

화면 설계

사용자 인터페이스

❖ User Interface(사용자 인터페이스 - UI)

- ✓ 사용자와 시스템 간의 상호작용이 원활 하게 이루어지도록 도와주는 장치나 소프트웨어
- ✓ UI가 필요한 이유
 - 막연한 작업 기능에 대해 구체적인 방법을 제시
 - 사용자의 편의성을 높임으로써 작업 시간 단축과 업무에 대한 이해도를 높임
 - 정보 제공자와 공급자의 원활하고 쉬운 매개 역할을 수행
- ✓ UI의 세 가지 분야
 - 정보 제공과 기능 전달을 위한 물리적 제어 분야
 - 콘텐츠의 상세적 표현과 전체적 구성에 관한 분야
 - 사용자의 편의성에 맞춰 쉽고 간편하게 사용 가능하게 하는 기능적 분야
- ✓ UI 구분
 - CLI(Command Line Interface): 명령과 출력이 텍스트 형태로 이뤄지는 인터페이스
 - GUI(Graphical User Interface): 아이콘이나 메뉴를 마우스로 선택하여 작업을 수행하는 그래픽 환경의 인터페이스
 - NUI(Natural User Interface): 사용자의 말이나 행동으로 기기를 조작하는 인터페이스
- ✓ UI의 설계 원칙
 - 직관성: 누구나 쉽게 이해하고 사용할 수 있어야 한다.
 - 유효성: 사용자의 목적을 정확하게 달성하여야 한다.
 - 학습성: 누구나 쉽게 배우고 익힐 수 있어야 한다.
 - 유연성: 사용자의 요구사항을 최대한 수용하며, 오류를 최소화하여야 한다.

UI 표준 및 지침

❖ UI 표준 및 지침

- ✓ UI 표준: 전체 시스템에 포함된 모든 UI에 공통적으로 적용될 내용
- ✓ UI 지침: UI 개발 과정에서 반드시 지켜야 할 공통의 조건

❖ UI 스타일 가이드: 개발자나 디자이너 등이 UI를 작성할 때 기준이 되는 규칙들로 구동 환경, 레이아웃, 네비게이션 등을 정의

❖ UI 스타일 가이드 작성 순서

- ✓ 구동 환경을 정의
 - 컴퓨터 운영체제를 확인
 - 웹 브라우저를 확인
 - 모니터 해상도를 확인
 - 프레임 세트를 확인
- ✓ 레이아웃 정의
 - 화면 구조를 정의하고 각 영역의 메뉴를 구성 하는 단계
 - 레이아웃 영역에는 Top, Left, Contents, Footer Area가 있으며, 기본적으로 Top, Left, Contents 영역으로 구성
 - 상단 메뉴(Top Area) : 필수 영역으로 시스템 전체 페이지에 동일하게 적용
 - 좌측 메뉴(Left Area) : 선택 영역으로 시스템 별 서브 페이지에 선택적으로 적용
 - 내용 구성(Contents Area) : 필수 영역으로 시스템의 전체 콘셉트를 나타내는 메인 이미지와 시스템 별로 필요한 콘텐츠를 표시함
 - 하단 메뉴(Footer Area) : 선택 영역으로 회사 상황에 따라 표시 여부를 결정

소프트웨어 개발을 위한 UI 표준 및 지침

❖ UI 스타일 가이드 작성 순서

✓ 네비게이션 정의

- 네비게이션의 메뉴 타입을 선택하여 적용하는 단계
- 네비게이션은 사용자가 원하는 정보를 빠르게 찾을 수 있도록 안내하는 것으로 메뉴, 버튼, 링크 등으로 구성

✓ 기능 정의

- 시스템에 적용할 업무 과정에서 일어나는 모든 기능 정의 활동이나 필요한 데이터 간의 관계 등을 논리적인 모델로 상세화 하는 단계

✓ 구성 요소 정의

- 화면에 표시할 그리드나 버튼 등을 정의하는 단계



UI 설계 도구

❖ 와이어프레임 (Wireframe)

