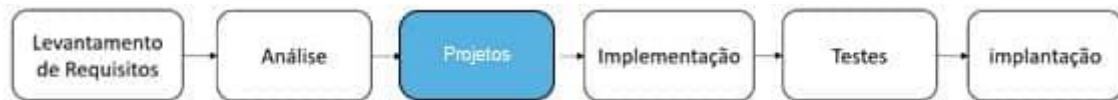


PRODUZIR O MODELO DE PROJETO PARA UM ESTUDO DE CASO DE MODELAGEM DE SISTEMAS EM UML

Daremos continuidade ao nosso estudo de caso, que trata do gerenciamento de uma rede de pousadas, desenvolvendo os principais modelos da etapa de **projeto** do processo de desenvolvimento de software genérico.



Etapa: Projeto.

MODELO DE INTERAÇÃO

O modelo de interação é composto por diagramas de sequência e/ou diagramas de comunicação, podendo ser iniciado na fase de análise ou na fase de projeto para representar os aspectos dinâmicos do sistema. Consideremos que o início da sua construção ocorre na etapa de projeto do processo, tendo como finalidade principal detalhar os objetos e mensagens requeridas na realização de um caso de uso.

Embora os modelos representem visões distintas do sistema, esses modelos são interdependentes, estando a modelagem de interação diretamente relacionada com a modelagem de casos de uso e a modelagem de classes.

DIAGRAMA DE SEQUÊNCIA

Este diagrama organiza os objetos participantes da interação na horizontal, sendo que cada objeto possui uma linha de vida e as mensagens entre esses objetos são representadas com linhas horizontais rotuladas partindo da linha de vida do objeto remetente e chegando à linha de vida do objeto receptor. A posição vertical das mensagens permite deduzir a ordem na qual elas são enviadas.

DIAGRAMA DE COMUNICAÇÃO

Este diagrama organiza os objetos em forma de nós, sendo adicionados setas e rótulos de mensagens nas ligações entre esses objetos; a ordem das mensagens é explicitada pela sua numeração sequencial no diagrama de comunicação.

Atenção

Os diagramas de comunicação e sequência representam as mesmas interações e devem estar consistentes com o diagrama de classes e com a respectiva descrição de caso de uso.

ESTUDO DE CASO – MODELO DE INTERAÇÃO

Daremos continuidade ao nosso estudo de caso, gerando o diagrama de sequência do caso de uso “**Registrar Reserva**”. Como afirmado anteriormente, esse diagrama possui uma interdependência com os modelos de casos de uso e modelo de classes, portanto vamos iniciar nossa abstração “dinâmica” com a descrição do referido caso de uso desenvolvida no tópico **Estudo de caso – modelo de casos de uso**, sendo parcialmente replicada a seguir:

Caso de uso: Registrar Reserva.

Atores: Cliente e Recepção (funcionário)

Pré-condição: Cliente ou Recepção deve estar identificado pelo sistema.

Fluxo Principal:

1. O sistema busca as pousadas disponíveis.
2. O Cliente/Recepção informa a pousada destino e as datas de *check-in* e *check-out*.
3. O sistema busca quartos disponíveis no período informado.
4. O sistema lista os quartos disponíveis para o período informado por tipo, incluindo a quantidade máxima de pessoas a serem acomodadas.
5. O Cliente/Recepção seleciona os quartos e informa a quantidade de pessoas por quarto selecionado, sendo no mínimo 1 (um) quarto e no máximo 3 (três) (de acordo com RN 2).
6. O sistema exige os dados de login do cliente para efetivação da reserva online.
7. ...

Fluxo Alternativo (6): reserva realizada por contato telefônico (recepção da pousada)

- a. O sistema exige dados de login do funcionário da recepção para efetivação da reserva por contato telefônico.
- b. O funcionário da recepção informa dados para login.
- c. O funcionário da recepção informa o CPF do cliente.
- d. O sistema valida o cliente.
- e. Prosseguir no passo 9.

Tabela: Alberto Tavares da Silva.

Temos que considerar na construção do modelo de interação o requisito não funcional “RNF 5 – O padrão de arquitetura adotado é o MVC (model-view-controller)”, pois novos objetos serão incorporados, a fim de permitir que o projeto do sistema tenha um nível de detalhamento suficiente para sua implementação.

