

UNIVERSIDADE DE LISBOA  
FACULDADE DE CIÊNCIAS  
DEPARTAMENTO DE INFORMÁTICA



## **Componentes Web/Mobile inovadores para soluções de gestão e distribuição de energia**

João Miguel Roque Mourão

**Mestrado em Engenharia Informática**  
Especialização em Engenharia de Software

Trabalho de projeto orientado por: Dr. Carlos Eduardo Ramos dos Santos  
Lourenço  
e  
Eng.º. Nuno Santos

2017



## Agradecimentos

Quero começar por agradecer à minha família, nomeadamente aos meus pais, irmãos, e sobrinhos, assim como igualmente aos meus amigos, toda a força com que procuraram sustentar os meus desânimos e toda a paciência demonstrada nos últimos tempos, sobretudo nos momentos em que manifestei uma notória impaciência e pouca disponibilidade para os escutar, sendo que todos eles, de uma forma ou de outra, me prestaram uma preciosa e inestimável ajuda no que diz respeito a enfrentar todos os contratemplos que foram surgindo aquando da concretização deste projeto. Apesar de na altura as suas sugestões e pequenas “intromissões” me terem parecido, por vezes, pouco convenientes, e até, talvez, inoportunas, hoje, no entanto, observando em retrospectiva, reconheço que se vieram a revestir de extrema importância, pois serviram para definir e redefinir prioridades sobre quem quero ser de hoje em diante.

À CGI, pois embora muitos dos seus elementos já me conhecessem, receberam-me novamente de braços abertos, possibilitando a realização deste projeto. É certo que, atendendo a todas as turbulências e reviravoltas subjacentes à organização deste trabalho, me vi na eminência de incomodar muita gente, desde o meu chefe, Nuno Santos, à equipa do Departamento Financeiro e às minhas queridas RH's, sendo, por isso, meu ensejo a todos agradecer a inestimável paciência e a perseverança. A todos eles, que carinhosamente me apelidaram de “O chato”, devo, indubitavelmente, o sucesso da concretização desta tese, ficando a dever-lhes uma sangria de espumante.

À FCUL, por todo o processo de transmissão de conhecimento, fulcral para o meu desenvolvimento como pessoa, aluno e profissional, onde, como é claro, aprendi muito a nível técnico, mas também a nível pessoal.

À Ana Filipa Martins, fiel companheira de ginásio, por toda a boa vontade, demonstrada em todos os momentos, especialmente naqueles em que me ouviu dizer mais de mil vezes a frase “tenho de escrever este relatório de projeto”, incentivando-me incansavelmente, mesmo quando a minha vontade de procrastinar se sobrepunha a tudo o resto.

A todos, mesmo os não referidos, mas cujo contributo se revelou prestimoso, o meu muito obrigado.



*O valor das coisas não está no tempo que elas duram,  
mas na intensidade com que acontecem,*

*por isso existem momentos inesquecíveis,  
coisas inexplicáveis e pessoas incomparáveis.*

*Maria Júlia Paes de Oliveira*



## Resumo

A regulação no setor das energias implica que sejam cumpridos certos requisitos e níveis de serviços prestados ao cliente final. Estes requisitos são impostos às entidades de gestão pelas entidades reguladoras, o que faz com que o não cumprimento destes níveis de serviço, resulte, muitas vezes, em coimas bastante pesadas.

Em Portugal, estas regulações e imposições de níveis de serviços prestados, devem-se à classificação atribuída de serviços essenciais às provedoras de energia (Distribuição de Água, Gás e Eletricidade), pois alguns clientes residenciais prestam serviços de saúde e necessitam de ser classificados como clientes prioritários/críticos. O mesmo é aplicado a serviços que, por falta da mesma, podem ter prejuízos elevados, requerendo então garantias para que lhes seja possível a exequibilidade desses serviços.

Devido a elevados critérios aplicados a essa unidade de negócios, era necessário então prover as entidades fornecedoras com as ferramentas, com as soluções e com os mecanismos corretos para lhes permitir responder a todos esses critérios e, conseqüentemente, para evitar coimas e lhes tornar possível prestar um serviço de qualidade superior.

Este relatório de projeto de mestrado focar-se-á nesse aspeto. *Diamond* constitui uma solução que irá permitir a gestão de ocorrências e de todos os problemas com elas relacionados.

A solução, iniciada ainda quando o atual grupo CGI pertencia ao grupo *Lógica*, começou a ser embrionariamente planeada e ulteriormente desenvolvida em 2010, só chegando, no entanto, a ver a luz do dia em 2012. A solução foi desenvolvida inicialmente em *Silverlight*, utilizando o componente *Bing Maps SilverLight* como gestor de cartografia base e Bases de dados *Oracle* como base de dados de armazenamento de todas as informações relacionadas com as ordens de serviço e ocorrências.

Os dias de hoje, passado um considerável número de anos, seguindo a premissa de ciclo de vida tecnológico, que especifica, aproximadamente de 3 a 5 anos até que uma tecnologia esteja ultrapassada ou morta, assumem-se como a altura de redesenhar, repensar, reimplementar e melhorar a solução, adicionando também novas funcionalidades para permitir que a solução se mantenha atual e que se adapte às novas necessidades que os novos tempos assim exigem.

**Palavras-chave:** SIG, IoT, Web, Mobile, Energias, KISS





# Abstract

The regulations in the Utilities sector, implies providing certain requirements and service levels towards the final client, this demands are imposed to the distributor entities by regulatory entities. Not achieving/accomplish these levels will often results in heavy fines.

In Portugal, these regulations are imposed due to distributor entities being considered as an essential service (Distribution of Water, Gas and Electricity), being that, some of the residential clients provide health care services and are classified as critical/priority services, requiring warranties to fulfil these services without interruption.

Due to the high standards applied to this business area, it was necessary to provide the distributor entities with capable tools, systems and mechanisms that would allow them efficiently respond to any interruptions and be able to meet the standards and surpass them as well.

The main focus of this Project Report, is *Diamond*, a solution that will be able to manage incidents, improving service levels, providing customers a proper and reliable service, assisting the distributor entities avoiding fines.

The solution started being planned and developed back in 2010, and only seeing the day of light in 2012, developed initially in Silverlight, and using Bing maps SilverLight as the base cartography, and map management system with oracle databases.

After some years and with the premise of technology life cycle, being in average from 3-5 years, it was necessary to redesign the solution, and, from scratch re-implement the old functionalities, improving them and bring some new ones to develop an improved solution. The Market was analysed, existing clients were questioned and a new solution was born to keep *Diamond* innovative and using the latest and most cutting edge technologies nowadays.

**Keywords:** GIS, IoT, WEB, UTILITIES, Mobile, KISS



# Conteúdo

Capítulo 1	Introdução .....	1
1.1	Motivação .....	1
1.2	Objetivos .....	2
1.2.1	Diamond Web .....	2
1.2.2	Diamond Web Manager Mobile .....	2
1.2.3	Diamond Field Agent .....	2
1.3	Estrutura do documento .....	3
Capítulo 2	O projeto .....	5
2.1	Sobre a CGI .....	5
2.2	Solução Diamond .....	5
2.3	Objetivos inicialmente pretendidos .....	5
2.4	Características associadas ao produto .....	7
2.5	Riscos .....	8
2.6	Competição no mercado .....	8
2.6.1	Estudo de mercado .....	8
2.6.2	Competição direta .....	9
2.7	Metodologia Agile – Scrum .....	12
2.8	Trello .....	13
2.9	Desenvolvimento .....	14
2.9.1	Equipa de desenvolvimento .....	14
2.9.2	Ferramentas de desenvolvimento .....	15
2.10	Ambiente Aplicacional .....	16
2.10.1	Requisitos Mínimos – Cliente .....	16
2.10.2	Requisitos – Servidor .....	17
2.11	Arquitetura da solução .....	18
2.11.1	Módulos internos .....	18
2.11.2	Módulos externos .....	19
2.11.3	Arquitetura e padrão de desenvolvimento .....	19
2.12	Evolução da Solução Diamond .....	21
2.13	Planeamento inicial e trabalho final .....	21
Capítulo 3	Trabalho Realizado .....	24
3.1	Diamond Web .....	24
3.2	Diamond Web Manager Mobile .....	28
3.3	Diamond Field Agent .....	32
3.3.1	Projeto Upgrid .....	32
3.3.2	Descrição e demonstração de funcionalidades .....	33
3.3.3	Demonstração .....	34
3.3.4	Modo colaborativo de desenho .....	37

Capítulo 4	Conclusões .....	41
4.1	Trabalho Desenvolvido.....	41
4.2	Dificuldades .....	41
4.3	Considerações Finais .....	42
4.4	Trabalho Futuro.....	42
Capítulo 5	Bibliografia .....	43

# Lista de Figuras

Figura 2.1 - Trello Scrum.....	13
Figura 2.2 -Trello específico para cliente .....	13
Figura 2.3 - Composição da equipa .....	14
Figura 2.4 - Ferramentas e tecnologias utilizadas .....	15
Figura 2.5 - Arquitetura da solução .....	18
Figura 2.6 - Arquitetura e tecnologia usada por camada .....	20
Figura 2.7 - Arquitetura Diamond Mobile .....	20
Figura 2.8 - Planeamento inicial (Alto nível).....	21
Figura 2.9 - Lista de esforço corrigido .....	22
Figura 2.10 - Calendário meio-termo .....	22
Figura 2.11 - Lista de esforço final.....	22
Figura 2.12 - Calendário Final .....	23
Figura 3.1 - Página inicial <i>viewer</i> com as <i>tile layers</i> ativas por defeito.....	25
Figura 3.2 - <i>Viewer</i> mostrando equipamento, <i>tile layers</i> desativadas .....	25
Figura 3.3 - Controlos <i>Viewer</i> .....	26
Figura 3.4 - Modo de rastreamento .....	27
Figura 3.5 - Página inicial .....	29
Figura 3.6 - Menu lateral .....	29
Figura 3.7 - Resultados do módulo de pesquisa .....	29
Figura 3.8 - Exibição de detalhes de chamada .....	29
Figura 3.9 - Página Inicial Listagem.....	30
Figura 3.10 - Menu lateral.....	30
Figura 3.11 - Resultados da pesquisa .....	31
Figura 3.12 - Exibição de detalhes da chamada (1/3).....	31
Figura 3.13 - Exibição de detalhes da chamada (2/3).....	31
Figura 3.14 - Exibição de detalhes da chamada (3/3).....	31
Figura 3.15 - Página inicial <i>Diamond Field Agent</i> .....	34
Figura 3.16 - Informação de ponto e lista de ocorrências .....	35
Figura 3.17 - Informação de ponto, StreetView e Esquema interno.....	36
Figura 3.18 - Informação de camadas e <i>clustering</i> de pontos .....	37
Figura 3.19 - Página inicial do desenho colaborativo.....	39
Figura 3.20 - Representação de uso do desenho colaborativo.....	40



# Lista de Tabelas

Tabela 2.1 - Comparativo de produtos concorrentes .....	11
Tabela 3.1 - Simbologia relevante.....	28





# Capítulo 1 Introdução

## 1.1 Motivação

O Mestrado em Engenharia Informática, na sua vertente de Engenharia de *Software*, contém um módulo, em que, aquando da frequência do segundo ano, proporciona ao mestrando a oportunidade de poder participar como estagiário em contexto de projeto, realizado numa entidade externa (como por exemplo numa empresa). A inclusão desta componente no Mestrado constitui, sem quaisquer dúvidas, uma mais-valia no que concerne ao alcance de um melhor desenvolvimento do carácter profissional de um aluno, proporcionando um contacto com novas situações através da exposição a uma realidade completamente diferente e de cariz mais pragmático, que é a que caracteriza o mundo empresarial, constituindo esta um passo fundamental para um futuro profissional de excelência, inevitavelmente consolidado pelo percurso na Faculdade de Ciências da Universidade de Lisboa.

Sendo a CGI uma empresa de origem canadiana são cada vez mais evidentes as sinergias de um grupo que está presente em 40 países, entre os quais Portugal, país em que, desde 2012, a multinacional está a levar a tecnologia nacional por vários continentes.

Assim, com um leque de mais de 4500 clientes nos vários setores de atividade em que opera, e com os quais trabalha em estreita parceria para melhor compreender as suas efetivas necessidades e assim poder adaptar os serviços e as soluções da empresa no sentido de potenciar o sucesso dos clientes esta é uma empresa dividida em inúmeras subsecções.

Com o intuito de poder concretizar o projeto que constitui objeto desta dissertação, todo o trabalho realizado ocorreu na subdivisão do Sudoeste Europeu, que contempla colaborações ativas entre Portugal, Espanha, Reino Unido, França e Itália.

