**Code Review Documentation**

**Functional Tests**

**Case 1**: Email login form test
- **Developer**: Andy Park
- **Preconditions**: Website is running and login form page is current displayed
- **Status**: no bugs

```ruby
class SessionsController < ApplicationController
  def new
    end

  def create
    @user = User.find_by_email(params[:session][:email].downcase)
    if @user && @user.authenticate(params[:session][:password])
      session[:user_id] = @user.id
      redirect_to '/profiles'
      flash[:success] = 'Signed in successful!'
    else
      flash[:danger] = 'Invalid email/password combo'
      render :new
    end
  end

  def destroy
    session[:user_id] = nil
    redirect_to '/'
  end
end
```

Session controller

"Welcome back, #{user.email}"

Signed in successful!

Successful Login case
**Case 2:** Password login form test

- **Developer:** Andy Park
- **Preconditions:** Website is running and login form page is currently displayed
- **Status:** no bugs

```ruby
class SessionsController < ApplicationController
  def new
    end
  end

  def create
    @user = User.find_by_email(params[:session][:email].downcase)
    if @user && @user.authenticate(params[:session][:password])
      session[:user_id] = @user.id
      redirect_to '/profiles'
      flash[:success] = 'Signed in successful!
    else
      flash[:danger] = 'Invalid email/password combo'
      render :new
    end
  end

  def destroy
    session[:user_id] = nil
    redirect_to '/'
  end
end
```

Session controller

"Welcome back, #{user.email}"

Signed in successful!

Successful login case
SQL user account database with encryption of password

**Case 3:** Search form test  
- Developer: Danny Nuch  
- Preconditions: Website is running and home page is displayed with search bar in top right corner  
- **Status:** no bugs

```ruby
def index
  @games = Game.all
  if params[:search]
    @games = Game.search(params[:search]).order("created_at DESC")
  else
    @games = Game.all.order("created_at DESC")
  end
end
```

Search form code

![Search form code](image)

**Case 4:** “Play” shuffle button test  
- Developer: Danny Nuch  
- Preconditions: Website is running and home page is displayed  
- **Status:** in progress

**Case 5:** Game “Show” page from games directory test  
- Developer: Danny Nuch  
- Preconditions: Website is running and login form page is current displayed  
- **Status:** in progress
Case 6: Game “Show” page from homepage test
   ❑ Developer: Danny Nuch
   ❑ Preconditions: Website is running and login form page is current displayed
   ❑ Status: in progress

Case 7: Description of game shown as pop-up when hovered over test
   ❑ Developer: Linda Nguyen
   ❑ Preconditions: Website is running and login form page is current displayed
   ❑ Status: no bugs

Pop-up code

```
</div>
<br />
<div class="col-md-3 portfolio-item">
<img src="/assets/es5/elderscrolls5.png" alt="" class="tooltip" title="The Elder Scrolls V: Skyrim is an open world RPG developed and published by Bethesda Softworks" />
</div>
```

Top Rated Games

- Final Fantasy XV
- Elder Scrolls V: Skyrim
- Pokemon Sun
- Fallout 4
- Pokemon X
- Tales of Berseria

© Powerplay

Screenshot of case

Case 8: User comment test
   ❑ Developer: Linda Nguyen
   ❑ Preconditions: Website is running and a game’s view page is displayed and user is signed in
   ❑ Status: no bugs
Comment code in view/comments/index.html.erb
# GET /comments/1/edit
def edit
end

# POST /comments
# POST /comments.json
def create
  @comment = Comment.new(comment_params)
  respond_to do |format|
    if @comment.save
      format.html { redirect_to @comment, notice: 'Comment was successfully created.' }
      format.json { render :show, status: :created, location: @comment }
    else
      format.html { render :new }
      format.json { render json: @comment.errors, status: :unprocessable_entity }
    end
  end
end

# PATCH/PUT /comments/1
# PATCH/PUT /comments/1.json
def update
  respond_to do |format|
    if @comment.update(comment_params)
      format.html { redirect_to @comment, notice: 'Comment was successfully updated.' }
      format.json { render :show, status: :ok, location: @comment }
    else
      format.html { render :edit }
      format.json { render json: @comment.errors, status: :unprocessable_entity }
    end
  end
end

# DELETE /comments/1
# DELETE /comments/1.json
def destroy
  @comment.destroy
  respond_to do |format|
    format.html { redirect_to comments_url, notice: 'Comment was successfully destroyed.' }
    format.json { head :no_content }
  end
end

Comments controller
Case 9: User rate game test
   - Developer: Andy Park
   - Preconditions: Website is running and a game’s view page is displayed and user is signed in
   - Status: in progress

Case 10: User write review test
   - Developer: Linda Nguyen
   - Preconditions: Website is running and a game’s view page is displayed and user is signed in
   - Status: in progress

Comment page layout on webpage

<table>
<thead>
<tr>
<th>id</th>
<th>text</th>
<th>game_id</th>
<th>created_at</th>
<th>updated_at</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>happy saturday!!</td>
<td>0</td>
<td>2017-04-23 04:34:36.181330</td>
<td>2017-04-23 04:34:36.181330</td>
</tr>
</tbody>
</table>

SQL: Comment database