Links

Heroku: https://eventnxt-fall2024-dev-67890e38df3f.herokuapp.com/
Linear: https://linear.app/eventnxt/team/EVE/all
Github: https://github.com/FashioNXT/EventNXT-Fall2024/tree/dev

Dates

7 October, 2024 - 18 October, 2024

Members Contributions

Project Manager	Duties		Estimated Time
Kiara Sims	Made user stories Check pull requests Communicated with client Made documents for sprint Made sure discussion assignments were done as a team Did rubocop fixes		10
Scrum Master	Duties		Estimated Time
Ming-Sheng Xu	Daily check ins with developers Made documents for sprint Check pull requests Data Validation Features of Box Office Sales Help other developers to write rspec.		10
Developers	User Stories	Points	Estimated Time
Neil			
	1. Feature: Seating Levels	1	1 hour
	2. Feature: Seating Summary	2	2 hours
	3. Feature: Booked Seats column	3	3 hours
	4. Feature: Committed Seats column	3	3 hours
Total		9	9 hours

Rishabh			
	5. Feature: Manage Invited Guests	5	
	6. Feature: Manage Invited Guests - Data Validation	2	
	7. Feature: Manage Invited Guests - Data Validation	2	
Total		9	9 hours
Parth	8. Feature: Drop down of event selection	2	
	9. Feature: No drop down of guest	2	
Total		4	4 hours
Surya			
	10. CRM Authentication	5	
Total		5	5 hours

Sprint Goal

After the bug fixes in Sprint 1, we are now adding improvements to the current features. After talking to the client these user stories are features that need to be working. These improvements are meant to make the application more user-friendly.

Sprint Achievements

The team completed the majority of the user stories for the improvements and rspec fixes are completed. Sprint 2 achievements are as follows:

- Seating level and Invited Guests are fixed
- Login and logout feature is fixed
- Overall progress with the application is being made.

Sprint backlog items and status

- *Box Office Sales - Data Validation -> Box Office Sales: Connect to Eventbrite
    - The tickets for data validation have been changed to several tickets about getting data from Eventbrite (see sprint-3 plan) because of the change of the client's requirement.

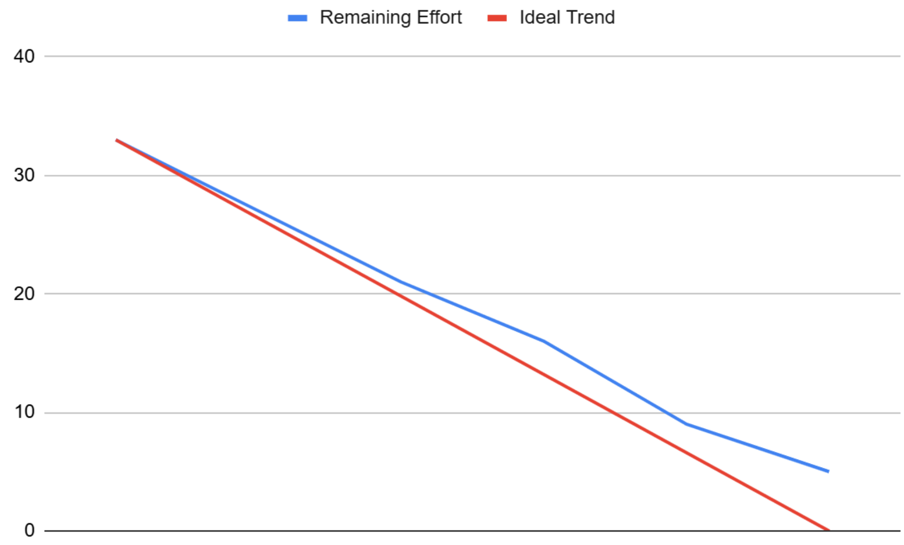
      Though we had spent time finishing those tickets in this sprint.
- *Feature: Drop down of event selection
    - Not completed due to being sick and midterm.

- *Feature: No drop down of guest
  - The rspec and cucumber scenarios are not completed due to being sick and midterm.
- *Feature: RSVP link
  - Put to backlog because the client's requirement is unclear, and wait for sprint-3 to decide how to change the user stories.

- *Feature: Referral link
  - Put to backlog because the client's requirement is unclear, and wait for sprint-3 to decide how to change the user stories.

