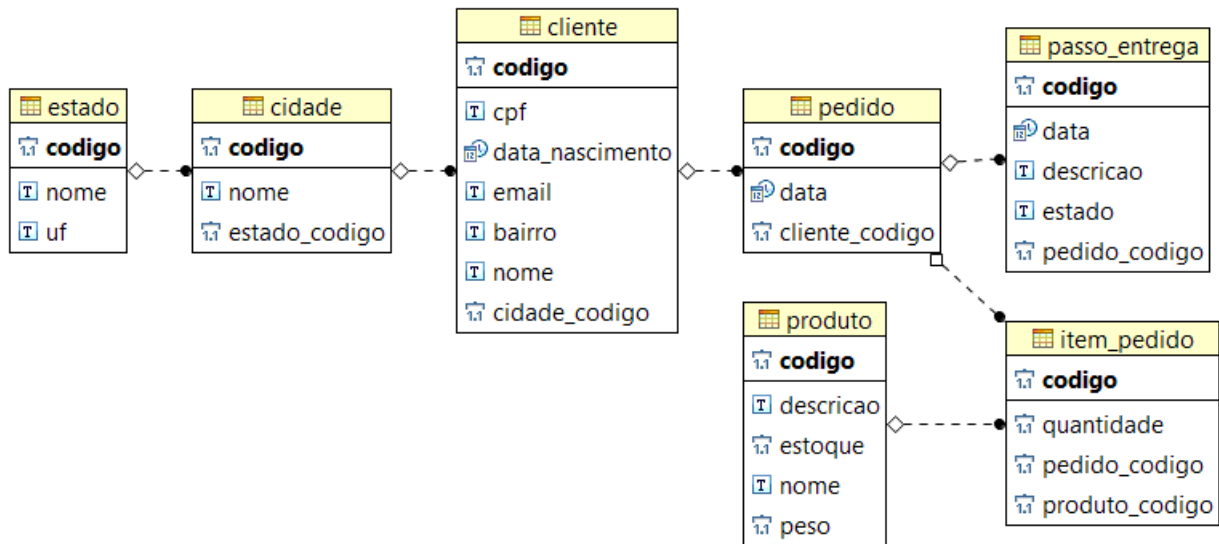


Sistema de Pedidos

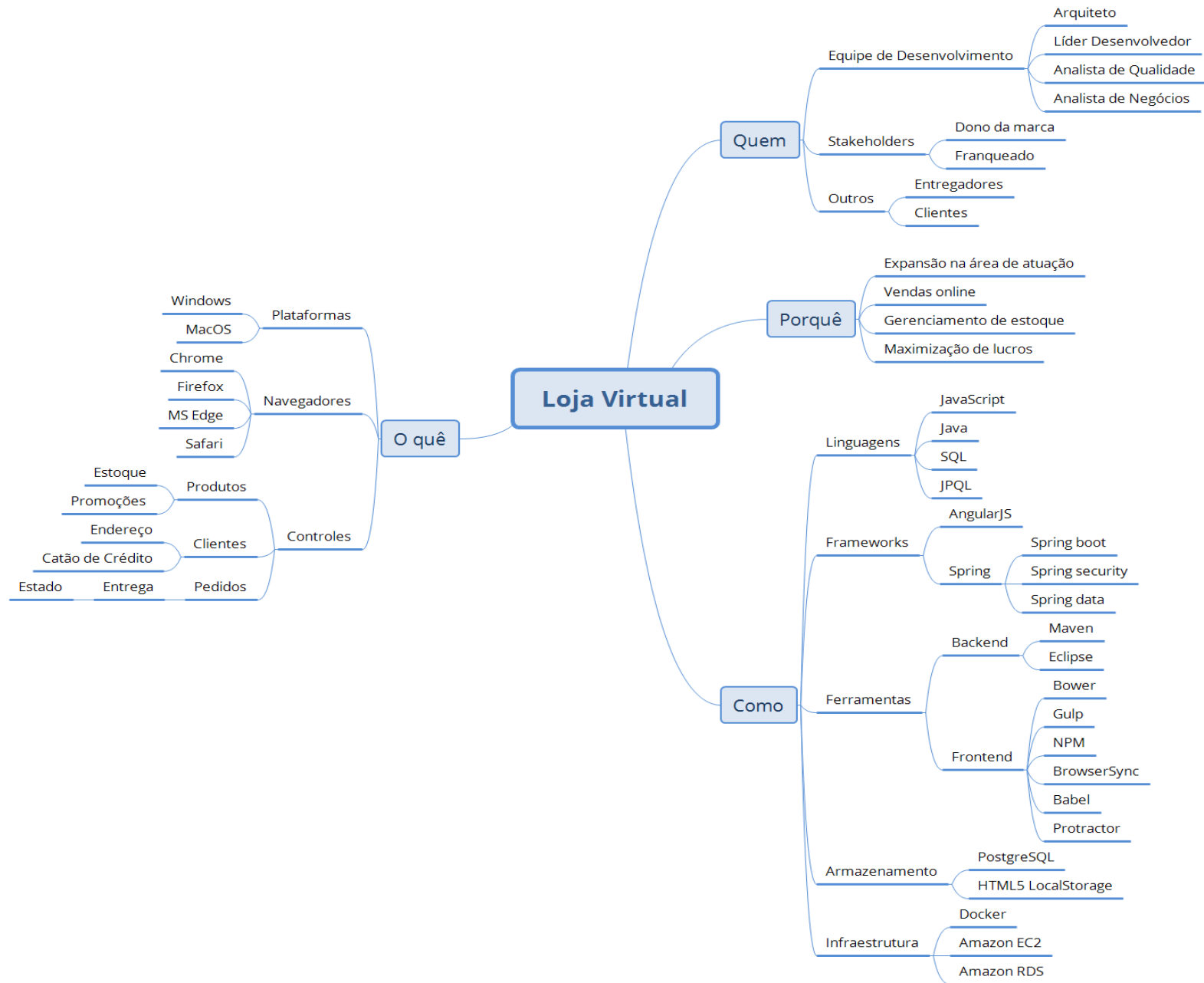
DEMO/Código-fonte

- <https://unopar.herokuapp.com/>
- <https://github.com/marcelothebuilder/UNOPAR-pedidos>

