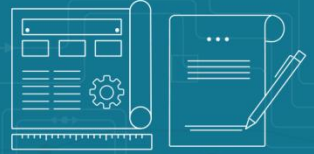


화면 구현

part 1



```
$('.menuBtn > .go').on('click', function(){
    var menuId = $(this).attr('id');
    switch (menuId) {
        case "goAbout":
            $('.slideWrap').slick('slickGoTo', 1);
            break;
        case "goContact":
            $('.slideWrap').slick('slickGoTo', 2);
            break;
        case "goHistory":
            $('.slideWrap').slick('slickGoTo', 3);
            break;
        case "goMain":
            $('.slideWrap').slick('slickGoTo', 0);
            break;
    }
});
```

UI 설계 확인 방법

학습목표

- UI 설계 문서에 대해 이해하고 이를 활용할 수 있다.
- 검토한 내용으로 이해관계자 사이에서 협의할 수 있다.

학습내용

- UI 설계 검토
- UI 설계 피드백

UI 설계 검토

1. 설계 개발 프로세스



1) UX/UI 기획

- 우리가 제공하려는 서비스에서 상호작용을 설계하여 좋은 사용성과 상호작용 수단과 방식을 제공

2) UI 설계 문서 작성

- 프로젝트 및 사업을 진행할 때 여러 이해관계자가 원활하게 커뮤니케이션을 하기 위한 문서 작성

UI 설계 검토

1. 설계 개발 프로세스

3) GUI 디자인

- 설계 문서를 기반으로 그래픽 디자인 작업

4) GUI 가이드라인

- 그래픽 작업을 통하여 생성된 이미지를 이용하여 화면에 배치하는 작업

5) 코딩 개발

- 개발자는 GUI 가이드 문서를 이용하여 실제 화면을 코딩

UI 설계 검토

2. UI 디자인 문서



상세 문서가
왜 필요할까?

이해관계자 간의
커뮤니케이션 도구

상세 수준에 따라 그래픽 디자인과 개발 결과가
달라질 수 있고, 개발 및 출시 후
UI 의도와 결정의 근거가 됨

1) 디자인 문서 작성법

화면 스케치

Wireframe
디자인

화면의
상세 요소에
대한 기술

동작에 대한
명세

UI 설계 검토

2. UI 디자인 문서

1) 디자인 문서 작성법

화면 스케치

- 주요 화면에 대한 컨셉 스케치
- 화면의 큰 스타일 및 대략적인 정보 배치
- 몇 가지 안에 대하여 내부 검토 후 선택

Wireframe 디자인

- **자세한 화면구성**
: UI 구성요소를 채워 넣는 일
- 실제 적용 가능한 UI 구성요소의 적용
- 실제 화면에서 보이는 것과 유사한 크기의 Font, Image Size 등으로 구성

UI 설계 검토

2. UI 디자인 문서

1) 디자인 문서 작성법

화면의
상세 요소에
대한 기술

- 자세한 화면의 구성 및 요소에 대한 설명
- 디자인 및 개발 시 주의사항 및 요구사항 설명

동작에 대한
명세

- 각각의 화면을 설명하는 구성 이외에 전체 화면 간의 흐름을 볼 수 있는 내용
- 화면 사이의 이동을 표현하기 위해 Flowchart와 유사한 도구를 사용

UI 설계 검토

2. UI 디자인 문서

2) 디자인 가이드라인 작성법

최우선 목적에
집중

- 사용자에게 가장 중요한 목적이 무엇인가 판단
- 최우선 목적에 맞게 디자인되었는지 고려

최대한
간결하게 작성

- 정말 필요한 정보인지 판단하여 필요한 정보만 삽입
- 가장 적당한 디자인을 통해 최대한의 효과 창출

정보를 적절한
스타일로 제공,
중요한 것은
강조

- 중요한 정보를 강조하여 사용자에게 전달력을 높임

UI 설계 검토

2. UI 디자인 문서

2) 디자인 가이드라인 작성법

동작하는 방법을
알려주는 디자인

- 사용자가 디자인을 통해 무엇을 어떻게 해야 할지 알려주는 형식의 디자인이 필요
- Color, Size, Font, Layout을 고려

인터랙션 활용

- 혁신적인 인터랙션을 통해 쉽게 사용할 수 있도록 유도

Pixel

- 해상도를 고려한 디자인 Pixel을 조정

Image 활용

- Text보다 Image를 활용하여 정확한 의미 전달

UI 설계 검토

2. UI 디자인 문서

2) 디자인 가이드라인 작성법



기타 고려해야 할 사항은?