Burndown

Sprint Burndown Chart						
User Stories	Initial Estimate					
	Day 0	Day 1	Day 2	Day 3	Day 4	Day 5
Seating Level	1	1	0	0	0	0
Seating Summary	2	1	1	0	0	0
Booked Seats column	3	0	1	1	1	0
Committed seats column	3	0	0	1	1	1
Manage Invited Guests	5	1	1	1	1	1
Manage Invited Guests - Data Validation	2	0	1	1	0	0
Manage Invited Guests - Data Validation	2	0	0	0	1	1
RSVP link	3	0	1	0	0	0
Referral link	3	0	0	0	0	0
Drop down of event selection	2	1	0	0	1	0
No drop down of guest	2	1	0	0	1	0
CRM Authentication	5	1	1	1	1	1
Remaining Effort	33	27	21	16	9	5
Ideal Trend	33	26.4	19.8	13.2	6.6	0



Documentation of Changes

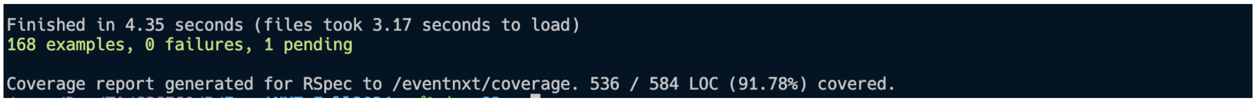
- Remove the extra authentication interface and its code.
    - We believe they were originally implemented by the previous teams for the convenience to test the app.
    - However, they do hide/remove it even in the production environment, and leave the code intervened with the code of the OAuth interface, leading to confusion both on the users and developers.
- Refactor the OAuth code, and create mock users in the development and testing environment.
    - The original OAuth code has some duplicated configuration and redundant controllers, which increase the complexity of the code base.
    - Since we removed the extra authentication interface, in order to test the app manually and run rspec/cucumber, we created some mock users which can be used to login without calling the actual API of OAuth providers.
- Remove the "test" directory, which is used for unit-tests supported by Rails natively.
    - The files are no longer needed because we currently have RSpec for unit-tests.

