

반응형 웹 페이지 구성

1. 목표

- HTML을 통한 웹 페이지 마크업 분석
- CSS 라이브러리의 이해와 활용
- 컴포넌트 및 그리스 시스템 활용
- 커뮤니티 서비스 반응형 레이아웃 구성

2. 준비사항

A. 개발도구

- i. Visual Studio Code
- ii. Google Chrome Browser
- iii. Bootstrap v5

B. 제공 Assets

images/	사용할 이미지들이 들어 있습니다.
01_nav_footer.html	모든 페이지가 공유할 Navigation bar 및 Footer입니다.
01_nav_footer.css	Navigation bar 및 Footer의 커스텀 CSS입니다.
02_home.html	사용자가 처음 접속하면 보게 될 페이지입니다.
02_home.css	home.html 의 커스텀 CSS입니다.
03_community.html	사용자들끼리 의견을 나누는 게시판입니다.
03_community.css	community.html 의 스타일의 커스텀 CSS입니다.

3. 요구사항

커뮤니티 서비스 개발을 위한 화면 구성 단계로, 유저가 보는 프론트 엔드를 개발합니다. 아래 기술된 사항은 필수적으로 구현해야 하는 내용입니다.

A. 01_nav_footer

i. 네비게이션 바(Navigation Bar)

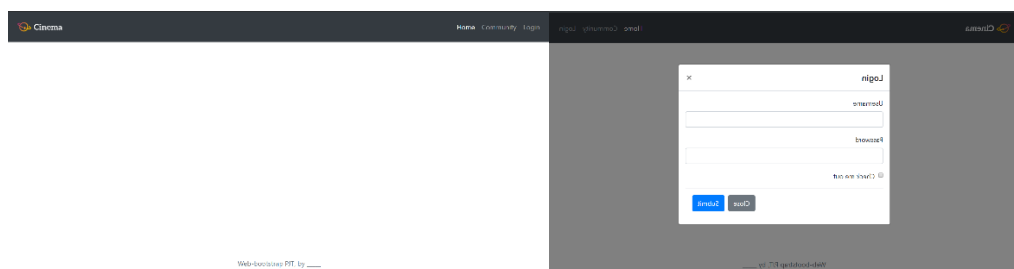
1. 네비게이션 바는 스크롤을 하더라도 항상 상단에 고정되어 있습니다.
2. 로고 이미지는 images 폴더 안의 logo.png파일을 사용합니다.

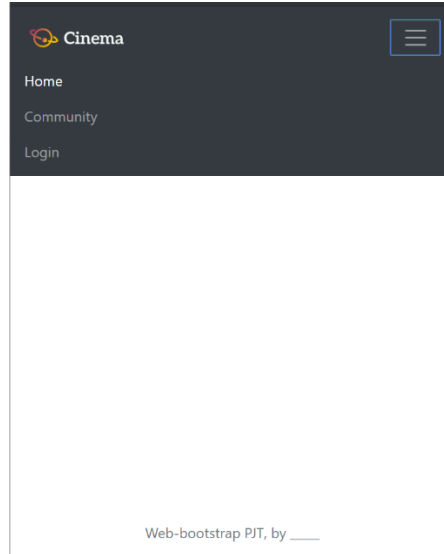
3. 로고 이미지는 클릭이 가능한 링크이며, 해당 페이지(02_home.html)로 이동해야 합니다.
4. 네비게이션 바 내부의 네비게이션 리스트(Home, Community, Login)는 ul 과 li 요소를 사용합니다.
5. 네비게이션 바 내부의 네비게이션 리스트(Home, Community, Login)는 오른쪽에 배치합니다.
6. 네비게이션 리스트의 각 항목들은 클릭이 가능한 링크이며, 해당 페이지 (02_home.html, 03_community.html, #)로 이동해야 합니다.
7. Viewport의 가로 크기가 768px 미만일 경우에는 네비게이션 리스트(Home, Community, Login)가 햄버거 버튼으로 교체되며, 클릭했을 시 세부 항목을 볼 수 있습니다.
8. 네비게이션 리스트(Home, Community, Login)의 항목들 중에서 Home을 [강조합니다.](#)
9. 네비게이션 리스트의 Login 항목은 클릭 시 요소가 [Modal 컴포넌트](#)를 통하여 나타납니다. (페이지 이동이 일어나지 않습니다.)
10. Modal 컴포넌트 내부에는 [form요소](#)를 배치합니다.
11. Modal 컴포넌트에서 form요소 내부의 비밀번호는 표시되지 않습니다.

ii. 푸터(Footer)

1. 푸터는 스크롤을 하더라도 항상 [하단에 고정](#)되어 있습니다.
2. 푸터에 작성된 내용은 수평으로 정렬되어 있습니다. (왼쪽, 오른쪽 여백이 같습니다.)
3. 푸터에 작성된 내용의 빈 부분('____')은 본인의 이름을 작성합니다.
4. 위에 명시된 내용 이외에는 자유롭게 작성합니다.

iii. 결과





B. 02_home

01_nav_footer를 완성했다면, 네비게이션 바와 푸터를 복사하여 사용합니다.

i. 네비게이션 바(Navigation Bar)

1. 네비게이션 리스트(Home, Community, Login)의 항목들 중에서 Home을 강조합니다.
2. Home 페이지는 크게 상단 Header와 하단 Section요소로 이루어져 있습니다.

