

대분류 / 20
정보통신

중분류 / 01
정보기술

소분류 / 02
정보기술개발

세분류 / 02
응용SW엔지니어링

학습모듈 / 04

04 화면 구현

LM2001020204_14v2

[NCS-학습모듈의 위치]

대분류	정보 통신
중분류	정보 기술
소분류	정보 기술 개발

세분류	능력단위	학습모듈명
SW아키텍처		
응용SW 엔지니어링	요구사항 확인	요구사항 확인
시스템 엔지니어링	어플리케이션 설계	어플리케이션 설계
DB엔지니어링	어플리케이션 구현	어플리케이션 구현
NW엔지니어링	화면 구현	화면 구현
보안 엔지니어링	데이터 입출력 구현	데이터 입출력 구현
UI/UX 엔지니어링	통합 구현	통합 구현
시스템SW 엔지니어링	개발자 테스트	개발자 테스트
	정보시스템 이행	정보시스템 이행
	제품소프트웨어 패키징	제품소프트웨어 패키징
	소프트웨어공학 활용	소프트웨어공학 활용

차 례

학습모듈의 개요	1
학습 1. UI 요구 사항 확인하기	
1-1 UI 요구 사항 검증	3
1-2 UI 프로토타이핑	9
• 교수·학습 방법	14
• 평가	15
학습 2. UI 설계하기	
2-1 UI 설계	17
• 교수·학습 방법	24
• 평가	25
학습 3. UI 구현하기	
3-1 UI 구현	27
3-2 UI 테스트	40
• 교수·학습 방법	43
• 평가	44
참고 자료	46
활용 서식	47

화면 구현 학습모듈의 개요

학습모듈의 목표

요구 사항 분석 단계에서 파악된 화면에 대한 요구 사항을 소프트웨어 아키텍처 단계에서 정의된 구현 지침 및 UI/UX 엔지니어가 제시한 UI 표준과 지침에 따라 화면과 폼, 메뉴 등의 구조와 흐름을 구현하여 반영할 수 있다.

선수학습

UML(Unified Modeling Language), HTML, CSS, JavaScript, 요구사항, 프로토타이핑, UI(User Interface)

학습모듈의 내용체계

학습	학습 내용	NCS 능력단위 요소		
		코드 번호	요소 명칭	수준
1. UI 요구 사항 확인하기	1-1 UI 요구사항 검증 1-2 UI 프로토타이핑	2001020204_14v2.1	UI 요구 사항 확인하기	2
2. UI 설계하기	2-1 UI 설계	2001020204_14v2.2	UI 설계하기	2
3. UI 구현하기	3-1 UI 구현 3-2 UI 테스트	2001020204_14v2.3	UI 구현하기	2

핵심 용어

요구 사항, 유스케이스, HTML, 프로토타이핑, CSS, JavaScript

학습 1

UI 요구 사항 확인하기 (LM2001020204_14v2.1)

학습 2 UI 설계하기(LM2001020204_14v2.2)

학습 3 UI 구현하기(LM2001020204_14v2.3)

1-1. UI 요구 사항 검증

학습 목표

- 응용 소프트웨어 개발을 위한 UI 표준 및 지침에 의거하여, 개발하고자 하는 응용 소프트웨어에 적용될 UI 요구 사항을 확인할 수 있다.

필요 지식 /

① 유스케이스(Use Case) 상세 검토

1. 유스케이스 상세의 용도

유스케이스 상세는 사용자와 시스템 간의 이벤트 흐름을 파악할 수 있게 하며 시스템과 사용자 사이의 상호 작용과 시스템 행위를 이해할 수 있다. 또, 이를 통해 UI 화면 구성을 유추할 수 있다.

2. 유스케이스 상세의 특징

- 유스케이스의 기능과 목적이 아닌 이벤트 흐름이 기술된다.
- 유스케이스에 속한 흐름만 기술된다.
- 유스케이스와 연결되지 않는 액터는 표현되지 않는다.
- 공통 용어집에 있는 용어가 사용된다.
- 직접적인 어휘가 사용된다(간단하게 기술하기 위해 복잡한 용어를 사용하지 않는다.)
- 간단하고 간결한 문장으로 표현된다.
- 부사는 사용되지 않는다.
- 복합문은 최대한 피해서 작성된다.

② 유스케이스(Use Case) 상세 예시

ID	U001	UseCase 명	도서목록 검색
작성자	000	수정자	000
최초 작성일	0000/00/00	최종 수정일	0000/00/00
Actor	고객		
사전 조건	없음		
사후 조건	없음		
기본 흐름	액터 행위	시스템 행위	
	1. 도서 검색을 선택한다. 3. 도서 검색에 필요한 조건을 선택한다. 4. 검색하기 원하는 단어를 입력하고 검색을 요청한다.	2. 도서 검색 화면을 표시한다. 5. 입력한 조건에 해당하는 도서 목록을 검색하여 보여준다.	
대체 흐름	액터 행위	시스템 행위	
예외 흐름	액터 행위	시스템 행위	
	4. 검색을 요청했으나 20초 이상 기다려도 응답이 없다.	5. 오류 메시지를 출력한다.	
포함/확장			
우선 순위	상		
사용 빈도	상		
업무 규칙			
특별요구사항			
비고	도서 검색 조건은 도서명, 핵심단어, 저자, 출판사, ISBN 중 하나이다.		

[그림 1-1] 도서 목록 검색 유스케이스 상세

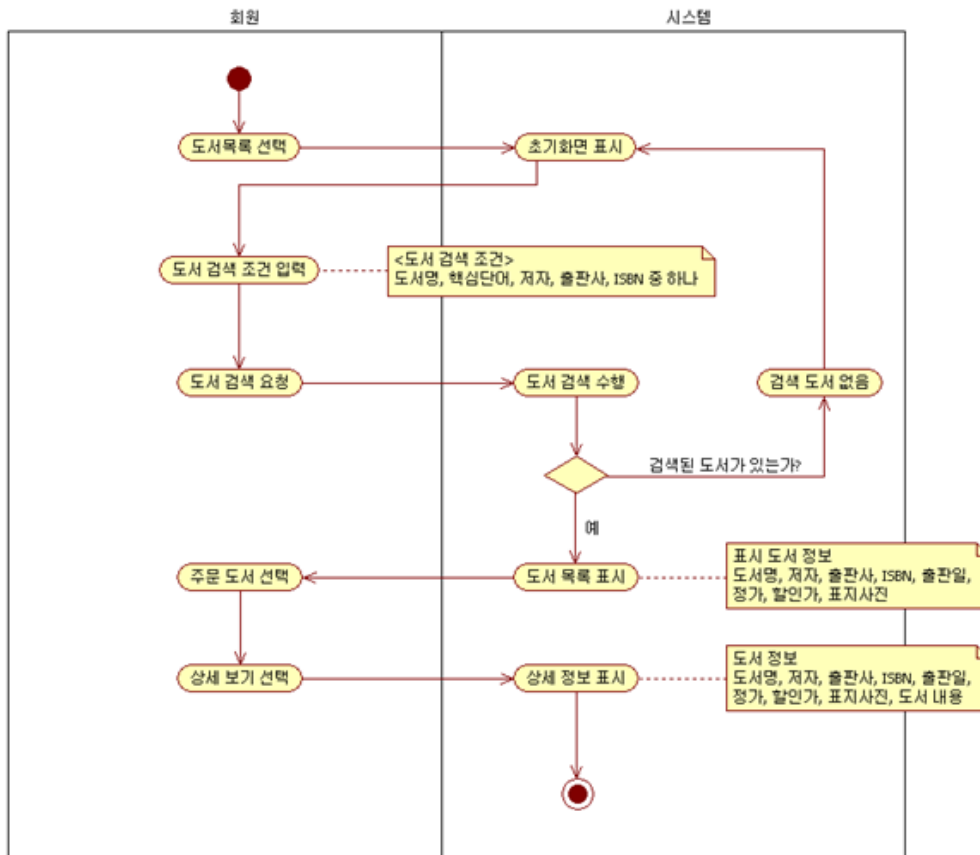
유스케이스 상세 결과물을 검토하여 UI에 표현되어야 하는 구성 요소들에 대해서 파악하고 검토할 수 있다. UI의 전체 구조를 파악하고, 내부에 구성될 요소들의 종류와 해당 요소들의 표현 방식 등을 검토할 수 있다.

③ 액티비티 다이어그램(Activity Diagram) 검토

유스케이스 상세 이벤트 흐름은 액티비티 다이어그램으로도 표현될 수 있으며, 액티비티 다이어그램으로 상세 이벤트 흐름이 표현되면 다음과 같은 이점을 갖는다.

- 이벤트 흐름 안에서의 복잡한 조건 로직을 쉽게 이해할 수 있는 시각적인 표현을 제공한다.
- 복잡한 이벤트 흐름을 사용자에게 효과적으로 표현할 수 있는 수단을 제공한다.
- 이벤트 흐름의 각 행위 사이의 종속성과 관계를 이해할 수 있게 한다.

④ 액티비티 다이어그램(Activity Diagram) 예시



[그림 1-2] 도서 목록 검색 액티비티 다이어그램

액티비티 다이어그램의 검토를 통해 전체 UI의 흐름을 파악할 수 있다. 사용자의 최초 접근부터 종료까지 전체의 흐름을 파악하고, 구성되어야 하는 UI 요소들의 순서와 흐름을 검토할 수 있다.

수행 내용 / UI 요구 사항 파악하기

재료 · 자료

- 유스케이스 상세, 액티비티 다이어그램

기기(장비 · 도구)

- 컴퓨터, 프린터
- 종이, 펜

안전 · 유의사항

- 사용자의 사용성을 적극 고려한다.
- 기술적 위험에 대해 고려한다.
- UI 전체 구조에 대해 고려한다.

