



# Desenvolvimento de Sistemas

---

## Ferramentas de testes: *bugtrackers* e automação

### Introdução

Como você já sabe, por meio dos testes é possível reduzir a probabilidade de defeitos não descobertos permanecerem no *software*. Para que esses defeitos sejam encontrados de forma antecipada, as atividades de testes devem iniciar o mais cedo possível no ciclo de vida de desenvolvimento de *software*. Além disso, para a realização de testes, existem ferramentas específicas para os tipos de *softwares*/sistemas e contextos. Por meio dessas ferramentas, os testes são executados, os resultados são reportados e as comparações com testes anteriores são realizadas, podendo serem executados os testes repetidamente em qualquer horário do dia.

A seguir, você verá o que são ferramentas de *bugtrackers* e o que são ferramentas de automação.

### Ferramentas de *bugtrackers*

As ferramentas de *bugtrackers* são de extrema importância para os testes. Serão abordados, aqui, o conceito de *bugtracker* e as ferramentas que são utilizadas para essa finalidade.

Você sabe o que são *bugtrackers*?

Eles têm como finalidade auxiliar a equipe de teste e de desenvolvedores a manter o histórico de todos os *bugs* que ocorreram no *software*. É nesse tipo de ferramenta que todos os dados referentes aos *bugs* encontrados são colocados. É um meio formal de sinalizar aos desenvolvedores que os *bugs* existem. Ocorre da seguinte forma: os *bugs* encontrados são cadastrados na ferramenta, para que, a partir disso, as equipes possam gerenciá-los.

# Ferramentas de automação



A automação de testes está em expansão e, diferente do que os leigos acreditam, não é um processo de testes em si, mas, sim, parte de um processo maior. A implantação de automação geralmente falha quando os testes são imaturos e as responsabilidades não são bem definidas, até mesmo por não terem profissionais suficientemente qualificados ou, ainda, por optarem por ferramentas sem ter a maturidade suficiente para o processo de testes.

O objetivo da automação de testes é a redução do envolvimento humano em atividades repetitivas e manuais, e o grande benefício em automatizar os testes está em aumentar a amplitude e a profundidade da cobertura de testes a serem realizados.

Antes da escolha de uma ferramenta de testes, tanto automatizados quanto manuais, deve-se levar em consideração a criação de uma prova de conceito. Além disso, deve-se ter certeza de que não ocorrem erros críticos e utilizar sempre a versão mais atualizada da ferramenta, realizando teste de regressão. Outros fatores muito importantes são o dimensionamento e a infraestrutura adequados, pois as ferramentas necessitam de computadores com alto desempenho, processadores rápidos e memória grande, os quais devem ser exclusivos para a automação de testes.

Lembrem-se de que a automação não substitui os testes manuais. Ela deve ser um adicional para agregar valor.

Agora, serão apresentadas algumas das ferramentas mais utilizadas em testes, tanto manuais quanto automatizados, ferramentas comerciais e gratuitas.

## Ferramentas de *bugtrackers*

### Selenium

Para testes automatizados relacionados a aplicações *web* pelo *browser*, há a ferramenta Selenium, utilizada para teste de aceitação, teste funcional, teste de *performance*/desempenho, teste de carga, teste de estresse, teste regressivo, desenvolvimento orientado a testes e desenvolvimento orientado a comportamento. É uma ferramenta gratuita e que fornece recursos de reprodução de gravação. Várias linguagens de programação podem ser escritas por testadores, por exemplo, JavaScript, PHP (*hypertext preprocessor*, ou pré-processador de hipertexto), Python, Ruby, Java, Perl, Groovy e C#.

O *download* da ferramenta pode ser realizado no *site* da Selenium (o qual você encontrar digitando “Selenium” no seu buscador), na aba **Download**. Já a documentação pode ser encontrada no mesmo *site*, clicando em **Documentação – Diretrizes – Tipos de teste**.

A ferramenta é uma extensão do Firefox e depende do *browser* para funcionar. Além disso, também existe disponível o núcleo da Selenium para que seja executado independentemente de plataforma e do Firefox, no qual é disponibilizado somente o JSFTR (JavaScript Functional Test Runner).

Após a instalação do Firefox, o próximo passo é fazer o *download* do Selenium IDE (*integrated development environment*). Para realizar o *download* da extensão, basta clicar em **Extensões** no navegador (ícone em formato de uma peça de quebra-cabeça) e digitar “Selenium” no campo **Encontrar mais extensões**, conforme apresentado na figura 1. Como resultado, será exibida uma tela com as extensões pertinentes ao que foi digitado, conforme apresentado na figura 2.

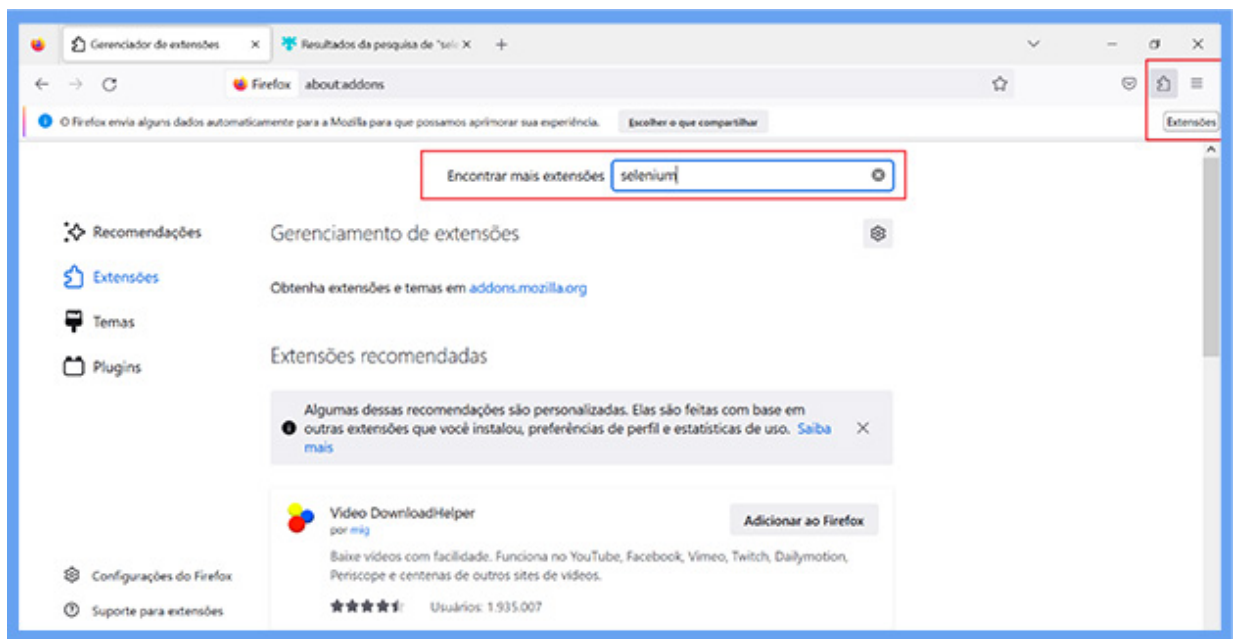


Figura 1 – Exemplo de como encontrar a extensão Selenium IDE no gerenciador de extensões do navegador Firefox

Fonte: Senac EAD (2023)

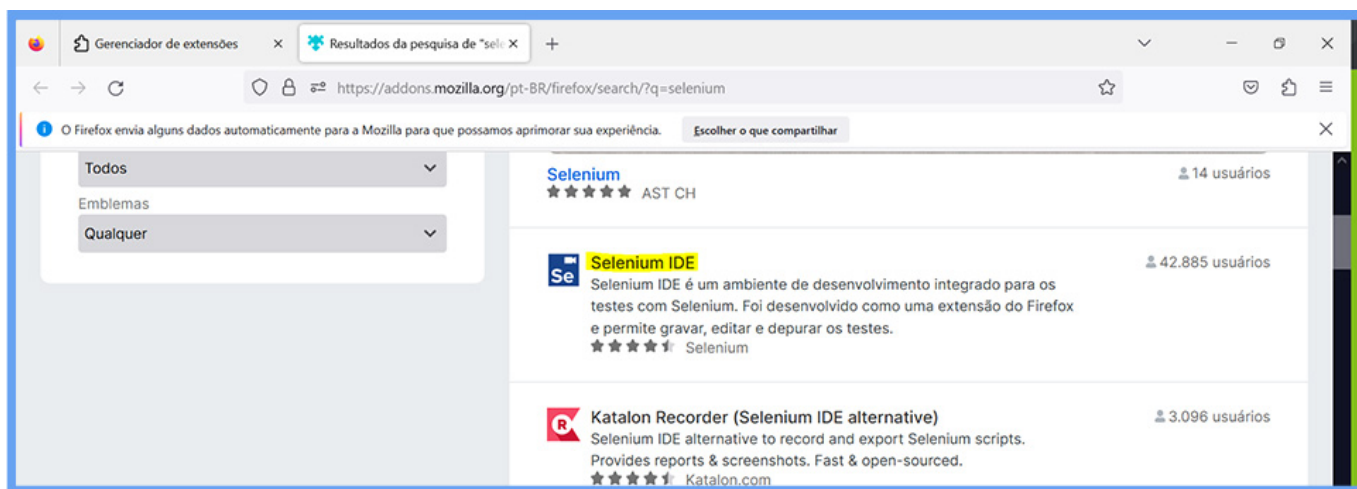


Figura 2 – Extensão Selenium IDE no gerenciador de extensões do navegador Firefox

Fonte: Senac EAD (2023)

A extensão será apresentada para ser realizado o *download* conforme apresentado na figura 3, bastando clicar em **Adicionar ao Firefox**.

