

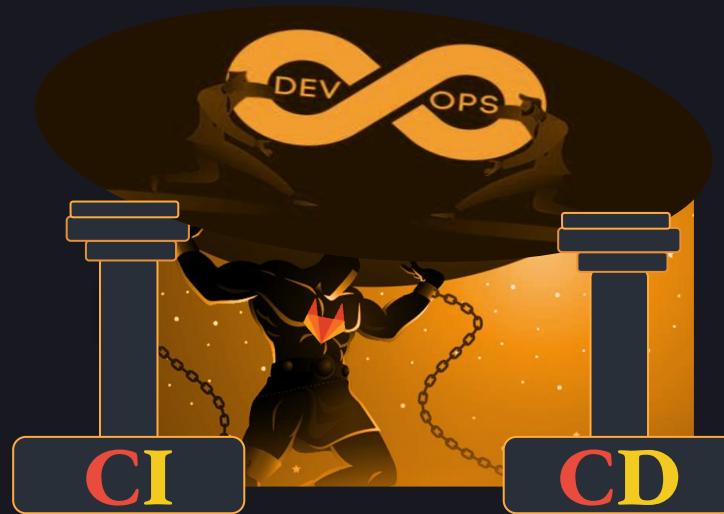


# Continuous Integration



# Continuous Deployment

The Foundation Of Modern DevOps

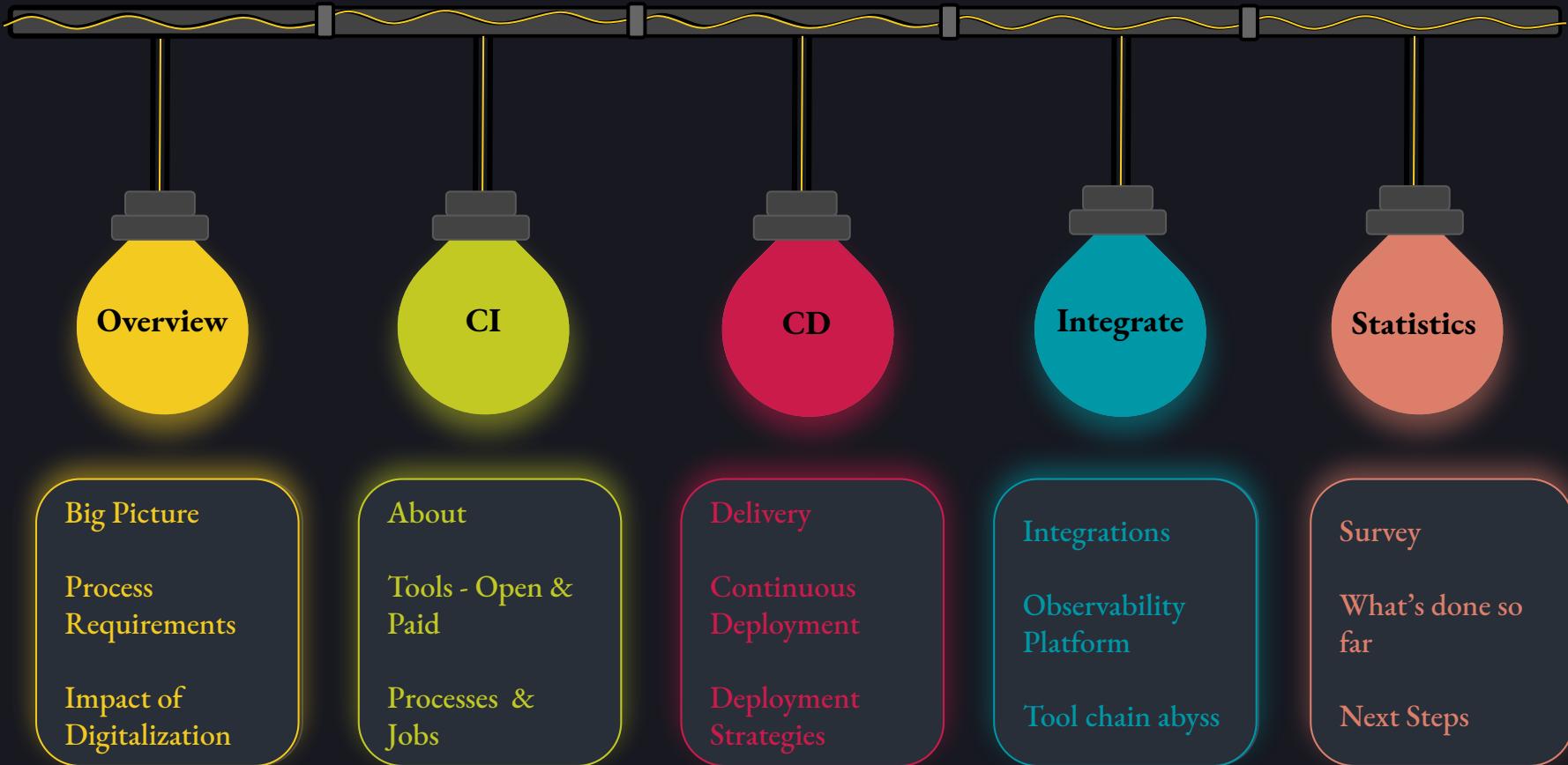


Pravin MJ



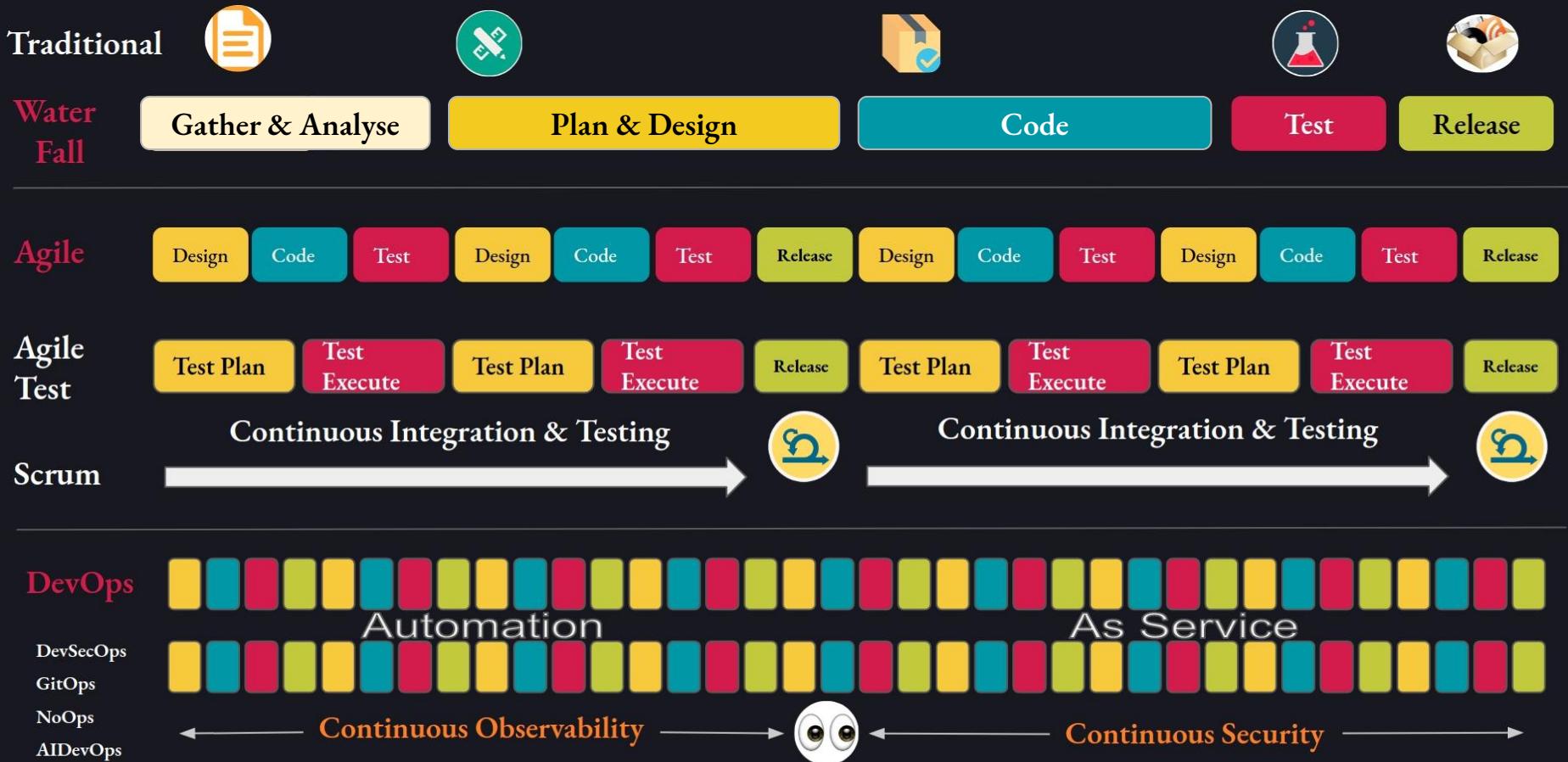


# Session Roadmap

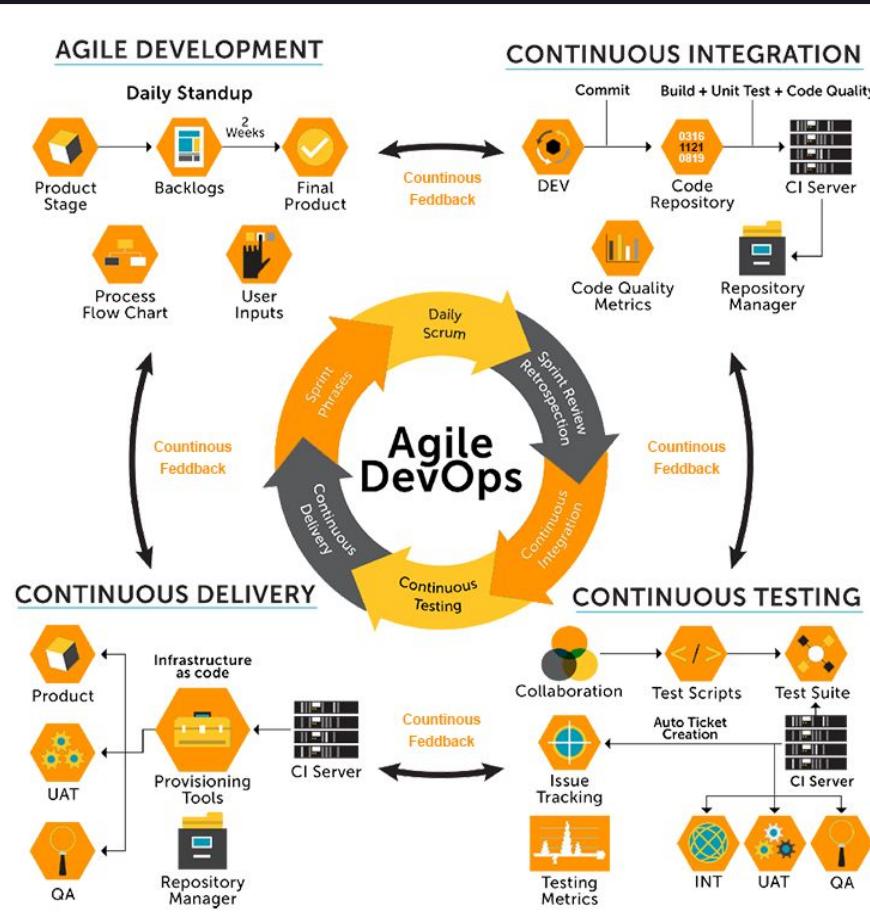




# The Path Of Software Enlightenment



# Agile To DevOps



Some Process Methodology in place

→ Choice of best value

BDUF  
JEDI

→ Cross Role responsibilities

→ But One POC

→ REST, GraphQL  
Microservices



# Software Is Eating The World

Everything is a Software  
Roles are Evolving

Management gets detailed  
Knowledge becomes open-sourced

Maintenance becomes complex  
Automation Eats Software

AI Will Eat Automation

Service / API-First



# To The Future



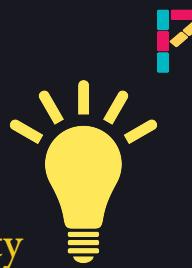
Python is in school  
YAML is **standardized**  
NoOps Movement



