

20231593 임수호 미디어디자인 소감

작업을 하면서 일상에서 마주칠 만한 것들을 구현하는 것에 초점을 맞추었습니다. 그래서 포춘쿠키를 처음으로 만들었습니다. `math.random`을 활용해서 단순하게 100개의 무작위 운세가 나오는 방식으로 제작했습니다. 이걸 만들어보고 나니, 클릭하면 실제 시간 기록 로그를 남기는 것도 가능하지 않을까? 싶어졌습니다. 그래서 `toLocaleTimeString` 기능을 검색을 통해 알게 되었고, 이를 한 번 활용해 보았습니다.

그 후에는 `math.random`과 간단한 연산자를 활용해서 주사위의 결과 값을 비교하는 기능도 만들어 보았습니다. 마찬가지로 비슷한 방식으로 저녁 메뉴를 추천해주는 프로그램을 만들었습니다.

이후 부터가 조금 재밌어지는데, hex값이 16진법이라는 것을 활용해서 6개의 자릿수를 모두 랜덤 16진수를 입력하는 코드를 만들어 컬러팔레트를 랜덤으로 생성해주는 코드를 제작했습니다. 또한, `math.random`과 `touch`를 통한 감산 기능을 결합해서, 과거 한 때 유행했었던 클리커류의 게임을 재현해보았습니다. 그리고 이번에는 js와 css를 연결하는 것을 시험해보고 싶어 버튼을 클릭할 때 마다 전구와 주변 밝기가 달라지는 전등 프로그램을 제작해보았습니다.

그 후에는 60진수를 활용해서 자동 비밀번호를 생성하는 프로그램을 만드려고 했으나, `math` 기능에 60진수로 변환해주는 기능이 없었기에, gpt의 힘을 빌려 60진수로 강제 변환하는 코드를 얻고 이를 바탕으로 n자릿수 비밀번호를 생성하는 코드를 제작했습니다. 그 후, XY좌표를 활용하면 재밌는 것을 만들 수 있지 않을까 싶어 버튼을 누르면 임의의 좌표값에 점이 찍히는 기능을 `.style`을 활용해 제작했습니다. 해당 작업에서 사용한 방식은 후에 더 다양한 버튼 상호작용으로 확장되었습니다. 잠시 쉬어가는 용도로 기존에 수업시간에 제작했던 랜덤 뽑기 시스템을 개선했습니다. 기존에 뽑기 횟수같은 것들이 보이지 않는다는 점을 개선했습니다. 그 후 카드 덱을 생성하고 카드를 뽑는 코드를 제작했습니다. 원하는 장수만큼 뽑고, 덱에는 정확한 카드를 남기는 과정에서 제가 뜻하는 대로 되지 않는 것들이 많았고, 이 부분을 gpt의 힘을 빌렸습니다. 그 후에는 실시간으로 상호작용이 눈에 보이는 버튼을 만들고 싶어서, 연타함에 따라 게이지가 변하는 버튼을 만들었습니다. 1틱을 정의함으로 소위 게임에서 이야기하는 시간성을 확보했고, 상단 바의 width를 조정하는 방식으로 바가 늘었다가 줄었다가 하는 것을 표현했습니다.

이후 제작하게 된 것은 이전에 다른 분이 하셨던 gpt기반 css 작업이 너무 감명 깊어서, 저도 한 번 gpt에게 css를 맡겨보았던 작업입니다. 또한, 가장 최근 술자리에서 했던 지하철 게임이 너무 머릿속을 맴돌기도 했고, 지하철 게임을 혼자 연습해볼 수 있으면 어떨까 하는 생각에 좀 무리수를 두었습니다. input이라는 것을 알게 되었고, 이를 array랑 연결지어서 제작했습니다. 중간에 워낙 문제가 많이 발생했던 탓에 gpt로 디버깅도 많이 진행했고, 이때부터 조금 구조적으로 코딩하려고 노력했습니다. 직전 작업에서 너무 고생했던 탓에, 이번 작업은 보다 쉬운 작업으로 진행하고자 했습니다. 앞에서 진행했던 무작위 위치에 점이 찍히는 버튼을 응용해서 버튼 자체가 무한으로 증식하는 연출을 해보았습니다.

이 이후에는 일종의 게임을 만들어보고 싶다는 욕구가 강해져서, 뼈대가 되는 로직을 코딩한 후, gpt의 도움을 얻어 제가 예상하지 못했던 변수나 오류에 대한 컨트롤과 작동의 유연성을 강화했습니다. 그 첫번째는 아이템 강화하기 게임입니다. `math.random`과 `if`를 활용해서 후반으로 갈수록 강화에 성공하기 어렵게 만들어보았습니다. 그 다음 작업은 정확하게 10초를 맞추면 경품을 주는 영상을 보고 영감을 받아 제작해보았습니다. `performance.now`를 통해 정확하게 시간을 받고 연산자를 통해 오차값을 받고 이를 절대값으로 치환해서 그 차이에 따라 코멘트가 나오는 방식으로 제작했습니다. 그 다음으로는 누구나 한 번쯤 해봤을 통에 칼을 꼽는 해적게임을 만들어보았습니다. 랜덤으로 24개의 구멍 중 하나에 해적 깃발이 위로 올라가는 효과를 발동시키는 버튼 트리거가 배당될 수 있도록 코딩했습니다. 그 후에는 간단한 사칙 연산 퀴즈를 만들어보았습니다 css를 이용해서 시험지같은 톤으로 제작했고, 오답의 숫자가 정답과 너무 달라지는 것을 방지하는 코드를 gpt를 이용해 추가했습니다.

