

Testes de Aceitação para Verificação de Entregas de Parceiros

Tatiana Dutra Borghi¹, Paula Gonçalves Povoas¹, Gizelle Sandrini Lemos²

¹Samsung R&D Institute Brazil (SRBR-M)
Av. dos Oitis, s/n – Distrito Industrial I – CEP 69007-002 – Manaus/AM – Brazil

²Samsung R&D Institute Brazil (SRBR-C)
Av. Cambacicas, 1200 – Parque dos Resedás – CEP 13097-104 – Campinas/SP – Brazil

{t.borghi,paula.povoas,g.lemos}@samsung.com

Abstract. *Samsung R&D Institute Brazil (SRBR) is one of many Samsung research centers in the world. SRBR team has worked in collaboration with Samsung headquarter (located at South Korea) and Brazilian outsourcing partners to produce software for Samsung products. SRBR team incorporated acceptance test on its software development process in order to assure that all software delivered by partners to SRBR complies with Samsung corporate process and SRBR requirements. This experience report presents the result of acceptance test activity incorporated on SRBR software development process and includes the result of acceptance tests in three projects developed in Manaus city (Brazil) by SRBR with partners participation.*

Resumo. *Samsung R&D Institute Brazil (SRBR) é um dos institutos de pesquisa da Samsung no mundo. O time do SRBR vem trabalhando em colaboração com a matriz da Samsung (localizada na Coreia do Sul) e instituições parceiras do Brasil para produzir software para produtos Samsung. O SRBR incorporou testes de aceitação em seu processo de desenvolvimento de software para assegurar que todo software produzido por parceiros do SRBR esteja de acordo com o processo corporativo da Samsung e requisitos propostos pelo SRBR. Este relato de experiência apresenta o resultado de testes de aceitação em três projetos desenvolvidos em Manaus com participação de parceiros.*

1. Introdução

SRBR é o instituto de pesquisa da Samsung no Brasil. Ele conta com times internos e parceiros para o desenvolvimento de projetos requisitados pela matriz na Coreia do Sul e por times de marketing da América Latina. A matriz na Coreia definiu um processo de desenvolvimento de software corporativo (PDSC) para regular todas as fases do ciclo de vida de desenvolvimento.

O governo brasileiro definiu leis que preveem exceção no pagamento de taxas para companhias que realizam pesquisa e desenvolvimento envolvendo parceiros brasileiros (institutos de pesquisa, universidades e agências governamentais). Por este motivo, o PDSC foi estendido para atender à necessidade de validação do trabalho dos parceiros.

Para projetos que envolvem parceiros externos, além das atividades da qualidade (testes, revisão de código, etc.) realizadas pelos próprios parceiros, identificou-se a necessidade de verificação adicional pelo SRBR como condição para a aceitação de cada entrega de software (release), pois alguns releases foram entregues com defeitos críticos resultando em reclamações dos usuários e retrabalho.

2. O Processo de Desenvolvimento de Software

O processo desenvolvido pela matriz da empresa (PDSC) é um processo híbrido que tem foco em desenvolvimento ágil, mas que também incorpora atividades da qualidade consideradas essenciais para o desenvolvimento de produtos de software confiáveis. Ele conta com três fases: Iniciação, para o planejamento e definição de escopo inicial; Construção, em que são gerados os releases utilizando-se Scrum [Sutherland, J. and Scwhaber, B. 2017] ou Kanban [Anderson, D. and Carmichael, A. 2016]; e, por fim, Transição, em que são realizadas as últimas atualizações nos documentos e a retrospectiva do projeto com seus resultados analisados e documentados (Figura 1). O time de desenvolvimento alocado ao projeto realiza todas as atividades em azul e parte das atividades em laranja. O time de *Software Engineering* (SE) atua como *agile coach* e auditor realizando as atividades em verde.

