

Propuesta de metodología que coordine el levantamiento de requerimientos en apoyo a los estudiantes de la Universidad estatal de Sonora, unidad académica Navojoa, para introducirlos en el desarrollo global de software

*Metodologia proposta para coordenar o levantamento das necessidades de apoio dos estudantes da Universidade Estadual de unidade acadêmica Sonora Navojoa, para apresentá-los para o desenvolvimento global de software*

**Josefina Ortega Ruiz**

Universidad Estatal de Sonora, México  
[josefina-ortega@live.com.mx](mailto:josefina-ortega@live.com.mx)

**Filiberto Valenzuela Mendoza**

Universidad Estatal de Sonora, México  
[filiberto.valenzuela@ues.mx](mailto:filiberto.valenzuela@ues.mx)

**Alma Isabel Arias Hurtado**

Universidad Estatal de Sonora, México  
[alma\\_isabela@hotmail.com](mailto:alma_isabela@hotmail.com)

**Margarita Soto Rodríguez**

Universidad Estatal de Sonora, México  
[maguysoto@hotmail.com](mailto:maguysoto@hotmail.com)

## Resumen

El presente trabajo contextualiza el desarrollo global de software (DGS) como una nueva tendencia de trabajo, así como la posibilidad de crear una metodología para coordinar el levantamiento de requerimientos que apoye a los estudiantes de la carrera de Ingeniería en Software de la Universidad Estatal de Sonora, Unidad Académica Navojoa, a incorporarse en

esta área. Para ello, se abordan las problemáticas actuales en DGS y los beneficios de esta nueva tendencia de desarrollo de software.

**Palabras clave:** metodología para DGS.

## Resumo

Este artigo contextualiza o desenvolvimento global de software (DGS) como uma nova tendência do trabalho ea possibilidade de criação de uma metodologia para coordenar o levantamento das necessidades para apoiar os alunos de Engenharia de Software da Universidade Estadual Sonora, Unidade Acadêmica Navojoa, para juntar-se nesta área. Para fazer isso, os problemas atuais na DGS e os benefícios desta nova tendência de desenvolvimento de software são abordados.

**Palavras-chave:** metodologia para a DGS.

**Fecha recepción:** Enero 2015

**Fecha aceptación:** Julio 2015

---

## Introdução

Hoje, graças aos avanços tecnológicos estão permitindo que as pessoas se comuniquem em um contexto de negócios, embora sejam em diferentes regiões. Algo semelhante acontece com o desenvolvimento de software: é possível gerar projetos com equipes cujos membros são quilômetros de distância, que é conhecido como um desenvolvimento global de software (DGS), que oferece benefícios como a redução Custos quando se trabalha com regiões onde o trabalho é mais barato, por exemplo, a Índia ea China. Além disso, a diferença de tempo entre diferentes países permite que o ciclo de desenvolvimento de software é quase constante e, portanto, pode reduzir significativamente o tempo de conclusão do projeto. Sempre que se fala das vantagens de um método de trabalho é conveniente investigar para descobrir e eliminar as desvantagens, melhorar a eficiência do equipamento DGS, maximizar os benefícios desta nova tendência e, acima de tudo, garantir que as novas gerações de estudantes Engenharia de Software da Universidade do Estado de Sonora envolvidos neste tipo de emprego e alcançar ligada com empresários que são, na prática, ao abrigo deste ambiente de desenvolvimento, ou que poderiam

criar entre equipes de desenvolvimento de estudantes, sob uma metodologia rigorosa levantamento de requisitos e trabalhar de forma independente dentro desse tipo de desenvolvimento de software.

O desenvolvimento de software é uma estrutura que é feito com práticas diferentes. Pressman (2008) menciona cinco práticas a serem realizadas em qualquer processo de falar sobre engenharia de software: comunicação, planejamento, modelagem, construção e implantação. Em seguida, a prática da comunicação é descrito.

O mesmo autor afirma que "o caminho de comunicação para o entendimento geralmente esburacada", o que significa que a comunicação eficaz é o maior desafio para um engenheiro de software. No livro de 10 Pressman importante para as exigências de elevação são princípios detalhados para ser bem sucedido. Um deles é "OTL", bem como a comunicação verbal existe gestual, que fornece mais informações sobre os requisitos do cliente. Mas o que acontece em ambientes DGS? O OTL é zero ou perto de zero, apesar de saber que os requisitos reunindo fase é a mais importante, pois determina o sucesso ou fracasso de um aplicativo. Na vida universitária é praticamente certo que os alunos não têm experiência suficiente para fazer um levantamento das necessidades, complicado em um ambiente global.

## **JUSTIFICAÇÃO**

A realidade atual dos sistemas de informação do desenvolvimento do projeto deve ser realizada com mais frequência em um ambiente distribuído, para que as equipes de desenvolvimento são formados em vários países ou empresas compõem sua equipe de desenvolvimento de software com as pessoas diferentes regiões de um mesmo país; assim, poupar custos e aumentar a sua competitividade. No entanto, a gestão para alcançar estas vantagens no ambiente de trabalho é mais complexa, uma vez que nenhuma boa comunicação entre as pessoas que trabalham em lugares diferentes, o produto final pode ser o caos, aumentando assim os custos.

