

# Game Design com Prototipação

**Tiago Tex Pine**  
tex@texpine.com

# Protótipos

Prototipação é o processo de criação de um ou vários modelos funcionais das suas idéias. O objetivo é testar a viabilidade de cada uma delas. **Com protótipos, você está tentando descobrir o game design.**

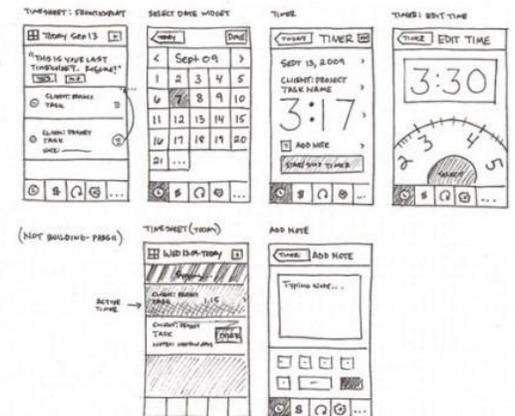
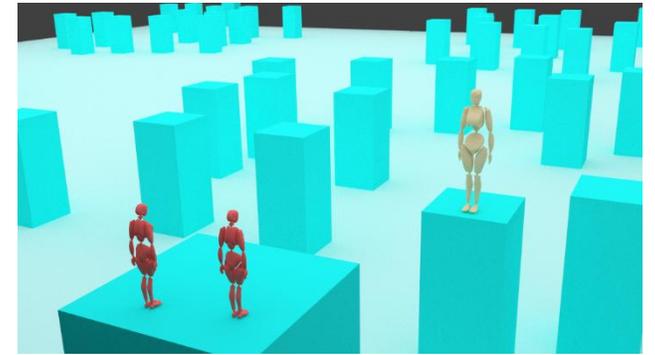
Protótipos são como rascunhos: crus, às vezes feios e inacabados. Porém toscos de propósito: ao eliminar todos os vários componentes de que um jogo digital é composto - tais como som, paleta da UI, direção da arte 3D, otimização da programação da física, etc - protótipos de game design lhe permite focar no essencial do game design.

Muitos iniciantes querem pular direto para "fazer o jogo de verdade". Mas um jogo digital de qualidade é um processo caro em termos de tempo e recursos. Começar sem um bom plano demandará re-trabalho e mais recursos depois.

# Protótipos

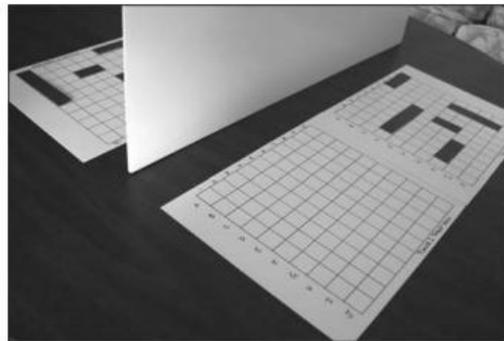
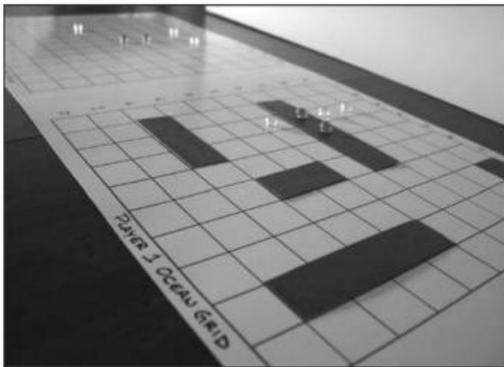
De modo geral, muitas coisas dentro de um projeto de jogo digital podem ser prototipadas:

- Core gameplay -> protótipos físicos e digitais (a seguir)
- Arquitetura de código e desenvolvimento -> experimentos com ferramentas.
- UX, o Flow de interfaces e estados do jogo -> HTML.
- Direção de arte -> "fakes" estáticos das telas principais do jogo e da UI
- Tecnologias server-side -> com apps para testar carga e resposta



