

Projeto

Projeto de PSI 2019/20

Objetivo do projeto

O projeto deverá desenvolver funcionalidades para um sistema de informação que suporte clientes de uma cadeia de hotéis a gerir as suas reservas.

Aspetos técnicos

Os projetos utilizarão servidores implementando o stack tecnológico MEAN (MongoDB, Express, Angular, NodeJS). Os clientes terão obrigatoriamente de ser *browsers web*. É ainda necessário estruturar o sistema de informação de acordo com as seguintes quatro camadas: 1) cliente *browser*; 2) servidor *web* com lógica de apresentação; 3) servidor aplicacional com regras de negócio; e 4) servidor de base de dados.

Plataforma de execução do projeto

- O projeto deve ser executado no servidor `appserver.alunos.di.fc.ul.pt`
- O acesso ao servidor é feito através de ssh sendo o username o número do grupo. A password inicial é também o número do grupo e deve ser alterada após o primeiro login

```
> ssh psi050@appserver.alunos.di.fc.ul.pt
```

- O appserver tem instalados node, npm e mongo. Todos os módulos necessários para funcionamento do projeto devem ser instalados recorrendo ao npm.
- Cada grupo tem uma base de dados criada no servidor mongo. O nome da base de dados é igual ao número do grupo (p.ex. o grupo psi050 deve usar a base de dados psi050).

- Cada grupo tem um utilizador no servidor mongo. O nome do utilizador é igual ao número do grupo. A password também é igual ao número do grupo. Para acederem à consola do mongo usem o comando (substituindo psiXXX pelo número do grupo)

```
mongo --username psiXXX --password --authenticationDatabase psiXXX appserver.alunos.di.fc.ul.pt/psiXXX
```

- Cada grupo tem dois portos abertos para acesso por http a servidores node. O primeiro porto no intervalo 3001 a 3035 e o segundo porto no intervalo 3051 a 3085. Por exemplo, o grupo psi003 deve usar os portos 3003 e 3053. É assim importante que configurem os servidores node (para o front-end e back-end) nesses portos.
- A forma de executar o servidor node que serve o front-end Angular deve ser a seguinte (onde o XXXX que define o porto deve ser o específico de cada grupo)

```
ng serve --port XXXX --host 0.0.0.0 --disableHostCheck true
```

- Para o servidor node que serve o back-end não é necessário mudar a forma de execução.
- A connection string para acesso à base de dados mongo deve ser a seguinte (onde devem substituir psiXXX pelo número do grupo)

```
mongodb://psiXXX:psiXXX@localhost:27017/psiXXX?retryWrites=true&authSource=psiXXX
```

Recomendação sobre o processo de desenvolvimento (controlo de versões)

É vivamente recomendado que utilizem a plataforma git (git.alunos.di.fc.ul.pt) durante o desenvolvimento do projeto. O uso de controlo de versões irá facilitar o trabalho de uma equipa de 5 pessoas e irá permitir passar o código desenvolvido nas máquinas de desenvolvimento para o servidor de demonstração de uma forma fácil.

Recomenda-se vivamente que criem um branch no repositório para cada tarefa que seja criada na ferramenta JIRA. O branch deve ser identificado pelo número da tarefa no JIRA de modo a permitir a sua identificação de uma forma quase imediata.

Execução do projeto

As características deste projeto requerem uma equipa composta por 5 a 6 elementos, com competências e responsabilidades bem definidas e distintas, que serão avaliadas. Devido à importância do projeto, o processo de avaliação seguirá regras rigorosas. A constituição das equipas é realizada na atividade [Grupos de PSI](#) na página de PSI no moodle. O projeto seguirá a metodologia de desenvolvimento de *software* Scrum, que possui papéis e atividades bem definidos, operacionalizada pela plataforma JIRA Agile.

Como parte da metodologia *Scrum*, o conjunto de funcionalidades a implementar durante cada *sprint* é definido no início do *sprint*. O *backlog* inicial é composto pelos seguintes itens (estes itens podem sofrer alterações a qualquer momento).

