



PERIVER PLATAFORMA

QUALITY ASSURANCE PLAN

Periver_QualityAssurancePlan_2008-02-28_v1.6.pdf

Versão 1.6

28 de Fevereiro de 2008

TABELA DE REVISÕES

Versão	Autores	Descrição da Versão	Aprovadores	Data
V1.0	João Gabriel Silva	Versão a utilizar pelos grupos de projecto de Engenharia de Software II, ainda sem a descrição dos métodos de teste.	-	29 de Março de 2004
V1.1	João Gabriel Silva	Acrescentada a descrição dos métodos de teste	-	18 de Abril de 2004
V1.2	João Gabriel Silva	Clarificadas as circunstâncias em que os erros detectados numa revisão ou inspecção são directamente anotados numa versão electrónica do documento, bem como o nome que esse ficheiro deve ter. Indicado que o moderador de uma revisão pode não aceitar algum dos erros assinalados pelos revisores. Clarificada a anotação da evolução do estado do documento nos relatórios de revisão e inspecção. Permitida a revisão apenas por uma pessoa, mas tendo isso de ser explicitamente indicado no Software Development Plan.	-	28 de Março de 2005
V1.3	João Gabriel Silva	Precisada a forma de atribuir nome aos ficheiros que contenham os relatórios de revisão.	-	5 de Abril de 2005
V1.4	João Gabriel Silva	Os testes de sistema passam a ser parte da especificação, o que resulta em diversos ajustes à fase do projecto em que são elaborados e a quem os elabora e aplica	-	20 de Março de 2006
V1.5	João Gabriel	Indicado nome do ficheiro que	-	2 de Abril de

	Silva	contem o relatório de inspecção, e os nomes a atribuir aos ficheiros quando o resultado de uma inspecção ou revisão é concatenado ou comprimido.		2006
V1.6	Luís Gomes	Adaptação do documento para o grupo GPSoft, alteração só do nome	-	28 de Fevereiro de 2008

ÍNDICE

Tabela de Revisões.....	2
Índice.....	4
1. Garantia de Qualidade	6
1.1 Âmbito	6
1.2 Aceitação formal.....	7
1.3 Aceitação final	7
2. Gestor de Qualidade	7
3. Inspeções.....	8
3.1 Introdução	8
3.2 Composição da Equipa	8
3.3 As fases	8
3.4 Modelo de Relatório de Inspeção.....	10
4. Revisões	13
4.1 Introdução	13
4.2 Composição da Equipa	13
4.3 As fases	13
4.4 Modelo de Relatório de Revisão.....	15
4.5 Modelo de Relatório de Síntese de Revisão	16
5. Unit Testing	19
6. Integration Testing.....	19
7. System Testing	20
7.1 Descrição dos testes	20

1. GARANTIA DE QUALIDADE

1.1 Âmbito

O Quality Assurance Plan define procedimentos para garantir a qualidade dos documentos produzidos no projecto. Note-se que o código também é considerado um documento, com a característica distintiva de ser escrito numa linguagem de programação, artificial, e não em linguagem natural.

Os procedimentos descritos são de dois tipos: aqueles que se aplicam a qualquer tipo de documento (incluindo código), e aqueles que se aplicam apenas a código.

Os procedimentos genéricos são:

- *Inspecções* - método formal muito rigoroso de revisão manual de documentos;
- *Revisões* - versão simplificada das inspecções, mas mantendo o carácter formal.

Os procedimentos específicos para o código são os seguintes:

- Testes unitários - testes aplicados pelo próprio programador a cada módulo de código por si produzido;
- Testes de integração - testes aplicados pelo próprio programador para verificar o correcto funcionamento do módulo que produziu quando associado a outros módulos já disponíveis;
- Testes de sistema - testes que cobrem toda a funcionalidade do software, a aplicar em diversas fases, no mínimo e obrigatoriamente a todo o software, no final, quando este estiver pronto.

