CSE 403

Software Engineering

Software development life cycle

Today

- Project proposals
  - Elevator pitches

- Software Development Life Cycle (SDLC)
  - What is it and why is it needed
  - Recurring themes and stages
  - Popular SDLC models and their tradeoffs

Project Proposal Pitches

Assignment 1: Project proposals

Elevator pitch: a brief, persuasive speech to spark interest in a product, project or idea. An elevator pitch is short, about the time you spend in an elevator, hence the name.

https://asana.com/resources/elevator-pitch-examples

You have 2.5 min for your project pitch to the class!

This is an example of how it could flow.

Proposal pitch template

Introduce yourself	
Present the problem	
Present your solution (This is your lucky day!)	
Share your value proposition	
Add a call to action	

Independently create two pitches with your proposal partner, then iterate and consolidate.

Mock product press release

Another option to creating a pitch:
Write a mock product press release!

Includes

- A catchy headline
- Problem trying to solve
- Value proposition
- How differs from competitors
- Release timing and teaser of future beyond release
- Quotes from well known users showing their delight

Great way to show the vision and get buy in

See: https://jdmeier.com/how-to-create-innovative-disruption-with-mock-press-releases/
https://www.linkedin.com/pulse/working-backwards-press-release-template-example-ian-mcallister

Time to Walk: An inspiring audio walking experience comes to Apple Fitness+

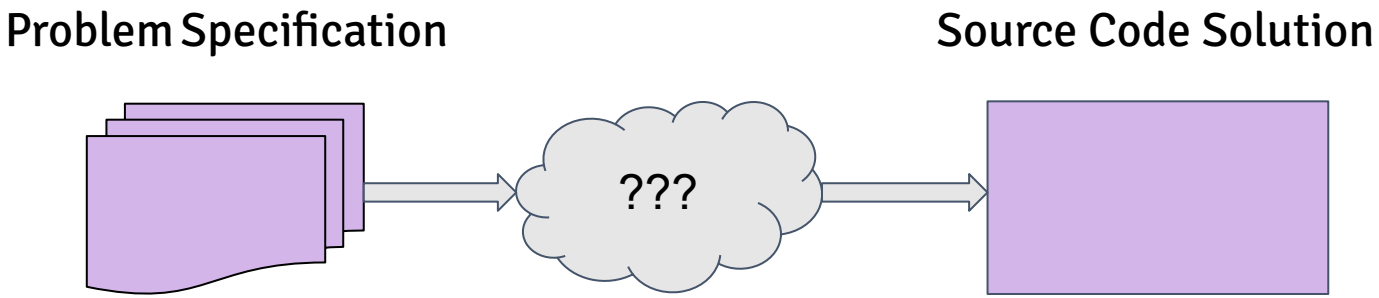
Episodes feature personal stories, photos, and music from influential people to inspire Apple Watch users to walk more

Cupertino, California — Apple today unveiled Time to Walk, an inspiring new audio walking experience on Apple Watch for Fitness+ subscribers, created to encourage users to walk more often and reap the benefits from one of the healthiest activities. Each original Time to Walk episode invites users to immerse themselves in a walk alongside influential and interesting people as they share thoughtful and meaningful stories, photos, and music. Time to Walk can be enjoyed anytime and anywhere with Apple Watch and AirPods or other Bluetooth headphones.

"Walking is the most popular physical activity in the world, and one of the healthiest things we can do for our bodies. A walk can often be more than just exercise: It can help clear the mind, solve a problem, or welcome a new perspective," said Jay Blahnik, Apple's senior director of Fitness Technologies. "Even throughout this challenging period of time, one activity that has remained available to many is walking. With Time to Walk, we're bringing weekly original content to Apple Watch in Fitness+ that includes some of the most diverse, fascinating, and celebrated guests offering inspiration and entertainment to help our users keep moving through the power of walking."

