

BACHARELADO EM ENGENHARIA DE CONTROLE E AUTOMAÇÃO

FREDERICO FERNANDES GONÇALVES BERNARDEZ

THAÍSSA FAGUNDES FERNANDES

DESENVOLVIMENTO DE SIMULADORES INDUSTRIAS COM DISPONIBILIZAÇÃO  
VIA WEB

CAMPOS DOS GOYTACAZES - RJ

2015

FREDERICO FERNANDES GONÇALVES BERNARDEZ

THAISSA FAGUNDES FERNANDES

DESENVOLVIMENTO DE SIMULADORES INDUSTRIAIS COM  
DISPONIBILIZAÇÃO VIA WEB

Trabalho de conclusão apresentado  
ao curso de Engenharia de Controle  
e Automação como requisito parcial  
para obtenção de título de  
Engenheiro de Controle e  
Automação.

Orientador: D.Sc Adelson Siqueira  
Carvalho

Instituto Federal Fluminense – IFF  
Campus Campos Centro  
Engenharia de Controle e Automação

CAMPOS DOS GOYTACAZES  
2015

Dados Internacionais de Catalogação na Publicação (CIP)

Biblioteca. Setor de Processos Técnicos (IFF)

B519d Bernardez, Frederico Fernandes Gonçalves.

Desenvolvimento de simuladores industriais com disponibilização via web / Frederico Fernandes Gonçalves Bernardez, Thaissa Fagundes Fernandes – 2015.

80 f.: il. color.

Orientador: Adelson Siqueira Carvalho.

Monografia (Engenharia de Controle e Automação). Instituto Federal de Educação, Ciência e Tecnologia Fluminense. Campus Campos Centro. Campos dos Goytacazes (RJ), 2015.

Referências: p. 54-55.

FREDERICO FERNANDES GONÇALVES BERNARDEZ

THAISSA FAGUNDES FERNANDES

DESENVOLVIMENTO DE SIMULADORES INDUSTRIAIS COM  
DISPONIBILIZAÇÃO VIA WEB

Trabalho de conclusão apresentado  
ao curso de Engenharia de Controle  
e Automação como requisito parcial  
para obtenção de título de  
Engenheiro de Controle e  
Automação.

Trabalho aprovado. Campos dos Goytacazes, 09 de dezembro de 2015

Examinadores:

---

Prof. D. Sc. Adelson Siqueira Carvalho  
Doutor em Informática na Educação – UFRGS  
Instituto Federal Fluminense Campus Campos-Centro

---

Prof. Fábio Junio dos Santos Coelho  
Instituto Federal Fluminense Campus Macaé

---

Prof. M. Sc. Rodrigo César Teixeira Baptista  
Mestre em Pesquisa Operacional e Inteligência Computacional  
Instituto Federal Fluminense Campus Campos-Centro

CAMPOS DOS GOYTACAZES

2015

*Dedicamos este trabalho à Deus e aos nossos familiares que estiveram sempre conosco durante esta longa caminhada.*

## **AGRADECIMENTOS**

À Deus por ter nos dado saúde e força para superar as dificuldades.

Ao nosso orientador, pelo empenho, suporte e dedicação à elaboração deste trabalho.

A todos da nossa família que, de alguma forma, incentivaram-nos na constante busca pelo conhecimento.

A todos que direta ou indiretamente fizeram parte da nossa formação, o nosso muito obrigado.

## **RESUMO**

Este trabalho apresenta o desenvolvimento de um simulador industrial de processamento de gás natural com disponibilização via web, que pode ser acessado via *link*. Através da integração dos softwares Matlab e ScadaBR, e do servidor MatrikonOPC, foi possível simular o uso dos controladores automáticos, o comportamento da planta industrial, a mudança de setpoint e seleção automático/manual dos controladores simulados, entre outros. O Matlab foi utilizado para criação do modelo matemático da planta, o ScadaBR para criação das telas de supervisão e disponibilização via web e o MatrikonOPC foi utilizado como servidor de comunicação entre os softwares Matlab e ScadaBR. Como última etapa do desenvolvimento do simulador, foram feitos testes de disponibilização em rede local e em seguida na web, permitindo que o acesso aos dados da planta simulada possa ser feito através de qualquer dispositivo que possua conexão com a internet. Além disso, foram criados grupos de acesso, para que, através de senhas pessoais, cada usuário só tenha permissão para acessar o que for de sua responsabilidade.

Palavras-chave: Simulador. MatrikonOPC. Matlab. Simulink. ScadaBR. Processamento de Gás Natural. Integração de Software.

## **ABSTRACT**

This coursework presents the development of processing gas plant simulator providing access via internet, that can be accessed through the link. By integrating Matlab and ScadaBR software, and MatrikonOPC Server, it was possible to simulate the use of automatic controllers, industrial plant behavior, changing the setpoint, automatic / manual selection of the simulated controllers, and more. Matlab was used to create the mathematical model of the plant, ScadaBR for creating the supervisory screens and make it available via internet. MatrikonOPC was used as a communication server between Matlab and ScadaBR software. As the last stage of development simulator, tests were made available in the local network and the web, allowing any device that has an internet connection, access to the simulated plant data. In addition, access groups were created that through personal passwords, each user has permission to access what is your responsibility.

**Keywords:** Simulator. MatrikonOPC. Matlab. Simulink. ScadaBR. Natural Gas Processing. Software Integration.

## **LISTA DE ABREVIASÕES**

DDE	Dynamic Data Exchange
ENIAC	Eletronic Numerical Integrator and Computer
GLP	Gás Liquefeito de Petróleo
GPSS	General Purpose Simulation System
GSP	General Simulation Program
LGN	Líquido de Gás Natural
OPC	OLE for Process Control
PID	Proporcional Integral Derivativo
SCADA	Supervisory Control and Data Acquisition
SLAM	Simulation Language for Analogue Modelling
UPGN	Unidade de Processamento de Gás Natural

## LISTA DE FIGURAS

Figura 1 - Diagrama de uma coluna de destilação .....	20
Figura 2- O gás associado .....	21
Figura 3 - O gás não associado .....	22
Figura 4 - Etapas de uma unidade de processamento de gás natural .....	23
Figura 5 - Tela do Watch List.....	27
Figura 6 - Lista das variáveis.....	30
Figura 7 - Tela do diagrama de blocos .....	32
Figura 8 - OPC Toolbox.....	34
Figura 9 - Tela principal do diagrama de blocos .....	35
Figura 10 - Malha de controle de vazão de entrada .....	36
Figura 11 - Malha de controle de vazão de refluxo.....	37
Figura 12 - Malha de controle de nível de fundo da torre .....	38
Figura 13 - Malha de controle de nível do vaso acumulador.....	39
Figura 14 - Malha de controle de pressão do vaso acumulador.....	40
Figura 15 - Malha de controle de temperatura do reervedor.....	41
Figura 16 - Criação de tela no Intouch.....	42
Figura 17 - Inserção dos data points no ScadaBR .....	43
Figura 18 – Tela da torre desbutanizadora .....	44
Figura 19 – Tela do vaso acumulador.....	45
Figura 20 - Tela do reervedor .....	46
Figura 21 - Telas dos controladores PID .....	47
Figura 22 - Criação de grupos de acesso .....	50
Figura 23 - Mensagem de erro.....	51
Figura 24 - Tela de instalação .....	56
Figura 25 - Tela de instalação .....	57
Figura 26 - Tela de opções de configuração.....	57
Figura 27 - Tela de seleção do Java .....	58
Figura 28 - Tela de login .....	58
Figura 29 - Tela do Data Source.....	59
Figura 30 - Propriedade do OPC DA.....	60
Figura 31 - Propriedades do OPC DA.....	61

Figura 32 - Propriedades do OPC DA.....	61
Figura 33 - Tela dos data points .....	62
Figura 34 - Tela dos data points .....	63
Figura 35 - Tela de representação gráfica .....	63
Figura 36 - Tela de representação gráfica .....	64
Figura 37 - Tela de representação gráfica .....	64
Figura 38 - Tela do servidor.....	66
Figura 39 - Tela do servidor.....	67
Figura 40 - Tela de inserção de novas variáveis.....	67
Figura 41 - Tela de seleção do tipo de variável.....	68
Figura 42 - Tela do servidor.....	69
Figura 43 - Tela do servidor.....	69
Figura 44 - Tela do servidor.....	70
Figura 45 - Tela do servidor.....	70
Figura 46 - Tela de mudança de valores das variáveis.....	71
Figura 47 - Tela de acesso ao Public View .....	72
Figura 48 - Tela de representação gráfica a partir do Public View.....	73
Figura 49 - Prompt de comando .....	73
Figura 50 - Configuração do bloco OPC configuration.....	75
Figura 51 - Configuração do bloco OPC configuration.....	76
Figura 52 - Configuração do bloco OPC configuration.....	76
Figura 53 - Configuração do bloco OPC configuration.....	77
Figura 54 - Configuração do bloco OPC write .....	78
Figura 55 - Seleção das variáveis.....	79
Figura 56 - Configuração do bloco OPC read.....	80

# SUMÁRIO

AGRADECIMENTO

RESUMO

ABSTRACT

LISTA DE FIGURAS

LISTA DE ABREVIACÕES

1.	INTRODUÇÃO .....	13
1.1.	APRESENTAÇÃO .....	13
1.2.	JUSTIFICATIVA.....	13
1.3.	OBJETIVOS .....	14
1.3.1.	Objetivo geral .....	14
1.3.2.	Objetivo específico .....	15
1.4.	ESTRUTURA DO TRABALHO.....	15
2.	REVISÃO BIBLIOGRÁFICA.....	17
2.1.	HISTÓRICO DE SIMULADORES.....	17
2.2.	SIMULAÇÃO DE PROCESSOS INDUSTRIAS.....	18
2.3.1.	Processamento de Gás Natural .....	21
2.3.2.	Gás Liquefeito de Petróleo (GLP).....	24
2.3.3.	Torre Debutanizadora.....	24
3.	MATERIAIS E MÉTODOS .....	27
3.1.	SOFTWARE DE CRIAÇÃO DE TELAS DE SUPERVISÃO .....	27
3.2.	SERVIDOR DE APLICAÇÃO JAVA PARA WEB .....	29
3.3.	SERVIDOR DE COMUNICAÇÃO OPC .....	29
3.4.	SOFTWARE DE MODELAGEM E SIMULAÇÃO DE SISTEMAS .....	30
4.	SIMULADOR DESENVOLVIDO.....	34
4.1.	SIMULAÇÃO MATEMÁTICA .....	34
4.2.	TELA DE SUPERVISÃO .....	41
4.3.	INTEGRAÇÃO DOS SOFWARES.....	47
4.4.	ACESSO VIA WEB .....	48
4.5.	CONTROLE DE GRUPO DE ACESSO POR SENHA .....	49
5.	CONSIDERAÇÕES FINAIS .....	52
6.	REFERÊNCIAS.....	54

ANEXO A - TUTORIAL DE INSTALAÇÃO E UTILIZAÇÃO SCADABR .....	56
ANEXO B - TUTORIAL DE UTILIZAÇÃO E DECLARAÇÃO DE VARIÁVEIS NO MatrikonOPC .....	66
ANEXO C – TUTORIAL DE DISPONIBILIZAÇÃO DA REPRESENTAÇÃO GRÁFICA EM REDE LOCAL .....	72
ANEXO D – TUTORIAL DE CONFIGURAÇÃO DOS BLOCOS OPC .....	75

## **1. INTRODUÇÃO**

Esse capítulo apresenta o assunto do trabalho, assim como os objetivos e a justificativa do tema abordado e especifica o que é exposto em cada um dos capítulos.

### **1.1. APRESENTAÇÃO**

Um simulador é uma ferramenta utilizada para que seja possível analisar o comportamento de dispositivos, geralmente caros, e sistemas em condições de operação reais, ou condições mais reais possíveis através de um modelo computacional. Os simuladores são amplamente aplicados para treinamento e qualificação de funcionários, tendo assim, grande importância na área industrial. Não se pode desprezar também o interesse das instituições de ensino na utilização destes simuladores para fins didáticos nas áreas de supervisão, instrumentação, controle, entre outros.

A simulação, segundo Pegden (1995) pode ser entendida como o processo de construção de um modelo representativo de um sistema real, bem como da realização de experiências com esse modelo com o intuito de conhecer melhor o seu comportamento e avaliar o impacto das estratégias alternativas de operação.

De acordo com Freitas Filho (2008), a técnica de modelagem e simulação de sistemas permite criar, em computadores, ambientes virtuais os quais imitam o comportamento de praticamente qualquer tipo de sistema.

Nos últimos anos, a oferta de programas e linguagens voltada à simulação de sistemas, tem sido crescentemente adotada tanto no meio acadêmico quanto no meio empresarial.

Com tantas possibilidades de escolha de ambientes de simulação, a maior dificuldade dos usuários desta ferramenta tem sido emprega-la corretamente no encaminhamento de soluções a seus problemas.

Segundo Banks e Sokolowski (2010), o termo modelagem e simulação referem-se ao desenvolvimento de um modelo, ou seja, é um sistema que será investigado ou estudado utilizando modelagem e simulação. Ao investigar um sistema, uma avaliação quantitativa é necessária para observar como o sistema se comporta com várias entradas e em diferentes ambientes.

## 1.2. JUSTIFICATIVA

O que impulsionou a realização deste trabalho foi o crescente uso de simuladores em atividades industriais em todos os setores, permitindo que se faça uma análise dos sistemas reais sem a necessidade de interferir no mesmo. A execução de modelo computacional através de simuladores permite o planejamento de sistemas futuros, consegue designar quais benefícios a implementação de um novo equipamento poderá alcançar e possibilita a demonstração de situações de emergências sem danificar o sistema real.

As vantagens pelas quais o uso de simuladores vem se intensificando pode ser encontrada em algumas literaturas, citadas abaixo, tais como:

- Podem ser usada para explorar novas políticas de escalonamento dos recursos, procedimentos operativos, regras de decisão, estruturas organizacionais, fluxos de informação, sem ser necessário interromper o normal funcionamento do sistema (Shannon, 1998).
- Permite identificar os pontos de estrangulamento da linha de produção, testar diversas opções com o intuito de conseguir aperfeiçoar o seu

funcionamento, identificando as causas dos atrasos no fluxo de materiais, de informação e de outros processos (Shannon, 1998; Banks, 2000).

- Permite conhecer melhor o sistema e identificar quais as variáveis que realmente influenciam o seu desempenho, servindo de suporte para uma melhor compreensão da realidade para explicar e descrever ou para apoio a decisão (Law, 1991; Ferreira, 1995; Rubinstein, 1998).

### **1.3. OBJETIVOS**

Esta subseção apresenta o objetivo geral e específico que impulsionaram o desenvolvimento deste trabalho.

#### **1.3.1. Objetivo geral**

O objetivo do presente trabalho é desenvolver simuladores industriais com distintas aplicações e funcionalidades. Estes simuladores serão baseados em modelos matemáticos e em sistemas dinâmicos criados no software Matlab/Simulink. A tela de supervisão foi criada no software ScadaBR. O trabalho também explica de forma detalhada os diferentes tipos de comunicação possíveis entre os softwares, indo mais a fundo no tipo de comunicação OPC e no servidor que é utilizado neste trabalho (MatrikonOPC). Além disto, explica o funcionamento dos outros softwares (Matlab/Simulink, ScadaBR), como são feitas as declarações de variáveis, criação dos modelos matemáticos e sistemas e como estabelecer a comunicação entre eles. Problemas e soluções referentes ao desenvolvimento dos simuladores também são abordados de forma minuciosa.

### **1.3.2. Objetivo específico**

Desenvolver um simulador de um processo de desbutanização do gás liquefeito de petróleo (GLP) que possa ser acessado e utilizado em sua plenitude via web, tendo acesso às telas com restrições por senha para definir grupos de acessos. Com isso, possibilita-se que com somente um modelo computacional pronto seja possível ter acesso ao processo simulado em qualquer lugar, a qualquer instante, de forma funcional. Além do mais, é possível que somente o operador responsável por determinada parte do processo tenha acesso à respectiva tela de supervisão via acesso pessoal (senha).

## **1.4. ESTRUTURA DO TRABALHO**

O capítulo 1 aborda as justificativas da escolha do tema e o objetivo geral e específico do trabalho.

O capítulo 2 apresenta uma revisão literária dos assuntos importantes para a criação do simulador, trazendo informações sobre a história dos simuladores, a utilização dos simuladores em processos industriais e o processamento do gás natural.

No capítulo 3 são apresentados os materiais e métodos utilizados para criação do simulador e a metodologia aplicada no desenvolvimento do mesmo.

O capítulo 4 se caracteriza de uma apresentação teórica-prática do simulador e também de como integrar os softwares para que os mesmos tenham uma comunicação capaz de permitir o funcionamento do simulador. Também expõe a utilização de senhas para definição de grupos de acesso.

O capítulo 5 apresenta as considerações feitas a partir do desenvolvimento do simulador. É neste capítulo onde se encontram as recomendações para os próximos trabalhos.

## **2. REVISÃO BIBLIOGRÁFICA**

Este capítulo aborda a revisão bibliográfica, o conceito, as propriedades e os fatores condicionantes do objeto de estudo que, por sua vez, fornece embasamento para o desenvolvimento do simulador, objeto de estudo desta monografia.

### **2.1. HISTÓRICO DE SIMULADORES**

A história da simulação pode ser escrita a partir de muitas perspectivas, por exemplo, a utilização de simulação (análise, treinamento, pesquisa); tipos de modelos de simulação (evento discreto, contínuo, discreto e continuo combinado); linguagens de programação ou ambientes de simulação (GPSS, SLAM, Arena, Simio) e domínios de aplicação (áreas de comunicações, fabricação, militar, transporte).

De acordo com Goldsman, Nance e Wilson (2010), em meados da década de 40, um acontecimento importante definiu o rápido crescimento do campo da simulação, a construção dos primeiros computadores eletrônicos de uso geral como o *Electronic Numerical Integrator and Computer* (ENIAC). Esta crescente disponibilidade de computadores eletrônicos permitiu uma rápida proliferação de técnicas de simulação e aplicações em muitas áreas.

A partir das décadas de 50 e 60, a simulação computacional teve aplicações em grandes corporações como, por exemplo, na indústria aeroespacial dos Estados Unidos da América. Com a modernização dos computadores, softwares e hardwares foram desenvolvidas ferramentas computacionais capazes de serem aplicadas em diversas áreas. A tecnologia disponível era oriunda dos conhecimentos científicos, matemáticos e estatísticos.

Um professor de pesquisa operacional da Universidade de Southampton, Keith Douglas Tocher, desenvolveu o *General Simulation Program* (GSP), o primeiro simulador de uso geral, como ferramenta para a construção sistematicamente de uma planta industrial que compreende um conjunto de máquinas (Goldsman, Nance e Wilson, 2010).

Na década de 70, com o desenvolvimento e uso de pacotes de linguagem, surgiram as linguagens de simulação baseadas em sistemas dinâmicos. Com esse avanço tecnológico, surgem os primeiros microcomputadores com linguagem genérica (Pascal e C) e de simulação (GPSS, SLAM, SIMAN, etc.). Deste modo, a utilização de simuladores passou a ser requerida por um número maior de empresas.

Segundo Goldsman, Nance e Wilson (2010), da década de 70 até o início da década de 80 houve uma grande expansão na área de simulação. Além das inovações técnicas na modelagem e análise de simulação que impulsionaram o crescimento do campo durante este período de expansão, houve desenvolvimentos importantes durante este período na construção da infraestrutura organizacional da comunidade internacional de simulação.

Na década de 80, a simulação começa a ser mais interativa com a introdução do computador e da animação, podendo assim ser realizada antes do início da produção. Os tipos de simulação são projetados para uma modelagem mais rápida e começam a ser restritos para certos tipos de sistemas, pois dispõem de elementos específicos para o uso.

A partir da década de 90, os simuladores possuem estação de trabalho de alto desempenho e grande capacidade de memória melhorando a animação e a facilidade de uso.

## 2.2. SIMULAÇÃO DE PROCESSOS INDUSTRIAS

É muito difícil formar um operador e realizar experiências sobre uma planta real. Existe também uma preocupação do corpo de gestão com a continuidade operacional das unidades de produção no sentido de procurar manter o processo de manufatura em constante emissão de produtos acabados, além de elevados padrões de produtividade e qualidade. Há uma grande necessidade de simuladores equipados com interface que corresponde às reais condições na sala de operação para que tudo que foi dito acima seja atendido.

Assim, como colocam Elshout e Wetherill (1987), é fundamental manter os operadores, pessoas que manuseiam os equipamentos dos processos produtivos, capacitados nas manobras para não permitir uma perda de produção. Além disso, ocorrendo uma parada, esses operadores devem também estar aptos para promover o retorno às condições normais de operação com rapidez por motivos econômicos e, principalmente, obedecendo rigidamente a todos os padrões de segurança.

Em relação ao treinamento dos operadores, Silva e Glória (2012) afirmam que:

Na indústria do petróleo, os operadores são treinados tipicamente em cursos de formação teórica. O conteúdo prático é verificado na própria planta com o acompanhamento de operadores antigos. Esse tipo de treinamento se mostra ineficaz, pois não expõe o operador a situações adversas que podem vir a ocorrer nas instalações dessa indústria. Além disso, a planta permanece estável em grande parte do tempo, não necessitando da intervenção do operador.

Assim como dito por Silva e Gloria (2012), a utilização de simuladores, além de todos os benefícios, aperfeiçoa o treinamento fazendo com que o operador vivencie todos os estágios de operação da planta de forma mais próxima do real quanto possível. Neste trabalho há também a possibilidade de acesso a telas de supervisão via web, facilitando ainda mais, aos operadores, o acesso e o treinamento.