Check Point 1

가독성이 보장되는
텍스트 크기와
색상 선택

Check Point 2

간섭없이
누를 수 있는
버튼 사이즈 고려

Check Point 3

다양한 디바이스에서
사용 가능한 디자인

Check Point 4

직접 입력보다
사용자가 선택할 수
있는 UI 사용

UI 설계 검토

3. UI 설계 시 고려사항

- 화면 설계 시 **실제 화면과 1:1 구도로** 작업해야 전체적인 콘텐츠의 구조 파악 가능
- 디자이너와의 **커뮤니케이션 원활**



**UI 화면 설계 작업을 실제 화면과 같은
구도로 작업해야 함**

콘텐츠 구성 및 배치는
반드시 **이유와 목적**이 있어야 함

UI 설계 검토

3. UI 설계 시 고려사항

사업의 전략



컨셉



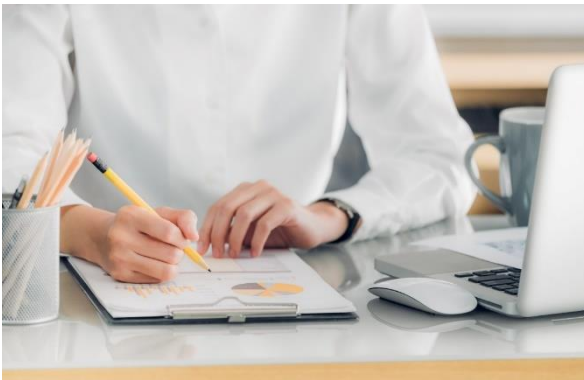
서비스 컨셉



메인-서브
페이지의 컨셉



UI 설계



모든 기획은
사용자의 입장을 고려

UI 화면 설계와 문서는
직관적이며 쉽게 구성



UI 설계 검토

3. UI 설계 시 고려사항



관련 문서

- Wireframe
- 스토리보드
- UI 시나리오(글과 그림으로 설명이 들어간 문서)
- Workflow
- IA(Information Architecture)
- DFD(Data Flow Diagram)

