|  |  |
| --- | --- |
| **Name** | **Traversal of Line Graphs in Coaster Visualization - UC1** |
| Summary | The user has the experience via the Oculus Rift of riding along a line graph that represents election polling percentages in a roller coaster style cart. They have rewind, fast forward, and pause/play control via a dual shock controller. They also have the option of moving the cart between the different lines of the line graph.  Data relevant to the point in the graph being traversed will be displayed on a heads up display. This will include poll date, next poll date, current polling percentage, next polling percentage, and the candidate's graph that is being traversed. There will also be a 2d version of the graph inset on the screen. |
| Rationale | Line graphs are normally viewed in 2d from a side view perspective. Viewing the line graph from the first person will offer a unique perspective of the data. |
| Users | All users |
| Preconditions | 1. Unity is installed on the user's system. 2. The Oculus Rift is hooked up and installed. 3. The Data Visualization project is open. 4. A properly formatted CSV file is placed in the resources file in the Unity project directory. |
| Basic Course of Events | 1. The user starts the program by pressing the play button in Unity. 2. The user is presented with an option to pick either the map visualization, or the coaster visualization. Using the Dual Shock controller, they select the coaster visualization 3. The user is presented with a menu allowing them to select any of the available candidate’s lines to begin riding. Using the dual shock controller, they will select their preferred candidate with the d-pad and confirm by pressing the X button. 4. A line graph is created based on data loaded from an imported CSV file. 5. The user has the experience of riding in a cart following a line graph with data from the CSV file mentioned in step 4. 6. At each new point on the graph, the user will pass through a ring which is labeled with the polling date and the candidate’s current polling percentage. 7. The traversal of the graph ends. The user can choose to go back to the menu screen or rewind the visualization. |
| Alternative Paths | 1. At any point during the visualization, the user can use the Dual Shock controller to switch the candidate's graph that they are riding on. 2. At any point during the visualization the user can rewind or fast-forward the visualization using the Dual Shock controller 3. At any point during the visualization, the user can pause or resume the visualization using the Dual Shock controller. |
| Postconditions | The visualization has reached its end and the user chooses to exit the program, rewind the visualization, or exit to the visualization selection screen. |

|  |  |
| --- | --- |
| **Name** | **Viewing Poll Data in Map Visualization - UC2** |
| Summary | The user will be viewing a map of the United States. There will be poles with multiple spheres along them coming out of the states that have polling data.  An animation will play when the user decides to activate it. At this point, there will be no poles shown. The animation will parse the polling data in date order. As different states have polling data become available the poles will emerge, and spheres representing the candidates will grow or shrink based on that candidate's popularity.  The user will have the option to select a specific pole to view stats about that state's polling data. |
| Rationale | This visualization will offer a unique perspective of an election, allowing users to see poll results as they became available over a period of time on a state by state basis laid out on a US map. |
| Users | All users |
| Preconditions | 1. Unity is installed on the user's system. 2. The Oculus Rift is hooked up and installed. 3. The Data Visualization project is open. 4. A properly formatted CSV file is placed in the resources file in the Unity project directory. |
| Basic Course of Events | 1. The user starts the program by pressing the play button in Unity. 2. The user is presented with an option to pick either the map visualization, or the coaster visualization. Using the Dual Shock controller, they select the map visualization. 3. The user chooses to start the visualization 4. Pins begin to appear on the US map as polling data becomes available for each state. Spheres color coded to candidates scale and rearrange themselves on their respective pins in the following way:    1. Spheres will grow and shrink based on a candidate's polling percentage.    2. Spheres will be rearranged so that the leading candidate is on top, and the lowest ranking candidate is on the bottom. |
| Alternative Paths | 1. At any point during the visualization, the user can select a specific pin (enter investigation state) to view stats about the polls conducted in that state that the pin is located. These will include candidate names, percentage of votes currently allocated to candidates, number of votes currently allocated to candidates, 2. At any point during the visualization the user can rewind or fast-forward the visualization. 3. At any point during the visualization, the user can pause or resume the visualization. |
| Postconditions | The visualization has reached its end and the user chooses to exit the program, rewind the visualization, or exit to the visualization selection screen. |

