CSC 217 Lab 04

Design

Lab Overview

- Deadlines & Reminders
- Project 1 Part 1 Advice
- Design Comparison
- Design Integration
- Design Extension

Deadlines & Reminders

- Deadlines
  - Check deadlines in lab writeup!
- Reminders
  - Run static analysis tools
  - Check Jenkins console output for compiler errors if red X
  - If you pair program, you MUST note it in the commit message or there will be deductions
  - Students who don’t let their partner contribute will receive deductions
  - Javadoc!
    - Generate HTML API pages from your Javadoc in the doc directory
    - Check Piazza “Javadoc 101” main points - class comments, method details, and throws

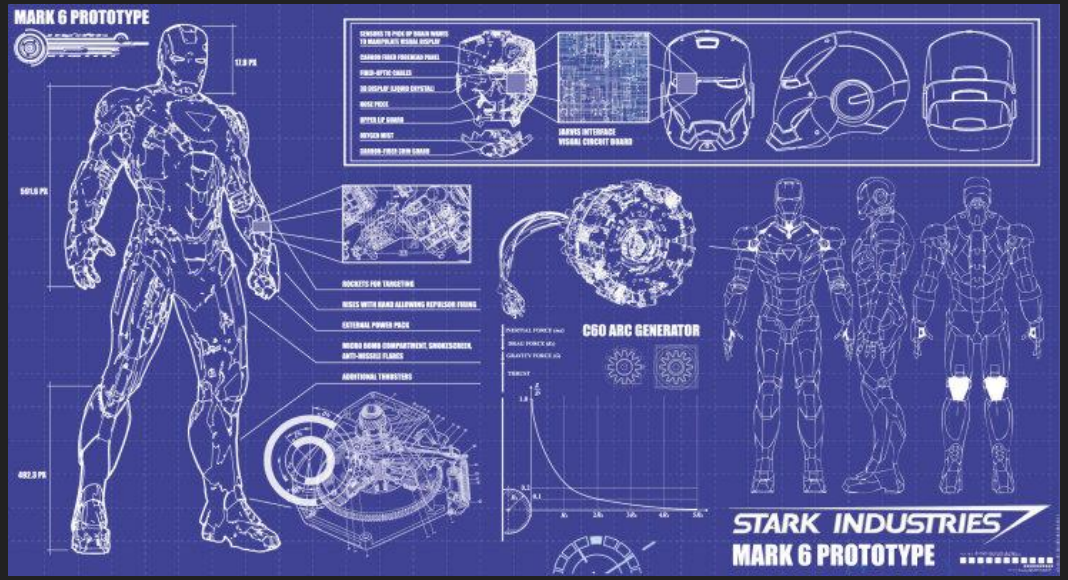
Advice for Project 1 Part 1

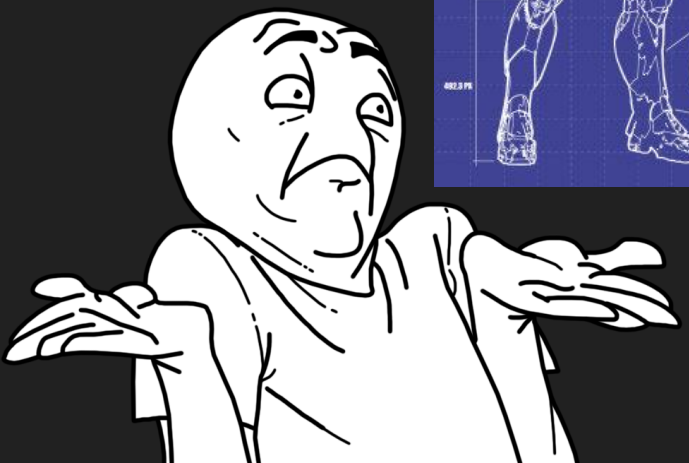
- Part 1 is the most “Unknown” assignment
- Classes:
  - Look for **nouns** in the project
  - Compare to PackScheduler/WolfScheduler
- Methods:
  - Walk through each Use Case & all subflows / alternate flows
  - Is there a chain of methods to call?
- Rationale:
  - It’s **75 points of the 105 points for the Part 1 grade**
  - Describe *each* field, *each* method, *each* class, and *each* relationship
  - Think of it like smushing and compressing *all your Javadoc* into one writeup

