 <small>ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO</small>	Tipo de Prova Época Especial	Ano letivo 2023/2024	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, 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 Época Especial	Ano letivo 2023/2024	Data
	Curso LEI/LSIRC	Hora	
	Unidade Curricular Paradigmas de Programação	Duração	

- As equipas de trabalho são constituídas por trabalhadores (**Employee**) e um líder (**EmployeeType** - **TEAM\_LEADER**). Cada equipa tem também um conjunto de equipamentos utilizados.
- Um trabalhador não pode estar em mais do que uma equipa.
- Têm de ser realizadas inspeções periódicas aos equipamentos e aos funcionários da empresa. Todas as inspeções devem estar registadas, identificando para cada uma a data e a avaliação (aprovado/não aprovado). A gestão de inspeções não se encontra especificado nos recursos disponibilizados, devendo cada grupo estudar a melhor abordagem para representar e gerir este tipo de informação. Considere ainda que:
  - No caso dos equipamentos, se a inspeção não for válida, o equipamento deverá ser identificado como inoperacional e deve ser registada a descrição do(s) problema(s) identificados
  - No caso dos funcionários, deverá ser registado o nome do médico de saúde e o seu número de licença. No caso da inspeção não aprovada, deve ainda ser identificado o número que dias que o funcionário ficará incapaz de exercer a sua função e o tipo de problema identificado que deverá pertencer a uma das seguintes categorias (fixas): oftalmologia, motricidade, mental e genérico.

Um estaleiro só será válido se:

- Existir um responsável com o tipo gestor (**MANAGER**).
- Contenha pelo menos uma equipa de trabalho.
- O alvará de construção com data válida de acordo com as datas de operação do estaleiro

A empresa deve que garantir que:

- O mesmo equipamento não poderá ser utilizado em mais do que um estaleiro/equipa em simultâneo
- A mesma equipa não poderá estar associada a mais do que um estaleiro em simultâneo

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.
- Acidente com máquina – Um tipo específico de acidente que ocorre obrigatoriamente entre um funcionário e uma máquina. Este tipo de acidente não se encontra especificado nos recursos disponibilizados, devendo cada grupo estudar a melhor abordagem para a representação deste tipo de acidente.

O gestor de eventos pode aceitar um conjunto de notificadores (**Notifier**) que poderão notificar um destinatário de acordo com um evento (**Event**). Cada notificador pode efetuar a notificação de modo distinto (por exemplo, email, escrever na consola, etc). Os eventos registados devem ainda ser comunicados com maior brevidade possível à seguradora contratada pela empresa.

Deverá obrigatoriamente existir um **Notifier** utilizado para reportar dados de incidentes à seguradora através da API pública disponibilizada, e um **Notifier** para apresentar notificações na consola para todos os gestores da empresa e líderes de cada equipa do estaleiro associado.

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.

Por vezes, não é possível comunicar no imediato as notificações, uma vez que pode não existir rede disponível. Por isso, a comunicação através das notificações deve ser assegurada de forma que nenhum

 <small>ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO</small>	Tipo de Prova Época Especial	Ano letivo 2023/2024	Data
	Curso LEI/LSIRC	Hora	
	Unidade Curricular Paradigmas de Programação	Duração	

evento fique por reportar. Este comportamento não está especificado nos recursos disponibilizados. Cada grupo deverá identificar a melhor abordagem para resolver o problema.

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.

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.

## Requisitos adicionais e obrigatórios

Para além dos requisitos apresentados, deverá:


- Desenvolver um componente de log que permita armazenar situações excecionais que ocorram durante o funcionamento da API. O componente é invalidado caso os registos sejam armazenados apenas em ficheiro ou apresentados na consola. **A qualquer momento deverá ser possível obter todas as exceções ocorridas num determinado período. Deve ser também possível exportar a informação para um ficheiro JSON.**
- Incorporar a API desenvolvida numa aplicação funcional que permita ao utilizador aceder a todas as funcionalidades apresentados nos recursos de suporte ao trabalho e no presente enunciado.

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.

## Descrição Técnica

Como suporte ao desenvolvimento da API, são disponibilizados um conjunto de recursos (*MA05\_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**

 <small>ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO</small>	Tipo de Prova Época Especial	Ano letivo 2023/2024	Data
	Curso LEI/LSIRC	Hora	
	Unidade Curricular Paradigmas de Programação	Duração	

**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 individualmente.

## Datas e considerações

O trabalho deve ser entregue na data especificada no Moodle na página da unidade curricular de Paradigmas de Programação em <http://moodle.estg.ipp.pt>.

A defesa do trabalho será realizada no dia do exame.

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:

- O trabalho desenvolvido deverá ser entregue através do moodle, através da submissão de um ficheiro com o nome PP\_<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.
  - 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>
 *
 */

```

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