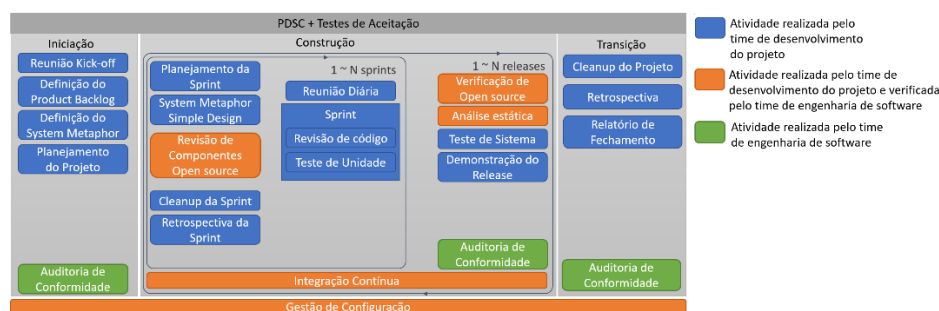


Figura 1. PDSC

3. Contextualização do Problema

Quando o SRBR começou a trabalhar com parceiros, o PDSC foi adotado sem nenhuma mudança. Porém, percebeu-se que a mera incorporação de software produzido por parceiros sem nenhuma verificação ou validação adicionais, além de uma breve demonstração dos requisitos, estava resultando em entrega de software com baixa qualidade em relação aos indicadores da qualidade monitorados pela empresa:

- Defeitos encontrados em testes por severidade;
- Número de componentes *open source* não-permitidos;
- Defeitos encontrados em análise estática por criticidade.

A percepção tardia de problemas relacionados aos indicadores acima causou, muitas vezes, aumento de custo dos projetos devido às extensões de contrato que se fizeram necessárias para tratar tais problemas.

4. A Atividade de Teste de Aceitação e a Verificação da Qualidade

Após alguns resultados ruins obtidos nos indicadores da qualidade mencionados e usuários finais relatarem problemas, foi incorporada ao PDSC uma atividade complementar chamada “Teste de Aceitação” realizada pelo time de SE antes da entrega de cada release pelo parceiro. Esta atividade se desdobra durante as fases de Iniciação e Construção como pode ser visualizado em amarelo na Figura 2. Além dos testes de aceitação, a verificação da qualidade realizada pelo SE durante as fases foi estendida nas fases de Iniciação e Construção.

- Na fase de Iniciação:
 - Validação do Plano de Trabalho do parceiro;

- Participação na reunião de *kick-off* do projeto;
- Validação do Plano de Testes do parceiro;
- Criação do Plano de Testes de Aceitação (cronograma e equipe, escopo de testes, procedimentos, ambiente necessário, critérios de entrada, parada e reteste, critérios de aceite, riscos relacionados);
- Na fase de Construção (antes da entrega de cada release):
 - Especificação dos casos de teste de aceitação (baseados em casos de sucesso e insucesso dos cenários de uso);
 - Validação dos critérios de entrada do teste de aceitação;
 - Validação dos requisitos através de uma “imersão” no software a ser entregue;
 - Participação na demonstração do release (review);
 - Preparação, execução e documentação dos testes de aceitação (*in company*);
 - Preparação, execução e documentação dos testes de aceitação (*field test*);
 - Participação na reunião de retrospectiva.

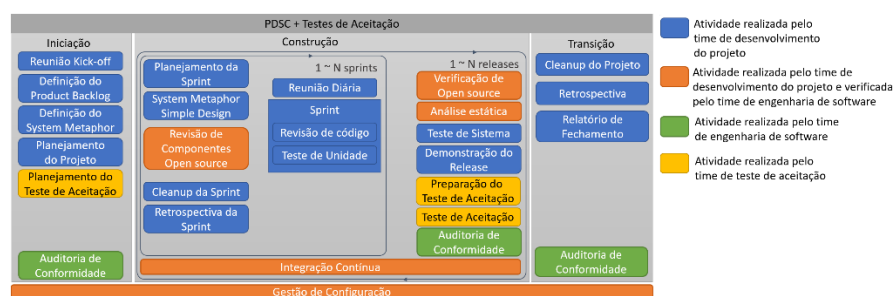


Figura 2. Passos do Teste de Aceitação no PDSC

5. Resultados do Teste de Aceitação e Verificação da Qualidade em Projetos

Foram redefinidas atividades de testes de aceitação e verificação da qualidade que passaram a ser realizadas como condição para a aceitação de releases. Alguns resultados preliminares foram publicados em 2018 [Lemos, et al 2018]. Neste mesmo ano foram incorporados testes em ambiente de usuário. Neste caso, os testes consistiram basicamente da execução dos cenários de uso (sucesso e insucesso) pelos usuários, monitorados pela equipe de teste de aceitação do SRBR.

