**LABDSOFT**

**ISEP – Mestrado em Engenharia Informática**

**Documento Arquitetura de Software**

**Banco de Gâmetas - GAM**

Trabalho prático nr.2 – iteração 1

**Histórico de Revisão**

|  |  |  |  |
| --- | --- | --- | --- |
| Versão | Autores | Descrição | Data |
| **1.0** | **Grupo 1**  Ana Barros  Daniel Bento  Manuel Correia  Maria Almeida  Tiago Gonçalves |  | **2017-10-22** |
| **1.2** | **Grupo 1**  Ana Barros  Daniel Bento  Manuel Correia  Maria Almeida  Tiago Gonçalves |  | **2017-11-26** |
| **1.3** | **Grupo 1**  Ana Barros  Daniel Bento  Manuel Correia  Maria Almeida  Tiago Gonçalves |  | **2017-12-10** |

2017

**Índice**

[1 Introdução 7](#_Toc500183016)

[1.1 Propósito 7](#_Toc500183017)

[1.2 Scope do produto 7](#_Toc500183018)

[1.3 Funcionalidades propostas 7](#_Toc500183019)

[2 Decisões tomadas 9](#_Toc500183020)

[3 Artefactos 10](#_Toc500183021)

[3.1 Diagrama de contexto 10](#_Toc500183022)

[3.2 Modelo de dados 11](#_Toc500183023)

[3.3 Diagrama de arquitetura 12](#_Toc500183024)

[4 Funcionalidades implementadas 13](#_Toc500183025)

[5.1 REQ 1 – Registo inicial do dador 13](#_Toc500183029)

[5.1.1 Análise de requisitos 13](#_Toc500183030)

[5.1.2 Diagrama de sequência 14](#_Toc500183031)

[5.1.3 Ilustração da funcionalidade 15](#_Toc500183032)

[5.2 REQ 2 – Validação dos dados do dador 16](#_Toc500183033)

[5.2.1 Análise de requisitos 16](#_Toc500183034)

[5.2.2 Diagrama de sequência 17](#_Toc500183035)

[5.2.1 Ilustração da funcionalidade 18](#_Toc500183036)

[5.3 REQ 3 – Gestão de perfis de utilizadores 19](#_Toc500183037)

[5.3.1 Análise de requisitos 19](#_Toc500183038)

[5.3.2 Diagrama de sequência 20](#_Toc500183039)

[5.3.3 Ilustração da funcionalidade 21](#_Toc500183040)

[5.4 REQ 4 – Configuração do template de perguntas médicas 22](#_Toc500183041)

[5.4.1 Análise de requisitos 22](#_Toc500183042)

[5.4.2 Diagrama de sequência 23](#_Toc500183043)

[5.4.3 Ilustração da funcionalidade 24](#_Toc500183044)

[5.5 REQ 5 – Consultas médicas via inquérito template 25](#_Toc500183045)

[5.5.1 Análise de requisitos 25](#_Toc500183046)

[5.5.2 Diagrama de sequência 26](#_Toc500183047)

[5.5.3 Ilustração da funcionalidade 27](#_Toc500183048)

[5.6 REQ 6 – Registo de amostras 28](#_Toc500183049)

[5.6.1 Análise de requisitos 28](#_Toc500183050)

[5.6.2 Diagrama de sequência 29](#_Toc500183051)

[5.6.3 Ilustração da funcionalidade 30](#_Toc500183052)

[5.7 REQ 7 – Pedido das análises de sangue iniciais 32](#_Toc500183053)

[5.7.1 Análise de requisitos 32](#_Toc500183054)

[5.7.2 Diagrama de sequência 33](#_Toc500183055)

[5.7.3 Ilustração da funcionalidade 34](#_Toc500183056)

[5.8 REQ 8 – Ciclo da dádiva (abertura e consulta de informação) 35](#_Toc500183057)

[5.8.1 Análise de requisitos 35](#_Toc500183058)

[5.8.2 Diagrama de sequência 36](#_Toc500183059)

[5.8.3 Ilustração da funcionalidade 37](#_Toc500183060)

[5.9 REQ 9 – Marcação de consultas 38](#_Toc500183061)

[5.9.1 Análise de requisitos 38](#_Toc500183062)

[5.9.2 Diagrama de sequência 39](#_Toc500183063)

[5.9.3 Ilustração da funcionalidade 40](#_Toc500183064)

[5.10 REQ 10 – Aprovação/Rejeição do ciclo da dádiva 41](#_Toc500183065)

[5.10.1 Análise de requisitos 41](#_Toc500183066)

[5.10.2 Diagrama de sequência 42](#_Toc500183067)

[5.10.3 Ilustração da funcionalidade 43](#_Toc500183068)

[5.11 REQ 11 – Lista de trabalhos do laboratório 44](#_Toc500183069)

[5.11.1 Análise de requisitos 44](#_Toc500183070)

[5.11.2 Diagrama de sequência 45](#_Toc500183071)

[5.11.3 Ilustração da funcionalidade 46](#_Toc500183072)

[5.12 REQ 12 – Registo de espermograma 47](#_Toc500183073)

[5.12.1 Análise de requisitos 47](#_Toc500183074)

[5.12.2 Diagrama de sequência 48](#_Toc500183075)

[5.12.3 Ilustração da funcionalidade 49](#_Toc500183076)

[5.13 REQ 13 – Validação do espermograma 51](#_Toc500183077)

[5.13.1 Análise de requisitos 51](#_Toc500183078)

[5.13.2 Diagrama de sequência 52](#_Toc500183079)

[5.13.3 Ilustração da funcionalidade 53](#_Toc500183080)

[5.14 REQ 14 – Criopreservação da amostra 54](#_Toc500183081)

[5.14.1 Análise de requisitos 54](#_Toc500183082)

[5.14.2 Diagrama de sequência 55](#_Toc500183083)

[5.14.3 Ilustração da funcionalidade 56](#_Toc500183084)

[5.15 REQ 15 – Visualização da alocação das amostras no Banco de criopreservação 58](#_Toc500183085)

[5.15.1 Análise de requisitos 58](#_Toc500183086)

[5.15.2 Diagrama de sequência 59](#_Toc500183087)

[5.15.3 Ilustração da funcionalidade 60](#_Toc500183088)

[5.16 REQ 17 – Registo de pedidos de gâmetas 61](#_Toc500183089)

[5.16.1 Análise de requisitos 61](#_Toc500183090)

[5.16.2 Diagrama de sequência 62](#_Toc500183091)

[5.16.3 Ilustração da funcionalidade 63](#_Toc500183092)

[5.17 REQ 18 – Registo de materiais usados (sem integração) 65](#_Toc500183093)

[5.17.1 Análise de requisitos 65](#_Toc500183094)

[5.17.2 Diagrama de sequência 66](#_Toc500183095)

[5.17.3 Ilustração da funcionalidade 67](#_Toc500183096)

[5 Funcionalidades não implementadas 67](#_Toc500183097)

[6 Pipeline 68](#_Toc500183098)

[7 Distribuição de tarefas 70](#_Toc500183099)

[8 Tempos 71](#_Toc500183100)

[9 Medidas de qualidade de software 72](#_Toc500183101)

**Índice de Ilustrações**

[Figura 1 - Diagrama de Contexto 7](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477199)

[Figura 2 - Modelo de dados 8](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477200)

[Figura 3 - Diagrama de arquitetura 8](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477201)

[Figura 4 - Excerto do ficheiro de resultados (testes) 9](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477202)

[Figura 5 – Artefactos criados após stage Archive 10](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477203)

[Figura 6 - Stage View Pipeline 10](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477204)

[Figura 7 - REQ\_1: Página Inicial 11](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477205)

[Figura 8 - REQ\_1: View de Registo do Dador 11](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477206)

[Figura 9 - REQ\_2: View Lista de Dadores 12](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477207)

[Figura 10 - REQ\_2: View da Validação de Dador 12](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477208)

[Figura 11 - REQ\_3: View da Lista de Perfis 13](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477209)

[Figura 12 - REQ\_3: View da Criação de um novo perfil 13](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477210)

[Figura 13: REQ\_3: View de edição de um perfil 14](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477211)

[Figura 14 - REQ\_4: View de edição de um questionário 14](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477212)

[Figura 15 - REQ\_5: View da lista de possíveis dadores 15](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477213)

[Figura 16 - REQ5: View dos detalhes do dador 15](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477214)

[Figura 17: REQ\_5: View do preenchimento do formulário 16](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477215)

[Figura 18 - REQ\_6: View da lista de amostras 16](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477216)

[Figura 19 - REQ\_6: View de criação de uma amostra 17](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477217)

[Figura 20 - REQ\_6 - View de detalhes de uma amostra 17](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477218)

[Figura 21 - REQ\_6: View de edição de uma amostra 18](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477219)

[Figura 22 - REQ\_7: View da listagem de dadores com amostras pendentes de análise 19](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477220)

[Figura 23 - REQ\_7: View dos resultados das análises gerados pela API iAnalysis 19](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477221)

[Figura 24 - REQ\_8: View dos processos pendentes e abertos 20](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477222)

[Figura 25 - REQ\_8: View de abertura do processo 20](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477223)

[Figura 26 - REQ\_9: View da lista de consultas 21](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477224)

[Figura 27 - REQ\_9: View da marcação de uma consulta 21](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477225)

[Figura 28 - REQ\_10: View da lista de processos pendentes de aprovação 22](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477226)

[Figura 29 - REQ\_10: View da aprovação/rejeição de um processo 22](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477227)

[Figura 30 - REQ\_11: View da lista de amostras por analisar 23](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477228)

[Figura 31 -REQ\_11: View da lista de amostras analisadas 23](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477229)

[Figura 32 - REQ\_12 – View da lista de espermogramas 23](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477230)

[Figura 33 - REQ\_12: View de criação de espermograma 24](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477231)

[Figura 34 - REQ\_12: View dos detalhes do espermograma 24](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477232)

[Figura 35 - REQ\_13: View da lista de espermogramas 25](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477233)

[Figura 36 - REQ\_13: View de aprovação/rejeição de um espermograma 25](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477234)

[Figura 37 - REQ\_14: View da lista de amostras analisadas 26](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477235)

[Figura 38 - REQ\_14: View dos detalhes de uma amostra 26](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477236)

[Figura 39 - REQ\_14: View da criopreservação de uma amostra 27](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477237)

[Figura 40 - REQ\_15: View da lista de alocação das amostras 28](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477238)

[Figura 41 - REQ\_17: View da lista de pedidos de gâmetas 28](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477239)

[Figura 42 - REQ\_17: View de detalhes de um pedido de gâmetas 28](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477240)

[Figura 43 - REQ\_17: View de edição de um pedido de gâmetas 29](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477241)

[Figura 44 - REQ\_17: View da criação de um pedido de gâmetas 29](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477242)

[Figura 45 - REQ\_18: View da lista de materiais usados 30](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477243)

[Figura 46 - REQ\_18: View do registo de material utilizado 30](file:///C:\Users\mcorreia\Documents\GitKrakenProjects\labdsoft-2017-g1\Documentacao\Entrega%202\relatorio_tp2_g1.docx#_Toc499477244)

**Índice de Tabelas**

Tabela 1 - Funcionalidades propostas 5

Tabela 2 – Distribuição tarefas 31

Tabela 3 - Duração de cada requisito 32

# Introdução

# Propósito

Este documento foi desenvolvido no contexto da segunda iteração do trabalho prático da disciplina de Laboratório Desenvolvimento Software (LABDSOFT), integrada no Mestrado de Engenharia Informática (MEI) do Instituto Superior de Engenharia do Porto (ISEP). Tem como propósito descrever os processos inerentes à implementação de alguns requisitos definidos para a aplicação de software - iGAM. No que diz respeito às funcionalidades a implementar, iremos garantir que o processo consegue fluir com todos os intervenientes, desde a sua inscrição inicial até à sua aprovação ou rejeição.

Ao longo do documento serão abordados tópicos relacionados com algumas das decisões tomadas pelo grupo, artefactos relevantes a nível arquitetural e tecnológico incluindo a especificação do pipeline criado. Para além disso, serão apresentadas as funcionalidades implementadas bem como a distribuição de tarefas entre o grupo de trabalho.

## Scope do produto

A GAM pretende uma solução de software que consiga monitorizar os dois grandes processos que neste momento existem apenas em formato papel. A gestão do conjunto de etapas que constituem os processos é outro grande foco.

O primeiro processo denomina-se por “**Ciclo de Dádiva**”, é muito rigoroso e constitui várias etapas. Existem dadores que realizam amostras no banco. Após verificação e validação de critérios bem definidos, o dador é considerado um dador “válido”. As amostras de espermatozoides ficam crio preservadas para mais tarde puderem ser usadas por casais inférteis.

O segundo processo denomina-se por “**Resposta a Pedidos de Gâmetas**”. Existem clínicas/hospitais que nas suas consultas de procriação recomendam técnicas de fertilidade como por exemplo, a fertilização in vitro com dadores anónimos. Estas clínicas solicitam amostras para puderem ser usadas nestes casais. O pedido é feito ao banco (formato papel), a respetiva clínica envia o conjunto de caraterísticas do casal e o banco é responsável por selecionar o dador que melhor obedece aos requisitos.

Em suma, o objetivo principal desta solução é automatizar estes dois processos, mas também ser capaz de apresentar ao cliente dados estatísticos relativos a ambos os processos.

## Funcionalidades propostas

Para esta iteração, temos um total de dezoito requisitos que pretendemos implementar no sistema de software iGAM e dez atores envolvidos (dador, informático, enfermeira coordenadora, médico, enfermeiro, embriologista, diretora laboratório, embriologista, laboratório sangue (externo), Clínica PMA) – Tabela 1.

Tabela 1 - Funcionalidades propostas

|  |  |  |
| --- | --- | --- |
| Id | Identificação | Ator |
| REQ\_1 | Registo inicial do dador (sem fotografia) | Dador |
| REQ\_2 | Validação dos dados do dador | Enfermeira coordenadora |
| REQ\_3 | Gestão de perfis de utilizadores | Informático |
| REQ\_4 | Configuração do template de perguntas médicas | Médico |
| REQ\_5 | Consultas médicas via inquérito template | Médico |
| REQ\_6 | Registo de amostras | Enfermeira |
| REQ\_7 | Pedido das análises de sangue iniciais | Médico |
| REQ\_8 | Ciclo da dádiva (abertura e consulta de informação) | Enfermeira coordenadora |
| REQ\_9 | Marcação de consultas | Enfermeira |
| REQ\_10 | Aprovação / Rejeição do ciclo da dádiva | Médico |
| REQ\_11 | Lista de trabalho do laboratório | Embriologista |
| REQ\_12 | Registo de espermograma | Embriologista |
| REQ\_13 | Validação do espermograma | Diretora laboratório |
| REQ\_14 | Criopreservação da amostra | Embriologista |
| REQ\_15 | Visualização da alocação das amostras no Banco de criopreservação | Embriologista |
| REQ\_16 | Integração com o laboratório de análises de sangue | Lab. Sangue (externo) |
| REQ\_17 | Registo de pedidos de gâmetas | Clínica PMA |
| REQ\_18 | Registo de materiais usados (sem integração) | Embriologista |

# Decisões tomadas

A nível de organização da equipa de desenvolvimento, optamos por utilizar a aplicação Slack como veículo de comunicação entre todos os elementos. O uso desta plataforma de comunicação permitiu-nos fazer chamadas de grupo, debater problemas, partilhar excertos de código e manter-nos informados sobre o estado de cada *issue*, uma vez que foi adicionado um *plugin* de integração com o Bitbucket. Importa também referir que no Bitbucket, foi criado um *board* associado ao Trello, onde foram sendo adicionadas as tarefas a desenvolver ao longo desta iteração, organizadas por etapas. Deste modo, todos os elementos da equipa de desenvolvimento sabiam o estado de cada *issue*.

Quanto ao desenvolvimento do produto, optamos por utilizar a linguagem de programação ASP.NET Core, uma vez que todos os elementos do grupo estavam familiarizados com a mesma. De realçar que a decisão recaiu sobre ASP.NET Core e não sobre ASP.NET pelo fato da primeira ser multiplataforma e modular, o que significa que as aplicações são mais leves e com melhor desempenho. Já a nível arquitetural, decidimos seguir uma arquitetura MVC (Model View Controller) porque, mais uma vez, os elementos do grupo já estavam familiarizados com a mesma e consideramos que nos dá garantias de manter o código organizado, estruturado e com uma adequada separação de responsabilidades. Uma das vantagens deste tipo de arquitetura é o fato de permitir a diversos elementos trabalhar sobre o mesmo repositório, sem que se verifiquem conflitos na junção de vários pedaços de código.

Por sua vez, quanto à persistência de dados, numa primeira instância recorremos a uma base de dados local providenciada pelo ambiente de desenvolvimento usado (Visual Studio 2017), onde implementamos um *script* para povoar a base de dados, de cada máquina onde corresse a solução, com alguns dados. Porém, esta solução não nos permitia apresentar uma aplicação escalável pelo que tomamos a decisão de migrar os dados para a *cloud*. Deste modo, temos uma única base de dados, partilhada e acessível por todos os elementos da equipa de desenvolvimento.

Por fim, como ferramenta de automação de todo o processo foi utilizada a plataforma Jenkins. O processo foi então dividido em cinco etapas, desde a conexão ao projeto alojado no Bitbucket, passando pela *build* do mesmo e a execução de alguns testes unitários, e terminando na publicação dos resultados dos testes e na geração de um zip com todos os dados.

