

ACCEPT – Aplicações Web para visualização e análise de dados de chão de fábrica

ACCEPT – Web Applications for visualization and analytics of shop floor data

Eduardo Miguel Ascenso Godinho

Centro de Investigação em Informática e Comunicações
Escola Superior de Tecnologia e Gestão de Leiria
Instituto Politécnico de Leiria
Leiria, Portugal
2171960@my.ipleiria.pt

Maria Beatriz Piedade

Centro de Investigação em Informática e Comunicações
Escola Superior de Tecnologia e Gestão de Leiria
Instituto Politécnico de Leiria
Leiria, Portugal
beatriz.piedade@ipleiria.pt

Resumo — No contexto atual da Indústria 4.0 e do controlo metrológico de produtos pré-embalados, procura-se que os dados provenientes de chão de fábrica sejam recolhidos de forma automática e que diferentes tipos de dispositivos digitais tenham acesso aos mesmos dados de forma ubíqua. Pretendeu-se desenvolver e integrar aplicações Web no sistema ACCEPT, sistema que permite a recolha, armazenamento, visualização e análise de dados de chão de fábrica. Neste artigo é descrito o processo de desenvolvimento das aplicações ACCEPT Quality Hub e ACCEPT Analytics e a sua integração neste sistema.

Palavras Chave – metrologia; dados de chão de fábrica; dashboard; analytics; aplicações Web.

Abstract — In the current context of Industry 4.0 and the metrological control of prepackaged products, it is intended that the shop floor data can be automatically collected and that different types of digital devices have access to the same data ubiquitously. It was intended to develop and to integrate Web applications on the ACCEPT system, a system that allows the collection, storage, visualization and analysis of shop floor data. This article describes the development process of the ACCEPT Quality Hub and ACCEPT Analytics applications and their integration in this system.

Keywords – metrology; shop floor data; dashboard; analytics; Web applications.

I. INTRODUÇÃO

O papel da metrologia na indústria é o de manter uma rastreabilidade das características dos produtos para garantir a sua fiabilidade e qualidade. [1]. Uma gestão eficaz dos meios de medição favorece as negociações com o cliente pela confiança acrescida que induz, sendo, por isso, um diferenciador positivo no domínio tecnológico e comercial. Atualmente existem um conjunto de normas legais que as empresas da área dos pré-embalados têm de cumprir. As empresas para terem controlo sobre os produtos, seja sobre a sua qualidade ou enchimento, têm de recolher, registar e analisar os dados do chão de fábrica, ou seja, os dados provenientes do ambiente industrial onde se encontram as linhas de produção. É neste contexto que foi desenvolvido o

sistema ACCEPT, sistema que permite automatizar o processo de recolha de dados de chão de fábrica, o armazenamento de dados e a sua visualização e análise [2].

No contexto atual da Indústria 4.0, pretende-se que diferentes tipos de dispositivos digitais tenham acesso aos mesmos dados de forma ubíqua [3]. Para que o sistema ACCEPT possa responder aos desafios tecnológicos subjacentes a este contexto foi necessário desenvolver várias aplicações Web para este sistema.

Neste artigo é apresentado e descrito o processo de desenvolvimento de duas aplicações Web, a aplicação ACCEPT Quality Hub e a aplicação ACCEPT Analytics. São também apresentados conceitos sobre a área da metrologia, sobre o controlo metrológico de pré-embalados e sobre a área de *analytics*. É ainda apresentado um estudo sobre aplicações Web na área de *analytics*. Este estudo teve como objetivo selecionar uma aplicação para ser integrada no sistema ACCEPT.

Este artigo está organizado da seguinte forma, a secção seguinte apresenta conceitos sobre a área da metrologia e sobre a área de *analytics*; na secção III é apresentada a arquitetura do sistema ACCEPT; na secção IV são descritos aspetos da fase de implementação e sobre os testes realizados, e, por último, na secção V é apresentada uma conclusão sobre o trabalho desenvolvido e a desenvolver no futuro.

II. ENQUADRAMENTO

A. Enquadramento concetual

1) Metrologia

A metrologia pode ser entendida decompondo a palavra em dois termos: “metro” exprime medição e “logia” significa conhecimento, estudo ou ciência, ou seja, a metrologia refere-se ao estudo da medição. O Instituto Português da Qualidade define a metrologia, no Vocabulário Internacional de Metrologia, como a ciência da medição e suas aplicações, englobando todos os aspetos teóricos e práticos da medição, qualquer que seja a incerteza de medição e o campo de aplicação [4].