Customer Meeting

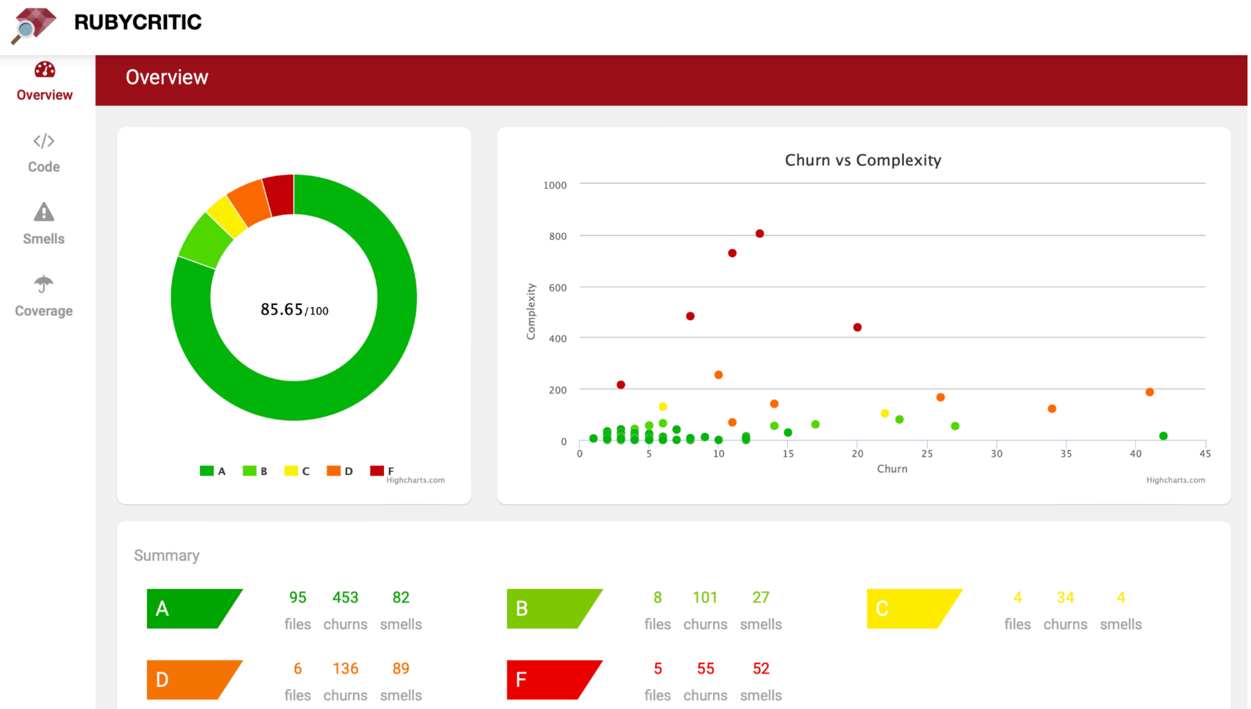
10/23/24

Customer was satisfied with the amount of work the team has completed so far. The customer still had a couple of concerns we are addressing. The customer also gave feedback to improve the features more to make it more user friendly.

Evaluations of Code and Test Quality



```
Finished in 4.35 seconds (files took 3.17 seconds to load)
168 examples, 0 failures, 1 pending

Coverage report generated for RSpec to /eventnxt/coverage. 536 / 584 LOC (91.78%) covered.
```