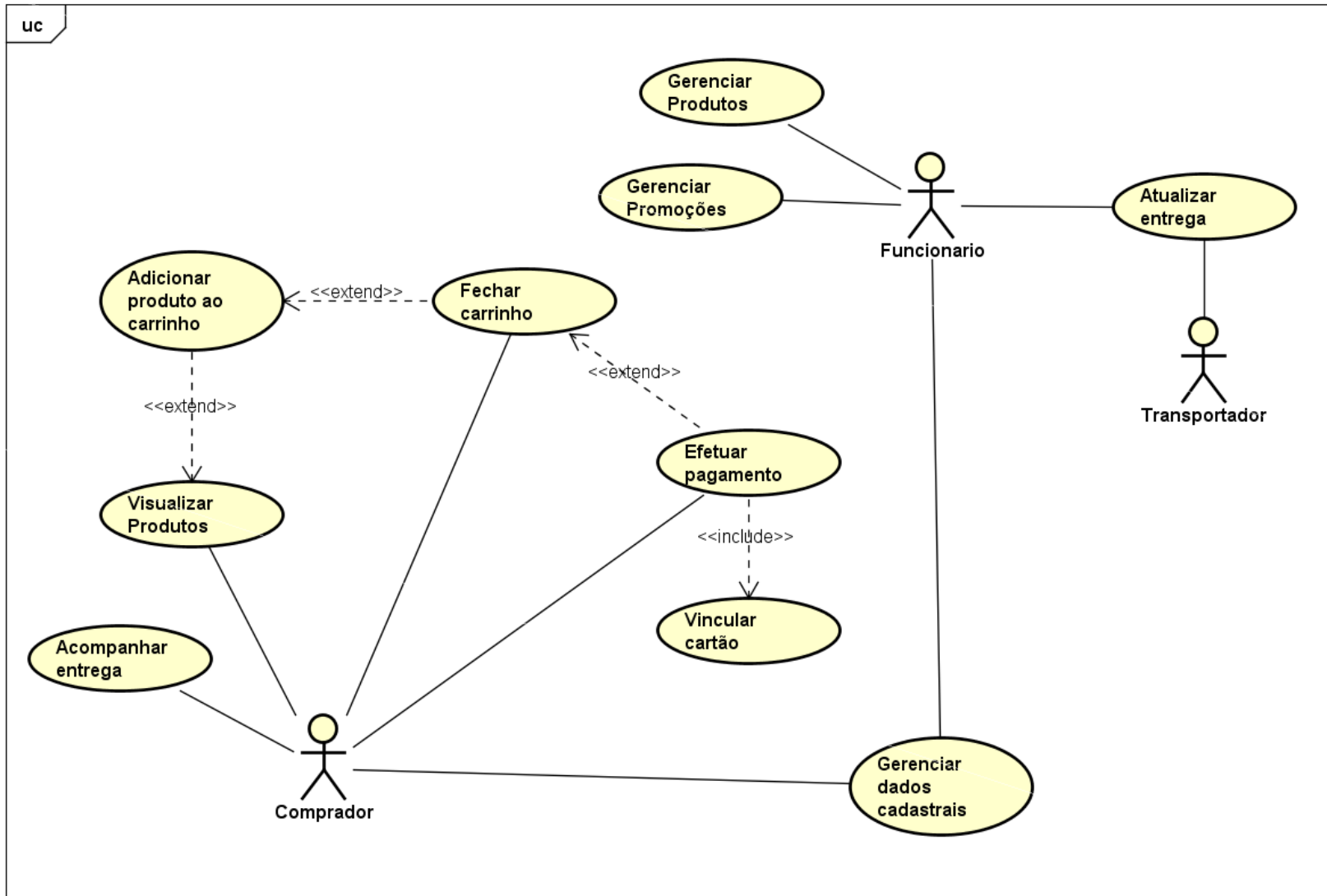
Database (Modelo)



