QualtricsReports Documenation

learningforpurpose.org

27 February 2017

Overview

The Qualtrics Reporting App is designed to present survey participants with a customized report after they have completed a survey. Qualtrics will redirect them to the reporting application and wil include relevant results from their survey in a custom URL. This URL will then be parsed by the Shiny application and a custom report generated.

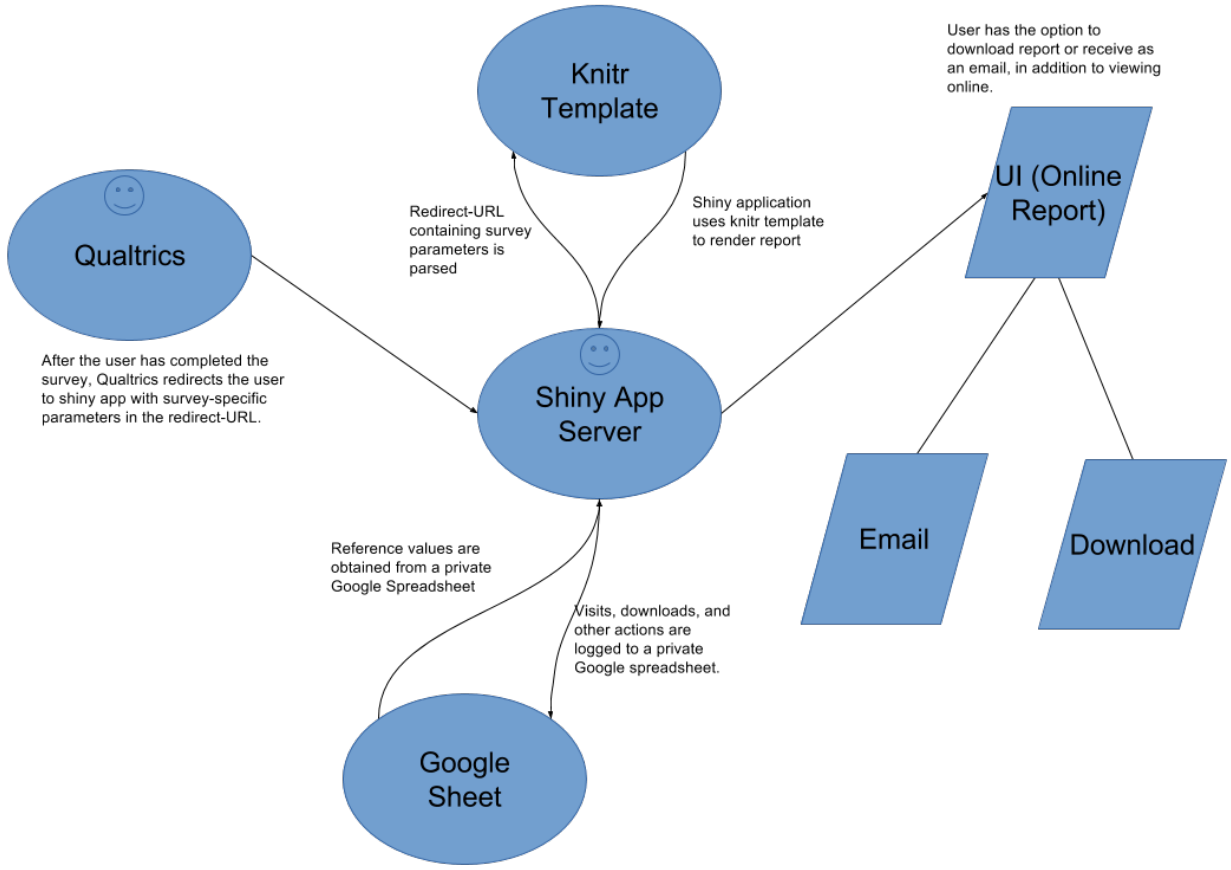

Figure 1: Qualtrics Reports Application Flow

Deployment

1. Make sure you have the most recent stable release from source control. You can do this by either:
    a. Downloading the code from the repository manually (here)
    b. Cloning the repository:

- In Windows use Git Bash, else use terminal
- `git pull origin master` while in the QualtricsReports folder on your local machine

2. Open QualtricsReports.Rproj from within the source code folder (should open in RStudio)
3. Create a copy of the file `settings_template.txt` and name it `settings.txt`. Edit `settings.txt` to reflect your desired settings and credentials. The settings are:
   - `email.username`: the username for the send-from email address when the user requests an emailed report (must be a gmail account, so this is the X from X@gmail.com)
   - `email.password`: the password for the send-from email address
   - `shiny.username`: your username for shinyapps.io
   - `shiny.token`: the token for your shinyapps.io account (https://www.shinyapps.io/admin/#/tokens)
   - `shiny.secret`: the secret for your shinyapps.io account token
   - `app.name`: the name that you want the application to have in the deployed URL (i.e. learningforpurpose.shinyapps.io/app.name
   - `log.spreadsheet.id`: the unique id for the google spreadsheet that will log visits to the application and hold the reference data
   - `offline.for.debugging`: whether to log visits to the application; keep this FALSE unless directed otherwise
4. Make sure that you have all of the relevant packages installed by running the following code. If a package is missing, you will get an error "there is no package called 'X' " . To solve this, use install.packages("X") in Rstudio before running the app.

```
install.packages("shiny")
install.packages("shinyjs")
install.packages("knitr")
install.packages("rmarkdown")
install.packages("mailR")
install.packages("googlesheets")
install.packages('rsconnect')
install.packages('httr')
install.packages('rematch')
```

