

# Aprender como equipa para trabalhar em equipa

## *Learn as a team to work in a team*

José Carlos Metrôlho, Fernando Ribeiro, Ana Paula Silva, Arlindo Silva, Fernando Barbosa

Instituto Politécnico de Castelo Branco

Castelo Branco, Portugal

[metrolho@ipcb.pt](mailto:metrolho@ipcb.pt), [fribeiro@ipcb.pt](mailto:fribeiro@ipcb.pt), [dorian@ipcb.pt](mailto:dorian@ipcb.pt), [arlindo@ipcb.pt](mailto:arlindo@ipcb.pt), [fsergio@ipcb.pt](mailto:fsergio@ipcb.pt)

**Resumo** — Existe atualmente em Portugal uma forte procura de profissionais na área das Tecnologias da Informação e Comunicação. Esta procura exige que as Instituições de Ensino Superior preparem os seus diplomados adequadamente, não só tecnicamente, mas também ao nível de competências que são vulgarmente designadas por *soft skills*. Este artigo pretende descrever uma estratégia recentemente implementada, e ainda a decorrer, em dois cursos do ensino superior da área das Tecnologias da Informação e Comunicação. Nela adotaram-se práticas que acreditamos poderem ser conducentes a uma sólida formação dos alunos não só ao nível técnico, como também ao nível da aquisição de conhecimento e prática das melhores metodologias de trabalho em equipa. O feedback dos empregadores foi, e tem sido, bastante importante para o delinear desta estratégia. Isto porque nos permitiu aferir o grau de facilidade com que os recém-licenciados se integram nas empresas após serem recrutados, e os aspetos que a influenciam. Pudemos, assim, constatar que para uma integração com sucesso, contribuem, não só os conhecimentos e competências técnicas que estes evidenciam, mas também um conjunto de aspetos comportamentais relacionados com as relações interpessoais, motivação manifestada, atitude perante desafios, entre outras. A abordagem que propomos visa orientar a formação dos alunos finalistas no sentido de facilitar a sua integração num futuro ambiente de trabalho. O propósito deste artigo é partilhar esta experiência, que avaliamos como positiva, com aqueles que trabalhem nesta área, contribuindo assim para a tão desejada aproximação Academia-Empresas.

**Palavras Chave** - *Agile; projeto; ensino; empregabilidade; soft skills.*

**Abstract** — There is currently a strong demand for Information and Communication Technologies professionals in Portugal. This demand requires that higher education institutions prepare their graduates adequately, not only in technical skills, but also in terms of the so-called soft skills. In this paper, we describe a strategy that is being implemented, in recent years, in two higher education Information and Communication Technologies courses. In it, several practices were adopted that we believe will lead to a successful preparation of students, not only on a technical level, but also in terms of the acquisition of knowledge and best practices regarding best teamwork methodologies. Employers' feedback has been essential for the implementation of this strategy, since it has allowed us to evaluate how it facilitates the integration of recent graduates, after being recruited, and the issues that influence that integration. We can conclude that, for a successful integration, there is an important contribution, not only from the technical knowledge and abilities they acquired, but also from a diverse set of behavioral aspects related to interpersonal relations, motivation, attitude towards challenges, among others. The

approach we propose is aimed at guiding the training of finalist students in order to facilitate their integration into a future work environment. Our purpose with this paper is to share this experience, which we evaluate as positive, with those who work in the area and thus contribute to the desired Academy-Business proximity.

**Keywords** - *Agile; project; education; Employability; soft skills.*

### I. INTRODUÇÃO

Um dos principais objetivos de um curso do Ensino Superior é o de dotar os seus alunos de conhecimentos e competências que lhes permitam, após a conclusão do mesmo, obter emprego na respetiva área de formação e prosseguir com sucesso a sua atividade. Na área das Tecnologias da Informação e Comunicação (TIC), esta formação vai para além dos conhecimentos teórico-práticos fundamentais e das tecnologias tradicionais das respetivas áreas, sendo cada vez mais comum a inclusão, nos planos de estudos das diferentes unidades curriculares, de temas e tecnologias emergentes que resultam da investigação aplicada na área. É de facto um fator de extrema importância, neste tipo de cursos, a constante atualização dos currículos de forma a facilitar a integração dos respetivos diplomados no mercado e trabalho. Em muitos casos, as próprias empresas participam na elaboração dos planos de estudos, permitindo assim a criação de formações adaptadas às necessidades desse mercado.