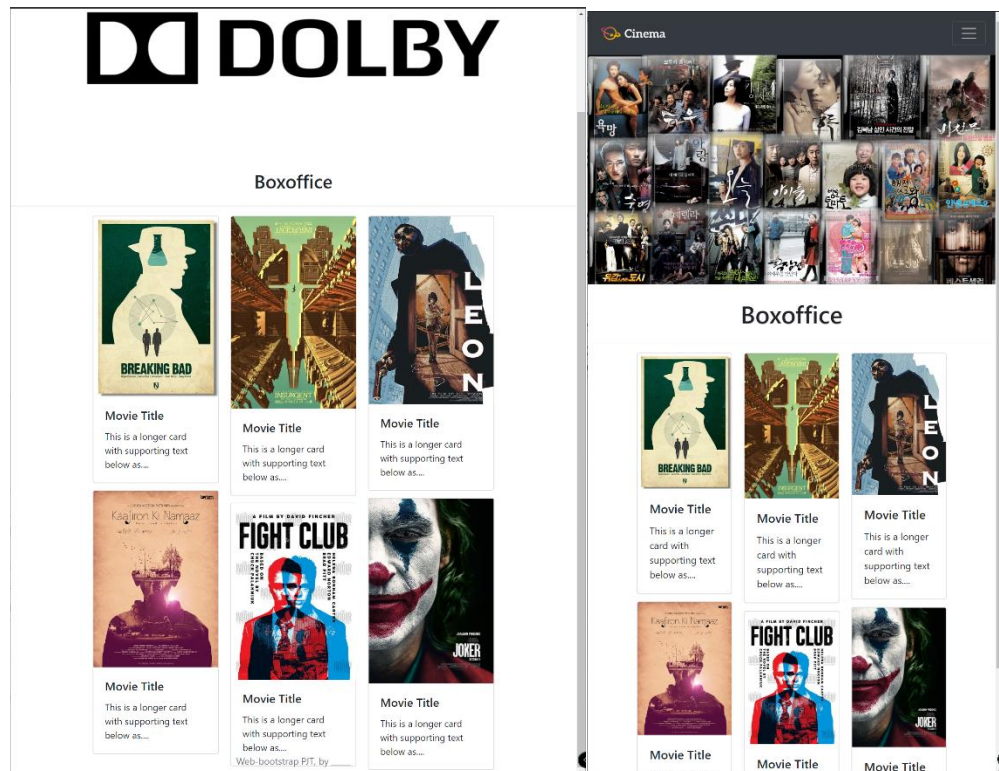
ii. Header

1. Header는 [Carousel 컴포넌트](#)로 구성합니다. 이미지는 최소 3장이며, 자동으로 전환됩니다. (images/ 폴더 안의 header 이미지들을 사용합니다.)

iii. Section

1. Section 내부의 개별 요소(article)들은 이미지, 제목, 설명을 포함하는 [Card 컴포넌트](#)로 구성합니다. (이미지는 images/ 폴더 안의 movie 이미지들을 사용합니다.)
2. 각 요소들은 좌우 일정한 간격으로 떨어져 있습니다. (간격은 자유롭게 설정 가능합니다.) 세로 간격은 일정하지 않아도 됩니다.
3. Section 내부의 요소(article)들은 Viewport의 가로 크기가 576px미만일 경우에는 한 열(row)에 1개씩 표시됩니다.
4. Section 내부의 요소(article)들은 Viewport의 가로 크기가 576px이상일 경우에는 한 열(row)에 2개 이상 자유롭게 표시합니다.
5. 위에 명시된 내용 이외에는 자유롭게 작성합니다.

iv. 결과



C. 03_community

01_nav_footer를 완성했다면, 네비게이션 바와 푸터를 복사하여 사용합니다.

i. 네비게이션 바(Navigation Bar)

1. 네비게이션 리스트(Home, Community, Login)의 항목들 중에서 Community를 강조합니다.
2. Community 페이지는 크게 게시판 목록, 게시판으로 이루어져 있습니다.
3. 게시판 목록과 게시판은 div.main 요소로 둘러싸여 있습니다.

