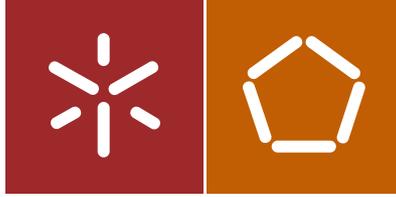




Universidade do Minho  
Escola de Engenharia

Carlos Diogo Lima Gonçalves

Ajustamento dos Papéis do RUP Reduzido  
no Desenvolvimento de Projetos em  
Contexto Académico



Universidade do Minho  
Escola de Engenharia

Carlos Diogo Lima Gonçalves

Ajustamento dos Papéis do RUP Reduzido  
no Desenvolvimento de Projetos em  
Contexto Académico

Tese de Mestrado  
Engenharia e Gestão de Sistemas de Informação

Trabalho efetuado sob a orientação do  
Professor Doutor Pedro Ribeiro

## Resumo

O *Rational Unified Process* globalmente denominado de RUP é uma *framework* de desenvolvimento de Software proposto pela IBM, com uma grande divulgação em termos mundiais. No entanto, não significa que seja uma *framework* simples de seguir e com uma curva de aprendizagem rápida, pelo contrário, é extremamente complexa e exige uma forte compreensão do seu funcionamento.

Esta *framework* é também utilizado no desenvolvimento de projetos de baixa complexidade, como por exemplo, em projetos universitários. No entanto, devido à complexidade da mesma, exige aos alunos um investimento grande de tempo na sua compreensão e na afetação de recursos humanos, podendo dificultar o seu desempenho e, eventualmente, afetar o sucesso do próprio projeto.

A nível nacional, grande parte das empresas de desenvolvimento de Software são consideradas Pequenas e Médias Empresas (PMEs). Estas organizações, de dimensões reduzidas, possuem um número limitado de recursos humanos nas suas equipas de desenvolvimento, o que dificulta a resposta aos desafios exigidos.

No sentido de aumentar a produtividade e desempenho dos colaboradores no desenvolvimento de projetos de pequenas dimensões, em 2007 foi proposto por Borges (Borges, 2007) um modelo Reduzido do RUP, onde a configuração do elenco processual do mesmo foi adaptada para que os recursos humanos das equipas de pequenas dimensões pudessem incorporar todo o leque de responsabilidades e, desta forma, utilizar a metodologia de forma competitiva, comparativamente a organizações de grande dimensão.

Apesar do modelo Reduzido do RUP ter sido desenvolvido, este não sofreu experimentação suficiente para o tornar válido em contextos reais. O propósito desta dissertação é avaliar o desempenho desse modelo em contexto académico simulando um contexto real, onde as equipas de desenvolvimento utilizam o modelo Reduzido do RUP para resolver os problemas propostos por um cliente real.

Adicionalmente pretende-se fazer, após análise dos resultados obtidos do estudo das equipas de desenvolvimento, um reajustamento do modelo Reduzido do RUP, no sentido de o

adequar ao nível de complexidade exigida e aumentar a taxa de sucesso em futuras aplicações em contexto real.

# Abstract

The Rational Unified Process, globally named RUP, is a software development framework proposed by IBM with a wide dispersion globally. However, it isn't meant to be seen as a simple framework to follow, with an easy learning curve, on the contrary, it is extremely complex and requires a strong understanding of its operation.

This framework is also used in the development of low-complexity projects, as an example, the university projects. Due this framework complexity, it demands from students a great amount of time invested in its understanding and in human resources allocation, consequently impaired the employees' performance, and additionally, it can in various situations affect the success of the project itself.

In Portugal, the majority of companies are considered Small and Medium Enterprises (SMEs). These organizations, that have reduced dimensions, have a limited amount of human resources in their development teams, making it difficult to respond to the required challenges.

In order to increase productivity and performance of employees in project development, in 2007, Borges proposed a RUP Reduced model (Borges, 2007), where the configuration of RUP roles has been adapted so that human resources within small organizations could incorporate the full range of RUP model responsibilities and with that, small development teams could compete against large organizations development teams.

Despite Reduced RUP model has been developed, it has not undergone enough experimentation in order to be applied in real contexts. The purpose of this dissertation is to evaluate the model's performance in an academic context, simulating a real context where development teams use the Reduced Model RUP to solve problems presented by a real client.

In addition it is intended to do, after analyzing the results of the development teams study case, an adjustment of the Reduced RUP model, in order to tune it to the level of complexity required and in the future increase the application success rate in a real organization context.



## Agradecimentos

Quero agradecer ao meu orientador Professor Pedro Ribeiro pela ajuda e orientação dada durante este longo percurso, assim como, a disponibilidade demonstrada.

Agradeço também à minha madrinha Lisete, pois sem ela nada disto seria possível, à minha tia Júlia pelo contínuo suporte e esforço para o meu sucesso durante todos estes anos e à minha tia Margarida pelo apoio dado durante o meu percurso académico.

Agradeço ainda ao resto da minha família e amigos pelo constante incentivo durante a realização da dissertação.

Agradeço às equipas de desenvolvimento da unidade curricular de Desenvolvimento de Aplicações Informáticas do segundo ano do Mestrado Integrado em Engenharia e Gestão de Sistemas de Informação, do ano letivo 2014/2015, que participaram nas entrevistas e, assim, tornaram possível a realização deste projeto.

Um agradecimento extra à minha incansável tia e amiga Júlia, por se voluntariar para realizar a revisão desta dissertação.



# Índice

Resumo.....	i
Abstract .....	iii
Agradecimentos .....	v
1 Introdução.....	1
1.1 Surgimento da Engenharia de Software e o seu desenvolvimento em Portugal .....	1
1.2 Competitividade das Pequenas ou Médias Empresas em Portugal.....	3
1.3 Objetivos propostos .....	5
1.4 Organização da Dissertação .....	6
2 Abordagem Metodológica .....	9
2.1 Escolha da abordagem .....	9
2.2 Estruturar o Estudo de Caso.....	11
2.3 Preparar para a Recolha de Dados.....	12
2.4 Recolher os Dados.....	12
2.5 Analisar os Dados Recolhidos.....	14
3 Revisão de Literatura .....	15
3.1 Contextualização do Rational Unified Process.....	15
3.2 Caracterização e Definição do <i>Rational Unified Process</i> .....	16
3.3 Definição e Caracterização do Elenco Processual do RUP .....	19
3.4 Identificação do elenco processual do RUP no contexto de PME.....	28
3.4.1 Identificação do modelo Base .....	31
3.4.2 Identificação do modelo Reduzido.....	37
3.5 Caracterização do elenco processual do modelo Reduzido.....	39
4 Acompanhamento das Equipas de Desenvolvimento de Projetos .....	81
4.1 Caracterização da maturidade das equipas de desenvolvimento .....	83
4.1.1 Avaliação da distribuição das faixas etárias e do género.....	84
4.1.2 Relações entre unidades curriculares, inscrições e estatuto do aluno .....	87
4.1.3 Distribuição dos papéis pelos elementos das equipas.....	91
4.1.4 Análise do conhecimento sobre as competências e responsabilidades .....	98
4.1.5 Sintetização e Conclusões .....	100
4.2 Análise do tempo investido pelas equipas no desenvolvimento do projeto.....	101
4.2.1 Análises dos dados relativos ao esforço investido pelas equipas .....	101
4.2.2 Conclusões e sugestões de reajustamento dos Papeis do modelo reduzido do RUP	111

5	Conclusões e Trabalho Futuro.....	118
5.1	Sintetização de Resultados.....	118
5.2	Trabalho Futuro.....	121
	Bibliografia .....	123
	Anexos.....	125
	Anexo A – Definição do Elenco Processual do RUP.....	126
	Anexo B – Comparação entre o modelo Base e o modelo Reduzido.....	133
	Anexo C – Gráficos inteiro de comparação do esforço dos papéis com o esforço médio definido .....	134
	Anexo D – Papéis do Modelo Base que não se encontram no Reduzido .....	136

## Índice de Tabelas

Tabela 1 - Diferenciação de estratégias de pesquisa (Yin, 2003).....	10
Tabela 2 - Descrição e Identificação dos cargos e responsabilidades elenco original.....	20
Tabela 3 - Grupos de divisão do elenco processual v2001.....	21
Tabela 4 – Identificação do elenco v2001 definido no RUP e respetivos distribuição por grupos.....	22
Tabela 5 - Identificação do elenco v2007 definido por Borges e respetivos distribuição por grupos .....	28
Tabela 6 - Ferramentas alternativas aos processos do RUP .....	30
Tabela 7 - Condições para ser considerado um papel principal.....	32
Tabela 8 - Definição do Modelo Base.....	35
Tabela 9 - Mapeamento do Modelo Base .....	37
Tabela 10 - Mapeamento do modelo Reduzido.....	39
Tabela 11 - Artefactos a desenvolver durante os momentos de avaliação .....	82
Tabela 12 - Número de intervenientes responsáveis por cada papel nos diferentes grupos de desenvolvimento.....	112
Tabela 13 - Horas médias investidas pelas equipas em cada papel.....	113
Tabela 14 - Distribuição dos Intervenientes pelos papéis do EPRR .....	115
Tabela 15 - Distribuição dos recursos humanos .....	116
Tabela 16 - Elenco Processual - Grupo <i>Analyst</i> .....	127
Tabela 17 - Elenco Processual - Grupo <i>Developer</i> .....	129
Tabela 18 - Elenco Processual - Grupo <i>Manager</i> .....	130
Tabela 19 - Elenco Processual - Grupo <i>Tester</i> .....	131
Tabela 20 - Elenco Processual - Grupo <i>Production and Support</i> .....	132
Tabela 21 - Elenco Processual - Grupo <i>Additional</i> .....	132
Tabela 22 - Comparação entre o modelo Base e o modelo Reduzido .....	133



## Índice de Figuras

Figura 1 – <i>Rational Unified Process</i> (Kruchten, 2003) .....	17
Figura 2 - Grupos etários dos intervenientes.....	85
Figura 3 - Distribuição do Género.....	85
Figura 4 - Estatuto dos Intervenientes do Projeto .....	87
Figura 5 - Relação entre Inscrições efetuadas no curso e o ano atual .....	88
Figura 6 - Relação entre as unidades curriculares em atraso e o ano atual de inscrição.....	89
Figura 7 - Relação entre Unidades Curriculares em atraso e Inscrições efetuadas .....	90
Figura 8 - Constituição dos Grupos de Desenvolvimento .....	91
Figura 9 - Distribuição dos papéis pelas equipas de desenvolvimento (1/4).....	92
Figura 10 - Distribuição dos papéis pelas equipas de desenvolvimento (2/4).....	92
Figura 11 - Distribuição dos papéis pelas equipas de desenvolvimento (3/4).....	93
Figura 12 - Distribuição dos papéis pelas equipas de desenvolvimento (4/4).....	93
Figura 13 - Distribuição dos papéis incluindo acumulação de responsabilidades (1/4) .....	95
Figura 14 - Distribuição dos papéis incluindo acumulação de responsabilidades (2/4) .....	95
Figura 15 - Distribuição dos papéis incluindo acumulação de responsabilidades (3/4) .....	96
Figura 16 - Distribuição dos papéis incluindo acumulação de responsabilidades (4/4) .....	96
Figura 17 - Experiência externa na execução das responsabilidades do papel atribuído .....	98
Figura 19 - Perceção das responsabilidades do papel que está a desempenhar .....	99
Figura 18 - Perceção das responsabilidades de todo o elenco processual do modelo .....	99
Figura 20 – Constituição dos grupos de desenvolvimento no final do projeto .....	101
Figura 21 - Número total de horas investido pelas equipas de desenvolvimento .....	102
Figura 22 - Média de horas investida por cada elemento das equipas de desenvolvimento.....	103
Figura 23 - Horas totais investidas nos momentos de avaliação em relação ao número de semanas ...	104
Figura 24 - Média de horas de cada elemento das equipas durante os diversos Momentos de avaliação .....	106
Figura 25 - Esforço total investido pelas equipas em cada papel para a realização do projeto.....	106
Figura 26 - Esforço médio investido em cada papel no decorrer do projeto.....	107
Figura 27 - Comparação entre o esforço de cada papel e o esforço normal médio no Momento 1 .....	109
Figura 28 - Comparação entre o esforço de cada papel e o esforço normal médio no Momento 2 .....	109
Figura 29 - Comparação entre o esforço de cada papel e o esforço normal médio no Momento 3 .....	110
Figura 30 - Comparação entre o esforço de cada papel e o esforço normal médio no Momento 5 .....	110
Figura 31 - Identificação do Modelo do EPRR.....	114
Figura 32 - Comparação entre o esforço de cada papel e o esforço normal médio nos diferentes Momentos .....	134



# 1 Introdução

Este capítulo dá início à dissertação e destina-se a dar uma visão global do cenário em que se encontra o desenvolvimento de Software em Portugal. Identifica as possíveis dificuldades pelas quais as PME's se encontram no momento, quando estão em concorrência com as grandes organizações, apresentam-se em seguida os objetivos desta dissertação, bem como, os contributos possíveis que esta tese pretende fornecer e, por último, a estrutura da própria dissertação.

## 1.1 Surgimento da Engenharia de Software e o seu desenvolvimento em Portugal

Após o surgimento do termo Engenharia de Software numa conferência em que *Fritz Bauer* participou onde discursou e definiu o termo anterior como, "... a utilização dos princípios básicos da engenharia para obter, de uma forma economicamente viável, Software fiável e que corra eficientemente em computadores reais" (Naur et al., 1968), verificou-se uma propagação da investigação científica e das suas práticas nessa área, fazendo com que o próprio termo fosse mais relevante para as práticas organizacionais.

Em 1970 houve inevitavelmente, em Portugal, o surgimento das primeiras investigações científicas e a sua aplicação ao nível académico na área de Engenharia de Software (Grancho, 2005). Décadas depois deste surgimento, houve um aumento drástico da procura de profissionais especializados neste setor de mercado muito superior ao número de profissionais capazes de obedecer a essas condições que eram formados nas universidades. Como consequência destes eventos, foi possível visualizar uma forte competição entre as organizações para contratar o máximo possível de profissionais qualificados da área, de forma a conseguirem responder à grande procura de mercado que se encontrava em expansão na altura.

Após alguns anos, perto de 2001, ocorreu uma recessão económica que afetou também a área de desenvolvimento de Software, devido ao facto de ter diminuído significativamente a procura de projetos de desenvolvimento, causando assim, um aumento da competitividade entre organizações e criando dependências entre as entidades públicas com o objetivo de amortizar a diminuição na procura de projetos de desenvolvimento de Software.

Isto fez com que as equipas de desenvolvimento de Software sofressem mais o efeito do *stress* e da pressão devido ao facto de serem obrigadas a demonstrar, o mais rapidamente possível, resultados de qualidade e utilizando o menor número de recursos possível, para que, a organização dessa equipa continuasse a ser seleccionada como a melhor para o desenvolvimento dos projetos.

Em termos genéricos, a qualidade de um produto é avaliada através do cumprimento das especificações inicialmente definidas para esse produto e não pelo seu aspeto elegante ou pelo facto de possuir a melhor tecnologia integrada (Fernandes & Duarte, 2005). Assim sendo, para melhor satisfação das partes interessadas é necessário que o processo de desenvolvimento do Software cumpra os requisitos recolhidos e seja acompanhado pelo cliente, durante o processo de desenvolvimento, de uma forma interativa (Duarte, 2002).

Apesar disso, no agregado de profissionais, existiam aqueles que consideravam não possuir competências profissionais suficiente para o trabalho do dia-a-dia a que estavam sujeitos. Na realidade, devido à falta de atividades práticas (e em alguns casos mesmo teóricas) que não constavam dos currículos das instituições académicas, nomeadamente dos institutos onde se formaram muitos dos profissionais, começa-se a notar uma insuficiência de técnicos especializados em determinadas áreas fundamentais ao desenvolvimento de projetos, como é o caso, de gestão de projetos, competências sociais e, de maior relevância ainda, capacidade de trabalhar em equipa, visto que, qualquer projeto, mesmo sendo de pequena complexidade, exige que se forme pelo menos uma equipa de trabalho para a resolução do problema.

Se, anteriormente, as capacidades técnicas elevadas eram suficiente para que os profissionais conseguissem lugar nas organizações, essa condição está ultrapassada pelo surgimento dos concorrentes internacionais, denominados “fábricas de Software” que, ao contrário de Portugal, possuem estruturas bem organizadas, solidificadas e com baixos gastos nos salários devido às suas localizações (Índia, China, etc.), como mostram os níveis na escala do CMM (Weber et al., 1996) (Fuggetta & Di Nitto, 2014). As pirâmides hierárquicas, que eram as estruturas organizacionais existentes na altura, criavam estruturas demasiado rígidas e pouco flexíveis para ser adequado a trabalhos em equipa e, com isso, dificultavam a comunicação entre os processos, fazendo assim com que todo o desenvolvimento do projeto despendesse mais tempo e recursos. No sentido de contrariar essas dificuldades, tornou-se premente uma reformulação

das áreas de ensino, levando assim a adoção do modelo de Bolonha, que originou uma revisão e reformulação nas ofertas das universidades, focando assim os conteúdos de ensino nas competências centrais que cada curso possuía e aumentando a quantidade de áreas de especialização das mesmas, proporcionando assim, melhor resposta ao mercado de trabalho.

Com isto, o surgimento de organizações orientadas a processos tornou-se uma realidade. Nessas organizações não existe a noção de cadeia hierárquica e os colaboradores da organização são divididos em grupos, com base nas competências específicas para que foram contratados. Essas competências são extremamente específicas e de diversas áreas do espectro como, por exemplo, a gestão, a logística e a informática dependendo, das necessidades da organização. Quando existe a necessidade de realização de um projeto, são selecionados elementos dos vários grupos e designado um gestor de projeto que se torna responsável por coordenar a equipa no decorrer do mesmo. Fica assim, atribuída a esta equipa a responsabilidade de realização do projeto e a qualidade do mesmo, fica dependente do sucesso da equipa na realização do processo.

Neste sentido é relevante enfatizar que, neste tipo de ambiente organizacional no qual o desenvolvimento de Software é influenciado pela forma como os processos organizacionais são tratados dentro da própria organização, existe uma necessidade de constante aperfeiçoamento da estrutura organizacional para o desenvolvimento do processo, de forma a melhorar a produtividade da organização, a qualidade do produto, a posição no mercado relativamente a concorrência, etc. e, sobretudo, a capacidade de responder a diversos tipos de pedidos. Em suma uma diferente estruturação da equipa permite uma melhor resposta do que a estrutura da organização convencional.

## **1.2 Competitividade das Pequenas ou Médias Empresas em Portugal**

Uma Pequena ou Média Empresa (PME) pode ser definida como uma organização que possui menos de 250 colaboradores e apresenta um volume de negócios igual ou inferior a 50 milhões de euros ou um balanço de menos de 43 milhões de euros (IAPMEI, 2016). De acordo com o IAPMEI, Portugal conta com 99,6% de todas as empresas nacionais como PMEs.

Segundo Beira (Beira et al., 2005), o espectro organizacional que agrupa o conjunto de produtores de Software é caracterizados por ser na sua maioria, organizações de pequena e média dimensão. Apesar disso, encontram-se em competição e concorrência com outras organizações

internacionais e nacionais de maior dimensão assim como organizações com recursos e capacidades de realizar esquemas de subcontratação.

Para além desta competitividade existe ainda a disponibilidade de recursos o que demarca a capacidade de aceder a ações de formação e de ferramentas e equipamentos mais poderosos e com maior facilidade de integração. Com isto existe uma necessidade de criar processos estruturais de trabalho que utilizem as capacidades e conhecimentos individuais de cada elemento de forma a aumentar o conhecimento partilhado entre diversos elementos da equipa (Krahn et al., 2014). Neste tipo de organizações existem problemas ao nível da coordenação devido ao facto dos processos não estarem normalizados e por isso possuem dependências em determinados recursos que são comuns a vários cargos, tendo como consequência diversos colaboradores a realizar o mesmo tipo de tarefa em momentos diferentes.

Outro problema encontrado pelas organizações onde o número de colaboradores é muito inferior ao dos grupos formados nas grandes organizações é a dificuldade de seguirem metodologias que foram criadas com o intuito de serem utilizadas por equipas que possuam um elevado número de membros, onde os problemas que surgem possuem um elevado nível de complexidade e requerem que esse mesmo problema seja repartido em partes mais simples e posteriormente distribuído por diversas equipas (Monteiro et al., 2012).

Existe ainda a situação onde uma organização não possui colaboradores suficientes para a formação de uma equipa que consiga obedecer às estruturas rígidas definidas pelos moldes que as grandes organizações utilizam, fazendo com que alguns colaboradores da organização necessitem de ocupar diversos cargos funcionais ao mesmo tempo o que na maior parte das vezes cria problemas de conflitos entre prazos e prioridade das tarefas solicitadas por diferentes responsáveis.

Deste modo, no desenvolvimento de sistemas, nomeadamente de Software, tem que se garantir que o produto final tenha qualidade e que o seu processo de desenvolvimento seja controlado. Torna-se necessário que tanto os produtos de Software como os processos que lhe deram origem sejam adequados ao tipo de organizações encontradas presentemente. A utilização de processos *standards* genéricos, bem definidos, documentados e geridos apresenta vantagens. Com isso existe uma necessidade de criar esses mesmos modelos para que sejam adaptados a pequenas equipas de desenvolvimento.

### 1.3 Objetivos propostos

Pelo descrito anteriormente esta dissertação tem como objetivo principal validar o elenco processual do modelo Reduzido do RUP proposto por Borges na sua dissertação (Borges, 2007), esse trabalho teve como objetivo encontrar ou caso necessário redefinir uma metodologia adequada a organizações de pequenas e médias dimensões, devido ao facto da maioria do tecido empresarial nacional ser dessas mesmas dimensões, assim como se necessário, redefinir algum dos papéis para que possam ser cumpridos os objetivos definidos para melhorar a produtividade organizacional, dos quais de destacam os seguintes:

- Uniformizar o esforço de cada papel do elenco processual do modelo Reduzido;
- Avaliar de que forma essa uniformização influenciou a produtividade das equipas;
- Diminuir o tempo e custo de formação das equipas até possuírem competências suficientes para desempenhar os papéis do modelo.

Para a realização deste projeto pretende-se efetuar um estudo de caso prático que irá envolver as equipas de 2º ano da unidade curricular de Desenvolvimento de Aplicações Informáticas, tendo por objetivo principal fazer o seu acompanhamento, com o intuito de recolher informações e dados acerca do funcionamento das mesmas equipas durante o decorrer do semestre letivo, para que, no final, possa ser possível uma análise e avaliação mais detalhada da configuração do elenco processual do RUP reduzido.

Como resultados, espera-se que através do estudo do caso prático exista uma criação de valor e feedback suficiente para permitir o tratamento futuro e contribuir para o melhoramento, ainda mais acentuado, dos papéis do modelo Reduzido do RUP para que possa ser implementado no futuro nas PMEs, tornando-as, assim, mais competitivos com o mercado concorrente atual, sendo estas, as organizações de grandes dimensões.

Por fim, é também oportuno identificar que ainda não se pretende com esta dissertação avaliar o desempenho do elenco processual em contextos reais, uma vez que, seria necessário um investimento elevado de tempo para a realização de tal processo, além de não se enquadrar nos objetivos desta dissertação. Em vez disso, pretende-se adequar ao contexto académico com o objetivo de recolher alguns dados e comportamentos que possam simplificar a implementação, em contexto reais, para futuras investigações.

## 1.4 Organização da Dissertação

Nesta secção será apresentada a organização estrutural desta dissertação, assim como, uma breve explicação do conteúdo que pode ser encontrado dentro de cada capítulo. Além do presente capítulo, esta dissertação contém, ainda, mais quatro capítulos, apresentados, de forma sucinta, de seguida:

1. Introdução: Este capítulo dá início à dissertação e destina-se a dar uma visão global do cenário em que se encontra o desenvolvimento de Software em Portugal. Identifica as possíveis dificuldades pelas quais as PME se encontram no momento, quando estão em concorrência com as grandes organizações. Seguem-se os objetivos desta dissertação, bem como, os contributos possíveis que esta tese pretende fornecer e, por último, a estrutura organizacional da própria dissertação.
2. Abordagem Metodológica: Neste capítulo é possível visualizar o tipo de abordagem metodológica que foi utilizada para a realização da componente prática desta dissertação, a razão pela qual foi escolhida essa abordagem metodológica, assim como, as fases e processos que devem ser seguidos, de forma a fazer uma utilização correta dessa mesma abordagem.
3. Revisão de Literatura: Revisão de Literatura: Este capítulo apresenta a revisão de literatura relacionada com o *Rational Unified Process*, fazendo inicialmente a identificação e descrição da sua arquitetura processual e, posteriormente, realizando uma análise cuidadosa ao elenco processual original (Rational, 1998). Por último podemos ainda encontrar neste capítulo a identificação e descrição do elenco processual do RUP reduzido definido e identificado por Borges (Borges, 2007) na realização da sua dissertação.
4. Acompanhamento de Equipas de Desenvolvimento de Projetos: Este capítulo envolve a componente prática desta dissertação, na qual foi aplicada a abordagem metodológica de estudo de caso às equipas de desenvolvimento da unidade curricular de Desenvolvimento de Aplicações Informáticas. Neste capítulo é ainda possível visualizar todos os dados recolhidos relacionados com os grupos de trabalho, bem como, as respetivas análises e conclusões. Por fim, apresentam-se

algumas sugestões de alterações aos modelos Reduzido e Base, de acordo com as conclusões retiradas, com base nos dados analisados.

5. Conclusões e Trabalho Futuro: Este capítulo tem como objetivo apresentar algumas conclusões resultantes da análise dos dados do modelo reduzido do RUP e que se espera que tenham utilidade prática para futuras adaptações. Outro assunto que abordado neste capítulo refere-se a propostas de temas para futuros projetos, que possam dar continuidade ao trabalho desenvolvido nesta dissertação.



## 2 Abordagem Metodológica

Este capítulo dá início à dissertação e destina-se a dar uma visão global do cenário em que se encontra o desenvolvimento de Software em Portugal. Identifica as possíveis dificuldades pelas quais as PME se encontram no momento, quando estão em concorrência com as grandes organizações. Seguem-se os objetivos desta dissertação, bem como, os contributos possíveis que esta tese pretende fornecer e, por último, a estrutura organizacional da própria dissertação.

### 2.1 Escolha da abordagem

O estudo de caso é uma abordagem metodológica de pesquisa muito usada em diversas áreas, onde podemos incluir ciências políticas, investigação da administração pública, estudos organizacionais e de gestão, entre outros, tais como, na orientação de dissertações nas áreas académicas ou em ramos profissionais como gestão de negócios, administração de negócios e gestão das ciências sociais (Yin, 1994).

Existem quatro outros tipos de estratégias que podemos usar no âmbito do estudo de caso: experimentação, questionários, análises de arquivos e histórica. A sua aplicação facilita a utilização da abordagem estudo de caso. Cada uma destas estratégias possui pontos fortes e limitações, para tirar partido dos pontos fortes é necessário saber as diferenças entre cada uma delas.

Estas abordagens podem ser usadas para o propósito de exploração, descrição e explicação, por outro lado, as três condições que distinguem as abordagens consistem em (Yin, 2003):

1. Tipo de questão que se pretende responder com a investigação;
2. A extensão de controlo que o agente consegue ter sobre o comportamento dos eventos.
3. A necessidade de prestar atenção a eventos contemporâneos em vez de utilizar os eventos históricos passados.

Uma forma simples de categorizar o tipo de questões necessárias para responder à primeira condição é: Quem?, Qual é?, Onde?, Como? e Porquê?

Na Tabela 1(Yin, 2003) encontram-se a relação entre as estratégias e as condições que distinguem as mesmas.

<b>Estratégia</b>	<b>Pergunta de Investigação</b>	<b>É necessário controlo sobre o comportamento do evento?</b>	<b>É necessário prestar atenção nos eventos contemporâneos?</b>
<b>Experimentação</b>	Como?, Porquê?	Sim	Sim
<b>Questionários</b>	Quem?, O que?, Onde?, Quantos são?, Quanto é?	Não	Sim
<b>Análises de Arquivos</b>	Quem?, O que?, Onde?, Quantos são?, Quanto é?	Não	Sim/Não
<b>Investigação Histórica</b>	Como?, Porquê?	Não	Não
<b>Estudo de Caso</b>	Como?, Porquê?	Não	Sim

Tabela 1 - Diferenciação de estratégias de pesquisa (Yin, 2003)

Definir a pergunta de investigação é o passo mais relevante na iniciação de um estudo de investigação e, por essa razão, é necessário algum tempo de reflexão e paciência para se obter uma questão que responda às necessidades da investigação.

Devido ao fato de não possuir controlo sobre o comportamento dos eventos e do tema desta dissertação se focar em eventos contemporâneos, apenas as estratégias Questionários e Estudo de Caso são válidas para escolha, mas como a questão que se pretende ver respondida é: “Como devo ajustar os papéis do RUP para aumentar a produtividade em ambiente académico?”, a estratégia que melhor se adequa é o estudo de caso.

As fases para a realização de um estudo de caso podem ser divididas em:

- a. **Estruturar o Estudo de Caso**
- b. **Preparar para a Recolha de Dados**
- c. **Recolher os Dados**
- d. **Analisar os Dados Recolhidos**

Cada uma destas fases possui dentro de si diferentes tipos de tarefas e atividades que devem ser executadas. Em seguida será explicado o que se deve realizar para cumprir com sucesso cada fase desta estratégia

## 2.2 Estruturar o Estudo de Caso

Depois da escolha do estudo de caso para a estratégia de investigação, foi necessário **estruturar o estudo de caso (a)** em si. Para isso, como em qualquer outro tipo de estratégia de investigação, foi necessário fazer um plano.

No planeamento do estudo de caso existem cinco componentes relevantes para a sua estrutura. A **Questão em Estudo (1)**, como já foi descrito em cima é a componente que é usada para distinguir qual a estratégia mais relevante a ser implementada no desenvolvimento do projeto (Yin, 2003).

Relativamente à segunda componente, **Proposições do Estudo (2)**, esta possui como objetivo principal redirecionar o foco de atenção para alguma coisa que deve ser investigada na área de estudo. Por exemplo, “Como se devem ajustar os papéis do RUP para aumentar a produtividade em ambiente académico?”, o “Como” captura o que está realmente interessado em ser respondido. Caso o “Como” não seja exatamente o que se pretende responder, só com a aplicação de uma proposição é possível mudar o rumo para a direção pretendida.

A terceira componente, **Unidade de Análise (3)**, que possui como objetivo a definição do “caso”, sendo que, o caso pode incluir indivíduos, grupos e organizações e que na maior parte das vezes, esta componente torna-se difícil de realizar devido ao facto de, frequentemente se encontrar dificuldades na descrição do caso (Yin, 1994). Na definição de caso, podemos definir barreiras temporais onde identificamos o início e o fim do caso e, com isto, conseguimos alguma flexibilidade na descrição do caso.

A componente de **Ligação de dados a Proposições (4)** e o **Critério de Interpretação de Resultados (5)** não estão muito bem definidos para o estudo de caso. Estas componentes representam os passos para a análise de dados na investigação dos casos de estudo. Em suma o planeamento deve indicar que tipos de dados devem ser recolhidos com as componentes **(1)**, **(2)**, **(3)** mas, deve também indicar o que se deve fazer com os dados depois de recolhidos, componentes **(4)** e **(5)** (Yin, 2003).

## 2.3 Preparar para a Recolha de Dados

A **preparação para a recolha de dados (b)** é uma etapa importante, complexa e difícil devido ao facto de que, se esta etapa não for definida corretamente, pode comprometer todo o estudo de caso. Para esta fase existem quatro tópicos relevantes que devem existir para que a recolha de dados seja realizada com mais clareza e qualidade (Yin, 2003).

**As capacidades sociais e cognitivas de um investigador** – Para se conseguir ser bom na investigação de um estudo de caso deve-se possuir boas competências sociais como: ser bom ouvinte, conseguir colocar questões que permitam extrair os dados pretendidos, ser flexível, adaptar-se facilmente a mudanças e, deve-se sobretudo, ter uma noção clara do problema em estudo.

**Treino e preparação para um caso de estudo específico** – Este tópico denota que cada estudo de caso é diferente e não pode ser guiado por fórmulas. O investigador deve tomar decisões inteligentes sobre que dados recolher.

**Desenvolvimento de um protocolo para a investigação** – A criação deste documento permite fornecer um protocolo onde se encontram as regras e os procedimentos necessários a serem seguidos na realização das atividades. O desenvolvimento deste documento é uma técnica usada para aumentar a fiabilidade da investigação e guia o investigador no processo de desenvolvimento do estudo de caso.

**Condução de um estudo de caso piloto** – Como fase final para a recolha de dados é necessário a realização de um estudo de caso piloto. O caso piloto ajuda o investigador a definir os planos de recolha de dados, o conteúdo dos dados assim como o tipo de dados a recolher.

## 2.4 Recolher os Dados

Relativamente à **recolha de dados (c)** pode ser realizada em muitas fontes de informação existentes, contudo, existem seis tipos de fontes de dados que são consideradas fontes de dados prioritárias e onde a maior parte dos dados devem ser recolhidos (Yin, 2003):

**Documentação** – A documentação é uma das maiores fontes de informação e é usada em praticamente qualquer estudo de caso. Pode ser encontrada em diversas formas e deve ser

inserido em qualquer plano de recolha de dados. Possui como pontos fortes, ser uma fonte de informação estável e ter informação com longos períodos de vida. Mas apresenta como fraquezas, a incerteza dos autores e a informação poder ser deliberadamente bloqueada.

**Documentos arquivados** – Os documentos arquivados podem ser bastante relevantes para uma grande parte dos estudos de casos. Exemplos de dados que podem ser recolhidos são: mapas, gráficos, listas de nomes, dados de questionários, como é exemplo, os censos e registos pessoais ou organizacionais. Para além dos pontos referidos anteriormente, trata-se, ainda, de uma fonte precisa e abundante de informação, contudo, possui com desvantagem as possíveis restrições de acesso à mesma.

**Entrevistas** – Outra das maiores fontes de informação em tempo real são as entrevistas. Esta afirmação pode parecer estranha devido à existência de questionários, mas com entrevistas podemos recolher dados de opinião aberta e em tempo real. Possui como ponto forte, o contacto direto com o caso em estudo, por outro lado, as suas fraquezas podem ser a má recolha de dados devido à má formulação de questões e receber respostas iguais ao que o entrevistador quer ouvir e não a opinião do entrevistado.

**Observação direta** – Fazer uma visita a uma área de estudo específica pode levar a recolha de informação relevante que de outra forma não seria possível recolher. Possui como ponto forte, a cobertura de eventos em tempo real, por outro lado, possui como ponto fraco, a quantidade de tempo consumido para a realização dessa atividade.

**Observações participativas** – Diferente da observação direta em que o investigador toma uma atitude passiva de observação externa face aos eventos, este modo de observação permite ao investigador assumir um papel dentro da situação de estudo de caso, permitindo assim, a participação nos eventos em estudo Possui como ponto forte, para além dos pontos da observação direta, a recolha precisa de comportamentos e motivos interpessoais, por outro lado possui como ponto fraco a manipulação de eventos.

**Artefactos físicos** – A fonte final de informação é a recolha de dados a partir de artefactos físicos ou culturais. Possui como ponto forte uma visão mais acertada, operações técnicas e características culturais, por outro lado, as suas fraquezas são a disponibilidade dos dados assim como a dificuldade na seleção.

## 2.5 Analisar os Dados Recolhidos

A **análise de dados (d)** é o ponto menos desenvolvido e mais difícil de realizar no que toca a estratégia de estudo de caso. O autor (Yin, 2003) identificou duas estratégias generalistas de análise de dados: “Apoiar-se em proposições teóricas” e “Desenvolver um caso descritivo”.

“Apoiar-se em proposições teóricas” é a estratégia preferencial uma vez que segue a proposição teórica que conduz ao caso de estudo. O objetivo e estrutura inicial do estudo de caso é baseado nessa proposição, o que permite retornar a resposta a um conjunto de perguntas de investigação, revisões de literatura e novas descobertas.

Esta proposição permite focar a atenção na informação mais relevante e ignorar outra informação que foi recolhida. Outra vantagem desta proposição é a ajuda que ela fornece na organização de todo o estudo de caso, assim como, também ajuda a definir outras explicações alternativas que necessitem de ser examinadas.

“Desenvolver um caso descritivo é a estratégia que opta pela definição de uma *framework* descritiva para organização do estudo de caso. Esta estratégia serve como uma alternativa, quando a proposição teórica não se encontra presente.

## 3 Revisão de Literatura

Revisão de Literatura: Este capítulo apresenta a revisão de literatura relacionada com o *Rational Unified Process*, fazendo inicialmente a identificação e descrição da sua arquitetura processual e, posteriormente, realizando uma análise cuidadosa ao elenco processual original (Rational, 1998). Por último podemos ainda encontrar neste capítulo a identificação e descrição do elenco processual do RUP reduzido definido e identificado por Borges (Borges, 2007) na realização da sua dissertação.

### 3.1 Contextualização do Rational Unified Process

O *Rational Unified Process* (RUP) é uma *framework* de processos de desenvolvimento de Software que fornece uma abordagem metodológica, no que diz respeito a tarefas e responsabilidades no desenvolvimento de Software organizacional. O objetivo principal é a criação de Software de alta qualidade que corresponda às expectativas do cliente, dentro do prazo e com o orçamento o mais reduzido possível (Kroll & Kruchten, 2003). No RUP podemos encontrar as melhores práticas de desenvolvimento de Software que foram recolhidas ao longo dos tempos, de forma a poder abranger um grande leque de projetos e organizações (Kruchten, 2003).

Para desenvolvimento de um projeto nas áreas das Tis, existe a necessidade de seguir outras metodologias que complementem a utilização do RUP. No caso que vai ser abordado nesta dissertação, de uma equipa académica constituída por quinze elementos, irão ser utilizados o *Capability Maturity Model Integration* (CMMI) para a qualidade de processos, o *Rational Unified Process* (RUP) para o desenvolvimento dos processos e o *Project Management Body of Knowledge* (PMBOK) para a gestão do projeto. Cada um destes possui responsabilidades diferentes na realização do projeto mas, tanto o CMMI como o PMBOK, são utilizados para dar suporte na utilização do RUP.

O PMBOK fornece os objetivos fundamentais para a gestão de projetos, independentemente do tipo de projeto que se está a realizar. O conceito básico para a aplicação deste processo envolve a utilização de cinco fases: *Initiating, Planning, Executing, Monitoring and Controlling, Closing*, (Project Management Institute, 2013) (Bourque & Fairley, 2014).

O CMMI é a integração de muitos outros modelos de *Capability Maturity Models* (CMM), (CMMI Product Team, 2010), que resulta numa melhoria dos modelos e práticas dos outros CMM's, que um só CMM não conseguia atingir. O modelo de qualidade usado para construir sistemas de processos otimizados é usado para tornar os processos já existentes em processos com maturidade superior e grau de incerteza menor.

Existem cinco níveis de maturidade para os processos no CMMI em que o inicial (um) é o nível onde não existe controlo nos processos e o nível cinco é o nível desejado em que a característica principal é o foco na melhoria contínua do mesmo.

Na secção abaixo estão definidas e identificadas as características da *framework* de desenvolvimento de Software RUP, onde podemos encontrar as dimensões do processo, sendo também caracterizado e descrito o elenco processual definido pela *Rational*.

### 3.2 Caracterização e Definição do *Rational Unified Process*

A arquitetura processual do RUP é decomposta em duas dimensões: as Fases do ciclo de vida e as disciplinas do processo como pode ser visualizado na Figura 1. O eixo horizontal representa as Fases do ciclo de vida do processo onde se pode verificar a sua natureza dinâmica. Este eixo é dividido em quatro fases distintas que podem possuir várias iterações cada, com a finalidade de se tentar aproximar ao máximo dos requisitos definidos. Estas fases encontram-se representadas da seguinte maneira (Borges, 2007), (Rational, 1998):

- ***Inception*** - Fase utilizada para explicar os objetivos do projeto e o modelo de negócio correspondente. Esta fase termina com o *milestone* denominado LCO (*Lifecycle Objective Milestone*).
- ***Elaboration*** – Fase utilizada para desenvolvimento e validação da arquitetura do sistema, tentando recolher os requisitos essenciais do projeto, de forma a conseguir calcular os custos associados e planear o seu desenrolar. Esta fase termina com o *milestone* denominado LCA (*Lifecycle Architecture Milestone*).
- ***Construction*** - Fase utilizada para implementar o sistema de Software de acordo com o que foi definido na *Elaboration* e permitindo assim o crescimento da mesma até estar pronta para ser distribuída pelos utilizadores. Esta fase termina com o *milestone* denominado IOC (*Initial Operational Capability Milestone*).

- **Transition** – Fase utilizada para transitar o sistema desenvolvido da organização para o cliente final, incluindo todo o processo de implementação, *testing*, formação de utilizadores e preparação das atividades de manutenção que poderão ser necessárias. Esta fase termina o ciclo de vida com a *milestone* denominado de PR (*Product Release Milestone*).

