

PROGRAMAÇÃO WEB 2018/2019 (MEI)

Enunciado do trabalho prático

Atributo	Valor
Designação	Serviço de análise e controlo de qualidade do serviço de helpdesk
Data de lançamento	2019-04-27
Trabalho em sala de aula	2019-06-22 (manhã e tarde).
Data de entrega	2019-06-29
Dimensão dos grupos	4 elementos

Existirá um momento intermédio para apresentação do trabalho ao docente da disciplina que terá lugar no dia 2019-06-01. O objetivo é fazer um ponto de situação em relação ao progresso do desenvolvimento e obter sugestões de melhoria por parte do docente.

Introdução

Os Sistemas de Gestão da Qualidade (SGQ) visam assegurar a execução sistemática e previsível dos vários processos de uma organização. Para tal é essencial estabelecer objetivos e indicadores que permitam aos gestores desses processos aferir a qualidade dos mesmos.

A empresa Jibs desenvolve software de gestão para diferentes sectores de atividade, possuindo uma base de clientes abrangente e diversificada. A Jibs considera que o seu serviço de Helpdesk (i.e., apoio ao cliente) é um dos seus processos mais importantes, pois uma parte substancial da sua receita provém da prestação deste serviço.

Para suportar o seu process de Helpdesk, a Jibs disponibiliza uma plataforma de helpdesk que permite aos seus clientes submeterem pedidos de suporte (é criado um projeto/área de helpdesk por cada produto implementado num cliente). Após a submissão de um pedido de suporte por um cliente, um colaborador afeto a um determinado produto, assume a responsabilidade de providenciar uma solução para o problema registando a resposta à solicitação na respetiva plataforma (e.g. a resposta a uma questão, ou informação que um problema reportado num servidor já se encontra resolvido).

As incidências de suporte podem ser de um dos seguintes tipos (`tracker.id`):

Id	Description
1	Bug
3	Support
9	Question
45	Report

Qualquer outro tipo de incidência que seja detetada deve ser ignorada para efeitos deste projeto.

Pretende-se que seja desenvolvida uma aplicação Web (back-end e front-end) que permita recolher informação da plataforma de helpdesk e disponibilizar ao gestor do processo indicadores relevantes sobre a presetação do serviço, incluindo a opinião dos seus clientes relativamente à qualidade dessa mesma prestação.

O trabalho deve ser desenvolvido recorrendo às tecnologias lecionadas ao longo da Unidade Curricular, nomeadamente, HTTP (e seus códigos de erro), NodeJS, Express, ReactJS, HTML 5, CSS 3, etc.

O trabalho deverá ser entregue sob a forma de um Docker container tendo como única dependência externa o Web service fornecido pela plataforma de helpdesk.

O trabalho deve também seguir uma metodologia assente num bom planeamento, onde se espera que as equipas de trabalho sejam capazes de definir e documentar, em sede de relatório técnico, os seguintes aspetos inerentes ao processo de desenvolvimento:

1. Entidades informacionais, suas propriedades e modelo de persistência
2. Formato dos dados trocados entre clientes e servidor (e.g. JSONP, XML, etc)
3. Esquema de identificadores das entidades informacionais (e.g. UUID, incrementais, etc)
4. Documentação da API desenvolvida
5. Planeamento de interfaces gráficas e modelo de interação (mockups)
6. Principais conclusões e ações de melhoria para o futuro

Back-end

A solução a desenvolver deverá consultar a API REST disponibilizada pela plataforma de helpdesk que lhe permite obter uma listagem de pedidos de suporte que foram respondidos desde uma dada data/hora, bem como informações sobre os utilizadores registados.

Após obter as informações sobre os pedidos de suporte, a solução deverá enviar um email ao cliente que criou a respetiva incidência para solicitar a sua opinião sobre a qualidade do serviço prestado.

O cliente tem a possibilidade de manifestar a sua opinião seguindo um de 5 links incluídos no email recebido. Cada link indica um nível de satisfação entre 1 e 5 (1 a 5 estrelas por exemplo). O email enviado

deve ser claro relativamente à incidência a que diz respeito, por exemplo, indicando o título da incidência, a data e hora da mesma, o nome do colaborador com quem interagiu ou qualquer outro elemento que se julgue relevante.