Developer  
Responsibilities  
Increases



Frameworks Comes for the Developer Rescue  
Tools Comes to Development Rescue  
PAAS, SAAS, IAS, MLAS For Business Rescue  
Automation & AI



**Full Stack** is No longer  
An End to End Tech-Stack  
Its an **End-To-End Responsibility**

**WASM**



Push For **Self Organizing Teams**





# Continuous Integration

- Integration of code changes frequently from multiple users to a single project
- Using Automation for Code Quality, Builds and Tests before merging to repository



AWS CodePipeline



Travis CI



circleci



Bitbucket



Bamboo

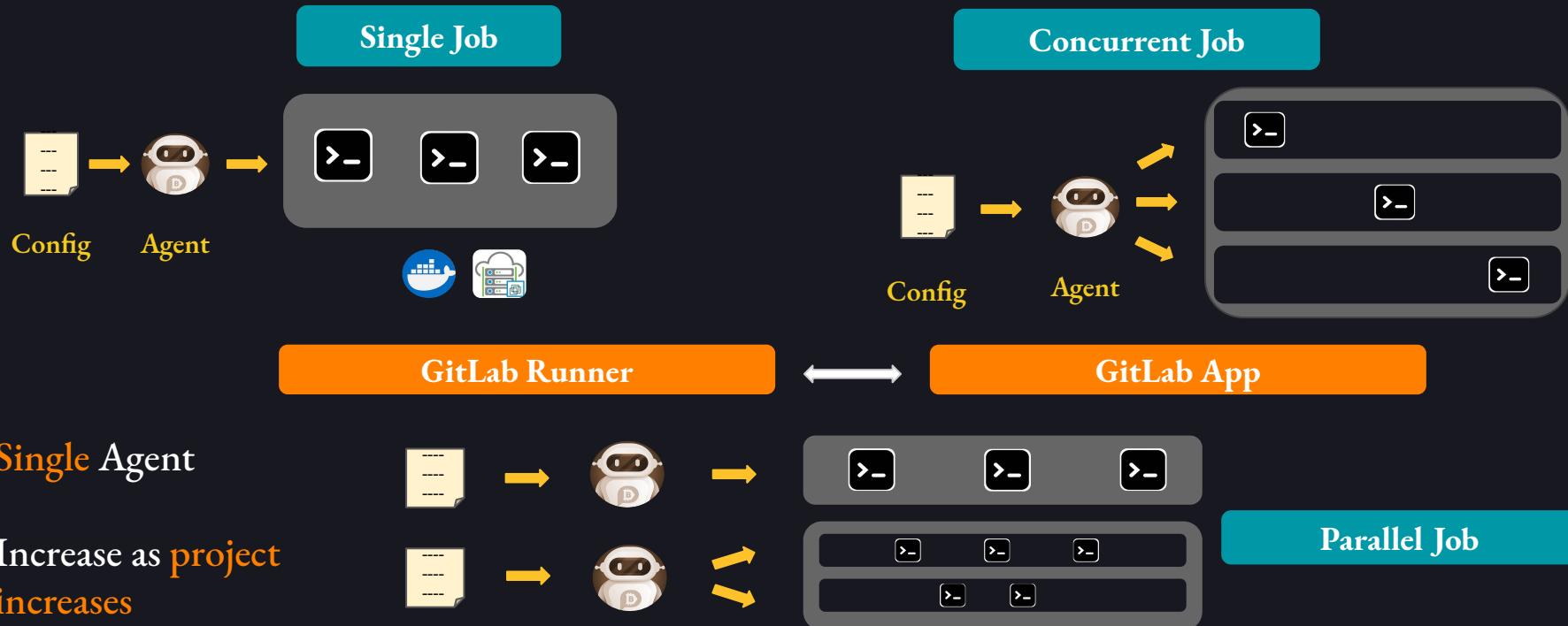


Azure Pipelines



# Continuous Integration

- Integration of code changes frequently from multiple users to a single project
- Using Automation for Code Quality, Builds and Tests before merging to repository





# Continuous Integration Jobs

## Python

Flake8

PyFlakes

PyCodestyle

McCabe Complexity

Pylint

UnitTest

PyTest

Django UnitTest

API

Django ~ Deploy Checklist

Bandit ~ Python

## Code Climate

## SonarQube

Static Check



Test Automation



Security Analysis



## Javascript

ESLint/TSLint

Build

Mocha

Jest

Jasmine

Puppeteer

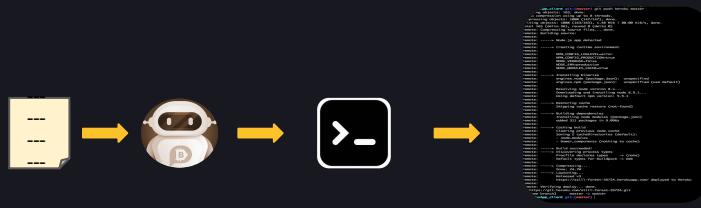
SAST/Eslint  
Security plugin

# Continuous Delivery

→ Integrated code changes to an environment (dev, staging, UAT)

→ Changes are continuously moved to production environment

Continuous Deployment



Target Machine



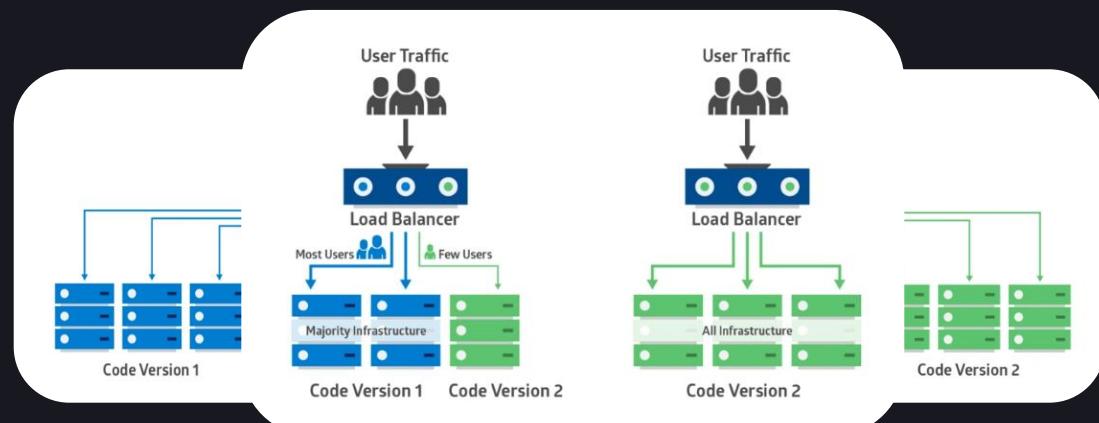
System Exit Code

Config    Agent    Shell Executor

Configure roll backs to any commits  
Heavily Container dependent  
**Rolling Deployment**  
**Blue Green Deployment**



