CSE1720

Week 02, Class Meeting 04 (Lecture 03)

Winter 2013 ◆ Tuesday, January 15, 2013

YORK UNIVERSITÉ UNIVERSITY

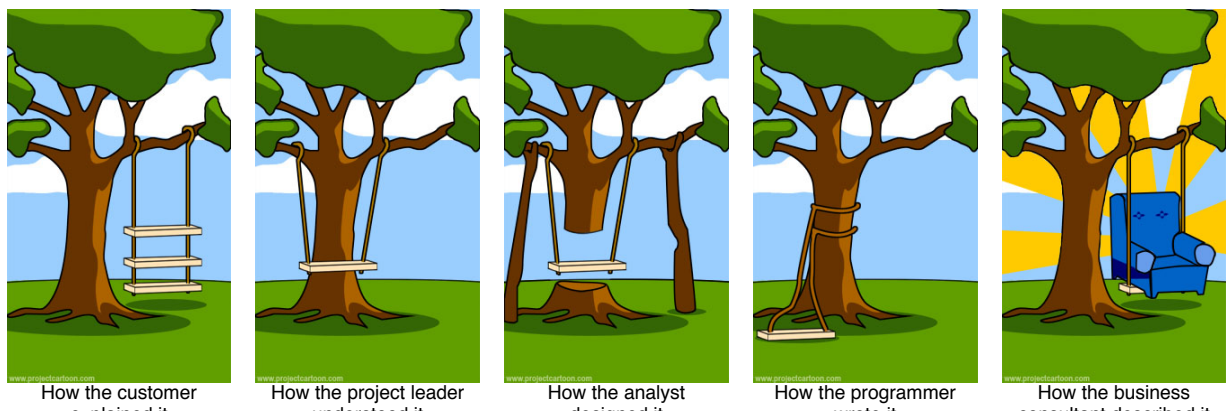
Objectives for this class meeting

- Complete requirements analysis for our game
- Articulate expectations for design specification
  - scene designs, storyboards
  - graphics vs widgets
  - atomic vs container widgets
- Articulate expectation for implementation
  - in-class exercise for class meeting 05 (Thurs, Jan 17)

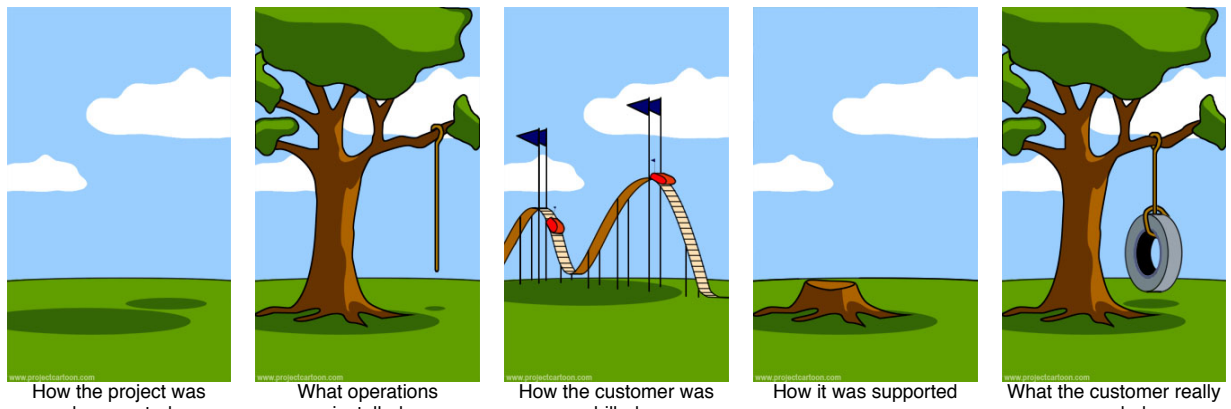
YORK UNIVERSITÉ UNIVERSITY

How Projects Really Work (version 1.0)

Create your own cartoon at www.projectcartoon.com



How the customer explained it | How the project leader understood it | How the analyst designed it | How the programmer wrote it | How the business consultant described it

