

|  |  |  |
| --- | --- | --- |
| **Sprint Documentation** | | |
| **Sprint #02** | **Start Date: 29/09/2024** | **Final Date: 12/10/2024** | |
| **Team Members** | | | |
| Edson Pereira De Carvalho | | | |
| Lemuel Prates Sobrinho Ferreira | | | |
| Maria Clara Borges Cardoso | | | |
| Rodrigo Yoshida Lombezzi | | | |
| Vinicius Lúcio Marcolino Da Silva | | | |

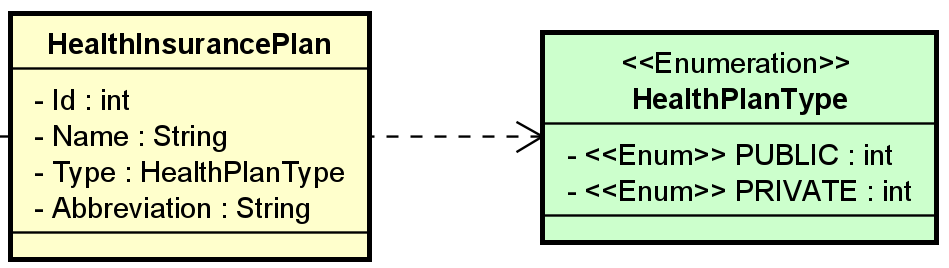
|  |  |  |  |
| --- | --- | --- | --- |
| **Sprint Backlog** | | | |
| Task# | Description | Start Date | Final date |
| 40 | Sprint 02 - Desenvolver a API para HealthInsurancePlan | 01/10/2024 | 07/10/2024 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Task Description** | | | | | |
| Task # | Description | Assigned To | Status | Estimated Hours | Logged Hours |
| 40 | Sprint 02 - Desenvolver a API para HealthInsurancePlan | Lemuel | Concluído | 8 | 6 |

|  |
| --- |
| **Sprint Results** |

Foi desenvolvido uma API para a classe HealthInsurancePlan. Esse processo envolveu a criação de diversos arquivos nas diferentes camadas do sistema para que as funções executassem corretamente.

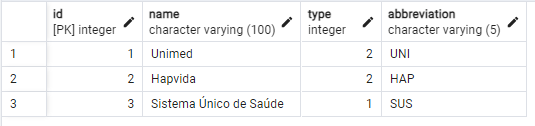
Primeiramente foram criados a model e o DTO conforme a seguinte representação no diagrama de classes:



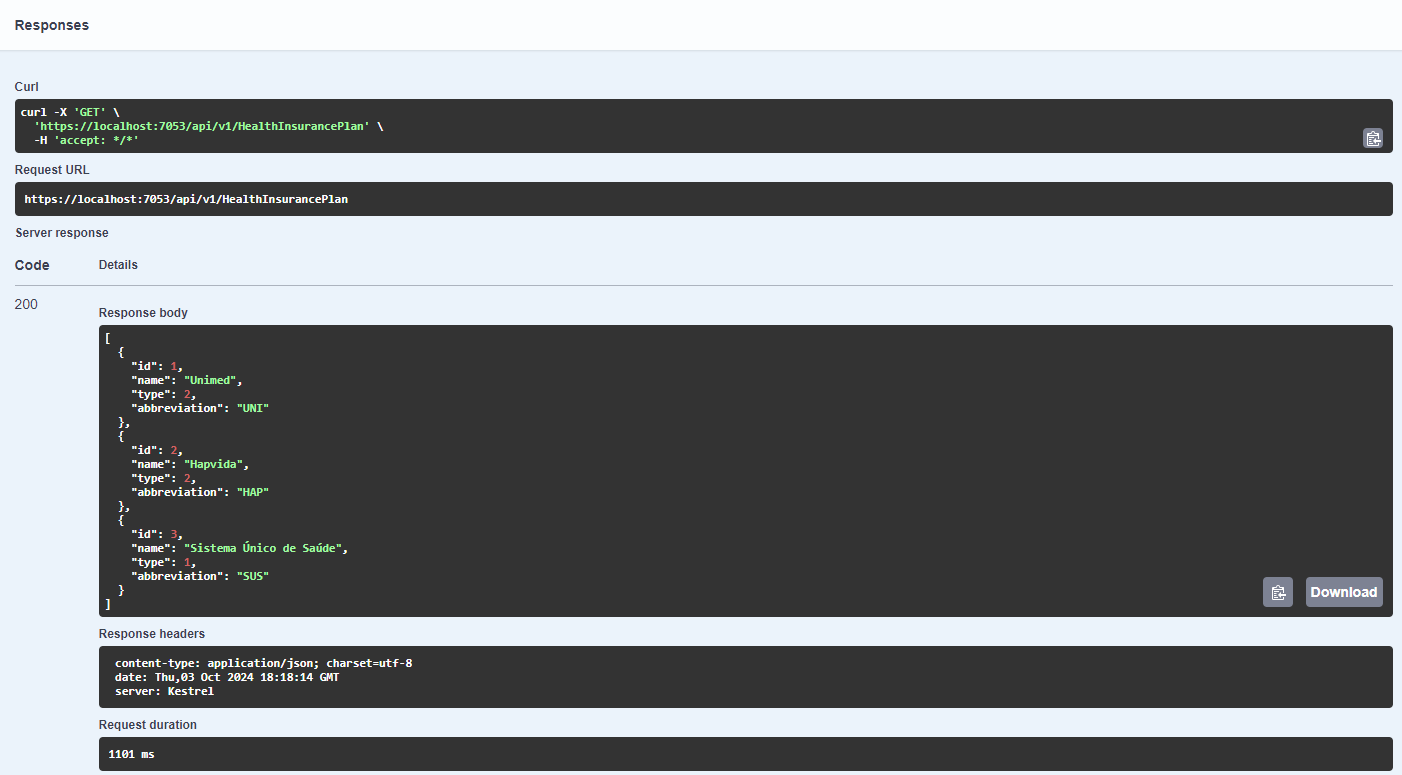
Os atributos são:

* Id, do tipo int, representa a chave primária dos registros nessa classe;
* Name, do tipo string, Varchar(100) no banco de dados, representa o nome fantasia do plano de saúde;
* Type, do tipo HealthPlanType, Varchar(1) no banco de dados, indica se o plano é público ou privado;
* Abbreviation, do tipo string, Varchar(5) no banco de dados, representa a abreviação do nome do plano de saúde.

Em seguida, foram criadas as camadas Data, Service e a Controller, e inseridos os seguintes registros como carga inicial de dados para a classe.



Na controller, as seguintes requisições são aceitas:

**HttpGet**: para buscar todos os registros da classe. 

**HttpGet com id**: para buscar um registro a partir do id fornecido. 

**HttpPost**: para criar um novo registro no banco de dados. 

**HttpPut**: para atualizar um registro a partir do id e dos dados fornecidos. 

**HttpDelete**: para remover um registro existente a partir do id passado na url da requisição. 