Porque projetos de software falham!

Por João Carlos Testi Ferreira

Software é complexo

de representation de de la complexión de la complexió

Software é abstrato

A tecnologia muda rapidamente
soda dia anco o organizadore deliración efeculador, logo alto entre como aplicação que ado
ser soda dia anco o organizadore deliración de efeculador, logo alto esta como a plicação que ado
ser soda deliración de esta abuseron. Se entre deliración de logo entre acidado en

A tecnología é um domínio vasto

Sento sós glas para en mais deversas cobas Guita mas se eglas medias con deservacionel
chromoléache. Um sentos problemo destas abresas se eglas medias con los en sus consideras de
consolegio. Las ministratos de praedictidades padre mediantes a conjunçãos de para questos,
consolegio. Las ministratos de praedictionel para reconsolegio. Las ministratos de para destador padre mediantes que desta conferior se conjunçãos para reconsolegio.

Requisitos são incompletos

Bir population die dieleks der contenentes statzen verschille pomei que contenen en provinciam genetic, index noves provinci que periorie mais provincia (quelle inteperante premiera se provincia, calentanda antida children antiques de indicentes vivors), destada contributo del monteceniglio des Nacionalismo de Cartaglier de provincia e participa. As melhores práticas não são maduras
so deservidamento de administrator presentamento apecas en contrato, com lambito

So deservolvimento de sellment pila nos presequentos aperas en sonimirir, mas tando om a mentranção dese estimante, los que prisco considerante questir que francissa om postencio de deservolvimento, mas que podem soble y per revent más resultada ao manifemção, los patidos em advorar paresente possos interpa de resperimentação, ado presidêncio conforme se elegante a deservo. A experimentação da tecnologia é incompleta

à terminique medio registemente, scompanisación tendéración e promoblados, bos Sistacos con el secuciospia a discubir medica resona seguer chaptera a conscience. El firminia desanacionida, a consequencia fisca e la adicación sociales promotes en posicion.

O trabalho repetitivo é automatizado

Deservolver seftivare não é um trabalho repetitivo. Cada projete é único e immentados diferentes. A autorastivação das atreidades de projeto não é algo visivol.

Desenvolvimento de software é pesquisa

Quanda varios automatizar un processo conseguineo identificar o caminho feliz, mas es demala isão surgem facilmente. O processo de identificação elest caminhas é de decaderia, trabalho em equipe e artiginam-se de pessoas que falsos dislatos differentes.

A construção é realmente o projeto

Diferente de outros projetos, projetar o software é de fato construí-les. O planejamento, por mais detalhación que seja, rafo é capaz de definir com precisio a forma como este será feito. Ocorrendo mudanças por necessidoses observados durante o projeto.

Mudanças são entendidas como simples

É fácil compreender a difficuldade de alterar de lugar uma ponte já construida. Com software esta visão não aconteo. Mudanças parecem faceis, é sempre muito difficil explicar a difficuldade de se lidar com mudanças de última bora.

Mudanças são inevitáveis

Em software esta é nossa único corteza. Outro corteza é que estas mudanças inito ocerteir orde menos imaginávamos. Não lai como persoar que leumos começar um projeto e segui-lo ani; o final sem mudanças. Na medida em que o apilware vali apravado firmo, vido sendo apravaldos estançais polesações em aprilace que apilware en filtra.

Referência

Software project secrets: why software projects fail. George Stepanek Editora Apress 18, sep, 2005

Porque projetos de software falham!

Por João Carlos Testi Ferreira

ftware é abstrato

é abstrato, você não pode tocá-lo. Sua essência é a reprodução de um comportamento. 2, por exemplo, podemos imaginar as possíveis jogadas do oponente em função de uma as apenas um movimento diferente do esperado e todo o plano de jogo cai por terra.

A tecnologia muda rapidamente

A cada dois anos os computadores dobram de velocidade. Hoje não vemos uma aplicação que não seja construída dentro de um framework. As melhores soluções de hoje certamente não serão as de amanhã. A tecnologia é muito volátil, em especial na área de software.

