

LEAP Life - Computer Literacy – January 7, 2026

Day 3



Department of Electrical & Computer Engineering



Day 3 Agenda

- **9:00**
 - 10-minute conversation on DNS
 - Overview of modern day software development processes - focus on Kanban
- **10:00**
 - Group Project Team Formation & Project Choices (suggestions available)
 - GitHub Kanban Config Overview & PullRequest Set Up
- **11:00 - 4:00**
 - Group Project Support
- **4:00 - 5:00**
 - Group Project Demos
 - ~ 10 minutes per groups

Day 2 Hands-on Exercise 4

- Set up Continuous Deployment of your Lambda function code via GitHub Actions
- Validate that you can make changes, have the code deployed automatically, and confirm behavior by invoking your function via a web browser



4

Day 2 Hands-on Exercise 4 - What did you learn?

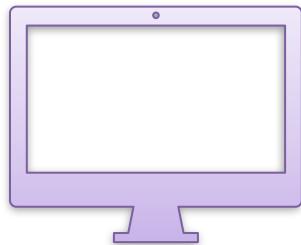


What is DNS?

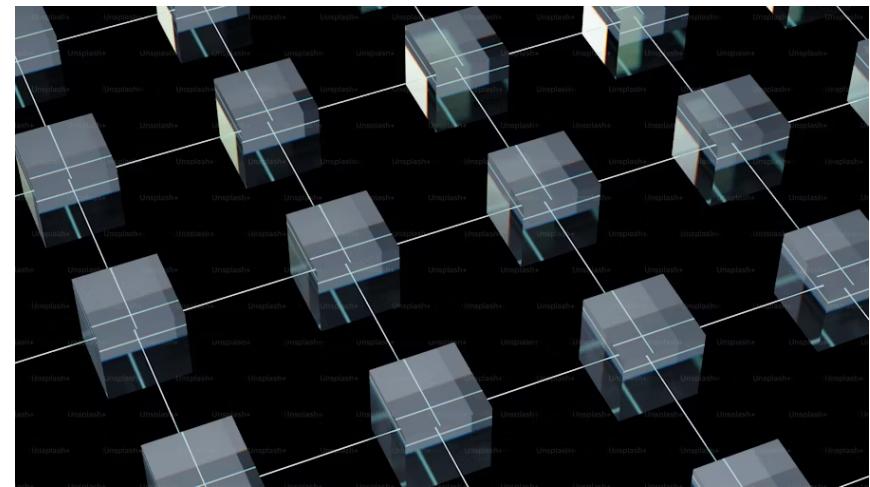


What is DNS?

www.wikipedia.com



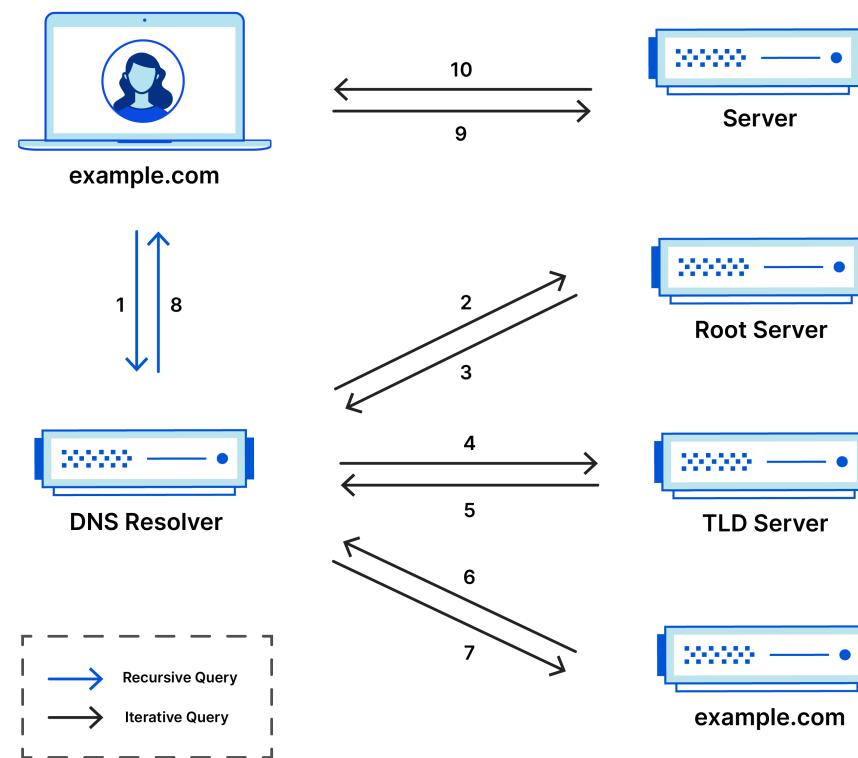
?



- DNS (Domain Name System) translates human-readable domain names (like wikipedia.org) into IP addresses (like 208.80.154.232)

What is DNS?