5. Authenticate with google to enable logging by running each line of code below one-by-one and following the prompts in the Rstudio console (make sure you login with a google account that has write access to the logging google document)

```
library(googlesheets)
token <- gs_auth(cache = FALSE)
gd_token()
saveRDS(token, file = "googlesheets_token.rds")
```

6. Test that the application runs locally on your machine by opening server.R and pressing the "Run App" button in the top right corner of the code window. Depending on your settings it will open in a pop-up window, an embedded Rstudio window, or a browser tab. When you are finished, or if you receive an error, remember to press the stop sign to stop the application from running.
7. Deploy the application to ShinyApps.io by running the file `deployment.R`.

```
source("deployment.R")
```

The application can take some time to deploy. Once it has, a browser tab should open automatically pointing to learningforpurpose.shinyapps.io/QualtricsReports. This is the live application.

8. Adjust the application's settings by finding the app in https://www.shinyapps.io/admin/#/applications/all and navigating to the settings tab. The settings you shoud choose are:

- General:
  - `Instance Size`
    - Large (1GB)
    - Select larger instances if you require more memory for your application. Free and starter plans are limited to a maximum of 1GB (Large).
  - `Instance Idle Timeout`
    - 14 min
    - Time after which your idle application will be stopped. The minimum is 5 minutes.
- Advanced:
  - `Max Worker Processes`
    - 1
    - Number of worker processes that can be started in a single instance.
  - `Max Connections`
    - 50
    - Number of concurrent connections allowed per worker process.
  - `Worker Load Factor`
    - 5 %
    - Threshold percentage after which a new connection will trigger the addition of a new worker process (limited to the Max Worker Processes limit).
  - `Connection Timeout`
    - 900 sec
    - Seconds of inactivity before a browser connection to a worker process is considered to be idle and is closed.
  - `Read Timeout`
    - 3600 sec
    - Seconds of browser to worker inactivity after which the connection is considered to be idle and is closed. Use a value of 0 for non-interactive applications, e.g. a dashboard.
  - `Startup Timeout`
    - 300 sec
    - Time that an instance will wait for a worker to start. Increase this number if you load a lot of data or have a longer application startup time. The maximum is 300 seconds.
  - `Idle Timeout`
    - 60 sec
    - Wait time in seconds before an idle worker process with no connections is shut down. The value must be between 5 and 60 seconds.
  - `Instance Load Factor`
    - 50 %
    - Threshold percentage after which a new connection will trigger the addition of an application instance (limited to the Maximum Instance Limit, free tier is 1).
  - `Start Count`:
    - 1
    - Number of instances to bring up when an application starts. Pick a higher number if you know that your application receives flash traffic.
  - `Package Cache`
    - Off
    - Enable/Disable the use of package caching when your application is built. Caching allows for faster deploy times, disable it only if you suspect that the build is failing due to caching.

Setting Up Multiple Surveys

Adding a new survey to the application requires 3 steps:

1. Define the name of the survey and its parameters in the redirect-URL from Qualtrics (`survey=report2&param1=one&param2=two`)

2. Add any reference parameters to the Log Google Sheet's Reference tab
3. Create a new knitr template with the same name (case-sensitive) as in the redirect-URL (`report2.Rmd`) and place it in the `report_templates` folder
4. Redeploy the application by sourcing the deployment script (`source("deployment.R")`)

The Shiny application will automatically render the correct report based on the "survey" parameter contained in the Qualtrics redirect-URL.

Using External Data

You should only reference survey-specific external data in the relevant post-survey report template (the knitr template). For instance, if the survey were assessing IQ and we wanted to show the user their IQ relative to the average, then we would include a variable for the average IQ in the report template and update it regularly.

To refer to a dynamically updated reference value, add it to the Log Google Sheet's Reference tab. The columns on that tab are defined as follows:

- Survey: same name as the "survey" variable in the redirect-URL and the name of the knitr template without the Rmd extension (e.g. ExampleReport.Rmd = ExampleReport)
- Parameter: the name of the parameter to be referenced. In the knitr template you will need to add a "ref." to this parameter to avoid overlapping names. It is okay to repeat variable names among different surveys because the application will only look for variables related to the survey it is trying to render (e.g. having IQ for SurveyA and IQ for SurveyB is okay)
- Value: the numeric value of the reference parameter

There is code in the "setup" (first) chunk of the sample template to demonstrate the functionality. After you have defined the variables and uploaded relevant data, you can refer to them as you would any other variable in a knitr document.