Deste modo, a CGI Portugal propôs a minha integração na sua equipa de SIG (Sistema Informação Geográfico), assim como a minha participação na equipa de *design* e de implementação do seu produto *Diamond*, que se encontrava já numa altura de fim de vida tecnológico e que necessitava de ser repensado e de ser desenvolvido de raiz. O principal objetivo deste redesenho foi a adoção de tecnologias de ponta, atuais, com eficácia demonstrada, com adesão por parte da comunidade e, preferencialmente, de fonte aberta (*Open source*). Tudo isto com o intuito de manter uma relação qualidade/ preço o mais competitiva possível e de produzir uma solução que realize os propósitos do cliente final.

## **1.2 Objetivos**

A solução *Diamond* é composta por um número elevado de módulos, neste relatório de projeto apenas serão aprofundados três módulos.

### **1.2.1 Diamond Web**

O *Diamond Web* tem como principal objetivo, ser capaz de gerir eficientemente as ocorrências/incidentes, dotando os seus utilizadores de ferramentas que possibilitem que as mesmas sejam resolvidas cooperativamente entre as diferentes equipas e que a resolução dos mesmos seja efetuada da forma mais célere possível. Assim, cumprindo níveis de serviço [10][11][12] acordados e sendo possível também superá-los, realizando uma melhor gestão de recursos, aumentando a satisfação do cliente e consequentemente evitar as pesadas multas impostas pelas entidades reguladoras.

### **1.2.2 Diamond Web Manager Mobile**

O *Diamond Web Manager Mobile*, é um componente que dá apoio aos gestores das unidades de negócio, uma vez que os mesmos são de elevada importância, relevo e conhecimento. Sendo requeridos em inúmeros sítios simultaneamente, os mesmos nem sempre conseguem ter um computador à sua disposição para aceder ao *Diamond Web*, este componente vem assim colmatar esta lacuna, possibilitando que os gestores tenham acesso às principais funções do *Diamond Web*, mas de forma simplificada em qualquer *tablet* ou *smartphone* de forma a executar tarefas independentemente da sua localização.

### **1.2.3 Diamond Field Agent**

O *Diamond Field Agent*, tem como principal objetivo colmatar um nicho que a solução *Diamond* não contemplava até então, cobrindo assim mais um módulo importante. Tendo como alvo os agentes no terreno, equipas e trabalhadores responsáveis por resolver as ocorrências. Esta ferramenta tem o intuito de passar toda a informação requerida pelo agente para completar as suas tarefas, identificar o mais rapidamente possível onde se encontra e caso necessite esclarecer dúvidas.

Possibilitando a cooperação entre o agente e um engenheiro/gestor do despacho onde ambos possam partilhar informação a fim de resolver casos mais complexos.

### **1.3 Estrutura do documento**

Este documento está organizado da seguinte forma:

- Capítulo 1 – Contexto do trabalho, resumo do trabalho desenvolvido enquadramento institucional;
- Capítulo 2 – Apresentação em pormenor dos objetivos do trabalho, metodologia utilizada, planeamento comparação entre planeamento efetuado;
- Capítulo 3 – Trabalho realizado, ferramentas utilizadas, contribuição no trabalho;
- Capítulo 4 – Conclusões, sumário do trabalho realizado e comentário crítico e possibilidades de trabalho futuro;
- Capítulo 5 – Bibliografia, glossário de acrónimos e lista de documentos utilizados.



## **Capítulo 2 O projeto**

### **2.1 Sobre a CGI**

A CGI é uma multinacional canadiana, com aproximadamente 70.000 profissionais, cobrindo as áreas geográficas da América do Norte, da América do Sul, da Europa e da Ásia-Pacífico. Foca-se na prestação de serviços para processos de negócio e de TI *End-To-End* que facilitam a evolução continuada dos negócios. Em Portugal existem várias áreas em que a mesma está envolvida, sendo uma das mais relevantes o setor das energias.

### **2.2 Solução Diamond**

A solução *Diamond* começou a ser planeada em 2010, projeto que se inseriu numa apresentação da CGI ao QREN (Quadro de Referência Nacional), assim, pensou-se na criação de uma solução/ projeto diferenciador, que pudesse ser realizada e integrada no mercado das empresas de distribuição de água, saneamento, gás e eletricidade. Após uma análise interna e uma análise de mercado foi notada a existência de várias soluções para a gestão de ocorrências na rede elétrica, contudo situação inversa à verificada para as redes de água, de saneamento e de gás, daí a oportunidade de desenvolvimento do produto *Diamond*, uma solução adaptável a qualquer um dos mercados.

### **2.3 Objetivos inicialmente pretendidos**

Para que esta solução se pudesse tornar competitiva, necessitava de preencher um elevado número de requisitos:

- Disponibilização de um conjunto de ferramentas para a execução de funções de gestão da rede;
- Permissão de acesso aos dados de incidentes a toda a organização;
- Gestão de incidentes, gestão da rede, processos e procedimentos de reposição de serviço;
- Gestão operacional da rede – Cartografia de Base e a Rede de Distribuição;
- Reporte automático de comunicação de avarias e notificações;

- Configuração do mecanismo automático do Tempo Estimado de Reposição (TER);
- Apresentação tabular da informação e representação em modo gráfico da rede no seu estado operacional e com coloração associada;
- Análise de previsão e identificação do dispositivo provável de falha;
- Gestão efetiva dos incidentes nas redes de água, de saneamento e de gás;
- Funções de Gestão e Mobilidade das equipas de piquete;
- Mecanismo de previsão assente em regras de negócio;
- Gestão efetiva de manobras planeadas e divulgação aos meios de comunicação social;
- Relatórios operacionais e históricos, incluindo o histórico de falhas por cada equipamento afetado;
- Indicadores de qualidade de serviço, de acordo com a regulamentação para o setor;
- Ferramentas de configuração, integração e de administração;
- Interface unidirecional com o SCADA (Supervisory Control and Data Acquisition), permitindo o envio de eventos (sinalizações e medidas);
- Interface com um sistema de mobilidade para despiste e para resolução atempada de uma avaria no terreno;
- Envio de OT – Ordens de Trabalho relativas a avarias e despistes, bem como o reagendamento e a reatribuição de uma OT a outra equipa;
- Interface com o *Call Center*;
- Emissão de alarmes quando for detetada a existência de clientes com prioridades de reposição (por exemplo, instituições de saúde) ou com necessidades especiais afetadas;
- Integração com sistemas SAP e com outros sistemas implementados nas empresas gestoras;
- Relatórios de qualidade de serviço para as entidades reguladoras;
- Contabilização de todas as interrupções e durações por cliente;
- Registo e consulta das alterações efetuadas, tornando o sistema completamente auditável.

## 2.4 Características associadas ao produto

O desenvolvimento do produto levou em linha de conta os seguintes aspetos genéricos:

**Modular** – Ser um sistema desenvolvido por módulos, permitindo assim a sua implementação diferenciada em cada uma das entidades, de acordo com as suas reais necessidades e a área de negócio (água, saneamento ou gás);

**Desempenho** – A arquitetura levou em linha de conta a otimização de todos os recursos e de todos os componentes de modo a obter um elevado desempenho;

**Compatibilidade/ Interoperabilidade** – O *Diamond* permite a interligação entre diversas tecnologias associadas aos diferentes sistemas com os quais irá interagir, garantindo as comunicações e as interligações entre eles;

**Disponibilidade** – Prevendo-se um elevado grau de utilização e importância do *Diamond* para o desempenho das empresas onde venha a ser instalado, o tempo de indisponibilidade deverá ser reduzido ao mínimo, sendo prevista a implementação de soluções de recurso para o caso de falha;

**Escalabilidade** – O produto *Diamond* será dimensionado de acordo com as necessidades de crescimento das áreas de negócio das empresas de distribuição de água e de gás para as quais se encontra focalizado, perspetivando ainda a sua adaptabilidade a outras áreas de negócio, nomeadamente no setor de energia;

**Capacidade de armazenamento** – Será prevista a capacidade de armazenamento suficiente, de acordo com o volume de informação exetável, para cada uma das empresas onde o produto *Diamond* seja instalado;

**Segurança** – O produto *Diamond* obedece às seguintes propriedades:

**Confidencialidade** – A informação será disponibilizada segundo critérios rigorosos;

**Integridade** – A informação não poderá ser destruída nem corrompida, para que o produto execute corretamente as suas funções;

**Autenticação** - Validação da identidade de um utilizador, dispositivo ou processo;

**Controlo de acesso** - Impedir o acesso não autorizado a um determinado recurso. Definição de regras específicas que definam quem deve ter acesso a cada recurso disponibilizado pelo *Diamond*;

**Privilégios mínimos** - Apenas serão concedidos aos utilizadores os privilégios necessários para a execução das suas tarefas.

## **2.5 Riscos**

Dada a situação difícil que a economia portuguesa e mundial está a atravessar neste momento, não poderá deixar de se considerar como “fator de risco” o desenvolvimento de um novo produto de *software* e a procura de mercado para o mesmo. No entanto, aposta-se no caráter diferenciador e competitivo que se pretende para o *Diamond*, face às soluções atualmente disponíveis no mercado, sempre numa perspetiva de recuperação económica a médio prazo.

## **2.6 Competição no mercado**

Este capítulo trata de uma análise comparativa do produto *Diamond* com outras soluções disponíveis no mercado, analisando as respetivas vantagens e desvantagens.

### **2.6.1 Estudo de mercado**

As empresas operadoras nos setores de Energia e de Água estão obrigadas pelo Regulamento da Qualidade de Serviço à disponibilização de um conjunto de indicadores às respetivas Entidades Reguladoras, bem como ao pagamento de compensações aos clientes afetados por interrupções do serviço para além dos limites regulamentados. Para que estes objetivos sejam tangíveis é necessário efetuar uma gestão rigorosa das ocorrências que se registam nas respetivas redes.

Pela análise efetuada, concluiu-se não existir em Portugal nenhum produto disponível no mercado, nomeadamente no setor da água e do saneamento, que responda a estas necessidades, assim pressupõe-se que o tratamento desta informação seja efetuado por aplicações locais e não de uma forma integrada com outros sistemas, pelo que se infere que o rigor da informação não será o mais adequado. Desta análise concluiu-se, ainda, que a oferta a nível mundial para esta problemática é muito reduzida.



## 2.6.2 Competição direta

Abaixo é apresentado um resumo comparativo entre o *Diamond* e as diferentes soluções disponíveis no mercado e que foram objeto de análise:

### **SLG – Gestão de ocorrências**

Trata-se de uma solução desenvolvida no mercado nacional pela SLG para a AGS (Administração de Gestão de Sistemas de Salubridade). As suas funcionalidades são bastante básicas, para o que se entende ser um sistema de gestão de ocorrências que efetua basicamente o registo numa base de Dados mySQL das ocorrências verificadas nas redes de distribuição de águas para as entidades concessionadas pela AGS, fornece ainda alguma informação sobre perdas, quantificando em metros cúbicos estas mesmas perdas em situações de rotura.

**Custo:** Baixo

**Concorrente direto?** Sim, para empresas de águas concessionadas pela AGS

### **CGI Utility Solutions – Outage Management Solutions**

Trata-se de uma solução desenvolvida para o mercado de eletricidade americano. É uma solução que concorre diretamente com o *Diamond*, no entanto ao tentar implementar-se este serviço em território Europeu, verificou-se que o mesmo tem um custo demasiado alto para a sua aceitação. O produto CGI PragmaLINE OMS permite a integração com os sistemas de gestão da rede elétrica e sistema de mobilidade.

**Custo:** Alto

**Concorrente direto?** Não diretamente, comporta apenas a gestão de ocorrências para a rede de distribuição elétrica. Fácil adaptação às redes de águas e de gás. O preço é bastante díspar.

### **Telvent Miner – Responder**

Trata-se de um sistema desenvolvido para a gestão de ocorrências na rede de distribuição elétrica. O sistema encontra-se desenvolvido em ArcGis, concentrando todas as suas funcionalidades dentro de um sistema SIG (Sistemas de Informação Geográfica). Ao contrário de alguns sistemas de gestão de ocorrências, que assentam na disponibilização de um sistema próprio, o *Responder* utiliza componentes comuns a outros sistemas.

**Custo:** (Sem informação)

**Concorrente direto?** Não. Apenas trata as ocorrências da rede elétrica.

### **Let Systems – eRespond**

Trata-se de um sistema desenvolvido para a gestão de ocorrências nas redes de distribuição de água, de gás e de eletricidade, permitindo a gestão de ocorrências planeadas e não planeadas. O produto eRespond encontra-se orientado para a Internet, desenvolvido em linguagem de programação Java e corre em servidores aplicativos J2EE (Java 2 Enterprise Edition) compatíveis. O eRespond utiliza XML (Extensible Markup Language), JMS (Java Message Service) e Web Services para fácil integração com outros sistemas abertos.

**Custo:** Os preços são calculados na mesma base do *Diamond*, mas bastante superiores aos valores propostos para este.

**Concorrente direto?** Sim, uma vez que a solução tem integrada a gestão de ocorrências para as redes elétrica, de gás e de água, fazendo dela o principal concorrente ao *Diamond*.

