로 울리는지 확인한다.
   => alarm이 제 시간에 제대로 울리지 않고, 설정한 시간에서 6초 정도 시간이 더 흐른 후에 울림.
9. timer를 1개 설정한 후에, time 모드 화면으로 전환하고 기다렸다가 설정한 시간이 만료되면 timer alarm이 제대로 울리는지 확인한다.
   => timer alarm이 설정한 시간이 만료되어도 울리지 않는다.

TestLink & Redmine – 8조

테스트 케이스	요약	상태	중요도	실행 형태
sma8-1 : R.0.0 시간측정	시간이 정상적으로 1초마다 업데이트 되는지 확인한다.	Draft	보통	수동
sma8-2 : R.0.1 윤년	날짜 설정 시 윤년을 고려하는지 확인한다.	Draft	보통	수동
sma8-3 : R.2.2 같은 시각의 알람	같은 시각의 알람이 2개 이상 설정된 경우에도 울린다.	Draft	보통	수동
sma8-4 : R.2.4 TimeKeeping 모드일 때 알람	TimeKeeping 모드 화면일 때, 알람이 제때 울리는지 확인한다.	Draft	보통	수동
sma8-5 : R.2.4 Timer 모드일 때 알람	Timer 모드 화면일 때, 알람이 제때 울리는지 확인한다.	Draft	보통	수동
sma8-6 : R.2.4 Stopwatch 모드일 때 알람	Stopwatch 모드 화면일 때, 알람이 제때 울리는지 확인한다.	Draft	보통	수동
sma8-7 : R.2.4 Tide 모드일 때 알람	Tide 모드 화면일 때, 알람이 제때 울리는지 확인한다.	Draft	보통	수동
sma8-8 : R.2.4 Moonphase 모드일 때 알람	Moonphase 모드 화면일 때, 알람이 제때 울리는지 확인한다.	Draft	보통	수동
sma8-9 : R.2.4 SelectMode 모드일 때 알람	SelectMode 모드 화면일 때, 알람이 제때 울리는지 확인한다.	Draft	보통	수동
sma8-10 : R.1.5 TimeKeeping 모드일 때 타이머	TimeKeeping 모드 화면일 때, 타이머가 제때 울리는지 확인한다.	Draft	보통	수동
sma8-11 : R.1.5 Alarm 모드일 때 타이머	Alarm 모드 화면일 때, 타이머가 제때 울리는지 확인한다.	Draft	보통	수동
sma8-12 : R.1.5 Stopwatch 모드일 때 타이머	Stopwatch 모드 화면일 때, 타이머가 제때 울리는지 확인한다.	Draft	보통	수동
sma8-13 : R.1.5 Tide 모드일 때 타이머	Tide 모드 화면일 때, 타이머가 제때 울리는지 확인한다.	Draft	보통	수동
sma8-14 : R.1.5 Moonphase 모드일 때 타이머	Moonphase 모드 화면일 때, 타이머가 제때 울리는지 확인한다.	Draft	보통	수동
sma8-15 : R.1.5 SelectMode 모드일 때 타이머	SelectMode 모드 화면일 때, 타이머가 제때 울리는지 확인한다.	Draft	보통	수동
sma8-16 : R.1.0 Timer의 올바른 시간 표시	타이머가 진행될 때, < 시 분 초> 가 시간 개념에 맞게 정확하게 표시되어 진행되는지 확인한다.	Draft	보통	수동
sma8-17 : R.3.0 Stopwatch의 올바른 시간 표시	스톱워치가 진행될 때, < 시 분 초 0.1초> 가 시간 개념에 맞게 정확하게 표시되어 진행되는지 확인한다.	Draft	보통	수동
sma8-18 : R.6.0 TimeKeeping 모드 화면일 때, modeSelect 진입	TimeKeeping 모드 화면일 때, modeSelect로 제대로 진입하는지 확인	Draft	보통	수동
sma8-19 : R.6.0 Timer 모드 화면일 때, modeSelect 진입	Timer 모드 화면일 때, modeSelect로 제대로 진입하는지 확인	Draft	보통	수동
sma8-20 : R.6.0 Alarm 모드 화면일 때, modeSelect 진입	Alarm 모드 화면일 때, modeSelect로 제대로 진입하는지 확인.	Draft	보통	수동
sma8-21 : R.6.0 Stopwatch 모드 화면일 때, modeSelect 진입	Stopwatch 모드 화면일 때, modeSelect로 제대로 진입하는지 확인.	Draft	보통	수동
sma8-22 : R.6.0 Tide 모드 화면일 때, modeSelect 진입	Tide 모드 화면일 때, modeSelect로 제대로 진입하는지 확인.	Draft	보통	수동
sma8-23 : R.6.0 Moonphase 모드 화면일 때, modeSelect 진입	Moonphase 모드 화면일 때, modeSelect로 제대로 진입하는지 확인.	Draft	보통	수동

Showing 1 to 23 of 23 entries Previous 1 Next

수정 및 요구사항 #49

Brute force testing report - fail result.

admin Admin이(가) 13일 전에 추가함.

