Sistema Tickents

Documentação de Testes

Tácio Breno Santos Almeida

01/12/2019

1. Análise do Sistema
   1. Introdução: o propósito do Tickents é ser uma plataforma para comprar ingressos de eventos variados, desde shows musicais até palestras acadêmicas, onde um administrador cadastra cada evento, assim como local e capacidade de pessoas dos mesmos, enquanto o usuário pode acessar todos os eventos cadastrados para ver mais informações e comprar ingressos.
   2. Requisitos funcionais:

|  |  |  |
| --- | --- | --- |
| Código | Título | Descrição |
| RF001 | Cadastro de local do evento | O sistema deve prover um cadastro de espaço com lotação |
| RF002 | Cadastro do evento | O sistema deverá prover cadastro de eventos, o cadastro de eventos deve ser feito para um espaço especifico, com um nome, data e hora e valor |
| RF003 | Cadastro do usuário | O sistema deve prover o cadastro de cliente, com nome, cpf, telefone |
| RF004 | Compra de ingresso | O cliente deve poder comprar um ingresso, gerado com um código único |

1.3 Requisitos não-funcionais:

|  |  |  |
| --- | --- | --- |
| Código | Título | Descrição |
| RNF001 | Usuário administrador | A criação de um usuário administrador no banco que seja o único com as permissões de cadastrar local e evento |
| RNF002 | Número do cartão de crédito | A compra de ingresso só poderá ser feita com o número de cartão 4000 0000 0000 0010, qualquer outro número deve retornar um erro |

1. Planejamento dos testes
   1. Cronograma:

|  |  |
| --- | --- |
| Teste | Duração |
| Validação dos campos de local do evento | 2 minutos |
| Validação dos campos de evento | 5 minutos |
| Validação dos campos de usuário | 5 minutos |
| Validação dos campos de compra | 3 minutos |

2.2 Recursos:

* Hardware - computador desktop ou notebook;
* Software - framework symfony, composer, php, doctrine, mysql e git;
* Pessoas - um dos desenvolvedores envolvidos.

1. Execução dos testes
   1. Validação dos campos de local do evento:

|  |  |
| --- | --- |
| Código | CT001 |
| Objetivo do teste | Garantir que os campos validam apenas os dados utilizáveis |
| Passos | 1. Entrar com o login de administrador; 2. Acessar a página de cadastro de local do evento através da barra de navegação; 3. Cadastrar um local com dados que não devem ser permitidos. |
| Resultado esperado | O local do evento apenas deve ser cadastro com no mínimo três caracteres e ao menos uma letra entre eles, sua capacidade só pode conter números, com um espectro de 50 a 50.000 |

3.2 Validação dos campos de evento:

|  |  |
| --- | --- |
| Código | CT002 |
| Objetivo do teste | Garantir que os campos validam apenas os dados utilizáveis |
| Passos | 1. Entrar com o login de administrador; 2. Acessar a página de cadastro do evento através da barra de navegação; 3. Cadastrar um evento com dados que não devem ser permitidos. |
| Resultado esperado | O nome do evento apenas deve ser cadastro com no mínimo três caracteres e ao menos uma letra entre eles, seu horário não pode ser anterior ao dia seguinte, a descrição com no máximo 255 caracteres e o valor precisa ser flutuante |

