



## Planeamento da Auditoria

Qualidade de Software

Blue Line

2019–2020

André Nicolau  
47880

Ana Pires  
55602

Sónia Costa  
55131

Pedro Costa  
55115

Pedro Lima Martins  
55100

Pedro da Nóbrega  
43580

## Conteúdo

<b>1 Purpose and scope of the audit</b>	<b>1</b>
<b>2 Audited organization</b>	<b>1</b>
<b>3 Software to be audited</b>	<b>1</b>
<b>4 Evaluation criteria</b>	<b>2</b>
<b>5 Auditor's responsibilities</b>	<b>3</b>
<b>6 Examination activities</b>	<b>3</b>
<b>7 Audit activity resource requirements</b>	<b>3</b>
<b>8 Audit activity schedule</b>	<b>4</b>
<b>9 Requirements for confidentiality</b>	<b>4</b>
<b>10 Checklists</b>	<b>5</b>
10.1 Processos de Software . . . . .	5
10.2 Especificação de Requisitos . . . . .	9
<b>11 Report formats</b>	<b>13</b>
<b>12 Report distribution</b>	<b>13</b>
<b>13 Follow-up Activities</b>	<b>13</b>
<b>14 Aprovação ou Reprovação do projecto</b>	<b>13</b>



## 1 Purpose and scope of the audit

A auditoria compreende os exames, análises, avaliações, levantamentos e comprovações, metodologicamente estruturados, para a avaliação da integridade, adequação, eficácia, eficiência e economicidade dos processos dos sistemas de informação e de controlos internos integrados ao ambiente, e de gestão de riscos, com a finalidade de dar apoio à administração da entidade no cumprimento dos seus objetivos.

A auditoria compreende: auditoria de processos e auditoria de requisitos.

A atividade da auditoria está estruturada em procedimentos com enfoque técnico, objetivo, sistemático e disciplinado, e tem por finalidade criar valor acrescentado à organização, apresentando contributos para o aperfeiçoamento dos processos, da gestão e dos controlos internos, por meio da recomendação de soluções para as não-conformidades apontadas nos relatórios.

A auditoria visa examinar um Sistema de Informação de Eventos (SIE) da Faculdade de Ciências da Universidade de Lisboa (FCUL), devendo estabelecer: a abrangência do exame (o que deve ser examinado); a oportunidade (quando deve ser realizado o exame); a extensão (a intensidade das verificações, o seu grau de detalhe, minúcia) e a qualidade dos exames (como devem ser realizados).

Assim, o principal objectivo desta auditoria de qualidade é avaliar se neste projecto de SIE as diversas equipas:

- Respeitam o processo de desenvolvimento
- Registam os defeitos encontrados
- Produzem as atas de reuniões
- Realizam as reuniões de revisões
- Realizam as documentações obrigatórias
- Envolvem clientes e utilizadores nos processos
- Atualizam o mapa de riscos do projecto

## 2 Audited organization

A organização a ser auditada será o grupo de alunos 012 da unidade curricular Projetos de Sistema de Informação da FCUL (ano lectivo 2013/2014).

## 3 Software to be audited

O alvo da auditoria é a documentação produzida relativa a um software que funciona como um sistema que permite gerir informações sobre eventos, e possibilita a estudantes e docentes a organizações de reuniões, seminários e outro tipo de eventos, nas instalações da FCUL. Esta documentação é composta por duas entregas: a primeira entrega é constituída por um relatório que retrata a fase de requisitos, o plano de projeto

e o mapa de Gantt. A segunda entrega é também composta por um relatório contendo eventuais melhorias ao primeiro já referido, com o acrescento da arquitectura de sistema, desenho do software, sequências de ecrãs e resultados de testes de software. Esta última entrega também inclui o Mapa de Gantt revisto.