상태:	신규	시작시간:	
우선순위:	보통	완료기한:	
담당자:	-	진척도:	
		추정시간:	

설명

Brute force testing 결과에 대해서는 첨부파일을 참고해주세요. # 첨부파일 1 참고.

11번 case. sma8 - 20 : R.6.0 Alarm 모드 화면일 때, modeSelect 진입 여부 확인.

Alarm 모드 화면일 때, reset 버튼을 4회 눌러 modeSelect로 제대로 진입하는지 확인.

```
=> stage_2050 30 page_reqModeSelect에서 '어느 모드에서든지'로 spec에 나와있었으나,
   확인 결과 Alarm 모드에서는 reset 버튼 4회 입력 시 modeSelect로 진입하지 않았음.
```

1.png (169 KB) 첨부파일 1 참고 admin Admin, 2019/05/31 08:06

2

V & V

T4 소감 - 강정모

- [Spec review]

  - spec의 양이 방대하고 중복되는 내용이 상당수 존재하나, stage 별로 내용이 변하는 것이 분명 존재하고 이후 System testing part의 expected output 을 생성하기 위해서는 반드시 거쳐야만 하는 단계였다.

- [System testing]

  - category partitional testing

    > 개념은 알고 있으나 testing 하는 system 에 따라 유효한 category 를 설정하는 것과 또한 해당 category 의 represent values 를 결정하는 것이 생각보다 어려웠던 것 같다.

  - pairwise testing

    > 설정한 category 를 어떻게 묶을 것인가, 를 또 고민해야했다.

  - brute-force testing

    > 경험이 쌓인다면 정말 강력한 검증 기법이 될 수 있다는 생각이 들었지만 경험이 많이 없어서 그런지 CPT 나 PWT 에서 크게 벗어난 case 는 없었던 것 같았다.

- [Static analysis]

  - 다양한 rule set 을 적용하여 code 를 분석했을 때, false alarm 과 minor 한 issue 들이 다수 존재했지만, 그 중에서도 몇 가지 critical 한 issue 들이 발견되는 것이 신기했다.

T4 소감 – 송상연

- 직접 카테고리를 만들어서 테스트 해보니 확실히 Brute Force Test처럼 막 테스트하는 것보다는 체계적이었고, 최대한 빈틈없이 category를 잘 나눠서 테스트하는 것이 중요하겠구나. 라고 느꼈다.

- 검증자의 입장에서 개발자의 코드를 분석하는 것이 새로운 경험이었다. 소모 팀에게는 미안하지만, Brute Force Test를 하기 위해 이것저것 실행시켜보면서 에러를 찾을 때면 굉장히 뿌듯했다. 😏

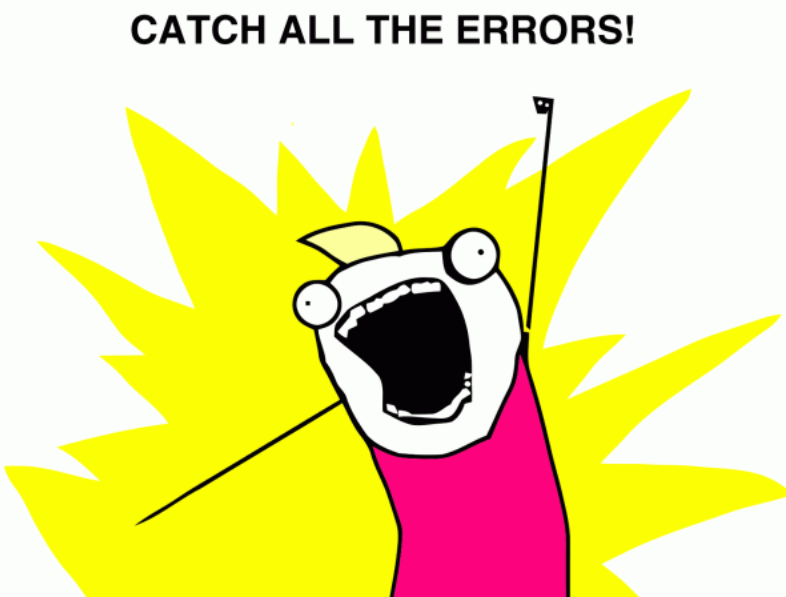
- 정적 분석을 할 때에는 ' 이런 코드 규칙까지 지켜야 해?' 싶은 것도 있었지만, 그저 프로그램이 목표하고자 한 기능들이 수행되게 짜는 것뿐만 아니라 코드를 짜는데 있어서 좋은 습관을 가져야겠구나. 라고 느꼈다. 🤔

T4 소감 - 신승화

- brute force 테스팅을 하면서 category partition이나 pairwise에서 잡아내지 못한 에러를 세세하게 잡아낼 때, 전문가의 역량이 가장 중요하구나 하는 것을 느꼈습니다.