# Artefactos

## Diagrama de contexto

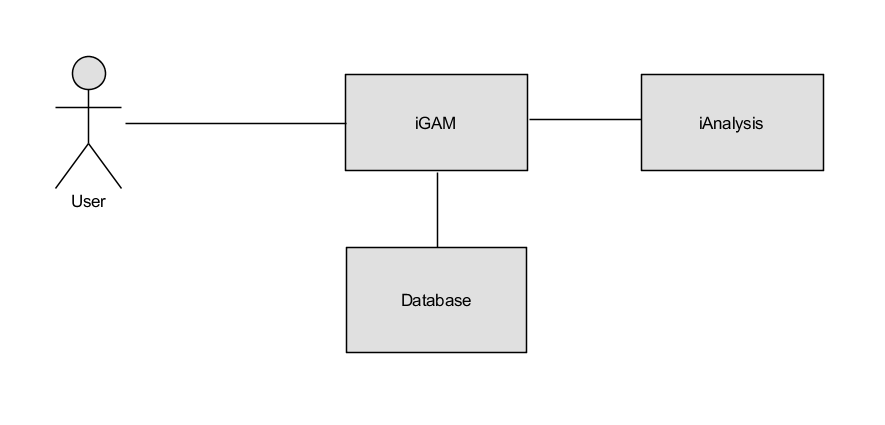


Figura 1 - Diagrama de Contexto

Na Figura 1 é apresentado o Diagrama de Contexto da aplicação, onde podemos ver que os utilizadores (Dador, Médico, Enfermeiro, Embriologista, Diretor Geral, Clinicas PMA, Informático e Assistente Social) vão interagir com a aplicação *web* iGAM. Esta aplicação pode ser executada a partir de qualquer navegador de Internet e, com exceção dos Dadores, para usufruir das suas funcionalidades é necessário ser um utilizador registado no sistema. Por sua vez, a plataforma iGAM vai interagir com uma base de dados alojada na *cloud*, para garantir a persistência de dados. Por fim, a plataforma iGAM recorre a uma *web* API, denominada iAnalysis, para onde são enviadas as amostras de sangue com o intuito de serem analisadas. Após a análise estar concluída, os resultados são devolvidos para a iGAM, que é responsável por os armazenar na base de dados referida anteriormente.

<Melhorar?>

## Modelo de dados

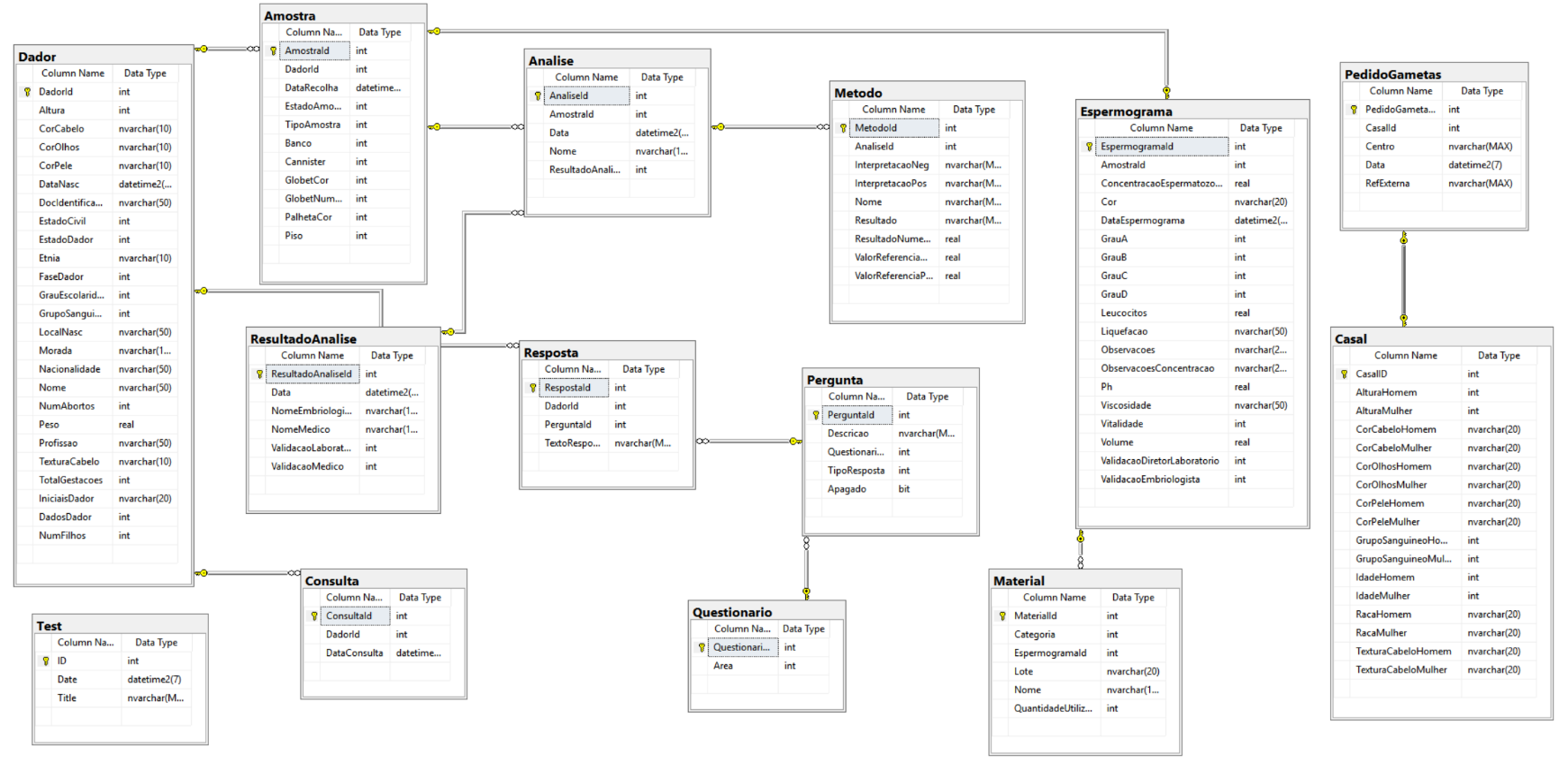


Figura 2 - Modelo de dados

<Inserir Descrição aqui>

## Diagrama de arquitetura

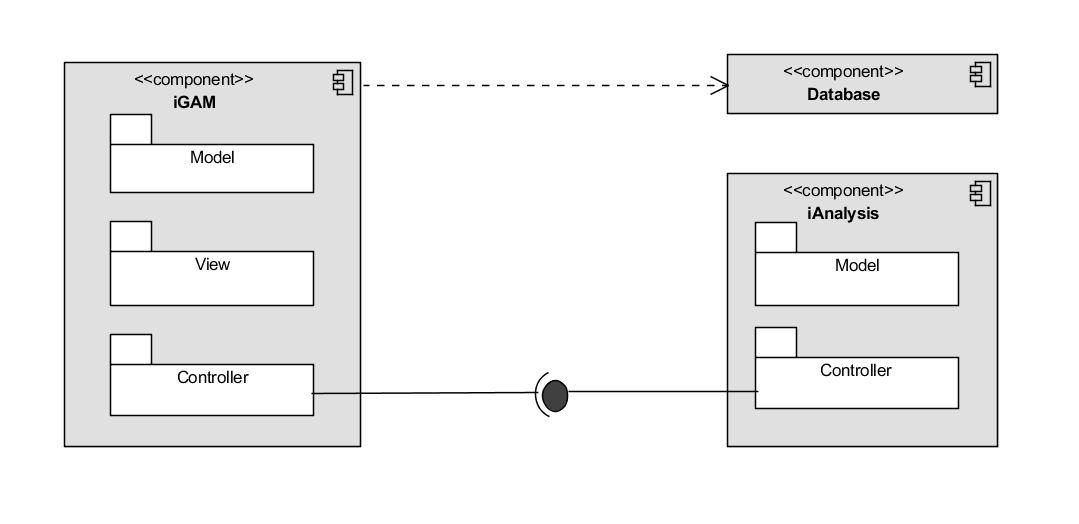


Figura 3 - Diagrama de arquitetura

<Diagrama de alto nível. Escolher uma das opções:

1 – Melhorar existente

2 – Deixar existente e adicionar um novo c/ mais detalhe>

# Funcionalidades implementadas



## REQ 1 – Registo inicial do dador

### Análise de requisitos

|  |  |
| --- | --- |
| Identificador | REQ 1 |
| Nome | Registo inicial do dador |
| Descrição | O(a) Dador(a), enquanto utilizador não registado do sistema, pode submeter a sua candidatura através da página inicial da iGAM. Esta candidatura ficará pendente de aprovação. |

|  |  |
| --- | --- |
| Fluxo de Eventos - Humano | Fluxo de Eventos - Software |
| 1 – O(a) Dador(a) acede à secção de registo de um novo dador | 1.1 – A view “XPTO” chama/faz …  1.2 – O Controller “ZXY” chama/faz … |
| 2 – O(a) Dador(a) insere os dados solicitados | Sem ação do sistema |
| 3 – O(a) Dador(a) submete o registo | 3.1 –  3.2 - |

|  |  |  |
| --- | --- | --- |
| Pré-condições | Pós-condições | Validações |
| Sem nada a registar | Os dados inseridos são validados pelo sistema | Os campos de texto não podem estar vazios |
|  | Os dados inseridos no sistema são armazenados numa base de dados | A idade deve ser um valor compreendido entre os 16 e os 60 anos |
|  | É gerado um identificador único para o(a) dador(a) | O NIF deve ser constituído por 9 dígitos |
|  |  | O contacto telefónico deve ser constituído por 9 dígitos |

### Diagrama de sequência

<Diagrama de sequência básico, que junta o Actor e a interação entre as diferentes camadas. Nos casos de uso mais completos/complexos, complementar o diagrama com uma breve descrição>

### Ilustração da funcionalidade



Figura 7 - REQ\_1: Página Inicial

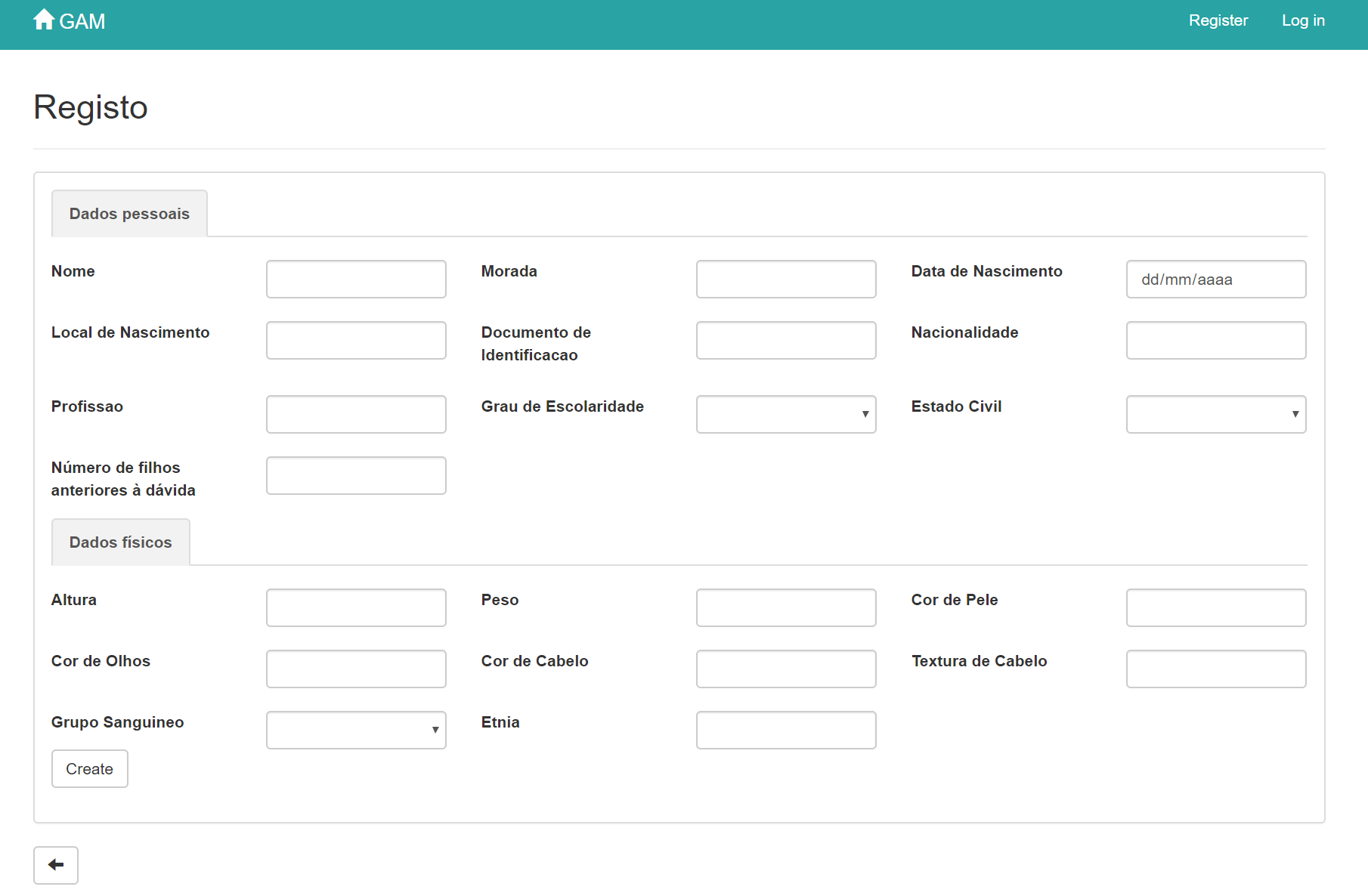


Figura 8 - REQ\_1: View de Registo do Dador

## REQ 2 – Validação dos dados do dador

### Análise de requisitos

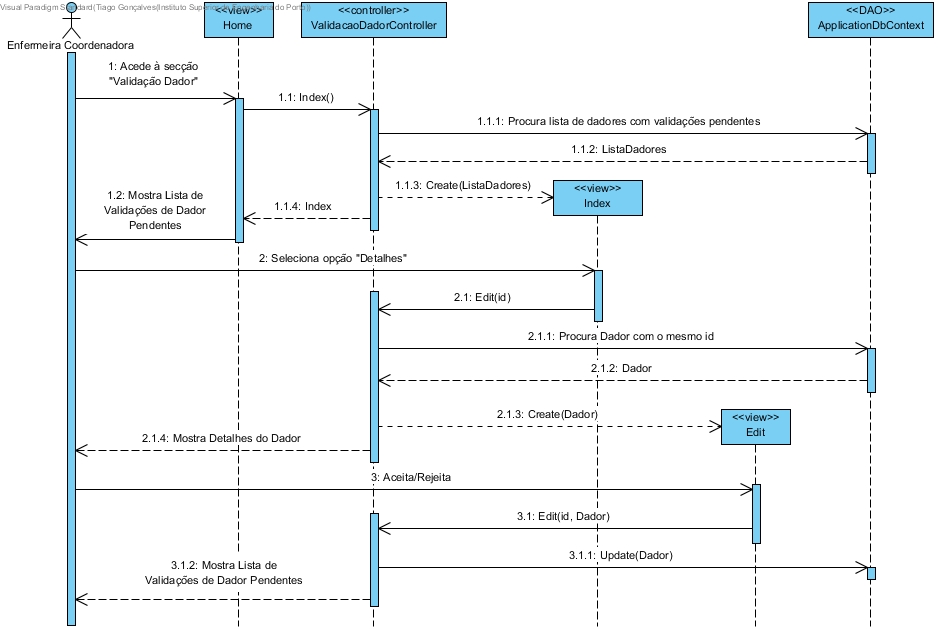
|  |  |
| --- | --- |
| Identificador |  |
| Nome | Validação dos dados do dador |
| Descrição | A Enfermeira Coordenadora, enquanto utilizador registado do sistema, pode aceder à secção de “Validação Dador” na sua página da iGAM para validação de dados relativos a pedidos de registo de dadores. Esta validação poderá ser aceite ou rejeitada. |

|  |  |
| --- | --- |
| Fluxo de Eventos - Humano | Fluxo de Eventos - Software |
| 1 – A Enfermeira Coordenadora acede à secção “Validação Dador” | 1.1 – A view “Home” executa o método Index do Controller “ValidacaoDadorController”  1.2 – O Controller “ValidacaoDadorController” carrega a view de “ValidacaoDador” associada, com dadores que tenham dados pendentes para validação |
| 2 – A Enfermeira Coordenadora seleciona a opção “Detalhes” de um registo pendente | 2.1 – A view “ValidacaoDador” executa o método Edit do Controller “ValidacaoDadorController”  2.2 – O Controller “ValidacaoDadorController” carrega a view de “ValidacaoDador” associada, com os detalhes do dador selecionado |
| 3 – A Enfermeira Coordenadora aceita ou rejeita o registo | 3.1 – A view “ValidacaoDador” executa o método Edit do Controller “ValidacaoDadorController”  3.2 – O Controller “ValidacaoDadorController” atualiza o estado da validação dos dados do dador para aceite ou rejeitado conforme a escolha |

|  |  |  |
| --- | --- | --- |
| Pré-condições | Pós-condições | Validações |
| A Enfermeira Coordenadora deve estar autenticada no sistema | O estado relativo à validação dos dados do dador é atualizado na base de dados | Sem nada a registar |

### Diagrama de sequência

<Diagrama de sequência básico, que junta o Actor e a interação entre as diferentes camadas. Nos casos de uso mais completos/complexos, complementar o diagrama com uma breve descrição>