Em Portugal, no caso dos cursos de licenciatura (seis semestres) em determinadas áreas, onde a velocidade de desenvolvimento tecnológico é elevada, nem sempre é possível cobrir todas as matérias e sobretudo todas as tecnologias. Assim, nestes casos, o foco será dotar os alunos de conhecimento e metodologias de trabalho que lhes permitam facilmente, após a conclusão do curso, adaptarem-se às exigências da entidade empregadora.

Para além da capacidade técnica e dos conhecimentos que os diplomados evidenciam (*hard skills*), as empresas valorizam, cada vez mais, aquilo que é geralmente designado por *soft skills*, ou competências não técnicas. Aqui, incluem-se várias qualidades que não estão relacionadas com os conhecimentos teórico-práticos adquiridos, mas antes com capacidades ao nível das relações interpessoais, habilidade na comunicação de ideias, atitude positiva perante desafios - aquilo que geralmente se designa por inteligência emocional e social do indivíduo.

Nos cursos de ensino superior, são várias as técnicas e metodologias aplicadas pelos professores de forma a evidenciar

e/ou reforçar as *soft skills* dos alunos. Esta diversidade resulta da opinião e interpretação que cada professor faz sobre este assunto. Há quem defenda que a autonomia do aluno deve ser privilegiada, outros que a autoafirmação em trabalho de equipa é mais importante, etc. Todos estes pontos de vista são relevantes e certamente que todos contribuem para o processo de amadurecimento intrínseco ao período de formação dos alunos. Para este processo, em muito contribuem a realização, sobretudo em grupo, de trabalhos práticos, e as respetivas apresentações e discussões, e a redação de relatórios técnicos. Estas atividades motivam a discussão de temas importantes para a formação dos alunos como são o caso do plágio e ética, cada vez mais pertinentes, considerando a abundância de informação que hoje em dia a Internet disponibiliza.

No entanto, como foi referido anteriormente, além dos conceitos fundamentais da área dos cursos, é especialmente importante dotar os alunos com capacidades de autonomia, autoconfiança e autoaprendizagem e metodologias de trabalho, que lhes permitam enfrentar a realidade do mercado de trabalho da melhor forma possível. Para esta vertente da formação, em muito contribuem as unidades curriculares de estágio ou projeto final de curso. Estas disciplinas, de duração semestral ou anual, devem colocar aos alunos finalistas desafios relacionados com áreas emergentes do ponto de vista tecnológico, mas também expô-los a métodos de trabalho semelhantes, o mais possível, aos que são usados por empresas da área.

Neste artigo será apresentada uma abordagem de ensino/aprendizagem que foi inicialmente pensada para as unidades curriculares de projeto de final de curso e que tem como objetivo contribuir para uma melhor formação dos alunos finalistas, preparando-os para responder positivamente ao que são os anseios das empresas empregadoras, facilitando, assim, a sua integração no mercado de trabalho.

## II. METODOLOGIAS DE DESENVOLVIMENTO DE SOFTWARE

As empresas que produzem software procuram acompanhar as tendências, tanto em termos de tecnologia como em termos de metodologias de desenvolvimento de software. Para isso, recrutam, ou formam internamente, os seus quadros técnicos para que estejam atualizados e certificados nas áreas mais relevantes para a sua atividade. No que diz respeito às metodologias de desenvolvimento de software, a conhecida metodologia *Waterfall*, ou Cascata, que dominou durante décadas na indústria [1], tem, desde a criação do Agile Manifesto, em 2001 [2], vindo a ser preterida por algumas empresas que optam por metodologias ágeis, tais como o Scrum [3], Extreme Programming (XP) [4], ou outras que seguem os princípios definidos naquele manifesto [2]. Estas novas metodologias ágeis elevam a importância dada à interação com o cliente e a recolha de feedback [1]. Além disso, estas metodologias, através de iterações curtas, possibilitam que o cliente receba regularmente novas funcionalidades, o que lhe permite aperceber-se do progresso do projeto e poder validar a direção que o mesmo esteja a seguir. Esta interação regular permite obter produtos de maior qualidade (resultado final correspondente ao definido como requisito inicial). Existem vários estudos comparativos que analisam as diferenças entre o uso da abordagem em Cascata e das abordagens ágeis, como é o caso de [5]. Estas metodologias ágeis são cada vez mais