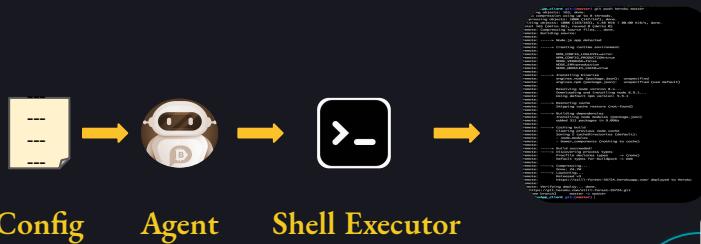
**Canary Deployment**



# Continuous Delivery

- Integrated code changes to an environment (dev, staging, UAT)
- Changes are continuously moved to production environment

Continuous Deployment



Target Machine



Config    Agent    Shell Executor

Configure roll backs to any commits  
Containers extends to stateful set  
**Rolling Deployment**  
**Blue Green Deployment**



**Canary Deployment**

GitLab



GitHub Actions

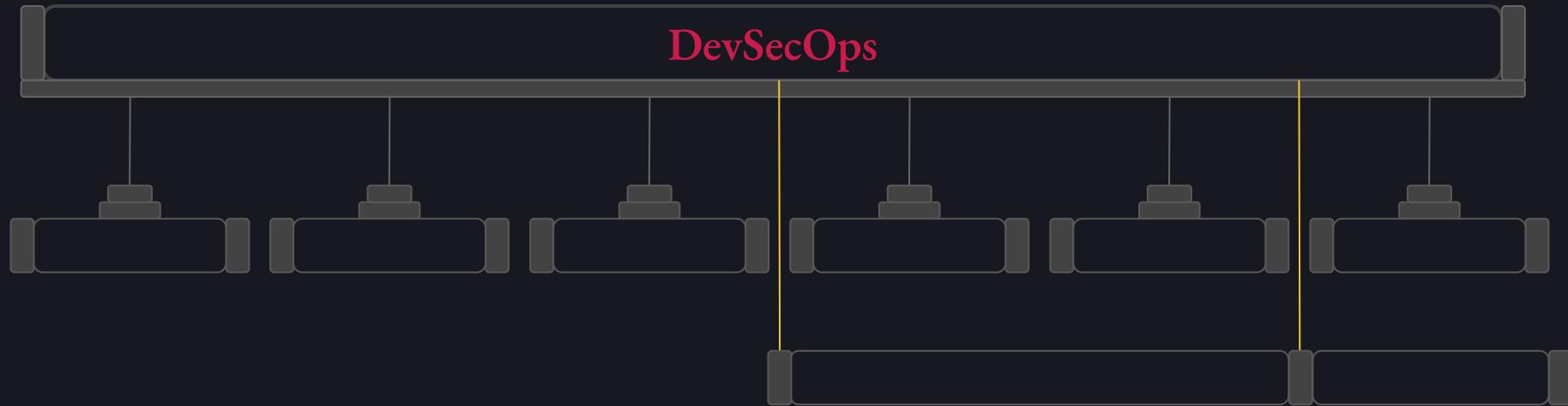


Alternatives





## DevSecOps

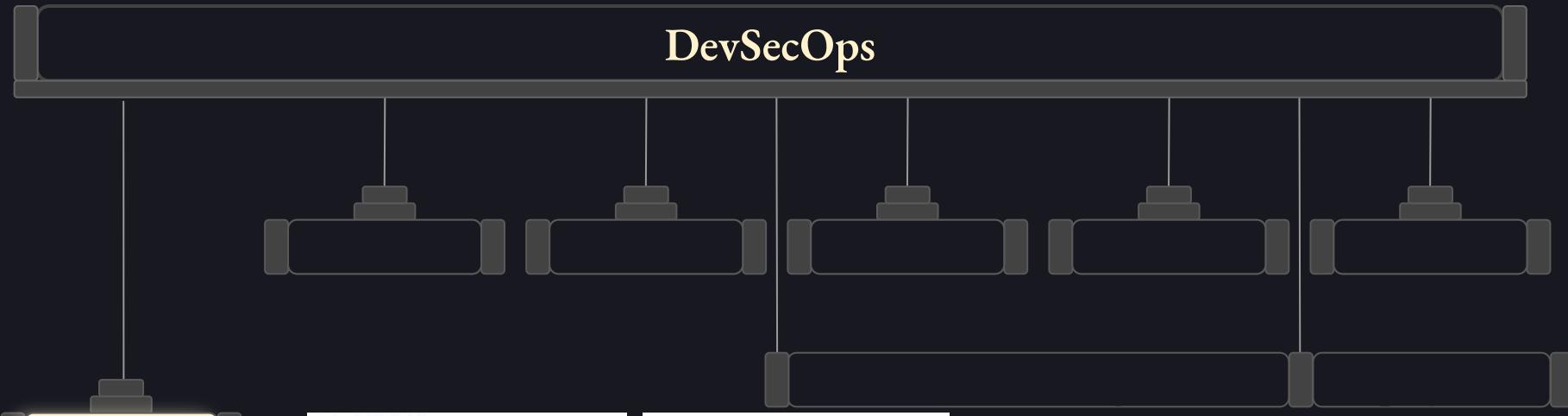


- ➡ Pipeline Minutes or Custom Runner configuration
- ➡ Test Results, Coverage, Monitoring tools, **Artifacts**,  
Package management, Container registry, Kubernetes & Serverless
- ➡ Completes the Development/operations Lifecycle
- ➡ Integrates with Open source Tools

*List of files and directories created by a job once it finishes*



## DevSecOps



## Requirements Document

**Development of Opti Admin Dashboard Module**

Description

General Details

Facilities

States and Regions

EMS Subscriptions

This template illustrates organizing the functional requirements for the product by system feature. It provides a general overview of the system's functionality and how it is organized by use case, mode of operation, user class, object class, functional hierarchy, or combinations of these, whatever makes the most logical sense for your product.