Após manifestar a sua opinião, o cliente deverá ser encaminhado para uma página de agradecimento que, simultaneamente, o informa que a plataforma recebeu a sua submissão com sucesso. Um dado cliente só deve poder votar uma vez, ou seja, o serviço, após receber uma votação para uma dada incidência de helpdesk não deve aceitar mais votações, apresentando informação em conformidade ao cliente.

A frequência de amostragem das incidências deve ser configurável (e.g. ficheiro de configuração ou variável de ambiente), porém não deverá ser superior a 5 minutos para não correr o risco do envio do email ser demasiado tarde.

Os Web services disponibilizados pela plataforma de helpdesk seguem a seguinte API:

HTTP Method	URL	Description
GET	<code>http://BASE_URL/api/v1/users</code>	Returns list of registered users
GET	<code>http://BASE_URL/api/v1/users/{id}</code>	Returns detailed information about the user with the id {id}
GET	<code>http://BASE_URL/api/v1/issues?after={datetime}</code>	Returns a list of Helpdesk requests that have been resolved after the {datetime} supplied to the service. The {datetime} parameter accepts values in ISO 8601 format, e.g. 2019-04-12T00:43Z.

Notas sobre a API:

- O endereço base da API é `https://redmine-mock-api.herokuapp.com`.
- Todos os endpoints apresentados respondem com uma mensagem em formato JSON ou JSONP e não carecem de autenticação.
- Os endpoints referentes a `users` suportam um parâmetro `forceMail` que permite redefinir o email do utilizador. Útil para efeitos de teste de envio de email (e.g. `http://BASE_URL/api/v1/users/{id}?forceMail=myemail@mail.com`)
- O endpoint referente a `issues` aceita um parâmetro adicional `limit` para limitar o número de issues devolvidos. Por omissão este valor está definido em 1000.
- O nome do produto e do cliente devem ser extraídos a partir do atributo `issues[].project.name`, e.g. no projeto com a designação `"Helpdesk | CA Datacom@Vectren"`, o nome do produto é `CA Datacom` e o nome do cliente é `Vectren`.

Para cada recurso de informação gerido pela nova aplicação de back-end devem ser criadas as operações

de CRUD (Create, Read, Update, Delete) necessárias para suportar a aplicação de Front-end, respeitando as boas práticas preconizadas pela filosofia RESTful em termos de passagem de parâmetros e mensagens de erro.

As respostas dadas pela API do serviço a desenvolver devem ser em formato JSONP para facilitar o processamento por parte de aplicações cliente. Para dar persistência aos dados, deve ser utilizada uma base de dados MongoDB.

NOTA: É importante salientar que um dado cliente apenas pode manifestar a sua opinião uma, e uma só, vez para uma dada incidência de helpdesk

Front-end

A segunda parte do trabalho foca-se no desenvolvimento de uma aplicação de front-end em ReactJS que permita ao gestor consultar um conjunto de indicadores considerados vitais para a adequada gestão do seu processo. Esses indicadores devem ser apresentados sob a forma de um dashboard que lhe permita consultar os indicadores e outras estatísticas consideradas relevantes de forma simples e imediata. Entre um intervalo de datas, o gestor deve poder saber:

Indicadores globais

1. O número total de pedidos de helpdesk
2. A percentagem de pedidos de helpdesk que não foram alvo de avaliação por parte dos clientes
3. O tempo médio de resposta a um pedido de helpdesk por cada nível prioridade de atendimento
4. A avaliação média da qualidade do serviço de helpdesk
5. O desvio padrão das votações dos clientes
6. Os colaboradores que responderam a pedidos com mais celeridade (ranking)
7. Os colaboradores que obtiveram as melhores pontuações por parte dos clientes (ranking)

Nota 1: O tempo de resposta mede-se através da diferença entre o momento em que o pedido foi criado e o momento em que foi fechado (atributos `closed_on` - `created_on`).

