

APDC Projeto: Edição 2020/2021

Exercício de avaliação Individual

5/Abril/2021

Âmbito da avaliação: A avaliação individual tem como objetivo a demonstração de competências adquiridas no decurso das sessões de apresentação e tutoriais de apresentação, com enfoque no desenvolvimento com a plataforma Google App. Engine e soluções Google Storage que serão utilizadas no projeto. A avaliação cobre ainda a aquisição de competências para desenvolvimento de serviços com endpoints REST bem como demonstração de execução de invocações REST cliente/servidor usando a ferramenta *Postman* (ou complementarmente e opcionalmente, clientes-piloto de demonstração com base em Javascript em Browsers).

Objetivo: O exercício a seguir enunciado deve ser feito como componente de avaliação obrigatória por parte de todos os alunos de APDC Projeto. Para efeitos de submissão da solução do exercício que deve ser feito individualmente, os alunos irão proceder à entrega dos materiais indicados a seguir. Para além da avaliação dos materiais a serem entregues nas datas indicadas, os alunos serão ainda convocados para uma prova oral individual.

Finalização da implementação do trabalho para entrega: 9/Abril/2021, 18h00

A solução do exercício e sua entrega envolver 3 componentes:

- A entrega da solução do exercício envolve 3 componentes:
 - A) Submissão do código: Arquivo ZIP** (Projeto Eclipse ou outro IDE - sources em repositório GitHub do aluno)
 - B) Submissão de respostas a um mini-teste (quiz) de 30 minutos, sobre a implementação e requisitos realizados**
 - C) Submissão de um vídeo (mpeg4, máx 10 min) com apresentação e demonstração da solução, a partir de um guião de passos de teste, como passos de comprovação da solução**
- **Guião de testes (passos de demonstração) sobre a solução desenvolvida**

Para a demonstração e teste da solução será disponibilizado um GUIÃO com um conjunto de passos de teste de demonstração do trabalho realizado. A realização deste guião e os resultados obtidos serão importantes para as respostas ao teste B e para demonstração e apresentação no vídeo C que deverá apresentar e demonstrar que o trabalho passa os testes do guião.

O Guião de testes deverá ser disponibilizado até 4ª Feira 7/ABRIL/2021, 20h00 (comunicação através de mensagem CLIP)

Entrega das 3 peças da realização do trabalho: Datas/Horas para entrega

- A) 6ª Feira, 9 de Abril/2021, 18h00: FORM (Googe Form) de submissão do trabalho (código fonte e URL GitHub)**
- B) 6ª Feira, 9/Abril/2021, das 18h30 às 19h00 (período em que o respetivo form estará aberto para respostas)**
- C) Sábado, 10/Abril/2021, até 12h59m: submissão via Form (Google Form)**

As referências para os forms de entrega bem como as indicações do Guião serão fornecidas até 5ª feira 8/Abril (comunicação através de mensagem CLIP)

1. Enunciado do exercício

Cada aluno deverá desenvolver um demonstrador de uma pequena aplicação de prova de conhecimentos e competências em programação e deployment com o Google App Engine e Google Storage.

1.1 Aplicação a desenvolver

A aplicação a desenvolver envolve dois componentes SERVIDOR e CLIENTE e deverá poder ser demonstrada através de invocações REST, com o CLIENTE demonstrado a partir da ferramenta Postman ou, opcionalmente, por um CLIENTE Web-Browser / Javascript). O SERVIDOR (integrando funcionalidade suportada em Google App Engine e dados persistentes (Google Storage) deverá poder ser demonstrado como aplicação CLOUD. A solução deve atender os requisitos do componente servidor (1.2), nomeadamente as **operações OP1 a OP7 na secção 1.2.1 (Funcionalidade e operações a desenvolver) e pelo menos três das operações indicativas referenciadas em 1.2.2 (Funcionalidade adicional a desenvolver).**

1.2 Componente Servidor

O componente servidor deverá poder executar no ambiente Cloud (*Google App Engine + Persistência*) tendo sido desenvolvido com as ferramentas e ambientes de programação das sessões de formação.

1.2.1 Funcionalidade e operações a desenvolver

OP1: Criação e registo de utilizadores (*account + perfil + role de utilizador*).

