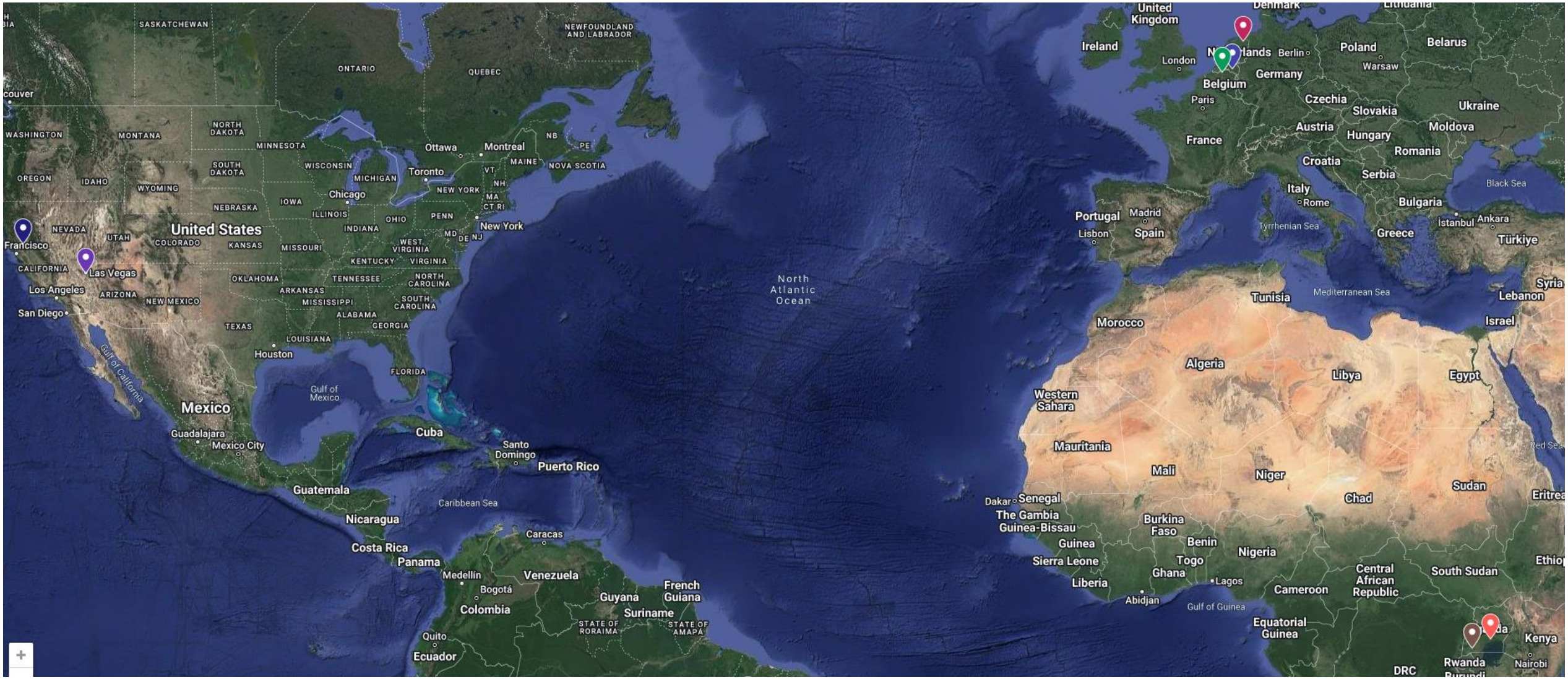
Software Product Design and Development I

Dr. John Businge
John.businge@unlv.edu

TA: Daniel Ogenrwot
ogenrwot@unlv.nevada.edu

My Journey to UNLV



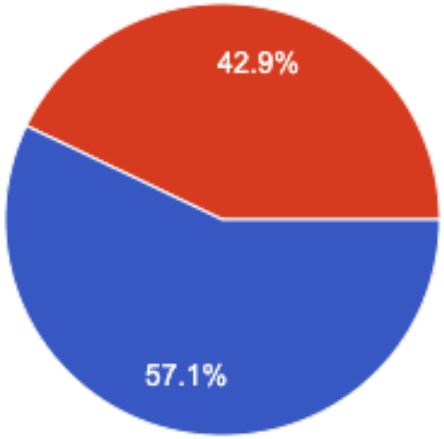
Administration

- Background Information survey.

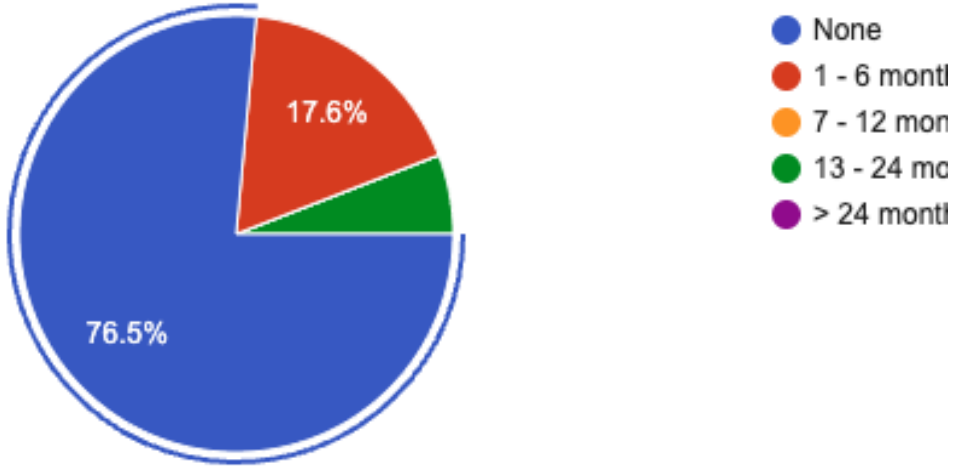
- Go to - https://johnxu21.github.io/teaching/CS472/

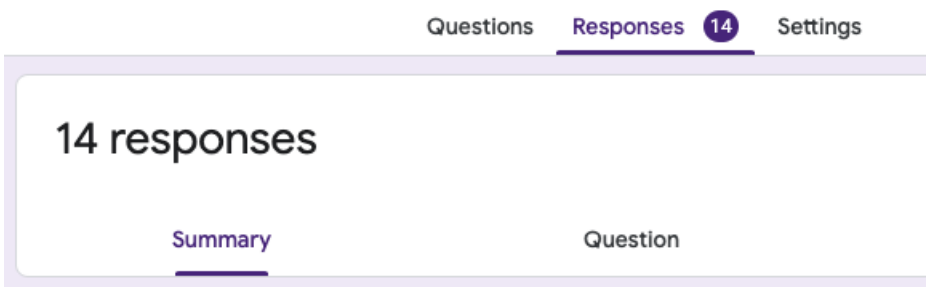
How many months of industry software development experience did you have before the beginning of class 472/672?