Deadline 4 days

DESIGN

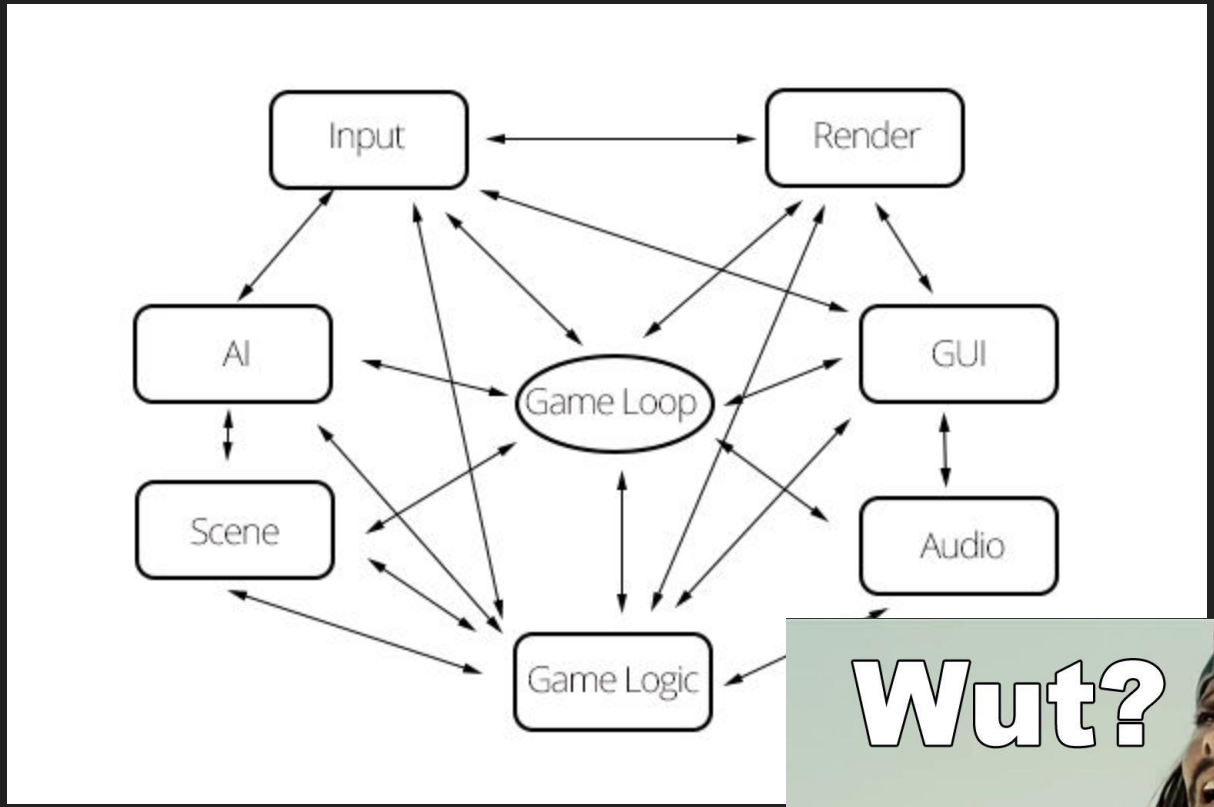
HOW DOES DO?

