

Gigseek Test Plan and Results

Aaron Verst and Ryan Gengler

Test Plan Introduction

To ensure a seamless user experience is achieved, testing will need to be done across multiple different components of our application. First, the applications architecture will need to be tested using simulated data. These tests will span over a variety of normal and abnormal use cases. Through conducting unit tests, we will ensure that our data is being properly transferred to our front end environment where it will be filtered and manipulated. Next, our front end functionalities will need to be tested to verify that the user is able to successfully utilize the features that we have provided. These features include searching, filtering, and accessing profiles. These tests will ensure that the end users are provided with a working and scalable environment that provides all of the features we have outlined.

Test Case Descriptions

CDT1.1 Central Database Test 1

CDT1.2 This test will ensure that our front end architecture has access to the flat file database where our account information is being stored

CDT1.3 This test will query the data from our database using the same operations that our front end architecture uses to query data. The results from this query will

provide us with the data that our frontend architecture has access to and will ensure that all permissions are set correctly.

CDT1.4 Inputs: The inputs for this test will be the query used by the frontend architecture and the flat file database that stores profile/account information.

CDT1.5 Outputs: All of the database records will return with the correct permissions and unauthorized data will not be returned.

CDT1.6 Both

CDT1.7 Whitebox

CDT1.8 Functional

CDT1.9 Unit Test

CDT2.1 Central Database Test 2

CDT2.2 This test will ensure that the front end architecture of a “Venue” profile has access only to “Performer” data when performing searches

CDT2.3 This test will query the data from our database using the same operations that our front end architecture uses to query data. The results from this query will provide us with the data that the front end architecture of a “Venue” has access to and will ensure that all permissions are set correctly.

CDT2.4 Inputs: The inputs for this test will be the query used by the frontend architecture of a “Venue” profile and the flat file database that stores profile/account information.

CDT2.5 Outputs: All of the database records will return with the correct permissions and unauthorized data will not be returned.

CDT2.6 Normal

CDT2.7 Whitebox

CDT2.8 Functional

CDT2.9 Unit Test

CDT3.1 Central Database Test 3

CDT3.2 This test will ensure that the front end architecture of a “Performer” profile has access only to “Venue” data when performing searches

CDT3.3 This test will query the data from our database using the same operations that our front end architecture uses to query data. The results from this query will provide us with the data that the front end architecture of a “Performer” has access to and will ensure that all permissions are set correctly.

CDT3.4 Inputs: The inputs for this test will be the query used by the frontend architecture of a “Performer” profile and the flat file database that stores profile/account information.

CDT3.5 Outputs: All of the database records will return with the correct permissions and unauthorized data will not be returned.

CDT3.6 Normal

CDT3.7 Whitebox

CDT3.8 Functional

CDT3.9 Unit Test

PWA1.1 Performer Web Admin Test 1

PWA1.2 This test will test the ability for a performer to filter the genre of music that venues would like to host.

PWA1.3 We will test this by navigating to the “Venue Menu” displaying all of the specific categories that can be filtered when the performer searches for specific venue. One of the filters will be defined as “genre”. The performer will have the option to choose from different genres such as “rock”, “EDM”, “Comedy”, etc. This “genre” will correspond to the type of performance that the venue is seeking. The filtered venues will then show up on the screen.

PWA1.4 Inputs: Clicking on the genre(s) that are desired by the performer.

PWA1.5 Outputs: A list of the venues that fit the filtered criteria.

PWA1.6 Normal

PWA1.7 Blackbox

PWA1.8 Functional

PWA1.9 Unit Test

PWA2.1 Performer Web Admin Test 2

PWA2.2 This test will test the ability of a performer to filter the location in which the venue is located.

PWA2.3 We will test this by navigating to the “Venue Menu” displaying all of the specific categories that can be filtered when the performer searches for specific venues. One of the filters will be defined as “origin”. The performer will have the option to choose from different locations such as “Cincinnati, OH”, “Atlanta, GA”, “Nashville, TN”, etc. The performer will have the option to choose more than one location to include all of the venues that they desire. The filtered venues will then show up on the screen.

