Project in Software Engineering Course
Specification of REST web API and Command-Line Interface

School of Electrical and Computer Engineering
National Technical University of Athens

Winter Semester 2022-2023

Prof. V.Vescoukis

REST web API: General principles

IntelliQ application must support a RESTful Application Programming Interface (REST API) for managing data and services. The API calls specified in this document must all be implemented by all students. The CLI should make use of the API calls, without implementing the functionality of the calls itself.

API documentation

The REST API should include documentation in the form of JSON file produced by the Postman app.

Base URL

Base URL of the REST API:

✓ https://{{host}}:91003/intelliq_api

Where {{host}} is the localhost or the host of the deployment.

Every individual resource (REST endpoint) which will be available in the REST API, will be accessible through the above base URL, as follows:

{baseURL}/{service}/{path-to-resource}

Where {service} is one of the endpoints, as mentioned below. For example, the endpoint for the retrieval of surveys questions of the questionnaire with ID "poll123" is:

```
https://localhost:91003/intelliq_api/info/poll123
```

Data formats

The REST API will produce data in UTF8 encoding, structured as a JSON string (content-type: application/json) and/or CSV format (content-type: text/csv). If both formats are implemented, the return format will be specified in the request as a query parameter:

{baseURL}/{service}/{path-to-resource}?format={json|csv}

✓ Error Handling

Each call to the API should return an appropriate HTTP status code, in the event of an error. In particular, the following error codes will be returned:

200	Success	Successful call returning a non-empty data payload
400	Bad request	In case the parameters given in a call are not valid (e.g. empty mandatory field)
401	Not authorized	In case the request is made by a non-authorized user
402	No data	In case the response is empty
500	Internal server error	In case of a general (back-end) error

Access and Management

Administrative Endpoints

The backend will support the following management endpoints. In case user authentication is implemented, the following endpoints will be accessible only by users with system administrator role:

✓**{baseURL}/admin/healthcheck**

Supports the GET request method and confirms end-to-end connectivity between the user and the database. That is, the back-end should check the connectivity with the DB to respond to the request. In case of a successful connection a JSON object: {"status":"OK", "dbconnection":[connection string]} is returned, otherwise {"status":"failed", "dbconnection":[connection string]} is returned. The connection string contains the necessary information required for the DB of your choice.

1. **{baseURL}/admin/questionnaire_upd** ✓
   Supports the http POST request method to upload a JSON file with data of a new questionnaire. In the http call, the file must be encoded as a “file” field in multipart/form-data encoding. The required JSON file structure is given in the appendix.
2. **{baseURL}/admin/resetall**
   Supports the http POST request method and initialized all system data, i.e. questionnaires, responses and users (if any). If successful, the JSON object: {"status":"OK"} is returned, otherwise {"status":"failed", "reason":"<...>} with information about the failure.
3. **{baseURL}/admin/resetq/:questionnaireID**
   Supports the http POST request method and deleted all questionnaire responses with id equal to questionnaireID. If successful, a JSON object: {"status":"OK"} is returned, otherwise {"status":"failed", "reason":"<...>} is returned, with information about the failure in the reason field.

System Operation

The 5 endpoints discussed below, will implement the necessary backend functions of the IntellQ software. They need to be implemented exactly as specified.

a. {baseURL}/questionnaire/:questionnaireID

HTTP GET request that returns an object containing the general information and questions of the questionnaire with identifier equal to *questionnaireID*, sorted by *questionID*.

Field	Type	Description
questionnaireID	String	The identification of the questionnaire
questionnaireTitle	String	The title of the questionnaire
keywords	List	List of keywords. Each keyword is a string.
questions	List	List of questions. For each question a JSON object containing the following:
qID	String	The ID of the question
qtext	String	The text of the question
required	String	"true" or "false"
type	String	"question" or "profile"

b. {baseURL}/question/:questionnaireID/:questionID

HTTP GET request that returns an object containing the full details of the *questionID* of the *questionnaireID*. The question options (answers) are sorted by *optID*.