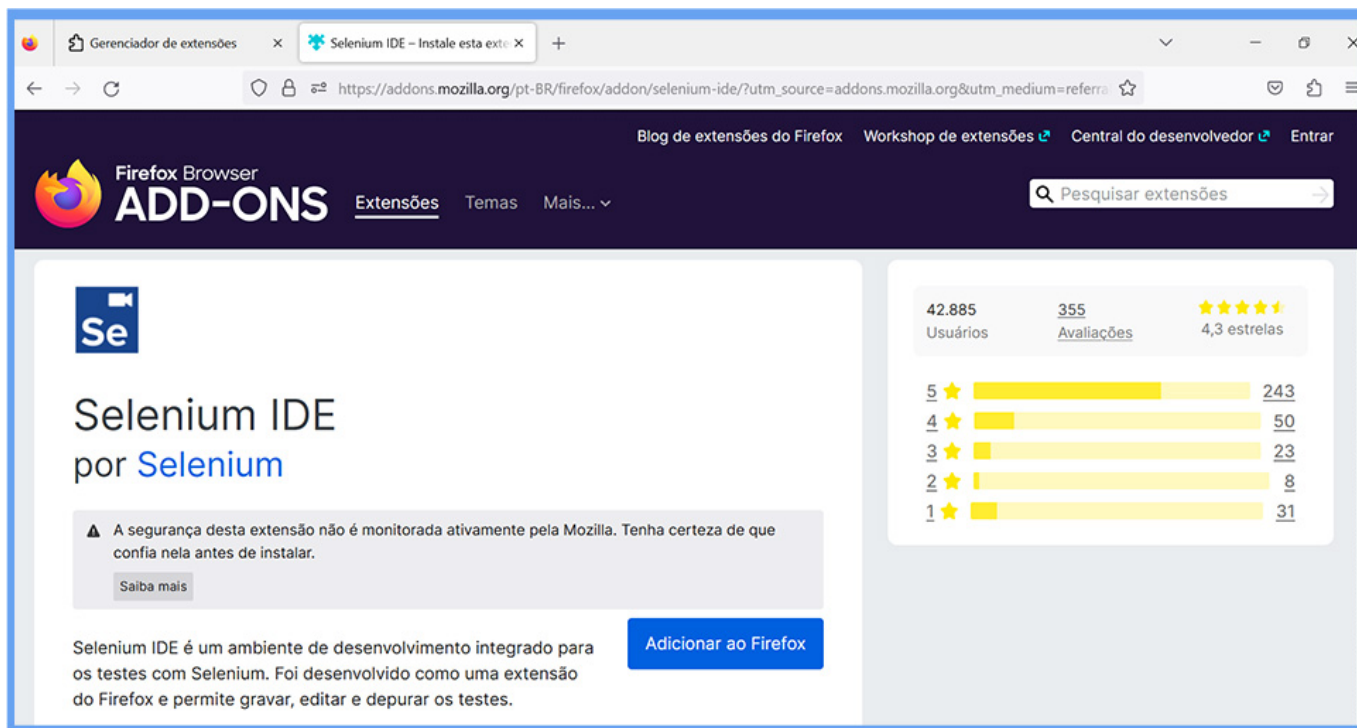


Figura 3 – Extensão Selenium IDE no gerenciador de extensões do navegador Firefox

Fonte: Firefox Browser (2022)

Em seguida, é necessário acessar a extensão por meio do ícone em formato de uma peça de quebra-cabeça e selecionar o que deseja fazer ao executá-la. Assim que estiver instalada, a extensão Selenium IDE está pronta para ser utilizada, assim como é demonstrado na figura 4. Ao selecionar essa opção, será visualizada a interface de configuração dos testes do Selenium IDE, onde é possível optar por gravar um novo teste de um novo projeto, abrir um projeto existente, criar um novo projeto ou simplesmente fechar a extensão.

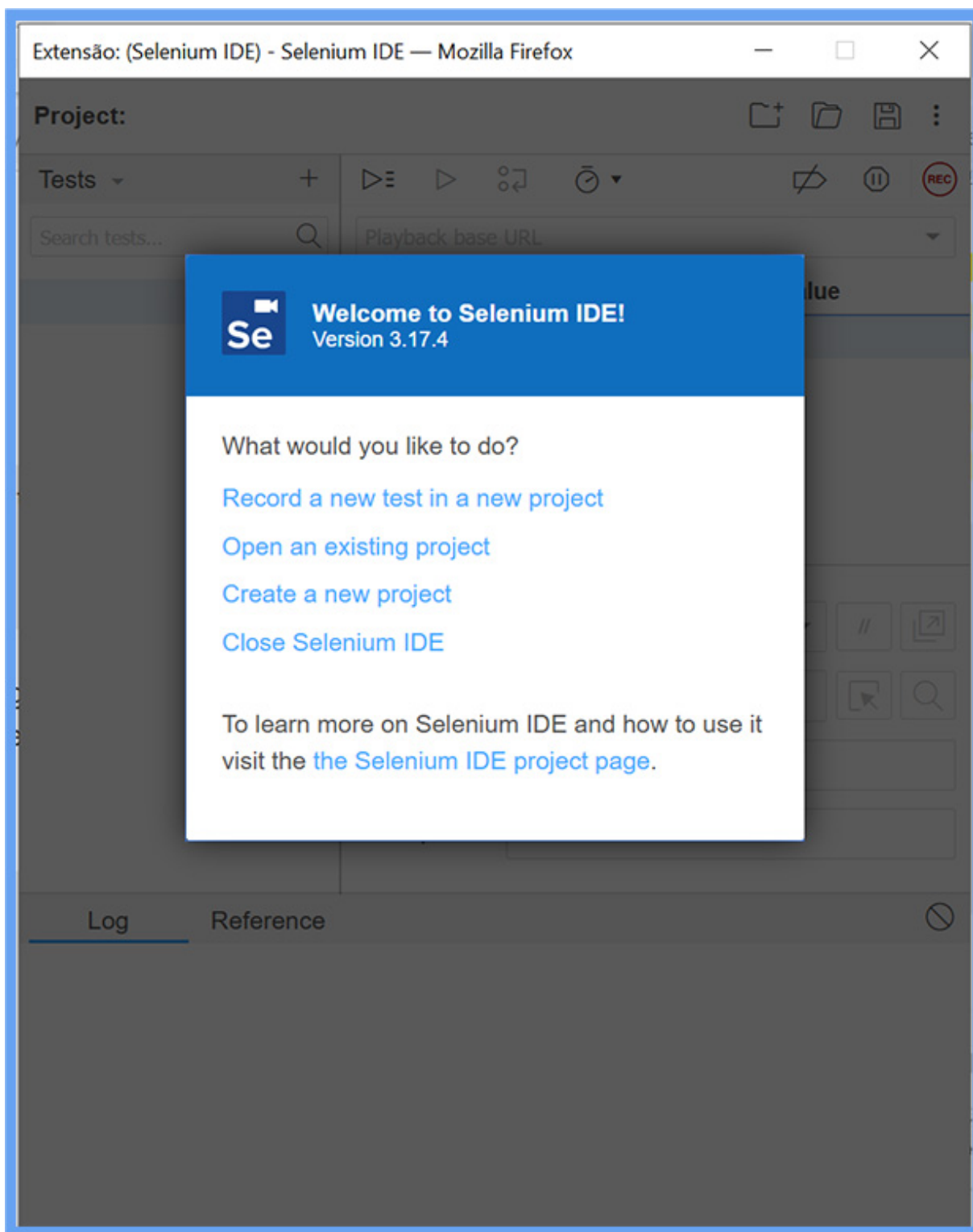


Figura 4 – Exemplo da ferramenta Selenium IDE após clicar para a sua execução

Fonte: Senac EAD (2023)

## ZipBoard

O zipBoard é uma ferramenta de rastreamento de *bugs* e *feedback* visual que permite aos desenvolvedores *web* trabalharem de maneira colaborativa em seus produtos e projetos, sejam eles *web apps*, sejam *sites*, entre outros. Essa ferramenta pode ser utilizada durante a criação de

novos *sites*, *e-commerce*, ou ainda para melhorias que sejam pertinentes a produtos já existentes, de forma que auxilie no gerenciamento de tarefas. É uma ferramenta gratuita para utilizar durante 15 dias. Após esse período, torna-se paga.

Algumas de suas características são: utilização *on-line* por meio do navegador sem qualquer necessidade de instalação de *software*; teste de usuário integrado e plataforma de rastreamento de *bugs*; relatórios de erros visuais, nos quais é possível reproduzir *bugs* facilmente e com a utilização de relatórios de *bugs* detalhados dispondo de informações relevantes; medição de desempenho; teste de usuário em que é possível coletar entradas de usuários e clientes sem sair do conteúdo; teste de *design* responsivo incluindo teste de capacidade de resposta do *site*; navegação em diferentes resoluções; capturas de telas; e arquivos *bugs* para resolução específica diretamente do navegador.

A ferramenta pode ser acessada no *site* do zipBoard (digite “zipBoard *bug tracking*” em seu buscador). Além disso, a inscrição é feita no próprio *site* e de forma gratuita, sem a necessidade de adição de cartão de crédito para o período pós-teste.

Na figura 5, é apresentado o *layout* da ferramenta, em que é possível cadastrar novos projetos e novas tarefas, deixar mensagens e verificar reportes. Também há a possibilidade de realizar integrações com outras plataformas, por exemplo, Jira, Slack, Microsoft Teams e Wrike para sincronização automatizada.

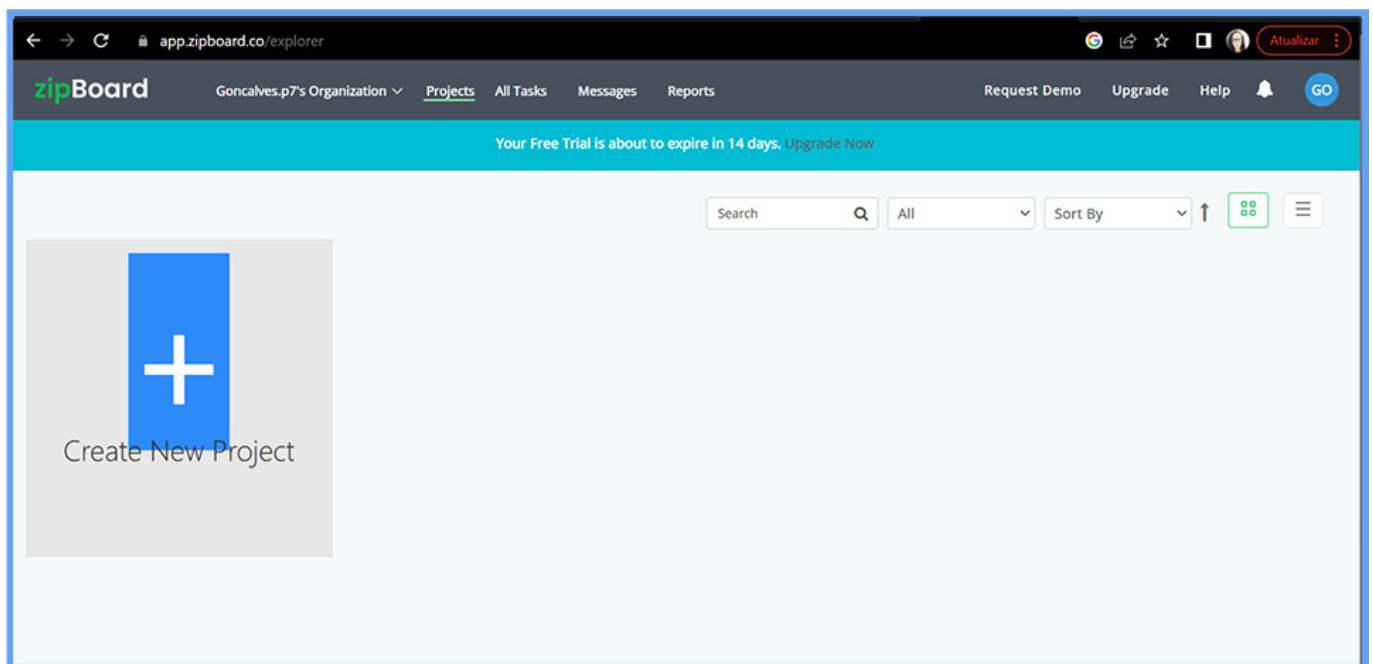


Figura 5 – Exemplo da ferramenta zipBoard

Fonte: Senac EAD (2023)