이 이후 작업부터는 보다 더 게임성을 띄는 작업을 준비해보았습니다. 먼저 첫번째는 슬롯머신입니다. 원래는 `math.random`을 활용하고자 했으나, 해당 방법으로 원통형의 회전하는 슬롯머신을 애니메이션하는 데에 한계를 느끼고 함수의 최대 최솟값을 활용해서 회전량을 결정하는 방식으로 제작했습니다. 그리고 gpt를 이용해서 사용자가 한번 누르고 결과가 나올 때까지 버튼을 반복해서 누르는 것을 방지하는 코드를 제작했습니다. 다음 작업은 두더지게임 또한 `math.random`으로 할 수 있지 않을까 해서 제작해보았습니다. 그러나 `math.random`으로 올라올 두더지에게 값을 할당하고 올라오게 하는 것까지는 성공했으나, 이후 단계에서 막혀 gpt를 통해 `classList`를 사용하는 법을 알게 되었습니다.

이후에는 제가 야구를 좋아해서 야구 관련 게임을 만들어 보고 싶었기에, 간단한 타격 시뮬레이션을 제작했습니다. `math.random`과 `if`를 활용해서 계속해서 하위 확률을 집어넣는 방식으로 타격매커니즘을 구현하고, 타율,출루율,장타율,ops 계산식을 구해서 집어넣어 구현했습니다. 버튼 레이아웃과 야구장 그림 배치가 원하는 대로 잘 되지 않아 gpt의 도움을 받아가며 진행했습니다.

그 다음은 아이디어가 살짝 고갈되어서 gpt에게 혹시 내가 전혀 모르고 있는 것 같아보이는 js기능 몇가지를 알려줄 수 있냐는 질문에서 출발한 작업입니다. gpt는 저에게 무언가를 기울이는 기능을 활용해 보라고 했고, 저는 이에 맞춰서 버튼을 누르면 바닥면이 기울어지고 이에따라 공이 굴러가는 연출을 하는 작업을 만들어 보기로 했습니다. 일종의 물리법칙을 만들어야하는 상황이었으나 저는 물리에 대해서는 전혀 몰랐기에 이러한 계산식과 값 설정에 있어서는 gpt에게 맡기고 저는 이렇게 얻은 값을 코드의 형태로 재구성하는 데에 힘을 썼습니다. 제 기억으로 이쯤부터인가 자꾸 화면 크기에 맞지 않는 문제가 발생해서 이에대한 디버깅을 지속적으로 gpt에게 부탁했었습니다.

다음은 앞선 작업에서 사용된 좌표나 좌표기반 이동, 그리고 확률적 요소를 병합해 인형뽑기 기능을 제작해보았습니다.

그 뒤에는 카드 뽑기를 만들었던 것을 기반으로 해서 짝맞추기 게임을 제작했습니다. 덱형성 부터 덱에서 뽑은 카드를 그리드를 통해 배치하는 것까진 순조로웠으나, 카드를 다시 자동으로 뒤집히게 하는 코드를 어찌해야할 바를 몰라 gpt에게 맡겼습니다.

그 뒤엔 컴퓨터가 랜덤으로 시범을 보이는 암호를 암기해서 따라누르는 프로그램을 제작했습니다. 제작 과정에서 암호를 정확히 단계별로 맞춰서 불이 들어오게 하기 위해서는 `promise`를 써야한다는 gpt의 피드백을 보고 사용해 보았습니다.

그 후에는 진짜 게임스러운 작업을 해보고 싶어졌습니다. 그래서 햄버거 만들기라는 게임을 제작해보았습니다. 햄버거 만들기는 윗빵과 아랫빵은 고정된 상태에서 랜덤한 재료가 사이에 끼어들어가는 구조로 제작했습니다. 햄버거를 완성했을 때 결과창이 나오길 바랐는데 이 부분은 직접 처음부터 끝까지 구현하기 어려워서 gpt의 도움을 받았습니다.

그 다음은 조금 다른 작업이긴 했는데, gpt로부터 js html css가 있으면 내부에 svg를 생성할 수 있다는 점을 알게되었습니다. 이를 응용하고 싶어져서, + - 버튼을 누르면 점이 추가되어 도형 형태가 바뀌는 간단한 코드를 제작해보았습니다.

마지막 3개 작업은 해보고 싶은게 명확했기에 다소 gpt를 많이 사용했습니다. 먼저 첫번째는 공 굴리는 것을 넘어서 물리엔진을 만들어보고 싶다는 생각이 들어 저울을 제작했습니다. gpt를 많이 사용했음에도 불구하고 수많은 버그가 발생해서 정말 오랜 시간동안 제작했습니다.

그 다음은 베스킨 라빈스 31 게임입니다. 작업 자체는 어렵지 않았는데, 상대역을 맡는 컴퓨터가 특정 전략을 수행하도록 로직을 만드는 것이 어려웠습니다. 그래서 컴퓨터의 로직을 제작하는 부분에서는 gpt를 사용했습니다.

마지막은 리듬게임입니다. 틱단위로 움직이던 연타버튼과, 애니메이션 효과 그리고 햄버거에서의 정답 판정 기능까지 고려해보았을 때 충분히 만들 수 있겠다는 판단이 들었고, 제가 뼈대를 짠 것을 바탕으로 gpt를 이용해서 확장했습니다. 모바일 환경에서 여러 문제가 발생해서 이를 수정하는데 정말 많은 시간을 들였습니다.

그리고 추가로 오직 gpt만 사용해서 지금까지 사용한 모든 개념을 응용한 온전한 게임을 하나 넣어보았습니다.

과제의 취지에 온전히 부합할진 모르겠으나 이런 것도 되네? 그럼 이것도? 그럼 저것도? 하면서 굉장히 흥미롭게 작업을 진행했습니다. 이제 코딩과 조금은 친숙해 지지 않았나 생각도 들었습니다.