**Team 7 (Voice) Business Contract**

Josh Corbin, Devin Kloos, Kevin Copper, Patrick Donavan, Donald Taylor

**Software Development Agreement**

This Software Development Agreement is made on April 11, 2017 between Team 7, whose principal place of residence is at Wright State University, and Dr. Nikolaos Bourbakis, a CEG4110 corporation with its principal place of business located at Room 207, Student Success Center.

The parties agree as follows:

1. **Definitions.** In addition to the terms defined above, the following definitions apply:

1.1. "Confidential Information" means all business-related information, written or oral, that the Customer discloses or makes available to the Developer, directly or indirectly, through any means of communication or observation.

1.2. "Software" means the programs created/used for the medical device voice detection system, and all associated documentation and other instructions.

1.3. "Specifications" has the meaning given to it in section 3.1.

2. **Software Development Services.** The Customer engages the Developer, and the Developer agrees, to perform services for the Customer to develop, deliver, and install/integrate the Voice Software in accordance with the terms of this agreement.

3. **Developer's Duties and Responsibilities**

3.1. Specifications. The Customer shall define the specifications, requirements, and deliverables.

3.2. Development. The Developer shall design, develop, and install/implement the Voice Software in accordance with the Specifications.

3.3. Delivery. The Developer shall use reasonable efforts to deliver the Voice Software to the Customer, no later than 1 day after the Effective Date OR as soon as practicable in accordance with the Specifications.

3.4. Installation. The Developer and other Teams will be responsible for conducting all activities required to integrate the voice module and install the complete Software at the Customer's premises.

4. **Acceptance**

4.1. Acceptance Period. The Customer will have 10 days following the date of installation/integration to assess and test the Software.

4.2. Completion. If the Developer in the Customer's opinion, delivers the Software in accordance with the Specifications, then the Developer will be deemed to have completed its delivery obligations.

4.3. Rejection. If the Developer in the Customer's opinion, fails to deliver the Software in accordance with the Specifications, the Customer shall detail in writing its grounds for rejection. In that case, the Developer shall promptly OR use reasonable efforts to correct the Software, in which case upon delivery of the corrected Software, the process of acceptance testing will restart.

4.4. Continued Failure. If the Developer's corrections, in the Customer's opinion, fail to deliver the Software in accordance with the Specifications, then the Customer may elect to either terminate this agreement or adjust the Specifications accordingly.

5. **Project Team.** The Voice “Project” shall be delivered primarily by Team 7. Team size will be 5 members and will not vary. Due to the fixed price nature of this statement of work, it is the responsibility of Team 7 to guarantee delivery of required functionality on specified date and hence to assign an appropriate team to meet the deadline.

“Client” (Dr. Bourbakis) will provide the necessary personnel to work alongside this team and provide timely feedback and any necessary information.

In addition to the stated above, Team 7 agrees to assign named personnel to this project that is deemed critical for the success of the “Project” (Voice) by the “Client” (Dr.Bourbakis). The list of such personnel is:

• Josh Corbin, Requirements

• Devin Kloos, Analysis

• Kevin Copper, Design

• Patrick Donavan, Implementation

• Donald Taylor, Testing

6. **Fees, Expenses, and Time**

6.1. Fees. The Customer shall pay the Developer

(a) $5,000 upon signature of this agreement as a down payment, and

(b) $200 for every hour spent working on the Voice Software.

6.2. Expenses. The Customer shall reimburse the Developer for all reasonable expenses that the Developer incurs in developing the Software, but only if the Customer has given its prior approval to the expenses.

6.3 Time. The Voice Project shall take the Developers (Team 7) roughly 40 hours.

6.4. Payment Due Date. All fees under this agreement will be due and payable in full to the Developer no later than 10 days after the date of the Developer's invoice.

6.5. Late Payments. Any amount not paid when due will bear interest from the due date until paid at a rate equal to [1]% per month ([12.68]% annually) or the maximum allowed by law, whichever is less.

7. **Requirements**

7.1 Project Outline. The voice recognition software will be used in a health care setting for Doctor's to be able to access and read out important and confidential information on their patient's medical device such as vitals and other data. The voice software will define simple words to use as "passwords". Then the voice software will check the pattern of the doctor's voice. If the voice recording is above 75% matching for the authorized Doctor, then he/she will be authenticated and be granted access to that medical devices data. If the voice pattern does not exceed 75% matching, then he/she can try again. Lastly, if the attempts to authenticate fail 3 or more times, then the device decides that this user attempting to authenticate is not the authorized Doctor and will not allow them access.

7.2 Application Domain. Health Care, Doctor's Office, Hospital

7.3 Need. Effective voice detection is used to authenticate a user (Doctor) so that he/she can read out vitals, data, information from that patient's medical device.

