

# Package ‘crisprShiny’

May 25, 2026

**Title** Exploring curated CRISPR gRNAs via Shiny

**Version** 1.9.0

**Date** 2026-02-23

**Description** Provides means to interactively visualize guide RNAs (gRNAs) in GuideSet objects via Shiny application. This GUI can be self-contained or as a module within a larger Shiny app. The content of the app reflects the annotations present in the passed GuideSet object, and includes intuitive tools to examine, filter, and export gRNAs, thereby making gRNA design more user-friendly.

**Depends** R (>= 4.4.0), shiny

**Imports** BiocGenerics, Biostrings, BSgenome, crisprBase, crisprDesign, crisprScore, crisprViz, DT, Seqinfo, htmlwidgets, methods, pwalign, S4Vectors, shinyBS, shinyjs, utils, waiter

**Suggests** BiocStyle, knitr, rmarkdown, shinyFeedback, testthat (>= 3.0.0), BSgenome.Hsapiens.UCSC.hg38

**biocViews** CRISPR, FunctionalGenomics, GeneTarget, GUI

**License** MIT + file LICENSE

**Encoding** UTF-8

**Roxygen** list(markdown = TRUE)

**RoxygenNote** 7.3.3

**VignetteBuilder** knitr

**BugReports** <https://github.com/crisprVerse/crisprShiny/issues>

**URL** <https://github.com/crisprVerse/crisprShiny>

**Config/testthat/edition** 3

**git\_url** <https://git.bioconductor.org/packages/crisprShiny>

**git\_branch** devel

**git\_last\_commit** 21f982d

**git\_last\_commit\_date** 2026-04-28

**Repository** Bioconductor 3.24

**Date/Publication** 2026-05-25

**Author** Jean-Philippe Fortin [aut, cre],  
Luke Hoberecht [aut]

**Maintainer** Jean-Philippe Fortin <fortin946@gmail.com>

## Contents

crisprServer . . . . .	2
crisprShiny . . . . .	3
crisprUI . . . . .	4
guideSetExample_basic . . . . .	5
guideSetExample_kras . . . . .	6
guideSetExample_kras_be . . . . .	6
guideSetExample_ntcs . . . . .	7
tooltipAnnotation . . . . .	7
tss_kras . . . . .	8
txdb_kras . . . . .	8

<b>Index</b>	<b>9</b>
--------------	----------

---

crisprServer	<i>Server component for crisprShiny App</i>
--------------	---

---

### Description

Server component for crisprShiny App. Not intended for direct use. (intended to be passed as server component of module in shinyApp function)

### Usage

```
crisprServer(id, guideSet, geneModel, title = NULL, useFilterPresets = TRUE)
```

### Arguments

id	Module ID.
guideSet	A GuideSet object.
geneModel	A GRangesList object obtained using <code>crisprDesign::TxDb2GRangesList</code> .
title	Optional title to display at head of app.
useFilterPresets	Whether to use preset filter values on app launch. See details section for <code>?crisprShiny</code> .

### Value

Shiny module server component.

### Examples

```
library(crisprShiny)
ui <- function(id){
  fluidPage(
    crisprUI(id)
  )
}

server <- function(id, gs){
  function(input, output, session){
    observeEvent(gs, {
```

```
        crisprServer(  
          id,  
          guideSet=gs,  
          geneModel=NULL,  
          useFilterPresets=TRUE  
        )  
      })  
    }  
  }  
  
myApp <- function(gs){  
  shinyApp(ui=ui("id"), server=server("id", gs))  
}  
  
if (interactive()){  
  data("guideSetExample_basic", package="crisprShiny")  
  myApp(guideSetExample_basic)  
}
```

---

crisprShiny

*Interactive visualization of GuideSets via Shiny applications*

---

## Description

Means to interactively visualize gRNAs within a GuideSet object via a Shiny application. Contents of the Shiny app reflect annotations contained within the GuideSet object, and provide intuitive controls to examine, filter, and export .csv tables of gRNAs.