Field	Type	Description
questionnaireID	String	The questionnaire identifier
qID	String	The id of the question
qtext	String	The text of the question
required	String	"true" or "false"
type	String	"question" or "profile"
options	List	List with the options
optID	String	The answer ID
opttxt	String	The text of the answer
nextqID	String	The identification of the next question

c. {baseURL}/doanswer/:questionnaireID/:questionID/:session/:optionID

HTTP POST request that registers the *optionID* response given in the *session* response event to the *questionID* question of *questionnaireID*. It does not return an object. The *session* identifier is a string of 4 random characters that correspond to the event of the questionnaire (note: not the question) being answered by a user.

d. {baseURL}/getsessionanswers/:questionnaireID/:session

HTTP GET request that returns an object containing the responses given to all the questions of *questionnaireID* during the *session* response event. Answers are sorted by *qID*.

Field	Type	Description
questionnaireID	String	The identifier of the questionnaire
session	String	The id of the response event
answers	List	List of selections made
qID	String	The question ID

Field	Type	Description
ans	String	The ID of the response given in the session response event

e. {baseURL}/getquestionanswers/:questionnaireID/:questionID

HTTP GET request that returns an object containing the answers given in all the answer events of *questionnaireID* to the question *questionID*. Answers are sorted by the order in which they were given.

Field	Type	Description
questionnaireID	String	The identifier of the questionnaire
questionID	String	Question Identifier
answers	List	List of selections made
session	String	The ID of the response event
ans	String	The ID of the response given in the session response event

Command Line Interface - CLI

General Principles

The CLI specification includes calls that are equivalent to those of the REST API. The CLI should only be available from the console (command line, ssh) of the system hosting the application you will build.

Data retrieval

The CLI will support JSON and CSV format, just like the REST API. The choice of format will be specified in a corresponding mandatory parameter as mentioned below. The CLI will be invoked from the command line with calls of the form:

```
$ se22XX scope --param1 value1 [--param2 value2 ...] --format fff
```

Where XX is your team ID and SCOPE as listed in the table below. If no parameters are given, the CLI-supported parameters for all scopes should be displayed. For example, assuming se22XX is the CLI executable you implemented, the call to retrieve the answers to question E15 of Q9A is:

```
$ ./se22XX getquestionanswers --questionnaire_id Q9A --question_id E15 --format json
```

The following per-scope parameters should be supported.

Scope	User Authentication (*)	Other Mandatory parameters	Corresponding REST API call
healthcheck	YES	None	/admin/healthcheck
resetall	YES	None	/admin/resetpasses
questionnaire_upd	YES	--source	/admin/questionnaire_upd
resetq	YES	--questionnaire_id	/PassesAnalysis
questionnaire	YES	--questionnaire_id	/questionnaire
question	YES	--questionnaire_id --question_id	/question
doanswer	YES	--questionnaire_id --question_id --session_id --option_id	/doanswer
getsessionanswers	YES	--questionnaire_id --session_id	/getsessionanswers
getquestionanswers	YES	--questionnaire_id --question)id	/getquestionanswers

The --source parameter passes the name of the JSON file containing the questionnaire. The JSON structure is given in the appendix.

The --format parameter is required for all calls and can have a json or csv value

