

Homework 3-1

1. Photo Organization : Requirements

a. Overall Objective of the Application

구글 포토와 같이 사진을 저장하고 볼 수 있는 웹 어플리케이션이다. 사용자가 원하는 사진들을 저장, 삭제하며 사진의 날짜, 사진의 내용에 따라 분류할 수 있다. 또한 사용자는 폴더를 만들어 사진들을 원하는 순서로 관리하고, 검색할 수 있을 뿐 아니라 다양한 SNS 와 웹페이지, 메일로 공유할 수 있다. 웹에 사진을 저장함으로써 사용자가 저장 공간을 들고 다닐 필요가 없도록 하며 동시에 다양한 SNS 를 하는 현대 사용자들에게 최적화된 Photo Organizer 를 제작하는 것을 목표로 한다.

b. Major Functional Requirements

- 사진 및 동영상 관리 : 사진 업로드, 삭제, 순서 정렬, 태그 추가, 복사를 가능하게 한다.
- 사진 보기 : 사진 미리보기가 가능하도록 한다. 사진을 클릭하면 크게 볼 수 있도록 한다.
- 동영상 보기 : 동영상 미리보기가 가능하도록 한다. 동영상을 클릭하면 크게 볼 수 있도록 한다.
- 사진 공유(SNS 연동 필요) : SNS 를 연동하여 한번에 여러 SNS 와 웹페이지에 사진 업로드를 가능하게 한다.
- 폴더 관리 : 폴더 생성, 삭제, 복사, 순서 정렬, 태그 추가를 가능하게 한다
- 날짜 별 사진 분류 : 자동으로 날짜별 사진 분류를 지원한다.
- SNS 별 사진 분류 (SNS 연동 필요): 자동으로 SNS 별 사진 분류를 지원한다.
- 검색 : 사진 검색을 가능하게 한다.
- 간단한 사진 수정 툴 : 사진 자르기, 회전, 수평 맞추기, 음영, 대비 조정을 지원한다.
- 보안 : 업로드 된 사진을 사용자의 허락 없이 접근할 수 없도록 한다.
- 로그인 기능 : 간단한 회원가입과 로그인 기능을 구현한다. 로그인만 하면 별도의 백업 없이도 이전의 정보를 불러올 수 있도록 한다.

c. Major UI Requirements

- Functional

- 사진 쉽게 업로드하기 : Drag and Drop 을 사용하여 사진의 업로드를 쉽게 할 수 있도록 한다.

- 중복된 사진 업로드 탐색하기 : 이미 업로드 되어 있는 사진과 같은 사진을 업로드 하는 것을 막기 위해 중복된 사진 업로드를 탐색하고, 중복된 사진 업로드가 감지되면 사용자에게 경고한다.
- 중복된 사진 이름 탐색하기 : 사진을 업로드 할 때, 중복된 사진 이름을 탐색하여, 같은 이름의 사진이 자동으로 덮어 씌워지는 것을 방지한다. 중복된 사진 이름이 감지 되면 사용자에게 경고한다.
- 편한 검색 기능 : 이전에 했던 검색을 저장하여 자주 하는 검색을 쉽게 하도록 한다. 태그와 날짜를 이용한 검색이 가능하도록 한다. 이를 통해, 태그 별 사진 분류가 가능하도록 한다. 검색의 범위를 넓혀 자동으로 태깅 한다. (ex. 종이를 검색하면 책도 나오도록)
- 주요한 폴더 별표 기능 : 사용자가 자주 사용하는 폴더를 자동으로 중요 폴더로 사용한다.

- **Non-Functional**

- 웹 표준 준수 : 웹 어플리케이션이므로 W3C 와 Ecma 의 웹 표준을 준수한다.
- 창 배치 : 네비게이션 창 위치를 알맞게 배치한다. 일반적으로 대부분의 웹 어플리케이션에서 사용하는 방식을 따른다. 왼쪽 위에 네비게이션 버튼을 두고, 이 것을 누를 경우 네비게이션 창이 띄워지는 방식으로 한다.
- 편한 공유 기능 : 공유 버튼의 위치를 알맞게 조정한다. 동시에 여러 SNS 에 업로드 하는 것이 쉽도록 사진 상세 보기 창에서 공유 버튼을 밖으로 노출 시킨다.
- 사용자 맞춤 : 한화면에 띄우는 사진의 개수를 사용자가 조정 할 수 있도록 한다. 모바일 기기에서는 두 손가락을 사용하여 줌인과 줌 아웃을 할 수 있도록 하여, 한 화면에 띄워지는 사진의 개수를 조정한다.
- 아이콘 기반 : 언어와 상관 없이 앱을 사용할 수 있도록 아이콘 기반의 버튼을 이용한다.

