Sharing Shiny Apps with Others Remotely

Sharing Apps to Run Locally

Once you've written your Shiny app, you can distribute it for others to run on their own computers—they can download and run Shiny apps with a single R command. This requires that they have R and Shiny installed on their computers.

If you want your Shiny app to be accessible over the web, so that users only need a web browser, see Deploying Shiny Apps over the Web.

Deploying Shiny Apps over Web

Once you've written your Shiny app, you can make it available to anyone who has a web browser, using our Shiny Server software. You can either host the applications on your own server, or let us host your Shiny applications for you.

Self-hosted Shiny Server

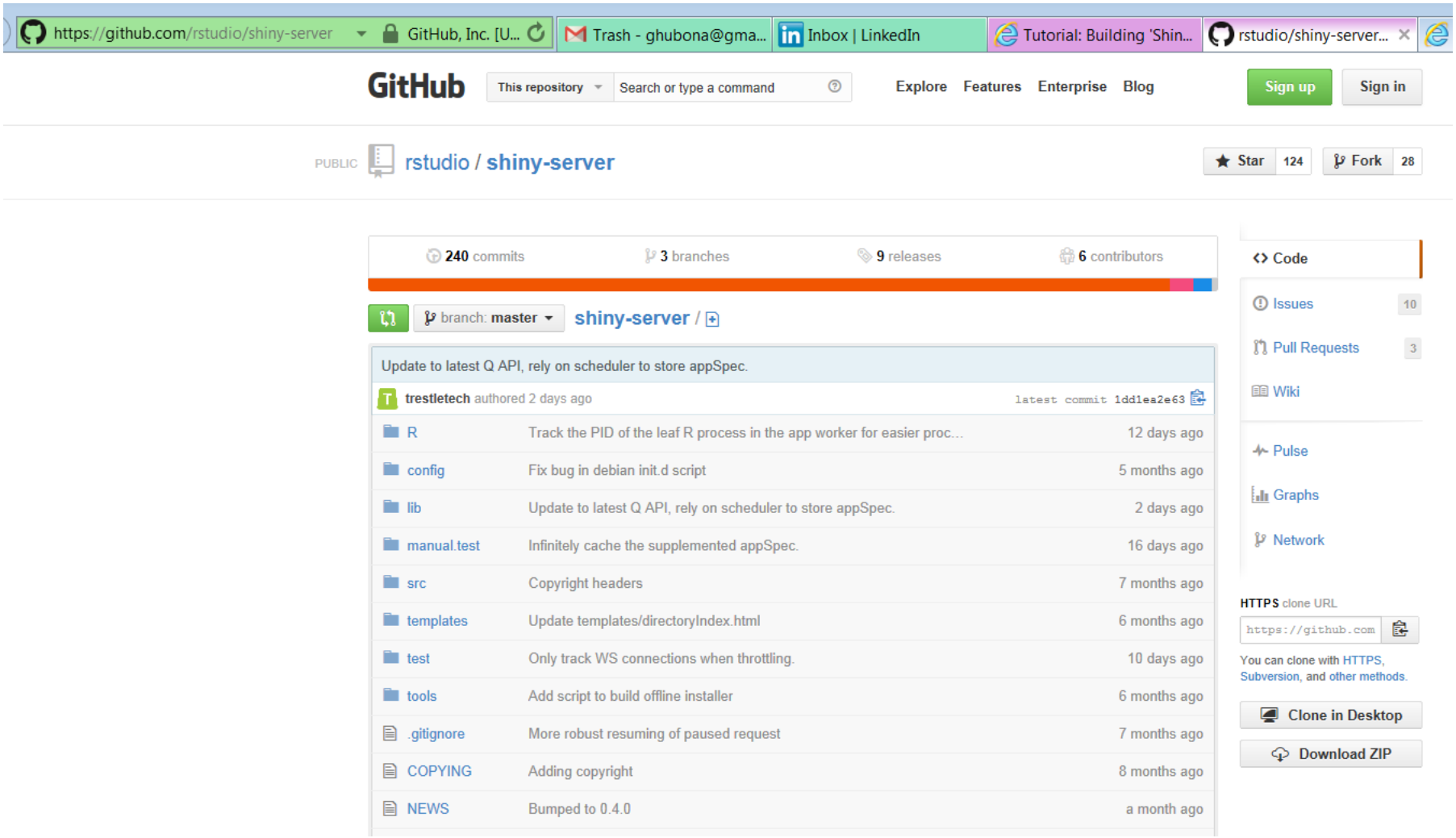
Next Slide . . .