Por João Carlos Testi Ferreira

Software é complexo

programa de ampanha este os positias mais tatricuales, delicalmente epitilizados e amente entrilogados da indicatis hamma tai lojo. Elos do adquiras com popa racino mais máveir de que aliquer mocos en popa ello se desperian, mos des laterapes e enceregames arma contra co se com de

Software é abstrato

drivere não é distria, é abemira, veoi sale pede teol los fina embesió é o repredição de um componitamente, o um jugo de traditivo, par exemplo, podemen imaginar as positivos jugadas do opositeire em finação de medio - nomas impedias, mas aposas um movilameno diferente dia comendo e rodo exista de los calisartentes.

A tecnologia muda rapidamente

A cada dois anos os computadores dabram de relocidade. Hoje não vemos uma aplicação que não seja construida dentro de um framework. As melhores soluções de hoje certamente não serão as de amanhã. A tecnologia é muito valúid, em especial na área de softmane.

A tecnologia é um domínio vasto

Terras subspites para as mais diversas cuisas. Carla uma se aplica melhor em determinada circumstitada. Um mesmo problema admire diversas soluções com base em um conjumo de tecnologias. Esta miriade de possibilidades pode consideras a aquisição de parses promos, componentes ou qualquer elemento que pasa ase incorporado.

Requisitos são incompletos

On regulation são obstitión dos continuantes. Algunina veres são pessoas que executara o processo em questão, curaos vereo, pessoas que grencialm este auroseisos. Quando estas pessoas pensoan nu processo, o chamado continho Relix costinua ser facilmente visited, os demais comissiones dos compilicacios. Normalmente: as variações dos processo e possibeis entro que acorrem nomente são licentificados tratificados.

As melhores práticas não são maduras

No deserrollrimano de sellmare não nos processamos aperass em cavatrás, mas também om a manutenção deste solhante, às bas peritats consideram questies que funcionado om percesso de deserrollations, na que perdem acidas por everm más establica na manutenção as pelíticas em sollware possuem pouco tempo de experimento qualo, não permitidos acultas oses importos timbos possuem pouco tempo de experimento qualo. So permitidos acultas oses importos timbos de companios de consecuencia permitidos acultas que sem importos timbos possuem pouco tempo de experimento.

A experimentação da tecnologia é incompleta

A tecnología muda rapidamente, acompanhando tendências e necessidades. Isso fiz com que temologías adetados muitos veces sequer cheguem a anadurecer, já ficando desatualizadas.

O trabalho repetitivo é automatizado

Desenvolver software não é um trabalho repetitivo. Cada projeto é único e traz resultados diferentes. A automatização das atividades de projeto não é algo viável.

Desenvolvimento de software é pesquisa

Quando vamos automatizar um processo conseguimos identificar o caminho feliz, mas os demais não surgem facilmente. O processo de identificação destes caminhos é de descoberta, trabalho em equipe e originam-se de pessoas que folam dialetes difenentes.

A construção é realmente o projeto

Diferente de outros projetos, projetar o software é de fato construí-lo. O planejamento, por mais detalhado que seja, não é capaz de definir com precisão a forma como este será feito. Ocorrerão mudanças por necessidades observadas durante o projeto.

Mudanças são entendidas como simples

É fácil compreender a dificuldade de alterar de lugar uma ponte já construida. Com software esta visão não acontece. Mudanças parecem fáceis, é sempre muito difícil explicar a dificuldade de se lidar com mudanças de última hora.

Mudanças são inevitáveis

Em software esta é nossa única certeza. Outra certeza é que estas mudanças irão ocorrer onde menos imaginávamos. Não há como pensar que iremos começar um projeto e segul-lo até a final sem mudanças. Na medida em que o software vai tomando forma, vão sendo percebidas situações indesejáveis ou melhorias que devem ser feitas.