SDLC: Software Development Life Cycle

SDLC: Here's the challenge



One solution: Ad-hoc code and fix

Specification
(maybe)

Deliver
(maybe)

SDLC: Code and fix

Pros:

- Little or no overhead - just dive in and develop, and see progress quickly
- Applicable *sometimes* for small projects, short-lived prototypes, and/or small teams

Cons:

- **Do you see any downsides to this approach?**

SDLC: Code and fix

Pros:

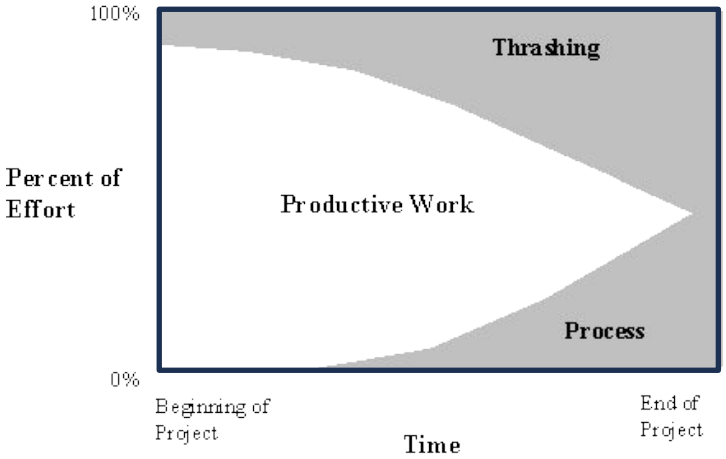
- Little or no overhead - just dive in and develop, and see progress quickly
- Applicable *sometimes* for small projects, short-lived prototypes, and/or small teams

Cons:

- **No way to assess progress, quality or risks**
- **Challenging to manage multiple developers – how synchronize your work**
- Harder to accommodate changes without a major design overhaul
- Unclear delivery of features (scope), timing, and support

Let’s look at data

Projects with little attention on SDLC process



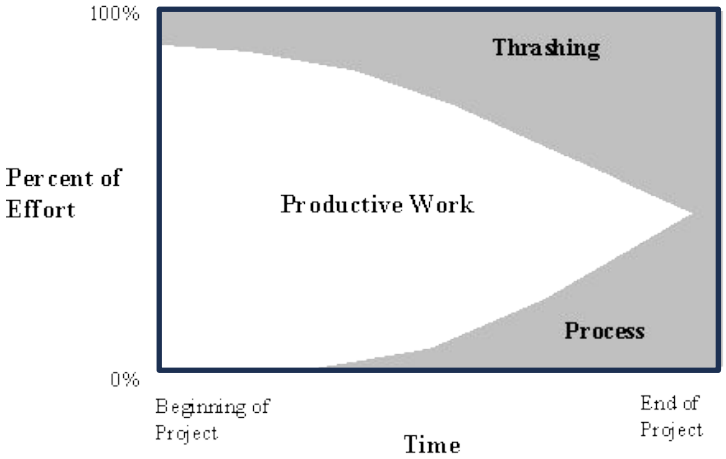
Thrashing:

Doing a lot of work but not making progress towards the actual goal!

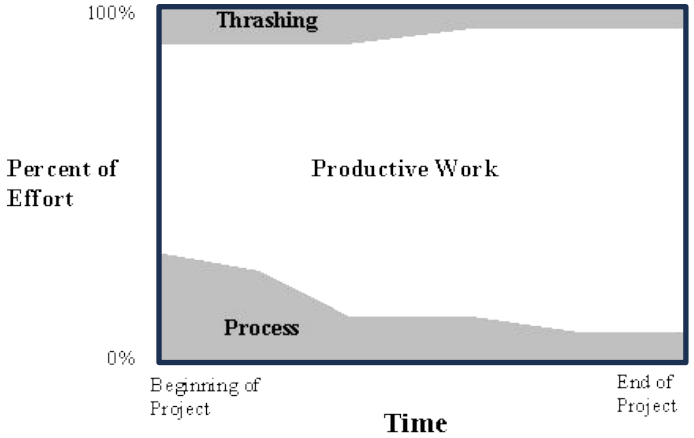
The Power of Process | Steve McConnell

Let’s look at data

Projects with little attention on SDLC process



Projects with early attention to SDLC process



The Power of Process | Steve McConnell

The software development life cycle (SDLC)

SDLC: produce software through a series of stages

- From conception to end-of-life.
- Can take months or years to complete.