2) Controlo metrológico de pré-embalados

Um produto pré-embalado é um produto embalado na ausência do consumidor, seja da área alimentar ou não, e que tenha declarada na sua embalagem uma quantidade nominal com um valor constante, desde que essa quantidade esteja compreendida entre 5g e 10kg ou 5mL e 10L [5]. Um produto não é considerado um pré-embalado se a quantidade declarada variar entre cada produto embalado, como, por exemplo, algumas embalagens de carne, queijo, etc., onde o preço pago depende do peso individual de cada embalagem, embora também seja embalado na ausência do consumidor.

O controlo metrológico de pré-embalados serve para garantir ao consumidor que, em média, cada embalagem contém o peso, ou volume de produto, nele declarado. O decreto-lei 291/90 de 20 de setembro obriga o cumprimento do controlo metrológico legal [6]. A realização das verificações metrológicas de pré-embalados é realizada por Organismos de Verificação Metrológica (OVM).

O processo de verificação consiste em utilizar instrumentos de medição nos produtos (picnómetros, balão volumétrico, entre outros), registar as medições e verificar se estão de acordo com as normas do embalador, que são:

- O conteúdo médio de um produto pré-embalado não deverá ser inferior à quantidade nominal nele marcada.
- Não deverá existir mais do que 1 embalagem em cada 40 (2,5%) que tenha uma quantidade inferior ao valor nominal menos o Erro Admissível por Defeito (EAD). Este erro varia de acordo com o valor declarado na embalagem e encontra-se no Quadro N.º 1 da Portaria 1198/91 de 18 de dezembro.
- Nenhum pré-embalado deverá conter uma quantidade inferior ao valor nominal menos o dobro do EAD [7].

Depois do controlo metrológico ser aprovado, é emitido um certificado e a possibilidade de colocar a marca de conformidade (Fig. 1) no rótulo do produto medido.

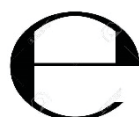


Fig. 1 - Marca de conformidade "e" de produtos pré-embalados

3) Analytics

Analytics é a prática de análise e interpretação de dados. O seu objetivo é permitir um processo de tomada de decisão mais eficiente. Pode ser aplicada em várias áreas de negócio e está ligada à área de *Business Intelligence* [8].

Na área de *Analytics* é frequente o recurso a visualização de dados. Estas visualizações podem ser através de tabelas ou de gráficos tais como gráficos de barras, gráficos de linhas, gráficos circulares, entre outros. Um conjunto de várias formas de representações de dados num só ecrã forma um *dashboard*. Os *dashboards* expõem dados que podem estar a ser atualizados em tempo real e podem ainda permitir que os mesmos possam ser filtrados, se forem interativos e tiverem

outros elementos presentes para o fazer, como caixas de texto ou de seleção.

Existem várias aplicações Web de *analytics* de acesso livre que possibilitam a criação de *dashboards*. A secção seguinte apresenta um estudo comparativo de várias aplicações.

B. Aplicações Web de analytics

Por forma a integrar no sistema ACCEPT uma aplicação Web de *analytics*, foi efetuada uma pesquisa e um estudo de aplicações Web de acesso livre existentes nesta área.

Da pesquisa realizada, foram destacadas três aplicações: Redash¹, Metabase² e Grafana³. Para cada uma destas aplicações, foi criado um *dashboard* com painéis de filtros, com painéis de visualização de contagens, com tabela de dados e com gráficos de linhas. O objetivo foi elaborar-se um estudo comparativo que permitisse selecionar a aplicação a utilizar no sistema ACCEPT. Na Tabela 1 é apresentada uma lista de funcionalidades com a indicação da aplicação em que existem.