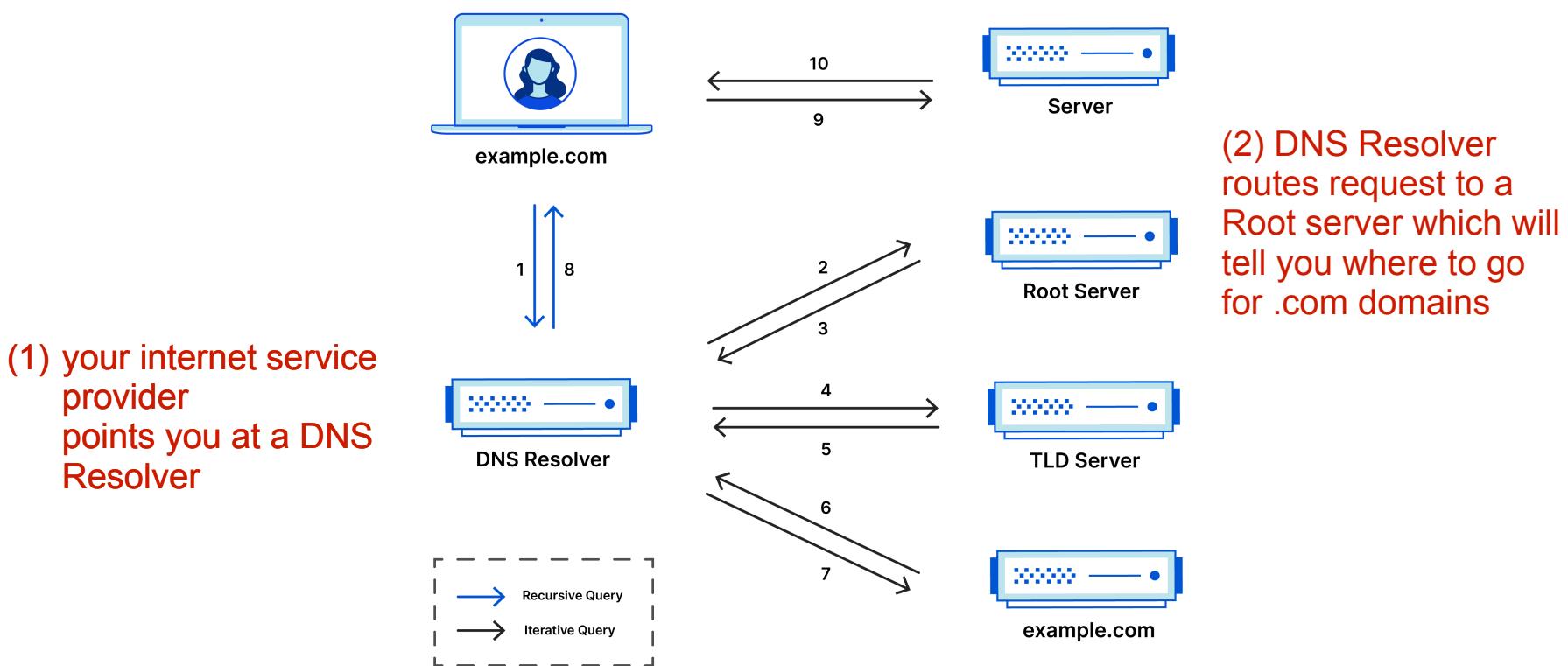
Complete DNS Lookup and Webpage Query



(1) your internet service provider points you at a DNS Resolver

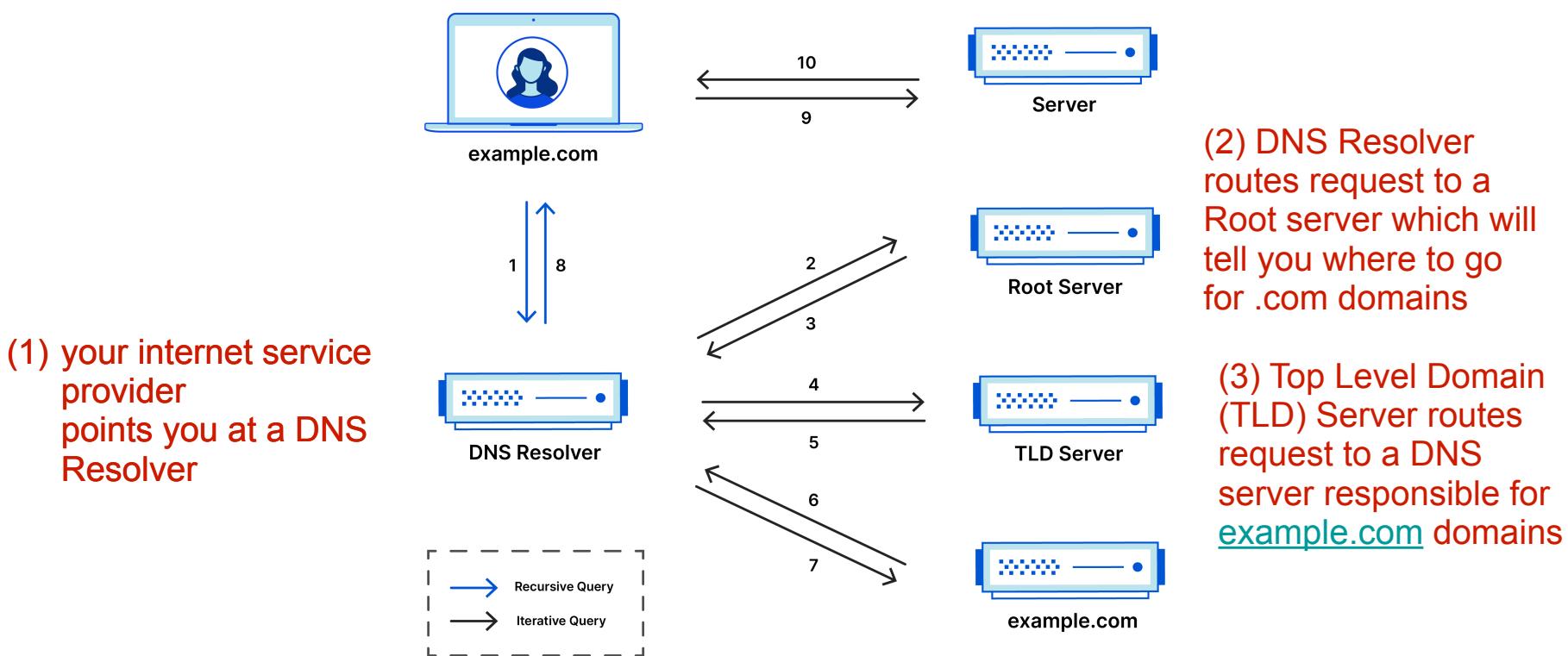
What is DNS?

Complete DNS Lookup and Webpage Query



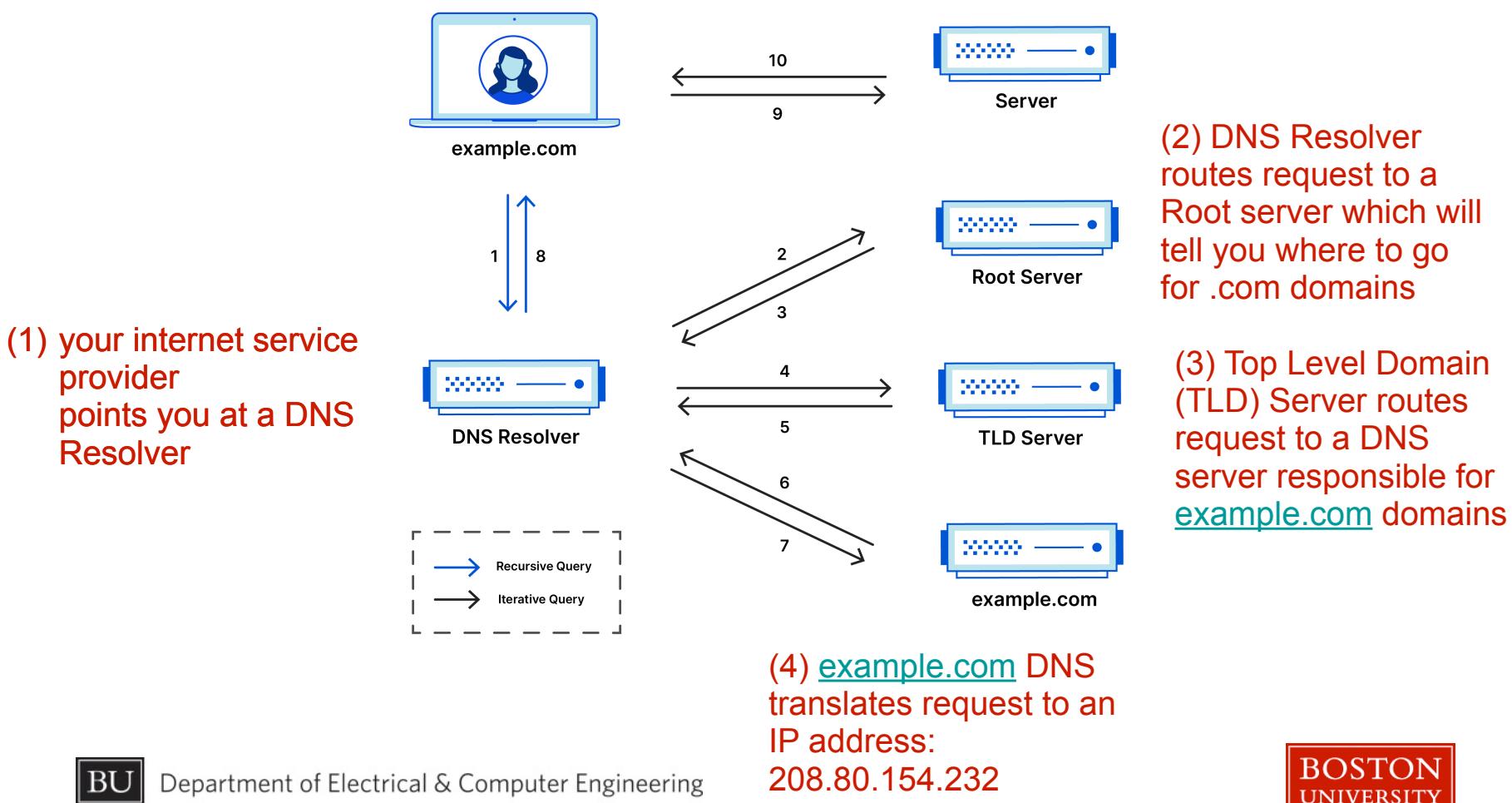
What is DNS?