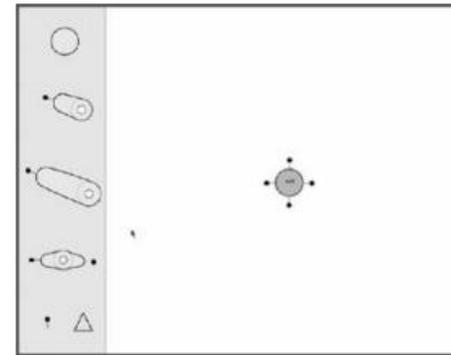
# Protótipos de Game Design

Podem ser feitos de duas formas: protótipos **físicos** e protótipos **digitais**.



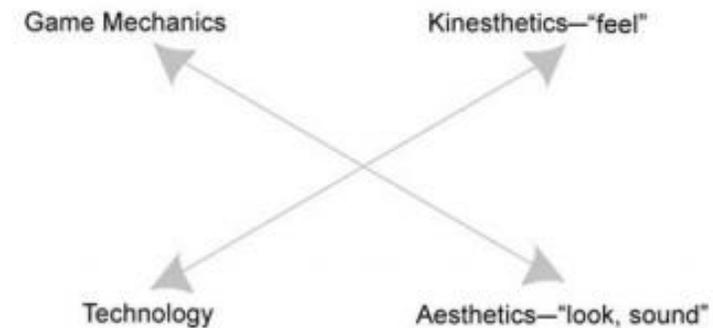
## Físicos

- Permite que você foque no design, e não no desenvolvimento.
- Sem apego a código e arte.
- Mudanças no design podem ser super rápidas.
- Permite que qualquer membro da equipe participe.
- Resposta em *realtime* ao feedback dos usuários. Mude as regras e features na mesma hora!