ensinadas em ambiente académico, sendo fundamental que a sua aplicação em projetos de software aconteça em contexto de equipa. Em [6] é explicada com detalhe a metodologia Scrum, nomeadamente no que respeita aos acontecimentos ao longo do desenvolvimento (Sprint, Sprint Planning, Daily Scrum, Sprint Review, Sprint Retrospective), os papéis na equipa Scrum (Product Owner, Scrum Master e equipa de desenvolvimento), os artefactos (Product Backlog, Sprint Backlog), e os valores que devem ser seguidos pelos diferentes intervenientes.

No caso de estudo que apresentamos neste artigo, como concretização da abordagem que propomos, seguimos uma metodologia Scrum, com algumas necessárias adaptações, dado que a atividade se desenrola em ambiente académico. Esta escolha deve-se ao facto desta metodologia ser uma das mais populares atualmente, e pelo facto de várias empresas com quem temos colaborado, e que têm recrutado diplomados dos nossos cursos, seguirem estas práticas.

## III. A NOSSA ABORDAGEM

Nesta secção, iremos relatar a nossa abordagem para o caso de projeto de fim de curso, não considerando a vertente de estágio curricular. O caso de estudo que aqui apresentamos, diz respeito a unidades curriculares de projeto de final de curso dos cursos que ministramos na área das TIC. Esta é uma área em permanente desenvolvimento, o que exige uma atualização pensada e cuidada dos planos de estudo dos cursos de forma a que estes reflitam aquilo que são as melhores práticas e tecnologias. É particularmente importante nestes casos considerar a célere evolução de tecnologias e de ferramentas de desenvolvimento de software.

Tradicionalmente, nos cursos da área de TIC, é comum que o trabalho desenvolvido nas unidades curriculares de projeto de final de curso, seja realizado por um aluno, ou por um par de alunos, orientados por um ou dois professores. Esta prática também acontece na nossa instituição e há razões que o justificam. No entanto, na nossa opinião, embora nela possam ser identificados aspetos positivos, esta abordagem limita a aquisição de competências, por parte dos alunos, importantes para um bom desempenho naquilo que é geralmente designado por trabalho de equipa. A estratégia que propomos visa explorar aquilo que são as melhores práticas em termos de Engenharia de Software e surge como resultado de uma experiência relevante de contacto permanente com empresas empregadoras da área das TIC. Como é sabido, cada vez mais o trabalho em equipa é valorizado como forma de atingir alto rendimento, nomeadamente na área de desenvolvimento de software.

Assim, na orientação dos trabalhos realizados no âmbito das unidades curriculares de final de curso, decidimos experimentar uma abordagem diferente da tradicional que consistia em adotar uma metodologia de desenvolvimento de projeto do tipo *waterfall/cascata* [7] e uma orientação quase de um para um, entre professor-aluno (um orientador para um ou dois alunos). Uma grande maioria dos projetos orientados nestes moldes são direcionados para fins académicos, muitas vezes porque os objetivos têm que ser adaptados ao tamanho da equipa de trabalho, o que condiciona a dimensão e o tipo de projetos propostos. Optamos, por isso, por uma nova abordagem, que apesar de ser conduzida em ambiente académico, permite o desenvolvimento de trabalhos de maior dimensão e contribui

