 <small>ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO</small>	Tipo de Prova Avaliação Final	Ano letivo 2021/2022	Data
	Curso LEI/LSIRC	Hora	
	Unidade Curricular Paradigmas de Programação	Duração	

Objetivos

Com a realização do trabalho prático, pretende-se que os alunos coloquem em prática todos os conhecimentos adquiridos na utilização do paradigma de programação orientado a objetos (POO) e a sua implementação na linguagem de programação Java, demonstrando as suas competências em:

- Conhecer e compreender os conceitos fundamentais associados ao paradigma da programação orientada a objetos;
- Conceber e implementar, para problemas concretos, soluções que tenham por base o paradigma da programação orientada a objetos.
- Reconhecer e compreender a semântica e a sintaxe da linguagem Java.
- Reutilizar, alterar e desenvolver código recorrendo à linguagem Java tendo em vista um determinado problema com regras semânticas específicas.

Considere ainda que:

- Não é permitida a utilização de API's/conceitos Java que não tenham sido alvo de lecionação na unidade curricular Paradigmas de Programação. Os alunos que pretendam utilizar API's adicionais devem atempadamente pedir autorização a um dos docentes da unidade curricular.
- Não é permitida a utilização de coleções Java predefinidas ([Java Collections Framework](#)).
- Os recursos de suporte ao trabalho referenciados no enunciado, são de utilização **obrigatória**.

Enunciado

Os grandes projetos de construção de infraestruturas representam tipicamente um processo complexo de gestão em que várias equipas de trabalhadores se encontram a realizar um conjunto de atividades num determinado local. As equipas deparam-se muitas vezes com problemas que embora possam não ser graves, necessitam de ser reportados aos níveis de gestão das empresas de forma a monitorizar e melhorar os vários processos organizacionais.

A comunicação destes eventos permite a elaboração de novas práticas e medidas que visam mitigar os problemas que decorrem durante uma obra e consequentemente melhorar a qualidade geral das condições de trabalho. Por exemplo, sinalização de incidências em obra que impeçam ou condicionem o decorrer dos trabalhos (condições meteorológicas ou mesmo condições de segurança), registo de acidentes com trabalhadores que estejam numa determinada equipa ou o registo de avarias ou mau funcionamento de máquinas em obra.

Cada equipa terá um tablet instalado no veículo de transporte. Caso ocorra algum evento relevante, qualquer trabalhador poderá dirigir-se ao tablet, realizar a autenticação com o seu cartão pessoal e registar o evento que será depois reportado a um conjunto de utilizadores da empresa.


Pretende-se que desenvolva uma API em linguagem Java que seja capaz de suportar os requisitos de uma ferramenta de registo e alerta de eventos ocorridos em obra.

Requisitos gerais

A solução a desenvolver deverá ser capaz de registar estaleiros de construção (**ConstructionSite**), bem como as equipas (**Team**) de trabalhadores (**Employee**) e os equipamentos (**Equipment**) que a constituem.

Considere ainda que:

- Um estaleiro de construção deverá ter sempre um funcionário com responsabilidades de gestor (**EmployeeType**).
- Os equipamentos podem ser de transporte, ferramentas, materiais, equipamento, Trabalho pesado, outros (**EquipmentType**) e podem estar num dos seguintes estados: operacional, em manutenção ou inoperativo (**EquipmentStatus**).

 <small>ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO</small>	Tipo de Prova Avaliação Final	Ano letivo 2021/2022	Data
	Curso LEI/LSIRC	Hora	
	Unidade Curricular Paradigmas de Programação	Duração	

- As equipas de trabalho são constituídas por trabalhadores (**Employee**) e pelo menos um líder (**EmployeeType** - **TEAM_LEADER**).

Um estaleiro só será válido se:

- Existir um responsável com o tipo gestor (**MANAGER**).
- Contenha pelo menos uma equipa de trabalho.
- Todas as equipas de trabalho devem ter um líder (**TEAM_LEADER**).
- Pelo menos um equipamento com estado de operativo (**OPERATIVE**).
- O alvará de construção com data válida.

De modo a registar os eventos (**Event**) que ocorrem durante o decorrer das obras, deve ser implementado um gestor de eventos (**EventManager**) que permita registar:

- Incidentes (*Incident*) - tipo de evento genérico que permite descrever uma ocorrência.
- Acidente (*Accident*) - incidente em que um funcionário está envolvido.
- Falha (*Failure*) - incidente em que um equipamento está envolvido.

O gestor de eventos pode aceitar um conjunto de notificadores (**Notifier**) que poderão notificar os funcionários de acordo com uma mensagem e prioridade (**EventPriority**). Cada notificador pode efetuar a notificação de modo distinto (por exemplo, email, escrever na consola, etc).

Quando um evento é reportado (incidente, acidente ou falha), deverá notificar o gestor e todos os líderes de equipas do estaleiro envolvido.


Os eventos registados devem ainda ser comunicados com maior brevidade possível à seguradora contratada pela empresa. Por vezes, não é possível comunicar no imediato o evento dado que pode não existir rede disponível. Por isso, a comunicação à seguradora deve ser assegurada de forma que nenhum evento fique por reportar. Este comportamento não está especificado nos recursos disponibilizados. Cada grupo deverá identificar a melhor abordagem para resolver o problema. A comunicação com a API web (pública) disponibilizada pela seguradora é encapsulada pela classe: **InsuranceReporter**. A identificação do grupo de trabalho com a API deverá ser realizada através da palavra "Grupo" seguida do número do grupo (*groupname*) e de uma chave de acesso (*groupkey*) composta, no mínimo por 15 caracteres. Não é necessário realizar registo na API. Cada grupo deve definir a sua chave e utilizar esse valor para suportar a comunicação com a API.