7.4 Users. People using this software will be the doctor of that patient.

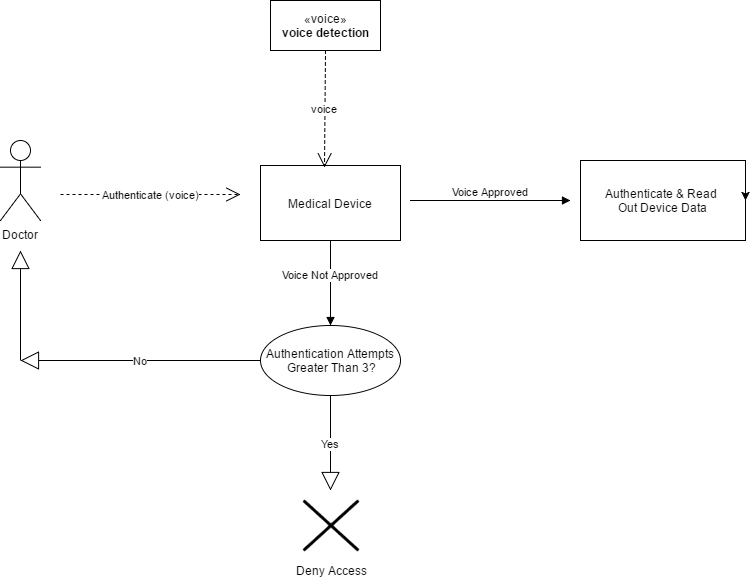
7.5 Deadline. April 27th, 2017.

7.6 Reliability. Product must be operational 99% of the time or the mean time between failures must be at least 4 months.

7.7 HIPPA. The product must be effective and accurate to ensure HIPPA/privacy laws are enforced. Only the authorized doctor should be granted access to the medical device's data which they are authorized to access (their own assigned patient's medical device).

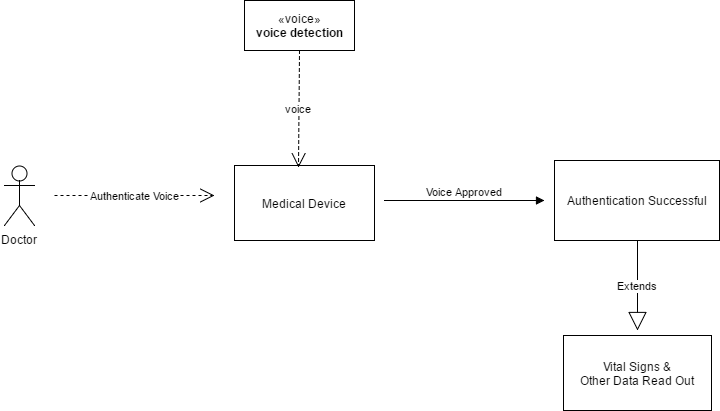
7.8 Glossary. "Confidential Information" means all business-related information, written or oral, that the Customer discloses or makes available to the Developer, directly or indirectly, through any means of communication or observation. "Software" means the programs created/used for the medical device voice detection system, and all associated documentation and other instructions.

7.9 Requirements Business Model.

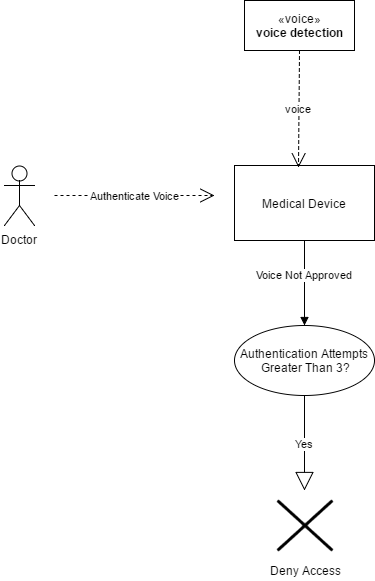


7.10 Requirements - Use Case Models.

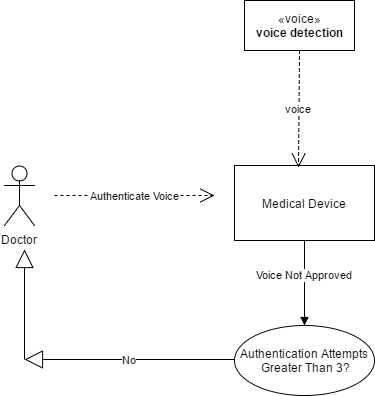
7.10.1 Doctor Successfully Authenticated.