## Jira Software

Jira é a principal ferramenta de rastreamento de *bugs*, rastreamento de problemas e gerenciamento de projetos. O Jira combina rastreamento de problemas, gerenciamento de projetos ágil, fluxo de trabalho personalizável e uma estrutura de integração por meio de *plugins* para aumentar a velocidade de sua equipe de desenvolvimento de *software*. Além disso, o Jira é gratuito para equipes de até 10 pessoas e centraliza relatórios de *bugs*, com a possibilidade de adicionar contexto a cada tarefa com rótulos, campos personalizados, anexos, entre outros.

Há a possibilidade de eliminar os *bugs* antes que problemas maiores ocorram, priorizando e atribuindo tarefas de acordo com a urgência em que se apresentam.

Ainda sobre as questões de *bugs*, é possível utilizar um fluxo de trabalho criado especificamente para a resolução deles, mantendo o time com foco no que for preciso. Na figura 6, é apresentado um exemplo de um quadro em que constam os *bugs* mapeados (todos os que têm o ícone quadrado vermelho com um círculo branco ao centro).

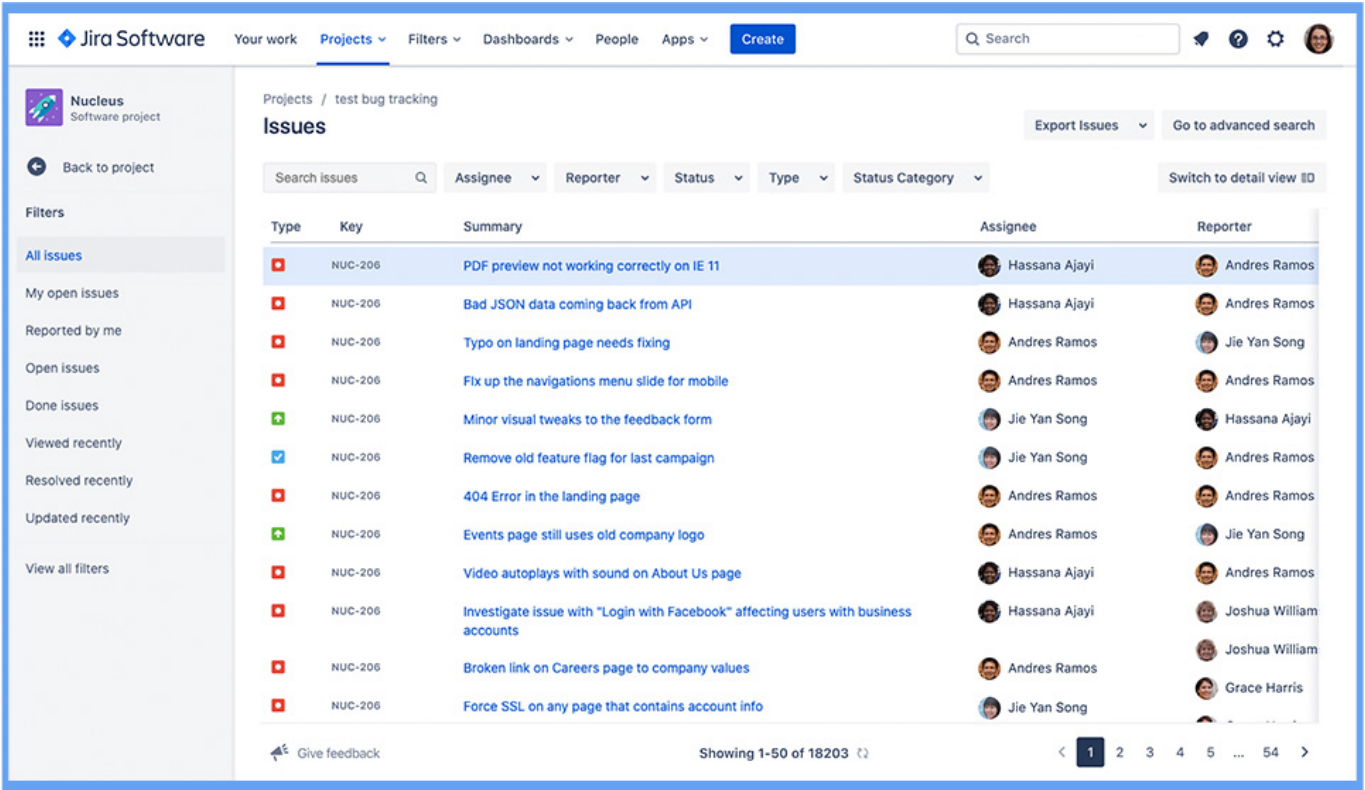


Figura 6 – Exemplo da ferramenta Jira

Fonte: Atlassian (c2023b)

Para iniciar os trabalhos no Jira, é necessária a criação da conta na plataforma. Ao acessar o endereço, aparecerá a tela apresentada na figura 7, na qual se pode escolher se a preferência é por criar uma conta com um *e-mail* ou, ainda, utilizar o Google.





Figura 7 – Exemplo da ferramenta Jira

Fonte: Atlassian (c2023c)

Após o cadastro (você precisa optar por uma das duas formas), aparecerá a tela de boas-vindas, na qual você disponibilizará o nome de seu *site*, conforme apresentado na figura 8 a seguir.

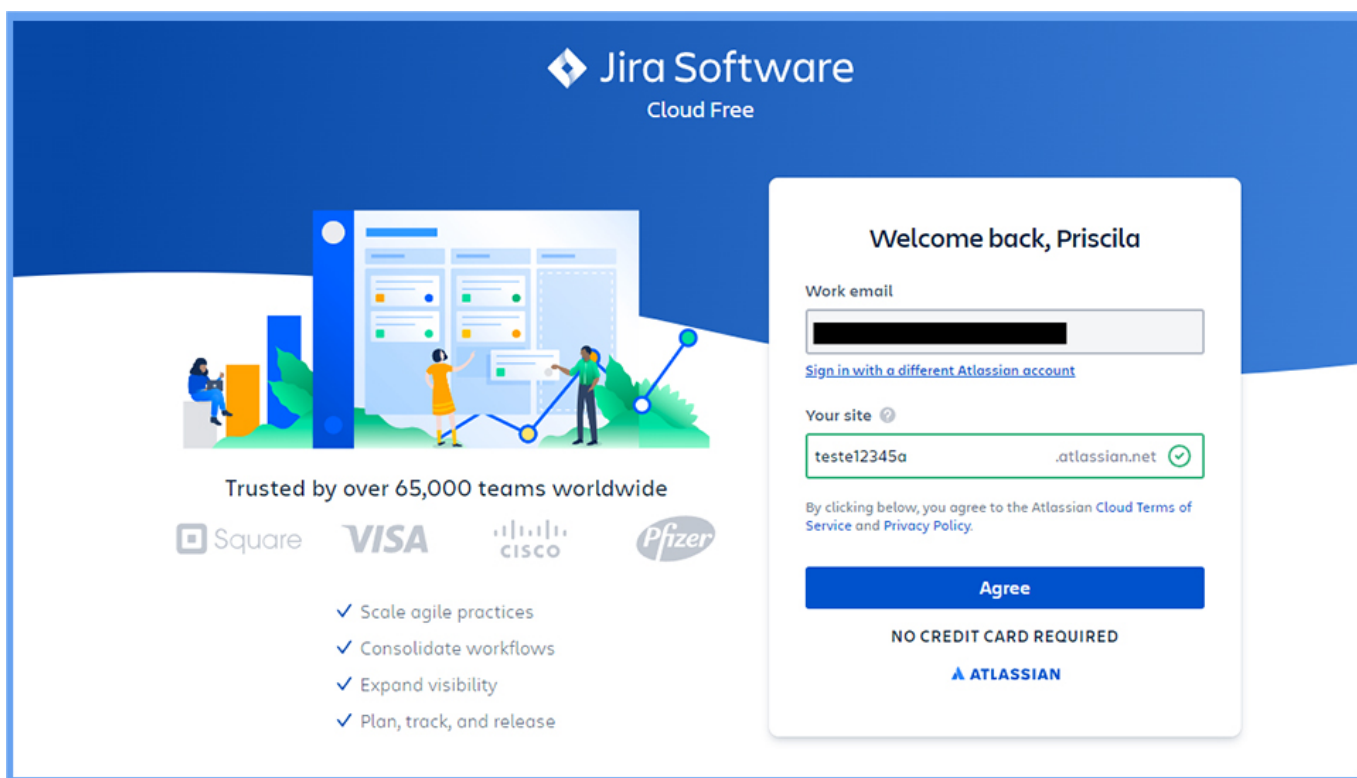


Figura 8 – Exemplo da ferramenta Jira

Fonte: Senac EAD (2023)



Após inserir as informações, a plataforma inicializará seu *board* e você precisará aguardar alguns minutos, conforme apresentado na figura 9.

