Project Background

Qualcomm is the connected processor company for the intelligent edge with leadership in wireless connectivity, RF front-end, high-performance, low-power computing, multimedia, and on-device intelligence. Within the Test Base Station Project we design internal custom, integrated cellular call flow test boxes that rival or exceed capabilities of external third-party solutions. The project has over 200 Engineers worldwide and has deployed over $200 million of equipment for 5G wireless verification. The team supports a product portfolio that includes 3G, 4G, 5G, and V2X products.

Customer support teams face challenges in quickly identifying the root causes of user issues, especially with complex systems that produce large log files. To solve this, the TBS team developed LogAn, a Python-based tool that leverages libraries like Pandas and Bokeh to analyze logs, visualize results, and suggest solutions. LogAn streamlines troubleshooting, improving support efficiency and user satisfaction.

LogAn GUI's front end is an Electron, TypeScript, ReactJS application that runs the LogAn analyzer on Windows. It uses HTTP/FastAPI for front-end and back-end interaction and can download and analyze logs from the cloud. LogAn GUI was developed by students at Colorado School of Mines in Fall 2024.

Project Description

The LogAn GUI requires enhancements to its reporting and visualization features. The following updates are proposed for the GUI application.

Deliverable 1: LogAn GUI Tables and Plotting

Building on current prototype work, the objective is to improve the GUI to support tables and plots at a professional standard.

Table Functionality:

- Optimize visual space usage within tables to manage verbose LogAn data, promoting efficient use of whitespace and text for usability.
- Provide users with customizable filters for highlighting text and numeric data using text and background colors, with options to save these settings.
- Enable saving of user preferences for column selection and filter configurations.
- Implement column and row filtering similar to Excel, including formula-based selections with selectable and clear all options.

Plotting Functionality:

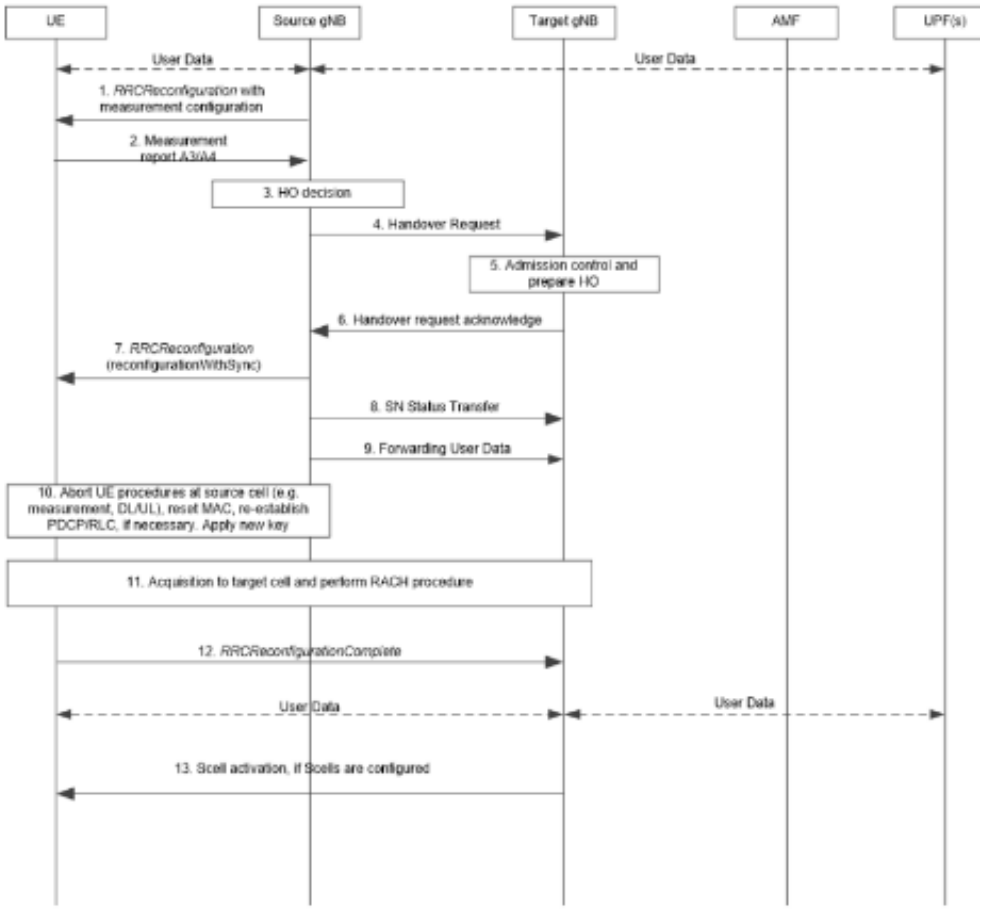
- Enhance readability by scaling x and y labels appropriately.
- Incorporate zooming capabilities and allow coordinated zooming across multiple plot instances.
- Add statistical plot types such as histograms and box-and-whisker plots.