|  |  |
| --- | --- |
| **Name** | **Loading a CSV File - UC3** |
| Summary | The user can load Data from a CSV file to create the visualizations that the program is capable of creating. |
| Rationale | This gives the program the versatility of being able to visualize any properly formatted CSV file. |
| Users | All users |
| Preconditions | 1. Unity is installed on the user's system. 2. The User has viewed the documentation explaining how to properly format a CSV file for use with the program. 3. The user has the project file for the data visualization program. 4. The user has hooked up an Oculus Rift to their computer. |
| Basic Course of Events | 1. The user has polling data which includes at least the following information for each poll:    1. End date of poll    2. Number of voters    3. Candidates    4. Polling percentages of each candidate    5. State of poll   2. The user formats the CSV to match the file format described in section 4.4.  3. The user opens the data visualization program.  4. On the main menu screen, the user selects the location of their CSV file from a dropdown list.  5. This will load the file into the Resources folder in the Unity  project directory.  6. The user can view the data in either the map or coaster visualization (both described.) |
| Alternative Paths | N/A |
| Postconditions | The loaded data can be viewed in the program. |

|  |  |
| --- | --- |
| **Name** | **Adding an Event Marker to Coaster Visualization - UC4** |
| Summary | The user can add an event marker to mark a specific point in the visualization to signify an even that causes the data to behave a certain way. The event marker will be a ring around the graph that the coaster can go through. There will be a line of text on the ring to label the event. |
| Rationale | The first person perspective can gain value from seeing a sudden rise, fall, or lull in polling numbers in correlation with a certain event. |
| Users | Advanced User |
| Preconditions | 1. Unity is installed on the user's system. 2. The user has data available to create an event marker. 3. The user understands the process of creating an event marker. |
| Basic Course of Events | 1. The user creates a properly formatted CSV file titled eventMarkers.csv and places it in the Resources folder in the project directory. 2. The user will start the program and choose the map visualization from the main menu. 3. The line graph is loaded based on information in the CSV file mentioned in UC-1. 4. The CSVReader.cs script will scan eventMarkers.csv. 5. The Event Marker object will be created at the point that represents the date derived from in step 4. 6. The text specified in the CSV file will be placed around the event marker. The text can have a maximum length of 30 characters. 7. The event will now be marked. |
| Alternative Paths | N/A |
| Postconditions | Event marker exists in the coaster visualization. |

|  |  |
| --- | --- |
| **Name** | **Rewinding Coaster Visualization - UC5** |
| Summary | The user can rewind the coaster visualization at any point after it starts in order to go back to a point of interest. |
| Rationale | If the user wants to examine a part of the visualization that been viewed already, they can easily navigate back to that part of the visualization. |
| Users | All Users |
| Preconditions | 1. Unity is installed on the user's system. 2. Data visualization project is open. 3. Coaster scene is open in Unity 4. Visualization has been started. |
| Basic Course of Events | 1. The user presses the L2 button on the Dual Shock Controller. 2. The user will have the perception of moving in reverse on the line graph.   3. The user lets go of the L2 button and the visualization resumes its normal forward trajectory. |
| Alternative Paths | N/A |
| Postconditions | The visualization resumes its normal forward trajectory from the point that left directional button was released. |

|  |  |
| --- | --- |
| **Name** | **Fast Forwarding Coaster Visualization - UC5** |
| Summary | The user can fast forward the coaster visualization at any point after it starts in order to go back to a point of interest. |
| Rationale | If the user wants to see the visualization more faster, or quickly get to certain point in the data, the user can fast forward. |
| Users | All Users |
| Preconditions | 1. Unity is installed on the user's system. 2. Data visualization project is open. 3. Coaster scene is open in Unity 4. Visualization has been started. |
| Basic Course of Events | 1. The user presses the R2 button on the Dual Shock controller. 2. The trajectory of the cart speeds up. 3. The user releases the R2 button. 4. The visualization resumes at the normal speed. |
| Alternative Paths | N/A |
| Postconditions | The visualization resumes at normal speed from the point that the right directional button was released. |