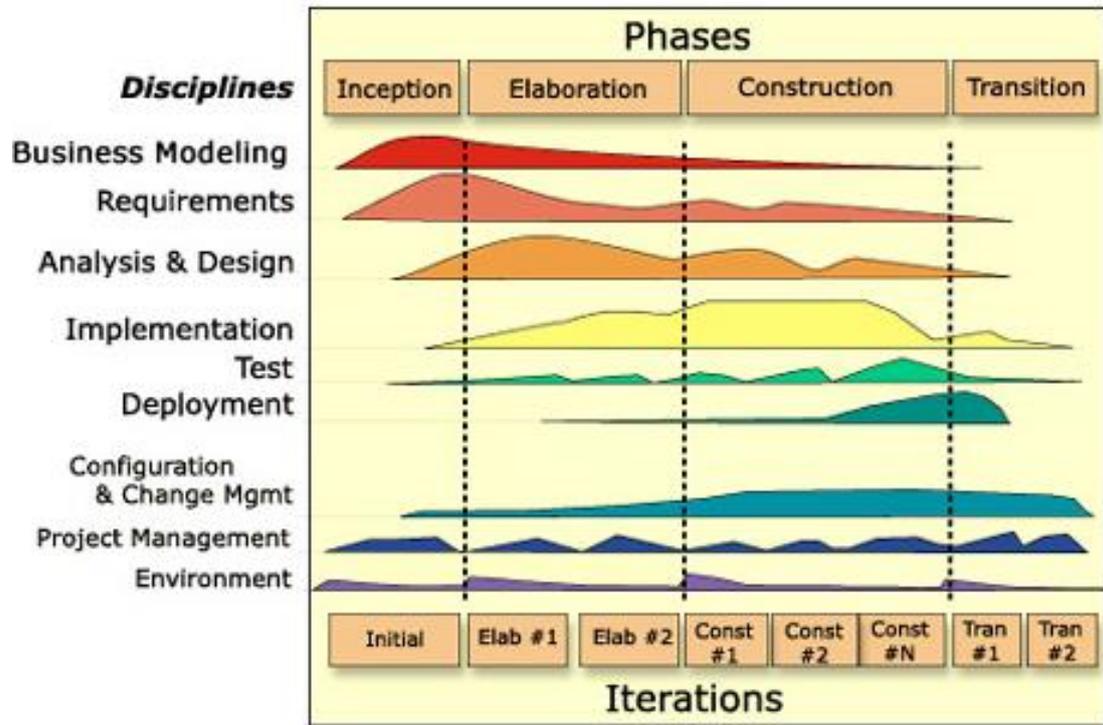


Figura 1 – *Rational Unified Process* (Kruchten, 2003)

O eixo vertical representa as disciplinas ou *workflows* caracterizados pelas atividades que são de maior relevância no processo de desenvolvimento de Software e possui uma natureza estática (Kruchten, 2003). Estas disciplinas encontram-se divididas da seguinte forma (Borges, 2007):

- *Business Modelling* – Atividade que tem como função o levantamento dos requisitos relevantes para o desenvolvimento do sistema, assim como, a descrição dos processos e da estrutura do negócio.
- *Requirements* – Atividade que tem como função a organização dos requisitos (esclarecimento, estruturação e documentação dos mesmos).
- *Analysis and Design* - Atividade que tem como função o desenvolvimento da arquitetura, assim como, do Software requerido.
- *Implementation* - Atividade que tem como função o desenvolvimento e a verificação do sistema pedido através de *debugging* e de *testing*.

- *Test* – Atividade que tem como função o teste da integração do sistema e a aceitação por outros sistemas.
- *Deployment* - Atividade que tem como função a criação e produção de manuais de utilização para o utilizador, a criação dos pacotes de instalação, entre outras tarefas semelhantes.
- *Configuration and Change Management* - Atividade que tem como função gerir as versões, assim como, os pedidos de manutenção ou de alterações.
- *Project Management* - Atividade que tem como função o planeamento, a gestão e a supervisão do projeto.
- *Environment* - Atividade que tem como função a acomodação do processo às necessidades da organização e a inclusão de ferramentas de suporte ao desenvolvimento.

Relativamente às atividades, cada uma delas corresponde a um foco de trabalho a ser realizado por uma ou várias unidades e resultam na produção de artefactos que, no seu contexto, são importantes para a realização do projeto (Borges, 2007). São estes artefactos que funcionam como peças de um puzzle para completar o projeto, ou seja, quantos mais artefactos se obtiver maior será a quantidade de informação útil e menor será a incerteza de se completar o projeto. Os artefactos são produzidos pelo papel mais especializado para a realização dos mesmos. Apesar do trabalho em equipa ser uma mais-valia e essencial na maior parte das atividades, no que toca a lidar com a complexidade do desenvolvimento de Software, não deixa de ser verdade que algumas atividades devem ser deixadas ao cuidado do colaborador mais especializado nessa função. Com isso o RUP, no seu modelo, propõe a criação de equipas multifuncionais que permitam ter especialistas em diversas áreas.

Na secção seguinte prosseguir-se-á com a descrição e caracterização do elenco processual do RUP, assim como, uma breve apresentação da forma como outros autores definiram o leque de responsabilidades do RUP, até chegarmos aos papéis atualmente definidos, no sentido de conseguirmos perceber melhor quais as responsabilidades de cada cargo e as suas transformações.

### 3.3 Definição e Caracterização do Elenco Processual do RUP

Agora que o processo do RUP se encontra genericamente definido, é chegada a altura de proceder a uma análise minuciosa do elenco processual do RUP, com vista a melhor entender a forma como foi desenvolvida a caracterização base do elenco processual reduzido, assim como, entender globalmente todas as responsabilidades e funcionalidades listadas no elenco processual do RUP. Segundo Borges (Borges, 2007) e Jacobson (Jacobson et al., 1999) o ponto de partida para uma melhor definição do elenco processual é a análise da génese do RUP que se pode expor a partir do *Unified Process*. Após a análise da obra, observa-se que Jacobson optou por caracterizar cada responsabilidade ao longo dos fluxos de trabalho, dispersando a respetiva definição ao longo de vários capítulos, o que torna mais complexa a perceção da mesma e mais confusa para os leitores. Para além disso, na obra de Jacobson (Jacobson et al., 1999), apenas se consegue identificar a participação de dez papéis distintos, sendo estas responsabilidades muito inferiores em número aos trinta e um papéis que se encontram no elenco processual do RUP.

Seguidamente apresenta-se a identificação e descrição dos papéis do elenco original pela ordem que são referenciados em Jacobson (Jacobson et al., 1999), com o objetivo de elucidar as responsabilidades que cada um desses cargos possui (Tabela 2).

Responsabilidade	Descrição
<i>System Analyst</i>	Responsável por todo o conjunto de requisitos que são modelados como casos de uso, incluindo todos os requisitos específicos a cada caso de uso. Para além disso, é responsável por delimitar o sistema, encontrar os autores, casos de uso e garantir que o modelo de casos de uso está completo e consistente. Apesar de ser responsável pelo modelo de caso de uso e pelos colaboradores intervenientes, não é responsável pelos casos de uso de uma forma individual, sendo essa tarefa atribuída ao <i>Use-Case Specifier</i> . Por fim é também o indivíduo que lidera o levantamento de requisitos.
<i>Use-Case Specifier</i>	Normalmente o levantamento de requisitos não pode ser feito apenas por um só elemento de equipa. Estes intervenientes auxiliam o <i>System Analyst</i> com essa tarefa, ficando assim responsáveis pela descrição, o mais detalhada possível, de um ou vários casos de uso. Para isso, estes intervenientes necessitam de contactar com os utilizadores finais correspondentes a estes casos de uso.
<i>User-Interface Designer</i>	Interveniente responsável por dar forma visual ao interface do utilizador. Apesar de não realizar a implementação da mesma, uma vez que é efetuada pelos <i>developers</i> durante a fase de conceção e implementação é responsável pela criação de protótipos para os casos de uso.
<i>Architect</i>	Durante a fase de análise, conceção e implementação assegurará a integridade dos respetivos modelos, de forma a garantir que os modelos se encontram corretos, perceptíveis e consistentes. Em adição à responsabilidade anterior, também é responsável pela arquitetura dos modelos de análise, conceção e implementação. Por fim o <i>Architect</i> é responsável ainda pelo mapeamento dos componentes executáveis para os nodos. Convém advertir que o <i>Architect</i> não é responsável pelo desenvolvimento contínuo e pela manutenção dos artefactos que se encontram nos modelos de análise e de conceção, sendo

	estes da responsabilidade do <i>Use-Case Engineer</i> , nem pelo desenvolvimento contínuo e manutenção dos artefactos que se encontram dentro dos modelos de implementação, que são da responsabilidade do <i>Component Engineer</i> .
<i>Use Case Engineer</i>	Cargo responsável pela integridade e conceção de uma ou mais concretizações de casos de uso, certificando-se assim que cumprem os requisitos na sua integridade. Existe a necessidade de garantir que todas as descrições textuais e diagramas que descrevem a realização do caso de uso são legíveis. É de observar que o <i>Use-Case Engineer</i> não é responsável pela conceção das classes, subsistemas, interfaces e relações de análise empregues na concretização do caso de uso, sendo essas funções da responsabilidade do <i>Component Engineer</i> . É de notar que o <i>Use-Case Engineer</i> é ainda responsável pela conceção das concretizações do caso de uso uma vez que, aquando da realização da análise e conceção existe uma transição natural.
<i>Component Engineer</i>	Responsável pela definição das responsabilidades, relações, requisitos especiais e atributos de uma ou mais classes de análise e das operações, métodos, atributos, relações e requisitos de uma ou mais classes de conceção, assegurando assim que cada uma delas obedece aos requisitos que lhe foram impostos pela realização do caso de uso em que participa. Tem como competência manter a integridade de um ou mais pacotes de análise, garantindo que os seus conteúdos estão corretos e que as suas dependências em relação a outros pacotes de análise são corretas e mínimas. Por fim, compete-lhe ainda garantir a integridade de um ou mais subsistemas de implementação. A maior parte das alterações a estes subsistemas são geridas durante a fase de conceção uma vez que, entre estes subsistemas e os subsistemas que são concebidos, existe uma correlação de um-para-um.
<i>System Integrator</i>	A integração do sistema é uma tarefa que está para além do âmbito individual de cada <i>Component Engineer</i> , sendo assim, o <i>System Integrator</i> responsável pela mesma. Ser responsável por esta tarefa engloba o planeamento da sequência de versões que são necessárias entre cada iteração, assim como, a integração de cada versão logo que as suas componentes estejam implementadas.
<i>Test Designer</i>	Cargo responsável pelo <i>design</i> e integridade do modelo de testes, assegurando assim que este cumpre os objetivos. Em adição, são responsáveis pela seleção e descrição dos casos de teste, assim como, os respetivos procedimentos de teste necessários, sendo, também, responsáveis por analisar e avaliar os testes de integração e sistema, após terem sido executados.
<i>Integration Tester</i>	A realização dos testes de integração para cada versão do fluxo de implementação é da responsabilidade deste cargo. Estes testes são realizados de forma a verificar se as componentes integradas em cada versão funcionam corretamente em conjunto. É de notar que em alguns projetos o papel de <i>System Integrator</i> e de <i>Integration Tester</i> são acumulados, de forma a minimizar a replicação de conhecimento comum entre colaboradores.
<i>System Tester</i>	Este cargo encontra-se encarregue da realização dos testes de sistema necessários para a criação da versão resultante de uma iteração completa. Estes testes são desenvolvidos e executados com o intuito de verificar as interações entre atores e o próprio sistema.

Tabela 2 - Descrição e Identificação dos cargos e responsabilidades elenco original

Depois da análise e descrição do elenco original, concluiu-se que esta obra não constitui uma boa base de partida para identificação do elenco processual nem para o estabelecimento de um referencial de ação, dada a “miríade de tarefas que usualmente recaem sobre cada participante num processo de desenvolvimento de Software” (Borges, 2007). Ou seja, apesar de apresentar de uma forma clara cada cargo e responsabilidade, não mostra a forma como cada um deve interagir no sistema.

Seguindo os mesmos passos que o autor Borges seguiu em (Borges, 2007), e de forma a tentar encontrar uma boa base de partida para a definição do elenco processual, optou-se pelo *Tutorial* HTML do RUP (Rational, 2001), sendo este datado de 2001, ou seja, sujeita a consideráveis desatualizações provenientes das revisões constantes a que a metodologia é sujeita periodicamente. Ainda assim, a análise de tal documento mostrou-se mais proveitosa do que a da obra de Jacobson (Jacobson et al., 1999), uma vez que, permitiu o acesso a informação sobre o elenco processual de uma forma mais direta.

Assim o modelo do RUP decidiu dividir o elenco processual em cinco grupos distintos e que trabalham em áreas do projeto diferentes, para que desta forma seja mais fácil a identificação da área a que cada papel corresponde, passando este elenco a ser denominado “elenco v2001”. Os grupos referidos possuem na sua constituição uma totalidade de trinta e um papéis que desempenham, dentro do próprio grupo, funções distintas. Os cinco grupos funcionais do elenco v2001 são identificados e definidos na Tabela 3.

Grupo	Descrição
<i>Analyst</i>	Associa-se a papéis que correspondem mais ao processo de levantamento de requisitos.
<i>Developer</i>	Enquadra-se nos papéis de criação e implementação do Software do sistema.
<i>Tester</i>	Abrange os papéis de <i>testing</i> e <i>debugging</i> .
<i>Manager</i>	Engloba os papéis de planeamento e gestão de processos de engenharia de Software.
<i>Other Roles</i>	Engloba todos os outros papéis necessários no suporte ao processo de desenvolvimento que não tenham sido referidos em nenhum dos outros grupos.

Tabela 3 - Grupos de divisão do elenco processual v2001

Na Tabela 4 seguinte, encontram-se identificados e caracterizados todos os cargos do elenco v2001 (Rational, 2001) dentro do seu grupo específico.

<i>Manager</i>	<i>Analyst</i>	<i>Developer</i>	<i>Tester</i>	<i>Others</i>
<i>Change Control Manager</i>	<i>Business Designer</i>	<i>Architecture Reviewer</i>	<i>Test Designer</i>	<i>Any Role</i>
<i>Configuration Manager</i>	<i>Business Model Reviewer</i>	<i>Capsule Designer</i>	<i>Tester</i>	<i>Course Developer</i>
<i>Deployment Manager</i>	<i>Business Process Analyst</i>	<i>Code Reviewer</i>		<i>Graphic Artist</i>
<i>Process Engineer</i>	<i>Requirements Reviewer</i>	<i>Database Designer</i>		<i>Stakeholder</i>
<i>Project Manager</i>	<i>Requirements Specifier</i>	<i>Designer</i>		<i>System Administrator</i>
<i>Project Reviewer</i>	<i>System Analyst</i>	<i>Design Reviewer</i>		<i>Technical Writer</i>

	<i>User Interface Designer</i>	<i>Implementer</i>		<i>Tool Specialist</i>
		<i>Integrator</i>		
		<i>Software Architect</i>		

Tabela 4 – Identificação do elenco v2001 definido no RUP e respetivos distribuição por grupos

No **Grupo 1**, “*Manager*”, encontram-se todos os papéis em que o seu foco e responsabilidade se inclinam para o planeamento, gestão e supervisão do processo e dos outros papéis. Os papéis que representam este grupo podem ser definidos como:

- *Change Control Manager* – Este papel tem como funcionalidade a supervisão do processo de controlo de mudanças. Normalmente é constituído por um grupo de intervenientes que engloba os clientes, *developers* e os utilizadores. No desenvolvimento de pequena escala ou dimensão, este papel pode ser atribuído ao gestor de projeto (*Project Manager*).
- *Configuration Manager* – Este papel tem como responsabilidade fornecer a infraestrutura de gestão de configurações para a equipa de desenvolvimento, que tem como funcionalidade o suporte da sua atividade. Permitindo, assim, a criação de um espaço de trabalho para uso dos *developers* e integradores para realizarem as ações de implementação e teste e facultando os artefactos que produzem. Este cargo fica também responsável por desenvolver o plano de gestão de configurações.
- *Deployment Manager* – Este papel tem como responsabilidade o planeamento da transição do produto para o utilizador, sendo também responsável pela documentação inerente aos processos.
- *Process Engineer* – Papel ao qual é atribuído a funcionalidade de desenvolvimento de processos de Software, onde se inclui toda a documentação pré desenvolvimento, assim como, a melhoria dos processos depois do desenvolvimento.
- *Project Manager* – Papel responsável pelo planeamento e atribuição de práticas, no sentido de fiabilizar e aumentar a qualidade dos artefactos produzidos pela equipa, sendo, também, responsável pela alocação de recursos, definição de prioridades, interações com clientes, assim como, por incentivar a equipa no cumprimento dos objetivos definidos.

- *Project Reviewer* – Responsabilidade de analisar e validar os artefactos do planeamento de projeto referentes a *milestones* relevantes ao ciclo de vida do mesmo. Estas validações são de extrema relevância, uma vez que, podem fazer com que um projeto seja cancelado, caso o planeamento tenha sido inadequado.

No **Grupo 2**, “*Analyst*”, encontram-se todos os papéis em que o seu foco e responsabilidades englobam o planeamento do âmbito, requisitos e modelação do negócio. Os papéis que representam este grupo podem ser definidos como:

- *Business Designer* – Papel que se encarrega do detalhe da organização, distinguindo os fluxos de casos de uso de negócio em relação a entidades e *workers*, sendo também responsável por definir responsabilidades, operações, atributos e a interação entre as mesmas.
- *Business-Model Analyst* – Possui a responsabilidade de revisão dos objetivos de negócio, assim como, dos modelos de casos de uso.
- *Business-Process Analyst* – Chefia e organiza a modelação dos casos de uso de negócio, delimitando a zona da organização necessária a modelar.
- *Requirements Reviewer* – Responsável pelo planeamento e revisão do modelo de casos de uso.
- *Requirements Specifier* – Papel responsável pelo detalhe de funcionalidades do sistema, especificando, mais aprofundadamente, requisitos de um ou vários casos de uso. “Também pode ser responsável pela manutenção da integridade de um determinado pacote de casos de uso” (Borges, Configuração do RUP com Vista à Simplificação dos Elencos Processuais em PMEs de Desenvolvimento de Software, 2007).
- *System Analyst* – Papel responsável por relacionar os requisitos com a modelação dos casos de uso, delineando os sistemas e objetivando as funcionalidades dos mesmos.
- *User-Interface Designer* – Papel responsável pela modelação, conceção e coordenação do interface de utilizador. O levantamento de requisitos de interface, a construção do protótipo e revisão de interface são também tarefas realizadas por este papel.

No **Grupo 3**, “*Developer*”, encontram-se todos os papéis em que o seu foco e responsabilidades se inclinam para o desenvolvimento e implementação da solução planeada. Os papéis que representam este grupo podem ser definidos como:

- *Architecture Reviewer* – Papel responsável pelo planeamento e condução, ao longo do ciclo do projeto, da arquitetura de *Software* intrínseca ao desenvolvimento do projeto.
- *Code Reviewer* – Responsável por assegurar a qualidade do código fonte, bem como, pelo planeamento e revisão do mesmo.
- *Database Designer* – Papel responsável pela redefinição de todos os parâmetros específicos correspondentes à base de dados e que “estejam relacionados com armazenamento, recuperação e remoção de objetos” (Borges, 2007).
- *Designer* – O *Designer* deve converter a arquitetura, concebida pelo arquiteto, numa solução coerente de componentes e modelos para poder ser implementada e detalhada, criando, ainda, operações e relações entre eles. Pode ser responsável por um ou mais pacotes de subsistemas.
- *Capsule Designer* – Possui as mesmas responsabilidades do *Designer* mas mais focada nos requisitos das componentes de *performance*.
- *Design Reviewer* - Responsável pelo planeamento e revisão do modelo de conceção, assim como, pela revisão de artefactos desenvolvidos por outros elementos.
- *Implementer* – Este papel tem as responsabilidades de programação de todas as componentes de um sistema, de acordo com as regras definidas no plano, para futuramente se passar à sua integração noutros subsistemas. Adicionalmente é ainda o responsável pela primeira linha de *debugging* do sistema.
- *Integrator* – Papel responsável pelo planeamento dos processos de integração provenientes dos *Implementers* e, conseqüentemente, responsável pela integração dos subsistemas em sistemas mais complexos denominados de *builds*.
- *Software Architect* – Líder e coordenador dos artefactos técnicos do projeto. Estabelece a estrutura geral da visão arquitetural, logo, em comparação com os restantes papéis, o arquiteto de *Software* necessita de ter uma visão do artefacto mais global ou abrangente do que profunda.

No **Grupo 4**, “*Tester*”, encontram-se todos os papéis em que o seu foco e responsabilidades se inclinam para o *teste* de todo o sistema desenvolvido pela equipa. Os papéis que representam este grupo podem ser definidos como:

- *Test Designer* – Responsável pelo planeamento, conceção, implementação e avaliação dos testes, onde se inclui a criação de plano e modelos de teste, implementação do procedimento, resultados de eficácia e abrangimento, assim como, a geração de sumários de avaliação de esforço dos testes.
- *Tester* – Responsável pela preparação e execução de um ou vários testes planeados pelo *Test Designer*. Assim sendo, faz uma avaliação à execução com o objetivo de reparar possíveis falhas que possam ocorrer durante os testes.

No **Grupo 5**, “*Others*”, encontram-se todos os papéis cujo foco e responsabilidades se inclinam para o suporte dos outros papéis organizacionais. Os papéis que representam este grupo podem ser definidos como:

- *Any Role* – Um papel que, se possuir as permissões necessárias, pode fazer *check-in* e *check-out* dos artefactos do projeto para realizar manutenções.
- *Course Developer* – Papel responsável pela formação e pelo desenvolvimento de material necessário para a mesma, de forma a instruir os utilizadores acerca do produto.
- *Graphic Artist* – Papel responsável pelo desenvolvimento das componentes de interface gráfico necessários ao produto.
- *Stakeholder* – Corresponde a qualquer elemento que seja afetado pelo resultado final do projeto. Este papel cobre, na sua maior parte, clientes, utilizadores finais, acionistas, etc.
- *System Administrator* – Papel responsável por manter o ambiente de desenvolvimento organizacional “saudável”, ou seja, é responsável por todas as componentes, quer *Hardware* quer *Software*, que estejam envolvidas no projeto, assim como, pela realização de cópias de segurança ao sistema.
- *Technical Writer* – Papel responsável pela produção de manuais ou instruções de suporte para utilizadores.

- *Tool Specialist* – Papel responsável pelas ferramentas de suporte ao projeto, incluindo a instalação, configuração e aquisição das mesmas.

Encontra-se, assim, identificado e definido o elenco v2001 utilizado pelo RUP. Verifica-se contudo uma incoerência no elenco v2001, devido à falta de determinados papéis existentes no elenco original (Borges, 2007). Pode-se verificar que *Component Engineer*, *Integration Tester*, *Use-Case Engineer* e *Use-Case Specifier* não se encontram presentes no elenco v2001, assim como, *Architect*, *System Integrator* e *System Tester* definidos no elenco original possuem uma designação diferente, sendo esta *Software Architect*, *Integrator* e *Tester* respetivamente.

De forma a complementar e estabilizar o elenco v2001, seguindo a sugestão de Borges (Borges, 2007), foi utilizado o elenco proposto por Kruchten (Kruchten, 2004), passando este a ser designado por elenco v2004.

Existem algumas diferenças entre o elenco v2001 e o elenco v2004, mas no que respeita às nomenclaturas dos papéis são bastante semelhantes. Uma diferença é que no elenco v2004 os papéis são vistos como *Workers*. Outra diferença notória entre os dois elencos é que o elenco v2004 possui apenas trinta papéis na sua constituição, sendo que o elenco v2001 possui no seu estado normalizado trinta e um.

Com a análise do Apêndice A da obra de Kruchten (Kruchten, 2004) e seguindo as sugestões de Borges (Borges, 2007), podem ser retiradas diversas conclusões relativamente a comparação das diversas versões de elencos, das quais se destacam as seguintes:

- *Requirements Reviewer* e *Project Reviewer* definidos no elenco v2001 não se encontram referenciados em elenco v2004.
- *Management Reviewer*, *Test Manager*, *Test Analyst*, *Reviewer* e *Reviewer Coordinator* são papéis ausentes no elenco v2001 e pertencentes ao elenco v2004.
- Os papéis *Architecture Reviewer*, *Business Reviewer*, *Code Reviewer* e *Design Reviewer* não se encontram identificados no apêndice A de elenco v2004 mas, encontram-se identificados no elenco v2001.

- *Component Engineer, Integration Tester, Use-Case Engineer* e o *Use-Case Specifier* referenciados em elenco original não se encontram presentes no elenco v2004.

Uma vez que não foi possível chegar a um consenso com a informação recolhida sobre esta matéria, houve uma necessidade de compilar os dados que foram apresentados ao longo desta secção, de forma a conseguir construir uma composição admissível do elenco processual para que sirva de base para a definição do elenco processual do modelo reduzido definido por Borges (Borges, 2007). Para finalizar esta secção e, no sentido de criar um elenco processual que seja o mais relevante para aplicação do RUP, será realizada uma correlação de toda a informação do elenco processual recolhida e apresentada nesta secção. Na tabela encontra-se identificados todos os papéis divididos pelos grupos correspondentes do elenco v2007, devido à extensão da caracterização do elenco v2007 e para não tornar demasiado extenso este capítulo, a mesma encontra-se no Anexo A.

<i>Manager</i>	<i>Analyst</i>	<i>Developer</i>	<i>Tester</i>	<i>Production and Support</i>	<i>Additional</i>
<i>Change Control Manager</i>	<i>Business-Model Reviewer</i>	<i>Architecture Reviewer</i>	<i>Integration Tester</i>	<i>Course Developer</i>	<i>Any Role</i>
<i>Configuration Manager</i>	<i>Business-Process Analyst</i>	<i>Capsule Designer</i>	<i>System Tester</i>	<i>Graphic Artist</i>	<i>Stakeholder</i>
<i>Deployment Manager</i>	<i>Business Designer</i>	<i>Code Reviewer</i>	<i>Test Analyst</i>	<i>Review Coordinator</i>	
<i>Management Reviewer</i>	<i>Requirements Reviewer</i>	<i>Component Engineer</i>	<i>Test Designer</i>	<i>System Administrator</i>	
<i>Process Engineer</i>	<i>Requirements Specifier</i>	<i>Database Designer</i>		<i>Technical Writer</i>	
<i>Project Manager</i>	<i>System Analyst</i>	<i>Designer</i>		<i>Tool Specialist</i>	
<i>Project Reviewer</i>	<i>Use-Case Specifier</i>	<i>Design Reviewer</i>			
<i>Test Manager</i>	<i>User-Interface Designer</i>	<i>Implementer</i>			

		<i>System Integrator</i>			
		<i>Software Architect</i>			
		<i>Use-Case Engineer</i>			

Tabela 5 - Identificação do elenco v2007 definido por Borges e respetivos distribuição por grupos

### 3.4 Identificação do elenco processual do RUP no contexto de PME

Após a análise descritiva e a caracterização do elenco processual do RUP, apresentada nas secções anteriores, é chegado o momento de aplicar essa informação ao contexto das PMEs ou equipas com dimensões relativamente pequenas.

Nas organizações anteriormente definidas, existe a necessidade de ter em conta certos fatores que as diferenciam das grandes organizações. Maioritariamente, nas organizações de pequenas dimensões, contrariamente às organizações de grandes dimensões, são apresentados evidências de uma estrutura organizacional descentralizada, colaboradores extremamente flexíveis e versáteis no desempenho das funções, bem como, uma informalidade nos processos. Com isso espera-se uma dificuldade na adaptação desse tipo de colaboradores a processos normalizados e com elevado nível de detalhe. Como adição aos fatores referidos acima, o RUP, sendo *architecture centric*, ainda assume que todos os indivíduos intervenientes no processo, possuem ou tem acesso a ferramentas de modelação *Unified Modeling Language* (UML), assim como também possuem a formação necessária para a utilizar de forma correta.

Assim, e apoiando o comentário de Borges (Borges, 2007), o estado atual em que o RUP se encontra não é plausível de aplicação no contexto das PMEs.

É de realçar que a desadequação da aplicação do mesmo, não é resultado do conceito ou da abrangência do RUP, já que este tenta atingir a maior extensão processual possível, indo ao máximo de pormenor no desenvolvimento dos seus artefactos. Este facto não é algo que deve ser apontado como negativo, uma vez que, os autores do RUP desenvolveram-no com o propósito de não ser de aplicação absoluta, mas sim, para ser uma metodologia suficientemente flexível para ser implementada parcialmente, de acordo com as necessidades e condicionalismos da organização em causa.

Neste sentido pretende-se que as organizações de dimensões reduzidas na área de desenvolvimento de Software, olhem para o RUP como um complemento no seu processo de desenvolvimento, tentado assim, implementar o mesmo de forma faseada, procurando encontrar e filtrar possíveis conceitos e processos de fácil contextualização à própria organização, fazendo com que os processos adicionais do modelo sejam adiados para adaptação futura.

Relativamente à dificuldade de acesso às ferramentas sugeridas pelo *Rational* para suportar o processo de aplicação do RUP podem ser ultrapassadas através de outro conjunto de ferramentas, sendo que as mesmas podem não fornecer uma integração tão transparente e fácil para o utilizador. Apesar de não fazerem parte das aplicações sugeridas pelo *Rational*, conseguem compensar em grande parte as carências que os elementos da organização possam ter. A Tabela 6 demonstra o tipo de ferramentas a utilizar numa determinada fase do processo, a que tipo de processo corresponde e uma descrição de possíveis problemas e respetivas soluções.

Processo	Ferramenta	Descrição
Levantamento de Requisitos	Processador de texto + ferramenta de modelação UML	A ferramenta sugerida pelo <i>Rational</i> é a <i>Rational RequisitePro</i> e pode ser substituída pela combinação de um processador de texto, como por exemplo o <i>Microsoft Word</i> , para fazer a descrição textual dos requisitos identificados e de uma aplicação de modelação UML.
Planeamento	Microsoft Project, <i>Mr.Project</i> ou <i>TeamWork</i>	Para o desenvolvimento do planeamento e criação dos diagramas de Gantt, poderá ser utilizada a ferramenta mais comum Microsoft Project ou, em caso de problemas orçamentais, pode-se optar pela ferramenta Mr.Project ou alternativa online <i>TeamWork</i>
Comunicação	Assembla, Facebook, Google Hangouts ou Skype	A comunicação entre elementos de equipa é um facto muito importante para determinar o sucesso de um projeto, mas, nem sempre é possível uma comunicação presencial. A <i>Assembla</i> é uma ferramenta que permite a utilização de fóruns e salas de <i>chat</i> de maneira livre e gratuita, enquanto o Facebook, a Google <i>Hangouts</i> e o Skype apesar de serem ferramentas utilizadas para atividades informais, podem ser utilizados como excelentes meios de comunicação, permitindo ainda a possibilidade de videochamada, em tempo real, para possíveis reuniões.
Gestão de Projeto	<i>JIRA</i>	Relativamente à atribuição de tarefas, planeamento de versões, <i>reporting</i> , controlo do Projeto, etc. já se encontram parcialmente cobertas pela ferramenta <i>Assembla</i> e o <i>site</i> online <i>TeamWork</i> . Pode ser

Processo	Ferramenta	Descrição
		ainda utilizada a ferramenta JIRA para complementar as funcionalidades das ferramentas indicadas acima.
Desenvolvimento	<i>Eclipse</i>	Existem várias ferramentas de suporte aos programadores que permitem o controlo das versões da qual é exemplo o <i>Subversion</i> presente no <i>Assembla</i> , que se pode encontrar na ferramenta Eclipse.
Gestão de Processo	<i>Eclipse Process Framework</i>	Apesar do RUP incluir uma ferramenta que possui como objetivo a adaptação do próprio processo, existem alternativas para a sua utilização, como é exemplo o <i>Eclipse Process Framework</i> , que permite definir qual o processo adotado pela organização, assim como, a criação e fornecimento de documentação semelhante à ferramenta oficial de apoio ao RUP.

Tabela 6 - Ferramentas alternativas aos processos do RUP

Tal como referido anteriormente, pretende-se clarificar e identificar os cargos existentes no RUP, que apesar de relevantes em projetos de maior complexidade possam ser encapsulados ou enviados para papéis secundários, com objetivo de identificar os papéis necessários para o sucesso da adaptação.

Até agora foi referido que o RUP possui aproximadamente quarenta papéis relevantes no processo de desenvolvimento de Software, sendo que, cada um desses possui um conjunto de responsabilidades distintas. Foi também referido que a maior parte das organizações ao nível nacional são consideradas PME e que grande parte das mesmas nem possui colaboradores suficientes para abranger o elenco processual do RUP. Mesmo que um colaborador possa executar mais do que um papel, uma acumulação excessiva ou errada de papéis pode revelar-se demasiado exigente e complexa, levando à diminuição da produtividade, tanto do interveniente, como do projeto em si.

De acordo com o exposto, pretende-se confirmar que, para ser suscetível a aplicação da metodologia num contexto de menor complexidade, o processo de desenvolvimento de Software deve ser suportado por um número muito mais limitado de intervenientes.

Desta forma conclui-se que existe a necessidade de percorrer três etapas, de forma a ser possível criar um processo eficiente de adaptação do conjunto de intervenientes do RUP aos papéis distintos existentes numa determinada organização. Sendo estas etapas as seguintes:

- Reduzir consideravelmente o número de papéis no processo de desenvolvimento de Software, com objetivo de tornar a sua compreensão mais fácil e, conseqüentemente, facilitar a sua aplicação.
- Identificar problemas de cruzamento de responsabilidades na acumulação de diversos papéis por parte dum colaborador.
- O mapeamento do elenco processual definido por Borges na sua obra (Borges, 2007) deve ser visto como uma proposta, devendo este, ser adaptado às necessidades individuais de cada organização.

Nas próximas duas subsecções serão apresentados os modelos que surgiram através da realização da etapa um, descrita acima, denominados de modelo Base (Borges et al., 2011) e modelo Reduzido, correspondentes ao processo de simplificação de menor e maior profundidade, onde se pretende responder a duas realidades distintas no contexto PME.

Dentro das PME, naturalmente, existem as empresas médias e as empresas pequenas. Nas organizações de dimensão média, os intervenientes procuram um processo que ajude a criar uma solução com elevados níveis de qualidade e a lidar com a complexidade existente nos projetos de média/ alta dimensão. Nas organizações de pequena dimensão pretende-se um processo que ajude a definir, de forma apropriada, o âmbito do projeto em que estão a participar, bem como, otimizar, a utilização de recursos com o objetivo de criar eficiência.

### 3.4.1 Identificação do modelo Base

De forma a concretizar os objetivos que Borges (Borges, 2007) tinha propostos, este autor realizou uma exaustiva e detalhada investigação ao elenco processual do RUP, com a intenção de identificar os papéis essenciais e os papéis secundários. Para isso o autor identificou três condições que um papel deveria ter para ser considerado principal (Borges et al., 2012).

Condição	Descrição
1	Possuir responsabilidades específicas que quando não cumpridas põem em causa o sucesso total do projeto.
2	Ser necessária formação ou necessitar de competências muito específicas e distintas de outros papéis.

3	Existir conflitos entre a utilização desse papel ou a acumulação de outro (s) para o qual (is) exista a necessidade de formação ou competências similares.
---	--

Tabela 7 - Condições para ser considerado um papel principal

Após definição das condições, os papéis definidos a seguir passam a ser considerados principais e irão ser incluídos no modelo base. Na Tabela 8 pode-se visualizar a razão de determinados papéis serem considerados fundamentais, assim como, as condições que se aplicam a cada um dos mesmos.

Modelo Base	Condição	Descrição
<i>Project Manager</i>	1/2	De todos os papéis encontrados no elenco, este pode ser considerado o que assume maior relevância e centralidade, uma vez que possui uma visão global de todo o projeto e, devido à sua interação com as entidades internas e externas ao projeto, espera-se do mesmo a criação de condições necessárias para o sucesso. Para o desempenho correto deste tipo de papel é necessário possuir conhecimentos básicos de gestão e, para além de saber a área de negócio do cliente, é aconselhável possuir formação e prática na utilização da metodologia de gestão de projetos. Ainda de salientar que as competências de negociação são uma mais-valia para este papel.
<i>Integrator</i>	1/2	Apesar de nos encontrarmos em contexto de PME, é normal verificar-se a existência de mais que um <i>Implementer</i> ligado a um projeto, sendo que cada um deles cumpre objetivos diferentes de acordo com que lhe foi atribuído. Neste sentido, é essencial a existência de alguém responsável por informar os <i>Implementers</i> do contexto global do projeto, assim como, por identificar tarefas e atribuir responsáveis às mesmas. Esta última condição impõe que o detentor deste cargo tenha formação apropriada, essencialmente na área de gestão de recursos humanos, na área de <i>Software Engineering Management</i> e <i>Software Engineering Process</i> .

<i>Project Reviewer</i>	3	Uma vez que este papel possui como responsabilidade a revisão e verificação de diversos artefactos que foram produzidos por outros papéis, não convém que o mesmo possua acumulação com outros papéis de forma a garantir imparcialidade e independência entre os papéis que produzem os documentos e os que realizam a sua revisão. Salienta-se, ainda, que este papel necessita de conhecimentos aprofundados na área de negócio em que o projeto se encontra envolvido.
<i>System Analyst</i>	1/2	A gestão do âmbito é essencial para o sucesso de qualquer projeto, devido à participação deste interveniente na identificação e documentação, rigorosa dos requisitos identificados, ou apresentados pelo cliente.
<i>Process Engineer</i>	1/2/3	Um projeto que envolva no desenvolvimento dos seus processos, a participação dos papéis identificados no decorrer deste capítulo, assume um nível de elevada complexidade que, caso não seja bem gerido, poderá resultar em consequências negativas para o mesmo. Má distribuição de responsabilidades pelos vários intervenientes, desperdício de recursos resultantes de uma falha na atribuição de responsabilidades aos papéis e áreas não cobertas pelo processo devido a omissões na atribuição de responsabilidades são três consequências negativas de uma má gestão do nível de complexidade dos processos.  Pelo referido acima, considera-se essencial a participação de um interveniente especializado com a gestão dos processos de desenvolvimento, no sentido realizar possíveis melhorias. Para finalizar este papel requer conhecimentos especializados no processo de desenvolvimento em utilização, sendo neste caso o RUP.
<i>Implementer</i>	1	Este papel é essencial por razões óbvias, não sendo possível um projeto ter sucesso sem que exista a sua implementação. Isto é, mesmo que a arquitetura e conceção do projeto sejam bem-sucedidos, falta a realização do passo seguinte.

<p>Software <i>Architect</i></p>	<p>1/2</p>	<p>Excetuando os projetos de pequena complexidade que podem ser realizados por uma ou duas pessoas, é indiscutível nos restantes, a necessidade de um responsável pela definição das bases onde todo o projeto irá encaixar, estimando os riscos existentes e a forma como se podem evitar. Espera-se ainda deste interveniente, uma definição rigorosa do esqueleto do sistema que irá ser criado, definindo funções e limites das componentes base. Para um bom desempenho desta função existe a necessidade de receber formação na área da arquitetura e de conceção de Software.</p>
<p><i>System Administrator</i></p>	<p>1/2</p>	<p>Em qualquer projeto na área de desenvolvimento de Software existe a dependência de infraestruturas. Ou seja, para o sucesso de um projeto deste tipo é necessário a existência de um responsável por garantir a satisfação de todas as necessidades relacionadas com as infraestruturas do resto da equipa. Salienta-se, ainda, que este interveniente deve possuir formação em várias áreas específicas, nomeadamente: administração de sistemas e configuração e otimização de motores de bases de dados.</p>
<p><i>Test Manager</i></p>	<p>1/3</p>	<p>A qualidade é um dos pontos principais num processo de desenvolvimento de Software , ou seja, uma cuidadosa gestão de qualidade ajuda, em muito, na melhoria do resultado final do processo, por melhorar a imagem do produtor de software, reduzir a ocorrência de bugs e, evitar incumprimentos contratuais (quer sejam de prazos ou de âmbito do projeto). Minimizar os recursos utilizados no suporte a clientes é mais um dos benefícios que este papel pode trazer ao projeto, tornando-se, assim, evidente que o sucesso do processo de desenvolvimento depende da qualidade dos produtos resultantes e, por esse mesmo motivo este papel é uma mais-valia como parte dos intervenientes principais para a realização de um projeto. Contudo, não é aconselhável a acumulação das responsabilidades deste papel com qualquer outro uma vez que, da mesma forma que no <i>Project Reviewer</i>, existe a necessidade de garantir imparcialidade entre o papel que</p>

		realiza uma atividade e o papel que testa a sua qualidade, sendo que a acumulação do mesmo com outras responsabilidades pode resultar na omissão de resultados de testes relevantes para avaliar o produto em causa.
<i>User-Interface Designer</i>	2	A intervenção deste papel num projeto depende muito da natureza dos requisitos definidos no mesmo. Não sendo sempre decisivamente necessário, é facto, que para haver um desempenho de excelência neste tipo de papel, verifica-se uma dependência direta com a necessidade de formação.
<i>Course Developer</i>	2	Uma vez que a responsabilidade deste cargo se suporta na capacidade de educar outros elementos na área que está a ser desenvolvida, existe a necessidade de desenvolver competências específicas em áreas não muito comuns nos outros papéis como a pedagogia e a didática.
<i>Database Designer</i>	2	Este papel é considerado essencial, não pela sua importância para o sucesso do projeto, mas antes pelo conhecimento específico necessário para uma boa utilização do mesmo.
<i>System Tester</i>	3	Como já foi referenciado anteriormente o <i>Test Manager</i> possui, entre outras responsabilidades, conceber o plano interno da qualidade. Porém, a operacionalização do mesmo é da autoria e responsabilidade do <i>System Tester</i> que, desenvolve tarefas diversificadas, desde revisão documentais, até aplicação de testes a artefactos do sistema. Com isto, verifica-se que este é um cargo essencial ao processo devido à eficácia dos processos de qualidade serem dependentes do mesmo.