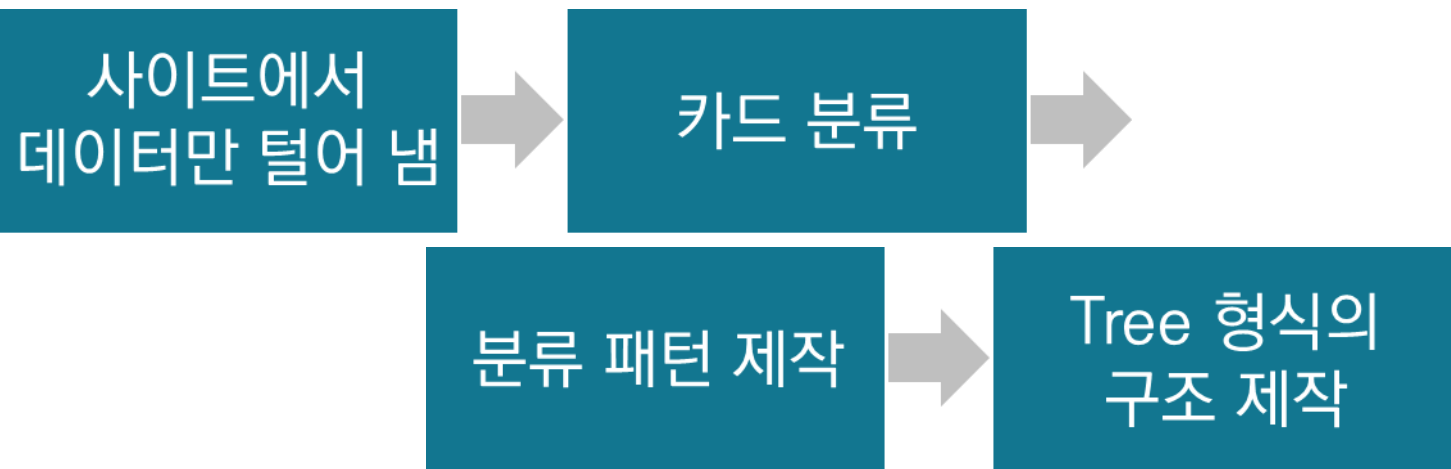
UI 설계 검토

3. UI 설계 시 고려사항

1) IA



IA란?



2) 사이트맵



사이트맵이란?

- IA에서 간단하게 페이지 중심으로 맵을 만든 것
- 사이트 전체를 쉽게 이해할 수 있음

UI 설계 검토

3. UI 설계 시 고려사항

3) DFD 및 기능설계서



DFD란?

- DFD는 데이터의 흐름 다이어그램을 뜻함
- 데이터가 어디서 와서 어디로 흘러가는지 정의가 필요할 때 반드시 만들어줘야 함



기능설계서란?

- 기능 Input과 Output에 대한 정의가 가장 중요함
- 무엇을 집어 넣어서 무엇을 내보낼 것 대한 정의가 있어야 함

UI 설계 검토

3. UI 설계 시 고려사항

3) DFD 및 기능설계서



Flowchart란?

- 사용자가 서비스를 이용하면서 여러 가지 시나리오를 조사, 여러 가지 조건들을 Flowchart로 작성
- 사용자 측면 뿐만 아니라, 데이터 측면에서도 필요한 경우 Flowchart를 만들 필요성이 있음

UI 설계 피드백

1. 디자인 검토 및 협업



그래픽 디자인과 **개발 간 조율과 협업**에 대한
UI 디자이너의 중요한 역할

UI 설계 피드백

1. 디자인 검토 및 협업

1) 초기 UI 디자인 리뷰 및 수정



목적

- 서비스 기획 의도, 법률, 보안, 사내 정책 등을 고려하여 검토



방법

- GUI 가이드

: 화면 설계에 대한 자세한 의도 전달 확인

- 개발

: 상세 UI 정의 누락 여부 확인, 기술 및 일정에 대한 개발 가능성 검토

UI 설계 피드백

1. 디자인 검토 및 협업

2) 디자인 및 개발 진행 중 UI 리뷰 및 수정



목적

- 디자인 중간 진행 상황 체크



방법

- GUI 가이드

: 의도와 다른 그래픽 디자인 결과에 대한 수정 사항 검토

- 개발

: 개발 중 구현 이슈, 일정상의 이슈 및 개발 결과와 UI가 다른 경우 확인

UI 설계 피드백

1. 디자인 검토 및 협업

3) 기획팀, 디자이너, 개발팀 협업 이슈

- **빠른 대안 찾기와 전체를 고려한 디자인 수정**

: 기획된 디자인으로 개발이 어려운 경우
다른 대안을 찾고, 그 문제점을 파악 후 알맞은
디자인으로 교체

- **개발 기간 단축을 위한 효율적인 업무 체계**

: 디자인 협업 소프트웨어를 활용한 실시간 업무
협업의 체계 도입

UI 설계 피드백

2. 디자인 검증

1) 사용성 테스트(Usability Test)



사용성 테스트(Usability Test)란?