para uma formação dos alunos mais próxima daquilo que são os requisitos das empresas recrutadoras. Estas usam metodologias de desenvolvimento de software eficazes, sendo que algumas, com as quais temos um contacto mais direto, seguem metodologias ágeis [4], onde predomina o trabalho em equipa. Assim, para criar um ecossistema deste tipo, foi proposto um projeto comum a ser realizado por vários grupos de trabalho. Formalmente, a tipologia da orientação continua a ser semelhante à tradicional: grupos de dois alunos (ou apenas um aluno) com um docente a supervisionar os trabalhos. Contudo, cada grupo trabalha para o desenvolvimento de um projeto que é parte integrante de um sistema comum mais complexo e abrangente. A necessidade de articulação das várias partes em desenvolvimento determina um trabalho em equipa mais alargado, como iremos ver mais à frente. Desta forma, existe a possibilidade de preservar os aspetos positivos da proximidade aluno-professor da orientação tradicional, mas simultaneamente explorar as vantagens da realização de trabalho em equipas mais alargadas e próximas da realidade profissional. A aquisição de experiência na integração de software e a planificação de tarefas no âmbito de um universo mais alargado de intervenientes são algumas destas vantagens. Como já referimos, este trabalho em equipa permite também envolver os alunos em trabalhos mais ambiciosos e com maiores índices de complexidade.

Para implementar esta estratégia foi criada uma equipa de cinco professores, os quais definiram o tema e requisitos do projeto a ser implementado pelos vários alunos. Após a definição do problema a resolver, foram lançadas quatro propostas de trabalho que no seu todo compõem o projeto objetivo. Cada proposta pode ser realizada por um ou dois alunos e será supervisionada por um orientador e um coorientador. No total, o número máximo de alunos orientados não ultrapassa os oito e cada docente participa na orientação e/ou coorientação de alguns dos trabalhos. A Figura 1 ilustra o conceito da relação de atores e frações do trabalho.

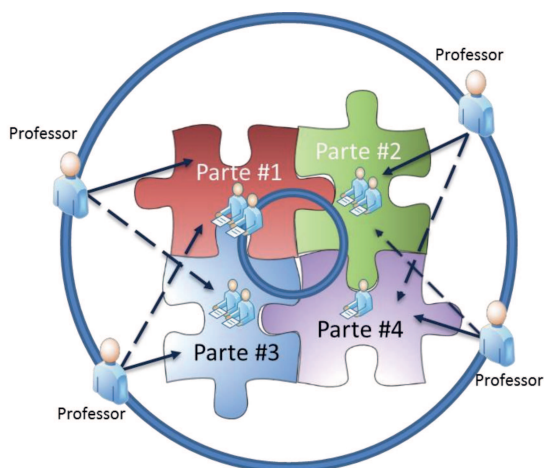


Figure 1. Relação entre atores.

Criam-se, assim, três dimensões de comunicação e interação entre os diferentes atores deste ambiente de desenvolvimento. Numa dimensão temos os supervisores (professores orientadores/coorientadores) que entre si comunicam com uma periodicidade quinzenal para fazer um ponto da situação do avanço do trabalho e atuam no seu conjunto como se de um

*Product Owner* em Scrum [6] se tratasse. São eles os *drivers* do trabalho e quem sabe em concreto os objetivos que devem ser alcançados em cada fase de desenvolvimento. Numa outra dimensão, temos a relação entre orientador/coorientador – aluno(s). Esta segunda dimensão garante um acompanhamento dedicado e explora aquilo que são as vantagens do tipo de orientação tradicional, as quais são também importantes para os alunos que nestas idades são naturalmente menos experientes. Finalmente, numa terceira dimensão, temos a interação entre os alunos dos diferentes projetos, quando expostos àquilo que são as características típicas de trabalho em equipa nesta área. Esta interação permite explorar aspetos importantes para a formação dos alunos, nomeadamente a necessidade de serem organizados, comunicativos e cumpridores de metas. Expõe, também, os alunos a uma certa concorrência entre grupos que os leva a quererem evidenciar-se no seio deste ambiente de desenvolvimento. Desta forma, a nossa abordagem elimina uma das desvantagens de equipas demasiado pequenas, e que já tivemos oportunidade de referir, não permitir projetos de elevada complexidade, mas também evita aquilo que são as desvantagens de equipas demasiado grandes.