- Epic1: Informações sobre os hotéis da cadeia
 - US8 - Como cliente quero consultar a oferta que a cadeia de hotéis oferece para poder decidir sobre os hotéis que me interessam
 - Prioridade: Alta
 - Critérios de aceitação:
 - A cadeia de hotéis deve ter uma página que liste cada hotel (apresentando no mínimo o seu nome e uma fotografia escolhida aleatoriamente a cada visualização da página).
 - A partir desta página deve ser possível aceder à página de cada hotel.
 - Deve ser possível aceder a toda a informação e efetuar todas as acções tanto num browser em desktop como em mobile.
 - US1 - Como cliente quero saber detalhes sobre os diferentes hotéis e os seus tipos de quartos para escolher o quarto com melhores condições
 - Prioridade: Alta
 - Critérios de aceitação:
 - Cada hotel deve ter uma página própria que apresente o seu nome, descrição, morada, coordenadas GPS, telefone, email, imagens, serviços disponíveis, total de quartos e total de quartos de cada tipo.
 - A partir da página de cada hotel devo poder aceder a uma visualização com informação detalhada dos quartos desse hotel que inclua os serviços disponíveis no quarto.
 - Deve ser possível visualizar lado a lado os serviços dos diferentes tipos de quarto.
 - Deve ser possível visualizar os serviços de cada quarto independentemente dos outros.
 - Deve ser possível aceder a toda a informação e efetuar todas as acções tanto num browser em desktop como em mobile.
 - US2 - Como cliente quero saber os preços por noite dos diferentes tipos de quartos de um hotel para escolher um quarto dentro do meu orçamento
 - Prioridade: Alta
 - Critérios de aceitação:
 - Na página do hotel, deve ser apresentado o preço do quarto mais barato, numa frase "Desde €xxx".

- Na visualização dos quartos do hotel deve ser apresentado o preço de cada quarto para as épocas baixa e alta (a época baixa decorre desde 15 de janeiro a 31 de maio e desde 30 de setembro a 15 de dezembro; a época alta decorre durante o resto do ano).
 - Na visualização dos quartos do hotel deve ser possível filtrar os quartos apresentados por preço (apresentar apenas quartos entre um limite inferior e um limite superior de preços, considerando ambos os preços de época baixa e alta).
 - Na visualização de cada tipo de quarto deve ser apresentado o preço do quarto para as épocas baixa e alta e a indicação de quando começa e acaba cada época.
 - Deve ser possível aceder a toda a informação e efetuar todas as ações tanto num browser em desktop como em mobile.
- US3 - Como cliente quero saber a disponibilidade temporal dos diferentes tipos de quartos de um hotel para escolher um quarto que esteja livre no período em que quero fazer a reserva
 - Prioridade: Média
 - Critérios de aceitação:
 - Na visualização dos quartos do hotel deve ser possível filtrar os quartos especificando uma data de entrada e uma data de saída para uma futura reserva e apresentado apenas os quartos disponíveis nessas datas.
 - Na visualização dos quartos de hotel deve ser possível calcular o custo de uma reserva nas datas especificadas para cada tipo de quarto.
 - Deve ser possível aceder a toda a informação e efetuar todas as ações tanto num browser em desktop como em mobile.
- Epic2: Gerir reservas de quartos
 - US4 - Como cliente quero poder reservar uma estadia num hotel através do site para não ter de gastar tempo com telefonemas ou deslocções até ao hotel
 - Prioridade: Média
 - Critérios de aceitação:
 - Na visualização dos quartos do hotel deve ser possível iniciar o processo de reserva de um quarto com as datas especificadas selecionando o tipo de quarto pretendido.