The team will collaborate with Qualcomm leads and relevant stakeholders to evaluate graphing and table display libraries and develop design strategies that enable a scalable user experience.

Deliverable 2: Call sequence visualization

- Users have requested the capability to visualize call sequences within the GUI, enabling comparison between actual and expected call flows. The new feature should provide a clear visual representation of these sequences and facilitate easy comparison against defined expectations.

1.1.1. Call Flow



- Development requirements include:
    - Back-end API enhancements to clearly communicate both the anticipated state and the observed state as analysed by the LogAn tool.
    - Front-end development to effectively visualize and present this data in an intuitive format consistent with the specified requirements.
    - We'd like the front end to be able to overlay the actual callflow as a visualization over the expected callflow – such that users can efficiently determine the failure point or deltas

The product is designed for operation on Windows 11, with development activities carried out in a Windows environment. Code repositories are managed on an internal GitHub server.

All development processes should conform to established best practices, including the implementation of unit tests, adherence to industry coding standards, and appropriate internal packaging methods. Qualcomm employs Gen AI tools to support code generation, unit testing, and documentation, thereby enhancing developer productivity. Development cycles are coordinated with the Qualcomm team using software sprints, and progress is systematically tracked through JIRA.

The selected team will engage in close collaboration with Qualcomm engineers based in the Boulder office as well as at other locations. The team will review all relevant documentation to gain a comprehensive understanding of the tool, its features, and APIs. To provide effective support, the team will familiarize themselves with historical procedures for log downloads and analysis, utilizing these insights to enhance tool functionality in line with Qualcomm's directives. Coordination with Qualcomm engineers is required to define the project's scope, deliverables, and timeline.

Additionally, the team must submit a High-Level Design document detailing their planned approach prior to commencing development. Recommendations for suitable technologies should be included to ensure optimal project execution. The completed product must be fully featured and ready for immediate deployment by Qualcomm personnel.

Desired Skill Set

- UX/UI Design Experience
- Python3 and TypeScript programming experience.
- Electron, ReactJS, Pandas, FastAPI knowledge/experience or desire to learn.
    - Our data is substantial in size, so familiarity/excitement about working with big data sets will be beneficial.
- Familiarity with git and working in github environment.

Preferred Team Size

Initiating with three people is ideal. The availability of additional resources would allow us to broaden the project's reach.

Location

The Qualcomm TBS project will provide support mostly locally from the Qualcomm Boulder and Bridgewater NJ office. Remote work is the expectation with possibilities of on-site for key needs or discussions.

Resources

Qualcomm intends to open a remote development environment with sufficient compute and virtual desktop portables for all required development.

Contact

Email Kevin Wolver at kwolver@qti.qualcomm.com for high level project questions. If selected, other engineers will be made available for guidance, leadership, and mentoring.

Background

TBS = Test Base Station