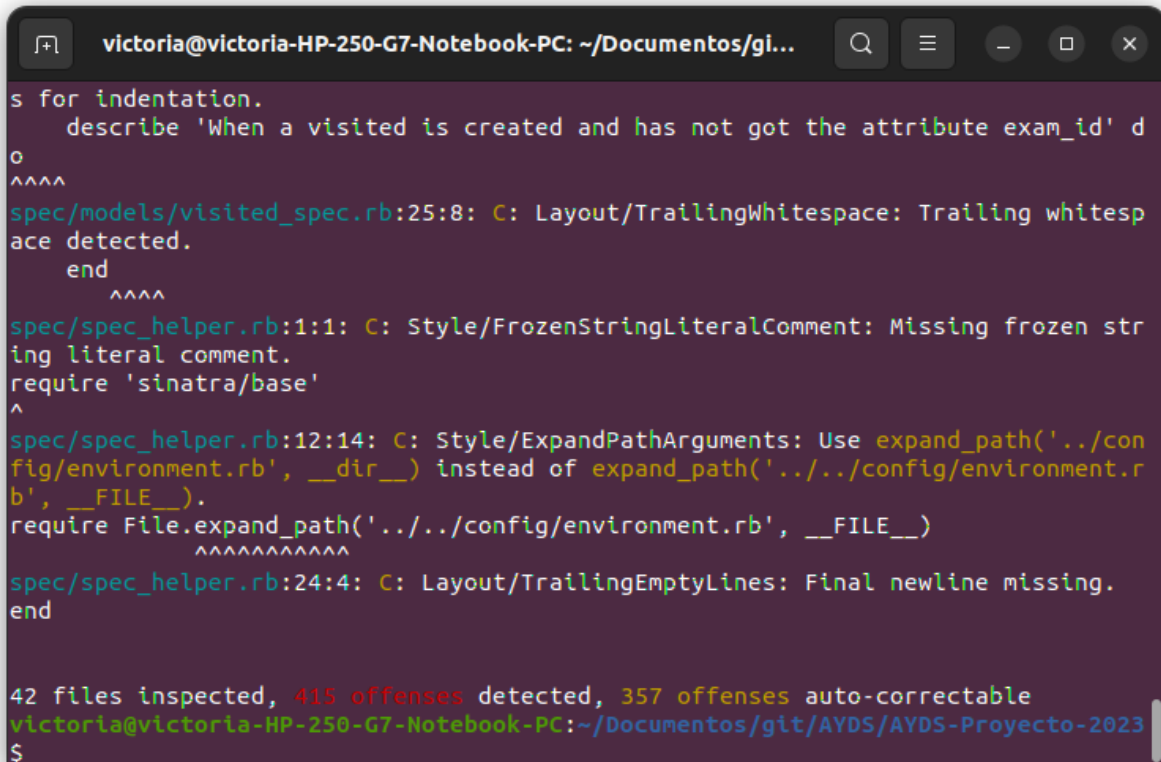


Informe RuboCop

Actividad número 4 del Taller de ingeniería de software.

Comenzamos instalando y ejecutando la herramienta RuboCop donde salieron los siguientes reportes:



```
victoria@victoria-HP-250-G7-Notebook-PC: ~/Documentos/gi...
s for indentation.
  describe 'When a visited is created and has not got the attribute exam_id' d
o
^^^^
spec/models/visited_spec.rb:25:8: C: Layout/TrailingWhitespace: Trailing whitesp
ace detected.
  end
  ^^^^^
spec/spec_helper.rb:1:1: C: Style/FrozenStringLiteralComment: Missing frozen str
ing literal comment.
require 'sinatra/base'
^
spec/spec_helper.rb:12:14: C: Style/ExpandPathArguments: Use expand_path('../con
fig/environment.rb', __dir__) instead of expand_path('../../config/environment.r
b', __FILE__).
require File.expand_path('../../config/environment.rb', __FILE__)
      ^^^^^^^^^^^^^^^
spec/spec_helper.rb:24:4: C: Layout/TrailingEmptyLines: Final newline missing.
end

42 files inspected, 415 offenses detected, 357 offenses auto-correctable
victoria@victoria-HP-250-G7-Notebook-PC:~/Documentos/git/AYDS/AYDS-Proyecto-2023
$
```

Donde nos informaba todas las ofensas encontradas, donde ejecutamos para que se corrijan las auto-correctable y nos quedaro

