Testing Document

Sowmya Movva: Scrum Master

Rimpy Saha: Product Owner

Tyvon Factor-Gaymon: Product Owner

George Gramatikov: Tester

Peter Fung: Tester

April 28, 2023

Table of Contents

Section 1 – Introduction

A local museum located on Niagara-On-The-Lake has contacted Brock University, St. Catherines with the task of designing, developing, and implementing an online interactive timeline application. This application will depict the events surrounding the history of Niagara, with special attention given towards the war of 1812.

This document is designed to reference the testing plan, approach, and overall framework that will be used to address the testing process of the https://badger-timeline.infinityfreeapp.com/src/ - site. Here, this document will introduce:

- Objectives
- Test Strategies
- Requirements
- Features
- Schedule
- Risks/assumptions
- Tools

Furthermore, this document will introduce the necessary steps taken in the following testing phases:

1. Unit testing
2. Component testing
3. System integration testing (Acceptance testing)

1.2 Project Overview

The https://badger-timeline.infinityfreeapp.com/src/ -site is a powerful web-application that will be responsible for facilitating the process of information passage from the client(Niagara-on-the-lake Museum) to the user(General public).

On the user side, this website will allow users to view information about the museum, the date of any events or workshops sponsored by the museum, create an account for email notifications, and will also have an interactive timeline application that will give the user an experience through Niagara's rich history. This web-app can be accessed and experienced through any device and in any orientation.

On the client side, this website will allow permitted staff members(admins) to edit content directly on the timeline such as texts and images. In addition, admins will also be allowed to edit content on the web such as its services, as well as adding any more additional information about the museum.

Section 2 – Objectives and Tasks

2.1 Objectives

The objective of this test is to verify that the web-application - https://badger-timeline.infinityfreeapp.com/src/ is working as intended. To verify, all major components of the application must work as intended. The web-application can be considered and thought of as two major components: the webpage component and the timeline component. Both of these components will each be tested separately to check that they are working according to specifications, before finally testing the system as a whole.

The objective of testing the web-page component is to check and verify the various features offered in the homepage. These tests can include and are not limited to checking all links found on the homepage, checking that each button found works as intended, to verify the login/register system, and also to check that the webpages can be readily accessible by various devices such as PCs, Notebooks, and mobile devices. All the while having a reasonable amount of expectations towards the UI/UX associated with each device.

The object of testing the timeline component is to check and verify that the various features implemented in the timeline are all working in accordance to its specifications. These tests can include but are not limited to checking each individual point on the timeline spectrum, checking the various ways to navigate the timeline, checking how the information is presented, editing features for admin users, the sub-timeline itself, image view function, and etc. The timeline application must also be able to operate on various devices such as PCs, Notebooks, and mobile devices, and have a reasonable amount of expectations towards the UI/UX associated with each device.

The final product of this test plan will yield both a production-ready web-application, and also a set of test-cases and scenarios that can be reused whenever a new feature is to be added or changed in the future and/or for maintenance purposes.

2.2 Tasks

This section highlights the various tasks presented in this test plan document. The Tasks are separated and listed by its responsibility as a whole towards the testing plan. These tasks can be subject to change, but can include and not limited to:

- Testing phase
- Post-testing
- Problem reporting
- Retesting
- Acceptance Testing
- Regression Testing

Section 3 – Scope

3.1 General

This section will describe which components of the system will be tested, basic and/or added functions that are specific to a component, interface of each component, how the component will interact once integrated with one another, and the expectations of the integrated system.

There are three components that make up the overall web-application system. These components consist of the database, web-pages, and the timeline application.

The database is wholly responsible for storing and retrieving information, and is used by both the web-page component and the timeline component. Once integrated with the web-page component, the database will be responsible for providing information about any upcoming events and workshops hosted by the museum, along with storing and authenticating information for requests involving login/registration. Furthermore, the database is responsible for storing roles that are associated with each user. Once integrated with the timeline component, the database will have the responsibility of providing stored information for each historical time-period and event for the timeline to display. In addition, the database will need to support the editing feature by having the ability to delete old entries and store new entries. All interactions involving the database will be done through either the web-page component or timeline application component. The user cannot directly access the database itself, it will remain independent.

The web-page component will be responsible for greeting the user and will act as the main hub of interaction for information. The web-page will not contain any information about the history of Niagara, but will instead contain information and content about the client(Niagara-On-The-Lake Museum). Content and features presented on the web-page component include information about services offered by the museum, information about the museum itself, up-coming events/workshops that are hosted by the museum, and a login/register feature. Once integrated with the database, the web-page component will retrieve login information, as well as store registration information from the database. The web-page will then get the role of the user, and display the correct actions associated with the role of the user. Once integrated with the timeline component, the web-page will be responsible for providing a user a path to the timeline application either through a link/button, or through the search feature.

The timeline application component is responsible for giving the user a rich walkthrough experience on the events and historical periods surrounding the Niagara region. Unlike the web-page component, this timeline component will be solely responsible for displaying any and all information regarding historical periods and any events surrounding those periods to the user. Once integrated with the database, the timeline component will retrieve the necessary information regarding history periods and events and display the information for the user to view. Any/all interactions involving the database will be done through the timeline component, such as when an admin wants to make an edit request for a historical event. Once integrated with the web-page application, the timeline component will have very minimal interaction with the web-page, since the timeline component will not need to rely on the web-page for anything. The two components will mostly be responsible for providing a link to each other.