7.10.2 Authentication Failed 3+ Times (Access Denied).

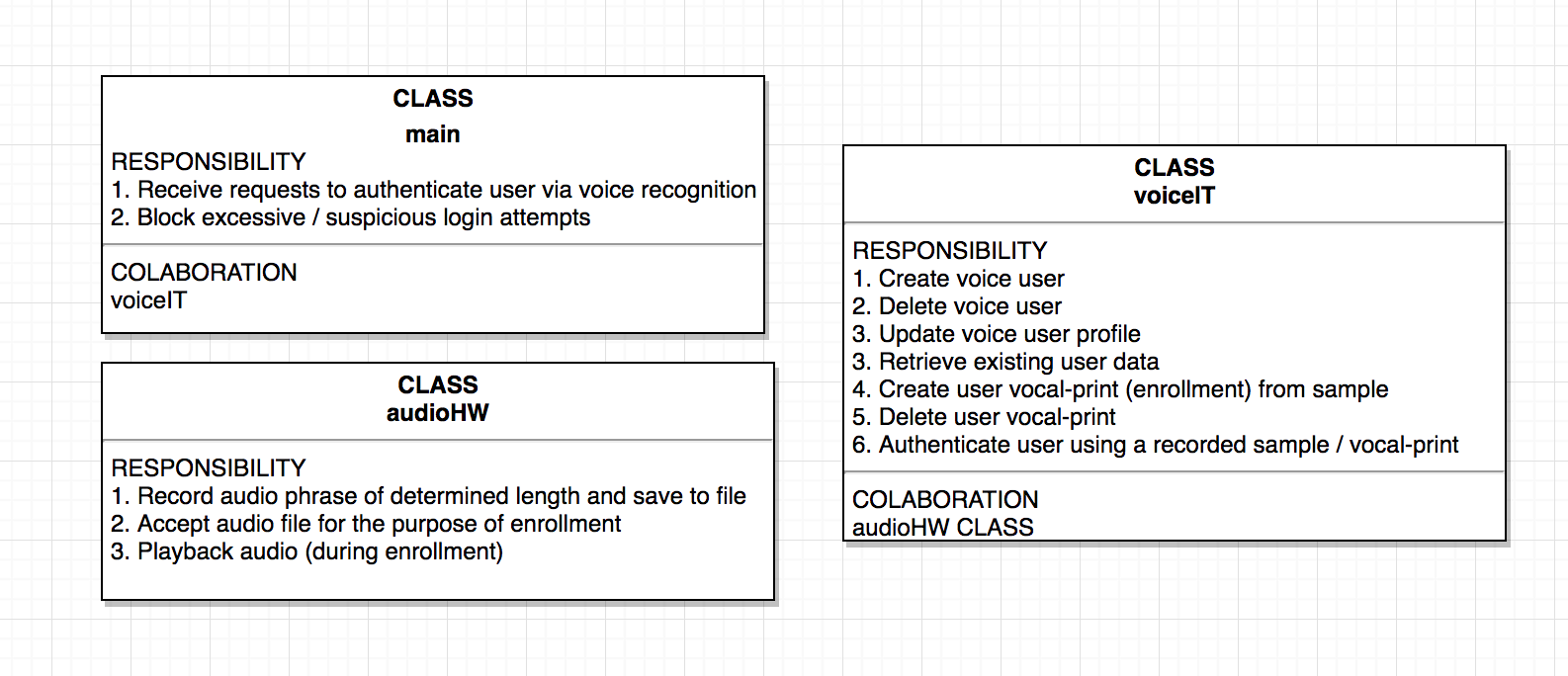


7.10.3 Authentication Failed Less Than 3 Times (Re-Attempt Authentication).

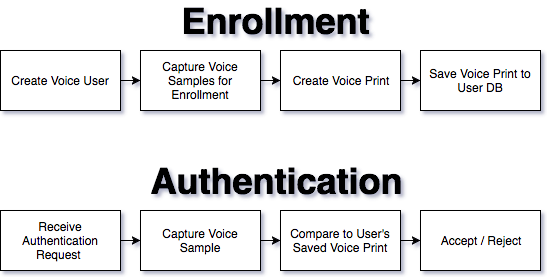


8. **Analysis**

8.1Analysis Class Responsibility Collaboration (CRC) Card.