### **CyberGear – DisSPatch**

Trata-se de um sistema desenvolvido para a gestão de ocorrências na rede de distribuição elétrica, permitindo a gestão de ocorrências planeadas e não planeadas.

**Custo:** Sem informação disponível.

**Concorrente direto?** Não. Apenas trata as ocorrências da rede elétrica.

### **Intergraph – Integrated Outage Management**






Trata-se de um sistema desenvolvido pela Intergraph para a gestão de ocorrências nas redes de distribuição de eletricidade e de gás. Este sistema encontra-se desenvolvido sobre o *software* GIS Geomedia, permitindo uma visualização em tempo real, pelas diferentes áreas das empresas, da informação residente na base de dados.

**Custo:** Ao nível dos propostos para o *Diamond*.

**Concorrente direto?** Sim, mas não na totalidade, comportando apenas a gestão de ocorrências para a rede elétrica e de gás. Fácil adaptação à rede de águas.

	Preço	Integração	Modularidade	Sectores	Funcionalidades
CGI - Diamond	baixo	elevado	elevado	elevado	elevado
SLG - GO	elevado	Muito baixo/Não existente	baixo	baixo	baixo
CGI - OMS	Muito baixo/Não existente	elevado	elevado	médio	elevado
Telvent Miner - Responder	N/D	elevado	elevado	médio	elevado
Let Systems - eRespond	médio	elevado	elevado	elevado	elevado
CyberGear - DISPATCH	N/D	N/D	N/D	baixo	baixo
Intergraph - IOM	baixo	elevado	médio	elevado	elevado

**Legenda:**

Muito elevado	
elevado	
médio	
baixo	
Muito baixo/Não existente	
Informação não disponível (N/D)	

**Tabela 2.1 - Comparativo de produtos concorrentes**

### Conclusão

As várias soluções analisadas (Tabela 2.1) apontam, de uma maneira geral, fundamentalmente para a gestão de ocorrências nas redes de distribuição de eletricidade, e aqui é realçado o carácter inovador do produto *Diamond*, nas vertentes gás, água e saneamento. Como exceção, aparece a solução *eRespond* da LET Systems, que é uma solução integrada, modular e que suporta a gestão de ocorrências nas redes de distribuição de eletricidade, de gás e de água.

Esta será a solução que mais se aproxima do que se pretende para o *Diamond*, e, como tal, servirá como referencial para definir o seu carácter inovador e diferenciador. Face ao *eRespond*, a solução *Diamond* tem a seu favor o preço de licenciamento proposto, que é muito abaixo dos valores praticados pela LET Systems.

## 2.7 Metodologia Agile – Scrum

O desenvolvimento de *software* é um processo de engenharia que necessita de procedimentos bem definidos. Ao contrário de outras áreas de engenharia, durante a sua execução não se obtém algo “visível”, como por exemplo, um edifício. No desenvolvimento de *software* com frequência ocorrem problemas de comunicação entre clientes finais e empresa de desenvolvimento e, por vezes, resulta num projeto em que a solução entregue está longe do esperado inicialmente.

Assim as metodologias *Agile* incutem uma maior proximidade com os *stakeholders*, possibilitando também uma grande alteração a nível do desenvolvimento, permitindo, assim, que também se torne possível um redirecionamento de prioridades, consoante a necessidade do mercado.

Com a metodologia *Scrum* [6] os projetos progridem em ciclos de desenvolvimento chamados *Sprints*. Neste projeto cada *Sprint* teve a duração aproximada de 20 dias úteis, consistindo geralmente em:

- 15 Dias úteis de desenvolvimento;
- 3 Dias de planeamento de futuras tarefas;
- 0-5 Dias úteis opcionais para alterações requeridas pelo *stakeholder*.

Nesta metodologia os clientes tornam-se parte da equipa de desenvolvimento, neste caso, em concreto, o gestor e *team leader* da CGI. Os mesmos, numa fase posterior, apresentarão as mesmas versões a possíveis clientes, assim como aos já existentes, trazendo o *feedback* dos mesmos para permitir um planeamento eficiente e o redirecionamento de prioridades caso fossem necessárias. As entregas são frequentes e mensuradas em funcionalidades 100% desenvolvidas.

Algo fulcral neste método é a transparência no planeamento e a elevada regularidade das reuniões a fim de monitorizar o progresso.

## 2.8 Trello

Para possibilitar a gestão e o controlo do projeto foi utilizada uma aplicação *Web*, com o nome *Trello*. O *Trello* é uma aplicação bastante versátil cuja principal função é guardar “coisas” em *cards* que permitem, assim, seguir de forma informal, mas que permitem controlar com eficiência o projeto.

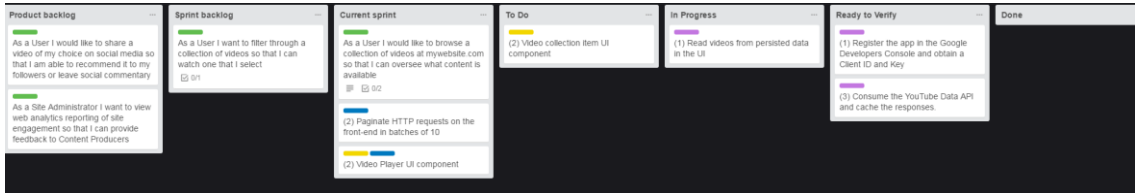


Figura 2.1 - Trello Scrum

O *Trello* guarda e organiza a informação nas seguintes categorias (Figura 2.1):

- **Product Backlog** – Mantendo uma lista priorizada das funcionalidades e dos requisitos;
- **Sprint Backlog** – Registrando a lista de *sprints* e de funcionalidades a serem entregues no fim do mesmo;
- **Current Sprint** – Mostrando qual o *sprint* que está a ser executado no momento;
- **To Do** – Lista em que são mostradas as funcionalidades que estão pendentes, atribuídas aos recursos no planeamento e priorizados;
- **In Progress** – Mostradas as funcionalidades que estão a ser executadas pela equipa;
- **Ready to verify** – Funcionalidades que estão prontas para ser testadas;
- **Done** – Tarefas que foram completadas a 100%.

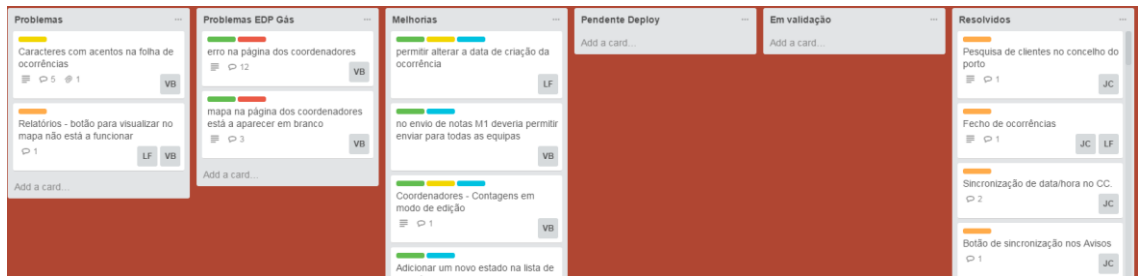


Figura 2.2 - Trello específico para cliente

Para manter também um *track* mais específico de cada cliente foram também criados “*trello's*” específicos (Figura 2.2) para cada um contendo campos adicionais, como:

- **Problemas** – Referidos aos problemas encontrados;

- **Melhorias** – Sugestões de melhorias, que são analisadas e que, consoante a exequibilidade das mesmas, saber se está incluído no propósito ou é um desenvolvimento específico;
- **Pendente *deploy*** – Erros reportados, melhorias feitas, estão prontas para irem para o cliente;
- **Em validação** – O cliente faz testes internos mais específicos a fim de validar as funcionalidades;
- **Resolvidos** – Todas as tarefas que foram executadas.

## 2.9 Desenvolvimento

### 2.9.1 Equipa de desenvolvimento



Figura 2.3 - Composição da equipa

A equipa de desenvolvimento (Figura 2.3), como especificado na metodologia *Scrum*, requer três intervenientes principais, o *Product Owner*, um *Scrum Master* e a equipa de desenvolvimento. Destacam-se o *Product Owner*, o Eng.º Nuno Santos; o *Scrum Master*, o Eng.º Joaquim Batista; como membros seniores da equipa de desenvolvimento, Jorge Cardoso e Vasco Barreto; por fim, João Mourão, João Dias, Alexander Fernandes e Miguel Melo, como restante equipa de desenvolvimento.

Eu integrei a equipa de desenvolvimento, onde fui responsável por implementar as componentes descritas ainda a abordar neste relatório, tendo sido supervisionado por elementos seniores, os quais têm, obviamente, um conhecimento mais aprofundado sobre a unidade de negócios e sobre a arquitetura de todo o sistema, sendo possível o esclarecimento de qualquer dúvida que surgisse.

## 2.9.2 Ferramentas de desenvolvimento

O desenvolvimento desta solução, como explicado anteriormente, foi dividido em três *milestones*, o *Diamond Web*, o *Diamond Web Manager Mobile* e o módulo *Diamond Field Agent*.

Começando com os componentes (Figura 2.4) que foram transversais à solução, neste caso as ferramentas de versionamento, a solução usada foi a Apache Subversion, utilizando um cliente específico, o TortoiseSVN. Este ponto deve-se, como já referido, à utilização de *Software Open Source* com alta aceitação pelo público.

Outras ferramentas que foram transversais às aplicações foram as de apoio ao desenvolvimento *web*, nomeadamente HTML 5, CSS 3 e Javascript [1]. Por gosto pessoal, a ferramenta principalmente utilizada para este tipo de desenvolvimento foi o Visual Studio Code, por ser uma ferramenta leve, *Open Source* e que permite que o desenvolvimento de componentes feitos pela comunidade que podem ser disponibilizados.



Figura 2.4 - Ferramentas e tecnologias utilizadas

Para acelerar o processo foram também utilizadas *Frameworks/Bibliotecas*, a fim de aumentar a conformidade em toda a aplicação em termos de estilos e também para que estes fossem aplicados uniformemente ao maior número de *web browsers* e de versões possíveis. A biblioteca usada foi o *JQuery* [2], que é uma biblioteca que simplifica no desenvolvimento *Javascript*, sendo bastante fácil de aprender. Para a parte de *Design a Framework* utilizada foi o Bootstrap, que, apesar de não ser uma *Framework*, é apelidada como tal pelos desenvolvedores. Esta é um grande ficheiro CSS que permite uma uniformização sobre os elementos de *layout* e os seus comportamentos. O desenvolvimento *web* foi parte integrante dos três módulos relatados neste relatório de projeto, sendo posteriormente explicado de forma mais aprofundada.

Nas componentes *Diamond Web* e *Diamond Web Manager Mobile*, para que houvesse informação registada, relacionada e que possuísse integridade, foi utilizado um sistema de base de dados relacionais, o Oracle 11g. Para a componente *Diamond Field Agent* foram utilizados dois tipos

de bases de dados, visto que a aplicação *mobile* necessitava que a informação fosse disponibilizada com alta rapidez e em tempo real. Assim, foi utilizada uma base de dados apelidada de *Online*, em que toda a informação é sincronizada em tempo real por todos os clientes, mais concretamente a Google Firebase Realtime Database, sendo uma base de dados em tempo real, também do tipo NoSQL (Non SQL or Non relational) e permitindo disponibilidade de informação, mesmo quando a aplicação fica *offline*.

Outra necessidade, em termos de dados, foi o acesso a grandes volumes de dados e de pesquisas rápidas (equipamentos de rede), em que a mesma informação não necessitava de ser relacionada. Nesse sentido, foi então utilizada a base de dados CouchDB, uma base de dados baseada em documentos que permite um rápido acesso à informação focada no princípio de fiabilidade dos dados e que possui também uma API (Application programming interface) HTTP/JSON (Hypertext Transfer Protocol/JavaScript Object Notation). Foi o ideal para o uso e integração com as linguagens WEB.

Para a gestão de mapas foram utilizados dois fabricantes: o Bing Maps [3], uma solução utilizada inicialmente, mas em fim do ciclo de vida, sendo de grande urgência a sua substituição, sendo assim, foi substituído pelo Bing Maps V8 Javascript SDK (Software Development Kit); e para a componente de *Diamond Field Agent*, de modo a haver diversidade e um *maior know-how*, foi utilizada a SDK da Google, a Google Maps API Javascript.

Para as camadas de BLL (Business Logic Layer) foram utilizadas duas ferramentas distintas, no *Diamond Web* e *Diamond Web Manager Mobile* os *webservices* foram desenvolvidos em C# utilizando a *Framework* WCF [4], para programar e compilar a mesma, o IDE (Integrated Development Environment) utilizado foi o Visual Studio Professional 2012.