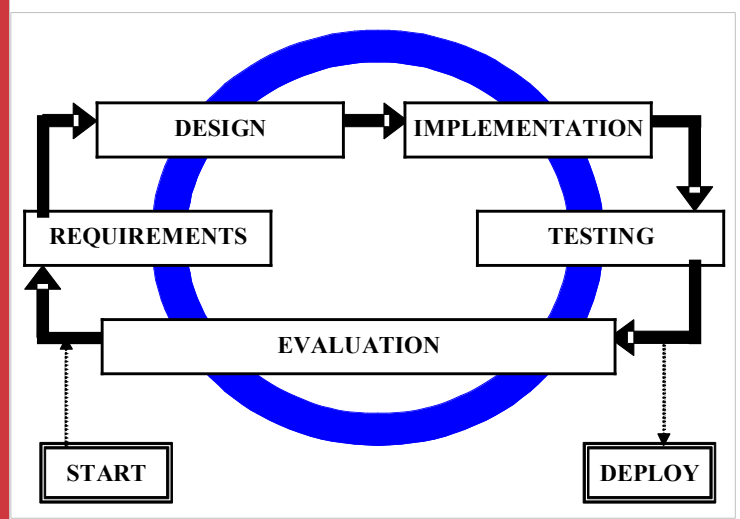


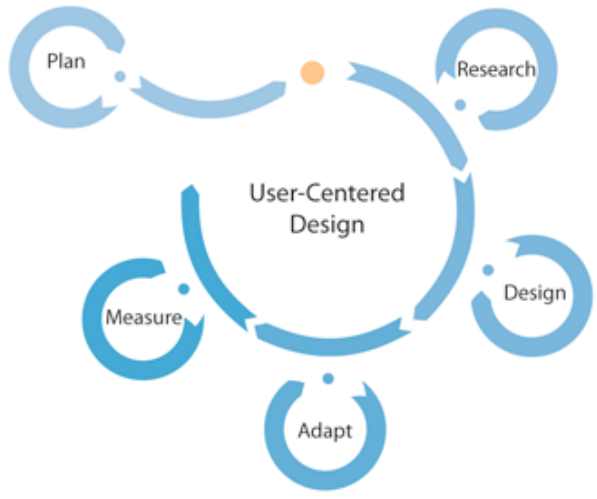
How the project was documented | What operations installed | How the customer was billed | How it was supported | What the customer really needed

Class Exercise

- Create a document (hardcopy or electronic) called “Requirements Analysis”
- Each student will maintain his or her own version of this document.
- You can expect this document to be refined and further developed during the term.
- You may be asked to submit this document at various points during the term.

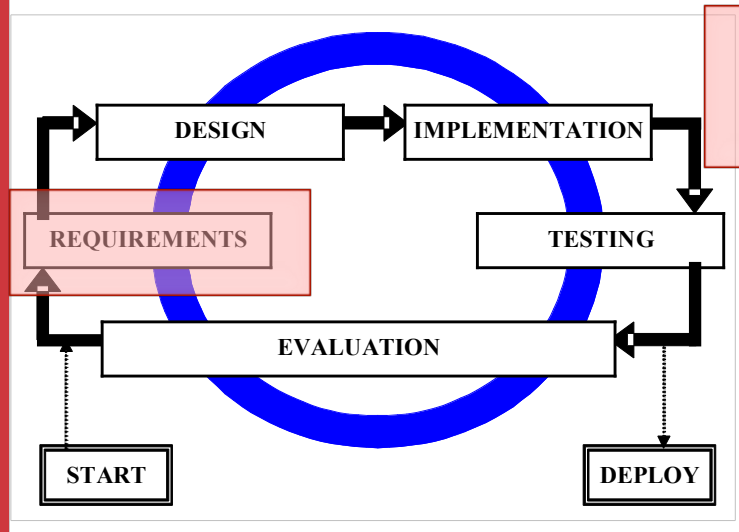
Overview

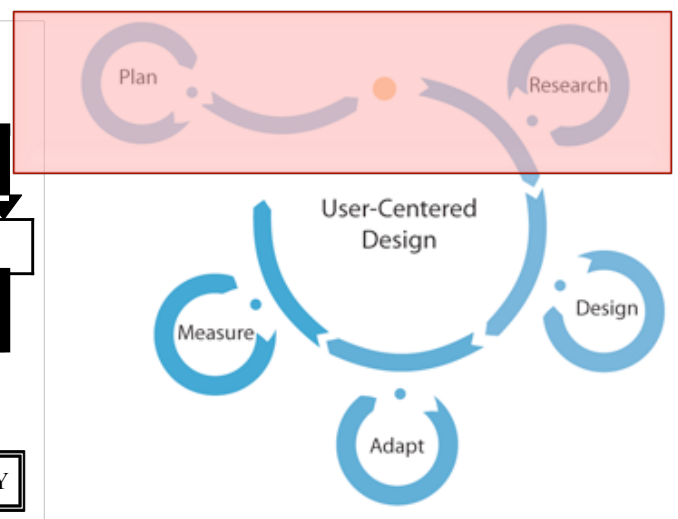




YORK UNIVERSITÉ UNIVERSITY

Overview





YORK UNIVERSITÉ UNIVERSITY

What is Requirements Analysis (RA)?