3.2 Tactics

This section will describe how the testing process of the scope will be carried out in regards to the exploratory, functional, and acceptance level of testing.

Exploratory level will be responsible for ensuring that unit level defects are removed before the next level of testing can start. This level of testing will be carried out in the web-application with the use of test scripts. This testing can be done by analyzing the software code and determining whether or not the unit will work as intended. Some examples include basic navigation and modules.

Functional level will be responsible for testing the function of each feature. This level of testing will be carried out on the web-application with the use of test-scripts. This testing can be done by feeding in an input for a given feature and then validating the output of the feature. Some examples include seeing whether the database has saved an entry, whether the user is able to log in using the correct credentials, and etc.

Acceptance level will be responsible for testing the overall functionality of the system. This level of testing will be carried out in the web-application without the use of test-scripts, as the goal is

to test from a user point of view. This testing can be done by taking a user story and attempting to validate the user story.

Section 4 – Testing Strategies

This section will describe the strategies used during the testing phase. The goal of this section is to verify the functionality of https://badger-timeline.infinityfreeapp.com/src/ -site is working according to its specifications. Here, the testing is divided into distinct stages, with each stage having a clearly defined object and goal.

To ensure a complete and realistic test, all test data and environments will be produced with the sole purpose of emulating a production environment as much as possible. Furthermore, each test will be a repeatable and quantifiable process for the purpose of producing a reliable and consistent testing kit.

4.1 Unit Testing

The unit testing stage will cover individual components in isolation. This is the defect testing stage, where individual functions and methods are tested to ensure that they are working as intended before being integrated to create a component.

This stage is done in both a black box and white box method of testing. The black-box tests will be conducted on the site, where testers will merely check for the existence of a unit. The white-box tests will involve a thorough look into the source code, where testers will be checking the functionality of an object or method via code.

The following includes a few examples test cases outlining the process of unit testing

Test case 1:	checking if homepage layout remains consistent when the window is in full screen.	1. Go to homepage using url: https://badger-timeline.infinityfreeapp.com/public_html/ 2. Open window to full screen 3. manually	Seeing how the homepage will look when the size of the window is. Checking the orientation in the order of: 1. Title and background 2.	Title and background is on top, followed by the 'view detail' cards in the sequential order of: Indigenious, african canadians, and european	As expected	Pass

		drag window screen to decrease size until it stops.	Indigenious card 3. African canadian card 4. european colonialist card 5. the war of 1812 section	colonialists and the 'war of 1812' section is on the bottom.		
Test case 2:	checking that are dates in column are all clickable and that an overlay will appear once clicked upon.	1. Go to edit content page using url: Go to homepage using url:https://badger-timeline.infinityfreeapp.com/public_html/views/pages/timeline 2. Click on all available dates in the row	Here we are checking to see if an overlay will appear once a date has been clicked. The test data will consist of all available dates from the base timeline. Such as: <5000, 1500, 1749, 1764, 1769, 1775, 1779, 1791, 1804, 1812, 1815, 1824, 1829, 1830, 1831, 1839, 1859, 1860, 1862, 1864, 1869, 1870, 1875, 1877, 1885, 1888, 1900, 1906, 1913, 1916, 1918, 1939	Once an timeline has been clicked, an overlay will appear.	As expected	Pass

Test case 3:	check text-to-speech feature on photocard/ overlay is accessible.	1. Go to edit content page using url: Go to homepage using url:https://badger-timeline.infinityfreeapp.com/public_html/views/pages/timeline 2. Click on all available timeline points in the row 3. use mouse to hover over the overlay that appears when user clicks on a timeline point 4. When card is flipped, use mouse to the speaker icon.	Here we are checking to see if a speaker icon will appear once a photocard overlay is flipped. The test data used includes: The timeline page:https://badger-timeline.infinityfreeapp.com/public_html/views/pages/timeline Timeline points: <5000, 1500, 1749, 1764, 1769, 1775, 1779, 1791, 1804, 1812, 1815, 1824, 1829, 1830, 1831, 1839, 1859, 1860, 1862, 1864, 1869, 1870, 1875, 1877, 1885, 1888, 1900, 1906, 1913, 1916, 1918, 1939	Once a timeline point has been clicked, an overlay will appear. When the user hovers their cursor over the overlay, the overlay will appear to 'flip' Once the card has flipped, a speaker icon will be visible on the top-left corner of the photocard/ overlay	As expected	Pass
Test case 4:	Check that subtimeline is accessible	1. Go to edit content page using url: Go to homepage using	Timeline page 1804 timeline	The line between the two points should appear	As expected	Pass

		url:https://badger-timeline.infinityfreeapp.com/public_html/views/pages/timeline 2. Use mouse to hover over space between 1804-1812	point 1812 timeline point space between the two points on the line	clickable, signifying that the timeline		
Test case 5:	Check to see if the 'email' field in the log in section is an editable field.	1. Go to homepage using url: https://badger-timeline.infinityfreeapp.com/public_html/views/user/account 2.Click on the email field 3. Input a string	1. Log in page 2. Log in section of Log in page 3. 'email' field 4. input string: "Leeebron James"	On the log in section, the 'email' field should be editable, and display the input string: "Leeebron James"	As expected	Pass
Test case 6:	Check to see if the 'password' field in the sign up section is an editable field.	1. Go to homepage using url: https://badger-timeline.infinityfreeapp.com/public_html/views/user/account	1. Log in page 2. Sign up section of Log in page 3. 'password' field	On the sign up section, the 'password' field should be editable, and display a hidden password of length 5	As expected	Pass

		2. Go to 'sign up' section of page 3.Click on the sign up section 4. Input a string	4. input string: "james"			

4.2 Component Testing

The component testing phase will cover components which are combined to form a composite component. This phase will come after the unit testing phase, to ensure that no potential problems will arise if a unit is not working properly.