With our Shiny Server software, you can deploy Shiny applications over the web so that users need only a web browser and your application's URL. You'll need a Linux server and Shiny Server.

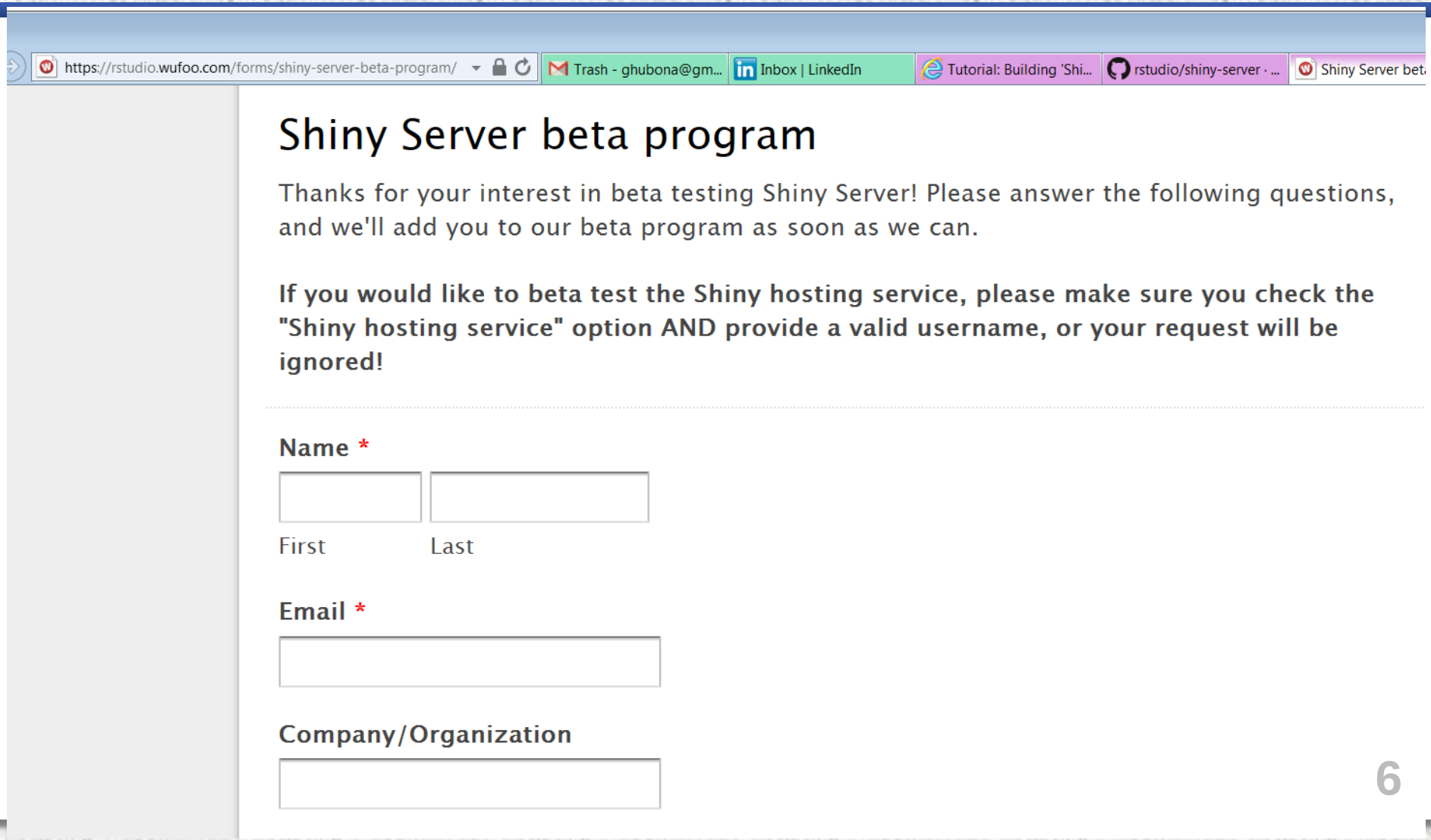
Shiny Server is free and open source, though in the future we will offer a commercially licensed edition with additional features for larger organizations. If you'd like to be notified of future beta releases of Shiny Server, please register now.