No *Diamond Field Agent*, a mesma foi disponibilizada em Node.js e NPM. Este último é um servidor que executa todo o código em *Javascript*, reduzindo assim a curva de aprendizagem de uma nova linguagem. É focada em eventos e é caracterizada por ser não bloqueante. O NPM é atualmente o maior ecossistema de bibliotecas *open source* para utilizar na implementação do Node.js.

Por fim, a aplicação *Diamond Field Agent*, por ser uma aplicação híbrida, foi então utilizado em parte da mesma linguagens web e noutra parte ferramentas específicas Android, utilizando código nativo Java, sendo utilizado como IDE da mesma o Android Studio com IntelliJ [5], que, apesar de não ser uma ferramenta de fácil utilização, a mesma é a oficial, por conseguinte, optou-se pela utilização da mesma.

## 2.10 Ambiente Aplicacional

### 2.10.1 Requisitos Mínimos – Cliente

O *redesign* da solução foi baseado em *HTML5* e *CCS3*. Tecnologias relativamente recentes, o que levou a que os requisitos mínimos por parte dos dispositivos clientes desktop sejam, um *web browser* que suporte *HTML5*, *CSS3* e que tenham a execução de *Javascript* ativa. Sendo, no entanto, recomendado o uso de um dos três maiores *players* no mundo dos *web browsers*:

- Internet Explorer 10+ (atualmente com o nome EDGE);
- Google Chrome 30+;



- Mozilla Firefox 35+.

No ambiente *mobile*, o desenvolvimento foi focado em Android, sendo que os requisitos são semelhantes aos dos clientes Desktops. No entanto, devido à arquitetura das versões do sistema operativo Android, o suporte mínimo está definido na versão 5.0 (*Lollipop, API Level 21*), por ser a primeira versão que começou utilizar a componente modular “*Android System WebView*” permitindo a abstração da versão do sistema operativo, anteriormente, cada versão do sistema Android tinha uma versão do *web browser* não permitindo a atualização da mesma.

## 2.10.2 Requisitos – Servidor

Devido à escalabilidade desta solução e à sua fácil flexibilidade o seu requisito mínimo é uma máquina que seja capaz de correr o sistema operativo Windows Server 2012 R2, os mesmos compreendem:

- **Processador:** 1.4GHz;
- **Memória Ram:** 512MB;
- **Espaço em disco:** 32GB.

No entanto, inerentemente à solução, a máquina necessita também de igualar ou superar os requisitos mínimos das seguintes aplicações:

- Microsoft. Net Framework 4.0;
- Microsoft IIS 7.5;
- Oracle 11g;
- CouchDB 1.6.1;
- Node.js;
- NPM.

Terá que ser garantido também conectividade entre as máquinas que estejam envolvidas ou que seja necessário a integração com as mesmas.

No entanto para um correto funcionamento da solução e boas prestações para uma solução com cerca de 50 utilizadores é recomendado o seguinte setup:

- **Processador:** Intel Xeon E5 Family;

- **Memória Ram:** 16GB;
- **Espaço em disco:** 1 TB;
- **Ligação:** *Gigabit*.

Tem sido testada a possibilidade de ser oferecida uma solução que permita ao cliente abstrair-se destes componentes e requisitos, utilizando soluções disponibilizadas e mantidas pela *Microsoft*, o produto *Azure*, contando com servidores *Web*, bases de dados, entre outros, estando a mesma já a ser implementada na solução *Diamond Field Agent*.

## 2.11 Arquitetura da solução

A arquitetura *Diamond*, como foi dito, é modular, podendo ser integrada de diversas formas em distintos sistemas, a arquitetura descrita neste capítulo tem como base a arquitetura implementada em dois dos grandes clientes que foram acompanhados.

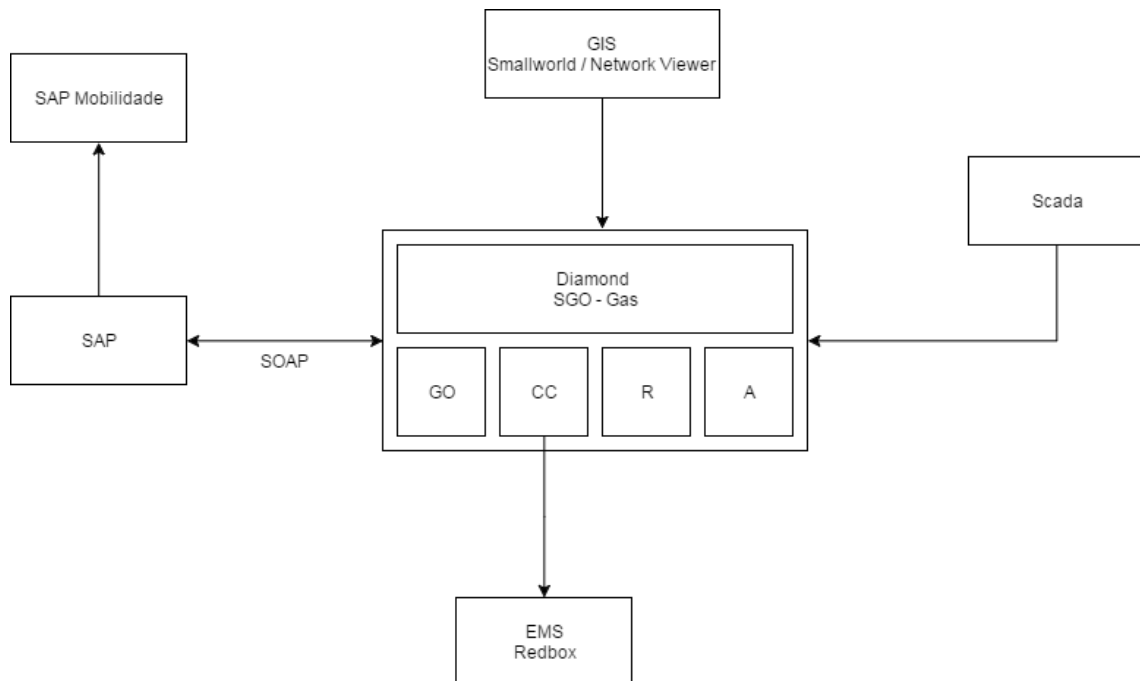


Figura 2.5 - Arquitetura da solução

### 2.11.1 Módulos internos

A (Figura 2.5) representa a arquitetura utilizada nesses clientes. A caixa no centro, representa a solução *Diamond*, a mesma está de momento implementada e a ser utilizada no seu total escopo dos desenvolvimentos feitos. É composta em suma por 4 componentes, o GO (Gestão de Ocorrências), CC (*Contact Center*), R (*Reporting*) e A (Ferramenta de Administração). A minha participação neste

projeto recai maioritariamente no módulo de GO, onde principais funções são a disponibilização da Interface (*PL*), da metodologia de negócio (*BLL*). O módulo CC, trata de executar tudo o que esteja relacionado com chamadas *inbound* e *outbound* relacionadas com a Gestão de Ocorrências. R é módulo de *reporting* que usa tecnologia *open source* BIRT. Por fim, a ferramenta de administração, que tem a função de gestão de parâmetros da unidade de negócios, esta, trata por exemplo da configuração de sincronização do sistema de autenticação com um sistema *AD* (Active Directory) ou com sistema *LDAP* (Lightweight Directory Access).

### 2.11.2 Módulos externos

Quanto a módulos externos, existem quatro diferentes, as bases dados de SIG do cliente, incluem, a existente cartografia, o seu mapeamento de equipamentos e toda a informação a eles relacionada, neste caso na tecnologia SmallWorld da GE (General Electric). SCADA, que é o módulo relacionado com a alarmística dos equipamentos de rede, onde os mesmos disparam alarmes e medidas que são posteriormente enviados para a solução *Diamond*, onde poder-se-ão gerar alarmes ou prever possíveis pontos de falha na rede. EMS Redbox, permite a comunicação entre a solução e o contact center com o exterior; por fim o módulo de SAP, é responsável por enviar e receber a informação entre os agentes no terreno e os engenheiros do despacho. Devido a contratos estritos entre a SAP e os clientes tem sido difícil de penetrar nesse nicho, no entanto o módulo *Diamond Field Agent* vem tentar substituir essas funcionalidades com a nossa solução *in-house*.

### 2.11.3 Arquitetura e padrão de desenvolvimento

Na versão anterior *Diamond*, foi utilizado o *design de pattern* multicamada [9] de forma bastante estrita, dividindo a solução em três camadas, a *DAL* (*Data Access Layer*), *BLL* (*Business Logic Layer*) e *PL* (*Presentation Layer*). Seguindo de forma estrita este *design pattern* requeria uma abstração total entre camadas, o que resultou num código bastante complexo, difícil de ler e manter. Foi então, decidido que, para o redesign desta aplicação, seria necessária uma alteração para aumentar a performance e utilizar capacidades nativas do sistema de SGBD. Resultando então, em um sistema com uma ligeira alteração (Figura 2.6), em que algum apoio à *BLL*, era desenvolvido na camada de *DAL*, sob a forma de *views* e *stored procedures*. Pelas pesquisas realizadas, opiniões divergem no entanto a mesma parece não violar o sistema de multi camada.

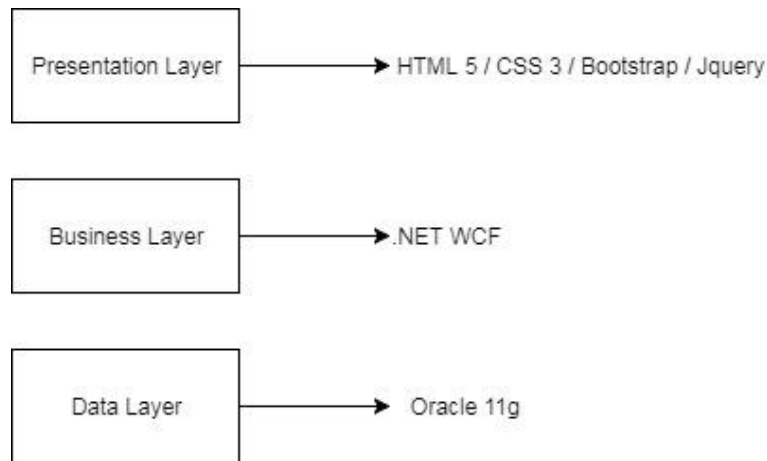


Figura 2.6 - Arquitetura e tecnologia usada por camada

As máquinas que correm os serviços da *BLL*, são sistemas *Microsoft Windows server 2008 R2 Standard* utilizando o *IIS (Internet Information Services)*, para dar o suporte necessário às comunicações *HTTP*, via *webservices*, utilizando o standard *REST (Representational State Transfer)*. Esses *webservices* foram implementados utilizando a *Proxy Design Pattern*, de modo a permitir uma interligação entre todos os componentes externos não sendo necessário uma nova implementação para cada um deles.

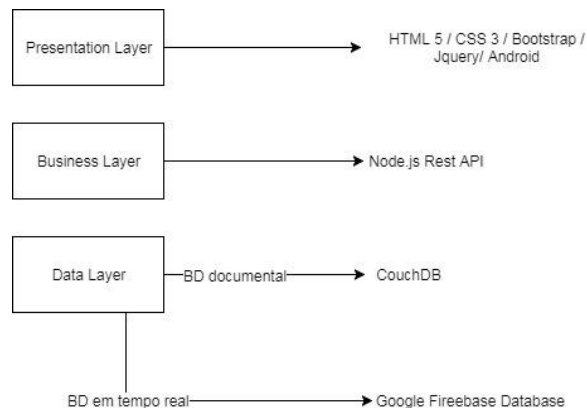


Figura 2.7 - Arquitetura Diamond Mobile

No novo módulo *Diamond Field Agent*, devido a diferentes pressupostos e especificações tecnológicas, a multicamada de três níveis foi cumprida à risca (Figura 2.7), no entanto por haver necessidade de dois tipos distintos de acessos de dados, a camada *Data Layer* contém tanto a base de dados de tempo real *Google Firebase Database* e a *CouchDB* [7] para salvar os documentos. Na *presentation layer* foi adicionada, a tecnologia *Android*, para implementar funcionalidades nativas, neste caso a implementação de notificações assim como o acesso a outras componentes nativas, como por exemplo localização e acesso à câmara.

## 2.12 Evolução da Solução Diamond

Ao longo da minha participação de projeto na CGI, a aplicação foi sendo melhorada, novos componentes foram desenvolvidos, o *Diamond Web*, foi a primeira componente a ser feito o *roll-out*, sendo o mesmo feito nos finais de março, a aplicação *Diamond Web Manager Mobile*, aproximadamente um mês após e a aplicação *Diamond Field Agent* que está neste momento no decorrer do seu *sprint* final, com data esperada de conclusão a 5 de julho, caso tudo corra em conformidade será lançado como módulo disponível a meados de julho.