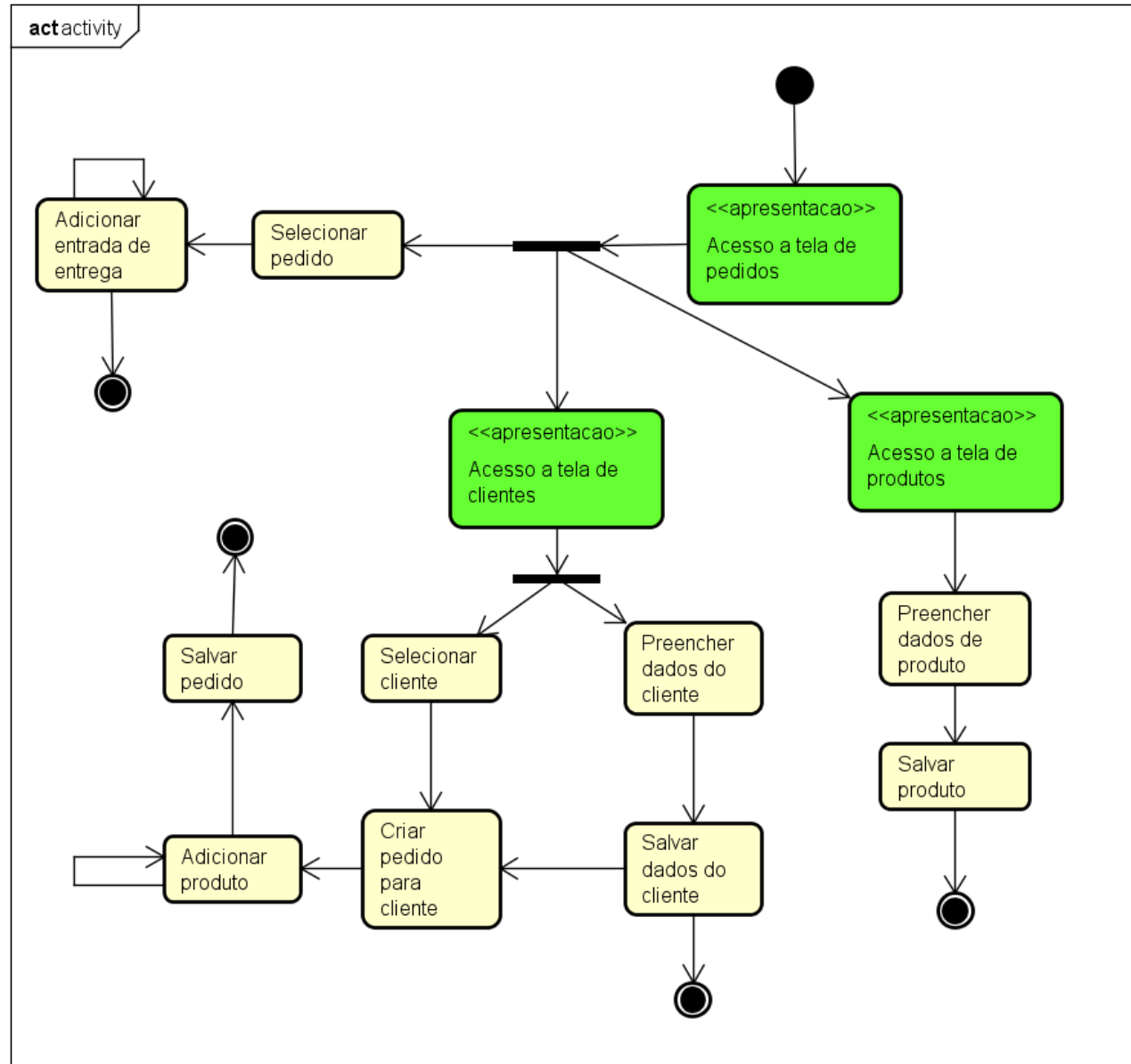
Mindmaps – Entrega de Valor



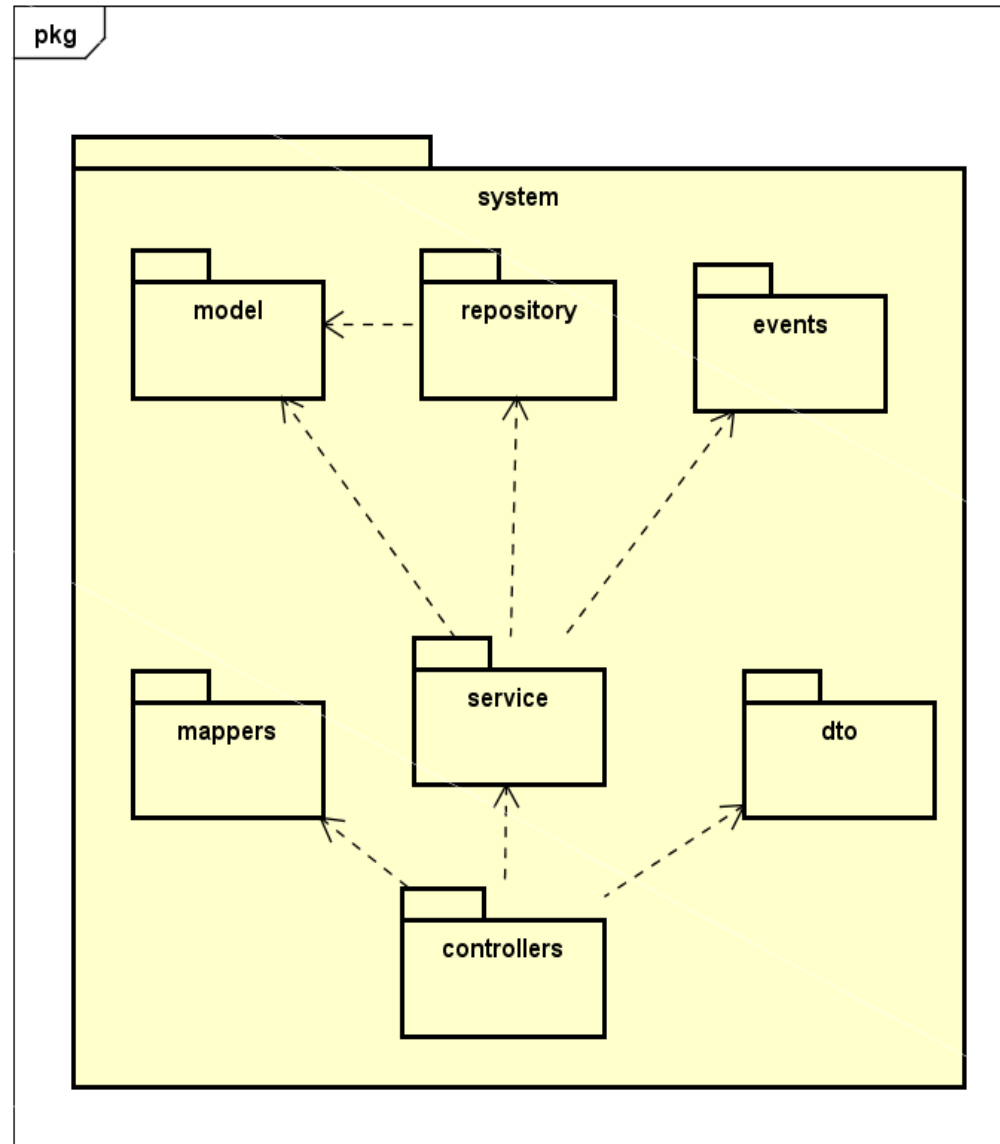