Current Semester



Last Semester





Inclusivity and Gradual Learning

- We want to build confidence for all students through gradual, structured learning
- Familiar material for experienced developers serves as refinement and perspective-sharing
- Help in reflecting real-world collaboration where diverse teams work together
- Encourages mutual support between **experienced** and **novice** peers

Software Product Design and Development I

High Quality Software

Flexible

Reusable

Maintainable

Collaboratively

10 – 11 developers / team project

Design Portfolio I

Design Portfolio II

>= 3 PRs
>= 3 PR reviews
/team member

Design Portfolio III

>= 3 PRs
>= 3 PR reviews
/team member

Agile methodology: Development delivers frequent, small software updates, enabling teams to quickly adapt to changes and continuously improving the product based on feedback.

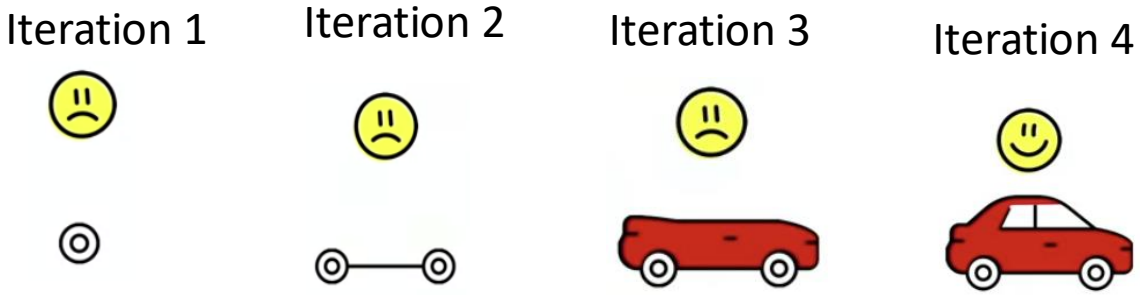
Minimum Viable product (MVP)

- During the DP we will focus on developing an MVP
- An MVP is the minimal thing that you can do to test a value hypothesis and gain learning and understanding
- MVP is focus on learning, not delivery
- Let us explain an MVP with an example using Agile developmemnt

Minimum Viable product (MVP)

Customer wants a red car

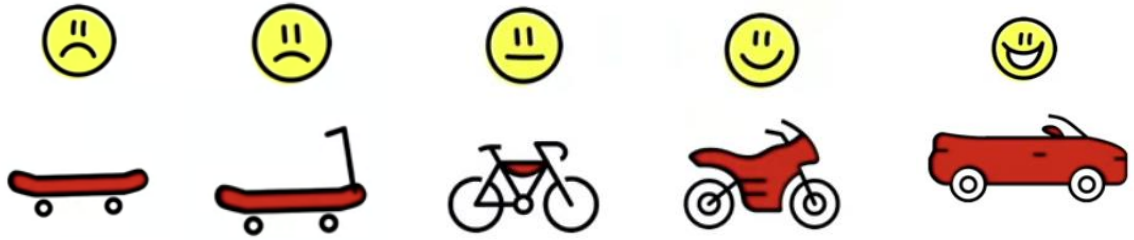
Agile Dev 1



No feedback

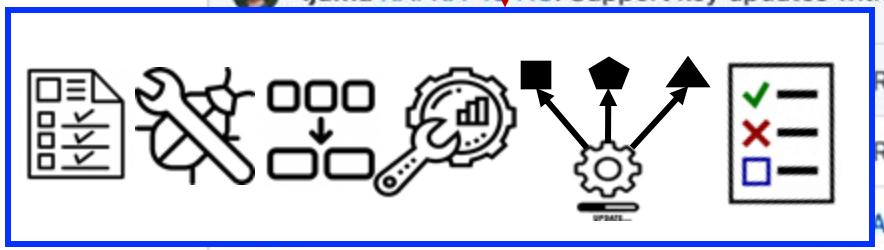
- Customer got exactly what they asked for
  - the dev team was just following a plan.
  - Increments were not useful.
- Team does not understand the value of MVP

Agile Dev 2



Valuable feedback on every iteration

- Customer got what they desired
  - Worked iteratively with the dev team.
- Developed something a little bit different but it's closer to what the customer really wanted.
- Giving the customer what they really want is the main purpose of delivering an MVP.
- A minimal viable product is a tool for learning.



apache / kafka Public

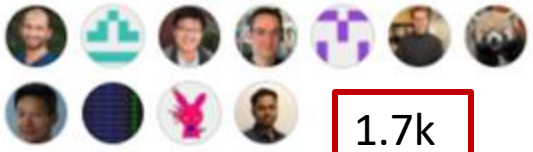
Watch 1.1k | Fork 11.3k | Star 21.5k

Pull requests 953 Open ✓ 11,016 Closed

ijuma KAFKA-1[illegible]418: Support key updates with TLS 1.3 (#11966) ... 5aed178 12 hours ago 9,874 commits

	R: Adding kafka-storag[illegible].bat file (similar to kafka-storage.sh) fo...	16 days ago
	R: Fix class comparis[illegible] in `AlterConfigPolicy.RequestMetadata...	8 days ago
	[illegible]966)	12 hours ago
config	MINOR	nfiguration files...	last month
connect	KAFKA	lLog using older...	5 days ago
core	MINOR	hBufferedRecei...	21 hours ago
docs	KAFKA	3)	22 hours ago
examples		s in examples README (#855...	2 years ago
generator/src		gged string fields in messag...	7 months ago
gradle		(#11885)	14 days ago
jmh-benchmarks		(#11870)	19 days ago
licenses	MINOR: Add missing licenses and update versions in LICENSE-binary...		7 months ago