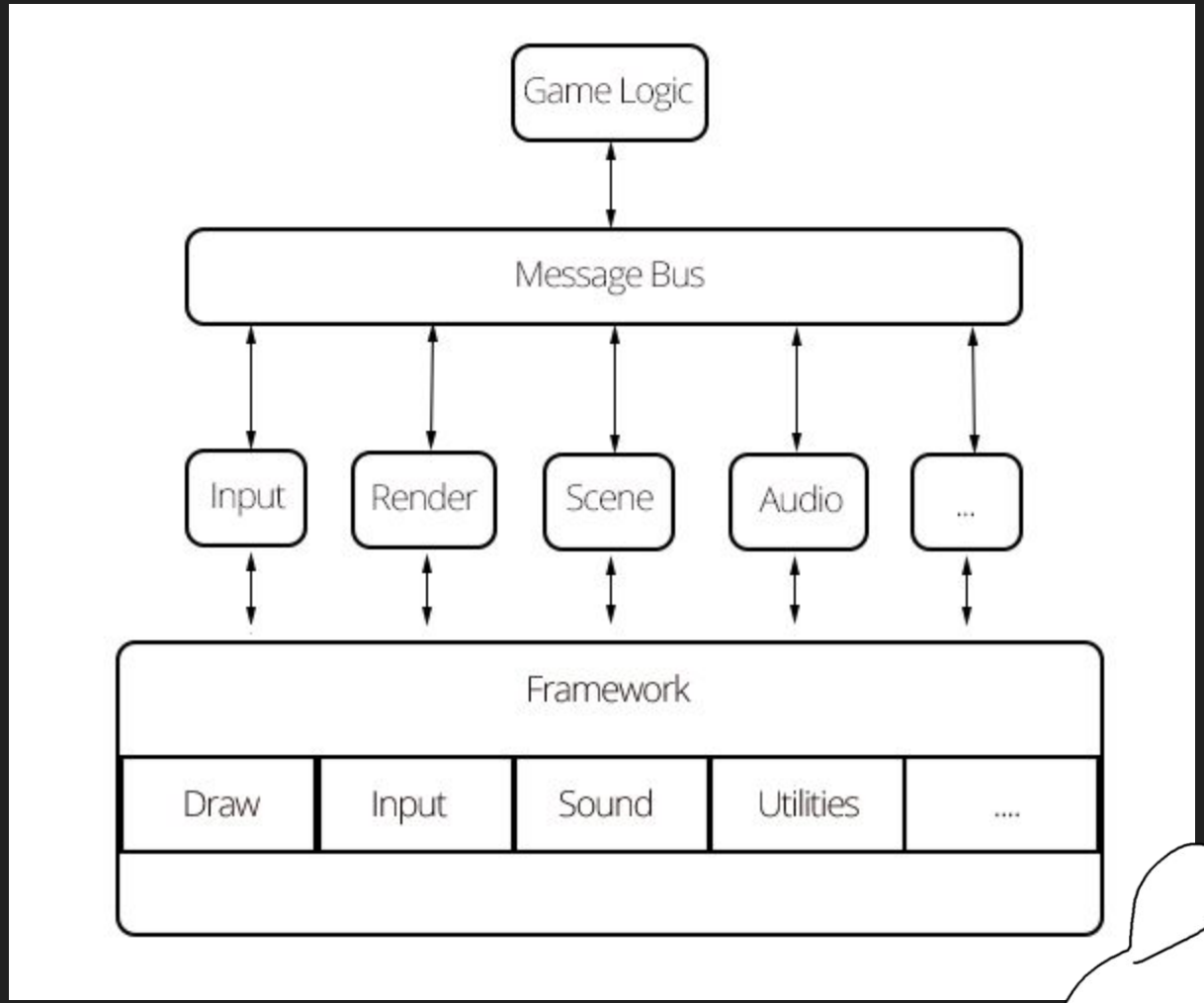




Bad Design

- Don’t make spaghetti!!
- Name things clearly
- Stick to conventions
- Document!
- Stay in touch with requirements