Diagrama de sequência

Retornando à nossa descrição, analisemos o passo 1, “**O sistema busca as pousadas disponíveis**”, estando a respectiva modelagem ilustrada a seguir. Nos diagramas de interação, por convenção de boa prática, empregamos os sufixos Model, View e Controller na denominação dos objetos, para indicar as camadas respectivas na fase de implementação do modelo.

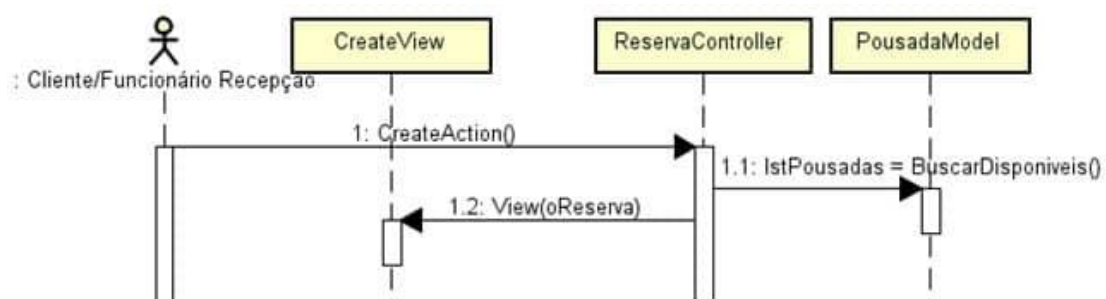


Diagrama de sequência parcial do caso de uso “Registrar Reserva” gerado pela ferramenta Astah.

A interação do ator, cliente ou funcionário da recepção definido no diagrama de casos de uso do estudo de caso, é iniciada pelo método **CreateAction** da classe controladora **ReservaController** que, por sua vez, instancia um objeto **PousadaModel**,

que é um objeto entidade identificado a partir do diagrama de classes de análise, a fim de que esse busque as pousadas disponíveis.

Em seguida, analisemos os passos 2, “**O Cliente/Recepção informa a pousada destino e as datas de *check-in* e *check-out***” e 3, “**O sistema busca quartos disponíveis no período informado**”, cujas modelagens estão representadas a seguir.

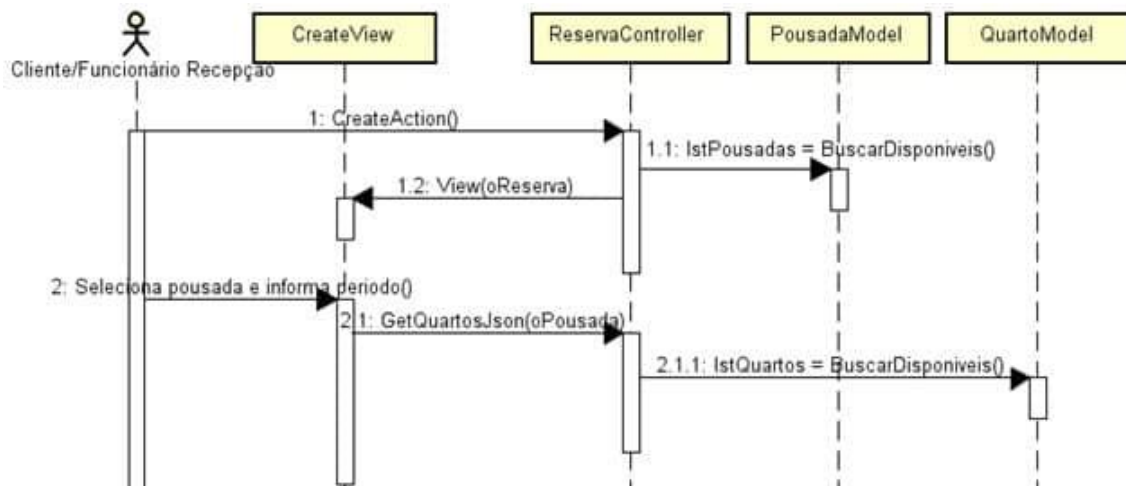


Diagrama de sequência parcial do caso de uso “Registrar Reserva” passo 2 gerado pela ferramenta Astah.

Após uma segunda interação do ator, ou seja, seleção da pousada, uma função JavaScript na camada View aciona um método na classe controladora a fim de buscar os quartos disponíveis para a selecionada pousada no período desejado, conforme observado na imagem a seguir.