Goals of each stage

- Define a clear set of steps to perform.
- Produce a tangible item.
- Allow for review of work.
- Specify actions to perform in the next stage.

Life-cycle stages

Virtually all SDLC models have the following stages

- Requirements
- Design
- Implementation
- Testing
- Maintenance

Key questions:

- How to combine the stages and in what order?
- How does this differ for *traditional vs. agile* models?

Major SDLC models

Traditional models

- Waterfall model
- Prototyping
- Spiral model
- ...

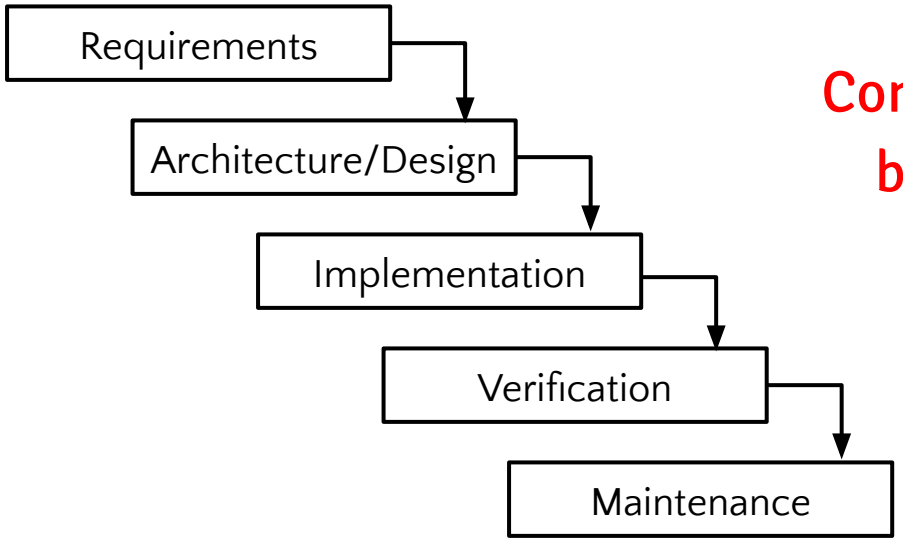
Agile models

- *XP (Extreme Programming)*
- *Scrum*
- ...

All models have the **same goals:**
manage risks and
produce **high quality software**.

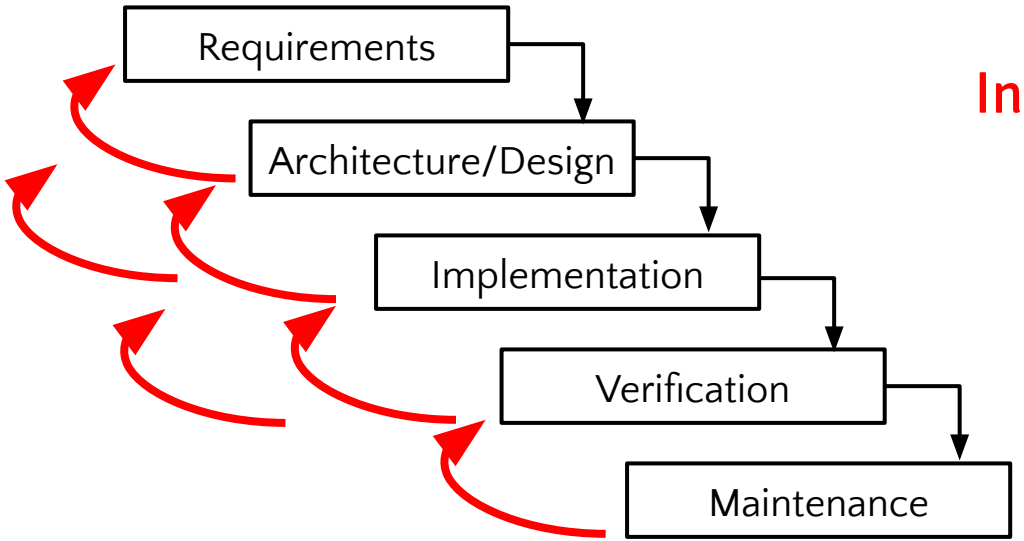
Traditional SDLC models

SDLC: Waterfall model



Conceptually very clean,
but what is missing?

