Dreamlines Task

Ahoi and welcome at Dreamlines!
We have prepared a little task for you to show us some of your coding skills.

After you finished the task, please send us an E-Mail with a zip-file or a download-link to the repository. Please attach some information on how we can start the application.
You can use any Node.js frameworks / libraries / servers / (Docker)-containers / databases etc. you think makes sense for the project, just make sure we are able to run your application on our machines.

If you have any questions about the task, feel free to contact me via Skype (yildirim.karal.dl) or email (yildirim.karal@dreamlines.de).

This is your task:

Build a small API with Node.js that provides **JSON** data via **HTTP** to be consumed by a frontend-app or other services.

You will be working with this dataset:

LINK (13mb CSV-file) https://bitbucket.org/netvacation/some-csv-file/raw/fae878b2146adb3c0d753212a0b288b40e529057/airport.csv

Task scope / Story:
We are planning to integrate content of customer airports reviews into several of our customer facing- and internal applications. Therefore we cooperate with a review-service that collects lots of customer reviews about airports. Right now the service supplier sends us a huge CSV-file every month with all of the reviews in the system.
Build a prototype API that provides us with the data from the review service.

Required endpoints for API:

api/all/stats

request: GET
returns: a collection of all airports stats,
the collection should be ordered by the count of reviews
each item in the collection should have the following information:

- names of the airport
- count of reviews in system

api/[airport]/stats

request: GET with [airport] as unique identifier returns: stats for a specific airport:

- airport name,
- count of reviews,
- average “overall_rating”
- count of recommendations “recommended”

api/[airport]/reviews

request: GET with [airport] as unique identifier
returns: a collection of reviews for the given airport
the collection should be ordered by “date”, so the latest review is returned as first element each review in the collection should have the following information:

- overall_rating
- recommendation - date
- author_country
- content

Optional:

- Provide a small user documentation for the API endpoints you implemented.
- Provide tests for your code where you think it makes sense.

- We want to be able to display only reviews that are in a given ‘overall_rating’-ratio for a given airport. For example: we only want the API to return reviews that are 2.0 or above for London Heathrow.

- The current CSV-file just represents the current state; we want to be able to update the data in the system with a “new” version of the CSV-file, implement a process to update the data source.