**STEPES-TR**

**Instituto Tecnológico de Aeronáutica - ITA**

**Definição da Estrutura Base do Banco de Dados - Módulo Analítico**

**TS#01 - US#127 - Sprint 1**

(Versão 0.1 12-10-2020)

Renata Bricio Vianna

Aline do Nascimento Rodrigues

1. **Objetivo**

O objetivo desta US é fazer o levantamento e análise dos requisitos para definir uma estrutura base para o armazenamento dos dados do projeto STEPES-TR.

1. **Levantamento e Análise dos Requisitos**

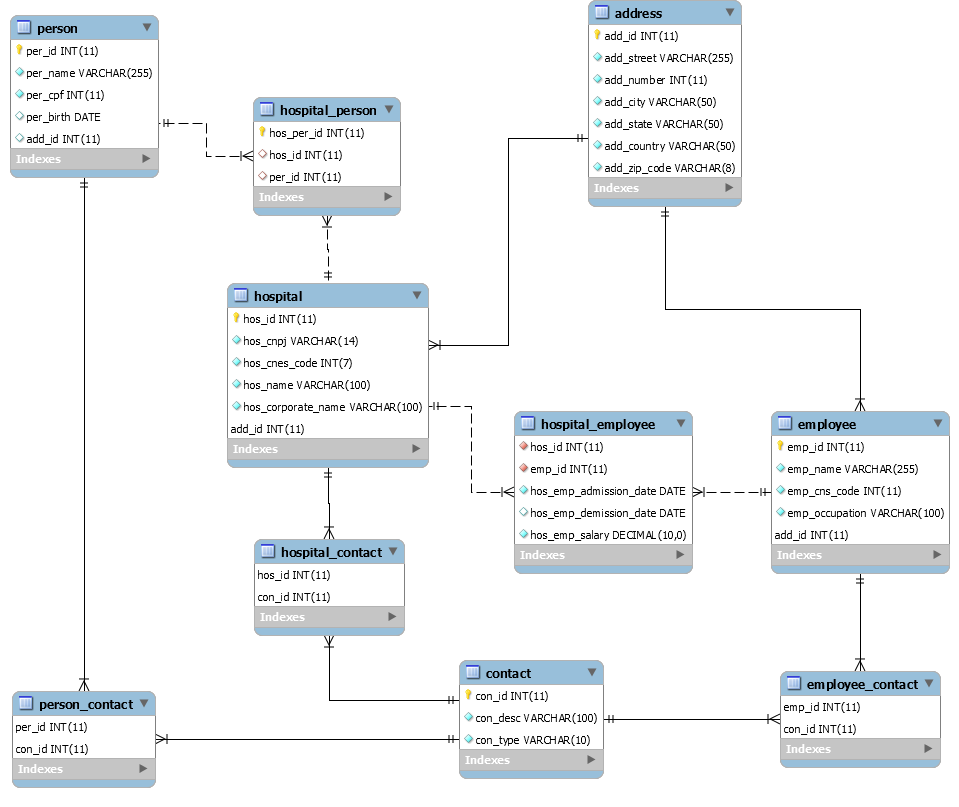
Através das Users Stories definidas no Product Backlog e reuniões realizadas com os integrantes dos demais times foram feitos o levantamento e a análise dos requisitos para a coleta de informações relevantes na definição dos dados, o banco de dados a ser utilizado bem como a sua modelagem.

Algumas questões foram levantadas relacionadas as seguintes USs:

* **US#128**: De acordo com a arquitetura definida do projeto, as informações de status do hardware não serão persistidas no banco de dados, pois elas serão necessárias somente para monitoramento em tempo real.

1. **Definição e Modelagem do Banco de Dados**

Nos dados de Paciente e Hospital que incluem gerenciamento de empregados, endereços e contatos foi definido que será reutilizado parte a modelagem feita no projeto STEPES-BD. O Banco de Dados utilizado foi o **MySQL**. A figura a seguir mostra o modelo ER inicial definido para o projeto.