Software é complexo

Os programas de computador são os produtos mais intrincados, delicadamente equilibrados e finamente entrelaçados da indústria humana até hoje. Eles são máquinas com peças muito mais móveis do que qualquer motor: as peças não se desgastam, mas elas interagem e escorregam-se umas contra as outras de formas que os programadores não podem prever. [Gleik 1992] (Tradução nossa)

Software é abstrato

Software não é físico, é abstrato, você não pode tocá-lo. Sua essência é a reprodução de um comportamento. Em um jogo de xadrez, por exemplo, podemos imaginar as possíveis jogadas do oponente em função de uma de nossas jogadas, mas apenas um movimento diferente do esperado e todo o plano de jogo cai por terra.





Requisitos são incompletos

Os requisitos são obtidos dos contratantes. Algumas vezes são pessoas que executam o processo em questão, outras vezes, pessoas que gerenciam estes processos. Quando estas pessoas pensam no processo, o chamado caminho feliz costuma ser facilmente visível, os demais caminhos são mais complicados. Normalmente as variações do processo e possíveis erros que ocorrem somente são identificados tardiamente.

Software é complexo

Os programas de computador são os produtos mais intrincados, delicadamente equilibrados e finamente entrelaçados da indústria humana até hoje. Eles são máquinas com peças muito mais móveis do que qualquer motor: as peças não se desgastam, mas elas interagem e escorregam-se umas contra as outras de formas que os programadores não podem prever. [Gleik 1992] (Tradução nossa)



Software é abstrato

Software não é físico, é abstrato, você não pode tocá-lo. Sua essência é a reprodução de um comportamento. Em um jogo de xadrez, por exemplo, podemos imaginar as possíveis jogadas do oponente em função de uma de nossas jogadas, mas apenas um movimento diferente do esperado e todo o plano de jogo cai por terra.

Requisitos são incompletos

Os requisitos são obtidos dos contratantes. Algumas vezes são pessoas que executam o processo em questão, outras vezes, pessoas que gerenciam estes processos. Quando estas pessoas pensam no processo, o chamado caminho feliz costuma ser facilmente visível, os demais caminhos são mais complicados. Normalmente as variações do processo e possíveis erros que ocorrem somente são identificados tardiamente.

Por João Carlos Testi Ferreira

A tecnologia muda rapidamente

A cada dois anos os computadores dobram de velocidade. Hoje não vemos uma aplicação que não seja construída dentro de um framework. As melhores soluções de hoje certamente não serão as de amanhã. A tecnologia é muito volátil, em especial na área de software.







As melhores práticas não são maduras

No desenvolvimento de software não nos preocupamos apenas em construir, mas também com a manutenção deste software. As boas práticas consideram questões que funcionaram com processos de desenvolvimento, mas que podem acabar por serem más escolhas na manutenção. As práticas em software possuem pouco tempo de experimentação, não permitindo avaliar seus impactos futuros.

A experimentação da tecnologia é incompleta

A tecnologia muda rapidamente, acompanhando tendências e necessidades. Isso faz com que as tecnologias adotadas muitas vezes sequer cheguem a amadurecer, já ficando desatualizadas. A consequência disso é a adoção de tecnologias imaturas em projetos.





A tecnologia muda rapidamente

A cada dois anos os computadores dobram de velocidade. Hoje não vemos uma aplicação que não seja construída dentro de um framework. As melhores soluções de hoje certamente não serão as de amanhã. A tecnologia é muito volátil, em especial na área de software.







As melhores práticas não são maduras

No desenvolvimento de software não nos preocupamos apenas em construir, mas também com a manutenção deste software. As boas práticas consideram questões que funcionaram com processos de desenvolvimento, mas que podem acabar por serem más escolhas na manutenção. As práticas em software possuem pouco tempo de experimentação, não permitindo avaliar seus impactos futuros.

sonware ramam:

Por João Carlos Testi Ferreira

A tecnologia muda rapidamente

A cada dois anos os computadores dobram de velocidade. Hoje não vemos uma aplicação que não seja construída dentro de um framework. As melhores soluções de hoje certamente não serão as de amanhã. A tecnologia é muito volátil, em especial na área de software.