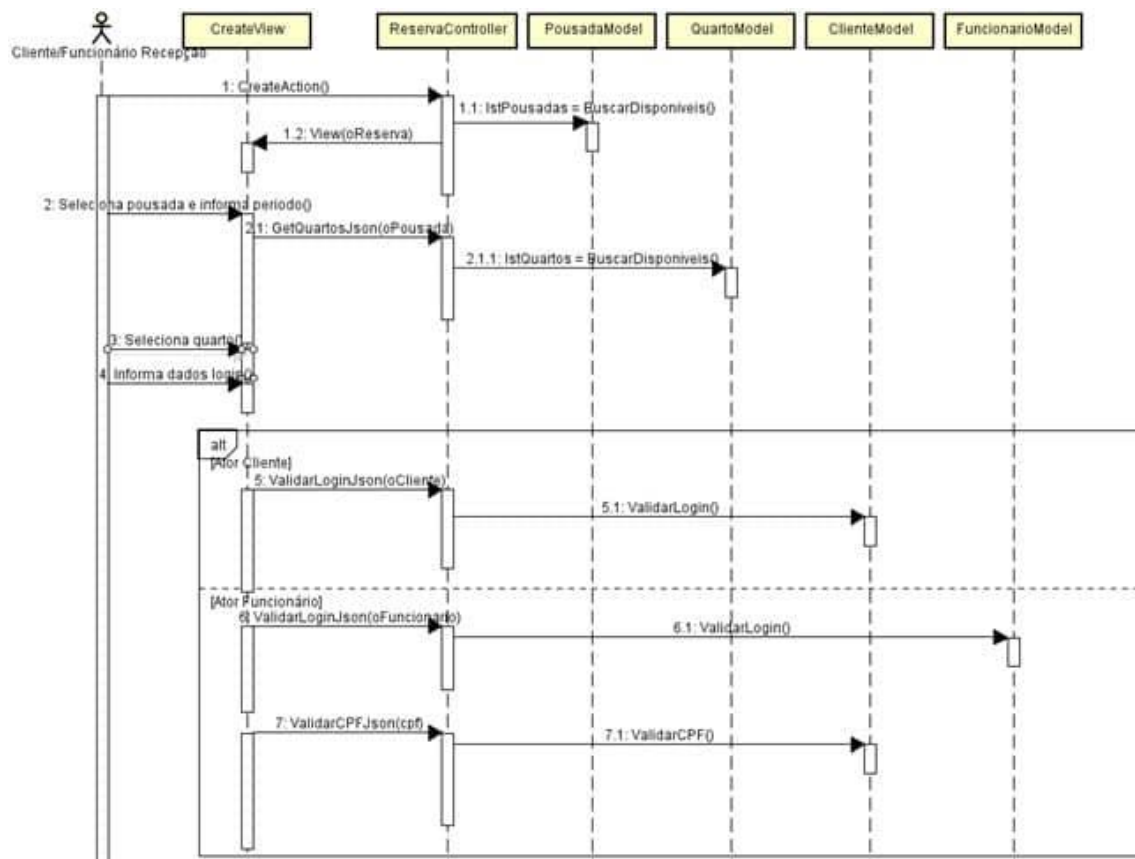
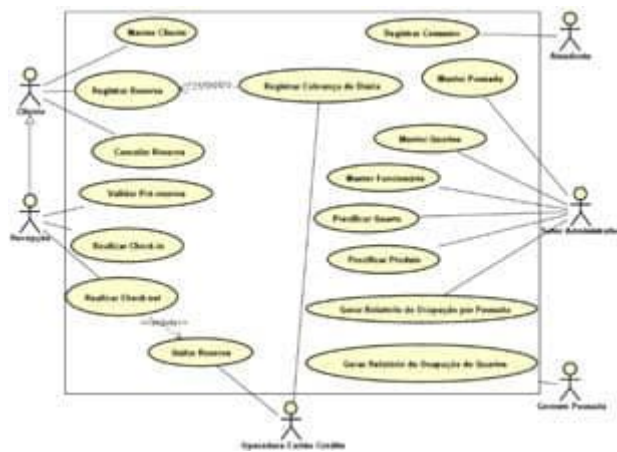


Diagrama de sequência parcial do caso de uso “Registrar Reserva” passo 3 gerado pela ferramenta Astah.