Two slides forward . . .

Shiny Server software



Register now link . . .



RStudio-hosted Shiny Server

Want to deploy over the web but prefer not to run your own server? We're currently beta testing a subscription-based hosting service for Shiny. To apply for a free beta test account, register now.

Same as previous slide . . .

Pros

- Easiest for your users—only a web browser is required
- No need to run your own server

Cons

- Code and data must be copied to our servers

More Ways to Deliver Shiny Apps to Run Locally

Gist

See next slide . . .

See two slides forward. . .

One easy way is to put your code on gist.github.com, a code pasteboard service from GitHub. Both server.R and ui.R must be included in the same gist, and you must use their proper filenames. See https://gist.github.com/3239667 for an example.

Your recipient must have R and the Shiny package installed, and then running the app is as easy as entering the following command:

```
shiny::runGist('3239667')
```

In place of `'3239667'` you will use your gist's ID; or, you can use the entire URL of the gist (e.g. `'https://gist.github.com/3239667'`).

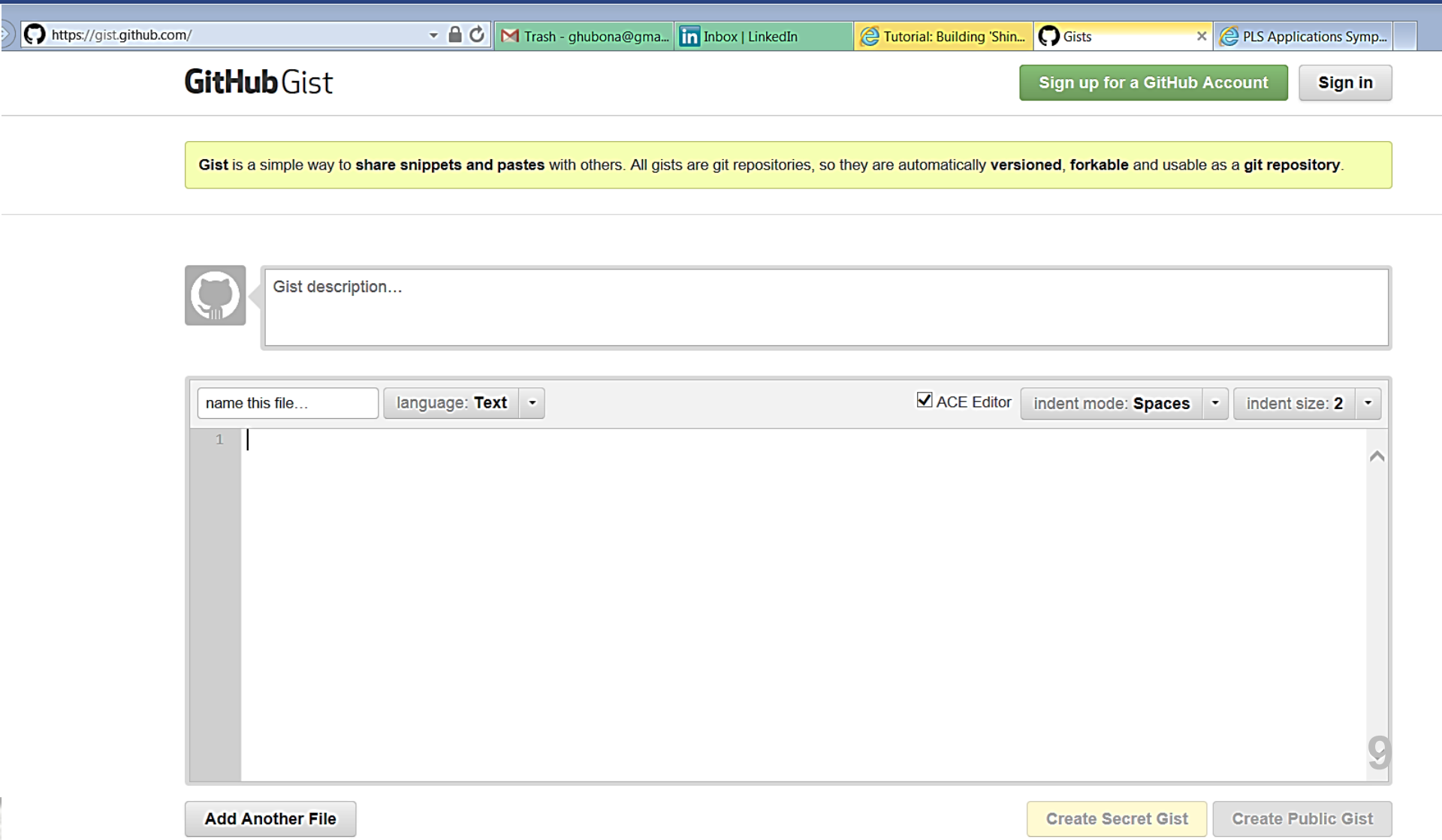
Pros

- Source code is easily visible by recipient (if desired)
- Easy to run (for R users)
- Easy to post and update

