

Revolutionize Clinical Trial Data Exploration with teal

Chendi Liao, Roche, Mississauga, Canada
Dony Unardi, Genentech, South San Francisco, USA

ABSTRACT

teal has evolved significantly over the past 6 years, from a small proof of concept to an industry-recognized open-source R-shiny framework that transforms the journey of clinical data exploration. Following our introduction at PHUSE US Connect 2023, we are thrilled to unveil the revamped data filtering panel that propels *teal* to greater heights. A suite of new features, including hierarchical filtering, module-specific filters, algorithm-based filter settings, reset and restore filter configurations, further pushes the boundaries of interactive and customizable clinical data exploration and reporting. *teal* also extends beyond organizational boundaries. This scalable and extensible product has fostered collaborative development across the pharmaceutical data science community. We will share how our partnership with other companies in the industry has cultivated more valuable extensions and further broadens the impactful reach of *teal*. Join us on this evolving journey, unite creative minds, to collectively achieve the vision of accelerating clinical insights.

INTRODUCTION

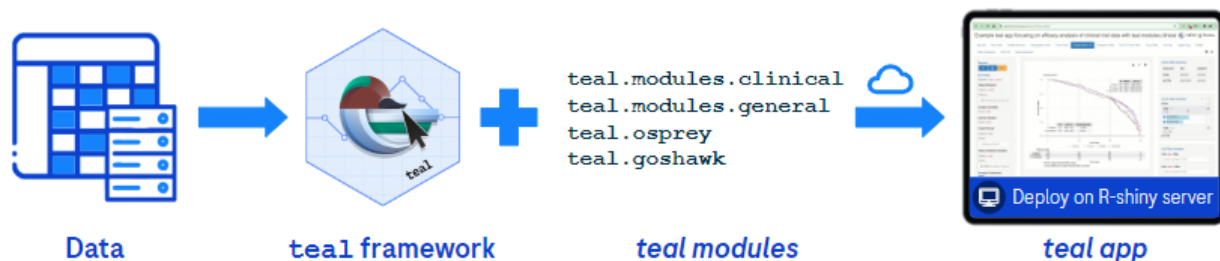
The demand for innovative and efficient data exploration solutions in the field of clinical trial data analysis has grown substantially in recent years. Traditional analytical approaches, that often involve time-consuming and repetitive tasks, are no longer sustainable for data scientists. In response to the needs for rapid insights generation and diverse ways of examining data, data scientists and software developers together at Roche/Genentech created an R-shiny based framework called *teal*, aiming to drive the advancement of interactive data exploration and visualization tools in the clinical trial space. It has been open-sourced to the public on GitHub since summer of 2022, and has just recently made its first official versioned release on CRAN in February 2024.

The *teal* framework is a modularized and scalable solution that streamlines the creation of interactive analytical web applications, known as *teal apps*, enabling data scientists and non-data scientists alike, to perform reproducible data exploration with ease. *teal* is designed to enhance data exploration efficiency from two perspectives without compromising on quality. For data scientists who are tasked with performing analysis, a large volume of analyses can be turned into elegant and powerful web applications within a matter of minutes, without requiring HTML, CSS, JavaScript or even R-shiny knowledge. On the other hand, for app users who are exploring and making interpretations on the data, new analysis and visualization can be created within seconds by adjusting a few user-interface (UI) components. By bringing data closer and faster to stakeholders, *teal* helps to drastically accelerate insights generation, ultimately leading to quicker and better-informed decisions.

TEAL FRAMEWORK OVERVIEW

DESIGN CONCEPT

Leveraging the concept of shiny modules [1], the design principle of *teal* framework centers around reusable and stackable building blocks, known as *teal modules*. Each *teal module* contains a single or a group of statistical analysis components in the form of data visualization, tabular summary and/or statistical model. *teal* framework serves as the platform where these *teal modules* integrates together, by providing a prebuilt R-shiny app shell that abstracts away the complexities of building an R-shiny app, such as configuring UI components and layouts, or mastering R-shiny namespace and reactivities. Out-of-the-box, it automatically embeds key mechanisms like dynamic data filtering, reproducible code generation, and analysis results reporting engine into the allowing app developers to focus on developing and configuring the appropriate *teal modules* for their statistical analysis needs.