Deve suportar-se o registo e criação de contas de utilizadores (devendo ser mostrada a possibilidade de registo de alguns utilizadores). O registo deve permitir que na criação de contas os utilizadores registem atributos (que para além dos indicados podem ser estendidos à escolha dos alunos), mas que devem incluir pelo menos os seguintes:

- Atributos obrigatórios para criação da conta:
 - Username (ou UserID): <string>
 - Email do utilizador: terá que ter formato <string>@<string>.<dom>
 - <dom> representa um top-level domain DNS
 - Password: PWD (invisível): *****
 - A inserção da PWD deve sempre ser confirmada duas vezes
 - As passwords podem ter uma regra de validação à escolha dos alunos (ex., comprimento mínimo, conter letras e números ou um ou mais caracteres de pontuação, etc).

Notas: *não vai ser preciso no exercício confirmarem a conta por um duplo fator de autenticação via Mail ou SMS (ou Token através calculado através de uma aplicação móvel autenticada) para ativar a conta, para mudar a password ou para mudar qualquer atributo à posteriori - o que pode vir a ser considerado na abertura de contas no projeto a desenvolver. Também não será preciso validar neste exercício o EMail ou verificar se o mesmo é correto ou se é válido para poder ser usado pelo utilizador que se está a criar.*

- Atributos que devem ser adicionados (podendo ser isso feito à posteriori, por parte dos utilizadores, após entrada em sessão autenticada username/password::
 - Perfil: <string> - com indicação de seleção “Público” ou “Privado”
 - Telefone Fixo - formato habitual

- (ex: +351 NNNNNNNNN)
- Telefone Móvel - formato habitual
 - (ex: +351 91NNNNNNNN ou 93NNNNNNNN ou 96NNNNNNNN)
- Morada: (ex., “Rua dos alunos de APDC20-21, 100, Piso 20)
- Morada complementar: (ex., APDC Project Innovation Center for Fresh Ideas”)
- Localidade: (ex. Caparica)- CP: código postal XXXX-XXX
 - *Nota: Por agora não é necessário validar nada da morada registada (ex., códigos postais, endereços existentes ou não, etc., o que poderia envolver checks via Google Maps ou por exemplo via <https://www.codigo-postal.pt>, https://github.com/centraldedados/codigos_postais, etc)*

• ROLES

Quando uma conta é criada por um utilizador final deverá existir um atributo invisível (ROLE), não alterável pelo utilizador, que pode ter um dos seguintes valores:

- “USER” (utilizador final - default ou front-end)
- “GBO” (utilizador de gestão BackOffice)
- “GA” (utilizador Backend de gestão da aplicação)
- “SU” (super-utilizador avançado com todos os poderes)

Na criação de contas, o atributo *ROLE* (como default) fica com o valor “USER”.

O atributo de *ROLE* de um utilizador apenas deve poder ser alterado numa operação de gestão que apenas os utilizadores “GA” podem realizar (ver à frente)

A mudança de um *Role* USER para Role GBO só pode ser feita por utilizadores com role GA ou SU.

A mudança de um Role USER para Role GA só pode ser feita em backend por um perfil “SU”.

Deve existir um utilizador com *Role* SU, criado como *bootstrap user*, em tempo de deployment/inicialização do sistema e antes deste entrar em produção.

• ESTADO DA CONTA

Quando uma conta é criada por um utilizador final deverá existir um atributo invisível (STATE), não alterável pelo utilizador, que pode ter um dos seguintes valores: *ENABLED* ou *DISABLED*

Na criação de contas, o atributo *STATE* (como default) fica com o valor “ENABLED”. Uma conta pode ser passada de *ENABLED* para *DISABLED* mas esta operação exige ser feita com base no *ROLE* em causa (ver á frente).

OP2: Remoção de contas de utilizadores.

Esta operação faz a remoção da conta de utilizador: a operação é executável por um utilizador com role USER desde que pretenda remover a sua própria conta. Remoção de contas de qualquer utilizador USER (só pode ser uma operação executável apenas por utilizadores com role GBO ou GA);

OP3: Modificação de atributos de utilizadores.

Operação executável por utilizadores com role USER, apenas podendo modificar os atributos do próprio utilizador (com excepção do *Role*..

OP4: Modificação do ROLE de um utilizador

Operação executável apenas por utilizadores com role GA ou SU se for para mudar o perfil de contas de utilizadores de USER para GBO, ou por utilizadores SU, para mudar contas com perfil de USER para GBO ou GA.

