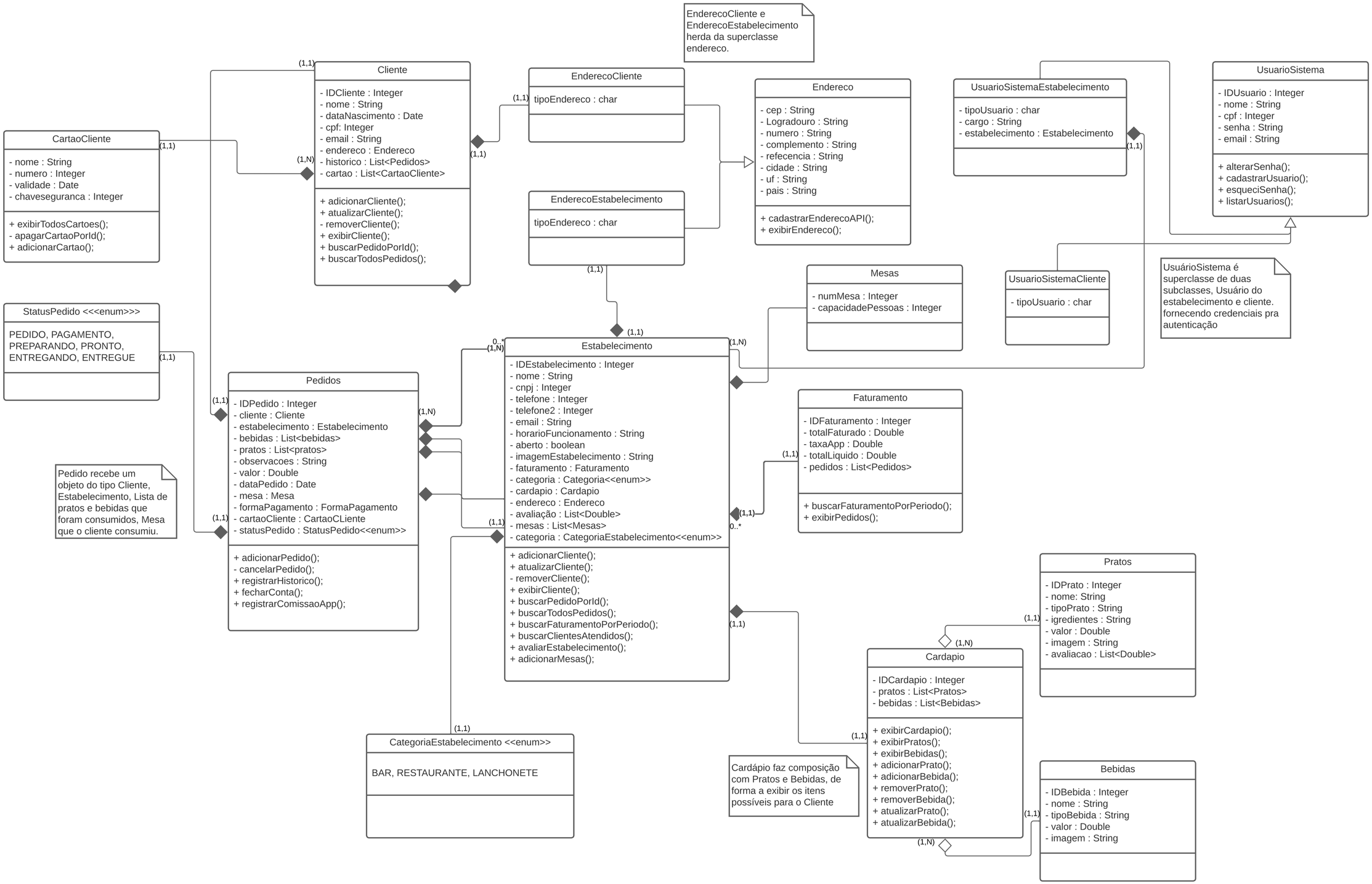
EnderecoCliente e EnderecoEstabelecimento herda da superclasse endereco.