- means determining the **needs** for a new software system
  - in our case, the system will be an **sw application**
  - needs to take into account the (possibly conflicting) requirements of **various stakeholders**
  - also applies when altering an existing application
- In User Centered Design (UCD), RA is broken down into:
  - Plan: determine all activities that will be needed and the necessary resources
  - Research: understand the users' goals and tasks and the market needs

YORK UNIVERSITÉ UNIVERSITY

Stakeholders

- users

But also, depending on the domain, …

- those who will install/operate/support the app
- those who will regulate the app
- those who need to integrate with the system

YORK UNIVERSITÉ UNIVERSITY

Stakeholders

Since this is a class project,

1. we ourselves (and our friends) will be serving as the **user population**
2. learning is important to this process, so there will be **learner stakeholder requirements**
   - such stakeholders are not normally part of the development process
3. no business stakeholders
   - we don't intend to create a business model; selling isn't important
   - somewhat atypical/unusual for business stakeholders to be are absent from the process

9

YORK UNIVERSITÉ UNIVERSITY

Success Criteria

aka *how do you know this phase has been completed correctly*?

Are the requirements:

- documented
  - available in a format for others to learn about?
- actionable
  - can this requirement be addressed through taking steps during design?
- measurable, testable
  - will it be possible to assess the degree to which this requirement has been met?
- traceable
  - is it possible to determine when in the iterative cycle this requirement arose? what was the reason for its inclusion?
- relevant
  - is this requirement related to the identified design or business needs or opportunities?
- sufficiently detailed
  - can system design take place on the basis of these requirments?

10

YORK UNIVERSITÉ UNIVERSITY

Learner Stakeholder Requirements

- “Standalone” application
  - not networked with other players and/or other “live” feeds
- Implementation language: Java Standard Edition (Java SE),
  - as opposed to Java ME, Java EE, JavaFX
- Desktop/Laptop application
  - not intended for handheld devices
- Input devices:
  - possible/allowable: keyboard, mouse
  - disallowed: accelerometer, other sensors
- Game structure:
  - must be single player, not multiplayer
  - if game is adversarial, then adversary (computer) shouldn’t require “intelligent” behaviour (strategic behaviours)

YORK UNIVERSITÉ UNIVERSITY

Other Stakeholder Requirements

- does the system need to be compatible and/or integrate with other systems?
- to what extent, if any, will the system need to be extended and/or modified at a later time?
- is security important? does the system need to withstand hostile acts?
- how robust does the system need to be? does it need to keep running even if a component fails or under low memory conditions?
- who is going to maintain the system? how easy should maintenance be?
- is any part of the system going to be re-used in another system?

YORK UNIVERSITÉ UNIVERSITY

User Requirements

- do we understand the user's needs and goals?
  - do we understand the user's "task"
- do we understand the user's context (the context for this task)?
  - other aspects/factors that need to be taken into account

User Requirements

Some terminology and framing:

- **users** have **needs** and **goals**
- **systems** are developed in order to allow users to **meet their goals** and to **address their needs**

For our game, what do we understand the user's needs and/or goals to be?

For our game, in which context(s) will the game be played?