****

8.2 Analysis Communication Diagram 1.

****

8.3 Analysis Communication Diagram 2.

****

9. **Design**

9.1 Design - Detailed Class Diagram



9.2 Design – Detailed Design of Methods

Class Name VoiceIt

Method name WriteCallback

Return type size\_t

Input argument(s) char \*contents, size\_t size, size\_t nmemb, void \*userp

Output argument(s) None

Error messages None

Files accessed None

Files changed None

Methods invoked readBuffer

Narrative

Method name read\_callback

Return type size\_t

Input argument(s) void \*ptr, size\_t size, size\_t nmemb, void \*userp

Output argument(s) None

Error messages None

Files accessed None

Files changed None

Methods invoked WriteThis, writeBuffer

Narrative

Method name getUser

Return type string

Input argument(s) string email, string password

Output argument(s) None

Error messages fprintf(stderr, "curl\_easy\_perform() failed: %s\n", curl\_easy\_strerror(res));

Files accessed None

Files changed None

Methods invoked readBuffer, CURL, CURLcode, curl\_easy\_strerror, curl\_easy\_perform, curl\_slist\_free\_all, curl\_easy\_cleanup, curl\_easy\_init, clear

Narrative

This REST API call is used to retrieve an existing user profile within the Voiceprint Developer Portal (VPDP) service. It retrieves an existing user profile record from the VPDP service database and returns true or false.

Method name setUser

Return type string

Input argument(s) string email,string password,string firstName,string lastName

Output argument(s) None

Error messages fprintf(stderr, "curl\_easy\_perform() failed: %s\n", curl\_easy\_strerror(res));

Files accessed None

Files changed None

Methods invoked readBuffer, CURL, CURLcode, curl\_easy\_strerror, curl\_easy\_perform, curl\_slist\_free\_all, curl\_global\_init, curl\_easy\_init, curl\_easy\_setopt, curl\_easy\_cleanup

Narrative

This REST API call is used to update an existing user profile within the Voiceprint Developer Portal (VPDP) service. It updates an existing user profile record within the VPDP service database and returns true or false.

Method name createUser

Return type string

Input argument(s) string email,string password,string firstName,string lastName, string phone1, string phone2, string phone3

Output argument(s) None

Error messages fprintf(stderr, "curl\_easy\_perform() failed: %s\n", curl\_easy\_strerror(res));

Files accessed None

Files changed None

Methods invoked CURL, CURLcode, curl\_easy\_strerror, curl\_easy\_perform, curl\_slist\_free\_all, curl\_global\_init, curl\_easy\_init, curl\_easy\_setopt, curl\_easy\_cleanup, readBuffer

Narrative:

Registers a new user profile within the Voiceprint Developer Portal (VPDP) service. Creates a new user profile record in the VPDP service database and returns true or false. Newly registered user profiles are enabled by default. Your DeveloperID is in the Welcome email you received when you registered.

Method name createUser

Return type string

Input argument(s) string email,string password,string firstName,string lastName

Output argument(s) email, password, firstName, lastName, "","",""

Error messages None

Files accessed None

Files changed None

Methods invoked createUser

Narrative

Registers a new user profile within the Voiceprint Developer Portal (VPDP) service. Creates a new user profile record in the VPDP service database and returns true or false. Newly registered user profiles are enabled by default. Your DeveloperID is in the Welcome email you received when you registered.

Method name deleteUser

Return type string

Input argument(s) string email,string password

Output argument(s) None

Error messages fprintf(stderr, "curl\_easy\_perform() failed: %s\n", curl\_easy\_strerror(res));

Files accessed None

Files changed None

Methods invoked curl\_easy\_strerror, curl\_easy\_perform, curl\_slist\_free\_all, curl\_global\_init, curl\_easy\_init, curl\_easy\_setopt, curl\_easy\_cleanup, readBuffer clear

Narrative

This REST API call is used to delete an existing user profile within the Voiceprint Developer Portal (VPDP) service. Deletes an existing user profile record from the VPDP service database and returns true or false.

Method name createEnrollment

Return type string

Input argument(s) string email,string passwrd, string pathToEnrollmentWav, string contentLanguage

Output argument(s) None

Error messages None

Files accessed fd

Files changed fd

Methods invoked clear, curl\_easy\_init, curl\_slist\_append, curl\_easy\_setopt, curl\_easy\_perform, curl\_slist\_free\_all, c\_str, curl\_easy\_cleanup, sha256, fopen, fstat, fileno

Narrative

This REST API call is used to create a new enrollment template for the specified user profile within the Voiceprint Developer Portal (VPDP) service. It creates a new enrollment template for the specified user profile in the VPDP service database and returns true or false. We recommend a minimum of three (3) enrollment templates per Voiceprint Phrase (VPP). Please Note: The Voiceprint Phrase's (VPP's) are Text-Dependent. The Minimum length of a VPP is 1.2 second. **Please note: You cannot use enrollment sound file for authentication. This is because of our anti- spoofing technology.** To manage the VPPs associated with your DeveloperID, please login to the developer portaland navigate to Voiceprint Phrases section. Please note: You can use DetectedVoiceprintText and DetectedTextConfidence to help decide which enrollmentID to keep or throw out and have the user record again based on text detected and its confidence.

