

리스트 UI의 페이지징을 위한 UX

이영주*

*청운대학교 멀티미디어학과
e-mail:yjlee@chungwoon.ac.kr

User Experience for Paging of List UI

Young-Ju Lee*

요약

본 연구에서는 데스크톱 환경에서 긴 스크롤에 대한 피로감을 가지는 콘텐츠의 길이로 인해 제공되던 페이지네이션이 모바일 환경에서 어떠한 변화를 가지게 되었는가 하는 의문으로 시작되었다. 다만 사용자가 모바일 환경에서 세로 방향으로의 페이지징에 거부감을 가지지 않는 바 데이터베이스로 구축되어 무한으로 제공되는 리스트 UI형태의 페이지징에 포커싱하여 사용자 경험에 대해 알아보았다. 리스트 UI에서의 페이지징의 종류는 숫자 페이지네이션과, 자동 무한 페이지징 그리고 수동 페이지징으로 구분할 수 있었다. 숫자로 구현된 페이지네이션은 모바일의 작은 화면에서는 구현되기 어려운 터치 환경을 가지고 있었으며 자동 무한 페이지징은 대량의 리스트 UI를 자동으로 반복하여 제공하기에 적합하다는 것을 알 수 있었다. 하지만 데이터의 과부하와 로딩의 제한이 문제가 될 수 있으며 종종 사용자가 원하는 콘텐츠를 다시 찾기 어렵다는 문제가 있음을 알 수 있었다. 수동 페이지징은 사용자가 직접 버튼을 탭하여 콘텐츠의 전환을 선택할 수 있다는 장점이 있어 데이터의 과부하는 선택적으로 제어할 수 있었다. 하지만 숫자로 된 명확한 페이지네이션이 아니기 때문에 사용자가 현재의 위치를 잃어버리거나 원하는 콘텐츠로의 회귀가 어렵다는 단점은 여전히 존재함을 알 수 있었다.

1. 서론

사용자는 디바이스 내에서 원하는 콘텐츠를 찾으면 위에서 아래로 읽는 습성을 가지고 있으며 이를 통해 안정을 찾는다. 텍스트를 한 줄 소비하면 다음 줄로 시선을 이동하는 것은 자연스러운 일이다. 그로 인해 온라인 콘텐츠는 기본적으로 세로 스크롤을 제공해 왔지만 데스크톱에서의 사용자 경험 연구를 살펴보면 지나치게 긴 스크롤은 시각적 피로감을 제공하기 때문에 긴 스크롤이 필요한 경우에는 콘텐츠를 분할하거나 페이지네이션을 제공하여 다음 페이지로의 이동을 권장하게 하였다. 페이지네이션은 게시판, 갤러리 등과 같이 주로 데이터베이스로 구축되어 무한 반복되는 대량의 콘텐츠에 적합한 UI형태로 이러한 형태의 오늘날 다양한 콘텐츠 서비스에 리스트 UI 형태로 활용되고 있다.

하지만 작은 화면의 모바일 디바이스에서는 작은 버튼으로 제공되는 다음 페이지 버튼이나 숫자로 연속되는 페이지네이

션이 터치인터페이스에 적합하지 않음으로 인해 무한 스크롤을 사용하고 있는 실정이다. 하지만 사용자는 콘텐츠가 언제까지 계속 될지를 알지 못하며 그로 인해 콘텐츠에 대한 부정적 견해가 발생할 수 있다.

따라서 본 연구에서는 모바일에서 사용되는 리스트 UI의 페이지징에 대해 알아보고 리스트 UI의 페이지징에 있어 임의의 양의 정보를 표시하는 올바른 사용자 경험에 대해 분석해 보고자 한다.

2. 모바일 화면 구조의 특징

모바일 환경에서의 콘텐츠는 데스크톱에서의 콘텐츠가 이동되어 온 경우가 많기 때문에 작은 화면에서 데스크톱의 콘텐츠를 보여주기 위해 데스크톱에서 보다 더 많은 콘텐츠의 배치가 발생하게 된다[1]. 따라서 자연스럽게 화면의 구조는 세로 방향의 기다란 구조를 가지고 일부 콘텐츠가 잘려서 보여 지지 않는 화면의 형태를 가지게 된다. 사용자가 처음 디바이스에서 보게 되는 화면의 경우 콘텐츠가 화면에 딱 차게

디자인 되는 경우 사용자는 화면에 스크롤이 작동한다는 것을 간과할 수 있다. 따라서 콘텐츠의 추가 배치를 알게 하기 위해서는 콘텐츠의 일부 또는 문자의 일부가 가려지도록 디자인 되어야 하며 반대의 경우에는 더 이상 페이지가 작동하지 않아 사용자가 스크롤 하지 않도록 콘텐츠 하단에 충분한 여백의 공간을 제공하여 이를 명확하게 해 주어야 한다.