## Usage

```
crisprShiny(guideSet, geneModel = NULL, useFilterPresets = TRUE)
```

## Arguments

guideSet	A GuideSet object.
geneModel	A GRangesList object obtained using <code>crisprDesign::TxDb2GRangesList</code> .
useFilterPresets	Whether to use preset filter values on app launch. See details.

## Details

### Preset filter values

Using sensible, preset filters can conveniently remove many poorer-quality gRNAs from view upon app launch. This can be done by setting `useFilterPresets=TRUE`, while passing `FALSE` will retain all gRNAs in `guideSet`. Of course, filters can still be adjusted within the app to either further refine or broaden the list of gRNAs to view. Setting `useFilterPresets=TRUE` (default) will impose the following filter criteria, as appropriate to the `guideSet`, upon app launch:

- spacers with polyT are excluded
- permissible spacer percent GC range set to [20, 80]
- spacers missing values for any score method (NA) are excluded

- (SpCas9 nuclease only) minimum permissible DeepHF and DeepSpCas9 scores set to 0.5
- spacers targeting repeat elements are excluded
- spacers overlapping SNPs are excluded
- (for GuideSets having gene annotation) spacers targeting the final 15% of the gene CDS (i.e., 3' end) are excluded

Filters will not be applied if the guideSet lacks the necessary annotation. For example, a guideSet lacking SNP annotation will not be filtered on the SNP criterium.

### Value

A Shiny app object.

### Examples

```
library(crisprShiny)
data("guideSetExample_kras", package="crisprShiny")
app <- crisprShiny(guideSetExample_kras)

if (interactive()) {
  shiny::runApp(app)
}
```

---

crisprUI

*UI component for crisprShiny App*

---

### Description

UI component for crisprShiny App. Not intended for direct use.

### Usage

```
crisprUI(id, cssFile = NULL)
```

### Arguments

<code>id</code>	Module ID.
<code>cssFile</code>	Optional path of CSS file to be included in rendering app. NULL gives default styling, found in <code>inst/www/styling.css</code> .

### Value

Shiny module UI component.

## Examples

```
library(crisprShiny)
ui <- function(id){
  fluidPage(
    crisprUI(id)
  )
}

server <- function(id, gs){
  function(input, output, session){
    observeEvent(gs, {
      crisprServer(
        id,
        guideSet=gs,
        geneModel=NULL,
        useFilterPresets=TRUE
      )
    })
  }
}

myApp <- function(gs){
  shinyApp(ui=ui("id"), server=server("id", gs))
}

if (interactive()){
  data("guideSetExample_basic", package="crisprShiny")
  myApp(guideSetExample_basic)
}
```

---

guideSetExample\_basic *Example of a [GuideSet](#) object storing gRNA sequences targeting the CDS of the human gene KRAS*

---

## Description

Example of a [GuideSet](#) object (with no additional annotation) storing gRNA sequences targeting the coding sequence of human gene KRAS (ENSG00000133703) for SpCas9 nuclease.

## Usage

```
data(guideSetExample_basic, package="crisprShiny")
```

## Format

A [GuideSet](#) object.

## Details

The object was obtained by using `crisprDesign::findSpacers` on a `GRanges` of the CDS region of human gene KRAS. See code in `inst/scripts/generateKrasData.R`.

guideSetExample\_kras *Example of a [GuideSet](#) object storing gRNA sequences targeting the CDS of the human gene KRAS*

---

### Description

Example of a fully annotated [GuideSet](#) object storing gRNA sequences targeting the coding sequence of human gene KRAS (ENSG00000133703) for SpCas9 nuclease.

### Usage

```
data(guideSetExample_kras, package="crisprShiny")
```

### Format

A [GuideSet](#) object.