UML - UseCase



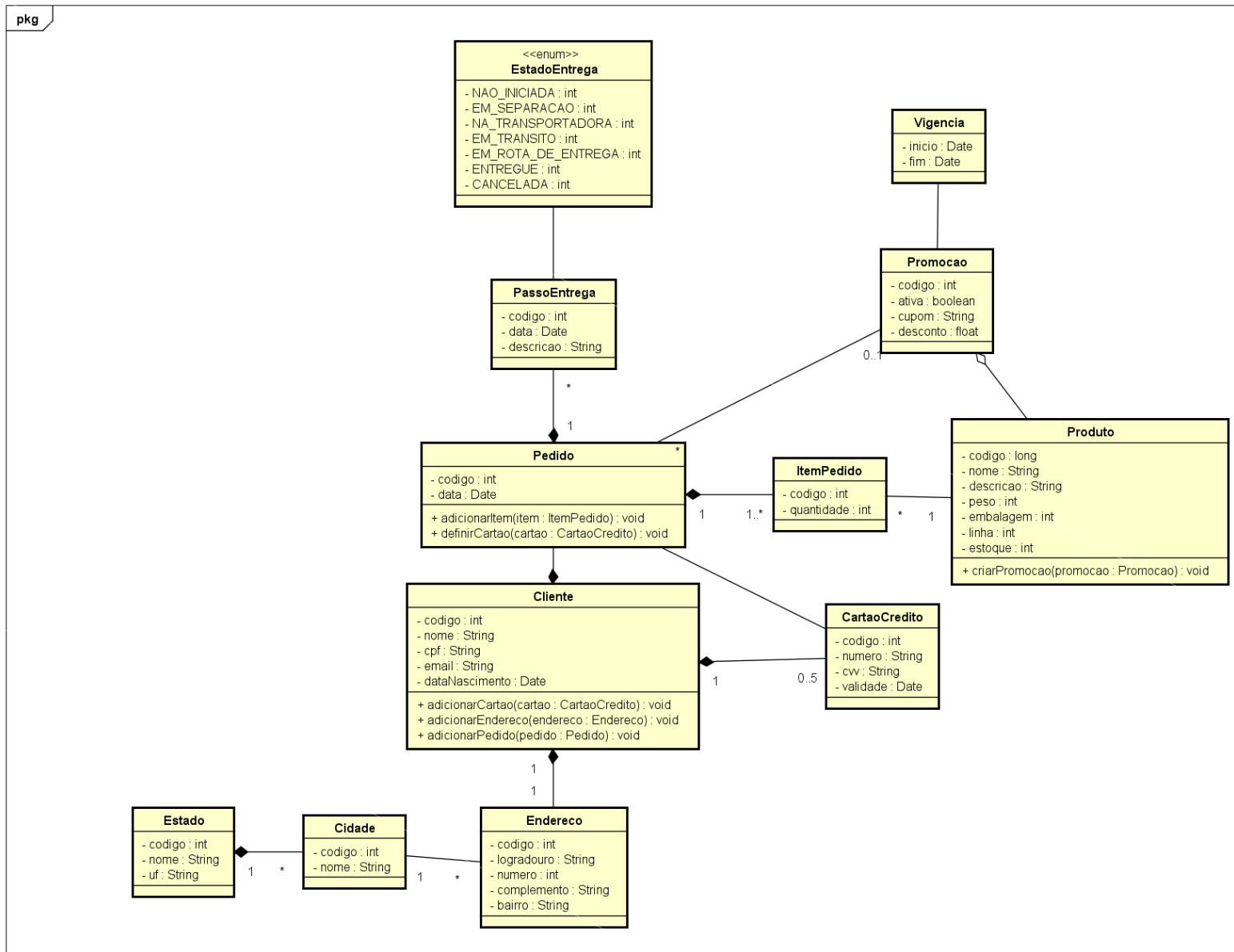
UML - Atividade



UML – Diagrama Pacotes



