|  |  |  |
| --- | --- | --- |
| Description: logo | **DEI** | **Departamento**  **Engenharia**  **Informática** |
|  | |
| **Integração de Sistemas**  Engenharia Informática  (3º ano, 1º sem - 2015/2016) | | |

***Data Integration Hub***

*Relatório do projeto*

|  |
| --- |
| ***Nome:***  ***Número:***  ***Turno Prático:***  ***Docente do Turno Prático:*** |

|  |
| --- |
| ***Nome:***  ***Número:***  ***Turno Prático:***  ***Docente do Turno Prático:*** |

|  |
| --- |
| ***Nome:***  ***Número:***  ***Turno Prático:***  ***Docente do Turno Prático:*** |

Índice

[1. Introdução 3](#_Toc433913618)

[2. Módulo 1 – Sensor Data Controller and Alarming Module   3](#_Toc433913619)

[3. Módulo 2 - DBPersistence 3](#_Toc433913620)

[4. Módulo 3 – Service Conector 3](#_Toc433913621)

[5. Requisitos não implementados 3](#_Toc433913622)

[6. Contribuição de cada elemento do grupo 3](#_Toc433913623)

[7. Conclusão 3](#_Toc433913624)

[*Anexo A* – Listagem dos Ficheiro XML utilizados nos vários módulos 4](#_Toc433913625)

[*Anexo B* – Outras informações 4](#_Toc433913626)

##### Introdução

[Breve descrição do trabalho. Notas introdutórias.

Nas seções seguintes deve descrever todos os pormenores de implementação dos vários módulos.]

No âmbito da Unidade Curricular de Integração de Sistemas foi-nos solicitado um projeto com o objetivo de pôr em prática os conhecimentos adquiridos ao longo do semestre.

O objetivo do projeto consiste em desenvolver uma aplicação que permitisse ler os dados de sensores (temperatura, humidade e pressão barométrica). Para essa comunicação é necessário existir um Sensor Data Controller (dados de sensores em ficheiro dll) e Communication HUB (Comunicação entre serviços) onde este vai receber dados do Sensor Data Controller, Alarming System (regras do alarme) que permite gerir as regras dos dados recebido pelo Communication HUB, DBPersistence que vai permitir ler os dados do Communication HUB e enviar esses dados para a base de dados e o Service API que fornece as informações exigidas pelo GUI cliente (Service Client application) onde o utilizador pode monitorar as informações dos sensores.

##### Módulo 1 – Sensor Data Controller and Alarming Module

[Descrição das funcionalidades principais. Análise das diversas componentes do módulo d desenvolver. Explicar detalhadamente a abordagem seguida na implementação do módulo Sensor Data Controller, implementação do HUB e do módulo de Alarming. Apresentaçaõ e justificação da estrutura adotada no ficheiro de configurações em XML. Possíveis subsecções sugeridas: Interface, Explicar Configurações; Funcionalidades; Extras, etc.]

O Sensor Data Controller é um serviço que permite ler o serviço Sensor Data Generator (DLL) onde contem os valores dos sensores e são enviadas para o Cummunication HUB onde esse vai guardar os dados em formato XML.

O serviço Alarming System serve para gerir as regras dos alarmes dos sensores (Temperatura, Humidade e Pressão), em aplicação em Windows Presentation Foundation (WPF) para ser mais fácil na compreensão dos utilizadores na utilização da aplicação como mostra na figura.

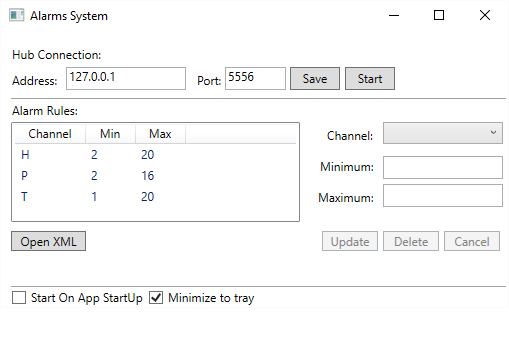


Figura 1-Aplicação Alarms System

Figura 2

Os sensores representam diferentes tipos de canais (T,H ou P) com os seus respetivos valores. A aplicação vai permitir ao utilizador definir regras para cada canais, o valor máximo e mínimo que serve para alertar ao Communication HUB que o valor do respetivo canal ultrapassa os valores definidos pelo utilizador. Vai ainda permitir definir o endereço IP e a porta para comunicar com o Communication HUB onde esse também terá os endereços. Essas mesmas regras são guardadas num ficheiro XML para depois serem publicados à Communication HUB informando que os valores ultrapassam os valores definidos nas regras. Ainda vai permitir ao utilizador visualizar o ficheiro XML através do botão “Open XML”. O serviço Alarming System, para obter os dados do Communication HUB, apenas é iniciado quando o utilizador clicar o botão “start” após definir o endereço IP e porta, e o serviço Communication HUB deverá também estar iniciado para que ele possa obter os dados do serviço Sensor Data Controller e enviar para o serviço Alarming System.

##### Módulo 2 - DBPersistence

[Explicação da abordagem adotada na implementação de todos as funcionalidades implementadas, esquema da base de dados e justificação do mesmo, apresentação e justificação da estrutura do ficheiro XML proposto para este módulo, etc. ]

O serviço DBPersistence permite ler os dados, em formato XML, armazenado no Communication HUB para serem analisados e enviar à base de dados.

##### Módulo 3 – Service Conector

[Explicação de todos as funcionalidades implementadas neste módulo ao nível da camada de serviço e da aplicação cliente, com justificação das mesmas.]

O serviço utilizado no Service Conector para aplicar em WebServices é SOAP, que é um protocolo de transferência de mensagens em formato XML para uso em aplicações. Uma das maiores vantagens disso é que várias linguagens e ferramentas conseguem ler e gerar mensagens facilmente, pois essas ferramentas corporativas tiram vantagem do padrão e possibilitam filtrar, enfileiramento, classificar e redirecionar as mensagens trocadas entre sistemas.

##### Requisitos não implementados

[Lista de todos as funcionalidades não implementadas, ou implementadas apenas parcialmente para cada módulo].

##### Contribuição de cada elemento do grupo

[Indicação clara das funcionalidades implementadas por cada membro do grupo em cada módulo].

Modulo 1 - Mickaël Gomes , 2130693

Modulo 2 –

Modulo 3 -

##### Conclusão

[Processo de desenvolvimento do trabalho, enfatizando eventuais dificuldades encontradas. Notas finais.]

##### *Anexo A* – Listagem dos Ficheiro XML utilizados nos vários módulos

[Colocar o XSD e respetivo XML para o(s) diverso(s) ficheiro(s) de XML que foram criados no projeto.]

##### *Anexo B* – Outras informações

[Logins e passwords necessárias para testar o trabalho. Devem também indicar todos os ficheiros de configuração que é necessário alterar para executar o trabalho corretamente.