Ao longo do desenvolvimento dos trabalhos é seguida uma metodologia similar ao Scrum [6], com necessárias adaptações. Assim, dado que tanto alunos como professores não estão dedicados a tempo inteiro ao desenvolvimento do projeto, os *sprints* são fixados em duas semanas e os *Daily Scrum* são substituídos por reuniões semanais envolvendo supervisor e aluno(s) de determinado grupo. Os supervisores (no seu conjunto) atuam como *product owners* e é suprimido o papel de Scrum Master [6]. As tarefas definidas inicialmente para serem desenvolvidas no projeto, *product backlog*, são definidas pelos supervisores e alunos numa reunião inicial conjunta, e os *sprint backlogs* são definidos no início de cada *sprint*, numa reunião conjunta também envolvendo todos os alunos e supervisores, para cada uma das partes do trabalho.

De seguida, apresentamos uma representação esquemática deste Scrum adaptado (ver Figura 2). Na parte superior do esquema podem ser observadas as fases de cascata e na parte inferior o Scrum adaptado.

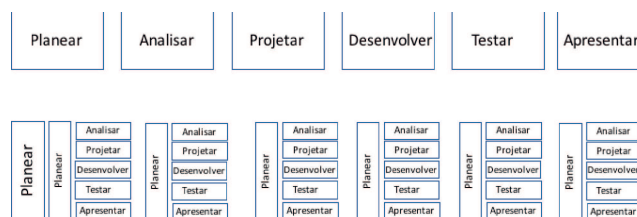


Figure 2. Cronograma de atividades ao longo do tempo.

#### IV. VANTAGENS E DESVANTAGENS

Tem-se constatado que com a metodologia tradicional (supervisor por cada aluno ou para de alunos) é frequente, ao fim de um semestre, os alunos não terem conseguido desenvolver um protótipo funcional do sistema proposto, limitando-se a apresentar o trabalho relativo às primeiras fases de desenvolvimento num método em cascata [7]. Nestes casos, é frequente os alunos apresentarem o estado da arte e a análise de requisitos e projeto (nomeadamente no que se refere à



modelação e indicação da arquitetura e ferramentas que serão usadas na implementação futura).

Com a abordagem proposta, verificou-se que os alunos, para além do que é habitual, conseguiram, no mesmo período de tempo, implementar parte do protótipo final. Além disso, e na nossa opinião talvez mais importante, os alunos revelaram um conhecimento coletivo daquilo que são os objetivos globais do projeto, uma noção clara, em termos de tempo e objetivos, das etapas a desenvolver e de que forma a sua equipa se integra com as restantes de forma a atingirem o objetivo comum. Outro aspeto que consideramos relevante é o facto desta abordagem permitir mesclar grupos de alunos com perfis diferentes num mesmo trabalho. Ou seja, os alunos dos diferentes grupos não têm que ser, necessariamente, todos do mesmo curso. Isto permite a integração de grupos de alunos de diferentes cursos, sendo assim possível expô-los a objetivos específicos do trabalho, focados nas matérias das formações que frequentam. É vantajoso para os alunos porque permite a partilha de informação, mas é também uma mais valia para o projeto cujas partes podem assim ser desenvolvidas por equipas com competências mais aprofundadas numa determinada área. Por exemplo, uma mesma equipa pode integrar grupos com competências mais dedicadas ao desenvolvimento de código, a questões de *storage*, a arquiteturas de cliente-servidor, ao desenvolvimento de interfaces e usabilidade, entre outras.

No entanto, aquela que nos parece ser a maior vantagem desta nova abordagem é o facto de conseguir expor os alunos a condições de trabalho muito semelhantes às que estes vão encontrar nas empresas onde irão trabalhar, quando terminarem o respetivo curso. Da experiência que adquirem, evidencia-se a capacidade de integração do trabalho desenvolvido, naquilo que é a aplicação completa para a qual trabalham. Esta competência de integração é muito valorizada pelos empregadores.

Quando se trabalha em equipa, pretende-se que o valor total seja superior à soma de todas as partes, mas para que tudo funcione corretamente o empenho e dedicação de todos os elementos da equipa deve ser semelhante. No caso da presente abordagem e devido à interligação e interdependência entre todos os membros, será necessário que todos produzam aquilo que se espera deles. Se alguns intervenientes não o fizerem poderão por em risco todo o projeto. Na versão tradicional, apenas um grupo seria afetado, mas nesta nova abordagem todos os grupos o serão. Por isso, é importante seguir as metodologias ágeis, as quais permitem avaliar se há atrasos em alguns dos componentes e tomar medidas que impeçam isso de acontecer.