Estes métodos aplicam-se em qualquer momento por decisão dos envolvidos, mas são obrigatórios em determinados momentos, nomeadamente aquando da aceitação formal e final de um documento. No Software Development Plan são indicados os procedimentos que devem ser aplicados para a aceitação formal e para a aceitação final de cada documento em concreto; aqui apenas são indicados requisitos mínimos.

Não é aqui especificado o papel exacto que o cliente deverá ter nos procedimentos descritos, pois isso depende muito das características específicas de cada cliente. A sua participação é indispensável no mínimo no processo de aprovação formal da especificação e na aceitação final do software desenvolvido, pelo menos. Idealmente deveria participar como inspector/revisor/testador de pleno direito nas inspecções/revisões/testes correspondentes, mas poderá ser necessário usar outro mecanismo. O que se deverá fazer em cada projecto concreto deverá ser especificado no respectivo Software Development Plan, mas terá sempre de haver uma assinatura formal de algum documento que concretize formalmente a aceitação do cliente da especificação e do produto final.

1.2 Aceitação formal

Um documento atinge o estado *baselined* assim que tenha sido submetido a uma inspecção ou revisão em que o resultado final seja a aceitação. Se se tratar de código, também terão tido de ser aplicados testes unitários, de integração e de sistema antes da revisão ou inspecção. Os testes de integração apenas poderão ser aplicados de forma parcial, se nem todos os módulos que seriam necessários para a sua aplicação plena estiverem disponíveis. Naturalmente, os testes de integração serão muito mais completos para os últimos módulos a serem desenvolvidos do que para os primeiros.

No caso de código, a sequência a seguir é a seguinte:

- testes unitários;
- testes de integração;
- testes de sistema (na parte aplicável);
- inspecção ou revisão.

1.3 Aceitação final

A versão entregue ao cliente é a última versão baselined. Pode haver condições adicionais (indicadas no Software Development Plan), por exemplo impostas pelo cliente. Este com frequência impõe um período de testes em condições reais, já nas instalações do cliente, aceitando o cliente o software apenas depois de passado esse período com sucesso. O "sucesso" nestas circunstâncias normalmente corresponde a não haver, no final desse período probatório, erros detectados que ainda estejam corrigidos.

No mínimo, a aceitação final está dependente da aplicação global dos testes de sistema ao software final. Como nessa altura os testes são aplicados a código já baselined, os erros detectados nessa altura deverão seguir o previsto no plano de controlo de alterações.

2. GESTOR DE QUALIDADE

Um dos membros da equipa de projecto terá as funções de Gestor de Qualidade.

O Gestor de Qualidade tem por função verificar que todos os procedimentos de qualidade são devidamente aplicados. Note-se que não é função do Gestor de Qualidade efectuar esses procedimentos; apenas verificar a sua correcta aplicação. Se da experiência prática resultar a necessidade de ajustar a definição de algum procedimento, o Gestor de Qualidade tem especial responsabilidade em promover esses ajustes, mas tal não é sua responsabilidade exclusiva, pois qualquer elemento da equipa o pode (e deve) também fazer.

3. INSPECÇÕES

3.1 Introdução

Uma inspecção é uma técnica de revisão manual de documentos. É uma técnica formal pois consiste num conjunto preciso de passos, aqui descritos, que têm de ser cuidadosamente aplicados. A sua vantagem essencial resulta de permitir uma boa detecção de erros numa fase inicial do desenvolvimento, em que o custo da respectiva correcção é muito mais baixo do que se forem descobertos apenas no final do projecto ou já em produção.

As inspecções não são um método de avaliação de desempenho. Quem se sente avaliado não participa com abertura e empenho na detecção de erros. As inspecções são um método de detecção de erros. Em consequência, os membros da equipa de inspecção devem ser pares do autor do documento inspecionado, e não superiores ou subordinados.

3.2 Composição da Equipa

Cada inspecção deverá ser efectuada por uma equipa composta por elementos que assumam os seguintes papéis:

Autor – Autor do documento inspecionado.