Finalmente, vamos analisar o passo 6, “**O sistema exige os dados de login do cliente para efetivação da reserva online**”.

Após uma quarta interação do ator, ou seja, informação de dados do login, ocorre a possibilidade de dois cenários estabelecidos pelos fluxos principal e alternativo, ou seja, reserva realizada pelo cliente (principal) ou pelo funcionário da recepção (alternativo). O operador “alt” permite representar essa abstração, pois caso o ator seja o cliente, será acionado o respectivo método de validação, e caso seja o funcionário, ocorrerá a respectiva validação de login, bem como a validação do CPF do cliente.

Na quinta interação, a confirmação da reserva, temos duas opções de acordo com o cenário; no cenário que a reserva é realizada pelo ator cliente, ocorre a extensão do caso de uso Registrar Cobrança de Diária, veja o diagrama de casos de uso do estudo de caso:



Extend em diagrama de casos de uso do estudo de caso gerado pela ferramenta Astah.

Caso essa cobrança seja confirmada, ou a reserva seja realizada pelo ator funcionário da recepção (cenário alternativo), ocorre o registro da reserva. Após a aplicação das abstrações descritas, propomos o diagrama de sequência ilustrado a seguir.

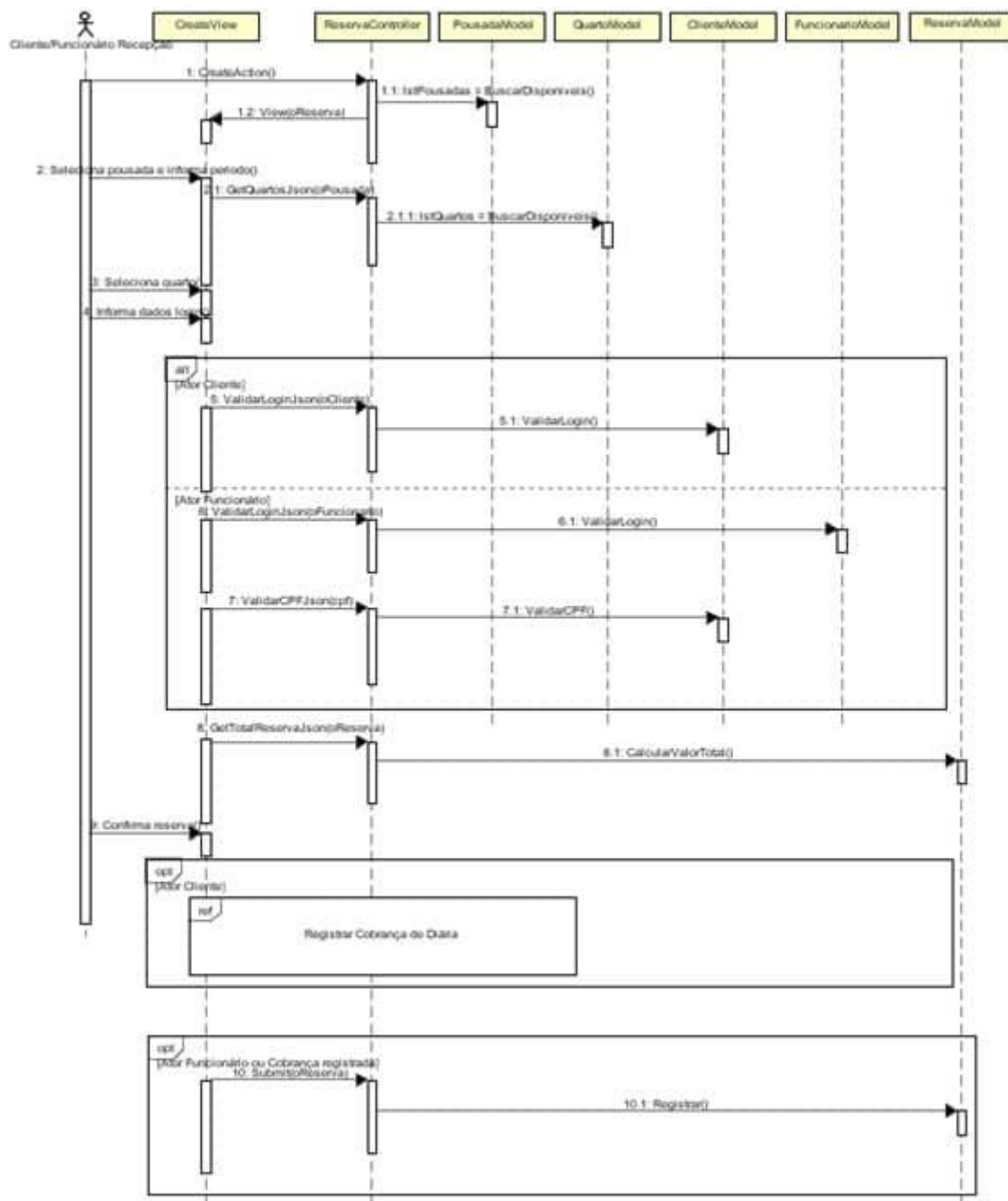


Diagrama de sequência do caso de uso “Registrar Reserva” gerado pela ferramenta Astah.

Diagrama de comunicação

Agora, vejamos a elaboração do diagrama de comunicação para o mesmo caso de uso, ou seja, “**Registrar Reserva**”. Nesse caso, podemos aplicar a mesma abstração aplicada na construção do diagrama de sequência, ou seja, a partir da descrição de caso de uso e do diagrama de classes identificar os objetos que fazem parte da solução do caso de uso e as respectivas trocas de mensagens, levando em consideração algum requisito não funcional relacionado com o padrão de arquitetura adotado (no caso MVC). A imagem a seguir ilustra o diagrama de comunicação proposto. Note que a sequência da execução é ordenada pelos números sequenciais das mensagens e o resultado das interações é o mesmo do diagrama de sequência.