PWA2.4 Inputs: Clicking on the location(s) that are desired by the performer.

PWA2.5 Outputs: A list of the venues that fit the filtered criteria.

- PWA2.6** Normal
- PWA2.7** Blackbox
- PWA2.8** Functional
- PWA2.9** Unit Test

PWA3.1 Performer Web Admin Test 3

PWA3.2 This test will test the ability of a performer to filter the capacity of the venue that they are searching for.

PWA3.3 We will test this by navigating to the “Venue Menu” displaying all of the specific categories that can be filtered when the performer searches for specific venues. One of the filters will be defined as “capacity”. The performer will have the option to choose from different capacity limits that include values such as 100, 500, 1000, etc. The performer will have the option to choose a capacity limit that corresponds with their target venue. The filtered venues will then show up on the screen.

PWA3.4 Inputs: Clicking on the capacity that is desired by the performer.

PWA3.5 Outputs: A list of the venues that fit the filtered criteria.

- PWA3.6** Normal
- PWA3.7** Blackbox
- PWA3.8** Functional
- PWA3.9** Unit Test

PWA4.1 Performer Web Admin Test 4

PWA4.2 This test will test the ability of a performer to filter the ammount of events that a venue has held.

PWA4.3 We will test this by navigating to the “Venue Menu” displaying all of the specific categories that can be filtered when the performer searches for specific venues. One of the filters will be defined as “events held”. The performer will have the option to choose from different gig limits that include values such as 25+, 100+, 1000+, etc. The performer will have the option to choose a gig limit that corresponds to the amount of performances a venue has held previously. The filtered venues will then show up on the screen.

PWA4.4 Inputs: Clicking on the gig limit that is desired by the performer.

PWA4.5 Outputs: A list of the venues that fit the filtered criteria.

PWA4.6 Normal

PWA4.7 Blackbox

PWA4.8 Functional

PWA4.9 Unit Test

PWA5.1 Performer Web Admin Test 5

PWA5.2 This test will test the ability for a performer to connect with a venue once the search filters have been applied and a specific venue has been selected by a performer.

PWA5.3 Once the performer has chosen a venue they would like to connect with, the venue will be notified that a venue wants to connect with them. Both the performer and the venue will have access to the other’s profile to do some research and give each side the option to ensure that the performer wants to perform at that venue and the venue wants the performer to perform at their venue. The venue will have a choice to

accept or deny this connection request. If this connection request is denied, then the test ends there, but if the connection request is accepted, the performer has to then accept or deny that acceptance, for security purposes. Once the performer has accepted the acceptance from the venue, the contact information from both the venue and performer, whether that is through a phone number, Instagram, etc., will be shared to each other to allow communication between the performer and desired venue.

PWA5.4 Inputs: Clicking “Accept” or “Deny” on connection request along with the acceptance request.

PWA5.5 Outputs: Contact information is shared between the performer and desired venue, or communication between the two is denied.

PWA5.6 Normal

PWA5.7 Blackbox

PWA5.8 Functional

PWA5.9 Unit Test

VWA1.1 Venue Web Admin Test 1

VWA1.2 This test will test the ability for a venue to filter the genre of performance that they seek from the performer.

VWA1.3 We will test this by navigating to the “Performer Menu” displaying all of the specific categories that can be filtered when the venue searches for specific performers. One of the filters will be defined as “genre”. The venue will have the option to choose from different genres such as “rock”, “EDM”, “Alternative”, “Comedy”, etc. The venue will

have the option to choose more than one option to include all of the performers that the Venue desires. The filtered performers will then show up on the screen.

VWA1.4 Inputs: Clicking on the genres that are desired by the venue.

VWA1.5 Outputs: A list of the performers that fit the filtered criteria.

VWA1.6 Normal

VWA1.7 Blackbox

VWA1.8 Functional

VWA1.9 Unit Test

VWA2.1 Venue Web Admin Test 2

VWA2.2 This test will test the ability for a venue to filter the origin in which the performers are from.