|  |  |
| --- | --- |
| **Name** | **Rewinding Map Visualization - UC6** |
| Summary | The user can rewind the map visualization at any point after it starts in order to go back to a point of interest. |
| Rationale | If the user wants to examine a part of the visualization that been viewed already, they can easily navigate back to that part of the visualization. |
| Users | All Users |
| Preconditions | 1. Unity is installed on the user's system. 2. Data visualization project is open. 3. Coaster scene is open in Unity 4. Visualization has been started. |
| Basic Course of Events | 1. The user presses the L2 button on the Dual Shock Controller. 2. Sphere objects will access information from candidate obects in reverse date order causing the visualization to move backwards in time. 3. The user lets go of the L2 key and the visualization resumes its normal forward-in-time trajectory. |
| Alternative Paths | N/A |
| Postconditions | The visualization resumes its normal forward trajectory from the point that R1 was released. |

|  |  |
| --- | --- |
| **Name** | **Fast Forwarding Map Visualization - UC7** |
| Summary | The user can speed up the map visualization if they wish to see it progress more quickly. |
| Rationale | If the user wants to skip to a certain point in the map visualization they can speed up the visualization to get the that point. |
| Users | All Users |
| Preconditions | 1. The user presses the R2 button on the Dual Shock Controller. 2. Sphere objects will access information from candidate obects in reverse date order causing the visualization to move backwards in time. 3. The user lets go of the R2 key and the visualization resumes its normal forward-in-time trajectory. |
| Basic Course of Events | 1. The user presses the R2 button on the Dual Shock Controller. 2. Sphere objects will access information from candidate obects in reverse date order causing the visualization to move backwards in time. 3. The user lets go of the R2 key and the visualization resumes its normal forward-in-time trajectory. |
| Alternative Paths | N/A |
| Postconditions | The visualization resumes its normal forward trajectory from the point that R2 was released. |

|  |  |
| --- | --- |
| **Name** | **Changing to Investigation State in Map Visualization - UC8** |
| Summary | When navigating in the Map Visualization, the user can switch into the investigation state, which will reveal more information about the specified state’s polls. |
| Rationale | The user may want to access more in depth information about a specific state. This will provide a convenient way to access that information. |
| Users | All Users |
| Preconditions | 1. User is viewing Map visualization in Navigation State. |
| Basic Course of Events | 1. User looks at an object. 2. Main camera will send a ray-cast towards that object.If the raycast hits the object, it will become illuminated. If the object is illuminated, the user has the option of pressing the X button to enter the investigation state. 3. The user presses the X button. 4. The pin object will get relevant information from its child spheres, which in turn get information from candidate objects. 5. This information will be displayed on a UI panel on the right hand side of the screen. Information displayed will include:    1. state name    2. the date of the position in the timeline    3. the number of observations    4. each of the candidates’ number of votes with percentages in parentheses 6. The user can press the O button when they no longer want to view the data. 7. The map visualization will resume its normal course.     . |
| Alternative Paths | N/A |
| Postconditions | The visualization resumes its normal course. |

|  |  |
| --- | --- |
| **Name** | **Selecting Coaster Visualization - UC9** |
| Summary | Upon the start of the application the user can select the coaster visualization from the main menu. |
| Rationale | The user may decide to view the coaster visualization of the data instead of the map visualization. |
| Users | All Users |
| Preconditions | 1. The user has started the application. |
| Basic Course of Events | 1. The user uses the Oculus Rift to look at the Coaster button on the Main Menu, thereby highlighting it. 2. The user presses the X button on the Dual Shock Controller, which brings up the Candidate Selection menu. 3. The user uses the Oculus Rift to look at the desired candidate line to ride. 4. The user presses the X button. |
| Alternative Paths | Before selecting the coaster visualization the user can perform UC3 to load a different CSV file into the applicaiton. |
| Postconditions | The coaster visualization begins in playback mode with the user riding the candidate line that was selected. The user can proceed with UC1. |