TABELA 1 – LISTA DE FUNCIONALIDADES DAS APLICAÇÕES

	Redash	Metabase	Grafana
Zoom nos gráficos.	✓	-	✓
Opções de filtragem direta de dados.	-	✓	✓
Criação de consultas com ambos SQL e seleção de campos por clique.	-	✓	-
Criação de <i>plugins</i> para painéis personalizados.	-	-	✓
Embutir <i>dashboard</i> noutra aplicação.	✓	✓	-
Embutir aplicação completa noutra aplicação.	✓	-	✓
API REST para controlar a aplicação através de outras aplicações.	-	-	✓
Permissões de utilizadores para limitar funcionalidades.	✓	-	✓
Apontar erros explícitos, na criação de consultas SQL.	✓	✓	-

Verifica-se que a aplicação Grafana tem mais vantagens e que estas são uma mais-valia. Por exemplo, permite que sejam criados *plugins* de painéis personalizados, o que dá muitas mais possibilidades de representação de dados e é importante para o caso de existirem clientes que solicitem gráficos à medida. Também tem uma API REST completa e documentada que possibilita a sua interação a partir de outra aplicação, o que facilita também a sua integração no sistema ACCEPT. Por estes motivos, foi selecionada a aplicação Grafana para ser a aplicação a utilizar para criação de *dashboards* e para servir como base da aplicação ACCEPT Analytics.

¹ <https://github.com/getredash/redash>

² <https://github.com/metabase/metabase>

³ <https://github.com/grafana/grafana>

III. ARQUITETURA DO SISTEMA ACCEPT

A Fig. 2 apresenta o esquema da arquitetura do sistema ACCEPT onde se encontra representada a proveniência dos dados (Chão de fábrica), onde são armazenados os dados (Base de dados SQL Server) e onde são apresentados ao utilizador (Aplicações cliente).



Fig. 2 - Esquema da arquitetura do sistema ACCEPT

O sistema assenta sobre uma arquitetura cliente-servidor em ambientes Microsoft Windows e Android. Os dados são provenientes das linhas de produção (chão de fábrica), sejam diretamente ou através de balanças, e são armazenados numa base de dados Microsoft SQL Server. Esta ligação entre linhas de produção e base de dados é feita através de um servidor que consiste num sistema com uma série de serviços que fazem a conexão entre as máquinas de fábrica e a base de dados [2].

As aplicações cliente estão subdivididas em aplicações Web, Desktop ou Mobile. Estas ligam-se a uma API intermédia para comunicar com a base de dados através de pedidos REST (*Representational State Transfer*). Foram apenas desenvolvidas aplicações Web, na *framework* Nuxt.js, que se ligam a uma API e base de dados já existentes. Esta ligação é feita através de uma biblioteca JavaScript chamada Axios, que é um cliente HTTP que corre em *browsers* e permite fazer pedidos REST.

IV. IMPLEMENTAÇÃO DAS APLICAÇÕES WEB

A. ACCEPT Quality Hub

O ACCEPT Quality Hub é uma aplicação Web que integra todas as aplicações Web do sistema ACCEPT, permitindo a autenticação de utilizadores e o posterior acesso às aplicações.

1) Layout e lista de aplicações

A Fig. 3 apresenta a página de início do Quality Hub, onde é apresentada uma lista de aplicações. A mesma lista também está acessível num menu do cabeçalho (também visível na mesma figura). Este menu facilita a mudança de aplicação a partir de qualquer página.

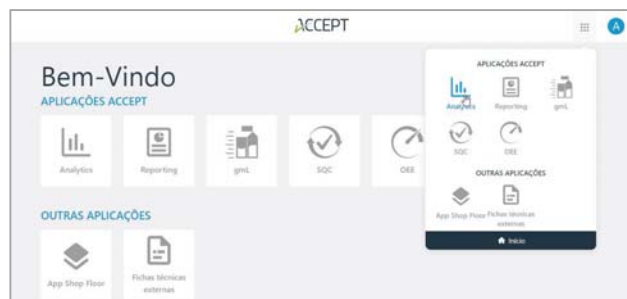


Fig. 3 - Página de início do Quality Hub com a lista de aplicações

2) Integração e carregamento de aplicações

A integração e carregamento de aplicações no *Quality Hub* foi um dos maiores desafios deste Projeto, pois não existia documentação nem exemplos funcionais para implementar esta solução de forma dinâmica e escalável.

O processo de implementação iniciou-se através da análise de um conjunto de aspetos, nomeadamente a análise de como um ficheiro estático HTML pode carregar uma aplicação Nuxt.js (ficheiro “index.html”) e também estudar o funcionamento do módulo Webpack. Este módulo empacota recursos JavaScript em ficheiros de *build* para poderem ser lidos num *browser*. Uma *build* de uma aplicação Web é uma versão compilada da aplicação, que pode ser corrida para a servir em modo de produção. Depois desta análise, procedeu-se a implementação do carregamento das aplicações. A análise é descrita na secção seguinte.

a) Análise do ficheiro “index.html”.