Uma planta simulada em um computador consegue uma redução do tempo de desenvolvimento, custos e riscos em comparação com a planta real, sendo possível também testar vários protótipos de *hardware*, que não são viáveis em um ambiente industrial. Deste modo, as simulações trazem muitas vantagens em relação à vida real,

tais como: permitem aos aprendizes jogarem com o tempo, enquanto a vida real não permite e permite um melhor acesso do instrutor ao aprendiz. As simulações podem ser instrumentadas a ponto de permitir que o instrutor acompanhe as fases de estudo até o momento em que se torna interessante a sua colocação sobre os seus principais fundamentos teóricos.

Quase todos os sistemas podem ser simulados de forma satisfatória, por exemplo, processos de manufatura, aviação, análise de mercado, etc. Estes exemplos mostram que há uma gama de utilidade para simulação, visando ajudar a resolver uma grande variedade de problemas.

Uma refinaria de petróleo dos EUA, Conoco, conforme demonstrado por Winter (1996), foi buscar na experiência da indústria de aviação a prática de uso de simuladores. Seus executivos perceberam que era muito importante promover as condições do mundo real a fim de elevar a experiência de seus operadores de controle. Eles observaram que os usos destes simuladores elevavam a velocidade de resposta de seus operadores de controle para atuar quando os processos apresentavam problemas. Como resultado, a Conoco conseguiu reduzir o tempo entre partida das máquinas e redução dos produtos entre 60% e 70%. Conseguiu, também, reduzir o tempo de preparo de um operador para assumir os controles das plantas de um período entre 12 e 18 meses para 6 meses.

Nas últimas décadas, as indústrias modernas foram obrigadas a operar de forma mais eficiente e econômica para atender não só a concorrência no intenso mercado de hoje, mas também o padrão de qualidade exigido pelo mesmo, encontrando na simulação um forte aliado para tal.

### **2.3. COLUNAS DE DESTILAÇÃO**

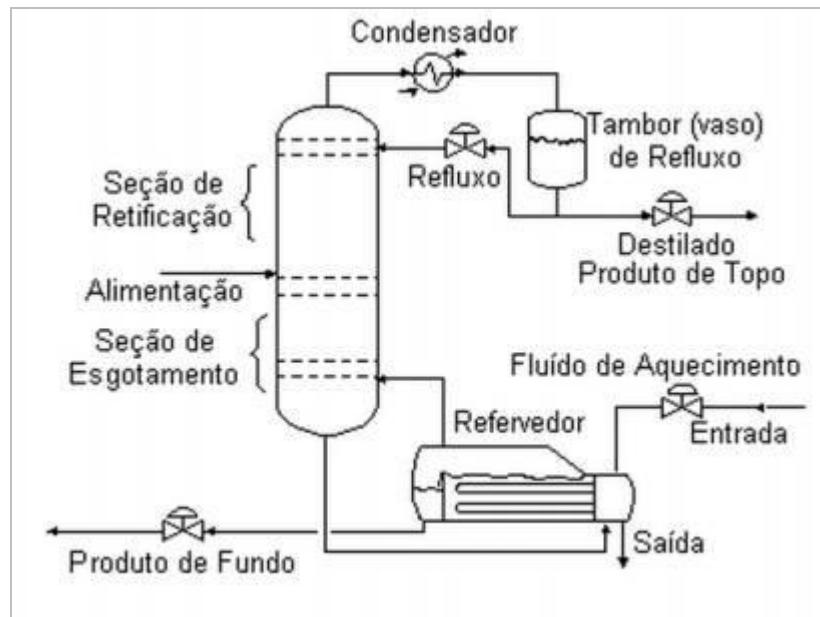
Colunas de destilação são bastante utilizadas na indústria química e de petróleo para separar componentes químicos em produtos purificados. A separação é

obtida através das diferenças na volatilidade (tendência a se vaporizar) entre os vários componentes químicos da mistura.

Em uma coluna de destilação, os componentes mais voláteis (leves ou destilado) são removidos na parte superior da coluna, e os componentes menos voláteis (pesados ou resíduos) são removidos a partir da parte mais inferior da coluna. Na Figura 1 é apresentado um modelo de uma coluna de destilação.

Segundo Lima e Coelho (2014), a coluna de destilação é o equipamento mais importante no processo de refino do petróleo e gás natural, e consequentemente geração de seus derivados, caracterizando-se por ser um elemento de extrema importância na indústria petroquímica. Dessa forma, essa torre é responsável por grande parte dos gastos iniciais e investimentos fixos relativos aos custos operacionais de produção, que são basicamente energéticos. Portanto, a otimização do processo de separação dos componentes da mistura de alimentação é indispensável na melhoria da eficiência da relação custo benefício, permitindo rendimentos econômicos maiores neste tipo de indústria.

Figura 1 - Diagrama de uma coluna de destilação



Fonte: Lima e Coelho, 2014

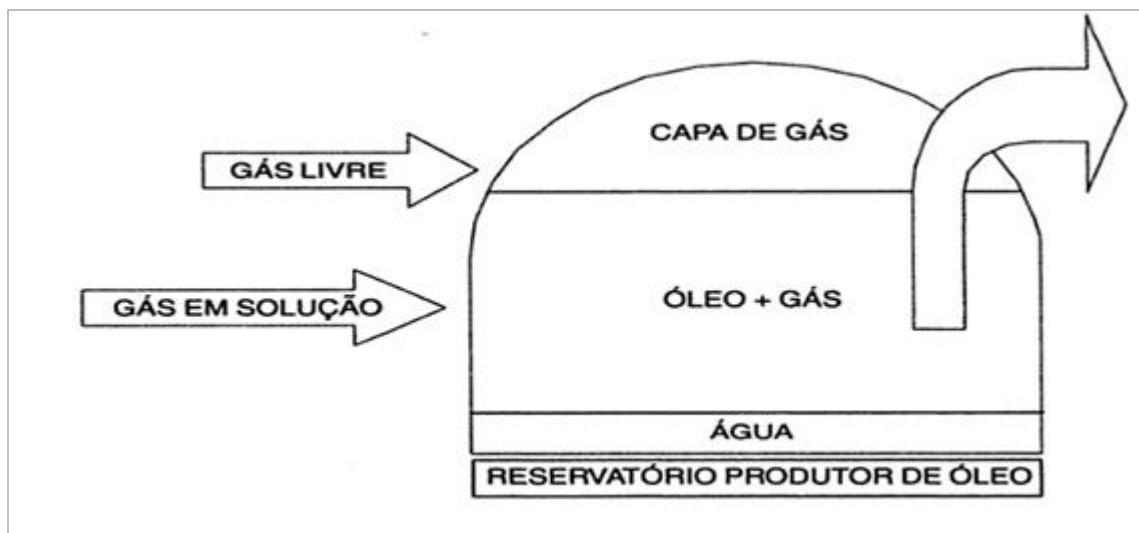
### 2.3.1. Processamento de Gás Natural

O gás natural é composto, principalmente, de metano, etano e propano. Outros hidrocarbonetos mais “pesados” vêm em menores proporções. Segundo Santos (2002), a composição do gás natural pode variar de campo para campo, devido ao tipo de matéria orgânica que lhe deu origem, aos processos naturais a que foi submetido, pelo fato de estar ele associado ou não associado.

O gás associado é assim chamado, pois se encontra misturado com óleo ou de acordo com Santos (2002), pode ser encontrado sobre a forma de capa de gás, isto é, uma parte superior da acumulação rochosa onde a concentração de gás é superior à concentração de outros fluidos como água e óleo. Quando o gás é encontrado nesta forma, normalmente, prioriza-se a produção do óleo utilizando o gás para manter a pressão do reservatório e após isto se inicia a produção de gás.

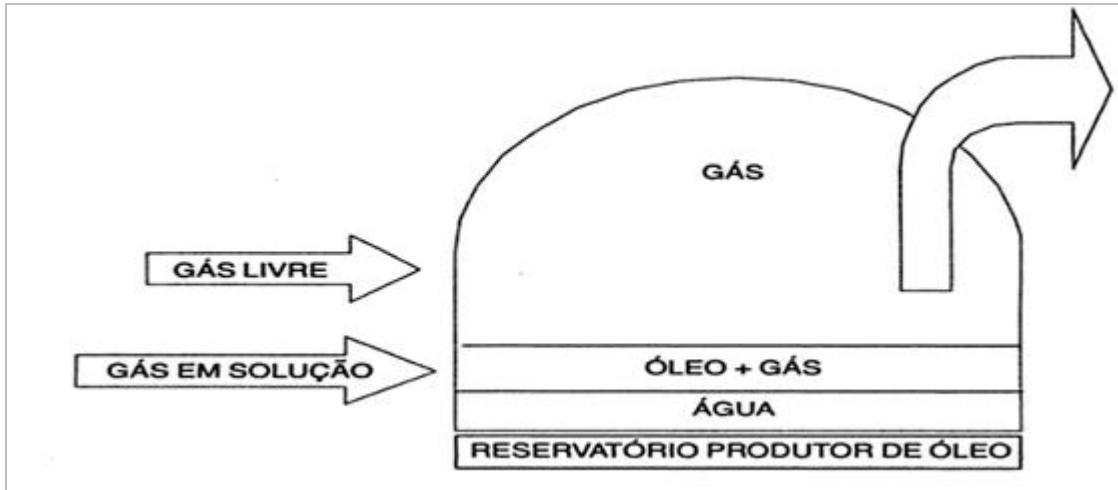
O gás não associado é basicamente o que não está misturado ao óleo e/ou água no reservatório, ou que possui estes misturados, porém em concentrações muito baixas. A diferença entre o gás associado e não associado pode ser vista na Figura 2 e Figura 3.

Figura 2- O gás associado



Fonte: Santos, 2002

Figura 3 - O gás não associado

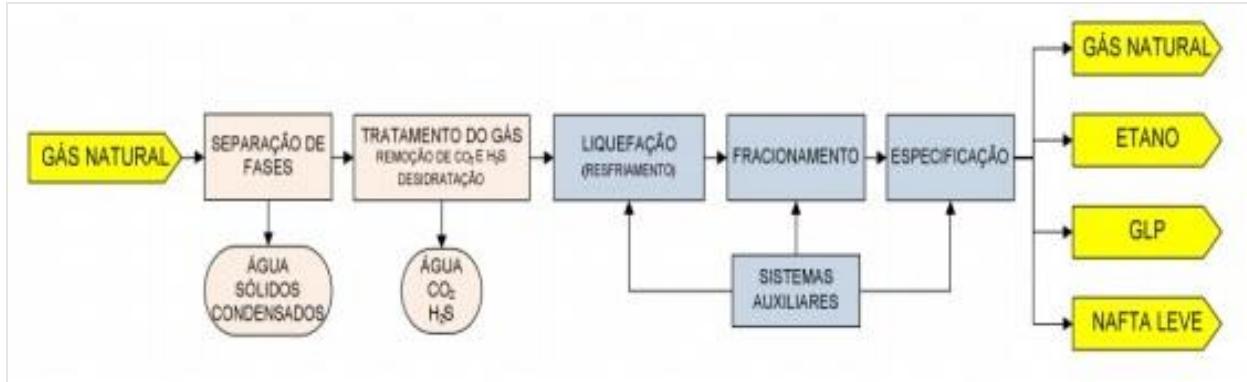


Fonte: Santos, 2002

As reservas de gás no Brasil são em sua maior parte constituídas de gás natural associado, ou seja, gás/óleo/água, juntamente com impurezas (areia). Posteriormente, é necessária a separação dessa mistura, uma vez que a indústria utiliza cada um separadamente para diversas aplicações. Contudo, nosso trabalho se delimitada ao processamento de condensado de gás natural.

O processamento de gás natural é realizado através de uma instalação industrial denominada Unidade de Processamento de Gás Natural (UPGN) composta das seguintes etapas: separação de fases, tratamento do gás e sistemas auxiliares (liquefação, fracionamento e especificação). Este processo pode ser ilustrado como mostra a Figura 4.

Figura 4 - Etapas de uma unidade de processamento de gás natural



Fonte: Cordeiro et al, 2011

O objetivo geral da UPGN é separar as frações pesadas (“ricas”), ou seja, hidrocarbonetos que possuem três ou mais carbonos, das leves (“pobres”), que são compostas por metano e etano.

De acordo com Sant’Anna (2005), para que a separação da mistura trifásica seja realizada, é necessário chamar os campos de produção (sejam eles marítimos ou terrestres) de “facilidades de produção”, que são instalações destinadas a efetuar, sob condições controladas, “o processamento primário de fluidos”, que corresponde à separação do óleo, do gás e da água com as impurezas em suspensão, o tratamento ou condicionamento dos hidrocarbonetos para que possam ser transferidos para as refinarias onde é efetuado o processamento propriamente dito (refino) e o tratamento da água para reinjeção ou descarte.

Após a separação da mistura água/óleo/gás, o gás passa por um tratamento no qual é removido água (desidratação), ácido sulfídrico (H<sub>2</sub>S) e dióxido de carbono (CO<sub>2</sub>). Na próxima etapa, o gás natural passa pelo processo de resfriamento e, com isso, é liquefeito. O GLP passa pelo processo de fracionamento, onde à medida que a temperatura se eleva há a separação dos hidrocarbonetos de forma a atender indústrias, residências, o setor automotivo e de geração de energia.

É importante ressaltar que o processo de desidratação é um dos mais importantes, pois a presença de água no duto permite a formação de hidratos, produzindo depósitos que reduzem o diâmetro da tubulação podendo impedir o fluxo de gás.

### **2.3.2. Gás Liquefeito de Petróleo (GLP)**

Segundo a ANP nº 9 de 21/01/2000, o GLP é uma mistura de hidrocarbonetos com alta pressão de vapor, obtida do gás natural em unidades de processo especiais, que é mantida na fase líquida em condições especiais de armazenamento na superfície.

O GLP tem uma grande aplicabilidade como combustível, devido a sua facilidade de armazenamento e transporte, a partir do engarrafamento em vasilhames. Dessa forma, o gás é mantido sob pressões que variam na faixa de 3 a 15  $kgf/cm^2$ , e nessas condições e à temperatura ambiente o mesmo se apresenta na fase líquida. Daí o nome gás liquefeito de petróleo (Lima e Coelho, 2014).

### **2.3.3. Torre Desbutanizadora**

A coluna de destilação utilizada neste trabalho foi a torre desbutanizadora, que consiste de uma unidade fundamental para o refino do gás natural e produção do GLP. Esta coluna tem como fluxo de entrada o líquido de gás natural (LGN), que é composto por uma mistura de hidrocarbonetos.

O LGN é atribuído à torre desbutanizadora, que é responsável por fazer a transferência de massa por meio da transferência de calor, para separar o GLP dos componentes pesados, também conhecidos como gasolina natural (produto de fundo da torre). Nesse processo os elementos mais leves do LGN são separados, graças a sua menor temperatura de ebulação, ou seja, maior volatilidade, formando uma composição gasosa (GLP), no topo da torre, constituído basicamente de propanos e butanos. E ainda um pouco de outros componentes, como os etanos, pentanos (i-pentanos e n-pentanos). Os componentes mais pesados do LGN são carreados para o fundo da coluna, sendo constituídos essencialmente de pentanos, hexanos, heptanos, octanos, e assim por diante, além de resíduos de butanos não vaporizados (Lima e Coelho, 2015).

De acordo com Campos e Teixeira (2006), a coluna desbutanizadora possui basicamente cinco graus de liberdade, associados as cinco variáveis controladas nesse sistema. Três dessas variáveis estão associadas ao controle de inventário da coluna, relacionado ao balanço de massa do sistema: o controle de pressão da coluna, o controle do nível do fundo da torre e a vazão de entrada. O controle correto dessas variáveis de processo garante uma operação satisfatória do mesmo, evitando que ocorra a inundação dos pratos e vasos de captação dos produtos. Porém, esse projeto foca no controle de uma das outras duas variáveis que interferem na qualidade e especificação do produto final. Essas variáveis são: a temperatura de fundo da torre, que é regulada pela vazão de óleo térmico, e a vazão de refluxo no topo da torre, que tem o objetivo de purificar o produto.

### **3. MATERIAIS E MÉTODOS**

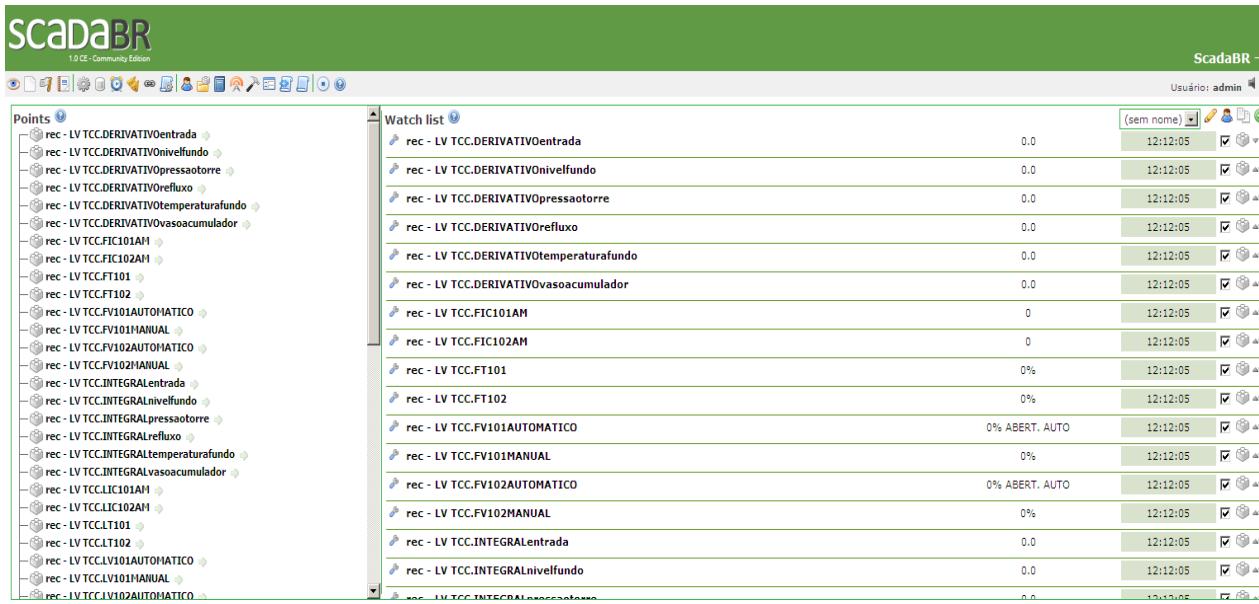
Neste capítulo são apresentados os *softwares* utilizados e a metodologia empregada no desenvolvimento do simulador. Na primeira parte será apresentado o *software* utilizado para a criação das telas de supervisão, o ScadaBR, indicando alguns recursos que foram essenciais para a criação do simulador. Na segunda parte deste capítulo, será apresentado o servidor de aplicação Java para *web*, o Tomcat. Na terceira parte, será apresentado o servidor de comunicação OPC utilizado. Já na quarta parte deste capítulo, será apresentado o *software* de modelagem e simulação de sistemas, o Matlab.

Optou-se pelo *software* ScadaBR devido ao fato de ser um *software* disponível em licença *Open Source* (*software* livre). As telas de supervisão (representação gráfica) construídas neste *software* proporcionarão uma interface mais próxima da realidade visando o mínimo de perda da fidelidade do comportamento do sistema real. O *software* Matlab/Simulink foi escolhido, pois desempenha devidamente as funcionalidades de um simulador. Além disso, possui algumas funções especiais como, por exemplo: mudança de velocidade de simulação, definição de condição de simulação, entre outros. O servidor MatrikonOPC possui licença *Open Source*, é usado para testes de aplicações (cliente-servidor) OPC e será responsável pela comunicação entre o *software* de supervisão e o de simulação matemática. O Tomcat foi utilizado, pois é um servidor *web* gratuito, garantindo assim a aplicação Java para *web*.

### 3.1. SOFTWARE DE CRIAÇÃO DE TELAS DE SUPERVISÃO

O ScadaBR é um software multiplataforma baseado em Java, e como o próprio nome sugere, do tipo SCADA (*Supervisory Control and Data Acquisition*), utilizado para desenvolvimento de aplicações automatizadas. É muito utilizado por possuir licença Open Source (software livre) e por ser um sistema de supervisão completo, possibilitando criação e edição de telas de supervisão (representação gráfica), geração de gráficos e relatórios, controle de acesso e comandos, controlar e ler dispositivos físicos (sensores, atuadores, chaves, motores, etc.), criação de alarmes, entre outros.

Figura 5 - Tela do *Watch List*



Fonte: ScadaBR

Para a melhor utilização do *software* é de extrema importância conhecer seus recursos. Os mais importantes utilizados neste trabalho são:

- *Data sources* e *Data points* – são as “fontes” de informações. No caso específico do nosso trabalho, este *data source* (basicamente é o “lugar” de onde os dados são recebidos) irá obter as variáveis (*data points*) que foram

previamente salvas no MatrikonOPC tendo que ser posteriormente habilitadas para que possam ser utilizadas no ScadaBR.