OP5: Modificação de estado de conta

Operação executável da seguinte forma:

- Utilizadores USER não podem mudar o atributo de qualquer conta (nem mesmo da sua)
- Apenas utilizadores com roles GBO, GA ou SU podem modificar o estado de contas de utilizadores USER
- Apenas utilizadores com roles GA ou SU podem mudar o estado de contas de utilizadores GBO
- Apenas um utilizador com role SU pode mudar o estado de contas de utilizadores GA

OP6: Login - Operação de “Login” de sessão

O utilizador fornece o IDENTIFICADOR (ex., Username ou User Id ou EMail) e a PWD Operação: login (user, pwd). Se as credenciais são corretas (autenticação pelo servidor): recebe um token para a sessão que entre quaisquer outros atributos terá:

- USER: o identificador (ou login User ID)
- ROLE: o role (ou código de role)
- VALIDITY: validade do token (correspondente à validade da sessão), com as propriedades:
 - VALID_FROM: data da emissão (tempo do servidor)
 - VALID_TO: data de expiração (tempo do servidor)
 - VERIFICADOR: VERIF VERIFICADOR: ex., um “magic number” emitido pelo servidor que traduz uma prova de autenticidade da emissão pelo servidor dos dados do token, para realização de operações na sessão de LOGIN. A forma de criar este VERIFICADOR é opcional. Os alunos devem investigar como gerar este verificador de modo a que o servidor tenha a certeza que foi o servidor a gerar e poder controlar a sua validade com prova na sua autenticidade,

OP7: LOGOUT - Operação de “Logout” da sessão

Cliente executa a operação:logout (user, token). Como resultado o servidor revoga o TOKEN (mesmo que esteja ainda válido) e a sessão é redirecionada no cliente para uma página que notifica LOGOUT (desaparecendo a informação que se “via” na sessão aberta). Após o Logout, não é possível realizar operações em nome do utilizador (mesmo que se reutilize um token que ainda estava válido), a não ser que o utilizador faça de novo LOGIN com sucesso em nova sessão e obtenha assim um novo TOKEN válido.

1.2.2 Funcionalidade adicional a desenvolver

Para além da funcionalidade indicada em 1.2.1, deverão ser suportadas adicionalmente pelo menos três operações (à escolha de cada aluno) que possam ser demonstradas para execução em sessões LOGIN de um utilizador, de acordo com o seu ROLE. Notar que todas as operações são feitas por invocações REST (suportadas do lado do servidor) sendo os pedidos REST (com suporte HTTP POST)

pedidos na forma genérica:

```
<Output> = operation( <Input>, <Token>) isto é:      {  
  
    operation: <ocid>  
    Input: input JSON data  
    Token: JSON token format  
    Outor: output JSON result data  
}
```

Se o <Token> estiver correto e válido (correspondendo a uma sessão válida do utilizador), esta será realizada pelo servidor (desde que o utilizador seja suposto ter permissões para essa operação dado o seu Role). Se não é esse o caso ou se a operação resultar em erro ou exceção (seja por não validade da sessão, token inválido, token de sessão expirada, ou seja por exemplo por Role não adequado), o resultado refletirá o erro. Quando uma sessão é expirada o utilizador será redirecionado para a página de LOGIN. O utilizador não pode ficar confrontado com um estado indeterminado sobre o que possa ter acontecido.

Os alunos devem implementar no mínimo uma das operações indicativas para cada grupo seguinte (G1, G2, G3), escolhendo pelo menos uma de cada grupo (podendo se assim o pretenderem selecionar opcionalmente mais do que uma, para efeitos valorativos).

Grupo 1: Operações exemplificativas em sessões de utilizadores de ROLE “USER”:

- OP8.1A - Dentro da sessão de LOGIN, mostrar os atributos do utilizador que estão registados na conta do utilizador com perfil público (todos os roles podem executar)
- OP8.1B - Após o LOGIN aparece imediatamente a lista de outros utilizadores (com role USER) registados no sistema que têm perfil público e que estão nesse momento em sessão no sistema
- OP8.1C - Após o LOGIN aparece imediatamente a lista de outros utilizadores registados no sistema (com role USER), que têm perfil público e que estão nesse momento em sessão no sistema
- OP8.1D - Mudança de password do utilizador na sessão: exige o fornecimento da password corrente e a nova password (que deve sempre ser confirmada duas vezes).

