AutoVerse – test scenarios

Team members:

* Szymon Kasprzycki 193065
* Aleksandra Jażdżewska 193601
* Michał Jakielaszek 193300
* Łukasz Kornacki 195613

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **id** | **Name** | **Description** | **Type** | **Preliminary activities** | **Post-completion activities** |
| 1 | Setup vehicle parameters | Setting car parameters (dimension, weight, fuel type) which will be considered when navigating. | Functional tests | Agree to location sharing | Go back to main screen |
| **#** | **Priority** | **Case Name** | **Initial conditions** | **Steps** | **Expected outcome** |
| 1.1 | 1 | Set vehicle dimensions | You are in the “Car parameters” segment | Press on windows near ‘Length’, ‘Height’ and ‘Width’ and specify them in given units. Click confirm. | Vehicle dimensions are saved and considered when navigating. |
| 1.2 | 2 | Set vehicle weight | You are in the ‘Car parameters’ segment’ | Press on windows near ‘Weight’ and specify it in given units. Click confirm. | Vehicle weight is saved and considered when navigating. |
| 1.3 | 2 | Set vehicle fuel type | You are in the ‘Car parameters’ segment’ | Press on the window near ‘Fuel type’ and choose fuel type corresponding to your vehicle from the list. Click confirm. | Vehicle fuel type is saved and considered when navigating with refueling option. |
| 2 | Browse the destination | Browsing the destinations from map text. | Functional tests | Agree to location sharing | Get a list of suggested destinations |
| **#** | **Priority** | **Case Name** | **Initial conditions** | **Steps** | **Expected outcome** |
| 2.1 | **1** | Browsing destination suggestions from text. | You are in the navigating screen and pressed to window to search via text. | Press on the window to input destination via text. | User's past destinations appear or common destinations in the area. |
| 2.2 | **2** | Browsing destinations on the map. | User is in the navigating window. | Drag through the map to find the desired location.  Press and hold on to it. | User see the destination address. |
| 3 | Select the destination | Selecting and confirming chosen destination. | Functional tests | Agree to location sharing.  Input the destination through text or choose from the map. | The destination is set, and possible paths show up. |
| **#** | **Priority** | **Case Name** | **Initial conditions** | **Steps** | **Expected outcome** |
| 3.1 |  | Select destination from suggested. | User clicked on the window to input the destination in the navigating screen.  User inputted the destination. | 1. Search through the suggestions list.  2. Press on the correct destination  3. Confirm destination | Destination for navigating is set. |
| 3.2 |  | Select destination from map | User chose the destination from the map. | 1. Search through the map.  2. Press and hold on the correct destination  3. Confirm destination | Destination for navigating is set. |
| 4 | Browse available paths | Check if browsing available paths to achieve destination works properly. | Functional testing? | 1. Agree to sharing the location  2. Have valid destination selected  3. Confirm destination | Go back to selecting the destination |
| **#** | **Priority** | **Case Name** | **Initial conditions** | **Steps** | **Expected outcome** |
| 4.1 | 1 | Browsing path without setting car parameters and with no need to refuel on the way | You chose valid (existing) destination and didn’t input any parameter in ‘Car parameters’ | Click ‘Confirm’ button after selecting destination | At least one most optimal path is shown on the screen along with estimated duration |
| 4.2 | 1 | Browsing path with setting car parameters | You chose valid (existing) destination and input at least one car parameter | Click ‘Confirm’ button after selecting destination | At least one path that is most optimal and suitable to car’s parameter is shown along with truck stands |
| 4.3 | 1 | Browsing path with intention to refuel | You chose valid (existing) destination and marked refueling on the way | 1. Click fuel icon  2. Click ‘Confirm’ button after selecting destination | At least one path, that is optimal and have cheap gas station on the way, is shown |
|  |  |  |  |  |  |
| 5 | Get specific path details | Check if viewing specific path details works properly. | Functional testing? | 1. Agree to sharing the location  2. Have valid destination selected  3. Confirm destination  4. Browse paths | Go back to selecting path |
| **#** | **Priority** | **Case Name** | **Initial conditions** | **Steps** | **Expected outcome** |
| 5.1 | 1 | Viewing path details of chosen path | 1. Browse paths | 1. Click on the path  2. Click ‘path details’ button  3. Click ‘back’ button to se details of other paths | Path details of selected path are shown |
|  |  |  |  |  |  |
| 6 | Select the path to destination | Check if selecting path that is going to be navigated?? Works properly | Functional testing? | 1. Agree to sharing the location  2. Have valid destination selected  3. Confirm destination | End being navigated by clicking ‘X’ button at the top |
| **#** | **Priority** | **Case Name** | **Initial conditions** | **Steps** | **Expected outcome** |
| 6.1 | 1 | Select path from the map | 1. Click on the path | 1. Click ‘confirm’ button | Navigation mode is set up, user is navigated to the destination using selected path |
| 6.2 | 1 | Select path from ‘Path details’ | 1. Click on the path  2. Click on the ‘Path details’ | 1. Click ‘Confirm’ button | Navigation mode is set on, user is navigated to the destination using selected path |
|  |  |  |  |  |  |