### Ilustração da funcionalidade



Figura 9 - REQ\_2: View Lista de Dadores

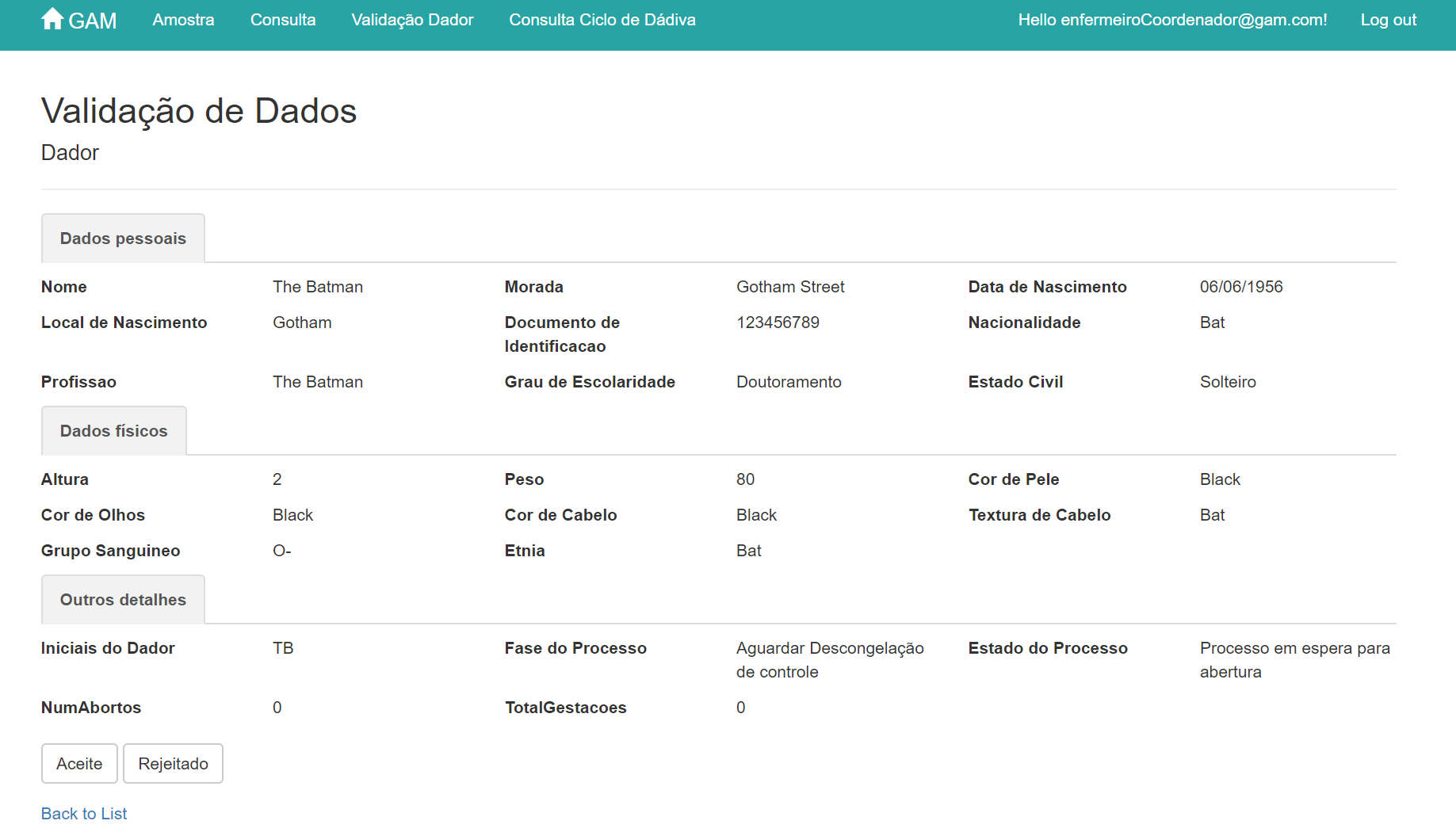


Figura 10 - REQ\_2: View da Validação de Dador

## REQ 3 – Gestão de perfis de utilizadores

### Análise de requisitos

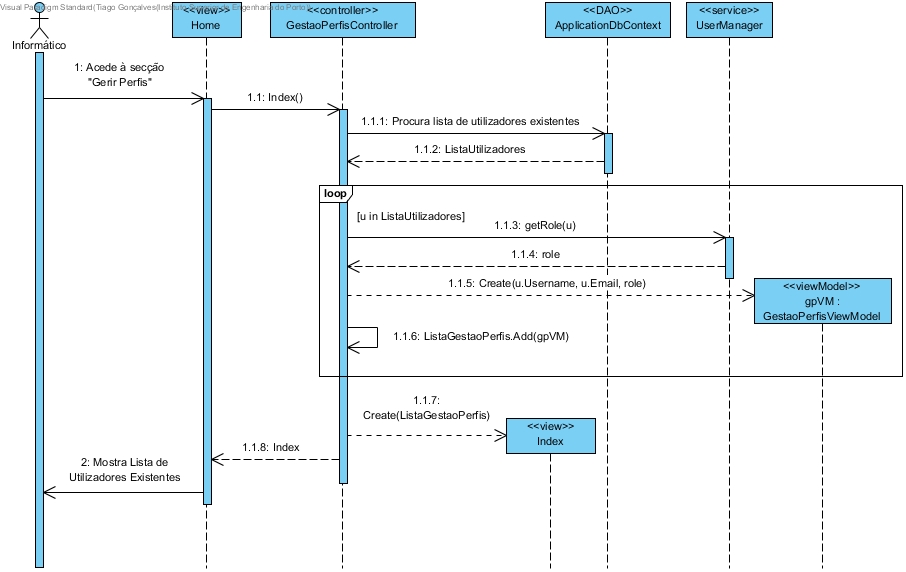
|  |  |
| --- | --- |
| Identificador |  |
| Nome | Gestão de perfis de utilizadores |
| Descrição | O Informático, enquanto utilizador registado no sistema, pode aceder à secção “Gerir Perfis” na sua página da iGAM para criar, editar, ou eliminar utilizadores que podem aceder ao sistema, atribuindo um perfil respetivo ao cargo do utilizador. |

|  |  |
| --- | --- |
| Fluxo de Eventos - Humano | Fluxo de Eventos - Software |
| 1 – O Informático acede à secção “Gerir Perfis” | 1.1 – A view “Home” executa o método Index do Controller “GestaoPerfisController”  1.2 – O Controller “GestaoPerfisController” carrega a view de “GestaoPerfis” associada, com os utilizadores existentes no sistema |
| 2 – O Informático pode criar um novo utilizador, ou pode editar ou remover um utilizador existente | 2.1 – A view “GestaoPerfis” executa o método Create, Edit, ou Delete do Controller “GestaoPerfisController” consoante a respetiva opção  2.2 – O Controller “GestaoPerfisController” carrega a view de “GestaoPerfis” associada, com um formulário para criação ou edição do utilizador, ou um pedido de confirmação para eliminação da conta, consoante a respetiva opção |
| 3 – O Informático insere os dados para criação e edição, ou consulta a informação para remoção | Sem ação do sistema |
| 4 – O Informático confirma a ação | 4.1 – A view “GestaoPerfis” executa o método Create, Edit, ou DeleteConfirmed do Controller “GestaoPerfisController” consoante a respetiva ação  4.2 – O Controller “GestaoPerfisController” cria, atualiza, ou elimina os dados de um utilizador do sistema consoante a ação |

|  |  |  |
| --- | --- | --- |
| Pré-condições | Pós-condições | Validações |
| O Informático deve estar autenticado no sistema | Os dados inseridos são validados pelo sistema (criação e edição) | Os campos de texto não podem estar vazios |
|  | Os dados inseridos no sistema são armazenados ou atualizados numa base de dados (criação e edição) | A password e a confirmação da password devem ser idênticas |
|  | Os dados são eliminados da base de dados (remoção) | A password deve conter um mínimo de 6 caracteres, um máximo 100 caracteres, letras maiúsculas, letras minúsculas, números e caracteres especiais |
|  | É gerado um identificador único para o utilizador (criação) |  |

### Diagrama de sequência

<Diagrama de sequência básico, que junta o Actor e a interação entre as diferentes camadas. Nos casos de uso mais completos/complexos, complementar o diagrama com uma breve descrição>



### Ilustração da funcionalidade

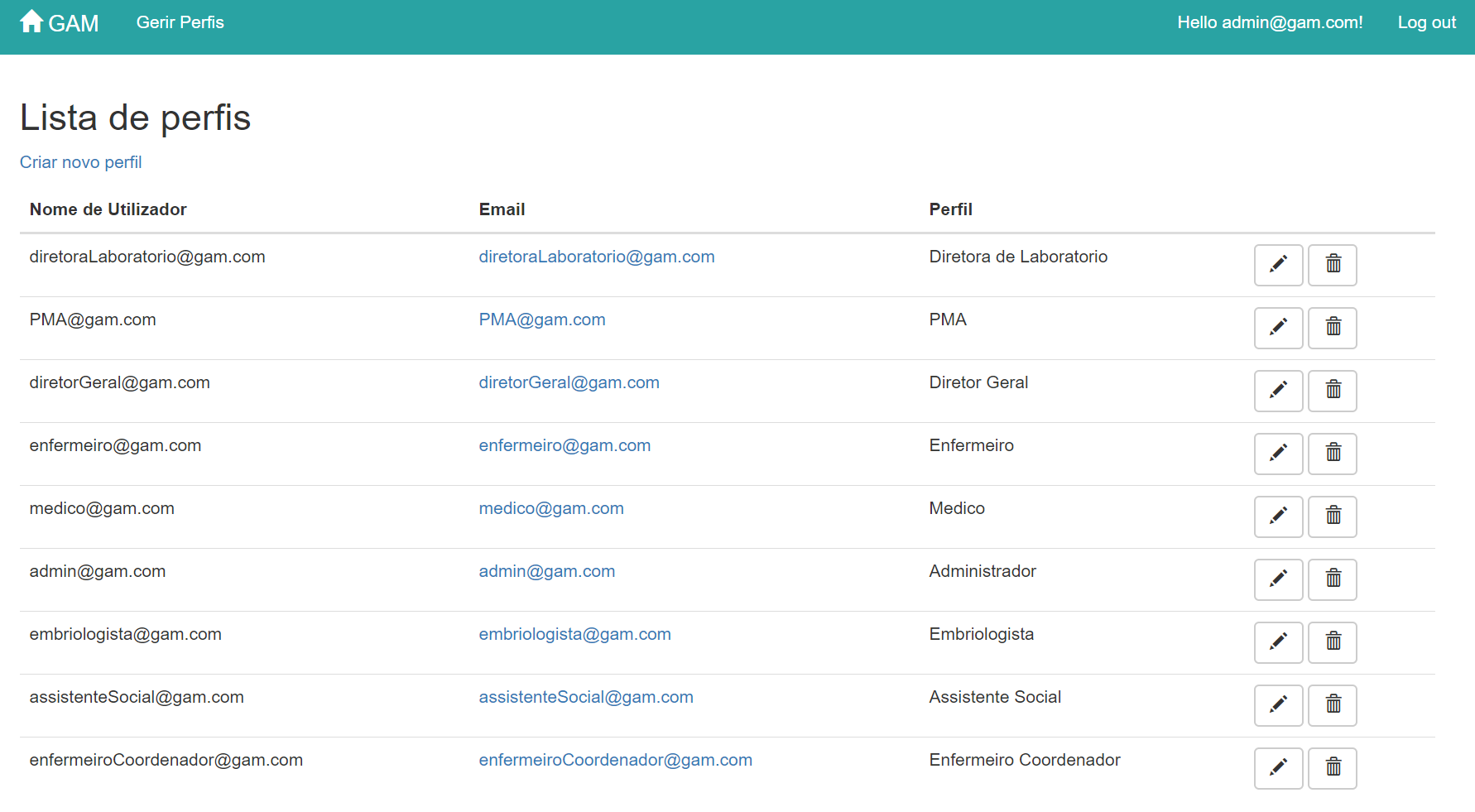


Figura 11 - REQ\_3: View da Lista de Perfis

Figura 12 - REQ\_3: View da Criação de um novo perfil

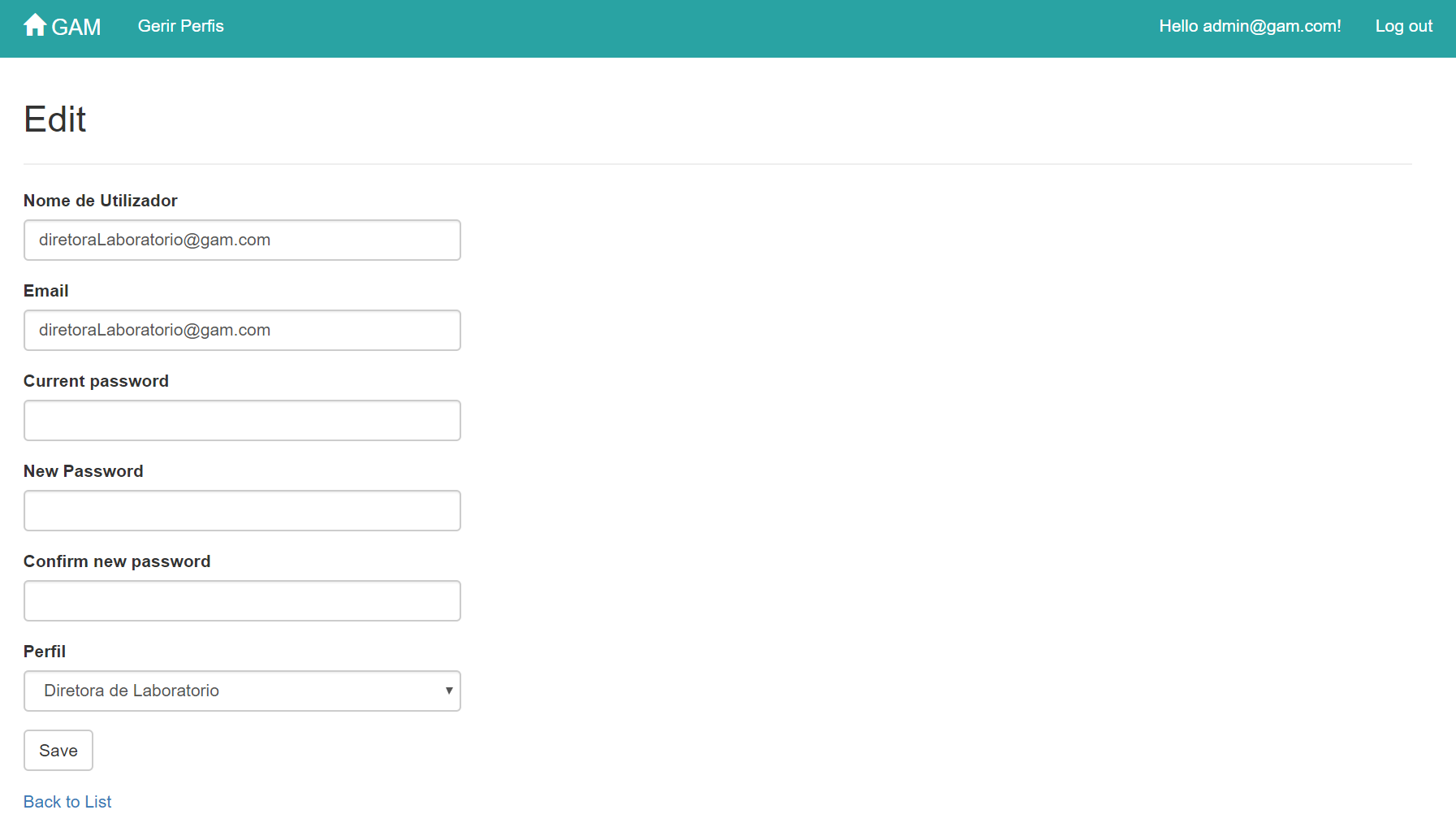


Figura 13: REQ\_3: View de edição de um perfil

## REQ 4 – Configuração do template de perguntas médicas

### Análise de requisitos

|  |  |
| --- | --- |
| Identificador |  |
| Nome |  |
| Descrição |  |

|  |  |
| --- | --- |
| Fluxo de Eventos - Humano | Fluxo de Eventos - Software |
| 1 – | 1.1 – A view “XPTO” chama/faz …  1.2 – O Controller “ZXY” chama/faz … |
| 2 – | Sem ação do sistema |
| 3 – | 3.1 –  3.2 - |

|  |  |  |
| --- | --- | --- |
| Pré-condições | Pós-condições | Validações |
| Sem nada a registar |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

### Diagrama de sequência

<Diagrama de sequência básico, que junta o Actor e a interação entre as diferentes camadas. Nos casos de uso mais completos/complexos, complementar o diagrama com uma breve descrição>

### Ilustração da funcionalidade

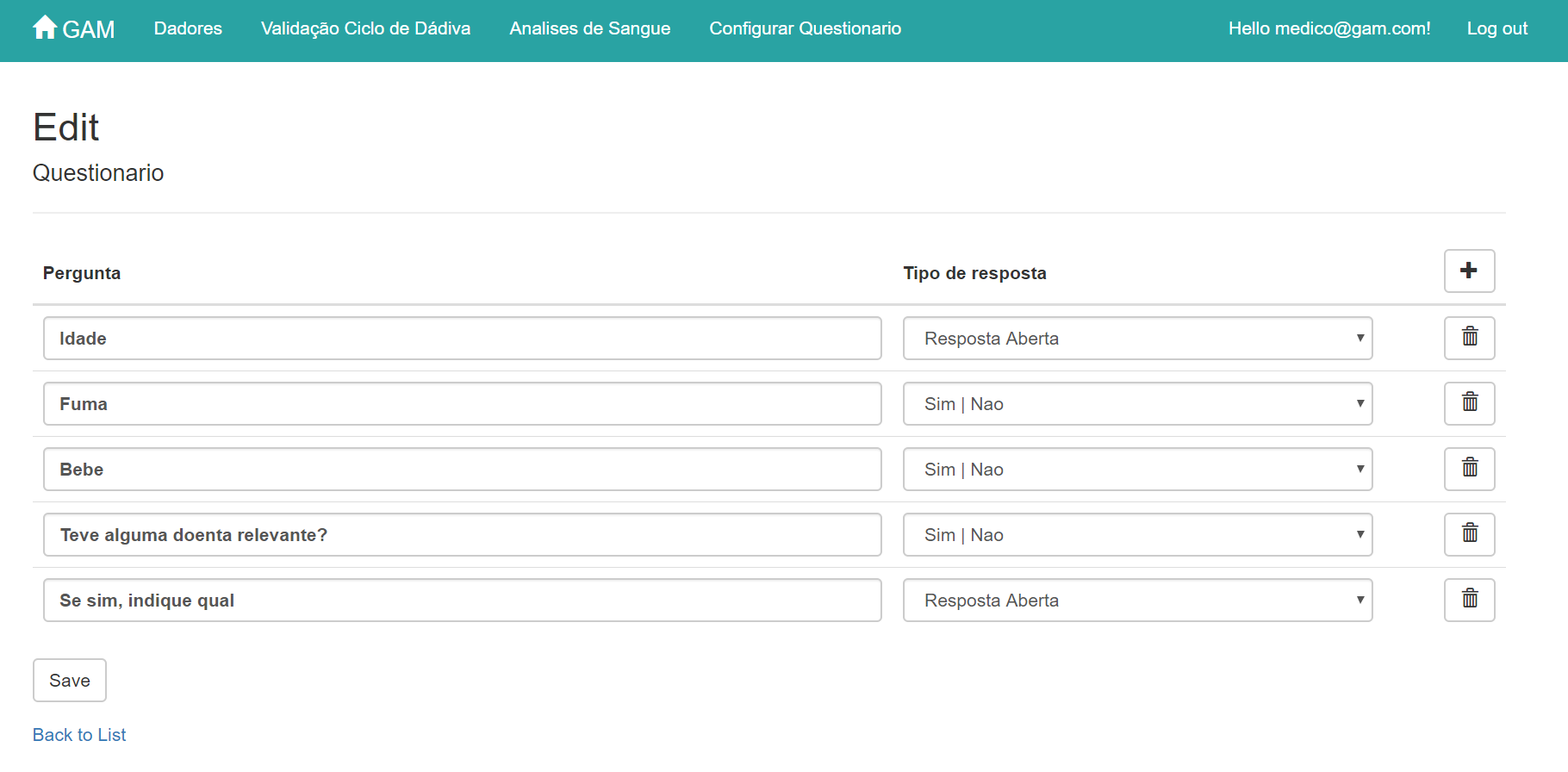


Figura 14 - REQ\_4: View de edição de um questionário

## REQ 5 – Consultas médicas via inquérito template

### Análise de requisitos

|  |  |
| --- | --- |
| Identificador |  |
| Nome |  |
| Descrição |  |

|  |  |
| --- | --- |
| Fluxo de Eventos - Humano | Fluxo de Eventos - Software |
| 1 – | 1.1 – A view “XPTO” chama/faz …  1.2 – O Controller “ZXY” chama/faz … |
| 2 – | Sem ação do sistema |
| 3 – | 3.1 –  3.2 - |

|  |  |  |
| --- | --- | --- |
| Pré-condições | Pós-condições | Validações |
| Sem nada a registar |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