A tecnologia é um domínio vasto

Temos soluções para as mais diversas coisas. Cada uma se aplica melhor em determinada circunstância. Um mesmo problema admite diversas soluções com base em um conjunto de tecnologias. Esta miríade de possibilidades pode considerar a aquisição de partes prontas, componentes ou qualquer elemento que possa ser incorporado.







ão são maduras

apenas em construir, mas também sideram questões que funcionaram bar por serem más escolhas na mpo de experimentação, não

A experimentação da tecnologia é incompleta

A tecnologia muda rapidamente, acompanhando tendências e necessidades. Isso faz com que as tecnologias adotadas muitas vezes sequer cheguem a amadurecer, já ficando desatualizadas. A consequência disso é a adoção de tecnologias imaturas em projetos.



A tecnologia é um domínio vasto

Temos soluções para as mais diversas coisas. Cada uma se aplica melhor em determinada circunstância. Um mesmo problema admite diversas soluções com base em um conjunto de tecnologias. Esta miríade de possibilidades pode considerar a aquisição de partes prontas, componentes ou qualquer elemento que possa ser incorporado.





A experimentação da tecnologia é incompleta

A tecnologia muda rapidamente, acompanhando tendências e necessidades. Isso faz com que as tecnologias adotadas muitas vezes sequer cheguem a amadurecer, já ficando desatualizadas. A consequência disso é a adoção de tecnologias imaturas em projetos.







O trabalho repetitivo é automatizado

Desenvolver software não é um trabalho repetitivo. Cada projeto é único e traz resultados diferentes. A automatização das atividades de projeto não é algo viável.

Desenvolvimento de software é pesquisa

Quando vamos automatizar um processo conseguimos identificar o caminho feliz, mas os demais não surgem facilmente. O processo de identificação destes caminhos é de descoberta, trabalho em equipe e originam-se de pessoas que falam dialetos diferentes.





A construção é realmente o projeto

Diferente de outros projetos, projetar o software é de fato construí-lo. O planejamento, por mais detalhado que seja, não é capaz de definir com precisão a forma como este será feito. Ocorrerão mudanças por necessidades observadas durante o projeto.

Mudar

É fácil comp construída. fáceis, é sen mudanças c



Mudanças são inevitáveis

Desenvolvimento de software é pesquisa

Quando vamos automatizar um processo conseguimos identificar o caminho feliz, mas os demais não surgem facilmente. O processo de identificação destes caminhos é de descoberta, trabalho em equipe e originam-se de pessoas que falam dialetos diferentes.



O trabalho repetitivo é automatizado

Desenvolver software não é um trabalho repetitivo. Cada projeto é único e traz resultados diferentes. A automatização das atividades de projeto não é algo viável.





A construção é realmente o projeto

Diferente de outros projetos, projetar o software é de fato construí-lo. O planejamento, por mais detalhado que seja, não é capaz de definir com precisão a forma como este será feito. Ocorrerão mudanças por necessidades observadas durante o projeto.



tivo. Cada projeto é único e traz idades de projeto não é algo Quando vamos automatizar um processo conseguimos identificar o caminho feliz, mas os demais não surgem facilmente. O processo de identificação destes caminhos é de descoberta, trabalho em equipe e originam-se de pessoas que falam dialetos diferentes.





A construção é realmente o projeto

Diferente de outros projetos, projetar o software é de fato construí-lo. O planejamento, por mais detalhado que seja, não é capaz de definir com precisão a forma como este será feito. Ocorrerão mudanças por necessidades observadas durante o projeto.

