

PROJETO ARBYTE – Construção de um sistema de acompanhamento de bugs

Descrição:

O objetivo desse projeto é construir, do zero, seu próprio sistema de acompanhamento de bugs. Esse sistema tem uma arquitetura bem característica, que se assemelha a sistemas de ticket de empresas de suporte, e provavelmente vocês já tiveram contato com algum sistema do tipo ao longo de suas vidas (seja do lado do cliente, seja do lado da empresa, seja do lado dos desenvolvedores).

Por que esse projeto:

Toda empresa de desenvolvimento que se preze vai querer explorar um pouco do seu portfolio em uma entrevista de emprego. Caso você não tenha uma construção em seu portfolio que você possa explorar a fundo, você ficará a mercê de testes técnicos e tarefas para serem entregues após alguns dias, com grandes chances de falha. Um sistema criado por você mesmo permite que você conduza a conversa de recrutamento na maneira que lhe convém, explorando os tradeoffs corretos e limitações que sua aplicação sabidamente tem.

Além desse aspecto, um sistema de acompanhamento de bugs pode ser usado para acompanhar os próprios bugs, permitindo que você registre o histórico do desenvolvimento de seu produto, mostrando-o para recrutadores. Ele também é um projeto com um frontend completo e um backend relativamente complexo, mas não é um negócio da NASA. Você consegue fazer se for dedicado e sagaz em sua construção.

Softwares de Referência:

Alguns sistemas de ticketing servem como ótimas referências. O principal deles sendo o JIRA. Separamos dois vídeos que mostram como o JIRA funciona, e você pode ter ideias acerca de como montar as suas funcionalidades.

<https://www.youtube.com/watch?v=vgKECiXbPQY> – Walkthrough de como funciona abrir um ticket no JIRA.

Outro software de referência é o sistema de ticketing do Zendesk.

<https://www.youtube.com/watch?v=T1Nu03XXxx4> – Walkthrough na Plataforma do Zendesk

Partes essenciais do software:

É importante que seu software contenha um módulo de registro e login, dado que todo ticket aberto é assinalado a um autor, e é sempre possível assinalar responsáveis para as tarefas.

É interessante seu software ter uma interface de kanban onde é possível acompanhar os status dos tickets abertos: sem resposta, em progresso, concluído, por exemplo.

É interessante dar suporte ao anexo de arquivos, como print screens e logs do sistema

A tela de entrada, após o login, costuma ser um dashboard mostrando os tickets abertos

Você pode ser extremamente criativo com os relatórios gerados por sua ferramenta, dando a visão dos tickets abertos por uma mesma pessoa, tickets resolvidos por uma mesma pessoa, tickets por status, e por aí vai. Lembre-se que você está criando uma ferramenta que eventualmente você vai utilizar com outros desenvolvedores para acompanhar os bugs no software que vocês vão produzir. Isso é de extrema valia.

Mensagem Final:

Eu acho que vocês terão enorme dificuldade para terminar esse projeto com qualidade até o final do curso. Ataquem-o com parcimônia e utilizem ajuda uns dos outros. O objetivo desse projeto é fazer vocês entenderem a construção de uma ferramenta profissional de fora a fora, que com certeza será usada em seu favor uma vez que você for apresentar seu projeto numa entrevista. Estamos aqui para auxiliar nas dúvidas do caminho.

Bom trabalho!