



# Image gallery Search

Back-End Engineer

## Intro

Imagine that you are involved in the development of a large file storage system. Special feature here is storing photos and images. We need to provide our users with the possibility to search stored images based on attribute fields.

---

## Requirements

1. We need to see your own code.
  2. The app should load and cache photos from our API endpoint `http://interview.agileengine.com`
  3. Obtain a valid Bearer token with valid API key (don't forget to implement invalid token handler and renewal)  
`POST http://interview.agileengine.com/auth`  
Body: `{ "apiKey": "23567b218376f79d9415" }`  
Response: `{ "token": "ce09287c97bf310284be3c97619158cfed026004" }`
  4. The app should fetch paginated photo feed in JSON format with the following REST API call (GET):  
`GET /images`  
Headers: `Authorization: Bearer ce09287c97bf310284be3c97619158cfed026004`  
Following pages can be retrieved by appending 'page=N' parameter:  
`GET /images?page=2`  
No redundant REST API calls should be triggered by the app.
  5. The app should fetch more photo details (photographer name, better resolution, hashtags) by the following REST API call (GET): `GET /images/${id}`
  6. The app should fetch the entire load of images information upon initialization and perform cache reload once in a defined (configurable) period of time.
  7. The app should provide a new endpoint: `GET /search/${searchTerm}`, that will return all the photos with any of the meta fields (author, camera, tags, etc) matching the search term. The info should be fetched from the local cache, not the external API.
  8. You are free to choose the way you maintain local cache (any implementation of the cache, DB, etc). The search algorithm, however, should be implemented by you.
  9. We value code readability and consistency, and usage of modern community best practices and architectural approaches, as well, as functionality correctness. So pay attention to code quality.
  10. Target completion time is about 2 hours. We would rather see what you were able to do in 2 hours than a full-blown algorithm you've spent days implementing. Note that in addition to quality, time used is also factored into scoring the task.
- 

## Expected Deliverables

1. Source code.
  2. Readme, with instructions, how to build and run.
-