- Representações Gráficas – é onde são criadas as telas de supervisão propriamente ditas. Comparando o ScadaBR com outros softwares de supervisão, a parte de representações gráficas deixa um pouco a desejar pois não tem muitos recursos. Mas, nesta parte podemos incluir monitoramento das variáveis, fazer *upload* de imagem de fundo (para melhorar a ergonomia e tornar as telas mais intuitivas possíveis), adicionar botões, gráficos, luzes de alarme e etc.
- *Watch List* – como o nome sugere, é onde confere e/ou muda os valores das variáveis. Nela estão listadas todas as variáveis que estão sendo utilizadas no processo (viabilizadas pelo *data source* vindo do MatrikonOPC) como pode ser visto na Figura 5.
- Importar/Exportar – esta ferramenta permite exportar e importar *data points*, *data sources*, representações gráficas, permitindo assim fazer um *backup* do sistema. Quando é realizada a exportação de um trabalho, o próprio programa, no caso, o ScadaBR, gera um arquivo em formato .ZIP, podendo este ser utilizado em outras versões do ScadaBR.
- Usuários – este ícone permite a criação, edição e exclusão de usuários com senha. Também permite limitar ou conceder acesso às variáveis, telas de supervisão, etc.

Após a configuração dos protocolos de comunicação e definição das variáveis, o ScadaBR permite também montar interfaces de operador *web* utilizando o próprio navegador. O tutorial de instalação e utilização do ScadaBR segue no “ANEXO A” deste trabalho.

### **3.2. SERVIDOR DE APLICAÇÃO JAVA PARA WEB**

O Tomcat é um servidor de aplicações Java para *web*. É distribuído como software livre, seja para fins comerciais ou não, e desenvolvido com código aberto dentro do conceituado projeto Apache.

Sua principal característica técnica é estar centrada na linguagem de programação Java. O Tomcat está escrito em Java e, por isso, necessita que uma versão do Java esteja instalada no mesmo computador onde ele será executado. No entanto, não basta ter a versão *runtime* do Java instalado, pois o Tomcat necessita compilar (e não apenas executar) programas escritos em Java.

### **3.3. SERVIDOR DE COMUNICAÇÃO OPC**

O MatrikonOPC é um servidor de comunicação gratuito usado para testes de ajuda e aplicações OPC de diagnóstico (clientes) e conexões. Ele é o responsável pela comunicação e troca de dados entre o ScadaBR e o Matlab.

A Figura 6 exemplifica uma lista de variáveis criada através do MatrikonOPC.

Figura 6 - Lista das variáveis

The screenshot shows the MatrikonOPC Server for Simulation and Testing application window. The title bar reads "MatrikonOPC Server for Simulation and Testing - LV TCC.xml". The menu bar includes File, Edit, View, Tools, and Help. The toolbar contains icons for New, Open, Save, Print, and Exit. The left pane displays the "Current configuration" tree, which includes "Server Configuration", "Alias Configuration", and "LV TCC". The right pane shows a table titled "Contents of alias group 'LV TCC'". The table has columns: Name, Item Path, Data Type, R/W, and Update Rate. The table lists various variables such as DERIVATIVO, FIC101AM, FT101, FV101AUTOM, INTEGRAL, and LIC101AM, each with its corresponding item path, data type (REAL8 or BOOLEAN), read/write status (R/W), and update rate (10). At the bottom of the table is a "Reset Statistics" button. The footer of the window shows the Matrikon logo, the number of clients (65), and the server time (23/9/2015 09:15:00).

Contents of alias group 'LV TCC':					
	Name	Item Path	Data Type	R/W	Update Rate
	DERIVATIVO	Bucket Brigade.Real31	REAL8	R/W	10
	DERIVATIVO	Bucket Brigade.Real35	REAL8	R/W	10
	DERIVATIVO	Bucket Brigade.Real39	REAL8	R/W	10
	DERIVATIVO	Bucket Brigade.Real42	REAL8	R/W	10
	DERIVATIVO	Bucket Brigade.Real45	REAL8	R/W	10
	DERIVATIVO	Bucket Brigade.Real62	REAL8	R/W	10
	FIC101AM	Bucket Brigade.Boolean1	BOOLEAN	R/W	10
	FIC102AM	Bucket Brigade.Boolean17	BOOLEAN	R/W	10
	FT101	Bucket Brigade.Real1	REAL8	R/W	10
	FT102	Bucket Brigade.Real11	REAL8	R/W	10
	FV101AUTOM	Bucket Brigade.Real15	REAL8	R/W	10
	FV101MANUA	Bucket Brigade.Real18	REAL8	R/W	10
	FV102AUTOM	Bucket Brigade.Real16	REAL8	R/W	10
	FV102MANUA	Bucket Brigade.Real19	REAL8	R/W	10
	INTEGRAL	Bucket Brigade.Real30	REAL8	R/W	10
	INTEGRAL	Bucket Brigade.Real34	REAL8	R/W	10
	INTEGRAL	Bucket Brigade.Real38	REAL8	R/W	10
	INTEGRAL	Bucket Brigade.Real41	REAL8	R/W	10
	INTEGRAL	Bucket Brigade.Real44	REAL8	R/W	10
	INTEGRAL	Bucket Brigade.Real61	REAL8	R/W	10
	LIC101AM	Bucket Brigade.Boolean6	BOOLEAN	R/W	10
	LIC102AM	Bucket Brigade.Boolean16	BOOLEAN	R/W	10

Fonte: MatrikonOPC

OPC é a sigla para “OLE for process Control”, onde OLE é “Object Linking and Embedding”. Os equipamentos e/ou softwares dotados de comunicação via OPC disponibilizam dados internos em uma interface simplificada, onde aplicações externas podem interagir com a leitura e/ou escrita de valores em parâmetros, registradores de programas, resultados, etc.

Aplicações de teste “ao vivo” de servidores OPC podem resultar em perda de dados de produção atuais. O servidor MatrikonOPC cria um ambiente simulado que mesmo diante de problema, nenhum dado do processo real é perdido.

O tutorial de instalação e utilização do MatrikonOPC segue no “ANEXO B” deste trabalho.

### 3.4. SOFTWARE DE MODELAGEM E SIMULAÇÃO DE SISTEMAS

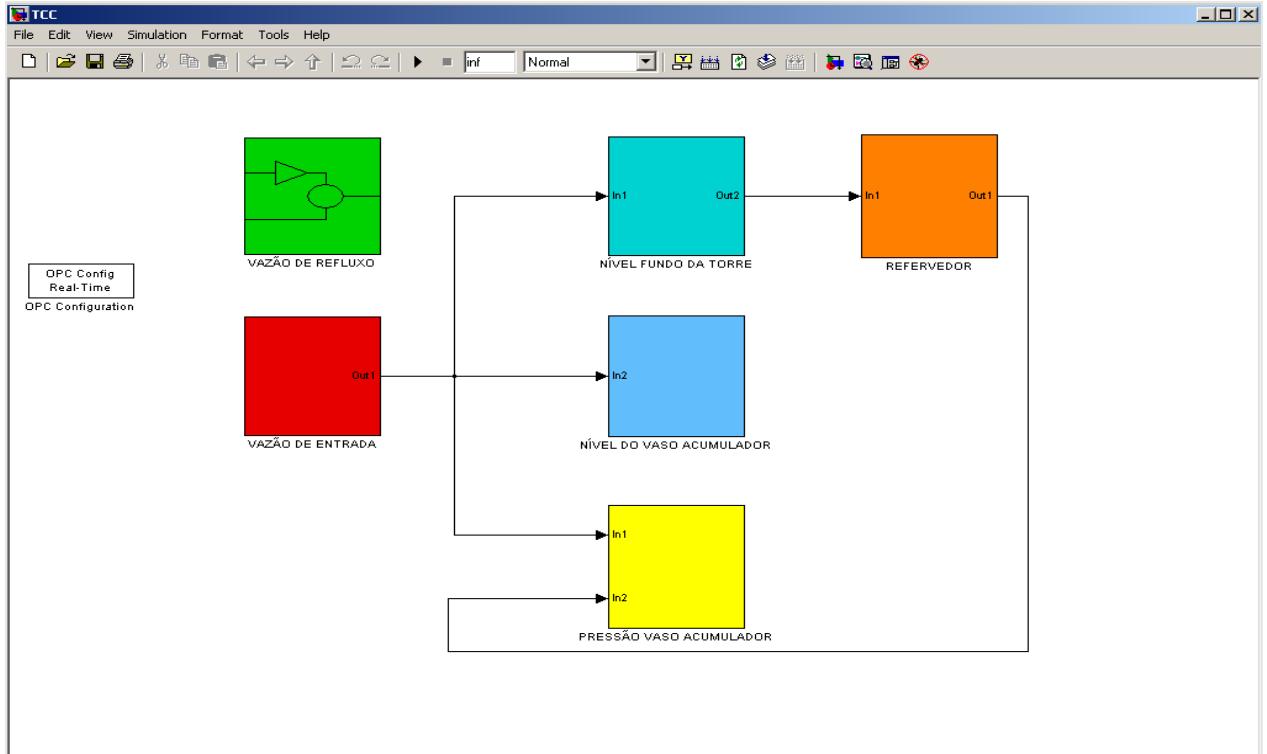
O Matlab é um software de alto desempenho dedicado a cálculos numéricos, análises numéricas, processamento de sinais, construção de gráficos, etc.

O Simulink é um pacote do Matlab e é utilizado para analisar sistemas dinâmicos através de modelagem e simulação. É também uma ferramenta extremamente poderosa, suportando sistemas lineares ou não, contínuos ou discretos e até mesmo a mistura dos dois. O Simulink utiliza uma interface gráfica de alto nível e bem intuitiva através de diagramas de blocos para a construção dos modelos. Ele já possui bibliotecas com blocos pré-definidos podendo outros novos serem adicionados. Existe também a possibilidade da criação de blocos a partir da necessidade do usuário. Após o desenvolvimento do modelo matemático é possível simulá-lo e analisar seu funcionamento através de gráficos, blocos *display*, entre outros.

Além disso, pelo Simulink ser integrado ao Matlab, é possível utilizar outras funcionalidades do Matlab para analisar os dados. Através de um bloco específico é possível mandar valores para o *workspace* do Matlab para visualização e/ou posterior utilização ou análise.

O Simulink é utilizado neste trabalho como ferramenta de simulação matemática onde toda a dinâmica do processo foi feita em estrutura de blocos, como mostrado na Figura 7. Através de um bloco “OPC Configuration” as informações serão lidas e/ou escritas no MatrikonOPC que será a “ponte” do Simulink com o ScadaBR.

Figura 7 - Tela do diagrama de blocos



Fonte: Lima e Coelho, 2014

Através do Simulink desenvolveu-se o simulador industrial o mais próximo do real possível. Com a utilização do ScadaBR desenvolveu-se uma tela de supervisão funcional, ergonômica e de padrão utilizado em industrias do ramo. E com a utilização do MatrikonOPC tem-se a integração entre os softwares criando-se assim um simulador industrial. Após esta etapa, através do ScadaBR colocamos o simulador na web para assim, termos o acesso remoto a planta industrial simulada.

No “ANEXO D” encontra-se o tutorial de configuração dos blocos OPC.

## **4. SIMULADOR DESENVOLVIDO**

A simulação computacional traz consigo grandes benefícios, além de ser uma excelente ferramenta de estudo. Assim como coloca Lima e Coelho (2014), uma vez que o sistema simulado representa o sistema real com uma fidelidade adequada, torna-se possível realizar testes e verificar o comportamento das diversas variáveis do processo.

A planta simulada foi criada com base no trabalho de conclusão de curso de Lima e Coelho (2014). Algumas modificações foram feitas, pois se mostraram necessárias para que todas as funcionalidades propostas fossem alcançadas. Para a criação da planta foram criadas funções de transferências (funções matemáticas), no Matlab, que representam o comportamento da planta real. E através do servidor de comunicação MatrikonOPC, o Matlab envia e recebe dados do software de supervisão ScadaBR. Toda esta integração de softwares possibilita que se chegue aos objetivos descritos neste trabalho de forma plena.

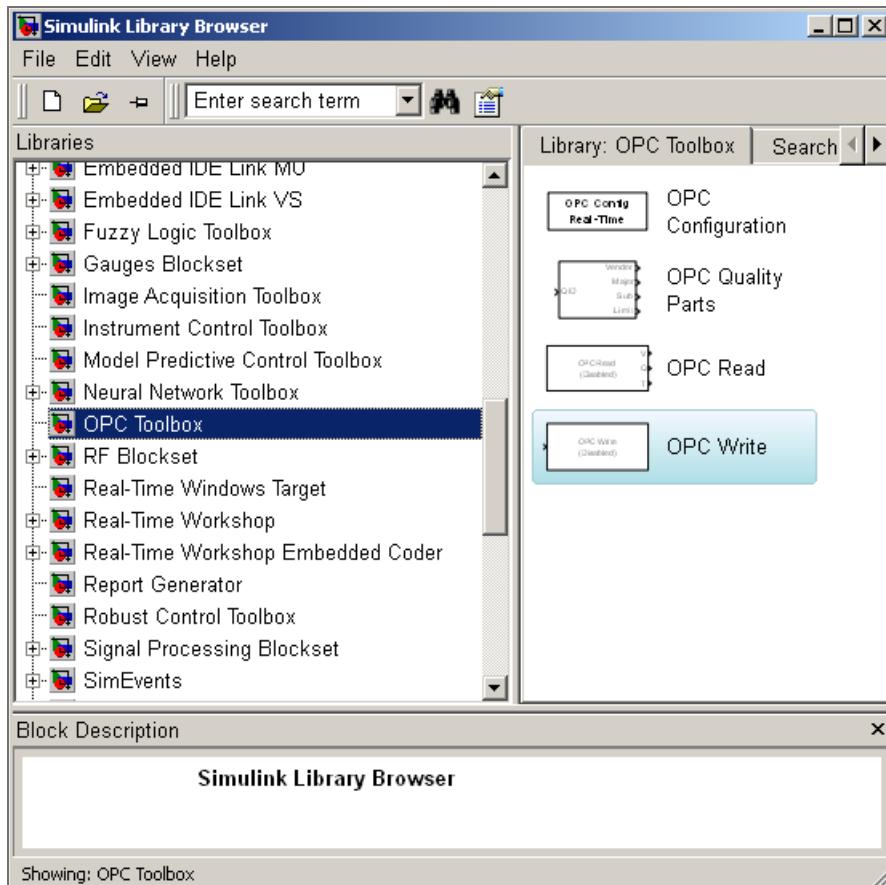
### **4.1. SIMULAÇÃO MATEMÁTICA**

A parte da implementação computacional para a criação do simulador do processo de desbutanização do gás de petróleo liquefeito (GLP) foi criada no Simulink/Matlab. O protocolo de comunicação utilizado foi o OPC e não o DDE

(*dynamic data exchange*). Para a comunicação DDE seria necessário criar uma função específica, Matlab *functions*, para cada entrada e saída do processo. Para a utilização do protocolo OPC foi necessário adicionar uma biblioteca OPC Toolbox no Simulink, ilustrada na

Figura 8. Com essa biblioteca é possível adicionar entradas e saídas através de blocos.

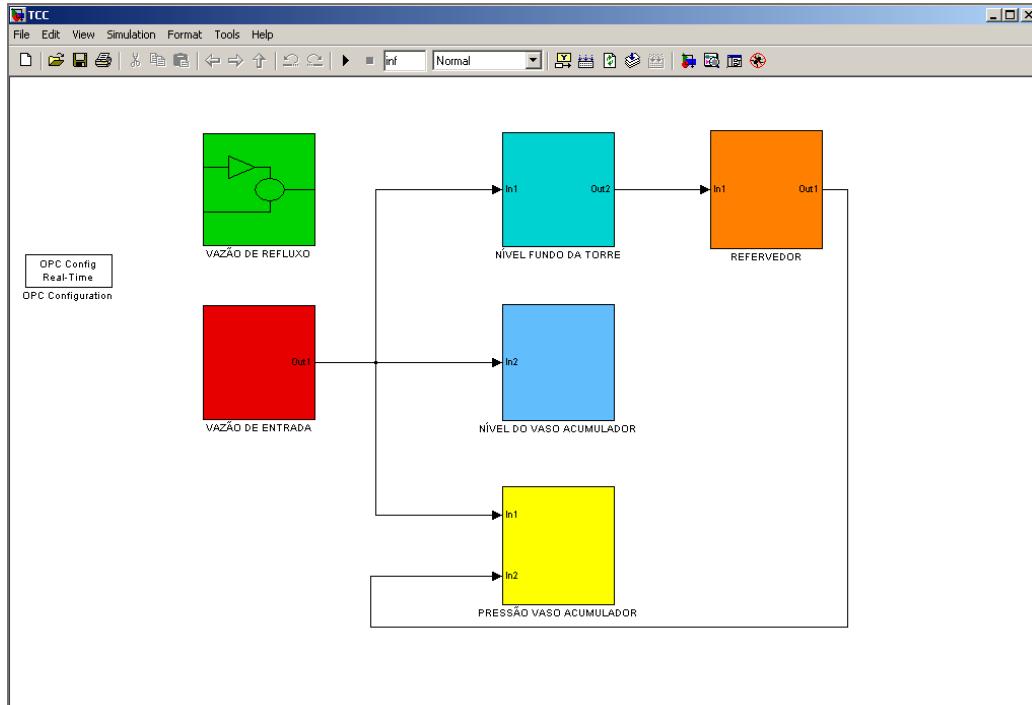
Figura 8 - OPC Toolbox



Fonte: Simulink

Tendo o OPC Toolbox sido instalado no Simulink, o próximo passo foi criar o diagrama de blocos no Simulink. A Figura 9 mostra a tela principal do diagrama de blocos criado, onde dentro estão as implementações matemáticas.

Figura 9 - Tela principal do diagrama de blocos



Fonte: Lima e Coelho, 2014

A Figura 10 corresponde ao controle de vazão de entrada. A estratégia de controle utilizada foi *feedback* negativa, também chamado de realimentação negativa. Este sistema tende a manter um estado de equilíbrio, que é garantido pelo laço de realimentação, onde utilizando um controlador proporcional integral derivativo (PID), mantem-se o valor de uma variável controlada (válvula de controle) dentro dos limites pré-ajustados (*setpoint*).

A saída do bloco de vazão de entrada influencia diretamente nas entradas dos blocos de nível de fundo da torre, nível do vaso acumulador e pressão do vaso acumulador.

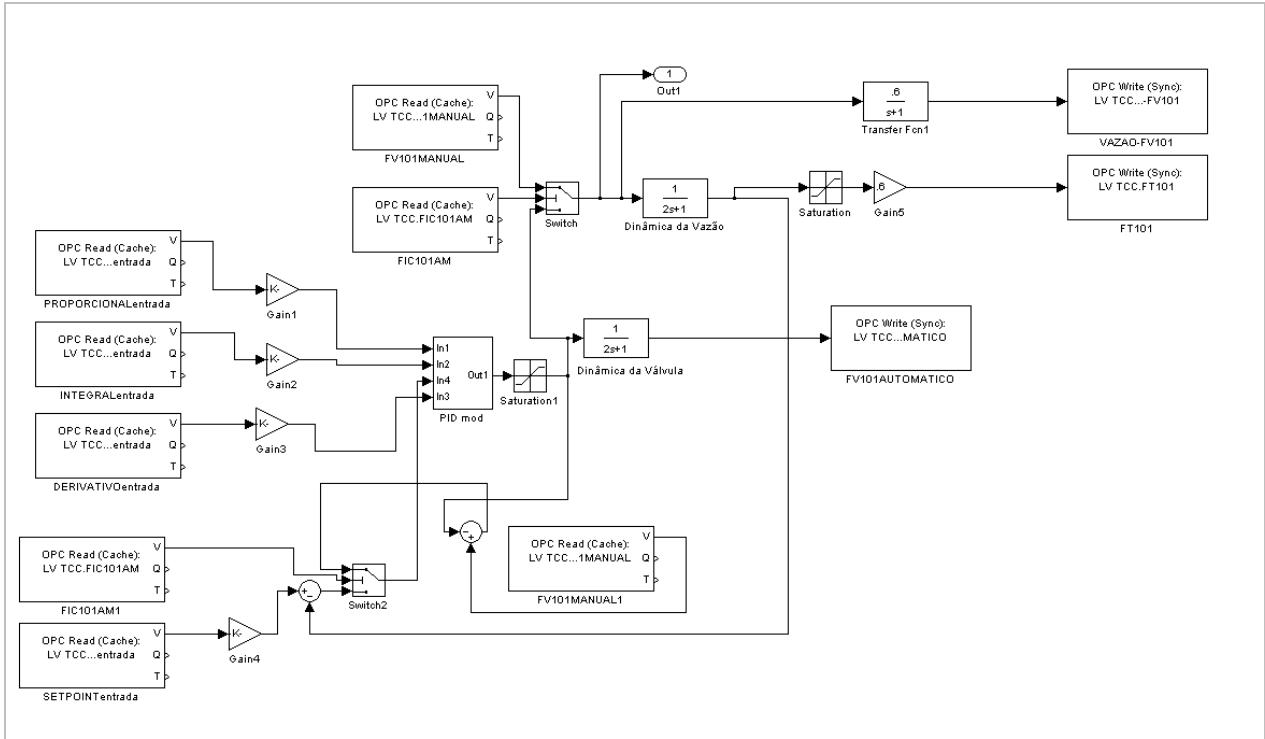
Os blocos OPC Read foram colocados para receber valores que são pré-estabelecidos no MatrikonOPC e podem ser futuramente modificados no ScadaBR. Já os blocos OPC Write enviam os valores para o MatrikonOPC e consequentemente para o supervisório, ScadaBR.

Os ganhos foram utilizados para alcançar valores desejados, por exemplo, o ganho de 0.6 foi adicionado juntamente com outros ganhos para manter a vazão em 60

$\text{m}^3/\text{h}$  quando a abertura da válvula for de 100%, fazendo com que essa proporcionalidade seja mantida. Também foram utilizados dois *switches* para que o sistema possa alternar de automático para manual e vice-versa.