Donde nos muestra que pudo corregir la mayoría de las ofensas, y quedaron para corregir manualmente

```
victoria@victoria-HP-250-G7-Notebook-PC: ~/Documentos/gi...
spec/models/exam_spec.rb:5:1: C: Metrics/BlockLength: Block has too many lines.
[100/25]
describe 'Exam' do ...
^^^^^^^^^^^^^^^^^^^^
spec/models/exam_spec.rb:6:3: C: Metrics/BlockLength: Block has too many lines.
[98/25]
  describe 'valid' do ...
  ^^^^^^^^^^^^^^^^^
spec/models/option_spec.rb:5:1: C: Metrics/BlockLength: Block has too many lines
. [26/25]
describe 'Option' do ...
^^^^^^^^^^^^^^^^
spec/models/user_spec.rb:5:1: C: Metrics/BlockLength: Block has too many lines.
[50/25]
describe 'User' do ...
^^^^^^^^^^^^^^^^
spec/models/user_spec.rb:6:3: C: Metrics/BlockLength: Block has too many lines.
[48/25]
  describe 'valid' do ...
  ^^^^^^^^^^^^^^^^^

42 files inspected, 72 offenses detected, 72 offenses auto-correctable
victoria@victoria-HP-250-G7-Notebook-PC:~/Documentos/git/AYDS/AYDS-Proyecto-2023
$
```

Donde de todas las ofensas las que corregimos fueron

models/exam.rb - 6 offenses

Line #11 – Convention: Naming/MethodName: Use snake_case for method names.

```
def isValid
```

Line #15 – Convention: Naming/MethodName: Use snake_case for method names.

```
def sumaPuntos
```

Line #23 – Convention: Naming/MethodName: Use snake_case for method names.

```
def restaPuntos
```

Line #31 – Convention: Naming/MethodName: Use snake_case for method names.

```
def restaVida
```

Line #35 – Convention: Naming/MethodName: Use snake_case for method names.

```
def estaEnRacha
```

donde se nos indicaba que la forma de escritura era de la snake_case, entonces en el modelo exam.rb y en los demas archivos donde se usaban estos métodos decidimos cambiar sus nombres de la siguiente manera como nos indicaba RuboCop

```

models > exam.rb
9   attribute :points_streak, :integer, default: 0
10
11   def valid?
12     | life.positive?
13   end
14
15   def suma_puntos
16     | if estaEnRacha
17     |   score + 20
18     | else
19     |   score + 10
20     | end
21   end
22
23   def resta_puntos
24     | if score.zero?
25     |   0
26     | else
27     |   score - 5
28     | end
29   end
30
31   def resta_vida
32     | life - 1
33   end
34
35   def esta_en_racha
36     | points_streak >= 5
37   end
38 end

```

En las ofensas encontradas en server.rb

server.rb - 11 offenses

Line #29 – **Warning:** Lint/AmbiguousBlockAssociation: Parenthesize the param `proc { File.join(root, 'views') }` to make sure that the block will be associated with the `proc` method call.

```
set :views, proc { File.join(root, 'views') }
```

lo que solucionamos con la línea

```
set :views, (proc { File.join(root, 'views') })
```

En este caso nos dijo que había líneas que podíamos mover fuera del condicional debido a que se repetían

Line #200 – **Convention:** Style/Identical/Conditional/Branches: Move @examen.save out of the conditional.

```
@examen.save
```

Line #205 – **Convention:** Style/Identical/Conditional/Branches: Move @examen.save out of the conditional.

```
@examen.save
```

por lo que el código paso de estar asi

```
post '/exam/play/incorrect' do
  if params[:id].present?
    @examen = Exam.find_by(id: params[:id])
    if @examen
      if @examen.score == 0
        @examen.life = @examen.restaVida
        @examen.points_streak = 0
        @examen.save
      else
        @examen.score = @examen.restaPuntos
        @examen.life = @examen.restaVida
        @examen.points_streak = 0
        @examen.save
      end
    end
  end
  erb :respuestaIncorrecta, locals: { examen: @examen }
end
```

a quedar asi

```
post '/exam/play/incorrect' do
  if params[:id].present?
    @examen = Exam.find_by(id: params[:id])
    if @examen
      @examen.score = @examen.resta_puntos unless @examen.score.zero?
      @examen.life = @examen.resta_vida
      @examen.points_streak = 0
      @examen.save
    end
  end
  erb :respuestaIncorrecta, locals: { examen: @examen }
end
```