3. 리스트 UI의 페이지 종류와 UX

3.1 숫자 페이지네이션

일반적으로 사용자는 모든 페이지의 콘텐츠를 탐색하면서 위에서 아래로 읽어 내려가는 습성을 가지고 있다[2]. 데스크톱의 환경에서는 콘텐츠의 양이 많아지면 자동으로 페이지의 우측에 스크롤이 생성되었으며 스크롤의 위치에 따라 페이지의 길이를 가늠할 수 있었다. 하지만 데스크톱의 화면을 디자인함에 있어 스크롤의 이동은 콘텐츠의 내용에 대한 인지부하를 발생시키는 요인 중 하나로 인식되어 스크롤을 사용하지 않고 맨 위로 이동할 수 있는 버튼을 활용하거나 화면 내에 서브 메뉴를 배치하여 앵커 기능을 통해 원하는 콘텐츠로의 이동을 원활하게 하는 기능을 추가하는 방식으로 사용자 경험을 개선하였다.

이는 각 메뉴 카테고리의 콘텐츠양이 많지 않은 경우에 주로 사용된 방식이며 방대한 양의 콘텐츠가 제공되는 경우에는 데이터베이스를 게시판 형태로 구축하고 화면 하단에 숫자로 페이지를 만들어 페이지를 넘겨 볼 수 있도록 하였다. 이러한 페이지는 주로 데스크톱에서 사용되어져 왔으나 그 디자인이 그대로 모바일로 옮겨져 한동안 모바일에서도 사용이 되어왔다. 하지만 작은 디바이스에서 숫자로 된 페이지네이션을 터치하기 위해서는 화면을 확대할 필요가 있고 다시 콘텐츠를 보기 위해 확대한 화면을 축소하는 등 여러 번의 불필요한 인터랙션이 발생하면서 무한 스크롤을 사용하기 시작하였다.

3.2 자동 무한 페이지

무한 페이지는 일반적으로 데스크톱의 게시판과 같은 리스트 UI 형태로 제공되며 임의의 콘텐츠 길이를 제공하고 나면 자동으로 현재 페이지에 다음 페이지를 추가로 표시하여 페이지링하는 것이다. 무한 페이지는 한 페이지의 단일 목록에 있는 모든 페이지를 표시하는 것처럼 보이지만 실제로는 콘텐츠를 작은 덩어리로 나눈 뒤 어느 정도 페이지가 내려가면 다음 페이지를 보여주는 방식으로 데이터를 전송해주는 방식으로, 사용자가 계속해서 페이지를 자동으로 스크롤하고 있는 것과 같은 느낌을 준다. 이때 스크롤은 사용자가 화면을 드래그하여 이동하는 만큼 따라 이동하지만 화면의 이동을 멈추

면 스크롤의 위치는 자동으로 화면의 중앙으로 재배치되어 더 많은 콘텐츠가 있음을 예측하게 해 준다.

무한 페이지는 사용자가 화면을 세로로 스크롤 할 때 다음으로 보게 될 가능성이 높은 콘텐츠를 미리 프리패치 하는 방식으로 로드하여 콘텐츠의 양을 사용자의 행동을 기반으로 계산한다. 최적화 된 프리 패치는 실제 페이지 시스템보다는 적은 데이터 전송을 포함 할 수 있지만 사용자가 너무 많이 또는 너무 빨리 스크롤 하면 네트워크에 따라 로딩에 과부하가 걸릴 수 있다. 이때는 반드시 화면 지연과 관련된 알림을 지원해 주어야 한다.

무한 페이지는 스크롤의 위치가 지속적으로 디바이스의 상하 방향의 중앙으로 자동 배치되는 관계로 사용자가 스크롤 목록의 최상단으로 돌아갈 수 있는 버튼을 제공해 주어야 한다. 또 사용자가 선택하여 탭 한 항목의 위치를 잃지 않도록 뒤로 가기 버튼을 선택했을 때 바로 직전의 페이지를 유지할 수 있도록 코딩되는 것이 바람직하다.

3.3 수동 페이지

수동 페이지는 수동 무한 페이지와 수동 제한 페이지로 구분할 수 있다. 수동 무한 페이지는 자동 무한 페이지 방식과 UI 형태는 유사하지만 스크롤의 위치가 수직 방향 상하로 이동을 하였을 때 자동으로 중앙으로 재배치되지 않고 리스트 화면의 하단에 배치되어 화면의 콘텐츠가 더 이상 프리패치 되지 않음을 알려준다. 다만 화면의 맨 마지막 리스트에는 좌우 중앙 정렬되는 더 보기 버튼이 추가되어 사용자가 선택적으로 콘텐츠를 프리패치 할 수 있게 한다는 점에서 자동 무한 페이지와 차이를 둔다.