Moderador – Responsável pela coordenação do processo de inspecção.

Leitor – Na reunião de inspecção é ele quem lê, interpreta e justifica em voz alta o documento em análise, para suscitar da parte dos outros membros da equipa de inspecção o máximo de detecção de erros.

Anotador – Deve preencher o relatório da reunião de inspecção, apresentado na secção 3.4. Se se tiver optado por anotar os erros directamente no documento em inspecção, deverá fazê-lo directamente na versão electrónica do documento, num formato que permita uma fácil localização dessas anotações.

Inspectores – Todos os membros da equipa de inspecção, quer tenham quer não tenham alguma das funções atrás enunciadas.

O número mínimo de elementos de uma equipa de inspecção são três: o moderador (que nessa circunstância também executa as funções de anotador), o leitor e o autor.

3.3 As fases

Uma inspecção passa pelas seguintes fases:

Planeamento – O autor começa por escolher um moderador. O moderador, que a partir daí conduz todo o processo de inspecção, começa por verificar se há condições para fazer a inspecção, nomeadamente:

- se o documento a inspeccionar não apresenta sinais evidentes de preparação descuidada, pois as inspeções não podem ser entendidas como uma forma de injectar qualidade em algo que foi mal construído. Um documento preparado sem cuidado deve ser liminarmente devolvido ao autor;

- se toda a informação necessária para proceder à inspecção está reunida. Por exemplo, para inspeccionar um documento de design é indispensável que o documento de arquitectura e a especificação correspondente estejam baselined e disponíveis. Tratando-se de código que utiliza serviços de outro módulo de software, tal como uma biblioteca gráfica ou um sistema de gestão de bases de dados, a especificação exacta desses serviços tem de estar disponível para todos os inspectores;

- se o documento a inspeccionar está em formato electrónico que permita anotar no próprio ficheiro os erros detectados, ou se existe uma maneira simples de identificação do local onde foi detectado um erro (por exemplo por numeração das linhas de um ficheiro de código), caso a anotação dos erros seja feita apenas no relatório de revisão.

O moderador, em conjunto com o autor, escolhe então os outros membros da equipa de inspecção, e em concertação com eles marca as datas e locais para a sessão de apresentação e a reunião de inspecção propriamente dita.

Reunião de Apresentação – Nesta reunião os membros da equipa de inspecção recebem o documento a inspeccionar e os outros documentos de suporte. O autor apresenta então a utilidade de cada um dos documentos de suporte, e descreve com detalhe a estrutura do documento a inspeccionar. No final da reunião todos os membros da equipa de inspecção devem conseguir navegar bem no documento a inspeccionar.

Preparação – Os inspectores analisam individualmente o documento em inspecção. Todos os detalhes devem ser escrutinados e, embora o objectivo não seja ainda uma detecção exaustiva de erros mas sim uma completa contextualização e compreensão do documento, todos os erros detectados devem ser anotados.

Reunião de Inspeção – Nesta reunião o leitor lê, interpreta e explica o documento. Os outros inspectores seguem e interpelam-no sempre que não perceberem ou não concordarem com alguma coisa.

O autor apenas intervém se solicitado pelo moderador, que o fará quando entender que os outros inspectores não conseguem entender a razão de ser de algum aspecto do documento, mesmo após algum debate entre eles. Apenas se encontrar um erro pode o autor intervir por sua iniciativa, assinalando-o para o anotador o registar.

O moderador deve manter a reunião sempre bem focada na detecção de erros e rejeitar liminarmente qualquer tentativa de discutir alternativas de implementação. Nem sequer se devem discutir as formas de corrigir os erros detectados - estes devem ser anotados apenas. Em caso de impasse nalguma situação, por exemplo quanto a classificar algo como erro ou não, o moderador tem voto de qualidade. Dada a grande concentração com que uma reunião de inspecção deve decorrer, não deverão ser feitas reuniões superiores a cerca de duas horas.