Contributors 884



1.7k

+ 873 contributors

Languages

Java 74.2% Scala 22.7%

Python 2.7% Shell 0.2%

Roff 0.1% Batchfile 0.1%

Apache Kafka is a distributed event store and stream-processing platform

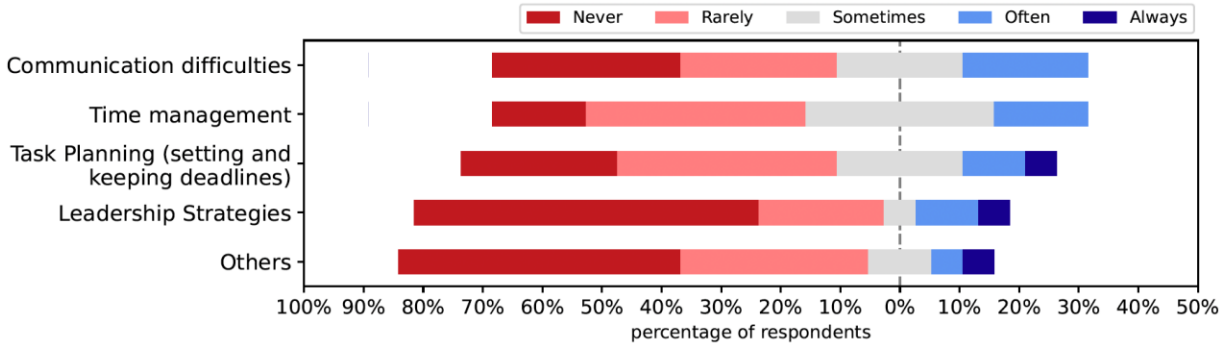
“People-related factors tend to be the greatest challenges—not technology.”

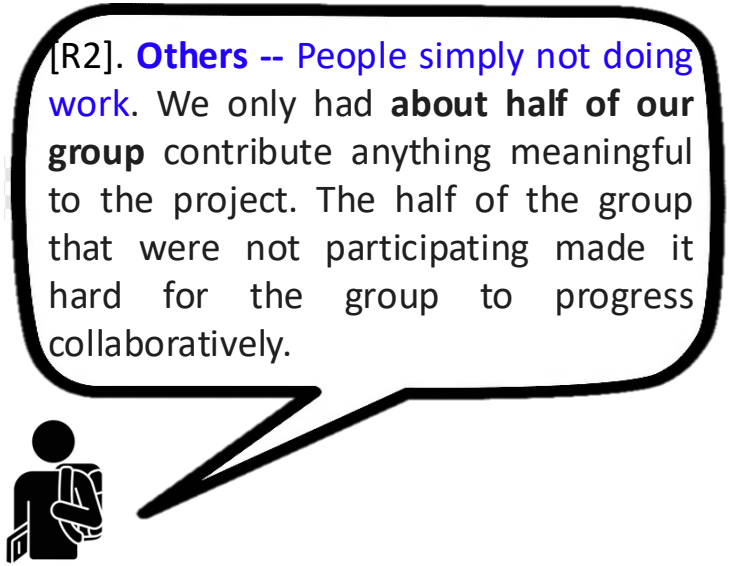
George Spafford,
Senior Director Analyst
at Gartner

Survey Results – Team Challenges

Qn.6: Please rank the following challenges that could have impeded effective teamwork.

Qn.7: If your ranking for "Others" in Qn.6 above was 4 or 5, kindly provide us what it represents.





What is social coding?

- Open source practice - Open Source for Inner Source
  - Inner source - adoption of open-source development practices, tools, and culture within an organization
- All repositories are public
- Everyone is encouraged to contribute
- Contribute back via Pull Requests