Complete DNS Lookup and Webpage Query



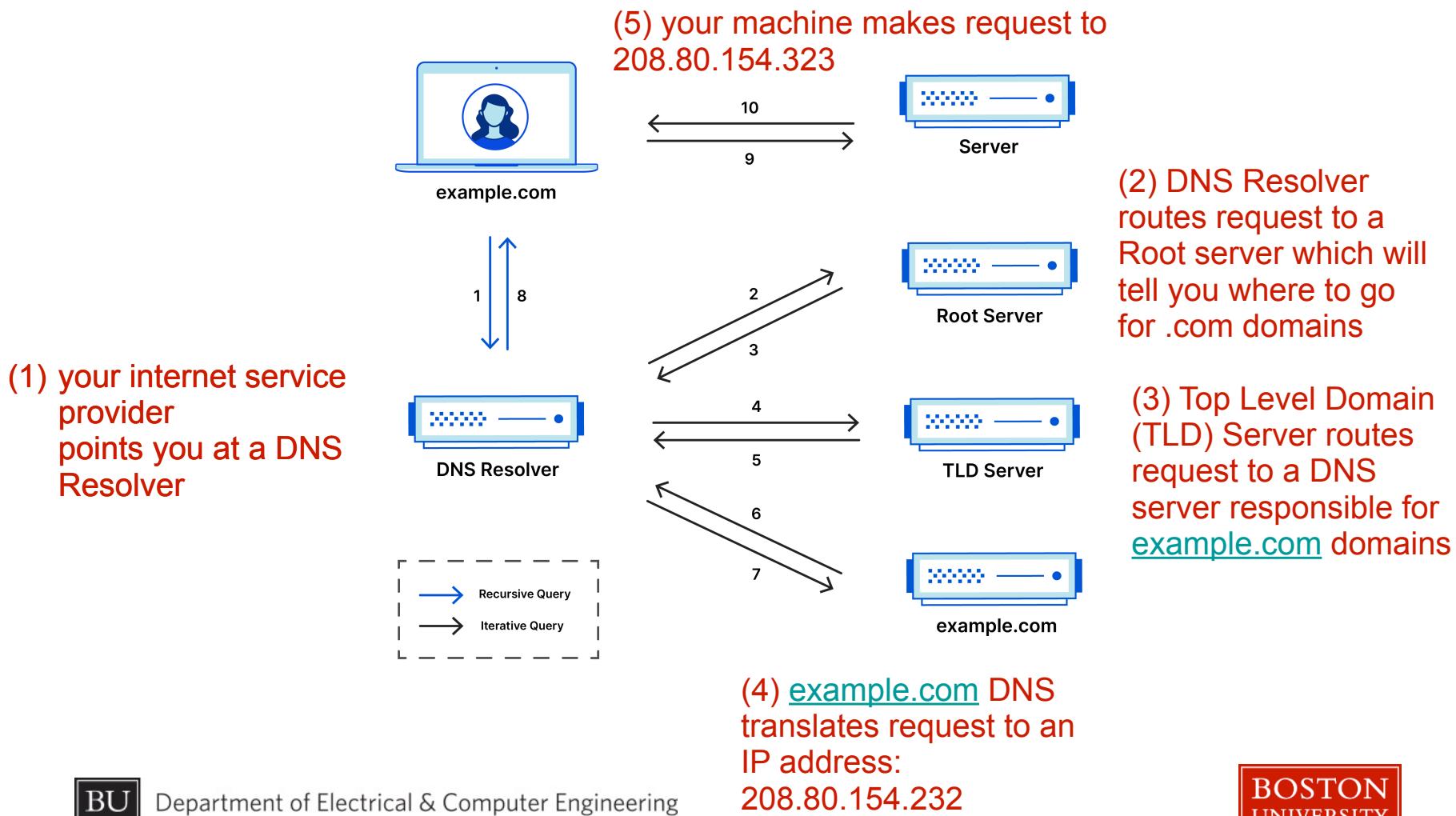
What is DNS?

Complete DNS Lookup and Webpage Query



What is DNS?

Complete DNS Lookup and Webpage Query



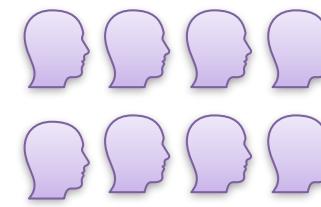
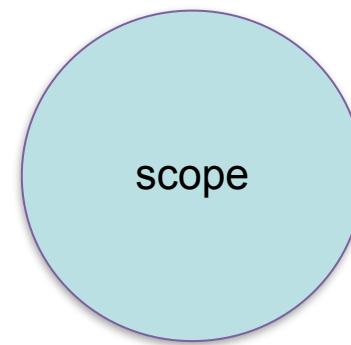
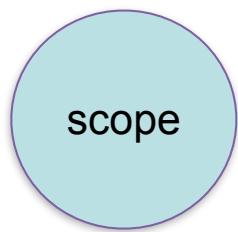
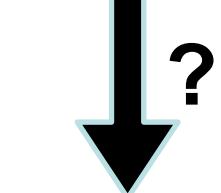
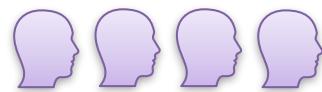
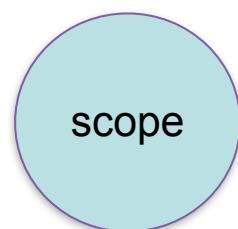
Iron Triangle