**System Features**

<This template illustrates organizing the functional requirements for the product by system feature. It provides a general overview of the system's functionality and how it is organized by use case, mode of operation, user class, object class, functional hierarchy, or combinations of these, whatever makes the most logical sense for your product.>

3.1 System Feature

<Don't forget to mention feature 1. State the feature name in just a few words>

3.1.1 Description and Priority

<Provide a short description of the feature and indicate whether it is High, Medium, or Low priority. You could also include specific priority component ratings, such as high, medium, low, etc. You could also include a relative scale from 1 to 100, where 100 is the highest priority and 1 is the lowest priority>

3.1.2 Scenario/Response Sequences

<List the sequences of user actions and system responses that stimulate the behavior of the feature. This will help the team understand how the feature will work in use cases>

3.1.3 Functional Requirements

<Identify the functional requirements associated with this feature. These are the extreme requirements that must be present in order for the user to carry out the feature. If the user cannot carry out the feature without this requirement, then the product should respond to anticipated error conditions or invalid inputs. Requirements should be listed in a numbered list and each requirement should have a placeholder to indicate when necessary information is not yet available>

<Each requirement should be uniquely identified with a sequence number or a meaningful tag and some text>

REF ID-1  
REF ID-2

3.2 System Feature 2 (and so on)

**4. External Interface Requirements**

4.1 User Interface

<Describe the logical characteristics of each interface between the software product and the users. This includes the same characteristics as the functional requirements, except that we are to follow accepted input conventions, standard buttons and functions (e.g., what we appear on every screen, keyboard shortcuts, error message display standards, and so on). Define

Software Requirements Specification for Project X

Page 3

another project, unless they are already documented elsewhere (for example, in the vision and scope document or the project plan).>

**3. System Features**

<This template illustrates organizing the functional requirements for the product by system feature. It provides a general overview of the system's functionality and how it is organized by use case, mode of operation, user class, object class, functional hierarchy, or combinations of these, whatever makes the most logical sense for your product.>

3.1 System Feature

<Don't forget to mention feature 1. State the feature name in just a few words>

3.1.1 Description and Priority

<Provide a short description of the feature and indicate whether it is High, Medium, or Low priority. You could also include specific priority component ratings, such as high, medium, low, etc. You could also include a relative scale from 1 to 100, where 100 is the highest priority and 1 is the lowest priority>

3.1.2 Scenario/Response Sequences

<List the sequences of user actions and system responses that stimulate the behavior of the feature. This will help the team understand how the feature will work in use cases>

3.1.3 Functional Requirements

<Identify the functional requirements associated with this feature. These are the extreme requirements that must be present in order for the user to carry out the feature. If the user cannot carry out the feature without this requirement, then the product should respond to anticipated error conditions or invalid inputs. Requirements should be listed in a numbered list and each requirement should have a placeholder to indicate when necessary information is not yet available>

<Each requirement should be uniquely identified with a sequence number or a meaningful tag and some text>

REF ID-1  
REF ID-2

3.2 System Feature 2 (and so on)

**4. External Interface Requirements**

4.1 User Interface

<Describe the logical characteristics of each interface between the software product and the users. This includes the same characteristics as the functional requirements, except that we are to follow accepted input conventions, standard buttons and functions (e.g., what we appear on every screen, keyboard shortcuts, error message display standards, and so on). Define

SRS



## User Stories

> Open

Mass Upload - Audit Log for file uploads with AWS

API AWS Integration Mass Upload #289 4d

PDF Report Generation - Front Page/Title Data

API Backend/Dynamic/Automation Report #290 2d

PDF Report Generation - Page 01/Single and Multi Property

API Backend/Dynamic/Automation Report #291

PDF Report Generation - Page 02/Consumption and Cost Data

API Backend/Dynamic/Automation Report #292

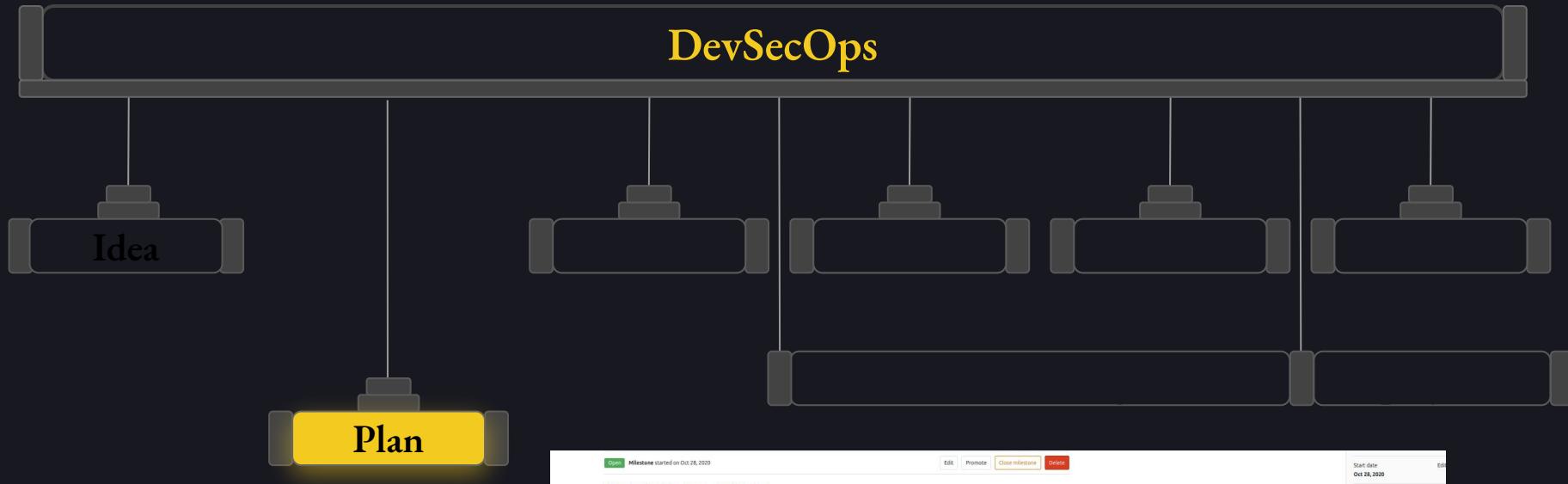
PDF Report Generation - Page 03/Chart Data

API Backend/Dynamic/Automation Report #293

Setup Celery Process and Implement mail trigger(For Production Instance)



## DevSecOps



➡ Set weight and Priority

➡ Add Milestones

➡ Time Tracking



[Open](#) | Milestone started on Oct 28, 2020

Edit Promote Close milestone Delete

### MILESTONE 06 - Water Utility Release

Water Utility related updates

(i) To display the existing electricity widgets and budget for water with "electricity consumption" & "demand" being shown as "water consumption" and the unit as gallons.