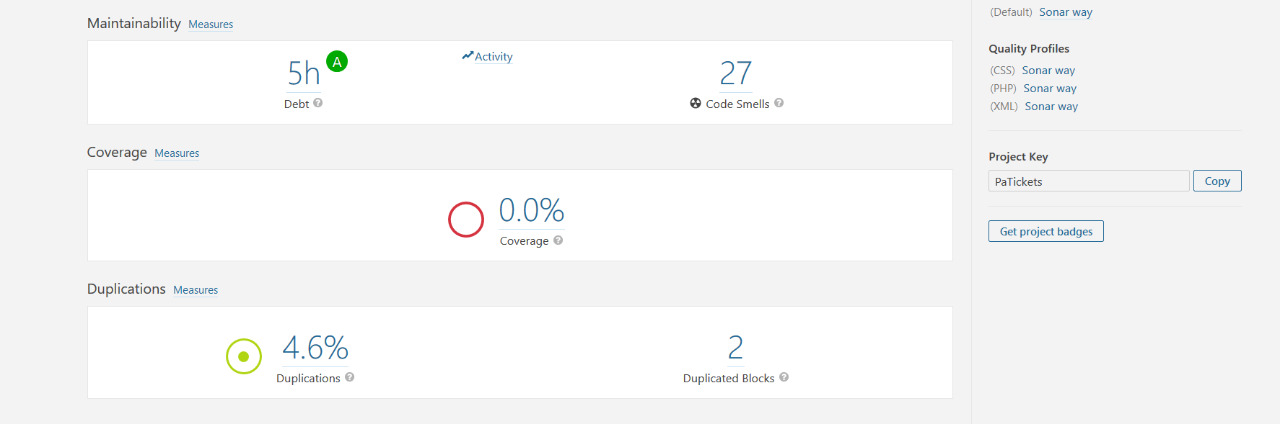
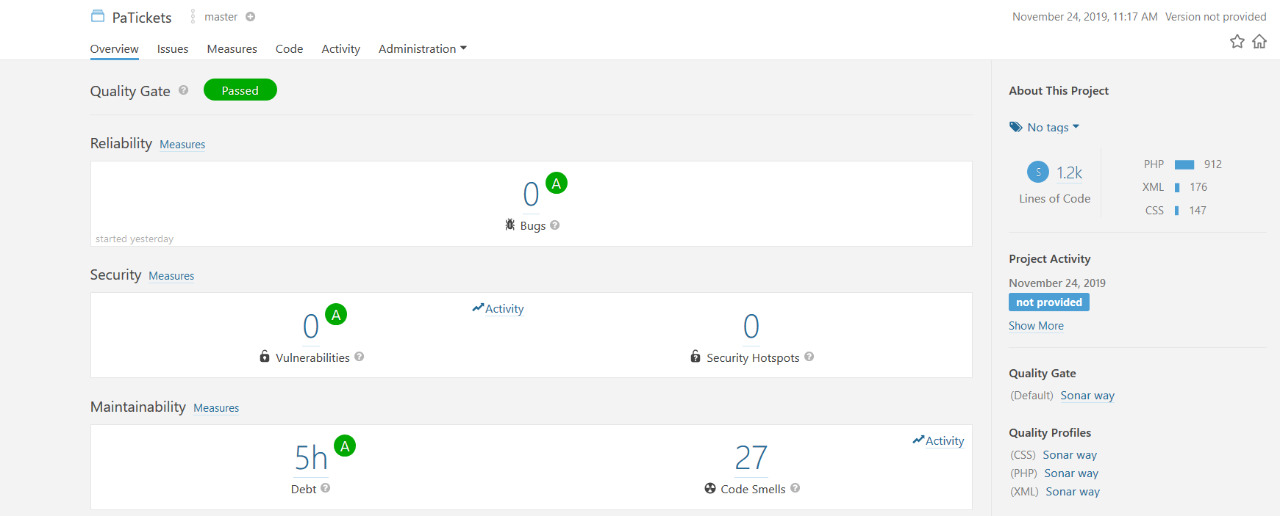
3.3 Validação dos campos de usuário:

|  |  |
| --- | --- |
| Código | CT003 |
| Objetivo do teste | Garantir que os campos validam apenas os dados utilizáveis |
| Passos | 1. Acessar a página de login de através da barra de navegação; 2. Acessar a página de cadastro através do link “Crie sua conta”; 3. Cadastrar um usuário com dados que não devem ser permitidos |
| Resultado esperado | O usuário deve ser cadastrado com um nome composto por ao menos duas palavras, cada uma com no mínimo três letras e nenhum número ou caractere especial, o email seguindo o padrão “[exemplo@email.com](mailto:exemplo@email.com)”, o cpf no formato “999.999.999-99” e o telefone com o molde “(99) 99999-9999” |

3.4 Validação dos campos de compra:

|  |  |
| --- | --- |
| Código | CT004 |
| Objetivo do teste | Garantir que os campos validam apenas os dados utilizáveis |
| Passos | 1. Acessar uma página de evento de através de um dos cards; 2. Clicar no botão verde “Comprar Aqui!; 3. Fazer uma compra com dados que não devem ser permitidos |
| Resultado esperado | A compra só pode ser concluída com o número de cartão 4000 0000 0000 0010, o código de segurança deve ser composto por três números e a validade não pode estar vencida |

1. Falhas identificadas
   1. Validação dos campos de local do evento: é cadastrado o nome com menos de três caracteres e a capacidade não está no espectro especificado;
   2. Validação dos campos de evento: é cadastrado com qualquer horário e o valor não aceita número flutuante;
   3. Validação dos campos de usuário: é cadastrado com nome composto por somente uma palavra, cpf e telefone ainda não estão restritos aos formatos indicados;
   4. Validação dos campos de compra: é concluída com um código de segurança acima e abaixo de três números, além disso, qualquer data de validade é aceita.
2. Ferramenta utilizada
   1. Descrição: foi utilizado o Sonar para coletar dados de análise de código, por ser bastante recomendada e trazer resultados muito significativos e precisos;
   2. Resultados:



5.3 Conclusões: as considerações feitas a partir das observações do Sonar são positivas, demonstrando que temos um código legível e seguro em mãos, os problemas explicitados pela ferramenta são pequenos e não será custoso resolvê-los, sendo que a prioridade seria diminuir as duplicações, já que muito dos code smells encontrados não são de fato negativos para o sistema.