Method name createEnrollment

Return type string

Input argument(s) string email,string passwrd, string pathToEnrollmentWav

Output argument(s) email,passwrd, pathToEnrollmentWav, ""

Error messages None

Files accessed fd

Files changed fd

Methods invoked createEnrollment

Narrative

This REST API call is used to create a new enrollment template for the specified user profile within the Voiceprint Developer Portal (VPDP) service. It creates a new enrollment template for the specified user profile in the VPDP service database and returns true or false. We recommend a minimum of three (3) enrollment templates per Voiceprint Phrase (VPP). Please Note: The Voiceprint Phrase's (VPP's) are Text-Dependent. The Minimum length of a VPP is 1.2 second. **Please note: You cannot use enrollment sound file for authentication. This is because of our anti- spoofing technology.** To manage the VPPs associated with your DeveloperID, please login to the developer portaland navigate to Voiceprint Phrases section. Please note: You can use DetectedVoiceprintText and DetectedTextConfidence to help decide which enrollmentID to keep or throw out and have the user record again based on text detected and its confidence.

Method name createEnrollmentByWavURL

Return type string

Input argument(s) string email,string passwrd, string urlToEnrollmentWav, string contentLanguage

Output argument(s) None

Error messages None

Files accessed None

Files changed None

Methods invoked CURL, CURLcode, clear, curl\_easy\_init, curl\_slist\_append, curl\_easy\_setopt, curl\_easy\_perform, curl\_slist\_free\_all, curl\_easy\_cleanup

Narrative

This REST API call is used to create a new enrollment template for the specified user profile within the Voiceprint Developer Portal (VPDP) service via a pre-recorded sound file. It creates a new enrollment template for the specified user profile in the VPDP service database and returns true or false. We recommend a minimum of three (3) enrollment templates per Voiceprint Phrase (VPP). Please Note: The Voiceprint Phrase's (VPP's) are Text-Dependent. The Minimum length of a VPP is 1.2 second. **Please note: You cannot use enrollment sound file for authentication. This is because of our anti- spoofing technology.** To manage the VPPs associated with your DeveloperID, please login to the developer portal and navigate to Voiceprint Phrases section. Please note: You can use DetectedVoiceprintText and DetectedTextConfidence to help decide which enrollmentID to keep or throw out and have the user record again based on text detected and its confidence.

Method name createEnrollmentByWavURL

Return type string

Input argument(s) string email,string passwrd, string urlToEnrollmentWav

Output argument(s) email,passwrd, urlToEnrollmentWav, ""

Error messages None

Files accessed None

Files changed None

Methods invoked createEnrollmentByWavURL

Narrative

This REST API call is used to create a new enrollment template for the specified user profile within the Voiceprint Developer Portal (VPDP) service via a pre-recorded sound file. It creates a new enrollment template for the specified user profile in the VPDP service database and returns true or false. We recommend a minimum of three (3) enrollment templates per Voiceprint Phrase (VPP). Please Note: The Voiceprint Phrase's (VPP's) are Text-Dependent. The Minimum length of a VPP is 1.2 second. **Please note: You cannot use enrollment sound file for authentication. This is because of our anti- spoofing technology.** To manage the VPPs associated with your DeveloperID, please login to the developer portal and navigate to Voiceprint Phrases section. Please note: You can use DetectedVoiceprintText and DetectedTextConfidence to help decide which enrollmentID to keep or throw out and have the user record again based on text detected and its confidence.

Method name getEnrollments

Return type string

Input argument(s) string email,string password

Output argument(s) None

Error messages fprintf(stderr, "curl\_easy\_perform() failed: %s\n", curl\_easy\_strerror(res));

Files accessed None

Files changed None

Methods invoked clear, curl\_easy\_init, curl\_slist\_append, curl\_easy\_setopt, curl\_easy\_perform, curl\_slist\_free\_all, c\_str

Narrative

This REST API call is used to retrieve the existing enrollment template(s) for the specified user profile within the Voiceprint Developer Portal (VPDP) service. It retrieves the existing enrollment template(s) for the specified user profile from the VPDP service database and returns true or false.

Method name deleteEnrollment

Return type string

Input argument(s) string email,string password,string enrollmentId

Output argument(s) None

Error messages None

Files accessed None

Files changed None

Methods invoked clear, curl\_easy\_init, curl\_slist\_append, curl\_easy\_setopt, curl\_easy\_perform, curl\_slist\_free\_all, c\_str

Narrative

This REST API call is used to delete an existing enrollment template for the specified user profile within the Voiceprint Developer Portal (VPDP) service. It deletes an existing enrollment template for the specified user profile from the VPDP service database and returns true or false.