Cliente
- IDCliente : Integer
- nome : String
- dataNascimento : Date
- cpf: Integer
- email : String
- endereco : Endereco
- historico : List<Pedidos>
- cartao : List<CartaoCliente>
+ adicionarCliente();
+ atualizarCliente();
- removerCliente();
+ exibirCliente();
+ buscarPedidoPorId();
+ buscarTodosPedidos();
EnderecoCliente
tipoEndereco : char
Endereco
- cep : String
- Logradouro : String
- numero : String
- complemento : String
- refecencia : String
- cidade : String
- uf : String
- pais : String
+ cadastrarEnderecoAPI();
+ exibirEndereco();
UsuarioSistemaEstabelecimento
- tipoUsuario : char
- cargo : String
- estabelecimento : Estabelecimento
UsuarioSistema
- IDUsuario : Integer
- nome : String
- cpf : Integer
- senha : String
- email : String
+ alterarSenha();
+ cadastrarUsuario();
+ esqueciSenha();
+ listarUsuarios();
CartaoCliente
- nome : String
- numero : Integer
- validade : Date
- chaveseguranca : Integer
+ exibirTodosCartoes();
- apagarCartaoPorId();
+ adicionarCartao();
EnderecoEstabelecimento
tipoEndereco : char
Mesas
- numMesa : Integer
- capacidadePessoas : Integer
UsuarioSistemaCliente
- tipoUsuario : char
UsuáriosSistema é superclasse de duas subclasses, Usuário do estabelecimento e cliente. fornecendo credenciais pra autenticação
StatusPedido <<<enum>>>
PEDIDO, PAGAMENTO, PREPARANDO, PRONTO, ENTREGANDO, ENTREGUE
Estabelecimento
- IDEstabelecimento : Integer
- nome : String
- cnpj : Integer
- telefone : Integer
- telefone2 : Integer
- email : String
- horarioFuncionamento : String
- aberto : boolean
- imagemEstabelecimento : String
- faturamento : Faturamento
- categoria : Categoria<<enum>>
- cardapio : Cardapio
- endereco : Endereco
- avaliação : List<Double>
- mesas : List<Mesas>
- categoria : CategoriaEstabelecimento<<enum>>
+ adicionarCliente();
+ atualizarCliente();
- removerCliente();
+ exibirCliente();
+ buscarPedidoPorId();
+ buscarTodosPedidos();
+ buscarFaturamentoPorPeriodo();
+ buscarClientesAtendidos();
+ avaliarEstabelecimento();
+ adicionarMesas();
Pedidos
- IDPedido : Integer
- cliente : Cliente
- estabelecimento : Estabelecimento
- bebidas : List<bebidas>
- pratos : List<pratos>
- observacoes : String
- valor : Double
- dataPedido : Date
- mesa : Mesa
- formaPagamento : FormaPagamento
- cartaoCliente : CartaoCLiente
- statusPedido : StatusPedido<<enum>>
+ adicionarPedido();
- cancelarPedido();
+ registrarHistorico();
+ fecharConta();
+ registrarComissaoApp();
Pedido recebe um objeto do tipo Cliente, Estabelecimento, Lista de pratos e bebidas que foram consumidos, Mesa que o cliente consumiu.
Faturamento
- IDFaturamento : Integer
- totalFaturado : Double
- taxaApp : Double
- totalLiquido : Double
- pedidos : List<Pedidos>
+ buscarFaturamentoPorPeriodo();
+ exibirPedidos();
Pratos
- IDPrato : Integer
- nome: String
- tipoPrato : String
- igredientes : String
- valor : Double
- imagem : String
- avaliacao : List<Double>
Cardapio
- IDCardapio : Integer
- pratos : List<Pratos>
- bebidas : List<Bebidas>
+ exibirCardapio();
+ exibirPratos();
+ exibirBebidas();
+ adicionarPrato();
+ adicionarBebida();
+ removerPrato();
+ removerBebida();
+ atualizarPrato();
+ atualizarBebida();
Cardápio faz composição com Pratos e Bebidas, de forma a exibir os itens possíveis para o Cliente
Bebidas
- IDBebida : Integer
- nome : String
- tipoBebida : String
- valor : Double
- imagem : String
CategoriaEstabelecimento <<enum>>
BAR, RESTAURANTE, LANCHONETE
(1,1)
(1,N)
0..*