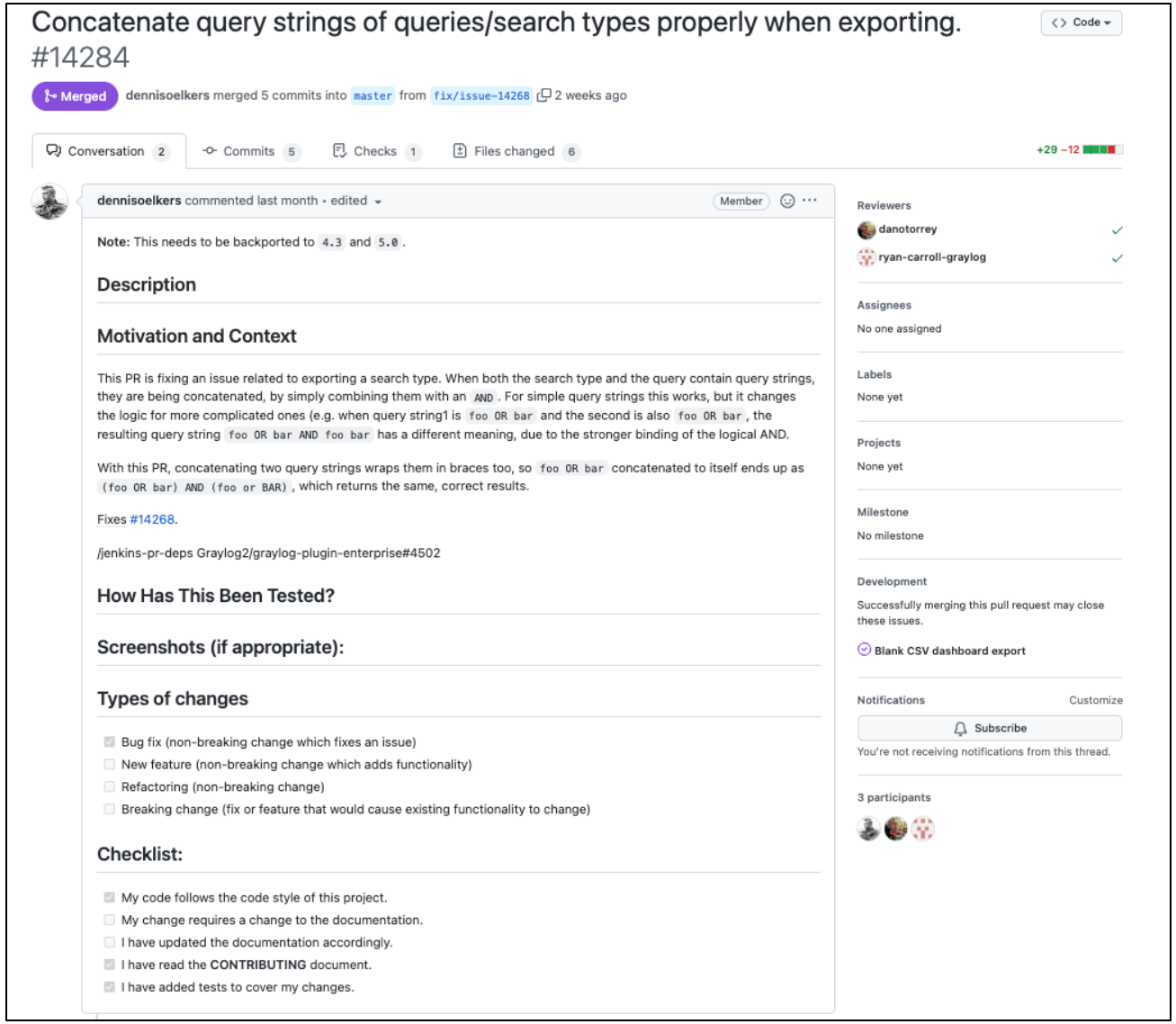
Git-Github repository guidelines

- Create a repository for a project
- Create a new branch for every issue
- Use a Pull Requests to merge to mainline
- Every Pull Request is an opportunity for code review

Git Feature branch workflow

Best Practices Pull Requests Documentation

https://github.com/Graylog2/graylog2-server/pull/14284



Concatenate query strings of queries/search types properly when exporting. #14284

Code

Merged dennisoelkers merged 5 commits into master from fix/issue-14268 2 weeks ago

Conversation 2 Commits 5 Checks 1 Files changed 6 +29 −12

dennisoelkers commented last month • edited Member

Note: This needs to be backported to `4.3` and `5.0`.

Description

Motivation and Context

This PR is fixing an issue related to exporting a search type. When both the search type and the query contain query strings, they are being concatenated, by simply combining them with an `AND`. For simple query strings this works, but it changes the logic for more complicated ones (e.g. when query string1 is `foo OR bar` and the second is also `foo OR bar`, the resulting query string `foo OR bar AND foo bar` has a different meaning, due to the stronger binding of the logical AND.

With this PR, concatenating two query strings wraps them in braces too, so `foo OR bar` concatenated to itself ends up as `(foo OR bar) AND (foo or BAR)`, which returns the same, correct results.

Fixes #14268.

/jenkins-pr-deps Graylog2/graylog-plugin-enterprise#4502

How Has This Been Tested?

Screenshots (if appropriate):

Types of changes

- [x] Bug fix (non-breaking change which fixes an issue)
- [ ] New feature (non-breaking change which adds functionality)
- [ ] Refactoring (non-breaking change)
- [ ] Breaking change (fix or feature that would cause existing functionality to change)

Checklist:

- [x] My code follows the code style of this project.
- [ ] My change requires a change to the documentation.
- [ ] I have updated the documentation accordingly.
- [x] I have read the **CONTRIBUTING** document.
- [x] I have added tests to cover my changes.

