

HiLingual Server Test Cases

Team 2

Garrett Davidson, Noah Maxey, Nate Ohlson, Riley Shaw

Validating session token

1. Title: Sending a request with a valid session token
Procedure: A user sends a POST request to the server containing a valid alphanumeric session token.
Result: The server responds that the session token is valid.
Severity: Critical
2. Title: Sending a request with a different user's session token
Procedure: A user sends a POST request to the server containing an active alphanumeric session token for a different user.
Result: The server responds that the session token is valid.
Severity: Critical
3. Title: Sending a request with an invalid session token
Procedure: A user sends a POST request to the server containing an invalid alphanumeric session token.
Result: The server responds that the session token is invalid.
Severity: Critical
4. Title: Sending a request with an improperly formatted session token
Procedure: A user sends a POST request to the server with a session token containing non-alphanumeric characters.
Result: The server responds that the session token is invalid.
Severity: Important
5. Title: Sending a request with an improperly non-ASCII session token
Procedure: A user sends a POST request to the server with a session token containing non-ASCII characters.
Result: The server responds that the session token is invalid.
Severity: Important
6. Title: Sending a request with an empty session token.
Procedure: A user sends a POST request to the server containing an empty session token.
Result: The server responds that the session token is invalid.
Severity: Workaround

7. Title: Sending a request with no session token
Procedure: A user sends a POST request to the server containing no session token.
Result: The server responds that the session token is invalid.
Severity: Workaround

Sending a request to a user id

8. Title: Sending a request with a valid user id
Procedure: A user sends a POST request to the server containing their valid alphanumeric session token and a valid recipient user id.
Result: The server responds that the user id is valid.
Severity: Important
9. Title: Sending a request with a user id for a nonexistent user
Procedure: A user sends a POST request to the server containing their valid alphanumeric session token and a recipient user id for a nonexistent user.
Result: The server responds that the user id is invalid.
Severity: Workaround
10. Title: Sending a request with an invalid id
Procedure: A user sends a POST request to the server containing their valid alphanumeric session token and a non-numeric string for the user id.
Result: The server responds that the user id is invalid.
Severity: Workaround
11. Title: Sending a request with an empty recipient
Procedure: A user sends a POST request to the server containing their valid alphanumeric session token and an empty recipient.
Result: The server responds that the user id is invalid.
Severity: Workaround
12. Title: Sending a request with no recipient specified
Procedure: A user sends a POST request to the server containing their valid alphanumeric session token and no recipient specified.
Result: The server responds that the user id is invalid.
Severity: Workaround

Picture messages

13. Title: Sending a picture message
Procedure: A user sends a POST request to the server containing their valid alphanumeric session token, a valid recipient user id, and valid PNG image data.
Result: The server returns 200.
Severity: Critical

14. Title: Sending a picture message with extra fields
Procedure: A user sends a POST request to the server containing their valid alphanumeric session token, a valid recipient user id, valid PNG image data, and additional fields with arbitrary data.
Result: The server ignores the additional fields and returns 200.
Severity: Workaround
15. Title: Sending a picture message to yourself
Procedure: A user sends a POST request to the server containing their valid alphanumeric session token, their own user id, and valid PNG image data.
Result: The server responds that they cannot send a message to themselves.
Severity: Workaround
16. Title: Sending a picture message with invalid image data
Procedure: A user sends a POST request to the server containing their valid alphanumeric session token, a valid recipient user id, and non-PNG data.
Result: The server responds that the image is invalid.
Severity: Important
17. Title: Sending a picture message with empty image data
Procedure: A user sends a POST request to the server containing their valid alphanumeric session token, a valid recipient user id, and an empty value for the image data.
Result: The server responds that the image is invalid.
Severity: Workaround
18. Title: Sending a picture message with no image data
Procedure: A user sends a POST request to the server containing only their valid alphanumeric session token and a valid recipient user id.
Result: The server responds that the image is invalid.
Severity: Workaround
19. Title: Retrieving your messages from nonexistent user
Procedure: A user sends a GET request to the server containing a valid session token and a nonexistent user id.
Result: The server responds that the user is invalid.
Severity: Workaround
20. Title: Retrieving your messages from a valid user that you have corresponded with
Procedure: A user sends a GET request to the server containing a valid session token and a valid user id for a user that they have corresponded with.
Result: The server responds with an array of messages with that user.
Severity: Important