수행 순서

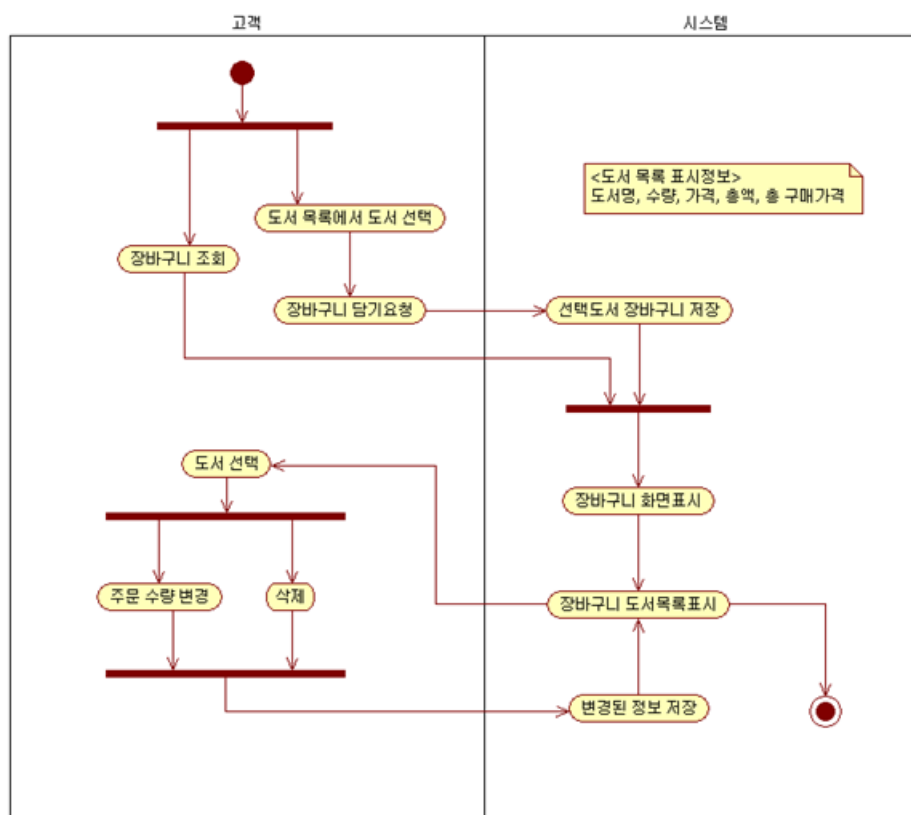
① 유스케이스 상세를 바탕으로 UI 구성을 검토한다.

ID	U001	UseCase 명	장바구니 관리
작성자	000	수정자	000
최초 작성일	0000/00/00	최종 수정일	0000/00/00
Actor	고객		
사전 조건	도서 목록 검색 UseCase를 실행한다.		
사후 조건	없음		
기본 흐름	액터 행위	시스템 행위	
	[장바구니 담기] 1. 검색한 도서 목록에서 원하는 도서를 선택하여 장바구니에 담기를 선택한다.	2. 장바구니 화면을 표시한다. 3. 선택된 도서를 장바구니에 담는다. 4. 장바구니에 저장된 도서 목록을 표시 한다. 2. 장바구니에 저장된 도서 목록을 표시 한다. 2. 장바구니에 저장된 도서 목록을 표시 한다. 5. 장바구니 도서목록을 변경한다. 6. 변경된 도서목록 정보를 표시한다.	
	[장바구니 조회] 1. 장바구니 조회를 요청한다.		
	[장바구니 변경] 1. 장바구니 조회를 요청한다.		
	3. 장바구니 도서 목록에서 도서를 선택 하고 수량을 변경하거나 삭제한다. 4. 변경내용을 저장한다.		
대체 흐름	액터 행위	시스템 행위	
예외 흐름	액터 행위	시스템 행위	
포함/확장			
우선 순위	중		
사용 빈도	상		
업무 규칙			
특별요구사항			
비고	장바구니 도서 목록에는 도서명, 수량, 가격, 총액, 총구매가격을 표시한다.		

[그림 1-3] 장바구니 유스케이스 상세

1. 전체 화면 구조에 대해서 구상해 본다.
 - 장바구니 화면에 표현되어야 할 기능들에는 어떤 것들이 있는지 나열해 본다.
 - 어떤 형태의 배치를 통해 각 기능을 표현할 것인지 설계하여 본다.
2. 화면에 표현되는 요소에는 어떤 것들이 필요할 것인지를 구상해 본다.
 - 장바구니 화면의 입력 요소들에는 어떤 것들이 필요할지 구상해 본다.
 - 각 입력 요소들의 형태와 입력 방법에 대해서 구상해 본다.
 - 입력 요소들 이외에 어떤 화면 요소들이 필요할지 구상해 본다.
 - 각 요소들의 배치에 대해서 구상하고 설계하여 본다.
3. 화면과 화면의 흐름에 대해서 구상해 본다.
 - 장바구니의 기능을 표현하기 위한 필요한 페이지들에 대해서 구상해 본다.
 - 각 화면들의 이동과 화면 흐름에 대해서 구상해 본다.
4. 검토한 내용을 토대로 유스케이스 상세에서 추가로 필요한 내용들은 없는지 판단해 본다.

② 액티비티 다이어그램을 바탕으로 UI 구성을 검토한다.



[그림 1-4] 장바구니 액티비티 다이어그램

1. 전체 화면 구조에 대해서 구상해 본다.
 - 장바구니 화면에 표현되어야 할 기능들에는 어떤 것들이 있는지 나열해 본다.
 - 어떤 형태의 배치를 통해 각 기능을 표현할 것인지 설계하여 본다.
2. 화면에 표현되는 요소에는 어떤 것들이 필요할 것인지를 구상해 본다.
 - 장바구니 화면의 입력 요소들에는 어떤 것들이 필요할지 구상해 본다.
 - 각 입력 요소들의 형태와 입력 방법에 대해서 구상해 본다.
 - 입력 요소들 이외에 어떤 화면 요소들이 필요할지 구상해 본다.
 - 각 요소들의 배치에 대해서 구상하고 설계하여 본다.
3. 화면과 화면의 흐름에 대해서 구상해 본다.
 - 장바구니의 기능을 표현하기 위한 필요한 페이지들에 대해서 구상해 본다.
 - 각 화면들의 이동과 화면 흐름에 대해서 구상해 본다.
4. 검토한 내용을 토대로 유스케이스 상세에서 추가로 필요한 내용들은 없는지 판단해 본다.

수행 tip

- UI 요구 사항을 만족시키기 위해서 사용자와 시스템의 행위를 이해하고, 주요 화면 구성의 주요 맥락을 검토하는 것이 중요하다.

1-2. UI 프로토타이핑

학습 목표

- 응용 소프트웨어 개발을 위한 UI 표준 및 지침에 의거하여, UI 요구 사항을 반영한 프로토타입을 제작할 수 있다.
- 작성한 프로토타입을 활용하여 UI/UX 엔지니어와 향후 적용할 UI의 적정성에 대해 검토할 수 있다.

필요 지식 /

① UI 프로토타이핑

프로토타이핑이란 해당 프로젝트의 모든 기능 또는 일부 기능을 그리거나, 묘사하고 테스트하는 활동을 의미한다.

1. 프로토타이핑의 종류

- 아날로그 프로토타이핑: 화이트보드 혹은 펜과 종이를 이용한 프로토타이핑



[그림 2-1] 아날로그 프로토타이핑의 예

- 디지털 프로토타이핑: 파워포인트, 아크로벳, 비지오, 옴니그래플, 악슈어, HTML 등을 이용한 프로토타이핑

2. 프로토타이핑의 특징

- 프로토타이핑의 단계는 사용자 경험과 관련된 이슈들을 정의하는 과정이므로 반복적으로 수행된다.
- 프로토타이핑은 결과물이 아닌 과정이다.

- 프로토타이핑 과정에 대한 결과는 UI/UX 디자인을 보완하고 개선할 수 있도록 실행 가능한 피드백이 이루어져야 한다.

② HTML을 이용한 프로토타이핑

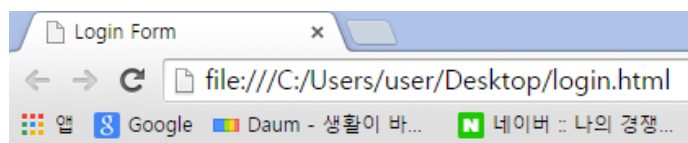
메모장이나 HTML 작성 도구들을 활용하면 HTML을 이용해 프로토타입을 제작할 수 있다.

1. 로그인 HTML 소스 코드

```
<!DOCTYPE html>
<html>
<head>
  <title>Login Form</title>
</head>
<body>
  <h1>Login</h1>
  <p><input type="text" name="login" value="" placeholder="Username or
Email"></p>
  <p><input type="password" name="password" value=""
placeholder="Password"></p>
  <p>
    <label>
      <input type="checkbox" name="remember_me" id="remember_me">
      Remember me on this computer
    </label>
  </p>
  <p><input type="submit" name="commit" value="Login"></p>

  <div class="login-help">
    <p>Forgot your password? <a href="">Click here to reset it</a>.</p>
  </div>
</body>
</html>
```

2. 로그인 HTML 프로토타이핑 결과



Login

☐ Remember me on this computer

Forgot your password? [Click here to reset it.](#)

[그림 2-2] 로그인 프로토타이핑 결과

③ HTML을 이용한 프로토타이핑의 특징 및 방법

HTML을 이용한 프로토타이핑은 먼저 HTML, CSS, JavaScript와 같은 언어들을 기본적으로 이해하고 활용할 수 있어야 가능하다. HTML을 이용한 UI 구성이 가능할 경우 다음과 같은 절차를 통해 프로토타이핑을 진행한다.

1. 프로토타이핑의 대상이 되는 UI 화면의 구조와 구성 요소들을 파악한다.
2. HTML의 다양한 태그를 통해 화면의 레이아웃을 구성한다.
3. 각 UI 표현 요소들을 적절한 HTML 화면 구성 요소들을 이용해 배치하고 표현한다.

수행 내용 / UI 프로토타입 작성하기

재료 · 자료

- HTML 편집 도구

기기(장비 · 공구)

- 컴퓨터, 프린터

안전 · 유의 사항

- 사용자의 사용성을 적극 고려한다.
- 기술적 위험에 대해 고려한다.

수행 순서

- ① 다음 회원가입 예시에 대해서 HTML을 이용한 프로토타이핑을 검토한다.