- 막연하게 7조 / 8조 중에서 8조가 훨씬 더 코드를 잘 짰다고 생각해서 정적 분석의 결과가 더 잘 나올 것이라고 생각했는데 의외로 결과가 비슷해서 정적분석은 코드가 잘 돌아가는 것과는 별개구나 하는 것을 알게 되었습니다.

- 테스팅을 진행할 때 소모팀이 실행 파일을 제대로 올려 주지 않아서 애를 먹긴 했지만, 소통을 통해 잘 해결해 나갔습니다.



3

Overall

T4 소감 - 강정모

- 매 순간이 쉽지 않았던 것 같은데 그만큼 얻어가는 것이 많은 수업이었다.
  학년이 올라갈수록 단순히 '만드는 것'은 6개월 코딩 학원 다닌 사람도 할 수 있지만
  '잘 만드는 것' 은 전공 지식과 고민이 필요한 문제라는 생각이 들었고,
  그런 점에서 '잘 만들었는가?' 라는 부분에 검증이 요구되는 것 같아서
  학기 초 모델링 강의를 취소 후 검증 강의로 옮겼는데 그만큼 무엇인가를 얻어가는 것 같다.
  (쉽게 접하기 어려운 영역과 지식인 것 같다.)

- Tool 간의 연동이 생각만큼 쉽지 않아서 CTIP 환경을 구축하는데 소비한 시간이 많은 게 아쉽다.
  (편의를 제공하기 위해 만든 tool이 오히려 사용자를 불편하게 만든다는 점에서 모순이 존재하는게 아닐까...)
  특히 redmine tool의 plugin 설치는 약 2주간에 걸쳐 계속 시도했는데 끝끝내 성공하지 못했다.
  이런 시행착오가 나중에 도움이 되었으면 좋겠다.

- 이번 한 학기 동안 검증이 생각만큼 쉬운 일이 아니고, 깊고 방대한 영역이며,
  정형화된 패턴이 항상 존재하는 것도 아니고 경험과 지식이 조화롭게 축적되어야 비로소 성장하는 영역이라는 것을
  실습을 진행하면서 잘 느낄 수 있었다. 정말 이론과 실전은 다르다는 것을 잘 알 수 있었다. (생각만큼 만만하지가 않다....)

- 단순히 '아는 것' 보다는 '정확하게 아는 것' 이 더 중요하다는 생각은 하고 있었지만, 이번 검증 강의에서 이론과 실전의 거리감을
  느껴보니 '정확하게 아는 것' 에서 더 나아가 '아는 것을 어떻게 사용하는 지' 까지 알아야 비로소 확실하게 안다고 말할 수 있지
  않을까 하는 생각이 들었다.

T4 소감 – 송상연

- CTIP환경 구축할 시 툴끼리 붙이는 게 버젼 충돌 등의 이유로 생각보다 너무 안 붙어서 설정을 여러차례 건드렸는데, 굉장히 계란으로 바위치기 하는 느낌이었다. (심지어 Redcase는 Redmine의 플러그인인데, 붙지 않았다..)

- 무료인 건 무료인 이유가 있다고 느꼈다..

- 소모 팀과의 소통은 원활했다. dslab페이지에 올라오지 않은 자료들을 요청하면, 즉각적으로 보내주었다. 초기 스펙에서 명시하였던 예외처리가 하기 어렵거나 시간이 부족하거나 등의 상황이 생기면, 코드가 아닌 스펙을 수정하는 방식은 조금 아쉬웠으나, 2차 시스템 테스트와 정적분석을 거치면서 테스트의 pass rate %가 올라가는 모습을 보니 덩달아 뿌듯하였다.

- 코드를 짜는 것도 어렵지만, 짜여진 코드를 정말로 잘 짠 코드인지 여러 방식으로 검증하는 것도 어렵구나 라고 느꼈다. 하지만 새로운 분야를 경험해보았고, 시야를 넓힐 수 있는 기회가 된 것 같다.

p.s. 블로그에 자세한 설명들을 올려주신 조대협, 꿈꾸는 태태태 씨에게 감사를 표합니다.

T4 소감 - 신승화

- 같은 회사끼리도 무료인 툴은 서로 붙지 않는 것을 보고 정말 '돈이 좋은 것이구나..' 하는 생각이 들었습니다... 그렇지만 이렇게 한번 해보는 것도 나중에 실무를 할 때에 큰 밑거름이 될 것 같아 의미 있는 수업이었다고 생각합니다!

- 팀 프로젝트를 하면서 다른 팀과의 커뮤니케이션을 통해 프로젝트를 한 경험은 이번이 처음이라 신선했습니다. 물론 소모팀과의 소통이 원활했기 때문에 무난하게 프로젝트를 마칠 수 있었지만, 그렇지 않았다면 꽤나 험난한 프로젝트가 되었을 것 같은 예감이 듭니다.
- 이번 프로젝트를 하면서 실습을 하는 것도 어렵지만, 문서 작업도 만만치 않다는 것을 느끼게 되었습니다. 😀

#6 Final Presentation

THANK YOU