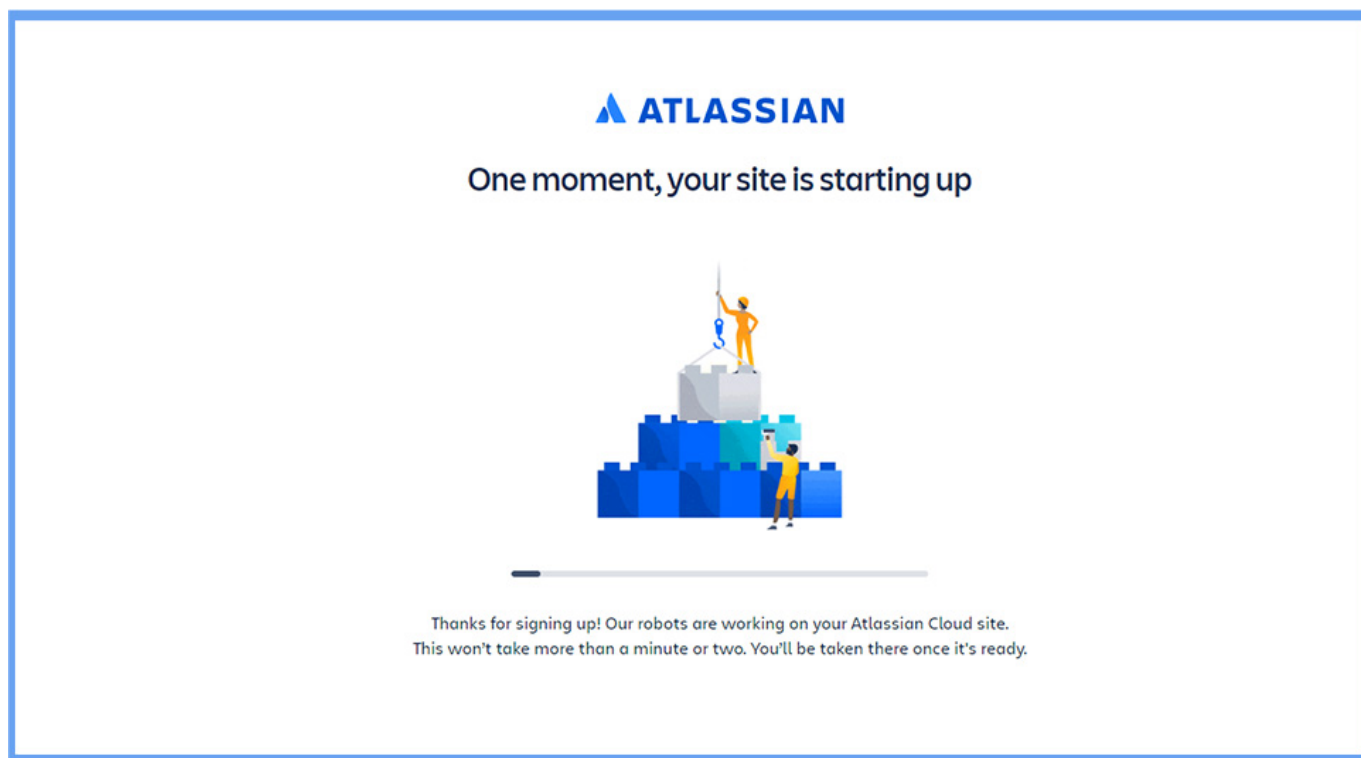


Figura 9 – Tela de criação do *board*

Fonte: Senac EAD (2023)

Os próximos passos são: adicionar as informações referentes ao projeto no qual está sendo utilizado o *template* de controle de *bugs* escolhido por você anteriormente, criar a chave que identificará o projeto e, finalmente, clicar em **Criar projeto**, conforme apresentado na figura 10.

Adicionar informações do projeto

Você pode alterar essas informações a qualquer momento nas configurações do seu projeto.

Nome

Teste 1234a

Chave

T1

Template

Alterar template

Controle de bugs

Gerencie listas de tarefas e bugs de desenvolvimento.

Criar projeto

Figura 10 – *Template* de adição de informações e finalização da criação do projeto

Fonte: Senac EAD (2023)

Há, ainda, a possibilidade de utilizar outras ferramentas que serão integradas e conectadas posteriormente, por exemplo, Slack (mensageiro), Microsoft Teams (mensageiro), GitHub (repositório de códigos), GitLab (repositório de códigos), Jenkins, entre outros.

Pronto! Agora você pode utilizar o *board* criado para o seu projeto. Para criar o primeiro item, basta clicar em **Criar** no menu superior, conforme demonstrado na figura 11.

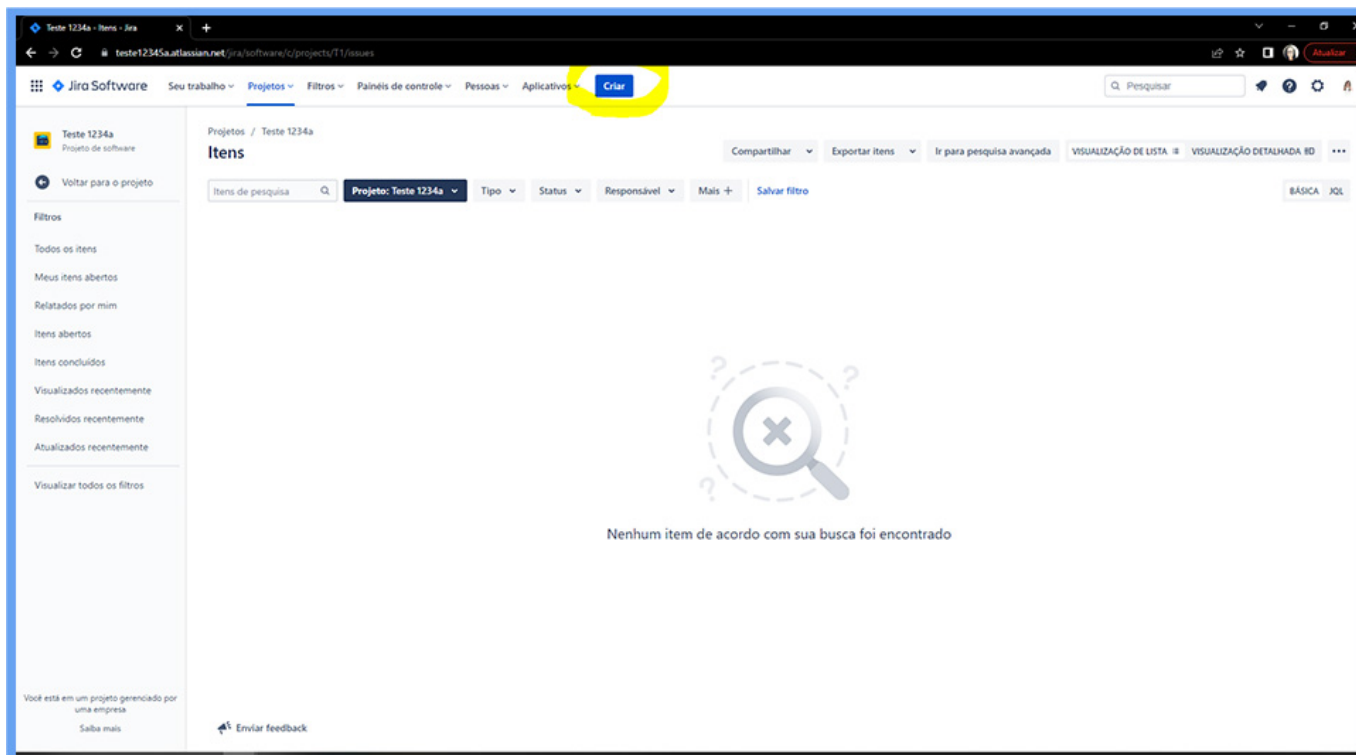


Figura 11 – *Board* onde ficarão os itens que serão criados no decorrer do projeto

Fonte: Senac EAD (2023)

Por meio da ferramenta, você pode capturar *bugs* em qualquer lugar em seus projetos de *software* com o Jira Software. É possível identificar um *bug*, criar um problema e adicionar todos os detalhes relevantes, incluindo descrições, nível de gravidade, capturas de tela, versão e muito mais, conforme apresentado na figura 12.

Os problemas podem representar qualquer coisa, desde um *bug* de *software* ou uma tarefa de projeto até um formulário de solicitação de licença, e cada tipo de problema exclusivo pode ter o próprio fluxo de trabalho personalizado.

