



Curso: <u>Tecnologia em Análise e Desenvolvimento de Sistemas</u>		Semestre: <u>5º Noturno</u>
Disciplina: <u>Desenvolvimento de Aplicações Distribuídas</u>		Professor: <u>Luiz Fernando</u>
Data: <u> </u> / <u> </u> / <u>2025</u>		
Aluno:		Nota:

Atividade 4 - Desenvolva uma aplicação distribuída

1. A aplicação deve conter dois microsserviços. Um para registro em um sistema de eventos e outro para enviar um e-mail sobre o cadastro efetivado (neste segundo pode aproveitar o mesmo que já desenvolvemos em sala). Para o microsserviço de eventos (Event) vocês devem registrar/cadastrar um evento com os seguintes campos: id (Integer ou Long e automático), maxParticipants (Integer), registeredParticipants (Integer), date (DateTime ou String: não é a data do cadastro e sim a data que o evento irá ocorrer, deve ser preenchida manualmente), title (String) e description (String). O sistema também deve armazenar tais dados em um banco de dados qualquer. Além disso, outro objeto de domínio deve ser criado, que é a subscrição ao evento (um evento pode ter muitas subscrições), com os seguintes campos: id (Integer ou Long e automático), event (Event), participantEmail (String). Esse objeto também deve ser armazenado em um banco de dados. Os métodos (endpoints) da api são: getAllEvents() que retorna uma lista de todos os eventos cadastrados; getUpcomingEvents() que retorna uma lista apenas dos eventos futuros (que irão ocorrer ainda); createEvent(Event event) que cria um evento a partir dos dados informados; e, por último, registerParticipant(Subscription subscription), que registra a subscrição do participante em um evento, para isso é necessário informar o id do evento pelo endpoint. Como regra ainda é necessário verificar neste serviço se o evento não está lotado e ao

subscriver deverá atualizar a quantidade de participantes registrados no evento. Faça os tratamentos necessários com mensagens e exceções.

2. Ao finalizar grave um vídeo explicando o código e mostrando um exemplo com o uso da mensageira e poste no moodle.