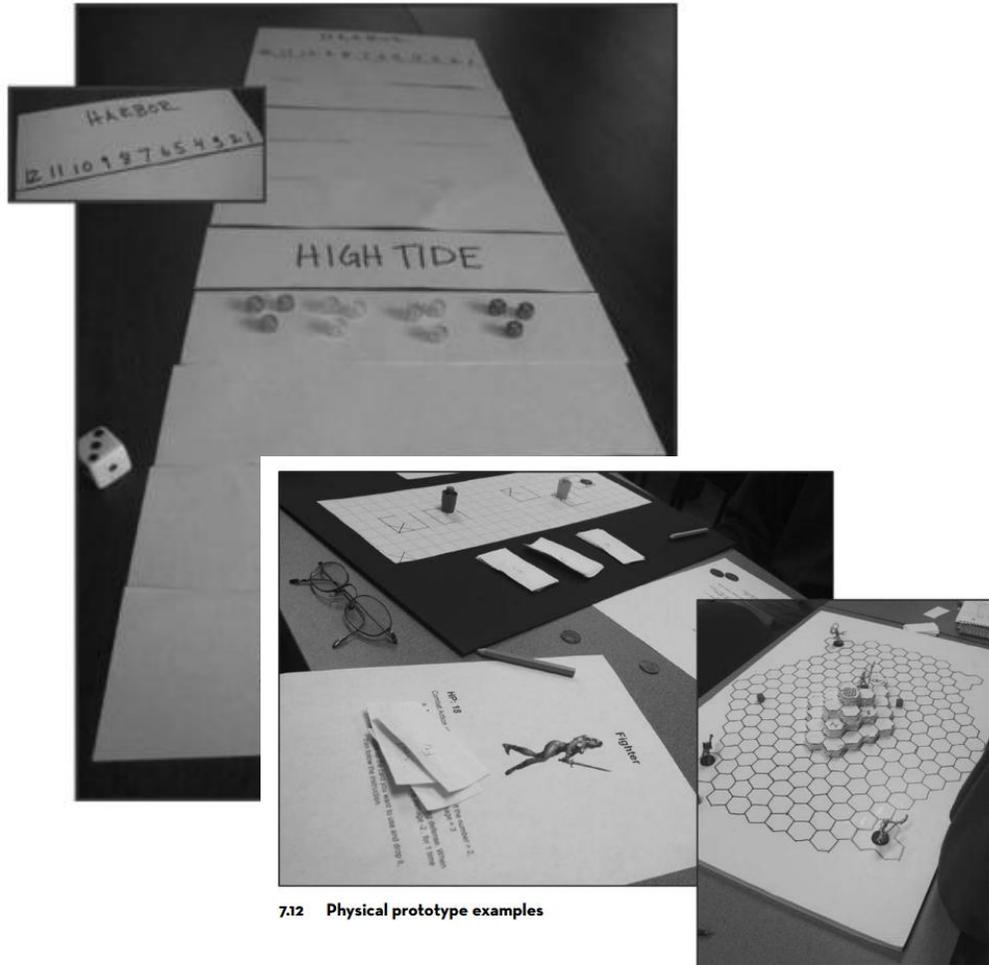


## Digitais

- Permite prototipar interações precisas em *real-time* e inteligência artificial.
- Permitem descobrir as outras áreas do design que os protótipos físicos não podem cobrir:



# Protótipos Físicos



7.12 Physical prototype examples

Protótipos físicos funcionam muito bem para:

- Board games
- Card games
- RPGs
- Jogos digitais com:
  - O loop baseado em turnos,
  - Com objetos que possam funcionar com no máximo 15 propriedades mais importantes.
  - Muito bom se o jogo demandar algum tipo de controle territorial.

# Protótipos Físicos

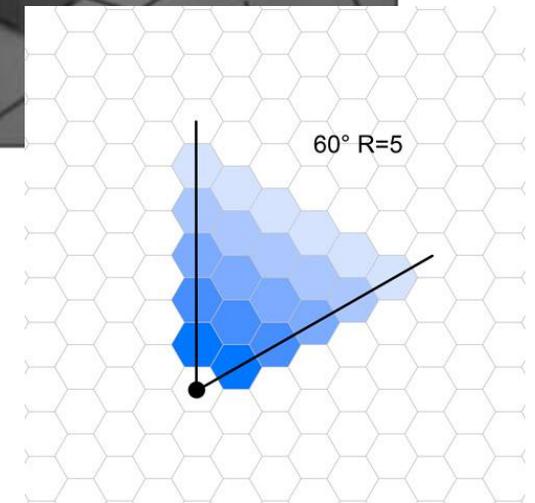
No entanto, com alguma criatividade, é possível abstrair até mesmo as mecânicas em *real-time*. Considere o seguinte protótipo de um *FPS*, cujo principal problema é similar movimento e tiro.

Cada jogador tem um peão e recebe 9 cartas:

- 1 de Mover 1 espaço
- 1 de Mover 2 espaços
- 1 de Mover 3 espaços
- 1 de Mover 4 espaços
- 2 de Virar para outra direção.
- 3 de Atirar

Regras básicas:

1. O cenário é montado com *um grid hexagonal* e paredes de papel.
2. Cada jogador escolhe 3 cartas e as colocar numa pilha na mesa.
3. Cada jogador vira suas cartas ao mesmo tempo.
4. Cartas de tiro são resolvidas com linhas imaginárias traçadas pelos jogadores a partir de "cones de visão" no grid hexagonal (que simula o FOV).