Method name authentication

Return type string

Input argument(s) string email,string passwrd, string pathToAuthenticationWav, string confidence, string contentLanguage

Output argument(s) None

Error messages None

Files accessed None

Files changed None

Methods invoked clear, curl\_easy\_init, curl\_slist\_append, curl\_easy\_setopt, curl\_easy\_perform, curl\_slist\_free\_all, c\_str, curl\_easy\_cleanup, sha256, fopen, fstat, fileno

Narrative

This REST API call is used to authenticate the specified user profile within the Voiceprint Developer Portal (VPDP) service. It authenticates the specified user profile in the VPDP service database and returns success or failure. Please Note: The Voiceprint Phrase's (VPP's) are Text-Dependent. The Minimum length of a VPP is 1.5 second. **Please note: You cannot use enrollment sound file for authentication. This is because of our anti- spoofing technology.** To manage the VPPs associated with your DeveloperID, please login to the developer portaland navigate to Voiceprint Phrases section. We recommend starting with 85% for the confidence parameter during testing. After becoming familiar with the API, you can tweak the confidence parameter (with a maximum around 91) in order to decrease false positives. Please note: You can use DetectedVoiceprintText and DetectedTextConfidence to help decide which authentication to keep or throw out and have the user record again based on speech text detected and its confidence.

Method name authentication

Return type string

Input argument(s) string email,string passwrd, string pathToAuthenticationWav, string confidence

Output argument(s) email,passwrd,pathToAuthenticationWav,confidence,""

Error messages None

Files accessed None

Files changed None

Methods invoked authentication

Narrative

This REST API call is used to authenticate the specified user profile within the Voiceprint Developer Portal (VPDP) service. It authenticates the specified user profile in the VPDP service database and returns success or failure. Please Note: The Voiceprint Phrase's (VPP's) are Text-Dependent. The Minimum length of a VPP is 1.5 second. **Please note: You cannot use enrollment sound file for authentication. This is because of our anti- spoofing technology.** To manage the VPPs associated with your DeveloperID, please login to the [developer portal](https://siv.voiceprintportal.com/developerlogin.jsp)and navigate to Voiceprint Phrases section. We recommend starting with 85% for the confidence parameter during testing. After becoming familiar with the API, you can tweak the confidence parameter (with a maximum around 91) in order to decrease false positives. Please note: You can use DetectedVoiceprintText and DetectedTextConfidence to help decide which authentication to keep or throw out and have the user record again based on speech text detected and its confidence.

Method name authenticationByWavURL

Return type string

Input argument(s) string email,string passwrd, string urlToAuthenticationWav, string confidence, string contentLanguage

Output argument(s) None

Error messages None

Files accessed None

Files changed None

Methods invoked clear, curl\_easy\_init, curl\_slist\_append, curl\_easy\_setopt, curl\_easy\_perform, curl\_slist\_free\_all, c\_str, curl\_easy\_cleanup, sha256

Narrative

This REST API call is used to authenticate the specified user profile within the Voiceprint Developer Portal (VPDP) service via a pre-recorded sound file. It authenticates the specified user profile in the VPDP service database and returns success or failure. Please Note: The Voiceprint Phrase's (VPP's) are Text-Dependent. The Minimum length of a VPP is 1.2 second. **Please note: You cannot use enrollment sound file for authentication. This is because of our anti- spoofing technology.** To manage the VPPs associated with your DeveloperID, please login to the developer portal and navigate to Voiceprint Phrases section. We recommend starting with 85% for the confidence parameter during testing. After becoming familiar with the API, you can tweak the confidence parameter (with a maximum around 91) in order to decrease false positives. Please note: You can use DetectedVoiceprintText and DetectedTextConfidence to help decide which authentication to keep or throw out and have the user record again based on speech text detected and its confidence.

Method name authenticationByWavURL

Return type string

Input argument(s) string email,string passwrd, string urlToAuthenticationWav, string confidence

Output argument(s) email,passwrd,urlToAuthenticationWav,confidence,""

Error messages None

Files accessed None

Files changed None

Methods invoked authenticationByWavURL

Narrative

This REST API call is used to authenticate the specified user profile within the Voiceprint Developer Portal (VPDP) service via a pre-recorded sound file. It authenticates the specified user profile in the VPDP service database and returns success or failure. Please Note: The Voiceprint Phrase's (VPP's) are Text-Dependent. The Minimum length of a VPP is 1.2 second. **Please note: You cannot use enrollment sound file for authentication. This is because of our anti- spoofing technology.** To manage the VPPs associated with your DeveloperID, please login to the developer portaland navigate to Voiceprint Phrases section. We recommend starting with 85% for the confidence parameter during testing. After becoming familiar with the API, you can tweak the confidence parameter(with a maximum around 91) in order to decrease false positives. Please note: You can use DetectedVoiceprintText and DetectedTextConfidence to help decide which authentication to keep or throw out and have the user record again based on speech text detected and its confidence.