## 4 Evaluation criteria

Nesta secção vamos definir os critérios de avaliação a ter em conta nesta auditoria, bem como a forma como os vamos classificar. Vamos basear-nos na secção 4.3 da norma IEEE 830-1998[1], nomeadamente:

- **correção** (escala de **1 a 3**): Considera-se que um requisito de software está correto se e só se está de acordo com os objetivos do software (neste caso, se está de acordo com o pedido no enunciado do projeto).
- **(não-)ambiguidade** (escala de **1 a 3**): Um requisito é não-ambíguo se e só se tiver apenas uma interpretação. Isto significa que o requisito, no produto final, deverá ser descrito por um único termo (que não se repete em outros requisitos).
- **completude** (escala de **1 a 3**): Um requisito considera-se completo se incluir:
  - todas as exigências significantes, quer se prendam com a funcionalidade, performance, limitações no design, atributos ou interfaces externas, bem como a forma como os requisitos externos impostos pelo sistema devem ser reconhecidos e tratados;
  - definições das respostas do software a todas as classes de input realizáveis em todas as situações realizáveis. Também têm que se especificar as repostas para inputs válidos e inválidos;
  - legendas e referências a todas as figuras, tabelas e diagramas do requisito, e definição de todas as condições e unidades de medida.
- **consistência** (escala de **1 a 3**): Refere-se à consistência interna, isto é, se um requisito de software está ou não em concordância com algum documento de alto-nível (por exemplo, requisitos do sistema).

Nestes três critérios acima classificamos como 1- requisito não cumprido e 3- requisito cumprido.

Também teremos em conta o critério de **aplicabilidade**, ou seja, se um requisito se aplica a este projeto de PSI - por exemplo, no caso de no enunciado do projeto enviado aos alunos não ser pedido algum dos requisitos, teremos que o considerar não aplicável (ao invés de não completo). Este critério será classificado de forma binária, ou seja **Sim** ou **Não**. caso não seja aplicável, o requisito não será avaliado por mais nenhum critério.

Para finalizar temos **severidade** (escala de **1 a 3**), onde iremos dar uma classificação final ao requisito desde 1(pouco severo) a 3(muito severo). Este critério será calculado de acordo com a média inversa dos outros critérios de avaliação. O objetivo deste critério é poder disponibilizar uma lista de observações (secção 9) onde se terá uma visão geral dos resultados desta auditoria.



## 5 Auditor's responsibilities

Neste projeto o grupo assume o papel de auditor, acumulando dois dos seus elementos a função de auditor líder ou auditor analista.

O auditor líder é responsável pela auditoria, o que inclui tarefas administrativas relativas à mesma, com o objetivo de garantir que será conduzida de maneira ordenada e que os seus objetivos serão cumpridos.

O analista deverá documentar anomalias, decisões e recomendações feitas pela equipa de auditoria.

É da responsabilidade dos auditores examinar produtos e relatar as suas observações, como definido no plano de auditoria.

Todos os auditores na equipa deverão fazer avaliações independentes e objetivas, livres de preconceitos e influências, ou então devem identificá-las e proceder com a aprovação do iniciador.

## 6 Examination activities

As atividades a desenvolver para a realização do relatório de auditoria são:

- preenchimento das checklists, com as devidas classificações, tendo por base o capítulo 4 do presente documento;
- elaboração do relatório de auditoria de acordo com os elementos da secção 8.7 da norma IEEE 1028-2008[2].

## 7 Audit activity resource requirements

Para a realização da auditoria será necessário um computador para a planificação da auditoria e papel/canetas. Para além disto, as normas: IEEE 1028-2008[2] para a preparação de gestão; IEEE 830-1998[1] para as especificações de requisitos de software; e a ISO/IEC/IEEE 12207-2008[3] para o estabelecimento de uma estrutura comum aos processos de ciclo de vida e de desenvolvimento de software. Serão ainda necessários os recursos fornecidos pela Prof. Paula Afonso através da plataforma Moodle, para a audição. Serão também necessários os relatórios que irão ser auditados.



## 8 Audit activity schedule

Nesta secção apresentamos um mapa da Gantt com a organização das atividades a realizar durante a auditoria. As atividades serão organizadas de acordo com a secção 8.5 da norma IEEE 1028-2008[2].