d. Other considerations

- 클라이언트 : 웹브라우저
- Target User : SNS 를 사용하고, 사진을 자주 찍는 20대에서 30대
- Device : 노트북은 물론 SNS 를 자주 사용하는 모바일 기기에서도 사용할 수 있도록 한다. 애플과 안드로이드 기기의 차이를 고려하여 두기기 모두에서 문제 없이 사용 할 수 있도록 한다.
- 자동 태그 자동 분류 : 태그를 자동으로 하기 위해서 어떤 알고리즘을 구현할지 고민한다. 현재 존재하는 Photo organizer 보다 더 좋은 성능을 구현하는 것을 목표로 한다.

2. Photo Organization : User Analysis

a. User Interviews

3 명의 20 대 대학생 을 대상으로 인터뷰를 진행하였다. Target User 들의 현재 사진 관리 현황과 Photo organizer 를 사용하는데 User 의 Requirement 를 알아보는 것을 목표로 인터뷰를 진행하였다. 인터뷰 질문은

- 1) Photo organizer 를 사용해 본 적이 있는가?
 - A. 사용을 해 보았다면 사용하면서 불편한 점이 있는지?
- 2) 사진을 평소에 많이 찍는 편인지?
 - A. 사진을 많이 찍는다면 사진은 어떻게 관리하는가
- 3) 사진은 어떻게 저장하는지?
 - A. 하드에 한다면 웹 저장할 수 있는거 놔두고 왜 하드에 저장하는지?
- 4) Photo Organizer 에 있었으면 하는 기능은?
- 5) SNS 는 사용하는지?
 - A. 사용한다면 사진 공유는 어떻게 하는지?
 - B. 몇가지 SNS 를 사용하는지?
 - C. SNS 에 업로드한 사진들의 관리는 어떻게 하는지?
- 6) Photo Organizer 에 있으면 하는 기능은?

으로 구성하였다.

b. Major direct User Requirements

- 자동 태그 기능 : 사진을 분석하여 자동으로 태깅을 해줬으면 좋겠다.
- 태그 기반 검색 시스템 : 태그를 검색하면 해당 태그가 있는 사진들 모아서 보여주는 기능이 있으면 좋겠다
- 무제한 용량 : 웹 대신 하드 를 사용하는 주요한 이유는 웹에 용량이 한정되어 있기 때문이다.
- 정보 소실 없음 : 한 기기에서 다른 기기로 로그인 할 때 자료소실이 없었으면 좋겠다. (ex. 아이폰에서 윈도우로 사진을 옮기면 아이폰에서 자동으로 태깅 해준 정보가 없어진다.)

c. Major Cognitive/Ergonomic Requirements

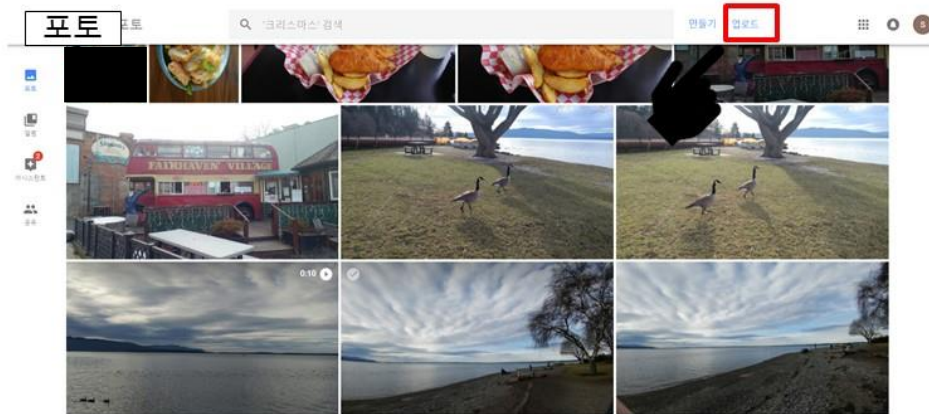
- 모바일 기기 역시 사용 할 수 있는 웹 앱 기반이기 때문에, 랩탑을 사용하지 못하여도 사용 할 수 있어 사용성이 매우 좋다. 때문에 키보드와 마우스를 사용 할 수 있을 정도의 신체능력이 필요하지 않다. 기본적으로 어떤 방식으로든 핸드폰을 터치할 수 있으면 된다.
- 사진을 구분하고 원하는 사진을 선택하기 위한 시각적 능력이 필요하다
- 대부분의 버튼을 언어가 아닌 아이콘으로 구성하였기 때문에 특정한 언어를 알고 있을 필요가 없다. 그러나 아이콘을 이해할 수 있을 정도의 인지 능력이 필요하다.