Reviewers
danotorrey ✓
ryan-carroll-graylog ✓

Assignees
No one assigned

Labels
None yet

Projects
None yet

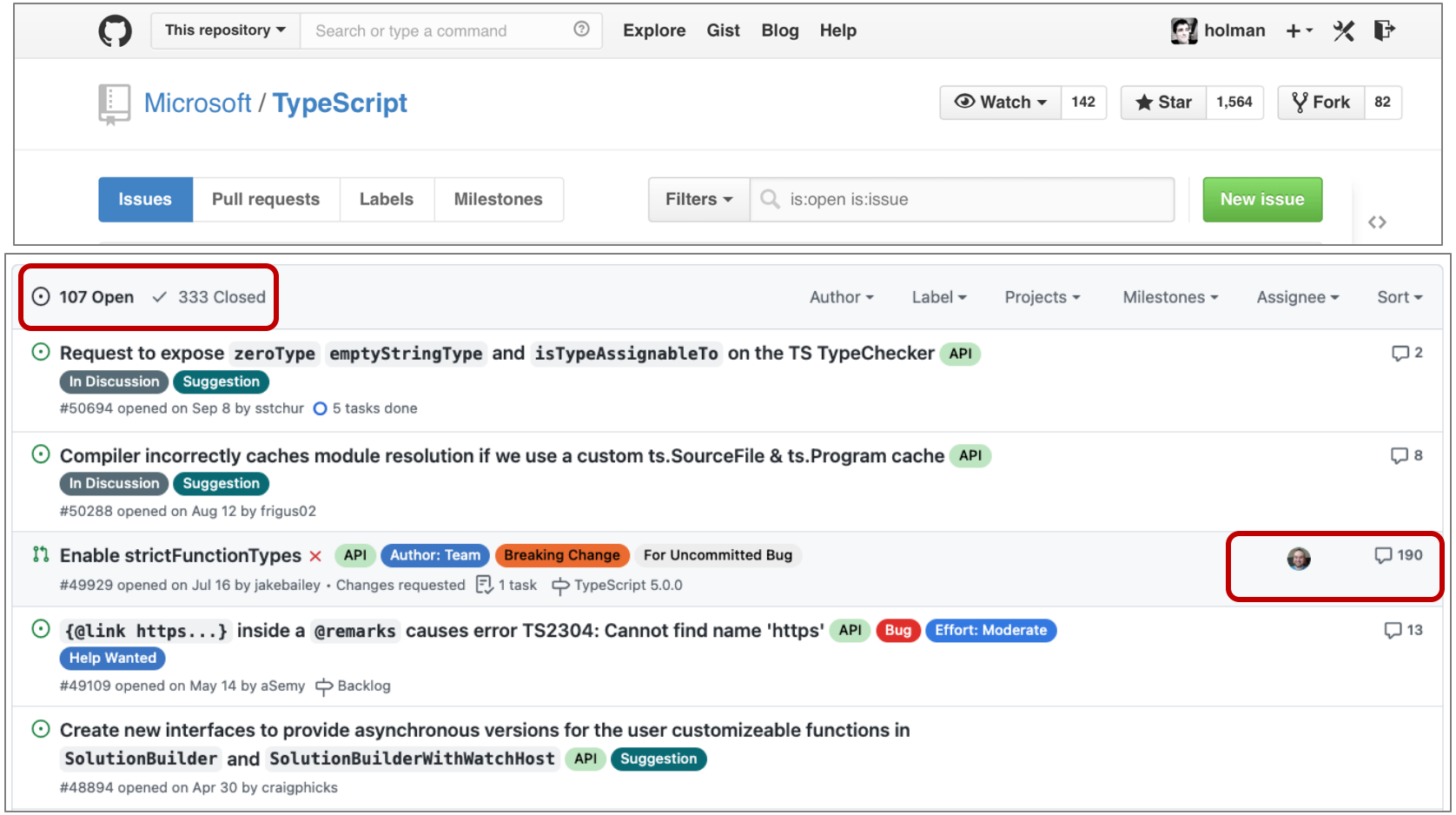
Milestone
No milestone

Development
Successfully merging this pull request may close these issues.
Blank CSV dashboard export

Notifications Customize
Subscribe
You're not receiving notifications from this thread.

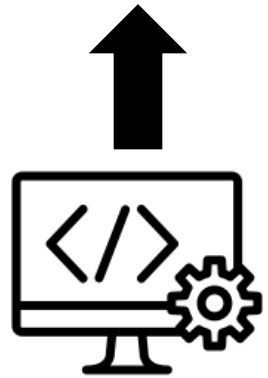
3 participants

Issue Tracker - GitHub



Issues

- Use cases
- Bugs to fix
- Features to add
- Documentation



Software Project

Testing

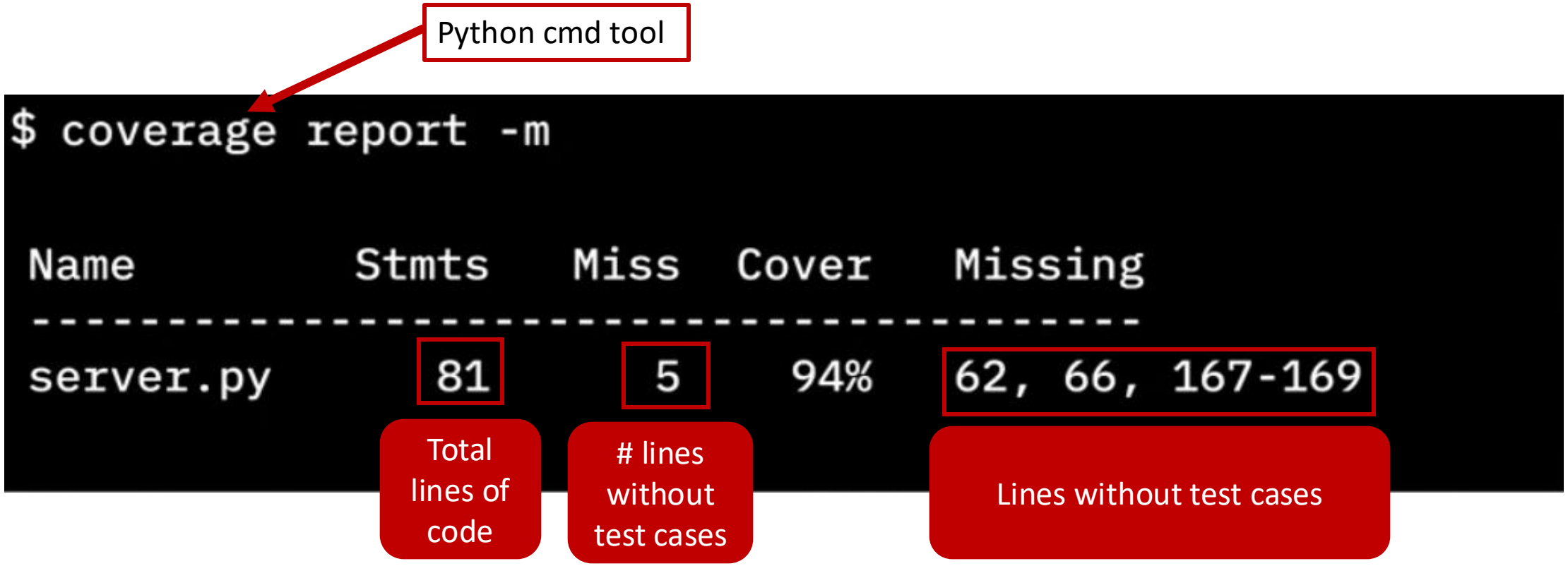
"If it is worth building, it is worth testing.

If it is not worth testing, why are you wasting your time working on it?"