SDLC: Waterfall model



In what contexts does this model work?

Honeywell's Flight Management System Selected By Airbus

Honeywell's solution will address the avionics needs of the Airbus A320, A330 and A350 aircraft fleet

Ahjay Rai
May 19, 2022

FDA U.S. FOOD & DRUG ADMINISTRATION

Home / Medical Devices / Device Advice: Comprehensive Regulatory Assistance / Overview of Device Regulation

Overview of Device Regulation

Share | Tweet | Linkedin | Email | Print

Overview of Device Regulation

A History of Medical Device

Introduction

FDA's Center for Devices and Radiological Health (CDRH) is responsible for regulating firms who manufacture, repackage, relabel, and/or import medical devices sold in the United States. In addition, CDRH regulates radiation-emitting products (medical and non-medical) such as lasers, x-ray equipment, microwave ovens and color televisions.

Likely parts of their SDLC is waterfall-like due to the upfront and regulated requirements

SDLC: Waterfall model pros and cons

Pros

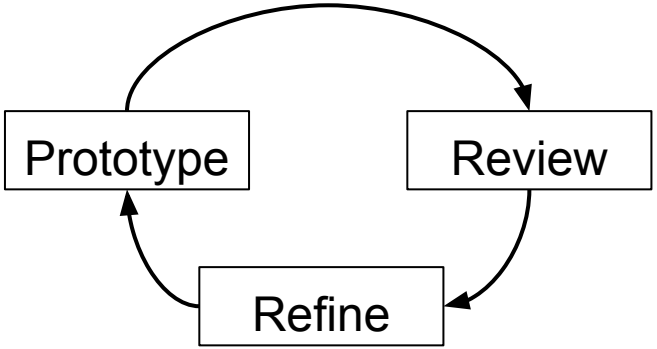
- Easy-to-follow, sequential model.
- Reviews ensure readiness to advance.
- Works well for well-defined projects (requirements are clear).

Cons

- Hard to do all the planning upfront.
- Final product may not match the client’s needs.
- Step reviews require significant effort.

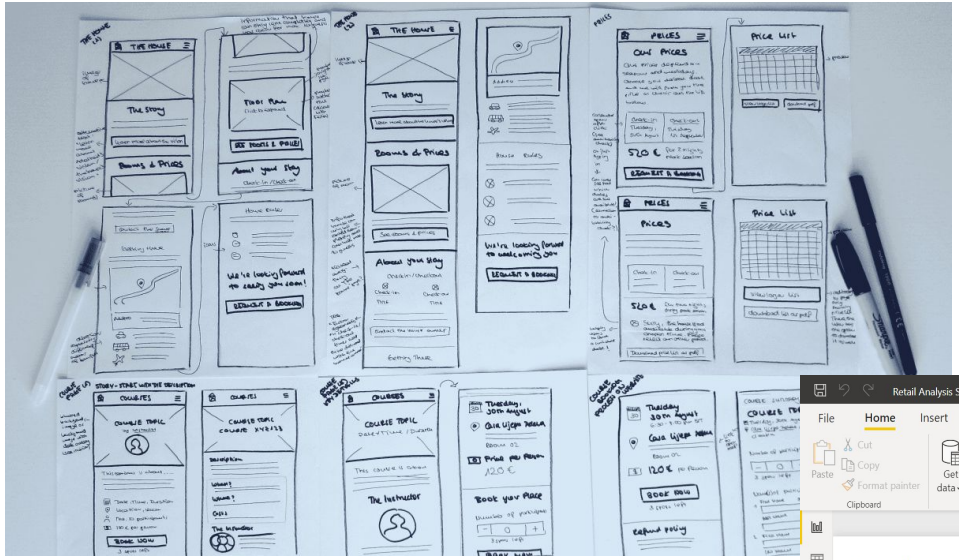
SDLC: Prototyping

- Bottom-up approach.
- Problem domain or requirements not well defined or understood.
- Create small implementations of requirements that are least understood.
- Requirements are “explored” before the product is fully developed.
- Developers gain experience when developing the “real” product.