This stage can be done using a black-box and white-box testing, where black-box tests will be conducted on the site itself, and the white-box tests will involve a thorough look into the source code and validating the input and output of test cases.

The following table includes a few examples test cases outlining the process of component testing:

Test case 1:	Check homepage visit now buttons	1. Go to homepage: https://badger-timeline.infinityfreeapp.com/public_html/ 2.Move mouse to hero section portion of the website 3. Click on 'visit now button'.	Homepage Visit now button	The 'visit now' button should lead the user directly to the timeline page.	As expected	Pass

Test case 2:	Check input requirements for the email field of the login section.	1. Go to homepage using url: https://badger-timeline.infinityfreeapp.com/public_html/views/user/account 2.Click on the email field 3. Input a string not containing an '@'	1. Log in page 2. Log in section of Log in page 3. 'email' field 4. input string: "LeeebronJames"	Here, the purpose of this test is to see how the system responds when a string that does not contain an '@' symbol is used as an input on the email field. The system should tell you that an '@' symbol is required within the input string	As expected	Pass
Test case 3:	Check input requirements for the email field of the login section.	1. Go to homepage using url: https://badger-timeline.infinityfreeapp.com/public_html/views/user/account 2.Click on the email field 3. Input a string containing '@' symbol and nothing else	1. Log in page 2. Log in section of Log in page 3. 'email' field 4. input string: "KD@"	Here, the purpose of this test is to see how the system responds when a string with an'@' symbol and empty space after, is used as an input In this case, the system should tell you that the following input	As expected	Pass

		following		"KD@" is incomplete.		
Test case 4:	Check Select Category feature	1. Go to homepage: https://badger-timeline.infinityfreeapp.com/public_html/views/pages/timeline 2. Go to the timeline section 3. Click on the Menu icon on the far right of the timeline 4. Select a category 5. Click apply	Timeline page Menu/Select Category icon Select option: War History Apply button	Here, the purpose of this test is to see how the system responds given an category to filter event. In this case, the timeline will hide all other points on the timeline except for the points that are relevant to war events, such as 1791, 1812, 1815, and 1916.	As expected	Pass
Test case 5:	Check input requirements for the sign up section by only entering an input for the 'Username' field and leaving all other fields empty	1. Go to homepage using url: https://badger-timeline.infinityfreeapp.com/public_html/views/user/account 2. Navigate to Sign Up	Log in page Sign Up section username input: "goatjames" sign up button	Here, the purpose of this test is to see how the system responds when only the 'username' field has an input and all other fields are left blank during	As expected	Pass

		section 3.Click on 'username' field 4. input a string		the sign up process. The system should respond by prevent the user from signing up by making the sign up button unclickable.		

4.3 System and Integration Testing

System integration testing is when all the components of the system is integrated together and tested as a whole system. The goal of system testing is to focus on testing the interaction between the components of the system. Here, the login page, profile page, homepage, timeline page, edit content page are integrated together with the database such that it produces a seamless and refined product. The following table below outlines a few sample testcases done on the complete system.

Test case 1	Check sign up feature	1. Go to log in page: https://badger-timeline.infinityfreeapp.com/public_html/views/user/account 2. enter valid inputs 3. click sign up button	Log in page Valid inputs for username, email, and password fields	The system will give a successful message notification to the user: "Registration successful! You can now log in."	As expected	Pass
Test case 2	Check sign up feature using pre-	1. Go to log in page: https://bad	Log in page A pre-	The system will give a unsuccessfu	As expected	Pass

	existing email in the database	ger-timeline.infinityfreeapp.com/public_html/views/user/account 2. enter an already pre-existing email address in the database 3. enter rest of fields with valid inputs	existing email on the database Valid inputs for username, password fields.	l message notification to the user: "Registration failed. Please try again."		
Test case 3	Check homepage when user is logged in	1. Go to homepage: https://badger-timeline.infinityfreeapp.com/public_html/ 2. naivgate to header section	Homepage	If the user is logged into an account, the homepage should reflect that fact. A 'user' icon will replace the log in link.	As expected	Pass
Test case 4	Check admin access for timeline page	1. Using a admin account, go to log in page: https://badger-timeline.infinityfreeapp.com/public_html/views/pages/timeline	Timeline page Timeline points: <5000, 1500, 1749, 1764, 1769, 1775, 1779, 1791, 1804, 1812, 1815, 1824, 1829, 1830, 1831, 1839, 1859,	The view and edit button are visible, when accessing the timeline using a admin account. This is because only admins	As expected	Pass