Tabela 8 - Definição do Modelo Base

De acordo com o exposto, a dimensão do elenco processual do RUP pode ser reduzida para treze cargos, de forma a tornar possível uma adoção do mesmo modelo pelas equipas das PME no setor de desenvolvimento de Software.

Apesar da quantidade de papéis ter sido reduzida de trinta e nove para treze, não significa que as responsabilidades dos outros vinte e seis sejam descartáveis ou menos relevante, por outro

lado, os restantes vinte e seis papéis serão mapeados e incorporados dentro dos papéis identificados anteriormente.

Apesar deste mapeamento ter sido realizado por Borges em (Borges, 2007) como um dos objetivos da sua dissertação, para esta dissertação não é de interesse visualizar a sequenciação de tarefas utilizada pelo autor para chegar a este mapeamento, por outro lado é do interesse visualizar a conclusão desse mesmo mapeamento. Com isso na Tabela 9 fica apresentado o mapeamento do modelo Base que foi definido por Borges.

<b>Grupo</b>	<b>Papéis do RUP</b>	<b>Modelo Base</b>
<i>Manager</i>	<b>Project Manager</b>	<i>Project Manager</i>
<i>Analyst</i>	<i>Business Process Analyst</i>	
<i>Manager</i>	<i>Change Control Manager</i>	
<i>Manager</i>	<i>Deployment Manager</i>	
<i>Analyst</i>	<i>Requirements Specifier</i>	
<i>Production and Support</i>	<i>Review Coordinator</i>	
<i>Tester</i>	<i>Test Analyst</i>	
<i>Developer</i>	<b>System Integrator</b>	<i>System Integrator</i>
<i>Developer</i>	<i>Capsule Designer</i>	
<i>Developer</i>	<i>Code Reviewer</i>	
<i>Developer</i>	<i>Designer</i>	
<i>Tester</i>	<i>Integration Tester</i>	
<i>Manager</i>	<b>Project Reviewer</b>	<i>Project Reviewer</i>
<i>Analyst</i>	<i>Business Reviewer</i>	
<i>Manager</i>	<i>Management Reviewer</i>	
<i>Analyst</i>	<i>Requirements Reviewer</i>	
<i>Analyst</i>	<b>System Analyst</b>	<i>System Analyst</i>
<i>Analyst</i>	<i>Business Designer</i>	
<i>Developer</i>	<i>Use Case Engineer</i>	
<i>Analyst</i>	<i>Use Case Specifier</i>	
<i>Manager</i>	<b>Process Engineer</b>	<i>Process Engineer</i>
<i>Developer</i>	<i>Architecture Reviewer</i>	
<i>Production and Support</i>	<i>Tool Specialist</i>	
<i>Developer</i>	<b>Implementer</b>	<i>Implementer</i>
<i>Developer</i>	<i>Component Engineer</i>	
<i>Developer</i>	<b>Software Architect</b>	<i>Software Architect</i>
<i>Developer</i>	<i>Design Reviewer</i>	
<i>Production and Support</i>	<b>System Administrator</b>	<i>System Administrator</i>
<i>Manager</i>	<i>Configuration Manager</i>	

<i>Manager</i>	<b>Test Manager</b>	<i>Test Manager</i>
<i>Tester</i>	<i>Test Designer</i>	
<i>Analyst</i>	<b>User-Interface Designer</b>	<i>User-Interface Designer</i>
<i>Production and Support</i>	<i>Graphic Artist</i>	
<i>Production and Support</i>	<b>Course Developer</b>	<i>Course Developer</i>
<i>Production and Support</i>	<i>Technical Writer</i>	
<i>Developer</i>	<b>Data Base Designer</b>	<i>Data Base Designer</i>
<i>Tester</i>	<b>System Tester</b>	<i>System Tester</i>
<i>Additional</i>	<b>Any Role</b>	-
<i>Additional</i>	<b>Stakeholder</b>	-

Tabela 9 - Mapeamento do Modelo Base

Dá-se então por concluída a apresentação do modelo Base, considerando-se que o elenco processual deste modelo é suficientemente simples para ser aplicado em organizações de média dimensão, sendo que fica apenas em falta um modelo capaz de satisfazer as condições necessárias para organizações de pequenas dimensões, sendo denominado por modelo Reduzido.

### 3.4.2 Identificação do modelo Reduzido

Pelo exposto anteriormente nota-se uma diminuição do esforço de aplicação do modelo nas organizações de média dimensão, contudo, o modelo base pode ainda ser demasiado complexo para aplicação em micro organizações ou organizações de pequenas dimensões. Com isso em mente, considera-se a necessidade de encontrar um modelo que possua um menor número de intervenientes na sua composição. Para a realização deste processo não é suficiente excluir alguns papéis e distribuir as responsabilidades pelos restantes, pois poderia levar à destruição do equilíbrio desenvolvido pelo modelo base, fazendo com que os intervenientes não fossem capazes de executar as funções de vários cargos com eficácia., Nesta perspetiva, Borges (Borges, 2007), propôs dar continuidade ao processo de simplificação usando como suporte as seguintes cláusulas:

- Identificação dos papéis essenciais, classificando-os como menos relevantes quando comparados com outros;
- Identificação do papel com condições mais favoráveis para encapsular as responsabilidades de intervenientes excluídos, de acordo com os perfis;
- Validação de novos mapeamentos, tendo o cuidado de não aumentar drasticamente a área de intervenção dos novos papéis, de forma a garantir que

os mesmo consigam desempenhar as suas responsabilidades de forma eficiente e em correlação com as novas responsabilidades intrinsecas a cada função.

Deve-se destacar que a aplicação do modelo reduzido em prol do modelo base, conduz, usualmente, a uma diminuição na qualidade da produção de artefactos ou, eventualmente, a maiores custos na produção dos mesmos. Com o exposto anteriormente passa-se então à apresentação do modelo reduzido sugerido por Borges e apresentando-se na Tabela 10 (Borges, 2007).

<b>Grupo</b>	<b>Papéis do RUP</b>	<b>Modelo Reduzido</b>
<i>Manager</i>	<b>Project Manager</b>	<i>Project Manager</i>
<i>Analyst</i>	<i>Business Process Analyst</i>	
<i>Manager</i>	<i>Change Control Manager</i>	
<i>Manager</i>	<i>Deployment Manager</i>	
<i>Analyst</i>	<i>Requirements Specifier</i>	
<i>Production and Support</i>	<i>Review Coordinator</i>	
<i>Tester</i>	<i>Test Analyst</i>	
<i>Analyst</i>	<i>System Analyst</i>	
<i>Analyst</i>	<i>Business Designer</i>	
<i>Analyst</i>	<i>Use Case Specifier</i>	
<i>Developer</i>	<b>System Integrator</b>	<i>System Integrator</i>
<i>Developer</i>	<i>Capsule Designer</i>	
<i>Developer</i>	<i>Code Reviewer</i>	
<i>Tester</i>	<i>Integration Tester</i>	
<i>Developer</i>	<i>Software Architect</i>	
<i>Developer</i>	<i>Design Reviewer</i>	
<i>Production and Support</i>	<i>Course Developer</i>	
<i>Developer</i>	<i>Data Base Designer</i>	
<i>Manager</i>	<b>Project Reviewer</b>	<i>Project Reviewer</i>
<i>Analyst</i>	<i>Business Reviewer</i>	
<i>Manager</i>	<i>Management Reviewer</i>	
<i>Analyst</i>	<i>Requirements Reviewer</i>	
<i>Manager</i>	<b>Process Engineer</b>	<i>Process Engineer</i>
<i>Developer</i>	<i>Architecture Reviewer</i>	
<i>Production and Support</i>	<i>Tool Specialist</i>	
<i>Developer</i>	<b>Implementer</b>	<i>Implementer</i>
<i>Developer</i>	<i>Component Engineer</i>	
<i>Developer</i>	<i>Designer</i>	
<i>Analyst</i>	<i>User-Interface Designer</i>	
<i>Production and Support</i>	<i>Graphic Artist</i>	

<i>Production and Support</i>	<i>Technical Writer</i>	
<i>Production and Support</i>	<b>System Administrator</b>	<i>System Administrator</i>
<i>Manager</i>	<i>Configuration Manager</i>	
<i>Manager</i>	<b>Test Manager</b>	<i>Test Manager</i>
<i>Tester</i>	<i>Test Designer</i>	
<i>Developer</i>	<i>Use Case Engineer</i>	
<i>Tester</i>	<b>System Tester</b>	<i>System Tester</i>
<i>Additional</i>	<b>Any Role</b>	-
<i>Additional</i>	<b>Stakeholder</b>	-

Tabela 10 - Mapeamento do modelo Reduzido

Encerra-se então a apresentação de ambos os modelos que foram desenvolvidos por Borges (Borges, 2007) durante o processo de investigação no desenvolvimento da sua dissertação. Adicionalmente, pode ser ainda visualizada uma comparação entre ambos os modelos no Anexo B, facilitando a observação dos papéis do modelo reduzido que foram encapsulados e os que se mantiveram como principais. Conclui-se assim, que o modelo reduzido depende fortemente de três intervenientes essenciais: o *Project Manager*, *Integrator* e o *Implementer*, sendo que se uma organização pretender obter sucesso no desenvolvimento de projetos, deverá assegurar um bom desempenho, principalmente nestes três papéis, não desprezando os outros.

Na secção seguinte serão caracterizadas, de forma minuciosa, as responsabilidades atribuídas a cada um dos papéis anteriormente definidos, com o objetivo elucidar sobre as responsabilidades dos intervenientes durante todo o ciclo do projeto.

### 3.5 Caracterização do elenco processual do modelo Reduzido

Uma vez identificados ambos os modelos criados a partir do RUP, ficam ainda por explicitar as responsabilidades de cada interveniente de forma a clarificar a sua área de intervenção. Assim, pretende-se facilitar às organizações a escolha dos intervenientes, adequando o seu perfil às responsabilidades exigidas por cada papel. Em seguida serão explicitadas de forma detalhada as responsabilidades de todos os intervenientes que compõem o elenco processual do modelo Reduzido. Os papéis que apenas fazem parte do elenco processual do modelo Base encontram-se caracterizados no Anexo D.

## *Implementer*

Este interveniente é a unidade central das equipas de desenvolvimento, sendo o mesmo, fundamental para a criação com sucesso dos produtos a serem entregues ao cliente. Por essa razão, espera-se deste interveniente o cumprimento de todas as boas práticas definidas pela organização assim como, um empenhamento para produzir artefactos de qualidade.

### **Responsável por...**

- Ter conhecimento das boas práticas definidas na organização, relacionadas com a criação de artefactos e desenvolvimento de Software.
- Executar de forma profissional e empenhada todas as tarefas atribuídas pelo *System Integrator*.
- Avisar o respetivo *System Integrator* de problemas (ex. incumprimento de prazos) ocorridos durante a execução de determinadas tarefas, aumentando, assim, a probabilidade do *System Integrator* corrigir a situação.
- Informar periodicamente o *System Integrator* da situação atual das tarefas pelas quais é responsável.

### **Perfil...**

Todas as condições apresentadas seguidamente não são obrigatórias, mas é aconselhável que os intervenientes satisfaçam as mesmas.

- Formação de 2º ciclo de Bolonha ou equivalente na área de informática.
- Formação aprofundada nas ferramentas utilizadas pela organização para a implementação.
- Formação avançada de notação UML.

<b>Participação nos Fluxos de Informação Internos</b>	<b>Participação nos Fluxos de Informação Externos</b>
<i>System Integrador</i> : Com objetivo lhe serem atribuídas as tarefas, assim como, de reportar a evolução das mesmas.	Não existe necessidade de intervenção deste interveniente em fluxos de informação externa.

### Funções Críticas...

- Garantir que é aplicada uma intervenção de correção, no caso de falha de cumprimento de objetivos.
- Produzir artefactos com o nível de qualidade exigido pela organização.

### O que deve fazer...

No Início do Projeto	No Final do Projeto
Verificar a situação atual do calendário de férias em vigor. Caso exista algum problema, alertar o <i>Project Manager</i> .	Informar o <i>System Integrator</i> da avaliação realizada aos componentes criados dentro da sua equipa de desenvolvimento.  Comunicar ao <i>Process Engineer</i> a avaliação relativamente ao processo de desenvolvimento de Software que foi utilizado no projeto, sendo ainda desejável, a sugestão de eventuais correções e melhorias.

No Início de cada iteração	Durante cada iteração	No Final de cada iteração
Solicitar ao <i>System Integrator</i> que o coordena, as tarefas correspondentes à iteração que se irá iniciar.	Ter conhecimento das boas práticas definidas na organização, relacionadas com a criação de artefactos e desenvolvimento de Software.	
	Executar de forma profissional e empenhada todas as tarefas atribuídas pelo <i>System Integrator</i> .	Garantir que o código fonte das componentes se encontra atualizado no sistema de controlo de versões.
	Notificar o <i>System Integrator</i> sempre que pretender fazer alteração nos períodos de férias.	Assegurar que o esforço alocado para a realização do seu trabalho se encontra atualizado.
	Avisar o respetivo <i>System Integrator</i> de problemas (ex. incumprimento de prazos) ocorridos durante a execução de determinadas tarefas, aumentando assim, a probabilidade do <i>System Integrator</i> corrigir a situação.	Verificar se pretende fazer alterações no seu calendário de férias que coincida com o início da iteração seguinte, devendo ser o <i>Project Manager</i> notificado de tal situação
	Informar periodicamente o <i>System Integrator</i> da situação atual das tarefas das quais é responsável.	

### *Process Engineer*

Dentro do elenco processual do modelo Reduzido, quase todos os papéis podem ser desempenhados por diferentes colaboradores, em diferentes projetos. O *Process Engineer*, sendo uma das suas responsabilidades a eficiente gestão e colocação dos diversos *Implementers*, *System Integrators* e *Software Architects* pelos vários projetos, é aconselhável que este cargo seja desempenhado pelo mesmo colaborador. Deste modo, o papel de *Process Engineer* serve como conector entre os vários projetos estabelecidos. Cabe, assim, ao *Process Engineer* a monitorização dos diversos *System Integrators* que se encontram a laborar no momento, de forma a assegurar que existe coerência e homogeneidade nos métodos de trabalho, o que, caso a situação o exija, facilita a troca de elementos entre equipas.

Relembra-se que, em algumas situações, a monitorização do *System Integrator* poderá ser adicionalmente acompanhada pelo *Software Architect*, com o intuito de ajudar o *System*

*Integrator* a desenvolver melhores soluções técnicas para os problemas encontrados. Nesta situação o *Process Engineer* deve garantir que o *Software Architect* fornece conhecimentos relevantes que ajudem no desenvolvimento profissional do *System Integrator*, assim como, deve assegurar que o *System Integrator* segue essas diretrizes.

Outro aspecto a salientar é a capacidade de guardar as componentes produzidas pela organização em outros projetos, para futuras utilizações, de forma a evitar a redundância e, conseqüentemente, aumentar a eficiência e qualidade, uma vez que, os componentes já foram desenvolvidos e testados em momentos anteriores e poderão ser reutilizados. Por fim, é de destacar, ainda, que este cargo é responsável por disponibilizar todas as ferramentas necessárias para o suporte ao processo de desenvolvimento.

#### **Responsável por...**

- Ser o conetor entre os diversos projetos que a organização desenvolve em simultâneo, sendo, sempre que necessário, o canal de comunicação entre os intervenientes dos projetos.
- Garantir uma gestão eficaz de todas as competências dos *Implementers*, *System Integrators* e *Software Architects* que coordena, no sentido de os formar para as responsabilidades que vão ter de acarretar.
- Escolher todos e cada um dos *Implementers*, *System Integrators* e *Software Architects* envolvidos em cada projeto de desenvolvimento.
- Garantir a disponibilidade de todas as ferramentas existentes para suportar o processo de desenvolvimento.
- Informar todos os *System Integrators* acerca dos perfis dos *Implementers* que estarão sobre a sua dependência.
- Monitorizar o trabalho colaborativo entre o *System Integrator* e o *Software Architect*.
- Avaliar, em conjunto com o *Project Manager* e o *Software Architects*, o desempenho dos *System Integrators* e tomar as medidas de correção necessárias, caso o desempenho demonstrado pelos mesmos não seja satisfatório.

- Assegurar uma reutilização eficiente dos elementos produzidos anteriormente pela organização.
- Avaliar, em conjunto com o *System Integrator*, o desempenho dos *Implementers* e tomar as medidas de correção necessárias, caso o desempenho demonstrado pelos mesmos não seja satisfatório.
- Garantir, em conjunto com o *Software Architect* e o *System Integrator*, que existe homogeneidade e coerência nos métodos de trabalho utilizados nos vários projetos, facilitando, assim, a estrutura de suporte aos processos de desenvolvimento e troca de *Implementers* entre diversas equipas, caso seja necessário.
- Avaliar, em conjunto com o *System Administrator* e o *Test Manager*, as propostas fornecidas pelos *System Integrators* relacionadas com a documentação necessária a produzir, de forma a garantir uma manutenção apropriada no futuro.
- Quando o *Project Manager* detetar anomalias no cumprimento de determinados objetivos, fazer os possíveis por as resolver, através da alocação de recursos humanos de outros projetos.
- Resolver situações de desentendimento entre o *Project Manager*, *System Integrator*, *Software Architect* ou *Implementer*.
- Assegurar a melhoria contínua e o cumprimento por parte de todos os intervenientes das suas responsabilidades no processo de desenvolvimento de Software.

### Perfil...

Todas as condições apresentadas seguidamente não são obrigatórias, mas é aconselhável que os intervenientes satisfaçam as mesmas.

- Possuir estatuto de responsável topo do departamento de desenvolvimento de Software.
- Formação de 2º ciclo de Bolonha ou equivalente na área de Informática, 2º ciclo em Engenharia de Software e 2º ciclo de Ciências Empresarias.
- Cinco anos de experiência no ramo de desenvolvimento de Software, utilizando métodos iterativos.

- Capacidade de decisão e negociação.
- Capacidade de trabalhar sobre pressão e tolerância à mesma.
- Formação avançada de notação UML.

Participação nos Fluxos de Informação Internos	Participação nos Fluxos de Informação Externos
<p>Software <i>Architect</i> (só no modelo Base): Com objetivo de avaliar as decisões de natureza técnica que originarão a arquitetura de desenvolvimento.</p> <p><i>System Administrator</i>. Com o objetivo de perceber a opinião do mesmo, em relação às propostas sugeridas pelo <i>System Integrators</i> relativamente à documentação a produzir durante o projeto, de forma a assegurar a manutenção futura do mesmo.</p> <p><i>Test Manager</i>. Com o objetivo de perceber a opinião do mesmo, em relação às propostas sugeridas pelo <i>System Integrators</i> relativamente à documentação a produzir durante o projeto, de forma a assegurar a manutenção futura do mesmo.</p> <p><i>System Integrator</i>. Com objetivo de:</p> <ul style="list-style-type: none"> <li>• Monitorizar as atividades de desenvolvimento de todas as equipas.</li> <li>• Garantir que o número de <i>Implementers</i> em cada equipa de trabalho é suficiente para o desenvolvimento do projeto.</li> <li>• Garantir uma homogeneidade no esforço de trabalho entre equipas.</li> <li>• Garantir a reutilização de componentes criadas anteriormente.</li> <li>• Avaliar as decisões de natureza técnica que originarão a arquitetura de desenvolvimento.</li> </ul> <p><i>Project Manager</i>. No sentido de o informar sobre o desempenho dos <i>System Integrator</i> envolvidos em cada projeto.</p>	<p>Não existe necessidade de intervenção deste interveniente em fluxos de informação externa.</p>

### Funções Críticas...

- Assegurar o cumprimento de todas as responsabilidades inerentes a todos os intervenientes no projeto.

- Assegurar que o *System Integrator* segue essas diretrizes do *Software Architect*.
- Alocar *Software Architects*, *System Integrator* e *Implementers* de forma otimizada aos diferentes projetos que acompanha.
- Assegurar que todos os colaboradores coordenados por si se encontram motivados para a realização das suas atividades.
- Assegurar que o *Software Architect* proporciona um acompanhamento devido ao *System Integrator*.

### O que deve fazer...

No Início do Projeto	No Final do Projeto
<p>Selecionar os <i>System Integrators</i>, <i>Software Architects</i> e <i>Implementers</i> que irão estar envolvidos em cada projeto.</p>	
<p>Informar todos os <i>System Integrators</i> acerca dos perfis dos <i>Implementers</i> que estarão sobre a sua dependência.</p>	<p>Verificar a existência de todos os documentos do projeto propostos pelo <i>System Analyst</i> e, também, se se encontram armazenados no repositório da organização.</p>
<p>Garantir, em conjunto com o <i>Software Architect</i> e o <i>System Integrator</i>, que existe homogeneidade e coerência nos métodos de trabalho utilizados nos vários projetos, facilitando, assim, a estrutura de suporte aos processos de desenvolvimento e a troca de <i>Implementers</i> entre diversas equipas, caso necessário.</p>	<p>Analisar as avaliações de competências realizadas por parte do <i>Project Manager</i>, <i>Software Architect</i> e <i>System Integrator</i>.</p>
<p>Avaliar, em conjunto com o <i>System Administrator</i> e o <i>Test Manager</i>, as propostas fornecidas pelos <i>System Integrators</i> relacionadas com a documentação necessária a produzir, de forma a garantir uma manutenção apropriada no futuro</p>	<p>Analisar, em cooperação com o <i>Software Architect</i>, as avaliações realizadas pelos <i>System Integrators</i> dos elementos reutilizados.</p>
<p>Avaliar, caso necessário, as decisões de natureza técnica que originarão a arquitetura de desenvolvimento.</p>	<p>Analisar, de acordo com as avaliações recolhidas anteriormente de diversos colaboradores e com as estratégias definidas pela organização, a necessidade de preparação dos colaboradores para assumirem papéis em projetos futuros.</p>
<p>Analisar se as competências exibidas pelos <i>System Integrators</i>, <i>Software Architects</i> e <i>Implementers</i> são adequadas para a necessidades do projeto, caso contrário tomar as devidas medidas para resolver tal carência.</p>	<p>(Apenas modelo Base) Avaliar as competências comportamentais dos <i>Software Architects</i> que fizeram parte dos projetos.</p>
<p>Avaliar se as ferramentas existentes atualmente na organização correspondem às necessidades para a realização do processo de desenvolvimento de</p>	<p>(Apenas modelo Base) Avaliar as competências técnicas dos <i>System Integrators</i> que fizeram parte dos projetos.</p>

Software, caso contrário tomar as devidas medidas para resolver tal carência.

No Início de cada iteração	Durante cada iteração	No Final de cada iteração
<p>Analisar, em colaboração com os <i>Software Architects</i>, as propostas criadas pelos <i>System Integrators</i> relacionados com as componentes a serem reutilizadas para o desenvolvimento.</p>	<p>Ser o conector entre os diversos projetos que a organização desenvolve em simultâneo, sendo, sempre que necessário, o canal de comunicação entre os intervenientes dos projetos.</p> <p>Quando o <i>Project Manager</i> detetar anomalias no cumprimento de determinados objetivos, fazer os possíveis através da alocação de recursos humanos de outros projetos para a resolução do mesmo</p> <p>Realizar reuniões periódicas com o <i>System Integrator</i> e com o <i>Software Architect</i> para saber o ponto de situação do projeto.</p> <p>Assegurar a melhoria contínua e o cumprimento por parte de todos os intervenientes das suas responsabilidades no processo de desenvolvimento de <i>Software</i> da organização.</p> <p>Resolver situações de desentendimento entre o <i>Project Manager</i>, <i>System Integrator</i>, <i>Software Architect</i> ou <i>Implementer</i>.</p>	<p>Avaliar, em conjunto com o <i>Project Manager</i> e o <i>Software Architects</i>, o desempenho dos <i>System Integrators</i> e tomar as medidas de correção necessárias, caso o desempenho demonstrado pelos mesmos não seja satisfatório.</p> <p>Avaliar conjuntamente com cada <i>System Integrator</i> a necessidade de alocar mais <i>Implementers</i> para o desenvolvimento da próxima iteração.</p> <p>Avaliar, em conjunto com o <i>System Integrator</i>, o desempenho dos <i>Implementers</i> e tomar as medidas de correção necessárias, caso o desempenho demonstrado pelos mesmos não seja satisfatório.</p> <p>Assegurar que o esforço alocado para a realização do seu trabalho se encontra atualizado.</p>

## *Project Manager*

Sendo o *Project Manager* o cargo com mais responsabilidades dentro da equipa organizacional, este é também o papel que influencia com maior peso o sucesso de um projeto de desenvolvimento de Software. Em traços gerais, este interveniente responsabiliza-se por garantir a monitorização geral do projeto e o cumprimento de todos os objetivos acordados com as entidades externas.

Deve, preferencialmente, ser o *Project Manager* a fazer a interligação entre o interior e o exterior da organização, garantindo uma perceção global de toda a informação do projeto relacionada com o âmbito do mesmo. Uma vez que, maioritariamente, estes projetos possuem grande complexidade, espera-se a existência de outros intervenientes, os *Analyst*, para o suportar, passando, assim, algumas das suas responsabilidades para os mesmos e ficando o *Project Manager* como responsável pela sua coordenação.

Adicionalmente, cabe ao Project Manager a gestão das relações entre os intervenientes, no sentido de evitar falhas de comunicação e ultrapassar situações de tensão, que possam influenciar o desempenho global do projeto. Destaca-se ainda que, dado ao seu grande espectro de responsabilidades, não se espera que o Project Manager possua formação avançada nas áreas técnicas, deixando essas incumbências a cargo dos *System Integrators* e dos *Software Architects*.

### **Responsável por...**

- Realizar a avaliação das propostas efetuadas pelos *System Integrators* relacionadas com o número e duração de cada iteração, assim como dos artefactos necessários a serem produzidos por cada equipa de desenvolvimento.
- Criar um plano de alto nível para ser apresentado ao *Project Reviewer*, bem como às entidades externas.
- Recolher os comentários das entidades externas relacionados com o plano produzido.
- Garantir uma imagem positiva da organização perante as entidades externas, através da aceitação do projeto produzido.
- Monitorizar o cumprimento de todos os objetivos propostos para o projeto e, quando detetada alguma anomalia, com o apoio do *Process Enginner* e respetivo

*System Integrator*, procurarem uma solução através da alocação de recursos. Caso tal situação não seja viável, comunicar com as entidades externa e procurar uma solução que seja benéfica para ambas as partes.

- Oficializar os pedidos de alteração de requisitos requeridos pelas entidades externas e comunica-las ao *System Integrator*.
- Recolher e analisar os pedidos de infraestruturas do projeto identificados pelo *Course Developer*, *Test Manager* e *System Integrator*, documentando e comunicando as necessidades dos mesmos ao *System Administrator*.
- Tornar-se o elo de ligação entre a organização e as entidades externas.
- Garantir a faturação do projeto, após as condições necessárias se verificarem.
- Garantir o cumprimento do plano definido e, conseqüentemente, a entrega atempada dos artefactos produzidos pela organização.
- Garantir o cumprimento das responsabilidades de todas as entidades externas que possam influenciar o desenrolar do projeto.
- Identificar, conjuntamente com o resto dos intervenientes, os fatores de risco para o projeto, analisar os riscos e criar medidas para a sua resolução.
- Elaborar *reports* periódicos do consumo de recursos e analisar comparações com iterações anteriores, de forma a verificar o estado atual.
- Garantir uma otimização dos custos do projeto da forma que encontrar mais pertinente, sem prejudicar o desempenho de nenhum interveniente no projeto.
- Garantir a criação de cópias de segurança de toda a informação relevante relacionada com o projeto desenvolvido.
- Ter conhecimento das boas práticas definidas na organização, relacionadas com a criação de artefactos documentais.
- Informar o *Process Enginner* da avaliação de competências realizada aos *System Integrators*.
- Controlar situações de conflito que possam existir entre intervenientes da organização.
- Realizar todas as ações relevantes para o sucesso do projeto que não estejam atribuídas a nenhum outro interveniente ou que necessitem de recursos humanos adicionais.

**Responsabilidades adicionais no modelo Base...**

- Orientar de forma eficiente o trabalho realizado pelos *Analysts* sob sua responsabilidade, tendo em atenção o seu perfil aquando da atribuição das tarefas.
- Fazer a avaliação da documentação produzida pelos *Analysts* de forma a assegurar o cumprimento do âmbito do projeto.
- Fazer a avaliação do plano do *Course Developer* relacionado com a formação a fornecer aos consumidores finais e as linhas de suporte.
- Fazer a avaliação da estruturação final da composição das linhas de suporte definidas pelo *Course Developer*.

#### **Responsabilidades adicionais no modelo Reduzido...**

- Monitorizar o processo de levantamento de requisitos, criando a documentação necessária, de forma a apresentar inequivocamente o âmbito do projeto.
- Avaliar a viabilidade de todos os requisitos identificados, caso não seja verificada a viabilidade do mesmo, e acordar, juntamente com entidades externas, uma solução que seja aceitável para ambas as partes.
- Documentar novos requisitos propostos pelas entidades externas, comunicando-os posteriormente ao *System Integrator* afetado por tal alteração.
- Identificar possíveis requisitos que não foram identificados pelas entidades externas, mas que se podem encontrar dentro das suas expectativas ou que podem criar uma oportunidade comercial.
- Fazer os possíveis no sentido de esclarecer todos os requisitos que não se encontram suficientemente descritos pelas entidades externas, de forma a alinhar o máximo possível as expectativas de ambas as partes.
- Criar e executar os planos de formação, que tem como destino, a formação dos utilizadores finais e das linhas de suporte ao cliente.
- Identificar, conjuntamente com os *System Integrators*, o formato e a localização da documentação de suporte que será desenvolvida por cada equipa de desenvolvimento.
- Garantir o cumprimento de todas as suas funções, por parte de todas as linhas de suporte.
- Caracterizar os perfis dos utilizadores finais.

- Analisar e identificar as infraestruturas necessárias para o bom funcionamento das diversas linhas de suporte.

### Perfil...

Todas as condições apresentadas seguidamente não são obrigatórias, mas é aconselhável que os intervenientes satisfaçam as mesmas.

- Formação de 2º ciclo de Bolonha ou equivalente na área de informática, complementada pelo 2º ciclo em gestão empresarial.
- Experiência de pelo menos um ano em desenvolvimento de Software utilizando métodos iterativos.
- Facilidade na tomada de decisão e negociação.
- Capacidade de lidar com situações de elevada pressão e *stress*.
- Formação avançada em notação UML.

Participação nos Fluxos de Informação Internos	Participação nos Fluxos de Informação Externos
<p><i>Process Engineer:</i> Com o objetivo de o informar sobre o desempenho dos <i>System Integrators</i>.</p> <p><i>System Integrator:</i> Com o objetivo de ser informado sobre o desempenho das linhas de desenvolvimento coordenadas pelo próprio <i>System Integrator</i>.</p> <p><i>System Administrator:</i> Com objetivo de lhe fornecer as informações relevantes relacionadas com a infraestrutura necessárias para o projeto.</p> <p><i>Test Manager:</i> Com objetivo de ser informado sobre o estado da situação do plano de auditoria de qualidade.</p> <p><i>Project Reviewer:</i> No sentido de lhe entregar o plano de projeto e comunicar os requisitos identificados.</p> <p>(Só no modelo Base) <i>Analyst:</i> Com objetivo ser informado sobre os requisitos identificados.</p>	<p><i>Project Manager (Externo):</i> Com objetivo de servir como comunicador entre as entidades externas e a organização, entregar os artefactos produzidos ao cliente final e obter aprovação dos artefactos produzidos. (Só no modelo Reduzido) Obter informações relevantes ao processo de levantamento de requisitos.</p>

(Só no modelo Base) *Software Architect*. Com objetivo de ser informado sobre os riscos técnicos do projeto identificados.

(Só no modelo Base) *User-Interface Designer*. Com objetivo de ser informado das necessidades de imagem e comunicação do projeto, assim como, ter conhecimento do desenvolvimento das suas atividades.

(Só no modelo Base) *Course Developer*. Com objetivo de se informado da proposta de estruturação e formação a aplicar aos utilizadores finais e as equipas de suporte.

### Funções Críticas...

- Monitorizar o desempenho dos seus colaboradores, assim como, protegê-los de situações de *stress* ou pressões externas que possam prejudicar o seu aproveitamento.
- Garantir uma análise cuidadosa dos requisitos que se pretendem alterar, antes de se validar a sua alteração.
- Garantir que os elementos da equipa se encontram motivados para a realização das suas atividades.
- Garantir a faturação do projeto, após as condições necessárias serem verificadas.
- Assegurar que as entidades externas avaliam, o mais cedo possível, os artefactos desenvolvidos, com objetivo de avaliar as expectativas globais da entidade e tomar medidas de acordo com a situação.
- Garantir que os *System Integrators* possuem conhecimento das atividades que estão a ser realizadas pelas outras equipas de desenvolvimento.
- (Só no modelo Reduzido) Garantir que as entidades externas expressem as suas necessidades de forma clara, com objetivo de definir os requisitos para o projeto.
- (Só no modelo Reduzido) Identificar requisitos não exequíveis, e procurar encontrar uma solução aceitável pela entidade externa e viável para ambas as partes.

## O que deve fazer...

No Início do Projeto	No Final do Projeto
<p>Realizar uma reunião, em conjunto com todos os intervenientes do projeto, com o objetivo de lhes explicar o âmbito do projeto e o que se espera de cada um dos intervenientes.</p>	
<p>Criar um plano de alto nível para ser apresentado ao <i>Project Reviewer</i> assim como as entidades externas.</p>	
<p>Identificar, conjuntamente com o resto dos intervenientes os fatores de risco para o projeto, analisar os riscos e criar medidas para a sua resolução.</p>	<p>Garantir a faturação total do projeto.</p> <p>Garantir a execução de cópias de segurança de toda a informação relevante para o projeto.</p>
<p>Analisar o calendário de férias de todos os intervenientes para o período que o projeto irá ser executado.</p>	<p>Pedir as entidades externas o preenchimento de um questionário, de forma a avaliar a satisfação com o trabalho desenvolvido pela organização.</p>
<p>Realizar avaliação das propostas efetuadas pelos <i>System Integrators</i> relacionadas com o número e duração de cada iteração, assim como, dos artefactos necessários a serem produzidos por cada equipa de desenvolvimento.</p>	<p>Assegurar que o <i>System Integrator</i>, <i>Test Manager</i>, <i>Course Developer</i> e <i>System Administrator</i> cumprem com as suas atividades de encerramento de projeto.</p>
<p>Recolher e analisar os pedidos de infraestruturas do projeto identificadas pelo <i>Course Developer</i>, <i>Test Manager</i> e <i>System Integrator</i>, documentando e comunicando as necessidades dos mesmo ao <i>System Administrator</i>.</p>	<p>Comunicar ao <i>Process Engineer</i> os resultados das avaliações realizadas aos <i>System Integrators</i>.</p>
<p>Publicar, no local definido pela organização, informações relacionadas com os elementos da organização e os respetivos papéis e lista de riscos do projeto.</p>	<p>Comunicar ao <i>Process Engineer</i> a avaliação relativamente ao processo de desenvolvimento de Software que foi utilizado no projeto, sendo ainda desejável, a sugestão de eventuais correções e melhorias.</p>
<p>Comunicar ao <i>Project Reviewer</i> o conjunto de requisitos identificados e a documentação respetiva.</p>	<p>(Só no modelo Reduzido) Verificar se a documentação de suporte desenvolvida durante a última iteração cumpre os requisitos definidos.</p>
<p>Garantir que as entidades externas possuem um responsável com poder de decisão e capacidade de avaliar os artefactos entregues pela organização.</p>	<p>(Só no modelo Reduzido) Realizar diversas chamadas teste às linhas de suporte para verificar se as mesmas conseguem responder com sucesso aos desafios propostos.</p>
<p>(Só no modelo Base) Fazer a avaliação da documentação realizada pelos <i>Analysts</i> no sentido de garantir que cumprem o âmbito do projeto.</p>	

(Só no modelo Base) Identificar, em conjunto com o *User-Interface Designer*, as necessidades de comunicação e de imagem do projeto.

(Só no modelo Base) Avaliar a proposta desenvolvida pelo *Course Developer*, relativamente a preparação dos conteúdos necessários para a aplicação do plano de formação às equipas de suporte e aos clientes finais.

(Só no modelo Base) Avaliar a proposta desenvolvida pelo *Course Developer*, relativamente à estrutura das linhas de suporte.

(Só no modelo Reduzido) Caracterizar os perfis dos utilizadores finais.

(Só no modelo Reduzido) Fazer a preparação dos conteúdos necessários para a aplicação do plano de formação às equipas de suporte e aos clientes finais.

(Só no modelo Reduzido) Fazer a revisão, conjuntamente com o *System Integrator*, do formato e localização da documentação de suporte que será desenvolvida na iteração seguinte.

(Só no modelo Reduzido) Identificar as infraestruturas necessárias ao bom funcionamento das linhas de suporte.

(Só no modelo Reduzido) Monitorizar o processo de levantamento de requisitos, criando a documentação necessária, com o intuito de apresentar, de forma inequívoca, o âmbito do projeto.

## *Project Reviewer*

Este interveniente, apesar de pertencer ao projeto, possui uma interação mais externa em comparação com os outros intervenientes, devendo, preferencialmente, contactar exclusivamente com o Project Manager. Sendo um agente influenciador, este interveniente deve utilizar o seu grau de influência para garantir as condições propícias ao sucesso do projeto. Em adição, é também responsável pela monitorização do Project Manager e pela avaliação do seu desempenho.

### **Responsável por...**

- Tentar minimizar os dados causados pelos problemas ocorridos durante a realização do projeto.
- Negociar com entidades externas concessões relevantes para a organização, como o caso de alongamento de prazos.
- Negociar, junto com as entidades externas, compensações devido ao incumprimento do âmbito do projeto.
- Garantir a escolha de um interveniente externo capaz de realizar as responsabilidades atribuídas ao Project Manager (Externo).
- Avaliar a documentação fornecida pelo *Project Manager*, relativa aos requisitos do projeto levantados.
- Garantir a disponibilidade de recursos humanos suficientes para que o projeto se desenvolva sem percalços.
- Garantir que se foca mais atenção nos requisitos classificados pelas entidades externas como mais importantes.
- Analisar e aprovar, caso seja pedido pelo Project Manager, o plano de alto nível do projeto.

### **Perfil...**

Todas as condições apresentadas seguidamente não são obrigatórias, mas é aconselhável que os intervenientes satisfaçam as mesmas.

- Conhecimentos na área de negócio que o projeto irá decorrer.
- Facilidade na tomada de decisão e negociação.

- Capacidade de lidar com situações de elevada pressão e *stress*.

Participação nos Fluxos de Informação Internos	Participação nos Fluxos de Informação Externos
<i>Project Manager</i> : Com objetivo de receber informação, relativa aos requisitos do projeto.	<i>Project Manager (Externo)</i> : Com objetivo de aplicar ações de influência externa.

### Funções Críticas...

- Produzir condições favoráveis para o desenvolvimento do projeto.
- Promover a criação de novas oportunidades comerciais.

### O que deve fazer...

No Início do Projeto	No Final do Projeto
<p>Avaliar, juntamente com o <i>Project Manager</i>, os requisitos identificados e a respetiva documentação.</p> <p>Analisar e aprovar, caso seja pedido pelo <i>Project Manager</i>, o plano de alto nível do projeto.</p>	<p>Verificar o cumprimento dos planos de alto nível da última iteração.</p> <p>Realizar a avaliação do projeto na globalidade, assim como, do <i>Project Manager</i> que o liderou.</p> <p>Comunicar ao <i>Process Engineer</i> a avaliação relativamente ao processo de desenvolvimento de Software que foi utilizado no projeto, sendo ainda desejável, a sugestão de eventuais correções e melhorias.</p>

No Início de cada iteração	Durante cada iteração	No Final de cada iteração
<p>Verificar o cumprimento do plano de alto nível da iteração anterior.</p> <p>Verificar, conjuntamente com o Project Manager, alterações de requisitos que foram identificados.</p> <p>Verificar, juntamente com o Project Manager, alterações requeridas por entidades externas ao plano de alto nível.</p>	<p>Tentar minimizar os danos causados pelos problemas ocorridos durante a realização do projeto.</p> <p>Negociar com entidades externas concessões relevantes para a organização, como o caso de alongamento de prazos.</p> <p>Negociar, junto com as entidades externas, compensações devido ao incumprimento do âmbito do projeto.</p> <p>Garantir a escolha de um interveniente externo capaz de realizar as responsabilidades atribuídas ao Project Manager (Externo).</p> <p>Avaliar a documentação fornecida pelo <i>Project Manager</i>, relativa aos requisitos do projeto levantados.</p> <p>Garantir a disponibilidade de recursos humanos suficientes para que o projeto se desenvolva sem percalços.</p> <p>Garantir que se foca mais atenção nos requisitos classificados pelas entidades externas como mais importantes.</p>	<p>Assegurar que o esforço alocado para a realização do seu trabalho se encontra atualizado.</p>

## *System Administrator*

Existe consenso geral na aceitação de que o sucesso no desenvolvimento de um projeto depende, não só mas em grande parte, de infraestruturas capazes de satisfazer as necessidades da equipa. O *System Integrator* poderá, caso possua competências necessárias, ter esta responsabilidade, contudo é aconselhável a existência de um colaborador próprio para a execução destas atividades, até porque, a utilização de um colaborador externo às várias equipas de desenvolvimento, permite a uniformização e coerências das infraestruturas utilizadas por todos, sendo que, ainda permite aos *System Integrator* focarem-se nas suas atividades das suas responsabilidades.