# Protótipos Físicos

Regras básicas, continuação:

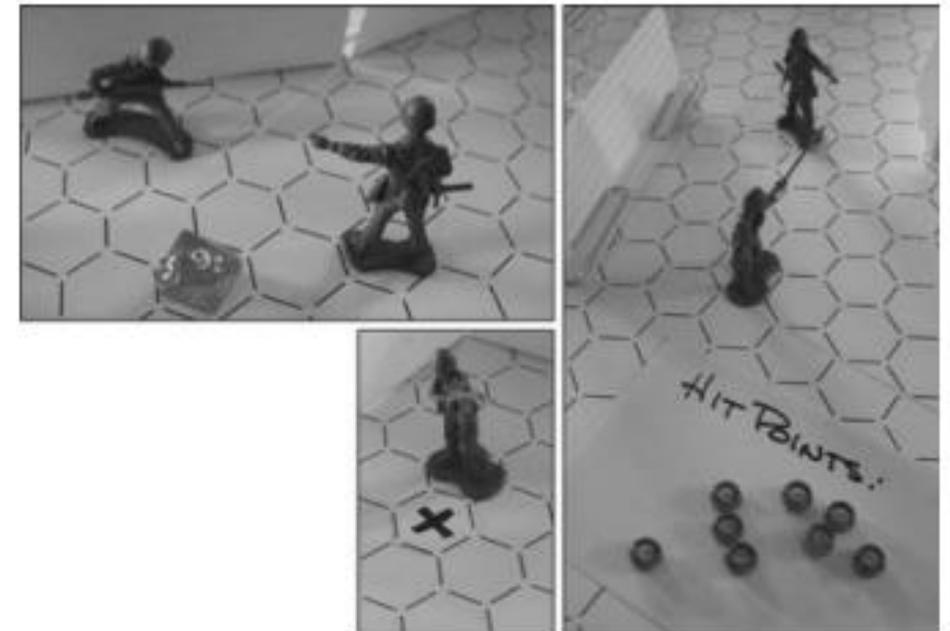
5. Cartas de virar de direção são resolvidas ao mesmo tempo.
6. Cartas de movimento são resolvidas ao mesmo tempo, porém "colisões" podem acontecer. Como resolvê-las?
7. Repita esses passos para cada carta no stack de 3 de cada jogador.

Certamente, tal protótipo não o ajudará a definir exatamente os features e regras do jogo final, mas já seria uma base sobre o qual você pode começar a introduzir features e regras dos aspectos "únicos" do seu jogo.

**Exemplo:** suponha que você queira introduzir um personagem com escudo invulnerável na "frente" dele vs. um personagem mais rápido. Apenas mude as cartas e as regras.

+Possíveis adições:

- Sistema de pontos
- Percentual de dano baseado na distância.
- Health
- Kits de Health
- Munição
- Armas diferentes



# Dicas Gerais



7.14 More physical prototype examples



- Caso esteja tentando simular movimento livre no espaço 3D, **um grid hexagonal** é melhor que um quadricular.
- Não abuse de dados ou sistemas aleatórios, a não ser que o jogo os tenha. Se o jogador deve estar no controle no jogo final, ele deve estar no controle no protótipo.
- Use outro designer como "inteligência artificial", e não os próprios playtesters.
- **Regras != Features.** A introdução de novas Regras visa corrigir problemas. A introdução de novos features é para expandir as possibilidades do gameplay.
- Protótipos físicos levam uns dias de refinamento até funcionarem em sua "forma básica". A maioria dos designers desiste aí. Mas se vc passar por essa etapa, terá um framework funcional sobre o qual testar outras idéias.