		2. verify that the 'view and edit button is visible	1860, 1862, 1864, 1869, 1870, 1875, 1877, 1885, 1888, 1900, 1906, 1913, 1916, 1918, 1939	have the privilege of editing the content on the timeline.		
Test case 5	Check date edit feature	1. Using an admin account, go to edit content page: https://badger-timeline.infinityfreeapp.com/public_html/views/pages/edit_content 2. Click on a time event from list 3. click on date field 4. input new date 5. Click save button	Edit content page Timeline events: <5000, 1500, 1749, 1764, 1769, 1775, 1779, 1791, 1804, 1812, 1815, 1824, 1829, 1830, 1831, 1839, 1859, 1860, 1862, 1864, 1869, 1870, 1875, 1877, 1885, 1888, 1900, 1906, 1913, 1916, 1918, 1939 four digit year such as: "1899" save button	The change is reflected onto the timeline.	As expected	Pass
Test case 6	Check edit information feature	1. Using an admin account, go to edit content page: https://badger-timeline.infinityfreeapp.	Edit content page Timeline events: <5000, 1500, 1749, 1764, 1769, 1775, 1779, 1791, 1804,	The change is reflected onto the timeline.	As expected	Pass

		com/public_html/views/pages/edit_content 2. Click on a time event from list 3. click on information field 4. input new string 5. Click save button	1812, 1815, 1824, 1829, 1830, 1831, 1839, 1859, 1860, 1862, 1864, 1869, 1870, 1875, 1877, 1885, 1888, 1900, 1906, 1913, 1916, 1918, 1939 Any string such as: "In the midst of Vimy Ridge, the troops from Canada....."			

The above tables of unit testing, component testing, and system integration testing are just a few examples of test cases done on the system. To view a full list of test cases, the link can be found here: https://github.com/sowmyamovva/NOTL-Museum-Interactive-Timeline/tree/main/Testing

4.4 Performance and Stress Testing

Stress testing is a non-functional software test. The goal of stress testing is to test an application while it is stressed. A stressed application is when it is working at its maximum limit. Find the maximum limit of our application and test our program to get as close to or reach this limit.

4. database can hold up to 8 records. Load all these records into our database and make sure no crashes occur.
5. Capacity of users on the website will need to be checked. Having 5 concurrent users on the website is max. Test to make sure 8 concurrent users will not crash the website.

ID	Scenario	Test Steps	Test Data	Expected Results	Actual Results	Pass/Fail

Case1	Test max records for database	Create 8 records Load records into database	8	No crashes	As Expected	Pass
Case2	Test max concurrent users on the website	Create 8 users All signed in at the same time All users on the website at the same time	8	No crashes	As Expected	Pass

Performance testing is a non-functional software test where the application will be tested on its stability, speed, scalability, and responsiveness during a given workload. Note that finding bugs is not the goal for performance testing. We are looking for long loading times, bottlenecking, slow response time, and poor scalability. For bottlenecking issues, a user's CPU, memory, internet, OS, disk space can all play roles in why they are experiencing issues.

6. Checking max concurrent users before software crashes
7. Check response time for a certain number of users ensuring it does not cause the application to be slow
8. Checking the execution time for our database when 8 records are read/written simultaneously
9. Check application with poor Wi-Fi connection
10. Check CPU and memory usage of application and database server during high concurrent usage of application.

ID	Scenario	Test Steps	Test Data	Expected Results	Actual Results	Pass/Fail
Case1	max concurrent users before software crashes	Create 5+ users All signed in at the same time	5+	No bottlenecking No long loading times	As Expected	Pass

		All users on the website at the same time		No bottlenecking No slow response times No poor scalability		
Case2	response time for a certain number of users	Create 5 users All signed in at the same time All users on the website at the same time	5	No bottlenecking No long loading times No bottlenecking No slow response times No poor scalability	As Expected	Pass
Case3	Checking the execution time for our database when 1000 records are read/written simultaneously	Create 8 records Load records into database	8	No bottlenecking No long loading times No bottlenecking No slow response times No poor scalability	As Expected	Pass
Case4	Check application with poor wifi connection	Use slow internet speeds	No data	No bottlenecking No long loading times No bottlenecking	As Expected	Pass

				No slow response times No poor scalability		
Case5	Check CPU and memory usage of application and database server during high concurrent usage of application.	Create 5 users All signed in at the same time All users on the website at the same time	No data	No bottlenecking No long loading times No bottlenecking No slow response times No poor scalability	As Expected	Pass

4.5 Manual Testing

This section covers the manual testing done the interactive timeline. The list following contains user essential features that have been thoroughly tested:

- Dark mode and light mode work as intended
- Scroll feature works for timeline correctly
- The '<< 'and '>>' chevrons to navigate left and right for the timeline work as intended

ID	Year	Test Features	Expected Results	Actual Results	Pass/Fail
test1	<5000	1. Button Highlights 2. Button Expands 3. Image Displays 4. Hover Feature 5. Text Displayed 6. Speech Functionality	1. The Button Highlights 2. The Button Expands 3. Correct Image Displays 4. Hover Feature flips image and keeps scaling 5. Text Displayed is related to correct year 6. Speech Functionality works and reads the text 7. Three Dots will display additional information 8. This will read the main point of the selected year	1. Yes 2. Yes 3. Yes 4. Yes 5. Yes 6. Yes 7. Yes 8. Yes	Pass