d. Scenarios

첫번째 시나리오는 Photo organizer 가 켜진 상황에서 photo organizer 에 사진을 업로드 하는 과정이다. 먼저 사진을 업로드 하는 것은 크게 두가지 방법이 있다. 먼저, 업로드 버튼을 눌러서 업로드 하는 방법과 바탕화면이나 폴더에서 바로 웹으로 드래그 앤 드롭을 통해 업로드 하는 방식이다. 두번째 시나리오는 photo organizer 가 켜져 있는 상황에서 페이스북과 인스타그램에 동시에 사진을 공유하는 과정을 서술하였다.

시나리오 1

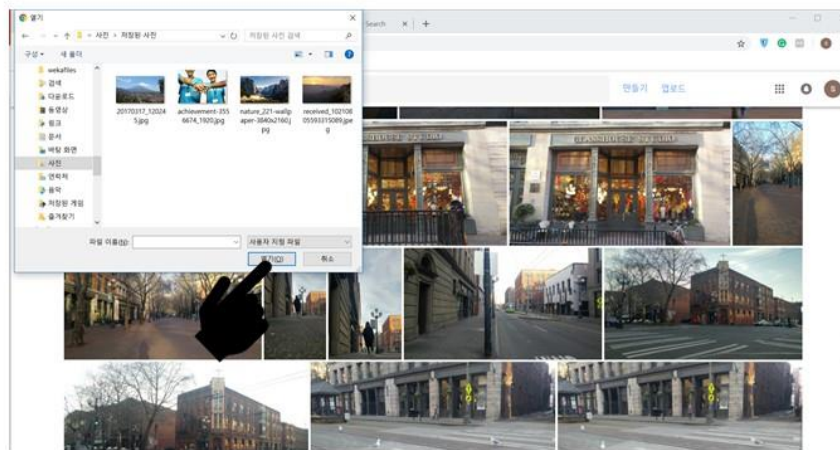
Task : 사진을 데스크탑에서 웹으로 업로드 한다.



1) 앱을 켜다. 업로드 버튼을 클릭한다.

시나리오 1

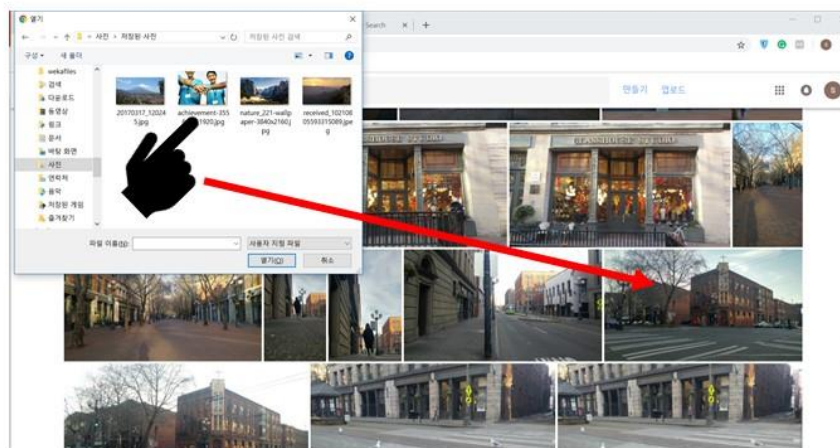
Task : 사진을 데스크탑에서 웹으로 업로드 한다.



2-1)원하는 사진을 클릭한다. 열기 버튼을 누른다.