Alguns autores (Leandro Lopes, Rafael Prikladnicki, Jorge Audy) indicam que as exigências de elevação é a fase mais crítica no desenvolvimento de software e ambientes é ainda mais complicada DGS como o enquadramento ideal é que todos os equipamentos projeto

(desenvolvedores, clientes e usuários) estão reunidos no mesmo lugar para facilitar a comunicação e alcançar o entendimento dos requisitos de uma forma favorável.

Para trabalhar em ambientes de desenvolvimento distribuídas, muitas vezes, a equipe é composta de pessoas que falam línguas diferentes, impedindo a comunicação entre eles e uma boa compreensão do que você quer alcançar. Além disso, o aspecto cultural não devem ser deixados fora do âmbito dos requisitos e coordenação das actividades entre os diferentes membros da equipe DGS. Esta forma de trabalho exige mais cooperação de pessoas que vêm de culturas diferentes, como há diversidade em seus estilos de comunicação e senso de tempo, dando origem a mal-entendidos graves que afetam negativamente o desenvolvimento de projetos de software (James D. & Herbsleb Deependra Moitra).

Estes aspectos que dificultam o trabalho no ambiente distribuído entre equipes de profissionais, crescer quando os estudantes de nível universitário desejam praticar os seus conhecimentos na área e juntar-se ao tipo de emprego, mas, devido à sua falta de experiência foi desmoralizando a sua primeira tentativas fracassadas. Como recém-licenciados que estão a entrar num ambiente normal de trabalho, onde os membros da equipe e os usuários estão próximos, no mesmo escritório ou cidade, ou a possibilidade de manter uma conversa em pessoa, viver de uma maneira mais complexa mesmo em ambientes distribuídos.

Há metodologias que suportam o desenvolvimento de software, como o RUP (Rational Unified Process), o que nos dá uma abordagem para construir um software implementando quatro fases: iniciação, elaboração, construção e transição (Larman, 2006). Nestas fases mostra o que deve ser coberto em cada e os produtos que recebem, mas não nos diz como.

Outro metodologias amplamente usados hoje são o ágil, tais como programação extrema (XP), Scrum, desenvolvimento de software adaptável, entre outros métodos, cujo principal objetivo é implementar um conjunto de melhores práticas para trabalhar de forma colaborativa em equipes e que de modo a obter bons resultados na execução de projectos, embora não especificamente detalhes como fazer um levantamento de requisitos, que é o que esta pesquisa se concentra.

Até à data, as metodologias acima ensinadas nas disciplinas Fundamentos da Engenharia de Software e Engenharia de Software em UES, UAN, mas não o suficiente para ajudar o aluno a se juntar no mundo do desenvolvimento de software global. Por isso é muito importante para criar

uma metodologia para apoiar os alunos da Universidade Estadual de unidade acadêmica Sonora Navojoa, a aderir a esta forma de trabalhar, como este cada vez mais prevalente. No entanto, ainda existem muitos problemas que criam incerteza no sucesso desta disciplina, mais no caso de alunos com nenhuma experiência de trabalho. Por esta razão, é muito importante a aplicação de métodos que aumentam a confiabilidade para continuar a beneficiar desta forma de trabalho.

Apesar do desafio, a DGS tem vindo a crescer a um ritmo considerável, o grau de tornando uma necessidade, uma vez que lida com questões como custo, escassez de recursos e localização do desenvolvimento perto dos clientes; Além disso, os problemas que limitam o seu desempenho óptimo deve ser reduzida.

## **PROBLEMA**

Vários autores têm estudado as vantagens e desvantagens da disciplina de desenvolvimento global de software (Oktaba 2010-2011) (Eckstein, 2010). Entre os problemas que são mencionados mais:

-A Falta de processos bem-sucedidos de organizar a coordenação de um projeto entre várias organizações ou equipes.

Problemas de comunicação: em face-off entre a equipe e comunicação com o cliente são frequentes problemas de compreensão, o que é ainda mais acentuada com as equipes que trabalham em ambientes distribuídos geograficamente, levando a mal-entendidos, omissão de informações, e assim por diante.

Cultural-diferenças: diferentes formas de pensar e de expressão, diferentes costumes e formas de trabalho estão latentes para ameaça a qualidade do produto de software.

Estes problemas são devido retrabalho, fazendo a vantagem de custo cessar, para o investimento de tempo extra.

Apesar destes obstáculos, esta prática é uma realidade na indústria de software que está crescendo conforme o tempo passa. Com base nestas situações, a seguinte questão:

Que impacto o desenvolvimento de uma metodologia para estudantes de Engenharia de Software da Universidade do Estado de Sonora, Unidade Acadêmica Navojoa, para que eles possam coordenar o levantamento de requisitos de software, em seguida, entrar no novo mercado de trabalho, como é o desenvolvimento global de software?