		7. Three Dot Feature 8. Speak feature			
test2	1500	1. Button Highlights 2. Button Expands 3. Image Displays 4. Hover Feature 5. Text Displayed 6. Speech Functionality	1. The Button Highlights 2. The Button Expands 3. Correct Image Displays 4. Hover Feature flips image and keeps scaling 5. Text Displayed is related to correct year 6. Speech Functionality works and reads the text	1.Yes 2.Yes 3.Yes 4.Yes 5.Yes 6.Yes	Pass
test3	1749	1. Button Highlights 2. Button Expands 3. Image Displays 4. Hover Feature 5. Text Displayed 6. Speech Functionality	1. The Button Highlights 2. The Button Expands 3. Correct Image Displays 4. Hover Feature flips image and keeps scaling 5. Text Displayed is related to correct year 6. Speech Functionality works and reads the text	1.Yes 2.Yes 3.Yes 4.Yes 5.Yes 6.Yes	Pass
test4	1764	1. Button Highlights 2. Button Expands 3. Image Displays 4. Hover Feature 5. Text Displayed 6. Speech Functionality	1. The Button Highlights 2. The Button Expands 3. Correct Image Displays 4. Hover Feature flips image and keeps scaling 5. Text Displayed is related to correct year 6. Speech Functionality works and reads the text	1.Yes 2.Yes 3.Yes 4.Yes 5.Yes 6.Yes	Pass
test5	1769	1. Button Highlights 2. Button Expands 3. Image Displays	1. The Button Highlights 2. The Button Expands 3. Correct Image Displays 4. Hover Feature flips image and keeps scaling	1.Yes 2.Yes 3.Yes 4.Yes 5.Yes	Pass

		4. Hover Feature 5. Text Displayed 6. Speech Functionality	5. Text Displayed is related to correct year 6. Speech Functionality works and reads the text	6.Yes	
test6	1775	1. Button Highlights 2. Button Expands 3. Image Displays 4. Hover Feature 5. Text Displayed 6. Speech Functionality	1. The Button Highlights 2. The Button Expands 3. Correct Image Displays 4. Hover Feature flips image and keeps scaling 5. Text Displayed is related to correct year 6. Speech Functionality works and reads the text	1.Yes 2.Yes 3.Yes 4.Yes 5.Yes 6.Yes	Pass
test7	1779	1. Button Highlights 2. Button Expands 3. Image Displays 4. Hover Feature 5. Text Displayed 6. Speech Functionality	1. The Button Highlights 2. The Button Expands 3. Correct Image Displays 4. Hover Feature flips image and keeps scaling 5. Text Displayed is related to correct year 6. Speech Functionality works and reads the text	1.Yes 2.Yes 3.Yes 4.Yes. 5.Yes 6.Yes	Pass
test8	1790	1. Button Highlights 2. Button Expands 3. Image Displays 4. Hover Feature 5. Text Displayed 6. Speech Functionality	1. The Button Highlights 2. The Button Expands 3. Correct Image Displays 4. Hover Feature flips image and keeps scaling 5. Text Displayed is related to correct year 6. Speech Functionality works and reads the text	1.Yes 2.Yes 3.Yes 4.Yes. 5.Yes 6.Yes	Pass

test9	1791	1. Button Highlights 2. Button Expands 3. Image Displays 4. Hover Feature 5. Text Displayed 6. Speech Functionality	1. The Button Highlights 2. The Button Expands 3. Correct Image Displays 4. Hover Feature flips image and keeps scaling 5. Text Displayed is related to correct year 6. Speech Functionality works and reads the text (re-click to stop reading)	1.Yes 2.Yes 3.Yes 4.Yes. 5.Yes 6.Yes	Pass
test10	1804	1. Button Highlights 2. Button Expands 3. Image Displays 4. Hover Feature 5. Text Displayed 6. Speech Functionality	1. The Button Highlights 2. The Button Expands 3. Correct Image Displays 4. Hover Feature flips image and keeps scaling 5. Text Displayed is related to correct year 6. Speech Functionality works and reads the text (re-click to stop reading)	1.Yes 2.Yes 3.Yes 4.Yes. 5.Yes 6.Yes	Pass
test11	1812	1. Button Highlights 2. Button Expands 3. Image Displays 4. Hover Feature 5. Text Displayed 6. Speech Functionality	1. The Button Highlights 2. The Button Expands 3. Correct Image Displays 4. Hover Feature flips image and keeps scaling 5. Text Displayed is related to correct year 6. Speech Functionality works and reads the text (re-click to stop reading)	1.Yes 2.Yes 3.Yes 4.Yes. 5.Yes 6.Yes	Pass
test12	1815	1. Button Highlights 2. Button Expands 3. Image Displays 4. Hover Feature	1. The Button Highlights 2. The Button Expands 3. Correct Image Displays 4. Hover Feature flips image and keeps scaling 5. Text Displayed is related to correct year	1.Yes 2.Yes 3.Yes 4.Yes. 5.Yes 6.Yes	Pass

		5. Text Displayed 6. Speech Functionality	6. Speech Functionality works and reads the text (re-click to stop reading)		
test13	1824	1. Button Highlights 2. Button Expands 3. Image Displays 4. Hover Feature 5. Text Displayed 6. Speech Functionality	1. The Button Highlights 2. The Button Expands 3. Correct Image Displays 4. Hover Feature flips image and keeps scaling 5. Text Displayed is related to correct year 6. Speech Functionality works and reads the text (re-click to stop reading)	1.Yes 2.Yes 3.Yes 4.Yes. 5.Yes 6.Yes	Pass
test14	1829	1. Button Highlights 2. Button Expands 3. Image Displays 4. Hover Feature 5. Text Displayed 6. Speech Functionality	1. The Button Highlights 2. The Button Expands 3. Correct Image Displays 4. Hover Feature flips image and keeps scaling 5. Text Displayed is related to correct year 6. Speech Functionality works and reads the text (re-click to stop reading)	1.Yes 2.Yes 3.Yes 4.Yes 5.Yes 6.Yes	Pass
test15	1830	1. Button Highlights 2. Button Expands 3. Image Displays 4. Hover Feature 5. Text Displayed 6. Speech Functionality	1. The Button Highlights 2. The Button Expands 3. Correct Image Displays 4. Hover Feature flips image and keeps scaling 5. Text Displayed is related to correct year 6. Speech Functionality works and reads the text (re-click to stop reading)	1.Yes 2.Yes 3.Yes 4.Yes. 5.Yes 6.Yes	Pass
test16	1831	1. Button Highlights	1. The Button Highlights	1.Yes	Pass