ii. 게시판 목록

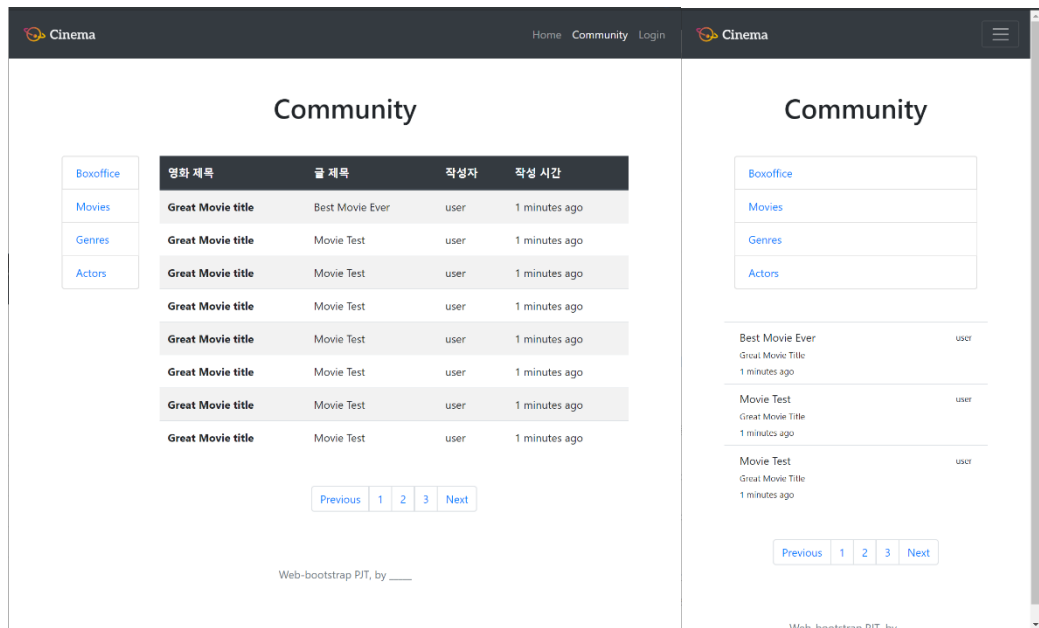
1. 게시판 목록은 aside 요소로 이루어져 있습니다.
2. 게시판 목록 내부의 각 항목들(Boxoffice, Movies, Genres, Actors)은 [List group 컴포넌트](#)를 활용합니다.
3. 게시판 목록 내부의 각 항목들은 클릭이 가능한 링크이며, 모두 동작은 하지 않습니다.

4. Viewport의 가로 크기가 992px 이상일 경우에는 게시판 목록 내부의 항목들(Boxoffice, Movies, Genres, Actors)은 div.main영역의 내부에서 좌측 1/6 만큼의 너비를 가집니다.
5. Viewport의 가로 크기가 992px 미만일 경우에는 게시판 목록 내부의 항목들(Boxoffice, Movies, Genres, Actors)은 div.main영역의 내부에서 전체만큼의 너비를 가집니다.

iii. 게시판

1. 게시판은 Viewport의 가로크기에 따라 전혀 다른 요소를 표시합니다.
2. Viewport의 가로 크기가 992px 이상일 경우에는 게시글들이 [표\(table\)요소](#)로 표시되며, div.main영역의 내부에서 우측 5/6 만큼의 너비를 가집니다.
3. Viewport의 가로 크기가 992px 미만일 경우에는 게시글들이 글(article)요소들의 집합으로 표시되고 가로선으로 구분합니다(스타일링은 자유롭게 진행합니다). div.main영역의 내부에서 전체만큼의 너비를 가집니다.
4. 게시글은 글 제목, 영화 제목, 사용자 id, 작성시간으로 구성되어 있으며 해당 구성을 유지해야 합니다.
5. 테스트 게시글의 개수는 2개 이상으로 자유롭게 구성할 수 있습니다.
6. [게시판 탐색기\(paginator\)](#)는 게시판 아래에 위치하며, 게시판과 같은 가로 폭을 가집니다.
7. 게시판 탐색기(paginator)는 자신의 영역 안에서 좌우 중앙 정렬되어 있습니다.
8. 게시판 탐색기(paginator) 내부의 요소들은 클릭이 가능한 링크이며, 모두 동작은 하지 않습니다.
9. 위에 명시된 내용 이외에는 자유롭게 작성합니다.

iv. 결과



4. 제출 안내

제출 마감시간은 18:00이며, 각 반 교수님을 lab.ssafy.com 프로젝트에 Maintainer로 등록해야 합니다.

반드시 학습 내용 및 소감을 README.md에 기록하여 제출합니다.

위에 명시된 사항은 최소 조건이며 추가적인 정보를 수정할 수 있습니다.

pjt03/

README.md

images/

01_nav_footer.html

01_nav_footer.css

02_home.html

02_home.css

03_community.html

03_community.css