- 사용자의 니즈를 발견하는 방법
- 상세한 문제점 진단 및 개선 방향 제시가 가능하고, 문제의 중요도를 판단해 작업의 우선순위를 정하는 데 도움이 됨

UI 설계 피드백

2. 디자인 검증

1) 사용성 테스트(Usability Test)

형성 사용성 테스트 (Formative Usability Test)

- 시스템 형성하는 과정에서 도움이 되는 테스트 방법
 - 사용성의 문제점 발견 혹은 디자인 결정에 이용
 - 문제점을 빠르게 찾을 수 있음

총괄 사용성 테스트 (Summative Usability Test)

- 개발 후반부에 시행
 - 수행 데이터 분석을 통해 데이터적인 검증결과를 도출

UI 설계 피드백

2. 디자인 검증

1) 사용성 테스트(Usability Test)

평가 절차 A

테스트 시나리오
수행과 측정 및 기록

- 실제 사용 상황을 가정해 미리 작성된 목표 시나리오에 맞춰 기기 조작 실행
- 수행 시간, 수행도, 특이사항, 주요 오류 유형 측정 및 기록

UI 설계 피드백

2. 디자인 검증

1) 사용성 테스트(Usability Test)

평가 절차 B

사용자 만족도 평가와
개선 보안 사항 발굴

- 제품에 대한 사용자 인터페이스의 주요 구성 요소별 사용자 만족도 평가
- 사용성을 높이기 위해 개선 보완해야 할 요소 발굴

UI 설계 피드백

2. 디자인 검증

1) 사용성 테스트(Usability Test)

평가 절차 C

문제점의 개선 및
보완을 위한 새로운
아이디어 탐색
(반복 실행)

- 주요 사용상 오류의 원인을 진단하고, 이를 개선 및 보완하기 위한 아이디어이션(Ideation) 진행
- 향후 사용자 인터페이스 개발을 위한 새로운 아이디어 탐색

UI 설계 피드백

2. 디자인 검증

1) 사용성 테스트(Usability Test)

항목	설명	측정 지표
작업 시간	<ul style="list-style-type: none"> - 사용자가 특정 목표를 완료한 시간 - 사용자가 특정 기능 또는 인터랙션 모드에 머무는 시간 (지속하는) 특정 이벤트를 수행하는 데까지 걸린 시간 - 입력 속도와 빈도 	<ul style="list-style-type: none"> - 완료 시간 - 로딩 시간 - 이벤트까지 걸린 시간 - 입력 시간
사용 패턴	<ul style="list-style-type: none"> - 사용자의 행동 또는 기능 사용 빈도 - 마우스 클릭의 수, 기능 사용의 수, 마우스 활용의 양 등 - 사용자의 접근의 수와 방문 빈도, 이용정보의 양 - 최적의 해결방법과 실제 행동 사이의 비율 	<ul style="list-style-type: none"> - 사용 빈도 - 정보 접근성 - 최선 해결책과의 편차

UI 설계 피드백

2. 디자인 검증

1) 사용성 테스트(Usability Test)

항목	설명	측정 지표
정확성	<ul style="list-style-type: none"> - 사용자가 특정 목표를 완료할 때까지의 에러의 양 - 사용자 인터페이스 조작 또는 위치 컨트롤의 정확도 - 검색 정보의 총량과 정확한 정보 사이의 비율 	<ul style="list-style-type: none"> - 에러율 - 공간 정확도 - 정보의 정확성
완성도	<ul style="list-style-type: none"> - 사용자의 목표 성공 여부에 대한 숫자 혹은 백분율(%) 	<ul style="list-style-type: none"> - 성공/실패 여부와 비율

UI 설계 피드백

2. 디자인 검증

1) 사용성 테스트(Usability Test)

항목	설명	측정 지표
학습 용이성	<ul style="list-style-type: none"> - 객체를 보는 즉시 사용하는 방법을 알 수 있는 정도 - 특정 목표를 달성하기 위해 필요한 기능 습득 정도 - 외부 요소가 중요한 정보와의 간섭 없이 습득하는 정도 	<ul style="list-style-type: none"> - 행동유도성 - 기억 용이성 - 단순성
일관성	<ul style="list-style-type: none"> - 동일한 상황이나 유사한 목표에 일관되고 표준화 된 체계를 가지고 있는지 여부에 대한 평가 - 사용자가 이해할 수 있는 인식 요소와의 상호작용을 제공하는지의 여부 	<ul style="list-style-type: none"> - 시각적 일관성 - 기능적 일관성 - 가독성 - 친숙성

UI 설계 피드백

2. 디자인 검증

2) Heuristic Evaluation



Heuristic Evaluation이란?