(ii) "Upload data": To have a separate "upload data" option available for water where both the consumption & demand data can be uploaded. The electricity data with available template will be uploaded and stored in the electricity table which will then be used for displaying the graphs.

(iii) Units of Measurement: Water has the following units: gallons, cubic feet, CCF, acre-foot. Show "gallons" in place of existing "KWH" and then change the data based on the conversion logic provided for these 4 units.

(iv) Secondary curves: We have secondary curves available for the following graphs: (a) Electricity Tracking Report | (b) Hourly Consumption Profile | (c) Monthly Consumption profile (which has CO2, IODD & Peak demand as secondary curves). The secondary curve temperature can be shown as temperature itself as we need that curve. However, we also need another secondary curve: precipitation (unit is inch). The temperature data can be scaled down (divided by 10 or divided by 5) and displayed as "precipitation" (inches). This is needed only for graphs (a) and (b). For (c) we already have three secondary curves. Two of these curves can be hidden and one can be shown as precipitation.

All issues for this milestone are closed. You may close this milestone now.

The tabs below will be removed in a future version

Learn more about issue boards, to keep track of issues in multiple lists, using labels, assignees, and milestones. If you're missing something from issue boards, please create an issue on GitLab's issue tracker.

Issues 14 Merge Requests 0 Participants 5 Labels 15

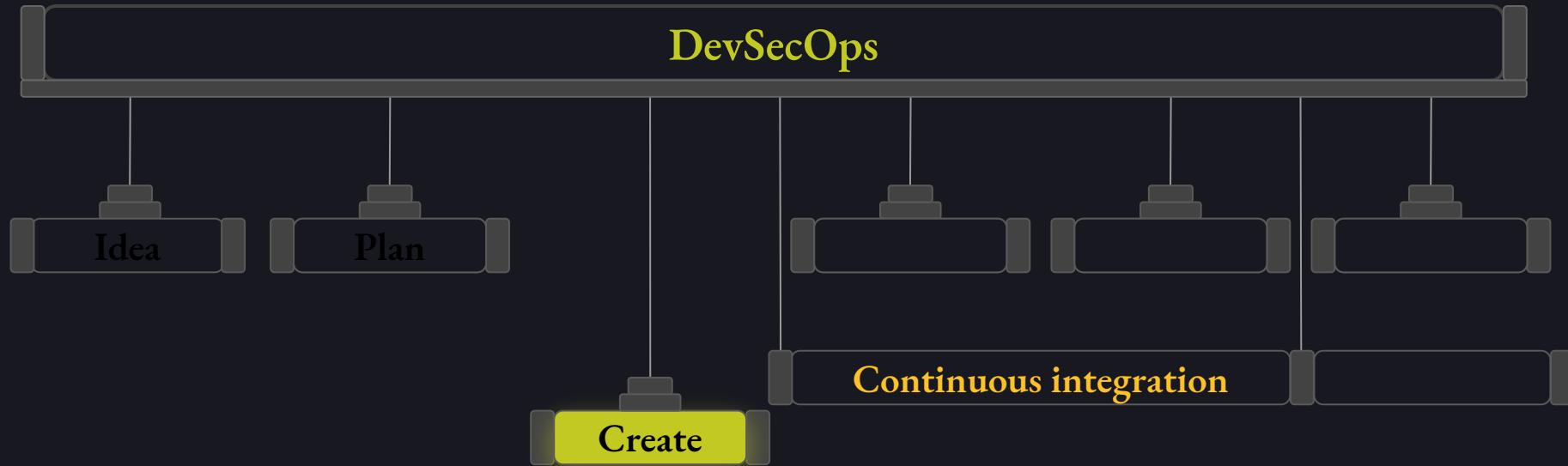
Unstarted Issues (open and unassigned) 0	Ongoing Issues (open and assigned) 0	Completed Issues (closed) 14
--	--------------------------------------	------------------------------

Defect - Water data are not removed when API is called  
#317 [Details](#) [Merge](#) [Close](#)

Issues in CI and pipelines



## DevSecOps



- Commit and tag
- Branching strategy
- Wiki

A screenshot of a GitLab issue page for a New API for Water Accounts. The page includes:

- Status**: Closed
- Opened**: 1 month ago by praveen
- Actions**: Reopen issue, New issue
- Title**: New API for Water Accounts
- Description**: A separate API to list facility water accounts
- Merge Requests**: 3 Related Merge Requests
  - !1131 New API for Water Accounts Merged
  - !1132 API for reset Water utility data, Remove water from other utility accounts Merged
  - !1162 Milestone 06-Water utility changes pushed to staging Merged
- Assignee**: praveen @praveen
- Milestone**: MILESTONE 06 - Water Utility Release
- Time tracking**: Spent 50m, Est 1h
- Due date**: No due date



## DevSecOps

Idea

Plan

Create

! Code quality degraded on 7746 points ?