시나리오 1

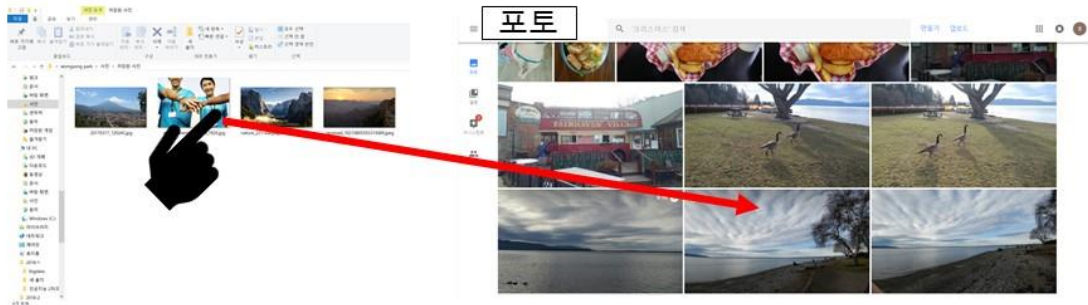
Task : 사진을 데스크탑에서 웹으로 업로드 한다.



2-2)원하는 사진을 클릭해서 웹으로 드래그 한 후 드롭한다.

시나리오 1

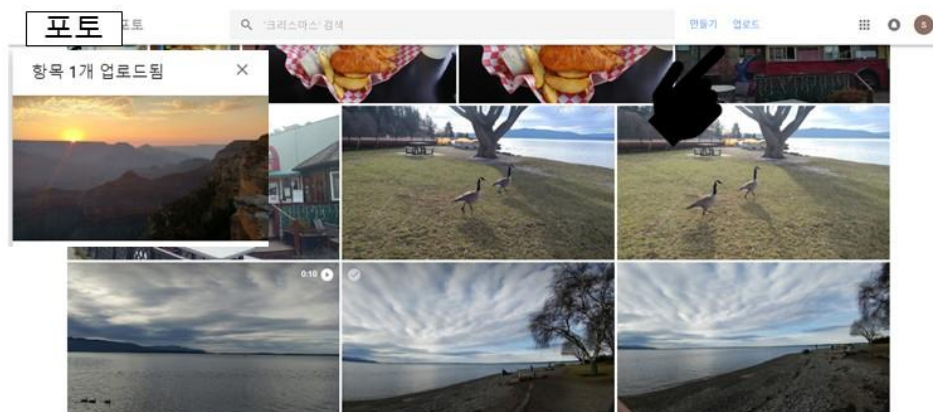
Task : 사진을 데스크탑에서 웹으로 업로드 한다.



2-3) 업로드 버튼을 누르지 않고 바탕화면에서 원하는 사진을 웹으로 드래그 한 후 드롭한다.

시나리오 1

Task : 사진을 데스크탑에서 웹으로 업로드 한다.



3) 업로드가 되었으면, 업로드가 되었다는 팝업 창이 뜬다.

시나리오 2

Task : 페이스북과 인스타그램에 동시에 사진을 공유한다.



시나리오 2

Task : 페이스북과 인스타그램에 동시에 사진을 공유한다.



2) 공유 버튼을 누른 후, 공유하고 싶은 sns를 선택한다.
다음을 눌러 다음화면으로 넘어간다.

시나리오 2

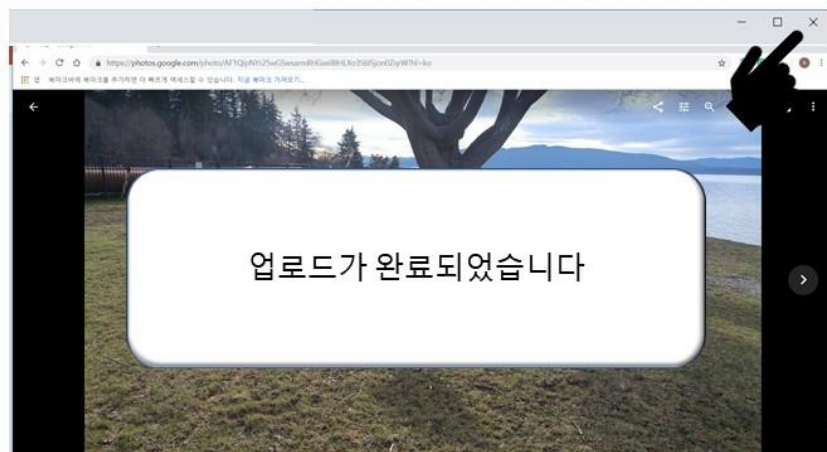
Task :페이스북과 인스타그램에 동시에 사진을 공유한다.