| 7 | Receive alerts while driving | Checking weather receiving alerts while driving works properly | Functional tests | Confirm the path that you are going to take and start the ride | Dismissing the alert by clicking on “X” button |
| **#** | **Priority** | **Case Name** | **Initial conditions** | **Steps** | **Expected outcome** |
| 7.1 | 1 | Receiving alerts while driving to chosen destination | There was a report submitted on the road by another app user | 1. Choose a path that you are going to take 2. View a report popping up at the screen | The driver is effectively reported about a road incident |
|  |  |  |  |  |  |
| 8 | Let other drivers know if alert is still valid | Checking weather letting other drivers know if alert is still valid works properly | Functional tests | Confirm the path that you are going to take and start the ride |  |
| **#** | **Priority** | **Case Name** | **Initial conditions** | **Steps** | **Expected outcome** |
| 8.1 | 2 | Leaving the report as it is valid | There was a report submitted on the road by another app user | 1. Choose a path that you are going to take 2. View a report popping up at the screen 3. Do not dismiss it as it’s valid | Report will still be treated as valid and shown to other app users taking the path |
| 8.2 | 2 | Dismissing the report as it’s not valid | There was a report submitted on the road by another app user | 1. Choose a path that you are going to take 2. View a report popping up at the screen 3. Dismiss the report using “X” button as it’s not valid anymore | Report will not be treated as valid anymore and will not be shown to other users taking the path |
| 9 | Receive alternative path suggestions | Checking whether receiving alternative path suggestions works properly | Functional tests | Confirm the path that you are going to take and start the ride | Accept or reject the path suggestion |
| **#** | **Priority** | **Case Name** | **Initial conditions** | **Steps** | **Expected outcome** |
| 9.1 | 2 | Accepting alternative path suggestion | Shorter path to chosen destination was detected | 1. Choose a path that you are going to take 2. View a path change suggestion on the screen 3. Accept the path | Driver is being navigated through a new – shorter path |
| 9.2 | 2 | Rejecting alternative path suggestion | Shorter path to chosen destination was detected | 1. Choose a path that you are going to take 2. View a path change suggestion on the screen 3. Reject the path | Driver is being navigated through primarily chosen path |
| 9.3 | 1 | Not accepting nor rejecting alternative path suggestion | Shorter path to chosen destination was detected | 1. Choose a path that you are going to take 2. View a path change suggestion on the screen 3. Ignore the path suggestion | Driver is being navigated through primarily chosen path |
| 10 | Report event on the road | Checking whether the functionality of reporting the unexpected event on road works properly | Functional tests | 1. Have connection to the internet  2. Agree to share your location  3. Open navigation in the AutoVerse app  4. Drive car with navigation | Close navigation by clicking “X” button in the top left corner of the screen. |
| **#** | **Priority** | **Case Name** | **Initial conditions** | **Steps** | **Expected outcome** |
| 10.1 | 1 | Report accident on the road | There is an unreported accident on the road that you’re currently driving | 1. Choose “report” button in the bottom left corner of the screen.  2. Choose the “accident” type of the report.  3. Click “submit report” button.  4. Confirm submitting by clicking “confirm” button. | The report is successfully submitted in the app and other drivers using the AutoVerse are being warned about this incident. |
| 10.2 | 1 | Report police patrol on the road | There is an unreported police patrol on the road that you’re currently driving | 1. Choose “report” button in the bottom left corner of the screen.  2. Choose the “police patrol” type of the report.  3. Click “submit report” button.  4. Confirm submitting by clicking “confirm” button. | The report is successfully submitted in the app and other drivers using the AutoVerse are being warned about the patrol. |
| 10.3 | 1 | Report roadblock | There is an unreported roadblock on the road that you’re currently driving | 1. Choose “report” button in the bottom left corner of the screen.  2. Choose the “roadblock” type of the report.  3. Click “submit report” button.  4. Confirm submitting by clicking “confirm” button. | The report is successfully submitted in the app and other drivers using the AutoVerse are being warned about the block. |
| 10.4 | 1 | Report car on the roadside | There is an unreported car on the side of the road that you’re currently driving | 1. Choose “report” button in the bottom left corner of the screen.  2. Choose the “car on roadside” type of the report.  3. Click “submit report” button.  4. Confirm submitting by clicking “confirm” button. | The report is successfully submitted in the app and other drivers using the AutoVerse are being warned about the car on the side of the road. |
| 10.5 | 2 | Make report with the location specified from the map | There is some unexpected event on the road in the other location than your actual location | 1. Choose “report” button in the bottom left corner of the screen.  2. Choose the “Choose from map” option for location.  3. Choose the type of report that you want to submit.  4. Choose a location from the map.  5. Click “confirm” button.  6. Click “submit report” to add report.  6. Confirm submission by clicking “confirm” button. | The report is successfully submitted in the app and other drivers using the AutoVerse are being warned. |