Para se analisar como um ficheiro estático HTML pode carregar uma aplicação Nuxt.js num *browser*, foi feita uma *build* de uma aplicação de teste, para esta ser analisada. Nesta *build* foi gerada a pasta “dist”, que contém todo o conteúdo necessário para a aplicação correr em modo de produção. O ficheiro de entrada nessa pasta que é lido por um servidor para servir a aplicação é o “index.html”. Esse ficheiro permite abrir a aplicação no *browser* de forma estática. Por isso, concluiu-se que, para carregar aplicações no Quality Hub, basta replicar de forma dinâmica o que faz esse ficheiro. Assim, foi analisado o ficheiro “index.html” e verificou-se que está a referenciar vários ficheiros JavaScript e um CSS. A Fig. 4 apresenta um excerto do conteúdo do ficheiro “index.html”, em que estão assinalados estes ficheiros através dos seus nomes.



Fig. 4 - Excerto do conteúdo do ficheiro “index.html”

Esses ficheiros são os ficheiros de *build* que o Webpack gerou ao empacotar toda a aplicação. Os ficheiros dos *scripts* têm nomes aleatórios e o Webpack cria-os para separar código em partes mais pequenas, com o objetivo de diminuir o tempo de carregamento deles no *browser* ao carregar à vez apenas os necessários [9]. Ainda no mesmo ficheiro “index.html”, também existe um elemento HTML com o ID “__nuxt”. Esse elemento HTML é o elemento de entrada onde a *framework* carrega a aplicação.

Com esta informação obtida, procedeu-se a implementação do sistema de carregamento de aplicações que funciona da mesma forma, a carregar ficheiros de *build* e a definir um elemento HTML com um ID para construir a aplicação.

b) Implementação do carregamento das aplicações

Teve de ser tido em conta o facto de os ficheiros dos *scripts* terem nomes aleatórios indetermináveis e que um *browser* não pode aceder ao sistema de ficheiros de um computador para saber o nome deles para os carregar. Por isso, antes de se fazer *build* de uma aplicação para ser integrada no Quality Hub, é necessária uma configuração específica do módulo Webpack e da *framework* de cada aplicação. Relativamente ao módulo Webpack, é necessário desativar as opções de divisão de ficheiros, com o objetivo de tornar os nomes fixos dos ficheiros de *build* de uma aplicação, que são “app.js”, “runtime.js” e “vendors.app.js”. De resto, foram configuradas novas propriedades como, por exemplo, de caminhos de ficheiros.

Para se poderem carregar ficheiros de *build* de uma aplicação numa página do Quality Hub, foi desenvolvido um *script*, que permite criar 3 elementos HTML de *scripts* e 1 de folha de estilos que têm os caminhos apontados para os ficheiros em questão. O *script* recebe por parâmetro o nome da aplicação que se quer abrir e referencia os ficheiros da pasta dessa aplicação. Quando o *script* é executado, são carregados os ficheiros de *build* da aplicação e assim esta é carregada sobre o ACCEPT Quality Hub.

A Fig. 5 mostra o resultado do HTML do Quality Hub quando é carregada uma aplicação chamada OEE. Nesta figura verificam-se os caminhos para os ficheiros de *build* da pasta da aplicação, o elemento HTML de entrada onde a aplicação é carregada e elementos do Quality Hub (elemento de entrada e cabeçalho). O resultado é a aplicação aberta com o cabeçalho do Quality Hub visível.



Fig. 5 - HTML resultante do Quality Hub com a aplicação OEE aberta

c) Integração de uma aplicação

No Quality Hub, para se adicionar uma aplicação nova, basta criar uma página com o nome da nova aplicação e importar o *script* descrito na secção anterior. Depois cria-se uma pasta com o nome da nova aplicação na pasta das aplicações. Na aplicação a ser integrada, depois de ter sido feita a configuração descrita anteriormente, faz-se a sua *build* e coloca-se a pasta resultante “dist” no Quality Hub.

Em suma, este sistema de carregamento de aplicações tem grandes vantagens que são importantes para a escalabilidade das aplicações Web do sistema ACCEPT, que são as seguintes:

- Instalação de uma nova aplicação é rápida visto que basta criar-se uma página nova no Quality Hub com o nome dela e uma pasta com o mesmo nome;
- Atualização de uma aplicação demora menos que um minuto, visto que basta substituir-se a pasta da *build*.