Appendix

Call example in JSON format for /admin/questionnaire_upd

```
{
  "questionnaireID": "QQ000",
  "questionnaireTitle": "My first research questionnaire",
  "keywords": [
    "footbal",
    "islands",
    "timezone"
  ],
  "questions": [
    {
      "qID ": "P00",
      "qtext": "Ποιο είναι το mail σας;",
      "required": "FALSE",
      "type": "profile",
      "options": [
        {
          "optID": "P00TXT",
          "opttxt": "<open string>",
          "nextqID": "P01"
        }
      ]
    },
    {
      "qID ": "P01",
      "qtext": "Ποια είναι η ηλικία σας;",
      "required": "TRUE",
      "type": "profile",
      "options": [
        {
          "optID": "P01A1",
          "opttxt": "<30",
          "nextqID": "Q01"
        },
        {
          "optID": "P01A2",
          "opttxt": "30-50",
          "nextqID": "Q01"
        },
        {
          "optID": "P01A3",
          "opttxt": "50-70",
          "nextqID": "Q01"
        },
        {
          "optID": "P01A4",
          "opttxt": ">70",
          "nextqID": "Q01"
        }
      ]
    },
    {
      "qID ": "Q01",
      "qtext": "Ποιο είναι το αγαπημένο σας χρώμα;",
      "required": "TRUE",
      "type": "question",
      "options": [
        {
          "optID": "Q01A1",
          "opttxt": "Πράσινο",
          "nextqID": "Q02"
        },
        {
          "optID": "Q01A2",
          "opttxt": "Κόκκινο",
          "nextqID": "Q02"
        },
        {
          "optID": "Q01A3",
          "opttxt": "Κίτρινο",
          "nextqID": "Q02"
        }
      ]
    },
    {
      "qID ": "Q02",
      "qtext": "Ασχολείστε με το ποδόσφαιρο;",
      "required": "TRUE",
      "type": "question",
      "options": [
        {
          "optID": "Q02A1",
          "opttxt": "Ναι",
          "nextqID": "Q03"
        },
        {
          "optID": "Q02A2",
          "opttxt": "Οχι",
          "nextqID": "Q04"
        }
      ]
    },
    {
      "qID ": "Q03",
      "qtext": "Τι ομάδα είστε;",
      "required": "TRUE",
      "type": "question",
      "options": [
        {
          "optID": "Q03A1",
          "opttxt": "Παναθηναϊκός",
          "nextqID": "Q04"
        },
        {
          "optID": "Q03A2",
          "opttxt": "Ολυμπιακός ",
          "nextqID": "Q04"
        },
        {
          "optID": "Q03A3",
          "opttxt": "AEK",
          "nextqID": "Q04"
        }
      ]
    },
    {
      "qID ": "Q04",
      "qtext": "Έχετε ζήσει σε νησί;",
      "required": "TRUE",
      "type": "question",
      "options": [
        {
          "optID": "Q04A1",
          "opttxt": "Ναι",
          "nextqID": "Q05"
        },
        {
          "optID": "Q04A2",
          "opttxt": "Οχι",
          "nextqID": "Q06"
        }
      ]
    },
    {
      "qID ": "Q05",
      "qtext": "Με δεδομένο ότι απαντήσατε [*Q04A1] στην ερώτηση [*Q04]: Ποια η
σχέση σας με το θαλάσσιο σκι;",
      "required": "TRUE",
      "type": "question",
      "options": [
        {
          "optID": "Q05A1",
          "opttxt": "Καμία",
          "nextqID": "Q07"
        },
        {
          "optID": "Q05A2",
          "opttxt": "Μικρή",
          "nextqID": "Q07"
        },
        {
          "optID": "Q05A3",
          "opttxt": "Μεγάλη",
          "nextqID": "Q07"
        }
      ]
    },
    {
      "qID ": "Q06",
      "qtext": "Είστε χειμερινός κολυμβητής",
      "required": "TRUE",
      "type": "question",
      "options": [
        {
          "optID": "Q06A1",
          "opttxt": "Ναι",
          "nextqID": "Q07"
        },
        {
          "optID": "Q06A2",
          "opttxt": "Οχι",
          "nextqID": "Q07"
        }
      ]
    },
    {
      "qID ": "Q07",
      "qtext": "Κάνετε χειμερινό σκι;",
      "required": "TRUE",
      "type": "question",
      "options": [
        {
          "optID": "Q07A1",
          "opttxt": "Σπάνια - καθόλου",
          "nextqID": "Q08"
        },
        {
          "optID": "Q07A2",
          "opttxt": "Περιστασιακά",
          "nextqID": "Q08"
        },
        {
          "optID": "Q07A3",
          "opttxt": "Τακτικά",
          "nextqID": "Q08"
        }
      ]
    },
    {
      "qID ": "Q08",
      "qtext": "Συμφωνείτε να αλλάζει η ώρα κάθε χρόνο;",
      "required": "TRUE",
      "type": "question",
      "options": [
        {
          "optID": "Q08A1",
          "opttxt": "Ναι",
          "nextqID": "Q09"
        },
        {
          "optID": "Q08A2",
          "opttxt": "Οχι",
          "nextqID": "-"
        }
      ]
    },
    {
      "qID ": "Q09",
      "qtext": "Με δεδομένο ότι απαντήσατε [*Q08A2] στην ερώτηση [*Q08]: Προτιμάτε
τη θερινή ή την χειμερινή ώρα;",
      "required": "TRUE",
      "type": "question",
      "options": [
        {
          "optID": "Q09A1",
          "opttxt": "θερινή",
          "nextqID": "-"
        },
        {
          "optID": "Q09A2",
          "opttxt": "Χειμερινή",
          "nextqID": "-"
        }
      ]
    }
  ]
}
```