|  |  |
| --- | --- |
| **Name** | **Selecting Map Visualization - UC10** |
| Summary | Upon the start of the application the user can select the map visualization from the main menu. |
| Rationale | The user may decide to view the map visualization of the data instead of the coaster visualization. |
| Users | All Users |
| Preconditions | 1. The user has started the application. |
| Basic Course of Events | 1. The user uses the Oculus Rift to look at the Map button on the Main Menu, thereby highlighting it. 2. The user presses the X button on the Dual Shock Controller.. |
| Alternative Paths | Before selecting the map visualization the user can perform UC3 to load a different CSV file into the applicaiton. |
| Postconditions | The map visualization begins in playback mode. The user can proceed with UC2. |

|  |  |
| --- | --- |
| **Name** | **Pausing/Playing Coaster - UC11** |
| Summary | While viewing the Coaster visualization the user can pause or resume playback of the visualization. |
| Rationale | The user may decide to pause the visualization to analyze all the lines in detail. |
| Users | All Users |
| Preconditions | 1. Successful completion of UC9 |
| Basic Course of Events | 1. While in playback mode the user presses the Start button on the Dual Shock controller. 2. The cart’s movement should stop. All UI elements should stop updating. 3. While paused, the user presses the Start button on the Dual Shock controller. 4. The visualization resumes its normal course. |
| Alternative Paths | The user can continue to look around and toggle the UI while the visualization is paused. |
| Postconditions | The visualization toggles between playback and paused states. |

|  |  |
| --- | --- |
| **Name** | **Pausing/Playing Map - UC12** |
| Summary | While viewing the Map visualization the user can pause or resume playback of the visualization. |
| Rationale | The user may decide to pause the visualization to analyze all the spheres in detail. |
| Users | All Users |
| Preconditions | 1. Successful completion of UC10 |
| Basic Course of Events | 1. While in playback mode the user presses the Start button on the Dual Shock controller. 2. The pins and spheres on the map should stop updating. All UI elements should stop updating. 3. While paused, the user presses the Start button on the Dual Shock controller. 4. The visualization resumes its normal course. |
| Alternative Paths | The user can continue to move, look around, toggle the UI, and enter investigation mode while the visualization is paused. |
| Postconditions | The visualization toggles between playback and paused states. |

|  |  |
| --- | --- |
| **Name** | **Switching Lines in Coaster Visualization - UC13** |
| Summary | While traversing a graph in the coaster visualization, the user can switch the line being traversed. |
| Rationale | The user may decide that they wish to traverse a different line than the one that they were originally riding on. |
| Users | All Users |
| Preconditions | 1. Coaster visualization has been chosen by user and is underway. |
| Basic Course of Events | 1. The user presses either the left or right d-pad button on the Dual Shock controller. 2. The visualization resets. 3. The user is now riding on the line to either the left or right (depending on input) of the line they were previously riding on.    1. If the user was on either the left-most or right-most line, they will start the visualization on the opposite side of the graph (e.g. left-most side if they were previously traversing right-most. |
| Alternative Paths |  |
| Postconditions | The user is riding on the candidate’s line of their choosing. |

|  |  |
| --- | --- |
| **Name** | **Headtracking - UC14** |
| Summary | At any point in the program, the user can look around with the Oculus Rift’s headtracking. |
| Rationale | The most natural user view input with the Oculus Rift is headtracking. |
| Users | All Users |
| Preconditions | 1. Program is loaded. |
| Basic Course of Events | 1. The user looks around, and the camera follows their head position. |
| Alternative Paths |  |
| Postconditions | The camera follows the user’s head position. |