A tabela a seguir mostra a descrição das tabelas

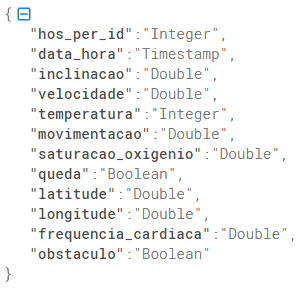
|  |  |
| --- | --- |
| **Tabela** | **Descrição** |
| address | Informações do endereço (hospital, empregado, paciente) |
| contact | Informações de contato (hospital, empregado) |
| hospital | informações do hospital |
| employee | informações do empregado |
| person | informações do paciente |
| hospital\_contact | referência de contato com hospital |
| employee\_contact | referência de contato com empregado |
| person\_contact | referência de contato com paciente |
| hospital\_employee | referência de empregado com hospital |
| hospital\_person | referência de paciente com hospital (id será utilizado para identificar paciente e hospital nos dados obtidos pelos sensores do exoesqueleto) |

Analisamos que pela proposta do projeto e o tipo de problema que estamos abordando, torna-se viável utilizar **NoSQL para os dados dos sensores do componente do exoesqueleto**. Um dos fatores mais relevantes que levamos em consideração foi o fato desse tipo de arquitetura possuir uma performance superior em relação aos banco de dados relacionais, principalmente na leitura dos dados, suportando baixa latência e alto desempenho, o que é vantajoso para dados coletados e analisados em tempo real.

Foi definido a utilização do Banco de Dados NoSQL **MongoDB** por ele ser OpenSource e bastante utilizado em projetos acadêmicos e profissionais.

[MongoDB](http://blog.4linux.com.br/curso-mongodb-presencial-sao-paulo/) é um banco de dados orientado a documentos. Estes documentos, conhecidos como collections, são semelhantes ao tipo JSON com esquemas dinâmicos, tornando a integração de dados em certos tipos de aplicativos mais fácil e rápido.

A figura a seguir mostra a defnição da **collection**, arquitetura base inicial proposta de acordo com a análise feita no documento de Product Backlog. A collection é referente aos dados que serão coletados dos sensores do componente do exoesqueleto do paciente.

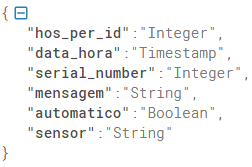


collection referente aos dados do componente do exoesqueleto.

Descrição das chaves do Object definidas do componente do exoesqueleto**:**

|  |  |  |
| --- | --- | --- |
| **Chave** | **Tipo** | **Descrição** |
| hos\_per\_id | Integer | Referência ao banco de dados relacional que identifica a relação com o hospital e paciente (tabela hospital\_person) |
| data\_hora | Timestamp | Data e hora da inserção dos dados |
| inclinacao | Double | Interruptor de inclinação do esqueleto para detecção de inclinação |
| velocidade | Double | Velocidade do exoesqueleto obtida por sensor de acelerômetro |
| temperatura | Integer | Temperatura obtida por sensor no exoesqueleto |
| movimentacao | Double | Movimentação do exoesqueleto obtida por sensor de giroscópio |
| saturacao\_oxigenio | Double | Saturação de oxigênio do sangue do paciente obtida por sensor óptico do exoesqueleto |
| queda | Boolean | Detecção de queda do exoesqueleto obtida através de um algoritmo |
| latitude | Double | Latitude da coordenada geográfica da localização do exoesqueleto |
| longitude | Double | Longitude da coordenada geográfica da localização do exoesqueleto |
| frequencia\_cardiaca | Double | Frequência cardíaca do paciente para detecção de atividades, esforço, emocional e/ou outros fatores, obtida por sensor no exoesqueleto |
| obstaculo | Boolean | Detecção de obstáculos próximo ao exoesqueleto obtida através de um sensor de IR |
| serial\_number | Integer | Número do série de identificação do dispositivo |

A figura a seguir mostra a collection **emergência** definida para as mensagens de emergências que serão acionadas pelo componente do exoesqueleto.



Descrição das chaves do Object definidas para acionamento de emergência**:**

|  |  |  |
| --- | --- | --- |
| **Chave** | **Tipo** | **Descrição** |
| hos\_per\_id | Integer | Referência ao banco de dados relacional que identifica a relação com o hospital e paciente (tabela hospital\_person) |
| data\_hora | Timestamp | Data e hora da inserção dos dados |
| serial\_number | Integer | Número do série de identificação do dispositivo |
| mensagem | String | Mensagem de emergência |
| automatico | Boolean | Mensagem automática (true) ou acionada pelo paciente (false) |
| sensor | String | Identificação do sensor que ocasionou a emergência (não obrigatório) |

1. **Conclusão**

Esta é definição inicial da modelagem do projeto, portanto ela pode sofrer alterações no decorrer do desenvolvimento e definições mais específicas e detalhadas que podem ocorrer durante as sprints.

As estruturas definidas na modelagem Relacional e noSQL já estão implementadas nos bancos de dados devidamente instalados e configurados em um servidor da AWS.