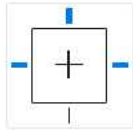


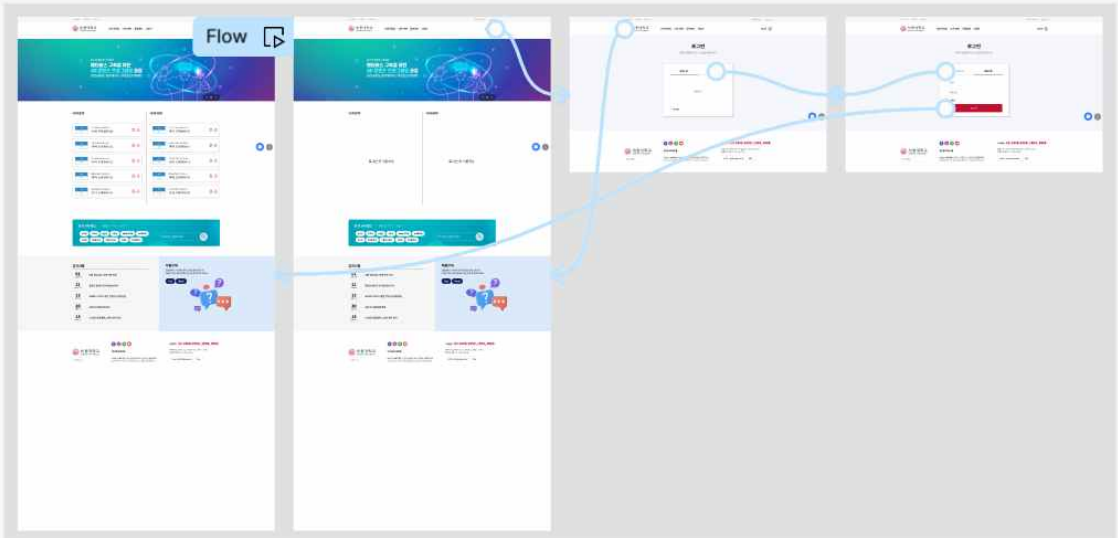


2024-1학기 창의학기제 주간학습보고서 (5주차)

창의과제	세종대학교 집현캠퍼스를 개선시킨 웹서비스 개발				
이름	박수진	학습기간	5/1~5/9		
학번	21011998	학습주차	5주차	학습시간	6
학과(전공)	인공지능학과	과목명	자기주도창의전공 I	수강학점	3
※ 수강학점에 따른 회차별 학습시간 및 10주차 이상 학습 준수					
금주 학습목표	제작한 페이지에 Constraint 적용, 로그인 페이지 UI 제작, 움직이는 프로토타입 제작				
학습내용	<div>1) 피그마 기능 학습</div> <ul style="list-style-type: none">Constraints : 디자인 요소의 크기와 위치를 제어하는 기능, 요소의 크기나 위치를 제한해서 다른 요소의 상대적인 위치나 크기를 유지할 수 있도록 함, 각각의 Constraints는 요소가 가로 또는 세로 방향으로 어떻게 동작하는지 결정함. 이 기능을 잘 활용하면 다양한 화면의 크기와 해상도에서 일관성있도록 효율적인 작업이 가능함. 좌, 우, 위, 아래에 적용을 할 수 있음, center에 유지시키거나 scale이 유지되도록 할 수 있음. <div>Constraints</div> <div></div> <ul style="list-style-type: none">Constraints 적용 예시 <p>파란색 직사각형에 Constraints가 left만 적용된 경우 프레임(회색 직사각형)의 크기가 줄어들거나 늘어날 때 파란색 직사각형은 그대로 이고 오른쪽 여백이 달라지는 것을 볼 수 있다. 그러나 Constraints를 left and right로 적용하면 프레임이 변해도 좌,우의 여백이 유지되면서 파란색 직사각형의 크기가 변하는 것을 알 수 있다. 이 기능을 활용하면 반응형 웹을 만들 수 있다.</p> <div><div>Constraints 좌 적용</div><div>Constraints 좌,우 적용</div></div> <div><div><div>원래 크기</div><div>크기 줄었을 때</div><div>크기 늘렸을 때</div></div><div><div>원래 크기</div><div>크기 줄었을 때</div><div>크기 늘렸을 때</div></div></div> <ul style="list-style-type: none">컴포넌트 : 반복적으로 사용되는 요소들을 컴포넌트화하여 효율적으로 작업할 수 있다.				



