How to edit the code for the Shiny app

App-2 is the code for the app which is embedded in the Ecobat website – this is the one you will want to edit.

App-1 and App-3 are also published but these are not embedded on the Ecobat website and were developed for testing purposes.

The R Markdown script in App-3 works in RStudio with csv files downloaded from Ecobat but the R Markdown scripts in App-1 and App-2 will not. App-3 will not work with csv files when deployed to Shiny. Therefore, developments should be made in App-3 and once it is confirmed they work, migrated to App-2.

App-3’s R Markdown script contains ..ï before location\_name in all places ‘location\_name’ occurs in the script to accommodate the UTF-8 symbol in the CSV download which for some reason R Studio can’t cope with but Shiny can.

R Studio

RStudio needs to be installed. You also will need to install and load the following packages:

* shiny
* rmarkdown
* ggplot2
* knitr
* pander
* plyr
* tidyr
* rcompanion

RSConnect

To be able to publish apps to the Ecobat Shiny Apps account, you will need to connect the computer to the Ecobat Account. To do this, log into shinyapps.io in a browser using the log in details below.

Username: [info@ecobat.org.uk](mailto:info@ecobat.org.uk)

Password: Pipistrelle1 (with the capital P)

Then you can follow the instructions on this page to link your computer up to the Shiny Apps account www.shinyapps.io/admin/#/dashboard or follow these steps

1. Install the rsconnect R package – can be installed directly from CRAN.

To make sure you have the latest version run following code in your R console:

install.packages('rsconnect')

1. Authorise account

The rsconnect package must be authorized to your account using a token and secret. To do this, go to [www.shinyapps.io/admin/#/tokens](http://www.shinyapps.io/admin/#/tokens) and follow the instructions to copy and paste the code containing the token and secret into your console to authorize your computer. Once you've entered the command successfully in R, that computer is now authorized to deploy applications to your shinyapps.io account.

1. Once the rsconnect package has been configured you can deploy apps to the Ecobat account/update current ones.

Editing the code

In Dropbox, Shiny Apps – App 2 is what you’re after.

This should contain server.R, ui.R, EcobatScript.Rmd, and two folders entitled www (containing ecobat-logo.png) and rsconnect. This configuration of files ***must stay as is***, with the logo in the www folder, and the R and Rmd files in the main folder.

You can open server.R, ui.R and EcobatScript.Rmd in RStudio.

server.R and ui.R are responsible for the Shiny app itself, and these shouldn’t need to be edited much if at all, excepting some potential aesthetic changes down the line.

EcobatScript.Rmd is responsible for producing the report - this is what generates graphs, tables etc. If there is a problem with the Shiny app not generating the report properly (i.e. people are getting server errors etc) this is the likely culprit. Therefore, this is the file you are most likely to want to edit.

As per above, any changes that need to be made should first be made in App-3 to allow you to test it in RStudio, but then the changes should be migrated across to App-2. This is a bit of a pain but it’s something I haven’t been able to find a workaround for yet.

For editing purposes, you will also need to change how R Markdown reads in files – chunks 2 and 3. To be able to test files, remove the hashtag from one of the lines of read.csv code in chunk 2, and change the file path to the file you are testing with. Add a # to df<-params$n in chunk 3. When you finish editing and are deploying to Shinyapps.io, don’t forget to switch the hashtags back (remove the hashtag from chunk 3 and add one back in to chunk 2) to ensure it reads it’s files correctly.

Once you have finished editing it, make sure your server.R, ui.R and EcobatScript.Rmd are all saved, and then open either the server.R or ui.R and look for the little blue symbol in the top right hand corner of the R script, next to where it says ‘Run App’.

A close up of a device

Description generated with high confidence

This is the publish button. To publish, click the little down arrow to the right of it, and select the title of the app (Data Analysis – it should be ticked). Check everything is in order (all the correct files are checked) and click Publish. It might take a little bit of time to publish (though the first time you publish it takes longer, subsequent updates should be quicker as the packages have already been loaded.)

Embedding Apps in Drupal

If you edit App-2 (the one embedded on the Ecobat website), when you publish the updated version, the embedded app will update automatically without you having to change anything in the embedded app. However, if you wish to embed a new app, or need to change something, this is how to embed apps within Drupal:

On the page where you want to embed the app, click edit, switch to plain text editor, change the text format to full HTML, and insert the following:

<iframe height="800" src="INSERT URL HERE" width="800"></iframe>

Click save