## 2.13 Planeamento inicial e trabalho final

Remontando a maio/junho de 2016, foi estabelecido um planeamento inicial por parte da CGI, sobre as tarefas expectáveis como seu colaborador (Figura 2.8).

Descrição de atividades	Out 16	Nov 16	Dez 16	Jan 17	Fev 17	Mar 17	Abr 17	Mai 17	Jun 17
Workshop com utilizadores para detalhe de requisitos	■								
Documentação de análise		■							
Reimplementação em HTML5 de funcionalidades actualmente existentes em Silverlight		■	■						
Implementação serviços REST				■					
Integração via web services com outros sistemas corporativos					■				
Implementação de novas funcionalidades sobre BD NoSQL					■				
Testes de software (testes unitários e de integração)						■			
Acompanhamento aos testes de aceitação							■		
Suporte ao roll-out							■	■	
Documentação da release/exploração								■	
Relatório de dissertação									■

Figura 2.8 - Planeamento inicial (Alto nível)

Com o decorrer do tempo as expectativas não só foram alcançadas, como também foram superadas, resultando assim na possibilidade de participar em mais partes do produto *Diamond*.

Sucedeu-se o *Diamond Web Manager Mobile* que foi uma componente de tamanho mais reduzido mas que continuou totalmente dentro das competências que eram esperadas por mim no decorrer de todo o projeto mesmo a nível inicial.

1	Task Name	Duration	Start	Finish
2	Company training	5d	09-19-16	09-23-16
3	Upgrade map to HTML5	60d	09-26-16	12-16-16
4	Development	30d	09-26-16	11-04-16
5	Map upgrade	23d	09-26-16	10-26-16
6	integration with the old solution	7d	10-27-16	11-04-16
7	Testing	20d	10-31-16	11-25-16
8	Internal Testing	10d	10-31-16	11-11-16
9	Client testing	10d	11-14-16	11-25-16
10	Client Solution Instalation	10d	11-28-16	12-09-16
11	Solution Documentation Revision	5d	12-12-16	12-16-16
12	Improvements - old components	35d	12-19-16	02-03-17
13	Development	25d	12-19-16	01-20-17
14	Testing & corrections	10d	01-23-17	02-03-17

Figura 2.9 - Lista de esforço corrigido

Na figura 2.9, observa-se o trabalho efetivamente realizado, com prospeções bastante positivas sobre o projeto executado. A começo de fevereiro já havia informações sobre a minha participação no próximo projeto.

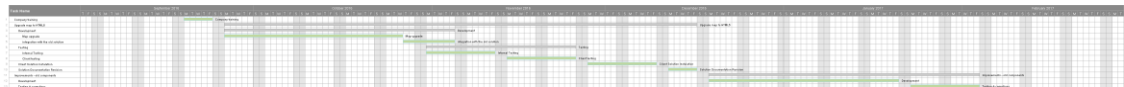
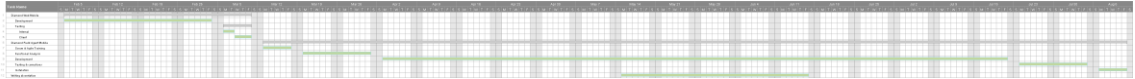


Figura 2.10 - Calendário meio-termo

O trabalho efetuado (Figura 2.10 e 2.11) permitiu fazer uma nova avaliação e planeamento de trabalho e este em acabou por ficar em concordância com o planeamento de trabalho projetado feito então.

1	Task Name	Duration	Start	Finish
2	Diamond Web Mobile	25d	02-06-17	03-10-17
3	Development	20d	02-06-17	03-03-17
4	Testing	5d	03-06-17	03-10-17
5	Internal	2d	03-06-17	03-07-17
6	Client	3d	03-08-17	03-10-17
7	Diamond Field Agent Mobile	110d	03-13-17	08-11-17
8	Scrum & Agile Training	5d	03-13-17	03-17-17
9	Functional Analysis	10d	03-20-17	03-31-17
10	Development	80d	04-03-17	07-21-17
11	Testing & corrections	10d	07-24-17	08-04-17
12	Instalation	5d	08-07-17	08-11-17
13	Writing dissertation	25d	05-15-17	06-16-17

Figura 2.11 - Lista de esforço final



**Figura 2.12 - Calendário Final**

Assim, como o planeado a meio-termo deste relatório (Figura 2.12), este foi escrito em paralelo com a implementação da componente *Diamond Field Agent*, sendo esta, apenas terminada após a data de entrega do mesmo.

## Capítulo 3 Trabalho Realizado

Devido ao anúncio do fim de vida tecnológico da componente Bing Maps Silverlight API, esta era a componente que necessitava de atenção premente, pois, o seu fim de vida estava agendado para 30 de novembro de 2016. A ser uma componente crítica, a mesma teve uma prioridade elevada na parte do redesenho. Após a data anunciada, a Microsoft referiu que continuaria com os serviços ligados durante algum tempo, a fim de facilitar a migração de todas as soluções nessa versão, mas já não suportaria nenhum pedido referente ao mesmo.

Neste relatório todos os componentes demonstrados no trabalho foram realizados/implementados por mim, quando for exibido algum componente ou parte não desenvolvido por mim, demonstrá-lo-ei por anotações.

### 3.1 Diamond Web

À data de início de participação de projeto na CGI a 16 de Setembro, visto que o meu conhecimento sobre este projeto e a lógica de negócio ser diminuta, era de alta importância a minha integração no desenvolvimento de algo com menor necessidade de possuir esse conhecimento, foi-me então incumbido a importante e crítica tarefa de integrar as antigas funcionalidades implementadas no componente apelidado de *Viewer*, da tecnologia *Bing Maps Silverlight SDK* para a mais recente *Bing Maps V8 Javascript SDK*. A mesma, foi testada inicialmente pelos membros seniores Jorge Cardoso e Vasco Barreto, chegando à conclusão inicial que, a mesma estaria possivelmente capaz de ser utilizada como parte integrante deste produto.

A arquitetura e o funcionamento da antiga componente é bastante diferente da nova solução, uma vez que *Silverlight* é executado no lado do servidor e exibido no cliente, ao contrário do *Javascript* em que é inicialmente requerido ao servidor e após receber o código do mesmo, o processamento e renderização são feitos no lado do cliente.

#### **Viewer**

*Viewer* (Figura 3.1 e 3.2) é o nome do primeiro controlo desenvolvido neste produto.

#### **Principais funcionalidades:**

- Sobreposição de *Tile Layers* a pedido do utilizador;
- Sobreposição de equipamentos com informação associada;



- Pesquisa de equipamentos;
- Pesquisa de moradas;
- “Rastreamento”, operação que permite com um clique aceder a informação sobre equipamentos e consegue em tempo real perceber onde se encontram por exemplo condutas de gás, válvulas e em tempo real dizer quais são as áreas que serão afetadas por uma possível intervenção.

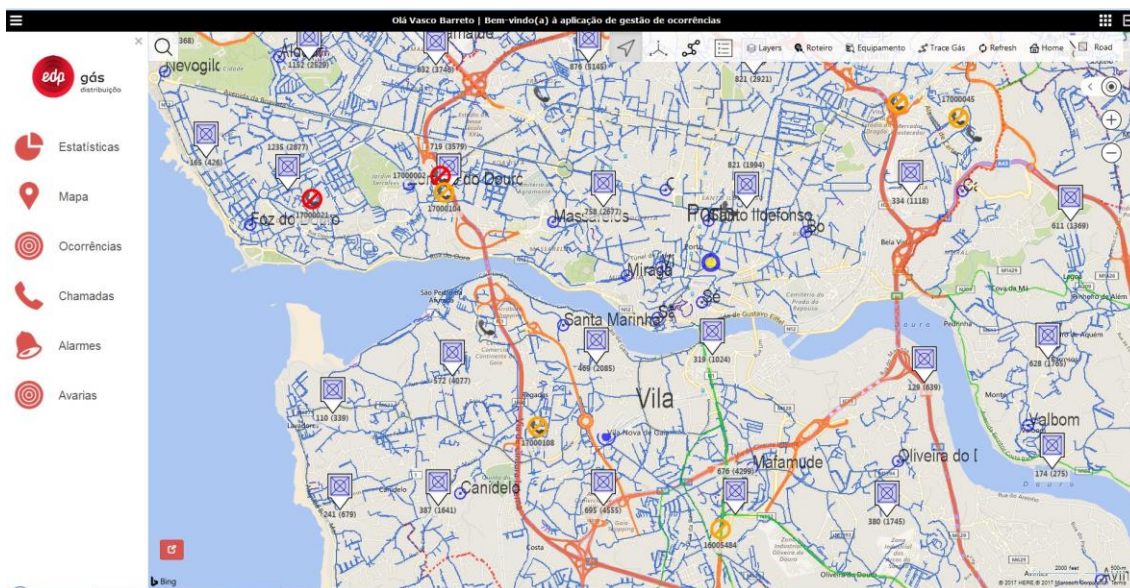


Figura 3.1 - Página inicial viewer com as tile layers ativas por defeito

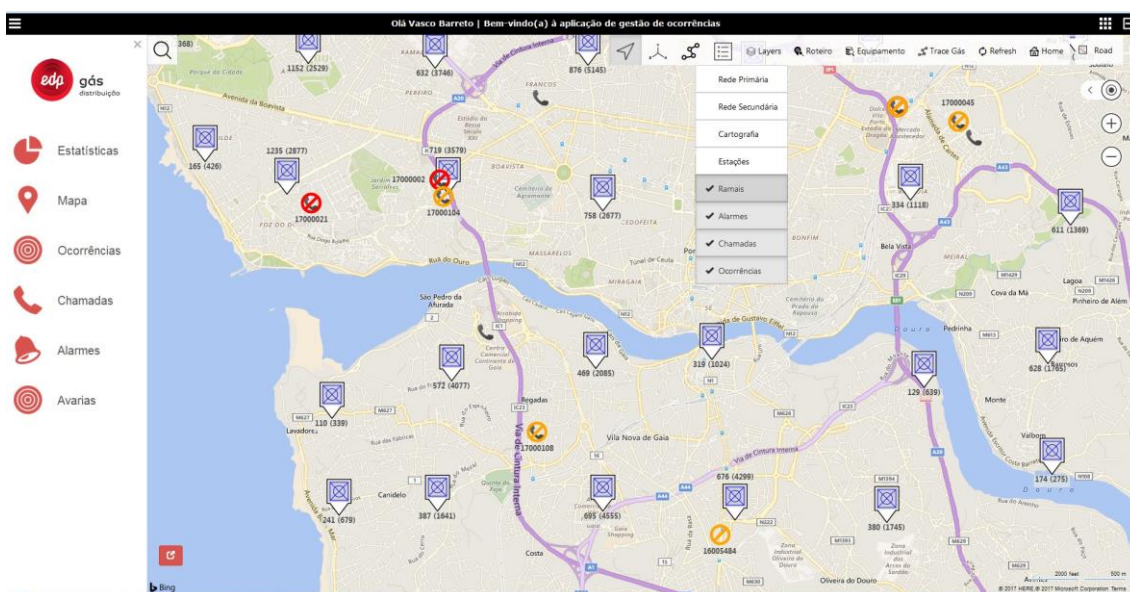


Figura 3.2 - Viewer mostrando equipamento, tile layers desativadas



Figura 3.3 - Controlos Viewer

De seguida serão descritos os controlos do Viewer (Figura 3.3):

1. **Modo de navegação;**

Permite livre navegação no mapa, os *inputs* do rato são os definidos por defeito pela Microsoft (Este botão também é representativo do modo atualmente selecionado);

2. **Modo de desenho**

Permite que seja selecionada uma coordenada do mapa, para ser associada a uma OS e integração com restantes componentes da solução desenvolvidos por outros colegas (Este botão é também representativo do modo atualmente selecionado);

3. **Modo de rastreamento**

Permite ativar o modo de rastreamento, permitindo ver as consequências na rede e clientes afetados em tempo real (Este botão é também representativo do modo atualmente selecionado);

4. **Modo detalhes de equipamentos**

Permite ativar os controlos para pesquisa de equipamentos (Este botão é também representativo do modo atualmente selecionado);

5. **Menu de camadas (Tile layers e pontos)**

Permite controlar a visibilidade de camadas escolhidas pelo utilizador;

6. **Pesquisa de informação de clientes**

Permite a pesquisa de informação sobre cliente;

7. **Pesquisa de equipamento**

Permite a pesquisa de informação sobre equipamento;

8. **Modo de rastreamento**

Igual ao ponto 3. (Este botão não é representativo do modo, sendo o mesmo feito no botão nº3);

### 9. Atualização do mapa

Permite refrescar o mapa, e atualiza toda a informação para a mais recente disponível, visto que podem ter ocorrido atualizações;

### 10. Home

Permite deslocar o mapa para uma coordenada e zoom predefinido;

### 11. Controles base Bing Maps V8

Controles básicos do Bing Maps V8.