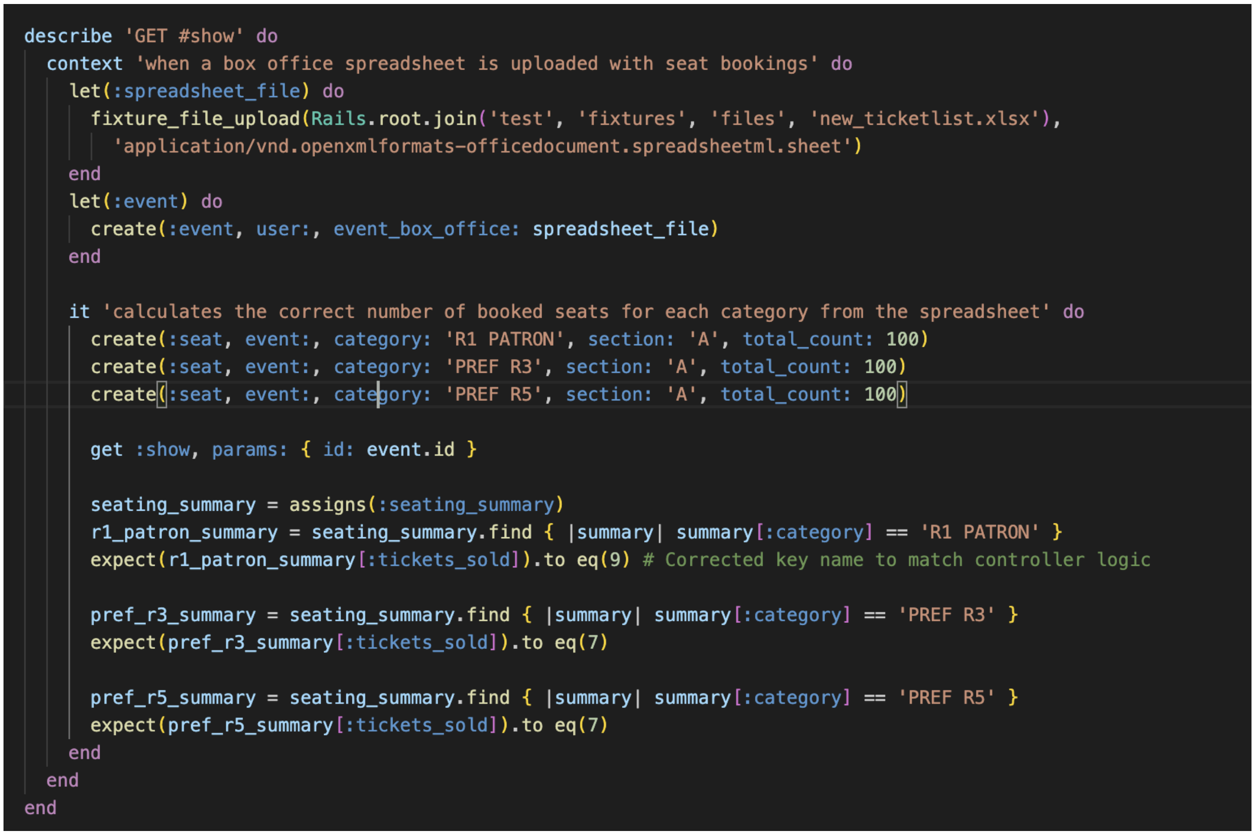


Summary

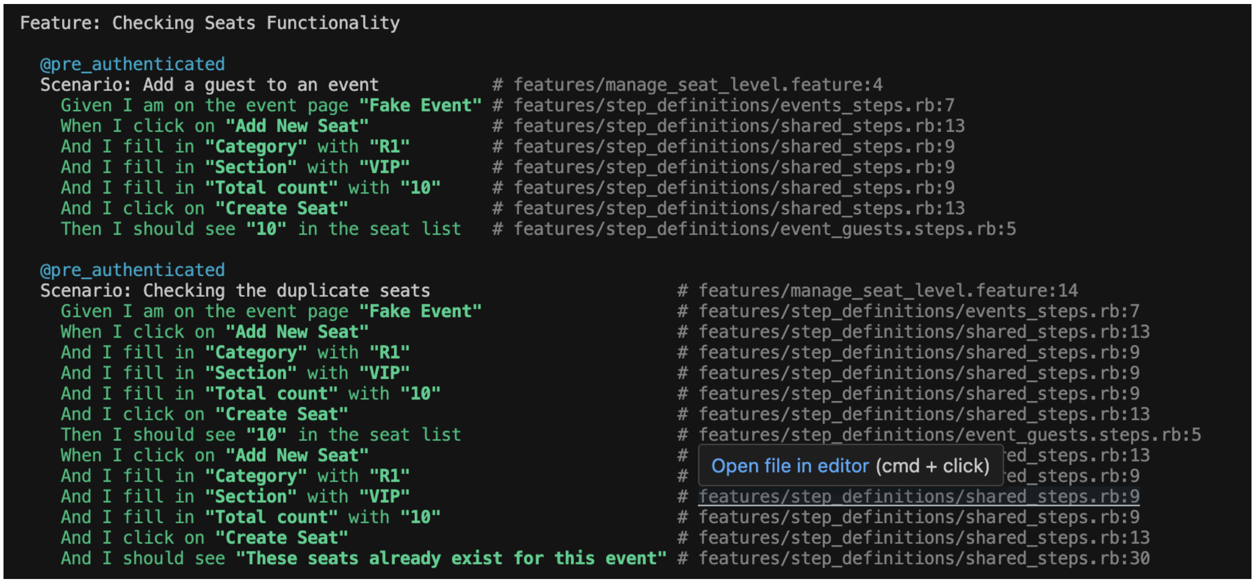
	files	churns	smells
A	95	453	82
B	8	101	27
C	4	34	4
D	6	136	89
F	5	55	52

bdd & tdd

- SEATING SUMMARY/ BOOKED SEAT COLUMN/COMMITTED SEAT COLUMN

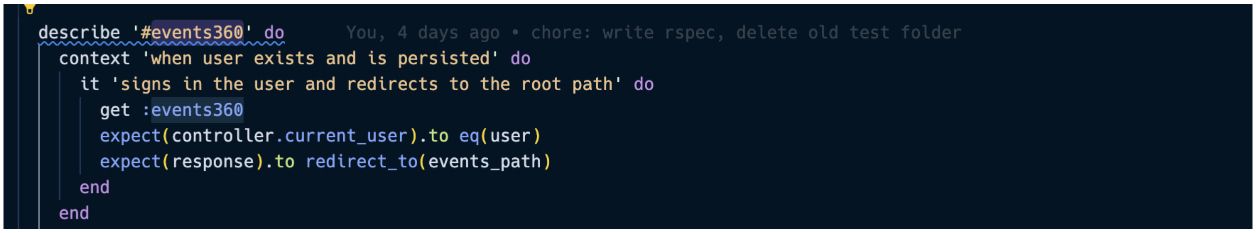


```ruby
describe 'GET #show' do
  context 'when a box office spreadsheet is uploaded with seat bookings' do
    let(:spreadsheet_file) do
      fixture_file_upload(Rails.root.join('test', 'fixtures', 'files', 'new_ticketlist.xlsx'),
        'application/vnd.openxmlformats-officedocument.spreadsheetml.sheet')
    end
    let(:event) do
      create(:event, user:, event_box_office: spreadsheet_file)
    end

    it 'calculates the correct number of booked seats for each category from the spreadsheet' do
      create(:seat, event:, category: 'R1 PATRON', section: 'A', total_count: 100)
      create(:seat, event:, category: 'PREF R3', section: 'A', total_count: 100)
      create(:seat, event:, category: 'PREF R5', section: 'A', total_count: 100)

      get :show, params: { id: event.id }

      seating_summary = assigns(:seating_summary)
      r1_patron_summary = seating_summary.find { |summary| summary[:category] == 'R1 PATRON' }
      expect(r1_patron_summary[:tickets_sold]).to eq(9) # Corrected key name to match controller logic

      pref_r3_summary = seating_summary.find { |summary| summary[:category] == 'PREF R3' }
      expect(pref_r3_summary[:tickets_sold]).to eq(7)

      pref_r5_summary = seating_summary.find { |summary| summary[:category] == 'PREF R5' }
      expect(pref_r5_summary[:tickets_sold]).to eq(7)
    end
  end
end
```