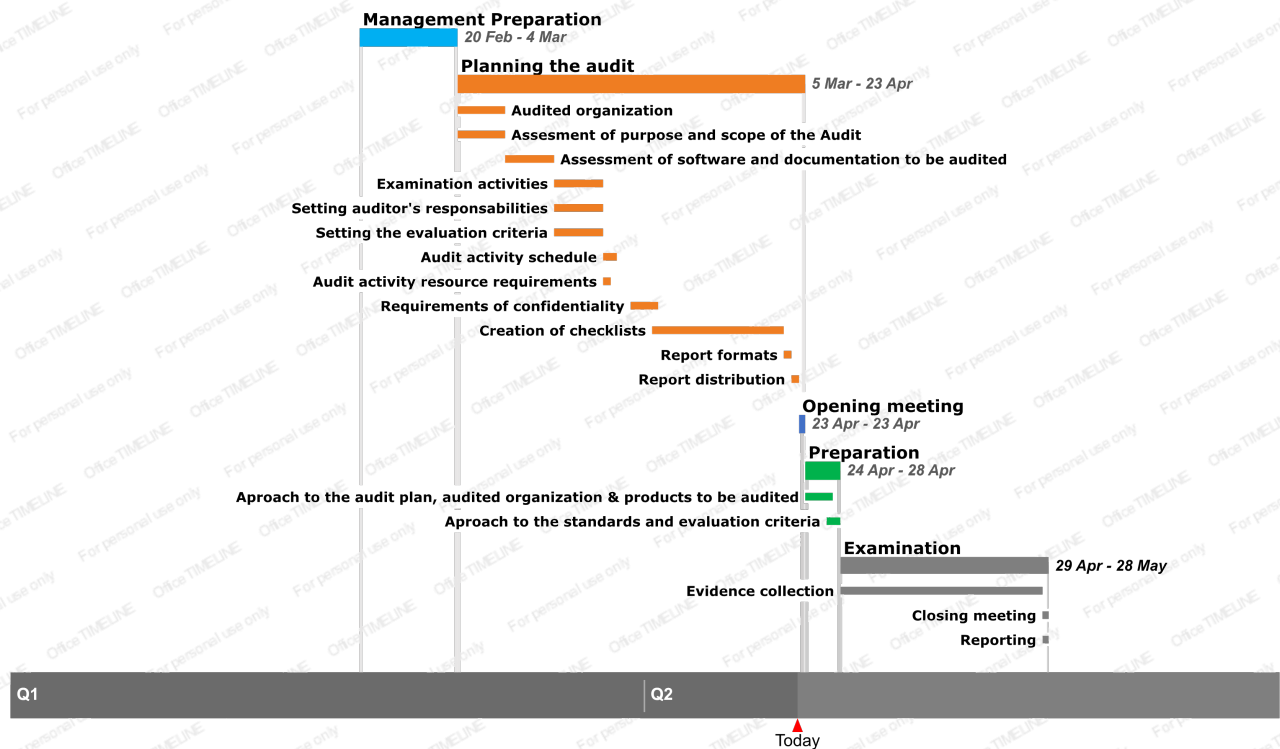


Figura 1:

## 9 Requirements for confidentiality

Este projeto contempla um acordo entre a nossa empresa e a organização a ser auditada, de modo a garantir a proteção e confidencialidade dos dados partilhados.



## 10 Checklists

Nesta secção serão descritas as checklists que serão usadas com o objetivo de auditar o projeto de software.

### 10.1 Processos de Software

As checklists desta secção têm como objetivo avaliar todos os processos envolvidos no desenvolvimento do software. Esta checklist está organizada de acordo com as secções 6 e 7 da norma IEEE 12207-2008[3]. Na tabela pode se ver os critérios, da esquerda para a direita: Aplicabilidade, Correção, Completude e Severidade.