Nota 2: Cada incidência possui um nível de prioridade de atendimento que pode assumir um dos seguintes valores:

ID	Value
3	low
4	normal
5	high

Indicadores por colaborador

1. O número total de pedidos de helpdesk
2. A percentagem de pedidos de helpdesk que não foram alvo de avaliação por parte dos clientes
3. O tempo médio de resposta a um pedido de helpdesk por cada nível prioridade de atendimento
4. A avaliação média da qualidade do serviço de helpdesk
5. O desvio padrão das votações dos clientes

Indicadores por produto

1. O número total de pedidos de helpdesk
2. A percentagem de pedidos de helpdesk que não foram alvo de avaliação por parte dos clientes
3. O tempo médio de resposta a um pedido de helpdesk por cada nível prioridade de atendimento
4. A avaliação média da qualidade do serviço de helpdesk
5. O desvio padrão das votações dos clientes
6. Os colaboradores que responderam a pedidos com mais celeridade (ranking)
7. Os colaboradores que obtiveram as melhores pontuações por parte dos clientes (ranking)

Indicadores temporais

1. Gráfico com a evolução do número de pedidos ao longo do tempo (entre datas)
2. Gráfico com a evolução da avaliação média da qualidade do serviço ao longo do tempo

A interface gráfica deverá ser apelativa, bem estruturada e utilizar metáforas visuais que sejam reconhecíveis pela generalidade dos utilizadores da Web. O uso de frameworks CSS é altamente recomendado.

Bonificações

Este trabalho prevê a possibilidade de serem demonstrados conhecimentos avançados e capacidade de investigação sobre matérias que transcendem o programa da Unidade Curricular. Assim, será dada uma bonificação de 2 valores (a somar à nota obtida no TP), aos grupos que demonstrarem capacidade para resolver um (e apenas um) dos seguintes desafios:

- Implementar o sistema de autenticação usando JSON Web Tokens tanto ao nível da API REST como ao nível da aplicação de Front-end (<https://jwt.io>)
- Implementar uma camada de serviço em GraphQL sobre a API desenvolvida (<https://graphql.org>)
- Implementar 10 novos indicadores relevantes para a gestão do processo de Helpdesk

Entregáveis

A entrega do trabalho deverá realizar-se através do envio para jferreira@ipca.pt dos materiais identificados

de seguida até à data da entrega de cada uma das partes do trabalho. O porta-voz do grupo deverá submeter um ZIP com os vários entregáveis perfeitamente identificados e com os números e nomes dos constituintes de cada grupo.

Os entregáveis esperados são:

1. Código fonte
2. Relatório de 6 páginas em formato Markdown e PDF segundo modelo fornecido pelo docente
3. Slides apresentados durante a defesa do trabalho
4. Instruções claras sobre como descarregar e executar o(s) Docker container(s)

Não serão aceites trabalhos entregues fora do prazo estabelecido. Durante a defesa do trabalho, todos os membros do grupo deverão estar presentes para defender a solução desenvolvida e aptos para responder a questões sobre o trabalho. Em caso de falta não-justificada aplica-se uma penalização de 50% na nota final do trabalho ao elemento faltoso.

Avaliação

Todos os trabalhos serão avaliados com base nos entregáveis e na defesa do trabalho, tendo por base os seguintes critérios de avaliação:

- 40% - Completude da solução apresentada
- 30% - Respeito pelas boas-práticas de programação Web (arquitetura, modelo de dados, usabilidade, etc.)
- 15% - Completude do relatório e qualidade da escrita
- 15% - Qualidade da apresentação/defesa do trabalho à turma

A defesa do trabalho realizar-se-á numa sessão plenária com uma duração de 15 minutos, acrescidos de 5 minutos para discussão, onde deverá ser feita uma apresentação do trabalho e principais decisões tomadas ao longo do projeto (slides) e uma demonstração ao vivo da solução final.

A nota final da disciplina atribuída a cada aluno corresponde à seguinte formulação:

1		$\text{NOTA FINAL} = \text{TP} * 0.7 + \text{E} * 0.3$
2		
3		$\text{TP} = \text{Nota do trabalho prático}$ $\text{E} = \text{Nota obtida no exame}$
4		

NOTE BEM: A deteção de fraude em algum dos trabalhos implica a reprovação à Unidade Curricular.