SE 3XA3: Modulule Interface Specification Plagiarism Check

Team 310, MXQ Gang
Mrinal Tiwari, tiwarim
Qifeng Xu, xuq14
Xu Wang, wangx147

April 7, 2020

Contents

1 MIS of System Utilities

1.1 Interface Syntax

1.1.1 Exported Access Programs

Name	In	Out	Exceptions
UserExists	string	boolean	-
VerifyPW	string, string	boolean	-
CountTokens	string	integer	-

1.2 Interface Semantics

1.2.1 State Variables

Not applicable

1.2.2 Environmental Variables

Not Applicable

1.2.3 Assumptions

‘ There will be no exceptions raised in this module as Utils module handles the exceptions when calling this module.

1.2.4 Access Program Semantics

UserExists(username):
Input: Username as string
Transition: Searches for the username in the database
Output: Boolean True or False
Exception: none

VerifyPW(username, password):
Input: Username as string, password as string
Transition: Compares the hashed password stored in the database to the password given as input
Output: Boolean True or False
Exception: none

CountTokens(username):
Input: Username as string
Transition: Searches for number of tokens stored in the database for that user.
Output: Returns the number of tokens as integer
Exception: none

2 MIS of Register Resource

2.1 Interface Syntax

2.1.1 Exported Access Programs

Name	URL	In	Out	Exceptions
post	/register	json	json	301

2.2 Interface Semantics

2.2.1 State Variables

namepass: Json input containing the name and password of the user to register.
name: user's name as string extracted from the Json.
password: user's password as string extracted from the Json

2.2.2 Environmental Variables

Not Applicable

2.2.3 Assumptions

It is assumed that user would not send a invalid json like json containing invalid number of inputs.

2.2.4 Access Program Semantics

post(Resource):
Input: Resource as json that is passed over the register URL of Restful API
Transition: Updates the database with the new user credential
Output: retJson as json that contains the reply message and status code
Exception: 301 - user already exists

3 MIS of Detect Resource

3.1 Interface Syntax

3.1.1 Exported Access Programs

Name	URL	In	Out	Exceptions
post	/detect	json	json	301, 302, 303

3.2 Interface Semantics

3.2.1 State Variables

namepassref : Json input containing username, admin password and refill amount.
name: user's name as string extracted from the Json.
password: User's password as string extracted from the Json
text1: First text as string extracted from the json
text2: Second text as string extracted from the json
~~admin_password: admin password as string extracted from the Json~~
~~refill_amt: the amount of tokens to refill as integer extracted from the Json~~

3.2.2 Environmental Variables

Not Applicable

3.2.3 Assumptions

It is assumed that user would not send a invalid json containing invalid number or type of inputs.

3.2.4 Access Program Semantics

post(Resource):
Input: Resource as json that is passed over the refill URL of Restful API
Transition: Adds the refill amount to the existing number of tokens
Output: retJson as json that contains the reply message and status code
Exception: 301 - invalid user, 302 - invalid password, 303 - out of tokens

4 MIS of Refill Resource

4.1 Interface Syntax

4.1.1 Exported Access Programs

Name	URL	In	Out	Exceptions
post	/refill	json	json	301,302, 304

4.2 Interface Semantics

4.2.1 State Variables

namepass: Json input containing the name and password of the user to register.
admin_password: admin password as string extracted from the Json.
refill_amt: the amount of tokens to refill as integer extracted from the Json.

4.2.2 Environmental Variables

Not Applicable

4.2.3 Assumptions

It is assumed that user would not send a invalid json like json containing invalid number of inputs.

4.2.4 Access Program Semantics

post(Resource):
Input: Resource as json that is passed over the register URL of Restful API
Transition: Updates the database with the new user credential
Output: retJson as json that contains the reply message and status code
Exception: 301 - invalid user, 302 - invalid password, 304 - wrong admin password

5 MIS of Integration Module

5.1 Interface Syntax

5.1.1 Exported Access Programs

Name	In	Out	Exceptions
createCorsrequest	method, url	CorsRequest	Error 404
makeCorsRequestRegister	string, string	-	Error 500
Registersend	-	-	-
makeCorsRequestCheck	string, string, string, string	-	Error 500
Checksend	-	-	-
makeCorsRequestRefill	string, string, int	-	Error 500
Refillsend	-	-	-

5.2 Interface Semantics

5.2.1 State Variables

Not applicable

5.2.2 Environmental Variables

~~Not Applicable~~
username : stores username entered by user in the textbox using keyboard
password : stores password entered by user in the textbox using keyboard
text1 : stores first text entered by user in the textbox using keyboard
text2 : stores second entered by user in the textbox using keyboard
adminPassword : stores admin password entered by user in the textbox using keyboard
Submit : responds to click made by user

5.2.3 Assumptions

It is assumed that the backend is running and is properly connected through Amazon Web Services.
~~There will be no exceptions raised in this module as Utils module handles the exceptions when calling this module.~~

5.2.4 Access Program Semantics

~~UserExists(username):~~
~~Input: Username as string~~
~~Transition: Searches for the username in the database~~
~~Output: Boolean True or False~~
~~Exception: none~~

~~VerifyPW(username, password):~~
~~Input: Username as string, password as string~~
~~Transition: Compares the hashed password stored in the database to the password given as input~~
~~Output: Boolean True or False~~
~~Exception: none~~

~~CountTokens(username):~~

~~Input: Username as string~~
~~Transition: Searches for number of tokens stored in the database for that user.~~
~~Output: Returns the number of tokens as integer~~
~~Exception: none~~

createCorsRequest(method, url):
Input: method in terms of GET or POSt, url as URL of the REST API
Transition: creates a HTTP Cors request of the url on the preferred method
Output: Returns the Cors request
Exception: Error 404 URL not found

makeCorsRequestRegister(uname, pass):
Input: username as string, password as string
Transition: Makes a HTTP request call of the url with username and password
Output: -
Exception: Error 500 Internal server error

RegisterSend():
Input: -
Transition: accepts the username and password from the text box
Output: -
Exception:-

makeCorsRequestCheck(uname, pass, text1, text2):
Input: username as string, password as string, text1 as string, text2 as string
Transition: Makes a HTTP request call of the url with username, password, text and text2
Output: -
Exception: Error 500 Internal server error

CheckSend():
Input: -
Transition: accepts the username, password, text1 and text2 from the text box
Output: -
Exception:-

makeCorsRequestRefill(uname, adminPass, refillAmt):
Input: username as string, admin password as string, refill amount as int
Transition: Makes a HTTP request call of the url with username, admin password and refill amount
Output: -
Exception: Error 500 Internal server error

RefillSend():
Input: -
Transition: accepts the username, admin password and refill amount from the text box
Output: -
Exception:-

6 Major Revision History

Date	Revision
February 3, 2020	Rough draft of sections
February 15, 2019	Revised sections
February 16, 2019	Revision 0 complete
March 12, 2020	MIS, Module Guide, Requirements Document, Test Plan, and Test Report revision 0
April 6, 2020	Updated sections, updated Integration Module