## Modo de Rastreamento

O modo de Rastreamento (Figura 3.4) é de elevada relevância por ser um componente inovador, tendo sido também bastante moroso em termos de desenvolvimento.

Este componente, era bastante desejado pelos clientes visto que os mesmos não conseguiam ter um feedback célere do lado do Despacho sobre quais as repercussões que alguma intervenção iria ter, uma vez que, antigamente, era necessário estudar a zona e fazer toda esta operação manualmente.

Quando este modo é ativado, o utilizador seleciona uma localização na rede que é enviado para serviços do sistema SIG do cliente devolvendo todos os equipamentos, o seu estado atual e formas de transporte (ex. Condutas), sobrepondo-o por cima do mapa base.

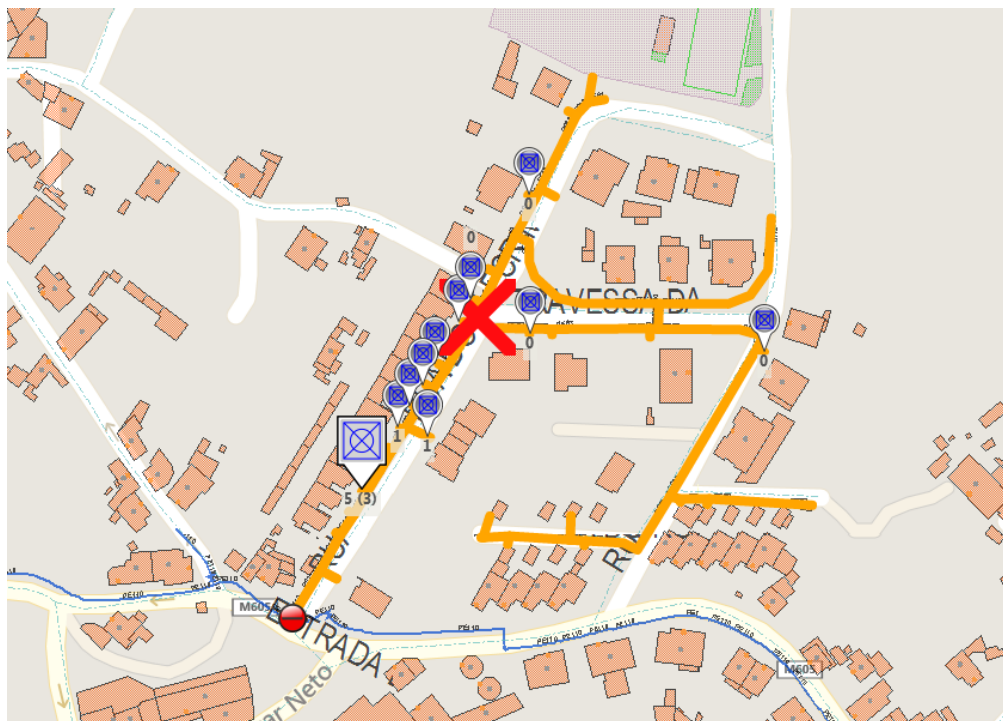


Figura 3.4 - Modo de rastreamento





Icon	Descrição
	Ponto inicial de trace
	Cluster de Contadores
	Contador
	Válvulas

Tabela 3.1 - Simbologia relevante

Na (Figura 3.4), observa-se, um ponto inicial de rastreamento. As linhas a laranja, representam as condutas de gás, observa-se também uma válvula que está fechada (significa que gás é transportado). A informação aqui apresentada não é extensa, pois a informação de desenvolvimento é parcial, no entanto em ambiente de produção, seria possível “abrir” e “fechar” válvulas e apareceria a restante rede e equipamentos intervenientes afetados.

## 3.2 Diamond Web Manager Mobile

Neste segundo, neste capítulo, organizado também de forma cronológica está descrito o módulo *Diamond Web Manager Mobile*. Este módulo, tem como finalidade dotar os gestores e engenheiros do Despacho, funcionalidades simplificadas e específicas à distância de poucos *clicks*, onde quer que se encontrem desde que tenha o telefone consigo.

### Protótipo/Mock up

Nas figuras 3.5 a 3.8, temos alguns dos *designs* feitos para demonstração ao cliente, os possíveis *layouts* da aplicação.



Figura 3.5 - Página inicial



Figura 3.6 - Menu lateral



Figura 3.7 - Resultados do módulo de pesquisa



Figura 3.8 - Exibição de detalhes de chamada

## Desenho Final

Após, aproximadamente um mês de desenvolvimento, o mesmo foi concluído com sucesso e com bastantes semelhanças com a prototipagem (Figuras 3.9 – 3.14).

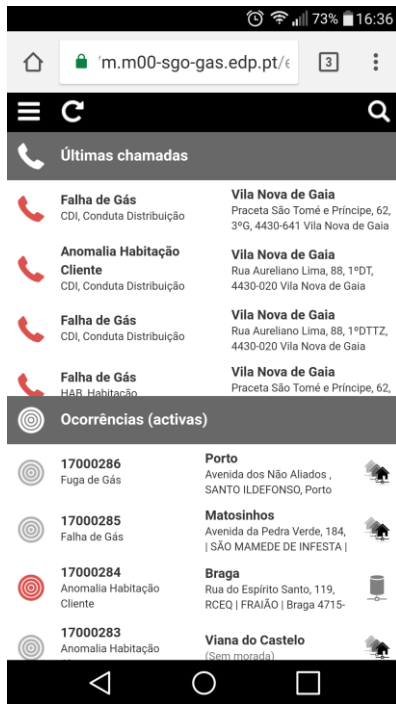


Figura 3.9 - Página Inicial Listagem

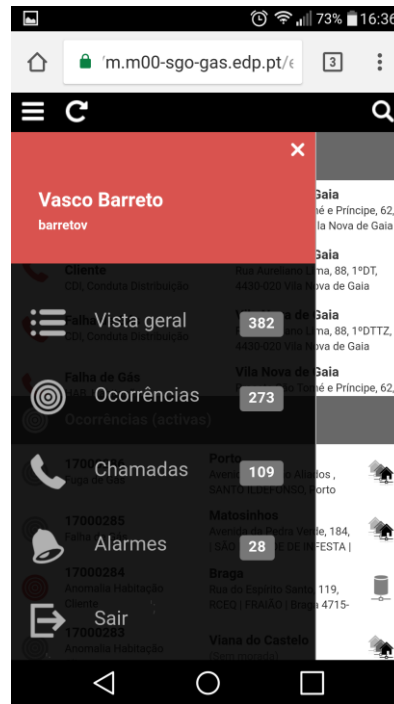


Figura 3.10 - Menu lateral



Figura 3.11 - Resultados da pesquisa



Figura 3.12 - Exibição de detalhes da chamada (1/3)



Figura 3.13 - Exibição de detalhes da chamada (2/3)

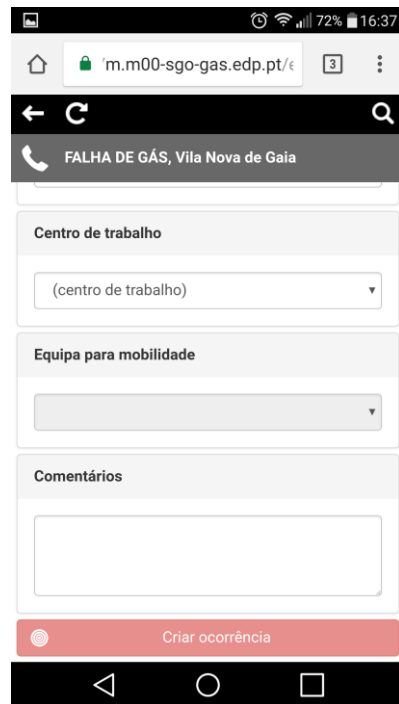


Figura 3.14 - Exibição de detalhes da chamada (3/3)

### **Principais funcionalidades:**

- Listagem e detalhes de ocorrências;
- Listagem e detalhes de alarmes;
- Listagem e detalhes de chamadas;
- Página inicial com listagem de ocorrências e chamadas, ambas tem uma importância superior a alarmes;
- Pesquisa de registos por tipo;
- Criação de ocorrências a partir dos menus de detalhes de alarmes e chamadas. Com associação de classificação, sintoma, companhia a dirigir, sistema de distribuição, centro de trabalhos, equipa e inserção de comentários.

Neste módulo, a minha participação não incluiu a funcionalidade de pesquisa e de login, as mesmas foram realizadas por um elemento sénior da equipa, que tinha melhor conhecimento e requisitos sobre esses métodos. A minha participação, incidiu sobre tudo o resto: Vista geral, Listagem de ocorrências, chamadas, alarmes e detalhes das mesmas. Sendo estas funcionalidades, em geral bastante semelhantes, sendo no entanto necessário alterar a sua especificação entre elas.

## **3.3 Diamond Field Agent**

Terceiro e último módulo a ser referido neste relatório de projeto, a aplicação *Diamond Field Agent*, um componente totalmente desenvolvido de raiz, complexo e inovador. Desenvolvido em proximidade com um dos nossos clientes atuais, estando também, a ser preparado para demonstração para o projeto UPGRID.

Este componente, já estava apontado como lacuna na solução *Diamond*, no entanto devido a contratos estritos com outras empresas a mesma nunca tinha tido possibilidade de penetração neste nicho, no entanto é algo que, esta componente da solução pretende colmatar.

### **3.3.1 Projeto Upgrid**

O projeto UPGRID começou no início de 2015 sob a iniciativa europeia H2020 (Horizon 2020), é um programa a ser desenvolvido pelo consórcio Europeu, composto por 19 parceiros de 7 Países europeus: Espanha, Portugal, Polónia, Suécia, Reino Unido, França e Noruega.

O projeto inclui quatro demonstrações que serão realizadas de Abril de 2015 até Julho de 2017, nos seguintes locais: Bilbao, no norte de Espanha, Parque das Nações em Lisboa (Portugal), Amal em Dalsland no sul da Suécia e Gdynia no norte da Polónia.



### 3.3.2 Descrição e demonstração de funcionalidades

As principais e mais relevantes funcionalidades deste módulo são:

**Principais funcionalidades:**

- Funcionamento em dispositivos mobile (Android de momento);
- Visualização de cartografia base, geográfica e esquemática;
- Sobreposição de *Tile Layers* controladas pelo utilizador;
- Sobreposição de camada de pontos com informação específica;
- Clustering de equipamentos;
- Pesquisa de moradas;
- Pesquisa de instalações/equipamentos;
- Relato de anomalias na rede notadas por piquetes;
- Visualização StreetView de componentes;
- Visualização de esquemas internos;
- Associação de ocorrências através do despacho;
- Listagem e acesso de informação às ocorrências associadas à equipa do utilizador;
- Alteração de estado de tarefas em ocorrência;
- Modo colaborativo de telefonia com o despacho;
- Modo colaborativo de *Sketching*.

### 3.3.3 Demonstração

De seguida serão explicados dos controles da aplicação (Figuras 3.15 – 3.18):

#### Página Inicial

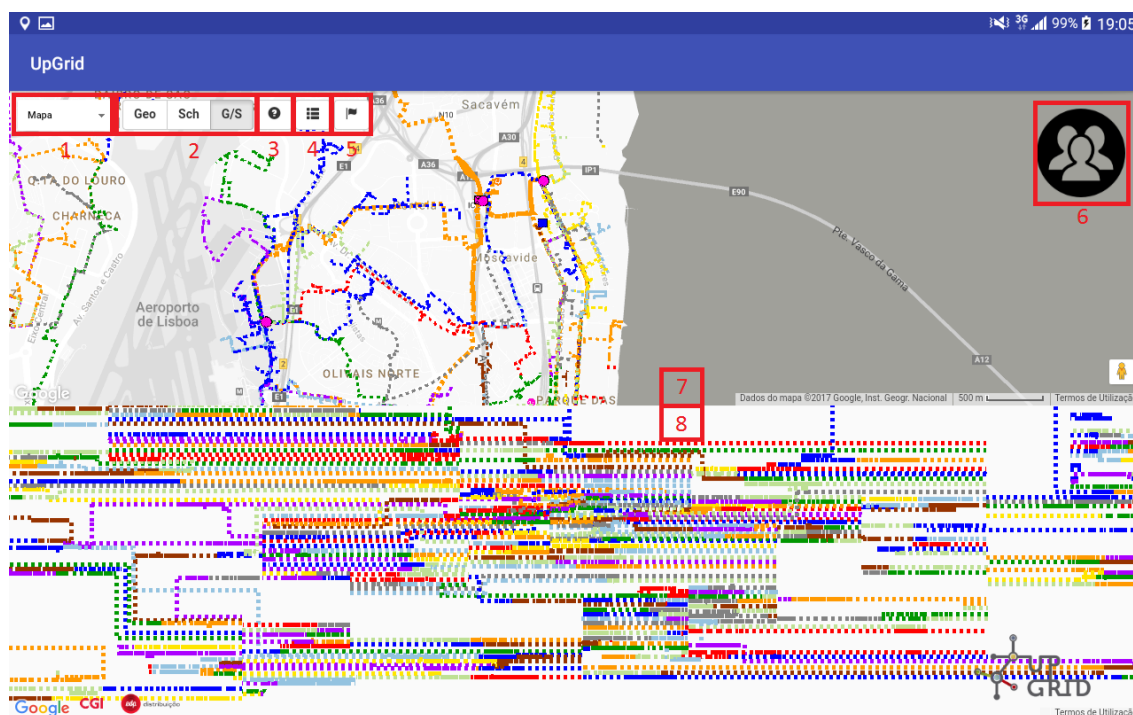


Figura 3.15 - Página inicial *Diamond Field Agent*

1. **Botão *standard* Google**  
Alternar entre modos de vista, estrada, satélite, entre outros;
2. **Mapas a exibir**  
Botão que permite seleccionar quais os mapas se pretende ter visíveis em ecrã. Só Geográfico, só esquemático ou ambos como (Figura 3.15);
3. **Pesquisa**  
Permite pesquisa de moradas e pesquisa de equipamentos por ID;
4. **Camadas e opções**  
Permite ativar e desativar as *Tile Layers* e camadas de pontos;
5. **Report de incidentes**  
Permite dar ao agente a possibilidade de reportar uma anomalia na rede sem necessidade de perda de contexto com a sua tarefa atual;

6. **Botão de informação de equipa**

Permite abrir o menu, onde se consultar a informação de ocorrências associadas à equipa do utilizador;

7. **Mapa geográfico**

Mapa de cartográfica geográfica;

8. **Mapa esquemático**

Mapa de cartografia esquemática.