Collapse

- ◆ Critical - Consider simplifying this complex logical expression.  
in [scripts/frontend/stylelint/stylelint-utils.js:15](#)
- ◆ Critical - Consider simplifying this complex logical expression.  
in [app/assets/javascripts/projects/settings/access\\_dropdown.js:248](#)
- ◆ Critical - CSRF vulnerability in OmniAuth's request phase  
in [Gemfile.lock:810](#)
- ▼ Major - Function 'buildConfig' has 7 return statements (exceeds 4 allowed).  
in [workhorse/main.go:67](#)
- ▼ Major - Function 'run' has 8 return statements (exceeds 4 allowed).  
in [workhorse/main.go:152](#)
- ▼ Major - Method 'Resizer.Inject' has 5 return statements (exceeds 4 allowed).  
in [workhorse/internal/imageresizer/image\\_resizer.go:164](#)
- ▼ Major - Method 'Resizer.tryResizeImage' has 7 return statements (exceeds 4 allowed).  
in [workhorse/internal/imageresizer/image\\_resizer.go:268](#)
- ▼ Major - Function 'unpackFileFromZip' has 7 return statements (exceeds 4 allowed).  
in [workhorse/internal/artifacts/entry.go:64](#)

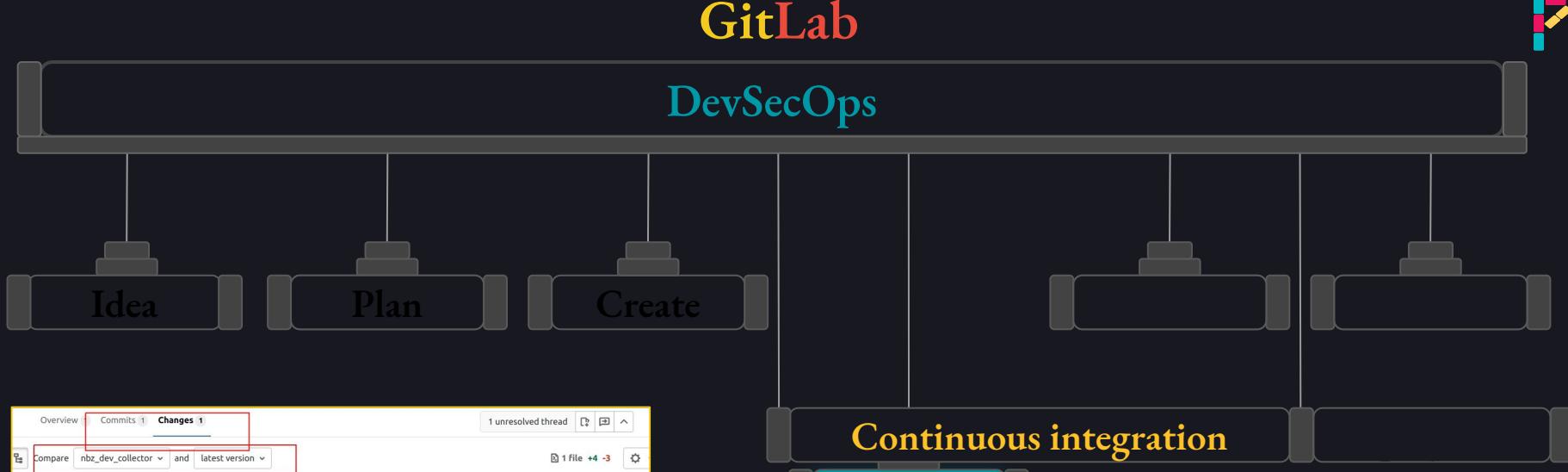
### Continuous integration

Verify

- ➡ Run jobs before MR
- ➡ Multiple grouped & parallel jobs
- ➡ Download Reports, Code Quality Widget



## DevSecOps



Code Review/Approve/Permissions



## DevSecOps

Idea

Plan

Create

Continuous integration

Verify

Code Review/Approve/Permissions

Start Discussions and Threads

praveen @praveen started a thread on the diff 32 seconds ago

main.py

```
553      # minus previous data from current entering data, which is convert into minute and  
554      # divided by difference minutes  
555      # print(current_data - board_total_active_data)  
556      processed_total_active_energy = float(current_data) - float(board_total_active_data)  
557      -  
558      +  
559      processed_total_active_energy = (  
560          float(current_data) - float(board_total_active_data)  
561      ) / float(seconds)  
562      b = float(current_data) - float(board_total_active_data)  
563      a = round((processed_total_active_energy), 2)  
564      -
```

Comment on lines 558 to -559

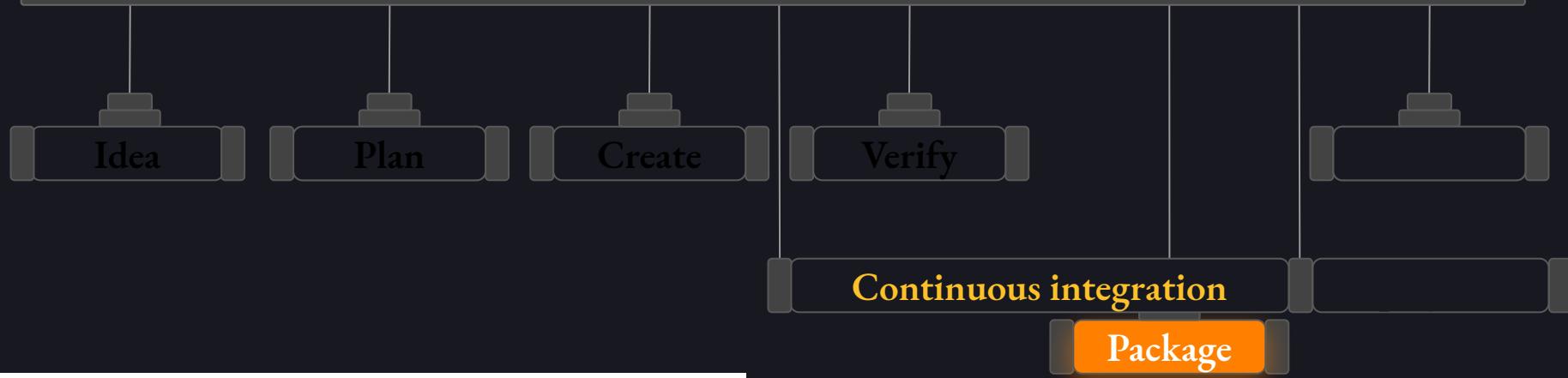
praveen @praveen · 42 seconds ago

Please follow Naming conventions

Reply... Resolve thread



## DevSecOps



### Container Registry

1 Image repository Expiration policy is disabled

With the GitLab Container Registry, every project can have its own space to store images. [More information](#)