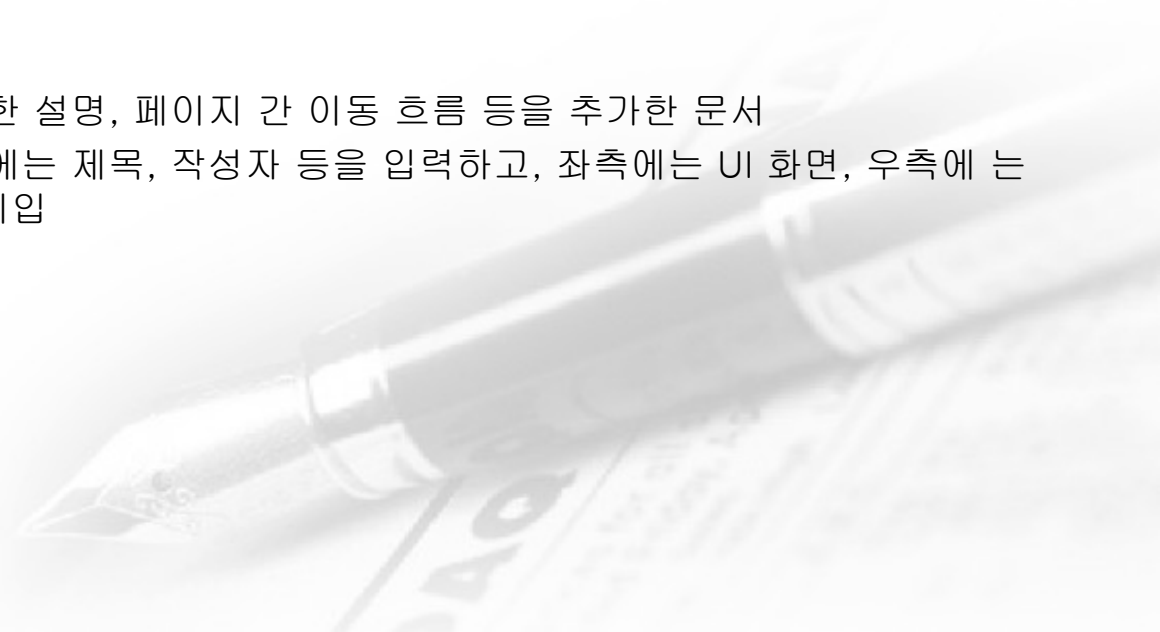
- ✓ 기획 단계의 초기에 제작하는 것으로, 페이지에 대한 개략적인 레이아웃이나 UI 요소 등에 대한 뼈대를 설계하는 단계
- ✓ 와이어프레임을 제작할 때는 각 페이지의 영역 구분, 콘텐츠, 텍스트 배치 등을 화면 단위로 설계

❖ 목업(Mockup)

- ✓ 디자인, 사용 방법 설명, 평가 등을 위해 와이어프레임보다 좀 더 실제 화면과 유사하게 만든 정적인 형태의 모형
- ✓ 시각적으로만 구성 요소를 배치하는 것으로 실제로 구현되지는 않는다.

❖ 스토리보드(Story Board)

- ✓ 와이어프레임에 콘텐츠에 대한 설명, 페이지 간 이동 흐름 등을 추가한 문서
- ✓ 스토리보드의 상단이나 우측에는 제목, 작성자 등을 입력하고, 좌측에는 UI 화면, 우측에는 디스크립션(Description)을 기입



UI 설계 도구

❖ 프로토타입(Prototype)