Terminada a reunião, existem 3 veredictos possíveis:

A – Aceitação – O documento não contém erros.

C – Algumas correcções – O documento apresenta pequenos defeitos cuja correcção pode ser verificada apenas pelo moderador, sem necessidade de reunir a equipa de novo.

R – Re-inspeccionar - O documento tem defeitos que resultarão em alterações significativas que requerem uma reinspecção. Essa reinspecção pode ser parcial ou total, em função da extensão das zonas alteradas pelas correcções. Cabe ao moderador decidir entre estas duas alternativas.

Correcção – O autor introduz as alterações necessárias para corrigir os erros detectados.

Acompanhamento – O moderador verifica se todos os erros detectados foram corrigidos. Caso verifique que a extensão das alterações introduzidas diferiu significativamente do previsto no final da reunião de inspecção, poderá mudar o veredicto de C para R ou vice-versa, bem como decidir em caso de R se deverá ser feita uma reinspecção total ou apenas parcial. A reinspecção, naturalmente, começa directamente na fase de preparação, ou até na fase de reunião de inspecção (cabendo ao moderador decidir) não sendo naturalmente necessária nenhuma nova reunião de apresentação.

3.4 Modelo de Relatório de Inspeção

Em cada reunião de inspecção ou reinspecção deverá ser preenchido o modelo a seguir, e colocado no repositório junto do documento inspeccionado. O nome do ficheiro contendo o relatório deverá ser igual ao nome do ficheiro inspeccionado acrescentado de "_relinsp_data", sendo "data" a data em que a reunião de inspecção terminou.

A tabela de descrição dos erros detectados pode ser deixada em branco, se os erros forem anotados electronicamente pelo anotador no documento em inspecção (o cabeçalho do relatório de inspecção tem sempre de ser preenchido). Nesse caso, o documento anotado (mas ainda não corrigido) deverá ser arquivado no repositório em conjunto com o relatório da inspecção, sendo o nome do ficheiro que contém as correcções igual ao do ficheiro original acrescentado de "_insp_data", sendo "data" a data em que a reunião de inspecção terminou, devendo ainda ser indicado na primeira linha da lista de erros do relatório de inspecção que os erros estão anotados nesse ficheiro, indicando também qual o respectivo nome. São possíveis soluções mistas, em que alguns erros são anotados no formulário a seguir, e outros na versão electrónica do documento. Caso o ficheiro anotado seja concatenado no mesmo ficheiro com o relatório de inspecção (ficando sempre o relatório em primeiro lugar), o nome do ficheiro resultante será o mesmo que teria se apenas contivesse o relatório de inspecção. Caso o relatório de inspecção e o ficheiro anotado sejam aglomerados no mesmo ficheiro comprimido, o nome do ficheiro comprimido será também o do relatório de inspecção (naturalmente com a extensão correspondente ao tipo de compressão usada, p.ex. "zip").

Após a introdução das correcções o moderador deverá indicar o seu veredicto final, se a avaliação final da reunião de inspecção for C. Caso a avaliação final da reunião de inspecção seja

R esse veredicto final não deverá ser preenchido, (a menos que perante a correcção de erros o moderador decida alterá-lo para A) pois o processo de revisão recomeça após a correcção dos erros decretados, sendo produzido novo relatórios de inspecção.

Relatório de Inspeção

Documento inspeccionado:

Avaliação final da reunião de inspeção: A-C-R

Data da reunião de apresentação:

Data da reunião de inspeção:

Duração da reunião de inspeção:

Veredicto após correcção dos erros, e data em que o moderador tomou a decisão: A-R - data

Papel	Elementos do Grupo
Moderador	
Autor	
Leitor	
Anotador	
Outros Inspectores	

Nº	Localização do erro	Descrição do erro
1		
2		
3		
4		
...		

