BandLab challenge

Backend Developer

What to build:

- “Imagegram” - a system that allows you to upload images and comment on them (no UI required)

User stories (where the user is an API consumer):

- As a user, I should be able to create new accounts
- As a user, I should be able to create posts with images
- As a user, I should be able to comment on a post
- As a user, I should be able to get a list of comments to a post
- As a user, I should be able to get a list of all posts from all users along with last 3 comments to each post
- As a user, I should be able to delete myself along with all my posts, images, and comments

Functional requirements:

- RESTful Web API (JSON)
- Accounts are authorized via “X-Account-Id: {some-UUID}” header
- API should accept the following image formats: .png, .jpg, .bmp.
- API should provide images only in .jpg format (the originals should be converted to .jpg)
- Posts should be sorted by the number of comments (desc)
- Comments should be sorted by date (desc)
- Any list resources (posts or comments) should support a cursor-based pagination

Non-functional requirements:

- Maximum response time for any API call except uploading image files - 50 ms
- Minimum throughput handled by the system - 100 RPS

Usage forecast:

- 1k uploaded images per 1h
- 100k new comments per 1h

Domain schema:

Account:

- Id
- Name

Post:

- Id
- ImageUrl
- Creator (account)
- CreatedAt (datetime)
- Comments

Comment:

- Id
- Content
- Creator (account)
- CreatedAt (datetime)

Preferable tools:

- ASP.NET Core / Azure Functions / AWS Lambda
- GitHub

Expected result:

- Source code
- Web API documentation

Notes:

- Active public application endpoints are not required
- Develop your system applying the best software development practices
- The non-functional requirements and the usage forecast should be only considered as design guidelines - there is no need to prove it
- You are recommended to spend no more than 5 hours on this challenge
- It’s ok to not cover all the stories - please focus on good design and clean implementation
- If you have any questions feel free to ask