

# Shiny Developer @XYZ

Using the biodiversity data, [available here](#) please build a Shiny app that visualizes observed species on the map. Data comes from the [Global Biodiversity Information Facility](#). We're primarily interested in how you build the app structure, how you implement the calculation logic and the quality of code. If you have any questions, please be sure to send us an email!

## Main task

### Business requirements

- General overview:
  - Build a dashboard that main purpose is to visualize selected species observations on the map and how often it is observed.
  - Original dataset is large and covers the whole world. Please use only observations from Poland.
- Specific requirements:
  - Users should be able to search for species by their vernacularName and scientificName. Search field should return matching names and after selecting one result, the app displays its observations on the map.
  - Users should be able to view a visualization of a timeline when selected species were observed.
  - Default view when no species is selected yet should make sense to the user. It shouldn't be just an empty map and plot. Please decide what you will display.
- [Optional] Use your creativity and add features that you would like to see in this application.

### Technical requirements

- Don't use any scaffolding tools like golem, packer, etc.
- Add readme that will help potential future developers of this app
- Deploy the app to shinyapps.io
- Decompose independent functionalities into shinyModules
- Add unit tests for the most important functions and cover edge cases.
- Share your solution using Github with [@xyz-hiring](#) user. Do not open your repo to the public.

## Extras

Below requirements are optional, but you can implement them if you apply for a more Senior role. Please add in the readme, which extra assignments you included in your solution.

### Beautiful UI skill

- Use CSS and [Sass](#) to style your dashboard. Make the dashboard look better than standard Shiny. The visual effect we aim for: [example 1](#),

### Performance optimization skill

- Use as big dataset as you can (not only limited to Poland). Make sure your app initializes fast and the search field is fast. Use as many optimization techniques as possible and describe them in readme.

### JavaScript skill

- Use JavaScript to create non-trivial visualization. It can be for example a timeline of selected species observations or additional information about point on the map (e.g. popup with the picture)

### Infrastructure skill

- *Instead of deploying the app to shinyapps.io, please deploy it to your own instance.*
  - *You can use any cloud provider you want (e.g. AWS, Azure, Google Cloud, etc).*
  - *You can use any deployment environment you want (e.g. Shinyproxy, Shiny Server, Connect etc).*