기본 정보 입력	
이름	<input type="text"/>
주민번호	<input type="text"/> - <input type="text"/>
아이디	<input type="text"/> @naver.com <input type="button" value="중복체크"/> 영문으로 시작하는 4~10자리의 영문, 숫자 조합만 가능합니다.
별명	<input type="text"/>
비밀번호	<input type="password"/> <input type="password"/> 확인을 위해 한번 더 입력해 주세요
비밀번호 힌트	다시 태어나면 되고 싶은 것은? <input type="button" value="답을 입력하세요"/>
생년월일	1999 <input type="text"/> 05 <input type="text"/> 06 <input type="radio"/> 양 <input checked="" type="radio"/> 음
핸드폰	010 <input type="text"/> 1111 - 111 <input checked="" type="radio"/> SKT <input type="radio"/> KTF <input type="radio"/> LGT
주로쓰는 메일	11111 <input type="checkbox"/> 네이버 메일 사용
거주지	부산 <input type="button" value="선택"/>
메일 수신 여부	<input checked="" type="checkbox"/> 네이버 뉴스레터 <input checked="" type="checkbox"/> 상품정보 메일 <input checked="" type="checkbox"/> 캐쉬정보 메일 <input type="checkbox"/> 게임정보 메일 <input type="checkbox"/> sms 정보 수신
<input type="button" value="확인"/>	

[그림 2-3] 회원 가입 프로토타이핑 예시

1. 전체 화면의 레이아웃을 어떻게 구성할지를 설계한다.
2. 화면 레이아웃을 구성하는 방법이 다양하기 때문에 어떤 형식을 통해 레이아웃을 구성할 것인지를 결정한다.

- ② 회원 가입 예시에서 HTML 구성 요소들을 파악한다.

1. 화면에 표현된 UI 요소들은 어떤 것들이 있는지 파악한다.(예, 텍스트 필드, 이미지, 체크 박스 등)
 - 텍스트 형태로 표현된 요소들은 어떤 것들이 있는지 그 종류와 수를 파악하고 표시한다.
 - 사용된 이미지가 있다면 그 크기와 위치 등을 파악한다.
 - 사용자 입력 요소의 형태를 파악한다.
2. 각 요소들을 표현할 때의 제약 조건에는 어떤 것들이 있는지를 파악한다.
 - 사용자 입력 요소에서 제약이 있는 것은 없는지 파악한다.(예, 특수 문자나 영문의 입력 제한)
 - ‘확인’ 버튼을 선택했을 경우 사용자 입력이 빠져서는 안되는 부분이 무엇인지를 파악한다.

3. 각 요소들을 표현하기 위한 HTML 요소는 어떤 것들인지를 파악한다.
4. CSS, JavaScript가 적용되어야 하는 부분을 파악한다.

③ HTML 편집 도구를 활용해 회원 가입 예시를 직접 프로토타이핑한다.

1. 프로토타이핑을 진행할 HTML 편집 도구를 선택하고 구성한다.
 - 문서 편집기 혹은 다양한 웹 브라우저 그리고 통합 개발 환경 도구 중에서 어떤 편집도구를 이용할 것인지 선택하고 작업 환경을 구축한다.
2. 프로토타이핑을 진행 과정에서 도출되는 문제점이나 변경이 필요한 부분 등을 정리한다.
 - 프로토타이핑을 진행하면서 사용자 편의를 위해 변경될 부분이 있는지 생각해 본다. (예, 사용자의 입력 방식이나 순서 등)
 - 회원 가입에 추가로 요청하거나 불필요한 입력 내용이 있는지 파악한다.

수행 tip

- HTML을 이용한 프로토타이핑을 위해서는 우선 HTML 및 CSS와 같은 언어에 대한 학습이 선행되어야 하며 다양한 형태의 화면을 토대로 연습을 진행해 보는 것이 중요하다.

학습 1 교수 · 학습 방법

교수 방법

- 유스케이스 상세, 액티비티 다이어그램, HTML 언어에 대한 이해 정도를 파악한 후 수업을 진행한다.
- 다양한 유스케이스 상세 및 다이어그램, UI 프로토타이핑 예시에 대한 PPT 자료를 제시한 후 설명한다.
- UI 프로토타이핑의 필요성에 대해 이해할 수 있도록 예시를 통해 지도한다.
- 다양한 UI 프로토타이핑 도구와 활용 방법에 대해 지도한다.
- UI 프로토타이핑 과정에 대해 단계적 실습이 이루어질 수 있도록 지도한다.

학습 방법

- 유스케이스 상세와 액티비티 다이어그램을 통한 사용자와 시스템 간의 상호 작용을 이해하고 이에 대한 UI 구성을 이해한다.
- 다양한 형태의 유스케이스 상세와 액티비티 다이어그램을 접해본다.
- UI 프로토타이핑의 목적과 활용에 대해 이해한다.
- UI 프로토타이핑 과정에 대해 실습한다.
- 여러 형태의 UI 프로토타이핑을 연습해 본다.

학습 1 평 가

평가 준거

- 평가자는 학습자가 수행 준거 및 평가 시 고려 사항에 제시되어 있는 내용을 성공적으로 수행하였는지를 평가해야 한다.
- 평가자는 다음 사항을 평가한다.

학습 내용	평가 항목	성취수준		
		상	중	하
UI 요구 사항 검증	- 응용 소프트웨어 개발을 위한 UI 표준 및 지침에 의거하여, 개발하고자 하는 응용 소프트웨어에 적용될 UI 요구 사항을 확인할 수 있다.			
UI 프로토타이핑	- 응용 소프트웨어 개발을 위한 UI 표준 및 지침에 의거하여, UI 요구 사항을 반영한 프로토타입을 제작할 수 있다.			
	- 작성한 프로토타입을 활용하여 UI/UX 엔지니어와 향후 적용할 UI의 적정성에 대해 검토할 수 있다.			

평가 방법

- 평가자 체크리스트

학습 내용	평가 항목	성취수준		
		상	중	하
UI 요구 사항 검증	- UI 설계서에 대한 이해 능력			
	- UI 설계서에 따른 UI 도출 능력			
UI 프로토타이핑	- 프로토타이핑 과정 및 결과에 대한 이해 능력			

- 포트폴리오

학습 내용	평가 항목	성취수준		
		상	중	하
UI 요구 사항 검증	- UI 설계서에 대한 이해 능력			
	- UI 설계서에 대한 분석 능력			
UI 프로토타이핑	- 아날로그 형태의 프로토타입 제작 능력			
	- 프로토타입 도구를 활용한 프로토타입 제작 능력			

피드백

1. 평가자 체크리스트

- UI 설계 문서에 대한 판독에 대해 평가한 후에 필요한 사항 등을 정리하여 돌려준다.
- UI 프로토타이핑 실습에 대해 살펴보고 개선해야 할 사항이나 특히 염두에 두어야 하는 사항 등을 정리하여 돌려준다.

2. 포트폴리오

- 실습한 UI 프로토타이핑 자료에 대해 평가한 후에 주요 사항에 대하여 표시하여 돌려준다.

학습 1	UI 요구 사항 확인하기(LM2001020204_14v2.1)
학습 2	UI 설계하기 (LM2001020204_14v2.2)
학습 3	UI 구현하기(LM2001020204_14v2.3)

2-1. UI 설계

학습 목표

- UI 요구 사항과 UI 표준 및 지침에 따라 화면과 폼의 흐름을 설계하고, 제약 사항을 화면과 폼 흐름 설계에 반영할 수 있다.
- UI 요구 사항과 UI 표준 및 지침에 따라 사용자의 편의성을 고려한 메뉴 구조를 설계할 수 있다.
- UI 요구 사항과 UI 표준 및 지침에 따라 하위 시스템 단위의 내외부 화면과 폼을 설계할 수 있다.

필요 지식 /

① UI 설계의 이해

UI의 설계부터 구현 및 검증까지는 해당 프로젝트의 시작부터 종료까지 전반에 걸쳐 진행되며 진행 내용은 다음과 같다.

1. UI 개발 목표와 범위 수립

해당 프로젝트의 목표 및 정의 단계에서 UI 부분의 목표와 범위에 대한 계획을 수립하는 단계로 프로젝트 계획 전반에 UI 계획을 반영한다. 이 과정을 수행하기 위해서는 해당 프로젝트의 이해 당사자(Stake Holder)의 UI 요구 사항을 조사하고, 정의하며 UI 트렌드 및 해당 서비스의 사용자를 분석한다.

2. UI 전략 수립

서비스를 사용할 사용자 조사 및 시장 조사를 통해 UI와 UX에 대한 전략을 수립하는 단계로 사용자는 물론 기술적 관점에서도 전략을 수립하는 것이 중요하다.

3. 사용자 요구 사항 분석 및 정의

사용자 조사와 페르소나(Persona) 결과를 기반으로 사용자의 요구사항을 추출하고 선별하여 사용자 요구 사항을 분석한다. 이를 통해 해당 시스템 전반에 적용되는 UI의 컨셉을

설계한다. 이때 페이퍼나 도구를 활용한 프로토타입(Prototype)의 제작이 진행된다.

4. UI 상세 설계

UI에 대한 콘셉트가 정의되면 이를 토대로 실제 설계와 구현을 위한 모든 화면에 대해 상세 설계를 진행하게 된다. UI의 기능에 대한 구조와 화면 간의 상호 흐름, 예외 처리 등의 UI 전반에 걸친 설계가 이루어진다.

5. 구현

UI 설계에 따른 구현에는 다양한 기술들이 사용되며 HTML5, CSS3, JavaScript 등과 같은 기본적인 언어들은 물론 jQuery와 같은 JavaScript의 라이브러리 또한 범용적으로 사용된다. 또, UI 구현은 UI 디자이너들과 협업으로 이루어지게 된다.

6. 테스트

테스트 단계는 일반 사용자를 대상으로 해당 개발 소프트웨어의 사용성을 검증하는 단계로 사용성에 대한 계획을 수립하고, 이 계획을 토대로 테스트를 진행하게 된다.

② UI 상세 설계

1. UI 시나리오 작성

(1) UI 상세 설계의 첫 단계는 시나리오 문서의 작성이다. 전체 기능을 한 번에 확인하고 이해할 수 있도록 기능 구조를 만들게 되는데, 보편적으로 계층 구조 또는 플로차트 형태로 작성하게 된다.