### **Responsável por...**

- Concretizar todas as necessidades relacionadas com as infraestruturas que forem solicitadas pelo *Project Manager*.
- Sugerir ao *Process Engineer* o formato e localização da documentação arquitetural que será desenvolvida ao longo do projeto.
- Garantir que todos os serviços externos cumprem as suas responsabilidades para com a organização.
- Garantir a execução dos testes de certificação e produção que foram calendarizados pelo *Project Manager*
- Identificar, conjuntamente com o *System Integrator*, os itens que requerem a sua monitorização.
- Identificar os riscos de projeto relacionados com as infraestruturas e, posteriormente, comunicar a sua probabilidade e gravidade ao *Project Manager*.
- Notificar o *Project Manager* sempre que pretender fazer alteração nos períodos de férias.
- Garantir a disponibilidade de um repositório com toda a documentação de instalação e manutenção, assim como, o respetivo Software utilizado para garantir o cumprimento das necessidades da organização.
- Garantir a segurança, fiabilidade e atualização necessárias a todas as infraestruturas utilizadas pelas equipas durante o projeto.

- Garantir que a dimensão das infraestruturas suporta os requisitos não funcionais propostos pelo *Project Manager*.
- Disponibilizar as equipas todas as informações de infraestruturas necessárias à manutenção das mesmas garantindo, assim, o seu bom funcionamento.

#### Responsabilidades adicionais no modelo Reduzido...

- Criar e configurar os motores de bases de dados que lhe foram atribuídos.
- Criar, em colaboração com o *Process Engineer*, políticas de segurança mais adequadas para os motores de base de dados.

#### Perfil...

Todas as condições apresentadas seguidamente não são obrigatórias, mas é aconselhável que os intervenientes satisfaçam as mesmas.

- Formação de 2º ciclo de Bolonha ou equivalente na área de informática.
- Experiência de pelo menos dois anos em Administração de Sistemas.
- Formação avançada na área de segurança de redes.
- Conhecimentos e experiência na configuração de protocolos e ferramentas de monitorização.

Participação nos Fluxos de Informação Internos	Participação nos Fluxos de Informação Externos
<p><i>Software Architect</i>: (Só modelo Base) Com objetivo de perceber os requisitos estabelecidos para as infraestruturas e analisar a viabilidade dos mesmos,</p> <p><i>Process Engineer</i>: Com o objetivo de apoiar na tomada de decisão relacionada com a forma e localização da documentação arquitetural que ira ser produzida durante o projeto.</p> <p><i>System Integrator</i>: Com objetivo de esclarecer incertezas que possam surgir no processo de passagem de certificação para produção e identificar os itens que necessitam de monitorização.</p> <p><i>Project Manager</i>: Com objetivo de informar sobre as necessidades de infraestrutura do projeto,</p>	<p><i>System Administrator</i> (Externos): Com objetivo de esclarecer dúvidas existentes relacionadas com estruturação das infraestruturas, assim como colaborar na resolução de problemas entre as infraestruturas internas e externas.</p>

assim como, trabalho realizado por si ao longo do projeto.

### Funções Críticas...

- Identificar requisitos não viáveis ou documentados com informação insuficiente, relacionados com as infraestruturas e alertar o *Project Manager*.
- Garantir que todos os serviços externos cumprem as suas responsabilidades para com a organização.

### O que deve fazer...

No Início do Projeto	No Final do Projeto
Concretizar todas as necessidades relacionadas com as infraestruturas que forem solicitadas pelo <i>Project Manager</i> .	Garantir a monitorização de todos os itens identificados.
Sugerir ao <i>Process Engineer</i> o formato e localização da documentação arquitetural que será desenvolvida ao longo do projeto.	Comunicar ao <i>Process Engineer</i> a avaliação relativamente ao processo de desenvolvimento de Software que foi utilizado no projeto, sendo ainda desejável, a sugestão de eventuais correções e melhorias.
Identificar, conjuntamente com o <i>System Integrator</i> , os itens que requerem a sua monitorização.	Disponibilizar as equipas todas as informações de infraestruturas necessárias para que consigam realizar manutenção das mesmas garantindo, assim, o seu bom funcionamento.
Identificar os riscos de projeto relacionados com as infraestruturas e, posteriormente, comunicar a sua gravidade e probabilidade ao <i>Project Manager</i> .	Realizar testes às linhas de suporte externo, de modo a avaliar se a forma de resposta das mesma se encontra de acordo com as condições contratadas-
Apresentar no local que a organização definiu, toda a caracterização das infraestruturas que foram definidas para o projeto, de forma a dar apoio ao desenvolvimento do projeto-	(Só no modelo Base) Realizar, sobre a supervisão do <i>Project Manager</i> , uma cópia de segurança de toda a informação relevante que esteja diretamente relacionada com as infraestruturas.
Verificar a situação atual do seu calendário de férias, caso exista alguma anomalia, alertar o <i>Project Manager</i> .	(Só no modelo Reduzido) Realizar, sobre a supervisão do <i>Project Manager</i> , uma cópia de segurança de toda a informação relevante que esteja diretamente relacionada com as infraestruturas e motores de bases de dados.
(Só no modelo Reduzido) Planear, conjuntamente com o <i>Project Manager</i> , as políticas de segurança que irão ser aplicadas no projeto relacionadas com as infraestruturas e motores de bases de dados.	

No Início de cada iteração	Durante cada iteração	No Final de cada iteração
<p>Garantir a execução dos testes de certificação e produção que foram calendarizados pelo <i>Project Manager</i></p> <p>Efetuar alterações às infraestruturas do projeto caso seja solicitado pelo <i>Project Manager</i>.</p> <p>Rever, conjuntamente com o <i>System Integrator</i> os itens que requerem monitorização.</p> <p>Verificar se pretende fazer alguma alteração no seu calendário de férias que coincida com o início da iteração. Em caso de alteração, comunicar ao <i>Project Manager</i>.</p> <p>(Só no modelo Reduzido) Verificar se as políticas de seguranças definidas se encontram a funcionar com sucesso, ou se existe a necessidade de proceder a alterações.</p>	<p>Avaliar a viabilidade dos requisitos estabelecidos para as infraestruturas, e sempre que não se verifique viável, ajudar a encontrar uma solução plausível e aceitável pelas entidades externas.</p> <p>Identificar requisitos não viáveis ou documentados com informação insuficiente, relacionados com as infraestruturas e alertar o <i>Project Manager</i></p> <p>Garantir a disponibilidade de um repositório com toda a documentação de instalação e manutenção, assim como, do respetivo Software utilizado para garantir o cumprimento das necessidades da organização.</p> <p>Garantir a segurança, fiabilidade e atualização necessárias a todas as infraestruturas utilizadas pelas equipas durante o projeto.</p> <p>(Só no modelo Reduzido) Criar e configurar os motores de bases de dados que lhe foram atribuídos.</p>	<p>Calendarizar junto do <i>Project Manager</i> a execução do processo de certificação da produção para a iteração que se segue.</p> <p>Assegurar que o esforço alocado para a realização do seu trabalho se encontra atualizado.</p>

## *System Integrator*

Este cargo encontra-se responsável pela monitorização da produção de todos os artefactos necessários para corresponder aos requisitos estipulados pelo *Project Manager*, por essa razão, juntamente com o *Project Manager* e com o *Process Engineer*, este papel é dos papéis com maior relevância dentro do elenco processual. Desta forma, este papel necessita não só de atribuir todas as tarefas aos *Implementers* pertencentes à equipa, bem como, de monitorizar o trabalho dos mesmos, no sentido de detetar potenciais problemas, tais como atrasos na execução de tarefas. Estas anomalias devem ser comunicadas e resolvidas conjuntamente com o *Project Manager*.

Uma vez que o *Project Manager* pode não possuir a formação técnica que os *Implementers* possuem, este pode apresentar dificuldades em lidar com as tarefas e atividades do projeto. Então, o *System Integrator* deve servir de elo de ligação entre os mesmos e libertar o *Project Manager* do peso que a coordenação de atividades dos *Implementers* acarreta. Outra situação passível de ocorrer é a existência de um número elevado de *Implementers*, que torna inviável a orientação dos mesmos por parte de um único colaborador. Para fazer face a esta situação, o *Project Manager* reparte os colaboradores por várias linhas de desenvolvimento e atribui a coordenação de cada linha a um *System Integrator*, otimizando assim o desenvolvimento e facilitando a monitorização de toda a equipa.

### **Responsável por...**

- Assegurar o cumprimento de todos os objetivos definidos para a equipa de desenvolvimento que coordena.
- Ter conhecimento detalhado da implementação desenvolvida pela sua equipa, sendo capaz de expor, de forma clara, a intervenientes externos e/ou internos tais detalhes, caso seja solicitado.
- Orientar de forma eficiente as atividades e tarefas de todos os *Implementers* que lhe foram atribuídos pelo *Process Engineer* e atribuir-lhes tarefas adequadas ao seu perfil.
- Realizar a avaliação periódica do código fonte, procurando inconformidades com as políticas definidas para o projeto e testando a sua qualidade.

- Uniformizar os métodos de trabalho da equipa de acordo com a diretrizes do *Process Engineer*.
- Sugerir ao Project Manager uma estimativa da duração de cada iteração assim como os artefactos técnicos e documentais a efetuar.
- Fazer o planeamento e execução dos componentes a serem desenvolvidos pela sua equipa de desenvolvimento, com objetivo de produzir as versões necessárias para cada iteração.
- Fazer o planeamento e execução dos testes de integração das versões desenvolvidas, com objetivo de certificar que tudo funciona corretamente em conjunto.
- Sugerir ao *Process Engineer* o formato e documentação a ser produzida durante o projeto.
- Garantir a qualidade na produção da documentação do projeto, bem como, a sua concretização atempada.
- Garantir que os *Implementers* possuem conhecimentos e observam as boas práticas de desenvolvimento de Software utilizadas dentro da organização.
- Avisar atempadamente o Project Manager de possíveis anomalias relacionadas com objetivos impraticáveis no período de tempo da iteração.
- Sugerir ao *Process Engineer* todas as componentes a serem aproveitadas ou não existentes, que possam ser utilizadas ou produzidas pela sua equipa.
- Tomar conhecimento do plano de auditoria de qualidade dos artefactos produzidos pela sua equipa de desenvolvimento.
- Ter uma ideia constante do calendário de férias de todos os *Implementers* que coordena.
- Servir de ligação entre o *Project Manager* e os *Implementers* que coordena.
- Garantir a produção de todos os documentos necessários para apoiar a execução dos processos relacionados com a equipa de desenvolvimento que coordena.
- Notificar o *Project Manager* sempre que pretender fazer alteração nos períodos de férias.
- Avaliar, conjuntamente com o *Process Engineer*, o desempenho global dos *Implementers* que se encontram sob sua orientação.
- Comunicar ao Project Manager as necessidades de infraestruturas identificadas.

- Garantir a disponibilidade do projeto no meio de desenvolvimento.

#### Responsabilidades adicionais no modelo Base...

- Definir as interfaces principais que irão pertencer às componentes do sistema.
- Apoiar o *Software Architect* na tomada de decisões técnicas relacionadas com a arquitetura de desenvolvimento que coordena.

#### Responsabilidades adicionais no modelo Reduzido...

- Avaliar, caso solicitado por um *Implementer*, as interfaces principais que irão pertencer às componentes do sistema.
- Propor ao *Process Engineer* as decisões técnicas relacionadas com a arquitetura de desenvolvimento que coordena.
- Identificar, calcular e medir os riscos técnicos inerentes ao projeto no decorrer de decisões arquiteturais e comunica-los posteriormente ao *Project Manager* e ao *Process Engineer*.
- Preparar os índices, views, constraints, etc. que sejam necessários para otimizar o modelos de dados suportados pelos motores de dados.
- Caracterizar, juntamente com o *Project Manager* toda a documentação que deve ser produzida pela equipa que coordena.
- Criar o modelo de dados que satisfaça as necessidades da equipa que coordena.

#### Perfil...

Todas as condições apresentadas seguidamente não são obrigatórias, mas é aconselhável que os intervenientes satisfaçam as mesmas.

- Formação de 2º ciclo de Bolonha ou equivalente na área de Informática.
- Dois anos de experiência em desenvolvimento de Software, utilizando métodos iterativos.
- Formação avançada em notação UML.

Participação nos Fluxos de Informação  
Internos

Participação nos Fluxos de Informação  
Externos

*Database Designer:* (Só no modelo Base) Com objetivo de fornecer informações relacionadas com os índices, views, constraints, etc. que sejam necessários para otimizar o modelos de dados, assim como, as políticas de segurança da equipa de desenvolvimento que coordena.

*Analyst:* (Só no modelo Base) Com objetivo de receber os requisitos do projeto e avaliar a sua exequibilidade.

*Software Architect:* (Só no modelo Base) Com objetivo de apoiar na tomada de decisões técnicas relacionadas com a arquitetura de desenvolvimento que coordena

*User-Interface Designer.* (Só no modelo Base) Com objetivo de apoiar a identificação das necessidades de interface e a comunicação com a equipa de desenvolvimento que coordena.

*System Administrator:* Com objetivo de esclarecer questões relacionadas com o processo de certificação/produção, assim como identificar os itens que necessitam de monitorização.

*Test Manager:* Com objetivo de prestar declarações relacionadas com o plano de qualidade, com os artefactos produzidos pela equipa que coordena e refletir sobre os resultados.

*Course Developer:* (Só no modelo Base) Com objetivo de criar em conjunto um plano de formação para educar os clientes finais do produto desenvolvido.

*Process Engineer:* com objetivo de:

- Informar da situação da equipa de desenvolvimento que coordena.
- Garantir a cooperação de todos os *Implementers*.
- Garantir que existe o reaproveitamento de componentes criados anteriormente.
- (Só no modelo Reduzido) propor e obter uma avaliação nas decisões técnicas relacionadas com a arquitetura de desenvolvimento que coordena.

*Project Manager:* Com objetivo de acordar ou esclarecer detalhes específicos, de desenvolvimento.

*Test Manager:* Com objetivo de prestar declarações relacionadas com o plano de qualidade, com os artefactos produzidos pela equipa que coordena e refletir sobre os resultados.

*Project Manager:* Com objetivo de informar da situação da equipa de desenvolvimento que coordena

*Implementer:* Com o objetivo de monitorizar e atribuir tarefas.

### Funções Críticas...

- Assegurar o cumprimento dos objetivos de cada iteração realizada pela equipa de desenvolvimento que coordena.
- Garantir que todos os colaboradores que coordena se encontram motivados para a realização das suas tarefas.
- Monitorizar o desempenho dos seus colaboradores, assim como, protegê-los de situações de *stress* ou pressões externas que possam prejudicar o seu aproveitamento.
- Detetar de forma atempada requisitos que não sejam exequíveis nos períodos da iteração ou que não estejam suficientemente explícitos.

### O que deve fazer...

No Início do Projeto	No Final do Projeto
Uniformizar os métodos de trabalho da equipa de acordo com a diretrizes do <i>Process Engineer</i> .	Garantir que o <i>Database Designer</i> e os <i>Implementers</i> que coordena cumpriram todas as atividades de fecho do projeto que lhes foram atribuídas.
Sugerir ao <i>Project Manager</i> o tempo estimado da duração de cada iteração, assim como, os artefactos técnicos e documentais a efetuar	Comunicar ao <i>Process Engineer</i> a avaliação de competências técnicas e comportamentais dos <i>Implementers</i> experienciadas ao longo do projeto.
Sugerir ao <i>Process Engineer</i> o formato e a documentação a ser produzida durante o projeto.	Comunicar ao <i>Process Engineer</i> a avaliação das componentes reutilizadas pela sua equipa de desenvolvimento, sendo ainda desejável, a identificação de eventuais correções e melhorias.
Comunicar ao <i>Project Manager</i> as necessidades de infraestrutura identificadas.	Realizar sobre a supervisão do <i>Project Manager</i> , uma cópia de segurança de toda a informação relevante que esteja diretamente relacionada com a equipa de desenvolvimento que coordenou durante o projeto.
Ter uma ideia constante do calendário de férias de todos os <i>Implementers</i> que coordena.	
Verificar a situação atual do seu calendário de férias e, caso exista algum problema, alertar o <i>Project Manager</i> .	

Publicar, no local definido pela organização, informações relacionadas com:

- Período e duração das iterações a realizar de acordo com o *Project Manager*.
- Formato e localização da documentação de acordo com o *Process Engineer*.
- Métodos de trabalhos acordados com o *Process Engineer*.

(Só no modelo Base) Apoiar o *Software Architect* na tomada de decisões técnicas relacionadas com a arquitetura de desenvolvimento que coordena.

(Só no modelo Reduzido) Propor ao *Process Engineer* as decisões técnicas relacionadas com a arquitetura de desenvolvimento que coordena.

(Só no modelo Reduzido) Identificar, calcular e medir os riscos técnicos inerentes ao projeto no decorrer de decisões arquiteturais e comunica-los, posteriormente, ao *Project Manager* e ao *Process Engineer*.

(Só no modelo Reduzido) Criar o modelo de dados que satisfaça as necessidades da equipa que coordena.

(Só no modelo Reduzido) Caracterizar, juntamente com o *Project Manager*, toda a documentação necessária a ser produzida pela equipa que coordena.

Comunicar ao *Process Engineer* a avaliação relativamente ao processo de desenvolvimento de Software que foi utilizado no projeto, sendo ainda desejável, a sugestão de eventuais correções e melhorias.

No Início de cada iteração	Durante cada iteração	No Final de cada iteração
Sugerir ao <i>Process Engineer</i> todas as componentes a serem aproveitadas ou não existentes que possam ser utilizadas ou produzidas pela sua equipa.	Assegurar o cumprimento de todos os objetivos que foram definidos para a sua equipa de desenvolvimento.	Fazer o planeamento dos testes de integração necessários sobre as versões desenvolvidas, com objetivo de se certificar que tudo funciona corretamente em conjunto.
Realizar a avaliação periódica do código fonte, procurando inconformidades com as políticas definidas para o projeto e testando a sua qualidade.	Identificar possíveis requisitos que não foram identificados pelas entidades externas, mas que se podem encontrar dentro das suas expectativas ou que podem criar uma oportunidade comercial.	Verificar que todas as componentes se encontram com a versão atualizada.
Tomar conhecimento do plano de auditoria de qualidade dos artefactos produzidos pela sua	Ter conhecimento detalhado da implementação desenvolvida pela sua equipa, sendo capaz de expor de forma clara, aos	Garantir que o <i>Database Designer</i> e os <i>Implementers</i> que coordena cumpriram todas as atividades de fecho de iteração que lhes foram atribuídas.

<p>equipa de desenvolvimento na iteração anterior.</p>	<p>intervenientes externos e/ou internos tais detalhes, caso seja solicitado.</p>	<p>Verificar o calendário de férias dos <i>Implementers</i> para o período da próxima iteração.</p>
<p>Fazer o planeamento dos componentes desenvolvidos pela sua equipa de desenvolvimento, com objetivo de produzir as versões necessárias para cada iteração.</p>	<p>Garantir que os <i>Implementers</i> possuem conhecimentos e observam as boas práticas de desenvolvimento de Software utilizadas dentro da organização.</p>	<p>Rever as necessidades de infraestruturas, comunicando-as, em caso de necessidade, ao <i>Project Manager</i>.</p>
<p>Atribuir tarefas a equipa de <i>Implementers</i> que coordena.</p>	<p>Realizar reuniões periódicas com os <i>Implementers</i> para saber o ponto de situação do projeto.</p>	<p>Verificar, juntamente com o <i>Process Engineer</i>, se o número de <i>Implementers</i> é suficiente para cumprir com os objetivos da iteração seguinte.</p>
	<p>Orientar de forma eficiente as atividades e tarefas de todos os <i>Implementers</i> que lhe foram atribuídos pelo <i>Process Engineer</i>. Atribuir tarefas a cada <i>Implementer</i> tendo em conta o seu perfil.</p>	<p>Fazer a avaliação, juntamente com o <i>Process Engineer</i>, dos <i>Implementers</i> que coordenou durante a iteração.</p>
	<p>Servir de ligação entre o <i>Project Manager</i> e os <i>Implementers</i> que coordena.</p>	<p>Assegurar que o esforço alocado para a realização das atividades se encontra atualizado.</p>
	<p>Avisar atempadamente o Project Manager de possíveis anomalias relacionadas com objetivos impossíveis de atingir durante o período de tempo da iteração.</p>	<p>Fazer a revisão, juntamente com o Project Manager, dos artefactos que deverão ser produzidos na próxima iteração, juntamente com as suas características técnicas.</p>
	<p>Garantir a disponibilidade do projeto no meio de desenvolvimento.</p>	<p>Verificar se pretende realizar férias durante o período da próxima iteração caso aconteça notificar o <i>Project Manager</i>.</p>
	<p>Aprovar a viabilidade dos requisitos para o sistema pelo qual é responsável.</p>	<p>(Só modelo Reduzido) Garantir a disponibilidade do material necessário para o suporte à formação.</p>
	<p>Documentar e comunicar ao <i>Project Manager</i> os novos requisitos encontrados durante a implementação.</p>	
	<p>(Só no modelo Base) Definir as interfaces principais que irão pertencer às componentes do sistema.</p>	
	<p>(Só no modelo Base) Identificar, juntamente com o <i>User-Interface</i></p>	

*Designer*, as necessidades de interface e comunicação relacionadas com a equipa de desenvolvimento que coordena.

(Só no modelo Reduzido)  
Identificar, calcular e medir os riscos técnicos inerentes ao projeto no decorrer de decisões arquiteturais e comunicá-los, posteriormente, ao *Project Manager* e ao *Process Engineer*.

(Só no modelo Reduzido)  
Preparar os índices, views, constraints, etc. que sejam necessários para otimizar o modelos de dados suportados pelos motores de dados pelo qual é responsável.

(Só no modelo Reduzido) Criar conteúdos apropriados à formação aos utilizadores finais do produto desenvolvido.

## *System Tester*

A intervenção dos *System Tester* durante o projeto possui como objetivo essencial o apoio ao *Test Manager*, garantindo que a qualidade dos artefactos que as entidades externas recebem é positiva e contribuindo para uma boa imagem da organização. Neste sentido espera-se que este interveniente realize com elevado desempenho todas as tarefas que lhe foram atribuídas pelo *Test Manager*.

### **Responsável por...**

- Executar, de forma excepcional, todas as suas tarefas, uma vez que influenciam a imagem organizacional, bem como, a perceção que as entidades externas vão ter do produto.
- Assegurar que são observadas as boas práticas no que diz respeito à criação de artefactos documentais que se encontram em vigor na organização.
- Garantir que todos os artefactos produzidos ao longo do projeto se encontram de acordo com os requisitos identificados pelo *Project Manager* ou pelo *System Analyst*.
- Garantir que o *Test Manager* possui conhecimento do tempo que o *System Tester* necessita para a realização das suas tarefas.
- Notificar o *Test Manager* sempre que pretender fazer alteração nos períodos de férias.
- Assegurar que todos os artefactos documentais possuam coerência entre si, isto é, não sejam afetados pelas diferentes características comportamentais e psicológicas dos diversos intervenientes.

### **Perfil...**

Todas as condições apresentadas seguidamente não são obrigatórias, mas é aconselhável que os intervenientes satisfaçam as mesmas.

- Formação de 2º ciclo de Bolonha ou equivalente na área de informática, caso seja o responsável pela análise de artefactos aplicativos.

- Conhecimentos de informática do ponto de vista do utilizador, caso não esteja responsável pela análise de artefactos aplicativos.
- Formação básica da notação UML.

Participação nos Fluxos de Informação Internos	Participação nos Fluxos de Informação Externos
<i>Test Manager</i> : Com objetivo de receber as tarefas que lhe são atribuídas e de, posteriormente, o informar sobre a evolução das mesmas.	Não existe necessidade de intervenção deste interveniente em fluxos de informação externa.

### Funções Críticas...

- Contribuir para a criação de uma imagem positiva da organização na ceio das entidades externas.
- Garantir a deteção atempada de erros que poderiam, de outra forma, proporcionar uma má experiência às entidades externas aquando da sua utilização.

### O que deve fazer...

No Início do Projeto	No Final do Projeto
Verificar a situação atual do seu calendário de férias, caso exista algum problema, alertar o <i>Test Manager</i> .	Comunicar ao <i>Process Engineer</i> a avaliação relativamente ao processo de desenvolvimento de Software que foi utilizado no projeto, sendo ainda desejável, a sugestão de eventuais correções e melhorias.

No Início de cada iteração	Durante cada iteração	No Final de cada iteração
<p>Colaborar na produção do plano interno de auditoria de qualidade, definido para todos os artefactos produzidos durante a iteração anterior, pedindo ao <i>Test Manager</i> para proceder à atribuição de tarefas para presente iteração.</p>	<p>Execução excecional de todas as suas tarefas uma vez que, influenciam a imagem organizacional, bem como, a perceção que as entidades externas vão ter do produto.</p> <p>Assegurar que são observadas as boas práticas no que diz respeito à criação de artefactos documentais que se encontram em vigor na organização.</p> <p>Garantir que todos os artefactos produzidos ao logo do projeto se encontram de acordo com os requisitos identificados pelo <i>Project Manager</i> ou pelo <i>System Analyst</i>.</p> <p>Notificar o <i>Test Manager</i> sempre que pretender fazer alteração nos periodos de férias.</p> <p>Assegurar que todos os artefactos documentais possuam coerência entre si, isto é, não sejam afetados pelas diferentes características comportamentais e psicológicas dos diversos intervenientes</p>	<p>Assegurar que o esforço alocado para a realização do seu trabalho se encontra atualizado.</p> <p>Verificar se pretende fazer alterações no seu calendário de férias que coincida com o início da iteração seguinte, devendo ser o <i>Test Manager</i> notificado de tal situação</p>

## *Test Manager*

Como já referido anteriormente este interveniente desempenha um papel de extrema importância, uma vez que é o responsável pelo planeamento e realização dos testes de auditoria de qualidade, de todos os artefactos documentais e aplicativos desenvolvidos pela equipa de desenvolvimento, garantindo, assim, que os mesmos se encontram de acordo com os requisitos levantados e caracterizados pelo *Software Architect*, ou caso não exista, pelo *Project Manager*.

Do exposto, pretende-se clarificar que este interveniente funciona como “degustador de vinhos” antes de o vender às entidades externas, devendo ser capaz de detetar problemas com baixa gravidade, mas que podem afetar a imagem da organização, caso sejam experienciados pela entidade externa. Outro ponto relevante é o facto de toda a documentação produzida no projeto ser realizada por diversos intervenientes com hábitos diferentes, níveis de formalidade na linguagem e formatações visuais distintas, o que leva à necessidade de um interveniente capaz de homogeneizar globalmente todo o documento, de forma a mostrar que a documentação foi produzida por uma organização e não por um conjunto de intervenientes distintos.

### **Responsável por...**

- Garantir a qualidade de todos os artefactos produzidos durante a execução do projeto.
- Garantir que as boas práticas de criação de artefactos definidas pela organização são aplicadas na produção dos documentos.
- Garantir que os artefactos documentais possuem coerência visual e terminológica.
- Garantir que todas as cópias de segurança realizadas sobre a supervisão do Project Manager possuem a informação correta.
- Criar, conjuntamente com o *Project Manager* e o *System Integrator*, o plano interno de qualidade, por forma a garantir que todos os artefactos se encontram de acordo com os requisitos identificados.
- Proceder à aprovação de todos os artefactos entregues às entidades externas.
- Comunicar periodicamente ao *Project Manager* e ao *System Integrator* o estado do cumprimento dos requisitos acordados com as entidades externas.

- Orientar de forma eficiente o trabalho realizado pelos *System Testers* sob sua responsabilidade, tendo em atenção o seu perfil aquando da atribuição das tarefas.
- Sugerir ao *Process Engineer* formato e localização da documentação arquitetural que será desenvolvida ao longo do projeto.
- Notificar o *Project Manager* sempre que pretender fazer alteração nos períodos de férias.

#### Responsabilidades adicionais no modelo Reduzido...

- Garantir a probidade de todos os casos de usos resultantes do modelo de casos de uso, assegurando que todos cumprem os respetivos requisitos.

#### Perfil...

Todas as condições apresentadas seguidamente não são obrigatórias, mas é aconselhável que os intervenientes satisfaçam as mesmas.

- Formação de 2º ciclo de Bolonha ou equivalente na área de informática.
- Experiência de pelo menos dois anos em desenvolvimento de Software, nomeadamente com a utilização de métodos iterativos.
- Formação avançada em notação UML.

Participação nos Fluxos de Informação Internos	Participação nos Fluxos de Informação Externos
<p><i>System Tester</i>: Com objetivo de conferir tarefas a realizar, assim como, monitorizar a realização das mesmas.</p> <p><i>Process Engineer</i>: Com o objetivo de apoiar na tomada de decisão relacionada com a forma e localização da documentação arquitetural que será ser produzida durante o projeto.</p> <p><i>System Integrator</i>: Com objetivo de esclarecer dúvidas relacionadas com a execução do plano interno de qualidades nos artefactos produzidos pela equipa de desenvolvimento, monitorizada pelo mesmo.</p>	<p>Não existe necessidade de intervenção deste interveniente em fluxos de informação externa.</p>

*Project Manager*: Com o objetivo de o manter informado relativamente a execução do plano interno de auditoria de qualidade, assim como, por aprovar a entrega de cada artefacto.

### Funções Críticas...

- Criar a imagem positiva da organização para como as entidades externas.
- Garantir que os *System Testers* se encontram motivados para desempenhar as suas funções.
- Otimizar o processo de teste dos artefactos de forma a diminuir o tempo de produção do projeto final e, conseqüentemente, entregar mais rapidamente o produto às entidades externas.
- Garantir que os problemas, mesmo de pequena dimensão, são reportados e corrigidos, de forma a evitar a degradação da imagem da organização perante as entidades externas.

### O que deve fazer...

No Início do Projeto	No Final do Projeto
<p>Criar, conjuntamente com o <i>Project Manager</i> e o <i>System Integrator</i>, o plano interno de qualidade, como forma de garantir que todos os artefactos se encontram de acordo com os requisitos identificados.</p> <p>Sugerir ao <i>Process Engineer</i> o formato e localização da documentação arquitetural que será desenvolvida ao longo do projeto.</p> <p>Verificar a situação atual do seu calendário de férias, caso exista algum problema, alertar o <i>Project Manager</i>.</p>	<p>Certificar-se que a documentação da última iteração foi produzida e cumpre com os requisitos definidos pelo <i>Process Engineer</i>.</p> <p>Monitorizar a execução do plano de controlo de qualidade que foi definido para os artefactos da última iteração.</p> <p>Comunicar a ambos, <i>Project Manager</i> e <i>System Integrator</i>, a avaliação realizada aos artefactos produzidos na última iteração.</p> <p>Garantir que o <i>System Tester</i> realizou todas as suas atividades de fecho de projeto.</p> <p>Comunicar ao <i>Process Engineer</i> a avaliação relativamente ao processo de desenvolvimento de Software que foi utilizado no projeto, sendo ainda desejável, a sugestão de eventuais correções e melhorias.</p>

Garantir que todas as cópias de segurança realizadas sobre a supervisão do *Project Manager* possuem a informação correta.

No Início de cada iteração	Durante cada iteração	No Final de cada iteração
<p>Fazer distribuição de tarefas por todos os <i>System Testers</i> que coordena.</p> <p>Monitorizar a execução do plano de controlo de qualidade que foi definido para os artefactos da iteração anterior.</p> <p>Comunicar a ambos, <i>Project Manager</i> e <i>System Integrator</i>, a avaliação realizada aos artefactos que foram produzidos na iteração anterior.</p> <p>Avaliar se a documentação da iteração anterior foi produzida e cumpre com os requisitos definidos pelo <i>Process Engineer</i>.</p>	<p>Realizar reuniões periódicas com os <i>System Testers</i> para saber o ponto de situação do projeto.</p> <p>Garantir que são utilizadas as boas práticas definidas pela organização, no que concerne a produção de documentos.</p> <p>Garantir que os artefactos documentais possuem coerência visual e terminológica.</p> <p>(Só no modelo Reduzido) Garantir a probidade de todos os casos de usos resultantes do modelo de casos de uso, assegurando que todos cumprem os respetivos requisitos</p>	<p>Informar o <i>Project Manager</i> e o <i>System Integrator</i> do cumprimento atual dos requisitos definidos com o cliente.</p> <p>Verificar se pretende fazer alterações no seu calendário de férias que coincida com o início da iteração seguinte, devendo ser o <i>Project Manager</i> notificado de tal situação</p> <p>Garantir que o <i>System Tester</i> realizou todas as suas atividades de fecho de iteração.</p> <p>Assegurar que o esforço alocado para a realização do seu trabalho se encontra atualizado</p> <p>Verificar se o calendário de férias dos <i>System Testers</i> não coincide com a execução da próxima iteração.</p>

No Início de cada iteração	Durante cada iteração	No Final de cada iteração
<p>Identificar, conjuntamente com o resto dos intervenientes, os fatores de risco para o projeto, analisar os riscos e criar medidas para a sua resolução.</p>	<p>Garantir o cumprimento das responsabilidades de todas as entidades externas que possam influenciar o desenrolar do projeto.</p>	<p>Avaliar em conjunto com as entidades externas o plano de alto nível efetuado, de forma a informar o <i>System Integrator</i> da lista de artefactos a produzir na próxima iteração.</p>
<p>Garantir a entrega dos artefactos produzidos na iteração anterior às entidades externas.</p>	<p>Obter comentários das entidades externas sobre a avaliação do projeto.</p>	<p>Avaliar a situação das condições necessárias para proceder à faturação de mais uma parcela do projeto.</p>
<p>Obter os comentários das entidades externas, relativos aos artefactos entregues que foram produzidos na iteração anterior.</p>	<p>Monitorizar constantemente o cumprimento dos objetivos da iteração. Caso seja detetada uma anomalia, avaliar, juntamente com o <i>Process Engineer</i> e o respetivo <i>System Integrator</i>, as possíveis soluções a serem aplicadas.</p>	<p>Avaliar, em conjunto com o <i>Process Engineer</i> e com o <i>Software Architect</i>, o desempenho do <i>System Integrator</i>.</p>
<p>Elaborar um relatório de alocação de recursos, tendo como base as iterações anteriores.</p>	<p>Realizar reuniões periódicas com os <i>System integrator</i> para saber o ponto de situação do projeto.</p>	<p>Analisar o calendário de férias de todos os intervenientes para o período da iteração seguinte.</p>
<p>(Só no modelo Base) Identificar, em conjunto com o <i>User-Interface Designer</i>, as necessidades de comunicação e imagem do projeto da iteração que irá ser iniciada.</p>	<p>Garantir uma otimização dos custos do projeto, da forma que encontrar mais pertinente, sem prejudicar o desempenho de nenhum interveniente no projeto.</p>	<p>Assegurar que o <i>System Integrator</i>, <i>Test Manager</i>, <i>Course Developer</i> e <i>System Administrator</i> cumprem com as suas atividades de encerramento de iteração.</p>
<p>(Só no modelo Base) Distribuir tarefas pelos <i>Analyst</i>, que as irão coordenar.</p>	<p>Conhecer as boas práticas existentes na organização relacionadas com a produção de artefactos documentais.</p>	<p>Assegurar que o esforço alocado para a realização do seu trabalho se encontra atualizado.</p>
<p>(Só no modelo Base) Verificar, em conjunto com os <i>Analyst</i> se a documentação dos requisitos precisa de ser atualizada.</p>	<p>Realizar todas as ações relevantes para o sucesso do projeto que não estejam atribuídas a nenhum outro interveniente, ou que necessitam de recursos humanos adicionais.</p>	<p>(Só no modelo Reduzido) Garantir a disponibilização do material de suporte à formação dos utilizadores finais.</p>
<p>(Só no modelo Reduzido) Preparar, em conjunto com o <i>System Integrator</i>, os conteúdos necessários para a aplicação do plano de</p>	<p>(Só no modelo Base) Realizar reuniões periódicas com os <i>Analyst</i> para saber o ponto de situação do projeto.</p>	<p>(Só no modelo Reduzido) Assegurar a existência de formação dos utilizadores finais e das linhas de suporte, caso se verifique essa necessidade.</p>
	<p>(Só no modelo Reduzido) Avaliar a viabilidade de todos os requisitos identificados. Caso se detete alguma anomalia, acordar, juntamente com</p>	

formação às equipas de suporte e aos clientes finais.	entidades externas, uma solução que seja aceitável para ambas as partes.	(Só no modelo Reduzido) Rever as necessidades de infraestruturas e verificar se existe necessidade de fazer alterações.
(Só no modelo Reduzido) Verificar se a documentação produzida relativamente à iteração anterior cumpre os requisitos do projeto.	(Só no modelo Reduzido) Documentar novos requisitos propostos pelas entidades externas, comunicando posteriormente ao <i>System Integrator</i> afetado por tal alteração.	
(Só no modelo Reduzido) Rever, conjuntamente com o <i>System Integrator</i> , o formato e localização da documentação de suporte que será desenvolvida na iteração que se irá realizar.	(Só no modelo Reduzido) Identificar possíveis requisitos que não foram identificados pelas entidades externas, mas que se podem encontrar dentro das suas expectativas ou que podem criar uma oportunidade comercial.	
	(Só no modelo Reduzido) Fazer todos os possíveis no sentido de esclarecer todos os requisitos que não se encontram suficientemente descritos pelas entidades externas, de forma a alinhar o máximo possível as expectativas de ambas as partes.	
	(Só no modelo Reduzido) Garantir o cumprimento de todas as suas funções, por parte de todas as linhas de suporte.	
	(Só no modelo Reduzido) Garantir a produção do material necessário de formação e suporte.	

Dá-se por concluída, nesta fase, a caracterização dos papéis do modelo Reduzido definido por Borges em (Borges, 2007) durante a realização da sua dissertação. É altura, agora, de aplicar estes modelos a uma situação atual.

No próximo capítulo iniciar-se-á a análise ao modelo caracterizado anteriormente, com o objetivo de o otimizar. Para esse efeito, irá-se efetuar um estudo de caso onde se irá aplicar esse modelo a equipas de desenvolvimento. O estudo de caso forçar-se-á no estudo de equipas de desenvolvimento em contexto académico, mais precisamente, as equipas de desenvolvimento de Software da unidade curricular de Desenvolvimento de Aplicações Informáticas de segundo ano do curso de Engenharia e Gestão de Sistemas de Informação.



## 4 Acompanhamento das Equipas de Desenvolvimento de Projetos

Este capítulo envolve a componente prática desta dissertação, na qual foi aplicada a abordagem metodológica de estudo de caso às equipas de desenvolvimento da unidade curricular de Desenvolvimento de Aplicações Informáticas. Neste capítulo é ainda possível visualizar todos os dados recolhidos relacionados com os grupos de trabalho, bem como, as respetivas análises e conclusões. Por fim, apresentam-se algumas sugestões de alterações ao modelo Reduzido de acordo com as conclusões retiradas, com base nos dados analisados.

Após descrição dos papéis do modelo Reduzido e do modelo Base, é chegada a hora de avaliar a sua praticabilidade numa envolvente profissional. Com isso em mente, em ambiente académico, foram selecionadas equipas de desenvolvimento de Software da Unidade Curricular de Desenvolvimento de Aplicações Informáticas (DAI) do segundo ano do Mestrado integrado em Engenharia e Gestão de Sistemas de Informação (MIEGSI), uma vez que as equipas utilizavam a *framework* do RUP e os papéis pertencentes ao modelo Reduzido

Desenvolvimento de Aplicações Informáticas é a unidade curricular de integração de vários conhecimentos adquiridos até ao momento, onde os alunos precisam de aplicar e relacionar vários modelos, assim como vários ambientes de desenvolvimento.

Os modelos utilizados durante o projeto são: o CMMI para otimização de processo, O PMBOK para a gestão de Projeto e o RUP como metodologia iterativa de desenvolvimento de Software. Os ambientes de desenvolvimento com que estes alunos interagem são: Java para o desenvolvimento *desktop* do produto, Base de Dados para o armazenamento de todas as informações relevante ao produto e *Web* para desenvolvimento online do produto. Sendo o cliente do projeto desta unidade curricular um cliente real, as equipas de desenvolvimento servem como um ótimo estudo de caso para avaliar, num ambiente experimental, o desempenho do elenco processual do RUP reduzido, sem correr os riscos que o insucesso de um projeto acarreta.

Relativamente ao contexto de DAI, o projeto é dividido em cinco momentos de avaliação, onde as equipas de desenvolvimento deverão apresentar determinados artefactos desenvolvidos durante esses momentos. A fase do ciclo de vida do RUP, assim como, os artefactos que devem ser entregues nos momentos de avaliação, podem ser visualizados na Tabela 11.

Momento 1	Momento 2 ( <i>Inception</i> )	Momento 3 ( <i>Elaboration</i> )	Momento 4	Momento 5 ( <i>Construction</i> )
<i>Software Development Plan</i>	<i>Issues List</i>	<i>Software Architecture Document</i>	Momento de avaliação individual do Alunos. Não existe entrega de nenhum artefacto documental.	<i>The System</i>
<i>Risk List</i>	<i>Iteration Assessment</i>	<i>Design Model</i>		<i>Deployment Plan</i>
<i>Iteration Plan</i>	<i>Project Measurements</i>	<i>Data Model</i>		<i>Review</i> e versões finais dos artefactos dos Momentos 1, 2 e 3
	<i>Review Record</i>	<i>Implementation Model</i>		
	<i>Status Assessment</i>	<i>Supplementary Specifications</i>		
	<i>Work Order</i>	<i>Test Suite</i>		
	<i>Vision</i>	<i>Test Automation Architecture</i>		
	<i>Business Case</i>	<i>User Support Material</i>		
	<i>Development Process</i>	<i>Iteration Plan (Construction)</i>		
	<i>Development Infrastructure</i>	<i>Review</i> e novas versões dos artefactos dos Momentos 1 e 2		
	<i>Glossary</i>			
	<i>Use Case Model</i>			
	<i>Domain Model</i>			
	<i>Prototypes</i>			
	<i>Iteration Plan (Elaboration)</i>			
	<i>Software Development Plan review</i>			
	<i>Risk List review</i>			

Tabela 11 - Artefactos a desenvolver durante os momentos de avaliação

Ao longo deste capítulo serão caracterizadas as equipas que participaram neste estudo de caso, a forma como os intervenientes foram distribuídos, analisada a distribuição de tarefas, assim como o tempo investido na sua realização. Por fim, apresentam-se as conclusões sobre o estado dos modelos, bem como a necessidade de implementar algumas alterações para cumprir os objetivos definidos inicialmente nesta dissertação.