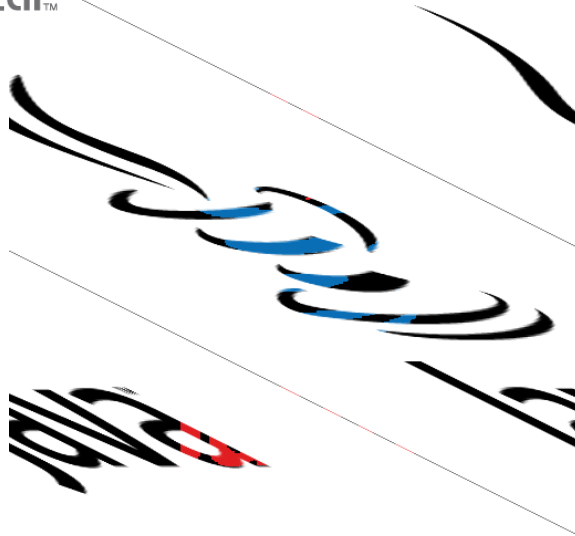
Classes



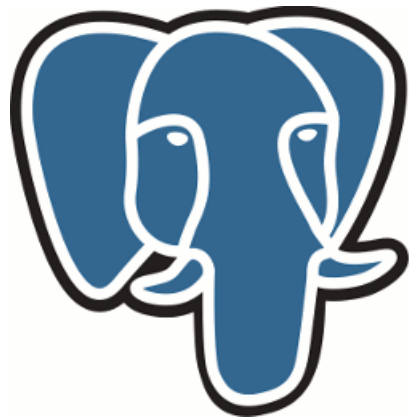