10. **Change Orders**

10.1. Changes. The Customer may at any time request changes to the Specifications.

10.2. Additional Time or Expense. If the proposed change will, in the Developer's reasonable opinion, require a delay in the delivery or installation of the Software or result in additional expense, then the Customer and the Developer shall confer. The Customer may in that case elect to either

(a) withdraw its proposed change, or

(b) require the Developer to deliver the Software with the proposed change, subject to the delay or additional expense or both.

10.3. Termination. If the Developer is unable to accept, or chooses not to accept, the change order, the Customer may terminate the agreement upon notice to the Developer.

11. **Training**

11.1. Scope. The Developer shall provide the Customer with 1 class session (110 minutes) of training OR such training as the Customer may reasonably request on the use of the Software.

11.2. Dates and Locations. The Developer shall conduct the training on the dates that the parties agree upon within Room 207, Student Success Center.

11.3. Costs. The training Developer shall provide the training at $100 per session.

12. **Support and Maintenance**

12.1. Initial Period. The Developer shall provide the Customer no support and maintenance services post April 25th due to class constraints. Until then, the Developer shall provide the Customer maintenance if needed.

13. **Term.** This agreement will begin on April 11, 2017 and continue until the Developer has performed all its obligations under this agreement, unless terminated earlier.

14. **Representations.** The Developer represents and warrants to the Customer as follows, acknowledging that the Customer is relying on these representations and warranties:

14.1. Right to Assign. The Developer is the sole author of the Software and has, and will have, the authority to assign rights to the Software to the Customer under this agreement.

14.2. No Infringement. The Developer's use of the Software will not infringe upon the intellectual property, contractual, or other proprietary or personal rights of any person.

15. **Warranties**

15.1. Limited Media Warranty. The Developer warrants that, for a period of 7 days following delivery of the Software, the media upon which the Software is delivered will be free from defects in materials and workmanship under normal use. The Developer's sole obligation in this regard will be to provide the Customer with a performing copy of the Software within a reasonable time following the Customer's written notice to the Developer of the defect.

15.2. Limited Support Warranty. For a period of 7 days following the delivery OR installation of the Software, the Developer shall perform its maintenance and support services consistent with generally accepted industry standards, but only if the Software is installed and operated in accordance with the Developer's documentation and other instructions.

15.3. Limited Performance Warranty. The Developer warrants that, for a period of 1 month starting on the date of delivery OR installation of the Software, the Software will perform substantially in accordance with the functional specifications set forth in the documentation, but only if the Software is installed and operated in accordance with the Developer's documentation and other instructions.

15.4. Disclaimer. The Developer does not warrant that the functions contained in the Software will meet the Licensee's requirements or operate in the combination desired by the Licensee, or that the Software's operation will be uninterrupted or error free. The Developer does not make and will not be liable for any warranties other than those expressly included in this agreement.

16. **Acknowledgments**

16.1. Independent Contractor. The Developer is an independent contractor. Nothing contained in this agreement creates a partnership, joint venture, employer/employee, principal-and-agent, or any similar relationship between the parties.

16.2. Proprietary Rights. The parties acknowledge that the development of the Software is "work for hire" within the meaning of the Copyright Act of 1976, as amended on one or more occasions, and that the Software will be the Customer's sole property.

16.3. Consent to Use of Data. The Developer may collect and use technical information gathered as part of its support services, but may only use this information to improve its products and services. The Developer shall not disclose this any of this information in a form that personally identifies the Customer or it clients.

16.4. Government End Users. If the Software and related documentation are supplied to or purchased by or on behalf of the United States Government, then the Software is deemed to be "commercial software" as that term is used in the Federal Acquisition Regulation system. The rights of the United States will not exceed the minimum rights set forth in FAR 52.227-19 for "restricted computer software". All other terms and conditions of this agreement otherwise apply.

17. **Confidentiality**

17.1. Confidentiality Obligations. During the term of this agreement and for 6 months afterward, the Developer shall hold all Confidential Information in confidence in accordance with the terms of this agreement.

17.2. Use Solely for Purpose. The Developer shall use the Confidential Information in accordance with, and solely for the purpose of providing its services under, the terms of this agreement.

18. **Assignment of Rights.** Upon acceptance of the Software and payment of all compensation due to the Developer, the Developer agrees to grant and assign, and hereby grants and assigns, to the Customer its entire interest in the Software (including all intellectual and other property rights).