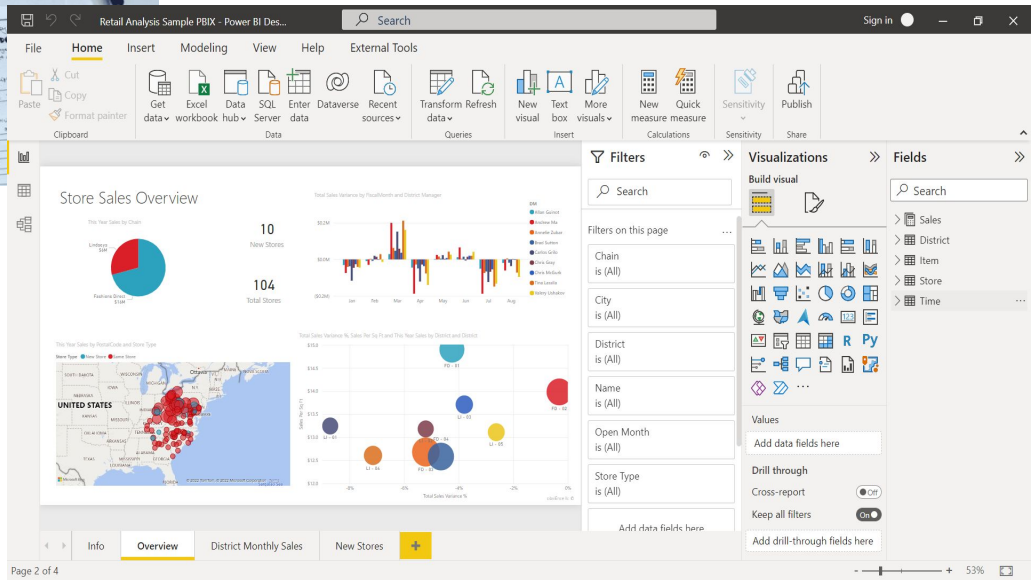


In what contexts does this model work?

UI prototyping is popular



https://internetdevels.com/blog/what-is-website-prototype-how-build-website-prototype



https://learn.microsoft.com/en-us/power-bi/fundamentals/desktop-what-is-desktop

SDLC: Prototyping pros and cons

Pros

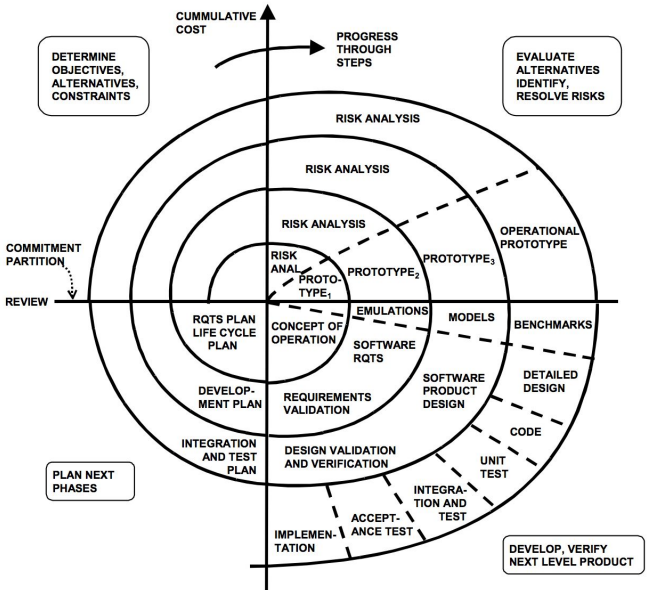
- Client involvement and early feedback.
- Improves requirements and specifications.
- Reduces risk of developing the “wrong” product.