Clausula	Processos/ Requisitos	Critérios de Avaliação				Observações
		Apl.	Corr.	Comp.	Sev.	
6	<b>System Life Cycle Processes</b>					
6.1	<b>Agreement Processes</b>					
6.2	<b>Organizational Project-Enabling Processes</b>					
6.2.1	<b>Life Cycle Model Management Process</b>					
6.2.1.3.1	Process establishment					
6.2.1.3.2	Process assessment					
6.2.1.3.3	Process improvement					
6.2.2	<b>Infrastructure Management Process</b>					
6.2.2.3.1	Process implementation					
6.2.2.3.2	Establishment of the infrastructure					
6.2.2.3.3	Maintenance of the infrastructure					
6.2.3	<b>Project Portfolio Management Process</b>					
6.2.3.3.1	Project initiation					
6.2.3.3.2	Portfolio evaluation					
6.2.3.3.3	Project closure					
6.2.4	<b>Human Resource Management Process</b>					
6.2.4.3.1	Skill identification					
6.2.4.3.2	Skill development					
6.2.4.3.3	Skill acquisition and provision					
6.2.4.3.4	Knowledge management					
6.2.5	<b>Quality Management Process</b>					
6.2.5.3.1	Quality management					
6.2.5.3.2	Quality management corrective action					
6.3	<b>Project Processes</b>					
6.3.1	<b>Project Planning Process</b>					



6.3.1.3.1	Project initiation					
6.3.1.3.2	Project planning					
6.3.1.3.3	Project activation					
6.3.2	<b>Project Assessment and Control Process</b>					
6.3.2.3.1	Project monitoring					
6.3.2.3.2	Project control					
6.3.2.3.3	Project assessment					
6.3.2.3.4	Project closure					
6.3.3	<b>Decision Management Process</b>					
6.3.4	<b>Risk Management Process</b>					
6.3.4.3.1	Risk management planning					
6.3.4.3.2	Risk profile management					
6.3.4.3.3	Risk analysis					
6.3.4.3.4	Risk treatment					
6.3.4.3.5	Risk monitoring					
6.3.4.3.6	Risk management process evaluation					
6.3.5	<b>Configuration Management Process</b>					
6.3.5.3.1	Configuration management planning					
6.3.5.3.2	Configuration management execution					
6.3.6	<b>Information Management Process</b>					
6.3.6.3.1	Information management planning					
6.3.6.3.2	Information management execution					
6.3.7	<b>Measurement Process</b>					
6.3.7.3.1	Measurement planning					
6.3.7.3.2	Measurement performance					
6.3.7.3.3	Measurement evaluation					
6.4	<b>Technical Processes</b>					
6.4.1	<b>Stakeholder Requirements Definition Process</b>					
6.4.1.3.1	Stakeholder identification					
6.4.1.3.2	Requirements identification					
6.4.1.3.3	Requirements evaluation					
6.4.1.3.4	Requirements agreement					
6.4.1.3.5	Requirement recording					
6.4.2	<b>System Requirements Analysis Process</b>					
6.4.2.3.1	Requirements specification					





6.4.2.3.2	Requirements evaluation					
6.4.3	<b>System Architectural Design Process</b>					
6.4.3.3.1	Establishing architecture					
6.4.3.3.2	Architectural evaluation					
6.4.4	<b>Implementation Process</b>					
6.4.5	<b>System Integration Process</b>					
6.4.5.3.1	Integration					
6.4.5.3.2	Test readiness					
6.4.6	<b>System Qualification Testing Process</b>					
6.4.6.3.1	Qualification testing					
6.4.7	<b>Software Installation Process</b>					
6.4.7.3.1	Software installation					
6.4.8	<b>Software Acceptance Support Process</b>					
6.4.8.3.1	Software acceptance support					
6.4.9	<b>Software Operation Process</b>					
6.4.9.3.1	Preparation for operation					
6.4.9.3.2	Operation activation and check-out					
6.4.9.3.3	Operational use					
6.4.9.3.4	Customer support					
6.4.9.3.5	Operation problem resolution					
6.4.10	<b>Software Maintenance Process</b>					
6.4.10.3.1	Process implementation					
6.4.10.3.2	Problem and modification analysis					
6.4.10.3.3	Modification implementation					
6.4.10.3.4	Maintenance review/acceptance					
6.4.10.3.5	Migration					
6.4.11	<b>Software Disposal Process</b>					
6.4.11.3.1	Software disposal planning					
6.4.11.3.2	Software disposal execution					
7	<b>Software Specific Processes</b>					
7.1	<b>Software Implementation Processes</b>					
7.1.1	<b>Software Implementation Process</b>					
7.1.1.3.1	Software implementation strategy					
7.1.2	<b>Software Requirements Analysis Process</b>					
7.1.2.3.1	Software requirements analysis					
7.1.3	<b>Software Architectural Design Process</b>					