### Diagrama de sequência

<Diagrama de sequência básico, que junta o Actor e a interação entre as diferentes camadas. Nos casos de uso mais completos/complexos, complementar o diagrama com uma breve descrição>

### Ilustração da funcionalidade



Figura 15 - REQ\_5: View da lista de possíveis dadores

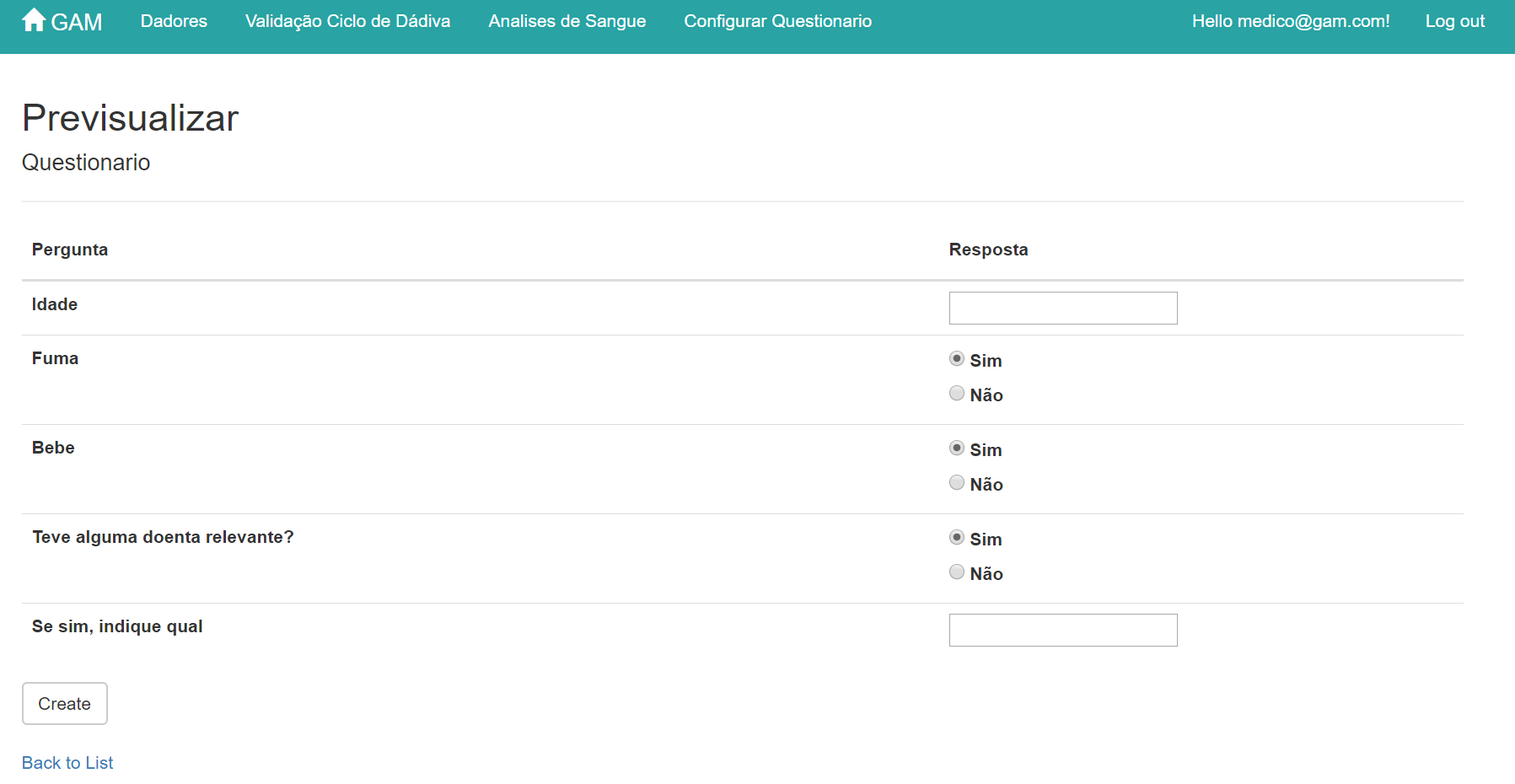


Figura 17: REQ\_5: View do preenchimento do formulário



Figura 16 - REQ5: View dos detalhes do dador

## REQ 6 – Registo de amostras

### Análise de requisitos

|  |  |
| --- | --- |
| Identificador |  |
| Nome |  |
| Descrição |  |

|  |  |
| --- | --- |
| Fluxo de Eventos - Humano | Fluxo de Eventos - Software |
| 1 – | 1.1 – A view “XPTO” chama/faz …  1.2 – O Controller “ZXY” chama/faz … |
| 2 – | Sem ação do sistema |
| 3 – | 3.1 –  3.2 - |

|  |  |  |
| --- | --- | --- |
| Pré-condições | Pós-condições | Validações |
| Sem nada a registar |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

### Diagrama de sequência

<Diagrama de sequência básico, que junta o Actor e a interação entre as diferentes camadas. Nos casos de uso mais completos/complexos, complementar o diagrama com uma breve descrição>

### Ilustração da funcionalidade



Figura 18 - REQ\_6: View da lista de amostras

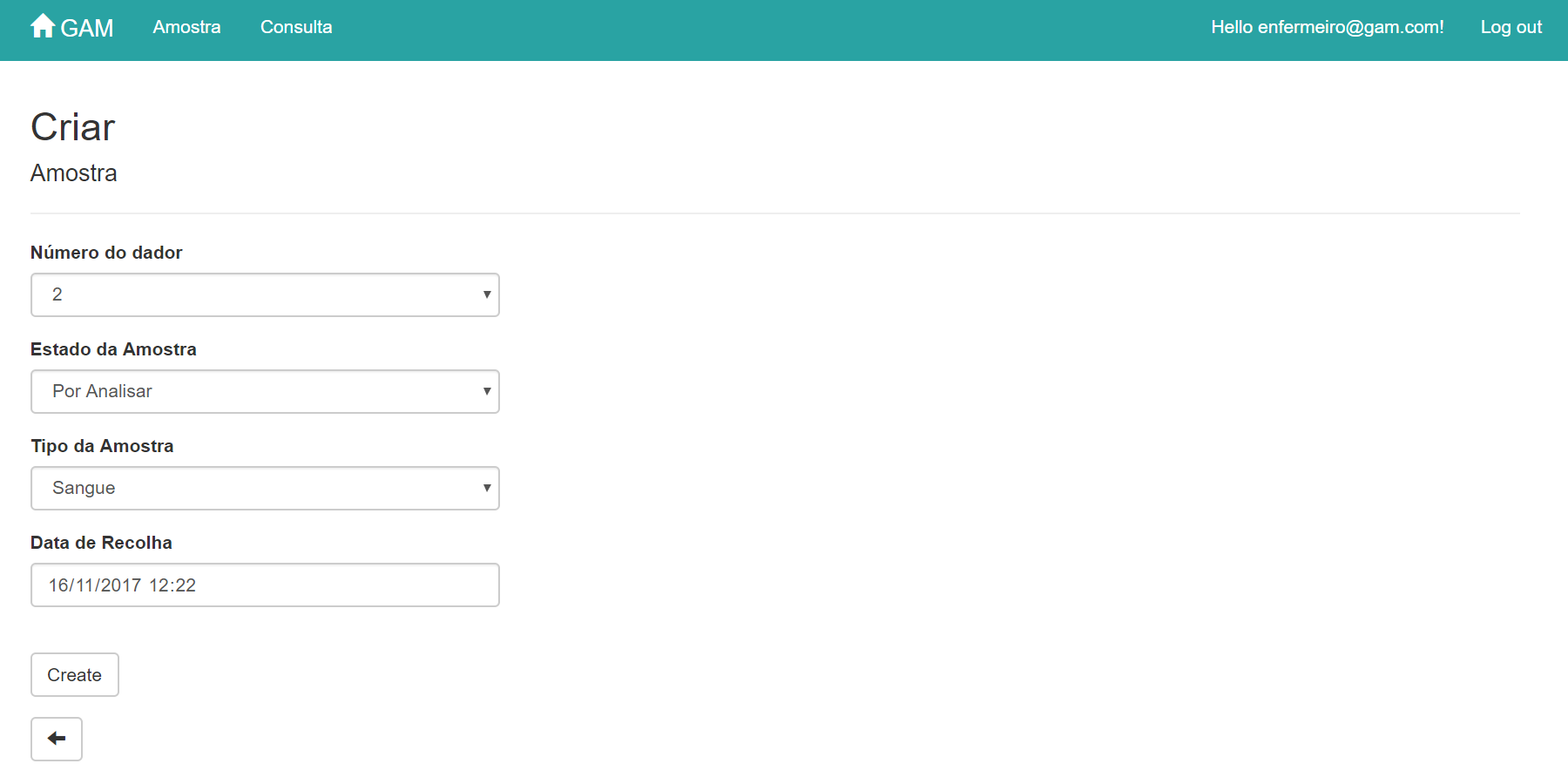


Figura 19 - REQ\_6: View de criação de uma amostra



Figura 21 - REQ\_6: View de edição de uma amostra

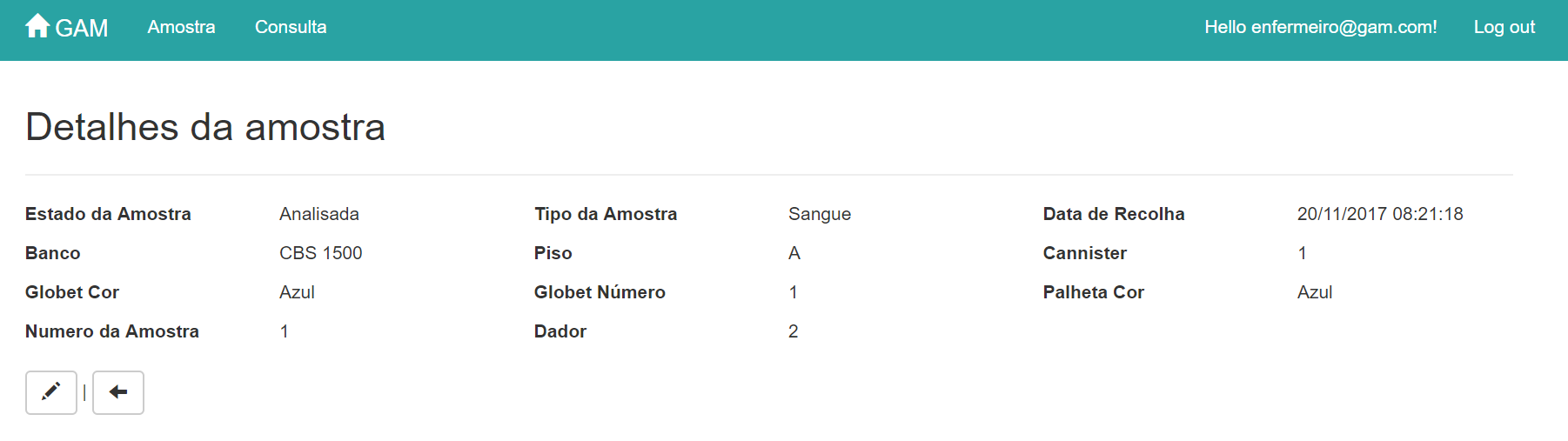


Figura 20 - REQ\_6 - View de detalhes de uma amostra

## REQ 7 – Pedido das análises de sangue iniciais

### Análise de requisitos

|  |  |
| --- | --- |
| Identificador |  |
| Nome |  |
| Descrição |  |

|  |  |
| --- | --- |
| Fluxo de Eventos - Humano | Fluxo de Eventos - Software |
| 1 – | 1.1 – A view “XPTO” chama/faz …  1.2 – O Controller “ZXY” chama/faz … |
| 2 – | Sem ação do sistema |
| 3 – | 3.1 –  3.2 - |

|  |  |  |
| --- | --- | --- |
| Pré-condições | Pós-condições | Validações |
| Sem nada a registar |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

### Diagrama de sequência

<Diagrama de sequência básico, que junta o Actor e a interação entre as diferentes camadas. Nos casos de uso mais completos/complexos, complementar o diagrama com uma breve descrição>

### Ilustração da funcionalidade

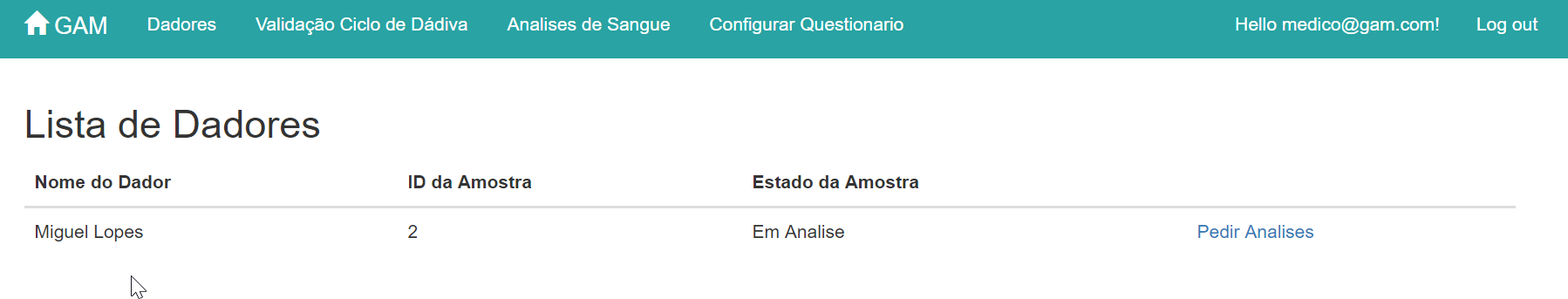


Figura 22 - REQ\_7: View da listagem de dadores com amostras pendentes de análise