## 4.1 Caracterização da maturidade das equipas de desenvolvimento

O estudo de caso realizado envolveu seis equipas de desenvolvimento, com uma média de dezasseis elementos por equipa, que realizaram o projeto em causa na unidade curricular “Desenvolvimento de Aplicações Informáticas” (DAI), do segundo ano, do curso do Mestrado Integrado em Engenharia e Gestão de Sistemas de Informação (MIEGSI) no ano letivo 2014/2015.

De forma a ser visível o nível de maturidade das equipas de desenvolvimento e a clarificar o conhecimento dos intervenientes que fizeram parte do projeto, foi realizado um questionário no início do mesmo (Momento 1, duas semanas depois do início do segundo semestre) com diversas questões, que tinham como objetivo a recolha de dados para responder aos pontos definidos, citados de seguida:

1. Idade;
2. Género;
3. Ano que está inscrito;
4. Estatuto trabalhador/estudante;
5. Unidades Curriculares em atraso;
6. Inscrições no Curso de MIEGSI;
7. Identificação do grupo;
8. Papel Atribuído de acordo com o modelo Reduzido
9. Conhecimento relativamente às responsabilidades do seu papel;
10. Conhecimento relativamente às responsabilidades de todos os papéis desempenhados pela equipa;
11. Experiência real em campo com o papel que lhe foi atribuído.

Os pontos (1 e 2) não são muito relevantes ao desempenho do intervenientes em si, mas ajuda a visualizar a faixa etária, assim como, verificar se existe distribuição uniforme do género. O ponto (3) pode influenciar o desempenho dos intervenientes e ajuda a distinguir a maturidade dos intervenientes.

Os pontos (4, 5 e 6) foram utilizados para tentar analisar a forma como as unidades curriculares em atraso influenciam perceção do desempenho do interveniente e a sua relação com o sucesso do projeto.

Os pontos (7 e 8) serviram para mostrar a distribuição dos intervenientes pelos grupos, assim como a distribuição dos papéis pelos grupos, com objetivo de analisar as diversas distribuições possíveis de ocorrer nos seis grupos, sendo que o número de intervenientes em cada grupo e os conhecimentos eram diferentes.

Os últimos três pontos (9, 10 e 11) servem para avaliar se os elementos dos grupos já possuíam algum conhecimento das suas responsabilidades como intervenientes, antes de iniciarem o projeto.

As próximas quatro subsecções incidirão na análise dos gráficos que resultaram da recolha de dados proveniente dos questionários, com objetivo de responder as métricas definidas anteriormente.

#### **4.1.1 Avaliação da distribuição das faixas etárias e do género**

Esta subsecção destina-se a apresentar e relatar os resultados dos pontos (1 e 2) dos questionários realizados às equipas, assim como a retirar breves conclusões sobre os mesmos. Em primeiro lugar, iremos analisar o nível de dispersão de idades dos noventa e dois (92) intervenientes que participaram no projeto, como mostra a Figura 2.

Depois de analisada a figura observa-se que a maioria dos intervenientes possuem idades entre os dezanove e vinte e um anos, sendo estes valores esperados atendendo a que a Unidade Curricular de DAI pertence ao segundo ano do curso de MIEGSI e a idade média dos alunos nesse ano é de vinte anos, apesar das idades dos intervenientes variarem entre os 19 e os 53 anos.

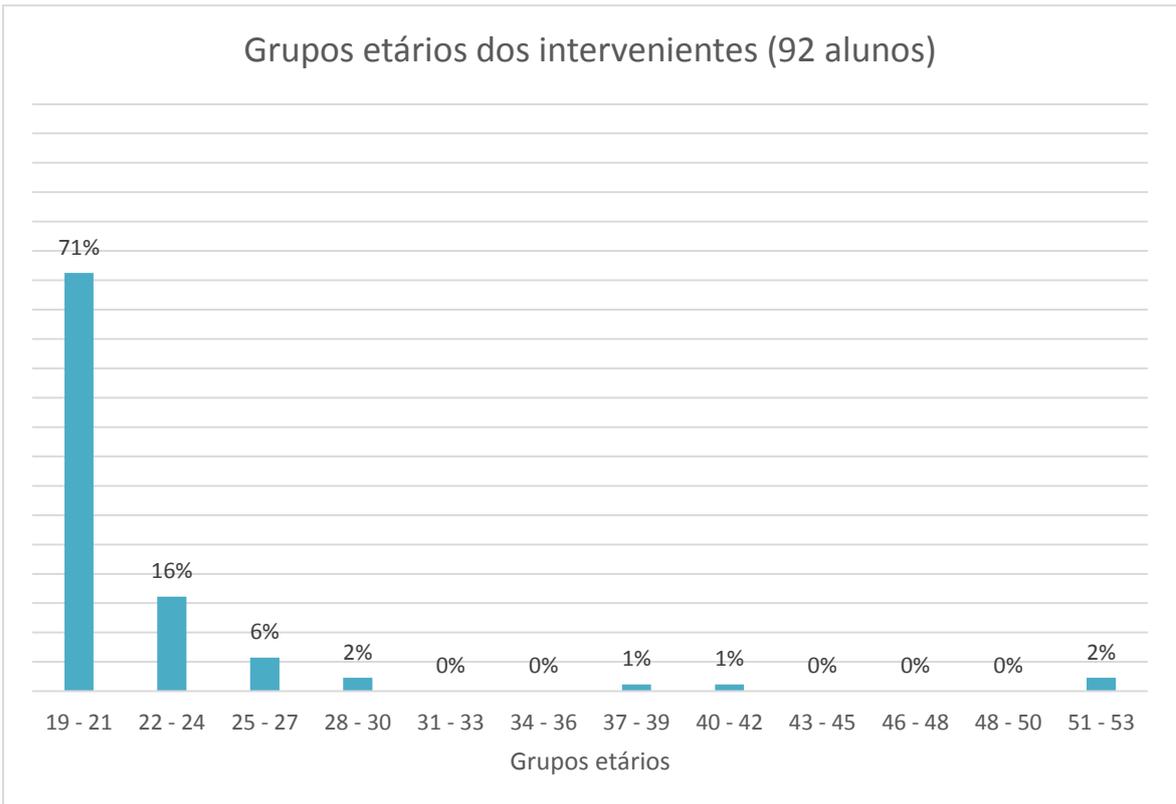


Figura 2 - Grupos etários dos intervenientes

Pode-se especular que dentro destes grupos de desenvolvimento existe um pequeno número de intervenientes que provavelmente possui uma maturidade superior à maioria, uma vez que, face ao grupo etário em que se encontram, possivelmente já possuem a experiência de trabalhar no mundo real.

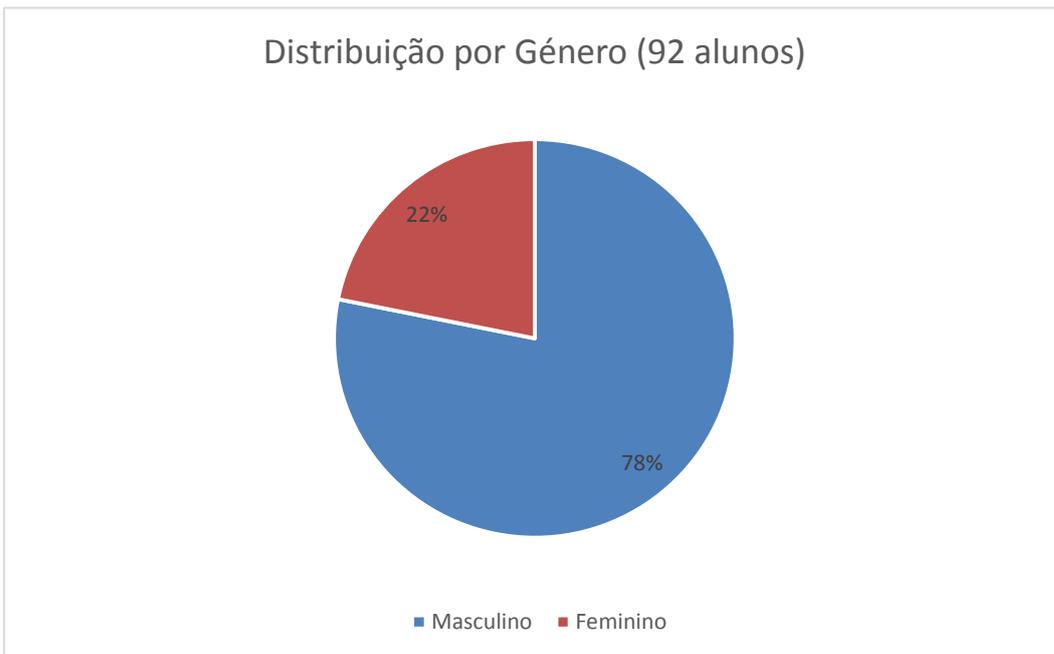


Figura 3 - Distribuição do Género

O segundo gráfico representado pela Figura 3, mostra a distribuição do género nas equipas de desenvolvimento. Apesar de, supostamente, em termos práticos, este gráfico não ter muita relevância, pretendia-se visualizar se existia situações de uma minoria acentuada num dos géneros que pudesse influenciar o desempenho desses mesmos intervenientes.

Contudo, observa-se que apesar do género masculino predominar, quase um quarto da população em estudo era do género feminino, com intervenientes dos dois géneros distribuídos de forma heterogénea dentro dos grupos. Não é possível, nem faz parte do âmbito deste projeto, estudar a correlação entre resultados obtidos e o género. Pode-se, apenas, concluir que ainda se constata uma predominância do género masculino nesta unidade curricular.

#### 4.1.2 Relações entre unidades curriculares, inscrições e estatuto do aluno

À semelhança da subsecção anterior, esta subsecção pretende analisar e comentar os resultados dos questionários relativamente aos pontos (3, 4 e 5) referidos anteriormente. Nesta subsecção pretendemos analisar a relação entre desempenho e o esforço acrescido a que estão sujeitos os intervenientes que possuem unidades curriculares em atraso, relativamente aos restantes. Pretende-se ainda ter uma ideia do rácio entre trabalhadores estudantes e alunos do regime normal, uma vez que os trabalhadores estudantes apesar de maiores responsabilidades, tendem a ter um desempenho inferior a um aluno do regime normal que cumpra todas as suas responsabilidades, devido ao desgaste provocado pelo emprego que possuem para além da vida académica.

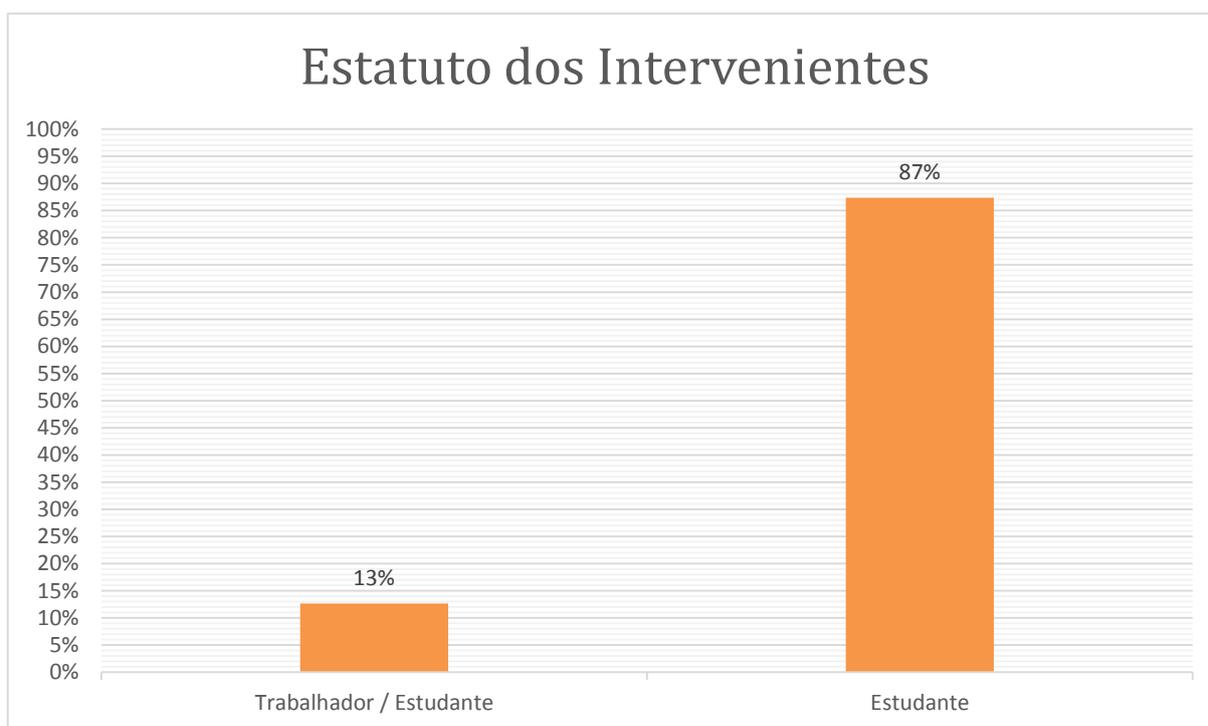


Figura 4 - Estatuto dos Intervenientes do Projeto

No gráfico identificado pela Figura 4, pode-se visualizar a distribuição do estatuto dos alunos que participaram no estudo de caso. Podemos observar que dentro das equipas de desenvolvimento existe uma pequena minoria de trabalhadores estudantes e que, uma vez que, existe um turno próprio para os alunos pós-laboral, um dos grupos será formado exclusivamente por alunos com este estatuto.

O gráfico da Figura 5 é o resultado da relação entre o ano atual em que os alunos se encontram e o número de inscrições realizadas no curso pelos mesmos até ao momento. Não existe nenhuma surpresa nos resultados do gráfico uma vez que para um aluno do regime normal que comece o percurso académico no curso de MIEGSI, é necessário o nível dois, ou seja, o segundo ano, para realizar a unidade curricular DAI. Assim, seria de supor a necessidade de no mínimo duas inscrições no curso. Contudo, por outro lado, curiosamente, existem alunos que se encontram presentes no segundo ano mas, que apenas possuem uma inscrição. Isto resulta provavelmente desses 6% serem alunos que vieram transferidos e, devido às equivalências, transitaram imediatamente para o segundo ano.

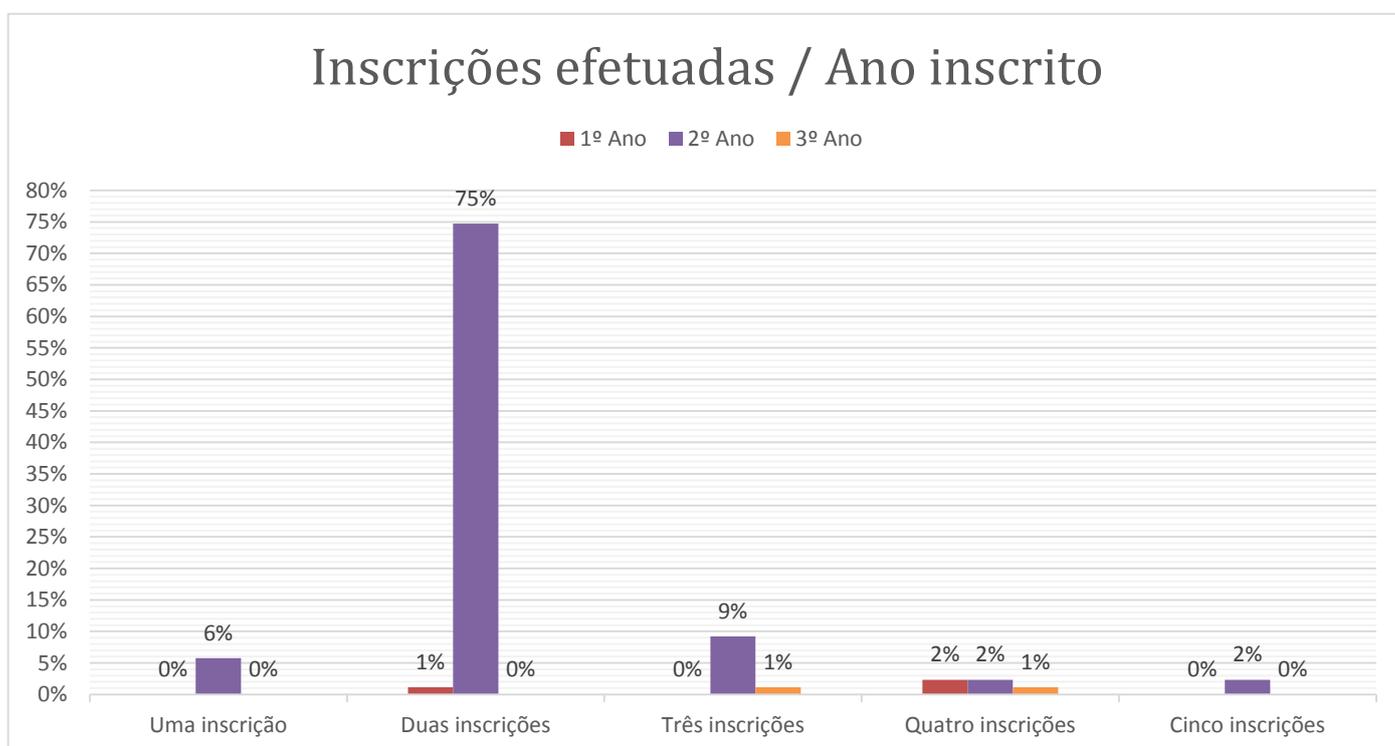


Figura 5 - Relação entre Inscrições efetuadas no curso e o ano atual

No gráfico da Figura 6, à semelhança do anterior pode-se visualizar a relação existente entre alunos com unidades curriculares em atraso e o ano em que atualmente se encontram. Após análise do mesmo, observa-se que existe uma distribuição constante dos alunos pelos números de unidades curriculares em atraso, o que não significa ser um sinal positivo. Normalmente o que se esperava era uma taxa elevada na zona das “Nenhuma UC”, significando, assim, que não possuía unidades curriculares em atraso e poderia investir a quantidade de tempo suposta normal para o sucesso do semestre. Pelo menos 55% dos alunos em estudo tiveram de investir mais tempo e esforço para concretizar, com o mesmo nível de sucesso, o semestre em causa.

Para terminar esta subsecção será apresentada uma relação entre as unidades curriculares em atraso e o número de atual de inscrições no curso, com objetivo de melhor identificar as percentagens de intervenientes que podem ter o seu desempenho influenciado por carga de trabalhos externa excessiva.

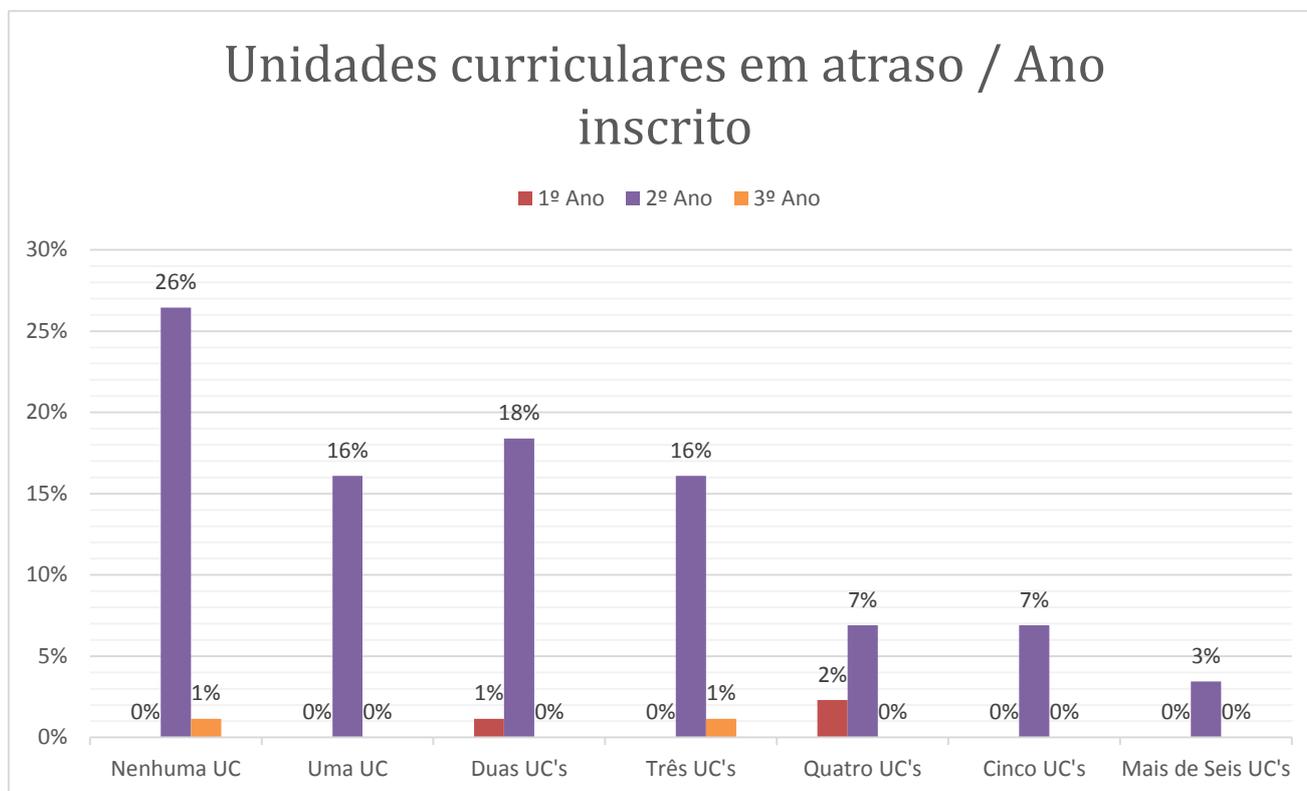


Figura 6 - Relação entre as unidades curriculares em atraso e o ano atual de inscrição

Na Figura 7 pode-se visualizar a relação entre as Unidades Curriculares em atraso e o número de inscrições efetuadas pelos alunos em estudo. Após análise ao gráfico, denota-se que, apesar do foco se encontrar nas duas inscrições, existe uma percentagem de 15,9% acima das duas inscrições, isto é, intervenientes/alunos que possuem pelo menos duas unidades curriculares em atraso. Apenas 2,2% dos alunos com duas inscrições possuem menos de uma unidade curricular em atraso, ou seja, possuem condições de trabalhar com desempenho normal, sem serem influenciados por cargas de trabalho extra. Avaliando agora a zona das duas inscrições, pode se visualizar usando a análise anterior que 39% dos alunos com duas inscrições possuem acima de duas unidades curriculares em atraso e, apenas, 36,8% dos alunos não possuem unidades curriculares em atraso.

Após análise conjunta dos três gráficos pode-se concluir globalmente que apenas 43,5% dos alunos em estudo têm menos de uma unidade curricular em atraso, ou seja, possuem qualificações para realizar o projeto com nível de esforço normal, sem necessitarem de investir

tempo extra no curso. Por outro lado, 56,5% dos alunos em estudo têm mais de duas unidades curriculares em atraso o que implica, como referido anteriormente, um aumento do tempo investido para obter sucesso a todas a unidades curriculares que atualmente possuem, sendo que esse investimento extra pode influenciar o desempenho no projeto de DAI.

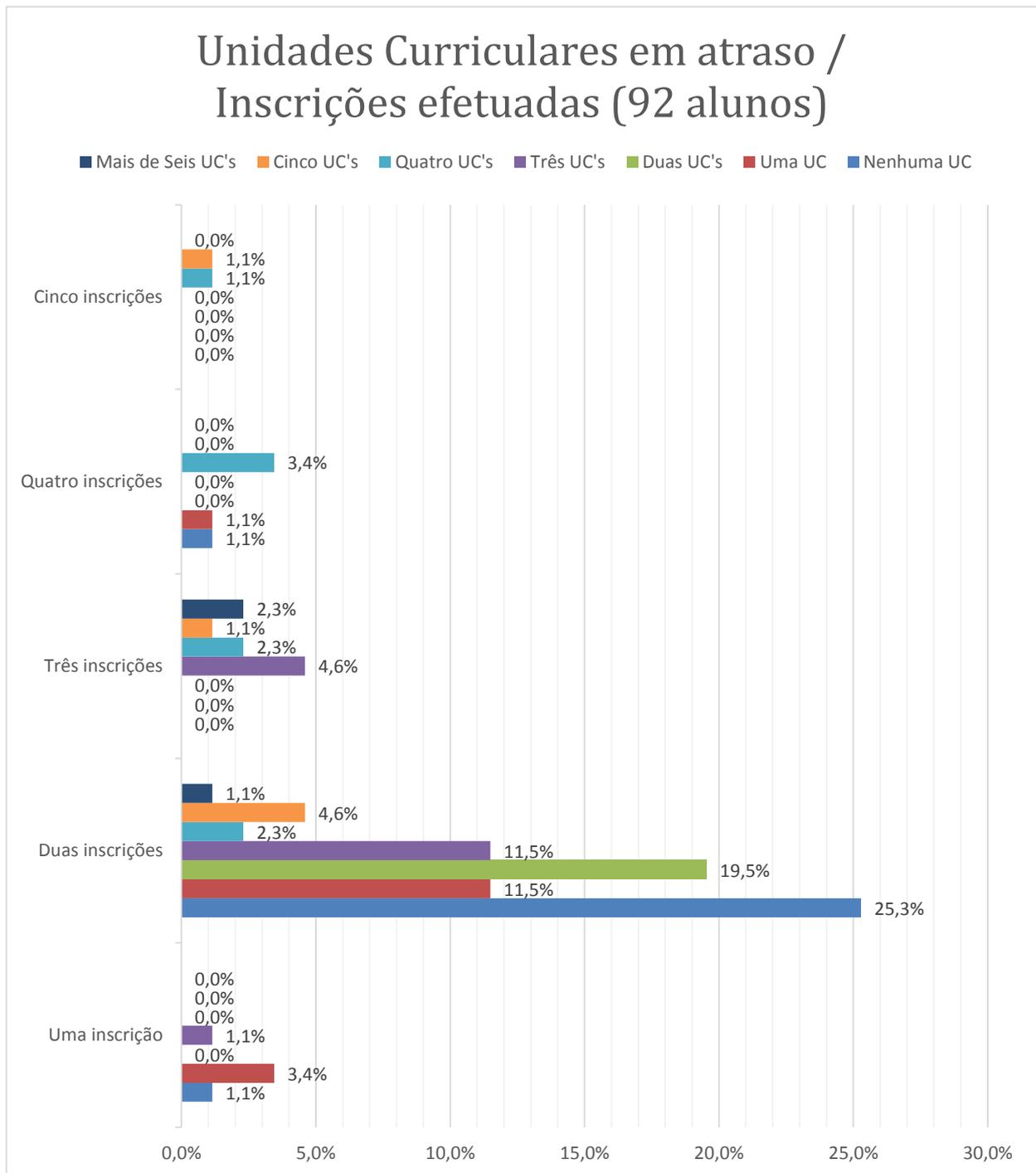


Figura 7 - Relação entre Unidades Curriculares em atraso e Inscrições efetuadas

Apesar destes dados não estarem diretamente ligadas com os papéis, ainda assim eles fornecem-nos informações relevantes que se devem ter em consideração para realizar um ajustamento dos papéis baseado no esforço que o papel teve durante o projeto.

### 4.1.3 Distribuição dos papéis pelos elementos das equipas

Nesta subsecção pretende-se comparar a forma como cada equipa escolheu o seu elenco processual, com o objetivo de no final analisar o esforço de cada grupo e compreender se existe a necessidade de reajustar algum papel ou se o grupo necessitava de mais intervenientes do mesmo tipo para dissipar o esforço extra realizado.

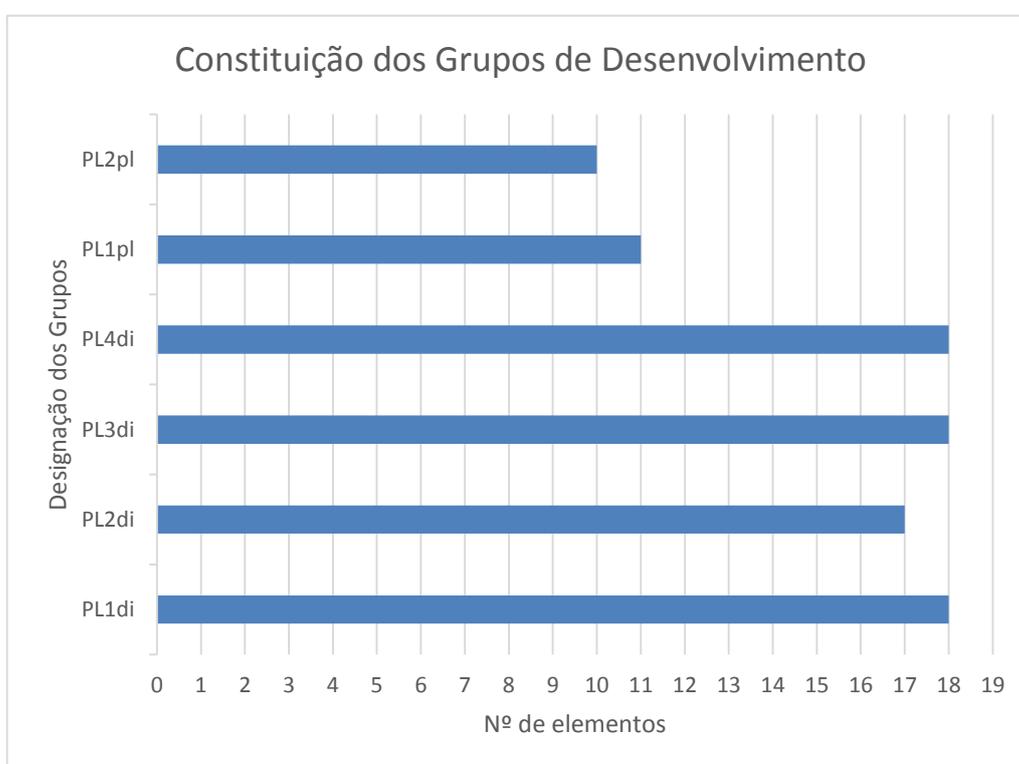


Figura 8 - Constituição dos Grupos de Desenvolvimento

O primeiro gráfico apresentado, Figura 8, mostra a forma como os grupos foram constituídos, podendo observar-se a existência de dois grupos com dez e onze elementos e quatro grupos com um maior número de elementos, isto é, entre dezassete e dezoito. Os primeiros dois grupos, cujos intervenientes pertencem ao turno do pós-laboral, por possuírem menor número de elementos, vão necessitar de atribuir vários papéis a um só interveniente, o que vai dificultar a realização de uma análise precisa aos gráficos. Por essa razão foi decidido desenvolver dois tipos de gráficos, um que demonstra o cargo principal de cada interveniente do grupo num rácio de um para um, ignorando os cargos extra atribuídos aos intervenientes, e outro o que mostra todos os

papéis atribuídos a um grupo, independentemente do número de intervenientes que esse mesmo grupo possui.

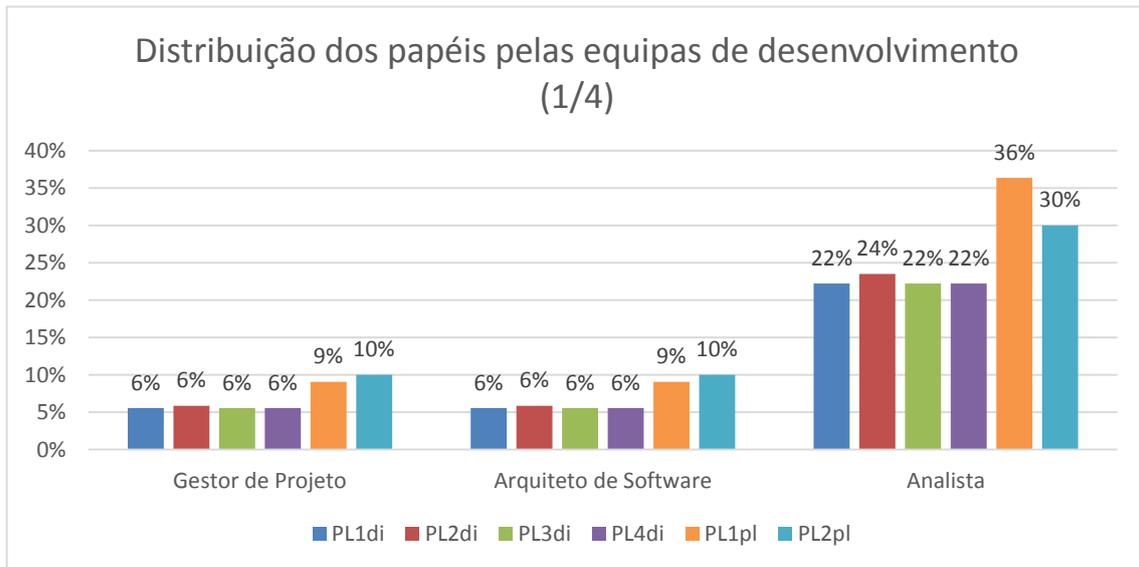


Figura 9 - Distribuição dos papéis pelas equipas de desenvolvimento (1/4)

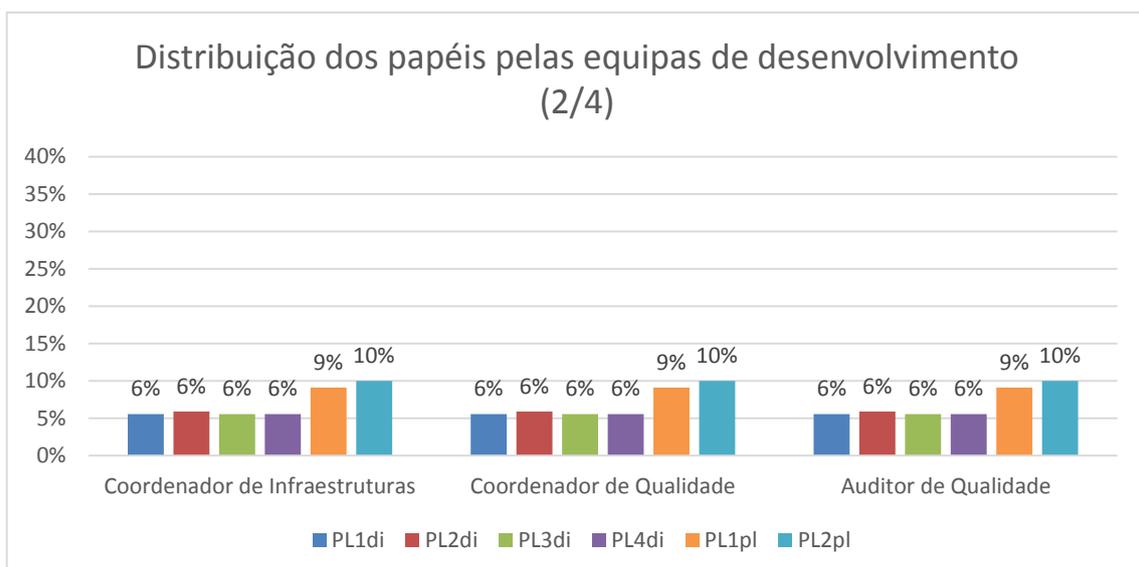


Figura 10 - Distribuição dos papéis pelas equipas de desenvolvimento (2/4)

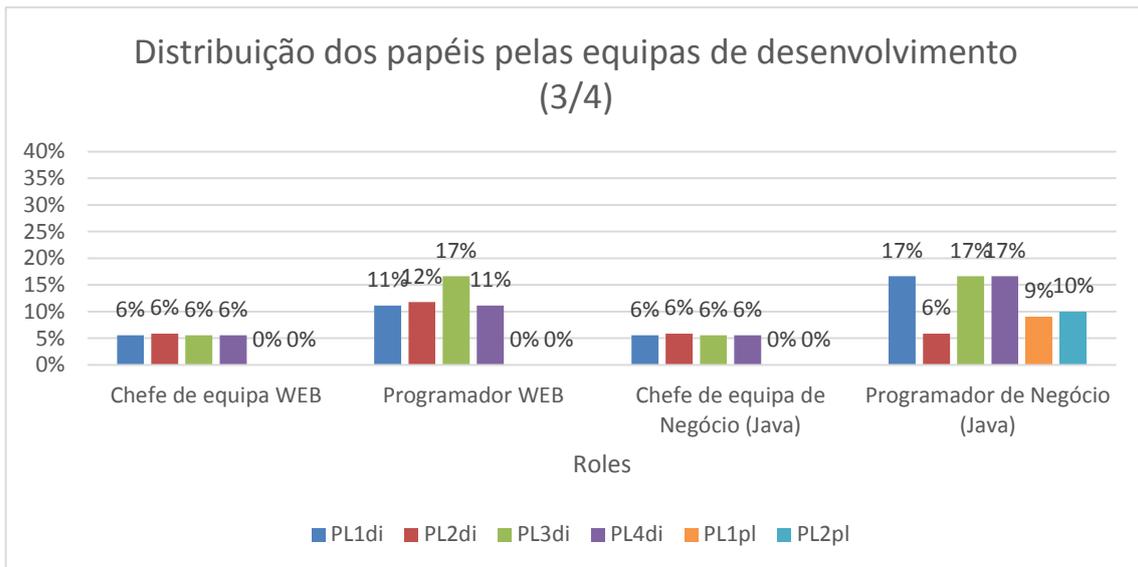


Figura 11 - Distribuição dos papéis pelas equipas de desenvolvimento (3/4)

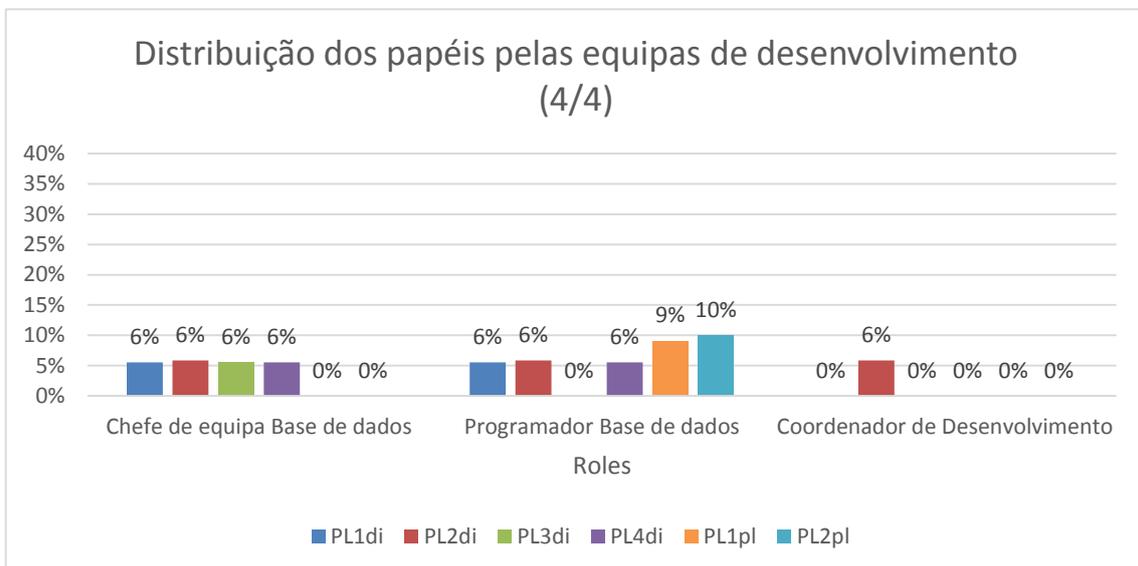


Figura 12 - Distribuição dos papéis pelas equipas de desenvolvimento (4/4)

As figuras representadas em cima, Figura 9, Figura 10, Figura 11 e Figura 12 devem ser analisadas como se fossem apenas uma, uma vez que a análise individual de papéis e a sua comparação entre equipas pode suscitar dúvidas e erros de análise. As percentagens de cada grupo atribuídas aos papéis devem ser lidas como a percentagem que a equipa X utilizou dos seus 100% para aquele papel. No caso da equipa PI2pl, apesar de investir 10% no Gestor de projeto não significa que tem mais gestores de projeto que as outras equipas, como já foi analisado na Figura 8, sabemos que esse grupo possui apenas dez elementos. Assim, conclui-se que apenas possui um Gestor de projeto, assim como os outros grupos. O objetivo destes gráficos não é comparar números de elementos mas sim o investimento do grupo nos papéis.

