# Puerto Rico Water Quality Assessment Application – Developers Guide

Contents

[Puerto Rico Water Quality Assessment Application – Developers Guide 1](#_Toc191376262)

[Introduction 2](#_Toc191376263)

[Purpose 2](#_Toc191376264)

[Primary Goal 2](#_Toc191376265)

[Target Audience 2](#_Toc191376266)

[Specific Objectives 2](#_Toc191376267)

[Overview 2](#_Toc191376268)

[Data Sources 2](#_Toc191376269)

[Workflow 3](#_Toc191376270)

[General 3](#_Toc191376271)

[UI Script 3](#_Toc191376272)

[Server Script 4](#_Toc191376273)

[Additional Resources 5](#_Toc191376274)

# Introduction

This document serves as a developers guide for the Puerto Rico Water Quality Assessment App, an R Shiny application designed to assist the Puerto Rico Department of Natural and Environmental Resources (DNER) in evaluating water body compliance with water quality standards and creating the Clean Water Act (CWA) 303(d) list. This guide provides a detailed overview of the application's structure, functionality, and development considerations.

# Purpose

## Primary Goal

To provide a tool for DNER to assess water body compliance with water quality standards.

## Target Audience

PR DNER personnel

## Specific Objectives

Retrieve and integrate water quality data from external sources. Provide a user-friendly interface for data exploration and analysis. Facilitate the identification of impaired waters.

# Overview

The application enables users to:

* Select a territory (currently Puerto Rico).
* Define a date range for water quality data retrieval.
* Review water quality standards (WQS) relevant to the assessment.
* Visualize and analyze water quality data compared to WQS.
* Generate plots of water quality data over time.

# Data Sources

* [Water Quality Portal](https://www.waterqualitydata.us/)
* [Office of Science and Technology Criteria Search Tool](https://www.epa.gov/wqs-tech/state-specific-water-quality-standards-effective-under-clean-water-act-cwa)
* [Assessment Total Maximum Daily Load (TMDL) Tracking and Implementation System (ATTAINS)](https://www.epa.gov/waterdata/upload-data-resources-registered-attains-users)

# Workflow

## General

1. **User Input:** Territory and date range selection.
2. **"Get Started" Click:** Triggers data retrieval and UI generation.
3. **Data Retrieval:** dataRetrieval::readWQPdata() fetches data.
4. **Data Processing:** Data is filtered, transformed, and plotted.
5. **Output Generation:** Data tables and plots are rendered.
6. **UI Update:** Reactive outputs are displayed in the tab panels.

**Key Functions:**

* ass.plot.f(): Generates ggplot objects, comparing water quality data to water quality standards.
* dataRetrieval::readWQPdata(): Retrieves water quality data.

## UI Script

* **Structure:**
  + Header: Includes EPA branding and navigation.
  + Collapsible panels: Provide overview and instructions.
  + Input controls: Territory selection (selectInput), date range selection (dateRangeInput).
  + Dynamic UI: "Get Started" button (uiOutput), tab panels (uiOutput).
  + Loading spinner: add\_busy\_spinner.
  + Footer.
* **Key Components:**
  + shinytheme("yeti"): Applies a clean theme.
  + HTML(): inserts raw html for the header and footer.
  + bsCollapse(): Creates collapsible panels.
  + selectInput(): Allows users to select the territory.
  + dateRangeInput(): Allows users to select a date range.
  + uiOutput(): Dynamically generates UI elements.
  + add\_busy\_spinner(): displays a loading symbol.
  + useShinyFeedback(): enables visual feedback.
* **Styling:**
  + style.css: External stylesheet for custom styling.
  + Inline CSS: For font styling.
  + EPA standard header.
* **Functionality:**
  + Provides input controls for data selection.
  + Dynamically generates the main application interface.

## Server Script

* **Structure:**
  + shinyServer(function(input, output) {...}): Defines the server logic.
  + Reactive expressions: reactive().
  + Event observers: observeEvent().
  + Output rendering: renderUI(), renderDataTable(), renderPlot().
* **Key Components:**
  + observeEvent(input$nextBtn, {...}): Triggers the creation of tab panels when the "Get Started" button is clicked.
  + renderUI(): Generates the tab panel UI.
  + renderDataTable(): Displays data in interactive tables.
  + renderPlot(): Generates plots of water quality data.
  + ass.plot.f(): A function that generates ggplot objects.
  + dataRetrieval::readWQPdata(): Retrieves water quality data from the Water Quality Portal.
* **Functionality:**
  + Retrieves water quality data from the Water Quality Portal.
  + Generates data tables and plots.
  + Dynamically updates the UI based on user inputs.
  + Generates plots that compare water quality data to water quality standards.
* **Data Handling:**
  + Retrieves data using dataRetrieval::readWQPdata().
  + Processes data using tidyverse functions.
  + Generates plots using ggplot2.
  + Filters data based on user defined date ranges.

# Additional Resources