

Introdução

- Dilema: - “Você não pode avaliar um desenho da interação antes que este esteja construído, mas depois de construído, mudanças no projeto são difíceis (se não impossíveis).
 - No começo do projeto não há nada para testes e construir algo para testar gasta tempo, dinheiro e é um alto investimento para um conceito de desenho que não foi avaliado - isso se torna um impasse e um dilema para os desenvolvedores.
- Solução - prototipagem rápida
 - Através da prototipagem rápida, um desenvolvedor da interação tem a oportunidade de avaliar projetos propostos o mais cedo possível.

5



- Prototipagem é uma técnica, não simplesmente uma ferramenta.
- Essa técnica pode ser eficaz mesmo com a utilização de desenhos manuais ou outros meios não automatizados.
- Um desenvolvedor pode fazer o papel do computador; tais métodos podem ser bastante eficazes.
- O uso da criatividade na construção de protótipos pode dispensar maiores esforços de implementação.

6



Motivação

- Maquetes, modelos e protótipos em geral oferecem a possibilidade de observar precocemente muitos aspectos sobre a natureza final do produto, avaliando idéias e pesando alternativas antes do comprometimento com o produto final.
- O conceito de prototipação é bem sedimentado em áreas mais bem estabelecidas como a engenharia, a arquitetura e a escultura; no entanto não tem sido muito utilizado no desenvolvimento de software interativo.
- Você encomendaria a construção de uma casa sem exigir uma planta ou maquete ?

7



- A técnica de prototipação e o desenho iterativo vem surgindo com ênfase no contexto de desenvolvimento de software interativo, principalmente para a interface do usuário
- Prototipação permite, como Hitchcock fazia com seus filmes, refinamento do produto baseado em feedback dos usuários.
- A prototipação permite a detecção precoce de problemas, reduzindo custos e melhorando a qualidade do produto.

8



- Prototipação também tem a vantagem de promover a participação / comprometimento do usuário.
- Protótipos permitem que os desenvolvedores observem o comportamento dos usuários e suas reações de forma semelhante ao que aconteceria com o produto final.

9



- O uso de Prototipação para apoiar desenho iterativo não significa que os desenvolvedores possam se dar ao luxo de serem preguiçosos ou descuidados com o projeto inicial.
 - A questão é simplesmente que não é possível, somente com o uso de diretrizes de projetos, ter um produto adequado da primeira vez.

10



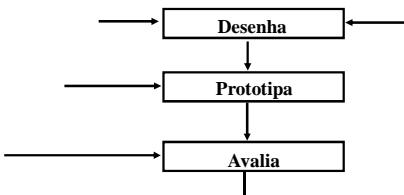
- Ciclo desenha – prototipa

-- avalia

- Desenho / redesenho

prototipação, implementação

avaliação, análise



11



- Os testes com os primeiros protótipos em estágios iniciais de desenho são geralmente realizados pelos próprios desenvolvedores, mas devem também incluir representantes de usuários
- Prototipagem rápida é uma técnica natural que começa com detalhes específicos de um projeto de interação, depois os estrutura e refina dentro do sistema.

12



- Trabalhando do concreto para o abstrato é um modo pelo qual o ser humano naturalmente investiga, entende e assimila novos conceitos e resolve novos problemas.
- Para os usuários e desenvolvedores, um protótipo é uma coisa concreta, enquanto especificações são conceitos abstratos.
- Para Ehn (1990), requisitos de projeto são melhores expressados *mostrando-os* do que somente *falando* sobre eles.

13



- Protótipos iniciais
 - Objetivo: testar globalmente a metáfora de interação
 - Fidelidade pode ser um pouco menor
 - Deve incluir funções representativas
 - seqüenciar subtarefas quando importante.
- Protótipos finais
 - Objetivo: testar partes importantes ou ainda não bem resolvidas.

14



Tipos de Protótipos

- Protótipos podem ser classificados quanto a:
 - Dimensão
 - Horizontal x Vertical
 - Global x Local

15



Classificação por dimensão

- Representação: como o desenho da interação é representado no protótipo?

16



Escopo

- O protótipo inclui todo o sistema ou somente a interface?
 - Quando representa-se somente a interface é chamado de fachada, modelo (mock-up).
 - Neste caso, perde-se em possibilidade de antecipação de funções do sistema.

17



Executabilidade

- O protótipo pode ser executado a qualquer tempo?
 - A implementação e protótipo podem se fundir.
 - Estágios intermediários de implementação podem impedir a exibição de protótipos.
 - Pode-se ter protótipo compilado/ligado ou interpretado.

18



Amadurecimento

- Como o protótipo evolui para o produto?
 - Durante o amadurecimento de protótipo para o produto, é usual um sistema interativo passar por várias etapas:
 - um ou mais protótipos;
 - uma implementação no desenvolvimento;
 - o produto final.
 - O amadurecimento pode ser revolucionário ou evolucionário.

19



Classificação Horizontal x Vertical (Nielsen, 93)

- Horizontal: menor profundidade maior "largura" com relação ao número de características (features) ou funções contempladas no protótipo.
 - A avaliação da usabilidade nesse tipo de protótipo é geralmente menos realística, mas abrange mais funcionalidades do sistema final.
 - São mais usados para protótipos iniciais.

20



Vertical

- Utiliza um menor número de características, abordadas em maior profundidade
 - São mais usados em protótipos quando o desenvolvimento do sistema está mais adiantado.

