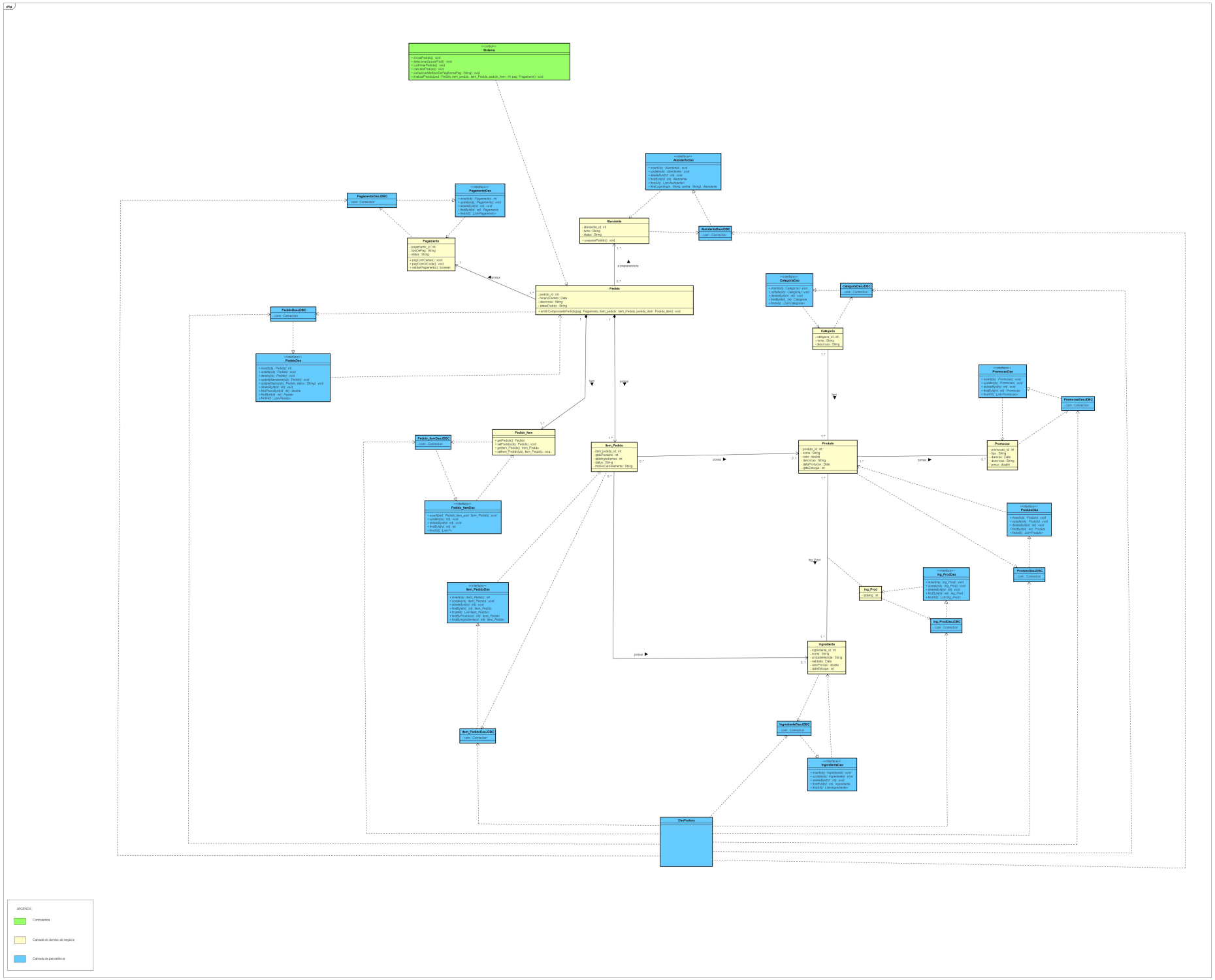
pkg
<<controller>>
Sistema
+ iniciarPedido() : void
+ adicionarItem(id) : void
+ confirmarPedido() : void
+ cancelarPedido() : void
+ comunicarInterfaceConfigRaspberryPag (String) : void
+ finalizarPedido(ped: Pedido, item_pedido: Item_Pedido, pedido_item: int, pag: Pagamento): void
<<interface>>
AtendenteDao
+ insertby(a: Atendente): void
+ updateby(a: Atendente): void
+ deleteby(id: int): void
+ findById(id: int): Atendente
+ findAll(): List<Atendente>
+ findLoginLogin(: String, senha: String): Atendente
AtendenteDaoJDBC
- conn: Connection
Atendente
- atendente_id: int
- login: String
- senha: String
+ prepararPedido(): void
é preparado por
1..*
0..*
<<interface>>
PagamentoDao
+ insertby(p: Pagamento): void
+ updateby(p: Pagamento): void
+ deleteby(id: int): void
+ findById(id: int): Pagamento
+ findAll(): List<Pagamento>
PagamentoDaoJDBC
- conn: Connection
Pagamento
- pagamento_id: int
- tipoDePag: String
- status: String
+ pagCom Cartao(): void
+ pagCom Cb(cb): void
+ validarPagamento(): boolean
<<value>>
1
1..*
Pedido
- pedido_id: int
- horaInsPedido: Date
- descricao: String
- statusPedido: String
+ emitirCorporativoPedido(pag: Pagamento, item_pedido: Item_Pedido, pedido_item: Pedido_Item): void
PedidoDaoJDBC
- conn: Connection
<<interface>>
PedidoDao
+ insertby(p: Pedido): void
+ updateby(p: Pedido): void
+ deleteby(p: Pedido): void
+ updateAtendente(p: Pedido): void
+ updateStatus(p: Pedido, status: String): void
+ deleteAll(id: int): void
+ findById(id: int): Pedido
+ findPedido(id: int): Pedido
+ findAll(): List<Pedido>
<<interface>>
CategoriaDao
+ insertby(c: Categoria): void
+ updateby(c: Categoria): void
+ deleteby(id: int): void
+ findById(id: int): Categoria
+ findAll(): List<Categoria>
CategoriaDaoJDBC
- conn: Connection
Categoria
- categoria_id: int
- nome: String
- descricao: String
1..*
tem
<<interface>>
PromocaoDao
+ insertby(p: Promocao): void
+ updateby(p: Promocao): void
+ deleteby(id: int): void