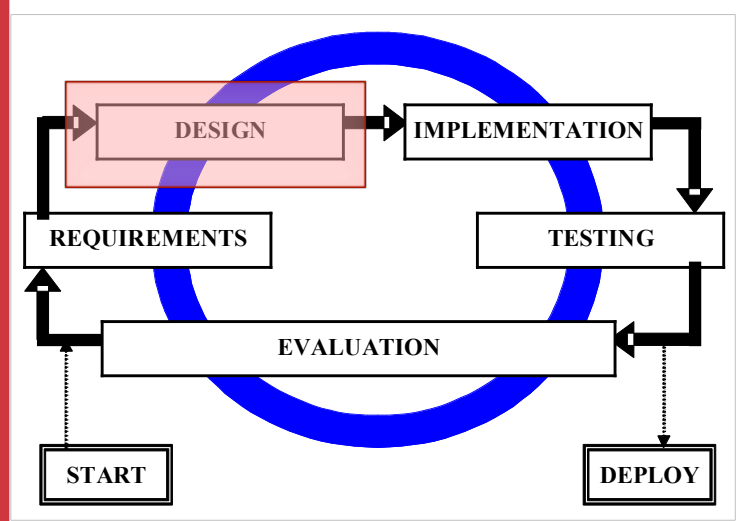
User needs and goals

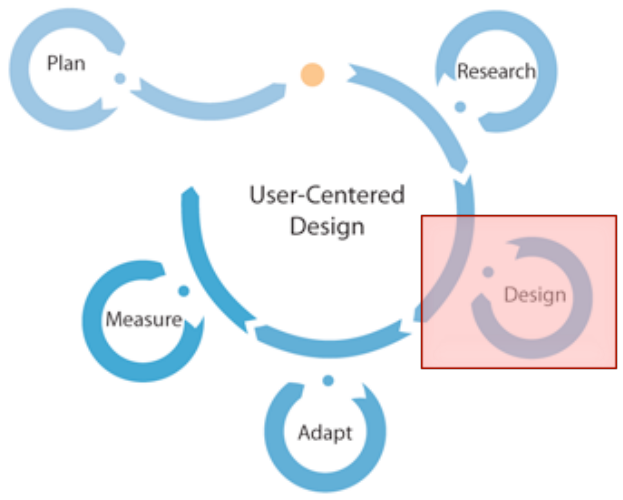
- _____
- _____
- …

User context

- _____
- _____
- …

Overview





YORK UNIVERSITÉ UNIVERSITY

What is Design Specification?

Specify all aspects of the system:

- the architecture
- the components (classes, modules)
- the interfaces
- other characteristics

Use a **modeling language** for the specification, for instance UML

YORK UNIVERSITÉ UNIVERSITY

What is Design Specification?

Success Criteria

- aka *how do you know this phase has been completed correctly?*
- does the **specification** meet the **requirements**?
- have all aspects of the system been **defined** sufficiently so that they can be implemented?
- in the case of UML class diagrams:
  - derive the class APIs
  - are the APIs detailed enough so that the classes can be implemented?
- the outcome of this phase is a **set of specifications**

YORK UNIVERSITÉ UNIVERSITY

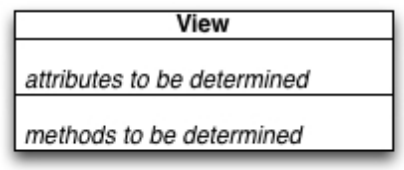
What is Design Specification?

Software design concepts that can be used:

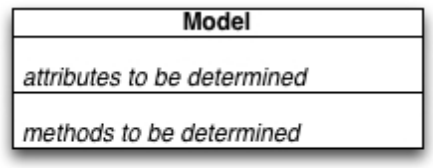
- abstraction, layered abstraction
- modularity, encapsulation, separation of concerns

YORK UNIVERSITÉ UNIVERSITY

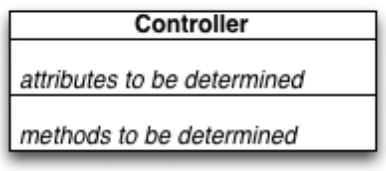
Architecture



this module implements:
- what does the user see/hear
- what actions can the user perform?
aka the user interface



this module implements:
- what is the state of the system?
such as
- what is the current score?
- whose move is next?
- how close to the end?
- what actions are allowed right now?



this module implements:
- the logic of the game
- given a user action, what is the impact on the state of the system?
- given other events (clock ticks, countdown timers), what are the impacts on the system's state

Course Evaluation

- **During Term 76%**
  - exercises **24%**
    - **lab exercise #1 : 2.5%**
  - written tests (x3 @ 12% each) **36%**
  - labtests (x3 @ 12% each) **36%**
- **Final Exam Period 24%**
  - written tests **12%**
  - labtests **12%**

For next class…

- have RA document prepared
- Be ready to do design specification for the view:
  - Read the following
    - Lesson: Using Swing Components
      - read “1 level deep”, except for the “How to…” subsection (read 2 levels deep)
      - http://docs.oracle.com/javase/tutorial/uiswing/index.html
    - Lesson: Overview of the Java 2D API Concepts (1 level deep)
      - read 1 level deep
      - http://docs.oracle.com/javase/tutorial/uiswing/index.html

YORK UNIVERSITÉ UNIVERSITY