21



Classificação Global x Local

- Global: visa representar o sistema inteiro, trabalhando tanto em alto nível, paradigmático, como em nível de detalhes.
 - Esta classificação é semelhante à de prototipação horizontal x vertical, mas um protótipo global pode também ter profundidade em algumas partes.
 - Um usuário pode, exercitando com uma protótipo global, ter uma boa percepção do produto final.
 - Protótipos globais são usados por todo o ciclo de prototipagem, do começo até o final.

22



Local

- É um protótipo de um detalhe específico, mas que seja importante o suficiente para potencializar a usabilidade de todo o sistema.
 - Um protótipo local é usado para avaliar alternativas de desenho para um detalhe particular de interação.
 - Um protótipo local é tipicamente isolado (*stand-alone*) dos protótipos restantes.
 - O protótipo local tende a ser usado por um breve período de tempo, quando detalhes específicos de uma ou outra questão de desenho estão sendo trabalhadas.

23



Vantagens

- Sistemas mais fáceis de aprender e utilizar.
- Desenvolvedores menos sujeitos a pressões de prazos.
- Código dos sistemas tende a ficar menor.
- Sistemas são desenvolvidos com menos esforço.

24





- Usuários demonstram um maior nível de satisfação e aceitação do produto.
- Maior entusiasmo dos usuários.
- Melhoria da comunicação sobre o sistema.
- Protótipos geram uma referência comum para discussões.
- Prototipação facilita uma resposta mais rápida dos desenvolvedores.

25


Synergia
Engenharia de Software e Sistemas



- Permitem um maior número de iterações no desenho.
- Facilitam a assimilação de um novo paradigma pelos usuários de um sistema que está sendo substituído.
 - Facilita quebrar a resistência de usuários a mudanças.

26


Synergia
Engenharia de Software e Sistemas



Riscos

- É perigoso adotar-se prototipação sem um comprometimento de todas as partes envolvidas.
- Gerentes podem ver a técnica como desperdício de dinheiro.
- Pode haver uma redução da disciplina da equipe que tende a enxergar a prototipação como um “treino” que não é para valer.
 - Prototipação não é “brinquedo” e deve ser feita com metodologia.

27


Synergia
Engenharia de Software e Sistemas



- Equipe de desenvolvedores e usuários podem perder entusiasmo após a apresentação de várias versões de protótipos.
- Prototipação pode ser em si um processo caro e trabalhoso.
 - Por exemplo, o protótipo de um sistema de controle de tráfego aéreo para a aviação americana custou milhões de dólares e demorou 2 anos.

28


Synergia
Engenharia de Software e Sistemas



- Protótipos podem ser confundidos pelos usuários com o sistema verdadeiro; isso cria falsas expectativas com relação a prazos.
- Usuários são bons em produzir críticas mas não são bons em antecipar ou articular necessidades.
- Há limitações quanto à fidelidade com que protótipos representam os sistemas verdadeiros: isso deve ser bem compreendido.
- É preciso cuidado para não criar-se falsas expectativas de provimento de mais funcionalidades do que o real.
 - Por exemplo, se o sistema final não permitirá o uso de mouse, também não deve o protótipo permitir.

29


Synergia
Engenharia de Software e Sistemas



- Plataforma de desenvolvimento do protótipo deve ser bem escolhida para permitir a representação das características desejadas do protótipo.
- Armadilha de “superdesenhar” (*overdesign*): tendência a sofisticar demais o protótipo.
- Apresenta dificuldades para o desenvolvedor acostumado com a abordagem *top-down* de projeto.
- Tendência do desenvolvedor a subestimar dificuldades de implementação de tarefas prototipadas como fachada (*stub*).

30


Synergia
Engenharia de Software e Sistemas



Ferramentas de Prototipação

- Permitem ganho em produtividade
 - Permitem diminuição significativa do tempo de desenvolvimento
- Muito importantes em grandes projetos
- Facilitam a gestão da evolução de desenhos de projetos, muito importante dada a natureza iterativa do ciclo desenho / avaliação

31


Synergia
Engenharia de Software e Sistemas



Características desejadas

- A escolha da ferramenta é muito importante para um projeto. Observar principalmente as seguintes características:
 - Facilidade de desenvolver e modificar telas.
 - Permite mistura de parte gráfica com textos ?
 - Suporta o tipo de interface que se deseja desenvolver.
 - Suporta uma boa variedade de dispositivos de entrada / saída.
 - É fácil ligar telas ou modificar ligações.

32


Synergia
Engenharia de Software e Sistemas

- Permite chamada de procedimentos externos.
- Permite importação de textos, gráficos e outras mídias.
- Possui atributos de usabilidade: facilidade de aprender, facilidade de uso, retenção, etc.
- Tem bom suporte de fornecedor.

33



Exemplos de Ferramentas

- *HyperCard* - ferramenta muito utilizada.
 - O *HyperCard* permite associação com a produção de documentação.
 - Ferramenta fácil de aprender, fácil de usar e possui biblioteca de objetos de interação embutida (botões, menus, e outros).
 - Possui linguagem script que permite extensões para objetos embutidos - *HyperTalk*.

34



- *SuperCard, SmathersBarnes Prototyper* - Macintosh
- *MacroMind Director* - produto baseado no Mac, originalmente criado para desenvolvimento instrucional e programas de ajuda ao aprendizado. Permite animações.
- *Visio* – diagramas em geral, Microsoft

35