4. REVISÕES

4.1 Introdução

As revisões são uma versão simplificada das inspecções, a aplicar a documentos menos críticos do que aqueles onde as inspecções são aplicadas. Trata-se também de uma técnica de análise manual de documentos, mas que não inclui a reunião de inspecção, baseando-se antes na revisão solitária feita pelos revisores (correspondente à fase de preparação das inspecções). No restante é muito próxima das inspecções; como nelas um aspecto essencial é o registo do resultado da revisão.

4.2 Composição da Equipa

Cada revisão deverá ser efectuada por uma equipa composta por elementos que assumam os seguintes papéis:

Autor – Autor do documento inspeccionado.

Moderador – Responsável pela coordenação do processo de revisão, e pelo preenchimento do relatório síntese da revisão, apresentado na secção 4.5.

Revisores – Todos os membros da equipa de revisão com excepção do autor, incluindo portanto o moderador. Todos devem preencher o relatório de revisão apresentado na secção 4.4.

Pode ser feita uma revisão apenas pelo moderador e autor, em casos mais simples. Nesses casos bastará preencher o relatório síntese da revisão, apresentado na secção 4.5. A aplicação de tal método simplificado de revisão terá sempre de ser resultado de uma decisão explícita, a indicar claramente no Software Development Plan.

4.3 As fases

Uma revisão passa pelas seguintes fases:

Planeamento – O autor começa por escolher um moderador. O moderador, que a partir daí conduz todo o processo de revisão, começa por verificar se há condições para fazer a revisão, nomeadamente:

- se o documento a rever não apresenta sinais evidentes de preparação descuidada, pois as revisões não podem ser entendidas como uma forma de injectar qualidade em algo que foi mal construído. Um documento preparado sem cuidado deve ser liminarmente recusado;

- se toda a informação necessária para proceder à revisão está reunida. Por exemplo, para inspeccionar um documento de design é indispensável que o documento de arquitectura e a especificação correspondente estejam baselined e disponíveis. Tratando-se de código que utiliza serviços de outro módulo de software, tal como uma biblioteca gráfica ou um sistema de gestão

de bases de dados, a especificação exacta desses serviços tem de estar disponível para todos os revisores;

- se o documento a inspecionar está em formato electrónico que permita anotar no próprio ficheiro os erros detectados, ou se existe uma maneira simples de identificação do local onde foi detectado um erro (por exemplo por numeração das linhas de um ficheiro de código), caso a anotação dos erros seja feita apenas no relatório de revisão.

O moderador, em conjunto com o autor, escolhe então os outros membros da equipa de revisão, e em concertação com eles marca a data para a reunião de apresentação.

Reunião de Apresentação – Nesta reunião os membros da equipa de revisão recebem o documento a rever e os outros documentos de suporte. O autor apresenta então a utilidade de cada um dos documentos de suporte, e descreve com detalhe a estrutura do documento a rever. No final da reunião todos os membros da equipa de revisão devem conseguir navegar bem no documento a rever.

Revisão – Os revisores analisam individualmente o documento em revisão. Todos os detalhes devem ser escrutinados e feita uma procura metódica de erros. Todos estes deverão ser devidamente anotados, ou como comentários bem identificáveis na versão electrónica do próprio documento a ser revisto, ou num formulário como o apresentado na secção 4.4, e o resultado da revisão enviado ao moderador. Mesmo que os problemas detectados sejam anotados no próprio documento, o formulário da secção 4.4 deverá ser sempre preenchido, indicando apenas na primeira linha da lista de erros que estes estão anotados no ficheiro original, indicando também qual o nome do ficheiro onde essas correcções estão introduzidas. São possíveis soluções mistas, em que alguns erros são anotados no formulário da secção 4.4, e outros na versão electrónica do documento.