7.1.3.3.1	Software architectural design					
7.1.4	<b>Software Detailed Design Process</b>					
7.1.5	<b>Software Construction Process</b>					
7.1.5.3.1	Software construction					
7.1.6	<b>Software Integration Process</b>					
7.1.7	<b>Software Qualification Testing Process</b>					
7.2	<b>Software Support Processes</b>					
7.2.1	<b>Software Documentation Management Process</b>					
7.2.2	<b>Software Configuration Management Process</b>					
7.2.3	<b>Software Quality Assurance Process</b>					
7.2.4	<b>Software Verification Process</b>					
7.2.4.3.1	Process implementation					
7.2.4.3.2	Verification					
7.2.5	<b>Software Validation Process</b>					
7.2.5.3.1	Process implementation					
7.2.5.3.2	Validation					
7.2.6	<b>Software Review Process</b>					
7.2.7	<b>Software Audit Process</b>					
7.2.8	<b>Software Problem Resolution Process</b>					
7.3	<b>Software Reuse Processes</b>					
7.3.1	<b>Domain Engineering Process</b>					
7.3.1.3.1	Process implementation					
7.3.1.3.2	Domain analysis					
7.3.1.3.3	Domain design					
7.3.1.3.4	Asset provision					
7.3.1.3.5	Asset maintenance					
7.3.2	<b>Reuse Asset Management Process</b>					
7.3.3	<b>Reuse Program Management Process</b>					



## 10.2 Especificação de Requisitos

Estão organizadas de acordo com as secções 4.3 (com os requisitos funcionais e não-funcionais listados pelos alunos nos próprios relatórios) e 5 da norma IEEE 830-1998[1], respectivamente.

Nas tabelas podem se ver os critérios Aplicabilidade, Correção, Não-Ambíguo, Completude, Consistência e Severidade.

Requisitos	Parâmetros de Análise					Observações
	Corr.	N-Amb.	Comp.	Cons.	Sev	
<b>Funcionais</b>						
<b>-Utilizador Estudante</b>						
Consultar os dados de um evento ao qual está convidado, incluindo a lista de convidados.						
Consultar a lista dos eventos e as datas de realização dos eventos aos quais está convidado, com possibilidade de imprimir esta lista.						
Indicar/Alterar a sua disponibilidade para um evento ao qual está convidado e se deseja que outros convidados possam saber se o utilizador vai ao evento ou não.						
Escrever comentários em eventos no qual é convidado.						
<b>-Utilizador Docente</b>						
Criar evento.						
Cancelar um evento que criou.						
Alterar os dados de um evento que criou.						
Seleccionar os convidados de um evento que criou.						
Adicionar um ficheiro a um evento que criou.						
Consultar a contagem das disponibilidades das pessoas para cada data admissível dos eventos que criou.						
Decidir a data e hora de realização dos eventos que criou.						
Consultar a listagem dos eventos e suas datas de realização alternativas ou decididas dos eventos do qual é criador.						

Indicar/Alterar a sua disponibilidade para um evento ao qual está convidado.						
Consultar os dados de um evento do qual é criador ou convidado.						
Imprimir a lista de eventos dos quais é criador ou convidado.						
Escrever comentários em eventos no qual é criador ou convidado.						
<b>Não Funcionais</b>						
É necessário um suporte informático mínimo (computador desktop ou portátil), com ligação a internet e um browser para aceder ao sistema.						
Os utilizadores devem ter os conhecimentos mínimos de informática na óptica do utilizador.						
O tempo de resposta do sistema deve ser suficientemente baixo para manter uma boa interação com o utilizador.						
Em caso de erro, o sistema deve recuperar um estado coerente de forma a que todas as acções que tinham sido realizadas no passado ainda sejam visíveis.						