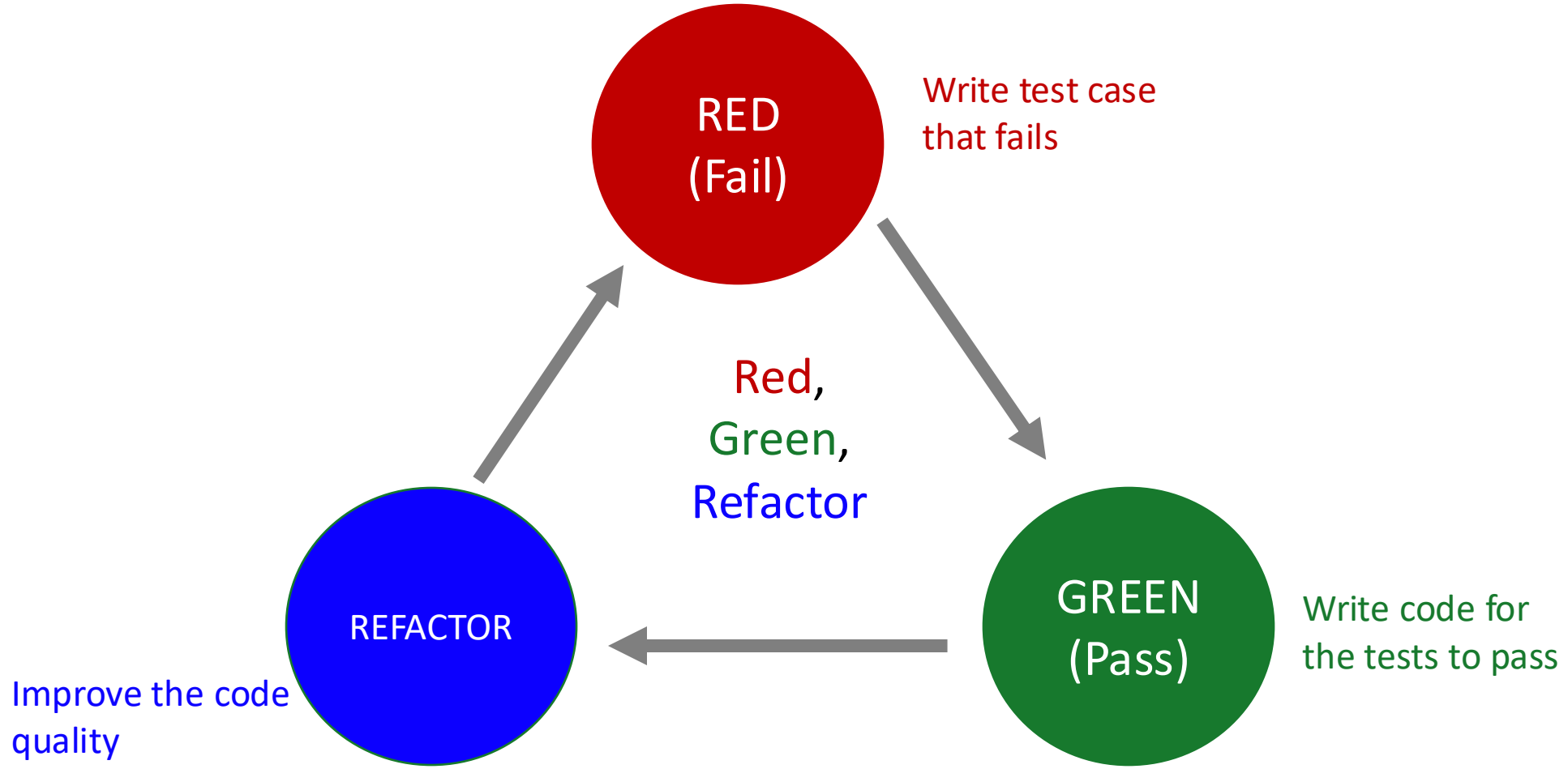
Scott Ambler, agiledata.org

Importance of test coverage

- High test coverage gives you confidence that your code works as expected
- Test coverage reports can reveal which lines of code were not tested

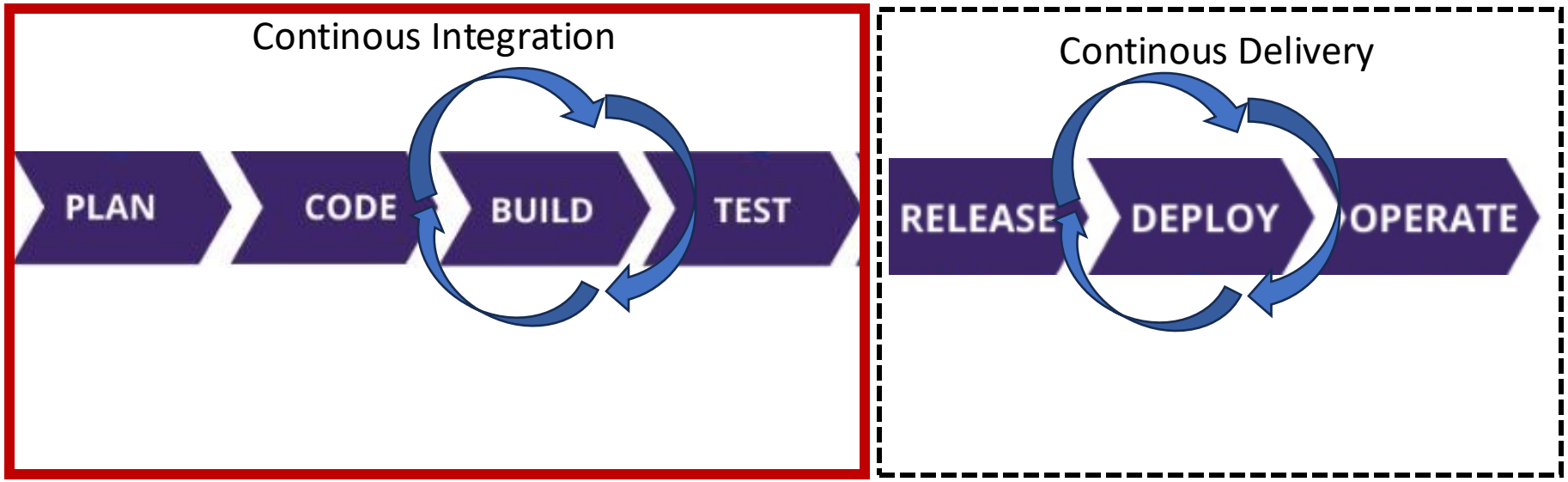


Basic TDD workflow



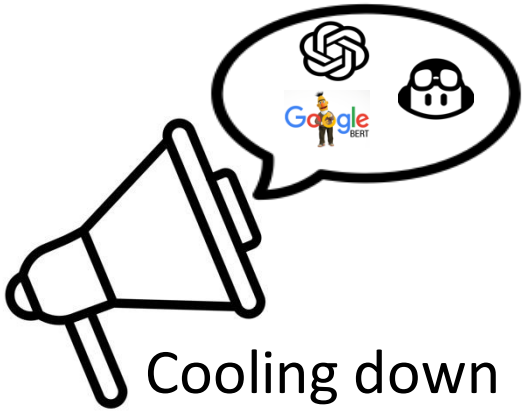
I have prepared the lab which will guide you in practicing with TDD