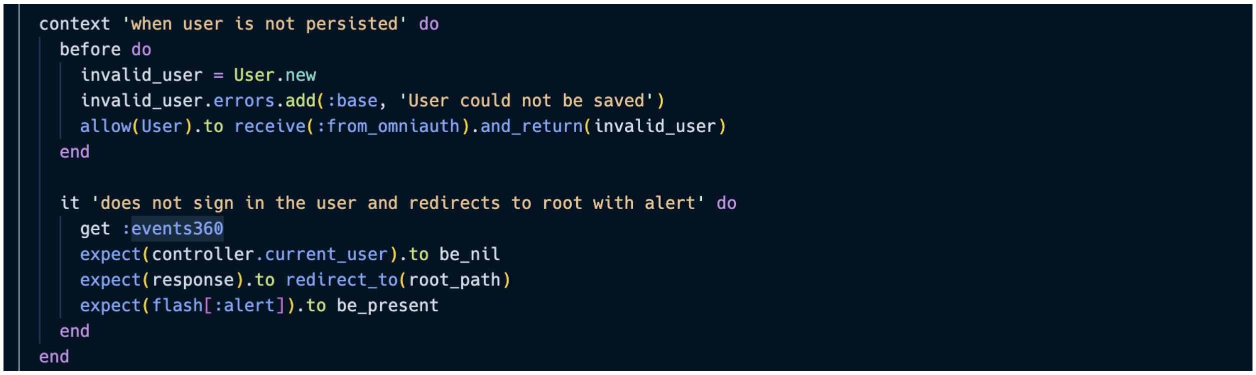


```
Feature: Checking Seats Functionality

  @pre_authenticated
  Scenario: Add a guest to an event           # features/manage_seat_level.feature:4
    Given I am on the event page "Fake Event" # features/step_definitions/events_steps.rb:7
    When I click on "Add New Seat"            # features/step_definitions/shared_steps.rb:13
    And I fill in "Category" with "R1"        # features/step_definitions/shared_steps.rb:9
    And I fill in "Section" with "VIP"        # features/step_definitions/shared_steps.rb:9
    And I fill in "Total count" with "10"     # features/step_definitions/shared_steps.rb:9
    And I click on "Create Seat"              # features/step_definitions/shared_steps.rb:13
    Then I should see "10" in the seat list   # features/step_definitions/event_guests.steps.rb:5

  @pre_authenticated
  Scenario: Checking the duplicate seats                        # features/manage_seat_level.feature:14
    Given I am on the event page "Fake Event"                   # features/step_definitions/events_steps.rb:7
    When I click on "Add New Seat"                              # features/step_definitions/shared_steps.rb:13
    And I fill in "Category" with "R1"                          # features/step_definitions/shared_steps.rb:9
    And I fill in "Section" with "VIP"                          # features/step_definitions/shared_steps.rb:9
    And I fill in "Total count" with "10"                       # features/step_definitions/shared_steps.rb:9
    And I click on "Create Seat"                                # features/step_definitions/shared_steps.rb:13
    Then I should see "10" in the seat list                     # features/step_definitions/event_guests.steps.rb:5
    When I click on "Add New Seat"                              #                              ed_steps.rb:13
    And I fill in "Category" with "R1"                          #                              ed_steps.rb:9
    And I fill in "Section" with "VIP"                          # features/step_definitions/shared_steps.rb:9
    And I fill in "Total count" with "10"                       # features/step_definitions/shared_steps.rb:9
    And I click on "Create Seat"                                # features/step_definitions/shared_steps.rb:13
    And I should see "These seats already exist for this event" # features/step_definitions/shared_steps.rb:30
```