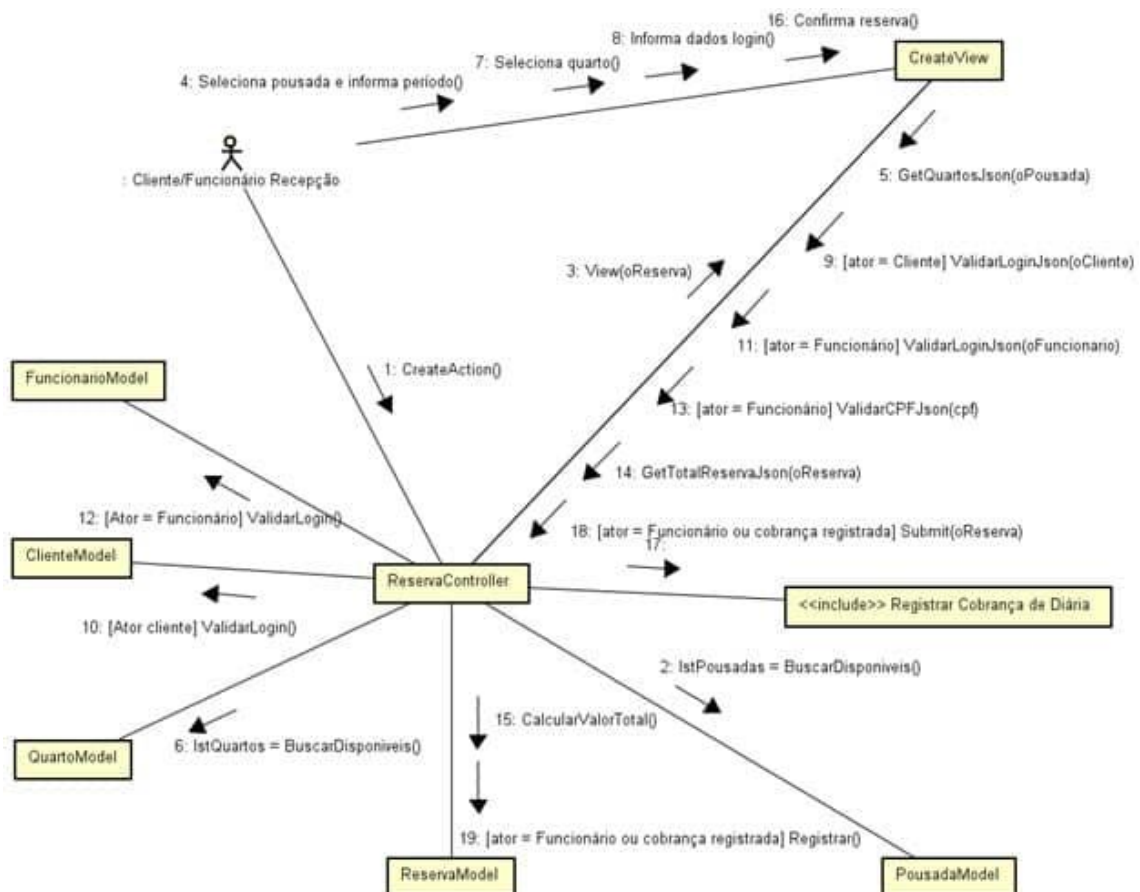


Diagrama de comunicação do caso de uso “Registrar Reserva” gerado pela ferramenta Astah.

MODELO DE CLASSES DE PROJETO

O modelo de classes de projeto é resultante de refinamentos no modelo de classes de análise, sendo incluídos detalhes úteis para a implementação das respectivas classes.

Dica

Como boa prática, sugerimos que esse modelo seja construído em paralelo com o modelo de interações.

ESTUDO DE CASO – MODELO DE CLASSES DE PROJETO

