**Name: Deepali Rawat**

**Suggestions for QRealTime plugin**

* Allow users to apply dynamic aggregations and filters to real-time data. In order to enable users to narrow down on certain subsets of data, this can entail integrating advanced filtering options based on attributes, spatial queries, time intervals, or custom expressions.
* Add real-time analytics capabilities to the plugin's capabilities. This can entail integrating with programmes or algorithms that analyse streaming data in real-time, such as anomaly detection, clustering, or pattern recognition.
* Expand the range of data sources and formats that are frequently supported by the plugin to provide connectivity possibilities for real-time applications, this may include interaction with databases, web services, sensor networks, or IoT platforms.
* Include a feature that enables users to configure individual alerts or notifications in response to real-time data situations. Users might, for instance, set criteria for specific data attributes, and the plugin would send out notifications or draw attention to the impacted features on the map when those thresholds are surpassed.
* Allow for the recording and archiving of real-time data streams for a future analysis or replay. Users could gather and save information from a certain time frame, giving them the opportunity to review and analyse it later or contrast it with earlier data.
* Establish reliable methods for validating and regulating data. Data that has been crowdsourced may contain mistakes, contradictions, and malicious additions. Users could check and confirm the data before adding it into their study by using the plugin's functionality to identify and flag possibly false or questionable data elements.
* Establish a reputation system that evaluates contributors' trustworthiness based on their prior contributions' calibre and performance. This could entail keeping track of elements like data consistency, accuracy, and conformity to predetermined data standards. The crowdsourced data might then be weighted or filtered based on the reputation of the contributors.
* Implement a versioning system for data that has been gathered from the public. It is crucial to monitor how the data has changed over time as contributors continue to update or correct their entries. Users should have access to earlier iterations of the data and be able to compare changes between them in order to see how the data has changed or been improved.
* Options for data filtering and crowdsourcing should be made available so that users can choose the crowdsourced data types they want to contribute or view. Users may want to provide particular standards or guidelines to guarantee that only reliable and pertinent data is displayed or accepted. This could entail creating data attribute filters, developing data moderation mechanisms, or establishing minimum reputation criteria for contributors.
* Include methods for people to offer input and report problems with crowdsourced data. This can be used to detect potential problems with data quality, offer recommendations for enhancements, or report any abuse or misuse of the crowdsourcing feature. Users' feedback loops can help to improve the plugin's crowdsourcing features over time.