The iron triangle of project management

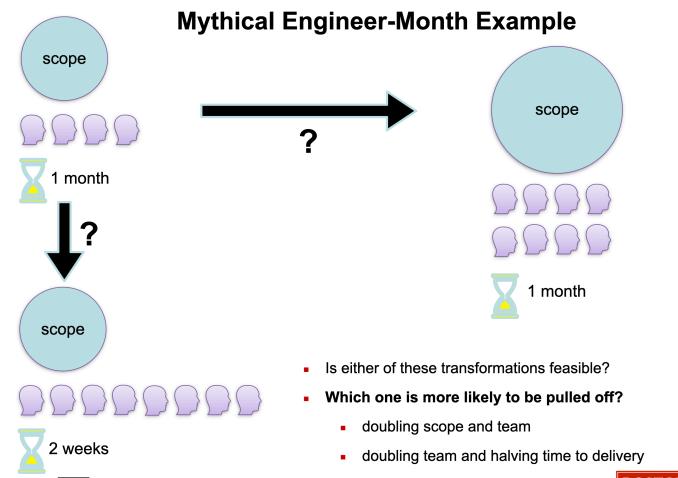


- No one wants to ship low quality software.
- We assume that the quality bar is very high and we are not willing to sacrifice on it.
- The other three dimensions play a game of tug of war:
 - Increasing Scope requires an increment of Time or Cost
 - Reducing Cost requires a reduction in Scope or increment of Time.
 - Reducing Time requires reducing Scope or an increment of Cost

Mythical Engineer-Month Example



- Is either of these transformations feasible?
- **Which one is more likely to be pulled off?**
 - doubling scope and team
 - doubling team and halving time to delivery



- Fred Brook's classic 1975 book *The Mythical Man-Month: Essays on Software Engineering*.
- The central idea is that adding more people to a late software project often makes it later, rather than speeding it up.
- Software engineering should not be thought of as linear work.
- **Communication overhead is exponential with the the number of people involved on a project.**

Roles in a Waterfall World

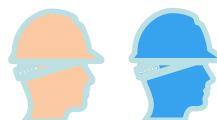
project planning



requirements & analysis



design



coding



testing



deployment



CEO



product manager



UX researcher



designer



developer



quality assurance



release engineer

Agile

Agile Development

is *adaptive* rather than predictive
is *people-oriented* rather than process-oriented



Martin Fowler



Department of Electrical & Computer Engineering

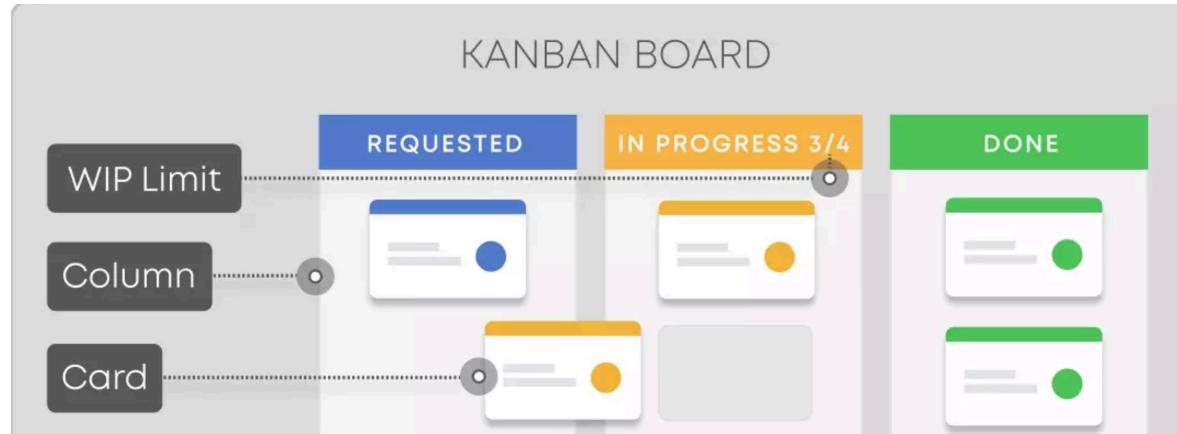


Agile

- There are multiple methodologies and processes that are considered **Agile**, we will dig into a few of them, but the main distinction from its predecessor methodologies:
 - In **Plan-driven** processes, e.g. Waterfall, success is measured according to how well development follows the detailed, predictive plan.
 - In **Agile** processes, careful plans are still made, but are thought to be in constant revision to reflect learnings and a quickly evolving ecosystem.
 - In **Agile** processes, success is measured based on value delivered by the software.



Kanban

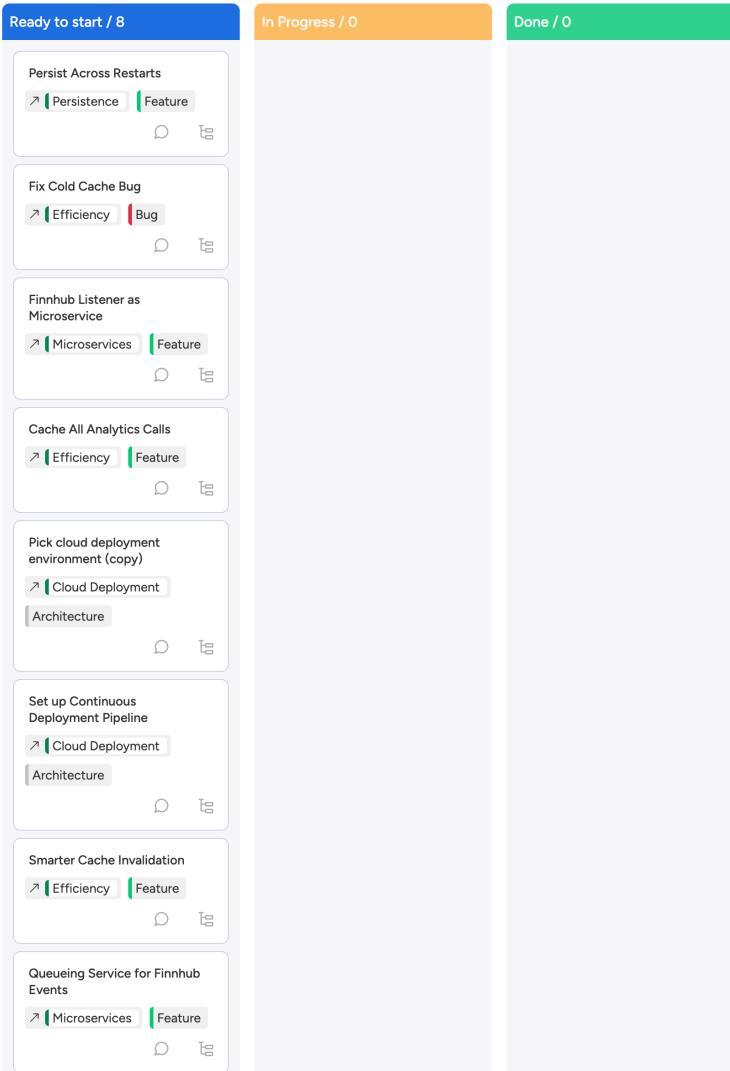


- Originated in manufacturing by Toyota in the 1940's
- Inspired by supermarket inventory practices of visualizing workflow and limiting Work In Progress (WIP)
- Physical Card System Core to the philosophy
- Popularity in software processes rose in the early 2000's
- Integrated into Agile philosophies