Aplicando os principais refinamentos no diagrama de classe de análise para o caso de uso Registrar Reserva ilustrado anteriormente no módulo 3, propomos o diagrama de classes de projeto apresentado a seguir. A sintaxe dos atributos foi definida no modelo de análise, sendo as visibilidades privativas (qualificador de visibilidade “-“).

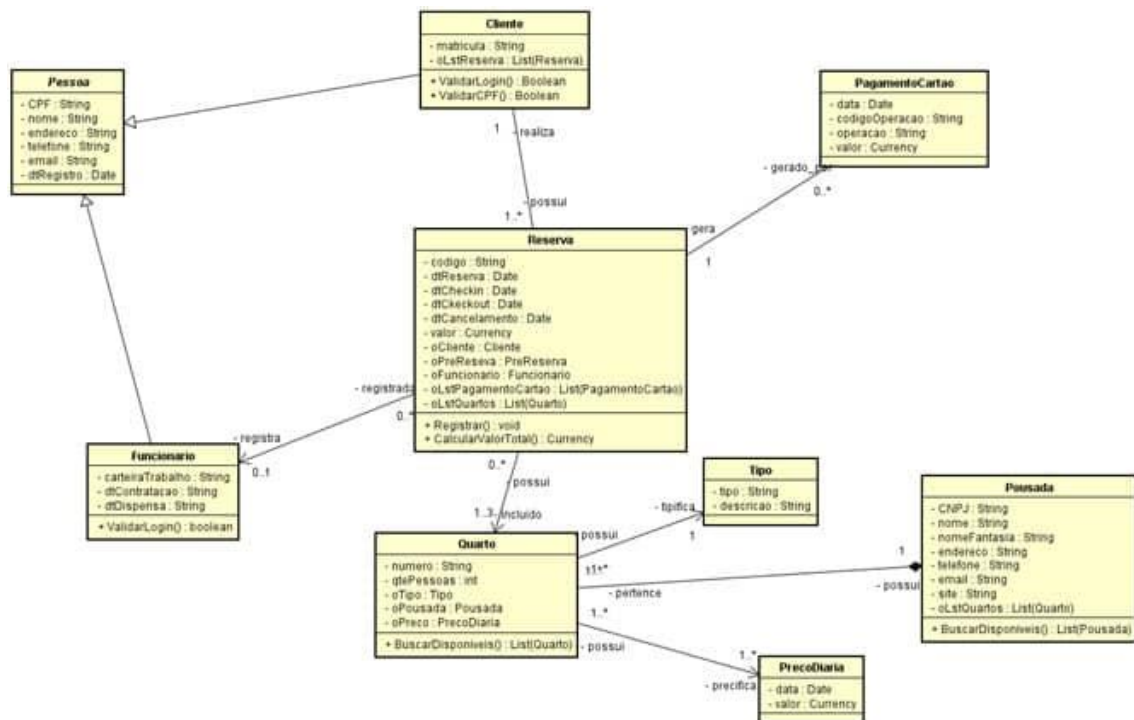


Diagrama de classes de projeto para o caso de uso Registrar Reserva gerado pela ferramenta Astah.