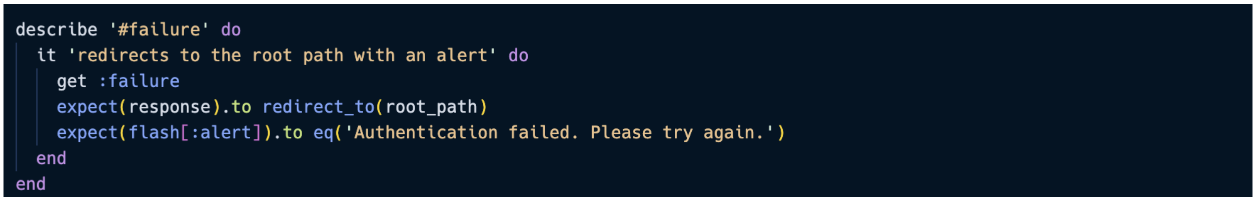
- CRM login



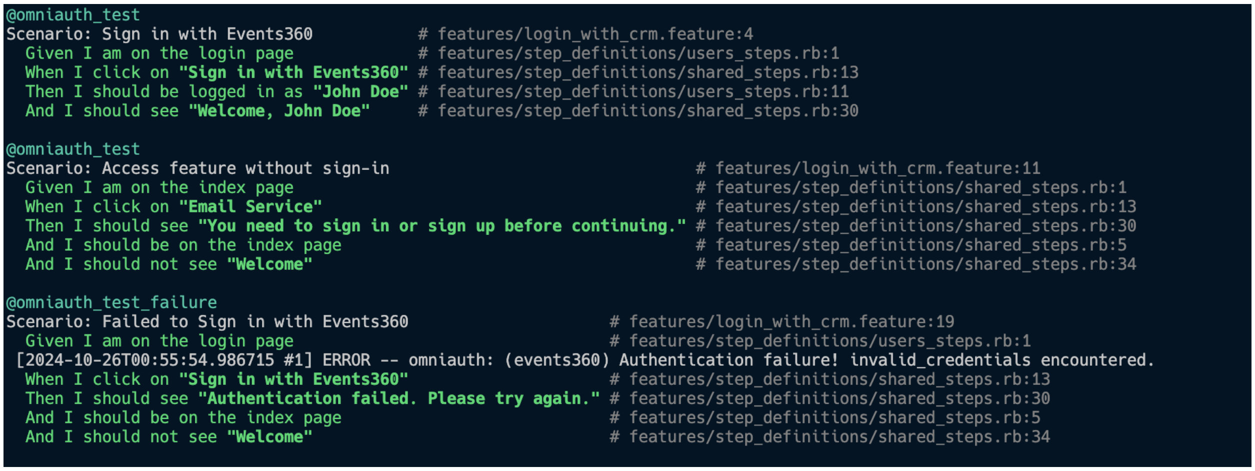
```
describe '#events360' do      You, 4 days ago • chore: write rspec, delete old test folder
  context 'when user exists and is persisted' do
    it 'signs in the user and redirects to the root path' do
      get :events360
      expect(controller.current_user).to eq(user)
      expect(response).to redirect_to(events_path)
    end
  end
```



```
  context 'when user is not persisted' do
    before do
      invalid_user = User.new
      invalid_user.errors.add(:base, 'User could not be saved')
      allow(User).to receive(:from_omniauth).and_return(invalid_user)
    end

    it 'does not sign in the user and redirects to root with alert' do
      get :events360
      expect(controller.current_user).to be_nil
      expect(response).to redirect_to(root_path)
      expect(flash[:alert]).to be_present
    end
  end
```