VWA2.3 We will test this by navigating to the “Performer Menu” displaying all of the specific categories that can be filtered when the venue searches for specific performers. One of the filters will be defined as “origin”. The venue will have the option to choose from different locations such as “Cincinnati, OH”, “Atlanta, GA”, “Nashville, TN, etc. The venue will have the option to choose more than one option to include all of the performers that the Venue desires. The filtered performers will then show up on the screen.

VWA2.4 Inputs: Clicking on the locations that are desired by the venue.

VWA2.5 Outputs: A list of the performers that fit the filtered criteria.

VWA2.6 Normal

VWA2.7 Blackbox

VWA2.8 Functional

VWA2.9 Unit Test

VWA3.1 Venue Web Admin Test 3

VWA3.2 This test will test the ability for a venue to filter the maximum amount of time in minutes in which the performers will play their music.

VWA3.3 We will test this by navigating to the “Performer Menu” displaying all of the specific categories that can be filtered when the venue searches for specific performers. One of the filters will be defined as “Max Play Time”. The venue will have the option to choose from different amounts of time such as “45 minutes”, “90 minutes”, “120 minutes”, etc. The venue will have the option to choose more than one option to include all of the performers that the Venue desires. The filtered performers will then show up on the screen.

VWA3.4 Inputs: Clicking on the amounts of time that are desired by the venue.

VWA3.5 Outputs: A list of the performers that fit the filtered criteria.

VWA3.6 Normal

VWA3.7 Blackbox

VWA3.8 Functional

VWA3.9 Unit Test

VWA4.1 Venue Web Admin Test 4

VWA4.2 This test will test the ability for a venue to filter the maximum travel radius that performers would generally travel.

VWA4.3 We will test this by navigating to the “Performer Menu” displaying all of the specific categories that can be filtered when the venue searches for specific performers. One of the filters will be defined as “Max Travel Radius”. The venue will have the option to choose from different genres such as “100 miles”, “200 miles”, “500 miles”, etc. The venue will have the option to choose more than one option to include all of the performers that the Venue desires. The filtered performers will then show up on the screen.

VWA4.4 Inputs: Clicking on the travel radii that are desired by the venue.

VWA4.5 Outputs: A list of the performers that fit the filtered criteria.

VWA4.6 Normal

VWA4.7 Blackbox

VWA4.8 Functional

VWA4.9 Unit Test

VWA5.1 Venue Web Admin Test 5

VWA5.2 This test will test the ability for a venue to connect with a performer once the search filters have been applied and a specific performer has been selected by a venue.

VWA5.3 Once the venue has chosen a performer they would like to connect with, the performer will be notified that a venue wants to connect with them. Both the performer and the venue will have access to the other’s profile to do some research and give each side the option to ensure that the performer wants to perform at that venue and the venue wants the performer to perform at their venue. The performer will have a

choice to accept or deny this connection request. If this connection request is denied, then the test ends there, but if the connection request is accepted, the venue has to then accept or deny that acceptance, for security purposes. Once the venue has accepted the acceptance from the performer, the contact information from both the venue and performer, whether that is through a phone number, Instagram, etc., will be shared to each other to allow communication between the venue and desired performer.

VWA5.4 Inputs: Clicking “Accept” or “Deny” on connection request along with the acceptance request.

VWA5.5 Outputs: Contact information is shared between the venue and desired performer, or communication between the two is denied.

VWA5.6 Normal

VWA5.7 Blackbox

VWA5.8 Functional

VWA5.9 Unit Test

GWA1.1 General Web Admin Test 1

GWA1.2 This test will test the ability for a user to authenticate their credentials with the LDAP servers, whether the user is logging in as a performer or venue.

GWA1.3 We will test this by opening up the application, with the page asking the user if one is a venue or performer. Once the selection by the user is made, a login screen will appear for the user to enter their credentials. In this case, the user does enter valid credentials and then we ensure that the user has successfully logged in. Once logged in, the user should be able to browse the various pages freely.

GWA1.4 Inputs: Valid credentials that log into the system.