Até o presente momento, testes de aceitação em ambiente de usuários foram realizados em três projetos (Projeto 1, Projeto 2 e Projeto 3). Duas pessoas fizeram parte destes testes. Apesar de haver alguma resistência inicial, nos três casos, os usuários logo perceberam que o time de teste de aceitação seria um aliado na melhoria dos produtos e houve pronta colaboração.

Vale salientar que, para os três projetos, os parceiros haviam realizado testes antes dos releases e não encontraram nenhum defeito. Entretanto, nos testes de aceitação (tanto *in company* quanto *field test*) vários defeitos foram notados como detalhado a seguir.

5.1. Resultados do teste de aceitação - Projeto 1

Contextualização: o projeto 1 consiste de um software com conteúdo educacional a ser usado em vilas do norte do país com conectividade limitada à Internet. Foram realizados dois ciclos de teste de aceitação “*in company*” e seis ciclos em ambiente do usuário (*field tests*). Os principais defeitos foram documentados na Tabela 1:

Tabela 1. Defeitos do Projeto 1

Tipo	Descrição	Ação Realizada pelo SRBR
Ambiente	Em uma das vilas apenas 10 dispositivos podiam ser conectados à Internet simultaneamente. Ao especificar os requisitos, o parceiro não considerou esta limitação.	Foi solicitada ao parceiro a configuração de limite de dispositivos conectados.
	Em um dos releases entregues o software mostrava uma mensagem de erro não compatível com o problema real que era o timeout causado pela largura limitada de banda.	Foi solicitado ao parceiro aumentar o tempo em que o software esperava pela resposta do servidor e a correção da mensagem de erro.
Usabilidade	Usuários apresentaram dificuldade num jogo do software devido à calibração da sensibilidade do mouse não ser adequada ao movimento.	Foi solicitado ao parceiro melhorar a calibração para diminuir a sensibilidade do mouse permitindo o correto uso.
	O calendário nativo do Android não era usado pelos usuários devido à má experiência do usuário na seleção da data e num defeito apresentado ao selecionar a data que levava ao crash do software.	Os defeitos foram reportados ao parceiro que substituiu o componente de seleção de data.
	O mecanismo de busca no software não era adequado à necessidade. Não havia, por exemplo, busca por capítulo de livro.	O defeito foi reportado e o parceiro realizou a adequação do mecanismo de busca conforme as necessidades do usuário.
Funcional	Não havia indicação da faixa etária adequada a cada jogo presente no software.	SRBR requisitou ao parceiro a inclusão da informação referente à faixa etária.
	Para alguns casos, o conteúdo incluído no banco de dados do software estava incorreto (com base no livro físico usado como fonte da informação).	SRBR requisitou ao parceiro a revisão do conteúdo do software com base no livro físico.
	A informação sobre os componentes <i>open source</i> usados pelo parceiro não foi especificada no software (<i>Open Source Announcement</i>).	SRBR requisitou ao parceiro a criação do <i>Open Source Announcement</i> no software.

5.2. Resultados do teste de aceitação - Projeto 2

Contextualização: o projeto 2 é uma solução web para uso por cartórios de registro civil da região norte do Brasil. Foram realizados três ciclos de teste (*field tests*). Testes “*in company*” não foram realizados devido a dependências de ambiente específicas. Os principais defeitos encontrados pelo time do SRBR estão descritos na Tabela 2:

