Foundations of Interactive Game Design (80K)

week three, lecture three
Today

- Prior 80K game demo
- Announcement
- Prototyping
- Quiz if there’s time
- What’s coming up
Demo: 

Lumos

by Ashwin Nathan & Ryan Ho
Announcement
PARTICIPATE IN A USER STUDY & WIN A $50 AMAZON GIFT CARD!!

Are you interested in taking part in an interesting experiment? And win a $50 gift card to boot? Then sign up! Chances of winning are 1 in 10!

1. Go to: http://goo.gl/DTdhT
2. Sign up,
3. Show up at Kresge 317 PC Lab in your scheduled time!!

Notes: You have to be a native speaker, over 18, and have previous gaming experience. Feel free to distribute this to your friends, and email the researcher Serdar Sali (sali@soe.ucsc.edu). We have limited slots so sign up now!!
What do you do after brainstorming?
Edit and refine

- Will Noah and the TAs think it is innovative?
- Do we have the skills needed?
- Do we have the time needed?
- Will we be happy to have made it?
- Do we know how to start prototyping a playable, formal system?
- Would it be compelling to play?
Prototypes
Why prototyping?

• You want to experiment with ways to turn your game concept into a system
• You want to involve your full team — and make sure you’re all on the same page
• You want rapid iteration
• You want to make sure the game works before you start writing code, making art, etc — so start with most important parts and biggest risks
What kind of prototype?

- Many games have deep gameplay systems — role-playing, real-time strategy, etc — so prototype the system.

- Other games have simpler systems and deeper level design — puzzle, platformer, adventure, etc — so prototype levels.

- Two examples from CMPS 170.
Stack and Deploy
Stack and Deploy is...

- A trading card game:
  - Players create decks from card collection
  - Play cards to create customized units
  - Commanders determine personality
Stack and Deploy is...

A strategy game:
- Autonomous units are deployed to battle
- Capture strategic objectives, combat enemies
- Destroy the enemy base!
Prototyping

- System-oriented prototyping
- Game map essential to system, prototyped different maps
- Different rules for combat and movement
- Different cards, different deck types against each other
- Game team member did AI’s work, tested with outsiders
Prototyping

• Board game tools are very effective for system-oriented prototyping

• Use maps, cards, dice, counters, etc

• Keep everything as reconfigurable as possible — be able to change while testing

• Test with people outside your group
Penumbra
Penumbra

Demons have entered your world and taken refuge in the shadows.

A puzzle game where shadows are manipulated by the rotation of the level.

Theme explored: dynamics of shadows

Creates a literacy relating object shape with the shadow it casts.
Level Progression – Intro

First level of the game.

Introduced to the concept of shadows as an obstacle.

Does not require use of the game play mechanic.
Level Progression - Rotation

Given the knowledge just gained, level appears impossible.
Rotating the world, which changes the lighting, reveals the solution.
Some prototype questions

• Will the basic mechanic work?
• Can we create levels that will lead people astray interestingly?
• Also discovered: play strategy they wanted to prevent
Prototyping
Penumbra physical prototypes

• Level-oriented prototyping, but with basic mechanic played in each level
• Figured out ways to make levels challenging
• Figured out how people respond to leading elements in the geometry (e.g., arches), their strategies for movement, etc
• Figured out camera can’t show full level
Textbook example
FPS Example

- We all believe you can paper prototype a board game—it’s already paper—but what about real-time computer games?
- What are some design questions you might ask?
- What are the basic actions a player can perform?
- How big should the level be?
- What’s the level design (for a particular level)?
- What objects are in the environment and how do they help or hurt the player (weapons, powerups, health…)?
- Where are spawn points?
Simulating real-time with a paper prototype

• To answer questions, may need to simulate real-time gameplay

• Build stack: each player chooses three action cards and places them face down

• Reveal: each player turns over his top card.

• Resolve shoot cards: players with shoot cards fire in the direction their unit is pointing in a straight line. Simultaneous shots are resolved with dice.

• Resolve turn cards: Players with turn cards turn their unit. The order of simultaneous turns is resolved with dice.

• Resolve move cards: Players with move cards move their units the number of spaces they selected. Resolve multiple move cards with dice. Cannot occupy the same cell.

• Repeat steps 2-5 for the second and third cards in the stack.
Physical prototypes

- What are the strengths and weaknesses of this kind of prototype?
  - Create much faster than on computer
  - Change much faster than on computer—even during playtest
  - Everyone can participate

- Allows rapid iteration
  - Can’t explore certain areas of game feel
  - Can’t figure out if game technology will work
  - Can’t execute complex processes—but can couple with computational support
Your prototypes

- Due in section next week, with your core concept document (outlined in syllabus)
- Should be playable for in-section demo
- For best results, have people outside your team play before you bring to section
- As you build, try things, revise, and try again
- Questions?
What if you can’t figure out how to prototype?
Prototype problems

- If you can’t figure out how to prototype your idea
- Brainstorm w/ your team
- Brainstorm w/ reader/tutors and/or TA
- If your TA agrees there’s no good physical prototyping strategy ... you can storyboard!
Storyboards
Storyboards

• For this course, storyboards will help you clarify how players begin playing your game.

• This will help clarify why players are playing.

• What teaches them how to play — what signals to try things, what feedback?

• What hooks them on the experience and keeps them playing?
Storyboards

• The images do not have to look pretty — though you can use storyboards to work out art style

• The idea is to work through the flow of potential sequences of actions

• What are the first things the player is asked to do? What happens with success or failure?

• Not asking for cutscenes
Creating storyboards

- Start out with what sequences you need to show — one for each version of each major action at the game opening

- To create initial images, hand sketch, use GIMP, screenshot other games, or...

- Organize the images, caption them, refine
Brief look at Celtx?

(Could also use Powerpoint, etc)
Also think of sequence relationships

(from Jakub Linowski)
Show, discuss, and argue storyboards as a team

Works better than doing it with design doc