19**. Indemnification**

19.1. Developer's Indemnity. The Developer shall indemnify the Customer and its officers, directors, employees, agents, and affiliates, against all claims, liability, costs, and expenses (including attorneys' fees) arising from any third party claim or proceeding against the Customer

(a) based on any claim that the Software [or any services performed under this agreement] infringes or violates any intellectual or other property right, or

(b) that alleges any negligent act or omission or willful conduct of the Developer

19.2. Notice of Claim. The Customer shall give prompt written notice to the Developer of any claim or potential claim for indemnification under this agreement.

19.3. Exclusive Remedies. The rights granted under this section 16 (Indemnification) are the exclusive remedies available under this agreement in connection with the claims and losses that this section addresses.

20. **Limitation of Liability.** Neither party will be liable for breach-of-contract damages that the breaching party could not reasonably have foreseen on entering into this agreement.

21. **Termination**

21.1. Termination upon Notice. Either party may terminate this agreement for any reason upon 14 days' Notice to the other party.

21.2. Termination for Cause. If either party

(a) commits a material breach or material default in the performance or observance of any of its obligations under this agreement, and

(b) the breach or default continues for a period of 7 days after delivery by the other party of written notice reasonably detailing such breach or default, then

the non-breaching or non-defaulting party may terminate this agreement, with immediate effect, upon written notice to the breaching or defaulting party.

21.3. Termination upon Insolvency. This agreement will terminate immediately upon the Developer's insolvency, bankruptcy, receivership, dissolution, or liquidation.

21.4. Effect of Termination

(a) Termination for Customer's Breach. In the event of termination of this agreement due to a material breach or default committed by the Customer,

(i) the assignment of rights to the Customer in this agreement will terminate, and

(ii) the Customer shall immediately stop using the Software and destroy or erase all copies in its possession or control.

(b) Termination for any other Reason. In the event of termination of this agreement for any other reason,

(i) the Customer will not be required to make payment for any Deliverables that the Customer had not accepted,

(ii) the Customer will continue to exercise all rights to the Software that it has acquired under this agreement,

(iii) the Developer shall immediately deliver to the Customer all Software, documentation, source code, and other Customer property in its possession relating to the Software and then destroy all copies in its possession or control, and

(iv) the Customer shall pay the Developer for all services rendered and work performed up to the effective date of termination, unless the Customer has terminated for cause, in which case it will only be required to pay fair value. The Developer shall provide the Customer with an invoice for its fees within 30 days of the effective date of the termination, and the Client shall pay the invoice within 14 days of receipt.

22. **General**

22.1. Entire Agreement. This agreement contains all the terms agreed to by the parties relating to its subject matter. It replaces all previous discussions, understandings, and agreements.

22.2. Amendment. This agreement may only be amended by a written document signed by both parties.

22.3. Assignment. The Developer may not assign this agreement or any of its rights under it. The Customer may assign this agreement or any of its rights without notice or the need for the Developer's consent.

22.4. Remedies Cumulative. Except as provided in section 16.3 (Exclusive Remedies), the rights and remedies available to a party under this agreement are cumulative and in addition to, not exclusive of or in substitution for, any rights or remedies otherwise available to that party.

22.5. Survival. Sections 15 (Confidentiality), 16 (Assignment of Rights), 17 (Indemnification), 18 (Limitation of Liability), and 19.4 (Effect of Termination) survive the termination or expiration of this agreement.

22.6. Severability. If any part of this agreement is declared unenforceable or invalid, the remainder will continue to be valid and enforceable.

22.7. Waiver. A party's failure or neglect to enforce any of rights under this agreement will not be deemed to be a waiver of that party's rights.

22.8. Governing Law. This agreement will be governed by and construed in accordance with the laws of the State of Ohio, without regard to its conflict of laws rules.

22.9. Dispute Resolution

(a) Arbitration. Any dispute or controversy arising under or in connection with this agreement will be settled exclusively by arbitration in Ohio, in accordance with the rules of the American Arbitration Association then in effect by 5 arbitrator(s).

(b) No Punitive Damages. The arbitrator(s) will not have the power to award punitive damages.

(c) Judgment. The successful party may enter the arbitral judgment in any court having jurisdiction.

22.10. Waiver of Jury Trial. The parties waive their respective rights to trial by jury in any action or proceeding involving this agreement or the transactions relating to its subject matter.

22.11. Headings. The headings used in this agreement and its division into sections and other subdivisions do not affect its interpretation.

22.12. Counterparts. This agreement may be signed in any number of counterparts, each of which is an original and all of which taken together form one single document.

22.13. Effectiveness of Agreement. This agreement is effective as of the date shown at the top of the first page, even if any signatures are made after that date.

This agreement has been signed by the parties.

Dr. Nikolaos Bourbakis

By: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Name:

Title:

Date

Team 7

By: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Name:

Title:

Date