CI/CD pipeline

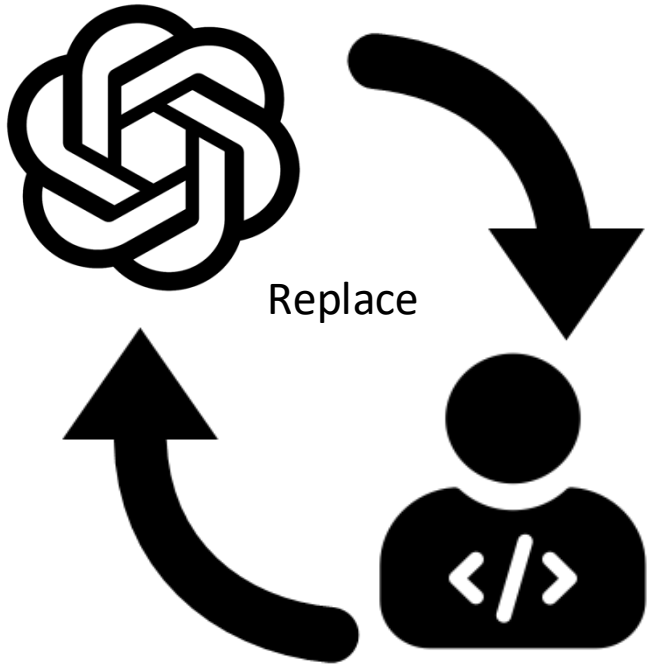


Using Generative AI to perform specific SDLC activities

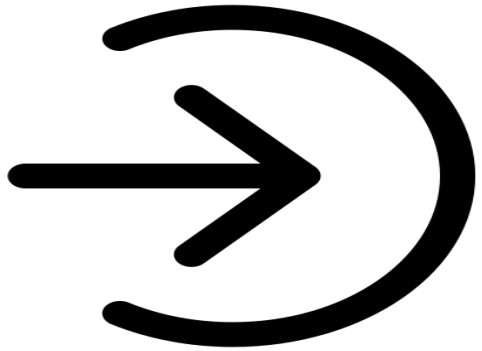
The Hype around Generative AI for Software Development