- ✓ 와이어프레임이나 스토리보드 등에 인터랙션을 적용함으로써 실제 구현된 것처럼 테스트가 가능한 동적인 형태의 모형
- ✓ 프로토타입은 사용성 테스트나 작업자 간 서비스 이해를 위해 작성하는 샘플
- ✓ 페이퍼 프로토타입: 아날로그적인 방법으로, 스케치, 그림, 글 등을 이용하여 손으로 직접 작성하는 방법
- ✓ 디지털 프로토타입: 파워포인트, 아크로벳, 비지오, 옴니그래플 등과 같은 프로그램을 사용하여 작성하는 방법

❖ 유스케이스(Use Case)

- ✓ 사용자 측면에서의 요구사항으로, 사용자가 원하는 목표를 달성하기 위해 수행할 내용을 기술한다.
- ✓ 사용자의 요구사항을 빠르게 파악함으로써 프로젝트의 초기에 시스템의 기능적인 요구를 결정하고 그 결과를 문서화 할 수 있다.



UI 요구사항 확인

- ❖ 새로 개발할 시스템에 적용할 UI 관련 요구사항을 조사해서 작성하는 단계
- ❖ 요구 사항 확인 순서
 - ✓ 목표 정의
 - 사용자들을 대상으로 인터뷰를 진행한 후 사용자들의 의견이 수렴된 비즈니스 요구사항을 정의
 - ✓ 활동사항 정의: 조사한 요구사항을 토대로 앞으로 해야 할 활동 사항을 정의
 - 여러 경로를 통해 수집된 사용자들의 요구 사항을 검토하고 분석하여 UI 개발 목적에 맞게 작성
 - ✓ UI 요구 사항 작성
 - UI 요구 사항을 UI 개발 목적에 맞게 실 사용자 중심으로 작성
- ❖ 요구 사항 작성 순서
 - ✓ 요구 사항 요소 확인: 요구 사항 요소의 종류와 각각의 표현 방식 등을 검토
 - ✓ 정황 시나리오 작성: 사용자가 목표를 달성하기 위해 수행하는 방법을 순차적으로 묘사
 - ✓ 요구 사항 작성: 정황 시나리오를 바탕으로 요구 사항 작성
- ❖ 요구 사항 요소
 - ✓ 데이터 요구: 사용자의 요구 모델과 객체들의 주요 특성을 기반으로 데이터 객체를 정리
 - ✓ 기능 요구: 목적 달성을 위해 무엇을 실행해야 하는지를 동사형으로 설명
 - ✓ 제품/서비스의 품질: 데이터 및 기능 요구 이외 품질, 서비스, 감성적인 품질 등을 고려
 - ✓ 제약 사항: 데드라인, 개발 비용, 시스템 준수에 필요한 규제 등

품질 요구 사항

❖ 품질 특성

- ✓ 기능성: 사용자의 요구 사항을 정확하게 만족하는 기능을 제공
- ✓ 신뢰성: 주어진 시간 동안 주어진 기능을 오류없이 수행할 수 있는 정도
- ✓ 사용성: 사용자와 컴퓨터 사이에 발생하는 어떠한 행위에 대하여 사용자가 정확히 이해하고 다시 사용하고 싶은 정도 - 사용자 중심
- ✓ 효율성: 사용자가 요구하는 기능을 얼마나 빠르게 처리할 수 있는지 정도
- ✓ 유지 보수성: 환경의 변화나 새로운 요구 사항이 발생했을 때 소프트웨어를 개선하거나 확장할 수 있는 정도
- ✓ 이식성: 다른 환경에서도 얼마나 쉽게 적용할 수 있는지 정도
- ✓ 일관성 : 버튼이나 조작 방법을 사용자가 기억하기 쉽고 빠른 습득이 가능하게 설계
- ✓ 단순성 : 조작 방법은 가장 간단하게 작동이 가능하도록 하여 인지적 부담을 감소
- ✓ 결과 예측 가능 : 작동시킬 기능만 보고도 결과 예측이 가능
- ✓ 가시성 : 주요 기능을 메인 화면에 노출하여 조작이 쉬워야 함
- ✓ 표준화 : 디자인을 표준화하여 기능 구조의 선행 학습 이후 쉽게 사용
- ✓ 접근성 : 사용자의 직무, 연령, 성별 등 다양한 계층을 수용
- ✓ 명확성 : 사용자가 개념적으로 쉽게 인지
- ✓ 오류 발생 해결 : 사용자가 오류에 대한 상황을 정확히 인지
- ✓ 유용성: 사용자가 시스템을 통해 원하는 목표를 얼마나 효과적으로 달성할 수 있는가에 대한 척도