+ findById(id: int): Promocao
+ findAll(): List<Promocao>
PromocaoDaoJDBC
- conn: Connection
Promocao
- promocao_id: int
- tipo: String
- duracao: Date
- descricao: String
- preco: double
0..1
possui
1..*
Produto
- produto_id: int
- nome: String
- valor: double
- descricao: String
- dataProducao: Date
- qtdeEstoque: int
1..*
<<interface>>
ProdutoDao
+ insertby(p: Produto): void
+ updateby(p: Produto): void
+ deleteby(id: int): void
+ findById(id: int): Produto
+ findAll(): List<Produto>
ProdutoDaoJDBC
- conn: Connection
Pedido_ItemDaoJDBC
- conn: Connection
Pedido_Item
- getPedido(): Pedido
- setPedido(p): Pedido: void
- getItem_Pedido(): Item_Pedido
- setItem_Pedido(p): Item_Pedido: void
<<interface>>
Pedido_ItemDao
+ insertped: Pedido, item_ped: Item_Pedido): void
+ deleteby(id): void
+ deleteAll(id): void
+ findById(id): int
+ findAll(): List<Int>
Item_Pedido
- item_pedido_id: int
- quantidade: int
- qtdeIngredientes: int
- status: String
- notesCancelamento: String
0..*
possui
1..*
<<interface>>
Ing_ProdDao
+ insertby(i: Ing_Prod): void
+ updateby(i: Ing_Prod): void
+ deleteby(id: int): void
+ findById(id: int): Ing_Prod
+ findAll(): List<Ing_Prod>
Ing_ProdDaoJDBC
- conn: Connection
Ing_Prod
- qtdeIng: int
Ing_Prod
<<interface>>
Item_PedidoDao
+ insertby(i: Item_Pedido): int
+ updateby(i: Item_Pedido): void
+ deleteby(id: int): void
+ findById(id: int): Item_Pedido
+ findAll(): List<Item_Pedido>
+ findByProduto(id: int): Item_Pedido
+ findByIngrediente(id: int): Item_Pedido
Item_PedidoDaoJDBC
- conn: Connection
possui
0..1
Ingrediente
- ingrediente_id: int
- nome: String
- unidadeMedida: String
- validade: Date
- valorUnitario: double
- qtdeEstoque: int
IngredienteDaoJDBC
- conn: Connection
<<interface>>
IngredienteDao
+ insertby(i: Ingrediente): void
+ updateby(i: Ingrediente): void
+ deleteby(id: int): void
+ findById(id: int): Ingrediente
+ findAll(): List<Ingrediente>
DaoFactory
LEGENDA:
Controladora
Camada do domínio do negócio
Camada de persistência