Retirando agora algumas conclusões relativamente à análise dos gráficos:

- Ambos os turnos de pós-laboral (PL1pl, PI2pl) decidiram não investir em intervenientes específicos para programação Web. O mais provável é que os intervenientes com os papéis de programador tenham de investir algum esforço na realização da implementação Web.
- Nenhum dos turnos de pós-laboral (PL1pl, PI2pl) possui chefes de equipa (*System Integrator*), o que sugere uma relação mais próxima entre o arquiteto de Software (*Software Architect*) e os programadores (*Implementers*).
- Existiu um maior investimento por parte dos grupos de pós-laboral (PL1pl, PI2pl) na componente de análise, analistas (*Analyst*), relativamente aos outros quatro grupos.
- O grupo PL2di foi o único grupo que investiu em intervenientes para o papel principal de Coordenador de Desenvolvimento (*Process Engineer*).
- Devido ao investimento realizado pelo grupo PL2di no Coordenador de Desenvolvimento (*Process Engineer*), consequentemente houve uma diminuição no esforço investido em programadores de negócio (*Implementers*).
- O grupo PI3di, apesar de possuir investimento em chefe de equipa de Base de dados, não investiu em programadores de base de dados, podendo-se inferir que este chefe de equipa ficou responsável por todos os cargos inerentes aos programadores de base de dados.
- Este grupo destaca-se, ainda, por ter feito um maior investimento que qualquer outro grupo em programadores WEB (*Web Implementers*).

Depois da análise feita aos gráficos anteriores, onde cada interveniente possuía apenas um papel pelo qual era responsável, apresentam-se, agora, os gráficos que relacionam o investimento real de cada grupo no desenvolvimento das funções referentes a cada papel e a respetiva análise. Salienta-se que a relação dos intervenientes com os papéis deixa de ser de um para um.

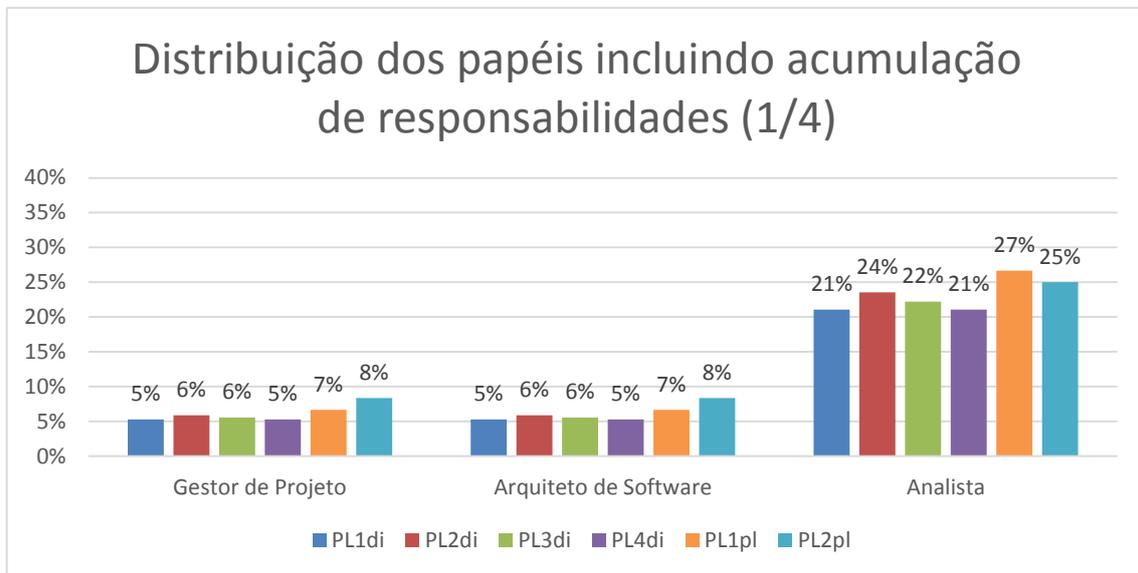


Figura 13 - Distribuição dos papéis incluindo acumulação de responsabilidades (1/4)

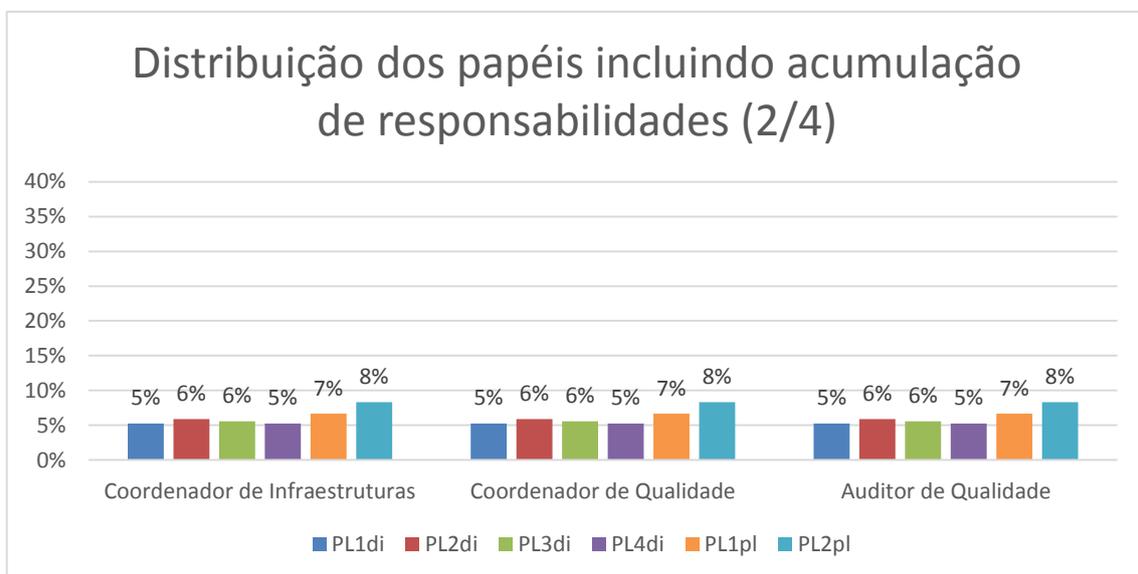


Figura 14 - Distribuição dos papéis incluindo acumulação de responsabilidades (2/4)

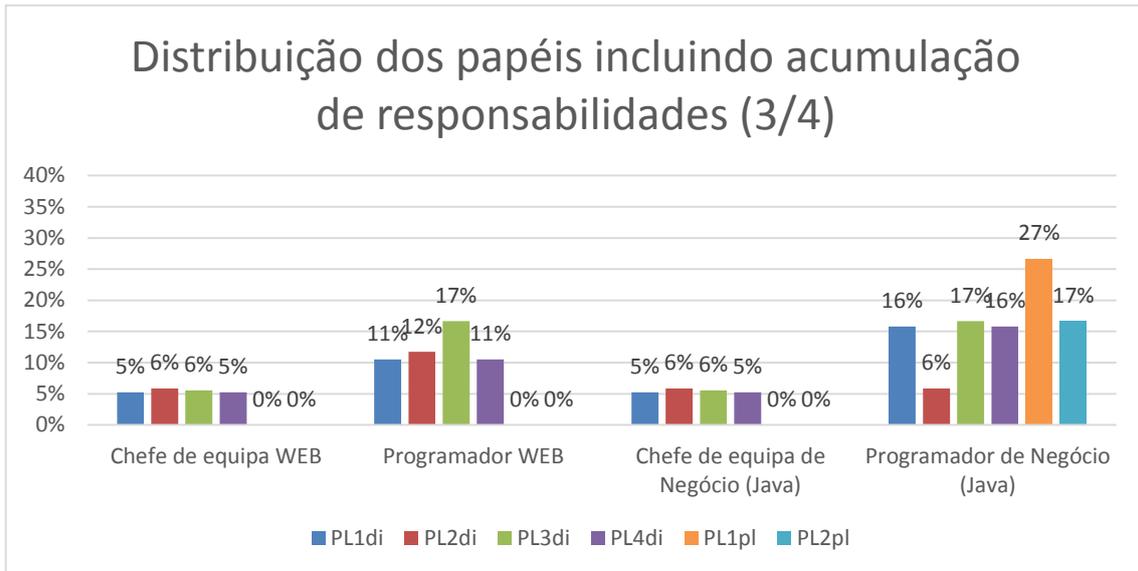


Figura 15 - Distribuição dos papéis incluindo acumulação de responsabilidades (3/4)

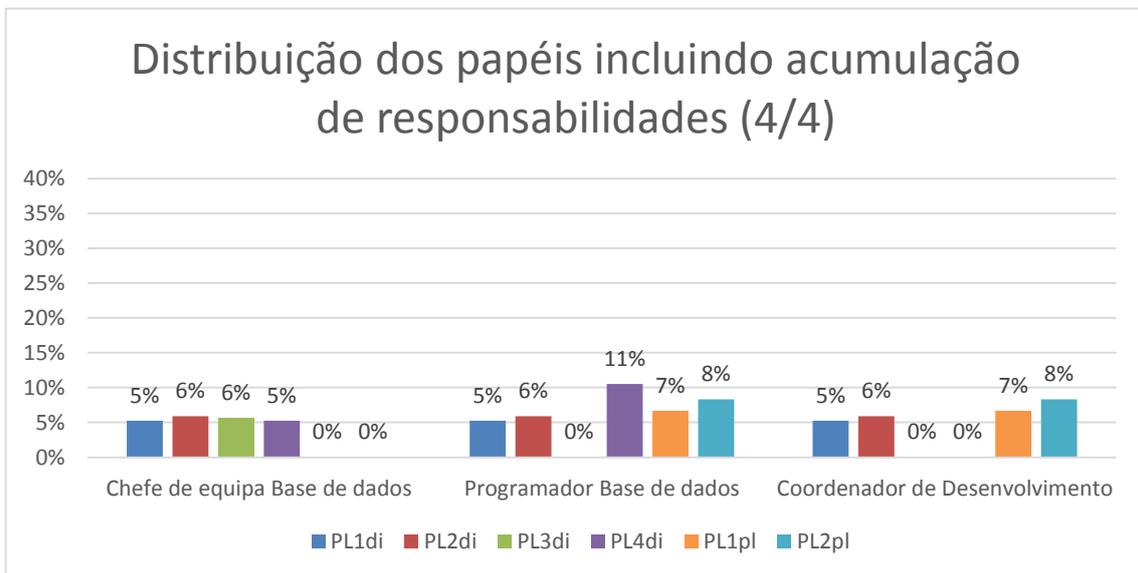


Figura 16 - Distribuição dos papéis incluindo acumulação de responsabilidades (4/4)

À semelhança do que foi feito nos gráficos anteriores, também os gráficos apresentados nas Figura 13, Figura 14, Figura 15 e Figura 16 devem ser analisados em simultâneo. Nestes gráficos a noção de interveniente deixa de existir, uma vez que cada interveniente pode ter um ou várias papéis. Com a análise destes gráficos consegue-se notar algumas diferenças, sobretudo nos dois turnos pós laboral onde o esforço é dispersado por vários papéis, uma vez que existem intervenientes, segundo os dados, com três papéis acumulados.

Retirando algumas conclusões adicionais depois da análise dos gráficos com acumulação de responsabilidades:

- Existem pelo menos três intervenientes com acumulação do papel de Coordenadores de Desenvolvimento (*Process Engineer*) e outros papéis.
- Existem Programadores de negócio (*Implementers*) do grupo PL1pl que possuem papéis acumulados com outros papéis, uma vez que o esforço investido nesse papel subiu de 9% para 27%.
- As duas equipas de pós- laboral (PL1pl e PL2pl) optaram por fazer um maior investimento em analistas (*Analysts*) que os outros grupos.
- Apesar da acumulação de cargos os grupos do Pós-laboral (PL1pl e PL2pl) decidiram não nomear nenhum interveniente para o papel de chefe de equipa (*System Integrator*).
- As equipas do turno diurno (PL1di, PL2di, PL3di e PL4di) tiveram um maior foco na parte de implementação, investimento de mais de 50% dos seus intervenientes nos papéis com essas responsabilidades. Os turnos do pós-laboral (PL1pl e PL2pl) fizeram um investimento superior de recursos na área de análise, tendo por isso, um investimento menor que 50% na área de implementação.

Da análise feita aos dados recolhidos pode-se, ainda, concluir que algumas equipas sentem mais necessidade de investir na modelação e análise do projeto, enquanto outros grupos preferem assegurar uma implementação segura e bem estruturada, aumentando o esforço individual a realizar pelos intervenientes na análise e modelação, mas diminuindo o esforço individual na fase de implementação.

#### 4.1.4 Análise do conhecimento sobre as competências e responsabilidades

Nesta subsecção será analisado o conhecimento dos intervenientes no início do projeto sobre as responsabilidades dos papéis do modelo do RUP reduzido, assim como perceber se algum dos intervenientes do projeto já teve oportunidade de experienciar alguma destas responsabilidades fora do contexto académico.

Analisando individualmente o gráfico da Figura 17 apenas se consegue concluir que, existe uma minoria dos intervenientes das equipas de desenvolvimento que já trabalhou no papel que lhe foi atribuído e, por essa razão, já possui conhecimento e experiência relativamente às responsabilidades e tarefas do mesmo. Adicionalmente, pode-se inferir que estes 13% de intervenientes possuem maior liberdade do desempenho das suas funções devido a experiência adquirida anteriormente.

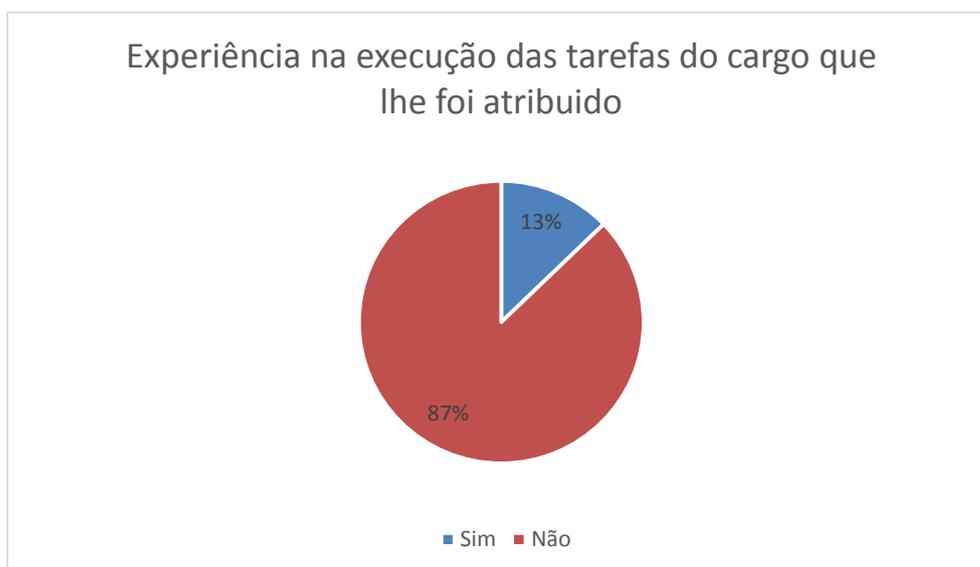


Figura 17 - Experiência externa na execução das responsabilidades do papel atribuído

Ao contrário do gráfico anterior, o gráfico da Figura 18 tenta demonstrar, na globalidade, a perceção dos intervenientes relativa ao papel que vão desempenhar no projeto, sendo que estes dados resultaram da pergunta: “Possui conhecimento das responsabilidades inerentes ao papel que lhe foi atribuído?”. Curiosamente, 99% dos intervenientes afirmam ter conhecimentos suficientes para desempenhar as funções do seu papel com sucesso. Um resultado surpreendente, tendo em conta que alguns intervenientes estão a ter contacto com alguns papéis pela primeira vez durante o percurso académico e apenas 13% deles teve experiências fora do contexto externo do meio académico.

No gráfico da Figura 19, pode ser analisado o nível de conhecimento atual dos intervenientes relativo as responsabilidades de todos os papéis pertencentes ao elenco processual do modelo Reduzido, sendo que a pergunta colocada para recolher estes dados foi: “Possui conhecimento das responsabilidades inerentes a cada papel definido no elenco processual do RUP?”. Sendo assim, pode-se observar que 90% dos intervenientes antes de se dar inicio a realização do projeto, já possuíam conhecimentos de todo o elenco processual. A análise deste gráfico resulta em dados ainda mais surpreendentes que o anterior. Pode-se constatar que grande parte dos papéis nunca foi desempenhado pelos intervenientes durante o percurso académico, sendo a única formação relacionada com os papéis recebida numa unidade curricular anterior a DAI, Processos e Metodologias de Software (PMS).



Figura 18 - Percepção das responsabilidades do papel que está a desempenhar



Figura 19 - Percepção das responsabilidades de todo o elenco processual do modelo

Procedendo, agora, à análise dos três gráficos em simultâneo pode-se concluir que apesar de apenas 13% dos intervenientes possuírem experiência a lidar com os papéis do modelo reduzido do RUP, somente 1% dos intervenientes necessitam de formação para que o desempenho do projeto não seja comprometido por falta de conhecimentos, sendo que o ideal seria que todos os intervenientes tivessem conhecimento de todas as responsabilidades de cada um dos papéis, elevando assim a percentagem que necessita de formação para 10%, de acordo com o gráfico da Figura 19.

#### 4.1.5 Sintetização e Conclusões

Terminada assim a caracterização e análise da maturidade dos intervenientes do projeto, podemos então retirar algumas conclusões relativas ao estado em que as equipas vão iniciar o projeto e a forma como o seu estado vai influenciar o desempenho do projeto.

Uma das conclusões mais relevantes é que mais de 55% dos intervenientes dos grupos poderão ter um desempenho inferior ao desempenho pretendido na realização do projeto, devido ao esforço extra que as unidades curriculares em atraso exigem.

Algumas das equipas possuem menos intervenientes que os elementos base necessários pelo modelo do RUP reduzido, resultando em diversas acumulações de papéis (*Software Architect* e *Implementer*), verificando-se, ainda, papéis que não entraram no elenco de algumas equipas de desenvolvimento (*System Integrator*).

Segundo as entrevistas realizadas, a falta de conhecimento relativamente às responsabilidades inerentes a cada papel não é um problema, uma vez que 99% dos intervenientes inquiridos afirmaram que possuem conhecimentos relativos as responsabilidades do cargo que lhes foi atribuído, pelo que julgam estar aptos para a realização das funções dos respetivos papéis.

Após a caracterização da maturidade das equipas de desenvolvimento procede-se, de seguida, à análise e estudo, sendo a secção seguinte, focada na análise e estudo do esforço investido pelas equipas de desenvolvimento no projeto.

## 4.2 Análise do tempo investido pelas equipas no desenvolvimento do projeto

Esta secção destina-se a expor e avaliar o esforço que as equipas de desenvolvimento investiram durante o decorrer do projeto. Por isso, todos os dados encontrados nesta secção foram recolhidos através de entrevistas às equipas e suportados pela documentação desenvolvida pelas mesmas.

Estas entrevistas foram realizadas no final das quinze semanas em que as equipas realizaram o projeto, com o objetivo de recolher o esforço individual investido em cada papel percebido pelos intervenientes do projeto.

### 4.2.1 Análises dos dados relativos ao esforço investido pelas equipas

Relativamente à análise dos dados, existem aspetos a ter em consideração, nomeadamente, que alguns elementos das equipas foram despedidos ao longo do projeto, sendo o seu esforço excluído da análise. Devido a escassez de respostas às entrevistas por parte da equipa do (PL4di), já que apenas um terço da equipa enviou o seu esforço, esta equipa foi excluída da análise de forma a não prejudicar a qualidade dos resultados. Os intervenientes que não possuem o esforço investido num determinado papel, passam a ter o esforço médio do papel como o seu esforço realizado (embora possa levar a uma pequena distorção dos dados).



Figura 20 – Constituição dos grupos de desenvolvimento no final do projeto

Antes de iniciarmos a análise dos esforços investidos pelas equipas de desenvolvimento é pertinente visualizar a constituição atual das equipas de trabalho. O gráfico representado pela Figura 20 tem como objetivo de elucidar sobre a constituição dos grupos no final do projeto. Devido

às razões apontadas anteriormente e ao “despedimentos” de alguns elementos das equipas, a constituição dos grupos sofreu ligeiras alterações comparativamente com o gráfico da Figura 8, que ilustra os dados os dados foram recolhidos no início do projeto. Reforça-se que a equipa PL4di foi excluída dos dados, passando a ser zero o número de alunos que constituem essa equipa.

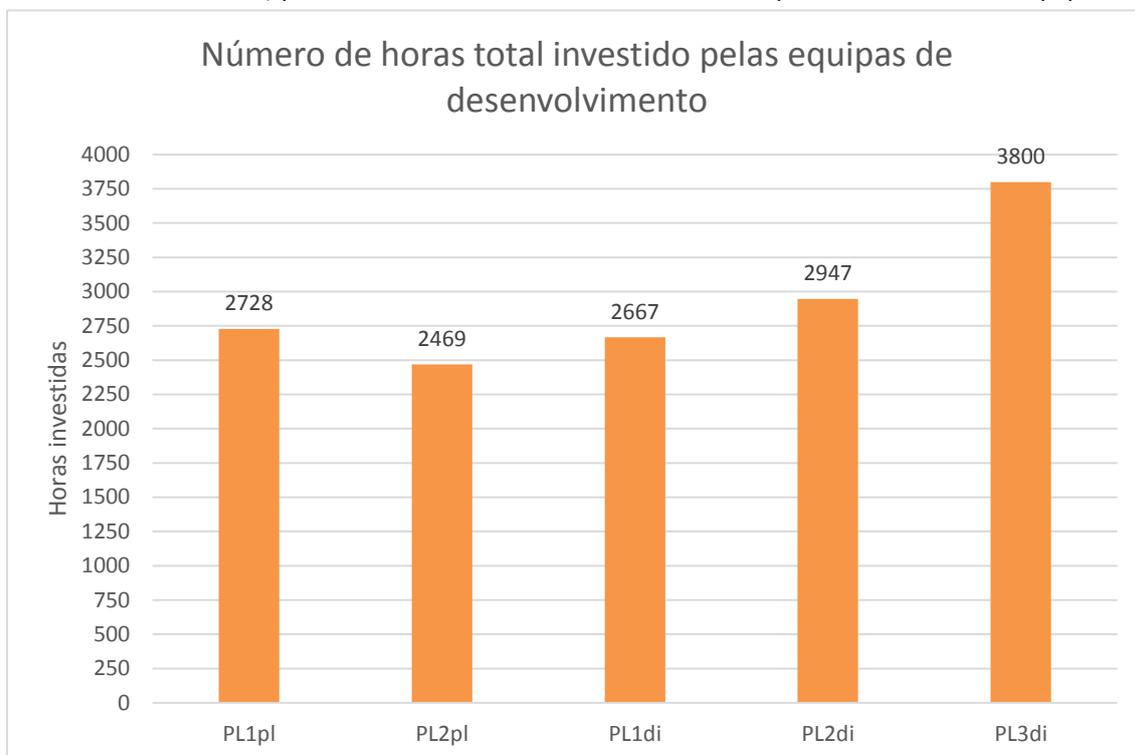


Figura 21 - Número total de horas investido pelas equipas de desenvolvimento

O número de elementos que foram “despedidos” das equipas, ou que não responderam ao questionário foram:

- Zero na equipa do PL1pl
- Zero na equipa do PL2pl
- Seis na equipa do PL1di
- Três na equipa do PL2di
- Um na equipa do PL3di
- Onze na equipa do PL4di

Começamos então por analisar o esforço global investido por cada equipa de desenvolvimento.

O gráfico da Figura 21 apresenta o esforço total investido pelas equipas de desenvolvimento, mostra-nos que, todas as equipas ultrapassaram as 2400 horas de trabalho,

sendo que o grupo PL3di chegou mesmo às 3800 horas, sendo isto devido a constituição da sua equipa de desenvolvimento.

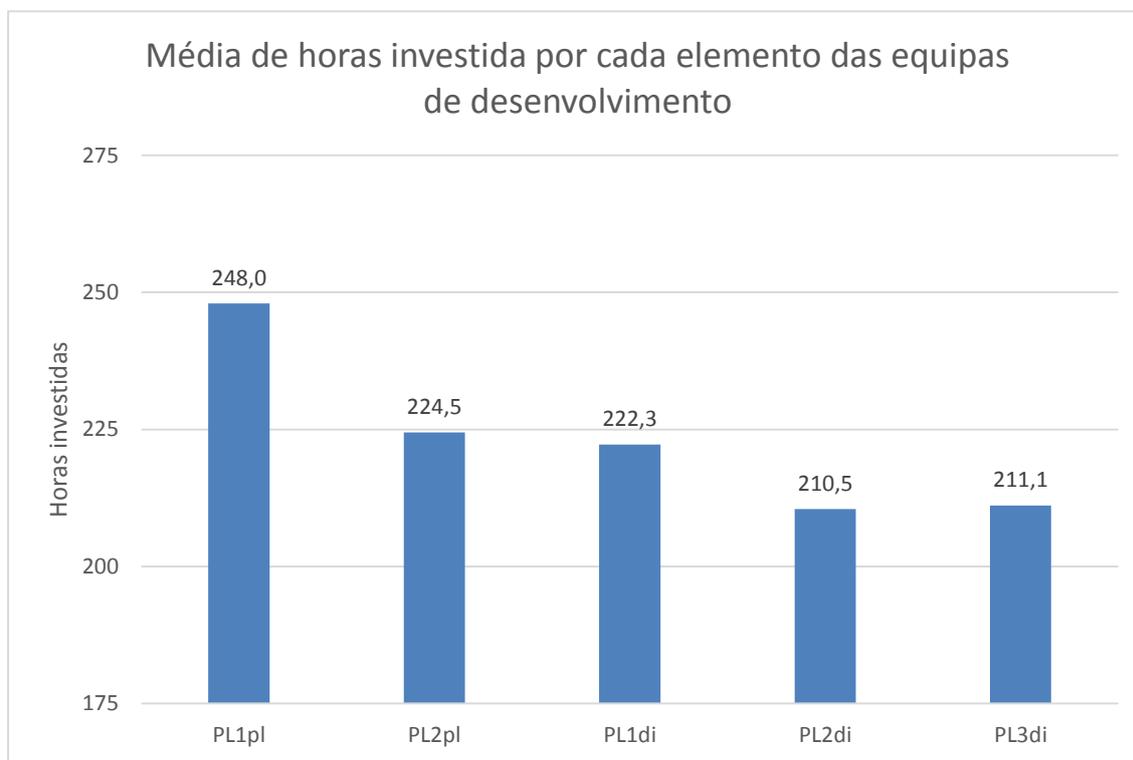


Figura 22 - Média de horas investida por cada elemento das equipas de desenvolvimento

Apesar do Grupo PL3di ter mais horas investidas que as outras equipas, quando visualizamos o esforço médio individual de cada elemento das equipas, observa-se que o Grupo PL3di não foi a equipa que teve maior esforço. No gráfico da Figura 21 está retratada a média de horas que cada elemento de uma equipa trabalhou, diferente do gráfico da Figura 22, pode-se observar que um aluno do grupo PL2pl trabalhou em média mais horas que um aluno do grupo PL3di. Isto acontece devido à diferença de número de intervenientes dos grupos existente.

Apesar disso, observa-se que em média todos os alunos do grupo trabalharam aproximadamente o número de horas/ECTS estabelecido pela universidade do Minho, sendo este 190 horas (Universidade do Minho, 2016). Apesar do esforço oficial ser de 280 horas, 190 horas são horas não presenciais e 90 horas são horas presenciais (aula Teóricas, Teórico/Práticas e Práticas). Adicionalmente partiu-se do princípio que os alunos utilizavam na sua totalidade as 90 horas presenciais.

É, ainda, pertinente analisar o esforço investido pelos grupos no desenvolvimento dos projetos, durante os períodos de trabalho que precedem os *milestones* de avaliação (Momento 1, Momento 2, Momento 3, Momento 5). Adicionalmente, os períodos de trabalho que precedem os milestones de avaliação possuem a seguinte duração.

- Momento 1 - Duas semanas úteis de trabalho;
- Momento 2 - Três semanas úteis de trabalho;
- Momento 3 - Seis semanas úteis de trabalho;
- Momento 5 - Quatro semanas úteis de trabalho.

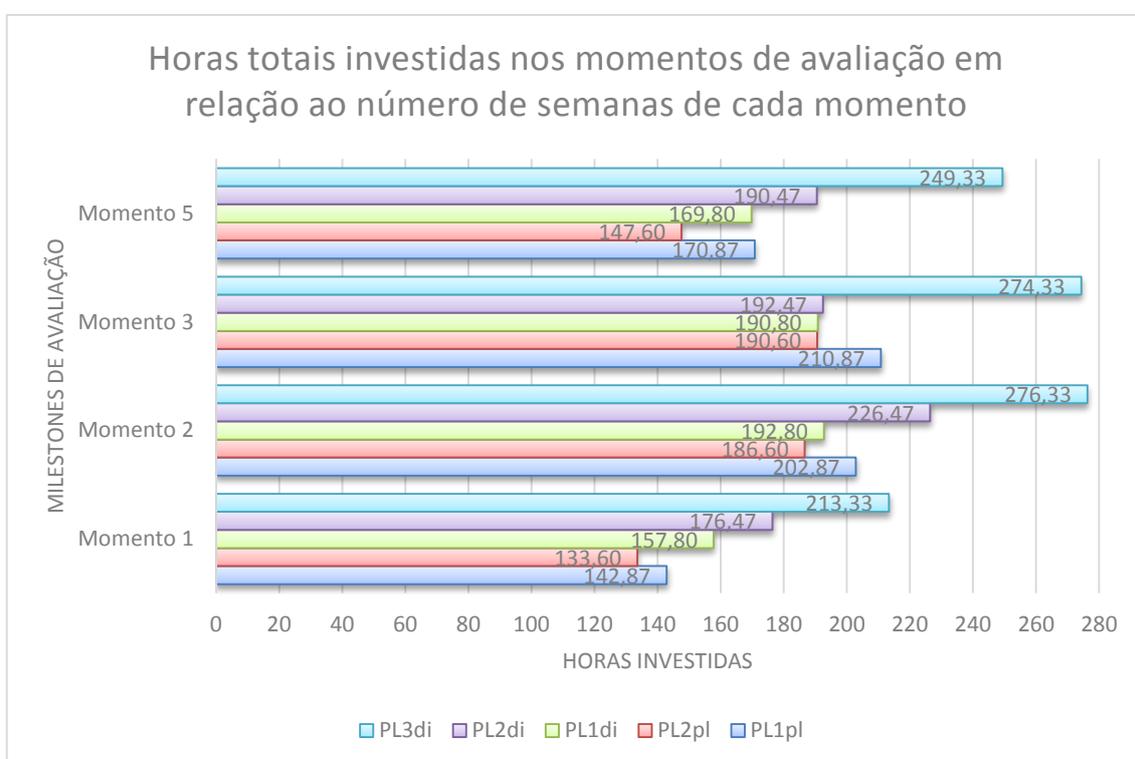


Figura 23 - Horas totais investidas nos momentos de avaliação em relação ao número de semanas

O gráfico da Figura 23 mostra a distribuição total de horas investidas por cada equipa durante os respectivos Momentos de avaliação, tendo em conta o número de semanas de trabalho a que cada Momento de avaliação está associado.

Pela análise do gráfico, pode concluir-se que as equipas tiveram um esforço diretamente proporcional ao número de semanas de cada *milestone*. Em semelhança aos gráficos anteriores podemos identificar na Figura 24, um gráfico que apresenta o esforço médio investido por cada elemento das equipas de desenvolvimento, nos diferentes momentos de avaliação. Neste

gráfico pode, ainda, ser visualizado uma comparação entre o esforço médio que um aluno deve investir de acordo com as normas (ECTS/Horas) da Universidade do Minho e o esforço médio que um aluno da unidade curricular investiu (Média horas das equipas).

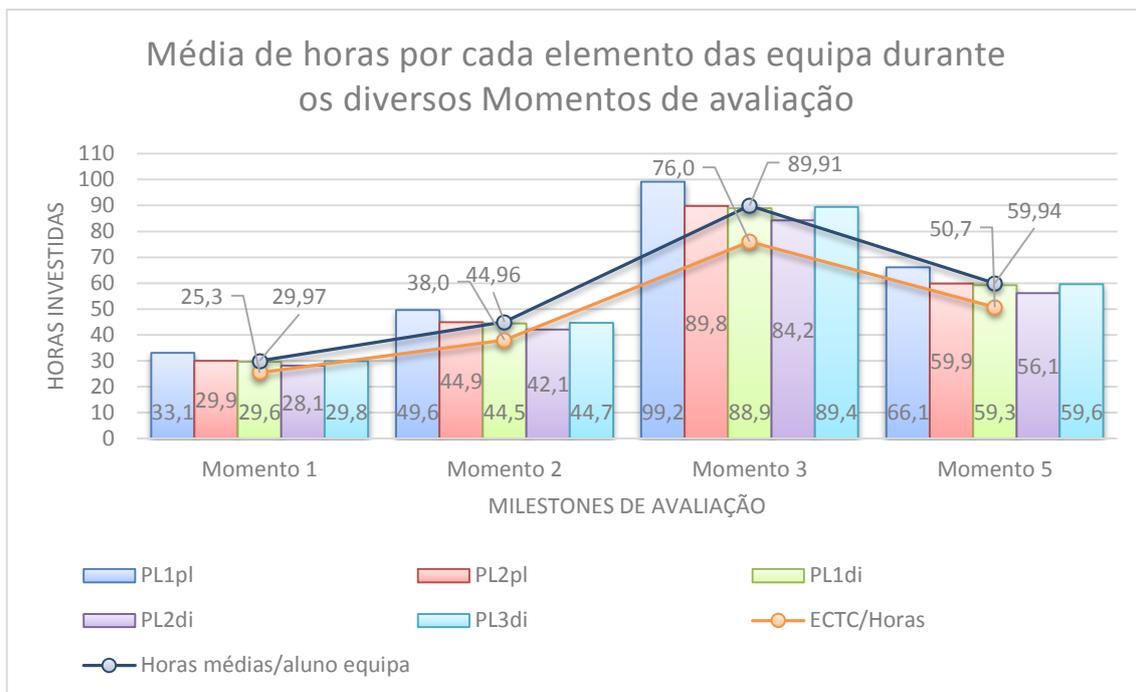


Figura 24 - Média de horas de cada elemento das equipas durante os diversos Momentos de avaliação

Contrariamente ao que pode ser interpretado pela leitura do gráfico da Figura 24, pode-se visualizar, pelo gráfico da Figura 22, que a equipa PL2pl, apesar de possuir o menor esforço total investido, foi das equipas em que cada interveniente investiu mais horas de trabalho. Após análise do gráfico da Figura 24, pode-se concluir que, em média, todos os elementos das equipas investiram, ligeiramente, mais horas do que as horas estipuladas pelas normas da Universidade do Minho.

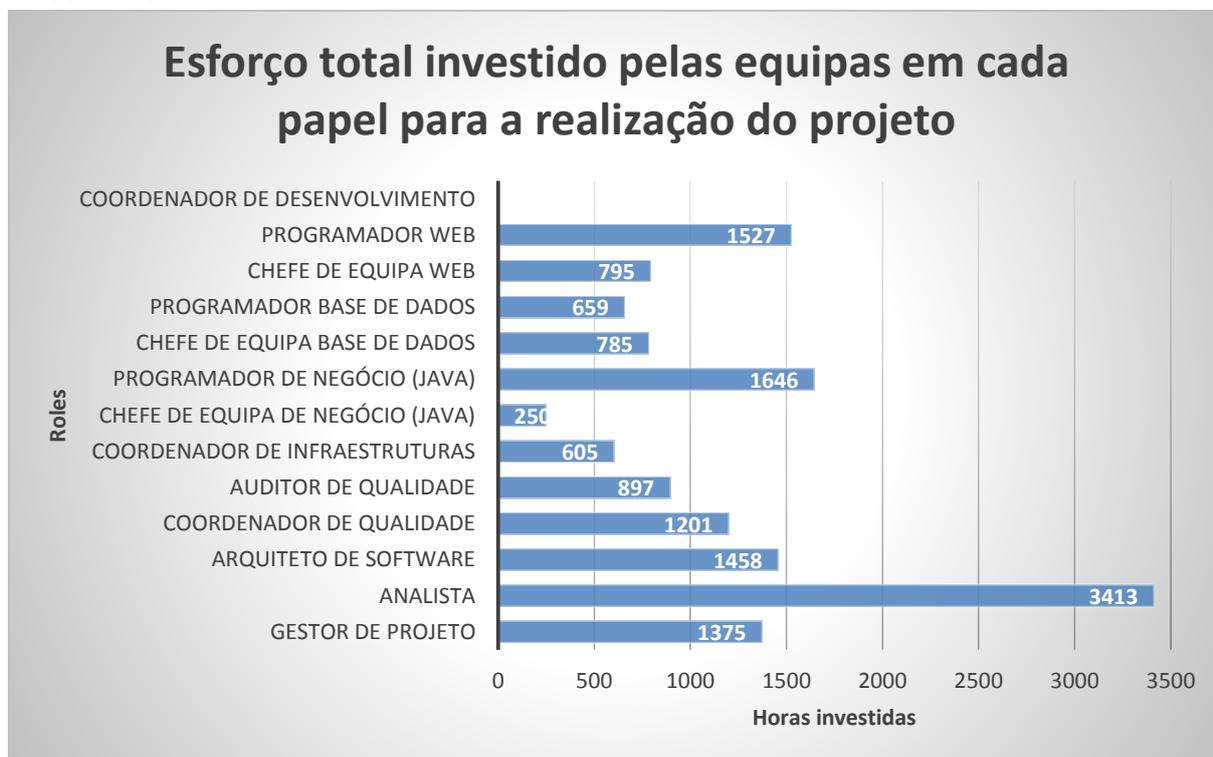


Figura 25 - Esforço total investido pelas equipas em cada papel para a realização do projeto

Uma vez analisado o esforço individual das equipas de desenvolvimento, resta-nos analisar o esforço investido pelos intervenientes em cada papel no desenvolvimento do projeto.

O gráfico da Figura 25 apresenta o esforço total investido pelas equipas de desenvolvimento em determinados papéis. Propositadamente, foi introduzido o papel de Coordenador de Desenvolvimento (*Process Engineer*), apesar do número de horas investido ser zero, de forma a informar que nenhum interveniente registou tempo investido nesse papel. Pode observar-se que existe um grande investimento em horas no papel de analista (*Analyst*), pelo número de intervenientes que desempenharam esse cargo.

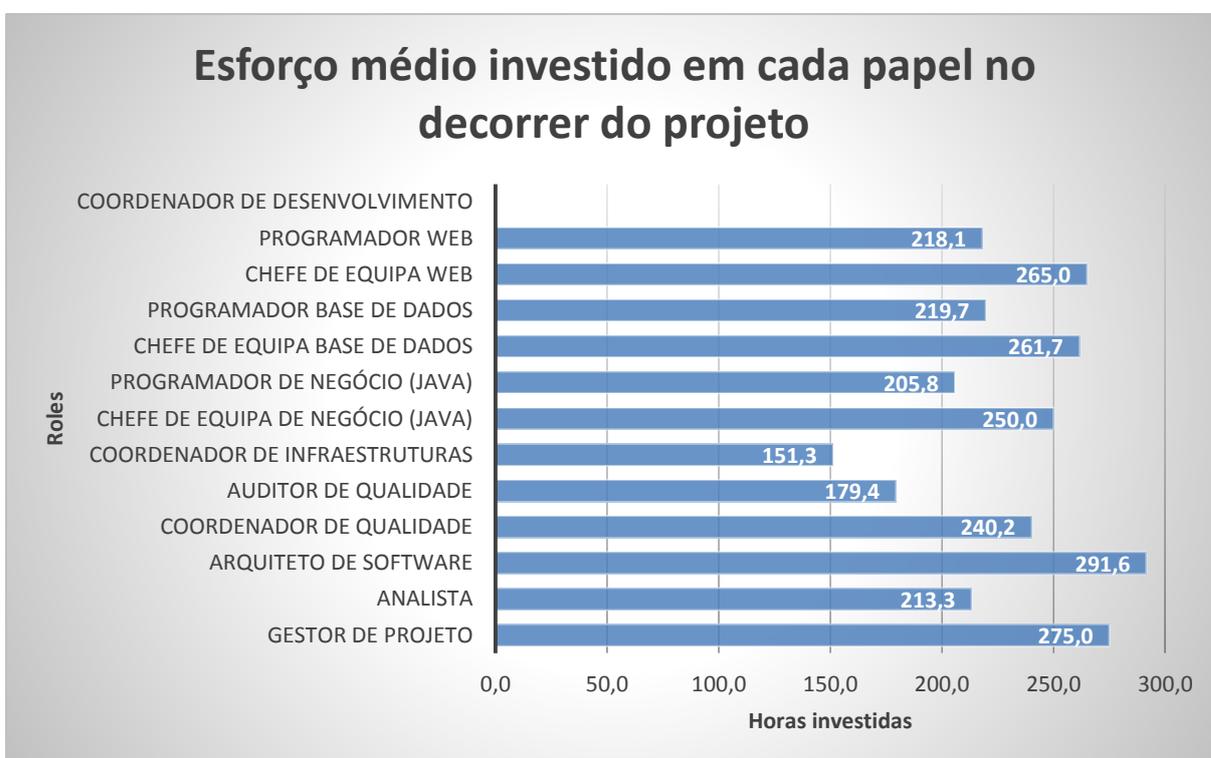


Figura 26 - Esforço médio investido em cada papel no decorrer do projeto

O gráfico da Figura 26 informa-nos do esforço médio de cada papel no decorrer do projeto. Através deste é possível observar que os analistas (*Analyst*), apesar de possuírem o maior esforço total investido, sendo esse esforço total investido por diversos intervenientes, resultam num esforço médio individual menor. Por outro lado os arquitetos de Software (*Software Architect*) possui um esforço muito superior, uma vez que este papel se encontra apenas distribuído por seis intervenientes, um em cada equipa de desenvolvimento, e as responsabilidades deste papel estão, em algumas equipas, agrupadas com as responsabilidades do chefe de equipa (*System Integrator*).