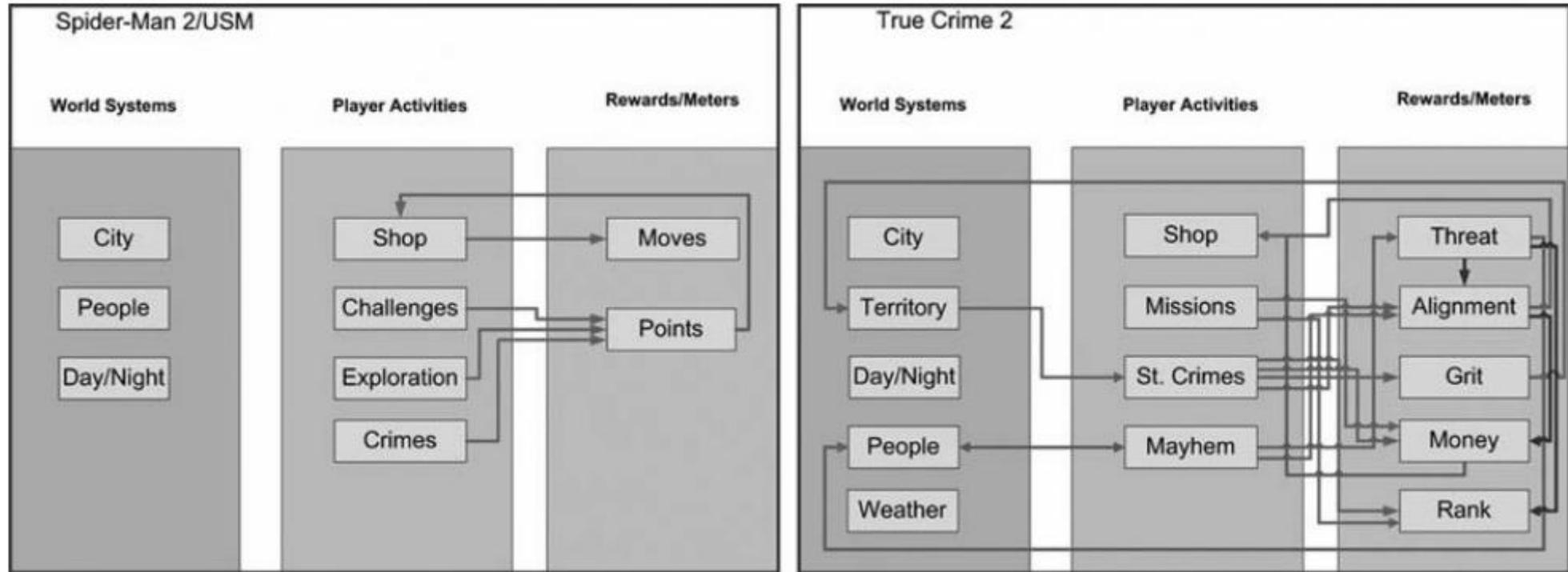
# Passos

1. Defina os Fundamentos do core gameplay
2. Defina a estrutura do protótipo - o que é melhor para simular as idéias do core gameplay?
3. Detalhes formais: Procedimentos e Regras específicas que tornam a coisa toda coerente.
4. Refinamento através de playtest.