- Atualização de uma aplicação não implica reiniciar o serviço do Quality Hub, mantendo toda a sua disponibilidade, porque os ficheiros são carregados no momento de abertura de uma aplicação;
- A abertura de uma aplicação não atualiza a página do *browser*, mantendo assim o conceito das SPA;
- Autenticação e contas de utilizadores são iguais para todas as aplicações porque é o Quality Hub que gere isso;
- A aplicação integrada mantém a possibilidade de correr normalmente de forma independente fora do Quality Hub;
- Um programador pode estar a desenvolver a aplicação fora do Quality Hub e evita conflitos e *merges* de código com outras aplicações.

3) Estilos das aplicações

De modo a manter a consistência de estilos entre as aplicações, para aproximar o design dos componentes com o que era pretendido e para acelerar o processo de desenvolvimento, foram definidos estilos iguais entre as aplicações. Para os componentes de introdução de dados, foi utilizada a biblioteca de componentes Vue, Element, visto que tem uma grande capacidade de configuração, personalização e a possibilidade de criação de um tema como o que foi feito.

Relativamente a componentes dinâmicos e genéricos em comum nas aplicações integradas, os componentes criados foram um menu lateral de mudança de páginas, um *breadcrumb* (Fig. 6), janela modal e caixa de notificação.

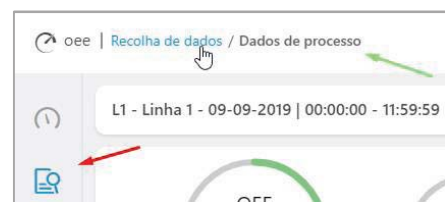


Fig. 6 - Breadcrumb e menu lateral numa aplicação dentro do Quality Hub

B. ACCEPT Analytics com o Grafana

O ACCEPT Analytics é uma aplicação Web que integra a aplicação Grafana, que permite a visualização e análise de dados através de *dashboards*.

1) Integração da aplicação Grafana

Esta aplicação tem como página inicial uma lista de pastas e *dashboards* (Fig. 7), que dão acesso direto aos *dashboards* no Grafana. Estes podem ser pesquisados através da caixa de pesquisa no topo da página ou também acedidos pelo menu dos favoritos.

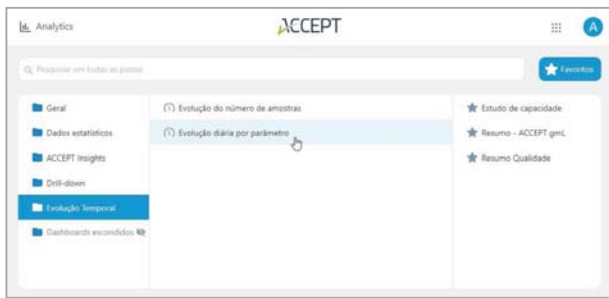


Fig. 7 - Página inicial do ACCEPT Analytics

A lista de pastas e *dashboards* é obtida com um pedido “GET” ao método de lista de *dashboards* da API do Grafana.

Um *dashboard* do Grafana é aberto ao selecionar-se numa das opções. Para isso, o Grafana está a funcionar numa *iframe* escondida por baixo da página. As *iframes* são *tags* HTML que servem para embutir documentos HTML dentro de *websites*. Para o Grafana estar a funcionar na *iframe*, tem de estar a correr num servidor no mesmo domínio mas noutra porta, porque tem um servidor e base de dados independentes. Para a se colocar a aplicação a correr como serviço, foi utilizada a ferramenta NSSM, que é um gestor de serviços Windows.

Ao clicar-se num *dashboard* da lista, o URL da *iframe* é alterado para o URL direto do *dashboard*. Enquanto está a carregar a nova página, a *iframe* continua escondida e é mostrado um *loading* no lado do Quality Hub. A Fig. 8 apresenta um *dashboard* aberto e o botão “Voltar” no canto superior esquerdo, que serve para voltar à lista de *dashboards*.