O nome do ficheiro com as anotações deve ser igual ao do ficheiro original, acrescido de "_rev_XYZ_data" sendo "XYZ" as iniciais do revisor. Assim, se o ficheiro em revisão tinha por exemplo o nome "Proj_NomeDocumento_2005-03-03_v0.8r.doc", o ficheiro com as anotações resultantes da revisão feita por Manuel Alves Redol a 5 de Março de 2005 deverá ter o nome "Proj_NomeDocumento_2005-03-03_v0.8r_rev_MAR_2005-03-05.doc"

Resultado da Revisão – O moderador analisa o resultado do trabalho dos vários revisores, tendo em conta as recomendações dadas nos respectivos relatórios de revisão (ver secção 4.4) e produz o relatório síntese de revisão (ver secção 4.5). O moderador pode discordar de alguns dos erros detectados pelos outros revisores, por considerar que não se trata de erros, mas de sugestões de alternativas ou de diferenças de opinião. Sempre que isso ocorra, o moderador tem de fornecer ao autor, ou a quem for fazer as correcções, uma descrição exacta de quais os erros a corrigir, o que pode envolver copiar para o relatório síntese apenas parte dos erros detectados pelos revisores. Esta cópia das listas de erros para o relatório síntese apenas é necessária se houver alteração da lista de algum dos revisores. Se as anotações foram feitas electronicamente no documento original, o moderador deverá retirar as anotações que não aceitou, sendo importante ter em conta que esta retirada tem sempre de ser feita numa cópia do ficheiro revisto pelo revisor em causa. O documento fornecido pelo revisor tem sempre de

ser mantido inalterado no repositório. A cópia alterada pelo moderador deverá ter o nome do ficheiro fornecido pelo revisor, acrescido de "_rev_XYZ_data" sendo "XYZ" as iniciais do moderador.

Em qualquer caso, o relatório síntese de revisão tem sempre de identificar claramente exactamente quais os ficheiros e/ou relatórios de revisão onde estão assinalados todos os erros a corrigir.

Tanto os revisores como o moderador podem dar três veredictos:

A – Aceitação – O documento não contém erros.

C – Algumas correcções – O documento apresenta pequenos defeitos cuja correcção pode ser verificada apenas pelo moderador, sem necessidade de re-revisão por todos os revisores.

R – Re-rever - O documento tem defeitos que resultarão em alterações significativas que requerem uma re-revisão. Essa re-revisão pode ser parcial ou total, em função da extensão das zonas alteradas pelas correcções. Cabe ao moderador decidir entre estas duas alternativas.

Correcção – O autor introduz as alterações necessárias para corrigir os erros detectados.

Acompanhamento – O moderador verifica se todos os erros detectados foram corrigidos. Caso verifique que a extensão das alterações introduzidas diferiu significativamente do previsto, poderá mudar o veredicto de C para R ou vice-versa, bem como decidir em caso de R se deverá ser feita uma re-revisão total ou apenas parcial. A re-revisão, se necessária, começa directamente na fase de "Revisão", não sendo naturalmente necessária nenhuma nova reunião de apresentação.

4.4 Modelo de Relatório de Revisão

Cada revisor deverá preencher, ao rever um documento, o modelo a seguir.

Nota - A tabela de descrição dos erros detectados pode ser deixada em branco, se os erros forem anotados electronicamente pelo revisor no documento em revisão (o cabeçalho do relatório de revisão tem sempre de ser preenchido). Nesse caso, o documento anotado deverá ser arquivado no repositório em conjunto com o relatório de revisão. O nome do ficheiro que contem este relatório de revisão deverá ser igual ao nome do ficheiro alvo de revisão, acrescentado de "_relrev_XYZ_data" sendo "XYZ" as iniciais do revisor, e "data" a data em que o revisor terminou a revisão. Caso o ficheiro anotado seja concatenado num mesmo ficheiro com o relatório de revisão (ficando sempre o relatório em primeiro lugar), o nome do ficheiro resultante será o mesmo que teria se apenas contivesse o relatório de revisão. Caso o relatório de revisão e o ficheiro anotado sejam aglomerados no mesmo ficheiro comprimido, o nome do ficheiro comprimido será também o do relatório de revisão (naturalmente com a extensão correspondente ao tipo de compressão usada, p.ex. "zip").

