

## Startup 06 Report

### Report: HTTP Requests / Routes

#### GET /user/:userId

Retrieves the user data for a specific user id. Replaces the getUserData function from the mock server. The HTTP request retrieves /user/:userId. Only a user of matching id may request the userId's data. Example: "A user with ID 1 may issue GET /user/1, whereas another user with differing ID cannot."

#### PUT /user/:userId

Updates a specific user's data. Replaces the updateUserData function from the mock server. The HTTP request updates /user/:userId. User's may only issue this request on their own data. However, Administrators will also have access to change user data for support purposes.

#### POST /email

Sends out an email to contact the developers. Replaces the sendContactEmail function from the mock server. The HTTP request creates a /email which will be sent out as an email to the contact email address. All users may issue this request using the contact page.

#### GET /user/cloudservices

Retrieves the user's cloud services data (which cloud services the user is connected to). Replaces the getUserCloudServices function from the mock server. The HTTP request retrieves /user/cloudservices. A user may only issue this request for their own personal data.

#### GET /user/cloudservices/:service/oauth

Retrieves the user's oauth for the given cloud service. Added function to properly handle retrieval of Google Drive and Dropbox accounts. The HTTP request retrieves /user/cloudservices/:service/oauth. A user may only issue this request for their own personal accounts.

#### GET /user/cloudservices/:service/files

Retrieves the user's specified cloud service files. Replaces the getFilesAndFolders function from the mock server. The HTTP request retrieves /user/cloudservices/:service/files. A user may only issue this request for their own personal accounts.

GET /user/cloudservices/:service/file/:fileId

Retrieves the user's specified file's share link. A user may only issue this request for their own personal accounts.

POST /user/cloudservices/:service

Creates new connection for user to a specific cloud service (Google Drive and Dropbox). Replaces addUserCloudServices function in the mock server. The HTTP request posts /user/cloudservices/:service. A user can issue this request with the click of the Connect Now button. Admins can make this request as well.

DELETE /user/cloudservices/:type

Removes the user's cloud services connection to that specific type of service (Google Drive or Dropbox). Replaces the removeUserCloudServices function from the mock server. The HTTP request deletes /user/cloudservices/:type. A user can issue this request with the click of the Delete Connection button. Admins can make this request as well

POST /device/url

Sends a URL to the socket connection on the device with the same device id as passed in the body. This is used by the mobile app to send a redirect URL to the device which the QR code was scanned off.

**You can also access the application at <https://nectari.me>**

Report: Special Server Setup Procedure

npm install, in both server and client

save 'credentials.json' from moodle page

place 'credentials.json' in server/src

npm run watch, in client

node src/server.js, in server

Report: Individual Contributions

Harsh Poddar - Cloud Services; GET /user/cloudservices,  
GET /user/cloudservices/:service/oauth, GET /user/cloudservices/:service/files  
POST /user/cloudservices/:service, DELETE /user/cloudservices/:type,  
GET /user/cloudservices/:service/file/:fileId, POST /device/url

Jackie Wang - User Profile; PUT /user/:userId

Brian Gregg - Mobile Version

Matthew Hinsley - Contact; POST /email

Ryan Fleischer - User Profile; GET /user/:userId

Cameron Pavao - Contact; POST /email