

소프트웨어를 만든다는 것



과 목 명: 소프트웨어프로젝트 I

지도교수: 송영록 교수님

학 과: 소프트웨어학부

학 번: 20213015

이 름: 송규원

제 출 일: 2021.06.04

1. 느낀 점

먼저, 나는 평소에 ‘소프트웨어를 개발한다는 것’이 단순히 코딩 작업을 핵심으로 나머지 추가적인 요소가 몇 가지 요구되는 것 그뿐이라고 생각했다. 그러나 본 강의 영상을 시청 후, 소프트웨어 개발은 내 생각보다 훨씬 다양하고 복잡한 과정을 요구하며 그 과정마다 꽤 많은 인력을 필요로 한다는 점이 인상 깊었다. 또한 소프트웨어를 개발한다는 것이 단순히 코딩 하나로 정의되는 것이 아닌 기획, 설계, 구현, 테스트 등과 같은 전 과정을 포함한다는 것을 알게 되었고, 추가적으로 이 과정이 전부가 아닌 유지 보수, 기술 지원, 기술 문서, 교육 등과 같은 세부적인 사항들을 필요로 한다는 점이 소프트웨어 개발을 단순하게 생각하던 기존의 내 모습을 되돌아보게 했다.

나는 소프트웨어를 전공하고 있지만 개발자 외엔 진출 분야에 대해 뚜렷이 알고 있는 게 없어 진로에 대한 고민이 꽤 많은 상태였는데, 소프트웨어 개발에 있어 각 단계마다 필요로 하는 기획자, 설계자, 테스터, 운영자 등의 역할과 각 역할들이 어떤 일을 수행하는지 강의 영상을 통해 알게 된 덕분에 진로에 대한 방향성을 잡아갈 지표가 생긴 좋은 기회가 되었다고 느꼈다.

기획, 설계, 구현, 테스트 과정을 거쳐 탄생한 소프트웨어가 장시간을 걸쳐 오랫동안 유지될 수 있는 비결은 성능 향상 덕분일 것이라고 두 번째 강의 영상 도입부를 보며 추측하곤 했는데, 이후 강의 영상에서 성능 향상을 포함한 형상 관리 과정과 목적이 상세히 설명되어 있었다. 이 덕분에 지속 가능한 소프트웨어를 개발하고 발전시키는 데에 있어 어떠한 기술적 요구 사항들이 부여되는지 알 수 있었고 이를 통해 훗날 내가 개발자 또는 개발에 이바지할 수 있는 사람이 되었을 때의 상황들과 그 상황들 속에서 내가 어떠한 행동을 취할지 강의를 보며 스스로 시뮬레이션 할 수 있던 점이 인상 깊었다.

2. 흥미로운 점

최근 진행한 AD 프로젝트를 수행하면서 앱 개발을 위한 기획 및 설계를 하는 것 자체가 생소한 탓에 기술적인 부분에서 특히 어려움을 많이 겪었고 기획 및 설계를 진행하는 매 순간 한계 상황을 마주하기 마련이었다. 그럴 때마

다 강의에서 다른 소프트웨어 개발 과정을 참고하며 차근차근 해결할 수 있던 점이 흥미로웠고, 한계 상황을 마주할 때 이를 피하려고 하기보다는 문제 상황으로 가정하고 성능 향상과 연관 지어 결과 및 결론을 도출하는 과정에서 버전 관리, 이슈 관리, 통합 빌드를 이용할 수 있던 점이 흥미로웠다.

또한 버전 관리에서 각 버전을 표기하는 숫자에 대한 설명을 강의에서 자세히 들을 수 있었는데, 일상생활에서 앱 또는 소프트웨어 업데이트 사항이 있을 때 이전에는 큰 관심 갖지 않았을 텐데 이제는 각 자리의 숫자가 어떤 변경 사항을 의미하는지 먼저 관심 가지고 바라보게 되었고 이러한 작은 변화들이 몸소 느껴지는 것이 새로웠다.

이슈 관리에서는 소프트웨어의 비가시성에 의해 변경 사항이 요구되는 경우 (Bug 발생, 새로운 기능 추가, 기존 기능 Update, 기술 변화 반영 등)마다 이를 어떻게 적용해야 하는지에 대해 상세히 알 수 있었다. 이전에는 소프트웨어 개발 후 발생하는 요구 사항을 모두 반영하는 줄만 알고 있었다면 이제는 요구 사항들을 경우에 따라 분류하고 각 경우마다 어떤 조치를 취해야 하는지에 대한 지표를 잡아갈 수 있는 점이 흥미롭게 느껴졌다.

통합 빌드에서는 통합 개발 환경 구축을 핵심으로 이슈 등록 후 로컬 저장소, 원격 저장소, 또 다른 서버 간의 컴파일 단계를 예시로 들어 컴파일 단계에서 문제 발생 유무에 따른 적절한 조치를 배울 수 있었다. 이슈 관리에서 모든 변경 사항이 반영되고 그 단계를 끝으로 세상 밖으로 배포할 수 있다고 생각하고 있었는데, 통합 빌드와 관련된 내용을 배우면서 소프트웨어 개발은 내가 생각했던 것보다 훨씬 복잡한 과정을 거친다는 것을 다시 한번 알게 되어 소프트웨어를 개발한다는 것 자체의 매력을 더 느끼는 계기가 되었다.

3. 세부적인 고찰

소프트웨어 개발에 대한 깊이 있는 내용을 학습할 수 있던 덕에 단순히 배우는 단계에서 끝마치기보다는 미래와 연관 지어볼 수 있었고, 훗날 내가 이 분야에서 어떤 역할을 수행할 수 있는지에 대한 기대감을 갖게 했다. 소프트웨어를 개발한다는 것을 새로이 정의할 수 있었고, 특히 개발 후 소프트웨어

발전 과정과 원리에 대해 자세히 배움으로써 소프트웨어는 개발하는 것이 다가 아닌 거란 걸 몸소 느끼는 계기가 되었다. 다양한 경우와 예시 사례들을 참고하여 내가 겪을 문제 상황을 시뮬레이션 할 수 있었으며, 실제로 최근 진행한 AD 프로젝트와 연관 지어 내가 기획 및 설계한 앱의 한계 상황과 성능 향상 논의에 있어 배운 내용을 참고하고 적용할 수 있었다.