

4조 최하경 세미 프로젝트 (23.03.09~23.09.20) 소감문

[발단]

자바를 배운지 4주차 쯤에는 언제 프로젝트를 시작하는지 궁금하고 기다려졌습니다. 하지만 그 이후 자료구조와 직렬화를 배우면서 이해를 못하는 부분이 생기는 와중에 강사님께서 세미 프로젝트를 시작한다고 하셨습니다. 듣고 싶었던 말을 막상 들으니 많이 떨리고 긴장되었습니다. 오로지 팀원들에게 민폐가 되지 말자는 다짐을 했었던 것 같습니다.

[전개]

강사님이 미리 만들어두신 팀으로 구성이 되었고, 처음으로 모여 회의를 하며 주제를 선정하는데에는 평화로웠습니다. 서로 적극적으로 의견을 내고 토론하면서 좋은 주제를 얻을 수 있었고, 그동안 배운 것들을 많이 활용할 수 있는 주제라고 생각했습니다. 그래서 모든 일이 다 잘 될 것만 같았고, 얼른 코드를 짜보고 싶은 마음이 들었습니다.

[위기-1]

하지만 친하게 지내던 사람과 함께 일을 한다는 건 새로운 상황이었습니다. 일을 진행하는데 있어서 서로의 가치관이나 취향에서 드러나는 확연한 차이가 있고, 어떻게 보면 정 반대의 성격을 가졌다고 할 수 있는 팀원과 일을 하게 되면서 엇갈리는 부분이 있었습니다. 하지만 모두의 목표는 같았습니다. 세미 프로젝트를 무사히 완성시키는 공동의 목표를 두고 생각해 볼 때, 모두가 잘 하고 싶은 마음에 그런 것이라는 이해를 가질 수 있었고, 금방 대화로 잘 풀어낼 수 있었습니다. 그 후 원활하게 팀장님의 리드 하에 큰 틀을 잡고, 두 명씩 팀으로 나눠 맡은 부분을 코드로 구현할 수 있었습니다.

[위기-2]

대략 6가지의 클래스로 나누어서 증명서 발급기를 구현하면서 가장 다루기 어려웠던 부분은 각 클래스간 동기화되어야 하는 정보였습니다. 학생 정보를 다루는 클래스에서 새로운 학생을 추가하여도, 로그인을 하는 클래스에서는 새로운 학생의 정보가 동기화 되지 않았습니다. 지금껏 써왔던 다른 클래스의 객체 생성과 호출의 개념과는 다른 무언가가 필요하다고 생각이 되었습니다. 그러다가 다른 조가 데이터베이스를 다루기 위해서 객체 직렬화를 사용한다는 말을 들었습니다. 자바 수업 막바지에 들었던 객체 직렬화를 사실 온전히 이해하지 못해서 쓸 엄두를 못내고 있었는데, 당장 눈앞의 문제를 해결하기 위해서 주먹구구식으로 객체 직렬화를 구현하였고, 학생 정보 데이터를 연동할 수 있었습니다. 이런 과정을 통해서 수업시간에 배운 것들은 절대 쓸모없는 것이 없다는 것을 느꼈습니다.

[위/가-3]

한 번 객체 직렬화를 하고 나니, 클래스간 동기화가 필요한 데이터가 있으면 모두 객체 직렬화를 통해서 연동할 수 있었고, 어려운 부분을 해결하고 나니 자신감이 붙었습니다. 강사님에게 첫 점검을 받으면서 현금 관리에 대한 볼륨감을 키워보라는 관리자 파트를 맡고있는 두 팀원이 고생하며 현금 관리를 구현하였습니다. 4명이라는 아기자기한 숫자로 구성된 팀에서 서로 막히는 부분을 도와주면서 팀워크, 합이라는 것도 생겨나갔습니다.