A API permite:

- Adicionar um evento. Os dados enviados para a adição dos eventos têm de obedecer a uma estrutura JSON específica:

```
{
  "groupname": "Grupo21",
  "groupkey": "...",
  "event": {
    "uuid": "abcd",
    "data": "2022-05-30",
    "priority": "High",
    "eventtype": "Accident",
    "title": "Avaria de máquina",
    "constructionsitename": "cs1",
    "details": "...",
    "employeenname": "João"
  }
}
```

- Remover e devolver (retornando um documento com os dados no formato JSON) todos os registos de eventos submetidos.

 <small>ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO</small>	Tipo de Prova Avaliação Final	Ano letivo 2021/2022	Data
	Curso LEI/LSIRC	Hora	
	Unidade Curricular Paradigmas de Programação	Duração	

Uma *dashboard* pública com os dados agregados de todos os grupos de trabalho encontra-se disponível no seguinte [link](#). A *dashboard* será enriquecida assim que novos dados sejam produzidos pelos grupos.

Com base na API desenvolvida, deverá apresentar uma aplicação funcional que permita ao utilizador através da consola todas as funcionalidades descritas, incluindo a apresentação dos eventos submetidos na API web da seguradora.

Descrição Técnica

Como suporte ao desenvolvimento da API, são disponibilizados um conjunto de recursos (*MA03_Resources*), de utilização obrigatória e que definem os contratos e definem o ponto de partida para o desenvolvimento da API. Os conteúdos fornecidos são um complemento ao presente enunciado, contendo informação específica sobre as particularidades de implementação de cada funcionalidade.

Deverá realizar a implementação do código necessário para suportar cada uma das operações definidas nos contratos. A existência dos contratos não deve ser impeditiva para a implementação de novas funcionalidades e/ou novos métodos ou classes. A utilização dos contratos constitui um ponto de partida, cujos ficheiros não podem ser alterados. **Caso não utilize os recursos disponibilizados, todo o trabalho é invalidado.**

Teste o mais exaustivamente possível o código que desenvolveu como resposta aos requisitos apresentados. Recorra a comentários JavaDoc e não só de modo a documentar, o mais exaustivamente possível, o código que desenvolveu.

Elaboração do trabalho

Este trabalho é realizado em grupo que deverá ser composto **por 2 alunos** da unidade curricular.

Datas e considerações

Os alunos devem comunicar atempadamente o seu grupo de trabalho na plataforma moodle, até ao dia **12 de junho de 2022**.

O trabalho deve ser entregue até às **23:55** horas do dia **25 de junho de 2022**, devendo a entrega ser feita através da página da unidade curricular de Paradigmas de Programação em <http://moodle.estg.ipp.pt>.

A defesa do trabalho será realizada na semana de **27 de junho**.


A defesa será realizada por turnos e a data exata para cada aluno será comunicada na plataforma moodle após a entrega do trabalho.

Considera-se por defesa satisfatória, quando o aluno demonstra que realizou o trabalho submetido e que **domina todos os conceitos de programação orientada a objetos aplicados na resolução do trabalho**. Tentativas de fraude, resultarão na avaliação do trabalho como: **Fraude Académica**.

Formato da Entrega

Os trabalhos entregues deverão evitar (se possível) utilizar caminhos absolutos ou endereços específicos, de modo a que possam ser facilmente utilizados em qualquer máquina. Para além disso, e no sentido de facilitar a receção dos vários trabalhos recebidos, estes deverão observar as seguintes regras:

- Todos os elementos do grupo deverão submeter o trabalho no link respetivo (Entrega do Trabalho);
- O trabalho desenvolvido deverá ser entregue através do moodle, através da submissão de um ficheiro com o nome PP_<nr_do_aluno>_<nr_do_aluno>.zip, contendo:
 - Os ficheiros criados incluindo o(s) projeto(s) do IDE Netbeans e uma pasta com a distribuição (jar) da solução proposta.

 <small>ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO</small>	Tipo de Prova Avaliação Final	Ano letivo 2021/2022	Data
	Curso LEI/LSIRC	Hora	
	Unidade Curricular Paradigmas de Programação	Duração	

- Recorra a comentários JavaDoc, e não só, de modo a documentar, o mais exaustivamente possível, o código desenvolvido.
- Cada ficheiro de código entregue por cada grupo terá de possuir no início do mesmo um comentário com pelo menos a seguinte informação (com as adaptações óbvias para cada aluno/grupo):

```

/*
* Nome: <Nome completo do aluno>
* Número: <Número mecanográfico do aluno>
* Turma: <Turma do aluno>
*
* Nome: <Nome completo do colega de grupo>
* Número: <Número mecanográfico do colega de grupo>
* Turma: <Turma do colega de grupo>
*/

```

Os alunos que não realizem a entrega do trabalho até à data/hora definida serão sujeitos a **penalização** ou a **invalidação do trabalho**.