(2) 모든 기능에 대해서 공통적으로 적용되는 UI 요소와 사용자와의 상호 작용에 대한 일반적인 규칙을 정의한다.

(3) 사용자와 화면과 화면의 상호 작용에 대한 흐름과 순서(Sequence), 분기(Branch), 조건(Condition), 루프(Loop) 등을 명시한다.

(4) UI에 대한 일반 규칙을 준수하고 각 기능의 상세 시나리오를 작성한다.

2. UI 시나리오 작성을 위한 일반적인 규칙에 대한 정의

(1) 주요 버튼의 위치와 기능

화면에서 공통으로 배치되는 주요 버튼들의 위치와 기능을 설명하는 것으로 화면들에 대한 일관성을 보장한다.

(2) 공통 UI 요소

다양한 형태의 입력 폼(일반 버튼, 체크 박스, 라디오 버튼, 텍스트 입력, 탭 등)과 각 UI 요소들에 대해 언제 사용되는지, 어떤 형태를 갖는지를 정의하며 사용자의 조작에 대해 어떻게 반응하는지를 설명한다.

(3) 기본 레이아웃

여러 화면에 공통적으로 나타나는 제목, 버튼들, 인디케이터 등의 위치와 속성을 설명한다.

(4) 기본 상호 작용 규칙

공통적으로 사용되는 조작 방법과 실행, 이전, 다음 등과 같은 화면 전환효과 등에 대해 설명한다.

(5) 공통 단위의 테스트 흐름

대다수의 기능에 공통적으로 사용되는 삭제, 검색 등에 대한 반응 흐름을 설명한다.

3. UI 시나리오 작성에서 고려할 사항

(1) 완전성(Complete)

누락되는 것이 없도록 최대한 상세하게 기술해야 하며 해당 시스템보다는 사용자의 사용 방법에 기초하여 작성해야 한다.

(2) 일관성(Consistent)

전체 화면과 UI 요소들은 일관성을 갖고 있어야 한다.

(3) 가독성(Readable)

표준화된 템플릿 등을 활용하여 작업자 간에 문서 통한 소통이 원활하여야 하며 문서를 쉽게 읽을 수 있어야 한다.

(4) 이해성(Understandable)

추상적이거나 보편적으로 사용하지 않고 이해하기 어려운 용어의 사용은 피한다.

(5) 수정 용이성(Modifiable)

수정이나 개선 사항에 대해서 시나리오에 반영하는 것이 쉬워야 한다.

(6) 추적 용이성(Traceable)

변경 사항에 대해서 언제, 어떤 부분이 어떤 이유로 변경 되었는지에 대해 쉽게 추적할 수 있어야 한다.

③ GUI 설계

GUI 설계를 통해 개발 소프트웨어의 시각적 개념을 정의하고 표현한다. 이를 통해 전체 UI 구성에 대해 이해할 수 있다.

1. 통일성(Unity)

GUI 설계에서 통일성은 UI 요소들이 공통적으로 갖는 특성을 뜻한다. 통일성을 나타내는 방법은 색상, 모양, 스타일, 요소들의 배치 등 다양하게 존재한다.

2. 우선순위

우선순위는 화면에 보이는 요소들의 나열되는 순서라고 할 수 있다. 우선순위는 사용자가 해당 페이지를 바라보는 시선에 따라 정해지며, 우선순위에 영향을 줄 수 있는 요소로는 UI 요소들의 위치, 색상, 크기 등이 있다.

3. 그 외의 요소들

GUI 설계에서 고려해야 할 부분은 통일성, 우선순위 외에도 비율과 균형, 인터랙션과 모션 등이 있다.

수행 내용 / UI 상세 설계하기

재료 · 자료

- 종이, 펜

기기(장비 · 공구)

- 컴퓨터, 프린터
- 인터넷

안전 · 유의사항

- 사용자의 사용성을 적극 고려한다.
- 누락되는 항목이 없도록 최대한 면밀히 검토하고 설계한다.
- 팀원간의 원활한 의사소통이 이루어질 수 있도록 한다.

수행 순서

- ① UI 상세 설계를 진행할 그룹이나 팀을 구성한다.

1. UI 상세 설계는 다수의 페이지를 구성하고 구조 및 디자인의 설계가 이루어져야 하기 때문에 팀 단위의 실습이 필요하다.

- 전체 구성원의 수를 고려하여 팀 인원을 산정한다.
 - 구성원의 능력차를 고려하여 팀 인원을 분배한다.
2. 팀 구성에 대한 기본 원칙을 확립한다.
 - 각 팀원의 역할을 명확히 분담한다.
 - 팀원 회의 및 기타 규칙을 정한다.

② 설계할 웹 프로젝트의 주제를 선정한다.

1. 팀원 간에 다양한 주제를 토의한다.
 - 어떤 웹 페이지를 설계할 것인지 논의 한다.(예, 쇼핑몰, 그룹웨어, SNS 등)
2. 일반적이며 벤치마킹이 용이한 주제로 선정한다.
 - 선정한 주제를 토대로 기존 웹 페이지들을 조사한다.
3. 주제 선정은 실습 기간 및 팀원의 인원에 따라 적절한 분량이 될 수 있도록 선정한다.
 - 선정한 주제에 대해 작업 규모를 산정한다.
 - 팀원들 간에 역할을 분담하고 업무량을 결정한다.

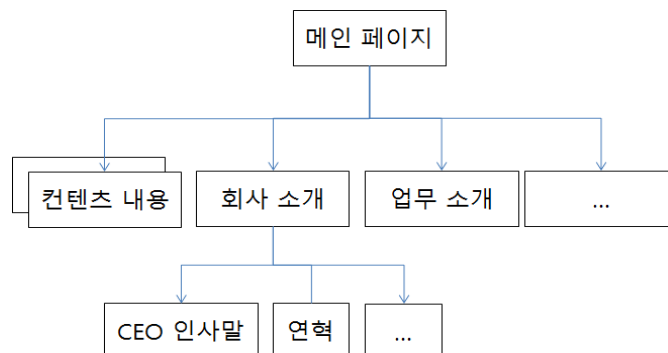
③ 각 구성원은 주제와 유사한 다양한 웹 사이트에 대한 벤치마킹을 진행한다.

1. 벤치마킹한 사이트에 대한 장점과 단점을 정리한다.

예) 사용자의 사용성은 어떠한가? 화면을 구성하는 요소들의 배치는 적절해 보이는가? 화면 간의 이동은 편리한가? 추가되어야 할 기능들은 없는가? 등
2. 구성원들과 벤치마킹한 사이트들을 선별하고 정리한다.

④ UI 상세 설계를 진행한다.

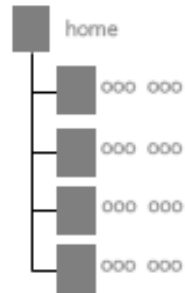
1. UI 시나리오 문서를 작성한다.
 - (1) 전체 페이지들의 계층 구조를 설계한다.



[그림 1-1] 계층 구조 설계의 예

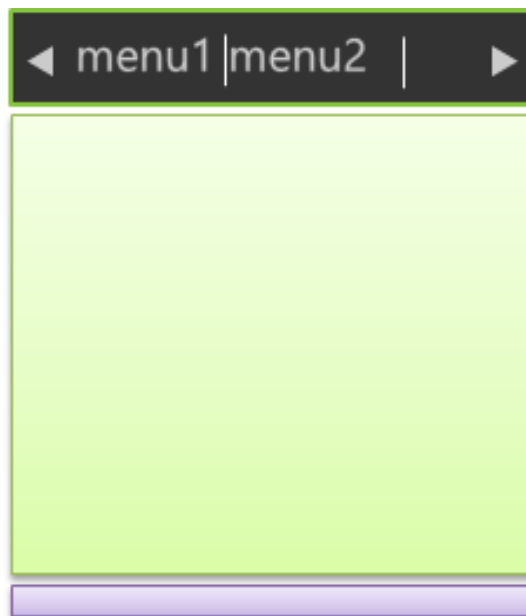
- (2) 전체 페이지에서 공통적으로 적용되는 UI 요소들에 대해 정의한다.
- (3) 메인 화면에 대한 레이아웃과 기능을 정의한다.
- (4) 각 페이지별 이동 경로를 설계한다.
- (5) 전체 사이트 맵을 작성한다.

SITEMAP



[그림 1-2] 사이트 맵의 예

- (6) UI 시나리오 내용을 구성원 간에 반복하여 검토한다.
2. GUI 템플릿 제작과 화면을 디자인한다.
 - (1) 화면 디자인에 활용될 레이아웃, 컬러 패턴, 디자인 요소 등을 정의한다.
 - (2) 디자인 요소들에 대한 정의 이후에 GUI 템플릿을 제작한다.



[그림 1-3] GUI 템플릿의 예

- (3) 각 페이지에 대한 세부 화면을 디자인 한다.
- (4) 디자인된 GUI 화면을 검토한다.

수행 tip

- UI 설계는 우선 많은 참고 자료를 보고 확인을 통해 그 구조와 표현 방법에 익숙해져야 한다. 또, 설계된 내용에 대해 구성원들과 반복적인 검토를 통해 세세한 부분까지 작업이 이루어져야 한다.

학습 2 교수·학습 방법

교수 방법

- 교육생들의 능력 및 성취도를 적합하게 선별하여 균형 잡힌 팀을 구성할 수 있도록 지도한다.
- 팀원 간의 원활한 의사소통을 위한 방법들에 대해서 지도한다.
- 교육생들의 수준에 맞는 주제 선정으로 위해 팀별 시나리오 및 벤치마킹 내용을 파악하여 학습 시간과 실습에 적합한 주제를 선정할 수 있도록 지도한다.
- 실제 UI 상세 설계 문서를 바탕으로 설계 문서 작성 방법 및 내용 도출 방법을 설명한다.
- 각 설계 절차에 대한 핵심 내용과 실습 방법에 대해서 설명한다.