Kanban Tenets

- Visualize all work
- Limit Work in Progress
- Keep Increments Small
- Manage flow, not individuals
- Create Feedback Loops
- Make Process Policies Explicit via Working Agreements
- Improve Collaboratively
- Celebrate Small Wins

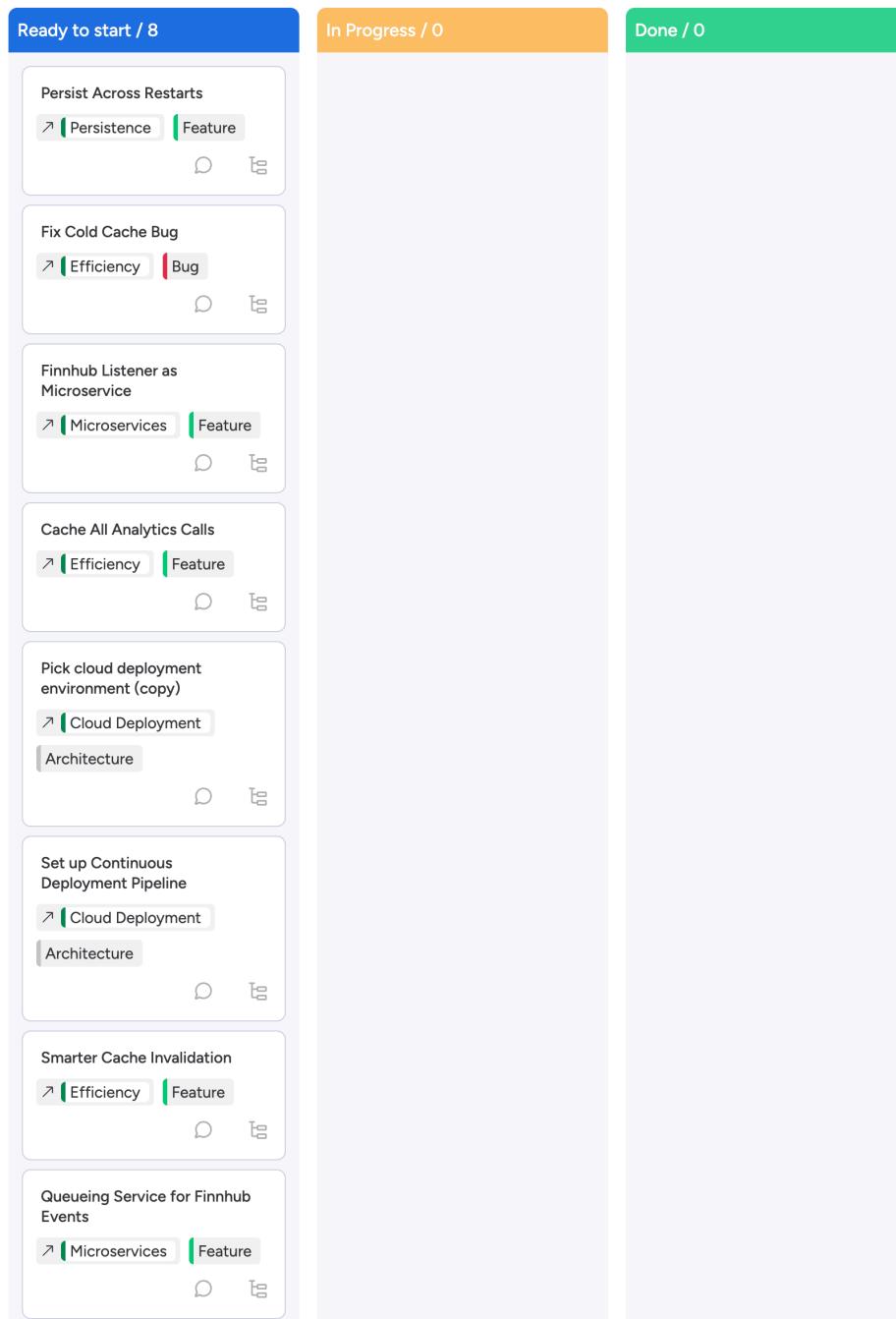




Kanban Board

- Only three states that we care deeply about:
 - What work is prioritized and imminently ready for us?
 - What are we working on currently?
 - What did we recently finish?
- Distinguish
 - Bugs
 - Features





Shar



Larry



Jane



Jarred



Donna



Department

- team of 5 developers
- Mentioned WIP (work in progress) limit earlier.
- **What is a reasonable limit for a team of 5?**