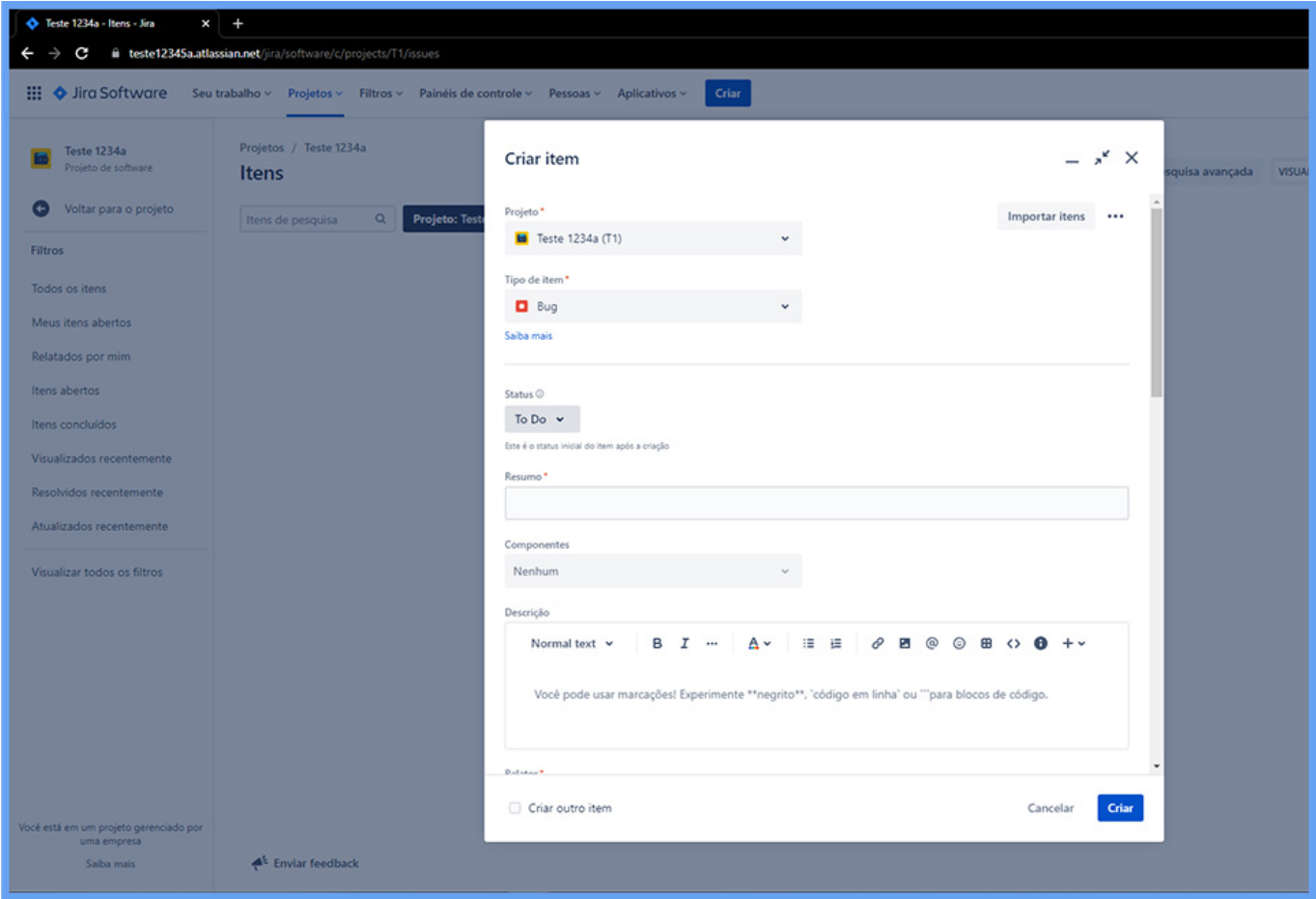


Figura 12 – Tela de cadastro de *bug*  
Fonte: Senac EAD (2023)

Os campos obrigatórios devem ser preenchidos, assim como os demais campos pertinentes ao projeto. Após o preenchimento, basta clicar em **Criar** e o item terá sido criado no *board* para ser acompanhado, conforme mostra a figura 13.

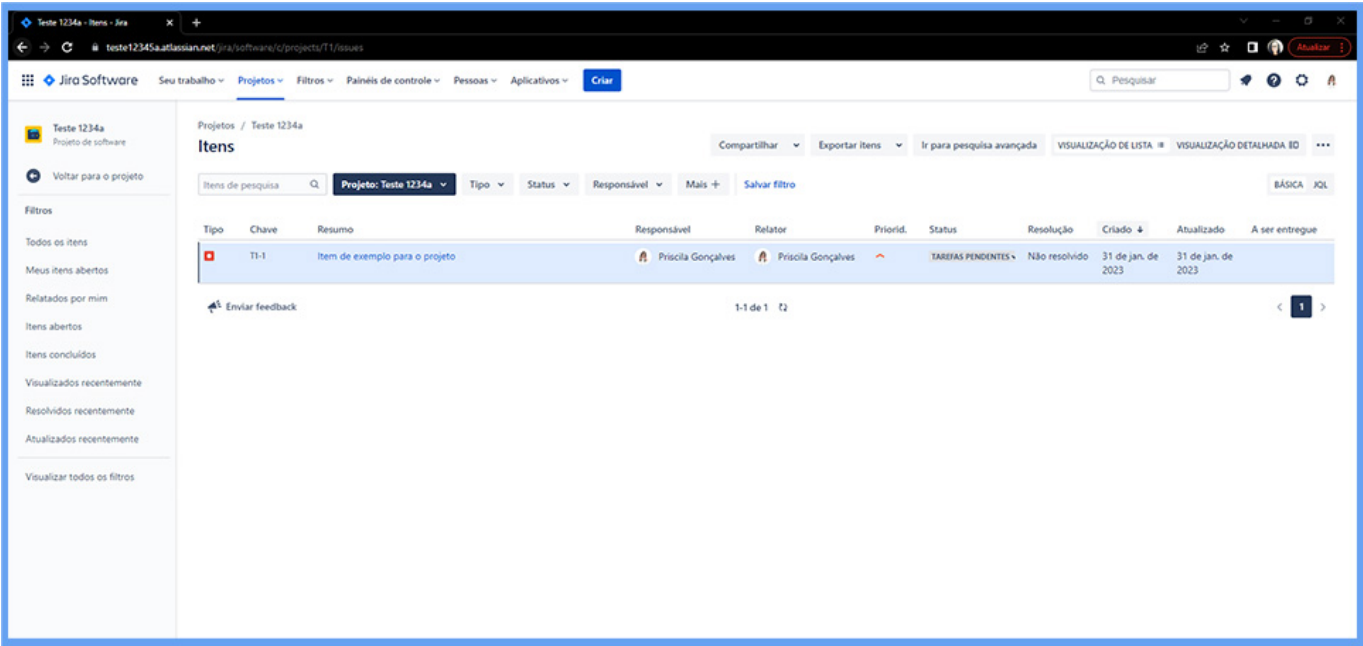


Figura 13 – Tela do *board* com o item cadastrado  
Fonte: Senac EAD (2023)

Quando os *bugs* são capturados, você pode priorizá-los de acordo com o nível de importância, urgência e capacidade da equipe, atribuindo-os com alguns cliques e priorizando-os por meio do “arrasta e solta” na lista de pendências ou, ainda, na coluna de tarefas pendentes da equipe, conforme apresentado na figura 14.

Sprint 2 3 issues

Start sprint

✓

ATMOS-25

Bad JSON data coming back from hotel..

SPACE TRAVEL PARTNERS

5

ATMOS-37

Warranty - it's not possible to redirect...

LARGE TEAM SUPPORT

5

ATMOS-9

When requesting user details, it shoul...

SEE SPACEEZ PLUS

5

+ Create issue

Backlog 19 issues

Create sprint

📌

ATMOS-7

After 100,000 requests, the SpaceEZ s...

SEE SPACEEZ PLUS

5

🔧

ATMOS-10

Session affinity - SessionBasedAnony...

LARGE TEAM SUPPORT

5

🚫

ATMOS-19

500 Error when requesting a reservati...

SEE SPACEEZ PLUS

5

✓

ATMOS-41

Bad JSON data coming back from hot...

SPACE TRAVEL PARTNERS

5

+ Create issue

Figura 14 – Exemplo de *backlog* de tarefas que o time tem para realizar

Fonte: Atlassian (c2023a)

Após a criação e a atribuição do item, é possível realizar a alteração de seu *status* na página inicial do *board*. Clicando em **Status**, aparecem as opções, conforme a figura 15.

Teste 1234a - Items - Jira

teste1234a.atlassian.net/jira/software/c/projects/T1/issues

Jira Software

Seu trabalho

Projetos

Filtros

Painéis de controle

Pessoas

Aplicativos

Criar

Teste 1234a

Projeto de software

Volta para o projeto

Filtros

Todos os itens

Meus itens abertos

Relatados por mim

Itens abertos

Itens concluídos

Visualizados recentemente

Resolvidos recentemente

Atualizados recentemente

Visualizar todos os filtros

Projetos / Teste 1234a

Itens

Compartilhar

Exportar itens

Ir para pesquisa avançada

Visualização de lista

Visualização detalhada

Básica

JQL

Item de pesquisa

Projeto: Teste 1234a

Tipo

Status

Responsável

Mais

Salvar filtro

Tipo

Chave

Resumo

Responsável

Relator

Priorid.

Status

Resolução

Criado

Atualizado

A ser entregue

T1-1

Item de exemplo para o projeto

Priscila Gonçalves

Priscila Gonçalves

TAREFAS PENDENTES

Não resolvido

31 de jan. de 2023

31 de jan. de 2023

A ser entregue

Enviar feedback

1-1 de 1

2

EM ANDAMENTO

IN REVIEW

CONCLUÍDO

Visualizar fluxo de trabalho

about:blank

13/24

Figura 15 – Exemplo de alteração de *status*

Fonte: Senac EAD (2023)

Há também como estar sempre informado rastreando *bugs* e problemas por meio do fluxo de trabalho de sua equipe. O Jira Software tem um poderoso mecanismo de fluxo de trabalho, recursos de planejamento, pesquisa abrangente e recursos de geração de relatórios projetados para ajudar a encontrar, registrar e rastrear *bugs* em seu *software*, conforme o exemplo mostrado na figura 16.

