

# CSCI397: Tools for the Software Life Cycle and Beyond

Or: What I Think You Should Know  
Before You Graduate

Jan 10, 2022

Sprenkle - CSCI397

1

1

## Software Tools

- What are they?
  - What is the goal of software tools?
- Why do we use them?
- Why do we develop them?

Jan 10, 2022

Sprenkle - CSCI397

2

2

## Tool Goal: Productivity

- Many available tools
  - UNIX & UNIX-like systems (e.g., Linux)
  - Open-source (Gnu, Apache, Eclipse)
  - Proprietary
  - Variety of purposes
- Know what (mostly, free) tools are available, what they do, how to use them

Jan 10, 2022

Sprenkle - CSCI397

3

3

## Tool Goal: Automation

- Often have to do a task over and over again
  - Time-intensive to do by hand
  - Shortcuts aren't enough
- What we want
  - Tools to make tasks easier
  - Scripts to be able to repeat the tasks easier

Jan 10, 2022

Sprenkle - CSCI397

4

4

## Tool Goal: Higher Quality Software

- Related to automation but different
- Efficiently debug, test, and deploy code

Jan 10, 2022

Sprenkle - CSCI397

5

5

## Common Interfaces to Tools

- Command Line Interface (CLI)
- Graphical User Interface (GUI)

What are the benefits and limitations  
of each type of tool?

Jan 10, 2022

Sprenkle - CSCI397

6

6

# Command Line Tools

## Benefits

- Flexible--lots of options
- After run once, can run again in same terminal using up arrow key or using !command
- Tab-completion
- Automation: Can be put into bash scripts and repeated

## Limitations

- Requires knowing name of command
- Requires knowing syntax of command, options
  - Easy to screw up!
- Slower learning curve

Jan 10, 2022

Sprenkle - CSCI397

7

7

# GUIs

## Benefits

- Require less knowledge of syntax
- Generally: faster learning curve

## Limitations

- Can require many clicks to do even simple operations
- May require a lot of set up/configuration
- Harder to automate, repeat tasks