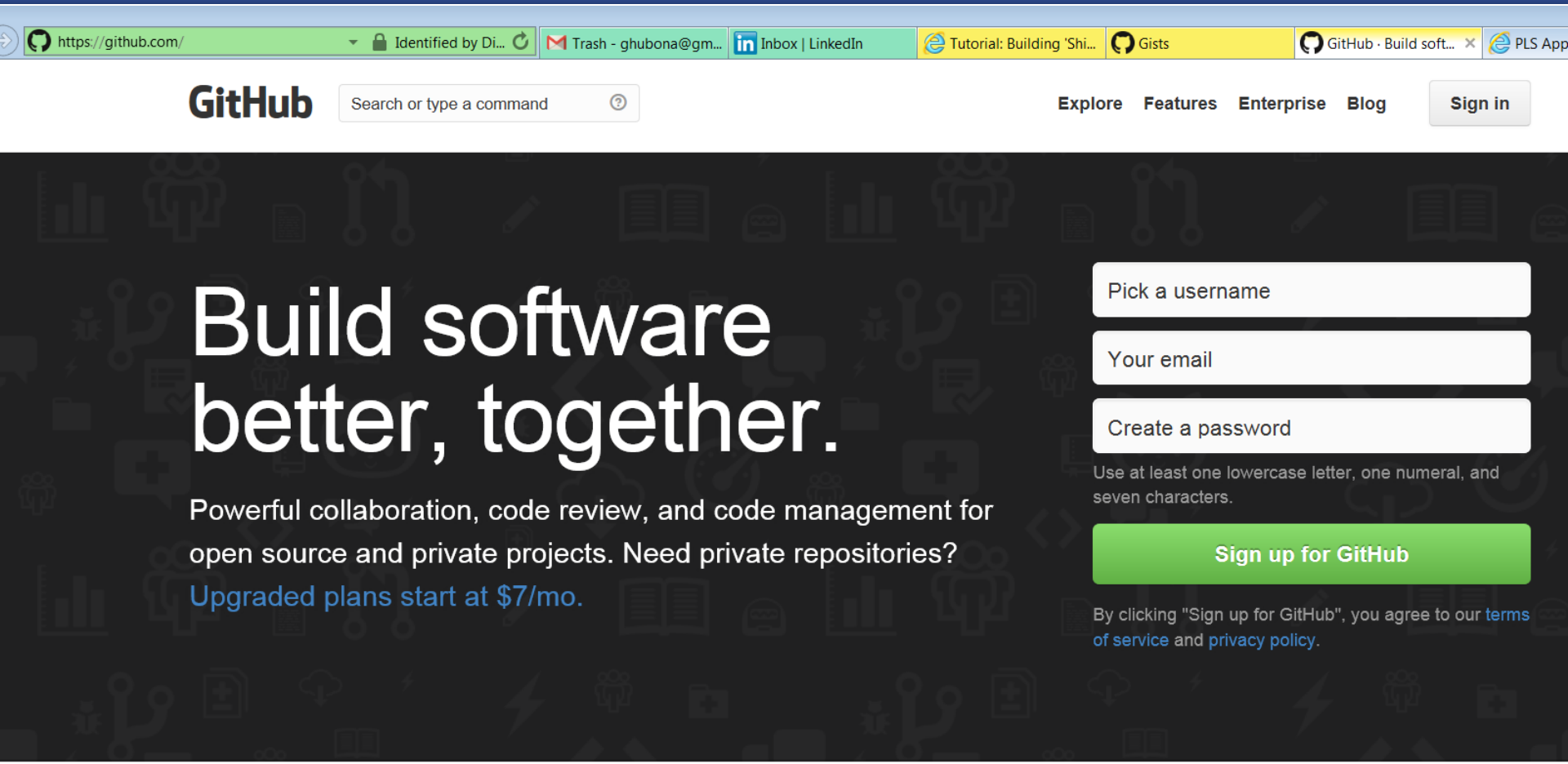
Cons

- Code is published to a third-party server

More Ways to Deliver Shiny Apps to Run Locally



More Ways to Deliver Shiny Apps to Run Locally



More Ways to Deliver Shiny Apps to Run Locally

GitHub repository

If your project is stored in a git repository on GitHub, then others can download and run your app directly. An example repository is at https://github.com/rstudio/shiny_example. The following command will download and run the application:

```
shiny::runGitHub('shiny_example', 'rstudio')
```

In this example, the GitHub account is `'rstudio'` and the repository is `'shiny_example'`; you will need to replace them with your account and repository name.

Pros

- Source code is easily visible by recipient (if desired)
- Easy to run (for R users)
- Very easy to update if you already use GitHub for your project
- Git-savvy users can clone and fork your repository

Cons

- Developer must know how to use git and GitHub
- Code is hosted by a third-party server

More Ways to Deliver Shiny Apps to Run Locally

Zip File, delivered over the web

If you store a zip or tar file of your project on a web or FTP server, users can download and run it with a command like this:

```
runUrl('https://github.com/rstudio/shiny_example/archive/master.zip')
```

The URL in this case is a zip file that happens to be stored on GitHub; replace it with the URL to your zip file.

Pros

- Only requires a web server for delivery

Cons

- To view the source, recipient must first download and unzip it

More Ways to Deliver Shiny Apps to Run Locally

Zip File, copied to recipient's computer

Another way is to simply zip up your project directory and send it to your recipient(s), where they can unzip the file and run it the same way you do (`shiny::runApp`).

Pros

- Share apps using e-mail, USB flash drive, or any other way you can transfer a file

Cons

- Updates to app must be sent manually

More Ways to Deliver Shiny Apps to Run Locally

Package

If your Shiny app is useful to a broader audience, it might be worth the effort to turn it into an R package. Put your Shiny application directory under the package's `inst` directory, then create and export a function that contains something like this:

```
shiny::runApp(system.file('appdir', package='packagename'))
```

where `appdir` is the name of your app's subdirectory in `inst`, and `packagename` is the name of your package.

Pros

- Publishable on CRAN
- Easy to run (for R users)

Cons

- More work to set up
- Source code is visible by recipient (if not desired)