Relatório de Revisão

Documento revisto:

Proposta de avaliação final da revisão: A-C-R

Data de conclusão da revisão:

Nome do Revisor:

Nº	Localização do erro	Descrição do erro
1		
2		
3		
4		
...		

4.5 Modelo de Relatório de Síntese de Revisão

Após a recepção dos relatórios de cada um dos revisores, o moderador preenche o modelo seguinte, devendo após a introdução das correções indicar o seu veredicto final, se a avaliação final era C. Caso a avaliação final fosse R o processo de revisão recomeça após a correção dos erros decretados, com novos relatórios de revisão. O nome do ficheiro que contem este relatório de revisão deverá ser igual ao nome do ficheiro alvo de revisão, acrescentado de "_relsintrev_ XYZ_data" sendo "XYZ" as iniciais do moderador, e "data" a data em que o moderador terminou a síntese.

Caso o relatório síntese de revisão e os outros relatórios e ficheiros anotados sejam aglomerados no mesmo ficheiro comprimido, o nome do ficheiro comprimido será também o do relatório síntese de revisão (naturalmente com a extensão correspondente ao tipo de compressão usada, p.ex. "zip").

Relatório de Síntese de Revisão

Documento revisto:

Avaliação final: A-C-R

Data da reunião de apresentação:

Data do preenchimento inicial deste relatório:

Veredicto após correcção dos erros, e data em que o moderador tomou a decisão: A-R - data

Papel	Elementos do Grupo
Moderador	
Autor	
Outros Revisores	

Observações justificativas para a avaliação final

Identificação exacta da localização das listas de erros a corrigir

Nº	Localização do erro	Descrição do erro
1		
2		
3		
...		

5. UNIT TESTING

Os testes unitários são executados pelo autor do código em cada módulo por si produzido. Entende-se aqui por "módulo" uma porção de código que seja programada de uma só vez, tal como por exemplo uma rotina ou pequeno conjunto de rotinas, ou uma classe ou pequeno conjunto de classes. O módulo deve ser testado quer em circunstância normais, quer em circunstâncias limite, quer em circunstâncias ilegais. Deve haver especial cuidado em garantir que o código de gestão de situações excepcionais seja também integralmente testado.

É importante que, tratando-se p.ex. de mais do que uma rotina, ao testá-las em conjunto, e havendo alguma rotina que apenas seja chamada nos testes por via indirecta (isto é, os testes não a chamam directamente, mas apenas através de outras rotinas que a usam) esse facto não limite a abrangência dos testes que lhe são aplicados. Se ao chamar essa rotina por via indirecta não for possível testá-la p.ex. em todas as situações limite dos valores dos parâmetros, então essa rotina deve ser também testada directamente.

Os testes unitários são testes informais, não sendo obrigatório ficarem registados. É naturalmente aconselhável que sejam guardados pelo menos os pequenos bocados de código que é em regra necessário desenvolver para chamar o código sob teste, para permitir reexecutar os testes na sequência de alguma alteração posterior. Embora não tenham que ser registados, é condição de submissão do código a aprovação formal que eles tenham sido devidamente executados, pelo que a entrega para aprovação formal de um pedaço de código implica da parte do respectivo autor o compromisso de os ter de facto executado.

6. INTEGRATION TESTING

Os testes de integração são executados pelo autor do código para verificar o correcto funcionamento do módulo que produziu quando este é integrado com outros módulos já disponíveis. O módulo deve ser testado quer em circunstância normais, quer em circunstâncias limite, quer em circunstâncias ilegais. Deve haver especial cuidado em garantir que o código de gestão de situações excepcionais seja também integralmente testado.

É importante que os testes incluam todas as variações possíveis de interacção entre o código novo e o código já existente com o qual está a ser feita a integração, particularmente no que diz respeito ao tratamento de situações excepcionais.