### Consulta de ponto e Lista de ocorrências

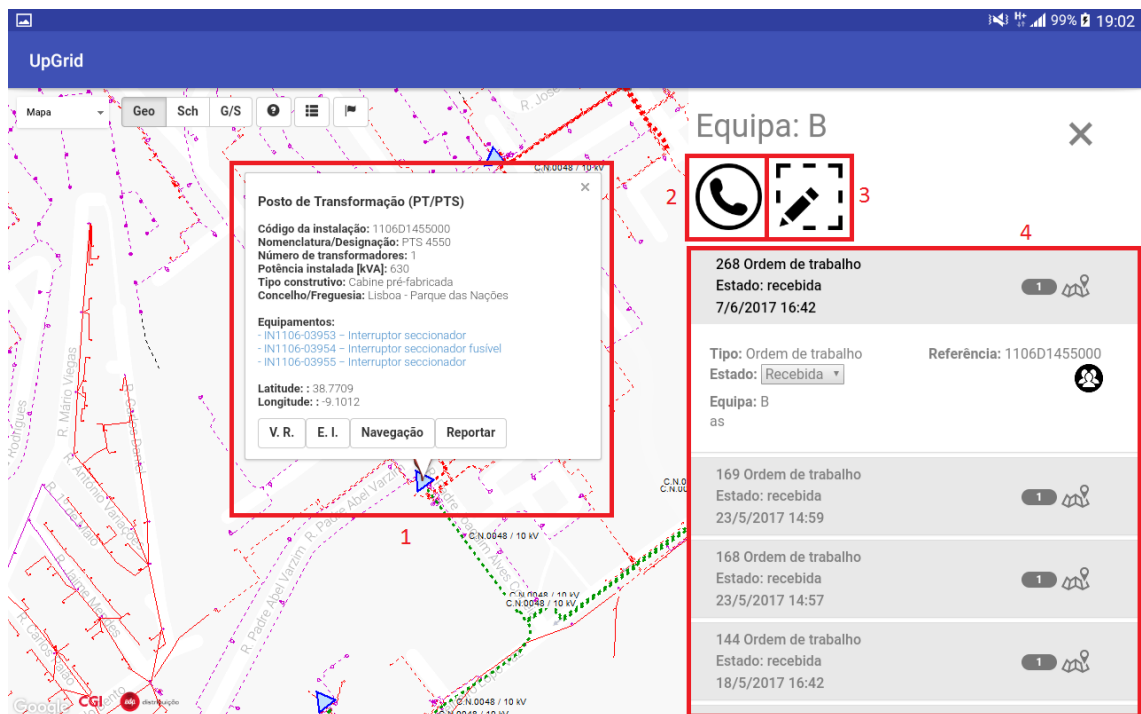


Figura 3.16 - Informação de ponto e lista de ocorrências

1. **Informação de ponto específico**

Permite visualizar informação específica de um ponto no mapa;

2. **Chamada para despacho**

Permite através de funcionalidades do Skype For Business a chamada para o despacho através de URI's;

3. **Modo colaborativo de desenho**

Permite aceder ao módulo de desenho colaborativo;

#### 4. Lista de ocorrências associadas

É possível, consultar informação referente à ocorrência, lançar a navegação para o ponto a intervir, alterar o estado da tarefa e fazer um *overview* dos pontos a intervir (ocorrência é constituída por uma ou mais tarefas).

### StreetView e esquema interno

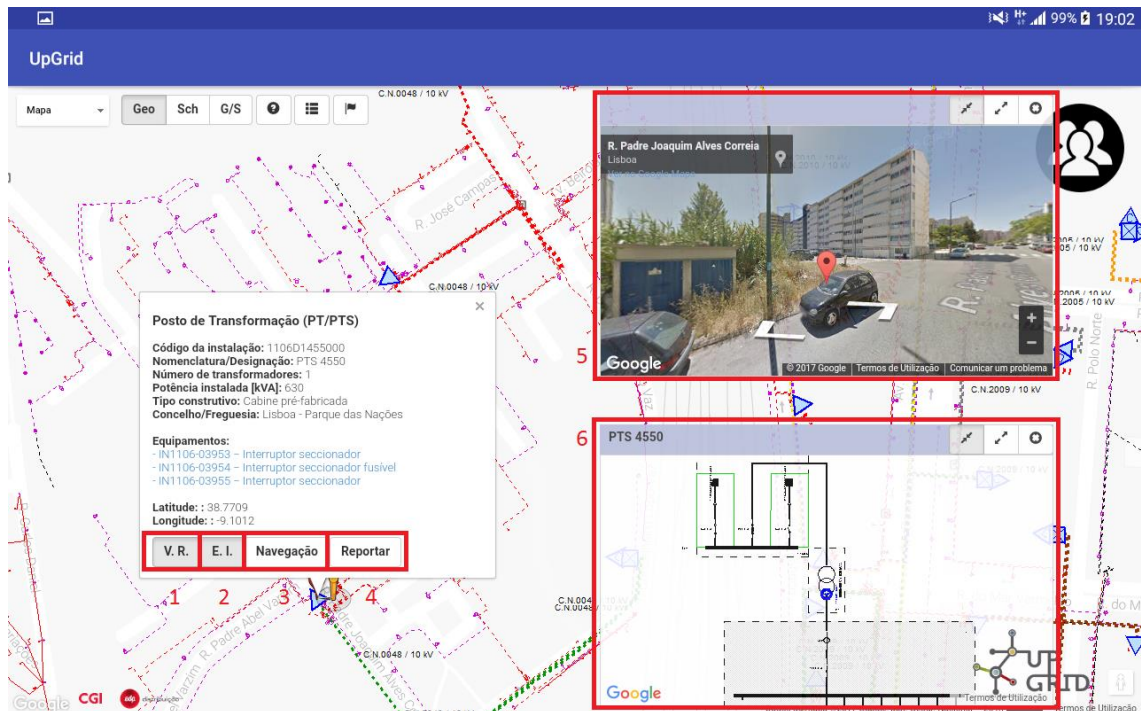


Figura 3.17 - Informação de ponto, StreetView e Esquema interno

#### 1. Botão vista de rua

Ativa o modo de vista de rua;

#### 2. Botão esquema interno

Ativa o modo visualização do esquema interno;

#### 3. Botão de navegação

Despoleta a navegação até ao ponto;

#### 4. Botão reportar dano

Permite associar as informações do equipamento para envio;

#### 5. Vista de rua

Permite visualizar o ponto utilizando o componente do Google Maps, o StreetView;

#### 6. Esquema interno

Permite visualizar o esquema interno do equipamento.

## Clustering de pontos e lista de camadas

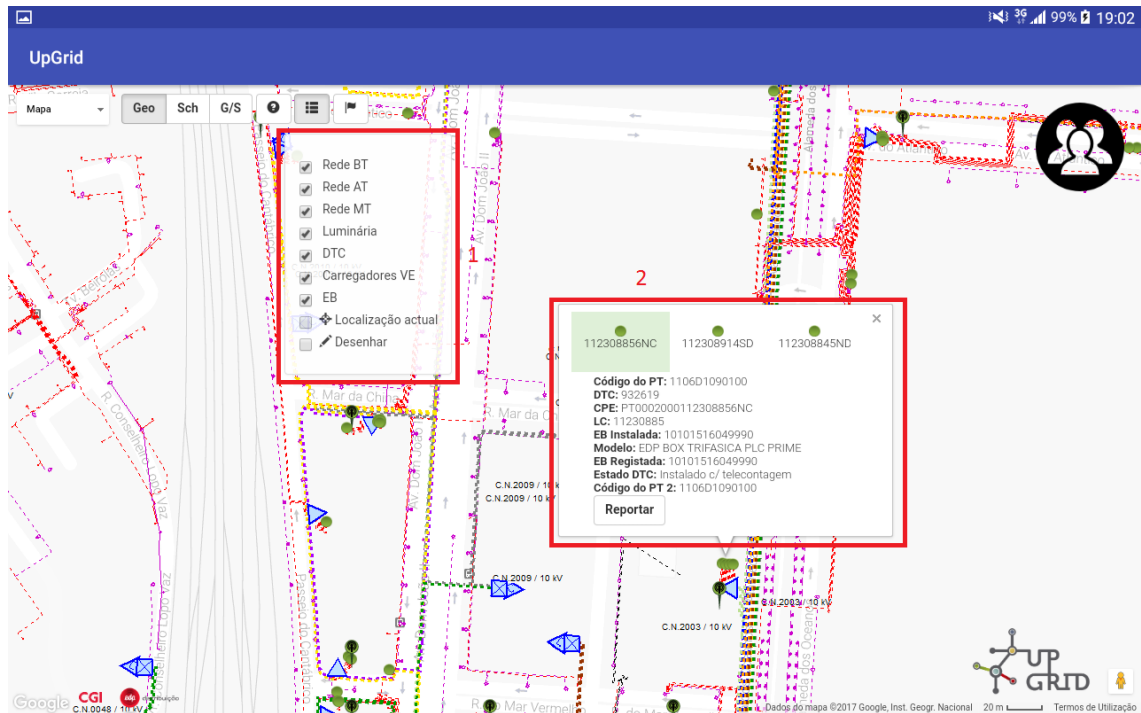


Figura 3.18 - Informação de camadas e clustering de pontos

### 1. Menu de camadas

Permite ativar as camadas de *Tile Layers* ou de informações em ponto à disposição do utilizador;

### 2. Clustering, informação de pontos

Permite consultar informação de pontos que estejam no mesmo ponto ou muito próximo.

## 3.3.4 Modo colaborativo de desenho

Este módulo, merece destaque neste relatório, pois não só, é inovador como também foi desenvolvido de raiz, o que aumentou a complexidade do mesmo, não tendo sido encontradas soluções existentes, quer gratuitas, quer pagas, que fornecessem o mesmo nível de compatibilidade e controle.

Algumas tecnologias *Open Source*, em alguns repositórios *GIT* foram analisadas, mas a metodologia ou ideologia dessas soluções não era totalmente aplicável ao que era inicialmente pensado, algumas eram aplicáveis mas não utilizavam as melhores condições de permitir uma fácil integração na solução.

Assim sendo, foi decidido implementar esta funcionalidade de raiz. À data da escrita deste relatório, a mesma ainda não se encontra totalmente terminada, mas apenas a cerca de 80% de conclusão.

## Objetivo

O objetivo pretendido desta funcionalidade, é a partilha de um *whiteboard* entre dois utilizadores, em que ambos conseguem visualizar e participar (com efeitos iguais) ambos os terminais em tempo real. Para ser utilizado, por exemplo, por um agente dentro de uma subestação onde existem muitos fusíveis de diferentes modelos e referências, com o objetivo de demonstrar com precisão o pretendido, ao engenheiro do Despacho e complementar a comunicação de telefónica. O que vem colmatar a falha de não ser possível nem fácil dar indicações por telefone sem uma imagem presente.