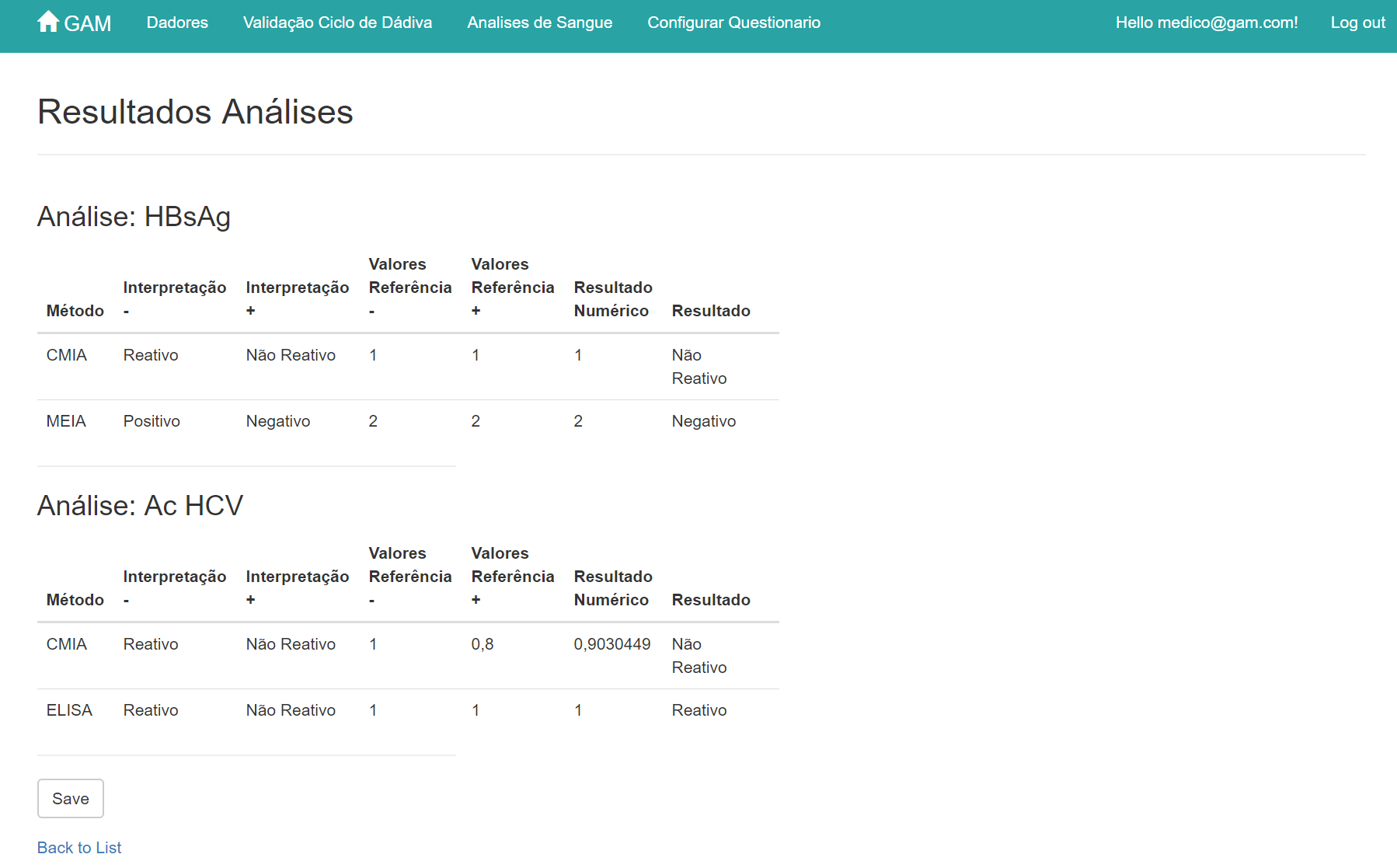


Figura 23 - REQ\_7: View dos resultados das análises gerados pela API iAnalysis

## REQ 8 – Ciclo da dádiva (abertura e consulta de informação)

### Análise de requisitos

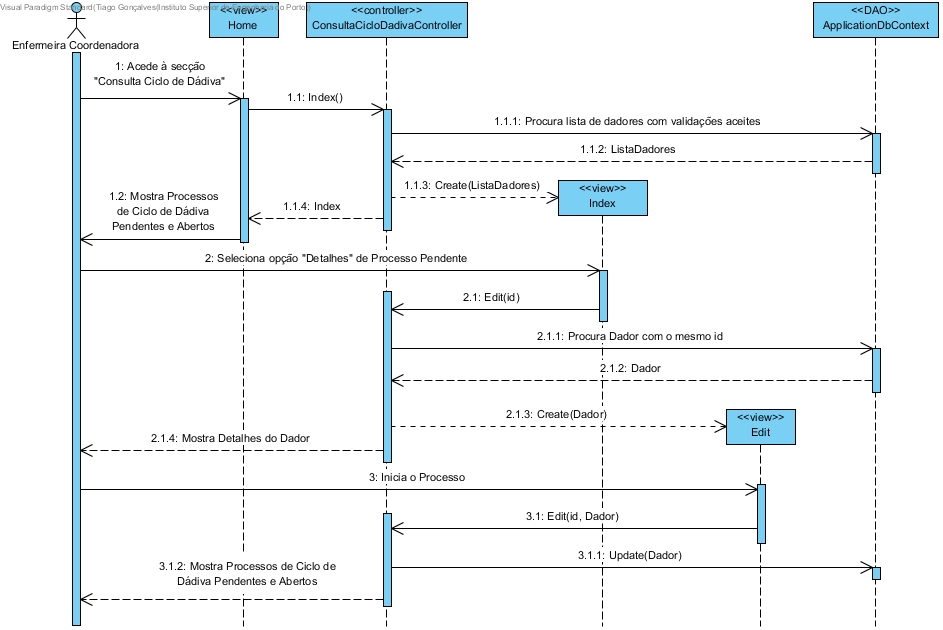
|  |  |
| --- | --- |
| Identificador |  |
| Nome | Ciclo da dádiva (abertura e consulta de informação) |
| Descrição | A Enfermeira Coordenadora enquanto utilizadora registada no sistema, pode aceder à secção “Consulta Ciclo de Dádiva” na sua página da iGAM para visualizar processos pendentes que podem ser iniciados, e processos abertos |

|  |  |
| --- | --- |
| Fluxo de Eventos - Humano | Fluxo de Eventos - Software |
| 1 – A Enfermeira Coordenadora acede à secção “Consulta Ciclo de Dádiva” | 1.1 – A view “Home” executa o método Index do Controller “ConsultaCicloDadivaController”  1.2 – O Controller “ConsultaCicloDadivaController” carrega a view de “ConsultaCicloDadiva” associada, com os ciclos de dadiva pendentes e aceites |
| 2 – A Enfermeira Coordenadora seleciona um ciclo de dádiva pendente ou aberto | 2.1 – A view “ConsultaCicloDadiva” executa o método Edit ou Details do Controller “ConsultaCicloDadivaController” consoante o caso  2.2 – O Controller “ConsultaCicloDadivaController” carrega a view de “ConsultaCicloDadiva” associada, com informação sobre o dador do respetivo ciclo de dádiva |
| 3 – A Enfermeira Coordenadora inicia o processo de ciclo de dádiva para ciclos pendentes | 3.1 – A view “ConsultaCicloDadiva” executa o método Edit do Controller “ConsultaCicloDadivaController”  3.2 – O Controller “ConsultaCicloDadivaController” atualiza o estado do processo do dador |

|  |  |  |
| --- | --- | --- |
| Pré-condições | Pós-condições | Validações |
| A Enfermeira Coordenadora deve estar autenticada no sistema | O estado do processo do dador é atualizado na base de dados | Sem nada a registar |

### Diagrama de sequência

<Diagrama de sequência básico, que junta o Actor e a interação entre as diferentes camadas. Nos casos de uso mais completos/complexos, complementar o diagrama com uma breve descrição>



### Ilustração da funcionalidade

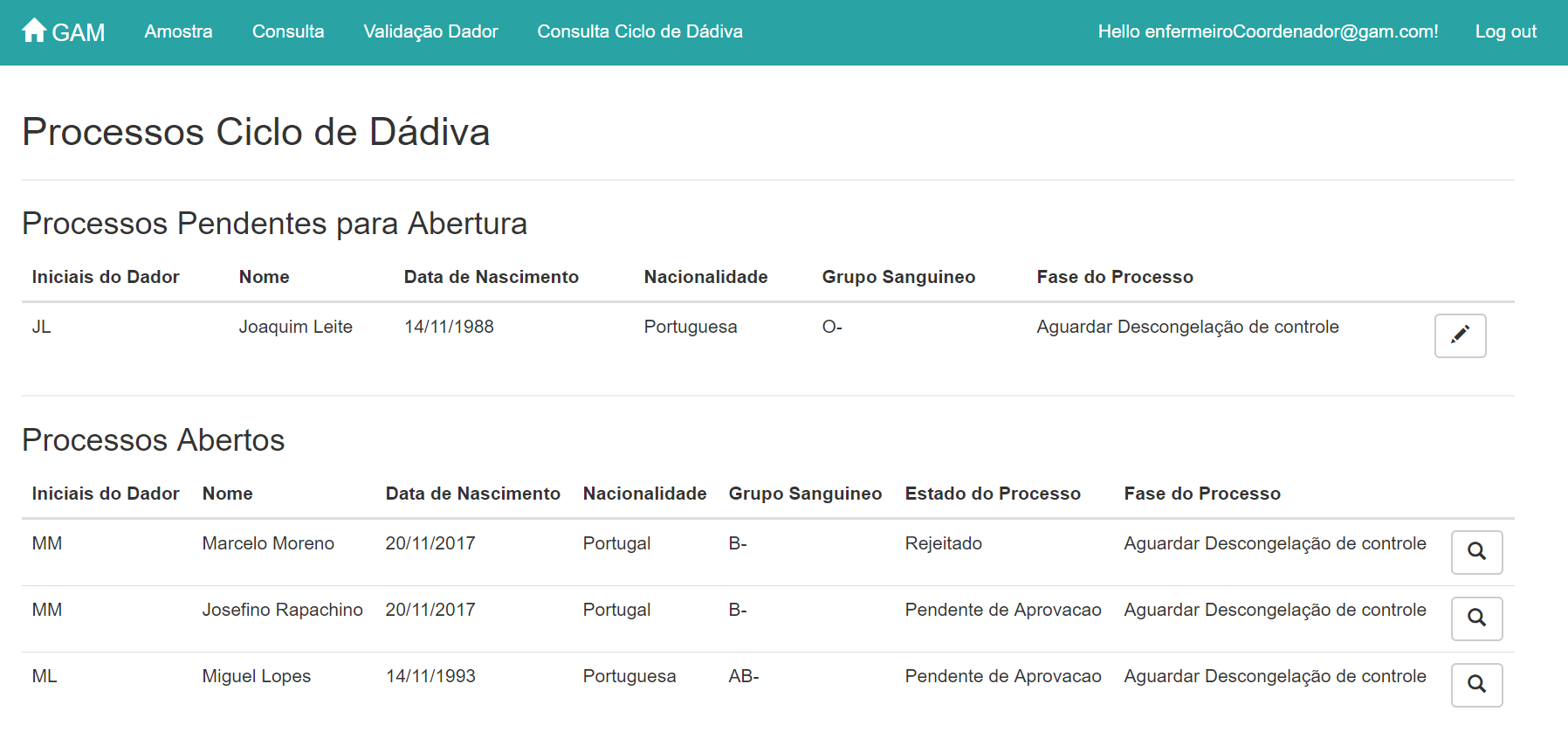


Figura 24 - REQ\_8: View dos processos pendentes e abertos

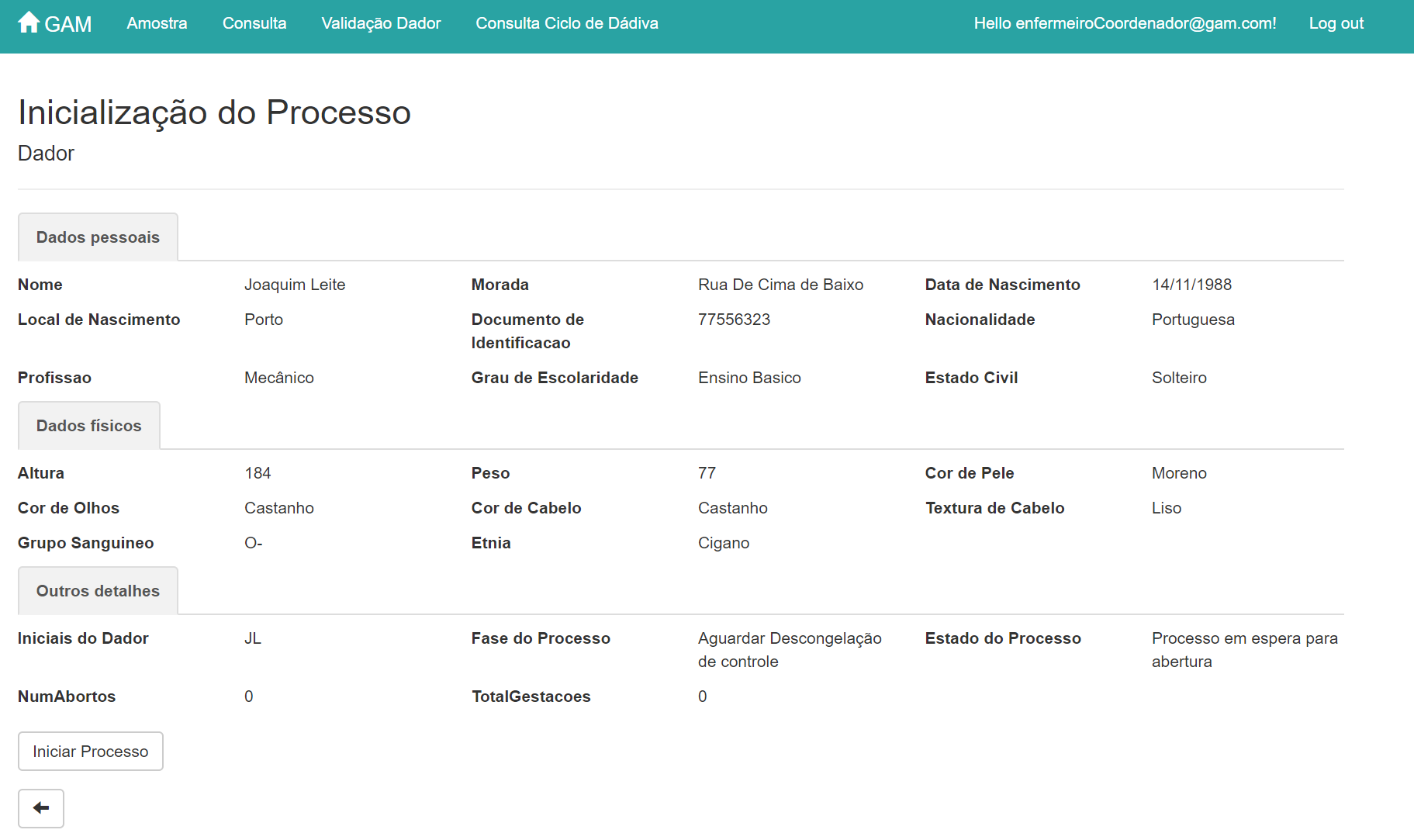


Figura 25 - REQ\_8: View de abertura do processo

## REQ 9 – Marcação de consultas

### Análise de requisitos

|  |  |
| --- | --- |
| Identificador |  |
| Nome |  |
| Descrição |  |

|  |  |
| --- | --- |
| Fluxo de Eventos - Humano | Fluxo de Eventos - Software |
| 1 – | 1.1 – A view “XPTO” chama/faz …  1.2 – O Controller “ZXY” chama/faz … |
| 2 – | Sem ação do sistema |
| 3 – | 3.1 –  3.2 - |

|  |  |  |
| --- | --- | --- |
| Pré-condições | Pós-condições | Validações |
| Sem nada a registar |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

### Diagrama de sequência

<Diagrama de sequência básico, que junta o Actor e a interação entre as diferentes camadas. Nos casos de uso mais completos/complexos, complementar o diagrama com uma breve descrição>

### Ilustração da funcionalidade

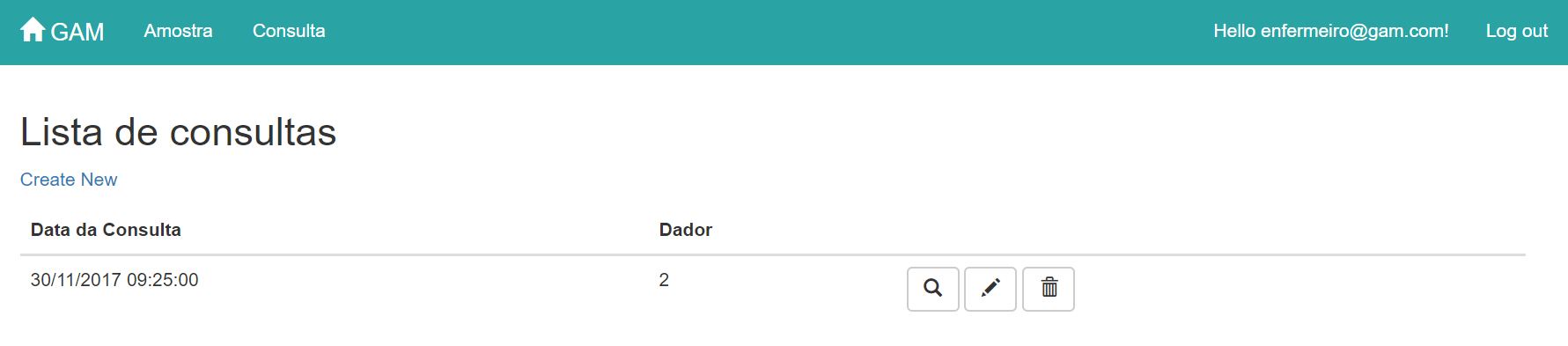


Figura 26 - REQ\_9: View da lista de consultas

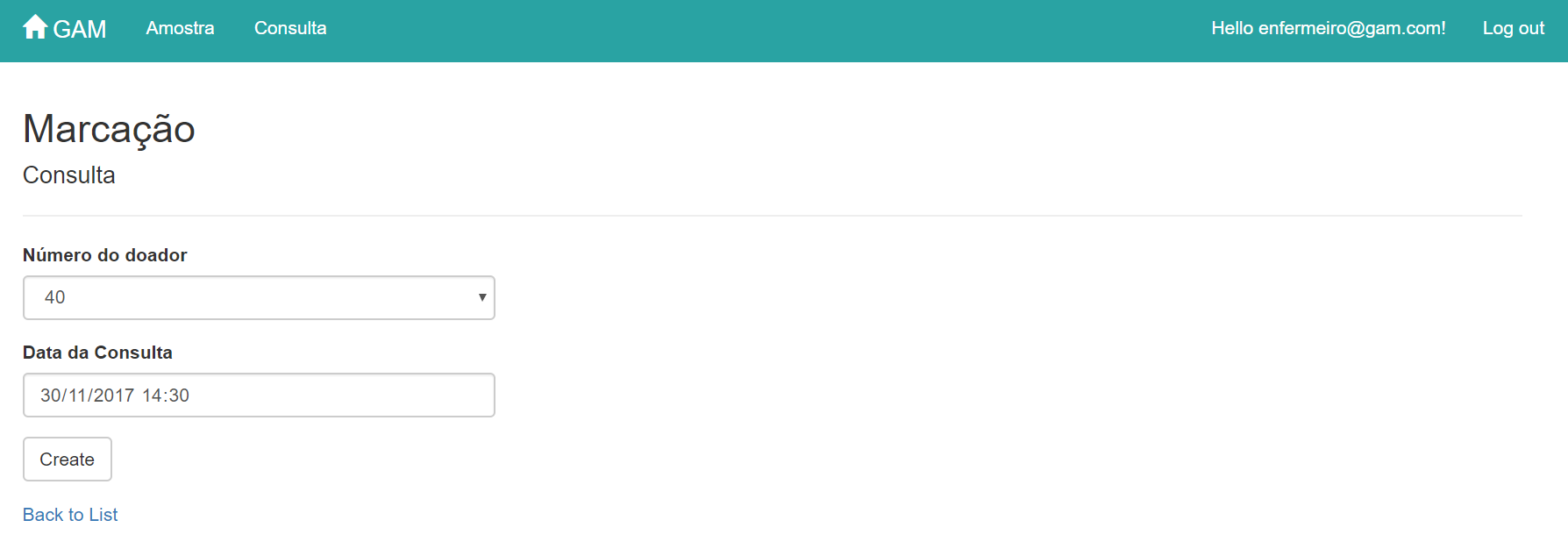


Figura 27 - REQ\_9: View da marcação de uma consulta

## REQ 10 – Aprovação/Rejeição do ciclo da dádiva

### Análise de requisitos

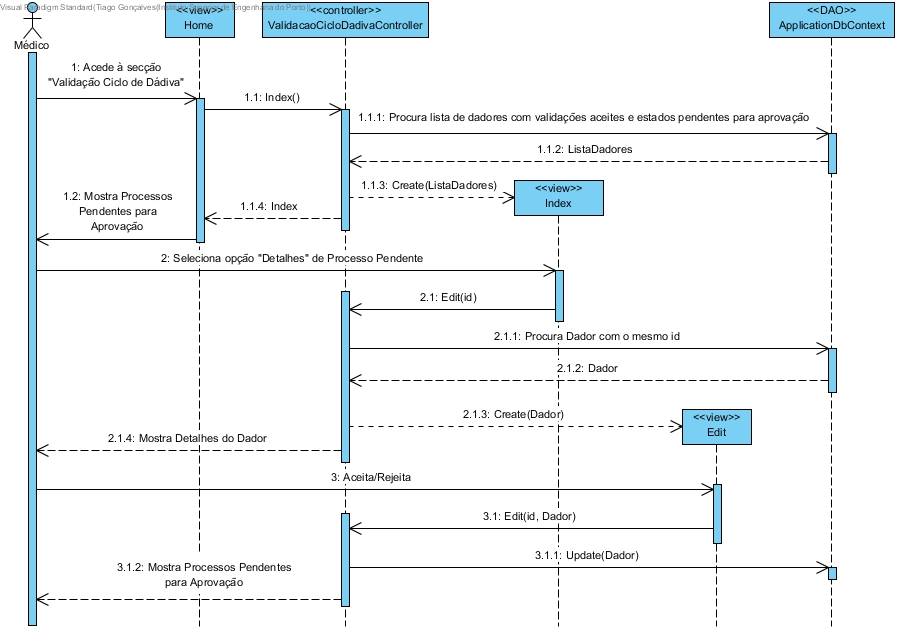
|  |  |
| --- | --- |
| Identificador |  |
| Nome | Aprovação/Rejeição do ciclo da dádiva |
| Descrição | O Médico enquanto utilizador registado no sistema, pode aceder à secção “Validação Ciclo de Dádiva” na sua página da iGAM para visualizar processos de ciclo de dádiva pendentes para aprovação. Estes processos podem ser aceites ou rejeitados. |

