### DM 112 – TRABALHO FINAL DA DISCIPLINA ATIVIDADE AVALIATIVA

Pós-Graduação em Desenvolvimento Mobile e Cloud Computing – Inatel
Pedro Piccichelli Carvalho
Lucas Bernardes Pereira

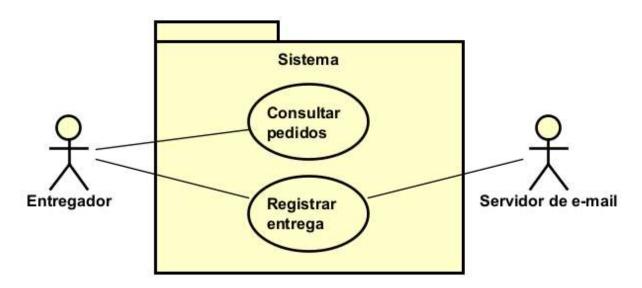


Figura 1: Caso de uso Provedor de Logística

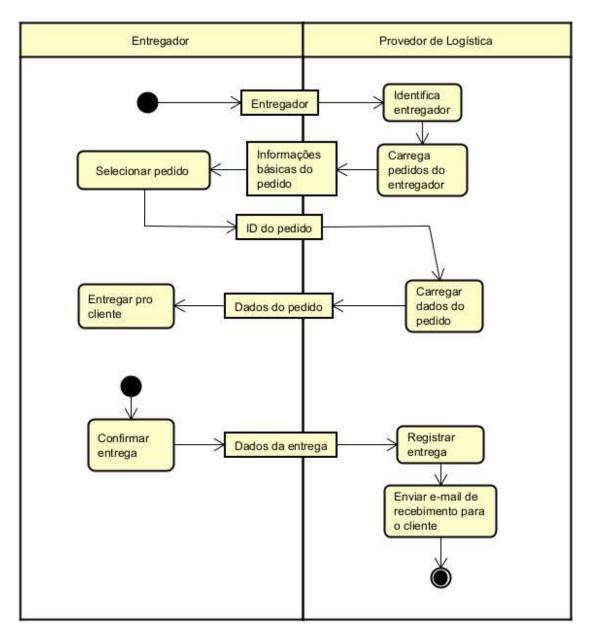


Figura 2: Modelo de processo da logística de entrega

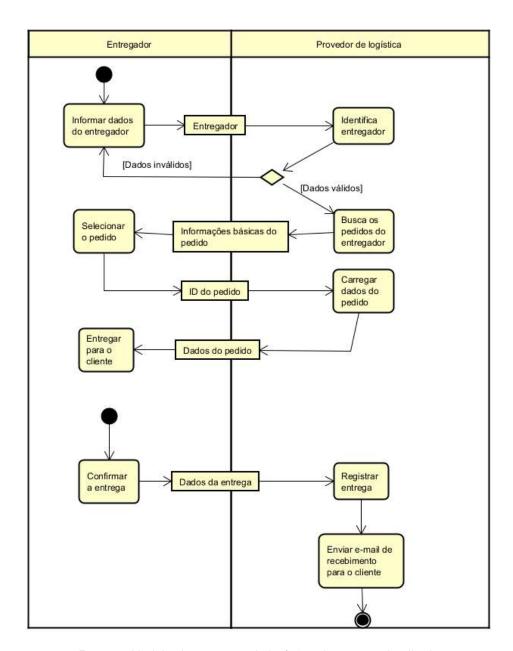


Figura 3: Modelo de processo da logística de entrega detalhado

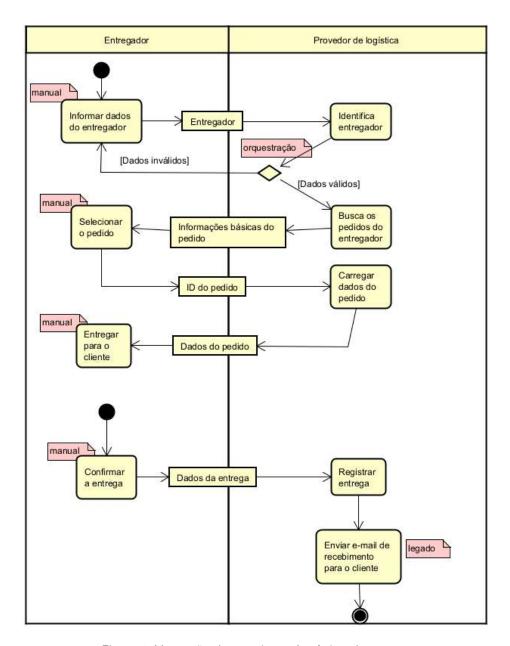


Figura 4: Marcação dos serviços - Logística de entrega

### Serviços candidatos agrupados em contexto lógico

- Pedido:
  - Buscar pedidos e devolver pedidos
  - Alterar stauts do pedido entre "Em espera", "Em trânsito" e "Entregue"
- Mensageiro:
  - Enviar e-mail para o cliente quando a entrega for realizada (legado)

#### Identificar a composição de serviços

Vamos criar um serviço de Entrega, que será responsável por orquestrar as chamadas de outros serviços. O serviço orquestrará as seguintes ações:

- Iniciar processo de entrega: Fará o processo de busca do pedido para o entregador e mudará o status do pedido
- Finalizar o processo de entrega: Esse serviço será responsável por alterar o status do pedido e mandar e-mail para o cliente