UI 설계

- ❖ 설계서의 작성 순서
 - ✓ UI 설계서 표지
 - ✓ UI 설계서 개정 이력
 - ✓ UI 요구사항 정의서
 - ✓ 시스템 구조
 - ✓ 사이트 맵
 - ✓ 프로세스 정의서
 - ✓ 화면 설계



UI 설계

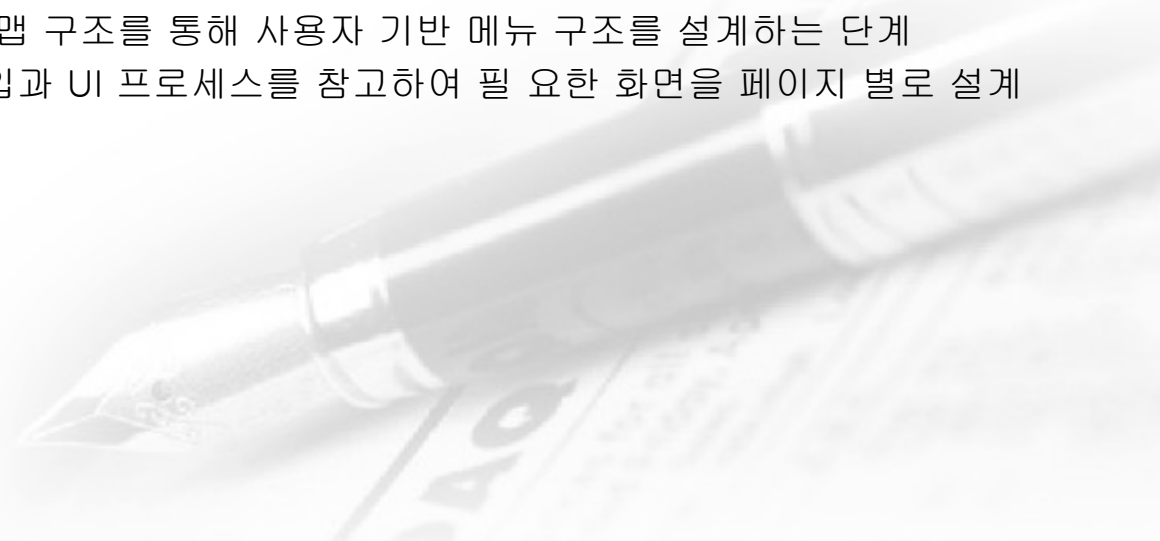
❖ UI 흐름 설계

- ✓ 업무의 진행과정이나 수행 절차에 따른 흐름을 파악하여 화면과 폼을 설계하는 단계
- ✓ UI 흐름 설계 순서
 - ❑ 기능 작성: 화면에 표현할 기능을 작성하는 단계
 - ❑ 입력 요소 확인: 화면에 표현되어야 할 기능을 확인한 후 화면에 입력할 요소를 확인하는 단계
 - ❑ Use Case 설계
 - UI 요구사항을 기반으로 UI UseCase 설계하는 단계
 - Use Case는 화면에 표현할 입력 요소들의 형태나 입력 방법, 배치 등을 고려해서 설계
 - ❑ 기능 및 양식 확인
 - 분석한 기능을 토대로 텍스트 박스, 콤보 박스, 라디오 박스, 체크 박스 등을 확인하고 규칙을 정의
 - 텍스트 박스(Text Box): 입력이 가능함을 표시
 - 콤보 박스(Combo Box): 목록에서 항목을 선택하거나 입력할 수 있음
 - 라디오 박스(Radio Box): 여러 개의 값 중 하나만을 선택
 - 체크 박스(Check Box) : 여러 개의 값 중 하나 이상을 선택 할 수 있음