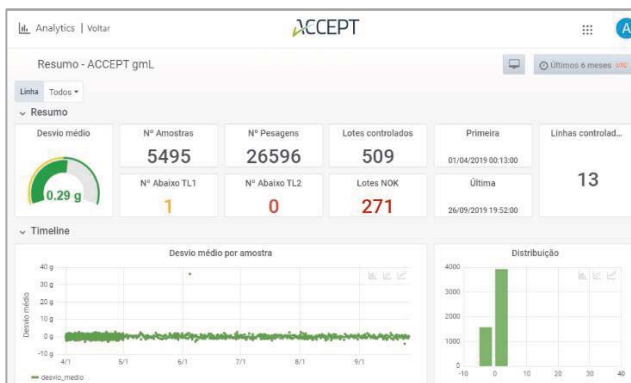


Fig. 8 - Dashboard aberto no ACCEPT Analytics

2) Autenticação

Um dos maiores desafios para integrar o Grafana foi a autenticação de utilizadores na aplicação a partir do Quality Hub, para o utilizador autenticado ser o mesmo de ambos os lados. O Grafana tem o seu próprio sistema de autenticação numa página de *login*, no entanto, não tem métodos na sua API para autenticar um utilizador remotamente pelo Quality Hub. E para se poderem usar os métodos da API para obter a informação, é preciso primeiro um utilizador estar já autenticado.

A solução foi passar o nome de utilizador e *password* pelos parâmetros de *query* do URL da *iframe*. Do lado do Grafana foi alterada a função de *login* para obter esses parâmetros a partir

do URL. Para a *password* não ir em texto limpo no URL da *iframe*, esta é encriptada no lado do Quality Hub e desencriptada do lado do Grafana. Para a encriptação foi utilizada a cifra de XOR em JavaScript, que utiliza uma chave única que só ambos os lados conhecem. Para aumentar a segurança, as instalações do Quality Hub e Grafana foram feitas com configuração HTTPS (*Hyper Text Transfer Protocol Secure*) nos servidores dos clientes. Assim, o URL com dados de autenticação não é lido por atacantes que estejam a fazer *sniffing* à rede.

3) Criação de dashboards

Foram criados diversos *dashboards*, sendo que, em cada instalação do sistema ACCEPT nos clientes, foram disponibilizados *dashboards* genéricos e à medida do cliente. A Fig. 9 apresenta um desses *dashboards*.



Fig. 9 - Exemplo de um dos dashboards criados

4) Criação e alteração de funcionalidades

Foi necessário criar e alterar algumas funcionalidades no Grafana. Toda a implementação foi feita analisando e alterando o código fonte do Grafana, que está desenvolvido em AngularJS e ReactJS. As novas funcionalidades implementadas foram:

- *Drill-down* em gráficos de barras: permite navegar para outro *dashboard* ao clicar numa barra de gráfico;
- Linhas verticais em gráficos: permite a criação de linhas para se visualizarem limites nos gráficos;
- Estilo de cores e tema do Grafana: para ter o mesmo esquema de cores do sistema ACCEPT;
- Traduções: todo o conteúdo apresentado ao utilizador final traduzido para português;
- Novo tipo de variável que devolve nome de utilizador: para se parametrizarem consultas pelo utilizador;

- Botões de alterar o tipo de gráfico: permite ao utilizador mudar o tipo de visualização de um gráfico para linhas, barras ou pontos;
- Exportação de painéis como imagens: permite ao utilizador exportar qualquer painel.

5) Instalação e alteração de Plugins

Os *plugins* do Grafana são extensões que se podem adicionar à aplicação. Existem *plugins* oficiais, ou seja, criados pela equipa de desenvolvimento da aplicação, Grafana Labs, ou criados pela comunidade.

a) Instalação

Para se instalar um *plugin*, basta colocar-se a pasta do *plugin* na pasta “plugins” do Grafana e reiniciar o serviço da aplicação. Os *plugins* instalados foram:

- Carpet plot: gráfico de mapa de calor;
- Statusmap: mapa de estados;
- Pie Chart: gráfico de pizza;
- Plotly: gráfico que utiliza a *framework* Plotly.js;
- Traffic Lights: painel com semáforos.

b) Alterações no *plugin* Plotly e no *plugin* Statusmap

Este *plugin* utiliza a *framework* Plotly.js, que é uma *framework* bastante extensa de gráficos. A *framework* permite uma grande configuração dos gráficos, no entanto, este *plugin* não permite grande parte das configurações que a *framework* disponibiliza. Por isso, foram adicionadas novas funcionalidades alterando o seu código fonte. As novas funcionalidades implementadas permitem:

- A definição do número de máximo de *labels* num gráfico de barras para evitar que fiquem sobrepostas, definição do ângulo de rotação das *labels* e a definição de margem extra para as *labels*, para o caso em que sejam compridas;
- A mudança de tipo de gráfico para múltiplas séries no mesmo painel: para permitir haver gráficos de linhas e barras no mesmo painel;
- Zoom através do rato no gráfico para alteração da escala dos eixos.

