

Package: PhysioRecipes (via r-universe)

October 3, 2026

Title Reproducible Analysis Case Studies for the PhysioExperiment Ecosystem

Version 0.7.4

Description A version-controlled database of self-contained, reproducible analysis case studies on public physiological data, built on the PhysioExperiment ecosystem. Each case states a proposition, answers a question on a public dataset and completes through to figures and a conclusion, with every reported number traced to an artifact and the whole op-DAG replayable byte-identically. The package ships the verified run bundles and a small API to browse the case metadata, open a case's bundle, and check the invariants that every case must satisfy.

License MIT + file LICENSE

Encoding UTF-8

Imports jsonlite, tools, utils

Suggests testthat (>= 3.2.0), PhysioLake, knitr, rmarkdown

VignetteBuilder knitr

RoxygenNote 7.3.3

Roxygen list(markdown = TRUE)

Config/testthat/edition 3

Repository <https://x-biosignal.r-universe.dev>

Date/Publication 2026-10-03 13:11:48 UTC

RemoteUrl <https://github.com/x-biosignal/PhysioRecipes>

RemoteRef HEAD

RemoteSha 881fbf8886ead9d423d0a624d42273e2499144de

Contents

case_bundle	2
case_path	2

escalations	3
get_case	4
list_cases	4
validate_cases	5

Index	6
--------------	----------

case_bundle	<i>Path to a case’s verified run bundle</i>
-------------	---

Description

The substrate bundle (frozen pre-registration, run manifest, op-DAG, terminal artifact, claims and verification report) — the evidence a reader re-runs or audits. The same shape `PhysioLake::physioPutBundle()` stores with lineage.

Usage

```
case_bundle(id)
```

Arguments

id Case id.

Value

Absolute path to the case’s bundle/ directory.

Examples

```
if (length(list.files(system.file("cases", package = "PhysioRecipes")))) {
  id <- list_cases()$id[1]
  list.files(case_bundle(id))
}
```

case_path	<i>Path to a case directory or one of its files</i>
-----------	---

Description

Path to a case directory or one of its files

Usage

```
case_path(id, ...)
```

Arguments

`id` Case id.
`...` Optional path components under the case directory (e.g. "bundle", "bundle", "prereg.json").

Value

An absolute path.

Examples

```
if (length(list.files(system.file("cases", package = "PhysioRecipes")))) {  
  id <- list_cases()$id[1]  
  case_path(id)  
  case_path(id, "case.json")  
}
```

escalations

Cases that escalate an open question to a paper

Description

The research queue: cases whose open_questions carry an escalate target.

Usage

```
escalations()
```

Value

A data.frame of id, title and escalation target(s).

Examples

```
if (length(list.files(system.file("cases", package = "PhysioRecipes"))))  
  escalations()
```

get_case	<i>Read a single case record</i>
----------	----------------------------------

Description

Read a single case record

Usage

```
get_case(id)
```

Arguments

id Case id (the case sub-directory name, e.g. "ecg-hrv-mitbih").

Value

The parsed case.json as a list.

Examples

```
if (length(list.files(system.file("cases", package = "PhysioRecipes"))))
  str(get_case(list_cases()$id[1]), max.level = 1)
```

list_cases	<i>The case database as one row per case</i>
------------	--

Description

Browse the whole corpus: id, title, modality, ecosystem packages exercised, dataset, reproducibility status and whether the case escalates an open question to a paper.

Usage

```
list_cases()
```

Value

A data.frame, one row per case.

Examples

```
if (length(list.files(system.file("cases", package = "PhysioRecipes"))))
  list_cases()
```

validate_cases	<i>Check the invariants every case must satisfy</i>
----------------	---

Description

Structural gate (dataset is public; every claim is GROUNDED in the case's `verification_report.json`; `n_grounded == n_claims`; a byte-identical replay is recorded; a verified/escalated case has all claims grounded). The pre-registration hash and per-case number tracing are checked by `tools/validate_case.py` in CI.

Usage

```
validate_cases(ids = list_cases()$id)
```

Arguments

ids	Cases to check (default: all).
-----	--------------------------------

Value

A `data.frame` with one row per case and a logical `ok` column.

Examples

```
if (length(list.files(system.file("cases", package = "PhysioRecipes")))) {  
  # Check a few cases (omit `ids` to check the whole corpus).  
  validate_cases(ids = utils::head(list_cases()$id, 3))  
}
```

Index

`case_bundle`, [2](#)

`case_path`, [2](#)

`escalations`, [3](#)

`get_case`, [4](#)

`list_cases`, [4](#)

`PhysioLake::physioPutBundle()`, [2](#)

`validate_cases`, [5](#)