



Pontificia Universidad Católica de Chile
Escuela de Ingeniería
Departamento de Ciencia de la Computación

Clase 9

Tests Proyecto Semetral

IIC3745 – Testing

Rodrigo Saffie

rasaffie@uc.cl

14 de agosto de 2020

Tests Proyecto Semestral

1. Arquitectura MVC

2. Tests de:

- modelos
- controladores
- vistas
- rutas

3. Herramientas de apoyo

4. Lecturas adicionales

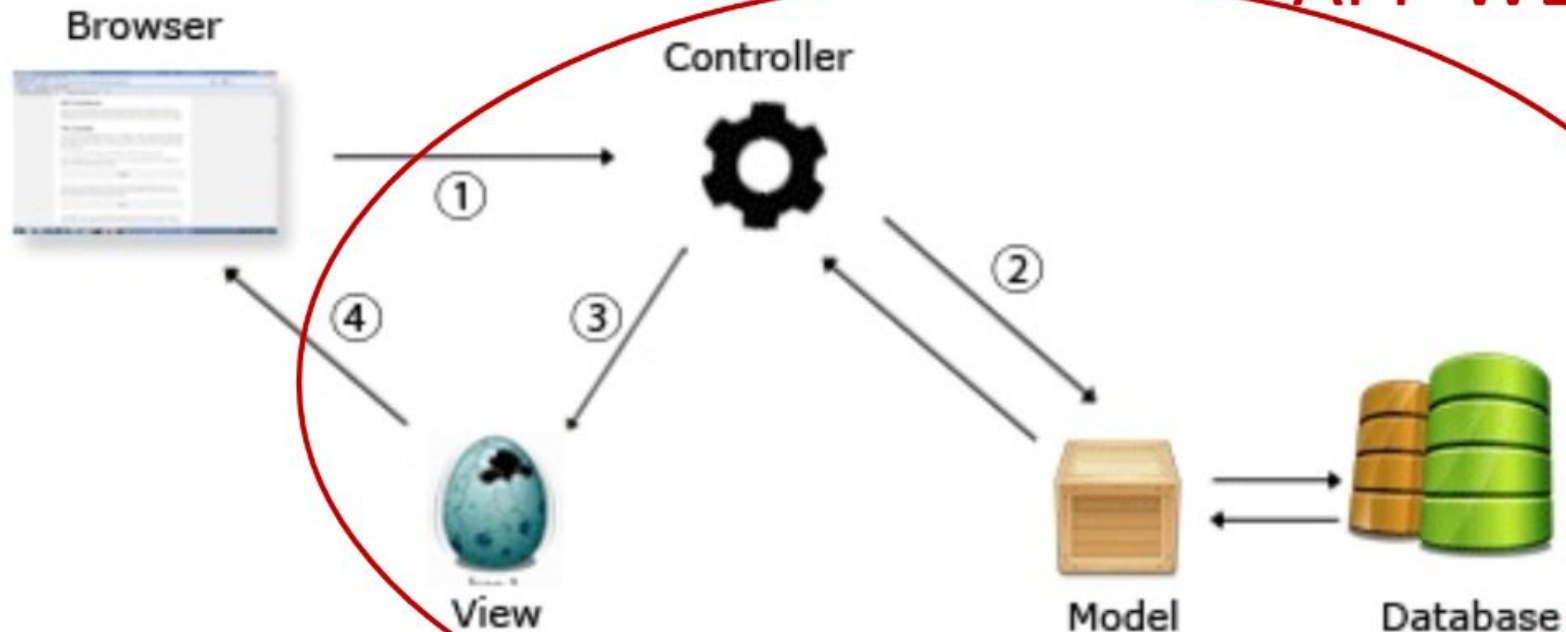
RoR: Arquitectura MVC

- **Modelo**: objetos y métodos con la lógica de la aplicación
- **Vista**: código para construir los documentos que se envían al usuario
- **Controlador**: código que sirve de nexo entre modelos y vistas

RoR: Arquitectura MVC

The Model-View-Controller architecture

APP WEB



Tests de modelos

- Sirven para probar lógica de los modelos de la aplicación
- Generalmente se prueba:
 - *Active Model*
 - validaciones
 - *Active Record*
 - relaciones entre modelos
 - validaciones en DB
 - Lógica de negocio propia al modelo
 - Si la lógica es compleja usar [services](#)

Tests de modelos

- Active Model:
 - GoRails: [How to Test Validations In Rails](#)
 - *Shoulda-matcher*:
 - [*numericality_of*](#)
 - [*presence_of*](#)
- *Active Record*
 - [*belongs_to*](#)
 - [*have_db_column*](#)
 - [*validates_uniqueness_of*](#)

Tests de controladores

- Sirven para probar el flujo de las *requests* y *responses* a la aplicación
- Generalmente se prueba:
 - Gestión de *parámetros* de las *requests*
 - Invocación de la lógica de negocio
 - Por ejemplo: diferencias si el usuario está autenticado o no
 - *Status codes* de las *responses*
 - Qué vistas, *layouts* y *templates* se *renderizan*
 - Manejo de *cookies*
 - *session*
 - *flash*
 - Invocación de *callbacks* antes y después del método

Tests de controladores

- Parámetros
 - [filter_param](#)
 - [permit](#)
- Status Codes: [respond_with](#)
- Vistas, layouts, templates:
 - [render_template](#)
 - [render_with_layout](#)
- Cookies
 - [set_session](#)
 - [set_flash](#)
- Callbacks
 - [use_before_action](#)

Tests de rutas

- Sirven para probar que las rutas de las aplicaciones con sus verbos *HTTP* existan
- Sirven para validar el diseño REST de la aplicación
 - Es una forma de garantizar “un contrato” sobre la interfaz
 - Especialmente útil cuando se expone una API
- No son útiles si se implementa un enfoque con [GraphQL](#)

Tests de rutas

- *routes*

Tests de vistas

- Sirven para probar los elementos que se dibujan en los templates de vistas
 - [view spec](#)

Herramientas de apoyo

- *Coverage*
 - [SimpleCov](#)
 - [CodeCov](#)
 - [CoverAlls](#)
- [fixtures](#) / [factory](#) [bot](#) [rails](#)
- [TimeHelpers](#)

Lecturas adicionales

- [Better Spec](#)
- [*Google - Testing on the Toilet*](#)
 - [Risk-Driven Testing](#)
 - [Prefer Testing Public APIs Over Implementation-Detail Classes](#)
 - [Writing Descriptive Test Names](#)
 - [Keep Cause and Effect Clear](#)
 - [Keep Tests Focused](#)
- [Thoughtbot - How We Test Rails Applications](#)



Pontificia Universidad Católica de Chile
Escuela de Ingeniería
Departamento de Ciencia de la Computación

Clase 9

Tests Proyecto Semetral

IIC3745 – Testing

Rodrigo Saffie

rasaffie@uc.cl

14 de agosto de 2020