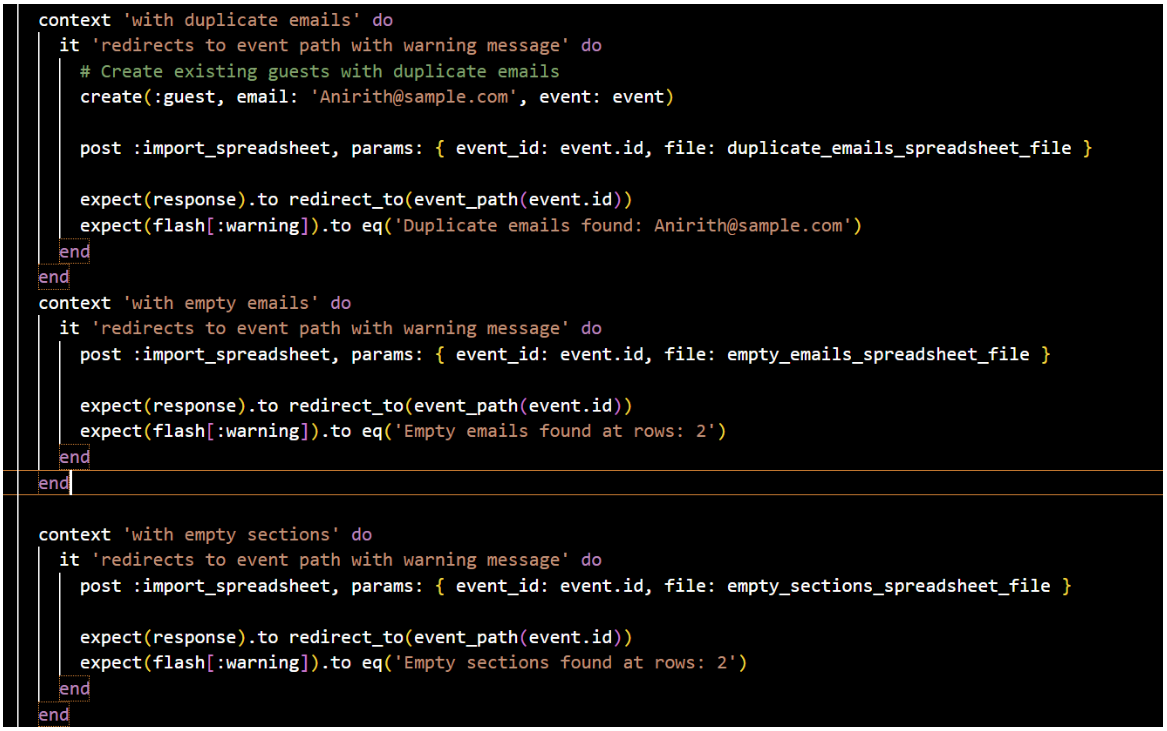


```
describe '#failure' do
  it 'redirects to the root path with an alert' do
    get :failure
    expect(response).to redirect_to(root_path)
    expect(flash[:alert]).to eq('Authentication failed. Please try again.')
  end
end
```

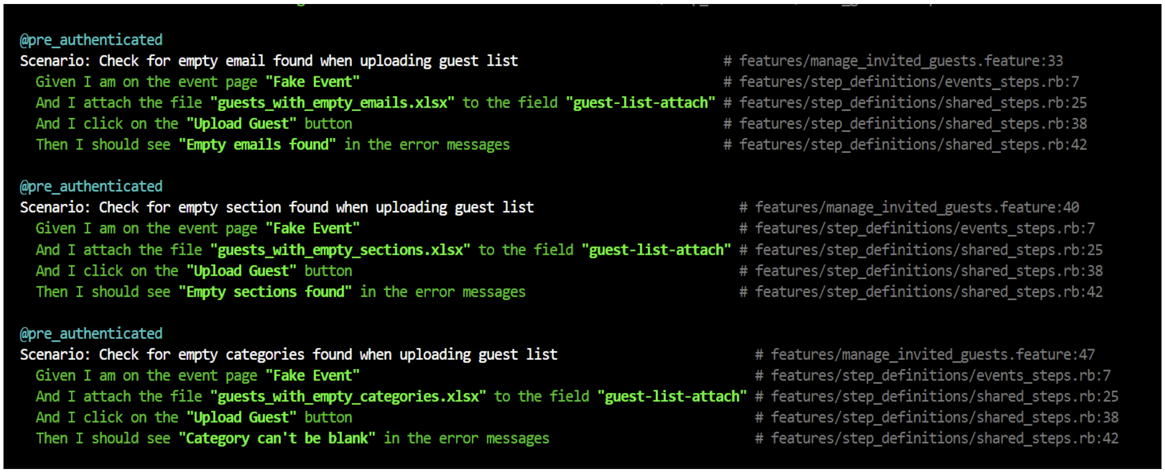


```
@omniauth_test
Scenario: Sign in with Events360          # features/login_with_crm.feature:4
  Given I am on the login page            # features/step_definitions/users_steps.rb:1
  When I click on "Sign in with Events360" # features/step_definitions/shared_steps.rb:13
  Then I should be logged in as "John Doe" # features/step_definitions/users_steps.rb:11
  And I should see "Welcome, John Doe"     # features/step_definitions/shared_steps.rb:30

@omniauth_test
Scenario: Access feature without sign-in                                  # features/login_with_crm.feature:11
  Given I am on the index page                                            # features/step_definitions/shared_steps.rb:1
  When I click on "Email Service"                                         # features/step_definitions/shared_steps.rb:13
  Then I should see "You need to sign in or sign up before continuing."   # features/step_definitions/shared_steps.rb:30
  And I should be on the index page                                       # features/step_definitions/shared_steps.rb:5
  And I should not see "Welcome"                                          # features/step_definitions/shared_steps.rb:34

@omniauth_test_failure
Scenario: Failed to Sign in with Events360                      # features/login_with_crm.feature:19
  Given I am on the login page                                  # features/step_definitions/users_steps.rb:1
 [2024-10-26T00:55:54.986715 #1] ERROR -- omniauth: (events360) Authentication failure! invalid_credentials encountered.
  When I click on "Sign in with Events360"                      # features/step_definitions/shared_steps.rb:13
  Then I should see "Authentication failed. Please try again."  # features/step_definitions/shared_steps.rb:30
  And I should be on the index page                             # features/step_definitions/shared_steps.rb:5
  And I should not see "Welcome"                                # features/step_definitions/shared_steps.rb:34
```