## WebRTC

Por ser uma funcionalidade, que exigia comunicação em tempo real, a primeira ideia que vem à cabeça no mundo do desenvolvimento web/híbrido é a utilização um servidor de *web sockets*. Felizmente várias implementações foram feitas, existindo bastante completa e conhecida, apelidada de *WebRTC* [8], que é um projeto gratuito, *Open Source* e *Open Project*. Permitindo comunicações em tempo real (RTC) por via de *API's*, este componente foi criado sendo atualmente otimizado e mantido por toda a comunidade. Também é suportado por grandes empresas como *Google*, *Mozilla* entre outras.

## PEERJS

Apesar de, a *API* ser simples, devido a muitas configurações específicas para a utilização da mesma, esta torna-se demasiado extensa para as mais comuns trocas de dados como as pretendidas. Assim sendo, foi também utilizado o *PEERJS*, que é uma implementação do *WebRTC* feita em forma de “*wrap*” que permite uma interface ainda mais simplificada e a com fácil integração em servidores com *Node.js*, que é o ideal nesta solução.

## Demonstração

Para evitar quaisquer problemas de compatibilidade, entre outros, foi decidido utilizar um componente nativo do *HTML5*, apelidada de *Canvas* (Figura 3.19). A mesma permite especificamente o desenho *on the fly* via *JavaScript*. Este elemento é apenas um *container* para gráficos, necessitando que, todas as operações a realizar pelo mesmo, sejam implementadas via *JavaScript*. O *Canvas*, disponibiliza apenas, funções básicas de desenho de caminhos, linhas, caixas, círculos, textos e permite adicionar imagens.



Figura 3.19 - Página inicial do desenho colaborativo

### 1. Informação WebRTC

Informação do componente *WebRTC*, onde é permitido saber o ID do utilizador X, para fornecer, o mesmo, ao utilizador Y de forma a iniciar a partilha. Essa comunicação passa a ser apelidada de sala, pois a colaboração pode ser para mais do que dois utilizadores, mas neste projeto apenas 1 para 1;

### 2. Ações sobre o Canvas

Grupos de ações, implementadas sobre o *canvas*, as mesmas incluem, limpar o *canvas*, tirar fotografia, adicionar imagem ao *canvas*, desenhar texto e desenhar com “lápiz” em diversas cores;

### 3. Canvas

Todas as operações, são realizadas sob este *canvas*, as operações são enviadas então por *WebRTC* para que todas as outras janelas de *canvas* possam representar o desenho pretendido.

## Exemplo demonstrativo

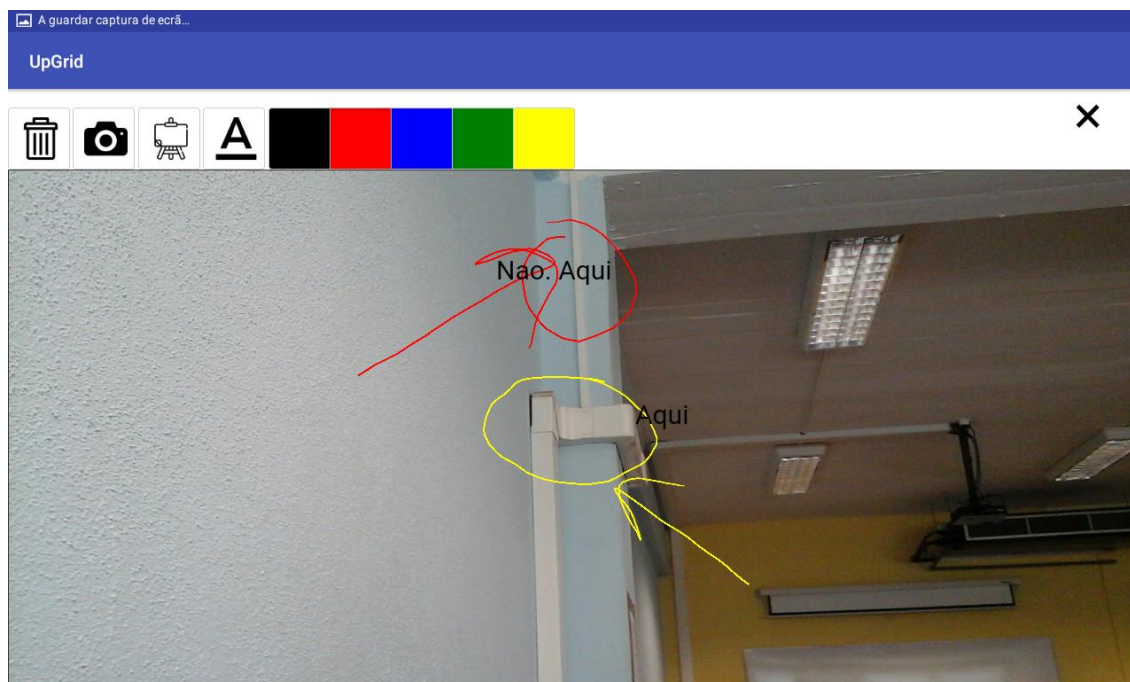


Figura 3.20 - Representação de uso do desenho colaborativo

Na figura 3.20, observa-se, um exemplo de uso, em que, o utilizador, no *tablet*, iniciou uma sessão e outro utilizador, num computador, assim sendo, consegue-se visualizar as alterações, adicionar e desenhar informação.



# Capítulo 4 Conclusões

## 4.1 Trabalho Desenvolvido

Este relatório, descreveu-se o esforço de desenvolvimento de um produto que pretende ser inovador, seja por funcionalidades, *standards* utilizados ou por preço competitivo, apesar de ser um produto, requer sempre altíssima atenção (pois a adaptação a cada cliente pode não ser linear ou requerer mesmo desenvolvimento específico).

Apesar de qualquer desvantagem que a solução possa ter, penso que na totalidade, a mesma seja um excelente projeto empresarial, possibilitando um desenvolvimento apropriado e positivo, como aluno e profissional a representar a Faculdade de Ciências da Universidade de Lisboa. Quanto à gestão do projeto, foi demonstrada confiança por parte de todos os elementos da equipa, dando sempre a possibilidade e liberdade de efetuarem as suas próprias escolhas, valorizando “as mentes novas” do projeto.

Foram descritos os módulos implementados, a divisão do produto *Diamond*, a divisão entre camadas e a sua integração em soluções existentes e a sua forma de comunicação, maioritariamente, através de *webservices*.

## 4.2 Dificuldades

É sempre um desafio importante para um estudante abraçar um projeto empresarial já inserido num contexto e negócio, o que faz com que a sua adaptação no mundo de trabalho seja mais eficaz. No entanto, isto implica uma muito maior rapidez de aprendizagem, tanto para o modelo de negócio como para todo o *know-how* de linguagens, tecnologias e paradigmas, que já estavam implementadas pela equipa.

Apesar de já contar com experiência profissional na área desde os 19 anos (tendo agora 26), este é sempre, um obstáculo inerente na vida. Cada projeto é diferente e a experiência anterior pode e deve complementar certas decisões ou preciosismos, mas, de modo geral, é sempre um desafio a utilização de novas ferramentas e tecnologias.

Todos estes princípios foram-me transmitidos pela Faculdade de Ciências da Universidade de Lisboa, e apesar de ter sido um enorme desafio, sem estes conhecimentos, a adaptação teria sido muito mais atribulada.

Outro obstáculo encontrado foi a calendarização do projeto, por ser um produto, em que, as prioridades podem mudar constantemente, sempre com o objetivo à adaptação ao mercado e aos desafios requeridos pelos clientes ou pelas entidades reguladoras, tendo esta sido bastante alterada ao longo do período de projeto. Apesar de tudo esta adaptação de calendarização, está implícita no modelo Agile, portanto verifica-se que o mesmo funciona efetivamente.

Contudo, considero que todos esses fatores contribuíram para o meu desenvolvimento pessoal e profissional, pois tudo o que aprendi será transportado para todas as outras decisões que venha a tomar.

### **4.3 Considerações Finais**

O mercado das Energias, é um mercado de relevância cada vez mais ascendente. A sua aplicabilidade no dia-a-dia, é cada vez maior estando todos nós a caminhar, cada dia mais, para um mundo das IoT, onde tudo converge para um mundo mais “Smartizado” onde tudo passou a ser de algo simples por exemplo contador, telefone ou relógio para *Smart-Meeter*, *Smart-Phone* e *Smart-Watch*.

Os objetivos inicialmente propostos, foram não só alcançados, como também superados, tendo sido realizado mais trabalho do que era expectável o que é algo admirável hoje em dia, tendo em conta que, de um modo geral no mercado de trabalho, corta-se nos tempos e nas previsões o que resulta no atraso de atividades.

### **4.4 Trabalho Futuro**

A colaboração no desenvolvimento de um produto, como neste projeto, nem sempre é linear, de momento com o objetivo mais próximo de terminar o ultimo componente referido neste relatório o *Diamond Field Agent*, de seguida e caso não hajam novas regulações impostas por parte das entidades reguladoras, será feita uma nova análise de mercado, para que seja possível uma penetração mais profunda e ampla em mais clientes. Por fim, abordar atuais clientes do produto e perceber que novas possibilidades podem ser desenvolvidas com o intuito de manter o estatuto de inovador da solução e agradar a totalidade dos clientes para que os mesmos revejam a CGI, os seus produtos e profissionais como um carimbo de qualidade.

## Capítulo 5 Bibliografia

- [1] HTML5, CSS3, Javascript development. URL: <https://www.w3schools.com/> - Consultado a: 10/06/2017
- [2] JQuery development. URL: <https://api.jquery.com/> - Consultado a: 10/06/2017
- [3] Bing Maps V8 Javascript SDK. URL: <https://www.bing.com/api/maps/sdkrelease/mapcontrol/isdk> - Consultado a: 24/03/2017
- [4] WCF usage. URL: <https://docs.microsoft.com/en-us/dotnet/framework/wcf/guide-to-the-documentation> - Consultado a: 31/03/2017
- [5] Android Design. Android open source project, 2017. URL: <https://developer.android.com/index.html> - Consultado a: 10/06/2017
- [6] Kenneth S. Rubin. Essential Scrum: A Practical Guide to the Most Popular Agile Process, 2012.
- [7] CouchDB Usage. URL: <http://docs.couchdb.org/en/1.6.1/> - Consultado a: 10/06/2017
- [8] WebRTC PEERJS usage. URL: <http://peerjs.com/> - Consultado a: 09/05/2017
- [9] Multi-tier architecture. URL: [https://en.wikipedia.org/wiki/Multitier\\_architecture](https://en.wikipedia.org/wiki/Multitier_architecture) - Consultado a: 10/06/2017
- [10] Parâmetros de nível de serviço elétrico. ERSE. URL: [http://www.erse.pt/pt/electricidade/regulamentos/qualidadedeservico/Documents/DR\\_Diretiva%2020-2013-Parametros.pdf](http://www.erse.pt/pt/electricidade/regulamentos/qualidadedeservico/Documents/DR_Diretiva%2020-2013-Parametros.pdf) - Consultado a: 10/06/2017
- [11] Parâmetros de nível de serviço Gás. ERSE. URL: [http://www.erse.pt/pt/gasnatural/regulamentos/qualidadedeservico/Documents/RQS\\_GN\\_abril\\_2013.pdf](http://www.erse.pt/pt/gasnatural/regulamentos/qualidadedeservico/Documents/RQS_GN_abril_2013.pdf) - Consultado a: 10/06/2017
- [12] Parâmetros de nível de serviço Agua, Saneamento e Resíduos. ERSAR. URL: [http://files.isec.pt/mwg-internal/de5fs23hu73ds/progress?id=Am6\\_dsF1dhzIfCv6ydFNHw7IdZTyL1YOg7zCYF64zwc,&dl](http://files.isec.pt/mwg-internal/de5fs23hu73ds/progress?id=Am6_dsF1dhzIfCv6ydFNHw7IdZTyL1YOg7zCYF64zwc,&dl) - Consultado a: 10/06/2017

# Acrónimos

AD – Active Directory  
AGS – Administração e gestão de sistemas de salubridade  
API – Application Programming Interface  
BLL – Business Logic Layer  
DAL – Data Access Layer  
ERSAR – Entidade Reguladora dos Serviços de Aguas e Resíduos  
ERSE – Entidade reguladora dos serviços energéticos  
HTTP – Hypertext Transfer Protocol  
IDE – Integrated Development Environment  
IQS – Indicadores de qualidade de serviço  
J2EE – Java 2 Enterprise Edition  
JMS – Java Message Service  
JSON – JavaScript Object Notation  
LDAP – Lightweight Directory Access  
NoSQL – Non SQL or Non relational  
OT – Ordens de trabalho  
PL – Presentation Layer  
QREN – Quadro de Referência Nacional  
RTC – Real Time Communication  
SCADA – Supervisory Control And Data Acquisition  
SDK – Software Development Kit  
SIG – Sistema de informação Geográfico  
TER – Tempo estimado de reposição  
URI – Uniform Resource Identifier  
URL – Uniform Resource Locator  
WCF – Windows Communication Foundation  
XML – Extensible Markup Language