

Code Less and Ship faster with OpenAI



Talk at saigon.rb on 21 March 2023.



How AI makes me better

3 use cases: LLM fill in the gaps where I...

1. am not good at
2. have no time for
3. am lazy or forgetful

Bonus: Use API for text + creativity only,
not for right answers

1. am not good at
2. no time
3. am lazy of forgetful

1. am not good at = **Frontend**

2. no time = ?

3. am lazy of forgetful = ?

1. am not good at = **Frontend**

2. no time = **Tests & Refactoring**

3. am lazy of forgetful =

1. am not good at = **Frontend**
2. no time = **Tests & Refactoring**
3. am lazy or forgetful = **Rails API**

1. Do frontend work

Items

- Thẻ yoga (17)
- Dụng cụ giảng dạy (26)
- Giáo trình đặc biệt (8)
- Cấp độ hạt (1)
- Cấp độ mầm (44)
- Cấp độ rễ (43)

Apply

Quantity

From:

To:

Filter

Active filters tags

Search by name

- Bao nhiêu con vật ở nông trại ✕ 3 Storage cấp độ hạt
- Một ngày phiêu lưu của Chuối Betsy ✕ 2 Storage cấp độ hạt
- Sinh nhật hồng hạc bên bờ biển ✕ 2 Storage cấp độ hạt
- Bồ công anh đón xuân ✕ 2 Storage cấp độ hạt
- Những con bọ sau vườn ✕ 2 Storage cấp độ hạt
- Những con bọ sống ở đâu ✕ 1 Storage cấp độ hạt
- Xung quanh cái hồ ✕ 2 Storage cấp độ hạt
- Chào các bạn màu sắc ✕ 1 Storage cấp độ hạt
- Không ở đâu bằng nhà ✕ 2 Storage cấp độ hạt
- Mèo làm gì mỗi ngày ✕ 2 Storage cấp độ hạt

< Prev 1 2 3 4 5 ... 22 Next >

New Item

I have this ViewComponent in Rails and I use stimulus. I want to hide all this code behind a box that looks like a dropdown box. If any tag is included (selected_values contains one), turn the select box green. Else gray. Only use stimulus and tailwind and Rails/ERB.

```
<%= helpers.turbo_frame_tag "tag_filter" do %>  
... (all my ERB code)
```

1-2 Tweaks

make the dropdown button look more like a dropdown menu with an upside / downside angle icon

rotate the angle up then the dropdown is expanded

Results

Tag Filter ^

Quantity ^

Active filters: name chuyển x

Warehouse

Rentals

Tag Filter v

Quantity ^

Giáo trình đặc biệt (8)

Cấp độ cây (40)

Thẻ yoga (17)

Dụng cụ giảng dạy (26)

tags giáo trình

ers x

QUANTITY



Example "make it pretty"

Add/take stock

Quantity:	Kind:	Comment:	<input type="button" value="Add quantity"/>
<input type="text" value="448"/>	<input type="text" value="Add"/>	<input type="text"/>	

This form looks ugly. Please make it pretty! [code]

Add/Take Stock

Quantity	Kind	Comment	<input type="button" value="Add Quantity"/>
<input type="text" value="448"/>	<input type="text" value="Add"/>	<input type="text"/>	

2a. Write tests faster

GPT4: Just throw it all in!

```
task copy: :environment do
  folder = Rails.root.join(ENV["folder_path"], "**/*.rb")
  ruby_files = FileList[folder]

  ruby_files.each do |file_path|
    puts "Ruby file: #{file_path.gsub(Rails.root.to_s, '')}"
    File.foreach(file_path) do |line|
      # exclude lines that start with a hash (#) after
      # optional whitespace
      next if line.match(/^ \s*#/)

      puts line
    end
  end
end
```

```
rake prompts:copy
```

```
folder_path=app/models | pbcopy
```

Write me an integration test for this Rails data model:

```
Ruby file: /app/models/api_key.rb
class ApiKey < ApplicationRecord
  include Discard::Model
  ...
Ruby file: /app/models/user.rb
... (+contents)
Ruby file: /app/models/event.rb ...
... (+contents)
```