- **평가 척도에 따른 준수 여부**를 사용성 전문가들이 판단하는 방법론
- 시스템 개발, 평가에 사용하며, 시스템 인터페이스의 문제점을 파악하기 위해 **평가 목적으로 개념화** 시킨 방법론

장점

- 전문가들만 모여지면 언제든지 활용할 수 있어 시간과 비용이 절약

단점

- 전문가 개인의 편견이 개입될 수 있음

UI 설계 피드백

2. 디자인 검증

2) Heuristic Evaluation

평가 단계

계획

평가자 결정

실행

결과 분석

평가 척도

시스템 상태의 시각화

시스템/실세계 일치

제어의 자유

일치의 표준

에러 방지

기억보다 인식

융통성과 효율성

간소화된 디자인

에러 인식, 진단, 복구

도움말과 문서화

UI 설계 피드백

2. 디자인 검증

3) Cognitive Walkthrough



Cognitive Walkthrough란?

인지적 시찰법

사용자 조사방법

- 시스템 이해의 난이도 및 배우기 쉬운지 평가

편리성 여부의 평가

- 설명 또는 안내 없이 사용하는 것

어떤 한 시점에서 적합한 행동을 취하는지 여부의 평가

- 연속된 행동을 단계별로 분석

UI 설계 피드백

2. 디자인 검증

3) Cognitive Walkthrough

A 단계

- 평가하고자 하는 인터페이스의 대표성을 가진 대상을 **사용자 관점**에서 선정

B 단계

- 행동 단계 예측 및 연속된 행동의 단계마다 **학습이론 기반의 질문**

C 단계

- **예상 행동을 유도**하는지의 여부 평가
- **시스템에서 제공하는 피드백**이 사용자가 원하는 방법으로 실행되는지 판단

UI 설계 피드백

2. 디자인 검증

3) Cognitive Walkthrough



장점

- 웹사이트 같은 애플리케이션의 기능 면에서 특정 범위를 정의하기 유용
- 해당 인터페이스가 효과적으로 탐색하기에 쉬운지, 어려운지를 보여주는 지표로 사용 가능
- 음성이나 화면 기반의 시스템 평가에 적합

핵심정리

1. UI 설계 검토

- UI 설계 개발 프로세스: UX/UI 기획 → UI 설계문서 작성 → GUI 디자인 → GUI 가이드라인 → 코딩 개발
- 우리가 제공하려는 서비스에서 상호작용을 설계하여 좋은 사용성과 상호 작용 수단과 방식을 제공 할 수 있도록 설계 되어야 함
- 설계 단계마다 각각의 산출물들을 이용하여 이해관계자들 간의 커뮤니케이션 수단으로 사용할 수 있음
- 디자인 가이드라인 작성 시 사용자의 최우선 목적에 맞는지 고려하여 작성하며 최대한 간결하고 정보를 적절하게 전달할 수 있어야 함
- 설계 시 고려사항으로는 실제 화면과 1:1 구도로 작업을 진행하며, 콘텐츠 구성 및 배치는 반드시 이유와 목적을 가져야 함
- 모든 기획은 사용자의 입장을 고려하여 설계되었는지 확인해야 함

핵심정리

2. UI 설계 피드백

- 초기 UI 디자인 피드백은 서비스 기획 의도, 법률, 보안, 사내 정책 등을 고려할 수 있음
- 디자인 및 개발 중 UI 피드백은 디자인 중간 진행 사항을 체크하여 개발 중 구현에 이슈가 있거나 일정상 이슈, 개발 결과와 다른 UI가 개발되었는지 확인할 수 있음
- UI디자인 검증 방법: 사용성 테스트, Heuristic Evaluation, Cognitive Walkthrough
- 사용성 테스트는 사용자의 니즈를 발견하는 방법으로 상세한 문제점 진단 및 개선 방향을 제시 가능한 방법
- Heuristic Evaluation은 평가 척도에 따른 준수 여부를 사용성 전문가들이 판단하는 방법론
- Cognitive Walkthrough는 인지적 시찰법이라고 하면 시스템을 얼마나 이해하고 배우기 쉬운지 평가하는 사용자 조사 방법론임