|  |  |
| --- | --- |
| Fluxo de Eventos - Humano | Fluxo de Eventos - Software |
| 1 – O Médico acede à secção “Validação Ciclo de Dádiva” | 1.1 – A view “Home” executa o método Index do Controller “ValidacaoCicloDadivaController”  1.2 – O Controller “ValidacaoCicloDadivaController” carrega a view de “ValidacaoCicloDadiva” associada, com os ciclos de dadiva pendentes para aprovação |
| 2 – O Médico seleciona um processo de ciclo de dádiva pendente para aprovação | 2.1 – A view “ValidacaoCicloDadiva” executa o método Edit do Controller “ValidacaoCicloDadivaController”  2.2 – O Controller “ValidacaoCicloDadivaController” carrega a view de “ValidacaoCicloDadiva” associada, com informação sobre o dador do respetivo ciclo de dádiva |
| 3 – O Médico aceita ou rejeita o processo de ciclo de dádiva selecionado | 3.1 – A view “ValidacaoCicloDadiva” executa o método Edit do Controller “ValidacaoCicloDadivaController”  3.2 – O Controller “ValidacaoCicloDadivaController” atualiza o estado do processo do dador para aceite ou rejeitado consoante a opção selecionada |

|  |  |  |
| --- | --- | --- |
| Pré-condições | Pós-condições | Validações |
| O Médico deve estar autenticado no sistema | O estado do processo do dador é atualizado na base de dados | Sem nada a registar |

### Diagrama de sequência

<Diagrama de sequência básico, que junta o Actor e a interação entre as diferentes camadas. Nos casos de uso mais completos/complexos, complementar o diagrama com uma breve descrição>



### Ilustração da funcionalidade

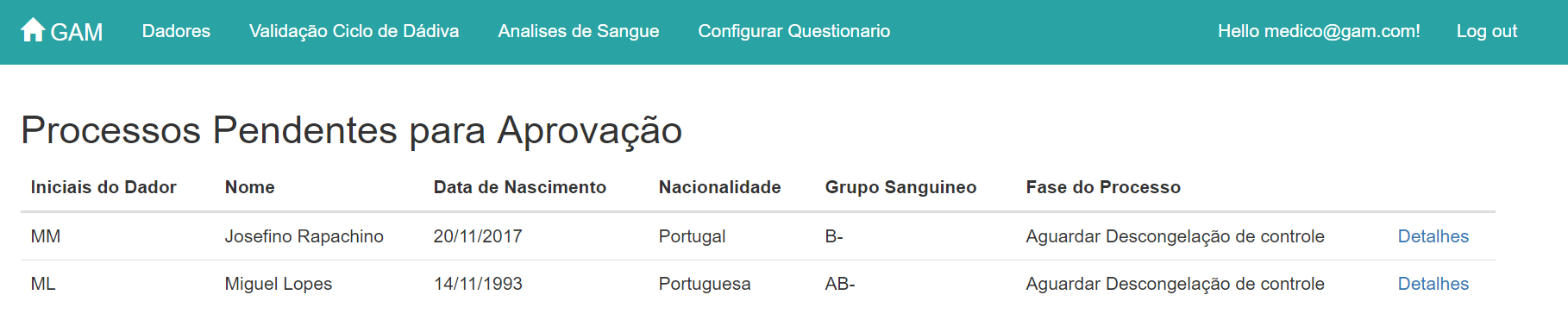


Figura 28 - REQ\_10: View da lista de processos pendentes de aprovação



Figura 29 - REQ\_10: View da aprovação/rejeição de um processo

## REQ 11 – Lista de trabalhos do laboratório

### Análise de requisitos

|  |  |
| --- | --- |
| Identificador |  |
| Nome |  |
| Descrição |  |

|  |  |
| --- | --- |
| Fluxo de Eventos - Humano | Fluxo de Eventos - Software |
| 1 – | 1.1 – A view “XPTO” chama/faz …  1.2 – O Controller “ZXY” chama/faz … |
| 2 – | Sem ação do sistema |
| 3 – | 3.1 –  3.2 - |

|  |  |  |
| --- | --- | --- |
| Pré-condições | Pós-condições | Validações |
| Sem nada a registar |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

### Diagrama de sequência

<Diagrama de sequência básico, que junta o Actor e a interação entre as diferentes camadas. Nos casos de uso mais completos/complexos, complementar o diagrama com uma breve descrição>

### Ilustração da funcionalidade



Figura 30 - REQ\_11: View da lista de amostras por analisar



Figura 31 - REQ\_11: View da lista de amostras analisadas

## REQ 12 – Registo de espermograma

### Análise de requisitos

|  |  |
| --- | --- |
| Identificador |  |
| Nome |  |
| Descrição |  |

|  |  |
| --- | --- |
| Fluxo de Eventos - Humano | Fluxo de Eventos - Software |
| 1 – | 1.1 – A view “XPTO” chama/faz …  1.2 – O Controller “ZXY” chama/faz … |
| 2 – | Sem ação do sistema |
| 3 – | 3.1 –  3.2 - |

|  |  |  |
| --- | --- | --- |
| Pré-condições | Pós-condições | Validações |
| Sem nada a registar |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

### Diagrama de sequência

<Diagrama de sequência básico, que junta o Actor e a interação entre as diferentes camadas. Nos casos de uso mais completos/complexos, complementar o diagrama com uma breve descrição>

### Ilustração da funcionalidade

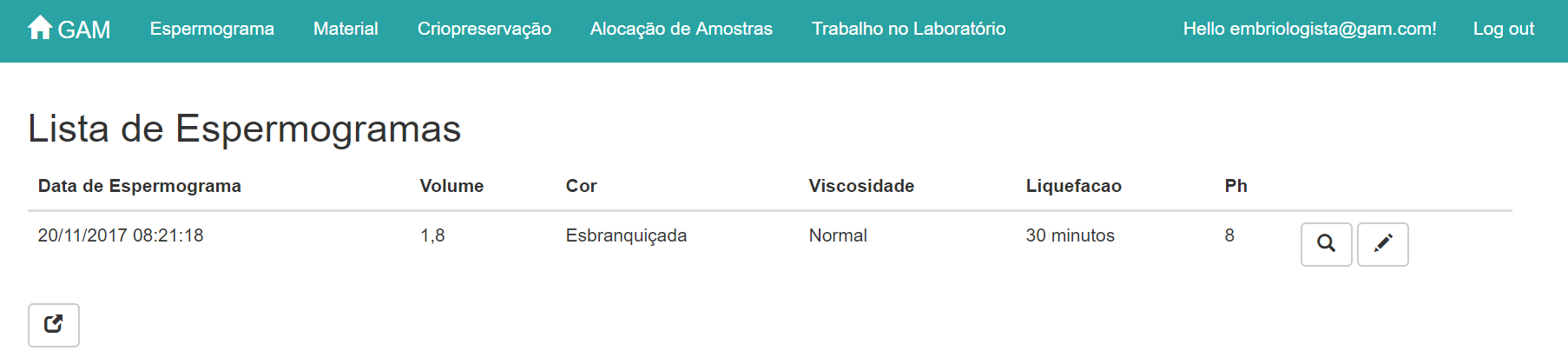


Figura 32 - REQ\_12 – View da lista de espermogramas

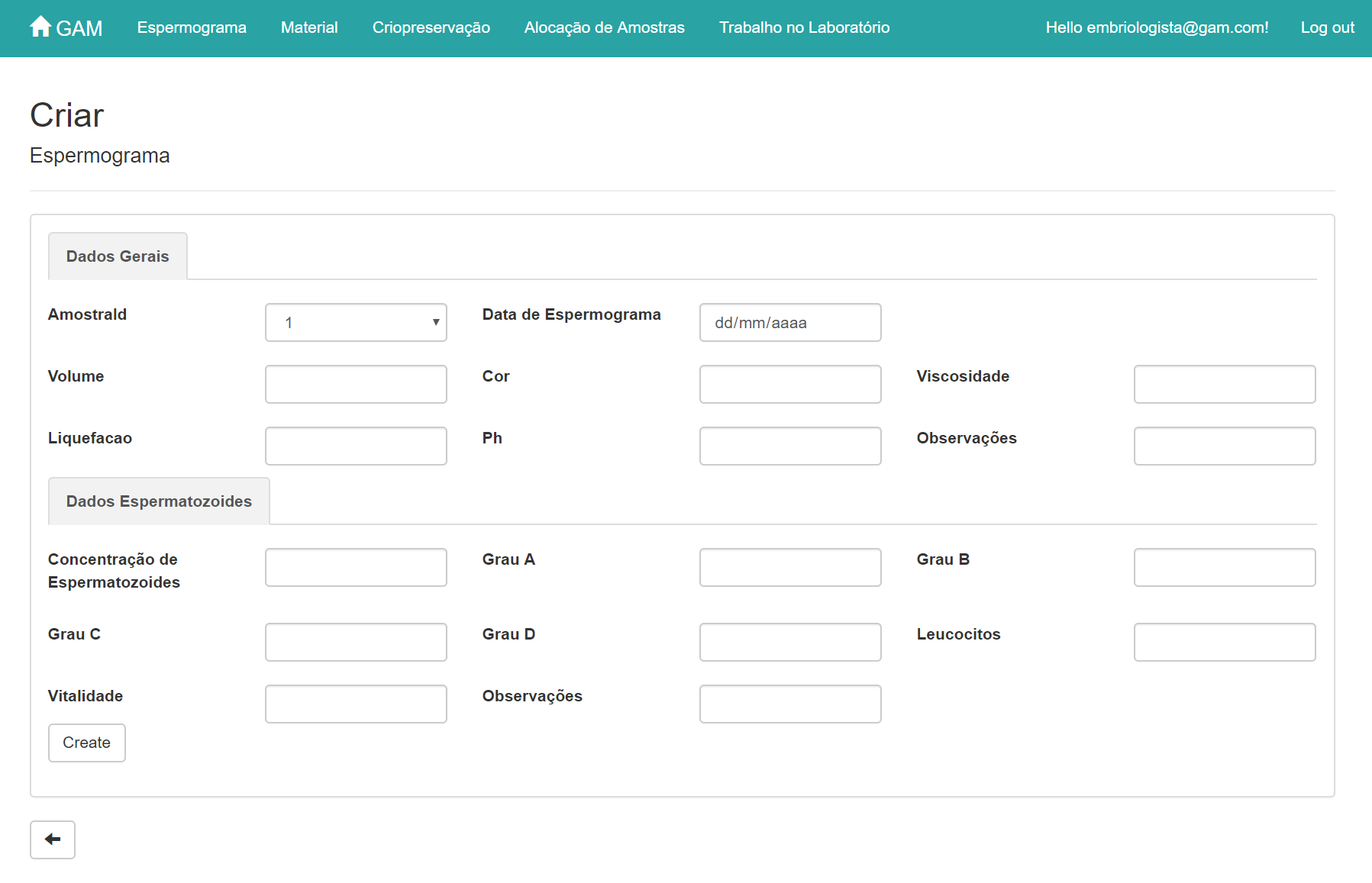


Figura 33 - REQ\_12: View de criação de espermograma

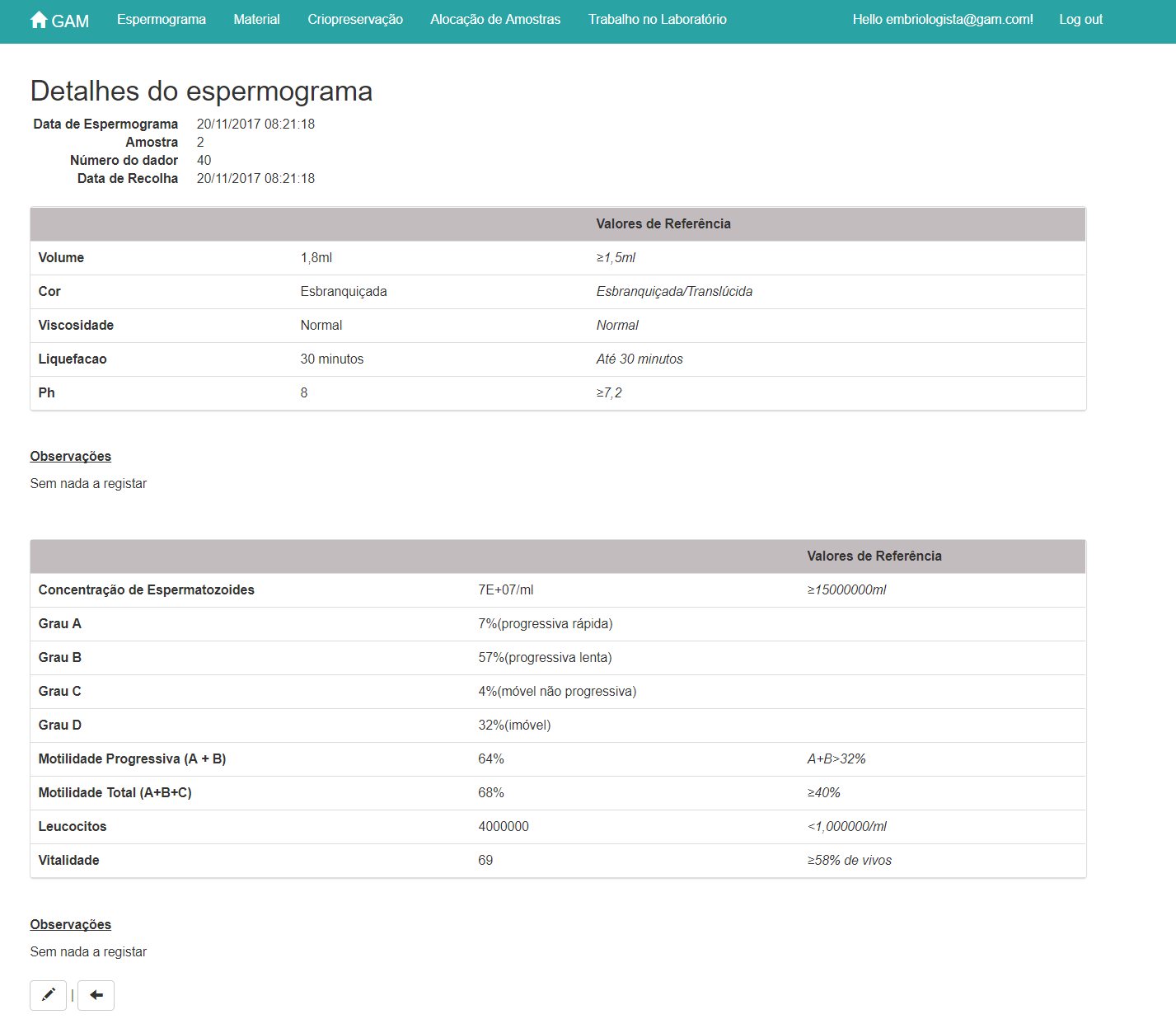


Figura 34 - REQ\_12: View dos detalhes do espermograma

## REQ 13 – Validação do espermograma

### Análise de requisitos

|  |  |
| --- | --- |
| Identificador |  |
| Nome |  |
| Descrição |  |

|  |  |
| --- | --- |
| Fluxo de Eventos - Humano | Fluxo de Eventos - Software |
| 1 – | 1.1 – A view “XPTO” chama/faz …  1.2 – O Controller “ZXY” chama/faz … |
| 2 – | Sem ação do sistema |
| 3 – | 3.1 –  3.2 - |

|  |  |  |
| --- | --- | --- |
| Pré-condições | Pós-condições | Validações |
| Sem nada a registar |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