21. Title: Retrieving your messages from a valid user that you have not corresponded with
Procedure: A user sends a GET request to the server containing a valid session token and a valid user id for a user that they have not corresponded with.
Result: The server responds with an empty array.
Severity: Workaround

Flashcards

22. Title: Sending flashcard to server
Procedure: A user sends a POST request to the server containing a valid session token and a flashcard.
Result: The server responds that it has saved the flashcard.
Severity: Important
23. Title: Receiving flashcards from server
Procedure: A user sends a GET request to the server containing a valid session token.
Result: The server responds with an array of flashcards.
Severity: Important
24. Title: Sending empty flashcard front
Procedure: A user sends a GET request to the server containing a valid session token and an empty value for the flashcard front and non-empty value for the flashcard back
Result: The server responds that the flashcard is invalid.
Severity: Workaround
25. Title: Sending empty flashcard back
Procedure: A user sends a GET request to the server containing a valid session token and an empty value for the flashcard back and non-empty value for the flashcard front
Result: The server responds that the flashcard is invalid.
Severity: Workaround
26. Title: Sending invalid flashcard front
Procedure: A user sends a GET request to the server containing a valid session token and a nonexistent flashcard front and non-empty existent flashcard back
Result: The server responds that the flashcard is invalid.
Severity: Workaround
27. Title: Sending invalid flashcard back
Procedure: A user sends a GET request to the server containing a valid session token and a nonexistent flashcard back and non-empty existent flashcard front
Result: The server responds that the flashcard is invalid.
Severity: Workaround

Profile

28. Title: Receiving profile from the server

Procedure: A user sends a GET request to the server containing a valid session token.

Result: The server responds with an object containing all the fields that make up the user's profile.

Severity: Important

29. Title: Sending an edited profile to the server

Procedure: A user sends a POST request to the server containing a valid session token and valid object with the user's profile information.

Result: The server responds that the profile has been updated

Severity: Workaround

30. Title: Sending an invalid edited profile to the server

Procedure: A user sends a POST request to the server containing a valid session token and invalid object with the user's profile information.

Result: The server responds that the profile edit is invalid.

Severity: Important

31. Title: Sending a profile to the server with an invalid age

Procedure: A user sends a POST request to the server containing a valid session token and object with the user's profile information and an age with non-numeric value.

Result: The server responds that the profile age is invalid.

Severity: Workaround

32. Title: Sending a profile to the server with an invalid age

Procedure: A user sends a POST request to the server containing a valid session token and object with the user's profile information and an age less than 0.

Result: The server responds that the profile age is invalid

Severity: Workaround

33. Title: Sending a profile to the server with an empty age

Procedure: A user sends a POST request to the server containing a valid session token and object with the user's profile information and an empty age.

Result: The server responds that the profile age is empty.

Severity: Workaround

34. Title: Sending a profile to the server with no age

Procedure: A user sends a POST request to the server containing a valid session token and object with the user's profile information and no age.

Result: The server responds that the profile has no age.