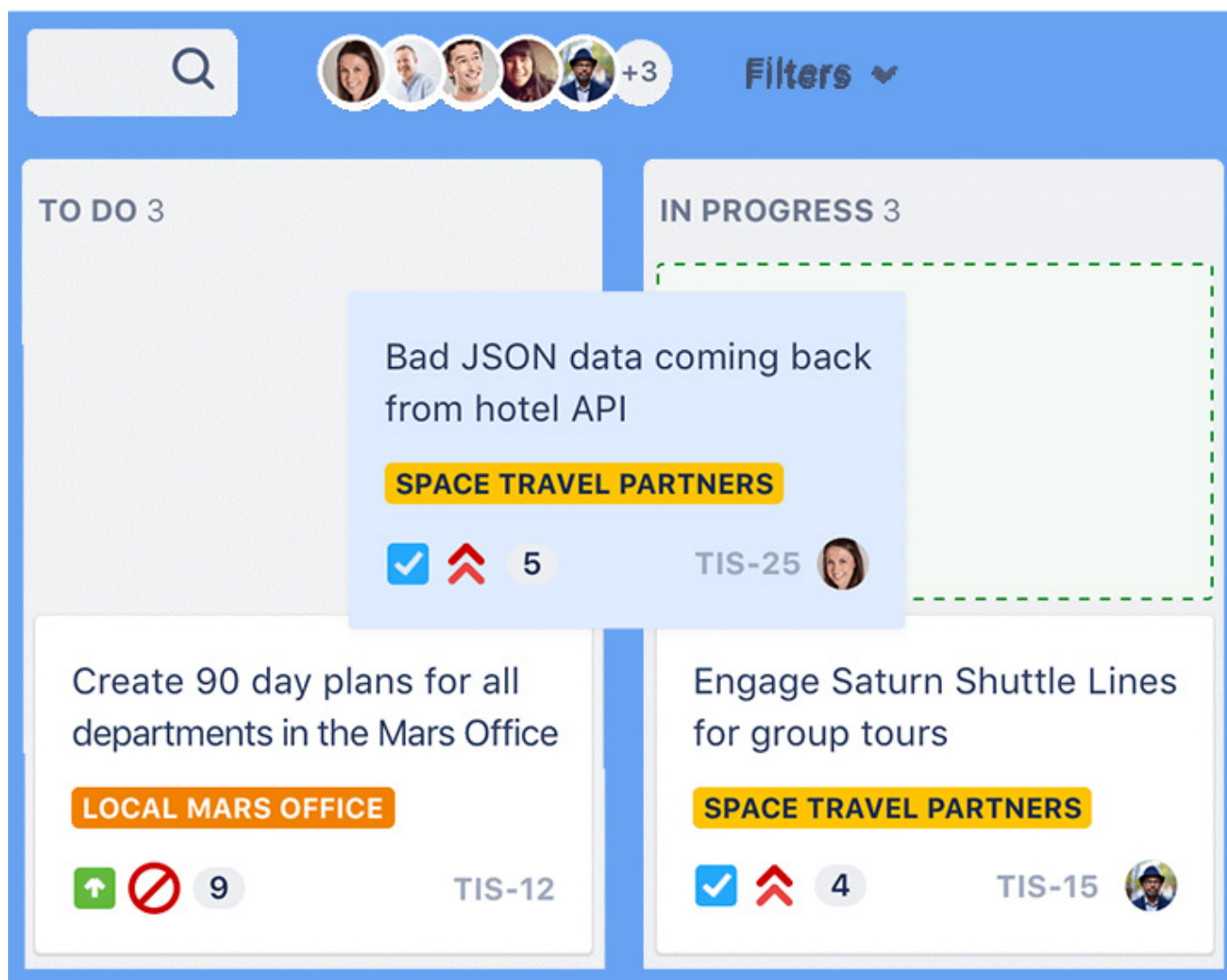


Figura 16 – Exemplo de fluxo de trabalho de equipe

Fonte: Site da Atlassian (c2023a)

Além disso, é possível manter-se atualizado com as notificações, pois não são somente as menções, e também configurar para que os responsáveis sejam notificados de forma automática a cada mudança de *status*, mantendo, assim, a visibilidade e a transparência entre todos os membros.



Agora que você já aprendeu sobre *bugtrackers* e ferramentas que exercem essa função, você aprenderá sobre as ferramentas de automação.

## Ferramentas de teste: automação

Há também a possibilidade de utilizar o Jira para a realização dos testes. O Jira é uma ferramenta de *software cloud* desenvolvida pela australiana Atlassian que permite monitorar tarefas e projetos de forma que seja garantido o gerenciamento em um local único. No Jira, há extensões que podem ser instaladas a fim de permitir que se façam inúmeras atividades, dentre elas a questão de testes, sejam manuais, sejam automatizados.

Uma dessas extensões a baixo custo após o período de experimentação (menos de 1 dólar por integrante de equipe para até dez pessoas) é a Xray. Essa extensão tem como objetivo o gerenciamento de teste nativo, ou seja, permite planejar, projetar e executar testes, assim como gerar relatórios deles.

Além disso, para cada fase de teste, é permitido que sejam utilizados os seguintes itens:



Fase do plano, em que constam os itens de plano de teste



Fase de projeto, em que a especificação é definida por meio de tipos de itens de pré-condição e teste



Fase de execução, em que estarão os itens de execução de teste



Fase de relatório, em que estão inclusos os itens de execução de teste, relatórios de cobertura de requisitos integrados e *tickets* utilizando as ferramentas do Jira

Será abordado, aqui, um passo a passo para a utilização do Xray, iniciando em como fazer a instalação. Para instalar, é necessário ter uma conta criada no Jira e procurar, na página do Atlassian Marketplace (digite “Atlassian Marketplace” no seu buscador), por “Xray Test Management for Jira”, selecionando a opção de fazer teste gratuito.

Agora que você já está com a extensão instalada, veja como escrever e especificar testes. É preciso saber que há três tipos de teste padrão: **pepino**, em que se pode escrever testes em um domínio legível para negócios e linguagem específica (Gherkin); **genérico**, em que há um teste não estruturado, sem etapas, sendo uma maneira de abstrair e ter a visibilidade de testes automatizados ou exploratórios; e **manual**.

Para criar um novo item de teste, clique em **Criar item** no menu superior da tela, conforme a figura 17 apresentada a seguir.

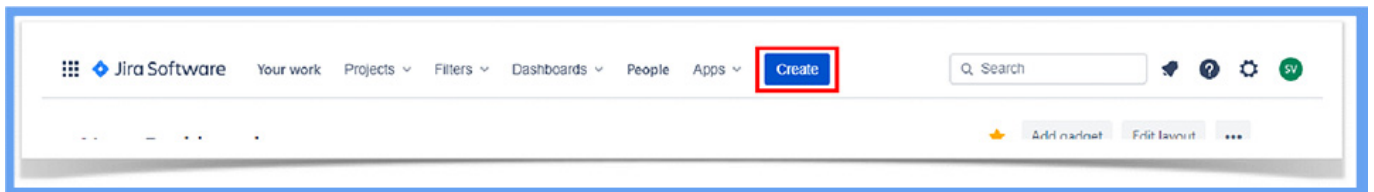


Figura 17 – Exemplo do menu de criação de item de teste

Fonte: Freire (c2023)

A próxima ação é selecionar o projeto no **Tipo de item** e selecionar **Testar**. Em seguida, digite a descrição resumida para o teste. É necessário preencher todos os campos obrigatórios que estão sinalizados com o asterisco e, na sequência, clicar em **Create**, se estiver tudo de acordo com o esperado, conforme apresentado na figura 18.

The image shows a 'Create issue' form in the Xray application. At the top right, there are two buttons: 'Import issues' and 'Configure fields'. The form contains the following fields:

- Project:** A dropdown menu with 'Calculator (CALC)' selected.
- Issue Type:** A dropdown menu with 'Test' selected. A help icon (?) is next to it.
- Summary:** A text input field.
- Component/s:** A dropdown menu.
- Description:** A rich text editor with a toolbar containing options for style, bold, italic, underline, text color, background color, link, unlink, list, and more.
- Fix Version/s:** A dropdown menu.

Below the 'Fix Version/s' field, there is a text prompt: 'Start typing to get a list of possible matches or press down to select.' At the bottom right of the form, there are three buttons: 'Create another' (with a checkbox), 'Create', and 'Cancel'.

Figura 18 – Exemplo do menu de criação de item de teste

Fonte: Freire (c2023)

Após completar essa primeira etapa, será abordado como organizar os testes no Xray. Essa é uma etapa de grande importância para as atividades de teste diárias. O Xray oferece duas formas: a primeira é usar os itens de conjuntos de testes, já a segunda é utilizar o repositório de

teste. Um conjunto de testes é uma forma de criar diversos grupos de testes, e um teste pode ser incluído em vários conjuntos. Já um repositório de testes é uma estrutura organizada em árvore no nível do projeto, o que torna possível que os testes sejam organizados por hierarquia nas pastas e nas subpastas.

Para criar um novo item de conjunto de testes, é necessário clicar em **Criar item**, na parte superior da tela, para que abra a referida página. Em seguida, selecione o “projeto” e vá em **Tipo de item, Conjunto de testes**. O próximo passo é digitar um resumo para o teste e preencher os campos obrigatórios. Realizado esse passo, basta clicar em **Criar**.

Para a criação de um repositório de teste, é necessária a criação de uma nova pasta.

Esse procedimento é feito da seguinte forma:

- ◆ Clicar na pasta.
- ◆ Clicar no ícone de adicionar (sinal de +), na parte superior da seção de pastas, conforme apresentado na figura 19.

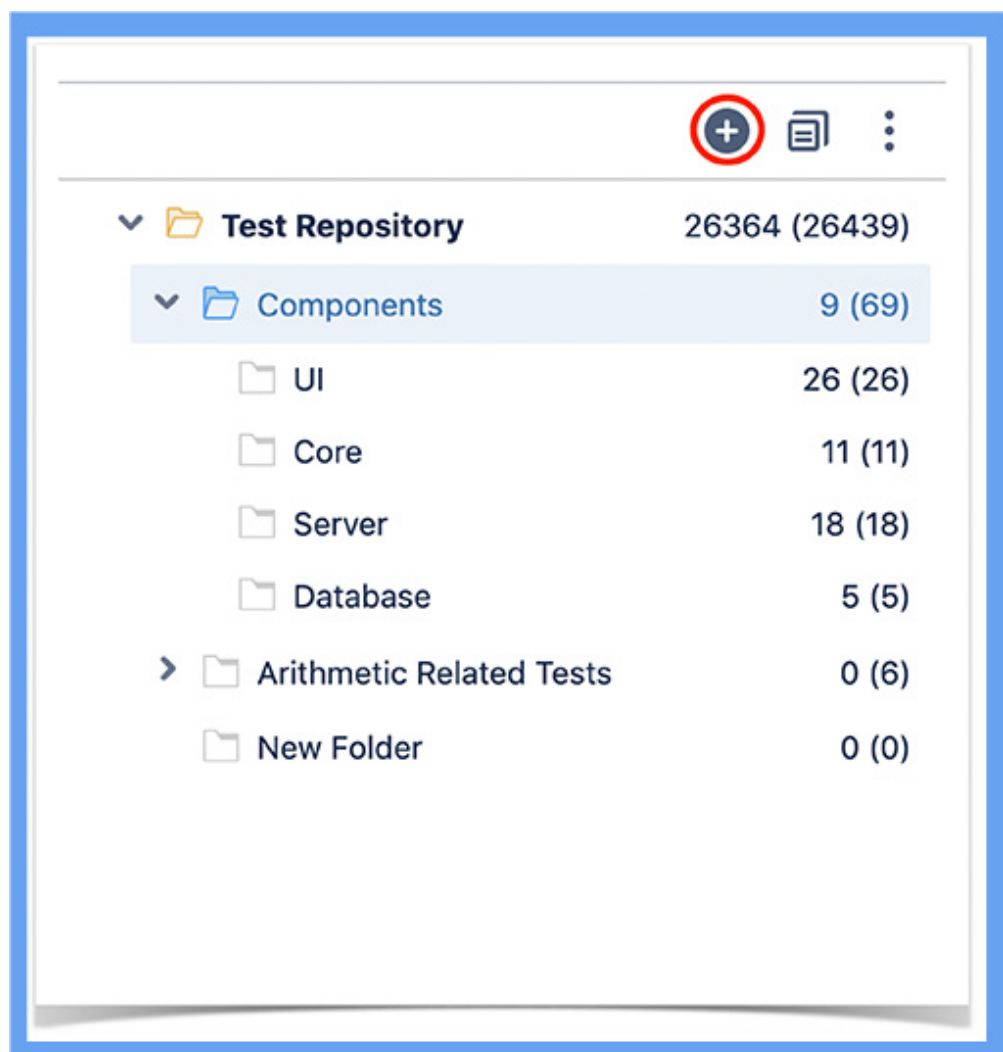


Figura 19 – Exemplo de criação de repositório de teste

Fonte: Freire (c2023)

- ◆ Digitar o nome desejado para a pasta na caixa de diálogo de criação de pasta, conforme apresentado na figura 20.
- ◆ Por fim, clicar em **Create** para criar a pasta.