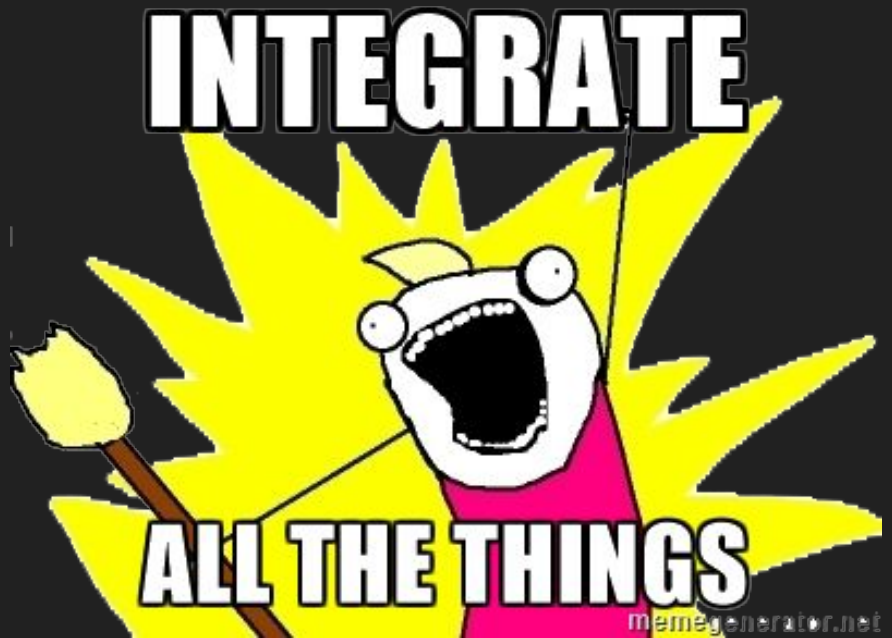

Input
Render
AI
GUI
Game Loop
Scene
Audio
Game Logic
Wut?


Game Logic
Message Bus
Input
Render
Scene
Audio
...
Framework
Draw
Input
Sound
Utilities
....

Design Comparison

- Read the requirements
- Compare and contrast the two designs
- Make a recommendation to management about the designs and any recommended changes
- Submit via Google Forms
- Expecting multiple sentences of detail, think:
  - coupling
  - cohesion
  - extensibility
  - redundancy
  - maintainability
  - encapsulation
  - ease of implementation
  - ease of client use
  - logical structure (it just “makes sense”)

INTEGRATION TIME!



Design Integration

Integrate WolfScheduler into PackScheduler!

- edu.ncsu.csc216.pack_scheduler.course
  - Straight forward copy of source files
  - Some updates to the tests to remove Event functionality
- edu.ncsu.csc216.pack_scheduler.io
  - Bring back together read/write functionality for CourseRecordIO and CourseRecordIOTest
  - Work with SortedList instead of ArrayList
  - Update imports
- edu.ncsu.csc216.pack_scheduler.catalog
  - CourseCatalog - New class (but you can use WolfScheduler and StudentDirectory to help you)
  - Same for CourseCatalogTest

TEST INTEGRATION

Design Extension

Design faculty functionality (as described in UC8, UC9, and UC10)!

(You aren't expected to implement faculty functionality…. yet!)

Start with one of the earlier designs and update as appropriate for faculty

- Dia
- UMLetino
- Draw.io
- Gliffy (14 day trial)
- LucidChart - (education account with NC State)
- PlantUML (free, difficult at first but super pretty in the end)

Wrap-Up & Team Reminders

General Wrap-Up

- Deadline Reminder (see the board)
- Exchange contact information with your partner
- Make a plan for finishing up the lab

Participation Outside of Lab (Guess which the teaching staff prefer?)

- If you pair program/design, **note that in the commit comments so everyone gets credit!**
- If you split the work, at least one contribution by each partner

Contribution

- If you pair program/design, you **MUST** note it in your commit messages or there will be deductions
- Students who don’t allow their partners to contribute will receive deductions
- Students who don’t contribute will receive deductions

Reflection & Collaboration

- Complete Collaboration Check-in before the lab deadline!

Record Tasks & Owners

Tasks only get done when someone owns them!

Identify the tasks required to complete Lab 4

- Edit README.md to list the tasks required to complete Lab 4 (at top of README - should come before Lab 3 tasks)
- Add an owner to each task
- Add a deadline to each task

Deadlines should be at least 48 hours before the lab deadline so team members can help out and finish the lab if a team member runs into issues.

Notify team early if you run into problems with your tasks!