Front-end



Backend

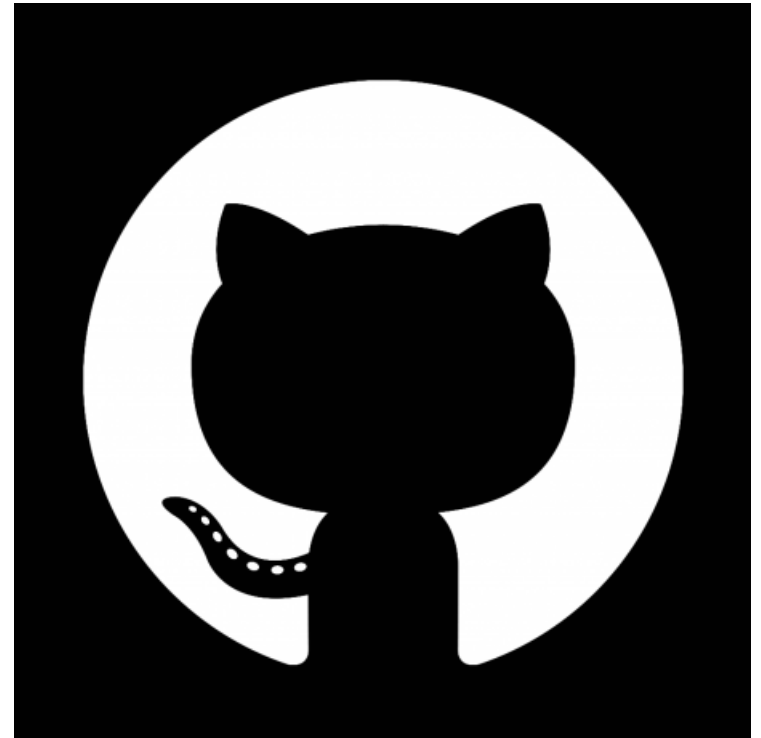
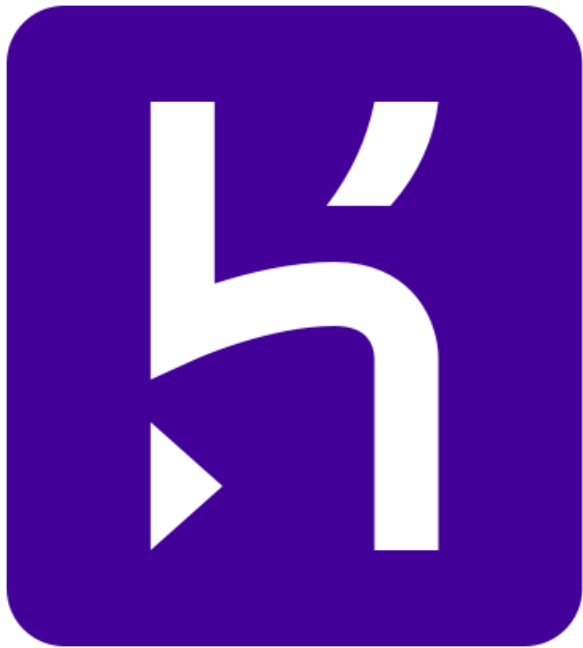