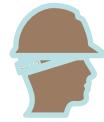
Shar



Larry



Jane



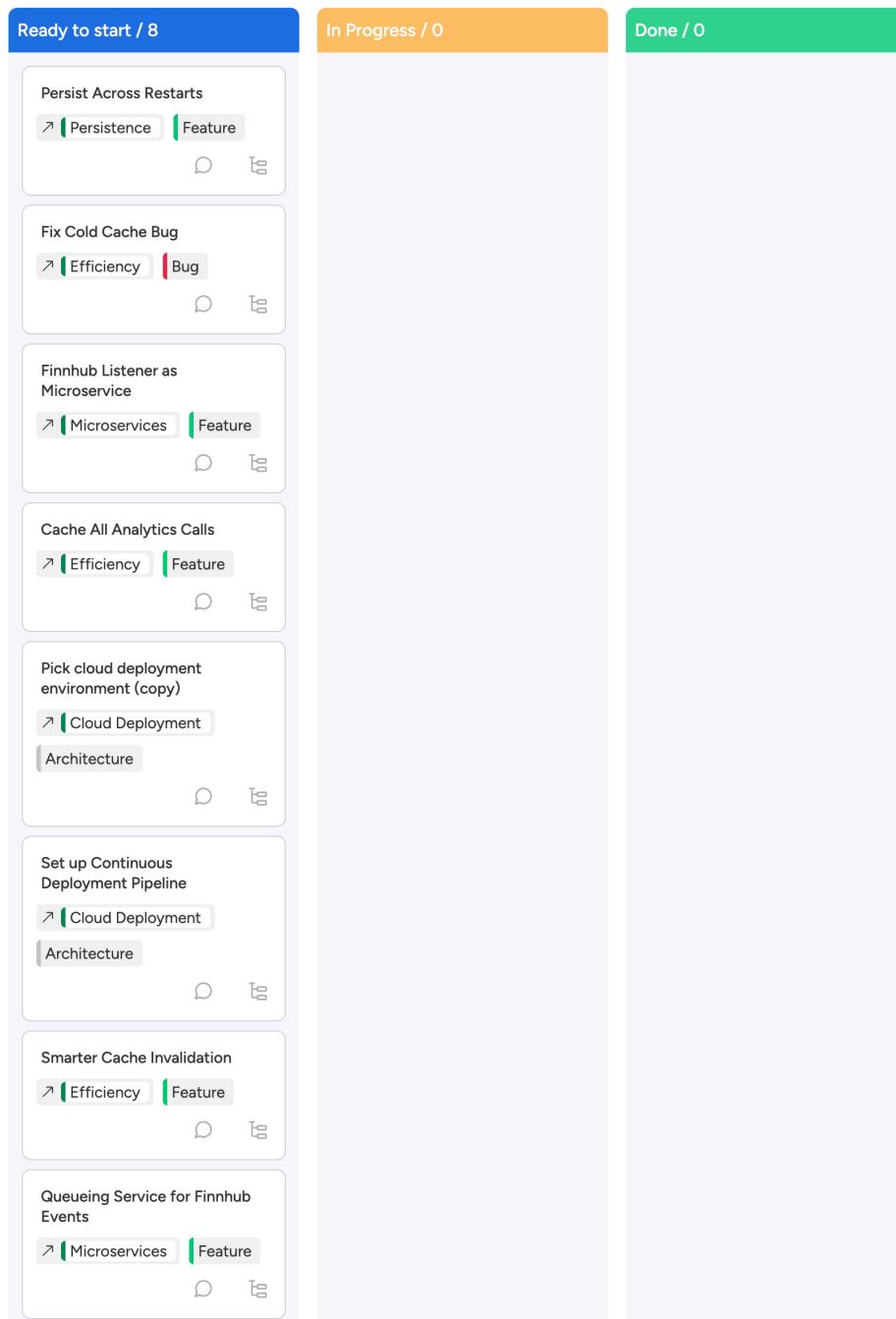
Jarred



Donna

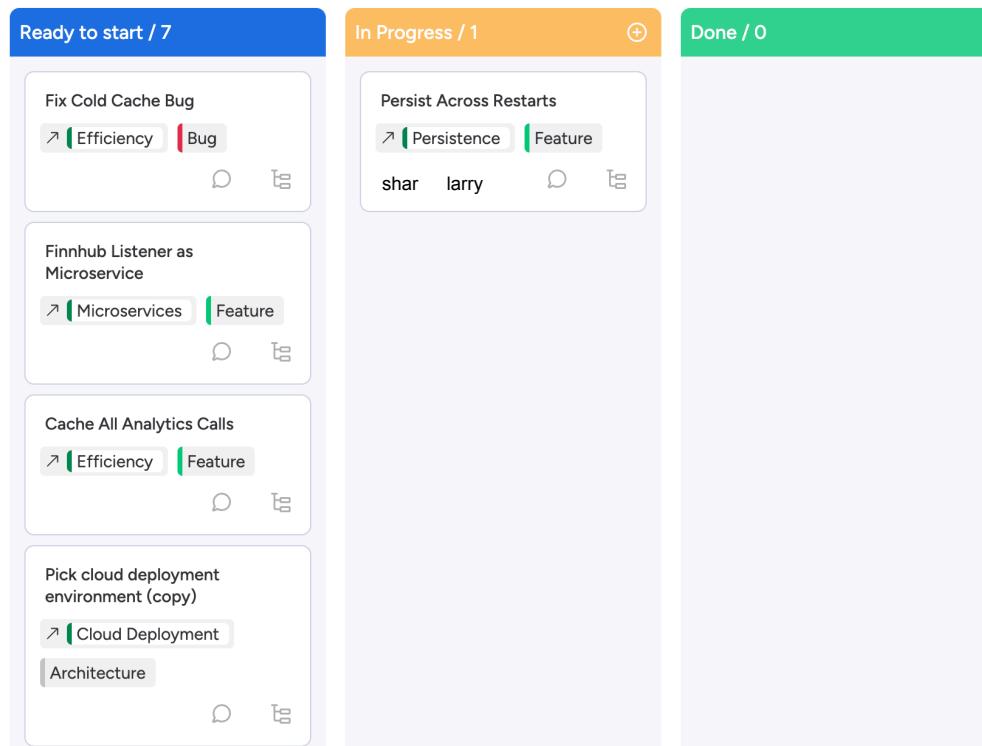


Department



- team of 5 developers
- starting with this prioritized roadmap
- question to ask:
 - who should start work on highest priority task?





Jane



Jarred



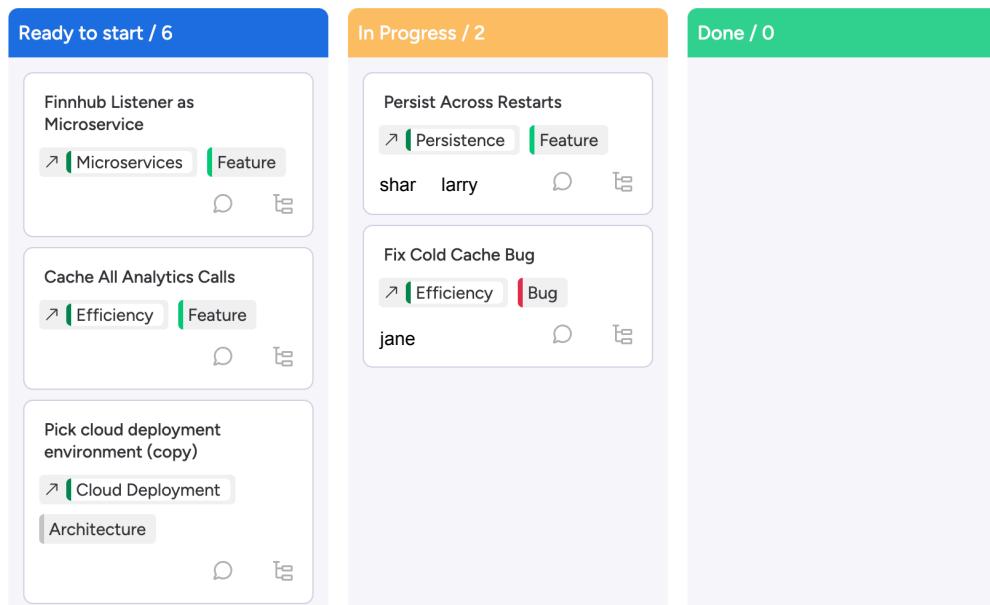
Donna

- Shar & Larry start work on persisting state across StockApp restarts.
- Still have 3 free engineers, what should they do?



Department of Electrical & Computer Engineering





- Jane has some background on caching and volunteers to take a look at the bug
- We avoid anyone working alone on a feature, but for smaller bugs its ok for Jane to look at this on her own
- What do Jarred and Donna do?



