

1 시작하기

개발을 알지 못하는 당신이 웹 개발을 시작한다면, 어디서부터 무얼 공부해야 할지라는 주제의 글입니다. 감히 누구도 편하게 얘기하기 어려운 주제입니다. 무능한 저 한 개인이 올바른 가이드를 제시해 드릴 수 없는 일입니다만, 무책임하게나마 감히 적어보겠습니다. 너무 신뢰하지 마시고 가벼이 읽어 주시고, 이렇게 생각하는 사람도 있구나 정도로 넘기시면 좋은 주제입니다.

간혹 주변에서 본격 웹 개발자가 되고 싶거나, 아니면 어떤 필요에 따라 웹 개발이라는 분야에 도전해 보려는 분들이 계신데, 마땅히 추천해 드릴 자료나 가이드가 너무 부족하다는 생각이 들었습니다. 그도 그럴 것이, 아직 웹 개발이라는 분야가 빠르게 성장하고 있고, 어제는 촉망받던 기술이 오늘의 레거시가 되버리는 상황이 반복되고 있어서, 정작 전업 개발자들조차 따라가고 배우기 벅찬 상황입니다. 뭐 하나 배워 놓으면 금세 쓰레기 지식이 되고 마는 거죠. 게다가 뭐 하나의 한 분야의 같은 일을 하는 데도 다양한 기술과 방법이 술하게 쏟아지면서 서로가 좋은 방법이라며 싸움에 가까운 토론을 벌이고 있어서 혼란스럽습니다.

이런 상황에, 누군가 새로이 웹 개발에 도전해서 무언가 간단하게나마 만들고 싶다는 생각이 있다 하더라도, 무언가를 공부하고 있으면, 다른 누군가가 나타나서, 그게 아니라 이 방법이나 이 기술을 써야 한다며 혼수를 두는데, 잘 모르는 입장에서는 팔랑귀가 펄럭여서 새로 처음부터 시작하게 되기도 합니다.

맞습니다. 더 새로운 기술과 더 나은 도구들이 분명히 있습니다. 그런데, 개발자마다 성향과 상황과 또 그들의 과거 경험에 따른 선호도가 너무도 다르고 다양해서, 잘 정리된 가이드가 나오기는 힘들어 보입니다. 지금도 없었고, 앞으로도 없을 것입니다. 이 글 역시 그런 충분한 가이드일 수는 없습니다.

상황이 이러니, 이런 방법으로 도전해 보면 어떨까요? 우선 무언가 직접 만들어서 개발한 웹서비스를 온라인에 공개해 보는 겁니다. 아무도 안 쓰더라도 상관없습니다. 나 혼자 들어가 보고, 친구들에게 링크를 보내줍니다. 들어와 보고 "이게 뭐냐?" 라고 의아해하며 그냥 나가더라도 상관없습니다. 한번 만들어나 봅시다. 우아하고 최신의 기술이 아니어도 괜찮습니다. 일단 무언가 보이게 해 봅시다.

2 목표

이 글의 목표는 공부해야 할 주제와 키워드를 제시하는 것입니다. 그 상세 내용은 직접 찾아보며 하셔야겠지만, 전체 적인 계획과 꾸준한 노력을 투자할 기준점을 하나 세워드리는데 이것이 목표입니다. 취직이나 훌륭한 개발자가 되는 것이 목표도 아닙니다. 단지 내가 공부해서 웹서비스 하나 만들어 보는 것이 목표입니다. 안 그래도 어려운 개발, 생소한 개발, 목표를 간단하게 잡아 봅시다. 거창한 대규모의 웹서비스 아닙니다. 깃해야 백 명 쓸까 말까 한 서비스, 간단한 서비스입니다.