This is not an integration test :(

```
RSpec.describe "Integration Test" do
  let(:user) { create(:user, email: "user@example.com") }
  let(:api_key) { ApiKey.generate(user) }
  let(:place) { create(:place, name: "Test Place") }
  let(:resource) { create(:resource, resourceable: place) }
  let(:person) { create(:person, first_name: "John", last_name:
"Doe") }
  let(:start_time) { Time.zone.now }
  let(:end_time) { 1.hour.from_now }
  let(:event) { create(:event, title: "Test Event", start_time:
start_time, end_time: end_time, resource: resource) }
  let(:participation) { create(:participation, event: event,
resource: resource) }

  it "creates and associates all models correctly" do
    user
    api_key
    place
    resource
    person
    event
    participation

    expect(user.email).to eq("user@example.com")
    expect(api_key.token).to be_present
    expect(api_key.user).to eq(user)
  end
end
```



However, this works

Give me some ideas for integration tests that test the collaboration of these models.

```
...  
Test stock changes.  
  Test creating stock changes with different kinds  
  (e.g., add, remove).  
  Test validation for presence and numericality of  
  quantity and initial_quantity.  
...
```

Take #9 and code it in rspec.



Here's an example of RSpec tests for StockChange (scenario #9) from the list above:



ruby

Copy code

```
# spec/models/stock_change_spec.rb
require 'rails_helper'

RSpec.describe StockChange, type: :model do
  let(:user) { create(:user) }
  let(:item) { create(:item, quantity: 10) }

  describe 'validations' do
    it { should validate_presence_of(:quantity) }
    it { should validate_numericality_of(:quantity).is_greater_than_or_equal }
    it { should validate_presence_of(:kind) }
    it { should validate_presence_of(:initial_quantity) }
    it { should validate_numericality_of(:initial_quantity).is_greater_than_ }
    it { should define_enum_for(:kind).with_values(%i[add remove]) }
  end

  describe '.create_and_apply' do
    context 'when adding stock' do
      it 'creates a stock change and updates the item quantity' do
        expect {
          StockChange.create_and_apply(item, 'add', 5, 'Test comment', user)
        }.to change { item.reload.quantity }.by(5)
      end
    end
  end
end
```

2b. Refactoring

The screenshot displays a web application interface with a filter sidebar on the left and a main content area on the right. The sidebar contains several filter options, each with a checkbox and a count in parentheses:

- Thẻ yoga (17)
- Dụng cụ giảng dạy (26)
- Giáo trình đặc biệt (8)
- Cấp độ hạt (1)
- Cấp độ mầm (44)
- Cấp độ rễ (43)

Below these options are two buttons: "Apply" and "Filter". The main content area shows a "Quantity" filter with a range selector. The range is currently set from 2 to 6. Below the range selector is a "Filter" button. The active filters are displayed as tags at the bottom of the page:

Active filters tags: dụng cụ giảng dạy × giáo trình đặc biệt × quantity from 2 × to 6 ×

Below the active filters, there are three items listed:

- Phù thủy Halloween 6 Storage giáo trình đặc biệt
- Trung Thu 4 Storage giáo trình đặc biệt
- Trung Thu gia đình 4 Storage giáo trình đặc biệt

I have 2 Rails ViewComponents, which filter in the view. They save via hotwire into the session.

How can I refactor this to build reusable filter components and not clutter my controller?

Here's the controller:

```
@items = @items.where("name ilike ?", "%#{params[:name]}%") if
params[:name].present?

if session.dig(SESS, "tags")&.size&.positive?
  @items = @items.tagged_with(names: session.dig(SESS, "tags"),
  match: :any)
end

if session.dig(SESS, "quantity", "from")
  @items = @items.where("quantity >= ?", session.dig(SESS,
  "quantity", "from"))
end

@items = @items.where("quantity <= ?", session.dig(SESS,
"quantity", "to")) if session.dig(SESS, "quantity", "to")

@pagy, @items = pagy(@items.load_async)
```

Result: great!

```
filter_components = {  
  "name" => ::Filters::NameFilterComponent,  
  "tags" => ::Filters::TagFilterComponent,  
  "quantity" => ::Filters::QuantityFilterComponent  
}  
  
# Instantiate the applicable filter components  
@filters.to_h.select { |key, _| filter_components.include?(key) }.each do |key, value|  
  @items = filter_components[key].apply_filter(@items, value)  
end
```


This has been done **many times before**

Reviews my structure:

This is my data model in Rails. Do you see any problems with it?