Os testes de integração são testes informais, não sendo obrigatório ficarem registados. É naturalmente aconselhável que sejam guardados pelo menos os pequenos bocados de código que é em regra necessário desenvolver para chamar o código sob teste, para permitir reexecutar os testes na sequência de alguma alteração posterior. Embora não tenham que ser registados, é condição de submissão do código a aprovação formal que eles tenham sido

devidamente executados, pelo que a entrega para aprovação formal de um pedaço de código implica da parte do respectivo autor o compromisso de os ter de facto executado.

7. SYSTEM TESTING

Estes são testes formais, isto é, deve ficar devidamente registada quer a definição dos testes quer o resultado da sua aplicação (o registo do resultado dos testes é obrigatório na fase de Aceitação Final). São testes caixa preta, que espelham apenas a especificação externa, no desconhecimento do código que concretiza essa especificação (que aliás em regra ainda nem sequer existe quando são definidos).

Os testes de sistema têm de ser definidos de maneira a cobrir integralmente a funcionalidade da aplicação em causa, incluindo as situações de erro. Os testes não podem provar a ausência de erros, apenas a sua presença. Os casos de teste devem por isso procurar deliberadamente sujeitar o programa às situações mais difíceis, sem naturalmente esquecer que toda a funcionalidade normal tem de ser devidamente testada.

Os testes de sistema vão sendo aplicados informalmente pelos programadores ao código que for sendo desenvolvido, podendo ser necessário adaptá-los quando ainda não existem todos os módulos.

Na fase de Aceitação Final os testes de sistema são aplicados de forma sistemática, eventualmente em paralelo por subequipas, todas elas independentes de quem programou a parte da aplicação em causa. Como são aplicados a um programa já baselined, os erros que estes testes detectarem devem ser corrigidos seguindo os procedimentos previstos no Change Control Plan para alterações a documentos já baselined. De uma forma geral, sempre que um problema for detectado numa parte da aplicação já baselined, o procedimento de controlo de alterações tem de ser seguido.

7.1 Descrição dos testes

Os documentos de descrição dos testes devem conter dois tipos de informação relativa aos testes.

Em primeiro lugar os critérios de cobertura. Sendo impossível proceder a testes exaustivos em programas não triviais, os testes a executar devem ser um subconjunto tão representativo quanto possível do conjunto dos testes possíveis. Os critérios usados para estabelecer essa representatividade, os chamados "critérios de cobertura", devem ser explicitamente listados no início do documento de descrição de testes.

Em segundo lugar deverão ser descritos os testes propriamente ditos, em formato tabular, sendo indicada para cada teste a seguinte informação:

- número do teste - esta numeração sequencial permite referenciar o teste com facilidade, sempre que necessário;
- condições de execução - descreve a situação em se deverá encontrar o programa quando o teste é aplicado, tais como conteúdo de eventuais bases de dados, estado de ligações de rede, parâmetros de lançamento de programa, etc. Em vez de indicar estas condições para cada teste poderão os testes que partilham as mesmas condições de execução serem agrupados numa sub-tabela, sendo antes dessa sub-tabela indicadas essas condições, bem como se os testes nela contidos deverão ser executados depois de recolocar as condições de partida no ponto indicado ou se devem ser executados em sequência, sendo portanto as condições de saída de um teste as condições de entrada do teste seguinte.
- entradas - valores de entrada a fornecer ao programa para aplicar o teste.
- saídas esperadas - saídas que o programa deverá produzir, caso se comporte de acordo com a especificação.
- referência ao protótipo e/ou Additional Spec Doc - indica qual o ecrã do protótipo ou a parte do Additional Spec Doc a que o teste diz respeito. Este campo ajuda a verificar a completude dos testes. Se alguma parte do protótipo ou do Additional Spec Doc não surgir, é porque o conjunto de casos de teste é incompleto.
- resultado - este campo apenas é preenchido quando o teste é aplicado, e indica se o programa se comportou como previsto, ou se houve algum desvio.