Antes de proceder a conclusões finais de análise é ainda pertinente análise um último gráfico. Os gráficos representados pelas Figuras: Figura 27, Figura 28, Figura 29 e Figura 30 demonstram a relação entre o esforço médio investido nos papéis, em diferentes momentos de avaliação, pelos intervenientes com o esforço médio que um aluno deveria ter segundo as normas definidas pela Universidade do Minho e o esforço médio de todos os alunos da unidade curricular.

De forma a fornecer uma perspetiva global, encontra-se no Anexo C, Figura 32, um gráfico que agrupa todos os momentos de avaliação, mas, como a sua visibilidade é limitada, optou-se por realizar uma análise dos gráficos de forma individual.

Relativamente a estes gráficos pode-se concluir que o papel de auditor de qualidade (*Test Manager*) e o coordenador de infraestruturas (*System Administrator*) possuíram um esforço inferior ao esforço médio estabelecido, à exceção do Momento 5, onde o coordenador de infraestruturas (*System Administrator*) teve um próximo do suposto.

O arquiteto de Software (*Software Architect*), apesar de ter tido um esforço normal durante o Momento 1, deve ter sentido necessidade de investir mais esforço durante o Momento 2 para que o projeto decorresse com normalidade, sendo que no Momento 3 possuía um esforço médio de quase 50 horas superior ao tempo médio normal estipulado.

Observa-se um aumento contínuo do esforço dos programadores (*Implementers*) no decorrer dos momentos, sendo que, no Momento 1 o seu esforço em média era inferior ao esforço esperado, mas durante os outros Momentos ultrapassou o esforço médio normal. Destaca-se, ainda, que o gestor de projeto (*Project Manager*) manteve o um esforço constante do Momento 3 para o Momento 5.

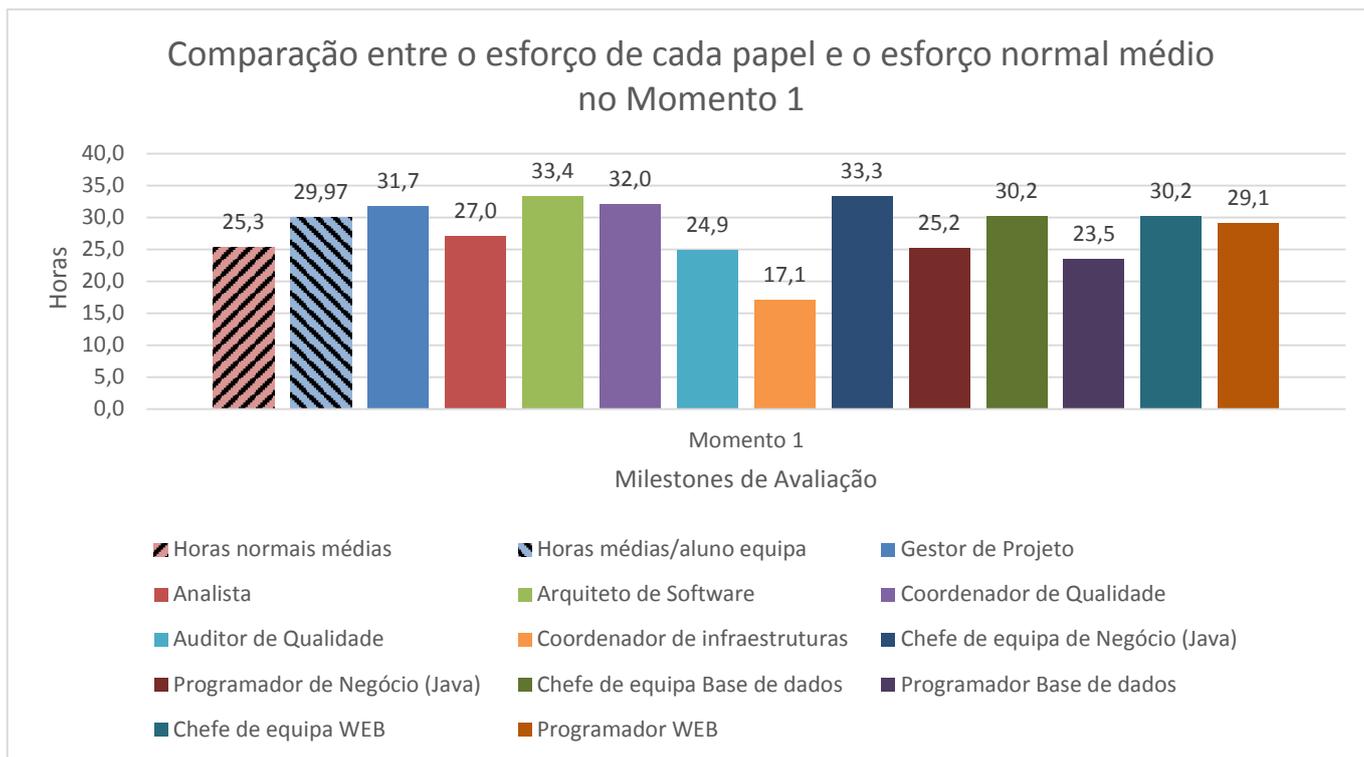


Figura 27 - Comparação entre o esforço de cada papel e o esforço normal médio no Momento 1

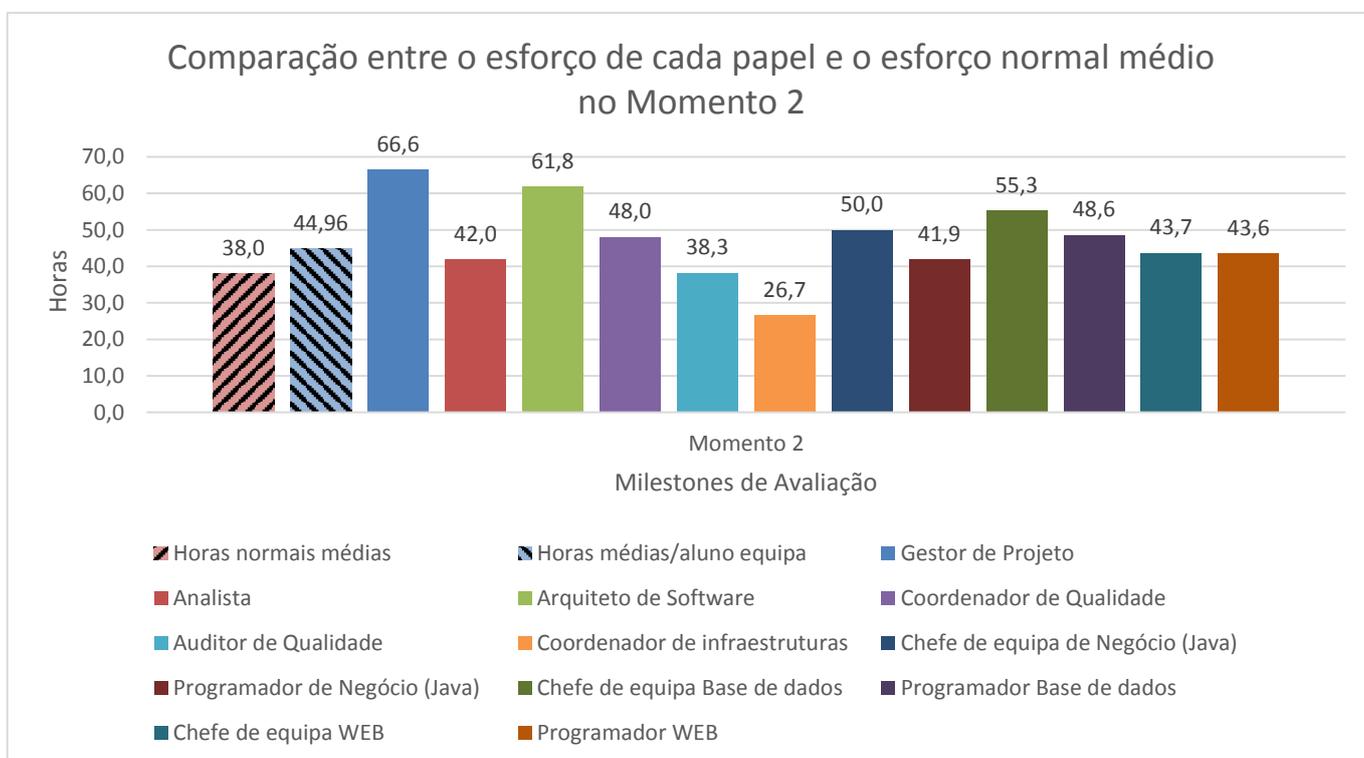


Figura 28 - Comparação entre o esforço de cada papel e o esforço normal médio no Momento 2

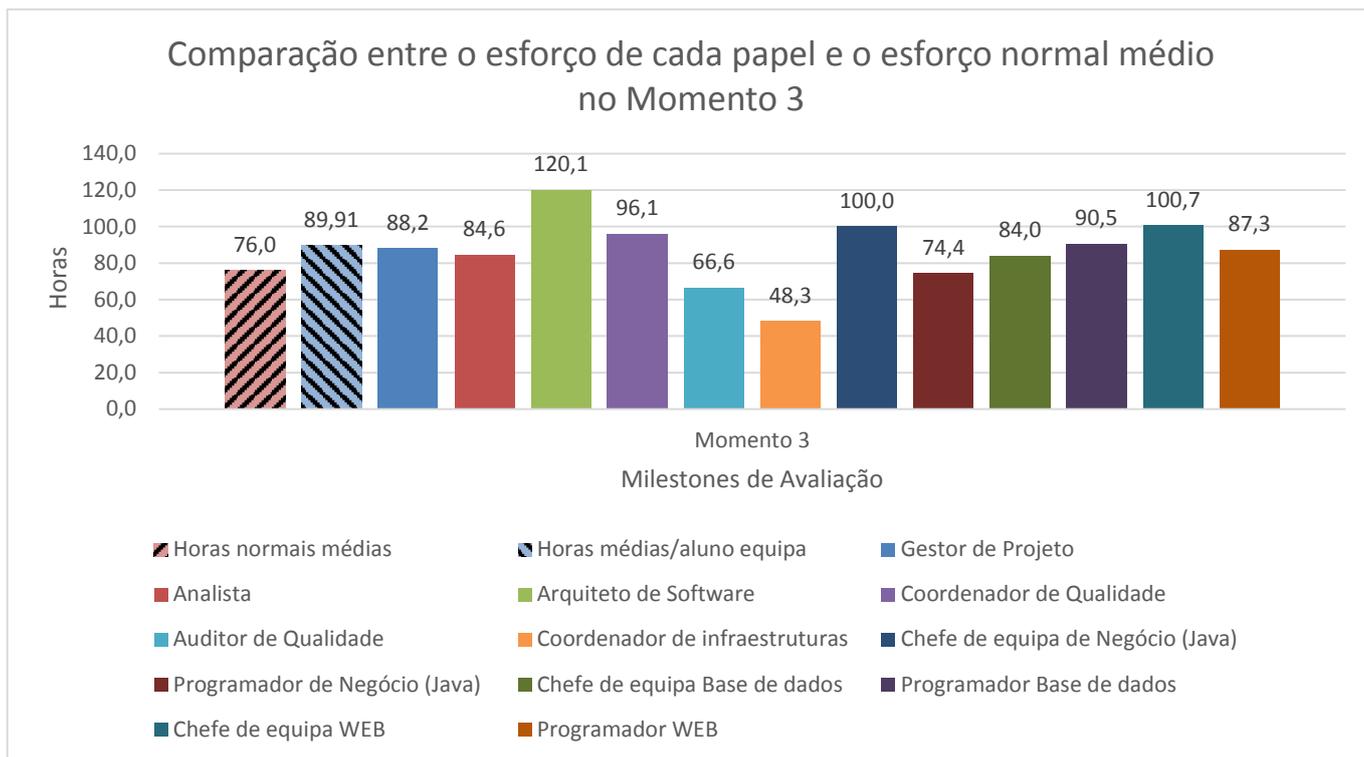


Figura 29 - Comparação entre o esforço de cada papel e o esforço normal médio no Momento 3

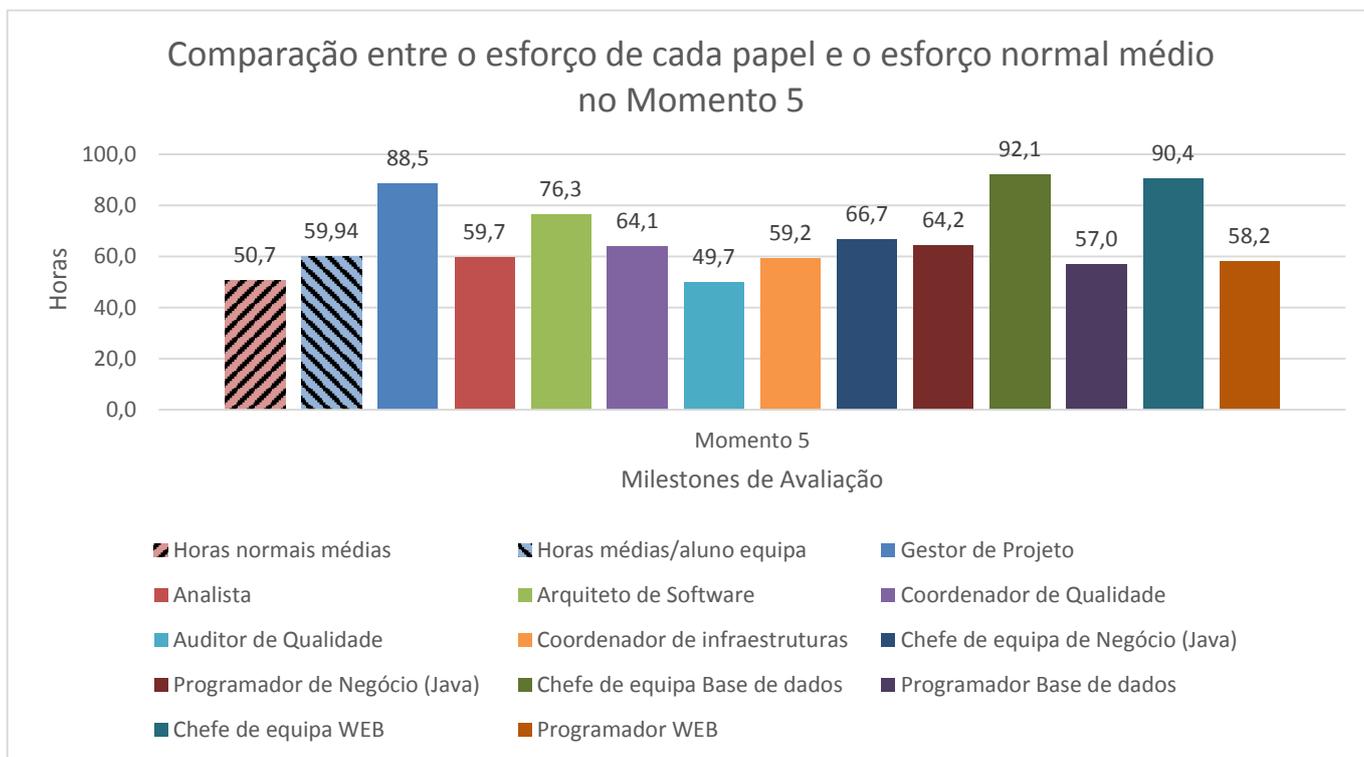


Figura 30 - Comparação entre o esforço de cada papel e o esforço normal médio no Momento 5

#### 4.2.2 Conclusões e sugestões de reajustamento dos Papeis do modelo reduzido do RUP

Esta subsecção aborda o propósito desta dissertação, ou seja, a realização dos ajustamentos que possam ser necessários fazer aos papéis do modelo reduzido do RUP, tendo em consideração todos os dados, análise e conclusões retiradas das secções produzidas anteriormente.

Após a análise de todos os gráficos e dados recolhidos das entrevistas e suportados pela documentação produzida pelos alunos foram retiradas algumas conclusões, entre todas as análises realizadas, existiram algumas que apenas nos permitiram enquadrar os dados e retirar conclusões relacionadas com as equipas de desenvolvimento, enquanto outros nos permitiram por em causa a distribuição do esforço investido pelos papéis no modelo Reduzido atualmente existente. Das evidências que afetaram significativamente o desempenho do projeto apresentam-se de seguida:

- O esforço do arquiteto de Software (*Software Architect*) manteve-se em todos os momentos do projeto acima do pretendido, chegando em certas alturas a subir para proporções muito elevadas.
- A ausência de um coordenador de infraestruturas (*System Administrator*) em algumas fases do desenrolar do projeto.
- A falta de presença no momento inicial do auditor de qualidade (*System Tester*).
- O analista (*System Analyst*), apesar de não pertencer ao modelo Reduzido foi utilizado pelas equipas de desenvolvimento com um esforço investido acentuado.

Dos tópicos referidos acima é de ter em consideração que 56,5% dos alunos a investir mais esforço poderia implicar uma modificação no desempenho dos mesmos, mas como a avaliação dos projetos não fez parte das métricas em estudo, não foi possível mostrar quanto as unidades em atraso afetaram o sucesso do projeto. Por outro lado pode especular-se que, uma vez diminuído o desempenho dos mesmos, aumentou o número de horas necessárias para cumprirem as suas responsabilidades, sendo esta a causa da maioria dos papéis possuírem valores ligeiramente superiores às normas definidas pela Universidade do Minho.

	PL1pl	PL2pl	PL1di	PL2di	PL3di
Gestor de Projeto	1	1	1	1	1
Analista	4	3	4	3	2
Arquiteto de Software	1	1	1	1	1
Coordenador de Qualidade	1	1	1	1	1
Auditor de Qualidade	1	1	1	1	1
Coordenador de infraestruturas	0	1	1	1	1
Chefe de equipa de Negócio (Java)	0	0	0	0	1
Programador de Negócio (Java)	3	1	0	1	3
Chefe de equipa Base de dados	0	0	1	1	1
Programador Base de dados	0	1	0	1	1
Chefe de equipa WEB	0	0	1	1	1
Programador WEB	0	1	1	2	3
Coordenador de Desenvolvimento	0	0	0	0	0

Tabela 12 - Número de intervenientes responsáveis por cada papel nos diferentes grupos de desenvolvimento

Antes de prosseguir para os outros três pontos que devem ser abordados, foquemo-nos agora na Tabela 12 onde se pode visualizar a forma como as equipas se distribuíram no projeto.

O PL1pl e o PL2pl, devido à falta de recursos humanos, tiveram uma distribuição idêntica de responsabilidades, não tendo sido atribuídos, em nenhuma das duas equipas, os cargos de chefe de equipa (*System Integrator*), nem de coordenador de desenvolvimento (*Process Engineer*), levou à necessidade de outros papéis cobrirem as responsabilidades inerentes aos cargos anteriormente referidos.

Todas estas responsabilidades foram adicionadas às responsabilidades do arquiteto de Software (*Software Architect*), contribuindo assim para um esforço acrescido resultante da acumulação de responsabilidades, como mostra na Tabela 13. As outras três equipas apesar de possuírem chefes de equipa, não possuem coordenador de desenvolvimento (*Process Engineer*), notando-se assim que nos grupos PL1di e PL2di o esforço dos arquitetos de Software (*Software Architect*) foi bastante superior ao normal, enquanto no grupo PL3di o esforço do gestor de projeto (*Project Manager*) foi notoriamente superior aos outros.

	PL1pl	PL2pl	PL1di	PL2di	PL3di
Gestor de Projeto	230,0	260,0	265,0	250,0	370,0
Analista	222,5	208,3	205,8	195,0	245,0
Arquiteto de Software	412,0	305,0	285,0	256,0	200,0
Coordenador de Qualidade	270,0	290,0	211,0	230,0	200,0
Auditor de Qualidade	190,0	167,0	210,0	180,0	150,0
Coordenador de infraestruturas	0,0	112,0	170,0	183,0	140,0
Chefe de equipa de Negócio (Java)	0,0	0,0	0,0	0,0	250,0
Programador de Negócio (Java)	245,3	240,0	0,0	205,0	155,0
Chefe de equipa Base de dados	0,0	0,0	240,0	215,0	330,0
Programador Base de dados	0,0	220,0	0,0	234,0	205,0
Chefe de equipa WEB	0,0	0,0	250,0	205,0	340,0
Programador WEB	0,0	250,0	213,0	202,0	220,0
Coordenador de Desenvolvimento	0,0	0,0	0,0	0,0	0,0

Tabela 13 - Horas médias investidas pelas equipas em cada papel

Concluindo-se então, que existe a necessidade de um interveniente para realizar as responsabilidades de coordenador de desenvolvimento (*Process Engineer*), sendo a outra opção a existência de um coordenador de desenvolvimento e um segundo interveniente com o papel de arquitetos de Software (*Software Architect*) de forma a dispersar as responsabilidades inerentes a esse papel.

Outra preocupação é o facto de auditor do qualidade e do coordenador de infraestruturas não possuírem um esforço aceitável durante todos os momentos do projeto. Uma vez que as equipas de desenvolvimento são equipas pequenas, nas quais provavelmente apenas existe um ou dois chefes de equipa (*System Integrator*) e um arquiteto de Software (*Software Architect*), as responsabilidades de elo de ligação entre gestor de projeto (*Project Manager*) e os papéis referidos anteriormente é mais simplificada. Sugere-se então, que as responsabilidades do coordenador de infraestruturas (*System Administrator*) sejam transferidas para o coordenador de desenvolvimento (*Process Engineer*), ficando este papel sem efeito no modelo Reduzido do RUP.

Relativamente à falta de esforço do auditor de qualidade (*System Tester*) em alguns momentos do projeto e do esforço acima da média do coordenador de qualidade (*Test Manager*), sugere-se que esses dois papéis agrupem as suas responsabilidades, denominado de “Gestor de Qualidade” ficando pelo menos dois intervenientes responsáveis pelo mesmo papel. Esta alteração ajudaria a aumentar ligeiramente o esforço investido pelo interveniente que desempenhou o papel de auditor de qualidade, assim como diminuir o esforço do interveniente que desempenhou o papel de coordenador de qualidade, que está ligeiramente acima da média. Desta forma o papel “Qualidade” (*Test Manager* e *System Tester*) passa a possuir todas as responsabilidades identificadas na Secção 3.5 do “*System Tester*” e do “*Test Manager*”.

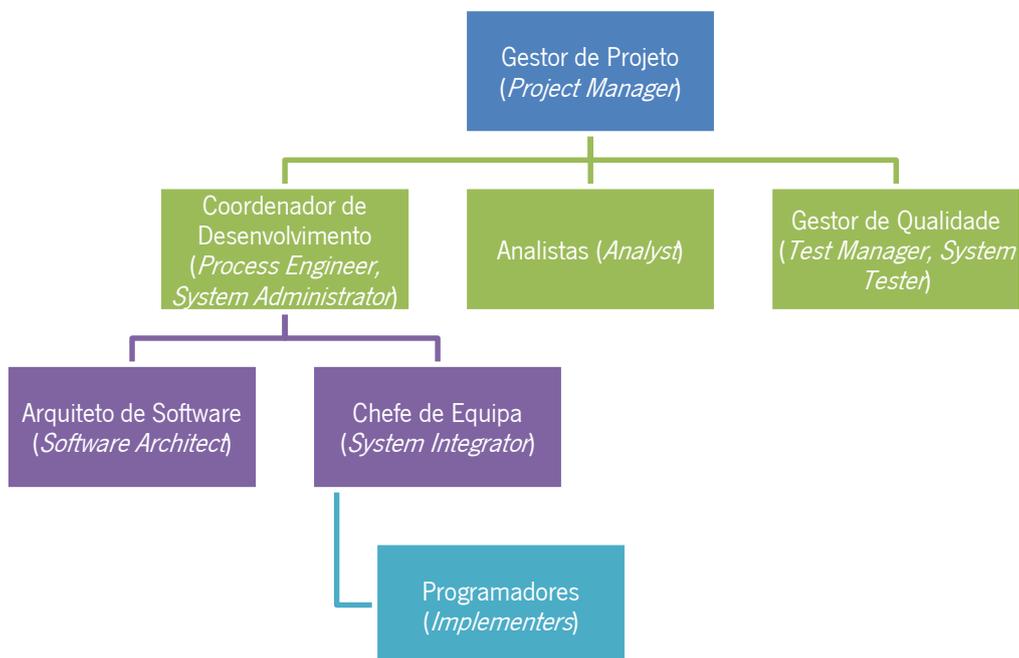


Figura 31 - Identificação do Modelo do EPRR

Adicionalmente, sugere-se que o papel de analista (*System Analyst*), que se encontra no modelo Base como um papel principal e no modelo Reduzido, se encontra agrupado com o papel gestor de projeto (*Project Manager*), seja separado das responsabilidades do mesmo. Uma vez que este papel possui um esforço investido que caso fosse adicionado ao esforço do gestor de projeto (*Project Manager*), resultaria numa de duas situações: numa diminuição da produtividade do mesmo em ambos os papéis ou no aumento do esforço investido por parte deste papel de forma a cumprir as responsabilidades de ambos os papéis.

Atendendo ao exposto, sugere-se, então, então, um modelo baseado nos dados adquiridos e conclusões retiradas que é apresentado de seguida na Figura 31. Este modelo reflete possíveis

correções para tornar o modelo Reduzido do RUP mais balanceado em termos de esforço, sendo esse modelo denominado de Elenco Processual Reduzido do RUP (EPRR).

Após apresentar do EPRR, apresenta-se, adicionalmente, uma forma alternativa de distribuição dos elementos extra das equipas de forma equilibrada, onde se pode observar que o número inicial de recursos humanos para aplicação desta distribuição é de 11 elementos e a distribuição dos restantes intervenientes por cada papel utiliza simples regras.

Papel	Número mínimo de intervenientes	Prioridade
Gestor de Projeto	1	-
Coordenador de Desenvolvimento	1	-
Arquiteto de Software	1	6
Chefe de Equipa	1	3 ou 4
Programadores	3	1, 2, 7, 8
Analistas	2	3 ou 4, 9, 10
Gestor de Qualidade	2	5

Tabela 14 - Distribuição dos Intervenientes pelos papéis do EPRR

Esta distribuição assenta sobre o pressuposto de que uma equipa de desenvolvimento não possui mais de 50 recursos humanos para distribuir, caso contrário não estaríamos a falar de uma organização de pequenas dimensões.

Seguidamente será explicitada a forma como todos os intervenientes devem ser distribuídos de forma a uniformizar o esforço das equipas de desenvolvimento para 11 elementos, 20 elementos, 30 elementos e 40 elementos, sendo este demonstrado na Tabela 15.

Papel	11 Elementos	20 Elementos	30 Elementos	40 Elementos
Gestor de Projeto	1	1	1	1
Coordenador de Desenvolvimento	1	1	1	1
Arquiteto de Software	1	2 (1+1)	2 (2+0)	3 (2+1)
Chefe de Equipa	1	2 (1+1)	3 (2+1)	4 (3+1)
Programadores	3	7 (3+4)	12 (7+4)	16 (12+4)
Analistas	2	4 (2+2)	7 (4+3)	9 (7+2)
Gestor de Qualidade	2	3 (2+1)	4 (3+1)	6 (4+2)

Tabela 15 - Distribuição dos recursos humanos

Caso o número de recursos humanos se encontre dentro de um destes intervalos, utiliza-se a menor distribuição (exemplo 24 elementos, utiliza-se a distribuição de 20 elementos) e os restantes são distribuídos de acordo com a coluna “Prioridade” da Tabela 14, onde o papel com o número 1 seria o primeiro a ser atribuído, seguidamente do número 2, utilizando a mesma regra para atribuir os papéis subsequentes. Em caso de prioridades iguais (no caso do Chefe de Equipa e do Analista), cabe a equipa decidir a sua distribuição. De forma a clarificar melhor esta distribuição ira-se realizar a atribuição de papéis a uma equipa de 25 elementos:

1. Primeiro procede-se a escolha do menor valor do intervalo pertencente a Tabela 15, sendo neste caso, o intervalo entre 20-30 elementos, sendo o menor valor 20 elementos, distribuindo assim os elementos de acordo com a coluna “20 elementos” da Tabela 14. Com isto encontram-se distribuídos 20 elementos dos 25 que se pretende distribuir.
2. Os restantes 5 elementos devem ser distribuídos seguindo a coluna “Prioridade” da Tabela 14.
3. O primeiro elemento dos 5 restantes receberá o papel de prioridade 1, sendo então atribuído o papel de Programador.
4. O segundo elemento dos 5 restantes receberá o papel de prioridade 2, sendo então atribuído o papel de Programador.
5. O Terceiro elemento dos 5 restantes receberá o papel de prioridade 3, que pode neste caso poderá ser um de dois papéis, ou Analista ou Chefe de Equipa, de acordo com as necessidades da equipa.

6. O Quarto elemento dos 5 restantes receberá o papel de prioridade 4, que será o papel restante da atribuição realizada no ponto 5.
7. O Quinto elemento dos 5 restantes receberá o papel de prioridade 5, sendo então atribuído o papel de Gestor de Qualidade.

Este método assegura, que existe uma distribuição uniformizadas dos recursos humanos no projeto. Adverte-se que este método não deve ser seguido como um dogma, a distribuição dos intervenientes, acima de tudo deve garantir o sucesso do projeto e um esforço balanceado de todos os intervenientes. Alerta-se ainda que esta distribuição pode resultar, em algumas situações irregulares, em um número superior ao necessário de alguns papéis, sendo da responsabilidade das equipas perceber a necessidade desse papel para o sucesso do projeto.

Dá-se então por concluído o capítulo prático, relacionado com o acompanhamento das equipas de desenvolvimento, onde se puderam realizar análises e retirar conclusões relevantes para o desenvolvimento de um novo modelo reduzido do RUP para pequenas empresas. No capítulo seguinte, que finaliza esta dissertação, apresentam-se as conclusões finais do projeto, bem como sugestões de propostas para futuros trabalhos.

## 5 Conclusões e Trabalho Futuro

Este capítulo tem como objetivo apresentar algumas conclusões que resultantes da análise dos dados com origem no modelo reduzido do RUP e que se esperam serem de utilidade para futuras adaptações. Outro assunto que é abordado neste capítulo refere-se a propostas de temas para futuros projetos, que possam dar continuidade ao trabalho desenvolvido nesta dissertação.

Ao longo do desenvolvimento do Capítulo 4, para além de se terem realizado diversas análises aos dados, chegando-se a algumas conclusões, foi também especulados alguns resultados, que não puderam ser validados nem testados devido a falta de determinadas métricas, que após a realização da iteração do estudo, fariam sentido estarem presentes nas entrevistas realizadas aos alunos. Neste capítulo será sintetizado de forma globais as conclusões do projeto de acordo com os requisitos definidos, assim como sugeridas potenciais trabalhos futuros de forma a facilitar a implementação do modelo reduzido do RUP em organizações de pequena dimensão.

### 5.1 Sintetização de Resultados

Esta secção destina-se a identificar de forma sintetizada os resultados obtidos durante o projeto assim como avaliar o cumprimento dos objetivos definidos inicialmente para esta dissertação.

Com a realização desta dissertação foi provado que o modelo Reduzido do RUP pode ser implementado em organizações de pequena dimensão, sendo que, apesar de não se ter conhecimento da avaliação das equipas de desenvolvimento, sabe-se que todas elas obtiveram aprovação na unidade Curricular de Desenvolvimento de Aplicações informáticas. Relativamente ao desempenho no cumprimento dos objetivos definidos para esta dissertação:

- O objetivo principal desta dissertação foi tentar encontrar um equilíbrio entre o esforço de cada papel do elenco processual. Este objetivo foi, dentro do possível cumprido com sucesso. O ajustamento dos papéis do modelo Reduzido do RUP foi realizado através da sugestão de algumas alterações ao modelo Reduzido do RUP atual, essas sugestões surgiram das análises realizadas aos dados levantados nas entrevistas feitas aos alunos que participaram no estudo de caso.

- O segundo objetivo passa por aumentar a produtividade das equipas de desenvolvimento em geral. Apesar de não ter sido testado com as métricas definidas inicialmente, espera-se que, através do balanceamento dos esforços se verifique um aumento da produtividade dos mesmos.
- O último objetivo centrava-se em diminuir o tempo e custo de formação das equipas até possuírem competências suficientes para desempenhar os papéis do modelo Reduzido do RUP, acabou por ser descartado uma vez que 99% dos alunos afirmavam possuir conhecimento das responsabilidades inerentes aos seus papéis, sendo por isso irrelevante analisar o esforço investido pelos mesmos na aquisição do conhecimento.

De uma forma global os objetivos principais desta dissertação foram cumpridos, embora alguns, necessitem de trabalho futuro para serem totalmente validados. Relativamente às conclusões obtidas no capítulo 4 pode-se verificar que:

- Algumas das métricas recolhidas, apesar de terem desempenhado um papel informativo, não tiveram muito impacto no ajustamento dos papéis. Foram recolhidas e utilizadas para identificar a maturidade das equipas. Exemplos disso são a faixa etária dos elementos, o género, estatuto do aluno e ano de inscrição.
- Por outro lado, outras métricas ajudaram, indiretamente a desenvolver o ajustamento realizado aos papéis do modelo reduzido do RUP, como é exemplo o número de unidades curriculares em atras ou os conhecimentos que o intervenientes tinham em relação às responsabilidades dos papéis no início do projeto.
- As métricas que diretamente ajudaram a sugerir ajustamentos ao modelo reduzido do RUP foram a constituição dos grupos (intervenientes distribuídos por cada papel e dimensão total das equipas) e o esforço investido pelos intervenientes (esforço total de cada grupo, esforço total de cada grupo nos diferentes momentos, esforço médio investido em cada papel do modelo Reduzido do RUP e, o esforço médio investido comparativamente ao esforço médio definido pelos créditos ECTS).
- Todas as métricas atrás definidas apoiaram no desenvolvimento desta dissertação, mas após realização da primeira iteração do modelo, foi notado que

provavelmente existiam outras métricas que poderiam ter enriquecido ainda mais o modelo, apoiando o novo modelo sugerido nesta dissertação. Duas das métricas principais seriam as notas individuais de cada grupo, de forma a poder medir o desempenho dos mesmos no projeto assim como avaliar o desempenho dos grupos em diferentes áreas de processos do CMMI.

Em suma e já exposto no capítulo anterior, relativamente ao modelo Reduzido do RUP utilizado pelos alunos na unidade curricular de Desenvolvimento de Aplicações Informáticas do mestrado integrado em Engenharia e Gestão de Sistemas de Informação e baseado nos dados recolhidos, pode afirmar-se que, a junção das responsabilidades do Chefe de Equipa (*System Integrator*), do Arquiteto de Software (*Software Architect*) e do Coordenador de Desenvolvimento (*Process Engineer*) não é uma opção viável, uma vez que se pretende uniformizar o esforço investido em todos os papéis.

No modelo Reduzido do RUP definido por (Borges, 2007), apenas o arquiteto de Software (*Software Architect*) e o Chefe de equipa (*System Integrator*), possuem responsabilidades acumuladas, sendo que foi opção dos alunos não definirem um Coordenador de Desenvolvimento (*Process Engineer*). Mesmo assim Sugere-se que esses papéis sejam repartidos assim como se encontra no modelo Base do RUP.

Outra conclusão relevante desta dissertação é o facto do Coordenador de infraestruturas (*System Administrator*) não ter um esforço proporcional durante todas as fases do projeto, sendo assim sugerido que as responsabilidades deste papel sejam agrupadas com as responsabilidades do Coordenador de Desenvolvimento (*Process Engineer*).

Sugere-se ainda, de forma a equilibrar o esforço entre o Auditor de Qualidade (*System Tester*) e o Coordenador de Qualidade (*Test Manager*), que as suas responsabilidades sejam agrupadas, surgindo assim o novo papel “Gestor de Qualidade”, sendo assim inicialmente definidos pelo menos dois intervenientes para desempenhar o novo papel.

Espera-se que este modelo, daqui a alguns anos, seja aplicado em organizações de desenvolvimento de Software de pequenas dimensões. Sendo isto provocado pela saída dos alunos da universidade do Minho após a conclusão dos seus percursos académicos e a entrada dos mesmos no mercado de trabalho com novas ideologias para valorizarem as organizações.

Apesar do esforço realizado para melhorar o modelo Reduzido, o Modelo EPRR ainda requer de experimentação antes de ser aplicado em contexto real. Este modelo encontra-se limitado pela falta de experiência por parte do autor em relação a contextos reais. Sendo assim possível encontrar no modelo soluções a problemas do estudo de caso que, simulava um contexto real mas que não deixava de ser um contexto académico, mas que aplicado no contexto real, poderia não possuir o mesmo efeito.

## 5.2 Trabalho Futuro

Dada a falta de experiência por parte do autor na definição inicial das métricas, assim como o limite temporal pelo qual estamos barrados, não é possível a exploração de outras vertentes que poderiam ter ajudado a enriquecer a validação do modelo desenvolvido por Borges em (Borges, 2007). Sendo assim, a continuidade desta dissertação pode passar por várias áreas:

- Seria interessante realizar o mesmo estudo mas utilizando a técnica de Observação Direta no estudo de caso, onde existiria contacto com as equipas e onde poderiam ser recolhidos dados que por entrevistas e análise documental não é possível recolher.
- Uma delas passa pela continuação direta desta dissertação de forma a avaliar o impacto que outras métricas (não experimentadas) teriam no modelo Reduzido do RUP.
- De forma a facilitar a compreensão, perceber as responsabilidades e aumentar o desempenho dos diversos papéis, seria interessante desenvolver de forma exhaustiva os diagramas de atividades que mostre de forma clara a sequência de tarefas que cada papel é responsável.
- Dentro da área da metodologia do RUP mas fora do elenco processual, sugeriria-se a simplificação dos artefactos do mesmo de forma a acompanhar a simplificação aplicada no elenco processual.
- Por fim o ideal seria testar a aplicação deste modelo num contexto real, onde se poderia realizar um estudo mais pormenorizado do modelo, sendo assim possível retirar conclusões que melhorariam o modelo, torando-o assim viável para aplicação em PMEs de contexto real.

- Adicionalmente sugeria-se a realização de um estudo de forma a investigar o sucesso dos papéis deste modelo quando aplicados em metodologias ágeis como o Agile Unified Process (AUP) ou o SCRUM.

## Bibliografia

- Beira, E., Kaldeich, C., Sousa, H., & Borges, P. (2005). Mapa TIC de Portugal : análise por distritos, 57.
- Borges, P. (2007). *Configuração do RUP com Vista à Simplificação dos Elencos Processuais em PMEs de Desenvolvimento de Software*.
- Borges, P., Monteiro, P., & Machado, R. J. (2011). Tailoring RUP to Small Software Development Teams. *2011 37th EUROMICRO Conference on Software Engineering and Advanced Applications*, 306–309. <http://doi.org/10.1109/SEAA.2011.55>
- Borges, P., Monteiro, P., & Machado, R. J. (2012). Mapping RUP Roles to Small Software Development Teams, 1–12.
- Bourque, P., & Fairley, R. E. (2014). *Guide to the Software Engineering Body of Knowledge (SWEBOK(R)): Version 3.0*. IEEE Computer Society Press.
- CMMI Product Team. (2010). *CMMI for Development, Version 1.3*. Carnegie Mellon University.
- Duarte, F. (2002). Engenharia de Software Orientada aos Processos.
- Fernandes, J. M., & Duarte, F. J. (2005). A reference framework for process-oriented software development organizations. *Software and Systems Modeling*, 4(1), 94–105. <http://doi.org/10.1007/s10270-004-0063-0>
- Fuggetta, A., & Di Nitto, E. (2014). Software [Process]. *Proceedings of the on Future of Software Engineering*, 1–12. <http://doi.org/10.1145/2593882.2593883>
- Grancho, N. (2005). Origem e Evolução Recente dos Sistemas de Informação Geográfica em Portugal Origem e Evolução Recente dos Sistemas de Informação Geográfica em Portugal, 167.
- IAPMEI. (2016). Definição de PME.
- Jacobson, I., Booch, G., & James, R. (1999). *The Unified Software Development Process*.
- Krahn, H., Rumpe, B., & Vollkel, S. (2014). Roles in Software Development using Domain Specific Modelling Languages.
- Kroll, P., & Kruchten, P. (2003). *Introducing the Rational Unified Process. Rational Unified Process Made Easy: A Practitioner's Guide to the RUP*.
- Kruchten, P. (2003). What Is the Rational Unified Process ? *Science*.
- Kruchten, P. (2004). The Rational Unified Process An Introduction, 336. <http://doi.org/10.1109/ICSE.2002.146346>
- Monteiro, P., Borges, P., Machado, R. J., & Ribeiro, P. (2012). A reduced set of RUP roles to small software development teams. *2012 International Conference on Software and System Process, ICSSP 2012 - Proceedings*, 190–199. <http://doi.org/10.1109/ICSSP.2012.6225964>
- Naur, P., Randell, B., & Committee, N. S. (1968). Software Engineering: Report of a conference sponsored by the NATO Science Committee, Garmisch, Germany, 7-11 Oct. 1968. In *NATO Software Engineering Conference* (p. 231).

