# POI-Search

**Classes/ Files:**

* ORS.Gui.Tool.SearchPoi.js (for arranging the visible parts of the search panel)
* ORS.Gui.SearchPoi in ORS.Gui.Search.js (for the logic to trigger requests)
* ORS.OpenLS.js (for building the actual requests and sending them to the service)

**2 options**

* Search for POI in visible map
* Search for POI along a given route

**Search for POI in visible map**

* Happens all in ORS.Gui.SearchPoi.observeSearchfieldInput()
* ORS picks the center point of the visible map
* ORS calculates the distance to the edges of the visible map
* ORS delegates this information to ORS.OpenLS.js to build the service XML request “within distance”
* The request is sent to the Directory service.
* ORS receives the results and displays them on the map.

**Search for POI near route**

* Checkbox “search near route” has to be checked
* Distance + measurement unit must be selected (warning is shown if selected distance to route exceeds 5000 meters. This is the max. supported value by the directory service)
* These checks are performed in ORS.Gui.SearchPoi. observeCheckbox() and .observeInputfield()
* In ORS.Gui.SearchPoi. observeSearchfieldInput(): Based on the points along the route (not just start and end point!) that are saved in the route object as olRepresentation and the entered maxDistance a new XML request is formed in ORS.OpenLS.js.
* This request is a BufferRequest that is sent to the WPS service.
* It returns a polygon with the selected distance around the given route.
* This polygon again is handed over to ORS.OpenLS.js to perform the actual POI request “withinBoundary”
* This request is sent to the Directory service
* ORS receives the results and displays them on the map.
* Note: the near route feature is not supported by POIbyName requests (no category like “mini golf” or “amenity”)