		2. Button Expands 3. Image Displays 4. Hover Feature 5. Text Displayed 6. Speech Functionality	2. The Button Expands 3. Correct Image Displays 4. Hover Feature flips image and keeps scaling 5. Text Displayed is related to correct year 6. Speech Functionality works and reads the text (re-click to stop reading)	2.Yes 3.Yes 4.Yes. 5.Yes 6.Yes	
test17	1839	1. Button Highlights 2. Button Expands 3. Image Displays 4. Hover Feature 5. Text Displayed 6. Speech Functionality 7. Three Dot Feature 8. Speak feature	1. The Button Highlights 2. The Button Expands 3. Correct Image Displays 4. Hover Feature flips image and keeps scaling 5. Text Displayed is related to correct year 6. Speech Functionality works and reads the text (re-click to stop reading) 7. Three Dots will display additional information 8. This will read the main point of the selected year	1. Yes 2. Yes 3. Yes 4. Yes 5. Yes 6. Yes 7. Yes 8. Yes	Pass
test18	1859	1. Button Highlights 2. Button Expands 3. Image Displays 4. Hover Feature 5. Text Displayed 6. Speech Functionality	1. The Button Highlights 2. The Button Expands 3. Correct Image Displays 4. Hover Feature flips image and keeps scaling 5. Text Displayed is related to correct year 6. Speech Functionality works and reads the text (re-click to stop reading)	1.Yes 2.Yes 3.Yes 4.Yes. 5.Yes 6.Yes	Pass
test19	1860	1. Button Highlights 2. Button Expands 3. Image Displays	1. The Button Highlights 2. The Button Expands 3. Correct Image Displays 4. Hover Feature flips image and keeps scaling	1.Yes 2.Yes 3.Yes 4.Yes. 5.Yes 6.Yes	

		4. Hover Feature 5. Text Displayed 6. Speech Functionality	5. Text Displayed is related to correct year 6. Speech Functionality works and reads the text (re-click to stop reading)		
test20	1862	1. Button Highlights 2. Button Expands 3. Image Displays 4. Hover Feature 5. Text Displayed 6. Speech Functionality	1. The Button Highlights 2. The Button Expands 3. Correct Image Displays 4. Hover Feature flips image and keeps scaling 5. Text Displayed is related to correct year 6. Speech Functionality works and reads the text (re-click to stop reading)	1.Yes 2.Yes 3.Yes 4.Yes. 5.Yes 6.Yes	Pass
test21	1864	1. Button Highlights 2. Button Expands 3. Image Displays 4. Hover Feature 5. Text Displayed 6. Speech Functionality	1. The Button Highlights 2. The Button Expands 3. Correct Image Displays 4. Hover Feature flips image and keeps scaling 5. Text Displayed is related to correct year 6. Speech Functionality works and reads the text (re-click to stop reading)	1.Yes 2.Yes 3.Yes 4.Yes. 5.Yes 6.Yes	Pass
test22	1869	1. Button Highlights 2. Button Expands 3. Image Displays 4. Hover Feature 5. Text Displayed	1. The Button Highlights 2. The Button Expands 3. Correct Image Displays 4. Hover Feature flips image and keeps scaling 5. Text Displayed is related to correct year 6. Speech Functionality works and reads the text (re-click to stop reading)	1.Yes 2.Yes 3.Yes 4.Yes. 5.Yes 6.Yes	Pass

		6. Speech Functionality			
test23	1870	1. Button Highlights 2. Button Expands 3. Image Displays 4. Hover Feature 5. Text Displayed 6. Speech Functionality 7. Three Dot Feature 8. Speak feature	1. The Button Highlights 2. The Button Expands 3. Correct Image Displays 4. Hover Feature flips image and keeps scaling 5. Text Displayed is related to correct year 6. Speech Functionality works and reads the text (re-click to stop reading) 7. Three Dots will display additional information 8. This will read the main point of the selected year	1.Yes 2.Yes 3.Yes 4.Yes 5.Yes 6.Yes 7. Yes 8. Yes	Pass
test23	1875	1. Button Highlights 2. Button Expands 3. Image Displays 4. Hover Feature 5. Text Displayed 6. Speech Functionality	1. The Button Highlights 2. The Button Expands 3. Correct Image Displays 4. Hover Feature flips image and keeps scaling 5. Text Displayed is related to correct year 6. Speech Functionality works and reads the text (re-click to stop reading)	1.Yes 2.Yes 3.Yes 4.Yes. 5.Yes 6.Yes	Pass
test24	1877	1. Button Highlights 2. Button Expands 3. Image Displays 4. Hover Feature 5. Text Displayed 6. Speech Functionality	1. The Button Highlights 2. The Button Expands 3. Correct Image Displays 4. Hover Feature flips image and keeps scaling 5. Text Displayed is related to correct year 6. Speech Functionality works and reads the text (re-click to stop reading)	1.Yes 2.Yes 3.Yes 4.Yes. 5.Yes 6.Yes	Pass
test25	1885	1. Button Highlights	1. The Button Highlights	1.Yes	Pass