	<p>컴포넌트화 하고자 하는 요소를 클릭하고 Create component를 클릭하면 컴포넌트가 된다. 생성된 컴포넌트는 마스터 컴포넌트인데 마스터 컴포넌트를 복제하면 인스턴스 컴포넌트가 생성된다. 마스터 컴포넌트를 변경하면 인스턴스도 같이 변경된다. 여러 개의 요소를 한 번에 컴포넌트로 등록 할 수 있음 (전체 선택 후 Create multiple components 클릭). 컴포넌트 안에 컴포넌트를 넣을 수 있음. 컴포넌트를 활용하면 같은 요소 여러 개를 변경할 때 효과적이다.</p> <ul style="list-style-type: none"> • 컴포넌트 활용 방법 <p>특정 요소를 컴포넌트화해서 사용한다. 컴포넌트 하나만 변경하면 해당 컴포넌트가 활용된 모든 페이지에 한 번에 변경 사항이 적용된다.</p> <p>2) UI 제작</p> <p>메인 페이지를 제작하기 위해 집현캠퍼스의 UI를 가져온 뒤 이를 기반으로 페이지를 제작했다. 그러나 이 방식을 사용했더니 계속 제작을 진행할수록 문제점들이 발생하기 시작했다. 기존 집현캠퍼스의 UI를 가지고 올 때 완전히 완벽하게 가지고 오는 것이 아니기 때문에 프레임이 지저분하게 설정이 되어 있었다. 프레임이 잘 설정이 되어있지 않으면 작업을 할 때도 효율적이지 않으며 Auto Layout이나 Constraint와 같은 기능들을 적용하기도 어렵다. 그래서 기존에 설정된 프레임 중에 불필요한 것들은 지우고 변경이 필요한 것들은 크기나 위치를 조절하여 프레임을 깔끔하게 바꾸었다.</p> <p>다음으로는 제작한 메인 페이지의 UI에 Constraint를 적용했다. 이는 각 요소의 폭을 자유롭게 조정하고 반응형 처리를 위해 필요한 작업이다. Nav Bar의 경우 항상 위쪽에 붙어있어야 해서 Constraint를 Top으로 설정하였다. 인기 키워드를 보여주는 부분의 경우 화면 크기가 변해도 좌우로 여백이 동일하게 움직여야 해서 Left and right로 설정하였다. 그리고 나머지 요소들은 대부분 Top, Left 혹은 Center로 설정했다.</p> <p>로그인 전의 페이지에서 로그인 후의 페이지로 가기위한 로그인 페이지를 제작했다. 기존의 로그인 방식은 세종포탈 아이디와 비밀번호로 로그인 하는 방법과 일반/이메일로 로그인하는 방법 이렇게 두 가지가 있었다. 이 중에서 세종포탈로 로그인 하는 방식은 그대로 유지하고 일반/이메일 로그인 방식 대신 얼굴인식을 통한 Face ID로 로그인 하는 방식을 추가하였다. Face ID로 로그인 하는 창에는 얼굴인식을 하여 인식한 얼굴이 보이도록 하는 창을 만들었다. Face ID의 경우 인식에 성공하면 자동으로 로그인이 되므로 로그인 버튼은 따로 추가하지 않았다. 그리고 세종포탈로 로그인하는 창에는 기존과 동일하게 아이디와 비밀번호를 입력할 수 있는 상자와 로그인 버튼을 만들었다. 이렇게 만든 로그인 하는 부분은 화면의 가운데에 위치시키고 위에는 메인 페이지의 Nav Bar를 그대로 유지하고 아래에는 메인 페이지의 footer를 그대로 유지하여 모든 페이지에 통일성을 주었다.</p>
학습방법	<p>유튜브에 올라온 피그마 기초에 대한 강의 영상을 시청하여 피그마를 다루기위해 필요한 개념들과 기본 기능들을 익혔다. 강의를 통해 배운 개념과 기능들을 적용하여 UI를 제작했다. 추가로 더 자세한 사항이 궁금한 것들은 책을 찾아보며 학습했다.</p>
학습성과 및 목표달성도	<ul style="list-style-type: none"> • 메인 페이지에 Constraint 적용 완료 • 로그인 페이지 UI (Face ID 로그인, 세종포탈 로그인) 제작 완료 • 움직이는 프로토타입 제작 완료

	<div> <div>로그인-2-1</div> <div>로그인-2-2</div> <div>  </div> </div>
<p>참고자료 및 문헌</p>	<ul style="list-style-type: none"> • 실전 피그마 (UI/UX 디자이너, 서비스 기획자를 위한 실무 입문서) • UX/UI 디자이너를 위한 실무 피그마 (디자인 시스템에서 개발 전달까지)
<p>내주 계획</p>	<ul style="list-style-type: none"> • 디자인 한 페이지를 html과 CSS로 구현 1 <ul style="list-style-type: none"> - 로그인 전 페이지 - 로그인 페이지

년 월 일

지도교수

(인)