Os trabalhos que se adaptam à aplicação desta metodologia devem possuir um conjunto de características que nem sempre pode ser fácil reunir. Como tivemos oportunidade de referir, para que esta abordagem possa ser útil para todos os intervenientes, o objetivo comum do trabalho deve ter uma complexidade e diversidade que justifique os diversos módulos. Estes devem ser geradores de volume de trabalho suficiente para tornar os objetivos parciais de cada suficientemente interessantes e motivadores para alunos e docentes. É fundamental haver pontos de interceção entre os trabalhos de forma a promover a interação entre os respetivos alunos e docentes e por essa via contribuir para a aprendizagem dos alunos. Por outro lado, a dependência entre módulos deve ser tal que caso haja insucesso na

implementação de um módulo, isso não comprometa demasiado os restantes.

Por último, estamos convencidos que a abordagem que aqui apresentamos, embora obrigue a uma disciplina e disponibilidade significativas por parte dos docentes, contribui para melhorar a formação dos alunos ao nível do trabalho em equipa e de engenharia de software, através de uma simulação em ambiente académico daquilo que será a experiência em ambiente empresarial.

## V. CONCLUSÃO

Neste artigo, pretendemos partilhar com os nossos pares uma abordagem que, na nossa opinião, poderá contribuir para melhorar a formação dos alunos finalistas e a sua integração no mercado de trabalho. Esta surge de uma experiência de vários anos a orientar projetos de final de curso e de um contato estreito com empresas da nossa área de conhecimento que nos permitiu perceber a sua forma de trabalhar e o que valorizam quando recrutam diplomados na área das TIC. Com base nesta experiência e também numa profunda reflexão sobre as anteriores metodologias de orientação, definimos a estratégia de orientação que aqui partilhámos e que estamos neste momento a implementar.

Embora esta experiência esteja apenas numa fase inicial, e se encontre ainda em curso, consideramos que os resultados conseguidos até agora são positivos e encorajadores. Nesta primeira fase, já pudemos verificar que os resultados apresentados pelos alunos em termos de volume de trabalho desenvolvido são superiores aos que tradicionalmente são apresentados pela abordagem tradicional. Estamos conscientes que isto se pode dever aos perfis dos alunos envolvidos e não ser exclusivamente resultado da abordagem seguida. Contudo, o contacto próximo entre alunos e orientadores, ao longo do desenvolvimento dos trabalhos, permitiu aos últimos observar um crescimento e amadurecimento dos alunos superiores ao habitual.

Finalmente, esta abordagem permitiu aos docentes propor o desenvolvimento de um projeto com uma dimensão significativamente superior aos tradicionais projetos de final de curso e com objetivos mais ambiciosos, tendo inclusivamente permitido já a redação de artigos na área do projeto proposto.

## REFERÊNCIAS BIBLIOGRÁFICAS

- [1] T. Palhoto, *Desenvolvimento Ágil de Software – Guia Prático*, 1.ª Edição. FCA Editora de Informática, Lda., 2016.
- [2] “Manifesto for Agile Software Development,” 2001. [Online]. Available: <http://www.agilemanifesto.org>. [Accessed: 17-Mar-2017].
- [3] K. Schwaber and J. Sutherland, “The Scrum Guide. The Definitive Guide to Scrum: The Rules of the Game,” 2016. [Online]. Available: <https://www.scrum.org>. [Accessed: 19-Mar-2017].
- [4] J. Shore and S. Warden, *The Art of Agile Development: Pragmatic Guide to Agile Software Development*. O'Reilly, 2007.
- [5] S. Palmquist, M. A. Lapham, S. Garcia-Miller, T. Chick, and I. Ozkaya, “Parallel Worlds: Agile and Waterfall Differences and Similarities,” Pittsburgh, PA, 2013.
- [6] C. Sims and H. L. Johnson, *SCRUM: A Breathtakingly brief and agile introduction*. Dymaxicon, 2012.
- [7] F. Tsui, O. Karam, and B. Bernal, *Essentials of Software Engineering*, 4th ed. Jones & Bartlett Learning, 2017.