사용자가 선택적으로 더 보기 버튼을 탭 한 경우 페이지는 자동으로 수직방향으로 상승하며 전체 페이지가 새로운 콘텐츠로 재배치되며 이때 스크롤의 위치는 수직방향 상단의 시작 지점에 배치되어 새로운 콘텐츠가 배치되었음을 알려준다. 이러한 방식의 수동 페이지는 사용자가 더 보기 버튼을 탭 할 때 마다 화면의 전환이 무한으로 반복되어 무한 페이지와 같이 콘텐츠의 전환이 무한으로 반복된다. 자동 무한 페이지에서와의 가장 큰 차이점은 페이지 내에서의 콘텐츠의 시작 지점과 끝 지점을 스크롤의 위치에 따라 파악할 수 있다는 점이지만 무한으로 반복되는 페이지의 전환은 이전의 콘텐츠를 다시 찾아가기 힘들다는 단점이 있으며 이는 자동 무한 페이지에서도 마찬가지이다.

수동 페이지는 수동 제한 페이지의 방식으로도 구현가능하다. 수동 제한 페이지 방식은 수동 무한 페이지 방식과 UI는 비슷하다. 다만 최 하단의 리스트에서 더 보기 버튼과 함께 일정 개수의 리스트 숫자를 보여줄 수 있도록 숫자를 입력해 주면 된다. 입력되는 숫자의 크기는 페이지에 처음 보여 지는

리스트의 개수만큼을 입력하면 사용자가 더 보기 버튼을 탭했을 때 페이징 효과처럼 화면을 새로운 콘텐츠로 리 로딩 하는 것이 가능하다. 다만 페이징처럼 숫자로 페이지를 표시하지는 않기 때문에 때때로 사용자는 수동 무한 페이징과의 차이를 느끼기 어려울 수도 있다[3]. 하지만 페이지 내에서 별도의 스크롤을 사용하지 않고 페이지의 콘텐츠를 전환 할 수 있다는 점에서 수동 무한 페이징과의 차이가 있다고 할 수 있다.

하지만 수동 제한 페이징 방식을 구현하는 경우 사용자는 별도의 화면 스크롤을 사용하지 않고 콘텐츠의 전환이 일어나기 때문에 주의를 집중하지 않으면 화면의 변화를 느끼지 못할 수도 있어 수동 제한 페이징 방식이 구현되는 경우에는 사진이나 이미지가 삽입되어 사용자가 주의를 집중하지 않고도 화면의 변화를 쉽게 느낄 수 있는 조건 하에서 사용하는 것이 바람직하다.

4. 결론

본 연구에서는 콘텐츠의 길이가 데이터베이스로 구축되어 무한으로 제공되는 경우 리스트 UI형태의 페이징의 사용자 경험에 대해 알아보기 위해 페이징의 종류를 숫자 페이지네이션과 자동 무한 페이징, 수동 페이징으로 구분하였다.

숫자 페이지네이션의 경우 데스크톱에 익숙한 UX이지만 모바일 환경에서 사용하기에는 숫자 페이지네이션의 버튼 디자인의 크기가 터치환경에 맞지 않아 권장하지 않는다. 자동 무한 페이징은 대량의 리스트 UI를 자동으로 반복하여 제공하기에 적합하지만 데이터의 과부하와 로딩의 제한이 문제가 될 수 있으며 종종 사용자가 원하는 콘텐츠를 다시 찾기 어렵다는 문제가 있음을 알 수 있었다. 수동 페이징은 수동 무한 페이징과 수동 제한 페이징으로 구분할 수 있었으며 두 경우 모두 사용자가 직접 화면의 끝으로 이동하여 다음 페이지로의 이동 버튼을 탭하여 콘텐츠의 전환을 선택할 수 있다는 장점이 있어 데이터의 과부하는 선택적으로 제어할 수 있다.

하지만 숫자로 된 명확한 페이지네이션을 제공하는 것이 아니기 때문에 여러 페이지의 전환이 일어난 경우는 앞서와 같이 사용자가 현재의 위치를 잃어버리거나 원하는 콘텐츠로의 회귀가 어렵다는 단점은 여전히 존재함을 알 수 있었다. 수동 제한 페이징의 경우에는 좀 더 단순한 페이징을 제공하지만 오히려 정지된 화면에서의 화면 전환이 일어나는 만큼 사용자의 주의를 요하는 UX가 될 가능성이 있으며 따라서 이미지나 사진과 같은 변화가 큰 콘텐츠와 함께 사용되는 경우에 적용하는 것이 바람직하다고 할 수 있다.

본 연구는 작은 화면 안에서 많은 콘텐츠를 무한으로 제공하기 위한 사용자 경험을 위한 기초 연구로서의 가치를 가지

지만 최근 다양한 화면 전환이 단일 화면에서 시도되고 있는 바 세로 방향의 페이징에 대한 연구로서의 한계를 가지고 있다.

참고문헌

- [1] Steven Hoober, “Designing Mobile Interfaces”, O’Reilly, 2012년.
- [2] Jenifer Tidwell, “Designing Interfaces Patterns for Effective Interaction Design”, O’Reilly Media , 2020년.
- [3] 오훈근, “미디어 공간에서 ‘사용자 경험’의 인지적 작용 요소 연구”, 한국공간디자인학회논문집, 제 4권 3호, pp. 35-42, 4월, 2009년.