Today

- Testing Lecture
- Riot Testing Lecture
- Break!

- In Class Work: FOCUS TEST
  - Plan Test
  - Run Test
  - Report Results
Testing.
Everyone needs it. How come nobody likes it?

Quality Assurance
“any systematic process of checking to see whether a product or service being developed is meeting specified requirements”

in other words…

Making sure your game does what you want it to do, and does it well.
who?
Types of Testing

- Technical Testing.
  - Looking for defects.

- Playtesting:
  - You play your game!

- User testing – someone else plays your game
  - Focus Testing
  - User Testing
Technical Testing Basics

- Technical Standards (Have them.)
- Bug database
- Build and test before check in
- Daily builds, daily play throughs
- Planned feature cuts
- Code Freezes (Code Reviews)
- Asset Freezes
Advanced Technical Testing

- Maintain ‘working’ feature lists
- Agree how features should work
- Test newly implemented features as soon as they are ‘done’.

- Checklist testing: methodical testing of all implemented features, all branches, all possible states. (Regression testing – testing features that haven’t changed since last checkin to make sure they are still working)

- Remember that checklist testing – and strict feature checking- doesn’t cover freeform play: getting in the game and just ‘doing stuff’ to see what happens. Test that too!
Be Kind To Your Fellow Engineers

Write good bugs, not bad.

- Good bugs include: What happened. What should have happened. How to reproduce. How common & serious the bug is. Supporting data, like screenshots, or numbers, is useful.

- Know the difference between a criticism and a critique.

- Read “Bugs & Bug Reporting” for more details.
Why: User Testing

- Engaging game play is hard: Find out how users react to your game.
- Gauging overall game difficulty/challenge.
- Test driven design – needs testing!
- Check your User Interface against actual users.
Test driven design does not mean ‘do what your testers tell you to!’
How: User testing

- Remember the ‘Experimental Method’? *Form a hypothesis, create a test for it, gather data, analyze results.*
  - *This isn’t really that different!*

- Have a Question (or a Goal) for each focus test!

- Develop a method: Standardize your approach.
  - Introductory Script
  - List of observations to look for
  - Set of questions to ask your testers

- Review your data, and decide what your team should do about it!
tl;dr

- For each test you run:
- Who are you testing with?
- What are you testing with?
- Why are you testing?
Data Collection

- Direct Observation
  - Ask players to talk out loud about what they are doing.
  - Take notes!

- Surveys
  - Not real time; people forget what they were doing quickly.

- Interviews
  - People don’t like to disappoint you, so they often soft pedal their comments.
Examples

- Observer script/survey
- Focus Test Form
Focus Test Workshop

- Prepare for test (15 minutes)
  - Decide on a question, data collection method(s),
  - create any useful paperwork

- Run Testing (60 minutes)
  - Everyone on your team should get a chance to observe at least one tester, and everyone on your team should play AT LEAST 2 games.

- Post test discussion (15) followed by quick report to the class on your results: what did you learn, how did it go?