

Nicolas Hafner @Shinmera

https://everything.shinmera.com

# Radiance - A Web Framework Environment for Common Lisp

### Usual Framework Focus

- Single application
- · Tightly integrated
- · Convenience through large toolsets

### Usual Framework Focus

- Single application
- Tightly integrated
- Convenience through large toolsets
- Often force libraries on you

### Usual Framework Focus

- Single application
- Tightly integrated
- Convenience through large toolsets
- Often force libraries on you
- Problematic to host multiple applications

### Radiance's Goals

- Many applications simultaneously
- Shared resources
- Exchangeable parts
- · Only use what you want

### Radiance's Goals

- Many applications simultaneously
- Shared resources
- Exchangeable parts
- Only use what you want
- · Fully specified

# A Tiny Example

- option: Pizza votes: 32
- option: Pasta votes: 4
- option: Lasagna votes: 12
- option: Crêpes votes: 9
- option: Mashed Potatoes votes: 2

Submit

```
(in-package #:rad-user)
(db:create 'vote '((option (:varchar 32))
                   (votes :integer)))
(define-page display #@"/" ()
 (r-forms:choose
   @"/api/vote" "id"
  (dm:get 'vote (db:query :all))))
(define-api vote (id) ()
 (with-model-save model ('vote (:= ' id id))
      (votes)
    (incf votes))
 (redirect #@"/")
 (api-output "Vote registered."))
```

```
(define-interface (database db)
  (defun create (collection structure
                 &key indices if-exists))
 (defmacro query (query-form))
 . . . )
(define-interface (data-model dm)
 (defun get (collection query
              &key (skip 0) (amount 0) sort))
 ...)
```

```
(define-module #:my-database
 (:use #:c1)
  (:implements #:database))
(in-package #:my-database)
(defun db:create (database db
                  &key indices if-exists)
  ...)
(defmacro db:query (query-form)
  ...)
```

### What Radiance Offers

Modules

Interfaces

Request handling

**URIs** 

**URI** Patterns

Resources

Bidirectional Routing

Dispatching

#### What Radiance Offers

Modules

Interfaces

Request handling

URIs

**URI** Patterns

Resources

Bidirectional Routing

Dispatching

Banning

Rate limiting

Administration interface

Cache handling

Sessions

User storage

User profiles

Logging

Database interaction

DB abstractions

Templating

Error handling

### What Radiance Offers

Modules Sessions

Interfaces User storage

Request handling User profiles

URIs Logging

URI Patterns Database interaction

Resources DB abstractions

Bidirectional Routing Templating

Dispatching Error handling

Banning Chatlog

Rate limiting Blog software

Administration interface File hosting

Cache handling Imageboard

**Authentication** 

Datahasa introchaction

## Blog

Index

#### 妖怪の世捨て人

Ramblings about stag beetles and lisp

#### A Not So Brief Update - Confession 50

It's been a while since I've made a blog about my personal life, even though I recall promising that I would. So, I suppose it can't hurt to quickly summarise what's been going on, and what will most likely be going on in the near future.



#### Using CL+Qt - Confession 49

2015.02.12

2015.02.22

Some deem it unfortunate, others are not bothered by it at all, but the fact remains that Common Lisp does not have a standard GUI toolkt. It does have a native toolkit called McCLIM, but due to general outdated-ness it is not a very attractive choice. Generally I'm not one to linger long on decisions when it comes to learning something, so after quickly evaluating the options I chose to try CommonQt, a library to allow using the Qt framework with CI.



#### Memoirs of a Lost Girl - Entry No. 7

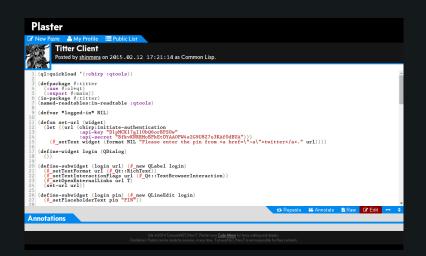
2015.02.00

It has been a long time since I've had the chance to write an entry. A lot has happened – too much to think about, let alone put it into words. Many a thing that I've experienced is still strange and foreign to me, as if from a bad fream.

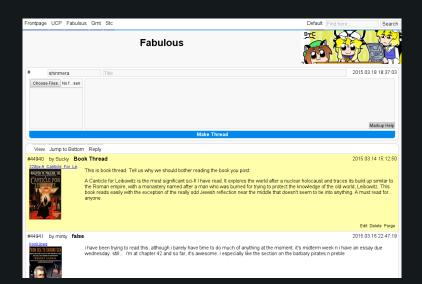
Running Tests in CL - Confession 48

2015.01.31

#### Paste Service



# **I**mageboard





https://github.com/Shirakumo/radiance

```
(ql-dist:install-dist
  "http://dist.tymoon.eu/shirakumo.txt")
(ql:quickload :radiance)
(radiance:startup)
```