Restful Service

Database? Text file? Request/response in XML/JSON?

Anything can be used as data access

# Car licenses

No POST operations allowed

PUT happens together with the successful full payment of the license fee from the driver or when an officer sends a rejection to the driver in the final step of the renewal process. Updates the car license with the driver’s temporary address from the renewal notice and the e-mail.

Get license number details: GET: <http://localhost:8080/Assignment2/licenses/_licid> Don’t use endpoint for license retrieval

GET without specifying license number should only be allowed by officer, and should happen when an officer starts the program to check which licenses expires within the next 60 days.

# Renewal notices:

Create renewal notice: POST: <http://localhost:8080/Assignment2/notices/>\_licid . The response’s location is the URL passed to the driver by e-mail. Don’t require license number, create all notices due, return list. Could replace notices with random string mapped to notice.

Keep track of assigned officer ID? Could be done by client app and managed in database

Response in JSON (returns fields for confirmation):

{

"response": {

“HTTPstatus”: “201 Created”

"location": “<http://localhost:8080/Assignment2/notices/_nid>“,

“\_nid”: <\_nid>,

“\_licid”: <\_licid>,  
 “address”: <address>,

“contact\_mail”: <mail>,

“status”: “new”

}

}

Potential XML

<response>

<Httpstatus> 201 Created </Httpstatus>

<\_nid>\_nid </\_nid>

</response>

In case of existing license:

{

"response": {

“HTTPstatus”: “409 Conflict”,

"errormessage": “There is already an existing renewal notice for this license number”

}

}

In case of server error:

{

"response": {

“HTTPstatus”: “500 Internal Error”,

"errormessage": “Something went wrong on the server side”

}

}

Retrieve renewal notice: GET: <http://localhost:8080/Assignment2/notices/_nid>

Response:

{

"response": {

“HTTPstatus”: 200 OK,

“\_nid”: <\_nid>,

“\_licid”: <\_licid>,  
 “address”: <address>,

“contact\_mail”: <mail>,

“status”: <status>

}

}

In case of non-existing notice:

{

"response": {

“HTTPstatus”: “404 Not Found”,

"errormessage": “No notice with that ID was found”

}

}

Update notice: PUT: <http://localhost:8080/Assignment2/notices/_nid> (same as GET) Send all info client side

{

"response": {

“HTTPstatus”: “200 OK”,

“\_nid”: <\_nid>,

“\_licid”: <\_licid>,  
 “address”: <address>,

“contact\_mail”: <mail>,

“status”: <status>

}

}

Delete notice: DELETE: <http://localhost:8080/Assignment2/notices/_nid>

{

"response": {

“HTTPstatus”: “200 OK”

}

}

DELETE should not be allowed to to without specifying \_nid

Running GET on [http://localhost:8080/Assignment2/notices/](http://localhost:8080/Assignment2/newnotice/) will return all notices:

{

“size”: <number of notices>

"response": [{

“HTTPstatus”: “200 OK”,

“\_nid”: <\_nid>,

“\_licid”: <\_licid>,  
 “address”: <address>,

“contact\_mail”: <mail>,

“status”: <status>

}]

}

# Payments

New payment: POST: <http://localhost:8080/Assignment2/notices/payments/_nid>

Using the same ID for payments as notices, supports the HATEOAS principle, making it easier to supply links to the next step in the process by simply adding /payments to the URL

Created either after “All good - yes” at driver side or “Accept - yes” by license renewal officer. Should return error if a payment exists with that renewal notice’s id. Should use PUT if request to extend is yes, POST otherwise.

{

"response": {

“HTTPstatus”: “201 Created”

"location": “<http://localhost:8080/Assignment2/notices/payments/_nid>“,

“\_pid”: <\_pid>,

“\_nid”: <\_nid>,  
 “amount”: <amount>,

“paid\_date”: <date>,

}

}

Update payment: PUT: <http://localhost:8080/Assignment2/notices/payments/_nid>

Only allowed by license renewal officer or updated by the system if the driver makes a partial payment. Returns 200 OK on success with the updated payment as confirmation, 404 NOT FOUND if no payment is found.

Get payment details: GET: <http://localhost:8080/Assignment2/notices/payments/_nid>. 404 NOT FOUND if no payment with that id is found

{

"response": {

“HTTPstatus”: “200 Ok”

"location": “<http://localhost:8080/Assignment2/notices/payments/_nid>“,

“\_pid”: <\_pid>,

“\_nid”: <\_nid>,  
 “amount”: <amount>,

“paid\_date”: <date>

}

}

Get list of all payments if no pid is set. This should only be allowed by an officer

# Rejection

Issued by officer. PUT: <http://localhost:8080/Assignment2/notices/_nid>

Request:

{

“request”: {

“status”: “rejected”

“reason”: “You failed”

}

}

Response:

{

"response": {

“HTTPstatus”: “200 OK”,

“\_nid”: <\_nid>,  
 “address”: <address>,

“contact\_mail”: <mail>,

“status”: “rejected”,

“reason”: “You failed”

}

}

# To next meeting

Set up Git repository

Progress on service implementation

Client app discussion