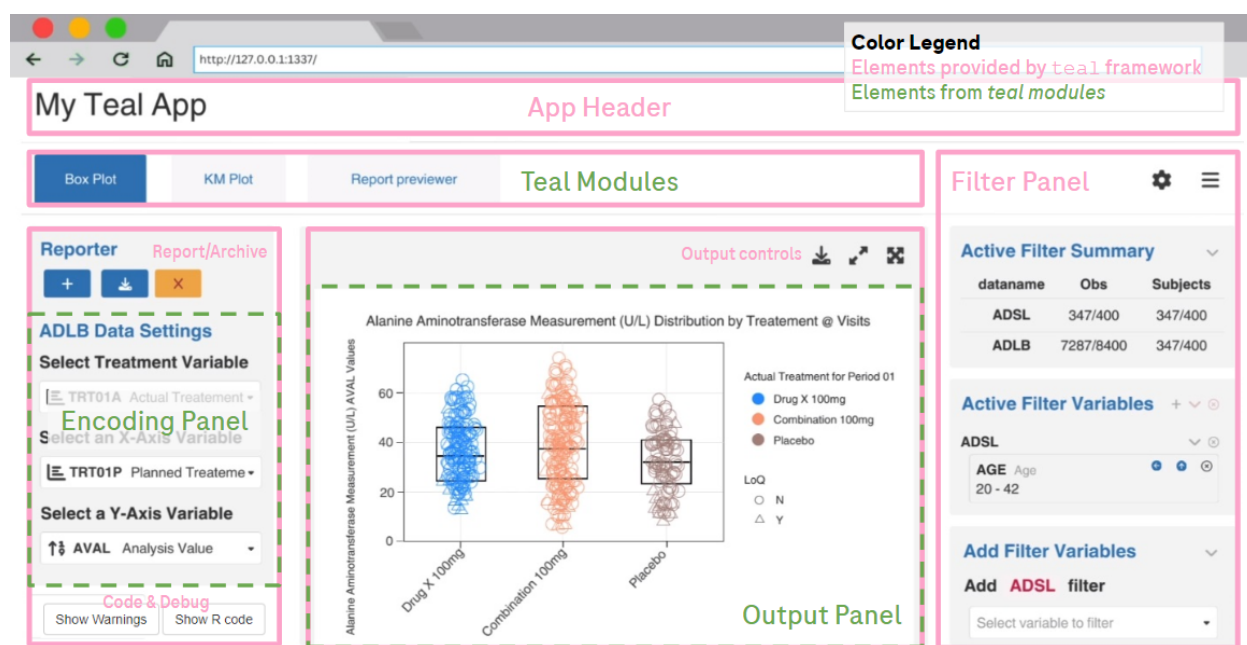


For users, the `teal` workflow starts with gathering and preprocessing the data into an analysis-ready shape. In a typical clinical trial setting, these data are most likely in the form of ADaM or SDTM structures. Once the desired *teal modules* have been selected for each or a combination of datasets, it becomes a matter of inserting them piece-by-piece into the `teal` framework to create a fully functional R-shiny app, known as *teal app*. Subsequently, *teal apps* can be published onto a R-shiny web server, accessible via an URL, and easily shared with stakeholders.

Overall, `teal` framework set a customizable but robust foundation that enables a modular approach to R-shiny app development, making it datatype agnostic, highly flexible, and facilitates collaboration and crowd-sourcing as an open-source software.

ANATOMY OF TEAL APP

Regardless of the amount or scope of the analysis contents, valuable functionalities that are important for traceability and reproducibility of interactive data exploration come automatically embedded in any *teal apps* that are created using the `teal` framework. To visualize more clearly the streamlined workflow that `teal` can bring to the development of data exploration tools, let's take a quick tour into the anatomy of *teal app*.