3) 각 SNS의 옆에 있는 텍스트 박스에 적고 싶은 말을 적는다.
업로드를 클릭하여 업로드 한다.

시나리오 2

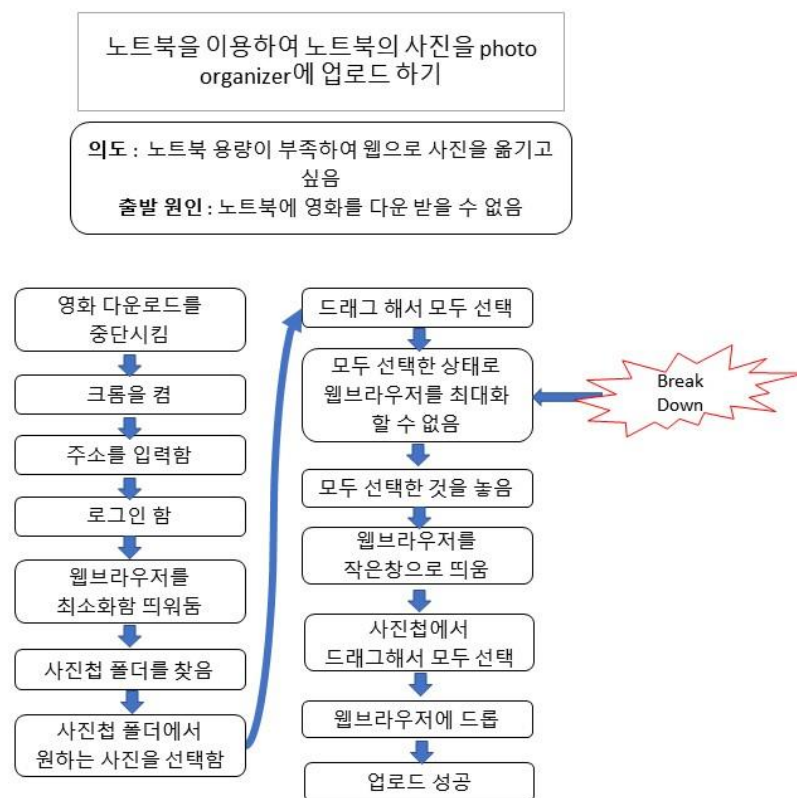
Task :페이스북과 인스타그램에 동시에 사진을 공유한다.



4)업로드가 완료되면 업로드가 완료되었다는 팝업창이 뜬다.
이웹 브라우저 상단의 x 표시를 눌러 웹을 나간다.

e. Rough Interaction model

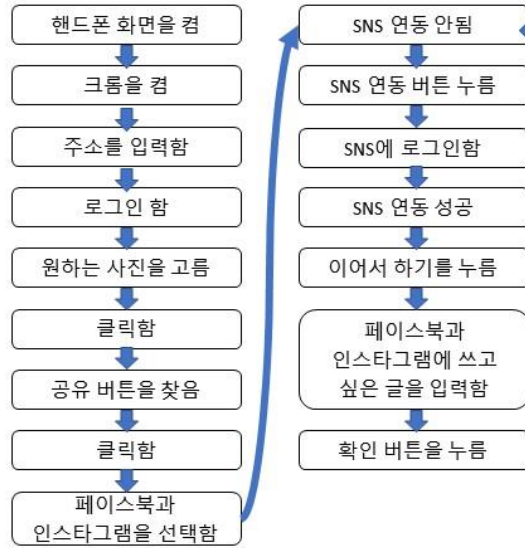
위와 같은 task 에 대해 시퀀스 모델을 사용하여 Interaction model 을 제작하였다. 그러나 위에서는 기기를 켜고 웹 브라우저를 실행시키는 것이 포함되어 있지 않지만, Interaction model 에서는 이를 포함하여 제작하였다. 두 Task 모두에서 사용자가 사용을 하다가 겪을 수 있는 Break down 을 발생시켰으며, 이를 해결하고 task 를 완수하기 까지의 과정을 포함하였다.



핸드폰으로 Photo organizer를 사용하여
페이스북과 인스타그램에 사진 올리기

의도 : 페이스북과 인스타그램에 먹은 음식을
업로드하려고 함

출발 원인 : 먹은 음식을 자랑하고 싶음



Break
Down