- Manage invited guests and data validations



```
context 'with duplicate emails' do
  it 'redirects to event path with warning message' do
    # Create existing guests with duplicate emails
    create(:guest, email: 'Anirith@sample.com', event: event)

    post :import_spreadsheet, params: { event_id: event.id, file: duplicate_emails_spreadsheet_file }

    expect(response).to redirect_to(event_path(event.id))
    expect(flash[:warning]).to eq('Duplicate emails found: Anirith@sample.com')
  end
end
context 'with empty emails' do
  it 'redirects to event path with warning message' do
    post :import_spreadsheet, params: { event_id: event.id, file: empty_emails_spreadsheet_file }

    expect(response).to redirect_to(event_path(event.id))
    expect(flash[:warning]).to eq('Empty emails found at rows: 2')
  end
end

context 'with empty sections' do
  it 'redirects to event path with warning message' do
    post :import_spreadsheet, params: { event_id: event.id, file: empty_sections_spreadsheet_file }

    expect(response).to redirect_to(event_path(event.id))
    expect(flash[:warning]).to eq('Empty sections found at rows: 2')
  end
end
```



```
@pre_authenticated
Scenario: Check for empty email found when uploading guest list                          # features/manage_invited_guests.feature:33
  Given I am on the event page "Fake Event"                                              # features/step_definitions/events_steps.rb:7
  And I attach the file "guests_with_empty_emails.xlsx" to the field "guest-list-attach" # features/step_definitions/shared_steps.rb:25
  And I click on the "Upload Guest" button                                               # features/step_definitions/shared_steps.rb:38
  Then I should see "Empty emails found" in the error messages                           # features/step_definitions/shared_steps.rb:42

@pre_authenticated
Scenario: Check for empty section found when uploading guest list                          # features/manage_invited_guests.feature:40
  Given I am on the event page "Fake Event"                                                # features/step_definitions/events_steps.rb:7
  And I attach the file "guests_with_empty_sections.xlsx" to the field "guest-list-attach" # features/step_definitions/shared_steps.rb:25
  And I click on the "Upload Guest" button                                                 # features/step_definitions/shared_steps.rb:38
  Then I should see "Empty sections found" in the error messages                           # features/step_definitions/shared_steps.rb:42

@pre_authenticated
Scenario: Check for empty categories found when uploading guest list                         # features/manage_invited_guests.feature:47
  Given I am on the event page "Fake Event"                                                  # features/step_definitions/events_steps.rb:7
  And I attach the file "guests_with_empty_categories.xlsx" to the field "guest-list-attach" # features/step_definitions/shared_steps.rb:25
  And I click on the "Upload Guest" button                                                   # features/step_definitions/shared_steps.rb:38
  Then I should see "Category can't be blank" in the error messages                          # features/step_definitions/shared_steps.rb:42
```