Referring to the figure above, a typical *teal app* consist of the following distinct areas:

- **App Header:** Shows the title of the web app
- **Teal Modules:** Offer each statistical analysis, data visualization and/or tabular summarization as separate tabs that users can toggle between. Distinct teal module tabs are associated with a different pair of encoding panel and analysis outputs.
- **Filter Panel** (right panel): A set of UI components that allow users to filter or subset the input data before being passed into the analysis modules.
- **Encoding Panel** (left panel): Various UI components that allow users to set or change the options and parameterization that goes into a data analysis, as defined by *teal modules*.
 - **Reporter:** Allow users to capture multiple outputs and analyses into a downloadable report for archival purposes, along with reproducible code used to generate the report.
 - **Show Warnings:** Verbatim printout of warnings/errors/message generated during the analysis run, this can be a useful debugging tool when encountering issues while developing the *teal app*.
 - **Show R Code:** Provides R code needed to reproduce the analysis/output shown in the Output Panel in a completely new R session.
- **Output Panel** (center panel): Where analysis results and visualizations are displayed. Results/outputs, that are defined by *teal modules*, are updated dynamically when changes are made in the Filter Panel or Encoding Panel.
 - **Output controls:** Set of widgets allowing users to download outputs, zoom into outputs (widen to screen width), or adjust dimension of graphical display within the Output Panel.

By providing an insertable and customizable shell that also comes with a set of consistent layout and functionality, the `teal` framework allows data scientists to bypass the mountain-worth of efforts needed to create them from scratch, and instead, devote their efforts on building up the analysis contents within each *teal module*.

TEAL MODULES

Although not considered core part of the `teal` framework itself, *teal modules* are a critical element of the success of `teal`. Any R-savvy data scientist will be able to develop a `teal module` for their own analysis needs, but the open availability of a more diverse and robust set of *teal modules* will further expand the application and impactful reach of `teal` into more data scientist communities.

With this in mind, the `teal` developer team at Roche/Genentech also created and open-sourced a large catalog of over 50 centrally curated and maintained *teal modules* for the analysis and visualization of clinical trial data. These plug-in-ready *teal modules* are organized into a collection of R packages, each with a focused theme.

- [teal.modules.clinical](#) - *teal modules* specializing in analyzing ADaM/CDISC data, as well as clinical trial reporting of safety, efficacy and patient profile data.
- [teal.modules.general](#) - *teal modules* for the visualization and exploration of any general data types.
- [teal.osprey](#) - *teal modules* specifically focusing on the analysis, visualization and reporting of early-phase clinical trial data with CDISC structure.
- [teal.goshawk](#) - *teal modules* specifically designed for the analysis, visualization and reporting of biomarker and longitudinal clinical trial data with CDISC structure.

These *teal modules* R packages are currently publicly available via GitHub. For the first two packages, we also plan to make their first versioned release on CRAN by the end of Q1 2024.

NEW DEVELOPMENTS IN TEAL

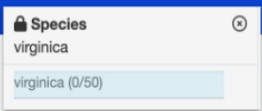
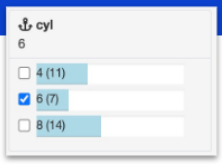
Since its inception 6 years ago, `teal` has evolved from a small proof of concept project to a well-established product that's recognized by the R community within data scientists working in the pharmaceutical industry. We formally introduced `teal` to the community of clinical data scientists at PHUSE US Connect 2023 last year. Nonetheless, we constantly strive to improve all aspects of the product, in our continued pursuit for innovation and trailblazing in the area of clinical data exploration. Since the previous conference, we've implemented several enhancements and added new features to `teal` framework, and also partnered with data scientists from other companies in the pharmaceutical industry to jointly develop valuable features and extensions of `teal`. The following sections highlight some of the key developments in `teal` over the past year.