Mudanças são entendidas como simples

É fácil compreender a dificuldade de alterar de lugar uma ponte já construída. Com software esta visão não acontece. Mudanças parecem fáceis, é sempre muito difícil explicar a dificuldade de se lidar com mudanças de última hora.





Mudanças são inevitáveis

Em software esta é nossa única certeza. Outra certeza é que estas mudanças irão ocorrer onde menos imaginávamos. Não há como pensar que iremos começar um projeto e segui-lo até o final sem mudanças. Na medida em que o software vai tomando forma, vão sendo percebidas situações indesejáveis ou melhorias que devem ser feitas.

Mudanças são entendidas como simples

É fácil compreender a dificuldade de alterar de lugar uma ponte já construída. Com software esta visão não acontece. Mudanças parecem fáceis, é sempre muito difícil explicar a dificuldade de se lidar com mudanças de última hora.







Mudanças são inevitáveis

Em software esta é nossa única certeza. Outra certeza é que estas mudanças irão ocorrer onde menos imaginávamos. Não há como pensar que iremos começar um projeto e segui-lo até o final sem mudanças. Na medida em que o software vai tomando forma, vão sendo percebidas situações indesejáveis ou melhorias que devem ser feitas.

é complexo

Software é abstrato

Software não é físico, é abstrato, você não pode tocó-lo. Sua escência é a regnodução de um comportamento. Em um jogo de nadrec, por exemplo, podemos imaginar se posciveis jugadas do oposervie em função de uma de mosais jugados, sua aspersão um movimento diferente de suspendo e toda o platos de jogo cai jor term.

A tecnologia muda rapidamente

A cada dois anos os computadores dobram de velocidade. Hoje não vemos uma aplicação que não seja construída dentro de um framework. As melhores soluções de hoje certamente não serão as de amanhã. A tecnologia é muito volátil, em especial na área de software.

A tecno

Temos soluções para tecnologias, Esta miri componentes ou qual













Requisitos são incompletos

processo em questão, outras vezes, pessoas que gerenciam estes processos. Quando estas pessoas pensam no processo, o chamado caminho feliz costuma ser facilmente visível, os demais caminhos são mais complicados. Normalmente as variações do processo e possíveis erras que ocorrem somente são identificados tandiamente.

As melhores práticas não são maduras

No desenvolvimento de software não nos preocupamos apenas em construir, mas também com a manutenção deste software. As buas práticas consideram questões que funcionaran com processos de desenvolvimento, mas que podem acabar por serem más escolhas na manutenção. As práticas em software possuem pouco tempo de experimentação, não permitindo avaliar seus impactos futuros.

A experimentação da tecnologia é incompleta

A tecnologia muda rapidamente, acompanhando tendências e necessidades. Isso faz com que as tecnologias adotadas muitas vezes sequer cheguem a amadurecer, já ficando desatualizadas. A eonsequência disso é a adoção de tecnologias imaturas em projetos.







O trabalho repetitivo é automatizado

Desenvolver software não é um trabalho repetitivo. Cada projeto é único e traz resultados diferentes. A automatização das atividades de projeto não é algo

Desenvolvimento de software é pesquisa

Quando vamos automatizar um processo conseguimos identificar o caminho feliz, mas os demais não surgem facilmente. O processo de identificação destes caminhos é de descoberta, trabalho em equipe e originam-se de pessoas que falam dialetos diferentes.





A construção é realmente o projeto

Diferente de outros projetos, projetar o software é de fato construí-lo. O planejamento, por mais detalhado que seja, não é capaz de definir com precisão a forma como este será feito. Ocorrerão mudanças por necessidades observadas durante o projeto.

Mudanças são entendidas como simples

É fácil compreender a dificuldade de alterar de lugar uma ponte já construída. Com software esta visão não acontece. Mudancas parecem fáceis, é sempre muito difícil explicar a dificuldade de se lidar com mudanças de última hora.





Referência

Software project secrets: why software projects fail.
George Stepanek
Editora Apress
18, sep, 2005