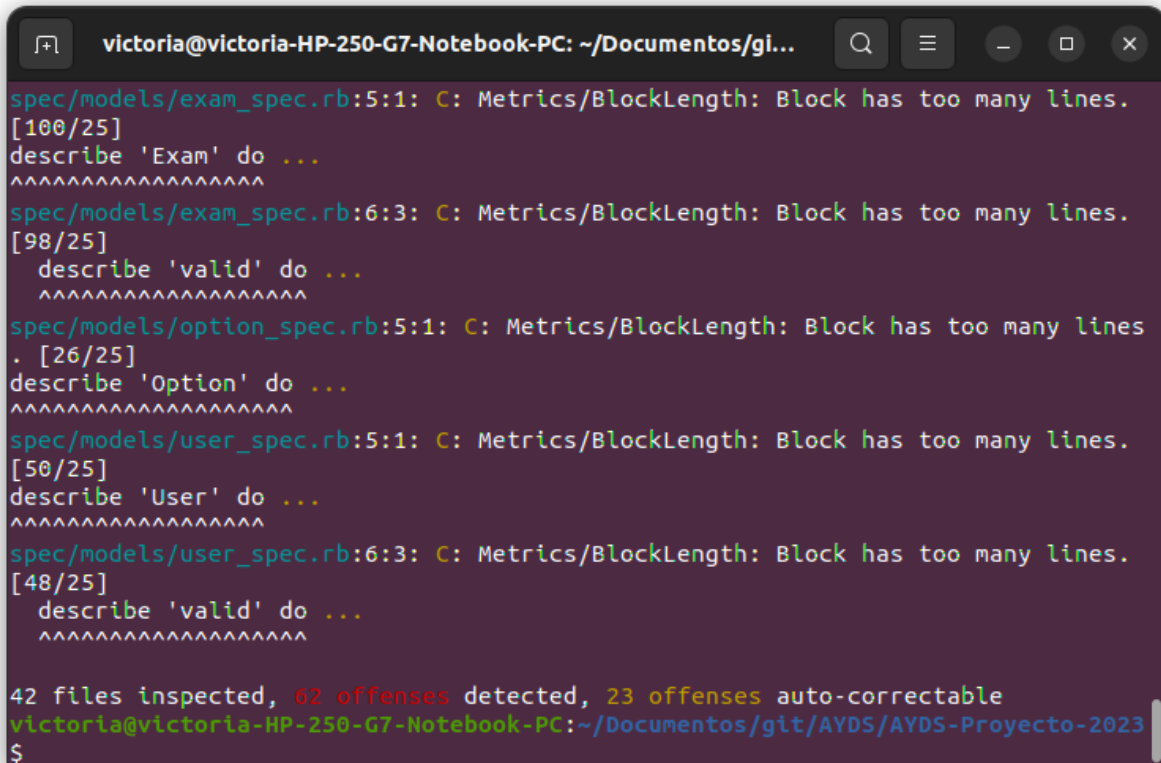
Por último una simple corrección de los comentarios que no utilizabamos símbolos ASCII

spec/app_spec.rb - 1 offense

Line #16 – Convention: Style/AsciiComments: Use only ascii symbols in comments.

```
expect(last_response.status).to eq(200) # Verifica el código de respuesta HTTP
```

La totalidad de las ofensas se encuentran en /reports/html/corrections1.html . Donde se encuentran muchas ofensas en base a los estilos y líneas de código muy largas las cuales decidimos dejarlas de esa manera. En conclusión nos quedaron las siguientes ofensas al ejecutar nuevamente RuboCop con las correcciones mencionadas.



```
victoria@victoria-HP-250-G7-Notebook-PC: ~/Documentos/gi...  
spec/models/exam_spec.rb:5:1: C: Metrics/BlockLength: Block has too many lines.  
[100/25]  
describe 'Exam' do ...  
^^^^^^^^^^^^^^^^^^^^^^^^^^^^  
spec/models/exam_spec.rb:6:3: C: Metrics/BlockLength: Block has too many lines.  
[98/25]  
  describe 'valid' do ...  
  ^^^^^^^^^^^^^^^^^^^^^^^^^^^  
spec/models/option_spec.rb:5:1: C: Metrics/BlockLength: Block has too many lines  
  . [26/25]  
describe 'Option' do ...  
^^^^^^^^^^^^^^^^^^^^^^^^^^^^  
spec/models/user_spec.rb:5:1: C: Metrics/BlockLength: Block has too many lines.  
[50/25]  
describe 'User' do ...  
^^^^^^^^^^^^^^^^^^^^^^^^^^^^  
spec/models/user_spec.rb:6:3: C: Metrics/BlockLength: Block has too many lines.  
[48/25]  
  describe 'valid' do ...  
  ^^^^^^^^^^^^^^^^^^^^^^^^^^^  
  
42 files inspected, 62 offenses detected, 23 offenses auto-correctable  
victoria@victoria-HP-250-G7-Notebook-PC:~/Documentos/git/AYDS/AYDS-Proyecto-2023  
$
```

Se mencionan estas 23 ofensas en el archivo ubicado en /reports/html/corrections4.html

También hemos refactorizado el server.rb utilizando controladores, usando la técnica de RuboCop extract method, que consiste en tomar una parte del código existente en un método y colocarlo en otro por separado. El objetivo es mejorar la claridad, reutilización y mantenibilidad del código.