미리 말씀드리자면, 아무리 최대한 간단하고 얇게만 파고들더라도 긴 시간이 필요합니다. 일 투자 시간에 따라 다르겠지만, 수개월에서 수년이 넘게 걸릴 수도 있습니다. 이 글의 내용을 다 배우더라도 투자하는 노력에 비해 당장 눈에 보이는 결과가 보잘것없을 겁니다. 그럼에도 불구하고 시작해보겠다는 의지가 있으시다면 계속 읽어주시기 바랍니다.

3 제공하는 가치

개발 전체 과정에 있어 필요한 주제를 최소한 한 가지씩 꼽아드리겠습니다. 많아 봐야 두세 가지 중 하나를 택하면 되게끔 알려드리겠습니다. 접근하지 말아야 할 주제도 적어드리겠습니다. 시간을 낭비하는 함정에 빠질 일을 막아드리겠습니다. 당신이 뛰어야 할 깊은 말이라면, 양 눈 옆에 가리개를 가려드리겠습니다. 앞만 보고 뛰세요. 최단코스를 찾는라 고민하지 마세요. 우리의 목표는 갈팡질팡하지 않고 목적지를 향해 가는 겁니다, 조금 돌아가더라도요.

4 제공하지 않는 가치

각 세세한 내용의 설명은 드리지 않습니다. 직접 해당 키워드로 찾아보시고 공부하셔야 합니다. 대부분의 내용은 이고잉님의 오픈튜토리얼스를 참고하시면 좋은 자료가 많을 것으로 생각합니다.

각 분야의 최고의 선택을 제시하지도 않습니다. 그저 되는 것 중에서, 제 개인이 선호하는 방법을 알려드립니다. 다른 개발자와의 의견은 얼마든지 다를 수 있습니다. 선택의 폭을 좁히는 것이 목표이지 최고 효율의 방법을 알려드리는 것이 목표는 아닙니다. 미리 귀뜸하자면, 이게 최선의 방법이라고 주장하는 사람의 말을 듣지 마시라는 겁니다. 거짓말장이이거나 아직 무지한 사람일 가능

성이 높습니다. 왜냐하면 이 바닥에서 최선의 방법을 제시하기는 꽤 어렵거든요.

5 공부해야 할 주제

정해진 카테고리에는 아님니다만, 대략 이렇게 나눠서 공략해 봅시다. 배우다 보면 한 주제에 대해서도 다양한 방법들이 준비합니다. 언젠가 더 진지해진다면 더 깊게 다뤄야 하겠지만, 지금은 한눈팔지 말고 아주 얇게만 파기로 합니다. 누가 뭐라고 해도, 여기 정리한 선택만 하고, 나머지 선택은 외면해 버립시다. 이 주제들을 다 훑고 나면, 그 후에 누가 좋다고 한 기술을 더 알아보기로 합니다.

* 프론트엔드 (front-end)- 이용자의 웹브라우저에서 직접적으로 보이는 부분을 다루는 기술들 * 백엔드 (back-end) - 웹브라우저가 주고받는 데이터를 기록하고 가져오는 등의 뒷 단의 곳을 일을 처리하는 기술들 * 데이터베이스 (database) - 백엔드가 다루는 데이터를 보관하고 검색해 오는 기술 * 네트워크 (network) - 각 컴퓨터 사이의 데이터 통신에 필요한 기술들 * 에디터/툴/버전 관리 시스템 (tools) - 각 주제 개발 작업에 필요한 도구들 선택과 활용 * 기초 자료 구조 (data structure) - 프론트엔드나 백엔드 프로그래밍 모드에 쓰이는 기초 지식 * 리눅스/도커/AWS - 백엔드와 데이터베이스를 운영할 기술과 환경

아직 이 주제들이 무얼 의미하는지 몰라도 됩니다. 오늘 글에서는 이 주제들이 무얼 의미하고 어떤 걸 키워드로 전해드릴 지만 적어보겠습니다.

6 프론트엔드

7 백엔드

8 데이터베이스

9 네트워크

10 텍스트 에디터

11 통합 개발 환경

12 버전 관리 시스템

13 기초 자료 구조

Hello world!