학습 방법

- 팀 구성원 간에 의사소통이 원활할 수 있도록 기본적인 규칙을 지정하고 준수할 수 있도록 한다.
- 팀 구성원 간 회의에 대한 전체 회의록을 작성하여 전체 진행 상황에 대한 이해를 갖고 학습할 수 있도록 한다.
- 팀을 구성하는 각 교육생들은 다양한 사이트들에 대한 벤치마킹 자료를 수집하고 분석하여 각 사이트들에 대한 장점과 단점을 비교 분석하고 학습한다.
- 교육생들은 팀별로 분석한 자료들을 토대로 임의의 주제를 선정 UI 상세 설계를 진행한다. 그리고 설계에서 도출되는 다양한 문서들을 작성하며 UI 설계 방식에 대해서 학습한다.
- 설계 과정에서 도출되는 결과물에 대해 검토하고 검증하여 각 결과물에 대해 이해할 수 있도록 한다.

학습 2 평 가

평가 준거

- 평가자는 학습자가 수행 준거 및 평가 시 고려 사항에 제시되어 있는 내용을 성공적으로 수행하였는지를 평가해야 한다.
- 평가자는 다음 사항을 평가한다.

학습 내용	평가 항목	성취수준		
		상	중	하
UI 설계	- UI 요구 사항과 UI 표준 및 지침에 따라, 화면과 폼의 흐름을 설계하고, 제약 사항을 화면과 폼 흐름 설계에 반영할 수 있다.			
	- UI 요구 사항과 UI 표준 및 지침에 따라, 사용자의 편의성을 고려한 메뉴 구조를 설계할 수 있다.			
	- UI 요구 사항과 UI 표준 및 지침에 따라, 하위 시스템 단위의 내외부 화면과 폼을 설계할 수 있다.			

평가 방법

- 평가자 체크 리스트

학습 내용	평가 항목	성취수준		
		상	중	하
UI 설계	- UI 화면의 구성과 각 화면의 흐름에 대한 이해			
	- 사용자의 사용 편의성, 구조와 표현 방법에 대한 이해 및 설계			
	- 각 화면에 대한 구성과 표현 방법, 설계 방법의 이해			

- 포트폴리오

학습 내용	평가 항목	성취수준		
		상	중	하
UI 설계	- UI 화면의 구성과 각 화면의 흐름에 대한 이해			
	- 사용자의 사용 편의성, 구조와 표현 방법에 대한 이해 및 설계			
	- 각 화면에 대한 구성과 표현 방법, 설계 방법의 이해			

피드백

1. 평가자 체크리스트

- 실습 과정에서 팀원들 간의 협력에 대해 살펴보고, 개선해야 할 사항이나 잘한 내용 등을 정리하여 돌려준다.
- 설계 과정에 대해 살펴보고 개선해야 할 사항이나 각 과정별 중요한 내용에 대해 정리하여 돌려준다.

2. 포트폴리오

- 설계에 대한 결과물을 평가한 후에 주요 사항에 대하여 표시하여 돌려준다.

학습 1	UI 요구 사항 확인하기(LM2001020204_14v2.1)
학습 2	UI 설계하기(LM2001020204_14v2.2)
학습 3	UI 구현하기(LM2001020204_14v2.3)

3-1. UI 구현

학습 목표

- 소프트웨어 아키텍처 세부 구현 지침과 UI 표준 및 지침을 반영하여, UI 요구 사항이 반영된 화면과 폼을 구현할 수 있다.
- 화면과 폼 흐름 설계에 따라, 사용자 접근성을 고려한 화면과 폼의 흐름 제어를 구현할 수 있다.
- 화면과 폼 흐름 설계에 따라, 감성 공학 기법을 고려하여 사용자가 접하는 화면, 폼, 메뉴, 흐름을 구현할 수 있다.

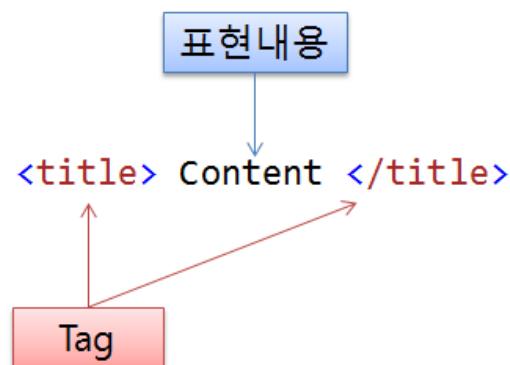
필요 지식 /

① HTML(Hyper Text Markup Language)

HTML은 마크업 언어로써 웹 페이지를 표현하는 언어이다.

1. HTML의 이해

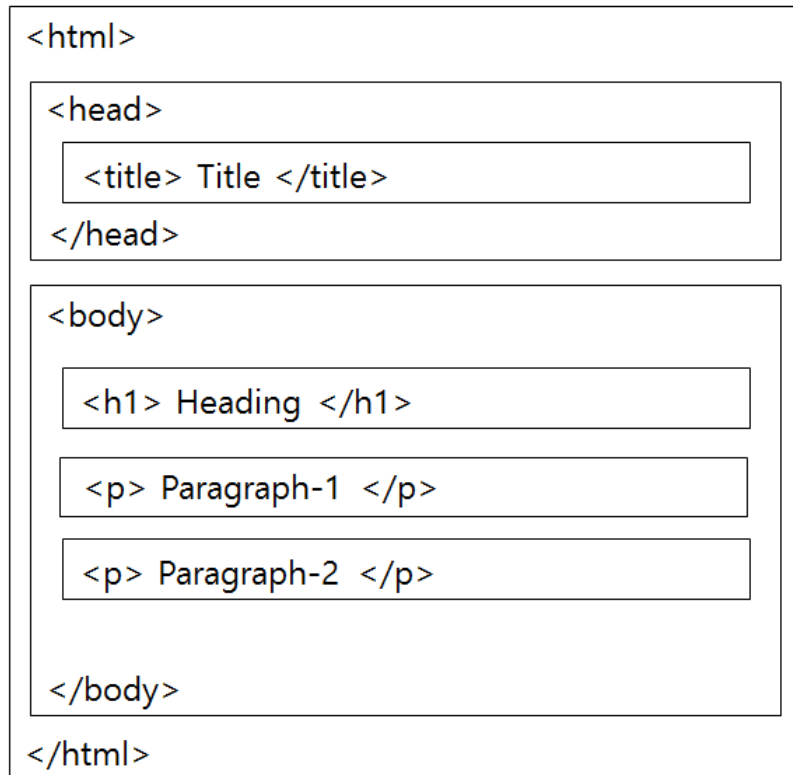
- HTML은 다양한 태그(Tag)들을 이용해 웹 브라우저에 보여질 내용들을 표현한다.



[그림 1-1] HTML 태그

- 웹 브라우저(IE, Chrome, Safari, Firefox 등)는 HTML 태그를 읽고 분석하여 표현 내용을 화면에 나타낸다. 이로 인해 서로 다른 웹브라우저라도 사용자에게는 동일한 형태의 웹 페이지를 표시할 수 있다.

2. HTML 페이지의 구조



[그림 1-2] HTML 페이지의 구조

HTML 페이지는 우선 크게 <head> 영역과 <body> 영역으로 구분할 수 있으며 <head> 영역에는 메타데이터(Metadata)에 대한 정보가 들어간다. 메타데이터는 해당 HTML 페이지의 정보라 할 수 있다. 그리고 실제 웹 브라우저에 표현되는 내용들은 <body> 영역에 위치하게 된다.

3. HTML 태그들의 이해

(1) HTML의 모든 태그들은 시작과 끝 태그로 이루어진다.

(2) HTML 태그들에는 속성을 지정할 수 있다.



[그림 1-3] HTML 태그의 속성

<a> 태그는 속성에 정의된 주소로 이동하게 하는 링크 태그이며 사용자가 이 태그의 내용을 선택하면 링크로 이동한다.

(3) HTML 태그에는 스타일을 적용할 수 있다.



[그림 1-4] HTML 태그의 스타일 속성

HTML 태그의 스타일 속성은 style="property:value" 형태로 지정한다.

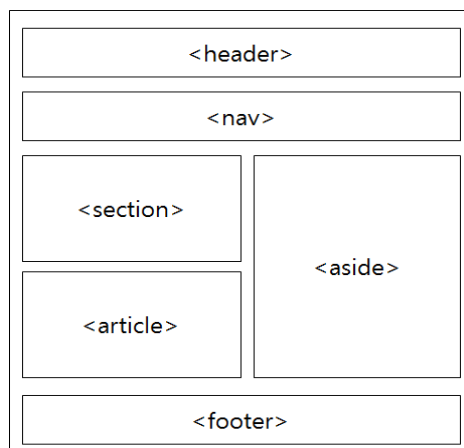
(4) HTML의 리스트 태그들

HTML의 태그들 중에는 리스트 형태로 표현 내용들을 나열할 수 있으며 순서를 갖는 리스트와 순서가 없는 리스트로 표현할 수 있다.

순서 없는 리스트	순서 있는 리스트
<pre> Coffee Tea Milk </pre>	<pre> Coffee Tea Milk </pre>

[그림 1-5] HTML 리스트 관련 태그

(5) HTML의 레이아웃 관련 태그



[그림 1-6] HTML 레이아웃 관련 태그들

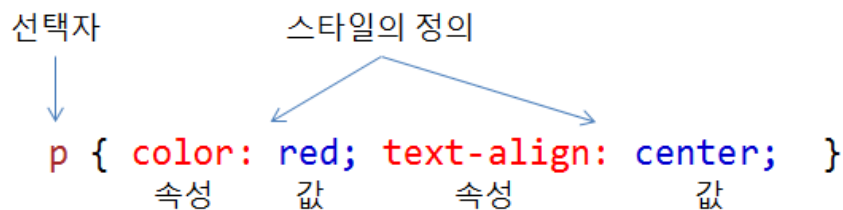
- header: 해당 페이지의 헤더 영역을 지정.
- nav: 본문의 주요 네비게이션 영역을 지정.
- section: 해당 페이지의 콘텐츠 영역을 지정할 때 사용하며 header, footer 태그와 비 교해서 영역을 구분 지정할 때 사용한다.
- article: 독립적인 콘텐츠 항목에 대한 영역을 지정.
- aside: 본문 내용 이외에 표현하고자 하는 기타 내용이 있을 경우 영역을 지정.
- footer: 본문 내용의 아래에 위치하도록 지정.