Outro *plugin* alterado foi o Statusmap, sendo que só foi feita uma alteração, que foi a de implementar a funcionalidade de *drill-down* nas células do mapa de calor.

C. Testes

Para assegurar a qualidade de uma aplicação antes desta ser colocada em produção, é necessário testá-la. Por isso, foram sempre feitos testes à medida que se iam desenvolvendo as aplicações, sendo estes os testes unitários. Os testes unitários consistem em validar dados de entrada, para verificação de falhas no resultado através dos dados de saída.

Quando as aplicações foram colocadas em produção nos servidores de clientes, os utilizadores realizaram testes de usabilidade. Os testes de usabilidade têm por objetivo verificar a facilidade de compreensão do fluxo de operação de uma

aplicação por parte de um utilizador. Caso fosse detetada alguma situação menos bem conseguida, esta era revista, tendo em conta o *feedback*, e resolvida rapidamente. A atualização da *build* de uma aplicação integrada no Quality Hub é feita em menos de 1 minuto, visto que basta colocar-se a *build* no projeto do Quality Hub do servidor do cliente, sem ser preciso reiniciar algum serviço.

V. CONCLUSÃO E TRABALHO FUTURO

No desenvolvimento deste Projeto, destaca-se o estudo de conceitos relacionados com a área da metrologia, bem como o seu papel na Indústria 4.0. Relativamente às aplicações Web desenvolvidas, foi relevante terem sido desenvolvidas novas funcionalidades para uma aplicação de grande notoriedade de criação de *dashboards*, a aplicação Grafana. Também se destaca o contributo no desenvolvimento de uma solução nova de integração de aplicações, mantendo a possibilidade de poderem correr independentemente de estarem, ou não, no Quality Hub. Destaca-se ainda a colocação em produção desta solução em ambiente industrial, o que possibilitou a empresas nacionais melhorar o processo de tomada de decisão, otimizar processos industriais do chão de fábrica e a garantir o cumprimento dos critérios legais do controlo metrológico de pré-embalados.

No que se refere a desenvolvimento futuro, tendo em conta a implementação da aplicação Quality Hub, o objetivo é poder criar-se novas aplicações Web à medida do cliente e efetuar a sua integração. Por fim, na aplicação Grafana poderão ser desenvolvidas funcionalidades novas na aplicação ou em *plugins* porque também há a possibilidade de se quererem novos tipos de gráficos ou painéis à medida.

AGRADECIMENTOS

Este trabalho foi financiado por Fundos Nacionais através da FCT – Fundação para a Ciência e a Tecnologia, I.P., no âmbito do projeto UID/CEC/04524/2020.

REFERÊNCIAS BIBLIOGRÁFICAS

- [1] E. Christian Raffaello Baldo, “A interação entre o controle de processos e a metrologia em indústrias de manufatura,” Tese de Mestrado, Universidade Federal de Santa Catarina, 2003.
- [2] E. Godinho, “ACCEPT – Aplicações Web para análise e visualização de dados de chão de fábrica,” Relatório de Projeto, IPEL - Escola Superior de Tecnologia e Gestão de Leiria, 2019. [Online]. Available: <http://hdl.handle.net/10400.8/4507>.
- [3] Stock, Tim, and Günther Seliger, “Opportunities of sustainable manufacturing in industry 4.0,” 2016.
- [4] IPQ, Vocabulário Internacional de Metrologia (VIM), 2012.
- [5] Portaria 1198/91 de 18 de dezembro, 1991.
- [6] Decreto-lei 291/90 de 20 de setembro, 1990.
- [7] Decreto-Lei 199/2008 de 8 de outubro, 2008.
- [8] M. Rouse, “business analytics (BA),” [Online]. Available: <https://searchbusinessanalytics.techtarget.com/definition/business-analytics-BA>.
- [9] Webpack, “Code Splitting | Webpack,” [Online]. Available: <https://webpack.js.org/guides/code-splitting/>.