1. 대략적으로 1달 정도의 시간이 흘렀습니다.1달의 시간 동안 느낀점과 간략한 소감을 작성해주세요.부가적으로 앞으로의 포부 또한 같이 작성해주시면 감사하겠습니다.

   1달의 시간을 하면서 새로운것을 많이 배우다 보니 머리를 많이 썻다는 느낌을 많이 받았고 새로운 기술을 접하다 보니 신기하고 재미있었습니다.

   앞으로도 새로운 언어를 배우고 다른 형태의 기술들을 많이 접할 때 최선을 다해 배우고 익히겠습니다.

2. 언어를 가리지 않고 Domain Driven Design을 수행하여 얻을 수 있는 이점들에 대해 기술하세요.

   유지 및 보수가 편리하다, 개발의 생산성이 높아진다, 확장성이 높아진다.

3. C/C++ 에서 Test 목적으로 GoogleTest를 사용하였습니다. TDD(Test Driven Development)를 통해 얻을 수 있는 이점들에 대해 기술하세요.

   내가 원하는 기능이 올바르게 작동하는지 판별이 가능하다.

   다른사람과의 협업과정에서문제발생을 빠르게 알 수 있다.

4. DDD(Domain Driven Design)과 TDD(Test Driven Development)의 관계에 대해 기술하세요.

   확장성을 늘리는 장점을 가진 DDD에 TDD로 안정성과 협업과정 속 발생할 문제를 도와준다.

5. 팀원들과 협업을 진행 할 때 중요하다 생각하는 요소들을 작성하고 각 요소들에 대해 조금 더 상세하게 기술하세요.

   하나의 프로젝트를 만들 때 같은 방향성과 같은 목적이 있는게 가장 중요하다고 생각됩니다. 방향성과 목적이 다르면 만드는 프로젝트의 기능이 의견에 맞지않아 재작업 및 많은 수정을 하여 생산성을 낮추게 됩니다. 소통의 문제도 중요하다 생각됩니다. 소통이란 서로 대화하며 서로의 생각을 나눈다고 생각되는데 이러한 소통이 원활하게 이루어지지 않으면 협업 과정에서 작게는 중복 작업 크게는 프로젝트의 롤백이 이루어 집니다.

6. 함수 포인터 테이블을 사용하는 이유에 대해 IoC 관점에서 기술하세요.

   함수 포인터 테이블을 통한 방법은 제어 권한을 내가 아닌 다른 대상에게 위임을 통해 작업의 구현과 수행을 분리하는 IOC의 방법을 사용했다고 생각됩니다. 함수 포인터 테이블을 사용한 이유가 외부에서 들어온 입력에 맞춰 올바른 함수를 실행해야 합니다. 그러므로 제어 권한은 외부의 입력에 맡기고 구현 및 수행을 함수 포인터 테이블이 실행합니다.

7. virtual method를 사용하는 이유에 대해 기술하시오.

   함수 포인터를 정의하며 virtual을 통해 함수 포인터 문법의 난해함을 최소화 해줍니다. 저희가 실제로 사용한 virtual은 간단히 어떤 함수를 만들것 인지를 선언하고 후에 재 정의하는 느낌을 통하여 함수 포인터의 역할을 했습니다.

8. 협력형 과제를 진행하는 상황 혹은 회사에서 업무를 진행하는 상황에서 갑자기 대응하기 어려운 문제를 마주하게 되었습니다. 이런 경우 여러분들은 어떤 형식으로 문제에 대응 할 것인가요 ? 어떻게 대응 할 것인지 상세하게 기술하세요.

   회사내의 이슈를 발행시키는 곳이 있다면 그곳에서 이슈를 발행하여 도움을 요청하거나 본인의 사수 및 본인보다 높은 직급의 사람들에게 도움을 요청할 것입니다. 만약 위의 두가지 방법이 모두 안되거나 제지를 당할경우 외부의 본인보다 더 많을 것을 아는사람(ex 강사님)을 통해 해결방안을 찾아 갑니다. 아무에게도 알리면 안되는 경우는 인터넷을 참고하여 대응하겠습니다.