Banco de dados

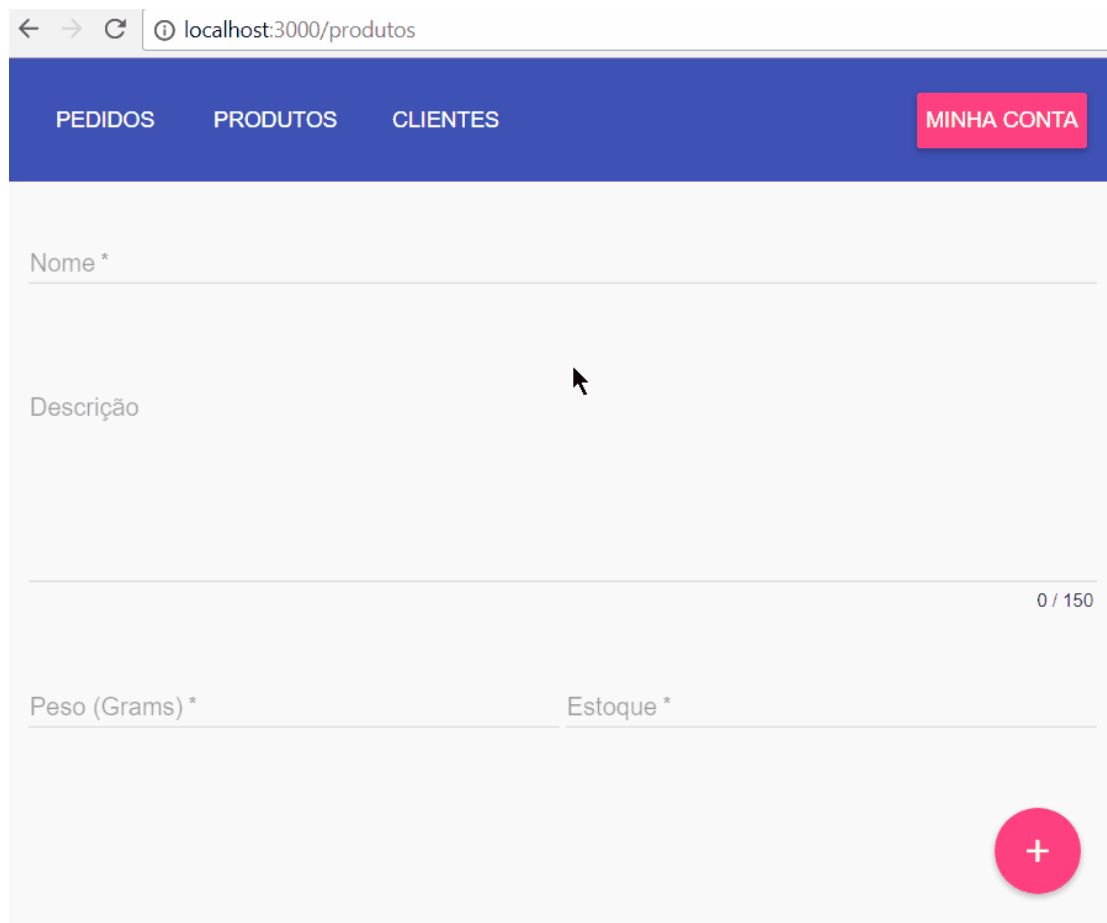


PostgreSQL
the world's most advanced open source database

Infraestrutura/Cloud/Versionamento



Cadastro de Produto



A screenshot of a web browser displaying a product registration form. The browser's address bar shows the URL `localhost:3000/produtos`. The page features a dark blue header with navigation links: **PEDIDOS**, **PRODUTOS**, and **CLIENTES**. A pink button labeled **MINHA CONTA** is positioned on the right side of the header. The main content area is light gray and contains the following form fields:

- Nome ***: A text input field.
- Descrição**: A large text area for product description, with a character count of **0 / 150** displayed at the bottom right.
- Peso (Grams) ***: A text input field.
- Estoque ***: A text input field.

A pink circular button with a white plus sign (+) is located in the bottom right corner of the form area.

Cadastro de Cliente

← → ↻ localhost:3000/produtos

PEDIDOS

PRODUTOS

CLIENTES

MINHA CONTA

Nome *

Descrição

0 / 150

Peso (Grams) *

Estoque *

+

Cadastro de Pedido

← → ↻ ⓘ localhost:3000/produtos

PEDIDOS PRODUTOS CLIENTES MINHA CONTA

Nome *

Descrição

0 / 150

Peso (Grams) * Estoque *

+

```
#include <exemplos>
```



```
3  @Entity
4  @Data
5  public class Cliente {
6      @Id
7      @GeneratedValue(strategy = GenerationType.SEQUENCE)
8      private Long codigo;
9
10     @NotBlank
11     private String nome;
12
13     @NotBlank
14     private String cpf;
15
16     @NotBlank
17     private String email;
18
19     @NotNull
20     @Temporal(TemporalType.DATE)
21     private Date dataNascimento;
22
23     @NotNull
24     @Embedded
25     private Endereco endereco;
26
27 }
28
```

```
18  @RestController
19  @RequestMapping("/produto")
20  public class ProdutoController {
21
22      @Autowired
23      private ProdutoRepository repository;
24
25      @Autowired
26      private ProdutoMapper mapper;
27
28      @PostMapping
29      public void novo(@Valid @RequestBody ProdutoDTO dto) {
30          repository.save(mapper.produtoDTOToProduto(dto));
31      }
32
33      @GetMapping
34      public List<ProdutoDTO> todos() {
35          return mapper.produtosToProdutoDTOs(repository.findAll());
36      }
37
38      @GetMapping(params = "nome")
39      public Page<ProdutoDTO> pesquisar(@RequestParam(name = "nome", required = false)
40          String nome, Pageable pagina) {
41          if (StringUtils.isEmpty(nome)) {
42              return repository.findAll(pagina).map(mapper::produtoToProdutoDTO);
43          }
44          return repository.findByNomeContainingIgnoreCase(nome, pagina)
45              .map(mapper::produtoToProdutoDTO);
46      }
47  }
48
49
```



That's all Folks!