- O processo de reserva deve pedir confirmação das datas e do tipo de quarto, podendo o cliente alterar quer as datas quer o tipo de quarto neste passo.
- O processo de reserva deve pedir os dados de identificação do cliente: nome, morada, telefone, email e NIF. O formato dos diferentes elementos fornecidos pelo cliente deve ser validado (os telefones devem incluir indicativo internacional).
- O processo de reserva deve pedir os dados de um cartão de crédito para pagamento: número do cartão, prazo de validade e cvv. O formato dos diferentes elementos fornecidos pelo cliente deve ser validado.
- O processo de reserva deve apresentar todos os detalhes relativos à reserva e pedir confirmação da reserva.
- Deve ser possível aceder a toda a informação e efetuar todas as acções tanto num browser em desktop como em mobile.
- US5 - Como cliente quero saber as datas de check-in e check-out da cada uma das minhas reservas para não me enganar nas datas
 - Prioridade: Baixa
 - Critérios de aceitação:
 - O cliente deve poder aceder a uma página com a sua área pessoal onde pode visualizar todas as suas reservas (passadas e futuras).
 - Para cada reserva na área pessoal o cliente deve poder visualizar qual o hotel, qual o tipo de quarto e quais as datas de check-in e check-out.
 - O cliente deve poder filtrar as reservas por data de check-in (mostrando apenas reservas entre um intervalo de datas).
 - Deve ser possível aceder a toda a informação e efetuar todas as acções tanto num browser em desktop como em mobile.
- US6 - Como cliente quero saber o preço de cada uma das minhas reservas para poder gerir o meu orçamento
 - Prioridade: Baixa
 - Critérios de aceitação:
 - Para cada reserva na área pessoal o cliente deve poder visualizar qual o preço total da reserva.
 - O cliente deve poder filtrar as reservas por preço da reserva (mostrando apenas reservas acima de um valor dado).
 - Deve ser possível aceder a toda a informação e efetuar todas as acções tanto num browser em desktop como em mobile.
- US7 - Como cliente quero mudar as datas das minhas reservas para que alguma alteração na minha agenda não me obrigue a cancelar a reserva

- Prioridade: Baixa
- Critérios de aceitação:
 - Para cada reserva futura na área pessoal o cliente deve poder alterar a data de check-in e/ou de check-out da reserva, desde que haja disponibilidade de quartos para as datas pretendidas.
 - O processo de alteração da reserva deve calcular o novo custo da reserva e pedir ao cliente para confirmar a alteração depois de apresentar o novo preço.
 - O processo deve apresentar ao cliente os dados do cartão de crédito.
 - Se o novo preço for superior ao preço original deve perguntar ao cliente se pretende pagar com o mesmo cartão ou com um novo cartão. Caso o cliente escolha um novo cartão deve pedir os dados do novo cartão. O formato dos diferentes elementos fornecidos pelo cliente deve ser validado.
 - Se o novo preço for inferior ao preço original deve informar o cliente que a diferença nos preços será creditada no cartão que foi usado para pagar.
 - Deve ser possível aceder a toda a informação e efetuar todas as ações tanto num browser em desktop como em mobile.

Podem aceder aos dados sobre os hotéis estão disponibilizados [neste documento](#).

As funcionalidades serão implementadas ao longo de 3 *sprints* de desenvolvimento de *software*, cada um com duas semanas de duração.

Em cada *sprint* pretende-se que seja construída uma versão completa de funcionalidades do sistema de informação na *framework* escolhida. No final de cada *sprint* é feita uma demonstração das funcionalidades em laboratório.

Entregas

Os vídeos que demonstram os resultados obtidos em cada *sprint* devem ser entregues através da atividade existente na página da disciplina no Moodle. O nome do ficheiro deve seguir o seguinte formato: *psi_xxx_sprint_y*, onde *xxx* é o número do grupo e *y* o número do *sprint*.

Avaliações

A avaliação do projeto tem em consideração a qualidade do desenvolvimento e as atividades realizadas nas aulas teórico-práticas. São atribuídas notas qualitativas após cada avaliação e uma nota quantitativa final de acordo com os seguintes critérios de ponderação:

- Sprint 1 - 33,33%

- Sprint 2 - 33,33%
- Sprint 3 - 33,33%

Last modified: Friday, 17 April 2020, 2:37 PM



PREVIOUS ACTIVITY
Grupos de PSI

NEXT ACTIVITY
Chat com grupos de PSI sobre o projeto