② CSS(Cascading Style Sheets)

단순 HTML 태그만을 이용해서는 표현하고자 하는 웹 페이지를 표현하는 데 한계가 있다. 다양한 웹 페이지의 표현을 하기 위해서는 CSS를 사용한다. CSS를 이용해 HTML 요소들을 화면에 어떻게 표현할지를 정의한다.

1. CSS 문법

CSS는 HTML 문서내에 작성되거나 외부 파일 형태로 작성 가능하며 표현 방법은 다음과 같다.



[그림 1-7] CSS 문법의 예

- 선택자(Selector)는 HTML 요소를 사용할 수도 있고 id나 class 형태로 정의할 수 있으며 스타일을 적용하기 위한 대상을 뜻한다.
- 스타일은 다양한 형태로 지정할 수 있으며 속성 : 값 형태로 지정한다.

2. CSS 선택자

(1) HTML 요소

HTML 요소를 선택자로 하여 스타일을 적용할 수 있으며 이 경우 해당 HTML 요소는 모두 스타일이 적용된다.

(2) id 선택자

특정 id를 부여하여 여기에 스타일을 지정할 수 있다.

```
#myid {  
    text-align: center;  
    color: red;  
}
```

[그림 1-8] id 선택자의 예

(3) class 선택자

class 선택자를 활용하면 특정 HTML 요소들을 그룹화하여 스타일을 지정하는 것이 가능하다.


```

.center {
    text-align: center;
    color: red;
}

p.center {
    text-align: center;
    color: red;
}

```

[그림 1-9] class 선택자

3. CSS를 적용하는 방법

CSS는 외부, 내부, 인라인(Inline) 스타일 세 가지 형태로 적용할 수 있다.

(1) 외부 스타일 시트의 적용

우선 .css 확장자를 갖는 스타일 시트 파일을 생성하여 이를 외부 링크 형태로 스타일을 적용할 수 있다. 이 경우 하나의 스타일 시트 파일로 다수의 웹 페이지에 동일한 스타일을 적용하는 것이 가능하다.

```

<head>
    <link rel="stylesheet" type="text/css" href="mystyle.css">
</head>

```

[그림 1-10] 외부 스타일 시트 적용의 예

(2) 내부 스타일 시트의 적용

HTML 태그의 <head> 태그 내부에 <style> 태그를 이용해 스타일 시트를 적용할 수 있다. 이 경우 해당 페이지에서만 스타일이 적용된다.

```

<head>
<style>
    body { background-color: linen; }
    h1 { color: maroon; margin-left: 40px;}
</style>
</head>

```

[그림 1-11] 내부 스타일 시트 적용의 예

(3) 인라인 스타일 시트의 적용

인라인 스타일 시트는 하나의 HTML 요소에 스타일을 적용할 수 있다.

<h1 style="color:blue; ">Heading</h1>

[그림 1-12] 인라인 스타일 시트 적용의 예

③ JavaScript

JavaScript는 객체 기반의 프로그래밍 언어로서 웹 브라우저에서 주로 사용된다. JavaScript를 이용하면 웹 페이지에서 발생하는 사용자 이벤트에 대한 처리가 가능하며 JavaScript의 내장 객체를 활용하면 다양한 형태의 웹 페이지를 구현할 수 있다.

1. JavaScript 구현 예

```
<!DOCTYPE html>
<html>
<head>
<script>
function myFunction() {
    document.getElementById("exam").innerHTML = "변경 후"
}
</script>
</head>

<body>

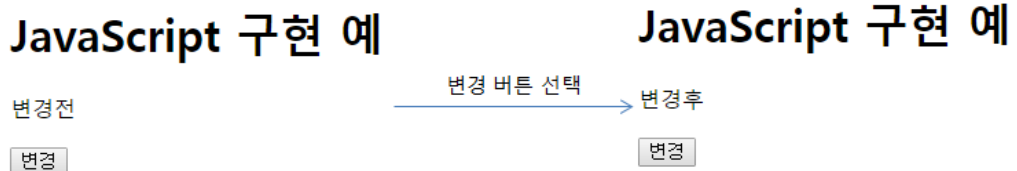
<h1>JavaScript 구현 예</h1>

<p id="exam">변경 전</p>

<button type="button" onclick="myFunction()">변경</button>

</body>
</html>
```

[그림 1-13] JavaScript 구현 예제 소스 코드



[그림 1-14] JavaScript 구현 예제 실행 결과

JavaScript 구현 예제를 보면 JavaScript를 이용해 HTML 페이지의 요소를 변경하는 것을 확인할 수 있다.

2. JavaScript 작성 방법

JavaScript는 HTML 문서 내에서 <script></script> 태그를 통해 작성되며, 작성되는 위치는 <head> 영역, <body> 영역 그리고 .js 확장자를 갖는 외부 파일 형태로 작성할 수 있다.

- <body> 영역 내부에 <script> 태그 내부에 JavaScript 코딩을 작성한다.

```

<!DOCTYPE html>
<html>
<body>

<h1>JavaScript 구현 예</h1>

<p id="exam">변경 전</p>

<button type="button" onclick="myFunction()">변경</button>

<script>
function myFunction() {
document.getElementById("exam").innerHTML = "변경후"
}
</script>
</body>
</html>

```

[그림 1-15] <body> 영역에 JavaScript 작성 예

- .js 확장자를 갖는 별도의 JavaScript 코드 파일을 생성하고 이를 HTML 페이지에 적용할 수 있다. 이 방식의 장점은 동일한 JavaScript 코드를 여러 HTML 페이지에 적용할 수 있다는 것이다.

```

<!DOCTYPE html>
<html>
<body>
<script src="myScript.js"></script>
</body>
</html>

```

[그림 1-16] 외부 JavaScript 파일을 적용하는 방법

3. HTML의 이벤트 처리

<script> 태그 내부의 JavaScript 코드는 해당 HTML 파일이 로딩될 때 한 번 실행된다. 이런 정적인 코드만을 사용할 경우 사용자 이벤트에 대한 다양한 처리에 어려움이 있으며 이를 위해 HTML에서는 HTML 태그의 여러 속성을 통해 JavaScript 코드를 처리할 수 있도록 하고 있다.

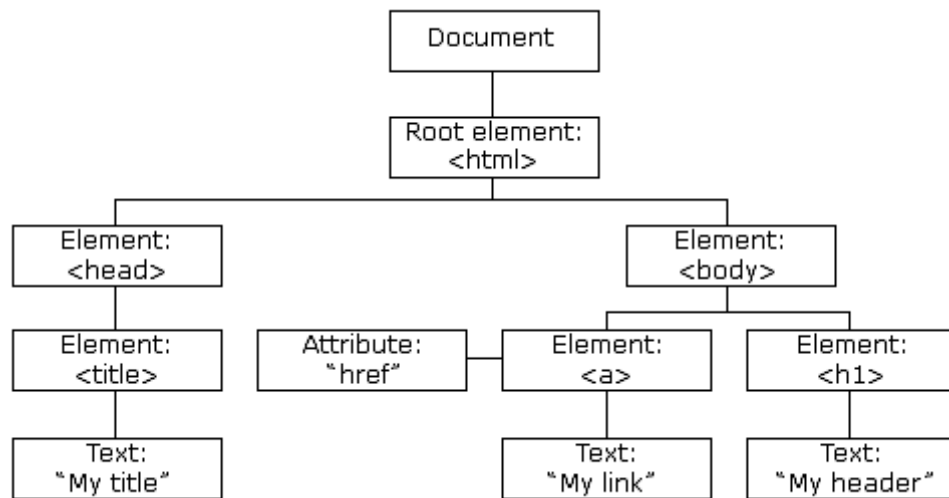
- onclick: 버튼과 같은 HTML의 다양한 폼 요소에서 지원하며 사용자가 해당 요소를 클릭할 때 동작한다. 만일 onclick 핸들러가 false를 반환할 경우 해당 요소의 어떠한 기능도 수행하지 않는다.
- onmousedown, onmouseup: onclick와 유사하게 동작하며 차이는 마우스를 누를 때와 놓았을 때를 구분하여 동작할 수 있다는 것이다.
- onmouseover, onmouseout: 마우스의 포인터가 해당 요소의 위로 올라오거나 벗어날 때 구동한다.
- onchange: <input>, <select>, <textarea> 요소에서 지원하며 해당 요소의 입력 포커스를

다른 곳으로 이동할 때 구동한다

- onload : <body> 태그에서 사용되며 해당 페이지와 해당 페이지에 연결된 외부 내용들이 완전히 로딩되었을 때 구동한다.

4. DOM(Document Object Model)과 JavaScript의 관계

DOM은 JavaScript에서 HTML 페이지의 요소에 접근하거나 변경할 수 있도록 하게 한다. 웹 브라우저는 웹 페이지가 로딩될 때 해당 페이지에 대한 DOM을 생성하며 이 HTML DOM은 해당 페이지 요소들을 계층 구조를 갖는 형태로 나열한다.



[출처]http://www.w3schools.com/js/js_htmlDOM.asp

[그림 1-17] HTML DOM의 예

이 HTML DOM을 통해 JavaScript는 다음과 같이 구현할 수 있다.