Cons

- Time/cost for developing a prototype may be high.
- Focus may be too narrow (no thinking outside the box).

SDLC: Spiral model

- Incremental/iterative model (combines the waterfall model and prototyping).
- Iterations called spirals.
- Activity centered:
  - Planning
  - Risk analysis
  - Engineering
  - Evaluation
- Phased reduction of risks (address high risks early).

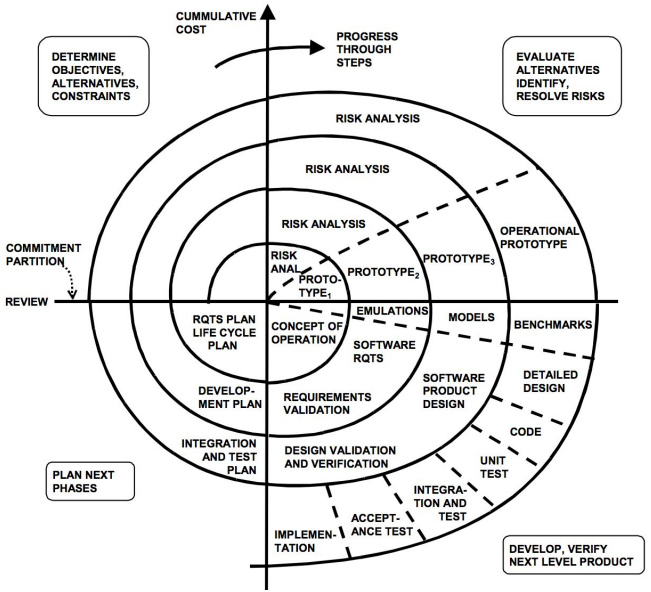


Boehm, *Spiral Development: Experience, Principles,and Refinements*

SDLC: Spiral model importance

A precursor to agile models:

Software development is based on iteration, using “risk reduction” as a criterion to prioritize activities at each iteration



Boehm, *Spiral Development: Experience, Principles,and Refinements*

SDLC: Spiral model pros and cons

Pros

- Early indication of unforeseen problems.
- Allows for changes.
- The risk reduces as costs increase.

Cons

- More complex to implement and run.
- Requires proper risk assessment.
- Requires a lot of planning and experienced management.

Agile SDLC models

SDLC: Agile models

Agile Manifesto (http://agilemanifesto.org/):

- ***Individuals and interactions*** over processes and tools
- ***Working software*** over comprehensive documentation
- ***Customer collaboration*** over contract negotiation
- ***Responding to change*** over following a plan.

SDLC: Agile models

Basics

- Maintain simplicity.
- Team members choose their own methods, tools etc.
- Continuous customer involvement.
- Expect requirements to change,
  focus on incremental delivery.
- Improve communication.

SDLC: Extreme Programming (XP)

Extreme Programming (XP)

- New versions may be built several times per day with products delivered to customers weekly.
- All tests must be run and pass for every build (may be combined with test-driven development).
- Adaptation and re-prioritization of requirements.
- Pair programming and continuous code review.

SDLC: Agile models pros and cons

Pros

- Flexibility (changes are expected).
- Focus on quality (continuous testing).
- Focus on communication.

Cons

- Requires experienced management and highly skilled developers.
- Prioritizing requirements can be difficult when there are many stakeholders.
- Best for small to medium (sub) projects.

What is the best SDLC model?

Why are there so many SDLC models?

Choices are good ☺️!

- **The choice depends on the project context and requirements**
- All models have the same goals:
  manage risks and produce high quality software
- **All models involve the same general activities and stages** (e.g., specification, design, implementation, and testing) and can be tailored
- Today's models focus on **customer feedback** and the ability to adapt to **changing requirements**

What model would you choose and why?

- A control system for anti-lock braking in a car.
- A hospital accounting system that replaces an existing one.
- An interactive system that allows airline passengers to quickly find replacement flights (for missed or bumped reservations) from airport terminals or a mobile app.
- A new web app for AI-based Q&A about recorded videos.