처음 증명서 발급기라는 주제를 선정할 때부터 재학생과 졸업생을 구분하는 부분을 구현하면서 또 어려움이 생겼습니다. 재학생과 졸업생은 데이터베이스도 다르고, 선택할 수 있는 증명서의 종류도 달랐습니다. 하지만 프로젝트의 볼륨감을 포기할 수 없어서 우리 팀은 재학생과 졸업생을 구분하는 것을 계속 진행하기로 했고, 로그인 클래스와 장바구니 클래스를 재학생용과 졸업생용으로 두 개씩 만들었습니다. 저는 장바구니 파트를 맡아서 여러 개의 벡터와 결제 파트를 구현하는데 온 힘을 쏟았습니다.

[절정]

하지만 프로젝트 마감 직전 주 금요일에 재학생과 졸업생의 수입을 연동할 수 없는 오류를 발견하였습니다. 해결할 방법은 졸업생용 관리자 클래스를 새로 만들거나 졸업생의 장바구니 클래스와 로그인 클래스를 삭제하는 것이었습니다. 팀원들 모두 며칠동안 코드만 보느라 지치고 예민한 상태였음에도 금요일날 카페에서 다같이 회의하며 해결책을 구했습니다. 결국 마감이 얼마 안남은 시점에 코드를 반쯤 잘라내고 수정을 거듭했습니다. 열심히 구현한 코드가 날아가 버리는 것이 아쉬웠지만, 최종의 목표를 위해 합리적인 결단을 내린 것이라고 생각했습니다. 그리고 또 하나의 복병은 예외처리였습니다. 입력을 받는 부분마다 예외처리를 해주어야 했고, 단순히 Exception만 사용하면 에러가 발생할 때마다 초기화면으로 돌아가버렸습니다.

[결말]

결국 주말에도 팀원들은 쉴 수 없었고, 계속해서 코드 수정과, 두더지 잡기처럼 어디선가 계속 나타나는 오류를 잡아내는데 시간을 보냈습니다. 모두가 주말에 힘 쓴 결과 부족한 부분은 있지만 거의 최종에 가까운 목표물을 산출할 수 있었습니다.

[배운점]

1. 수업을 들으면서 배웠던 것들을 사실 완전히 나의 것을 소화하지 못했었는데, 프로젝트를 하면서 실제로 프로그램을 구현하면서 배웠던 것들을 적용함으로써 저의 것을 소화할 수 있었습니다.
2. 팀 프로젝트를 하며 가장 중요한 점은 팀원 간의 대화, 정보 교류라는 것을 깨달았습니다. 특히 코드를 보며 수정할 부분에 대해서 정확하게 팀원들과 공유해야만 나중에 큰 오류가 없음을 경험할 수 있었습니다.
3. Open API를 통해 수업시간에 배우지 않았던 메소드들을 활용하면서 더 다양한 기능들을 사용할 수 있어서 뿌듯했습니다.
4. 데이터베이스를 다룰 자료구조로 무엇이 좋을까 고민하고 공부하면서 수업시간에는 잘 확립되지 않았던 개념들이 정립될 수 있었습니다. 이를 통해 수업은 듣는 것 만으로는 절대 소화될 수 없고, 직접 복습을 해야만 나의 것으로 만들고 이해할 수 있음을 다시금 느꼈습니다.
5. 프로그램을 구현하면서 반복되는 패턴을 코드로 짤 때가 많았는데, 이를 통해 이제는 구현해야 할 조건들을 보면 어떤 알고리즘으로 짜야하는지 감이 잡히게 되었습니다.
6. 세미 프로젝트를 하면서 4조 뿐만 아니라 다른 조의 코드도 읽어보고 설명을 들으면서, 낯선 코드도 이제는 차분히 읽으면서 알고리즘을 파악하고 읽을 수 있게 되었습니다.
7. 나중에 예외처리를 하면서 느낀 점은 처음 코드 짤 때 미리 생각하고 예외처리를 하자입니다.