- 해당 페이지의 요소 변경이나 추가, 삭제
- 해당 페이지의 요소에 대한 속성 변경이나 추가
- 해당 페이지의 CSS의 변경이나 추가, 삭제
- 새로운 HTML 이벤트의 생성

수행 내용 / UI 구현하기

재료 · 자료

- HTML 편집 도구

기기(장비 · 공구)

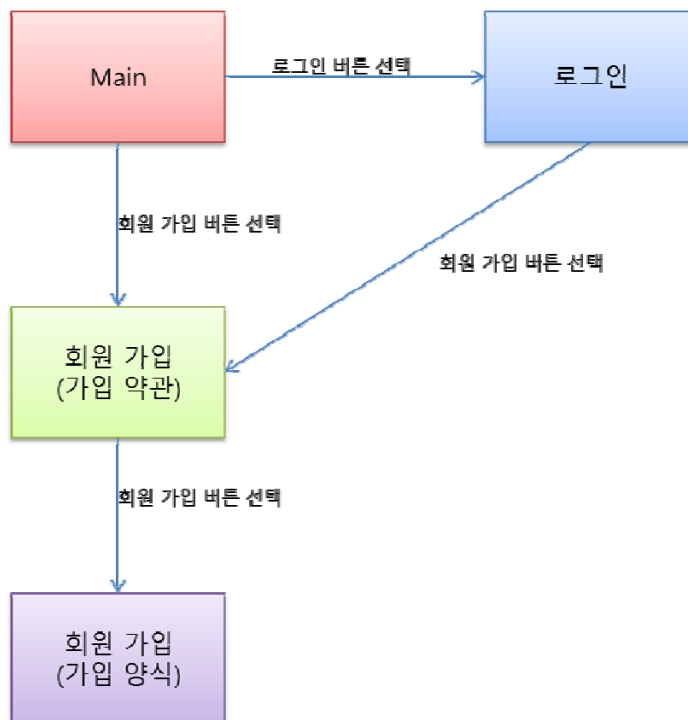
- 컴퓨터, 프린터

안전 · 유의 사항

- 사용자의 사용성을 적극 고려한다.
- 기술적 위험에 대해 고려한다.

수행 순서

① 다음은 실습을 위한 예제 페이지들의 구성에 대한 설명이다.



[그림 1-18] HTML 실습 예제에 대한 전체 페이지의 구성

1. 전체 페이지는 메인, 로그인, 회원 가입(가입 약관), 회원가입(가입 양식) 페이지로 구성한다.
2. 메인 페이지에서는 로그인과 회원 가입 페이지로 이동이 가능하다.
3. 로그인 페이지에서는 회원 가입 페이지로 이동한다.
4. 회원 가입(가입 약관) 페이지에서는 회원 가입(가입 양식) 페이지로 이동한다.
5. 메인 페이지를 제외한 모든 페이지에서는 Back 버튼을 통해 이전 페이지로 이동이 가능하다.

② 다음 메인 페이지의 레이아웃 구성을 보고 HTML, CSS, JavaScript를 활용해 메인 페이지를 구현한다.

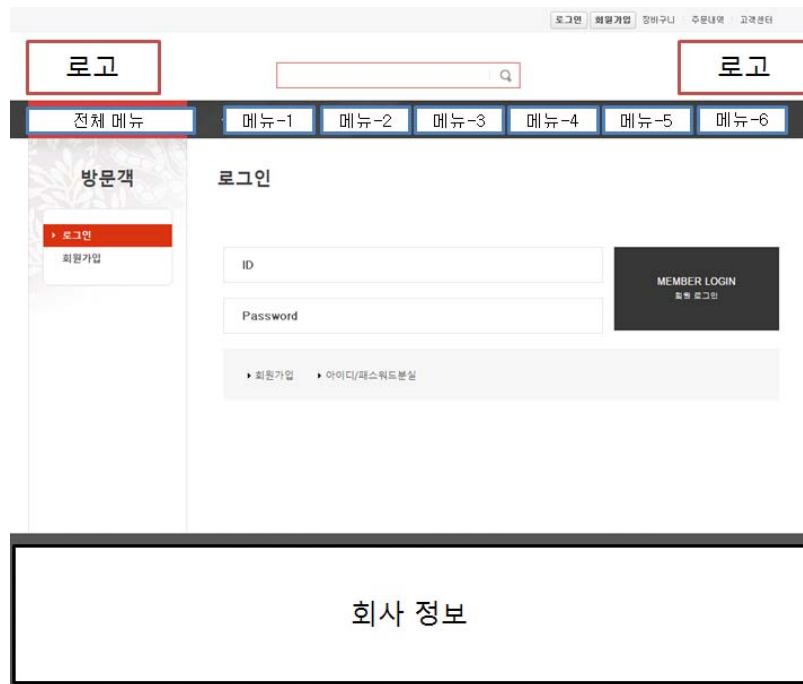


[그림 1-19] 메인 페이지에 대한 레이아웃

1. 메인 페이지의 구성을 확인하고 헤더(Header), 바디(Body), 풋터(Footer) 영역을 구분한다.
2. 각 영역을 구성하는 HTML 요소에 대해서 파악한다.
3. HTML, CSS를 이용해 메인 페이지를 구현한다.
4. 이미지가 사용되는 영역은 임의의 이미지들을 이용해 내용을 구성할 수 있도록 한다.
5. 페이지 이동에 대한 버튼은 로그인과 회원 가입 버튼으로 한다.

6. 로그인 버튼을 선택하면 로그인 페이지로 이동할 수 있도록 구현한다.
7. 회원 가입 버튼을 선택하면 회원 가입(가입 약관) 페이지로 이동할 수 있도록 구현한다.

③ 다음 로그인 페이지의 레이아웃 구성을 보고 HTML, CSS, JavaScript를 활용해 로그인 페이지를 구현한다.



[그림 1-20] 로그인 페이지에 대한 레이아웃

1. 로그인 페이지를 구성하는 HTML 요소에 대해서 파악한다.
2. HTML, CSS를 이용해 로그인 페이지를 구현한다.
2. 이미지가 사용되는 영역은 임의의 이미지들을 이용해 내용을 구성할 수 있도록 한다.
3. 회원 로그인 버튼을 선택하면 ID, Password 영역을 검사하여 입력 내용이 없을 경우 경고창이 화면에 나타날 수 있도록 구현한다.
4. 로그인 버튼을 선택하면 현재 페이지가 새롭게 갱신 되도록 구현한다.
5. 회원가입 버튼을 선택하면 회원 가입(가입 약관) 페이지로 이동할 수 있도록 구현한다.

- ④ 다음 회원 가입(가입 약관) 페이지의 레이아웃 구성을 보고, HTML, CSS, JavaScript를 활용해 회원가 입(가입 약관)페이지를 구현한다.



[그림 1-21] 회원 가입(가입 약관) 페이지에 대한 레이아웃

1. 회원 가입(가입 약관) 페이지를 구성하는 HTML 요소에 대해서 파악한다.
 2. HTML, CSS를 이용해 회원 가입(가입 약관) 페이지를 구현한다.
 3. 이미지가 사용되는 영역은 임의의 이미지들을 이용해 내용을 구성할 수 있도록 한다.
 4. 회원 가입 버튼을 선택하면 약관의 동의에 대한 체크 박스가 선택되었는지 검사하고, 체크되지 않았을 경우 경고창이 화면에 나타날 수 있도록 구현한다.
 5. 회원 가입 버튼을 선택하면 회원 가입(가입 양식) 페이지로 이동할 수 있도록 구현한다.
 6. 이전으로 버튼을 선택하면 로그인 페이지로 이동할 수 있도록 구현한다.
- ⑤ 다음 회원 가입(가입 양식) 페이지의 레이아웃 구성을 보고, HTML, CSS, JavaScript를 활용해 회원 가입(가입 양식) 페이지를 구현한다.

방문객

로그인

회원가입

회원가입

| | | |
|---------|---|-----------------------------|
| 아이디* | <input type="text"/> | 아이디 중복체크 |
| 별명* | <input type="text"/> | 닉네임 중복체크 |
| 비밀번호* | <input type="password"/> | 확인 <input type="password"/> |
| 성명* | <input type="text"/> | |
| 전자우편* | <input type="text"/> | 직접입력 ▼ |
| 주소* | 000-000 <input type="text"/> 우편번호 찾기 <input type="text"/> | 상세주소를 입력해주세요. |
| 전화번호* | 선택 ▼ - <input type="text"/> - <input type="text"/> | |
| 휴대전화* | 선택 ▼ - <input type="text"/> - <input type="text"/> | |
| 생년월일 | <input checked="" type="radio"/> 양력 <input type="radio"/> 음력 선택 ▼ 년 선택 ▼ 월 선택 ▼ 일 | |
| 결혼 | <input checked="" type="radio"/> 미혼 <input type="radio"/> 기혼 | |
| 추천인 아이디 | <input type="text"/> | |

OK
회원가입

BACK
이전으로

[그림 1-22] 회원 가입(가입 양식) 페이지에 대한 레이아웃

1. 회원 가입(가입 양식) 페이지를 구성하는 HTML 요소에 대해서 파악한다.
2. HTML, CSS를 이용해 회원 가입(가입 양식) 페이지를 구현한다.
3. 회원 가입 버튼을 선택하면 생년월일, 결혼, 추천인 아이디를 제외한 항목들에 대해 작성되지 않은 항목에 대한 안내 메시지가 화면에 나타나도록 구현한다.
4. 모든 항목이 작성되고 회원 가입을 선택하면 메인페이지로 이동할 수 있도록 구현한다.
5. 이전으로 버튼을 선택하면 이전 페이지로 이동할 수 있도록 구현한다.

수행 tip

- UI 구현은 여러 웹 페이지들을 참고하고 분석하여 다양한 표현 방법을 익혀야 한다. 따라서 다른 개발자들의 작성 코드들을 분석하는 것이 익숙해야 하며 참고가 될 만한 훌륭한 웹 페이지에 대한 정보를 찾아보는 것이 중요하다.

3-2. UI 테스트

학습 목표

- 구현된 화면, 폼, 메뉴, 흐름을 테스트할 수 있는 테스트 케이스를 작성하고 단위 테스트를 수행하기 위한 테스트 조건을 명세화할 수 있다.

필요 지식 /

① 사용성 테스트(Usability Test)

사용성 테스트 기법은 UI 테스트에 대한 보편적인 방법으로 크게 사용성 테스트에 대한 계획 수립, 사용성 테스트 설계, 사용성 테스트 수행과 결과 검토로 구분하여 진행된다.

1. 사용성 테스트 계획 수립

사용성 테스트 계획 단계에서는 테스트의 목표와 테스트 도구, 테스트 방법 등에 대한 계획을 수립한다.

- 테스트 목적을 명확히 정의한다.
- 테스트에 대한 환경과 사용자를 결정한다.
- 진행할 테스트(Task)에 대한 내용과 질문 사항을 정의한다.