Para representar o comportamento do processo durante variações na entrada, implementamos as funções de transferência. O controle PID empregado neste sistema foi modificado para que seus parâmetros pudessem ser alterados pelo MatrikonOPC ou pelo software de supervisão.

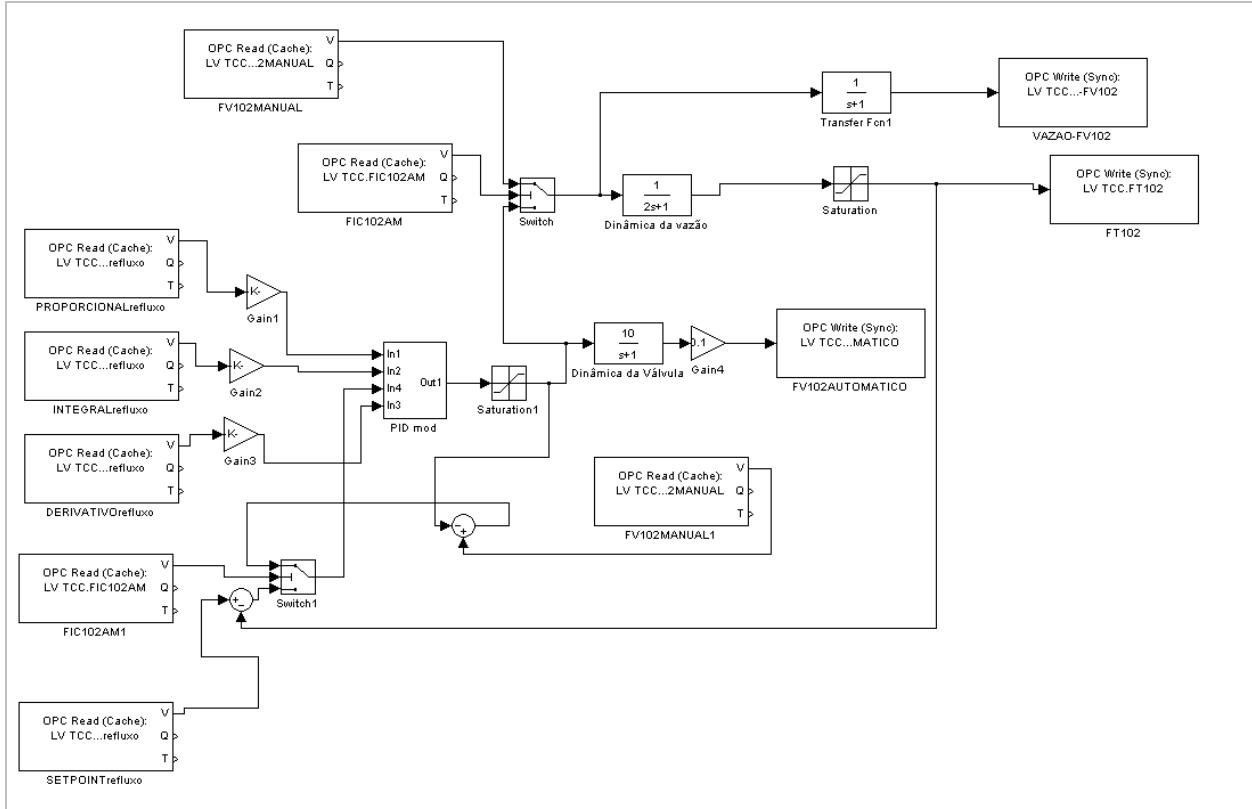
Figura 10 - Malha de controle de vazão de entrada



Fonte: Adaptado de Lima e Coelho, 2014

Na Figura 11 temos o controle da vazão de refluxo, que se comporta de forma semelhante à vazão de entrada. Com isso, foi utilizada a mesma estratégia de controle (*feedback negativo*), onde o controlador recebe parâmetros similares (proporcional, derivativo e integral).

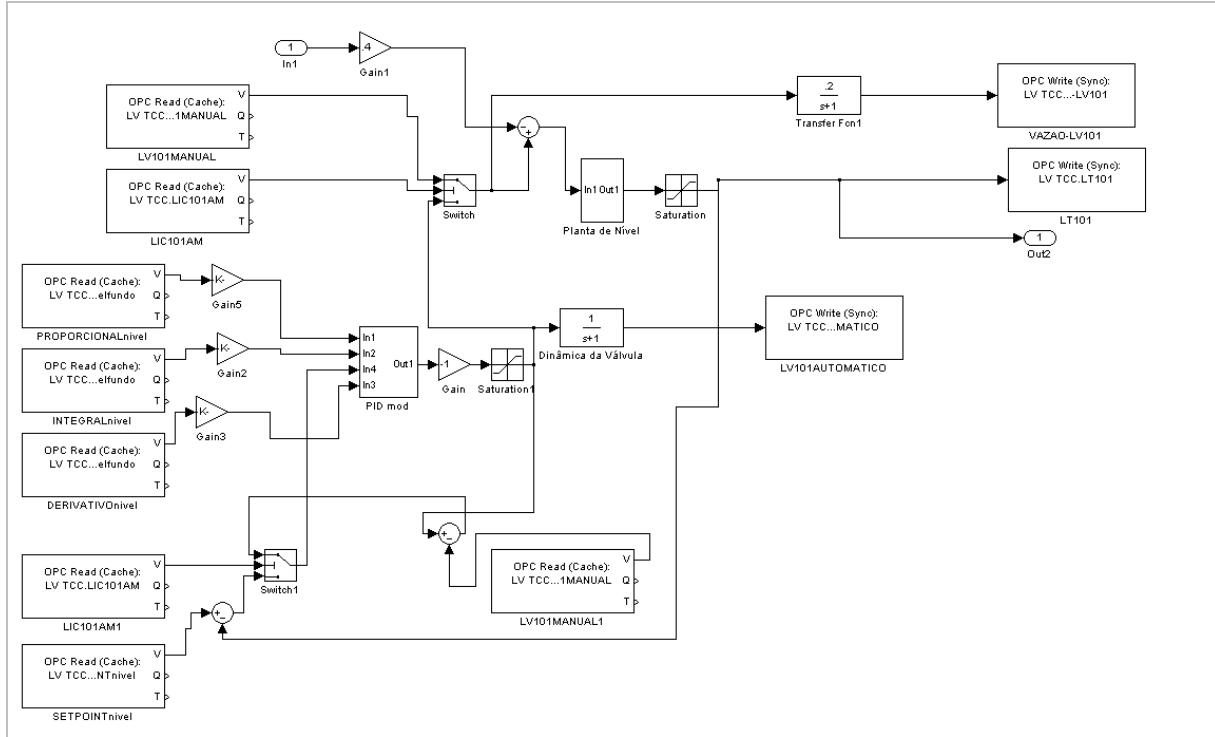
Figura 11 - Malha de controle de vazão de refluxo



Fonte: Adaptado de Lima e Coelho, 2014

A Figura 12 apresenta uma estratégia de controle *feedback* de nível, onde a válvula de controle possui ação reversa.

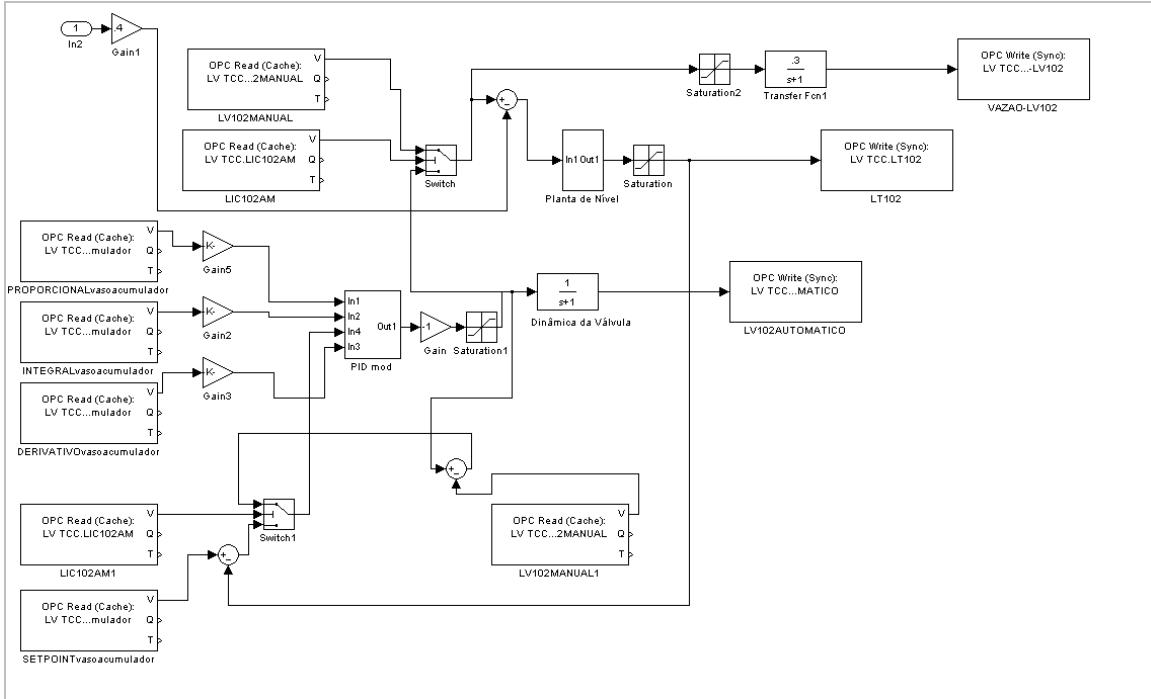
Figura 12 - Malha de controle de nível de fundo da torre



Fonte: Adaptado de Lima e Coelho, 2014

A Figura 13 representa o controle de nível do vaso acumulador que, assim como a Figura 12, possui uma estratégia *feedback* com ação reversa na válvula de controle.

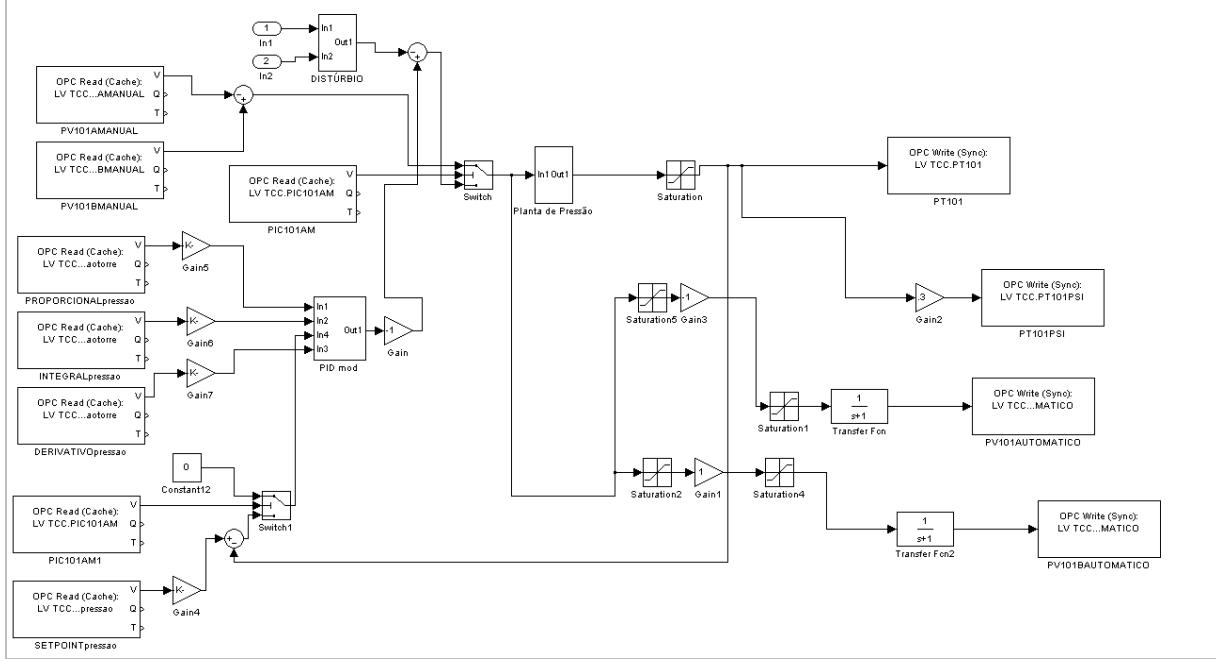
Figura 13 - Malha de controle de nível do vaso acumulador



Fonte: Adaptado de Lima e Coelho, 2014

A Figura 14 corresponde ao controle de pressão do vaso acumulador. Diferente de todas as estratégias utilizadas no simulador, neste bloco utilizou-se a estratégia de controle *Split-Range*, na qual possui a sua faixa dividida, que consiste em um sinal controlando dois elementos finais de controle. O objetivo desta malha é controlar a pressão interna do vaso acumulador de tal forma que, se a pressão interna aumenta, a válvula PV-101A começa a fechar diminuindo o fluxo de gás. Se o fechamento da válvula PV-101A não for suficiente para diminuir a pressão do vaso, a válvula PV-101B começa a abrir liberando gás, consequentemente, aliviando a pressão interna do vaso.

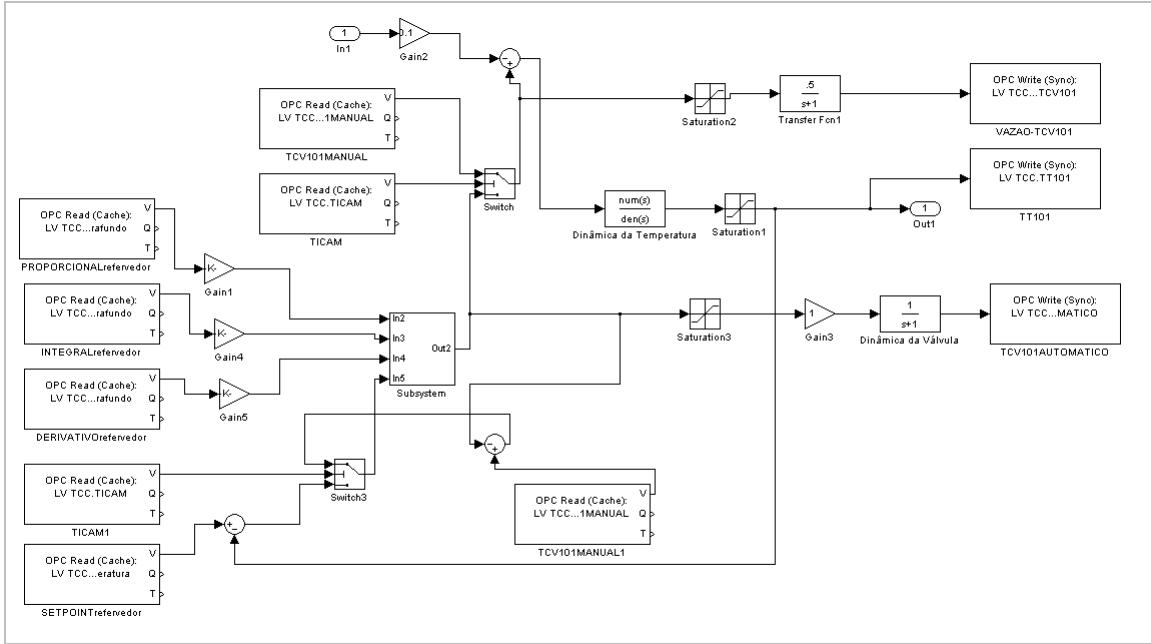
Figura 14 - Malha de controle de pressão do vaso acumulador



Fonte: Adaptado de Lima e Coelho, 2014

Na Figura 15 também se utilizou a estratégia de controle *feedback*, porém com uma peculiaridade. Por esta malha ser de temperatura, considerada uma variável “lenta”, houve a necessidade de receber um retardo de transporte para que o simulador apresentasse uma dinâmica mais próxima do real.

Figura 15 - Malha de controle de temperatura do referededor



Fonte: Adaptado de Lima e Coelho, 2014

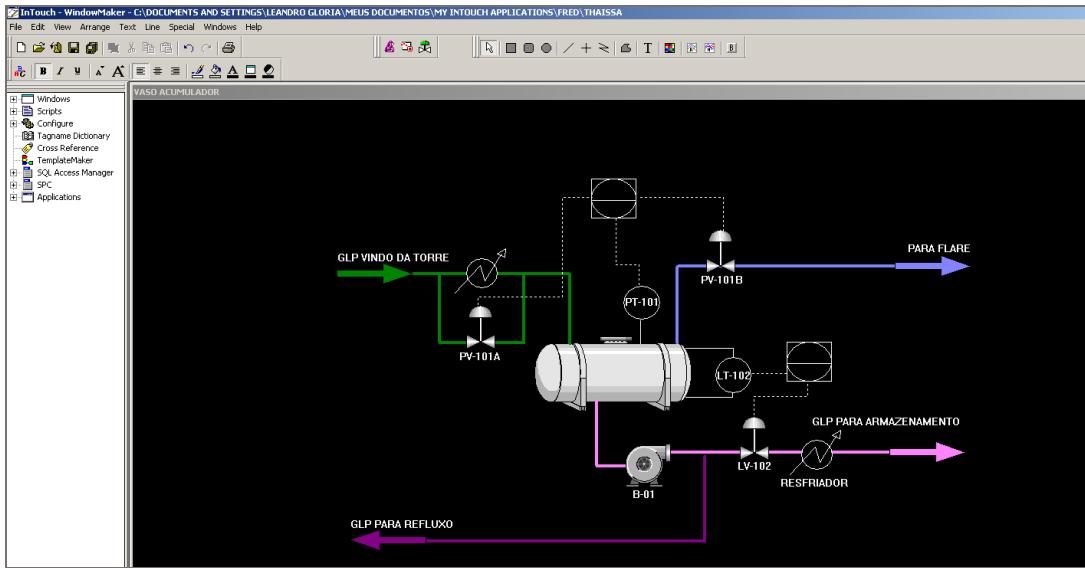
É importante ressaltar que em todas as malhas de controle foram utilizados ganhos, com valor de 0.01, após os blocos de leitura dos parâmetros dos controladores. Utilizamos estes ganhos, pois era necessário que os valores dos parâmetros tivessem casas decimais para um ajuste “mais fino” do controlador.

## 4.2. TELA DE SUPERVISÃO

No ScadaBR foram criadas telas para que a visualização do funcionamento do sistema fosse possível. Também foram adicionadas funcionalidades às telas de supervisão como: acionamentos, alerta de alarmes, medidores de vazão, nível e etc. Desta maneira, o usuário pode interagir com o processo de forma que o mesmo possa realizar mudança de *setpoint*, parâmetros de controle, entre outros de acordo com o que necessita para obter o resultado do processo.

Para a criação das telas no ScadaBR, de forma que elas fiquem o mais próximo das telas utilizadas no dia a dia dos operadores do processo, as telas foram criadas em outro *software* e somente a imagem foi importada para o ScadaBR, como podemos ver na Figura 16.

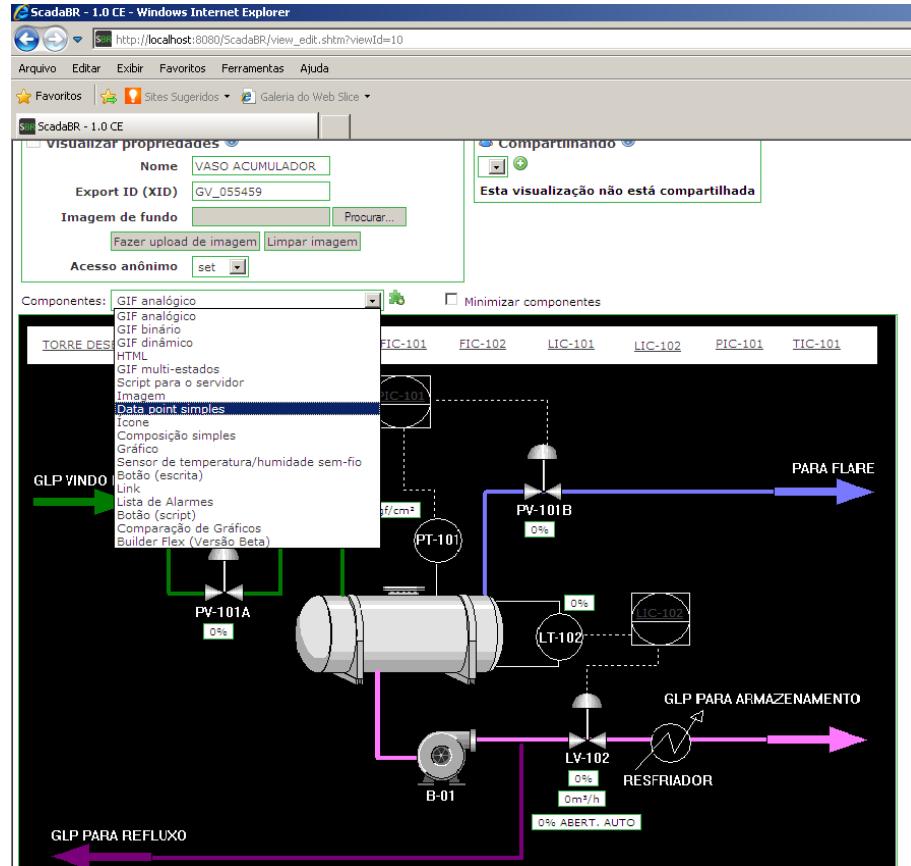
Figura 16 - Criação de tela no Intouch



Fonte: Os autores

Após colocar a imagem no plano de fundo, os *data points* são inseridos e configurados de acordo com a sua utilização, isso pode ser visto na Figura 17.