7.13 Physical prototype with procedures outlined

# Protótipos Digitais

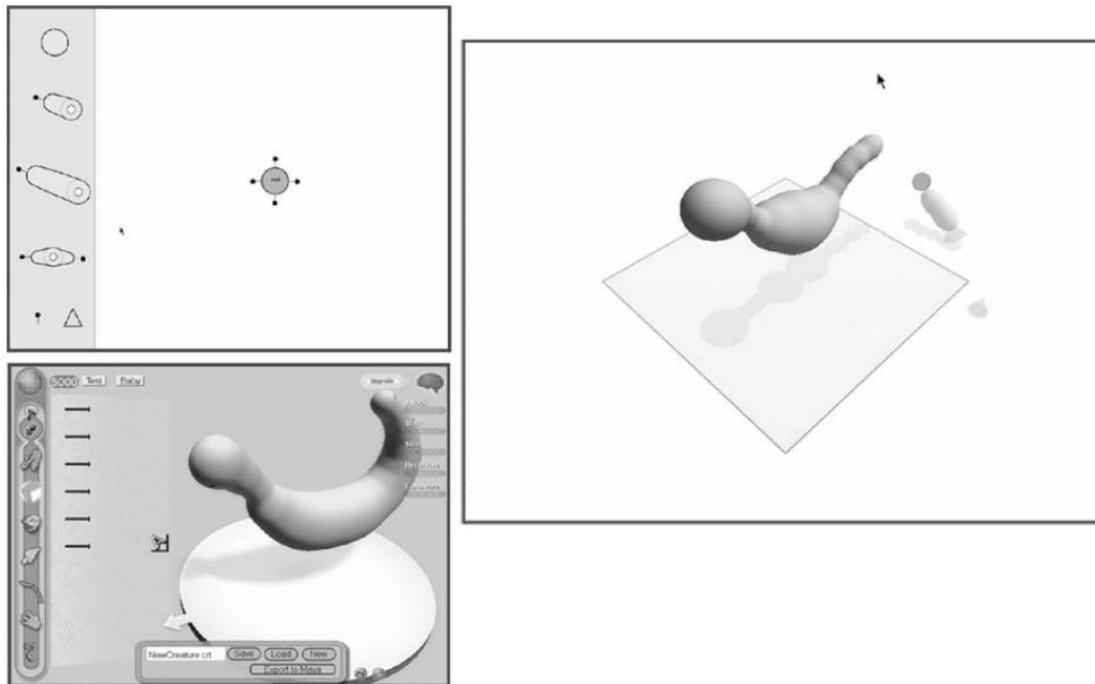


Diagrams courtesy of Activision Central Design (Jeff Chen and Carl Schnurr)

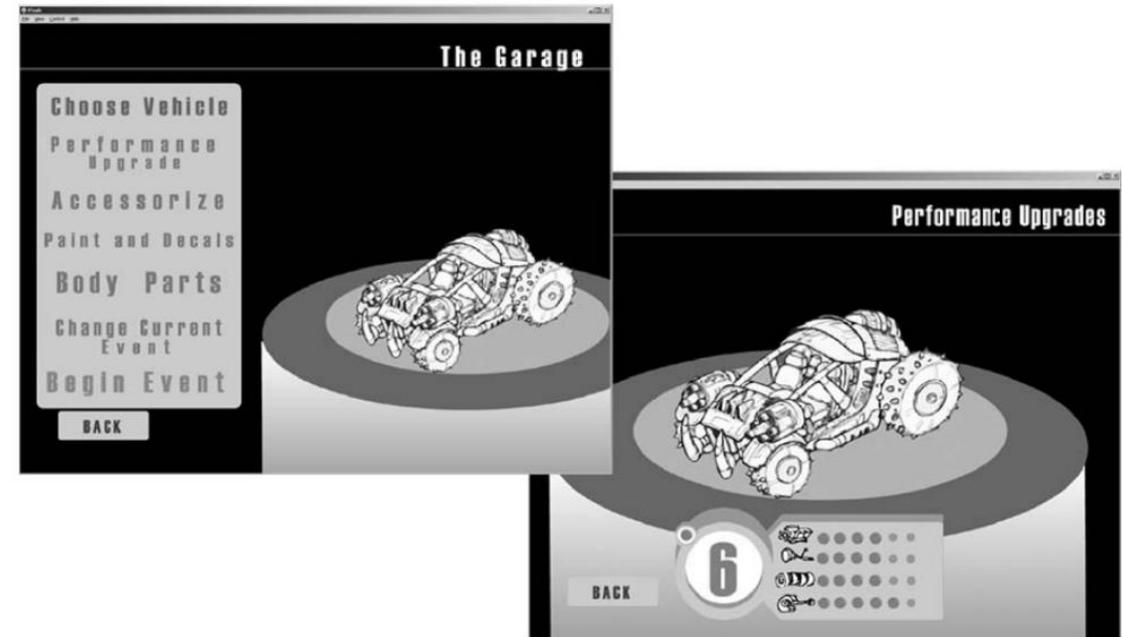
Idealmente, um protótipo digital deveria ser feito para cada área principal do seu core loop. No exemplo acima, em True Crime 2, um protótipo para o Shop, um para Missions, etc.