Jan 10, 2022

Sprenkle - CSCI397

8

8

## Course Content

- Unix tools
- Bash scripting
- Software development tools

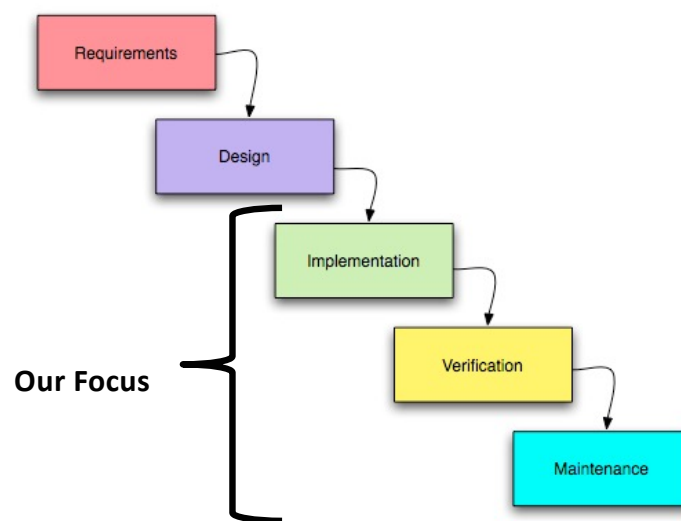
Jan 10, 2022

Sprenkle - CSCI397

9

9

## Software Life Cycle: Waterfall Model



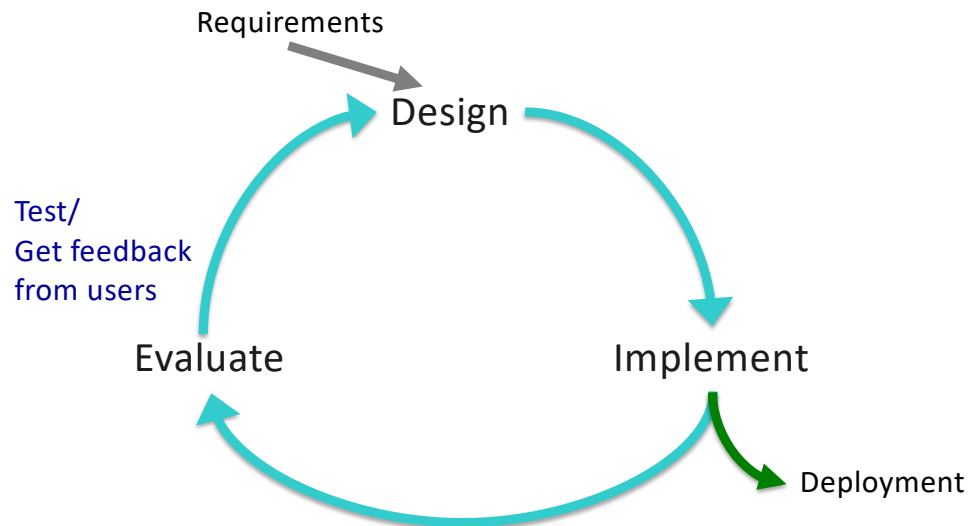
Jan 10, 2022

Sprenkle - CSCI397

10

10

## Iterative Design



Jan 10, 2022

Sprenkle - CSCI397

Often using Agile Development

11

## Why This Class?

- My graduate school experience
- Feedback from alumni and friends in industry

Jan 10, 2022

Sprenkle - CSCI397

12

12

## Course Objectives

- At the end of this course, you will be able to
  - Use a variety of Unix tools
  - Apply a variety of tools to automate many tasks
  - Describe the use of state-of-the-art software tools for developing and maintaining large software systems, based on hands-on experience
  - Discuss when best to use different tools, the limitations of the tools, and what they have to offer
  - Discuss the challenges and strategies in building software tools
  - Communicate technical content in both oral and written forms
  - *Learn a new tool on your own*

Jan 10, 2022

Sprenkle - CSCI397

13

13

## Goals Not on the Syllabus

- Improve your productivity
- Unix confidence/proficiency
  - To intermediate user
- Improve your confidence
  - Less intimidated by lingo; installing, learning new tools
- Better at collaborating with others
- Resume builder!
- Tech culture

Jan 10, 2022

Sprenkle - CSCI397

14

14

## Non-Goals

- To make you a Unix System Administrator
- To make you a senior software engineer
  - Or even an intermediate software engineer
  - Tools, development styles, approaches are changing and company-specific
- To give you real industry experience
  - We can only get a fake approximation

Jan 10, 2022

Sprenkle - CSCI397

15

15

## Expectations

- Material is most relevant in context
  - Need to make it relevant to you
  - What would you like to do—now or in the future?
  - What tools interest you?
- Actively explore tools
  - Try out everything we do
  - Make mistakes and learn from them

Jan 10, 2022

Sprenkle - CSCI397

16

16



## Class Details

- Course web page
  - <http://cs.wlu.edu/~sprenkle/cs397>
  - Check schedule frequently for updates
- Classes are in-person
- Classes will be recorded on Zoom and posted in Yuja on Canvas
- If you're not feeling well, you can attend the class on Zoom **BUT** that should not be a frequent occurrence

Jan 10, 2022

Sprenkle - CSCI397

17

17

## Grading

- (60%) Individual programming, reading, analysis, and homework assignments
- (25%) Final Tool Project (demo, documentation, etc.)
- (15%) Professionalism, participation, and attendance
  - Includes installing software before class, in-class assignments where you get an acceptable (check) or not-acceptable(check -), interacting with/asking questions of guest speakers

Jan 10, 2022

Sprenkle - CSCI397

18

18

## Your #1 Tool: A Search Engine

- You have a problem
- Need to find resources
- What helps make your search successful depends on what you're searching for
  - Searching for tools is different than searching for help on solving programming/software problems

Jan 10, 2022

Sprenkle - CSCI397

19

19

## Searching: Resolving Bugs

- Quote error messages, removing your specific code/classes
  - This is why I request students send me text (copy/paste the error message) instead of screenshots
- If you don't quite get what you want, refine query based on results

Jan 10, 2022

Sprenkle - CSCI397

20

20

## Searching: Tools

- Goal: find a tool that solves a problem
- Problem: tech lingo and buzzwords make it hard to decipher what the tool is actually about

Jan 10, 2022

Sprenkle - CSCI397

21

21

## Buzzword Bingo



- Applied to our class: read a tool's splash page or hear a talk, and note which of the buzzwords they mention

<https://dilbert.com/strip/1994-02-22>

Jan 10, 2022

Sprenkle - CSCI397

22

22

## Beyond the Buzzwords: Lingo

- All the terms, acronyms about software development can be intimidating and overwhelming
- It's hard to cut through it all and understand what it means

Jan 10, 2022

Sprenkle - CSCI397

23

23

## In-Class Assignment: Lingo Bingo

- (It's not actually Bingo)
- We are going to crowd source this living document to help us understand some common software industry terms
- Go to <https://wlu.box.com/v/CSCI397-LingoBingo>
  - Pick some terms to look up and define
- Want definitions for *all* terms at the end of the class period
  - Include common acronyms/tools, images, ... as appropriate
- Check the box when there is at least one definition for the term
- Can have multiple definitions for terms from different students
- Can add terms

Jan 10, 2022

Sprenkle - CSCI397

24

24

## Lingo Bingo Analysis

- How successful were your searches?
  - How did you improve your searches with subsequent searches?
- What was easy to figure out? What was more difficult?
- What other terms should be added to this list?

Jan 10, 2022

Sprenkle - CSCI397

25

25

## Looking Ahead

- Read (or watch) “How to Be a Wizard”
  - Geared toward a software developer, but there is useful stuff in here for students in a 300-level software tools course
  - Write a response on Canvas under the Discussions page by Tuesday at 11:59 p.m.
  - We’ll discuss on Wednesday
- Also Wednesday: Unix basics!

Jan 10, 2022

Sprenkle - CSCI397

26

26