Podemos observar que as mensagens definidas no diagrama de sequência apresentado anteriormente foram incorporadas como métodos na respectivas classes, tal como a mensagem “10.1 Registrar()” implementada na classe destino “Reserva”, incluindo o tipo de visibilidade, no caso públicas (qualificador de visibilidade “+”), e tipo de retorno.

Vamos exemplificar refinamentos relacionados com a navegabilidade entre classes (repare nas multiplicidades de cada associação):

(a)

ocorrência de uma associação bidirecional entre Cliente/Reserva, pois a navegabilidade não está representada por flechas, ou seja, o envio de mensagens deverá ocorrer em ambos os sentidos, sendo que um objeto Cliente poderá instanciar uma coleção de objetos Reserva (veja atributo oLstReserva) e um objeto Reserva poderá instanciar um objeto Cliente (veja o atributo oCliente).

(b)

ocorrência de uma associação unidirecional entre Reserva/Quarto, pois a navegabilidade está representada por uma flecha, ou seja, o envio de mensagens deverá ocorrer em um único sentido, sendo que um objeto Reserva poderá instanciar uma coleção de objetos Quarto (veja atributo oLstQuarto).

Nosso diagrama de classes de projeto para o caso de uso Registrar Reserva permite ilustrar, também, o refinamento relacionado com o conceito de classe abstrata, que permite organizar hierarquias gen/spec e a implementação do princípio do polimorfismo. A classe Pessoa é uma classe abstrata estando a mesma em itálico.

Aplicando a abstração descrita para o caso de uso “Registrar Reserva” para os demais casos de uso, teremos o diagrama de projeto do sistema.

Dica

Os métodos de acesso get/set, ou as propriedades, não estão representados em função da geração automática pelos compiladores modernos a partir dos atributos.

MODELO DE IMPLEMENTAÇÃO

O modelo de implementação é composto de um ou mais diagramas de componentes, que permite representar, de forma gráfica, os componentes do sistema e suas dependências.

ESTUDO DE CASO – MODELO DE IMPLEMENTAÇÃO

Retornemos ao nosso estudo de caso, ainda na fase de projeto do processo. A imagem a seguir ilustra uma proposta para o diagrama de componentes, considerando os modelos de casos de uso (módulo 2), de classes (módulo 3) e de interação (módulo 4).

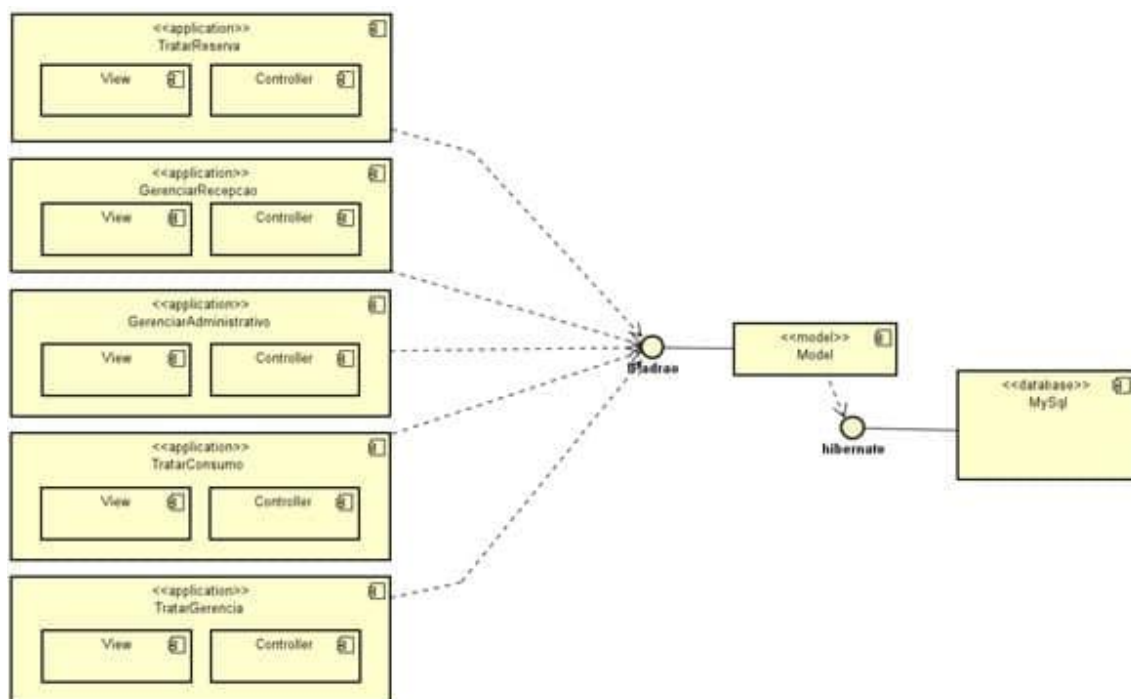


Diagrama de componentes para o estudo de caso gerado pela ferramenta Astah.