2. 사용성 테스트 설계

- 테스트 참가자 선정: 인원수, 참가 대상 결정
- 테스트 정의: 테스트를 진행하는 인원이 해야 할 과업에 대해 정의한다.
- 체크 리스트 작성

3. 사용성 테스트 수행

- 효율성 체크: 특정 테스트를 진행하는 데 소요되는 시간과 몇 단계를 거쳐야 테스트가 완료되는지를 체크
- 정확성 체크: 테스트를 진행하는 사용자가 얼마나 많은 오류를 일으키는지, 오류를 일으켰을 경우 어떤 결과를 사용자가 보게 되는지를 체크

② 테스트 케이스의 작성

테스트 케이스는 어떤 실행 환경에서 어떤 입력 값들을 부여하여 원하는 결과가 정확히 나오는지를 판단할 수 있도록 작성한다.

1. 테스트 케이스의 대표적인 구성 요소

(1) 식별 번호

- 테스트 케이스를 식별하기 위해 부여하는 번호
- 구분을 위한 식별자와 일련 번호로 구성된다.

(2) 사전 조건

- 테스트가 수행되기 위한 사전 조건에 대한 정보
- 테스트를 위한 데이터에 대한 정의 혹은 구동 환경에 대한 정보

(3) 테스트 수행 절차

- 테스트가 수행되기 위한 절차에 대한 내용

(4) 기대 결과

- 테스트 종료 후에 정확한 동작에 대한 판단을 위한 근거 정보

2. 테스트 케이스의 작성 예

<표 1-1 > 테스트 케이스의 작성 예

| 식별
번호 | 구분 | 사전 조건 | 수행 절차 | 기대 결과 |
|----------|-------------|---|---|--|
| XX.1.1 | ID 중복
체크 | 테스트 데이터
(1234, Hong, 홍,
공백, %\$#@%) | 회원 가입
페이지에서 데이터
입력 후 ID 중복체크
버튼 선택 | 영문 ID를 제외한 다른
데이터 입력에 대한
오류 메시지 출력 |
| .. | .. | .. | .. | .. |

수행 내용 / 테스트 케이스 작성하기

재료 · 자료

- 펜, 종이

기기(장비 · 도구)

- 컴퓨터, 프린터

안전 · 유의 사항

- 사용자의 사용성을 적극 고려한다.
- 기술적 위험에 대해 고려한다.

수행 순서

① 테스트를 진행하기 위한 UI 페이지를 선정한다.

1. 진행한 프로젝트에서 테스트를 진행할 UI 페이지를 선정한다.
 - 사용자의 접근이 많은 페이지를 선정한다.
 - 사용자 입력 내용이 많은 페이지를 선정한다.

② 사용성 테스트를 위한 계획을 수립한다.

1. 테스트를 진행 방식을 결정한다.
2. 테스트를 진행할 인원을 결정한다.
3. 테스트 진행 방식을 논의하고 결정한다.

③ 사용성 테스트를 위한 설계를 진행한다.

④ 테스트 케이스를 작성한다.

⑤ 구현한 웹 페이지들에 대한 테스트를 진행한다.

수행 tip

- UI 테스트는 개발자의 관점이 아닌 사용자의 관점에서 진행하여야 하며 이를 위해서는 객관성을 갖는 테스트 참가자를 선정할 수 있어야 한다.

교수 방법

- 교육생의 HTML, CSS, JavaScript 언어에 대한 이해 정도를 파악한 후 수업을 진행한다.
- 표준이 될 수 있는 다양한 웹 페이지를 바탕으로 HTML 요소와 CSS 표현 기법을 설명한다.
- 웹 페이지에서 발생할 수 있는 다양한 사용자 이벤트들에 대한 예제들을 활용하여 JavaScript 언어를 이용한 이벤트 처리를 설명한다.
- 교육생들이 구현한 UI 페이지를 토대로 테스트를 진행할 수 있도록 실제 테스트 케이스에 대한 예시를 들어 설명한다.
- UI 테스트에 대한 필요성과 이해를 돕기 위한 사례를 통해 교육생들의 이해를 돕도록 한다.

학습 방법

- UI 설계 문서를 토대로 UI 화면을 구성하는 HTML 요소들을 파악하고 CSS를 활용한 페이지 디자인을 학습한다.
- JavaScript 언어를 활용해 다양한 사용자 이벤트 처리에 대해 이해하고, 구현한 UI 페이지에 이를 적용하여 JavaScript 언어를 학습한다.
- 다양한 웹 페이지들을 벤치마킹하고 참고하여 HTML5, CSS3, JavaScript에 대한 활용 방법을 익히도록 학습한다.
- 구현한 UI 페이지에 대한 검토를 통해 부족한 부분이나 확장, 혹은 보완이 필요한 부분들 판단해 보고 구현과 검토의 반복을 통해 UI 페이지 언어에 대한 구현 숙련도를 높인다.
- 구현한 UI 페이지를 대상으로 테스트 케이스를 작성하고 테스트를 진행함으로써 UI 테스트 방법에 대해 학습한다.

학습 3 평가

평가 준거

- 평가자는 학습자가 수행 준거 및 평가 시 고려 사항에 제시되어 있는 내용을 성공적으로 수행하였는지를 평가해야 한다.
- 평가자는 다음 사항을 평가한다.

| 학습 내용 | 평가 항목 | 성취수준 | | |
|--------|--|------|---|---|
| | | 상 | 중 | 하 |
| UI 구현 | - 소프트웨어 아키텍처 세부 구현 지침과 UI 표준 및 지침을 반영하여, UI 요구 사항이 반영된 화면과 폼을 구현할 수 있다. | | | |
| | - 화면과 폼 흐름 설계에 따라, 사용자 접근성을 고려한 화면과 폼의 흐름 제어를 구현할 수 있다. | | | |
| | - 화면과 폼 흐름을 설계에 따라, 감성공학 기법을 고려하여 사용자가 접하는 화면, 폼, 메뉴, 흐름을 구현할 수 있다. | | | |
| UI 테스트 | - 구현된 화면, 폼, 메뉴, 흐름을 테스트할 수 있는 테스트 케이스를 작성하고 단위 테스트를 수행하기 위한 테스트 조건을 명세화 할 수 있다. | | | |

평가 방법

- 평가자 체크 리스트

| 학습 내용 | 평가 항목 | 성취수준 | | |
|--------|--|------|---|---|
| | | 상 | 중 | 하 |
| UI 구현 | - 요구 사항에 만족하는 UI 화면을 구성하고 구현하는 능력 | | | |
| | - 전체 화면들의 구조와 화면 전환에 대한 이해를 바탕으로한 UI 구현 능력 | | | |
| | - 사용자와 UI화면 간의 상호 작용을 이해하고 구현하는 능력 | | | |
| UI 테스트 | - UI 테스트 기법에 대한 이해와 테스트 케이스의 작성 및 테스트를 진행하는 능력 | | | |

- 포트폴리오

| 학습 내용 | 평가 항목 | 성취수준 | | |
|--------|--|------|---|---|
| | | 상 | 중 | 하 |
| UI 구현 | - 요구 사항에 만족하는 UI 화면을 구성하고 구현하는 능력 | | | |
| | - 전체 화면들의 구조와 화면 전환에 대한 이해를 바탕으로한 UI 구현 능력 | | | |
| | - 사용자와 UI화면 간의 상호 작용을 이해하고 구현하는 능력 | | | |
| UI 테스트 | - UI 테스트 기법에 대한 이해와 테스트 케이스의 작성 및 테스트를 진행하는 능력 | | | |

피드백

1. 평가자 체크리스트

- HTML5, CSS3, JavaScript에 대한 이해 정도를 파악하고, 부족한 부분에 대해 정리하여 돌려준다.
- 테스트 케이스 작성에 대해 체크하고 보완이 필요한 부분에 대해 정리하여 돌려준다.

2. 포트폴리오

- 구현한 UI 화면에 대해 평가한 후에 보완할 부분과 잘못된 부분, 잘된 부분에 대하여 표시하여 돌려준다.
- 작성한 테스트 케이스와 이를 통해 테스트한 결과에 대해 부족한 부분에 대하여 표시하여 돌려준다.

참고자료



- 고경희(2015) 『HTML5+CSS3 웹 표준의 정석』, 이지스퍼블리싱 .
- 권기태 · 남영광(2008). 『소프트웨어 공학』, 홍릉과학출판사 .
- 서우중 · 홍태호(2013). 『시스템 분석 및 설계』, 생능출판사 .
- 데이비드 플래너건(2008) 『자바스크립트 완벽 가이드』, 인사이트 .
- 윤인성(2012) 『모던 웹을 위한 JavaScript jQuery 입문』, 한빛미디어.
- 이지현 · 이춘희(2013). 『UX 디자인 프로젝트 가이드 2』, 위키북스 .



UI 시나리오 스토리보드

| Title | | | | | | | |
|---------|--|-----------|--|---------|--|-----|--|
| UI code | | Page name | | 작성자 | | 작성일 | |
| Path | | | | version | | | |
| | | | | 상세 설명 | | | |
| | | | | | | | |

NCS 학습모듈 개발진

(대표집필자)

강석진(이비스툼)

(집필진)

김보운(이화여자대학교)

김홍진(LG CNS)

유은희

장현섭((주)커리텍)

주선태(T3Q)

진권기(이비스툼)

최재준

(검토진)

김승현(경희대학교)

엄기영(우리에프아이에스)

장온순(한국IT컨설팅)

조상욱(세종대학교)

조성호(삼성카드)

(개발기관)

최가원(한국소프트웨어기술진흥협회)

이두현(한국소프트웨어기술진흥협회)

(연구기관)

옥준필(한국직업능력개발원)

김상진(한국직업능력개발원)

김성남(한국직업능력개발원)

김지영(한국직업능력개발원)

문한나(한국직업능력개발원)

홍서희(한국직업능력개발원)

*표시는 NCS 개발진임

※ 본 학습모듈은 자격기본법 시행령 제8조 국가직무능력표준의 활용에 의거하여 개발하였으며
저작권법 25조에 따라 관리됩니다.

※ 본 학습모듈은 <http://www.ncs.go.kr>에서 확인 및 다운로드할 수 있습니다.



www.ncs.go.kr