Figura 17 - Inserção dos *data points* no ScadaBR



Fonte: Os autores

Depois das telas serem criadas, para serem utilizadas, deve-se clicar no botão salvar, abaixo da representação gráfica. Cada tela possui um endereço diferente. Estes endereços, com suas devidas modificações, são utilizados posteriormente pelo usuário da web para ter acesso ao simulador remotamente.

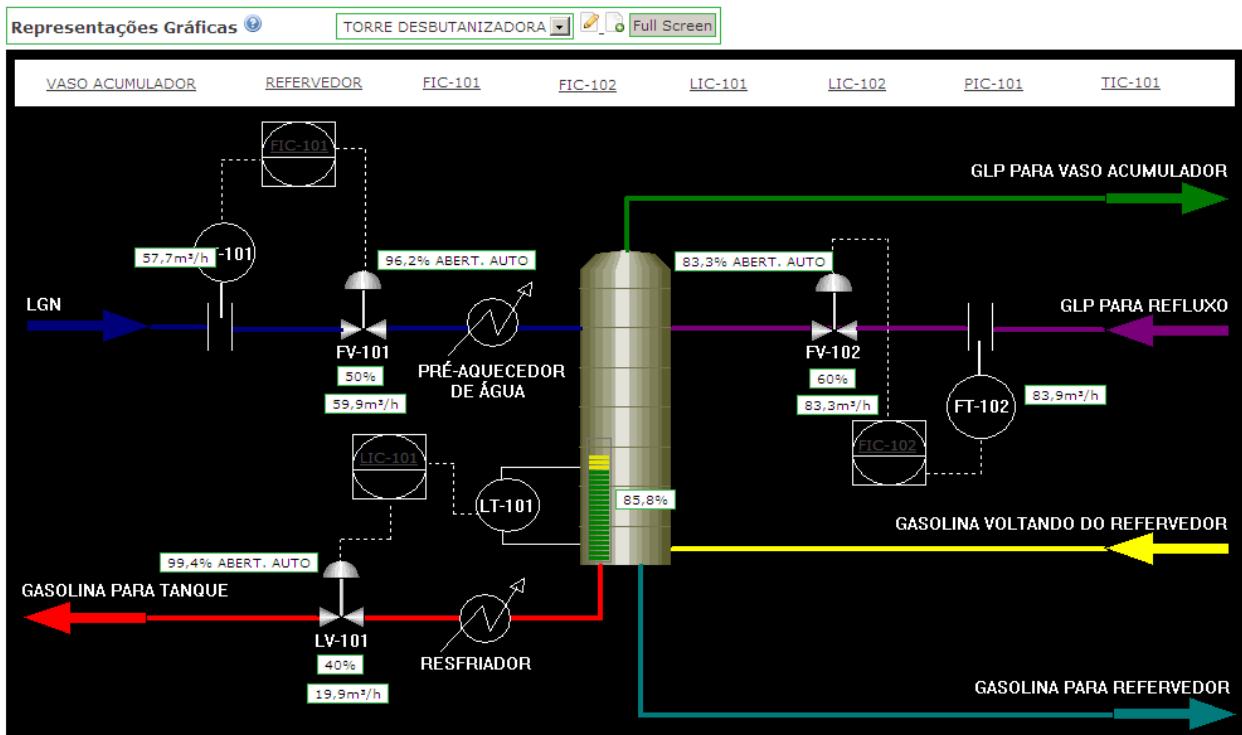
A Figura 18 mostra a tela principal referente à torre desbutanizadora. Nela temos três malhas de controle referentes às seguintes variáveis de processo: vazão de entrada, vazão de refluxo e nível de fundo.

Na malha de entrada de LGN, a vazão é controlada através do controlador FIC-101, que atua sobre a válvula FV-101. Ao clicar sobre o ícone do controlador, o operador poderá modificar os parâmetros proporcional, integral e derivativo, e também será possível alterar o valor do *setpoint*.

A malha de refluxo consiste da entrada do GLP vindo do vaso acumulador, que ajuda a manter o gradiente de temperatura da torre e melhorar a pureza do GLP. O controle é realizado através do controlador FIC-102 que atua na válvula FV-102. Assim como na malha de entrada, os parâmetros do controlador e o *setpoint* podem ser modificados pelo operador.

A outra malha presente na torre desbutanizadora é a de nível de fundo. Esta controla o nível de gasolina que é proveniente do referedor. Quando este nível está acima do desejado, o controlador envia um sinal para a válvula LV-101 fazendo a abrir e diminuindo assim o nível de fundo da torre.

Figura 18 – Tela da torre desbutanizadora

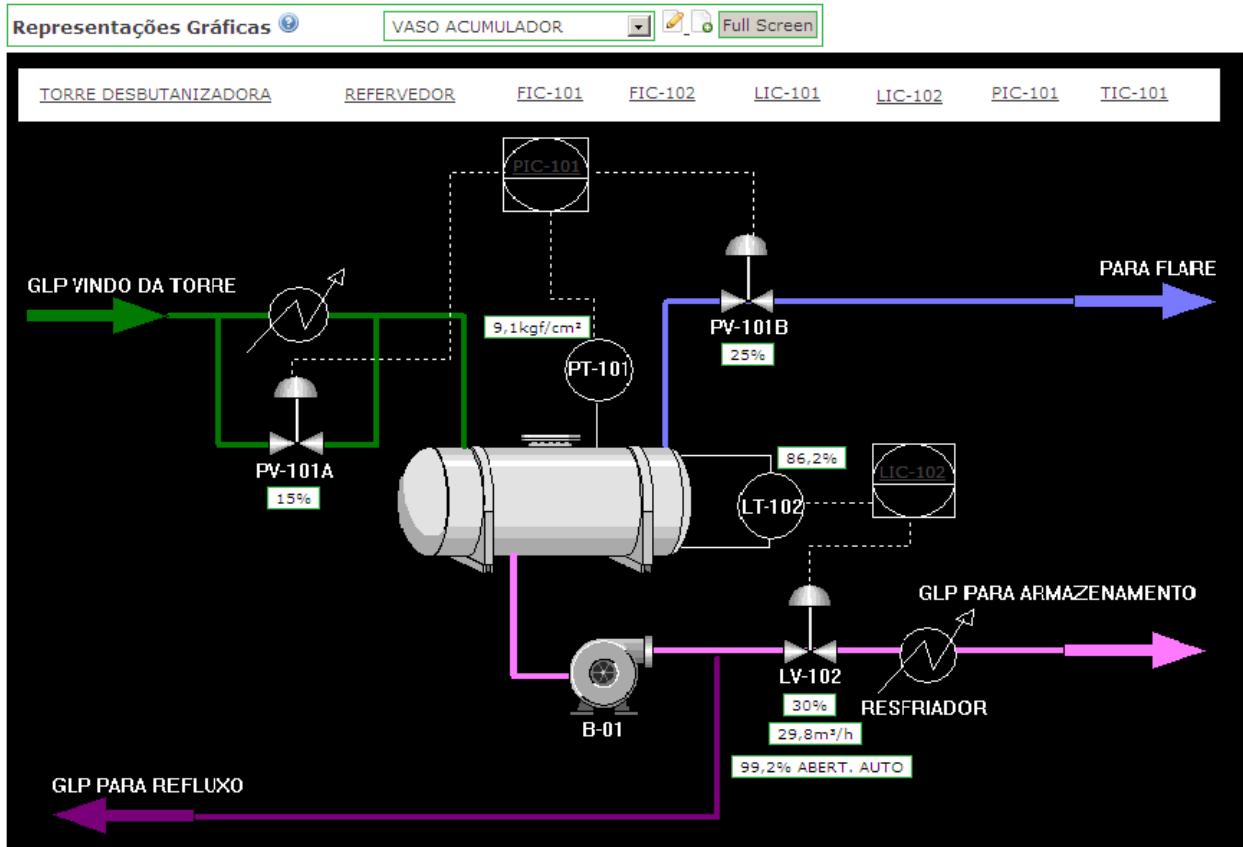


Fonte: Os autores

A Figura 19 se refere à tela do vaso acumulador, que possui duas malhas de controle: pressão e nível. A malha de nível do vaso controla o nível de GLP vindo da torre.

Na malha de pressão, diferentemente das outras malhas, utiliza-se um controlador para controlar duas válvulas: PV-101A e PV-101B. Com isso, a pressão do vaso é controlada com apenas um sinal. Para diminuir a pressão, a válvula PV-101B direciona o GLP excedente para o *flare*, onde será queimado.

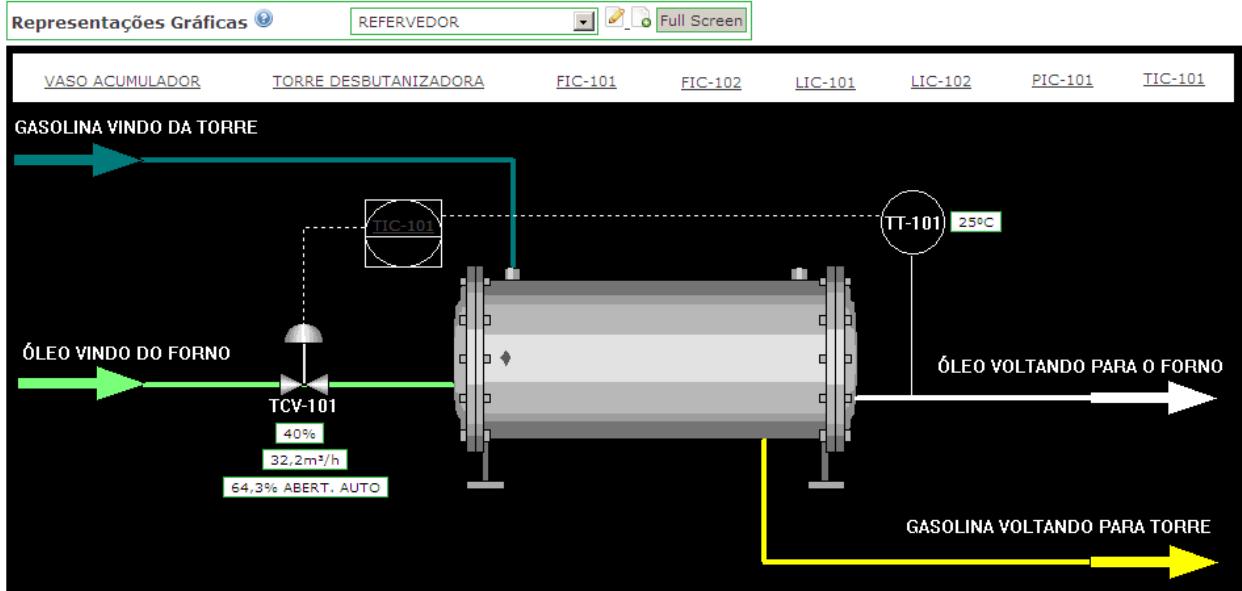
Figura 19 – Tela do vaso acumulador



Fonte: Os autores

A tela do referedor, Figura 20, apresenta somente uma malha de controle, a de temperatura de fundo da torre. A variação de temperatura é controlada através da mudança de vazão do óleo térmico vindo do forno. Quando a temperatura do óleo que sai do referedor é inferior ao pretendido, o controlador TIC-101 manda sinal para aumentar a abertura da válvula TCV-101, fazendo com que entre mais óleo térmico.

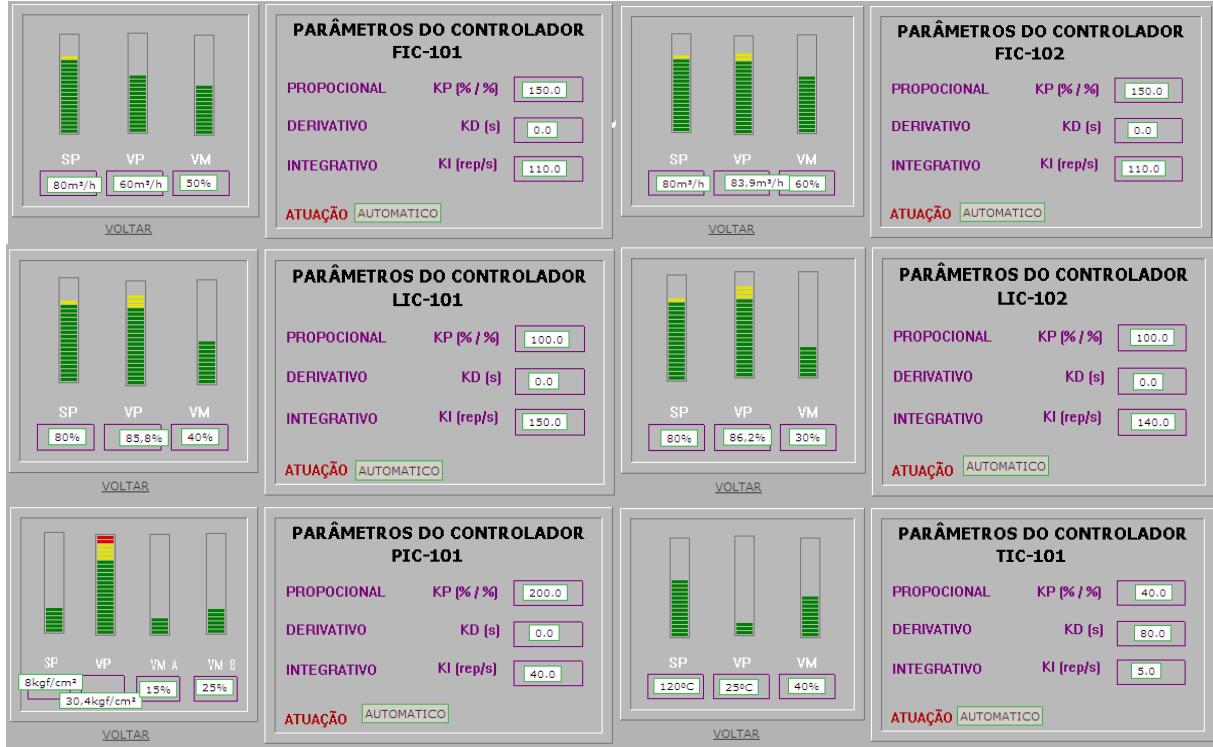
Figura 20 - Tela do referedor



Fonte: Os autores

A Figura 21 representa a tela dos controladores, onde através do comportamento do sistema é possível mudar os parâmetros de controle, passar de automático para manual e vice-versa, acompanhar valores das variáveis do processo, variáveis manipuladas e *setpoint*, mudar o valor do *setpoint* quando em automático, etc.

Figura 21 - Telas dos controladores PID



Fonte: Os autores

#### 4.3. INTEGRAÇÃO DOS SOFTWares

Para integrar o software de simulação e o de supervisão utilizou-se o servidor de comunicação MatrikonOPC. Foi necessária a instalação de uma biblioteca específica, OPC Toolbox, no Simulink para que a comunicação pudesse ser estabelecida. Com isso, utilizando blocos de escrita e leitura, foi possível enviar e receber dados do MatrikonOPC. O “ANEXO D” mostra como deve ser configurado os blocos OPC que são utilizados no Simulink.

Após a configuração dos blocos, é necessário que seja feita a declaração das variáveis no MatrikonOPC, como mostra o “ANEXO B”, e que elas sejam dispostas online.

Como última etapa do processo de integração dos softwares, assim como exposto no “ANEXO A”, deve-se conectar o ScadaBR com o servidor OPC. Com isso, obtém-se todas as variáveis no ScadaBR para que as mesmas sejam utilizadas nas telas de supervisão.

#### 4.4. ACESSO VIA WEB

O ScadaBR, por ter sua tela de supervisão hospedada em uma página da web, possibilita que a mesma seja disponibilizada em uma rede local ou em outra rede qualquer.

Através do *public view* consegue-se colocar a tela de supervisão em rede local, como evidenciado no “ANEXO C”. Para disponibilizar o simulador na web em uma rede diferente, os passos são parecidos, mas algumas modificações têm que ser feitas:

- Deve-se ter um IP fixo de internet para que a aplicação rode neste endereço. Para ter acesso ao IP dinâmico (da web) para a realização de testes, utilizamos o site “[www.myip.com.br](http://www.myip.com.br)”, este nos fornece o IP o qual está conectado. Este IP dinâmico é alterado automaticamente de tempos em tempos, por isso é necessário ter um IP estático;
- É necessário fazer modificações no roteador para que seja possível disponibilizar a página na web. Para fazer estas modificações é necessário entrar nas configurações do roteador. O roteador utilizado neste trabalho foi um roteador Sagemcom GVT e para acessar a página de configuração do roteador basta conectar o computador a ele e, na página da web, digitar o seguinte endereço de IP: 192.168.25.1 (cada modelo de roteador tem um endereço de IP diferente). Em seguida coloque o *login* de acesso e a senha que são admin e gvt12345, respectivamente. Estes são os dados de *login* e senha padrão, caso nunca tenham sido alterados. O que deverá ser alterado nas configurações do roteador é o intervalo de portas, quando este é habilitado, as solicitações de comunicação enviadas pela internet para a porta

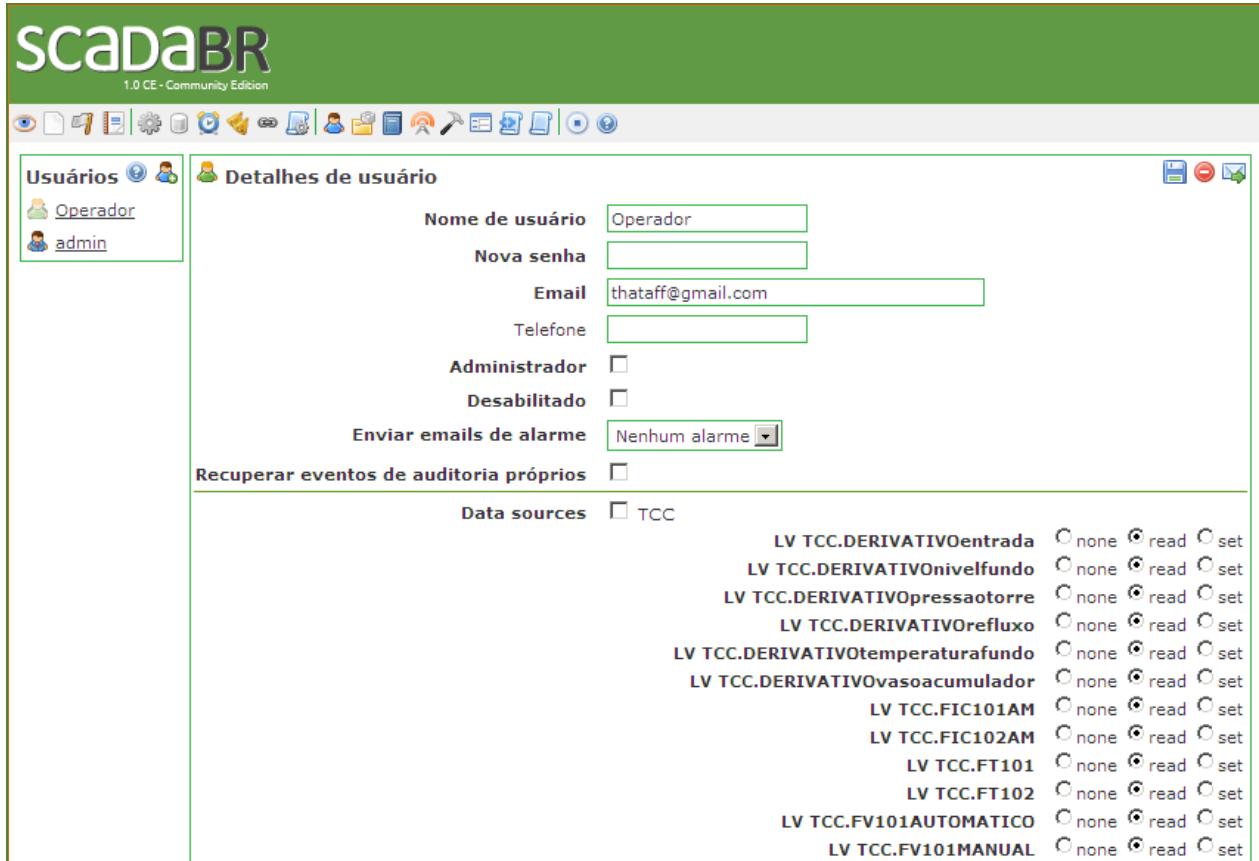
WAN do roteador serão direcionadas para o endereço IP da LAN que foi especificado. Neste caso, o intervalo das portas deverá ser de 8000 a 9000, pois a porta utilizada pelo ScadaBR é a 8080. O IP de destino, ou IP LAN, é o IP da máquina onde a aplicação está rodando, que pode ser encontrado através do comando “ipconfig” no Prompt de Comando.

Tendo feito todas as configurações e estando o ScadaBR funcionando em modo “set” ou “read”, já é possível acessar o simulador através de outra rede pelo endereço da representação gráfica.

#### **4.5. CONTROLE DE GRUPO DE ACESSO POR SENHA**

Como objetivo final do trabalho, ficou estabelecida a criação de grupos de acesso, onde através de senhas pessoais cada operador só tem acesso ao que lhe cabe. Com isso, foi feito um exemplo onde foram criados dois grupos de acesso: admin e operador, como pode ser visto na Figura 22.

Figura 22 - Criação de grupos de acesso



Fonte: ScadaBR

O admin é o administrador do supervisório e, com isso, tem acesso à edição de todos os data points, telas de supervisão, pode criar e excluir grupos de acesso, etc. O operador só terá acesso ao que o administrador liberar, incluindo visualização de telas de supervisão, de *data points*, modificação de valores, entre outros. Se o operador não tiver permissão para modificar um valor e mesmo assim tentar, o ScadaBR exibirá uma mensagem de erro, como mostrado na Figura 23.