Call example in JSON format for /questionnaire/:questionnaireID

```
{
  "questionnaireID": "QQ000",
  "questionnaireTitle": "My first research questionnaire",
  "keywords": [
    "footbal",
    "islands",
    "timezone"
  ],
  "questions": [
    {
      "qID ": "P00",
      "qtext": "Ποιο είναι το mail σας;",
      "required": "FALSE",
      "type": "profile"
    },
    {
      "qID ": "P01",
      "qtext": "Ποια είναι η ηλικία σας;",
      "required": "TRUE",
      "type": "profile"
    },
    {
      "qID ": "Q01",
      "qtext": "Ποιο είναι το αγαπημένο σας χρώμα;",
      "required": "TRUE",
      "type": "question"
    },
    {
      "qID ": "Q02",
      "qtext": "Ασχολείστε με το ποδόσφαιρο;",
      "required": "TRUE",
      "type": "question"
    },
    {
      "qID ": "Q03",
      "qtext": "Τι ομάδα είστε;",
      "required": "TRUE",
      "type": "question"
    },
    {
      "qID ": "Q04",
      "qtext": "Έχετε ζήσει σε νησί;",
      "required": "TRUE",
      "type": "question"
    },
    {
      "qID ": "Q05",
      "qtext": "Με δεδομένο ότι απαντήσατε [*Q04A1] στην ερώτηση [*Q04]: Ποια η
σχέση σας με το θαλάσσιο σκι;",
      "required": "TRUE",
      "type": "question"
    },
    {
      "qID ": "Q06",
      "qtext": "Είστε χειμερινός κολυμβητής",
      "required": "TRUE",
      "type": "question"
    },
    {
      "qID ": "Q07",
      "qtext": "Κάνετε χειμερινό σκι;",
      "required": "TRUE",
      "type": "question"
    },
    {
      "qID ": "Q08",
      "qtext": "Συμφωνείτε να αλλάζει η ώρα κάθε χρόνο;",
      "required": "TRUE",
      "type": "question"
    },
    {
      "qID ": "Q09",
      "qtext": "Με δεδομένο ότι απαντήσατε [*Q08A2] στην ερώτηση [*Q08]: Προτιμάτε τη θερινή ή την χειμερινή ώρα;",
      "required": "TRUE",
      "type": "question"
    }
  ]
}
```

Call example in JSON format for /getsessionanswers/:questionnaireID/:session

```
{
  "questionnaireID": "QQ000",
  "session": "ATBP",
  "answers": [
    {
      "qID ": "P00",
      "ans": "<*>"
    },
    {
      "qID ": "P01",
      "ans": "P01A2"
    },
    {
      "qID ": "Q01",
      "ans": "Q01A2"
    },
    {
      "qID ": "Q02",
      "ans": "Q02A2"
    },
    {
      "qID ": "Q03",
      "ans": "Q03A3"
    },
    {
      "qID ": "Q04",
      "ans": "Q04A1"
    },
    {
      "qID ": "Q05",
      "ans": "Q05A2"
    },
    {
      "qID ": "Q06",
      "ans": "Q06A2"
    },
    {
      "qID ": "Q07",
      "ans": "Q07A3"
    },
    {
      "qID ": "Q08",
      "ans": "Q08A1"
    },
    {
      "qID ": "Q09",
      "ans": "Q09A2"
    }
  ]
}
```