Severity: Workaround

35. Title: Sending a profile to the server with an empty name
Procedure: A user sends a POST request to the server containing a valid session token and object with the user's profile information and an empty name.
Result: The server responds that the profile name is empty.
Severity: Workaround
36. Title: Sending a profile to the server with no name
Procedure: A user sends a POST request to the server containing a valid session token and object with the user's profile information and no name.
Result: The server responds that the profile has no name.
Severity: Workaround
37. Title: Sending a profile to the server with an invalid language
Procedure: A user sends a POST request to the server containing a valid session token and object with the user's profile information and language that doesn't match one of our supported languages.
Result: The server responds that the profile language is invalid.
Severity: Workaround
38. Title: Sending a profile to the server with an empty language
Procedure: A user sends a POST request to the server containing a valid session token and object with the user's profile information and an empty language.
Result: The server responds that the profile language is empty.
Severity: Workaround
39. Title: Sending a profile to the server with no language
Procedure: A user sends a POST request to the server containing a valid session token and object with the user's profile information and no language.
Result: The server responds that the profile has no language.
Severity: Workaround
40. Title: Sending an edited profile to the server
Procedure: A user sends a POST request to the server containing a valid session token and valid object with the user's profile information.
Result: The server responds that the profile has been updated.
Severity: Workaround
41. Title: Sending invalid profile edit
Procedure: A user sends a POST request to the server with a valid session token and an invalid object with the user's profile information.
Result: The server responds that it was a bad request.
Severity: Workaround

42. Title: Sending a new profile to the server

Procedure: A user sends a POST request to the server containing a new auth token, user id, and valid object with the user's profile information.

Result: The server responds that the profile has been created.

Severity: Important

43. Title: Sending an invalid new profile to the server

Procedure: A user sends a POST request to the server containing a new auth token, user id, and invalid object with the user's profile information.

Result: The server responds that the profile was invalid.

Severity: Important

Push Notifications

44. Title: Receiving a push notification

Procedure: User A sends a POST request to the server containing a message, the id of User B as the recipient, and a valid session token.

Result: User B should receive a notification.

Severity: Workaround

45. Title: Receiving a push notification

Procedure: User A sends a POST request to the server containing a message, an invalid recipient user id, and a valid session token.

Result: The server responds that the recipient is invalid.

Severity: Workaround

46. Title: Receiving a push notification

Procedure: User A sends a POST request to the server containing a message, the id of User B as the recipient, and an invalid session token.

Result: The server responds that the session token is invalid.

Severity: Workaround

Audio Message

47. Title: Sending an audio message

Procedure: A user sends a POST request to the server containing their valid alphanumeric session token, a valid recipient user id, and valid audio data.

Result: The server returns 200.

Severity: Important

48. Title: Sending an audio message with extra fields

Procedure: A user sends a POST request to the server containing their valid alphanumeric session token, a valid recipient user id, valid audio data, and additional

fields with arbitrary data.

Result: The server ignores the additional fields and returns 200.

Severity: Workaround

49. Title: Sending an audio message to yourself

Procedure: A user sends a POST request to the server containing their valid alphanumeric session token, their own user id, and valid audio data.

Result: The server responds that they cannot send a message to themselves.

Severity: Workaround

50. Title: Sending an audio message with invalid audio data

Procedure: A user sends a POST request to the server containing their valid alphanumeric session token, a valid recipient user id, and invalid m4a data.

Result: The server responds that the audio is invalid.

Severity: Workaround

51. Title: Sending an audio message with empty audio data

Procedure: A user sends a POST request to the server containing their valid alphanumeric session token, a valid recipient user id, and an empty value for the audio data.

Result: The server responds that the audio is invalid.

Severity: Workaround

52. Title: Sending an audio message with no audio data

Procedure: A user sends a POST request to the server containing only their valid alphanumeric session token and a valid recipient user id.

Result: The server responds that the audio is invalid.

Severity: Workaround

Text Message

53. Title: Sending a text message

Procedure: A user sends a POST request to the server containing their valid alphanumeric session token, a valid recipient user id, and a non-empty message string.

Result: The server returns 200.

Severity: Critical

54. Title: Sending a text message with an invalid session token

Procedure: A user sends a POST request to the server containing an invalid session token, a valid recipient user id, and a non-empty string.