### Details

The object was obtained by applying all available add\* annotation functions in `crisprDesign` on a `GuideSet` storing gRNAs targeting the CDS region of human gene KRAS. See code in `inst/scripts/generateKrasData.R`.

---

guideSetExample\_kras\_be  
*Example of a [GuideSet](#) object storing gRNA sequences targeting the CDS of the human gene KRAS*

---

### Description

Example of a [GuideSet](#) object storing gRNA sequences targeting the coding sequence of human gene KRAS (ENSG00000133703) for BE4max nuclease.

### Usage

```
data(guideSetExample_kras_be, package="crisprShiny")
```

### Format

A [GuideSet](#) object.

### Details

The object was obtained by applying all base-editor-specific annotation functions in `crisprDesign` on the CDS region of human gene KRAS. See code in `inst/scripts/generateKrasData.R`.

---

guideSetExample_ntcs	<i>Example of a <a href="#">GuideSet</a> object storing gRNA sequences targeting the CDS of the human gene KRAS and NTCs</i>
----------------------	--

---

### Description

Example of a fully annotated [GuideSet](#) object storing gRNA sequences targeting the coding sequence of human gene KRAS (ENSG00000133703) and some non-targeting controls (NTCs) for SpCas9 nuclease.

### Usage

```
data(guideSetExample_kras, package="crisprShiny")
```

### Format

A [GuideSet](#) object.

### Details

The object was obtained by applying all available add\* annotation functions in `crisprDesign` on a [GuideSet](#) storing gRNAs targeting the CDS region of human gene KRAS and some NTCs. See code in `inst/scripts/generateKrasData.R`.

---

tooltipAnnotation	<i>List of tooltip annotations</i>
-------------------	------------------------------------

---

### Description

Tooltip messages for `crisprShiny` app stored in a list object.

### Usage

```
data(tooltipAnnotation, package="crisprShiny")
```

### Format

A list object.

### Details

See code in `inst/scripts/generateKrasData.R` for tooltip annotations.

---

tss_kras	<i>Example of a GenomicRanges object storing annotated TSS ranges for the human gene KRAS</i>
----------	---

---

**Description**

Example of a GenomicRanges object storing annotated TSS ranges for the human gene KRAS (ENSG00000133703).

**Usage**

```
data(tss_kras, package="crisprShiny")
```

**Format**

A GenomicRanges object.

**Details**

The object was obtained by subsetting genomic ranges in tss\_human from the crisprDesignData package for the KRAS gene (Ensembl ID: ENSG00000133703). See code in inst/scripts/generateKrasData.R.

---

txdb_kras	<i>Example of a CompressedGenomicRangesList object storing annotated ranges for the human gene KRAS</i>
-----------	---

---

**Description**

Example of a CompressedGenomicRangesList object storing annotated genomic ranges for the human gene KRAS (ENSG00000133703).

**Usage**

```
data(txdb_kras, package="crisprShiny")
```

**Format**

A CompressedGenomicRangesList object.

**Details**

The object was obtained by subsetting all genomic ranges annotations in txdb\_human from the crisprDesignData package for the KRAS gene (Ensembl ID: ENSG00000133703). See code in inst/scripts/generateKrasData.R.

# Index

## \* datasets

- guideSetExample\_basic, [5](#)
- guideSetExample\_kras, [6](#)
- guideSetExample\_kras\_be, [6](#)
- guideSetExample\_ntcs, [7](#)
- tooltipAnnotation, [7](#)
- tss\_kras, [8](#)
- txdb\_kras, [8](#)

- crisprServer, [2](#)
- crisprShiny, [3](#)
- crisprUI, [4](#)

- GuideSet, [5–7](#)
- guideSetExample\_basic, [5](#)
- guideSetExample\_kras, [6](#)
- guideSetExample\_kras\_be, [6](#)
- guideSetExample\_ntcs, [7](#)

- tooltipAnnotation, [7](#)
- tss\_kras, [8](#)
- txdb\_kras, [8](#)