Figura 23 - Mensagem de erro



Fonte: ScadaBR

Para fim de testes neste trabalho, o operador está autorizado a visualizar todas as telas do supervisório e todos os *data points*, porém não pode modificar nenhum destes parâmetros. Caso queira fazer todas estas modificações, deve-se entrar na parte de usuário, como administrador, e nesta página, estas configurações podem ser feitas, incluindo mudança de senhas, permissão de acesso as variáveis, etc.

## **5. CONSIDERAÇÕES FINAIS**

O simulador industrial é uma ferramenta extremamente poderosa podendo reduzir de forma massiva os custos de treinamento, diminuindo riscos e etc. O simulador proposto funcionou como esperado, porém encontramos pontos positivos e negativos.

Além dos pontos positivos citados acima, constatou-se que é possível criar um simulador com funcionalidades bem aproximadas da realidade operacional, incluindo o uso de senhas para controle de acesso pessoal para cada operador, visando assegurar que cada um só tenha acesso ao que é de sua responsabilidade. Também é extremamente relevante falar da facilidade que a utilização de um simulador que está disponível na *web* traz, onde o operador, estudante, professor podem acessá-lo em casa, no escritório, em uma sala de treinamento, entre outros, tornando a utilização mais acessível.

Os pontos negativos encontrados no trabalho foram o não determinismo devido ao cascataemento dos softwares, onde existe um atraso no envio e recebimento de dados, ou seja, não é em tempo real. Também foram encontrados problemas com o software de supervisão devido o mesmo não possuir grandes quantidades de elementos gráficos para o desenvolvimento das telas e também por não ter uma periodicidade de atualização, acarretando em problemas como: não funcionar em todas as versões do Windows, travamentos de tempos em tempos, entre outros.

No *link* disponível ([https://www.youtube.com/watch?v=\\_E1npIFztSg](https://www.youtube.com/watch?v=_E1npIFztSg)) pode-se observar o simulador funcionando no computador onde foi instalado. Através de outros dispositivos como tablet, celulares e outro computador foi possível acessar o simulador via *web* utilizando o *link* criado para o acesso.

Para trabalhos futuros propõem-se o desenvolvimento de outros tipos de simuladores, tentando melhorar a velocidade da comunicação entre os *softwares* utilizados, bem como utilizar outros *softwares* e protocolos de comunicação para fazer comparação de performance entre os novos *softwares* utilizados e os que foram utilizados neste trabalho.

## **6. REFERÊNCIAS**

- BANKS, J. **Introduction to Simulation**, in Proceedings of 2000 Winter Simulation Conference, J.A. Joines, R. R. Barton, K. Kang, and P. A. Fishwick, eds.
- BANKS, C. M.; SOKOLOWSKI, J. A. **Modeling and Simulation Fundamentals: Theoretical Underpinnings and Practical Domains**. Publicado por John Wiley & Sons, Inc., Hoboken, New Jersey.
- CAMPOS, M. C. M. M.; TEIXEIRA, H. C. G. **Controles típicos de Equipamentos e Processos Industriais**. 1. ed. [S.I.]: Edgard Blucher, 2006.
- ELSHOUT, R. V.; WETHERILL, F. E. **Effective Operator Training Program**. Chemical Engineering, 1987.
- FERREIRA, J. J. P. **Suporte do Ciclo de Vida dos Sistemas Integrados de Fabrico através de Modelos Executáveis sobre Infraestruturas de Integração**. Tese de Doutoramento em Engenharia Eletrotécnica e de Computadores, Faculdade de Engenharia da Universidade do Porto, 1995.
- FREITAS FILHO, J. de. **Introdução à Modelagem e Simulação de Sistemas**. Ed. Visual Books, 2008.
- GOLDSMAN, D.; NANCE, R. E.; WILSON, J. R. **A Brief History of Simulation Revisited**, in Proceedings of the 2010 Winter Simulation Conference, B. Johansson, S. Jain, J. Montoya-Torres, J. Hugan, and E. Yucesan, eds.
- LAW, A. M.; KELTON, W. D. **Simulation Modeling & Analysis**. Second Edition, Industrial Engineering Series, Ed. McGraw-Hill International Editions, 1991.

LIMA, D. G. de; COELHO, F. J. dos S. **Proposta de Otimização da Produção de Gás Liquefeito de Petróleo na Torre Desbutanizadora.** Instituto Federal de Educação, Ciência e Tecnologia Fluminense, Campos dos Goytacazes, 2014.

PEGDEN, C. D.; SHANNON, R. E.; SADOWSKI R. P. **Introduction to simulation using SIMAN.** Ed. Mc Graw-Hill, 1995.

RUBINSTEIN, R. Y.; MELAMED, B. **Modern Simulation and Modeling.** Wiley Series in Probability and Statistics, Applied Probability and Statistics Section, A Wiley-Interscience Publication, John Wiley & Sons, INC., 1998.

SANT'ANNA, A. A. **Simulação de Processamento de Gás Natural em Plataforma Off-Shore.** Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2005.

SANTOS, E. M. dos. **Gás natural: estratégias para uma energia nova no Brasil.** São Paulo: Annablume, Fapesp, Petrobras, 2002.

SHANNON, R. E. **Introduction to the Art and Science of Simulation,** in Proceedings of 1998 Winter Simulation Conference, D.J.Medeiros, E.F. Watson, J.S. Carson and M.S. Manivannan, eds.

SILVA, D. R. da; GLÓRIA, L. S. **Simulador de uma Unidade de Processamento de Condensado de Gás Natural.** Instituto Federal de Educação, Ciência e Tecnologia Fluminense, Campos dos Goytacazes, 2012.

WINTER, P. **Simulator Speeds Operator Training.** Chemical Engineering: Plant Notebook, New York, 1996.

## ANEXO A - TUTORIAL DE INSTALAÇÃO E UTILIZAÇÃO SCADABR

A instalação do ScadaBR deve prosseguir de acordo com as ilustrações abaixo. Assim que a tela de instalação abrir, clique em “Próximo” e uma nova tela irá abrir. Nesta nova tela aparecerá o termo de acordo do ScadaBR, para que possa prosseguir com a instalação é necessário estar de acordo com o termo. Para isso basta clicar em “Eu concordo” para continuar com a instalação.

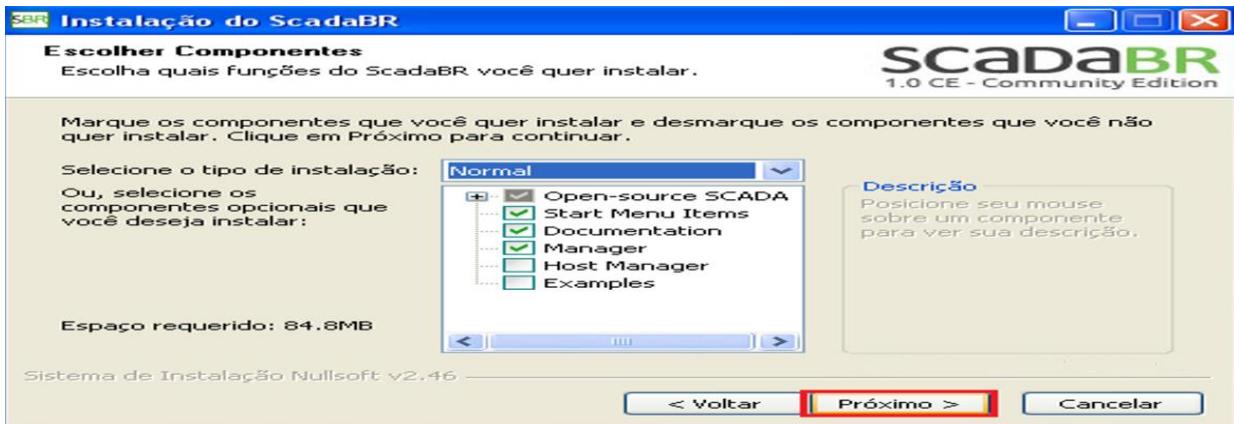
Figura 24 - Tela de instalação



Fonte: ScadaBR

Depois de ter concordado com o termo, uma nova tela irá aparecer. Nesta nova tela o usuário poderá escolher o tipo de instalação ou selecionar os componentes opcionais que deseja instalar. Após selecionar clique em “Próximo”.

Figura 25 - Tela de instalação



Fonte: ScadaBR

Uma nova tela aparecerá para que seja escolhida a pasta destino do arquivo. Em seguida clique em “Próximo” para dar continuidade a instalação. Uma tela de configuração básica do Tomcat aparecerá solicitando a “porta do conector” e deverá ser preenchido de acordo com a ilustração abaixo. Após o preenchimento clique em “Próximo” para prosseguir.

Figura 26 - Tela de opções de configuração

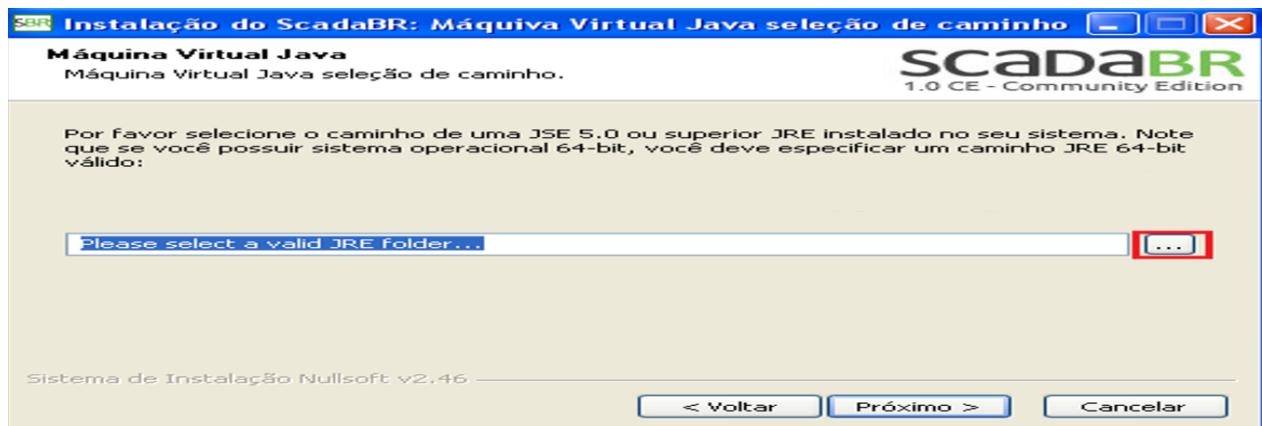


Fonte: ScadaBR

A próxima tela será para localizar a pasta onde está instalado o Java. O Java tem que ser previamente instalado no computador. É obrigatório ter o Java instalado

antes de iniciar a instalação do ScadaBR. Assim que o diretório for encontrado, selecione a pasta e clique em “Próximo” para prosseguir.

Figura 27 - Tela de seleção do Java



Fonte: ScadaBR

Feito isso, o ScadaBR começará a ser instalado.

Após a instalação do ScadaBR o próximo passo é a utilização do mesmo. Primeiro dê 2 cliques no ícone que, provavelmente, aparecerá na área de trabalho do computador. Quando a tela inicial do ScadaBR abrir, será necessário preencher os campos “User id” e “Password”. O preenchimento será conforme a ilustração abaixo.

Figura 28 - Tela de login

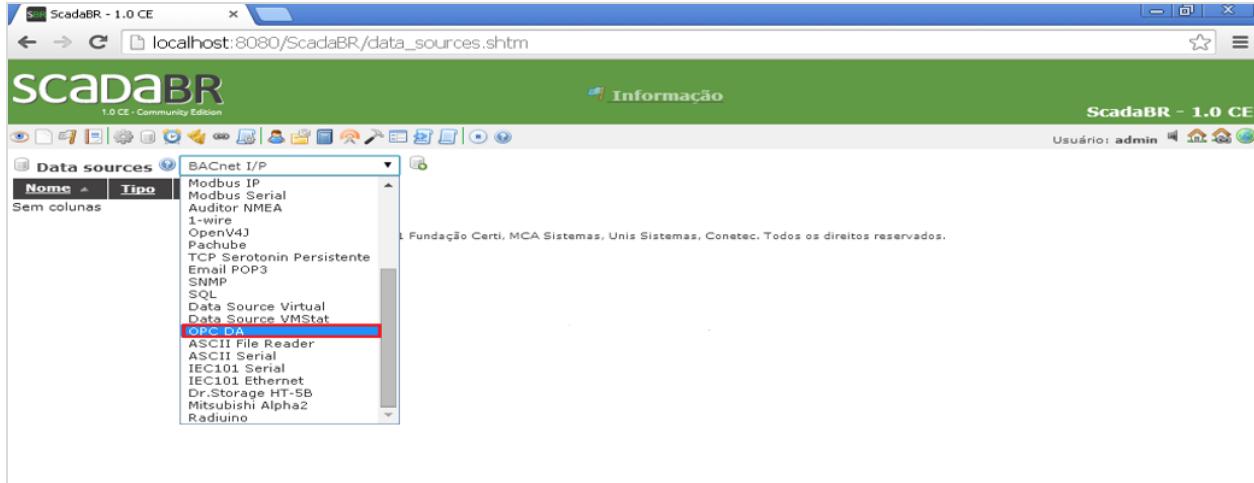


Fonte: ScadaBR

Após o preenchimento deve-se clicar em “Login”, uma nova tela aparecerá. Nesta nova tela é selecionado o tipo de “Data Source” que será utilizado. Como

estamos usando o MatrikonOPC, o tipo de “*Data Source*” que deverá ser utilizado é o “OPC DA”.

Figura 29 - Tela do Data Source



Fonte: ScadaBR

Após selecionar o tipo de “*Data Source*” deve-se clicar no botão adicionar, uma nova tela aparecerá. Esta nova tela deverá ser preenchida de acordo com a configuração do computador. O preenchimento deverá ser da seguinte forma:

Nome: da sua escolha

*Export ID (XID)*: é preenchido automaticamente

*Host*: nome do computador

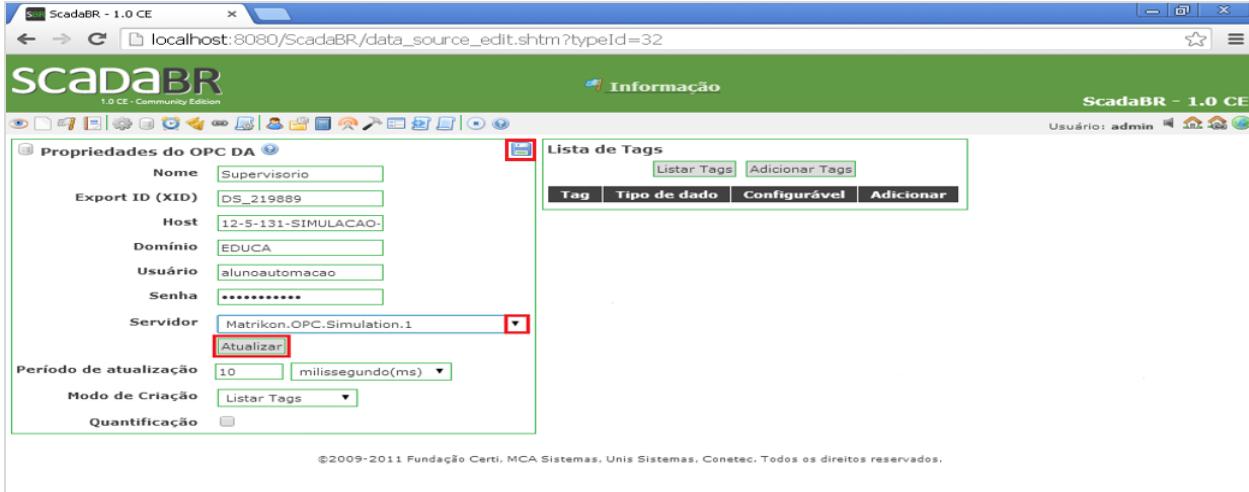
*Domínio*: domínio do computador, caso não houver, preencher com “localhost”

*Usuário e senha*: do próprio computador, é necessário que o computador tenha uma senha

O servidor deverá ser selecionado de acordo com o que estiver sendo utilizado, neste caso estamos utilizando o MatrikonOPC.

O período de utilização sugerido é de 10 milissegundos (ms).

Figura 30 - Propriedade do OPC DA



Fonte: ScadaBR

Após o preenchimento, clique em “Atualizar”, caso o MatrikonOPC não apareça automaticamente no item servidor, clique sobre a seta e selecione manualmente. Assim que estiver tudo preenchido, clique em “Salvar” e logo em seguida em “Habilitar” para que seja possível listar os *tags* (os *tags* são criados no MatrikonOPC, no “ANEXO B” possui um tutorial).

Para adicionar os *tags*, deve-se clicar em “Listar Tags”, em seguida aparecerá uma lista com as variáveis. Deve-se apenas selecionar as que serão utilizadas. Após selecionar as variáveis, clicar em “Adiconar Tags”.

Figura 31 - Propriedades do OPC DA

The screenshot shows the ScadaBR software interface. On the left, there is a configuration panel titled "Propriedades do OPC DA" (OPC DA Properties) with fields for Name, Export ID (XID), Host, Domínio, Usuário, Senha, Servidor, Período de atualização, Modo de Criação, and Quantificação. On the right, there is a table titled "Lista de Tags" (List of Tags) with columns for Tag, Tipo de dado (Type), Configurável (Configurable), and Adicionar (Add). The "Adicionar" button is highlighted with a red box. The table lists various tags such as Teste.Teste3, Teste.Teste1, Teste.Teste2, #MonitorACIFile, @Clients, Bucket Brigade.ArrayOfReal8, Bucket Brigade.Boolean, Bucket Brigade.Int1, Bucket Brigade.Int2, Bucket Brigade.Int4, Bucket Brigade.Money, Bucket Brigade.Real1, Bucket Brigade.Real4, Bucket Brigade.Real8, Bucket Brigade.Real9, Bucket Brigade.String, Bucket Brigade.Time, Bucket Brigade.UInt1, Bucket Brigade.UInt2, Bucket Brigade.UInt4, Random.ArrayOfReal8, and Random.ArrayOfString.

Fonte: ScadaBR

Após adicionar os tags é necessário habilitá-las. É possível habilitá-las uma a uma clicando no ícone em vermelho ou habilitá-las todas de uma vez clicando no ícone indicado.

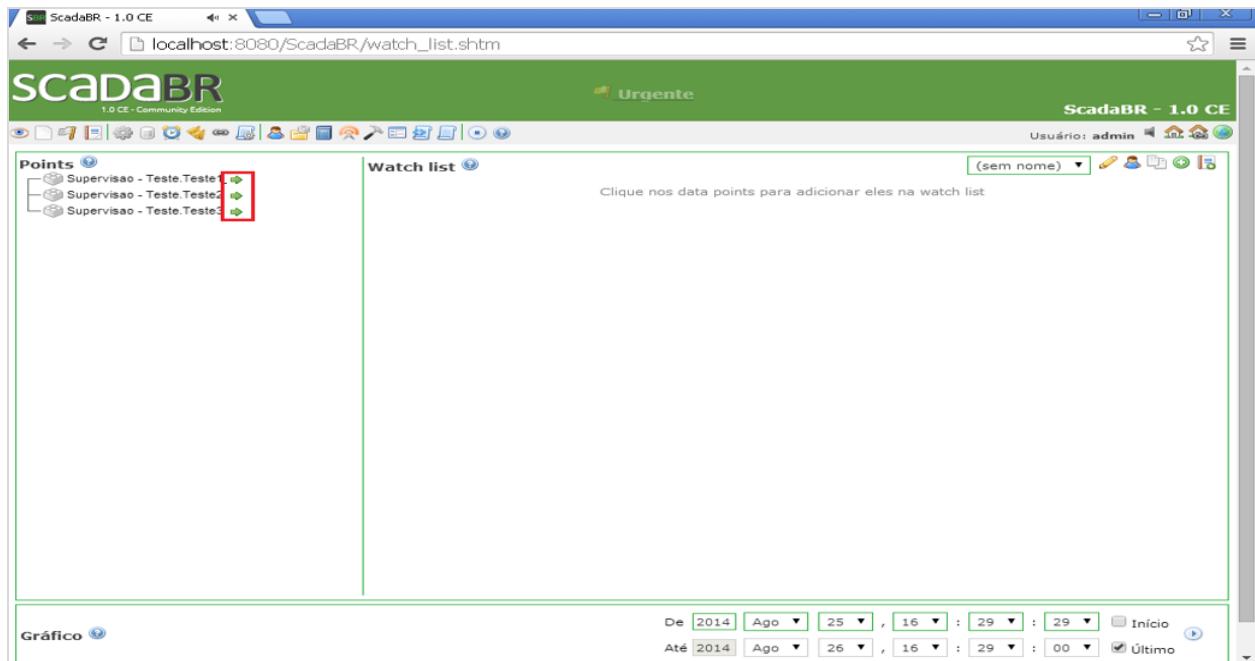
Figura 32 - Propriedades do OPC DA

The screenshot shows the ScadaBR software interface. On the left, there is a configuration panel titled "Propriedades do OPC DA" (OPC DA Properties) with fields for Nome, Tipo de dado, Status, and a "Detalhes do data point" (Data point details) section. The "Nome" field is set to "Teste.Teste2" and the "Export ID (XID)" field is set to "DP\_008186". On the right, there is a table titled "Data points" with columns for Nome, Tipo de dado, and Status. The table lists data points such as Write.Error.Int1, Write.Error.Int2, Write.Error.Int4, Write.Error.Money, Write.Error.Real4, Write.Error.Real8, Write.Error.String, Write.Error.Time, Write.Error.UInt1, Write.Error.UInt2, Write.Error.UInt4, Write.Only.ArrayOfReal8, Write.Only.ArrayOfString, Write.Only.Boolean, Write.Only.Int1, Write.Only.Int2, Write.Only.Int4, Write.Only.Money, Write.Only.Real8, Write.Only.String, Write.Only.Time, Write.Only.UInt1, Write.Only.UInt2, and Write.Only.UInt4. The "Status" column shows icons indicating the status of each data point.