#### Identificando os serviços

Analisando o projeto, foram identificados os serviços: Pedido e Mensageiro

- Pedido: O serviço de pedido será responsável por procurar buscar e retornar um pedido e atualizar o status do pedido entre "Em espera", "Em trânsito" e "Entregue".
- Mensageiro: O serviço do mensageiro será responsável por enviar o e-mail para o cliente

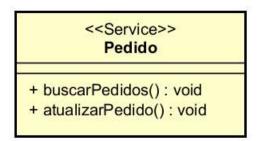




Figura 5: Relação de serviços refinada

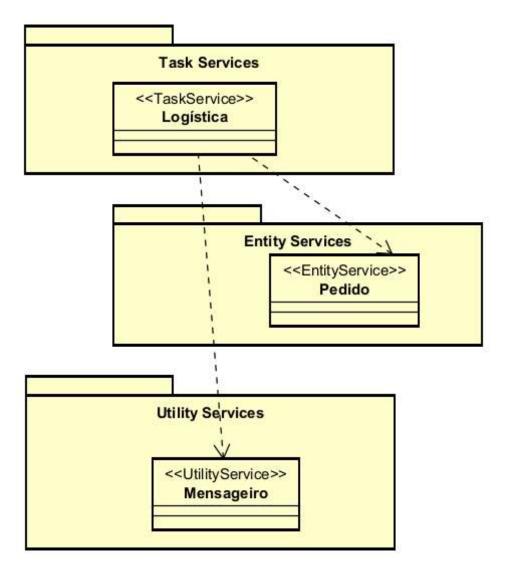


Figura 6: Relacionamento entre os serviços

# <<TaskService>> LogisticService

- + startDeliveryOfOrder(orderId : int) : OrderStatus
- + confirmDeliveryOfOrder(orderld : int, cpfReceiver : String, data : Date) : OrderStatus

## <<EntityService>> OrderService

- + getOrder(orderId : int) : Order
- + searchOrder(deliveryId : int) : List<OrderInfos>
- + startOrderDelivery(orderId: int): void
- + confirmOrderDelivery(orderld : int, cpfReceiver : String, data : Date) : void

## <<UtilityService>> EmailService

+ sendEmail(from : String, to : String, content : String) : MailStatusResponse

Figura 7: Diagrama de classes sobre os serviços da logística detalhados