		2. Button Expands 3. Image Displays 4. Hover Feature 5. Text Displayed 6. Speech Functionality	2. The Button Expands 3. Correct Image Displays 4. Hover Feature flips image and keeps scaling 5. Text Displayed is related to correct year 6. Speech Functionality works and reads the text (re-click to stop reading)	2.Yes 3.Yes 4.Yes. 5.Yes 6.Yes	
test26	1888	1. Button Highlights 2. Button Expands 3. Image Displays 4. Hover Feature 5. Text Displayed 6. Speech Functionality 7. Three Dot Feature 8. Speak feature	1. The Button Highlights 2. The Button Expands 3. Correct Image Displays 4. Hover Feature flips image and keeps scaling 5. Text Displayed is related to correct year 6. Speech Functionality works and reads the text (re-click to stop reading) 7. Three Dots will display additional information 8. This will read the main point of the selected year	1.Yes 2.Yes 3.Yes 4.Yes 5.Yes 6.Yes 7. Yes 8. Yes	Pass
test27	1900	1. Button Highlights 2. Button Expands 3. Image Displays 4. Hover Feature 5. Text Displayed 6. Speech Functionality	1. The Button Highlights 2. The Button Expands 3. Correct Image Displays 4. Hover Feature flips image and keeps scaling 5. Text Displayed is related to correct year 6. Speech Functionality works and reads the text (re-click to stop reading)	1.Yes 2.Yes 3.Yes 4.Yes. 5.Yes 6.Yes	Pass
test28	1906	1. Button Highlights 2. Button Expands 3. Image Displays 4. Hover Feature	1. The Button Highlights 2. The Button Expands 3. Correct Image Displays 4. Hover Feature flips image and keeps scaling 5. Text Displayed is related to correct year	1.Yes 2.Yes 3.Yes 4.Yes 5.Yes 6.Yes 7. Yes 8. Yes	Pass

		5. Text Displayed 6. Speech Functionality 7. Three Dot Feature 8. Speak feature	6. Speech Functionality works and reads the text (re-click to stop reading) 7. Three Dots will display additional information 8. This will read the main point of the selected year		
test29	1913	1. Button Highlights 2. Button Expands 3. Image Displays 4. Hover Feature 5. Text Displayed 6. Speech Functionality	1. The Button Highlights 2. The Button Expands 3. Correct Image Displays 4. Hover Feature flips image and keeps scaling 5. Text Displayed is related to correct year 6. Speech Functionality works and reads the text (re-click to stop reading)	1.Yes 2.Yes 3.Yes 4.Yes 5.Yes 6.Yes	Pass
test30	1916	1. Button Highlights 2. Button Expands 3. Image Displays 4. Hover Feature 5. Text Displayed 6. Speech Functionality	1. The Button Highlights 2. The Button Expands 3. Correct Image Displays 4. Hover Feature flips image and keeps scaling 5. Text Displayed is related to correct year 6. Speech Functionality works and reads the text (re-click to stop reading)	1.Yes 2.Yes 3.Yes 4.Yes. 5.Yes 6.Yes	Pass
test31	1918	1. Button Highlights 2. Button Expands 3. Image Displays 4. Hover Feature 5. Text Displayed 6. Speech Functionality	1. The Button Highlights 2. The Button Expands 3. Correct Image Displays 4. Hover Feature flips image and keeps scaling 5. Text Displayed is related to correct year 6. Speech Functionality works and reads the text (re-click to stop reading)	1.Yes 2.Yes 3.Yes 4.Yes. 5.Yes 6.Yes	Pass

test32	1939	1. Button Highlights 2. Button Expands 3. Image Displays 4. Hover Feature 5. Text Displayed 6. Speech Functionality	1. The Button Highlights 2. The Button Expands 3. Correct Image Displays 4. Hover Feature flips image and keeps scaling 5. Text Displayed is related to correct year 6. Speech Functionality works and reads the text (re-click to stop reading)	1.Yes 2.Yes 3.Yes 4.Yes. 5.Yes 6.Yes	Pass

Category (Timeline Feature)

ID	Select Category	Expected Results	Actual Results	Pass/Fail
test1	Indigenous History	1. Correct years are displayed when filter is selected 2. Years Work as intended	1. Correct years are displayed 2.The years' work as intended	Pass
test2	European Settlers	1. Correct years are displayed when filter is selected 2. Years Work as intended	1. Correct years are displayed 2.The years' work as intended	Pass
test3	War	1. Correct years are displayed when filter is selected 2. Years Work as intended	**1.** Correct years are displayed 2.The years' work as intended	Pass

test4	Transportation	1. Correct years are displayed when filter is selected 2. Years Work as intended	1. Correct years are displayed 2.The years' work as intended	Pass
test5	Fishing	1. Correct years are displayed when filter is selected 2. Years Work as intended	1. Correct years are displayed 2.The years' work as intended	Pass
test6	Historical Figures	1. Correct years are displayed when filter is selected 2. Years Work as intended	1. Correct years are displayed 2.The years' work as intended	Pass