Jarred

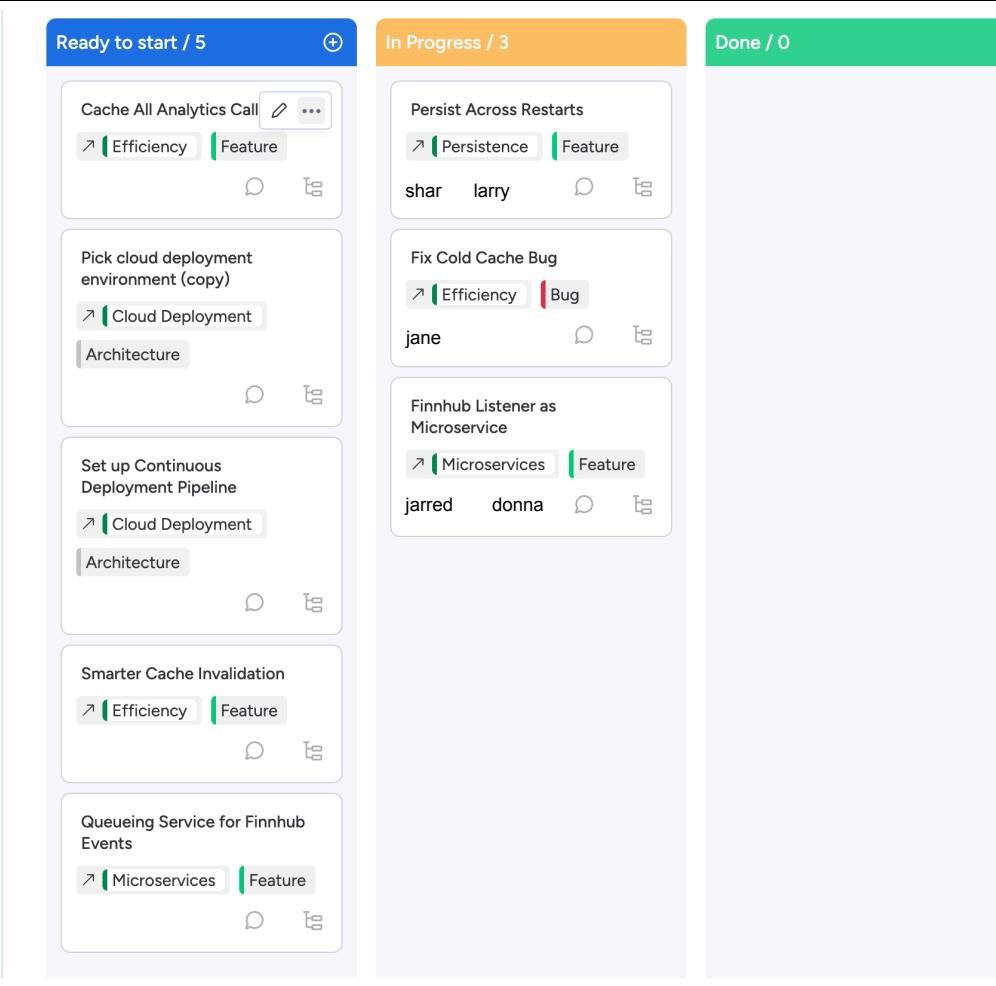


Donna



Department of Electrical & Computer Engineering

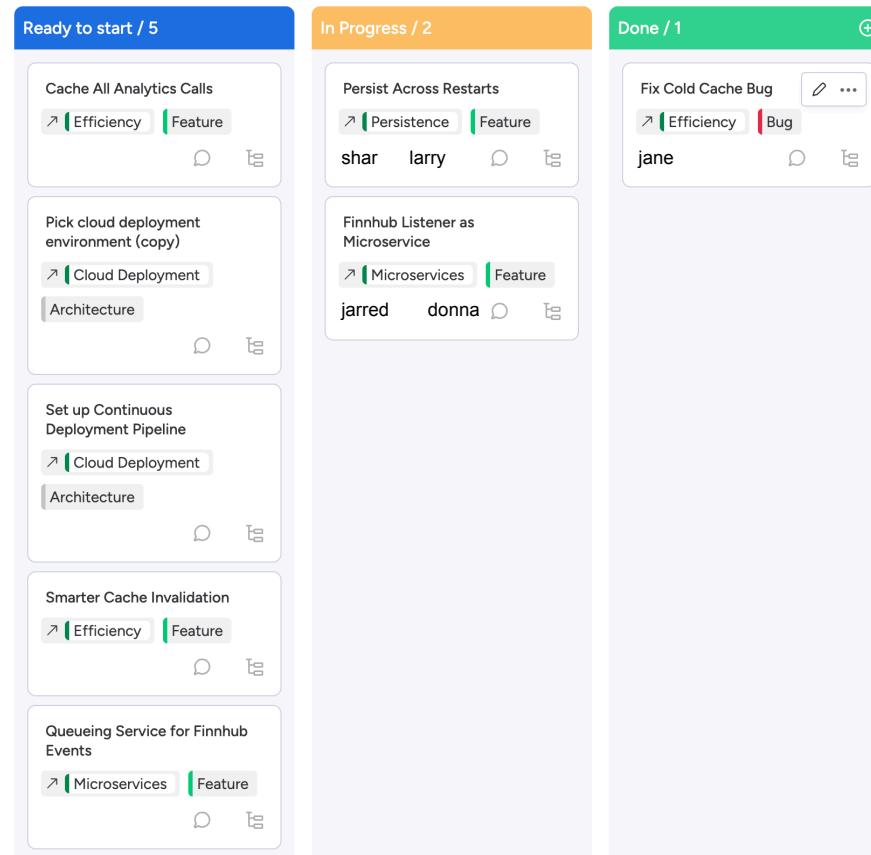




- Jarred and Donna pick up the Finnhub Listener as Microservice work and now our team are all making progress towards our highest priority tasks...



Jane



- One day later...
- Jane ships the cache bug fix (jarred took an interrupt to review her Pull Request)
- Jane has now freed up, what should she do?