9. Backlog를 작성함으로서 얻을 수 있는 이점들에 대해 기술하세요.

   중복작업의 예방 및 프로젝트의 목적과 방향성을 일치시킬 수 있고 어떤 일을 하고있는지 파악이 쉬워 도움을 주거나 받기가 용이합니다 이러한 과정을 통해 생산성을 높이고 소통이 원활하게 이루어 집니다.

10. 여러분들은 이미 Board (게시판) 을 Console UI와 함께 구현해봤습니다. 이 때 여러분들이 각자 분업하여 만들었던 작업들이 존재합니다. 여러분들이 각자 분업하여 만든 내용물들이 다른 사람들의 결과물과 원활하게 결합하지 못하여

폐기처분하는 경우도 있었습니다. 혹은 끔찍한 혼종이 탄생하는 경우도 있었습니다.이런 현상이 나타나게 된 근본적인 이유에 대해 기술해봅시다.

백로그 작성이 올바르게 안되어 서로 어떤일을 하는지 어떤 목적성을 가졌는지 몰랐고 결합시 어떤 정보를 필요로 하고 어떤 정보를 받아와야 하는지를 모르기 때문에 터졌습니다. 깃의 잘못된 사용의 예시로는 본인의 내용과 깃의 최종본이 달라 이상한 내용이 생성되고 최신 업데이트 버전에서 옛날 버전으로 롤백되었습니다.

11. 기능 단위로 Backlog 를 작성하는 경우 어떤 문제들이 발생 할 수 있는지 기술해봅시다.

기능 단위의 백로그를 작성했을 경우 목적이 불분명 해져서 소통의 이슈가 발생할 수 있습니다. 또한 복잡성이 증가함에 따라 목적성이 퇴색이 됩니다.

12. https://github.com/crazyfire29/SDC-Comprehensive-Evaluation/commit/f46784c5129ca48537658f381df9ce0572f46614

21. 현재 실시간으로 분석되는 영상 시스템을 사용하고 있습니다. 이 영상 시스템은 촬영하는 구간에 사람이 몇 명 있는지 확인 할 수 있습니다. 이러한 시스템을 특정 시간에 사람들이 붐비는 구간에서 사용하고자 합니다. 영상 시스템의 목적은 버스 정류장에 사람이 몇 명 있는지 판정하려고 합니다. 어떻게 DB에 데이터를 저장해야 스토리지를 효율적으로 사용 할 수 있을까요

사람이라고 판되고 일정 시간동안 같은 공간에서 있을 경우 일정 시간마다 데이터 베이스에 숫자를 갱신하게 합니다.

22. 포인터가 필요한 이유가 무엇인지 기술하세요.

메모리를 동적으로 할당하여 사용하기 위해

23. 모든 함수들이 자신만의 개별적 공간인 Stack을 사용합니다. 함수들끼리 서로 Stack 을 공유하지 않는 이유에 대해 기술하세요.

함수 간의 데이터 누출을 방지하고 프로그램의 안정성을 높입니다.

24.
https://github.com/EDDI-RobotAcademy/SDC-Comprehensive-Evaluation/commit/358efa90cca4b23e42e9ba06187b0dc24a36f661

26.

33. 처음에는 Domain 분리가 필요 없다 생각하였던 특정 Entity 내부의 어떤 멤버 변수 (필드) 가 있습니다. 시간이 지남에 따라 시스템이 커졌고 위 내부 변수가 수행하는 작업의 복잡도가 높아지고 있습니다. 여러분들이라면 이 녀석을 어떻게 처리하겠습니까 ?

새로운 domain으로 분리하여 기존 domain의 역할을 덜어줍니다

34.