1. In the `Participation` model, you have a validation that ensures uniqueness of `event` scoped to `resource`. It is recommended to add a database index to enforce this uniqueness constraint. Add the following line to a new migration:

```
ruby
```

 Copy code

```
add_index :participations, [:event_id, :resource_id], unique: true
```


These are the main issues I found in your code. Addressing them should help improve the data model. Please double-check the rest of the code to make sure it meets your requirements and adheres to best practices.

3a. Code lazily




The image shows a screenshot of the GitHub Copilot application card on the GitHub marketplace. On the left is the Copilot logo, a blue and white robot head with goggles. To the right, the text reads 'GitHub Copilot' followed by the version number 'v1.78.9758'. Below this, it says 'GitHub' with the GitHub logo and 'github.com', followed by a download icon and the number '4,193,504'. To the right of that is a star rating of four full stars and one half star, with '(830)' in parentheses. Underneath the star rating is the tagline 'Your AI pair programmer'. At the bottom left of the card is a blue button labeled 'Installing' and a gear icon.

GitHub Copilot v1.78.9758

GitHub  github.com |  4,193,504 | ★★★★★☆ (830)

Your AI pair programmer

Installing 

```

4
5 # Creates an item, then adds stock to initialize.
6 # Then rents the item out. Tries to rent it out again (while not having enough stock)
7 # Then returns the first rental.
8 RSpec.describe "Rental system", type: :integration do
  let(:item) { create(:item) }
  let(:user) { create(:user) }
  let(:rental) { create(:rental, item: item, user: user) }

  before do
    item.add_stock(1)
  end

  it "rents an item, then returns it" do
    rental = item.rent(user)
    expect(rental).to be_persisted
    expect(item.stock).to eq(0)
    expect(item.rentals.count).to eq(1)

    expect { item.rent(user) }.to raise_error(Item::NotEnoughStockError)

    rental.return!
    expect(item.stock).to eq(1)
    expect(item.rentals.count).to eq(0)
  end
end
9
10 end

```

Copilot works great for contained methods and algorithms. Not great for structures.

3b. When I forget the API

write me the relations in the model StockChange corresponding to this:

```
class Rental < ApplicationRecord
  belongs_to :item
  belongs_to :renter, class_name: 'Person',
    foreign_key: 'renter_id'
  belongs_to :checkout, class_name: 'StockChange',
    foreign_key: 'checkout_id'
  belongs_to :return, class_name: 'StockChange',
    foreign_key: 'return_id', optional: true
end
```

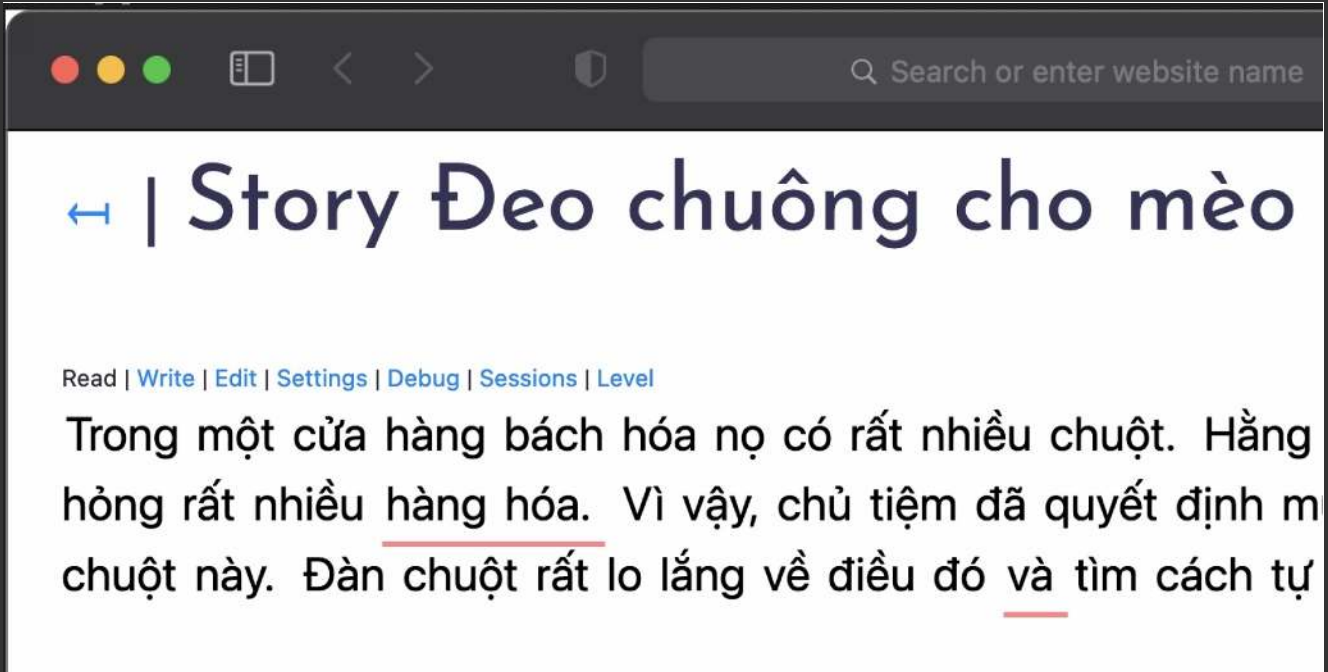
Result:

```
class StockChange < ApplicationRecord
  belongs_to :item
  belongs_to :user

  has_one :checkout_rental, class_name: 'Rental',
    foreign_key: 'checkout_id',
    dependent: :restrict_with_exception
  has_one :return_rental, class_name: 'Rental',
    foreign_key: 'return_id',
    dependent: :restrict_with_exception
end
```