Fonte: ScadaBR

Após terminar as configurações do *data source*, já é possível clicar na “*Watch List*” para visualizar as variáveis. Clique sobre as setinhas para adicionar os *data points* na *watch list*.

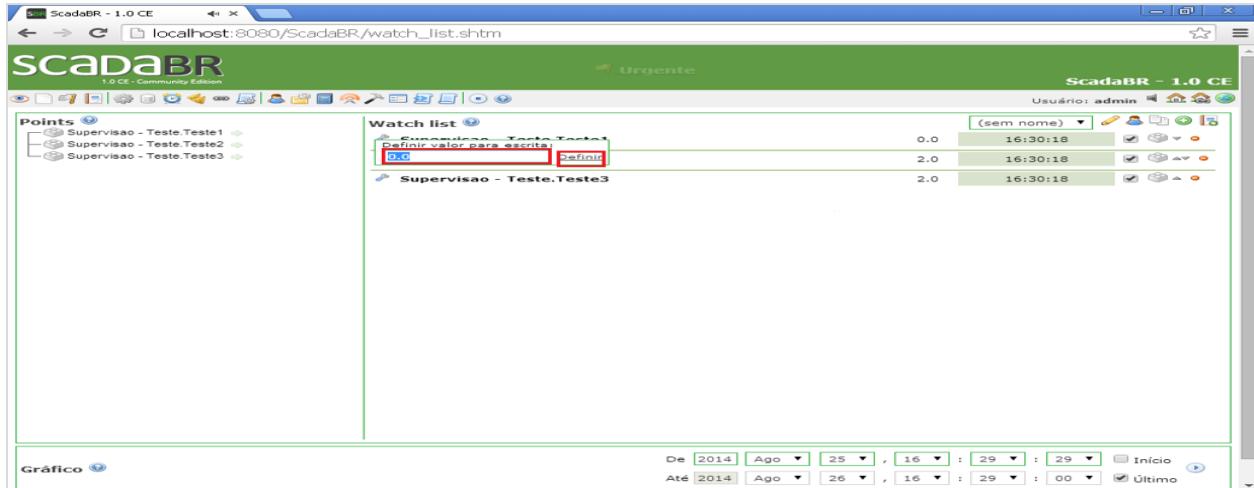
Figura 33 - Tela dos data points



Fonte: ScadaBR

Para colocar um novo valor da variável, basta clicar sobre a variável, digitar um novo valor e clicar em “Definir”.

Figura 34 - Tela dos data points



Fonte: ScadaBR

Após ter adicionado as variáveis na *watch list* já é possível criar uma representação gráfica. Clique sobre o botão indicado para criar uma representação gráfica e depois clique em “Adicionar representação”.

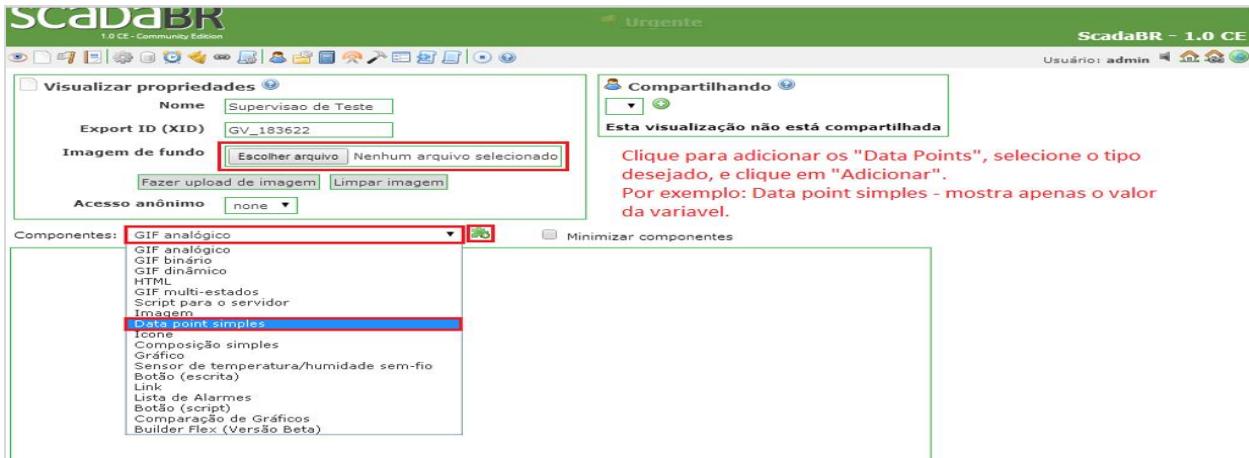
Figura 35 - Tela de representação gráfica



Fonte: ScadaBR

Uma nova tela aparecerá contendo os componentes, também é possível fazer o *download* de alguma imagem para utilizá-la como imagem de fundo. Na lista dos componentes possui vários itens como: *data points* (monitoramento das variáveis), criação de gráficos, botões, etc.

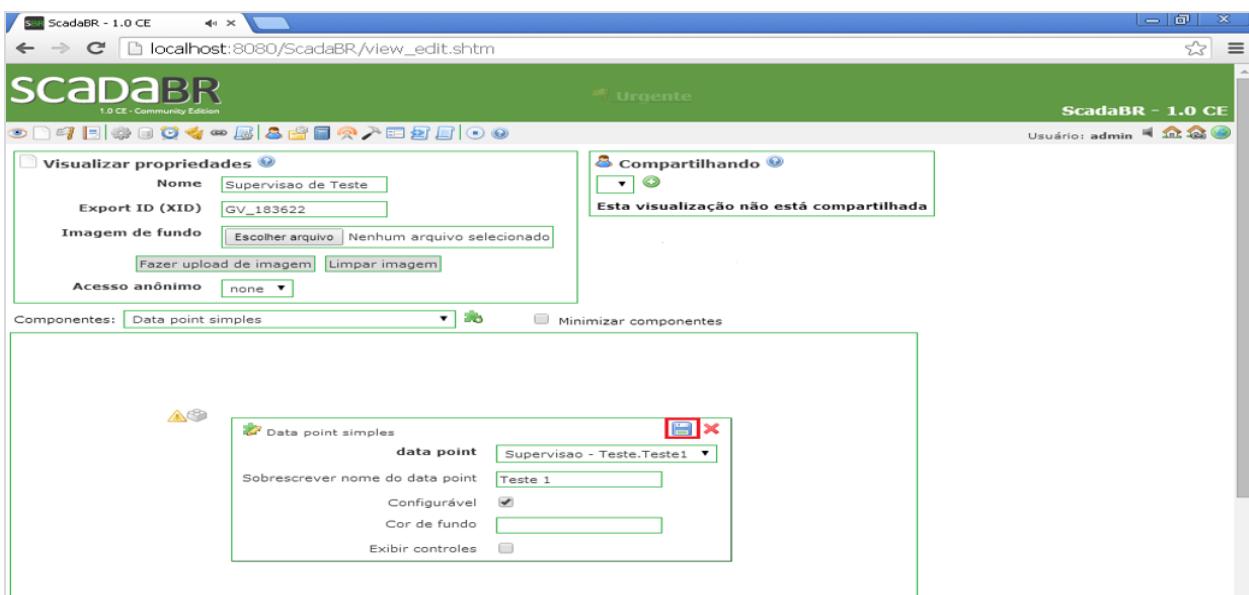
Figura 36 - Tela de representação gráfica



Fonte: ScadaBR

Para configurar um componente basta clicar sobre o mesmo, e uma tela de configuração aparecerá. Na tela deve ser selecionado o *data point*, ou seja, a variável que deseja relacionar ao componente, também possui a função configurável, que permite ou não que a variável tenha o seu valor alterado enquanto a tela estiver “rodando”.

Figura 37 - Tela de representação gráfica



Fonte: ScadaBR

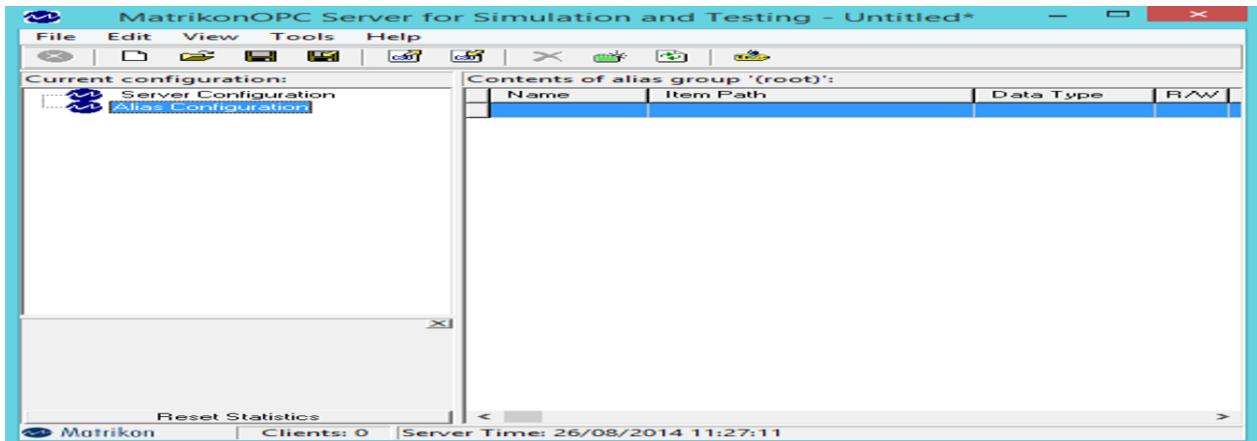
## **CONSIDERAÇÕES IMPORTANTES**

- Para instalação do ScadaBR é necessário ter o Java instalado no computador
- Lembrar sempre de iniciar o Tomcat antes de iniciar o ScadaBR. O *download* do Tomcat pode ser feito pelo próprio site do Tomcat.
- O ScadaBR só está disponível para o Windows XP ou 7, e pode ser baixado gratuitamente pelo próprio site do ScadaBR.

## ANEXO B - TUTORIAL DE UTILIZAÇÃO E DECLARAÇÃO DE VARIÁVEIS NO MatrikonOPC

O primeiro passo é abrir o arquivo de nome MatrikonOPC Server for Simulation and Testing. A tela abaixo aparecerá.

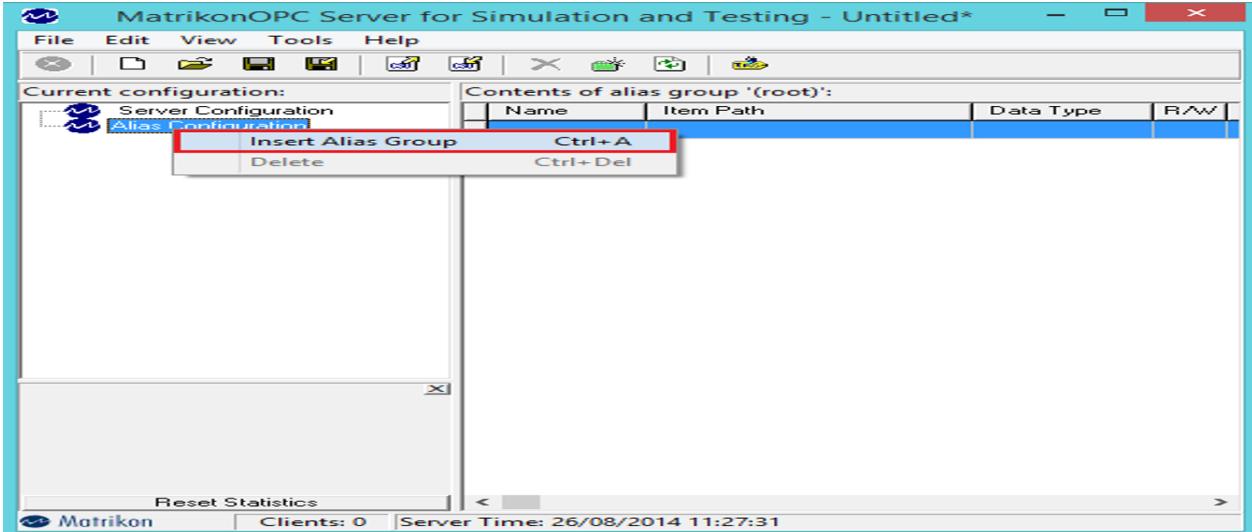
Figura 38 - Tela do servidor



Fonte: MatrikonOPC

Assim que a tela aparecer, clique com o botão direito sobre “Alias Configuration” e selecione a opção “*Insert Alias Group*”.

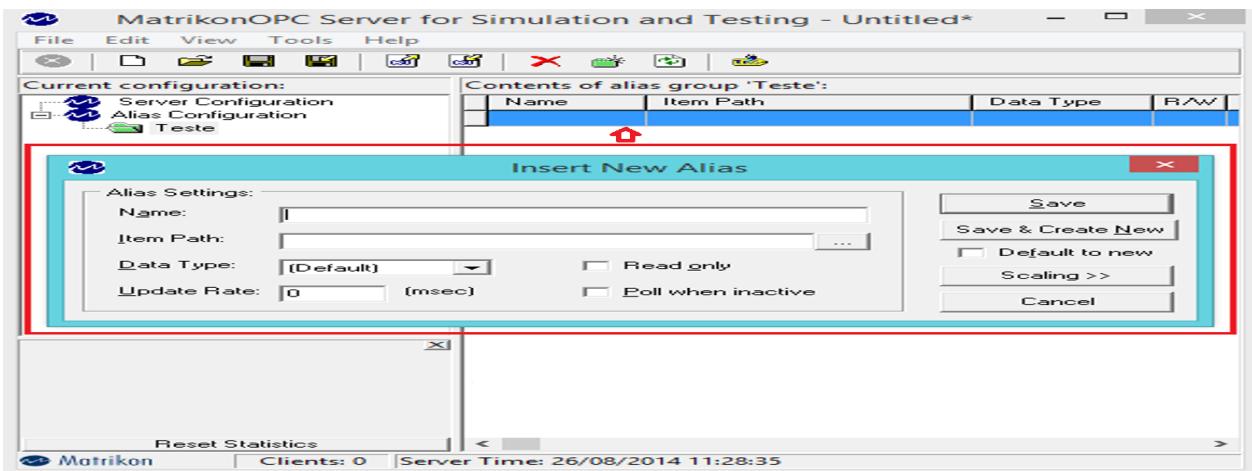
Figura 39 - Tela do servidor



Fonte: MatrikonOPC

Logo em seguida será criada uma pasta, onde as variáveis serão armazenadas. Clique na seta indicada para adicionar uma nova variável. Uma nova tela será mostrada para preenchimento dos campos.

Figura 40 - Tela de inserção de novas variáveis

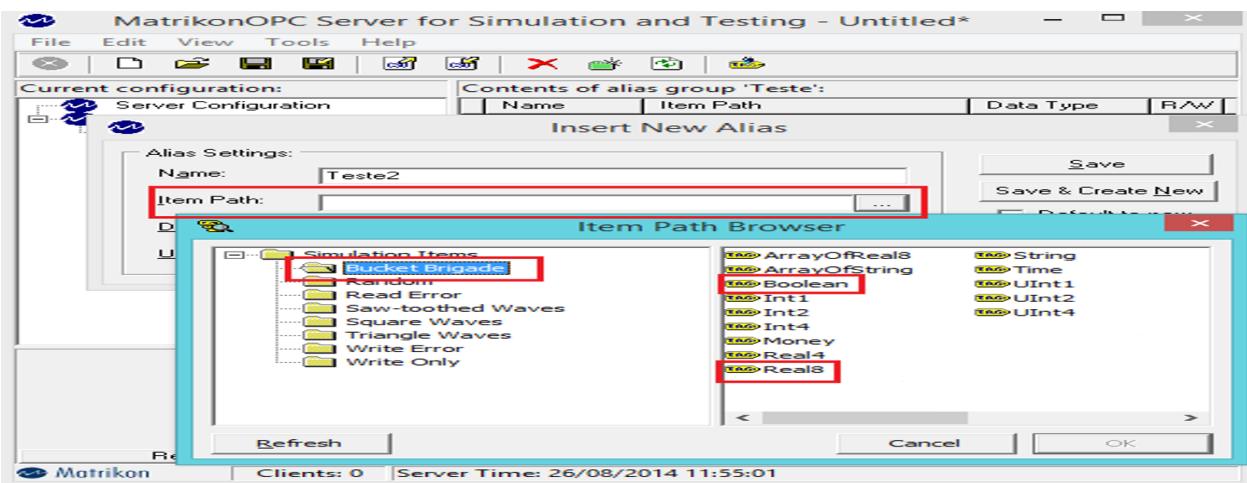


Fonte: MatrikonOPC

O preenchimento deverá ser da seguinte forma: *name* (nome da variável de sua escolha. É aconselhável colocar um nome de acordo legível para facilitar na hora

da utilização do mesmo); *item path* (seleciona o tipo de variável, por exemplo, real, booleano, inteiro, etc); *data type* (será de acordo com o tipo de variável, por exemplo, se a variável for real, selecionar o “REAL8”); *update rate* (é aconselhável colocar o valor 10 ms). Abaixo uma ilustração de como selecionar o tipo de variável. Clique em “*Item Path*”, depois em “*Simulation Items*”. Em seguida clique em “*Bucket Brigade*” e selecione o tipo de variável.

Figura 41 - Tela de seleção do tipo de variável

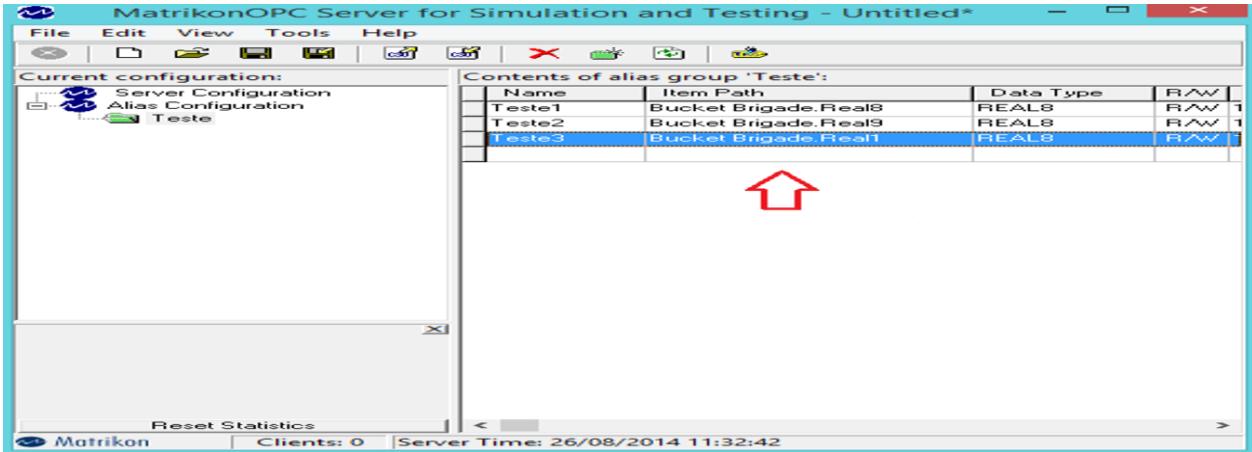


Fonte: MatrikonOPC

Assim que terminar de preencher, selecione “Save” para salvar ou “Save & Create New” para salvar esta variável e criar outra. Lembre-se que ao criar outra variável do mesmo tipo, o valor que está escrito após o “Real” no “*Item Path*” deverá ser modificado. Além disso, pode ser colocado qualquer número, contando que NUNCA seja um número já utilizado.

Após terem sido criadas, as variáveis ficam listadas conforme ilustração abaixo na tela do MatrikonOPC.

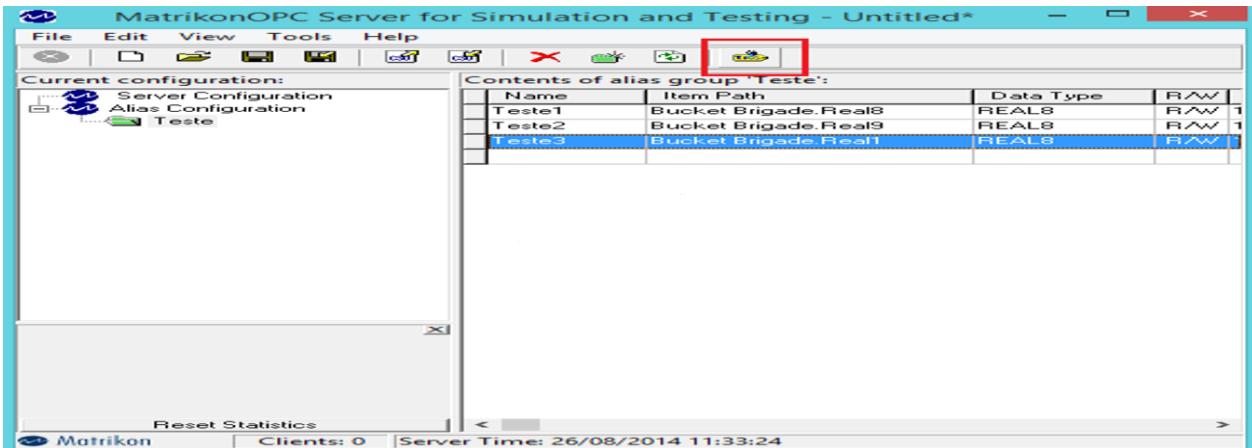
Figura 42 - Tela do servidor



Fonte: MatrikonOPC

Após criar as variáveis, é necessário visualizá-las para validá-las. Para fazer isto, clique no botão “View OPC Tags for this Server”.