REDESIGNED FILTER PANEL

The Filter Panel is an essential component and one of the most beloved functionality of the `teal` framework, empowering users to subset their data during exploration. Driven by user feedback, we've redesigned this important feature by enhancing its customization capabilities, expanding flexibility in its usage, and adding new features and new UIs to make data exploration with `teal` even more smooth and convenient.

EXPANDED FILTER TYPES AND MODULE-SPECIFIC CONFIGURATION

Flexibility in data exploration is a must, however, statistical vigilance should not be overlooked, particularly in the clinical trial space. In addition to the original interactive filters, where *teal app* users can create new filters on-the-fly, we added new filter types to the `teal` framework, giving some additional control to the clinical data scientist.

Fixed filter	Anchored filter	Fixed & Anchored	Custom expression
 <pre>teal_slice(dataname = "iris", varname = "Species", id = "filter1", selected = "virginica", fixed = TRUE)</pre>	 <pre>teal_slice(dataname = "mtcar", varname = "cyl", id = "filter2", selected = 6, anchored = TRUE)</pre>	 <pre>teal_slice(dataname = "mtcar", varname = "gear", id = "filter3", selected = c(3,5), anchored = TRUE, fixed = TRUE)</pre>	 <pre>teal_slice(dataname = "iris", id = "custom", title = "Custom Expression", expr = "Species %in% c('virginica', 'setosa') & Sepal.Length > 1")</pre>

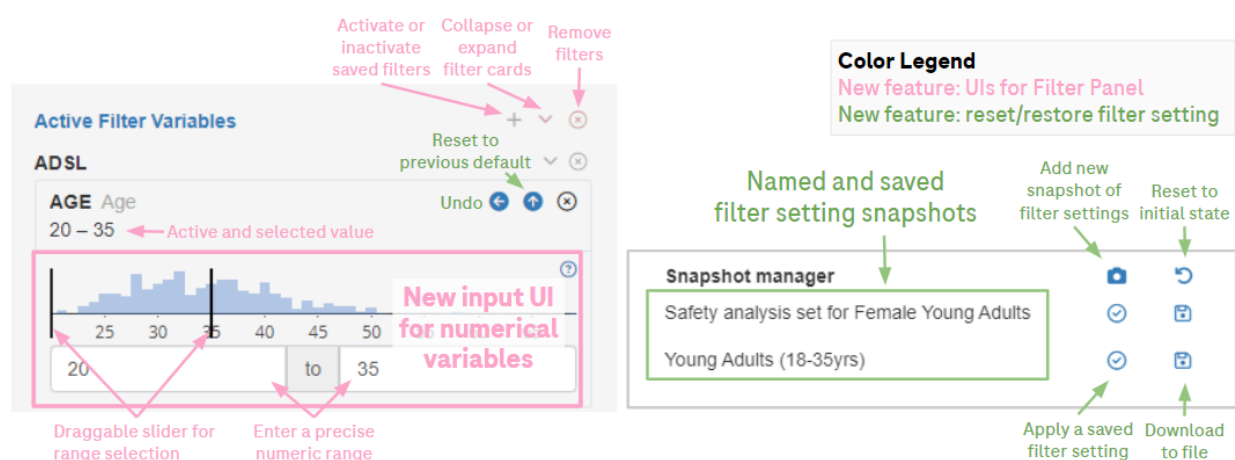
As shown in the chart above, with just a few parameter adjustments using the `teal_slice()` function, these filter types can be set as defaults during *teal app* configuration steps based on the analysis need.

- **Fixed filter:** Filters where filter conditions (selected value of the filter) are fixed, unchangeable and non-interactive, they can only be inactivated or removed from the associated analysis.
- **Anchored filter:** These filters are still interactive, where the filter condition is modifiable by the *teal app* user, however, they cannot be removed from the specified analysis, thus will always be activated and applied.
- **Fixed and anchored filter:** Combining the properties of the two filter types above, these filter conditions cannot be modified nor removed. Nonetheless, they will be displayed in the filter panel to inform the teal app user that they are being applied to the current analysis.
- **Filters with custom expression:** For filter conditions that are more complex, consisting of more involved logic or multiple dataset variables, allowing teal users to preset the filter condition through code expressions.