4. Use it as a general API



Text #17 **Origin:** I have a cat...

Lang: vi

Prompts: Translate this into vi: I have a cat. Her name is Lucy. ⚡ | **Tokens:** 45

| *Tôi có một con mèo. Tên cô ấy là Lucy.*

Sentences: Tên cô ấy là Lucy [{"vi":"Tên"}, {"vi":"cô ấy"}, {"vi":"là"}, {"vi":"Lucy"}] ⚡ |

I had hope for GPT-4 :(

Playground

Split this into words: Cô ấy cảm thấy buồn ngủ.

Request AI Text

◀ Prev

1

2

3

4

5

6

Next ▶

Split this into words: Cô ấy cảm thấy buồn ngủ. ⚡ |

Cô ấy / cảm thấy / buồn ngủ.

Time: 1.793787 | Tokens: 45 | gpt-3.5-turbo

Split this into words: Cô ấy cảm thấy buồn ngủ. ⚡ |

Cô, ấy, cảm, thấy, buồn, ngủ.

Time: 2.913913 | Tokens: 47 | gpt-4

Maybe this is not the use case after all



Here is the JSON array containing the parts of the sentence from the given text:

```
[  
  {  
    "subject": "Cô ấy",  
    "verb": "cảm thấy",  
    "adjective": "buồn",  
    "noun": "ngủ"  
  }  
]
```

Great for creative things

Write me a story about 2 french bulldogs who are the dogs of the owner of a coffee shop in Thảo Điền.

Sentences: 0

Text 111

Queries: ⚡ | Tokens: 147

As the coffee shop was closing for the night, the two French Bulldogs wandered around the empty tables, with a ball they had found in their mouth. Suddenly, they heard a sound coming from the back door and ran towards it, excitedly barking. The owner of the coffee shop emerged, surprised to see his playful dogs with a new toy, which he took from them and threw, as they chased after it happily, marking the end of another successful day at the coffee shop.

Sentences: 0

Continue story, include this word:

Request Next Text

Figured out tokenization:

Split this into Vietnamese words that make sense.
For every correct word use the format {"vi": <word>, "en": <English translation>} : " Tôi có một ..."

Origin Who are... generated by translate/vi

Who are you trolling today?

Text 126

Queries: ⚡ | Tokens: 36

Hôm nay bạn đang gây rối ai vậy?

1 Sentences: ⚡ |

Hôm nay, bạn đang gây rối ai vậy.

Transition

continue ▾

Transition info

Request Next Text

Conclusion

- AI is not "Intelligent". It only knows how to figure out the next word, based on a lot of inputs.
- You won't get novel ideas. But you get over the boring stuff, fast.
- Think: What can you build with it?

Photo by [Zac Wolff](#).

2023, TillCarlos.com.