UI 설계

❖ UI 상세 설계

- ✓ 실제 설계 및 구현을 위해 모든 화면에 대해 자세하게 설계를 진행하는 단계
- ✓ UI 상세 설계 순서
 - ❑ 요구사항 확인: UI 상세 설계를 위한 요구사항을 최종적으로 확인하는 단계
 - ❑ UI 설계서 표지 및 개정 이력 작성
 - UI 설계서 표지는 다른 문서와 혼동되지 않도록 프로젝트 명이나 시스템 명을 포함시켜 작성
 - UI 설계서 개정 이력은 UI 설계서가 수정될 때마다 어떤 부분이 어떻게 수정되었는지를 정리해 놓은 문서
 - 처음 작성 시 첫 번째 항목을 '초안 작성' 버전(Version)을 1.0 으로 설정
 - UI 설계서에 변경 사항이 있을 때마다 변경 내용을 적고 버전을 높임
 - ❑ UI 구조 설계: UI 요구사항과 UI 프로토타입에 기초하여 UI 구조를 설계하는 단계
 - ❑ 메뉴 구조 설계: 사이트 맵 구조를 통해 사용자 기반 메뉴 구조를 설계하는 단계
 - ❑ 화면 설계: UI 프로토타입과 UI 프로세스를 참고하여 필 요한 화면을 페이지 별로 설계하는 단계



UI 설계

❖ UI 시나리오 문서

- ✓ 사용자 인터페이스이 기능 구조, 대표 화면, 화면 간 인터랙션의 흐름, 다양한 상황에서의 예외 처리 등을 정리한 문서
- ✓ 요건
 - ❑ 완전성
 - ❑ 일관성
 - ❑ 이해성
 - ❑ 가독성
 - ❑ 수정 용이성
 - ❑ 추적 용이성



User Experience

❖ UX

- ✓ 사용자가 시스템이나 서비스를 이용하면서 느끼고 생각하게 되는 총체적인 경험으로 단순히 기능이나 절차상의 만족 뿐 만 아니라 사용자가 참여, 사용, 관찰하고, 상호 교감을 통해서 알 수 있는 가치 있는 경험
- ✓ 기술을 효용성 측면에서만 보는 것이 아니라 사용자의 삶의 질을 향상시키는 하나의 방향으로 보는 새로운 개념
- ✓ UI가 사용성, 접근성, 편의성을 중시한다면 이러한 UI를 통해 사용자가 느끼는 만족이나 감정을 중시
- ✓ 특징
 - ❑ 주관성(Subjectivity): 사람들의 개인적, 신체적, 인지적 특성에 따라 다르므로 주관적
 - ❑ 정황성(Contextuality): 경험이 일어나는 상황 또는 주변 환경에 영향을 받음
 - ❑ 총체성(Holistic): 개인이 느끼는 총체적인 심리적, 감성적인 결과

❖ HCI(Human Computer Interaction/Interface)

- ✓ 사람이 시스템을 보다 편리하고 안전하게 사용할 수 있도록 연구하고 개발하는 학문으로 최종 목표는 시스템을 사용하는데 있어 최적의 사용자 경험(UX)을 만드는 것
- ✓ 사람과 컴퓨터의 상호작용을 연구해서 사람이 컴퓨터를 편리하게 사용하도록 만드는 학문

❖ 감성 공학:

- ✓ 제품이나 작업환경을 사용자의 감성에 알맞도록 설계 및 제작하는 기술로 인문 사회 과학, 공학, 의학 등 여러 분야의 학문이 공존하는 종합 과학