# Protótipos Digitais

De modo que quando falamos em "fazer um protótipo", estamos realmente falando em fazer vários deles, dependendo do quão variado é o design completo do jogo.



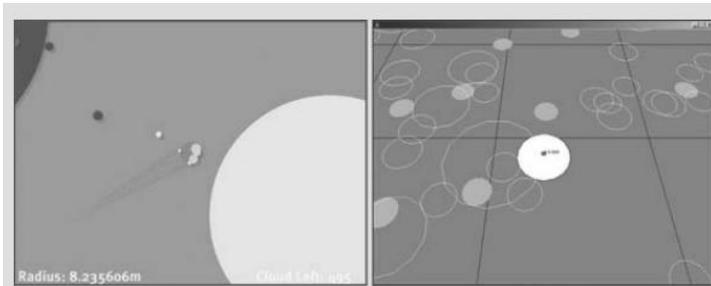
8.3 2D and 3D creature editor prototypes for Spore



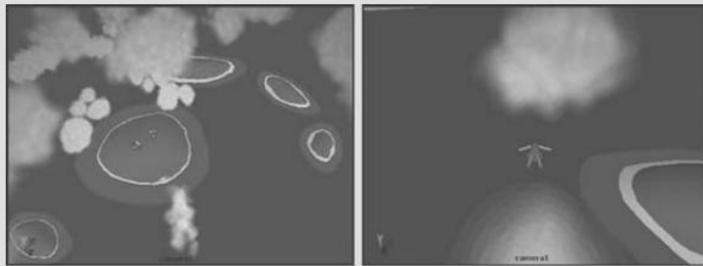
8.6 Interface prototypes for Jak X: Combat Racing

# Protótipos do jogo "Cloud"

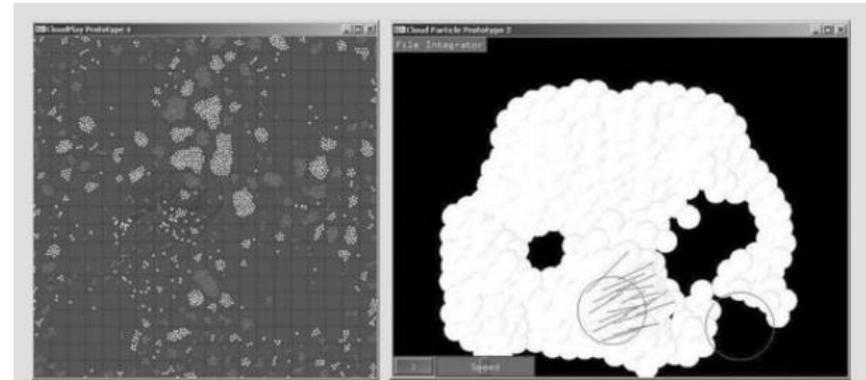
A título de ilustração, observe como os designers de Cloud dividiram seus muitos protótipos:



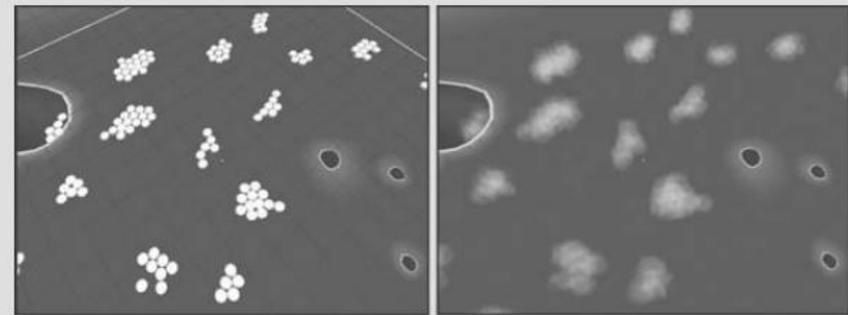
**Figure 1** 2D gathering clouds prototype for Cloud, left, and control, camera, and rudimentary gameplay prototype, right



