URL shortener

Assignment: Make an HTTP service that serves to shorten URLs, with the following functionalities:

- Registration Web address (API)
- Redirect client in accordance with the shortened URL
- Usage Statistics (API)

Assignment description:

1. Basic architecture

The service should have two parts: configuration and user.

1.1. Configuration part

The configuration part is invoked using REST calls with JSON payload and is used for:

a) Opening of accounts
b) Registration of URLs in the 'Shortener' service
c) Displaying stats

a) Opening of accounts

HTTP method	POST
URI	/account
Request type	application/json
Request Body	JSON object with the following parameters: • AccountId (String, mandatory) Example: { AccountId : 'myAccountId'}
Reponse Type	application/json
Response	We distinguish the successful from the unsuccessful registration. Unsuccessful registration occurs only if the concerned account ID already exists. The parameters are as follows: • success: true | false • description: Description of status, for example: account with that ID already exists • password: Returns only if the account was successfully created. Automatically generated password length of 8 alphanumeric characters Example {success: 'true', description: 'Your account is opened', password: 'xC345Fc0'}

b) Registration of URLs

HTTP metoda	POST
URI	/register
Request type	application/json
Request Headers	**Authorization** header with Basic authentication token

Request Body	JSON object with the following parameters:
	• url (mandatory, url that needs shortening)
	• redirectType : 301 | 302 (not mandatory, default 302)
	
	Example: {
	url: 'http://stackoverflow.com/questions/1567929/website-safe-data-access-architecture-question?rq=1',
	redirectType : 301
	}

Reponse Type	application/json
Response	Response parameters in case of successful registration are as follows: • shortUrl (shortened URL) Example: { shortUrl: 'http://short.com/xYswlE'}

c) Retrieval of statistics

HTTP metoda	GET
URI	/statistic/{AccountId}
Request Headers	Set ***Authorization*** header and authenticate user
Reponse Type	application/json

Response	The server responds with a JSON object, **key:value** map, where the key is the registered URL, and the value is the number of this URL redirects..
	Example:
	{
	'http://myweb.com/someverylongurl/thensomedirectory/: 10,
	'http://myweb.com/someverylongurl2/thensomedirectory2/: 4,
	'http://myweb.com/someverylongurl3/thensomedirectory3/: 91,
	}

1.2. Redirecting

Redirecting the client on the configured address with the configured http status.

2. General requirements

- Use Java programming language
- Pay attention that the response http statuses comply with the REST standards (list status http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html).
- Service should be written as 'executable' or as 'deployable' package of your choice (eg war, jar)
- The application should not require any additional (external) configuration, meaning it should not have dependencies that aren't declared in POM.
- The application should work out of the box, on first run without any aditional configuration
- In accordance with the above claim, it is not allowed to use databases unless they are embedded, therefore, built into the application itself
- It is allowed to use any framework
- Make a help page (url: /help) containing instructions for installation, launching and usage
- Deliver the source code with all dependencies for Java, preferably as a Maven project

Important note: send your solution through a link on cloud-based sites (Drobpox, Bitbucket, GitHub, WeTransfer, etc..), or otherwise we will not be able to receive it due to possible system safety issues