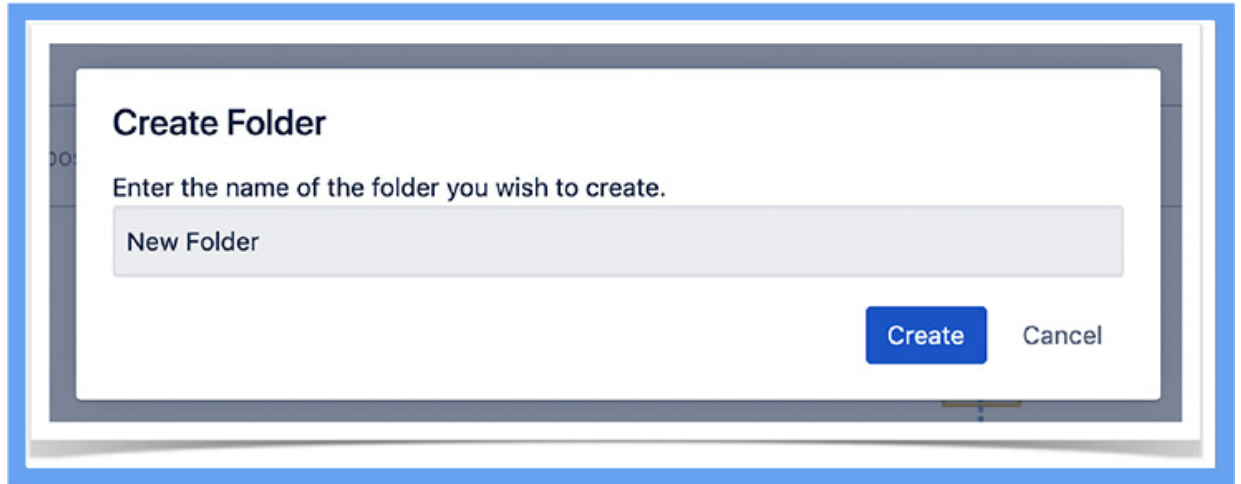


Figura 20 – Exemplo de criação de repositório de teste

Fonte: Freire (c2023)

Conforme visto até o momento, você já sabe como criar um conjunto de testes e como criar um repositório de testes. O próximo passo será verificar como planejar testes. Mas você sabe para que serve o planejamento de testes? O planejamento de testes consente que seja decidida por você a estratégia de teste, em que estão inclusos os itens que você quer validar, a forma de fazer essa validação, se serão testes manuais ou automatizados, de que maneira os recursos serão alocados e o momento e por quem os testes serão executados. Dessa forma, é possível a priorização de testes ou a concentração em requisitos específicos. E de que forma é possível fazer isso no Xray no Jira? Siga em frente para aprender!

Realize o passo a passo para a criação do plano de teste:

- ◆ Primeiro, é necessário clicar em **Create**, na parte superior da tela, para abrir a página **Criar item**, conforme apresentado na figura 21.

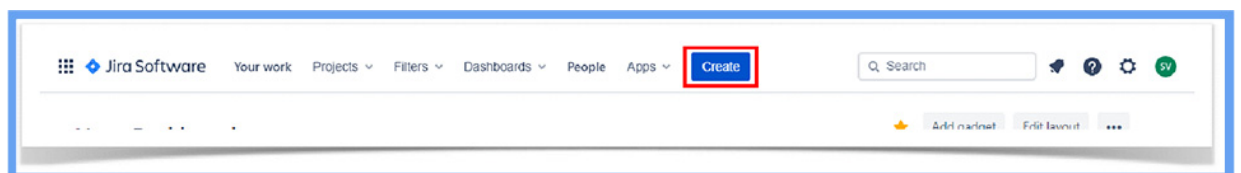
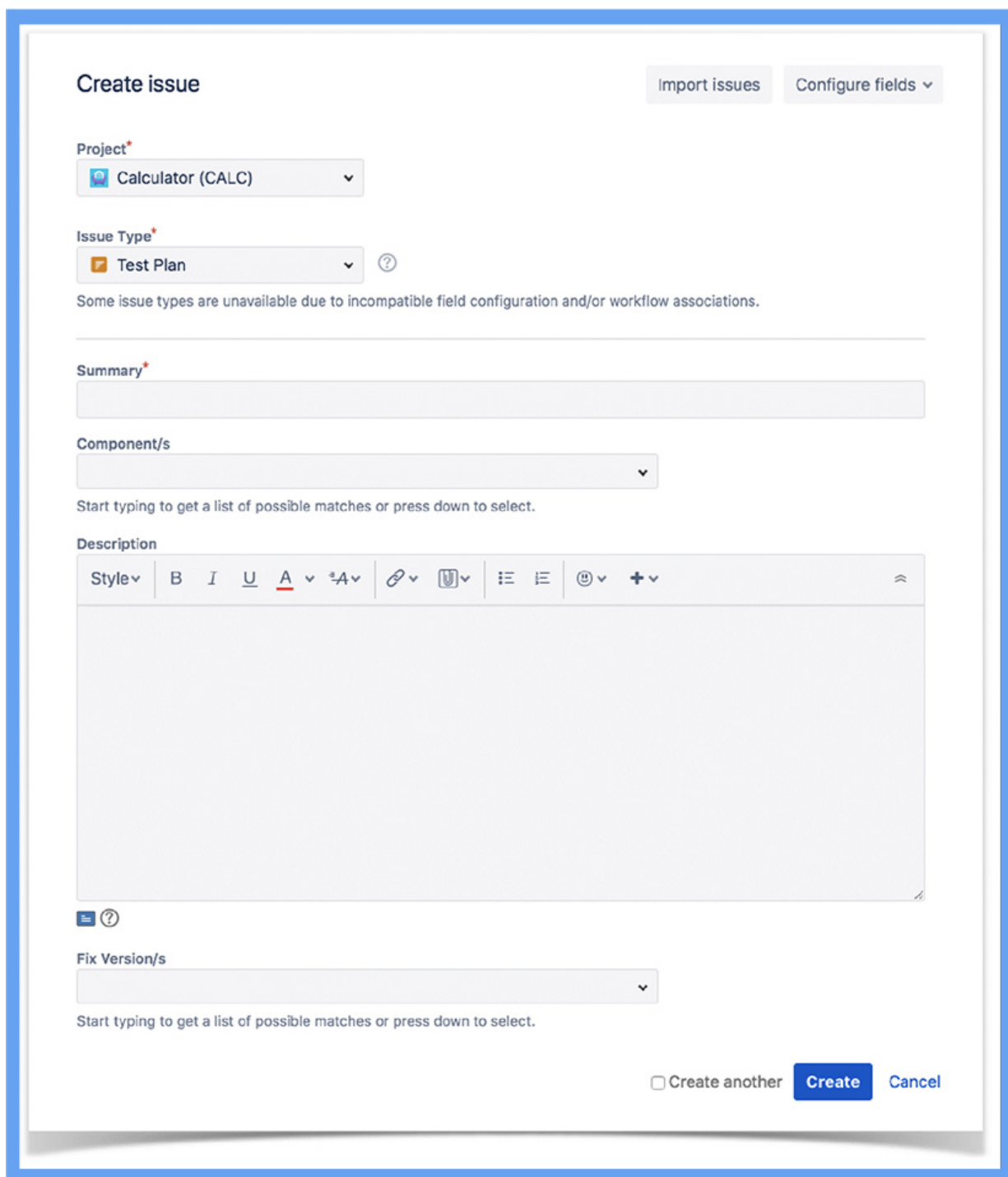


Figura 21 – Exemplo de criação de plano de teste.

Fonte: Freire (c2023)

- ◆ A etapa seguinte consiste em selecionar o projeto e, em **Tipo de item**, selecionar **Plano de teste**.
- ◆ Da mesma forma como ocorreram em etapas de processos anteriores, é necessário preencher um resumo e todos os campos obrigatórios presentes na tela e, finalmente, clicar no botão **Create**, conforme apresentado na figura 22.



The screenshot displays the 'Create issue' form in Jira. At the top right, there are buttons for 'Import issues' and 'Configure fields'. The 'Project' field is set to 'Calculator (CALC)'. The 'Issue Type' field is set to 'Test Plan', with a note below it stating: 'Some issue types are unavailable due to incompatible field configuration and/or workflow associations.' The 'Summary' field is empty. The 'Component/s' field is also empty, with a prompt to 'Start typing to get a list of possible matches or press down to select.' The 'Description' field is a large text area with a rich text editor toolbar above it. At the bottom, there is a 'Fix Version/s' field, also empty, with the same selection prompt. The bottom right corner features a checkbox for 'Create another', a blue 'Create' button, and a 'Cancel' button.

Figura 22 – Exemplo de criação de plano de teste

Fonte: Freire (c2023)

Por fim, há a etapa de execução dos testes. A execução dos testes acontece entre o desenvolvimento de *scripts* e relatórios e a análise de resultados do teste. Essa execução é um tipo de item que acrescenta uma coleção de testes escolhida pelo usuário, em que ele monitora e



verifica se os testes estão ocorrendo de acordo com o esperado no contexto e no ambiente destinado. Agora que você já sabe quando ocorre a execução de testes, veja como criar um item de execução de teste:

- ◆ Clicando no sinal de soma apresentado no canto superior esquerdo da página, abrirá a página de criação de item, conforme pode ser visto na figura 23.