Result: The server responds that the session token is invalid.

Severity: Important

55. Title: Sending a text message with an invalid recipient
Procedure: A user sends a POST request to the server containing their valid alphanumeric session token, an invalid recipient user id, and a non-empty string.
Result: The server responds that the recipient is invalid.
Severity: Important
56. Title: Sending a text message with an empty message
Procedure: A user sends a POST request to the server containing their valid alphanumeric session token, an invalid recipient user id, and an empty string.
Result: The server responds that the message must not be empty.
Severity: Important
57. Title: Sending a text message with no message
Procedure: A user sends a POST request to the server containing only their valid alphanumeric session token and a valid recipient user id.
Result: The server returns 200.
Severity: Workaround

Translations

58. Title: Sending a translation request
Procedure: A user sends a POST request to the server containing their valid session token, the requested phrase to be translated, and the language the requested phrase should be translated to.
Result: The server returns the translated string.
Severity: Critical
59. Title: Sending a translation request with a requested phrase translation of size greater than 500 characters
Procedure: A user sends a POST request to the server containing a valid session token, a string longer than 500 characters to be translated, and the language the requested phrase should be translated to.
Result: The server responds that the requested phrase is too long.
Severity: Workaround
60. Title: Sending a translation request with an empty requested translation phrase
Procedure: A user sends a POST request to the server containing a valid session token, an empty requested translation phrase, and the language the requested phrase should be translated to.
Result: The server responds that the requested phrase is empty.
Severity: Workaround
61. Title: Sending a translation request with a requested phrase translation that is in multiple languages

Procedure: A user sends a POST request to the server containing a valid session token, a requested phrase translation that is in multiple languages, and the language the requested phrase should be translated to.

Result: The server responds that the requested phrase is not valid.

Severity: Workaround

62. Title: Sending a translation request with a requested phrase translation with multiple alphabets.

Procedure: A user sends a POST request to the server containing a valid session token, a requested phrase to be translated written with multiple alphabets, and the language the requested phrase should be translated to.

Result: The server responds that the requested phrase is not valid.

Severity: Workaround

63. Title: Sending a translation request with a target language that does not match our supported languages.

Procedure: A user sends a POST request to the server containing a valid session token, a requested phrase to be translated, and a target language that does not match one of our supported languages.

Result: The server responds that the requested target language is not supported.

Severity: Workaround

Matching

64. Title: Sending a matching request and receiving non-matching language learners

Procedure: Fill the database with users that match the requester's language selection. A user sends a GET request to receive their match list.

Result: The server returns an array of profiles, containing at least one user who does not have matching selection as the requesting user.

Severity: Important

65. Title: Sending a matching request and receiving matches

Procedure: Fill the database with users that match the requester's language selection. A user sends a GET request to receive their match list.

Result: The server returns a non-empty array of profiles, each of which matches the user's language selection.

Severity: Important

66. Title: Sending a matching request and receiving more than 10 users

Procedure: Fill the database with more than 10 users that match the requester's language selection. A user sends a GET request to receive their match list.

Result: The server returns an array of no more than 10 matching profiles

Severity: Workaround

Authorization

67. Title: Sending an authorization token to sign in via Facebook/Google
Procedure: A user sends a POST request to the server containing a valid Facebook/Google id, and the corresponding Facebook/Google authorization token.
Result: The server returns the requestor's HiLingual session token.
Severity: Important
68. Title: Sending an authorization to sign in via Facebook/Google with invalid id
Procedure: A user sends a POST request to the server containing an invalid Facebook/Google id, and a valid Facebook/Google authorization token.
Result: The server responds that the authorization failed.
Severity: Important
69. Title: Sending an authorization to sign in via Facebook/Google with invalid token
Procedure: A user sends a POST request to the server containing a valid Facebook/Google id, and an invalid Facebook/Google authorization token.
Result: The server responds that the authorization failed.
Severity: Important