A partir do empacotamento de casos de uso, propomos uma solução (entre outras muitas!) onde cada pacote corresponde a uma aplicação. Cada aplicação possui suas próprias classes controladoras (pacote Controller) e views (pacote View); entretanto, as classes do tipo model (**stereotype**) são acessadas por todas as aplicações por uma interface IPadrao. Como especificado no requisito não funcional RNF 2 no módulo 1, o banco de dados é o MySQL, sendo o framework de persistência o Hibernate (RNF 4). Lembramos que esse diagrama representa um alto nível de abstração, pois podemos detalhar cada aplicação com um respectivo diagrama de componentes.

MODELO DE IMPLANTAÇÃO

O modelo de implantação é composto de um ou mais [diagramas de implantação](#) e define os nós de processamento disponíveis, ou seja, os componentes físicos do sistema e suas interdependências. A referida modelagem cresce de importância em função da complexidade do sistema.

ESTUDO DE CASO – MODELO DE IMPLANTAÇÃO

A imagem a seguir ilustra uma proposta de diagrama de implantação para o nosso estudo de caso, sendo necessários apenas três nós computacionais para sua implantação.

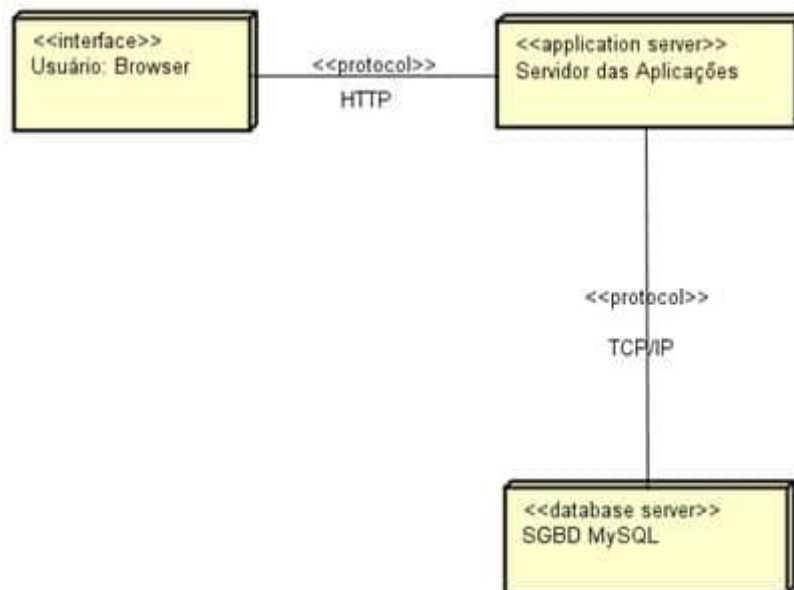


Diagrama de

implantação para o estudo de caso gerado pela ferramenta Astah.

Neste módulo, demos continuidade ao estudo de caso que trata do gerenciamento de uma rede de pousadas desenvolvendo os principais modelos da etapa de projeto do processo de desenvolvimento de software genérico.

Neste módulo vimos os seguintes modelos:

Modelo de interação

Permite identificar os objetos e respectivas interações, por meio de mensagens, na realização de casos de uso, sendo esse modelo composto por diagramas de sequência e/ou diagramas de comunicação.

Modelo de classes de projeto

É gerado a partir do refinamento do modelo de classes de análise, ou seja, transformações aplicadas nos atributos, operações e associações. Esse modelo é construído em paralelo com o modelo de interações, contendo detalhes úteis para a implementação das classes nele contidas.

Modelo de implementação

Composto de diagramas de componentes, permite identificar os componentes e suas dependências.

Modelo de implantação

Composto por um ou mais diagramas de implantação, define os nós **computacionais necessários à implantação do sistema.**