✓ controllers

■ exams_controller.rb

■ practice_controller.rb

■ users_controller.rb

Donde en estos archivos hemos puesto las rutas correspondientes a cada interfaz.

Exam:

```

class ExamsController < Sinatra::Base
  set :views, File.expand_path('../views', __dir__)

  post '/exam' do
    erb :index3
  end

  post '/exam/play' do
    user = User.find_by(id: session[:user_id])
    @username = user.username.to_s
    @contador ||= 0
    @contador2 ||= 0
    @preguntas = Question.all.shuffle
    @examen = Exam.find_or_create_by(id: params[:id])
    @visited = Visited.all
    @visited.each do |v|
      @contador2 += 1 if v.examen_id == @examen.id
    end
    @preguntas.each do |q|
      @preguntas -= [q] unless Visited.find_by(idq: q.id, examen_id: @examen.id) == nil
    end
    if @contador2 == 17
      user = User.find_by(id: session[:user_id])
      if user.total_score < @examen.score
        user.total_score = @examen.score
        user.save
      end
      erb :end
    elsif !Exam.valid(@examen) # You, yesterday * Refactoring (Controllers)
      user = User.find_by(id: session[:user_id])
      if user.total_score < @examen.score
        user.total_score = @examen.score
        user.save
      end
      erb :lost
    else
      @opciones = [@preguntas[@contador].option.option, @preguntas[@contador].option.option2, @preguntas[@contador].option.correct]
      @opciones = @opciones.shuffle
      Visited.create(idq: @preguntas[@contador].id, examen_id: @examen.id)
      @examen.save
      erb :quiz, locals: { examen: @examen, contador: @contador }
    end
  end

  post '/exam/play/correct' do
    logger.info(params)
    if params[:id].present?
      @examen = Exam.find_by(id: params[:id])
      if @examen

```

Practice:


```

class PracticeController < Sinatra::Base

  set :views, File.expand_path('../views', __dir__)  You, yesterday * Refactoring (Controllers)

  post '/practice' do
    erb :practica
  end
  post '/practice/theoric' do
    erb :teorico
  end

  post '/practice/play' do
    @preguntas = Question.all.shuffle
    @contador ||= 0
    if @contador < @preguntas.length
      @contador += 1
      @practica = Practice.find_or_create_by(id: params[:id])
    end
    @opciones = [@preguntas[@contador].option.option, @preguntas[@contador].option.option2, @preguntas[@contador].option.correct]
    @opciones = @opciones.shuffle
    erb :practicaQuiz
  end

  get '/practice/signal' do
    @signals = Question.where(theme: 1).shuffle
    @count ||= 0
    if @count < @signals.length
      @count += 1
      @practica = Practice.find_or_create_by(id: params[:id])
    end
    @options = [@signals[@count].option.option, @signals[@count].option.option2, @signals[@count].option.correct]
    @options = @options.shuffle
    erb :signal
  end

  post '/practice/definition' do
    @definition = Question.where(theme: 2).shuffle
    @count ||= 0
    if @count < @definition.length
      @count += 1
      @practica = Practice.find_or_create_by(id: params[:id])
    end
    @options = [@definition[@count].option.option, @definition[@count].option.option2, @definition[@count].option.correct]
    @options = @options.shuffle
    erb :definition
  end

  post '/practice/safety-road' do
    @safety = Question.where(theme: 3).shuffle
  end
end

```

User:

```

class UsersController < Sinatra::Base

  set :views, File.expand_path('../views', __dir__) # You, yesterday • Refactoring (Controllers)

  get '/' do
    erb :index
  end

  post '/' do
    user = User.find_by(username: params[:username], password: params[:password])
    if !user.nil?
      session[:user_id] = user.id.to_s
      @user = User.find_by(id: session[:user_id])
      erb :index2
    else
      redirect '/register'
    end
  end

  get '/register' do
    erb :register
  end

  post '/register' do
    if params[:password] == params[:password2]
      user = User.create(username: params[:username], email: params[:email], password: params[:password])
      redirect '/'
    else
      redirect '/register'
    end
  end
end

```