Cooling down

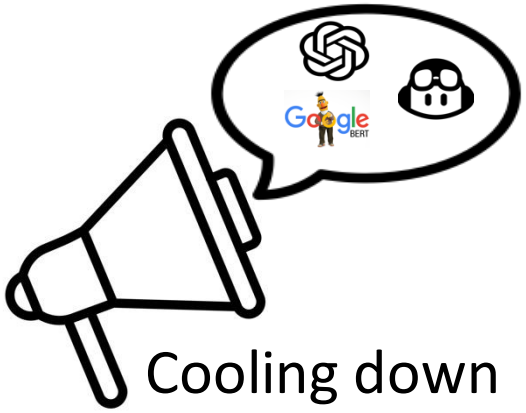


Replace

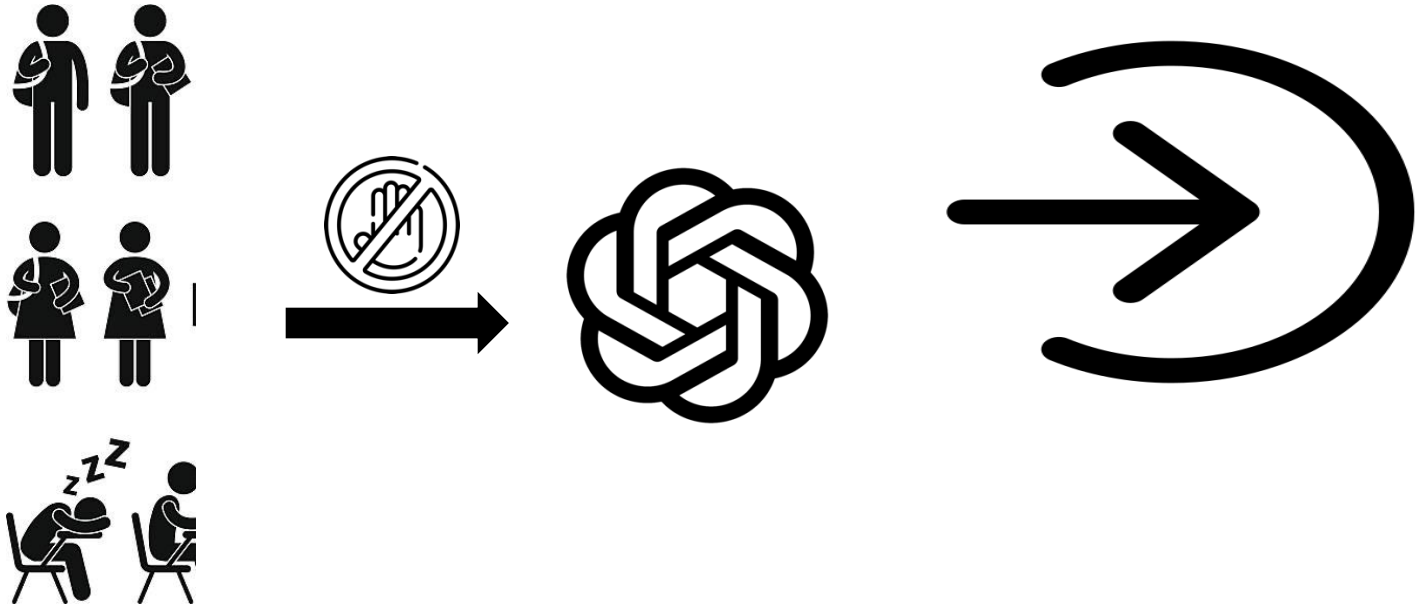


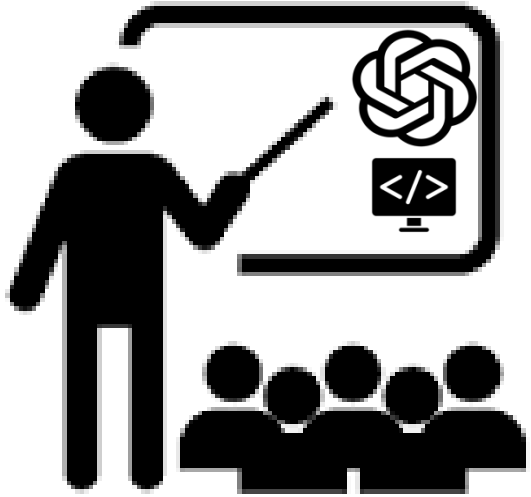
Partner

The Hype around LLMs for Software Development



Cooling down





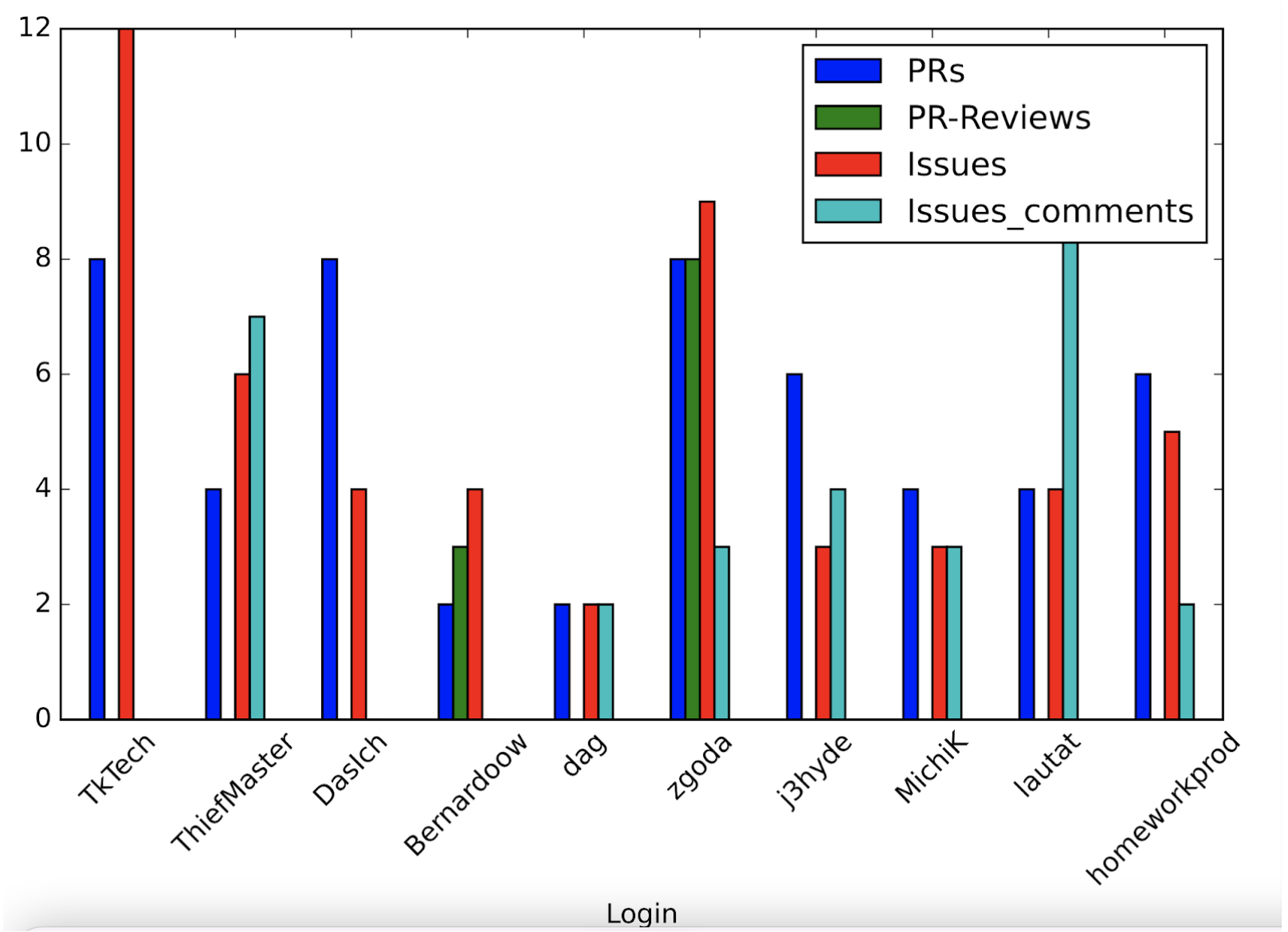
What SE Tasks have been addressed to date using LLM4SE

SE Activity	SE Task		Total
Requirements engineering	Anaphoric ambiguity treatment (3)	Requirements term identification (1)	11
	Requirements classification (3)	Coreference detection (1)	
	Requirement analysis and evaluation (2)	Traceability automation (1)	
Software design	GUI retrieval (1)	Software specification synthesis (1)	3
	Rapid prototyping (1)		
Software development	Code generation (62)	Agile story point estimation (1)	136
	Code completion (16)	API documentation smell detection (1)	
	Code summarization (10)	API entity and relation extraction (1)	
	Code understanding (7)	Code optimization (1)	
	Code search (5)	Code example recommendation (1)	
	Program synthesis (5)	Control flow graph generation (1)	
	API recommendation (2)	Data analysis (1)	
	API synthesis (2)	Identifier normalization (1)	
	Code comment generation (2)	Instruction generation (1)	
	Code representation (2)	Type inference (1)	
	Method name generation (2)	Others (11)	
Software quality assurance	Test generation (8)	Bug localization (1)	24
	Vulnerability detection (7)	Failure-inducing test identification (1)	
	Test automation (4)	Flaky test prediction (1)	
	Verification (2)		
Software maintenance	Program repair (23)	Duplicate bug report detection (1)	58
	Code review (6)	Decompilation (1)	
	Debugging (4)	Program merge conflicts repair (1)	
	Bug report analysis (3)	Sentiment analysis (1)	
	Code clone detection (3)	Tag recommendation (1)	
	Logging (2)	Vulnerability repair (1)	
	Bug prediction (1)	Commit classification (1)	
	Bug triage (1)	Traceability recovery (1)	
	Bug report replay (1)	Others (6)	
Software management	Effort estimation (1)		1

Hou et al. LLMs for SE: A Systematic Literature Review

- https://arxiv.org/pdf/2308.10620.pdf
- Analyzed 229 research papers on the subject
- Read Section 6 of the paper to find which papers have addressed the SE tasks.

Assessment


12
10
8
6
4
2
0
PRs
PR-Reviews
Issues
Issues_comments
TkTech
ThiefMaster
DasIch
Bernardoow
dag
zgoda
j3hyde
MichiK
lautat
homeworkprod
Login

A list of previous projects by students

https://github.com/orgs/UNLV-CS472-672/repositories?type=all