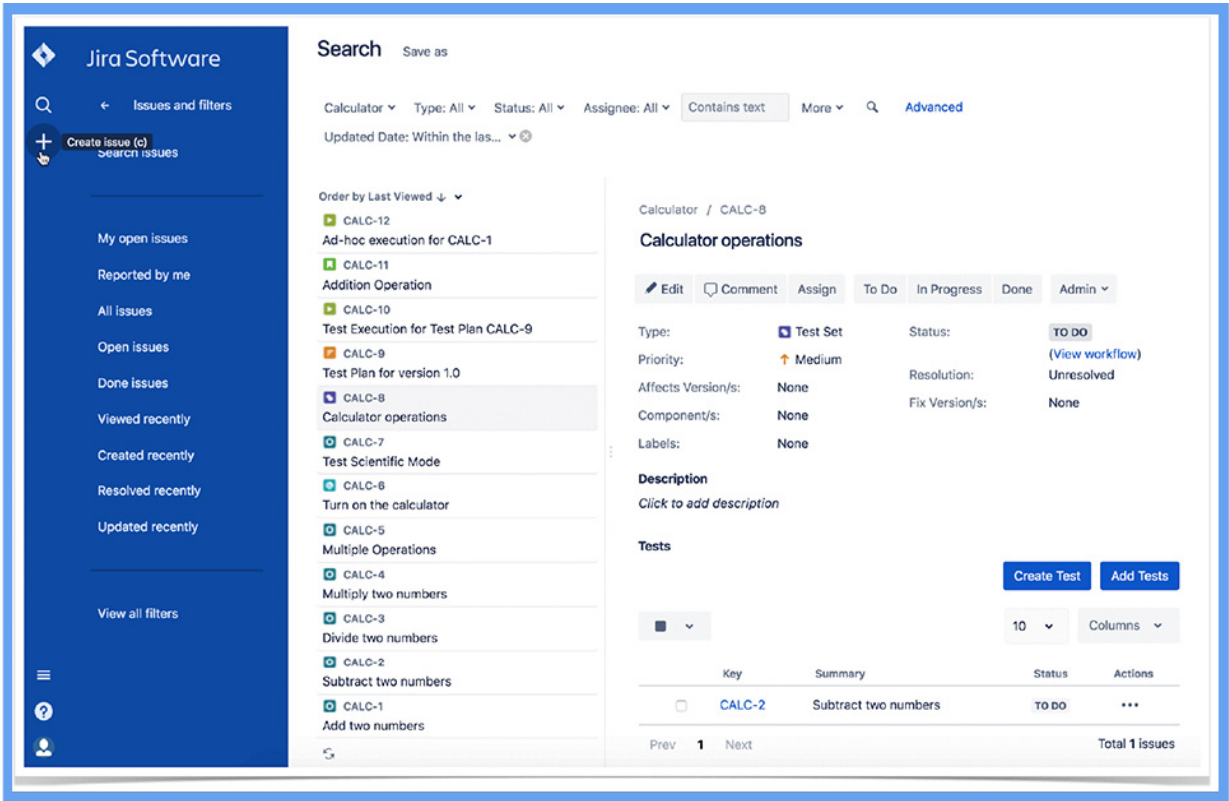


Figura 23 – Exemplo de página onde inicia a criação de um item de execução de teste

Fonte: Freire (c2023)

O próximo passo é selecionar o projeto e, em **Tipo de item**, selecionar **Execução de teste**.

É necessário digitar um resumo para a execução de teste e preencher os campos obrigatórios.

Na página **Criar item**, é preciso preencher os campos personalizados de execução de teste, incluindo as datas de início e fim, bem como a revisão.

É possível criar uma série de execuções de testes semelhantes no mesmo projeto, sendo apenas necessário selecionar **Criar outro** na parte inferior da página. Quando estiver tudo como deseja, clicar em **Create**.



Diante do que foi apresentado, pode-se concluir que o Xray é uma boa opção para utilizar no Jira, dispondo de vários recursos e sendo uma ferramenta intuitiva para o usuário.

Há outras ferramentas que podem ser utilizadas para a gestão de testes, algumas delas comerciais (pagas), como Mercury Interactive Test Director, Compuware QA Director e IBM Rational Test Manager, e outras *open source* (gratuitas), como Bugzilla Test Runner, Software Testing Automation Framework (STAF), JWebUnit, Bugzilla (testes funcionais), JMeter (*performance*), Mantis e phpBugTracker (sistemas *bugtracking*).

A ferramenta Bugzilla é um sistema de gestão de casos de teste que permite aos programadores a manutenção e a gestão de *bugs* do *software* em questão. Além disso, permite registrar alterações de código e *bugs* (problemas), comunicação entre membros da equipe, submissão e revisão de pacotes e gestão de QA (*quality assurance*). Na figura 24, é possível visualizar a interface na aplicação.



Figura 24 – Interface do Bugzilla

Fonte: Bugzilla (2023)

A ferramenta Mercury Interactive – Quality Center é uma suíte de aplicações para testes de *software* que permite gerir e automatizar o processo de QA e tem central de testes baseada na *web*. Sua ferramenta de gestão de testes é o TestDirector. Além disso, dentro do processo de teste, há grande destaque para metodologia, estrutura, organização e documentação.

O TestDirector é utilizado para identificar e definir os requisitos funcionais e de *performance* de teste e para realizar os testes. Também é possível desenvolver o plano de testes, executar os testes e registrar os defeitos encontrados e sua correção. Por meio da ferramenta, definem-se requisitos e propriedades, árvores de requisitos, agrupamento de requisitos por grupos funcionais e tipos, bem como seus estados, alterações de requisitos com seu respectivo histórico, anexação de documentos nos requisitos, envio de dados por *e-mail* e três visões da gestão de requisitos, sendo elas, respectivamente: lista de requisitos e propriedades, testes de cada requisito e execuções de teste de cada requisito.

Já os planos de teste precisam ser elaborados e organizados de acordo com as unidades funcionais da aplicação que serão testadas. Ainda, é preciso priorizar as relações hierárquicas e de dependência entre testes.

A seguir, na figura 25, apresenta-se a interface da ferramenta. É importante salientar que é uma ferramenta de mercado, portanto não é gratuita, e pode ser acessada digitando “TestDirect” em seu buscador.

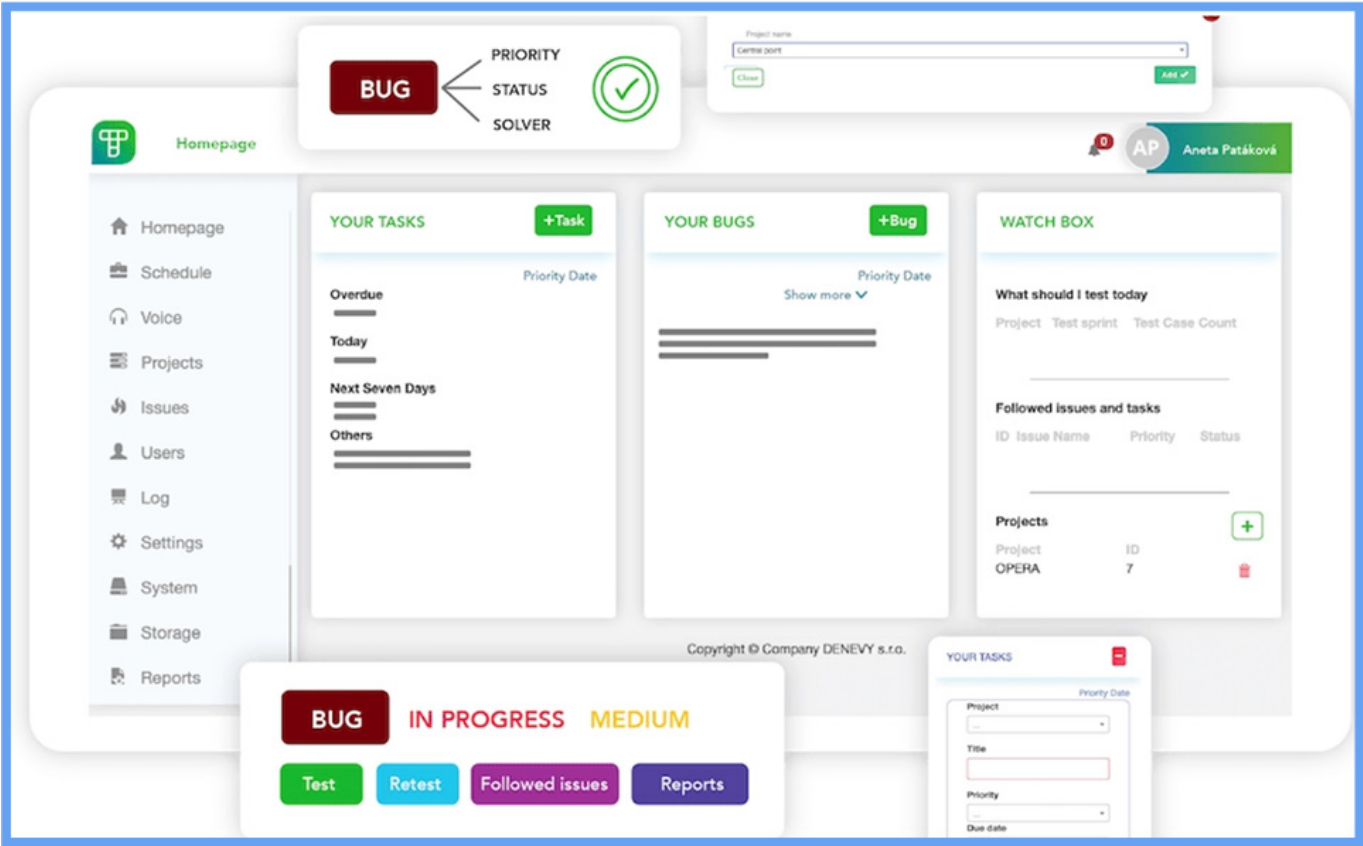


Figura 25 – Interface do *dashboard* do TestDirector  
Fonte: TestDirector (2023)

A Mercury afirma que essa é uma ferramenta líder de mercado. Além de ser uma ferramenta completa e em nuvem, há a opção de instalação no próprio servidor da empresa, em que se deve ter ao menos dez usuários, optar pelo plano *enterprise* e cumprir uma série de requisitos de infraestrutura necessários.

## Encerramento

Espera-se que este conteúdo tenha conseguido contribuir para o seu conhecimento acerca de ferramentas de testes manuais e automatizados. Aqui, você pôde conhecer as principais ferramentas, onde encontrá-las e a forma de utilizá-las. Diante do que foi abordado, agora você pode optar por ferramentas de mercado ou gratuitas e ainda entender como aplicá-las no dia a dia.