Bounded Value Analysis on Filter Range (Timeline Feature)

ID	Filter Range	Expected Results	Actual Results	Pass/Fail
test1	0-5000	All years between 0-5000 will be displayed along with working functionalities.	All years display and work as expected	Pass
test2	2000-5001	All years between 2000-5001 will be displayed along with working functionalities.	No years are displayed as it should as none exist.	Pass
test3	1500-4999	All years between 1500-4999 will be	“<5000” year does not display as it should.	Pass

		displayed along with working functionalities.	All other year's display and work as expected	
test4	0-10000	All years between 0-10000 will be displayed along with working functionalities	All years are displayed and working which is incorrect because year "<5000" should not be displayed.	Fail
test5	1700-1800	All years between 1700-1800 will be displayed along with working functionalities.	All years are displayed and working	Pass

4.6 User Acceptance Testing

User acceptance testing is the last phase of the software testing process. Conduct your tests on the software by its intended audience. Determine if all requirements of your contract for the application are met. This testing ensures your product is finished, effective in its goal, or does not have faults/bugs.

The test cases will be based on few examples of user stories created during the earlier stages.

Stakeholder	User story	Steps	Pass/fail
All	As any role, I want to be able to register and log in	1. Go to homepage, 2. Click on 'log in' 3. Sign up 4. Log in	**Pass**
Editor	As an editor, I want to be able to make edits without having access to sensitive information so that all information is easily editable	1. using an admin account, go to homepage 2. click 'Visit now' 3. Click 'edit' button on timeline page	**Pass**
Editor	As an editor, I want to be able to have access to content material but not user information.	1. using an admin account, go to homepage	**Pass**

		2. click ‘Visit now’ 3. Click ‘edit’ button on timeline page	
User	As a user, I would like to have audio and visuals to create an immersive experience that compliments intractability so that the user can be engaged as much as possible without being physically present.	1. Go to homepage 2. Click ‘Visit now’ 3. Select timeline point 4. hover over photocard 5. Click speaker icon	**Pass**
User	As a user, I would like to have accessibility features available such as text to speech audio or audio description so that users that are visually impaired or disabled can still interact with the virtual museum features.	1. Go to homepage 2. Click ‘Visit now’ 3. Select timeline point 4. hover over photocard 5. Click speaker icon	**Pass**
Member	As a member, I want to be able to manage my user profile	1. Log in 2. Click on user profile icon 3. modify account	**Pass**
User	As a user, I want to be able to search for info using filters so that I can pinpoint exactly what I’m looking for	1. Go to homepage 2. Click ‘Visit now’ 3. Select filters on top	**Pass**
User	As a user, I want to be able to hover over something with my mouse, so that I can know the functionality.	1. Go to homepage 2. Click ‘Visit now’ 3. Select timeline point 4. hover over photocard	**Pass**

Section 5 – Requirements

5.1 Hardware Requirements

Include the minimal specifications required to run applications without issues. This will include the processor speed, memory and disk space. Ensure your hardware exceeds these requirements for adequate performance.

5.2 Environmental Requirements

Different browsers such as Google Chrome, Microsoft Edge, Safari, and Firefox. All these browsers will be used and tested on different operating systems such as Windows, macOS, and Linux.

Section 6 – Test Schedules

This section will describe the schedule in which each testing phase will take place. Below is a table detailing such schedule:

Date	Phase	Members involved
April 5 - April 8	Unit Testing	Development team, Testing team
April 9 - April 12	Component Testing	Development team, Testing team
April 12 - April 14	System Integration testing	Development team, Testing team
April 14 - April 18	User Acceptance Testing	Development team, Testing team

Section 7 – Control procedures

Reporting problems

Document any incidents or problems encountered during the testing process.

Changes made

Document the modifications made to the software to fix any problems.

Section 8 – Risk and Assumptions

This section underscores the risks and assumptions regarding the testing plan and schedules.

8.1 Risks

Some risks abouts the schedule includes delays of the testing process, as some bugs may take longer to fix than others. To address this, the team has agreed on the strategy of continuous iteration, where if a testing phase gets done before its scheduled end date, then the team will move onto the next phase immediately on the same date. This will allow for more overhead near the end of the schedule.

Failure to identify complex functionalities and time required to develop those functionalities. Having other group members help with the functionality when it becomes too complex, or time begins to run low. Speaking up and accepting help is a must to ensure the whole team can continue to move along without having down time.

8.2 Assumptions

This testing plan outline is created with assumptions in mind. Few of those assumptions include:

1. The web-application works consistently across all platforms.
2. Defects are reported through a chain of command.
3. There is no downtime for the environment during testing due to either updates or maintenance.
4. The database is treated as a black-box for most of the test-cases
5. The exploratory testing level will be carried out on the first day.
6. Production data is readily available at the start of each test.
7. The test cases follow the base timeline application without any edits done on the timeline.

Section 9 – Tools

Over the scope of this project, there have been numerous tools that the team has considered using. Though, the team narrowed down the selection to only two tools: JUnit and Selenium web driver. Both tools were used to create test scripts for the product, however, the tools were then later discontinued from its uses due to the scope of the project. To the team, the scope of the project was much too small to justify the uses automating the test cases. Instead, the team decided to manually test the product using written test cases. However, the code developed for the tools can still be found on the GitHub repository using this link: https://github.com/sowmyamovva/NOTL-Museum-Interactive-Timeline/tree/main/Testing