Moreover, these filter types can be applied to analysis in a module-specific approach, where the filter condition won't be applied to all modules within a *teal app*, but rather, can differentiate from one to the next. This is especially useful for *teal apps* built for clinical trial data that require multi-faceted analysis and exploration in safety, efficacy, and/or exploratory topics, where a different set of filters or subpopulation is only relevant for a certain analysis. While creating the *teal app*, teal users can now apply different filter types or filter configurations as default between each or each set of *teal modules* based on the analysis content. As an example in clinical trial setting, data scientists creating the *teal app* may choose to add a fixed and anchored filter of "safety-evaluable population" for all safety analysis-related *teal modules*, while applying a fixed and anchored filter of "intention-to-treat population" for *teal modules* designed to analyze efficacy data.

NEW FILTER PANEL UIs

As UI/UX designs are crucial to any interactive tools, we've made several enhancements to help elevate the user experience. For example, the range selector UI in older teal versions was clunky and difficult to perform fine adjustments on the selected values. In the most current version of teal, we offer a new range selector input UI for numerical variables, by combining the convenience of a draggable slider with the precise control of a numeric input box – the best of both worlds (shown in below figure on the left panel). Other examples of UI enhancements include the addition of an information banner that displays the filter condition beneath the name and label of each filter, a toggle button to collapse or expand the available filter cards, and a mechanism to activate or inactivate pre-specified filters without removing them from analysis.



RESET AND RESTORE FILTER SETTING

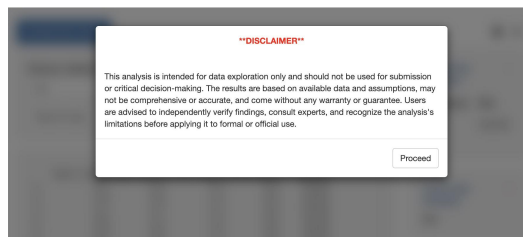
Data exploration usually is an iterative process that often requires the explorer to navigate back and forth between a diverse set of analysis configurations. We considered the typical user workflow within a *teal app*, and designed new features to further improve the user experience of data exploration with teal. For each filter, users can now easily undo a change they've made to filter condition or reset to default filter condition, now with just a few clicks, greatly boosts the efficiency of the exploration process.

Lastly, we introduced a feature called 'Snapshot Manager' (shown in above figure on the right panel), which enables *teal app* users to save and name one or more sets of filter settings that they created during the interactive app session. Furthermore, this feature supports *teal app* users to restore a saved filter settings and reapply to the data or analysis, download filter settings to file and share with other users who are using the same *teal app* for data exploration. This feature greatly enhances collaboration and facilitates knowledge sharing among *teal app* users, such as various stakeholders within a clinical trial study.

NEW FEATURE: LANDING POPUP

Due to its quick yet flexible interactivity, *teal* can empower stakeholders to self-serve and create exploratory analysis spontaneously. On the other hand, there are also concerns about what is the best approach to ensuring statistical validity and responsible use of the insights generated from *teal apps*. While the responsibility and decision of gatekeeping ultimately resides with data scientists who share the *teal app*, as product developers, we also implemented a new feature in the form of displaying a popup window with customizable messages via the `landing_popup_module()` function. This will allow the creator of the *teal app* to embed any welcome messages and/or disclaimer statements as the first thing any app users will see when accessing the app.

```
teal::landing_popup_module(  
  title = tags$b("Welcome", style =  
    "color:red;")  
  content = "A place for welcome messages  
or disclaimer statement.",  
  buttons = modalButton("Proceed")  
)
```