## **PRESSUPOSTOS**

A metodologia proposta para coordenar de forma eficiente o levantamento de requisitos de software, ajuda os alunos a UES UAN para participar nesta nova forma de trabalho de forma eficiente, de modo que o problema de retrabalho causada por reduções mal-entendidos nas especificações fornecidas pelo cliente, impactando positivamente na redução de custos de produção, tempo de desenvolvimento e entrega de uma funcionalidade completa do software, e gerar a confiança de que os alunos das UES pode ser considerado para o trabalho na significa.

## **OBJETIVO GERAL**

Desenvolver e avaliar uma metodologia para coordenar o levantamento de requisitos de software em ambiente de desenvolvimento distribuído global que apoia os estudantes da UAN UES a juntar-se que esta abordagem funcione.

## **OBJETIVOS ESPECÍFICOS**

1. Fornecer uma ferramenta metodológica para coordenar o levantamento de requisitos para ajudar a reduzir o problema da re-trabalho da DGS, impactando os custos de produção e entrega do software cliente com funcionalidade completa eo tempo previsto.
2. Avaliar a metodologia formado por alunos do equipamento UES UAN.
3. Verifique através de um estudo comparativo a redução do tempo de re-trabalho no desenvolvimento de software, considerando sistemas projetados em tempos iguais, um deles usando seu modo tradicional de elevação e outros requisitos usando uso metodologia desenvolvida por esta proposta.
4. Apoiar os estudantes da UAN UES para entrar na área de trabalho em um ambiente de desenvolvimento de software global.

## **METAS**

1. Desenvolver uma metodologia para coordenar o levantamento das necessidades para orientar os participantes para capturar uma clara especificação do cliente e projeto para os outros membros do projecto de uma forma compreensível, sem interpretações ambíguas acomodar o equipamento.
2. Proporcionar através da ferramenta metodológica, um modelo para o levantamento de requisitos de software projeta ambientes DGS.

## **MÉTODO**

Nesta pesquisa várias fases representava uma metodologia que é feito especificamente para o levantamento de requisitos em um ambiente de DDS foram desenvolvidos. Os referidos passos são:

1. Inicie uma investigação completa das metodologias atuais de desenvolvimento de software orientado, a fim de assimilar as questões abordadas e resgatar a parte relativa à elicitação de requisitos; em seguida, suas vantagens e desvantagens para chegar a uma conclusão porque eles ainda têm esse problema nesta área são discutidos.
2. Em seguida, realizar uma entrevista com analistas de sistemas de diferentes fábricas que operam no modo de software ambiente distribuído, para reunir informações importantes da operação em si sobre os detalhes dos problemas atuais relacionados ao levantamento de requisitos.
3. Em seguida, proceder para desenvolver uma metodologia e baseia-se em métodos de pesquisa e outras informações recolhidas a partir de especialistas na área que tenham sido directamente afectadas pelo problema que esta tese estudos.

4. Uma vez que você tem a metodologia elaborada, os alunos escolhem um projeto no modo DGS, onde testa a eficiência da metodologia desenvolvida; Esta seleção é feita minuciosamente, a fim de obter resultados eficazes que ajudam a concluir se a metodologia desenvolvida fornece apoio importante na elicitação de requisitos, para diminuir o problema de mal-entendidos nos requisitos de software.

### **Contribuições esperadas**

Apesar complicação apresenta os requisitos de extração no processo de desenvolvimento de software, nenhuma metodologia contribuindo especificamente para esta atividade; se feita incorretamente, ele irá falhar no projeto, mesmo nos casos em que as partes interessadas para se comunicar cara a cara. Este problema aumenta ambientes DGS, acima. No entanto, esta forma de trabalhar no ambiente global tornou-se uma alternativa recorrente pelas organizações que têm a necessidade de fornecer desenvolvimento de software mais clientes, aproveitando o baixo custo de mão de obra ea enorme quantidade de desenvolvimento profissional em algumas regiões (Damian, 2006).

Devido a isso, é importante para fornecer uma metodologia eficaz que suporta a fase mais crítica do desenvolvimento de software, ou seja, levantar requisitos para mais e mais estudantes têm melhores bases na vida universitária e aderir sem maiores problemas na vida profissional e, acima de tudo, ser incorporadas a este tipo de trabalho na DGS, que de acordo com pesquisa tem muito a oferecer novos profissionais de desenvolvimento de software, independentemente do local onde vivem.

### **Bibliografía**

Damian, D. M. (2006). Global Software Development: How far have we come?

Eckstein, J. (2010). Agile Software Development with Distributed Teams. Dorset House Publishing.

James D. Herbsleb and Deependra Moitra, L. T. (s.f.). Global Software Development.

Larman, C. (2006). UML y Patrones. Pearson.

Leandro Lopes, Rafael Prikladnicki, Jorge Audy. (s.f.). Requirements specification in distributed software development a process proposal. Requirements Specification in Distributed Software Development a Process Proposal. Porto Alegre, RS, Brazil.

Oktaba, H. (2010-2011). Desarrollo global de software, retos y factores de éxito. Software Guru, 60.

Pressman, R. S. (2008). Ingeniería del Software. México, McGraw-Hill.