### Diagrama de sequência

<Diagrama de sequência básico, que junta o Actor e a interação entre as diferentes camadas. Nos casos de uso mais completos/complexos, complementar o diagrama com uma breve descrição>

### Ilustração da funcionalidade

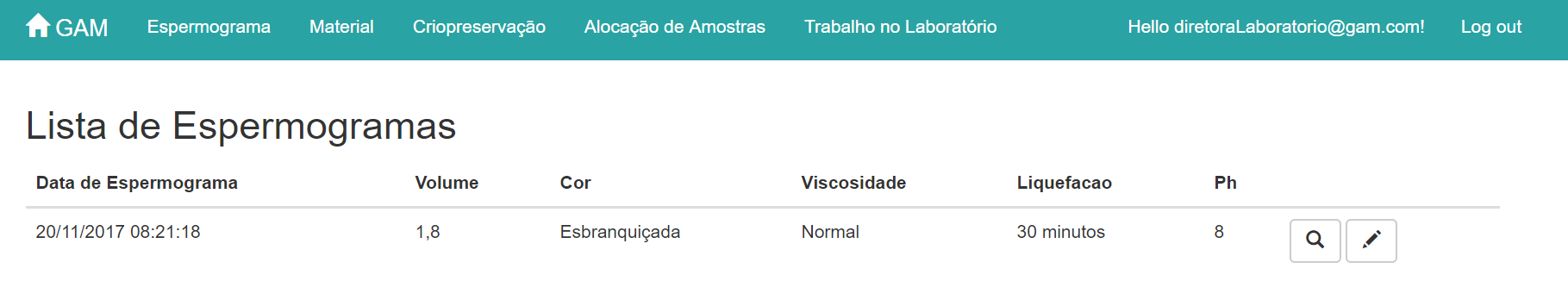


Figura 35 - REQ\_13: View da lista de espermogramas

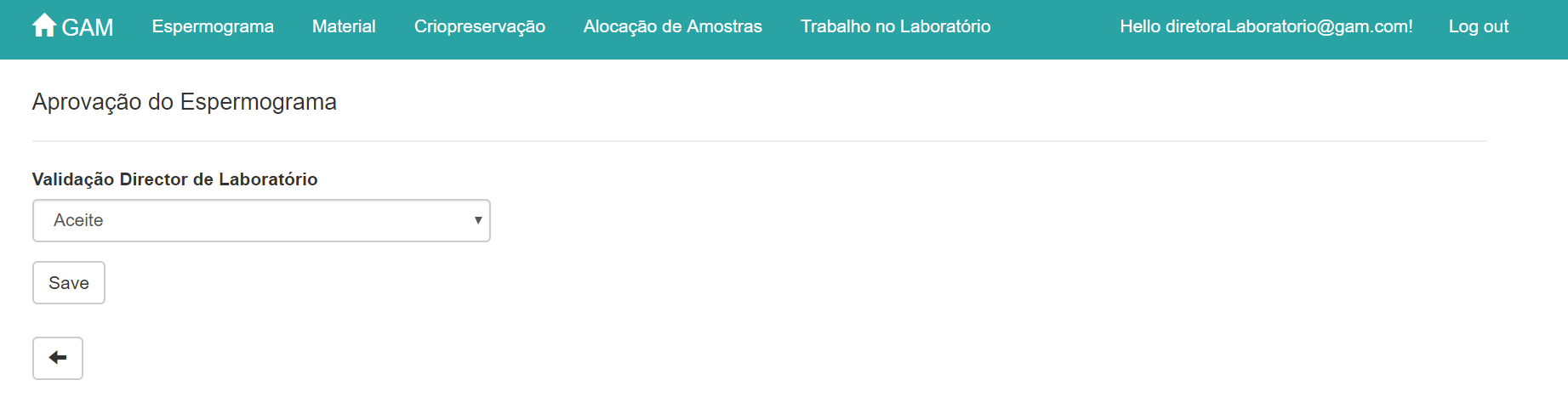


Figura 36 - REQ\_13: View de aprovação/rejeição de um espermograma

## REQ 14 – Criopreservação da amostra

### Análise de requisitos

|  |  |
| --- | --- |
| Identificador |  |
| Nome |  |
| Descrição |  |

|  |  |
| --- | --- |
| Fluxo de Eventos - Humano | Fluxo de Eventos - Software |
| 1 – | 1.1 – A view “XPTO” chama/faz …  1.2 – O Controller “ZXY” chama/faz … |
| 2 – | Sem ação do sistema |
| 3 – | 3.1 –  3.2 - |

|  |  |  |
| --- | --- | --- |
| Pré-condições | Pós-condições | Validações |
| Sem nada a registar |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

### Diagrama de sequência

<Diagrama de sequência básico, que junta o Actor e a interação entre as diferentes camadas. Nos casos de uso mais completos/complexos, complementar o diagrama com uma breve descrição>

### Ilustração da funcionalidade

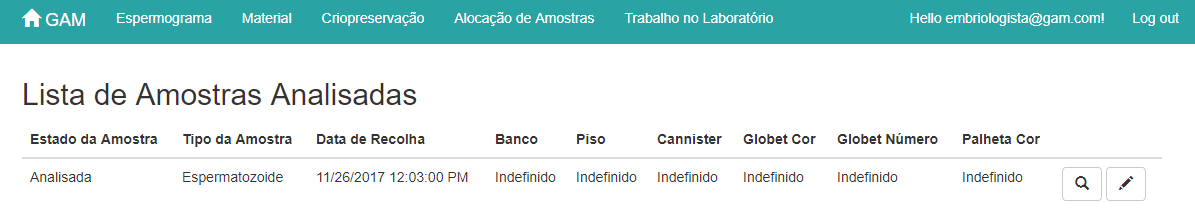


Figura 37 - REQ\_14: View da lista de amostras analisadas



Figura 38 - REQ\_14: View dos detalhes de uma amostra

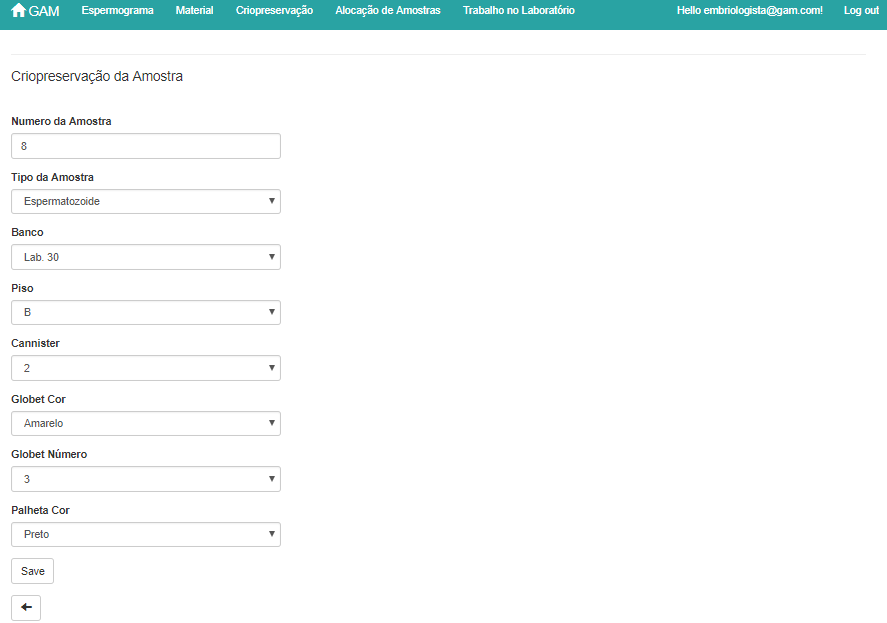


Figura 39 - REQ\_14: View da criopreservação de uma amostra

## REQ 15 – Visualização da alocação das amostras no Banco de criopreservação

### Análise de requisitos

|  |  |
| --- | --- |
| Identificador |  |
| Nome |  |
| Descrição |  |

|  |  |
| --- | --- |
| Fluxo de Eventos - Humano | Fluxo de Eventos - Software |
| 1 – | 1.1 – A view “XPTO” chama/faz …  1.2 – O Controller “ZXY” chama/faz … |
| 2 – | Sem ação do sistema |
| 3 – | 3.1 –  3.2 - |

|  |  |  |
| --- | --- | --- |
| Pré-condições | Pós-condições | Validações |
| Sem nada a registar |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

### Diagrama de sequência

<Diagrama de sequência básico, que junta o Actor e a interação entre as diferentes camadas. Nos casos de uso mais completos/complexos, complementar o diagrama com uma breve descrição>

### Ilustração da funcionalidade



Figura 40 - REQ\_15: View da lista de alocação das amostras

## REQ 17 – Registo de pedidos de gâmetas

### Análise de requisitos

|  |  |
| --- | --- |
| Identificador |  |
| Nome |  |
| Descrição |  |

|  |  |
| --- | --- |
| Fluxo de Eventos - Humano | Fluxo de Eventos - Software |
| 1 – | 1.1 – A view “XPTO” chama/faz …  1.2 – O Controller “ZXY” chama/faz … |
| 2 – | Sem ação do sistema |
| 3 – | 3.1 –  3.2 - |

|  |  |  |
| --- | --- | --- |
| Pré-condições | Pós-condições | Validações |
| Sem nada a registar |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

### Diagrama de sequência

<Diagrama de sequência básico, que junta o Actor e a interação entre as diferentes camadas. Nos casos de uso mais completos/complexos, complementar o diagrama com uma breve descrição>