Conteúdos		Ap.	Corr.	Comp.	Sev	Observações
Introduction	<b>Purpose</b>					
	Delineate the purpose of the SRS.					
	Specify the intended audience for the SRS.					
	<b>Scope</b>					
	Identify the software product(s) to be produced by name (e.g., Host DBMS, Report Generator, etc.).					
	Explain what the software product(s) will, and, if necessary, will not do.					
	Describe the application of the software being specified, including relevant benefits, objectives, and goals.					
	Be consistent with similar statements in higher-level specifications (e.g., the system requirements specification), if they exist.					
	<b>Definitions, acronyms, and abbreviations</b>					
	<b>References</b>					
	Provide a complete list of all documents referenced elsewhere in the SRS.					
	Identify each document by title, report number (if applicable), date, and publishing organization.					
	Specify the sources from which the references can be obtained.					
	<b>Overview</b>					
	Describe what the rest of the SRS contains.					
	Explain how the SRS is organized.					
	<b>Product perspective</b>					
	System interfaces					
	User interfaces					
	Hardware interfaces					
	Software interfaces					
	Communications interfaces					
	Memory					
	Operations					

**Overall  
description**



*10.2 Especificação de Requisitos*

**Specific  
requirements**

	Site adaptation requirements					
	<b>Product functions</b>					
	<b>User characteristics</b>					
	<b>Constraints</b>					
	<b>Assumptions and dependencies</b>					
	<b>Apportioning of requirements</b>					
	<b>External interfaces</b>					
	<b>Functions</b>					
	<b>Performance requirements</b>					
	<b>Logical database requirements</b>					
	<b>Design constraints</b>					
	<b>Standards compliance</b>					
	<b>Software system attributes</b>					
	<b>Organizing the specific requirements</b>					
	<b>Additional comments</b>					
<b>Supporting information</b>						
	Sample input/output formats, descriptions of cost analysis studies, or results of user surveys					
	Supporting or background information that can help the readers of the SRS					
	A description of the problems to be solved by the software					
	Special packaging instructions for the code and the media to meet security , export, initial loading, or other requirements.					

## 11 Report formats

Esta auditoria terá como outputs um Plano de Auditoria, um relatório de Auditoria e duas apresentações (correspondentes a cada relatório).

Os relatórios seguirão o mesmo formato, incluindo uma folha de rosto com a identificação da equipa responsável pela auditoria, seguida de uma tabela de conteúdos que inclui elementos da secção 8.5.2 da norma IEEE 1028-2008[2] (para o plano), e da secção 8.7 da norma IEEE 1028-2008[2] (para a auditoria em si). Todas as páginas a partir da tabela de conteúdos são paginas no rodapé e no cabeçalho, mencionando a secção correspondente e o logo da empresa **Blue Line**.

As duas apresentações seguirão o mesmo formato. Constituem capa, com título da fase de trabalho e nome da empresa, seguida de lista de conteúdos abordados. Todos os slides das apresentações serão paginadas, e no final uma página de agradecimento.

## 12 Report distribution

A distribuição dos dois relatórios, nomeadamente o Plano de Auditoria e a Auditoria, será feita através da página da Unidade Curricular Qualidade de Software no Moodle da Faculdade de Ciências da Universidade de Lisboa.

## 13 Follow-up Activities

Nesta segunda fase do relatório existem follow-up activities dos quais são:

\*\*\*(tenho esta nota do que a prof. disse: O follow up é basicamente o número de dias/horas para a equipa de projeto corrigir os erros - no segundo relatório quando detetarmos os erros planeamos quantos dias seriam necessários para que a equipa de projeto alterasse tudo o que dissemos que estava errado)\*\*\*

## 14 Aprovação ou Reprovação do projecto

Após uma avaliação do relatório quer da fase 1, quer da fase 2, a equipa BlueLine decidiu aprovar o projecto. Esta aprovação deve-se ao facto de no relatório constatar que o número total de funcionalidades que foram implementadas no sistema ser superior ao número total de funcionalidades planeadas pela equipa de alunos da cadeira de PSI (como se pode ver no relatório da fase 2 desde a Pág.2 até à Pág.). É de referir que os alunos de PSI não tiveram muito tempo para realizar o projecto.



## Referências

- [1] “IEEE Recommended Practice for Software Requirements Specifications,” *IEEE Std. 830-1998*, 1998.
- [2] “IEEE Standard for Modeling and Simulation,” *IEEE Std. 1028-2008*, 2008.
- [3] “Systems and software engineering — Software life cycle processes,” *IEEE Std. 12207-2008*, 2008.