Grupo 2: Operações exemplificativas em sessões de utilizadores de ROLE “GBO”:

- OP8.2A - Após o LOGIN aparece uma lista de todos os utilizadores registados e respectivos ROLES e perfis (público ou privado);
- OP8.2B - Mostrar todos os utilizadores registados e respectivos ROLES bem como perfis (público ou privado) e que estão logged em sessão (nesse momento)
- OP8.2C - Dado qualquer utilizador registado, obter os dados (atributos) do seu registo;
- OP8.2D - Mostrar todos os utilizadores dado um certo ROLE
- OP8.2E: Mostrar as contas que foram criadas num certo período (de: <data> a <data>)

Grupo G3: Operações exemplificativas em sessões de utilizadores de ROLE “GA”:

- OP8.3A - Remover ou desabilitar/habilitar uma conta de um utilizador (que tenha perfil USER ou GBO). O utilizador em causa deixará de usar o sistema em próximo login e verá a sua sessão expirada se no momento da remoção ou desabilitação estiver em sessão (logged) e com um token válido.
- OP8.3B - Qualquer uma das operações anteriores de utilizadores dos tipos USER ou GBO

1.3 Componente Cliente

O componente CLIENTE deverá ser desenvolvido como requisitos obrigatórios numa das seguintes opções. Os alunos podem apresentar (opcionalmente) a demonstração das duas opções, mas tal será apenas usado como informação valorativa.

Mínimo obrigatório:

- Funcionalidade cliente demonstrada com POSTMAN: o ambiente de trabalho (workspace) Postman deve estar preparado para mostrar execuções com testes REST sobre todas as operações REST pre-definidas em coleções, para as operações desenvolvidas e incluído scripts de teste para demonstração do correto funcionamento das operações REST (com base nos seus endpoint URLs e respectivos parâmetros)

Opcional e valorativo

- Cliente JAVASCRIPT/Browser (demonstrável com utilização em pelo menos dois browsers diferentes: (ex., dois diferentes de entre Google *Chrome*, Mozilla *Firefox*, Apple *Safari*, MS *MS Explorer*, MS *Edge*, etc.) no computador do aluno
- Cliente ANDROID/JAVA (que pode ser demonstrável em ambiente de emulação *Android Studio* ou mostrando em dispositivo *smartphone/tablet* do aluno).

2. Critérios de Classificação

Como indicado (*ver quadro inicial de Datas Importantes*) será depois disponibilizado um guião de testes e demonstrações associados aos requisitos deste enunciado. Este guião estará associado a passos de testes para demonstradores das funcionalidades indicadas (tendo como referência os requisitos obrigatórios). Assim, os requisitos serão demonstrados “passo a passo” obedecendo a tempos definidos. O sucesso de realização de cada passo da demonstração e o cumprimento do tempo definido para cada passo são fatores que determinam a avaliação individual. Cada passo realizado e demonstrado na sessão de avaliação terá uma pontuação que será como referência a seguinte:

- De início: Ter o código pronto e submetido na data indicada, com as três peças indicadas (20%)
- Até ao 2º min.: Conseguir criar o projeto, fazer importação do código desenvolvido entregue e fazer a sua compilação a partir do projeto local novo que foi criado no IDE: (+10%)
- Até ao 4º minuto: conseguir fazer deployment para o ambiente Cloud (GAE + Persistência) e mostrar que está pronto a operar em 2 min.: (+ 20%)
- Do 5º ao 15º Minuto: Demonstrar/Mostrar em sequência cada uma das seis operações (requisitos mínimos) (6 x 8% = +50 %)

Não conseguir realizar os passos em tempo útil ou os passos não conseguirem ser mostrados a prova de demonstração terminará no passo em que se ficar. A submissão do trabalho deverá ser acompanhada com o preenchimento de auto-avaliação com base no mesmo guião usado na verificação “presencial” e que os alunos apresentarão como auto-avaliação inicial (tendo em conta as métricas anteriores) e que será depois confirmada (ou seja, verificada) na sessão de avaliação “presencial”.

Todas as demais informações complementares sobre submissão do trabalho e sua demonstração constarão do referido Guião que estará disponível na data indicada (ver Datas Importantes).