Tabela 2. Defeitos do Projeto 2

Tipo	Descrição	Ação Realizada pelo SRBR
Ambiente	A indisponibilidade, em alguns cartórios, do equipamento mandatório a ser integrado ao software impossibilitou uso de parte do software.	SRBR solicitou ao parceiro a implementação de tela de configuração de dispositivos. Assim, o software passou a habilitar apenas funcionalidades que dependiam de dispositivos disponíveis.
Usabilidade	Usuários reportaram dificuldades com interfaces devido à baixa usabilidade. Em adição reportaram ausência de manuais e falta de treinamentos.	SRBR registrou defeitos de usabilidade específicos e solicitou treinamentos e manuais de usuário para as funções mais frequentemente executadas.
	Em alguns casos o usuário tentava atualizar a tela (após atualização nos dados), mas a interface continuava a apresentar dados desatualizados.	SRBR reportou o defeito ao parceiro.
	Alguns campos de entrada de dados não tinham limite ou máscara permitindo que o usuário entrasse dados inválidos. Ex: CPF.	SRBR reportou os defeitos ao parceiro.
	Havia interfaces com palavras com erros de grafia.	SRBR reportou os defeitos ao parceiro.
Funcional	Em muitos requisitos apenas o fluxo principal foi tratado pelo software. Ex: mesmo usuário cadastrado mais de uma vez.	Todos os casos de defeitos em fluxos alternativos foram reportados ao parceiro.
	Não havia a possibilidade de reimpressão de documentos oficiais já impressos.	O parceiro corrigiu o defeito após reporte pelo SRBR.
	A informação sobre os componentes <i>open source</i> usados pelo parceiro não foi especificada no software (<i>Open Source Announcement</i>).	SRBR requisitou ao parceiro a criação do <i>Open Source Announcement</i> no software.

5.3. Resultados do teste de aceitação - Projeto 3

Contextualização: O projeto 3 é um software de realidade virtual para o treinamento de usuários em cenários de ergonomia. Houve cinco ciclos de teste de aceitação (*field test*) neste projeto. Os principais defeitos encontrados estão descritos na Tabela 3:

Tabela 3. Defeitos do Projeto 3

Tipo	Descrição	Ação Realizada pelo SRBR
Usabilidade	Muitas interfaces no software tinham descrições incoerentes e complexas dos cenários de ergonomia. O usuário realizando a experiência com os óculos de realidade virtual muitas vezes não entendia o propósito do cenário.	SRBR requisitou ao parceiro a alteração das descrições dos cenários de ergonomia não adequados para que fosse atendido o objetivo do software que era o treinamento em ergonomia.
	O tutorial presente no software era complexo para o perfil dos usuários e, em muitos casos, estes não conseguiam atingir o objetivo.	SRBR reportou o defeito ao parceiro e requisitou a melhora em pontos do tutorial.
	Em jogos com demonstração de cenários, os personagens deveriam realizar movimentos corretos ergonomicamente. Porém, em alguns casos os personagens ficavam fora do campo de visão do usuário final (quando vestindo os óculos de realidade virtual). Assim, a demonstração da posição não era realizada adequadamente.	Sendo o objetivo dos jogos de demonstração orientar os usuários quanto ao correto posicionamento em determinados cenários, SRBR solicitou ao parceiro a mudança para a melhor visualização do personagem nos jogos.
Funcional	O software conta com personagens que imitam movimentos humanos. Porém, em alguns cenários a perna do personagem sentado ultrapassava o limite da mesa ou de uma caixa sendo levantada o que foge ao real e não atingia a experiência desejada ao usuário.	SRBR reportou o defeito ao parceiro.
	Quando o usuário simulava, com a ajuda dos óculos de realidade virtual, o levantamento de um objeto, o usuário conseguia levantar outras partes do cenário que não deveriam participar da experiência. Em alguns casos o software ficava travado após estes movimentos incorretos.	SRBR solicitou que apenas alguns objetos pré-definidos pudessem ser levantados/movidos.
	Em alguns casos, um laser era habilitado para que o usuário indicasse objetos na tela (usando um joystick, além dos óculos de realidade virtual). O laser ultrapassava os objetos selecionados.	SRBR solicitou ao parceiro que o laser fosse limitado à apenas o objeto selecionado.
	No tutorial, em algumas situações o joystick não funcionou corretamente.	SRBR solicitou ao parceiro que refizesse a calibração do joystick.

Como pode ser notado nas Tabelas 1, 2 e 3, havia vários defeitos nos produtos de software a serem entregues ao SRBR que não foram localizados durante os testes realizados pelos parceiros. A Tabela 4 apresenta o detalhamento dos defeitos encontrados. No total, para os três projetos, foram encontrados 140 defeitos sendo que 32% deles eram de severidade A ou B de acordo com os conceitos a seguir:

- A: Defeito crítico que impede o uso do software;
- B: Defeito que produz inabilidade intermitente de uso ou impacto de nível médio no uso do software;
- C: Design/funcionalidade não implementada, mal implementada ou erroneamente implementada, mas que não afetava o uso do software;
- D: Sugestão de melhoria baseado na especificação e na experiência dos usuários durante os testes de aceitação.