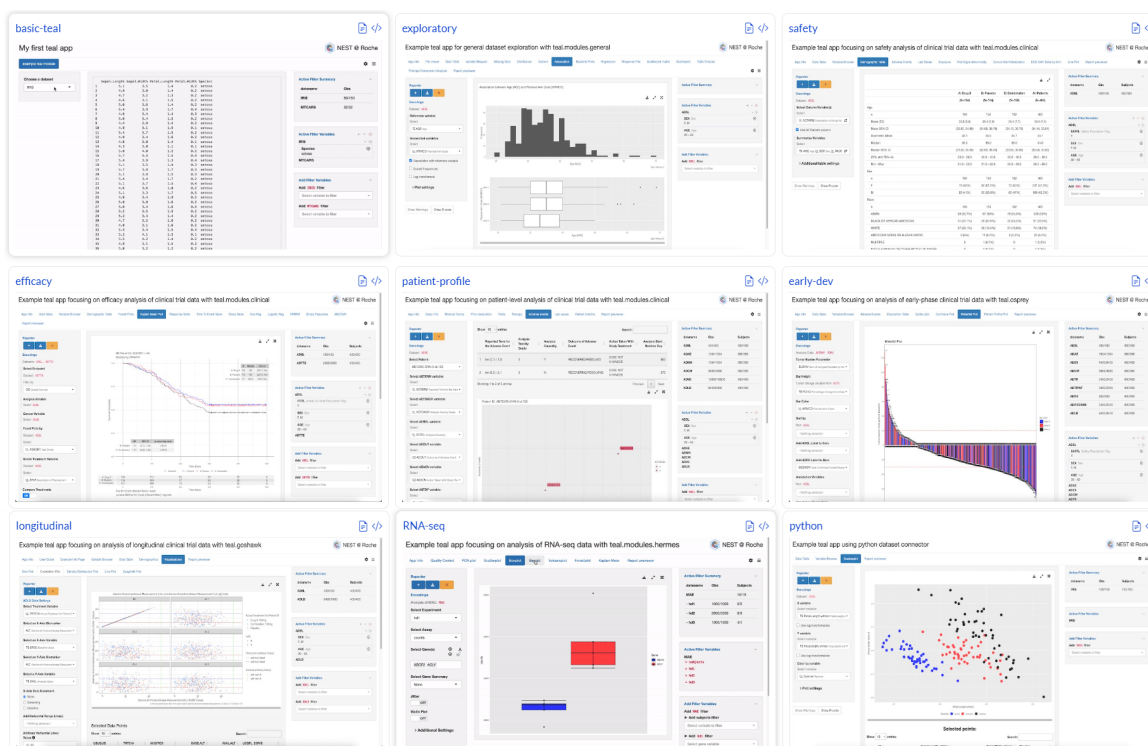


Above left is an example code for the new popup feature, where the `title` and `content` arguments of the function are fully compatible with any HTML-modifiers via `shiny::tags` options [2]. Above right is an example of how the landing popup mechanism is displayed upon opening the *teal app*. Contents of the main *teal app* are blurred out and won't be accessible to the app user until the landing popup is consciously closed by the app user.

REVAMPED TEAL.GALLERY

[teal.gallery](#), as the name suggests, is a gallery of themed sample *teal apps* created with the *teal* framework and the large catalog of over 50 centrally curated and maintained *teal modules* with analysis templates. It was designed not only to showcase the large catalog of plug-in-ready modules, but also provide the corresponding R codes for each example app, which can serve as template codes for *teal* users working with clinical trial data.

Previously, in order to see and play around with these sample apps, users have to first install the *teal.gallery* R package then interact with it through the R console, which is inconvenient and not very user-friendly for new R or R-shiny users. We recognized this shortcoming, and revamped it into a true gallery-like repository where the animated demo, the corresponding R code, and a URL linking to the live sample app are displayed side-by-side. Users, regardless of their R skills or their familiarity with *teal*, are able to view and explore the sample apps at their own leisure. We also incorporated a robust continuous integration and continuous deployment (CI/CD) pipeline, that will automate the deployment of the sample apps using the latest stable version of *teal* framework and *teal module* packages, making the whole process seamless and always up-to-date.