Figura 43 - Tela do servidor

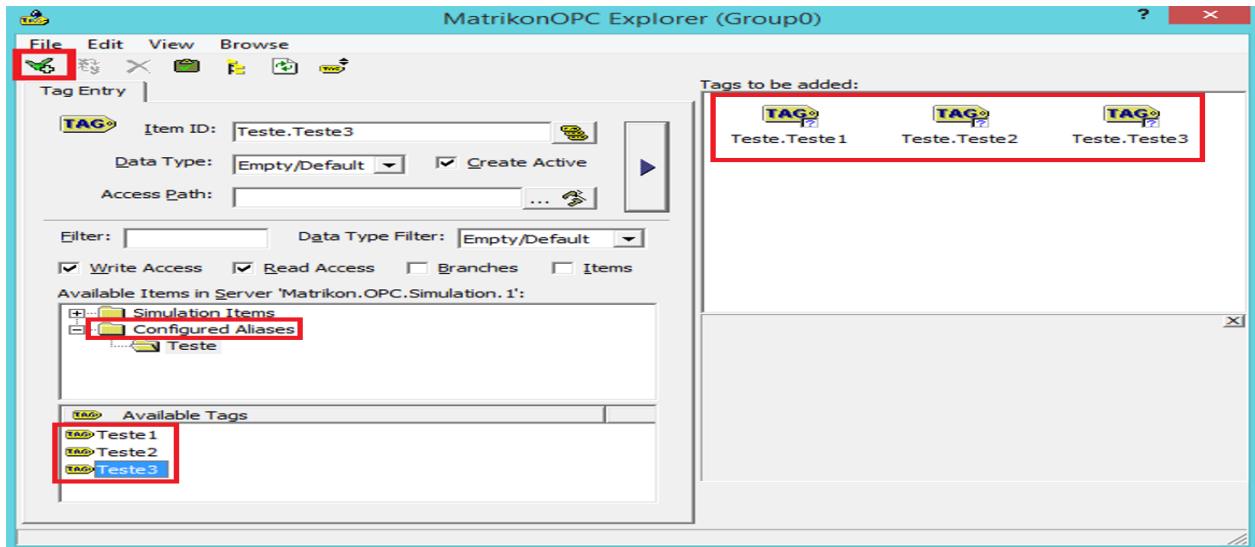


Fonte: MatrikonOPC

Uma nova tela aparecerá automaticamente. Para localizar as variáveis, clique em “Configured Aliases” e selecione o grupo desejado (o mesmo grupo que foi criado no MatrikonOPC). Em seguida, dê dois cliques sobre cada variável no campo “Available Tags” fazendo com que elas apareçam no campo “Tags to be added”. Após

adicionar todas as variáveis, clique no botão marcado na ilustração abaixo para validá-las.

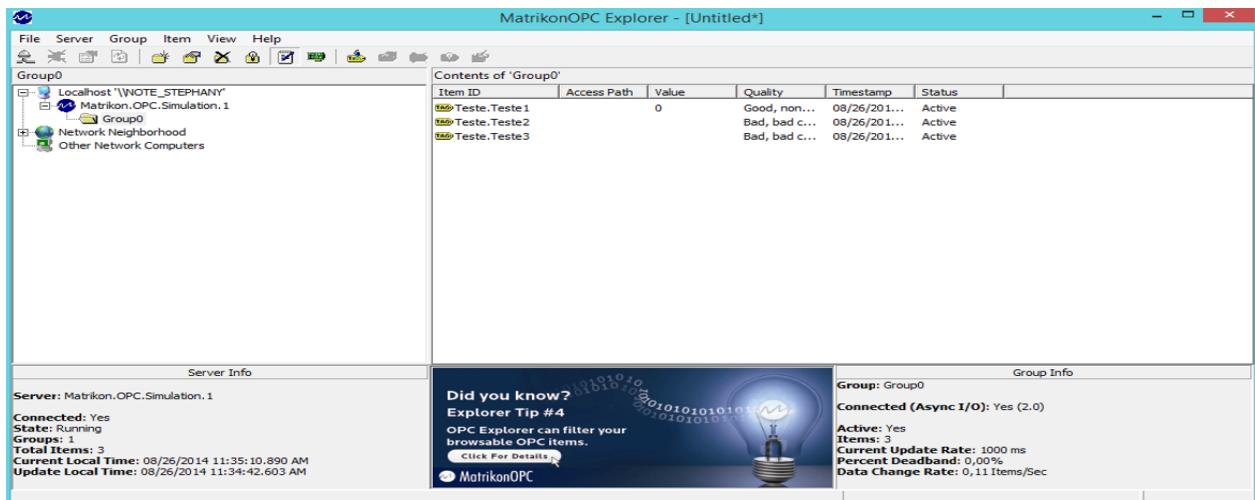
Figura 44 - Tela do servidor



Fonte: MatrikonOPC

Uma nova tela aparecerá automaticamente.

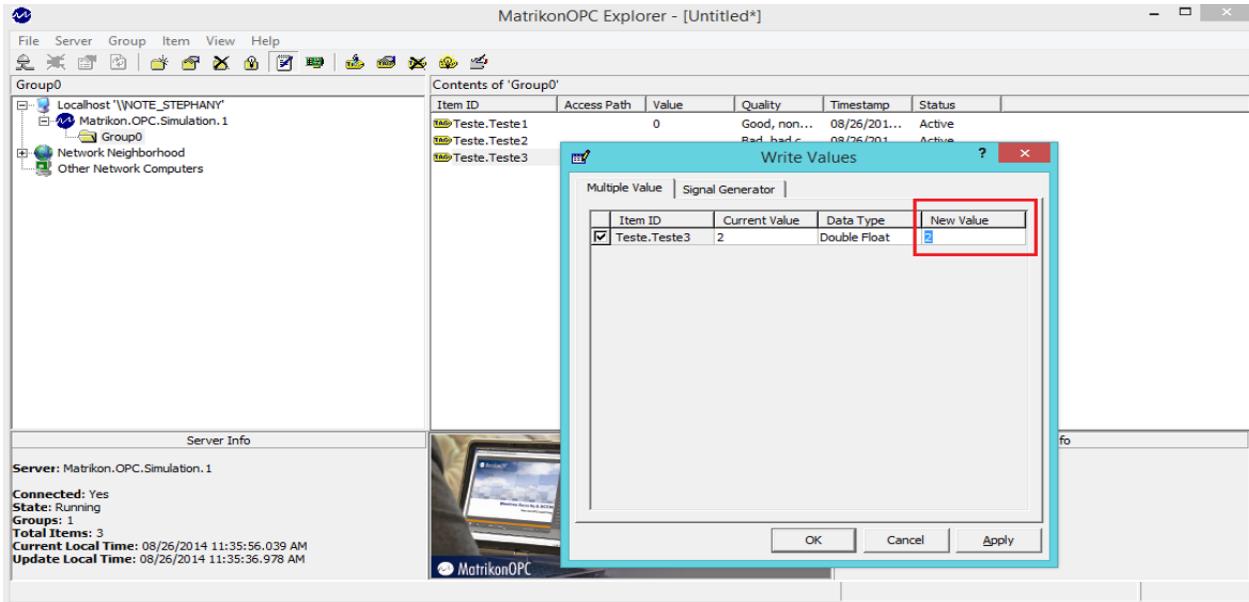
Figura 45 - Tela do servidor



Fonte: MatrikonOPC

Caso alguma variável apareça com “Bad” no item “Quality”, basta clicar sobre ela e alterar o seu valor. Logo após ela aparecerá como “Good”.

Figura 46 - Tela de mudança de valores das variáveis



Fonte: MatrikonOPC

Caso o campo “New Value” (para alterar seu valor) não esteja habilitado, vá até a aba “Signal Generator”, clique em “Start”, aguarde 1 ou 2 segundos e clique em “Stop”. Assim, a variável aparecerá como “Good”.

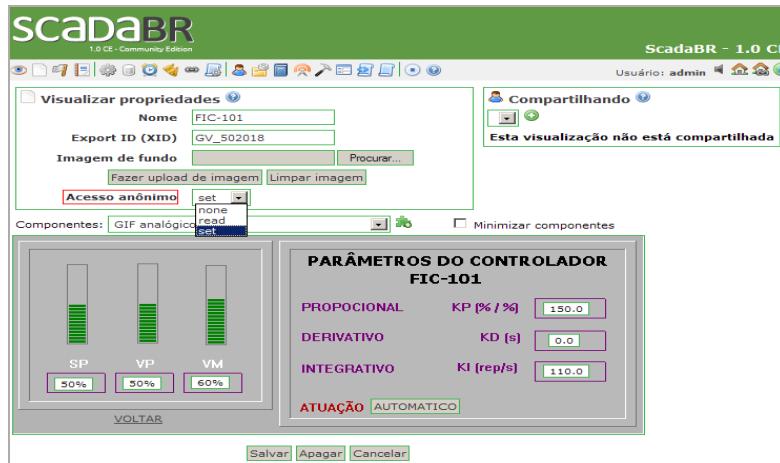
Feito isso as variáveis já estão prontas para serem utilizadas.

## ANEXO C – TUTORIAL DE DISPONIBILIZAÇÃO DA REPRESENTAÇÃO GRÁFICA EM REDE LOCAL

Para colocar a representação gráfica em rede local, temos algumas configurações a fazer:

- Para conseguir realizar o acesso pelo *public view*, o “Acesso Anônimo” da Representação Gráfica deve estar em modo “*set*” (autoriza o usuário a ver e modificar parâmetros) ou “*read*” (autoriza o usuário somente a visualizar a tela de supervisão). Se o “Acesso Anônimo” estiver em “*none*”, a representação ficará desabilitada;

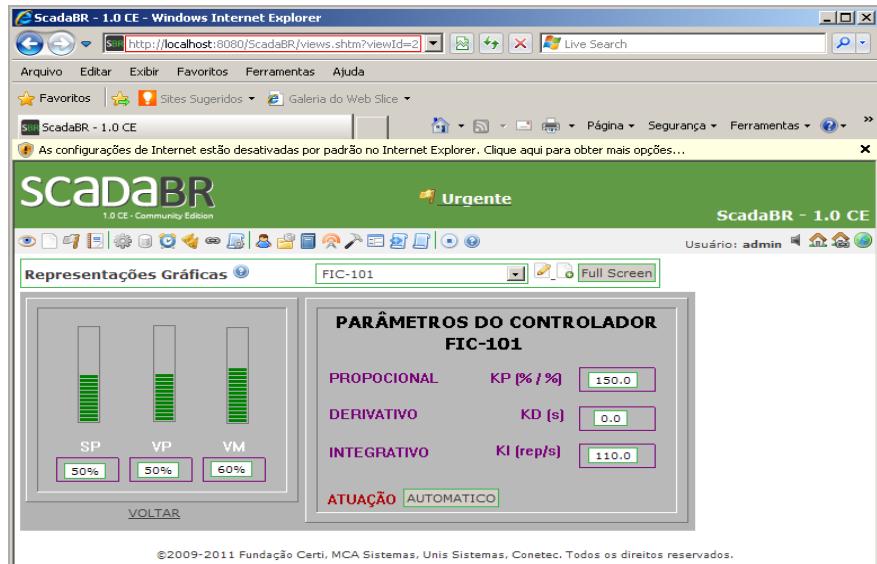
Figura 47 - Tela de acesso ao Public View



Fonte: ScadaBR

- Obter o endereço da representação gráfica, no caso do exemplo abaixo, o endereço é “localhost:8080/ScadaBR/views.shtm?viewId=2”

Figura 48 - Tela de representação gráfica a partir do Public View



Fonte: ScadaBR

- Obter o endereço de IP da máquina onde a aplicação está rodando através do comando “ipconfig” no Prompt de Comando do Windows.

Figura 49 - Prompt de comando

```
C:\> Prompt de comando
Microsoft Windows XP [versão 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\Leandro Gloria>ipconfig

Configuração de IP do Windows

Adaptador Ethernet Conexão local:
    Sufixo DNS específico de conexão . . . : Home
    Endereço IP . . . . . : 192.168.25.9
    Máscara de sub-rede . . . . . : 255.255.255.0
    Gateway padrão. . . . . : 192.168.25.1

C:\Documents and Settings\Leandro Gloria>
```

Fonte: Windows

- Após isso, abrir o navegador e digitar o endereço da representação gráfica com as seguintes modificações:

Endereço da representação gráfica:

localhost:8080/ScadaBR/views.shtm?viewId=2

Endereço de acesso: 192.168.25.9:8080/ScadaBR/public\_view.htm?viewId=2

- No geral, o endereço de acesso ficará da seguinte forma:

**IP da máquina**:8080/ScadaBR/public\_view.htm?viewId=N, onde N é o número da página da representação gráfica, caso a representação gráfica tenha várias páginas, cada uma terá um valor, que pode ser visto quando ela está no modo editar.

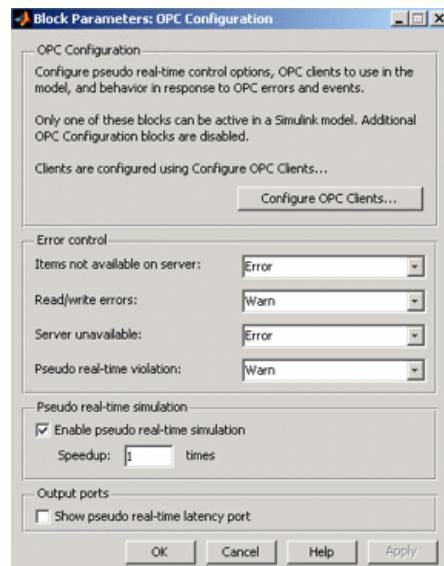
- É importante observar que o *firewall* do Windows deve estar desativado, pois se não estiver nada irá funcionar.

## ANEXO D – TUTORIAL DE CONFIGURAÇÃO DOS BLOCOS OPC

Para se comunicar com servidores OPC do Simulink, primeiro precisa configurar esses servidores. O bloco de configuração OPC gerencia e configura servidores OPC para um modelo Simulink. Cada bloco OPC *Read* ou OPC *Write* usa um servidor dos servidores configurados, e define os itens a ler ou escrever.

Primeiro clique duas vezes no bloco de configuração OPC para abrir o diálogo parâmetros.

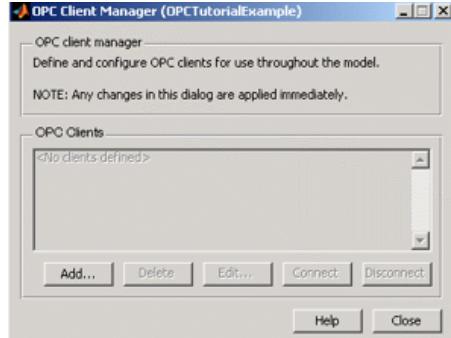
Figura 50 - Configuração do bloco OPC configuration



Fonte: Matlab/Simulink

Em seguida, clique em “Configure OPC clients” para abrir o “OPC Client Manager”.

Figura 51 - Configuração do bloco OPC configuration



Fonte: Matlab/Simulink

Clique em “*Add*” para abrir a caixa de diálogo “*OPC Server Properties*”, depois clique em “*Select*” para escolher um servidor da lista de servidores OPC disponíveis.

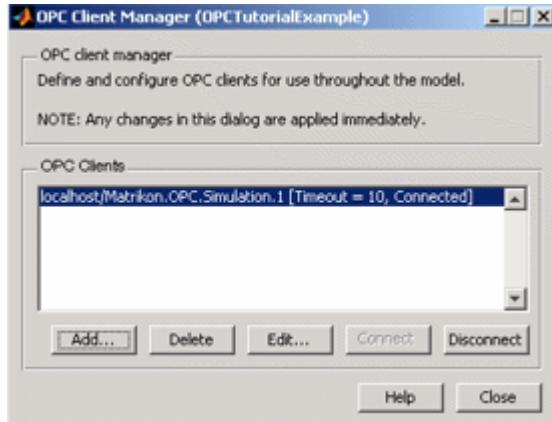
Figura 52 - Configuração do bloco OPC configuration



Fonte: Matlab/Simulink

Clique em “*ok*” para adicionar o servidor OPC em “*OPC Clients*”.

Figura 53 - Configuração do bloco OPC configuration

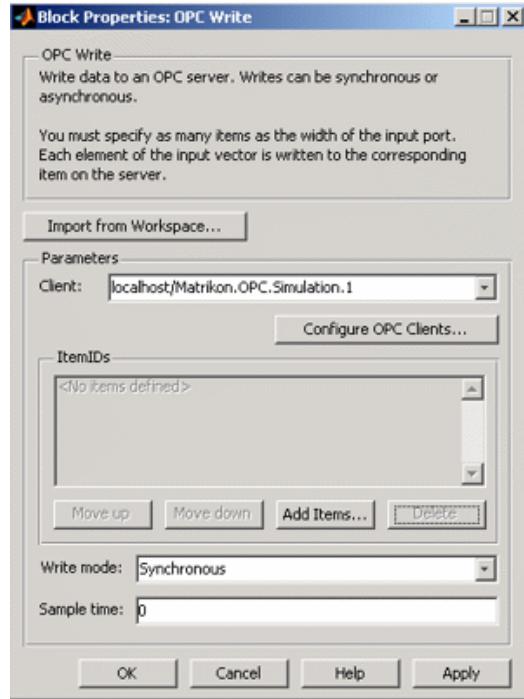


Fonte: Matlab/Simulink

Clique em “ok” na caixa de diálogo da configuração OPC para fechar.

Para definir os parâmetros dos blocos OPC *Read* e OPC *Write* deve clicar duas vezes sobre o bloco. Ao abrir a caixa de diálogo, o servidor MatrikonOPC é automaticamente selecionada como cliente OPC. É necessário especificar as variáveis para a escrita ou leitura.

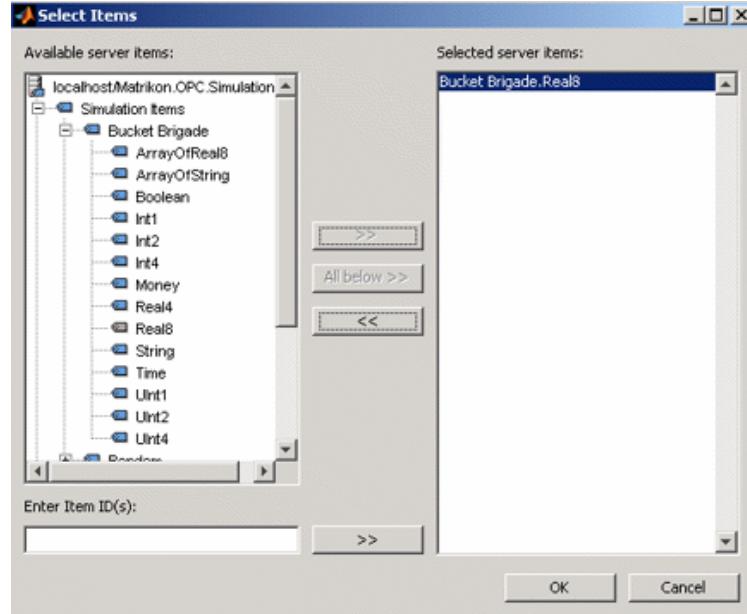
Figura 54 - Configuração do bloco OPC write



Fonte: Matlab/Simulink

Em seguida, clique em “*Add Items*” para exibir um navegador com as variáveis que estão disponíveis para serem usadas. Expanda o nó “*Simulation Items*” e depois expanda o nó “*Bucket Brigade*”. Selecione a variável que deseja e clique em “>>” para adicionar a variável na lista de itens selecionados.

Figura 55 - Seleção das variáveis

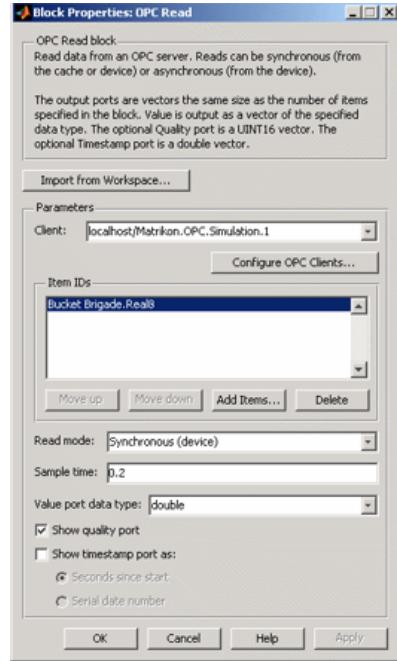


Fonte: Matlab/Simulink

Em seguida, clique em “ok” para adicionar o item “*Bucket Brigade.Real8*” na lista “*ItemID’s*”. Clique em “ok” para aceitar as alterações e fechar o diálogo.

A diferença entre os blocos OPC *Read* e OPC *Write* é que no bloco de leitura é necessário definir o modo de leitura para “*Synchronous (device)*” e o tempo de amostra para 0,2. Também desmarque a opção “*Show timestamp port*”.

Figura 56 - Configuração do bloco OPC read



Fonte: Matlab/Simulink