Embora a atividade de teste de aceitação tenha resultado num número maior de releases e o tempo investido tenha sido considerável (Tabela 5), foi essencial para a maturação dos produtos. Em especial, os testes de campo contribuíram para a melhoria da

usabilidade e correção de defeitos relacionados à infraestrutura. Lembrando houve continuação do desenvolvimento nos parceiros durante a realização dos testes de aceitação. Houve extensão no cronograma em virtude da correção dos defeitos encontrados, porém tais extensões não foram classificadas como atrasos em virtude do ganho na qualidade e estabilidade final dos produtos de software entregues.

Tabela 4. Defeitos por Severidade nos Projetos 1, 2 e 3

Projetos Severidade	Projeto 1				Projeto 2				Projeto 3				Total			
	A	B	C	D	A	B	C	D	A	B	C	D	A	B	C	D
Defeitos encontrados em Acceptance Test – in company	5	19	28	5	0	0	0	0	0	0	0	0	5	19	28	5
Defeitos encontrados em Acceptance Test – field test	1	3	1	6	1	6	15	3	1	9	20	17	3	18	36	26
Total	6	22	29	11	1	6	15	3	1	9	20	17	8	37	64	31
%	41%		43%	16%	28%		60%	12%	21%		43%	36%	32%		46%	22%

Tabela 5. Sumário de Releases e Ciclos de Teste de Aceitação nos Projetos 1, 2 e 3

Projeto	Tempo medio para a release	Releases Planejadas	Releases adicionadas devido a defeitos encontrados em teste de aceitação	Total de Releases	Ciclos de testes de aceitação planejados	Ciclos de teste de aceitação adicionados	Total de Ciclos de Teste de Aceitação	Tempo total da equipe de teste de aceitação (dias)	Defeitos encontrados (Tabela 4)
Projeto 1	4 semanas	8	4	12	4	3	7	80	68
Projeto 2	15 semanas	4	0	4	4	0	4	16	25
Projeto 3	3 semanas	1	5	6	1	3	4	15	47

5. Conclusões e Trabalhos Futuros

Embora o teste de aceitação em campo tenha demandado mais tempo do que apenas o teste de sistema realizado pelo parceiro, o ganho no total de defeitos com severidades “A” ou “B” encontrados foi bastante relevante justificando sua realização. Houve pronta colaboração dos usuários o que contribuiu para a melhoria da maturidade do software entregue. Além disso, a presença da equipe de teste de aceitação em campo ajudou a aumentar a confiança dos usuários no software.

Sendo que um dos critérios de aceitação do SRBR é que não haja defeitos de severidade A ou B num release entregue, os parceiros, após o estabelecimento dos testes de aceitação, passaram a ser mais cuidadosos antes de submeter um release ao SRBR.

Como desafios futuros, o SRBR pretende:

- Incluir o desempenho em teste de aceitação de projetos passados em futuras candidaturas de um potencial parceiro;
- Usar os resultados de testes de aceitação já realizados para direcionar os treinamentos ministrados aos parceiros para minimizar a recorrência de defeitos;
- Automatizar parte dos testes de aceitação para diminuir o tempo para emissão do resultado da aceitação de um release;
- Através de pesquisas em campo, entender melhor o perfil dos usuários antes do início dos testes de aceitação.

Referências

- Sutherland, J. and Scwhaber, B. (2017) “The Scrum Guide - The Definitive Guide to Scrum: The Rules of the Game”. Nov 2017.
- Anderson, D. and Carmichael, A. (2016) “Essential Kanban Condensed”. Seattle,
- Lemos, G. Costa, M. Borghi, T. and Povoas, P. (2018) “Validation of Outsourcing Teams Work on Agile Projects of Samsung R&D Institute Brazil”. Proceeding of the 13th International Conference on Global Software Engineering. Pages 11-15. Gothenburg, Sweden. May 27 - 29, 2018. ISBN: 978-1-4503-5717-3.