TUTORIAL & GUIDES

While `teal` is an effective tool for data exploration and insights acceleration, we acknowledge that it is also sophisticated, seemingly complex for new users, and can be rather challenging for any users wanting to develop their custom *teal modules* or contribute directly to `teal` framework development. Therefore, it is imperative that guides and documentations are clear, concise, and comprehensive to facilitate effective understanding and implementation.

Over the past year, we've improved on existing documentation and created many new ones for technical-oriented `teal` users with strong R or R-shiny skills. Articles such as the *Technical Blueprint*, which outlines the fundamentals principles of `teal` framework or migration guides that offer instructions to up-version existing *teal apps* with the newest `teal` release, will provide important information for easy adoption for any prospective `teal` users.

Looking towards the rest of 2024, we aim to complete even more training materials for all levels of `teal` users, including the *teal app* users, who may not even be data scientists or familiar with data or code. We will be creating tutorial videos and user guides for both `teal app` users and app developers alike. There will also be a new upcoming Coursera course titled "*Hands On Clinical Reporting Using R*", offered by Genentech, with one of the chapters dedicated to applying `teal` in a clinical reporting setting. We also warmly welcome any feedback and suggestions of documents or materials that `teal` users may find useful.

CROSS-INDUSTRY COLLABORATIONS

Roche/Genentech is spearheading a shift towards open-source backbone in our next generation of data science platform. One major benefit of open-source software is the ability to collaborate with external partners, without worrying about the implications and barriers of proprietary or competitive intelligence.

In the past year, we engaged in many fruitful discussions and partnered with data scientists from four other pharmaceutical companies to cultivate new ideas on the product roadmap. To give a flavor of these exciting developments, these include new features enhancements such as addition of customizable data processing modules, or new `teal` framework extensions such as a marketplace that enable `teal` users to pick out *teal modules* into a "shopping cart" that subsequently generates a fully functional `teal app`. Although most of these projects are still in their infancy, nevertheless, we are enthusiastic about the prospect of productive collaborative development that will further broaden the impactful reach of `teal`.

CONCLUSION

`teal` is a true recipe for quick insights generation and continues to transform the way we conduct clinical trial data exploration. Here at Roche/Genentech, our `teal` development team continues to strive for improvements and innovation, to further push the boundaries of interactive and customizable data analysis and reporting. We are thrilled to share this innovative and transformative product with the world by open-sourcing it and releasing it on CRAN, and we have many more exciting plans in 2024 and beyond. Join us on this evolving journey, unite creative minds, to collectively achieve the vision of accelerating clinical insights across the entire industry.

REFERENCES

- [1] Wickham, Hadley. Mastering Shiny, Chapter 19: Shiny modules. Available from: <https://mastering-shiny.org/scaling-modules.html>
- [2] Grolemond, Garrett. Shiny HTML Tags Glossary. Available from: <https://shiny.posit.co/r/articles/build/tag-glossary/>

ACKNOWLEDGMENTS

We would like to thank all past and present developers that contributed to any of the `teal` framework R packages.

RECOMMENDED READING

- PHUSE US Connect 2023 Presentation: Enabling Comprehensive Interactive Data Analysis and Exploration in R with an Open-Sourced, Scalable Shiny Framework, [presentation material](#) and [recording](#)
- `teal` web manual: <https://insightengineering.github.io/teal>
- `teal` Github repository: <https://github.com/insightengineering/teal>
- `teal.gallery` demo apps: <https://insightengineering.github.io/teal.gallery/>
- NEST R product family website: <https://insightengineering.github.io/nest/>

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Author Name: Chendi Liao

Company: Roche

Email: chendi.liao@roche.com

Co-Author Name: Dony Unardi

Company: Genentech

Email: unardi.dony@gene.com

Brand and product names are trademarks of their respective companies.