### Ilustração da funcionalidade

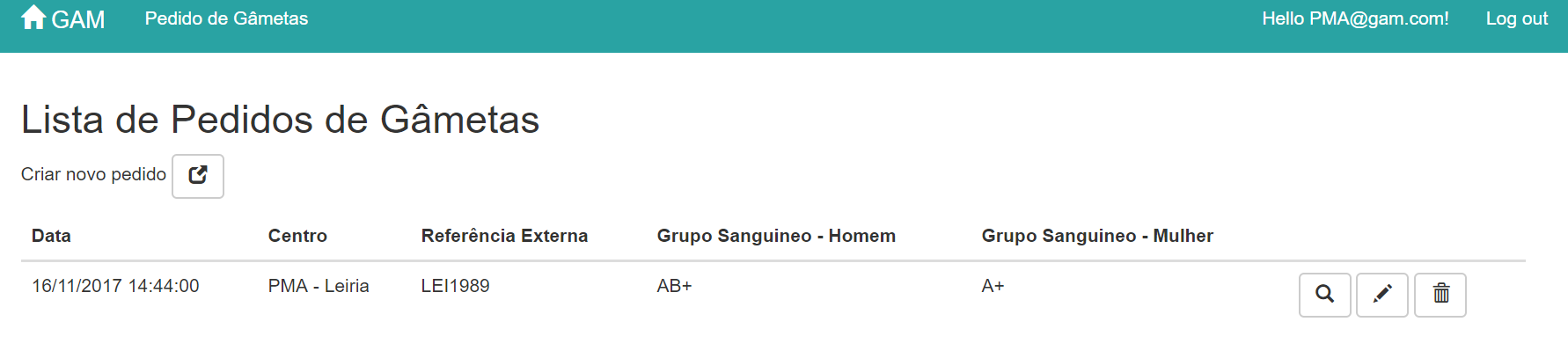


Figura 41 - REQ\_17: View da lista de pedidos de gâmetas

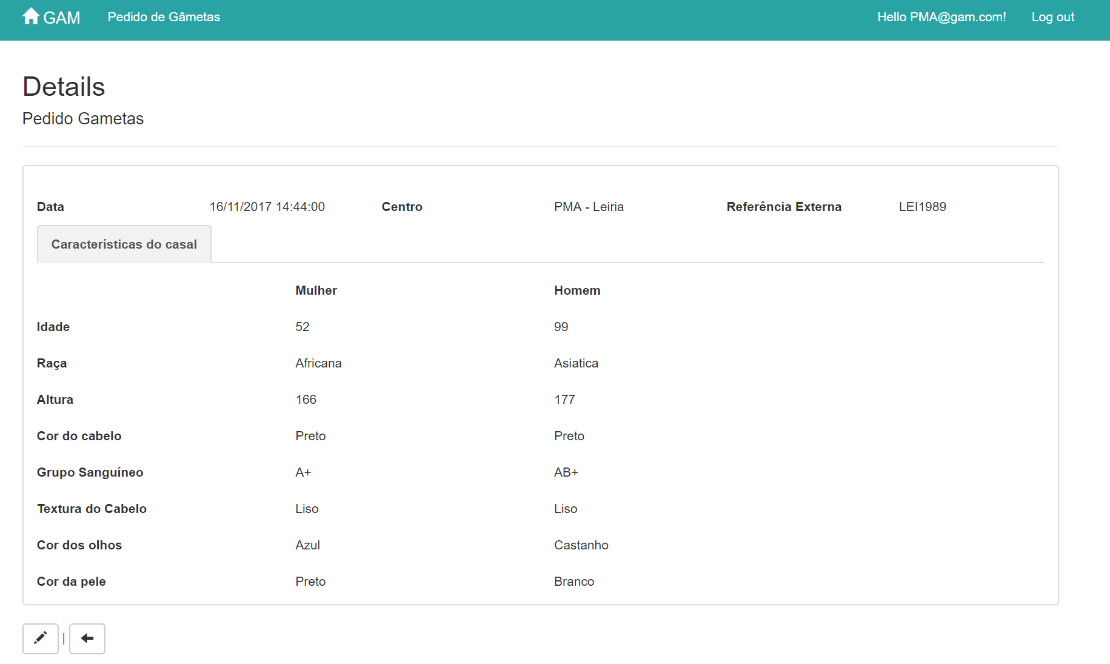


Figura 42 - REQ\_17: View de detalhes de um pedido de gâmetas

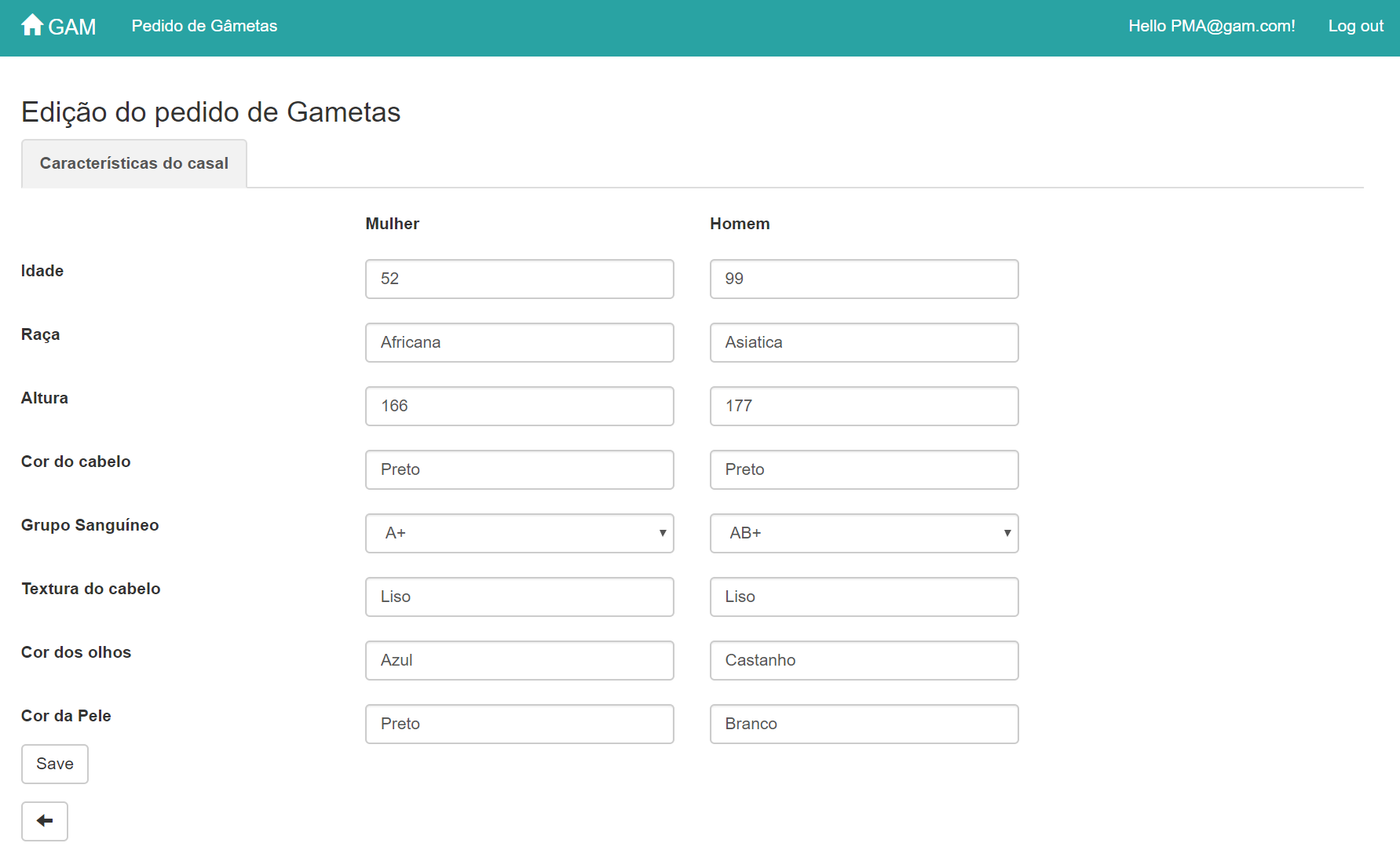


Figura 43 - REQ\_17: View de edição de um pedido de gâmetas

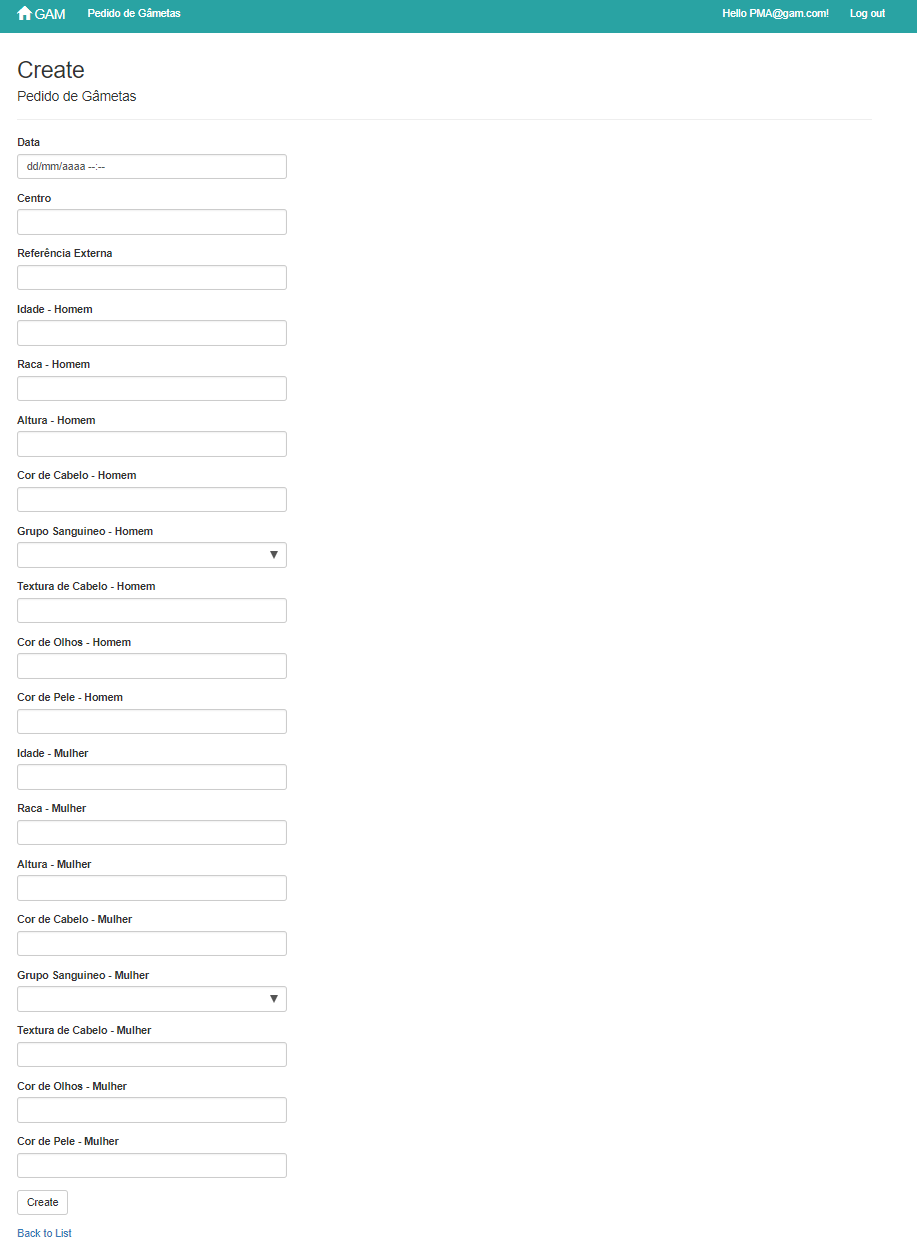


Figura 44 - REQ\_17: View da criação de um pedido de gâmetas

## REQ 18 – Registo de materiais usados (sem integração)

### Análise de requisitos

|  |  |
| --- | --- |
| Identificador |  |
| Nome |  |
| Descrição |  |

|  |  |
| --- | --- |
| Fluxo de Eventos - Humano | Fluxo de Eventos - Software |
| 1 – | 1.1 – A view “XPTO” chama/faz …  1.2 – O Controller “ZXY” chama/faz … |
| 2 – | Sem ação do sistema |
| 3 – | 3.1 –  3.2 - |

|  |  |  |
| --- | --- | --- |
| Pré-condições | Pós-condições | Validações |
| Sem nada a registar |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

### Diagrama de sequência

<Diagrama de sequência básico, que junta o Actor e a interação entre as diferentes camadas. Nos casos de uso mais completos/complexos, complementar o diagrama com uma breve descrição>

### Ilustração da funcionalidade

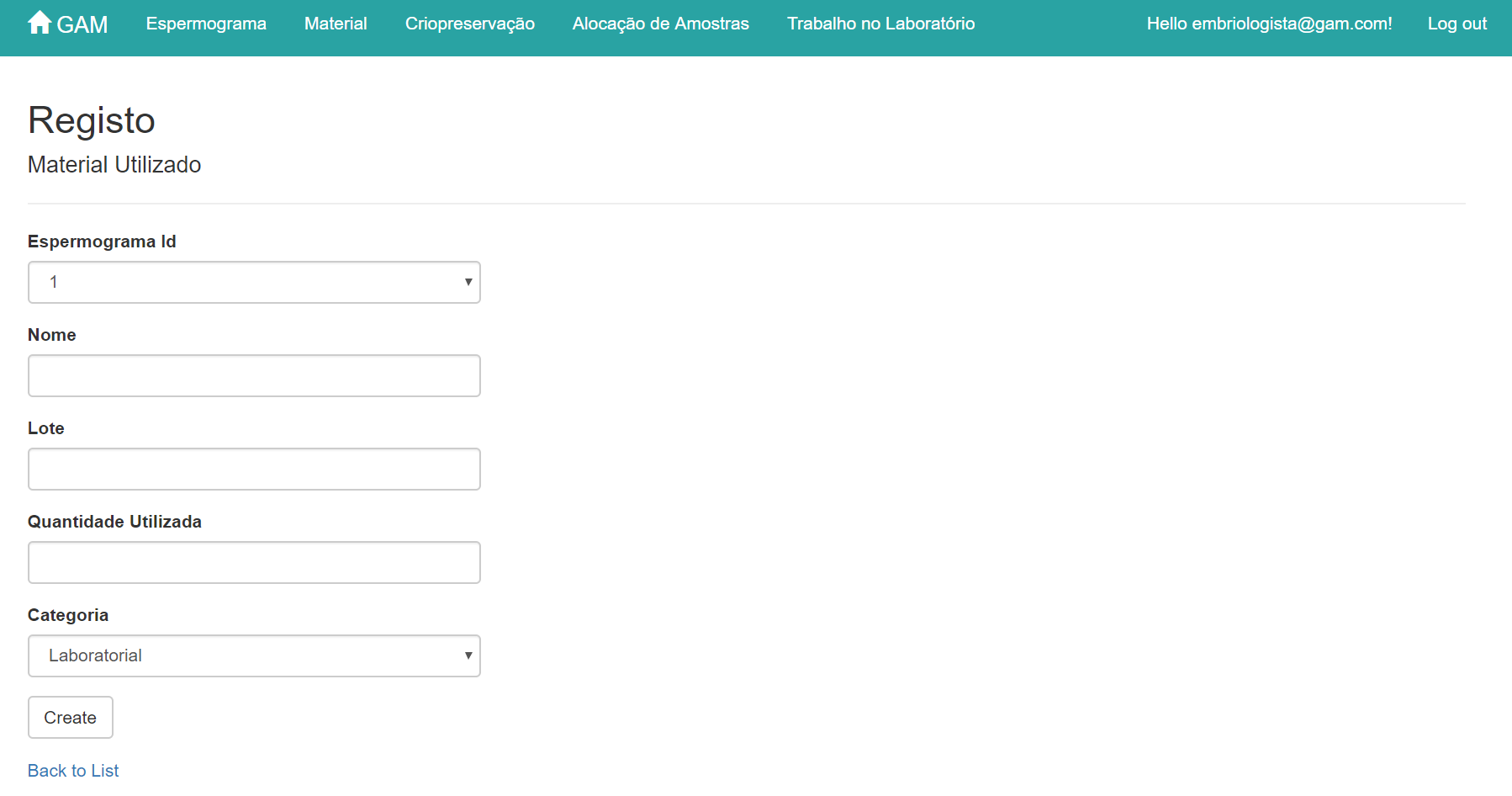


Figura 46 - REQ\_18: View do registo de material utilizado



Figura 45 - REQ\_18: View da lista de materiais usados

# Funcionalidades não implementadas

Todas as funcionalidades escalonadas para desenvolvimento neste *sprint* foram implementadas com sucesso. Contudo, a aprovação das mesmas ainda se encontra pendente da avaliação do público alvo.

# Pipeline

Para esta primeira iteração criamos um pipeline para o processo de desenvolvimento do *software*. Para a definição e implementação do pipeline, utilizamos o Jenkins, ferramenta de integração contínua e automatizada. O Jenkins auxilia na automatização do processo de desenvolvimento de *software* combinando os conceitos de *continuous integration* e *continuous delivery*.

A definição do script encontra-se no ficheiro *Jenkinsfile* na raiz do projeto. O script contém cinco *stages*:

* **Checkout**
  + Responsável por verificar a conexão ao repositório no Bitbucket, através de uma chave ssh

* **Build**
  + Responsável por compilar a solução GAM.sln no diretório correto
* **Unit Tests**
  + Responsável por compilar a solução GamTest.sln (testes unitários)
* **Publish Unit Test Results**
  + Responsável por criar um ficheiro de resultados (falhas) referente aos testes unitários (os ficheiros ficam guardados na pasta “TestResults”) e, deste modo, podemos analisar o número total de testes, testes que passaram/falharam, entre outros)

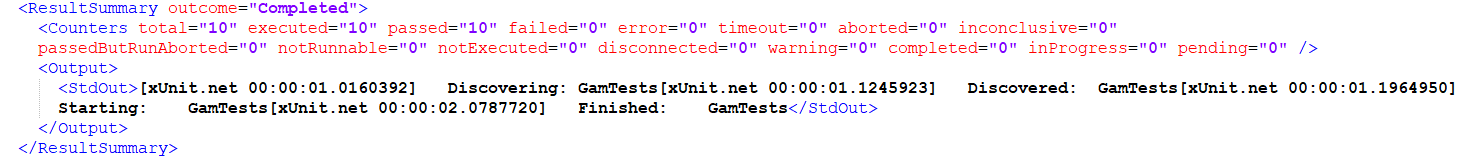


Figura 4 - Excerto do ficheiro de resultados (testes)

* **Archive**
  + Responsável por arquivar todos os artefactos

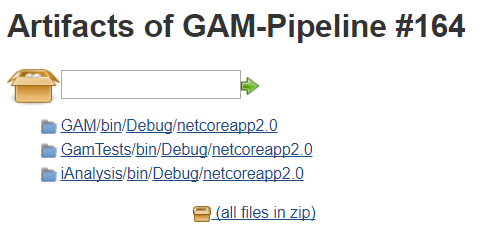


Figura 5 – Artefactos criados após stage Archive

É de salientar que o pipeline realiza *builds* periódicas de hora a hora. Para a implementação do pipeline foi necessário a instalação de alguns *plugins*, tais como: *Credentials Plugin, Git Plugin, MSBuild Plugin, Nuget Plugin, Pipeline, Pipeline: Stage View Plugin, MSTest Plugin e XUnit Plugin.*

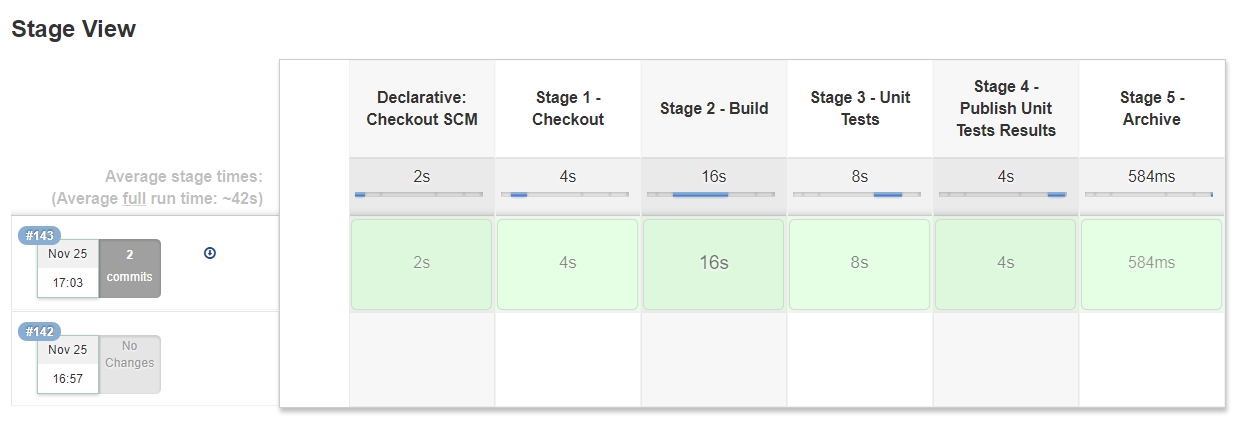


Figura 6 - Stage View Pipeline

# Distribuição de tarefas

A distribuição de tarefas foi realizada através de um *board*, usando a ferramenta Trello. Esta ferramenta é muito flexível, auxilia na criação de tarefas podendo a mesma ser atribuída a um membro da equipa e passar por diferentes estados (To Do, Doing, Done). Os requisitos são denominados *cards*, e podem ser deslocados consoante o estado em que se encontram.

Na tabela 2 apresentamos os requisitos que foram atribuídos a cada membro da equipa. É de salientar que o requisito 1 foi desenvolvido por toda a equipa.

Tabela 2 – Distribuição tarefas

|  |  |  |
| --- | --- | --- |
| Id | Identificação | Membro |
| REQ\_1 | Registo inicial do dador (sem fotografia) | Todos |
| REQ\_2 | Validação dos dados do dador | Tiago Gonçalves |
| REQ\_3 | Gestão de perfis de utilizadores | Tiago Gonçalves |
| REQ\_4 | Configuração do template de perguntas médicas | Daniel Bento |
| REQ\_5 | Consultas médicas via inquérito template | Daniel Bento |
| REQ\_6 | Registo de amostras | Maria Almeida |
| REQ\_7 | Pedido das análises de sangue iniciais | Manuel Correia |
| REQ\_8 | Ciclo da dádiva (abertura e consulta de informação) | Tiago Gonçalves |
| REQ\_9 | Marcação de consultas | Ana Barros |
| REQ\_10 | Aprovação / Rejeição do ciclo da dádiva | Tiago Gonçalves |
| REQ\_11 | Lista de trabalho do laboratório | Daniel Bento |
| REQ\_12 | Registo de espermograma | Maria Almeida |
| REQ\_13 | Validação do espermograma | Maria Almeida |
| REQ\_14 | Criopreservação da amostra | Ana Barros |
| REQ\_15 | Visualização da alocação das amostras no Banco de criopreservação | Maria Almeida |
| REQ\_16 | Integração com o laboratório de análises de sangue | Manuel Correia |
| REQ\_17 | Registo de pedidos de gâmetas | Manuel Correia |
| REQ\_18 | Registo de materiais usados (sem integração) | Ana Barros |
|  | Definição do pipeline | Ana Barros |

# Tempos

Tabela 3 - Duração de cada requisito

|  |  |  |  |
| --- | --- | --- | --- |
| Id | Identificação | Membro | Duração |
| REQ\_1 | Registo inicial do dador (sem fotografia) | Todos | 9h |
| REQ\_2 | Validação dos dados do dador | Tiago Gonçalves | 1h |
| REQ\_3 | Gestão de perfis de utilizadores | Tiago Gonçalves | 8h |
| REQ\_4 | Configuração do template de perguntas médicas | Daniel Bento | 8h |
| REQ\_5 | Consultas médicas via inquérito template | Daniel Bento | 4h |
| REQ\_6 | Registo de amostras | Maria Almeida | 3:30h |
| REQ\_7 | Pedido das análises de sangue iniciais | Manuel Correia | 4h |
| REQ\_8 | Ciclo da dádiva (abertura e consulta de informação) | Tiago Gonçalves | 3h |
| REQ\_9 | Marcação de consultas | Ana Barros | 2h |
| REQ\_10 | Aprovação / Rejeição do ciclo da dádiva | Tiago Gonçalves | 2h |
| REQ\_11 | Lista de trabalho do laboratório | Daniel Bento | 2:30h |
| REQ\_12 | Registo de espermograma | Maria Almeida | 2:30h |
| REQ\_13 | Validação do espermograma | Maria Almeida | 3h |
| REQ\_14 | Criopreservação da amostra | Ana Barros | 4h |
| REQ\_15 | Visualização da alocação das amostras no Banco de criopreservação | Maria Almeida | 3:30h |
| REQ\_16 | Integração com o laboratório de análises de sangue | Manuel Correia | 7:30h |
| REQ\_17 | Registo de pedidos de gâmetas | Manuel Correia | 4h |
| REQ\_18 | Registo de materiais usados (sem integração) | Ana Barros | 2h |
|  | Definição do pipeline | Ana Barros | 7h |

Quanto ao tempo que demorou cada requisito a ficar concluído, importa referir que se trata de uma estimativa uma vez que não foi utilizada nenhuma ferramenta para a medição real do tempo. Este é porventura um dos pontos sobre os quais devemos refletir e tentar melhor no próximo *sprint*. De realçar que o desenvolvimento do trabalho destinado a este *sprint* foi sempre pautado por uma grande entreajuda dos elementos da equipa de desenvolvimento, tendo sido constante a colaboração de vários elementos para a realização de diversos *issues*. Por fim, importa referir que, como é natural, foram existindo variadas tarefas ao longo do processo de desenvolvimento (como por exemplo a resolução de *bugs*), que ocuparam tempo considerável aos elementos do grupo e que não foi possível quantificar e associar à tabela apresentada anteriormente. Os valores mencionados na tabela 3 perfazem um total de 100 horas de trabalho de desenvolvimento.

# Medidas de qualidade de software

Quanto a medidas de qualidade de software, a equipa de desenvolvimento tentou sempre manter o código limpo e isolado por camadas, de modo a construir uma estrutura organizada e escalável. No mesmo setor da escalabilidade encontra-se a nossa decisão de migrar a base de dados do projeto para a *cloud*. Por fim, desenvolvemos alguns testes unitários às funcionalidades da aplicação. Contudo, o nível de profundidade dos mesmos não está ainda nos parâmetros que consideramos necessários para garantir uma aplicação altamente fiável, pelo que este é, sem sombra de dúvida, um dos pontos sobre o qual o grupo de trabalho vai refletir de modo a tentar melhorar na próxima iteração.