**Performed tests:**

* 1. Set vehicle dimensions – no comments. Functionality works as intended.
  2. Set vehicle weight – no comments. Functionality works as intended.
  3. Set vehicle fuel type – no comments. Functionality works as intended.

2.1 Browsing destination suggestions from text – suggestions of the most fitting destination based on user’s input are shown. Works as expected.

2.2 Browsing destinations on the map – in the lo-fi prototype it is not yet possible to drag through the map. The rest of the functionality works as intended.

3.1 Select destination from suggested – no comments. Functionality works as intended.

3.2 Select destination from map – no comments. Functionality works as intended.

4.1 Browsing path without setting car parameters and with no need to refuel on the way – A few most optimal paths are shown on the screen and user can choose the desired one. Works as expected.

4.2 Browsing path with setting car parameters – paths suitable for specified car parameters are shown. Works as expected.

4.3 Browsing path with intention to refuel – suitable paths with gas stations along the way are shown on the screen. Works as expected.

5.1 Viewing path details of chosen path – No comments. The functionality works as intended.

6.1 Select path from the map – No comments. The functionality works as intended.

6.2 Select path from ‘Path details’ – No comments. The functionality works as intended.

7.1 Receiving alerts while driving to chosen destination – An exemplary alert is shown on the screen after a delay on the navigating screen.

8.1 Leaving the report as it is valid - No comments. The functionality works as intended.

8.2 Dismissing the report as it’s not valid - No comments. The functionality works as intended.

9.1 Accepting alternative path suggestion – a new path is shown to the user.

9.2 Rejecting alternative path suggestion – the path of navigation remains unchanged. Works as expected.

9.3 Not accepting nor rejecting alternative path suggestion – the path of navigation remains unchanged. Works as expected.

10.1 Report accident on the road – works as expected.

10.2 Report police patrol on the road – no comments. Works as expected.

10.3 Report roadblock – no comments. Works as expected.

10.4 Report car on the roadside – no comments. Works as expected.

10.5 Make report with the location specified from the map – no comments. Works as expected.

**Link to the figma project:**

<https://www.figma.com/file/OcTXEvC2tXPmnzKVQKhPsy/HCI---AutoVerse?type=design&node-id=0%3A1&mode=design&t=HuTRVmjmYKDlENZX-1>