GWA1.5 Outputs: A successful login and the ability to browse pages freely. All pages properly redirect to the login page and the login successfully authenticates to the LDAP server.

GWA1.6 Normal

GWA1.7 Blackbox

GWA1.8 Functional

GWA1.9 Unit Test

GWA2.1 General Web Admin Test 2

GWA2.2 This test will test the ability for a user to authenticate their credentials with the LDAP servers, whether the user is logging in as a performer or venue.

GWA2.3 We will test this by opening up the application, with the page asking the user if one is a venue or performer. Once the selection by the user is made, a login screen will appear for the user to enter their credentials. In this case, the user enters invalid credentials and then we ensure that the user does not successfully log in. The user should not be able to browse the various pages freely.

GWA2.4 Inputs: Invalid credentials that keep the user from logging into the system.

GWA2.5 Outputs: An unsuccessful login, not giving the user the ability to browse pages freely. An error message is received by the user and they do not have access to any pages within the application.

GWA2.6 Normal

GWA2.7 Blackbox

GWA2.8 Functional

GWA2.9 Unit Test

FS1.1 **Full System Test 1**

FS1.2 This test will be used to see that the full system works together well from the venue's perspective.

FS1.3 The user will log in as a venue, and once their login is successful, the venue will have access to the various web pages on this application. The venue will then go to the "Performer Menu" and provide search filters to find a performer that is ideal for that specific venue. Once the performer that the venue is looking for is found, they will make a connection request to the desired performer. The performer will accept that connection request and then the venue will accept the acceptance request. Contact information between the two are shared in which they can now communicate more specific matters off of the application.

FS1.4 Inputs: Correct login information, clicking on specific search filters, and clicking "Accept" on the connection and acceptance requests.

FS1.5 Outputs: Venue has access to various web pages on the application, specific desired performers are shown for the venue to choose from, and contact information between the venue and desired performer are shared.

FS1.6 Normal

FS1.7 Blackbox

FS1.8 Functional

FS1.9 Integration

FS2.1 Full System Test 2

FS2.2 This test will be used to see that the full system works together well from the performer's perspective.

FS2.3 The user will log in as a performer, and once their login is successful, the venue will have access to the various web pages on this application. The performer will then go to the "Venue Menu" and provide search filters to find a venue that is ideal for that specific performer. Once the venue that the performer is looking for is found, they will make a connection request to the desired venue. The venue will accept that connection request and then the performer will accept the acceptance request. Contact information between the two are shared in which they can now communicate more specific matters off of the application.

FS2.4 Inputs: Correct login information, clicking on specific search filters, and clicking "Accept" on the connection and acceptance requests.

FS2.5 Outputs: Performer has access to various web pages on the application, specific desired venues are shown for the performer to choose from, and contact information between the performer and desired venue are shared.

FS2.6 Normal

FS2.7 Blackbox

FS2.8 Functional

FS2.9 Integration

Test Case Matrix

	Normal/ Abnormal	Blackbox/ Whitebox	Functional/ Performance	Unit/ Integration
CDT1	Both	Whitebox	Functional	Unit
CDT2	Normal	Whitebox	Functional	Unit
CDT3	Normal	Whitebox	Functional	Unit
PWA1	Normal	Blackbox	Functional	Unit
PWA2	Normal	Blackbox	Functional	Unit
PWA3	Normal	Blackbox	Functional	Unit
PWA4	Normal	Blackbox	Functional	Unit
PWA5	Normal	Blackbox	Functional	Unit
VWA1	Normal	Blackbox	Functional	Unit
VWA2	Normal	Blackbox	Functional	Unit
VWA3	Normal	Blackbox	Functional	Unit
VWA4	Normal	Blackbox	Functional	Unit
VWA5	Normal	Blackbox	Functional	Unit
GWA1	Normal	Blackbox	Functional	Unit
GWA2	Normal	Blackbox	Functional	Unit
FS1	Normal	Blackbox	Functional	Integration
FS2	Normal	Blackbox	Functional	Integration