**Figure 2** Camera simulation prototype; left shows camera zoomed out for view of entire sky, right shows camera zoomed in to fly close to the character



**Figure 3** Particle simulations prototypes



**Figure 4** Cloud simulation layer (left) and with rendered clouds overlying simulation (right)

# Os "3 Cs"

## 1. Câmera

- Definições básicas: Isométrica, Lateral, first-person, third-person.
- Mas mais importante: *como a câmera se comporta / reage / dá feedback* à medida que o jogador executa ações?
- Observe os sutis porém importantes movimentos de camera aqui conforme o personagem esquiva, aplica o golpe ou é golpeado: <https://www.youtube.com/watch?v=eDmVSQU6cni>

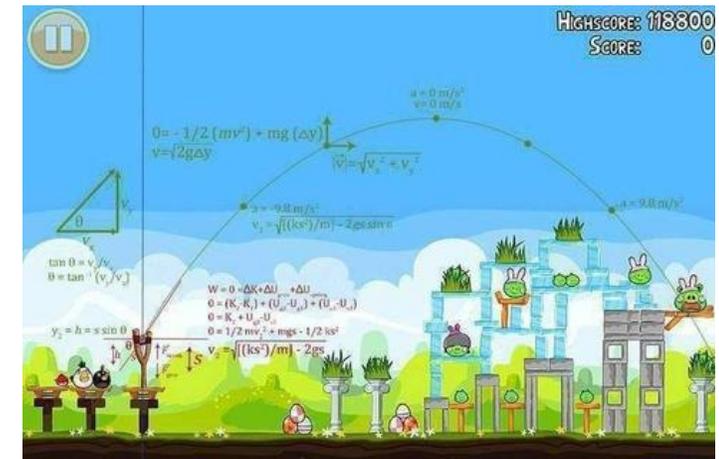


## 2. Controles

- É **muito importante** prototipar controles. Tempo e precisão de resposta de qualquer coisa que o jogador faça é crítico para a experiência de qualquer jogo.
- Controles que não espondem quebram a imersão e comunicam baixa qualidade.

## 3. Character (Personagem)

- Coberto na aula passada, a prototipação nesse caso são os muitos rascunhos e interações até se chegar num resultado final..



# Protótipos Digitais

Protótipos digitais são **aplicativos descartáveis** cujo único propósito é testar um conceito de game design. A regra mais importante de Protótipos Digitais para programadores:

**O código vai ser JOGADO FORA!**

O objetivo é **descobrir como fazer**, e não fazer o jogo em si. Use outras linguagens e frameworks se cumprir o objetivo de game design mais rápido.

# Regras Gerais

Regras gerais para trabalhar com protótipos:

1. **Sempre comece com uma Pergunta.** É ela quem vai determinar o que você vai fazer. Prototipar por prototipar é perda de tempo.
2. **Mantenha a mentalidade de experimentação.** Designers às vezes se apegam demais aos sistemas e demoram muito tempo para mostrá-los, tentando deixá-los "perfeitos" e com o "feeling certo". Mostre e teste!
3. **Trabalhe rápido:** De modo geral, a produção e teste de protótipos deve levar *entre 2 dias e 2 semanas*, e não mais que isso.
4. **Trabalhe barato:** faça da forma mais rápida possível. Padrões de código não importam, use GOTO. Arte é legal, mas simplesmente imprima ou copie o que achar na internet, tipo DeviantART.
5. **Decomponha os problemas:** faça um protótipo para cada coisa. Fazer um app com todos os features é fazer o jogo inteiro.

# Exercícios



1. Faça o design de um protótipo físico para *Portal*.
2. Faça o design de um protótipo físico para *Hotline Miami*.  
<http://hotlinemiami.com/>
3. Apresente idéias para um protótipo físico **para o seu jogo**.
4. Apresente plano para protótipo digitais **do seu jogo**.