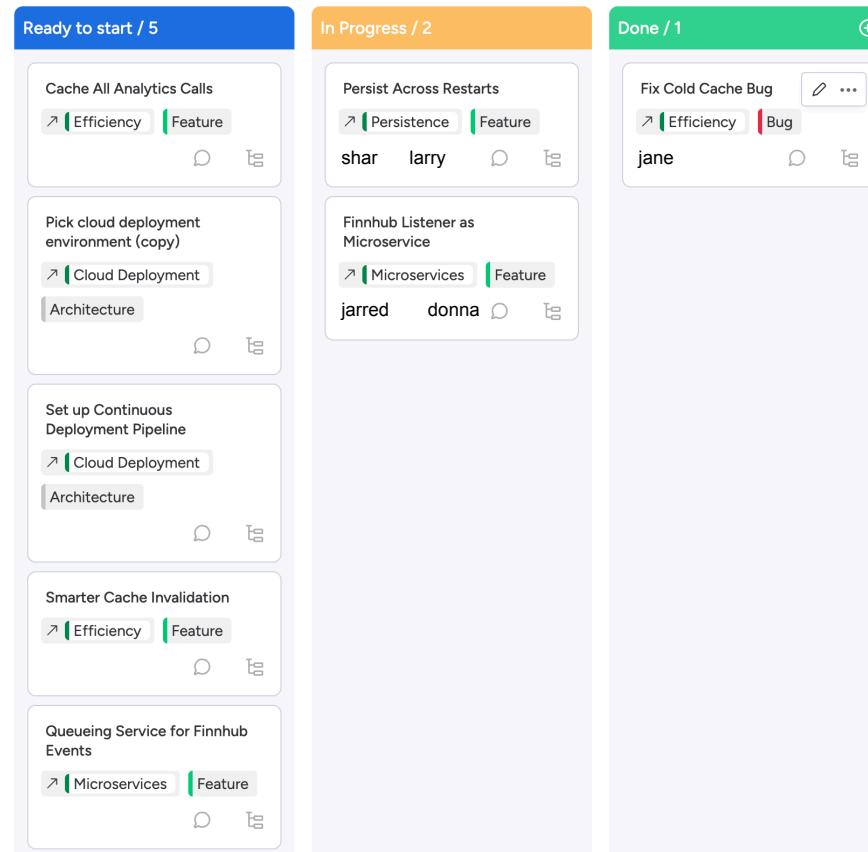


Department of Electrical & Computer Engineering





Jane



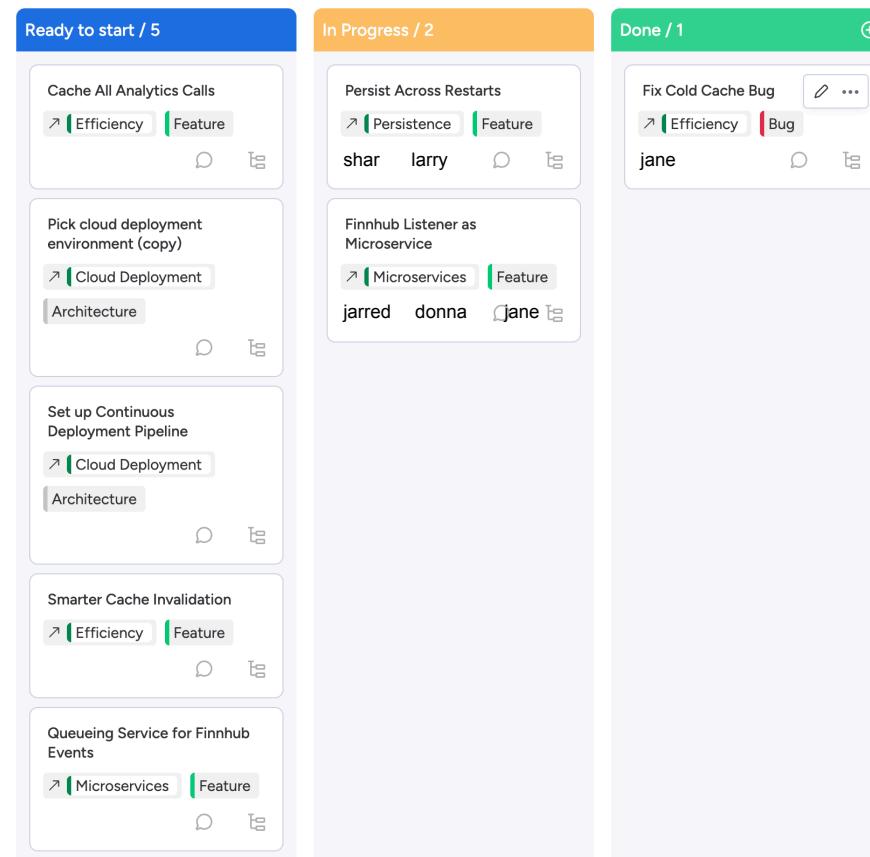
- First, everyone gives Jane a high five for shipping the fix!

- Pulling a new feature would push our WIP too high - 3 features across 5 engineers is too much.
- Jane should ask Shar & Larry and Jarred & Donna which team could use more help and join one of the two in progress features based on their response and her skill set.



Department of Electrical & Computer Engineering





- Jane has some experience with WebSockets and Jarred and Donna tell her that there is plenty of work left to do on their project, so she joins their team.



Kanban Tenets

- Visualize all work
- Limit Work In Progress
- Keep Increments Small





developer

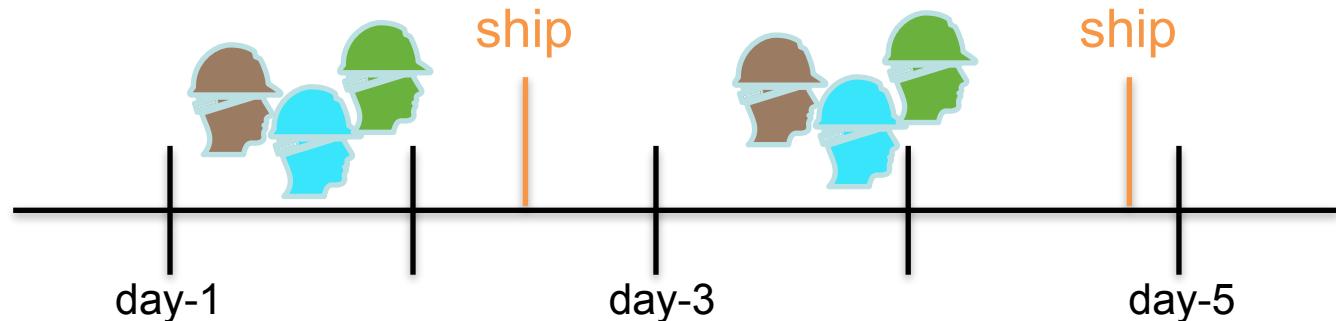


quality
assurance



release
engineer

Merging of Roles



- two factors pushing for the merging of these 3 roles:
 - automation of testing and integration + deployment
 - compressed time frames & iterations
- resulting in ...
 - DevOps!



Department of Electrical & Computer Engineering



Day 3 Final Projects

- Work in Teams of 4 or 5 - people at your table are a great group to work with!
- Pick one of your repos as the one to use for your final project and:
 - Set up a Kanban board in your GitHub Repo
 - Invite others from your team to that repo
 - For the repo that you use, require that:
 - changes be merged into main via PullRequests
 - tests pass on the PullRequest branch before they can be merged
- Given that you have access to Lambda functions and S3, think about what kind of interesting / useful function - API based application you could build in one day
 - Add tasks to your Kanban board to reflect this and plan the implementation.



Final Project - Ideas

1. Big Data!

- Measure processing time for large CSV's (10,000 rows then 100,000 rows, then 1,000,000 rows)
- Can you improve initial processing time by changing the algorithms and data structures you are using?
- Can you improve initial processing time by splitting the work up and invoking your lambda multiple times in parallel?

2. Choose Your Own Adventure!

- Have your lambda render an HTML with options that are links back to your lambda that process choice and present further options
- Be creative! Do you need special handling for every state?
- Can some game states be handled in a generic way?
- Can multiple players play your game at the same time?



Final Project - Ideas

3. Headline Tracker

- Your lambda is schedule to run once every few minutes
- On each run it:
 - loads a designated file from S3
 - finds out the current headline on [NYT.com](#)
 - appends it to the S3 file along with timestamp
- As a result, the file in S3 has the minute-by-minute headline tracking for the day

4. Stock Tracker

- Your lambda is schedule to run once every few minutes
- On each run it:
 - reads a CSV from S3 that contains stock symbols
 - for each stock symbol in the file it
 - calls a public API to check the current price of the stock
 - writes the price evolution of that stock to a different file in S3
 - optionally identifies interesting price fluctuations





Keep in touch! - esolovey@bu.edu