- Project Management Institute. (2013). *Project Management Body of Knowledge A Guide to the Project Management Body of Knowledge*. <http://doi.org/10.1002/pmj.20125>
- Rational. (1998). Best practices for software development teams. *A Rational Software Corporation White Paper*, 1–21.
- Rational. (2001). Rup Tutorial.
- Universidade do Minho. (2016). Créditos ECTS na UMinho. Retrieved from <http://www.sri.uminho.pt/Default.aspx?tabid=10&pageid=506&lang=pt>
- Weber, C. V., Curtis, B., & Chrissis, M. B. (1996). The capability maturity model: Guidelines for improving the software process. *Computers & Mathematics with Applications*, *31*(7), 143. [http://doi.org/10.1016/S0898-1221\(96\)90461-7](http://doi.org/10.1016/S0898-1221(96)90461-7)
- Yin, R. (1994). Case Study Research., 1–5.
- Yin, R. (2003). *Case Study Research, Design and Methods*. SAGE Publications.

## Anexos

## Anexo A – Definição do Elenco Processual do RUP

<i>Role</i>	<i>Descrição</i>
<b><i>Analyst</i></b>	
<i>Business-Model Reviewer</i>	Possui a responsabilidade de revisão dos objetivos de negócio, assim como, dos modelos de casos de uso relacionados com o âmbito de <i>Business Modeling</i> .
<i>Business-Process Analyst</i>	Chefia e organiza a modelação dos casos de uso de negócio, delimitando a zona da organização necessária a modelar.
<i>Business Designer</i>	Papel que é encarregado do detalhamento da organização distinguindo os fluxos de casos de uso de negócio em relação a entidades e <i>workers</i> . É ainda responsável por definir responsabilidades, operações atribuídas e a interação entre as mesmas.
<i>Requirements Reviewer</i>	Responsável pelo planeamento e revisão do modelo de casos de uso.
<i>Requirements Specifier</i>	Papel responsável pelo detalhamento de funcionalidades do sistema, especificando mais aprofundadamente requisitos de um ou vários casos de uso. “Também pode ser responsável pela manutenção da integridade de um determinado pacote de casos de uso”(Borges, 2007)
<i>System Analyst</i>	Papel responsável por liderar e coordenar a identificação dos requisitos com a modelação dos casos de uso, delineando os sistemas e objetivando a funcionalidade do mesmo. Garante, ainda, que não existem inconsistências, nem incoerências no modelo de casos de uso. Sendo responsável pelo modelo de casos de uso não é responsável por cada caso de uso individualmente, sendo essa responsabilidade transferida para o <i>Use Case Specifier</i> . Por fim, é também o indivíduo que lidera o levantamento de requisitos.
<i>Use-Case Specifier</i>	Normalmente, o levantamento de requisitos não pode ser feito apenas por um só elemento de equipa. Estes intervenientes auxiliam o <i>System Analyst</i> com essa tarefa, ficando assim responsáveis pela descrição, o mais detalhada possível, de um ou vários casos de uso. Para isso, estes intervenientes necessitam de contactar com os utilizadores finais correspondentes a estes casos de uso.
<i>User-Interface Designer</i>	Papel responsável pela modelação, conceção e coordenação do interface de utilizador. O levantamento de requisitos de interface, a construção de

	protótipo e a revisão de interfaces são também, funcionalidades realizadas por este papel.
--	--

Tabela 16 - Elenco Processual - Grupo *Analyst*

<i>Role</i>	<i>Descrição</i>
<b><i>Developer</i></b>	
<i>Architecture Reviewer</i>	O papel responsável pelo planeamento e condução do desenvolvimento do projeto.
<i>Capsule Designer</i>	Papel responsável pela conceção da cápsulas, assegurando que o sistema consegue responder eficientemente e eficazmente a eventos, de acordo com os requisitos.
<i>Code Reviewer</i>	Responsável por assegurar a qualidade do código fonte e a sua correspondência com os requisitos definidos para o projeto, assim como, pelo planeamento e revisão do mesmo. Para além disso, fica a cabo deste mesmo role o <i>reporting</i> das conclusões resultantes do processo.
<i>Component Engineer</i>	Responsável pela definição das responsabilidades, relações, requisitos especiais e atributos de uma ou mais classes de análise e das operações, métodos, atributos, relações e requisitos de uma o mais classes de conceção, assegurando assim que, cada uma delas, obedece aos requisitos que lhe foram impostos pela realização do caso de uso em que participa. Tem como competência manter a integridade de um ou mais pacotes de análise, garantindo que os seus conteúdos estão corretos e que as suas dependências em relação a outros pacotes de análise são corretas e mínimas. Por Fim, compete-lhe ainda a integridade de um ou mais subsistemas de implementação. A maior parte das alterações a estes subsistemas são geridas durante a fase de conceção, uma vez que, entre estes subsistemas e os subsistemas que são concebidos existe uma correlação.
<i>Database Designer</i>	Papel responsável pela redefinição de todos os parâmetros específicos correspondentes à base de dados e que “estejam relacionados com armazenamento, recuperação e remoção de objetos” (Borges, 2007).
<i>Designer</i>	O Designer deve converter a arquitetura concebida pelo arquiteto numa solução coerente e detalhada de componentes e modelos para poder ser

<i>Role</i>	<b>Descrição</b>
	implementada, para poder ser implementada, criando operações e relações entre eles. Pode ser responsável por um ou mais pacotes de subsistemas.
<i>Design Reviewer</i>	Responsável pelo planeamento e revisão do modelo de conceção, assim como, pela revisão de artefactos desenvolvidos por outros elementos.
<i>Implementer</i>	Este papel tem a responsabilidade de testar diversas componentes de um sistema, de acordo com as regras definidas no plano, para futuramente se passar à sua integração noutros subsistemas.
<i>System Integrator</i>	Papel responsável pelo planeamento dos processos de integração provenientes do <i>Implementer</i> e, conseqüentemente, pela integração dos subsistemas em <i>builds</i> . Nele podemos incluir o planeamento da sequência de versões que são necessárias em cada iteração e a integração de cada versão.
<i>Software Architect</i>	<p>Líder e coordenador dos artefactos técnicos do projeto. Possui responsabilidades a vários níveis durante as diferentes fases do processo, nomeadamente:</p> <ul style="list-style-type: none"> <li>• Estabelecer a estrutura geral da visão Arquitetural. Logo, em, comparação com os restantes papéis, o arquiteto de Software necessita de ter uma visão do artefacto mais global ou abrangente do que profunda;</li> <li>• Participar no levantamento de requisitos para que consiga descrever a visão arquitetural do modelo de casos de uso;</li> <li>• Garantir, durante a fase de análise, a consistência do modelo de análise, assegurando que o mesmo está correto e legível. Dentro desta fase é ainda responsável pela arquitetura do modelo de análise.</li> <li>• Garantir, durante a fase de conceção, a instalação e a integridade do modelo de conceção assegurando assim a sua consistência e legibilidade. Dentro desta fase é ainda responsável pela arquitetura do modelo de conceção.</li> <li>• Assegurar, durante a fase de implementação, a integridade do modelo de implementação e garantir que o modelo está consistente, correto e perceptível. Dentro desta fase é ainda responsável pela arquitetura do modelo de implementação.</li> </ul>

<i>Role</i>	<b>Descrição</b>
	<ul style="list-style-type: none"> <li>Fazer o mapeamento das componentes de execução, resultantes do processo de implementação.</li> </ul>
<i>Use-Case Engineer</i>	<p>Cargo responsável pela integridade e conceção de uma ou mais concretizações de casos de uso, certificando-se assim que cumprem os requisitos na sua integridade. É necessário garantir que todas as descrições textuais e diagramas que descrevem a realização do caso de uso sejam legíveis. Saliente-se que o <i>Use-Case Engineer</i> não é responsável pela conceção das classes, subsistemas, interfaces e relações de análise empregues na concretização do caso de uso, sendo essas funções da responsabilidade do <i>Component Engineer</i>. É de notar que o <i>Use-Case Engineer</i> é, ainda, responsável pela conceção das concretizações do caso de uso, uma vez que, aquando da realização da análise e conceção existe uma transição natural.</p>

Tabela 17 - Elenco Processual - Grupo *Developer*

<i>Role</i>	<b>Descrição</b>
<b><i>Manager</i></b>	
<i>Change Control Manager</i>	<p>Este papel tem como funcionalidade a supervisão do processo de controlo de mudanças. Normalmente é constituído por um grupo de interessados que engloba os clientes, <i>developers</i> e os utilizadores. No desenvolvimento de pequena escala ou dimensão, este papel pode ser atribuído ao gestor de projeto (Project Manager).</p>
<i>Configuration Manager</i>	<p>Este papel tem como responsabilidade fornecer a infraestrutura de gestão de configurações para a equipa de desenvolvimento que tem como funcionalidade o suporte da sua atividade. Permite, assim, a criação de um espaço de trabalho para os <i>developers</i> e <i>integrators</i> realizarem as ações de implementação e teste e facultando os artefactos que produzem. Este cargo fica também responsável por desenvolver o plano de gestão de configurações.</p>

<i>Role</i>	<b>Descrição</b>
<i>Deployment Manager</i>	Este papel tem como responsabilidade o planeamento da transição do produto para o utilizador, ficando, também, responsável pela documentação inerente aos processos.
<i>Management Reviewer</i>	Este papel tem como principais responsabilidades a avaliação do planeamento do projeto, assim como, dos artefactos principais que são desenvolvidos nos momentos chave do ciclo de vida do projeto.
<i>Process Engineer</i>	Papel ao qual é atribuído a funcionalidade de desenvolvimento de processos de Software. Isto inclui toda a documentação pré desenvolvimento assim como a melhoria dos processos depois do desenvolvimento.
<i>Project Manager</i>	Papel responsável pelo planeamento e atribuição de práticas no sentido de fiabilizar e aumentar a qualidade dos artefactos produzidos pela equipa, sendo também responsável pela alocação de recursos, definição de prioridades, interações com clientes, assim como, por incentivar a equipa no cumprimento dos objetivos definidos.
<i>Project Reviewer</i>	Tem como responsabilidade analisar e validar os artefactos do planeamento de projeto referentes a <i>milestones</i> relevantes ao ciclo de vida do mesmo. Estas validações são de extrema relevância, uma vez que podem fazer um projeto ser cancelado, caso o planeamento tenha sido inadequado.
<i>Test Manager</i>	Este papel encontra-se responsável pelo sucesso na execução dos testes, sendo desta forma o promotor da qualidade e esforço na realização dos mesmos. Com isso em mente, este papel tem de conseguir desenvolver e realizar o planeamento e gestão dos recursos existentes, assim como, resolver potenciais problemas que possam impedir a realização de determinados testes.

Tabela 18 - Elenco Processual - Grupo *Manager*

<i>Role</i>	<b>Descrição</b>
<b><i>Tester</i></b>	
<i>Integration Tester</i>	A realização dos testes integração para cada versão do fluxo de implementação é da responsabilidade deste cargo. Estes testes são realizados de forma a verificar se as componentes integradas em cada versão funcionam corretamente em conjunto. É de notar que, em alguns

<i>Role</i>	<b>Descrição</b>
	projetos, o papel de <i>System Integrator e Integration Tester</i> são acumulados, de forma a minimizar a replicação de conhecimento comum entre colaboradores.
<i>System Tester</i>	Este cargo encontra-se encarregue da realização dos testes de sistema necessários para a criação da versão resultante de uma iteração completa. Estes testes são desenvolvidos e executados com o intuito de verificar as interações entre atores e o próprio sistema.
<i>Test Analyst</i>	Este papel é responsável pela identificação e definição dos testes que sejam necessários realizar. Responsabiliza-se por supervisionar o desenvolvimento e os resultados dos testes realizados em cada ciclo de testes, assim como, por avaliar a qualidade da experiência resultante da realização dos mesmos. Possui, ainda, uma responsabilidade acrescida ao nível dos intervenientes, devendo representar de forma adequada as necessidades dos <i>stakeholders</i> que não têm uma representação de forma direta ou regular no projeto.
<i>Test Designer</i>	Responsável pelo planeamento, conceção, implementação e avaliação do teste. Inclui, assim, a criação de planos e modelos de teste, gestão da integridade dos modelos para <i>testing</i> (garantindo que estes cumprem os seus objetivos), implementação dos procedimentos, avaliação da extensão dos testes (resultados e eficácia), assim como, a geração de sumários de avaliação de esforço dos testes.

Tabela 19 - Elenco Processual - Grupo *Tester*

<i>Role</i>	<b>Descrição</b>
<b><i>Production and Support</i></b>	
<i>Course Developer</i>	Papel responsável pela formação e pelo desenvolvimento de material necessário para a mesma de forma a instruir os utilizadores acerca do produto.
<i>Graphic Artist</i>	Papel responsável pela componente gráfica inerente ao pacote de distribuição do produto”.
<i>Review Coordinator</i>	Papel responsável por diminuir a dificuldade das inspeções formais e revisões, tendo por isso, que assegurar que as mesmas ocorram quando

<i>Role</i>	<b>Descrição</b>
	necessário, sem problemas e orientadas de acordo com os processos definidos previamente.
<i>System Administrator</i>	Papel responsável por manter o ambiente de desenvolvimento organizacional “saudável”, ou seja, é responsável por todos as componentes quer de Hardware quer de Software que estejam envolvidos no projeto, assim como, pela realização de cópias de segurança ao sistema.
<i>Technical Writer</i>	Papel responsável pela produção de manuais ou instruções de suporte para utilizadores.
<i>Tool Specialist</i>	Papel responsável pelas ferramentas de suporte ao projeto, incluindo a instalação, a configuração e a aquisição das mesmas.

Tabela 20 - Elenco Processual - Grupo *Production and Support*

<i>Role</i>	<b>Descrição</b>
<b><i>Additional</i></b>	
<i>Any Role</i>	Papel que, caso possuas as permissões necessárias, pode fazer <i>check-in</i> e <i>check-out</i> dos artefactos do projeto para realizar manutenções do sistema de controlo de configurações.
<i>Stakeholder</i>	Corresponde a qualquer elemento que seja afetado pelo resultado final do projeto. Este papel cobre na sua maior parte clientes, utilizadores finais, acionistas, etc.

Tabela 21 - Elenco Processual - Grupo *Additional*

## Anexo B – Comparação entre o modelo Base e o modelo Reduzido

<b>Modelo Base</b>	<b>Papéis do RUP</b>	<b>Modelo Reduzido</b>
<i>Project Manager</i>	<b>Project Manager</b>	<i>Project Manager</i>
	<i>Business Process Analyst</i>	
	<i>Change Control Manager</i>	
	<i>Deployment Manager</i>	
	<i>Requirements Specifier</i>	
	<i>Review Coordinator</i>	
	<i>Test Analyst</i>	
<i>System Analyst</i>	<b>System Analyst</b>	
	<i>Business Designer</i>	
	<i>Use Case Specifier</i>	
<i>System Integrator</i>	<b>System Integrator</b>	<i>System Integrator</i>
	<i>Capsule Designer</i>	
	<i>Code Reviewer</i>	
	<i>Integration Tester</i>	
<i>Software Architect</i>	<b>Software Architect</b>	
	<i>Design Reviewer</i>	
<i>Course Developer</i>	<b>Course Developer</b>	
<i>Data Base Designer</i>	<b>Data Base Designer</b>	
<i>Project Reviewer</i>	<b>Project Reviewer</b>	<i>Project Reviewer</i>
	<i>Business Reviewer</i>	
	<i>Management Reviewer</i>	
	<i>Requirements Reviewer</i>	
<i>Process Engineer</i>	<b>Process Engineer</b>	<i>Process Engineer</i>
	<i>Architecture Reviewer</i>	
	<i>Tool Specialist</i>	
<i>Implementer</i>	<b>Implementer</b>	
	<i>Component Engineer</i>	
<i>System Integrator</i>	<i>Designer</i>	<i>Implementer</i>
<i>User-Interface Designer</i>	<b>User-Interface Designer</b>	
	<i>Graphic Artist</i>	
<i>Course Developer</i>	<i>Technical Writer</i>	
<i>System Administrator</i>	<b>System Administrator</b>	<i>System Administrator</i>
	<i>Configuration Manager</i>	
<i>Test Manager</i>	<b>Test Manager</b>	<i>Test Manager</i>
	<i>Test Designer</i>	
<i>System Analyst</i>	<i>Use Case Engineer</i>	
<i>System Tester</i>	<b>System Tester</b>	<i>System Tester</i>

Tabela 22 - Comparação entre o modelo Base e o modelo Reduzido





## Anexo D – Papéis do Modelo Base que não se encontram no Reduzido

Os papéis identificados e caracterizados neste anexo encontram-se apenas dentro do modelo Base do RUP.

### *Course Developer*

Como em qualquer tipo de negócio, o sucesso de uma organização depende em muito da opinião e ideia que o utilizador final possui do produto. Na área de negócio de desenvolvimento de Software é extremamente relevante a forma como o cliente experiencia o produto depois de receber, sendo da responsabilidade do Course Developer melhorar a forma como o cliente interage com o produto. Assim, enquadram-se nas responsabilidades exigidas a este interveniente preparar e ministrar formações, produzir manuais de suporte à configuração e instalação do produto e, ainda, preparar e formar as linhas de suporte, que deverão estar funcionais a partir do momento em que o produto é apresentado ao cliente

#### **Responsável por...**

- Identificar, em conjunto com o *System Administrator*, a documentação de suporte propícia a ser produzida por cada equipa de desenvolvimento, de forma a cumprir os requisitos definidos.
- Caracterizar o perfil dos utilizadores finais do projeto.
- Criar e executar os planos de formação, tendo como alvo os utilizadores finais e as linhas de suporte ao cliente.
- Analisar e reportar ao Project Manager necessidades de infraestruturas, de forma a obter um bom funcionamento das linhas de suporte.
- Estruturar adequadamente as diferentes linhas de atendimento.
- Garantir o cumprimento de todas as suas funções, por parte de todas as linhas de suporte.
- Conhecer as boas práticas existentes na organização relacionadas com a produção de artefactos documentais.

- Notificar o *Project Manager* sempre que pretender fazer alteração nos períodos de férias.

### Perfil...

Todas as condições apresentadas seguidamente não são obrigatórias, mas é aconselhável que os intervenientes satisfaçam as mesmas.

- Experiência de pelo menos dois anos na prestação de formação e suporte a clientes finais.
- Conhecimentos de informática na ótica do utilizador.

Participação nos Fluxos de Informação Internos	Participação nos Fluxos de Informação Externos
<p><i>Software Architect</i>: Com objetivo de perceber os requisitos estabelecidos para o suporte.</p> <p><i>System Integrator</i>: Com objetivo de criar um plano de formação adequado para ministrar aos clientes finais e as equipas de suporte.</p> <p><i>Project Manager</i>: Com objetivo de sugerir o plano de formação a ser implementado assim como as diferentes linhas de suporte ao consumidor.</p>	<p><i>Course Developer</i> (Externo): Com objetivo de esclarecer a estruturação das linhas suporte, bem como a distribuição de responsabilidades entre as linhas de suporte internas e externas a organização.</p>

### Funções Críticas...

- Garantir que o cliente final possui uma experiência agradável na utilização do produto desenvolvido, esclarecendo-o sobre questões relacionadas com a configuração e utilização do sistema, ou mesmo, fornecendo-lhe uma formação adequada.
- Garantir que as linhas de suporte possuem formação suficiente para responder eficientemente ao cliente.

## O que deve fazer...

No Início do Projeto	No Final do Projeto
<p>Caracterizar o perfil dos utilizadores finais do projeto.</p> <p>Sugerir ao <i>Project Manager</i> a composição estrutural das equipas de suporte.</p> <p>Sugerir ao <i>Project Manager</i>, após discussão com o <i>System Integrator</i>, o plano de formação que deve ser aplicado aos clientes finais e equipa de suporte.</p> <p>Identificar, em conjunto com o <i>System Administrator</i>, a documentação de suporte propícia a ser produzida de acordo com cada equipa de desenvolvimento.</p> <p>Identificar, conjuntamente com os <i>System Integrators</i>, o formato e localização da documentação de suporte que será desenvolvida por cada equipa de desenvolvimento.</p> <p>Analisar e reportar ao <i>Project Manager</i> necessidades de infraestruturas, de forma a obter um bom funcionamento das linhas de suporte.</p> <p>Verificar a situação do calendário de férias em vigor ou proposto e, caso exista alguma anomalia, alertar o <i>Project Manager</i>.</p>	<p>Realizar diversas chamadas teste às linhas de suporte, com o intuito de verificar se as mesmas conseguem responder com sucesso aos desafios propostos.</p> <p>Certificar-se que a documentação de suporte da última iteração foi produzida e cumpre com os requisitos definidos.</p> <p>Comunicar ao <i>Process Engineer</i> a avaliação do processo de desenvolvimento de Software que foi utilizado no projeto, sendo ainda desejável a sugestão de eventuais correções e melhorias.</p>

No Início de cada iteração	Durante cada iteração	No Final de cada iteração
<p>Identificar, em conjunto com o <i>System Administrator</i> a documentação de suporte propícia a ser produzida de acordo com cada equipa de desenvolvimento.</p> <p>Averiguar se documentação de suporte da iteração anterior foi produzida e cumpre os requisitos estipulados.</p> <p>Fazer revisão, conjuntamente com o <i>System Integrator</i>, do formato e localização da documentação de suporte que será desenvolvida na iteração seguinte.</p> <p>Verificar se pretende fazer alguma alteração no seu calendário de férias que coincida com o início da iteração. Em caso de alteração, comunicar ao <i>Project Manager</i>.</p>	<p>Fazer a preparação dos conteúdos necessários para a realização do plano de formação às equipas de suporte e aos clientes finais.</p> <p>Conhecer as boas práticas existentes na organização relacionadas com a produção de artefactos documentais.</p> <p>Avaliar a viabilidade dos requisitos que foram definidos para o suporte, caso não seja viável, procurar uma solução alternativa aceitável pelas entidades externas.</p> <p>Garantir o cumprimento de todas as responsabilidades por parte das linhas de suporte de qual é responsável.</p> <p>Garantir a produção do material de formação e suporte.</p> <p>Identificar requisitos com detalhe insuficiente e comunicar ao <i>Software Architect</i> ou ao <i>Project Manager</i>, caso o primeiro não exista, tal situação.</p>	<p>Garantir a disponibilidade do material de suporte e formação aos utilizadores finais.</p> <p>Aplicar formação às equipas de suporte e aos utilizadores finais, caso se mostre necessário.</p> <p>Verificar se as linhas de suporte se encontram com infraestruturas que satisfaçam as suas necessidades. Caso não se verifique essa situação, contactar o <i>Project Manager</i>.</p> <p>Assegurar que o esforço alocado para a realização do seu trabalho se encontra atualizado.</p> <p>Verificar se o calendário de férias dos <i>System Testers</i> não coincide com a execução da próxima iteração.</p>

## *Database Designer*

O *Database Designer* intervém quando os intervenientes não se encontram familiarizados com certos requisitos na utilização de bases de dados. A utilização deste interveniente que, devido à sua experiência e formação específica, se encontra confortável com a sua utilização, contribuirá de forma positiva para o sucesso do projeto, garantindo assim, a conceção dos modelos de bases de dados de forma que outros intervenientes provavelmente não conseguiriam.

### **Responsável por...**

- Conceber juntamente com o *System Integrator* os modelos de dados que irão apoiar os motores de bases de dados.
- Configurar, manter e otimizar os motores de bases de dados.
- Escolher juntamente com o *System Integrator* as políticas mais apropriadas para a segurança dos motores de bases de dados.
- Criar juntamente com o *System Integrator* os índices, *views*, *constraints*, etc. necessários para que a utilização dos motores de bases de dados sejam otimizados.
- Notificar o *Project Manager* sempre que pretender fazer alteração nos períodos de férias.

### **Perfil...**

Todas as condições apresentadas seguidamente não são obrigatórias, mas é aconselhável que os intervenientes satisfaçam as mesmas.

- Formação de 2º ciclo de Bolonha ou equivalente na área de informática, mais precisamente, na conceção e normalização de modelos de dados.
- Formação relacionada com gestão e configuração de motores de bases de dados.
- Experiência de pelo menos dois anos em instalação, configuração e administração dos motores de dados.

<b>Participação nos Fluxos de Informação Internos</b>	<b>Participação nos Fluxos de Informação Externos</b>
<i>System Integrador</i> : Com objetivo de recolher informações necessárias sobre os modelos de	Não existe necessidade de intervenção deste interveniente em fluxos de informação externa.

dados, necessidades de *views*, *triggers*, etc. e sobre as políticas de segurança dos motores de bases de dados.

### Funções Críticas...

- Assegurar o funcionamento otimizado de todos os motores de bases de dados pelos quais é responsável.
- Garantir a normalização de todos os modelos de dados que são suportados pelos motores de bases de dados pelos quais é responsável.

### O que deve fazer...

No Início do Projeto	No Final do Projeto
Conceber juntamente com o <i>System Integrator</i> os modelos de dados que irão apoiar os motores de bases de dados.	Realizar, sobre a supervisão do <i>Project Manager</i> , uma cópia de segurança de toda a informação relevante que esteja diretamente relacionada com as bases de dados do projeto.
Escolher juntamente com o <i>System Integrator</i> as políticas mais apropriadas para a segurança dos motores de bases de dados.	Comunicar ao <i>Process Engineer</i> a avaliação relativamente ao processo de desenvolvimento de Software que foi utilizado no projeto, sendo ainda desejável, a sugestão de eventuais correções e melhorias.
Verificar a situação atual do seu calendário de férias, caso exista algum problema, alertar o <i>Project Manager</i> .	

No Início de cada iteração	Durante cada iteração	No Final de cada iteração
<p>Fazer revisão juntamente com o <i>System Integrator</i> dos modelos de dados a que os motores de bases de dados vão dar apoio, de forma a identificar possíveis alterações a serem realizadas durante a iteração.</p> <p>Verificar se a política definida para a segurança dos motores de base de dados se encontra corretamente aplicada e em funcionamento, ou se existe a necessidade de efetuar alguma alteração.</p>	<p>Gerir e otimizar o funcionamento dos motores de bases de dados pelos quais é responsável.</p> <p>Criar, juntamente com o <i>System Integrator</i>, os índices, <i>views</i>, <i>constraints</i>, etc. necessários para que a utilização dos motores de bases de dados sejam otimizados.</p>	<p>Assegurar que o esforço alocado para a realização do seu trabalho se encontra atualizado.</p> <p>Verificar se pretende fazer alterações no seu calendário de férias que coincida com o início da iteração seguinte, devendo ser o <i>Project Manager</i> notificado de tal situação.</p>

## *System Analyst (Analyst)*

A participação deste interveniente é altamente aconselhada em determinadas situações, uma vez que, existe a possibilidade do *Project Manager* não possuir os conhecimentos técnicos necessários, tais como, o conhecimento da notação UML, imprescindível para concretizar o processo de levantamento de requisitos. A utilização de um ou vários *System Analysts* irá ser indispensável para delinear de forma correta e explícita o âmbito do projeto, minimizando assim a possibilidade da criação de um produto diferente das expectativas das entidades externas. Ainda que o *Project Manager* possua competências técnicas necessárias para a realização de tais atividades, será ainda favorável a utilização de *System Analysts* quando os requisitos possuírem uma complexidade ou dimensão tal que o *Project Manager* individualmente não possui capacidade de caracterizar de forma detalhada ou atempada ou ambos

### **Responsável por...**

- Coordenar todo o processo de levantamento de requisitos, quer funcionais quer não funcionais, de forma a produzir os documentos necessários para delimitar o âmbito do projeto.
- Identificar possíveis requisitos que não foram referidos pelas entidades externas, mas que se podem encontrar dentro das suas expectativas, ou que podem criar uma oportunidade comercial.
- Validar a exequibilidade de todos os requisitos identificados. Sempre que tal não aconteça, contactar as entidades externas de forma a renegociar os requisitos ou, possivelmente, tentar encontrar uma solução que satisfaça ambos os lados.
- Fazer todos os possíveis no sentido de esclarecer todos os requisitos que não se encontram suficientemente descritos pelas entidades externas, de forma a alinhar o máximo possível as expectativas de ambas as partes.
- Supervisionar a realização de todos os casos de uso que resultem do modelo de casos de uso, de forma a garantir a integridade e assegurar são cumpridos na totalidade os respetivos requisitos.
- Identificar novos requisitos apresentados durante a realização do projeto por parte das entidades externas e formalizar internamente as alterações, comunicando-as ao *Project Manager*.

- Notificar o *Project Manager* sempre que pretender fazer alteração nos períodos de férias.

### Perfil...

Todas as condições apresentadas seguidamente não são obrigatórias, mas é aconselhável que os intervenientes satisfaçam as mesmas.

- Formação de 2º ciclo de Bolonha ou equivalente na área de informática.
- Conhecimentos relacionados com a área de negócio em que o projeto se encontra envolvido.
- Formação aprofundada de notação UML e das técnicas de gestão de requisitos.
- Experiencia de pelo menos um ano em desenvolvimento de Software utilizando métodos iterativos.

Participação nos Fluxos de Informação Internos	Participação nos Fluxos de Informação Externos
<p><i>Project Manager:</i> Com objetivo de lhe transmitir os requisitos identificados e oportunidades de negócio identificadas.</p> <p><i>System Integrador:</i> Com objetivo de lhe transmitir os requisitos para o sistema pelo qual é responsável.</p> <p><i>System Administrator:</i> Com objetivo de lhe transmitir os requisitos estabelecidos para a infraestrutura e de validar a exequibilidade dos mesmos.</p> <p><i>Course Developer:</i> Com objetivo de lhe transmitir os requisitos estabelecidos para o suporte e de validar a sua exequibilidade.</p>	<p><i>Project Manager:</i> Com objetivo de obter mais informações relevantes ao processo de levantamento de requisitos</p>

### Funções Críticas...

- Ajudar as entidades externas na identificação das necessidades, de forma a clarificar posteriormente os requisitos necessários.
- Identificar requisitos não exequíveis e encontrar soluções alternativas que possam ser aceites pelas entidades externas.
- Assegurar que o *Project Manager* possui conhecimento atualizado de todos os requisitos identificados.

- Comunicar as novas oportunidades comerciais do projeto ao *Project Manager*.

### O que deve fazer...

No Início do Projeto	No Final do Projeto
<p>Coordenar todo o processo de levantamento de requisitos, quer funcionais que não funcionais, de forma a produzir os documentos necessários para delimitar o âmbito do projeto.</p> <p>Verificar a situação atual do seu calendário de férias, caso exista algum problema, alertar o <i>Project Manager</i>.</p>	<p>Comunicar ao <i>Process Engineer</i> a avaliação relativamente ao processo de desenvolvimento de Software que foi utilizado no projeto, sendo ainda desejável, a sugestão de eventuais correções e melhorias.</p>

No Início de cada iteração	Durante cada iteração	No Final de cada iteração
<p>Analisar juntamente com o Project Manager se existe a necessidade de atualização da documentação dos requisitos.</p>	<p>Identificar possíveis requisitos que não foram identificados pelas entidades externas, mas que se podem encontrar dentro das suas expectativas, ou que podem criar uma oportunidade comercial.</p> <p>Identificar novos requisitos apresentados pelas entidades externas durante a realização do projeto e formalizar internamente as alterações, comunicando-as ao Project Manager.</p> <p>Fazer todos os possíveis no sentido de esclarecer todos os requisitos que não se encontram suficientemente descritos pelas entidades externas de forma a alinhar o máximo possível as expectativas de ambas as partes.</p> <p>Validar a exequibilidade de todos os requisitos identificados e, sempre que essa situação não se verifique, contactar as entidades externas, de forma a renegociar os requisitos ou tentar encontrar uma solução que satisfaça ambos os lados.</p> <p>Supervisionar a realização de todos os casos de uso que resultem do modelo de casos de uso, de forma a garantir a sua integridade, e assegurar que cumprem totalmente os respetivos requisitos.</p>	<p>Assegurar que o esforço alocado para a realização do seu trabalho se encontra atualizado.</p> <p>Verificar se pretende fazer alterações no seu calendário de férias que coincida com o início da iteração seguinte, devendo ser o <i>Project Manager</i> notificado de tal situação</p>

## Software *Architect*

Devido ao facto das organizações PME terem recursos humanos escassos e, conseqüentemente, apresentarem dificuldades em ministrar toda a formação necessária para prepararem os colaboradores, nem sempre é possível conseguir colaboradores com perfil totalmente adequado para se iniciar um novo projeto. Assim, existe a necessidade de encarregar um *System Integrator* com essas responsabilidades, que poderá não possuir os conhecimentos técnicos necessários para orientar autonomamente todo o espectro de *Implementers* que se encontram no projeto, de forma a conseguir adquirir as soluções técnicas mais otimizadas para organização. Espera-se, então, o auxílio do Software *Architect* no desempenho das funções do *System Integrator* no processo de tomada de decisão, já que possui uma maior experiência nas áreas técnicas. Assim, garante-se um aumento da qualidade do espectro de soluções possíveis a implementar sem influenciar os recursos humanos envolvidos no projeto.

### Responsável por...

- Identificar e informar o *Project Manager* e o *Process Engineer* sobre os possíveis riscos do projeto associados às decisões arquiteturais tomadas.
- Utilizar ações que permitam ao *System Integrator* melhorar as suas competências técnicas.
- Discutir, em conjunto com o *Process Engineer* e o *System Integrator*, as decisões principais de carácter técnico que originaram a arquitetura de desenvolvimento.
- Recomendar ao *Process Engineer* a aplicação de métodos de trabalho uniformes comuns a toda a equipa de desenvolvimento.
- Rever, se solicitado pelo responsável, os interfaces das componentes principais do sistema.
- Apoiar o *Process Engineer* na avaliação do desempenho do respetivo *System Integrator*.
- Fazer um acompanhamento constante das novas tendências tecnológicas, de forma a identificar potenciais tecnologias úteis para o projeto.
- Notificar o *Project Manager*, assim como o *System Integrator*, sempre que pretender fazer alteração nos períodos de férias.

## Perfil...

Todas as condições apresentadas seguidamente não são obrigatórias, mas é aconselhável que os intervenientes satisfaçam as mesmas.

- Formação de 2º ciclo de Bolonha ou equivalente na área de informática, com um complemento de 3º ciclo em Engenharia de Software.
- Experiência mínima de cinco anos no desenvolvimento de Software.
- Capacidade, iniciativa de autoaprendizagem e vontade de conhecer novidades.
- Capacidades de Análise elevadas.
- Formação aprofundada da notação UML.

Participação nos Fluxos de Informação Internos	Participação nos Fluxos de Informação Externos
<p><i>Process Engineer:</i> Com objetivo de:</p> <ul style="list-style-type: none"><li>• Notificar sobre a contribuição do <i>System Integrator</i>.</li><li>• Identificar os possíveis riscos do projeto associados com as decisões arquiteturais tomadas.</li><li>• Discutir as decisões principais de carácter técnico, que originaram a arquitetura de desenvolvimento.</li><li>• Recomendar ao <i>Process Engineer</i> a aplicação de métodos de trabalho uniformes a toda a equipa de desenvolvimento.</li></ul> <p><i>System Integrator:</i> Com objetivo de discutir as decisões principais de carácter técnico, que originaram a arquitetura de desenvolvimento.</p> <p><i>Project Manager:</i> Com objetivo de Identificar e informar o <i>Project Manager</i> sobre os possíveis riscos do projeto associados com as decisões arquiteturais tomadas.</p>	<p>Não existe necessidade de intervenção deste interveniente em fluxos de informação externa.</p>

## Funções Críticas...

- Auxiliar o *System Integrator* na resolução de problemas, gerando um maior espectro de soluções para os problemas técnicos encontrados.

- Ajudar no desenvolvimento das competências de cariz profissional do *System Integrator*.

### O que deve fazer...

No Início do Projeto	No Final do Projeto
<p>Identificar e informar o <i>Project Manager</i> e o <i>Process Engineer</i> sobre os possíveis riscos do projeto associados com as decisões arquiteturais tomadas.</p> <p>Discutir, em conjunto com o <i>Process Engineer</i> e o <i>System Integrator</i> as decisões principais de carácter técnico, que originaram a arquitetura de desenvolvimento.</p> <p>Rever, se solicitado pelo responsável, os interfaces das componentes principais do sistema.</p> <p>Recomendar ao <i>Process Engineer</i> a aplicação de métodos de trabalho uniformes a toda a equipa de desenvolvimento.</p> <p>Verificar a situação atual do seu calendário de férias, caso exista algum problema, alertar o <i>System Integrator</i>.</p>	<p>Apoiar o <i>Process Engineer</i> com a avaliação do desempenho do respetivo <i>System Integrator</i> ao longo do projeto.</p> <p>Comunicar ao <i>Process Engineer</i> a avaliação das competências do <i>System Integrator</i> que o mesmo acompanhou ao longo do projeto</p> <p>Comunicar ao <i>Process Engineer</i> a avaliação relativamente ao processo de desenvolvimento de Software que foi utilizado no projeto, sendo ainda desejável, a sugestão de eventuais correções e melhorias.</p>

No Início de cada iteração	Durante cada iteração	No Final de cada iteração
<p>Apoiar o <i>Process Engineer</i> na avaliação das propostas dos diversos <i>System Integrators</i> relacionadas com o reaproveitamento ou não, das componentes utilizadas para o desenvolvimento.</p>	<p>Contactar o <i>System Integrator</i> e o <i>Process Engineer</i>, sempre que justificável, comunicando-lhe alguma alteração da arquitetura, assim como, os riscos associados à mesma, sendo isto seguido da atualização dos respetivos artefactos documentais.</p> <p>Ajudar no desenvolvimento das competências de cariz profissional do <i>System Integrator</i>.</p> <p>Fazer um acompanhamento constante das novas tendências tecnológicas, de forma a identificar potenciais tecnologias úteis para o projeto.</p>	<p>Apoiar o <i>Process Engineer</i> na avaliação do desempenho do respetivo <i>System Integrator</i>.</p> <p>Assegurar que o esforço alocado para a realização do seu trabalho se encontra atualizado.</p> <p>Verificar se pretende fazer alterações no seu calendário de férias que coincida com o início da iteração seguinte, devendo ser o <i>Test Manager</i> notificado de tal situação.</p>

## *User-Interface Designer*

Quando existe dentro da organização uma equipa formada por elementos cujo a formação incidiu profundamente na área da informática, verifica-se que os mesmos produzem resultados mais focados nos requisitos funcionais. Involuntariamente, estes profissionais tendem a ignorar as componentes estéticas, bem como, a informação de determinadas funcionalidades, que podem parecer de conhecimento geral mas, para utilizadores menos experientes ou menos capacitados, poderá tornar-se uma dificuldade. Salienta-se, contudo, que algumas das entidades externas, para além de apreciarem o resultado final primam pela aparência do seu produto. Assim, é de elevada importância, em projetos com relevância, a presença de um elemento com um espírito diferente, com conhecimentos em áreas como comunicação e imagem, de forma a consiliar os aspetos referidos anteriormente.

### **Responsável por...**

- Certificar o cumprimento de todas as carências relacionadas com a comunicação e imagem do projeto.
- Encontrar, em conjunto com o *Project Manager*, todas as necessidades relacionadas com a comunicação e imagem que estão previstas para o projeto.
- Administrar todas as interfaces gráficas utilizadas no decorrer do projeto.
- Solicitar ao *Project Manager*, quando possível, todos os recursos relacionados com as entidades externas necessários à realização das atividades.
- Notificar o *Project Manager* sempre que se pretender fazer alterações no(s) períodos de férias.

### **Perfil...**

Todas as condições apresentadas seguidamente não são obrigatórias, mas é aconselhável que os intervenientes satisfaçam as mesmas.

- Formação de 2<sup>º</sup> ciclo de Bolonha ou equivalente na área de ciências da comunicação e multimédia.
- Conhecimentos das ferramentas de tratamento de imagem utilizados pela organização.

- Forte sentido estético.

Participação nos Fluxos de Informação Internos	Participação nos Fluxos de Informação Externos
<p><i>System Integrator:</i> Com o objetivo de detalhar as necessidades de imagem a ser implementadas no desenvolvimento do projeto.</p> <p><i>Project Manager:</i> Com objetivo de conhecer as necessidades de imagem e comunicação do projeto, assim como, informar o interveniente do desenvolvimento das suas atividades.</p>	<p>Não existe necessidade de intervenção deste interveniente em fluxos de informação externa.</p>

### Funções Críticas...

- Assegurar que a organização transmita, às entidades externas, uma imagem positiva do desenvolvimento dos seus projetos.

### O que deve fazer...

No Início do Projeto	No Final do Projeto
<p>Encontrar, em conjunto com o <i>Project Manager</i>, todas as necessidades relacionadas com a comunicação e a imagem que estão previstas para o projeto.</p> <p>Verificar a situação atual do seu calendário de férias, caso exista algum problema, alertar o <i>Project Manager</i>.</p>	<p>Comunicar ao <i>Process Engineer</i> a avaliação relativamente ao processo de desenvolvimento de Software que foi utilizado no projeto, sendo ainda desejável, a sugestão de eventuais correções e melhorias.</p>

No Início de cada iteração	Durante cada iteração	No Final de cada iteração
<p>Rever, em conjunto com o <i>Project Manager</i>, todas as necessidades relacionadas com a comunicação e a imagem que estão previstas para iteração que se inicia.</p>	<p>Verificar o cumprimento de todas as carências relacionadas com a comunicação e imagem do projeto.</p> <p>Comunicar com o <i>System Integrator</i> com o objetivo de detalhar as necessidades de imagem a ser implementadas no desenvolvimento.</p> <p>Administrar todas as interfaces gráficas utilizadas no decorrer do projeto.</p> <p>Solicitar ao <i>Project Manager</i>, quando possível, todos os recursos relacionados com as entidades externas necessárias à realização das atividades.</p>	<p>Assegurar que o esforço alocado para a realização do seu trabalho se encontra atualizado.</p> <p>Verificar se pretende fazer alterações no seu calendário de férias que coincida, com o início da iteração seguinte, devendo ser o <i>Test Manager</i> notificado de tal situação</p>