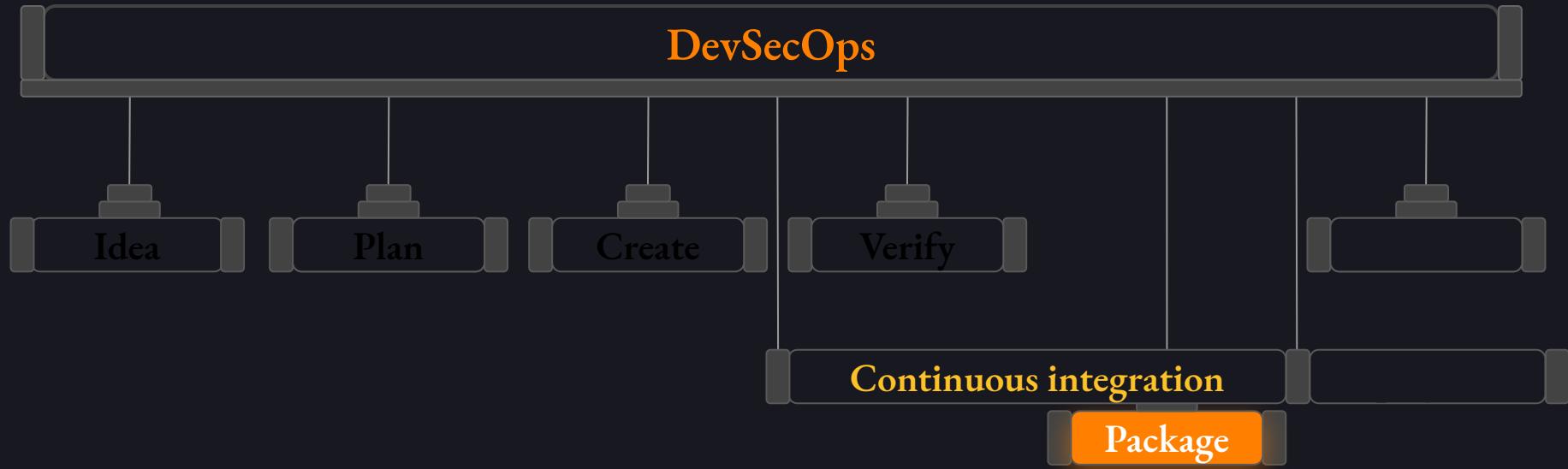
q

enterprise/rnd/fastapi/ Root image	1 Tag
------------------------------------	-------

➡ Upload & maintain Container images



## DevSecOps



### Package Registry

1 Package

Publish and share packages for a variety of common package managers. [More information](#)

Filter results

Published

NiPyPi  
0.0.1 PyPI

Manually Published  
Created just now

- ➡ Upload & maintain Container images
- ➡ Pypy, Npm, Conan, Maven



## DevSecOps



```
(1.12.1)
122 Collecting xlrd==2.0.1
123   Downloading xlrd-2.0.1-py3-none-any.whl (96 kB)
124 Requirement already satisfied: setuptools>=3.0 in ./venv/lib/python3.8/site-packages (from gunicorn==20.1.0->-r requirements.txt (line 13)) (44.0.0)
125 Installing collected packages: et-xmlfile, numpy, openpyxl, six, python-dateutil, pandas, xlrd
126 Successfully installed et-xmlfile-1.1.0 numpy-1.20.3 openpyxl-3.0.7 pandas-1.2.4 python-dateutil-2.8.1 six-1.16.0 xlrd-2.0.1
127 Running Model Migrations
128 Migrations for 'customer':
129   customer/migrations/0009_auto_20210608_1358.py
130     - Alter field category on masterdata
131     - Create constraint master_data_uniq_id on model masterdata
132
133 Running Database migrations
134 Operations to perform:
135   Apply all migrations: admin, auth, contenttypes, customer, sessions, user
136 Running migrations:
137   Applying customer.0009_auto_20210608_1358... OK
138
139 Restarting Service
140 Restarting Nginx
✓ 142 Saving cache for successful job
143 Creating cache default...
144 WARNING: ~/cache/pip/: no matching files
145 Archive is up to date!
146 Created cache
✓ 148 Cleaning up file based variables
150 Job succeeded
```

### Continuous integration

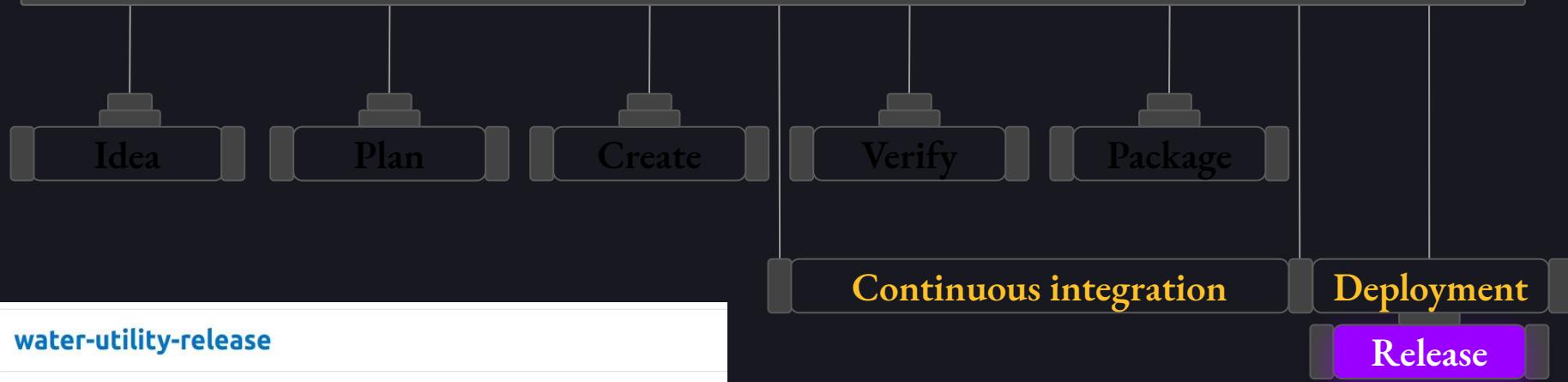
### Deployment

### Release

- ➡ Deploy to environment
- ➡ View environment and deployments
- ➡ Web Hooks and Monitoring Integrations



## DevSecOps



### water-utility-release

100% complete      Milestone      Issues 14

— MILESTONE 06 - Water Utility Release      Open: 0 • Closed: 14

▼ Assets 4

- Source code (zip)
- Source code (tar.gz)
- Source code (tar.bz2)
- Source code (tar)

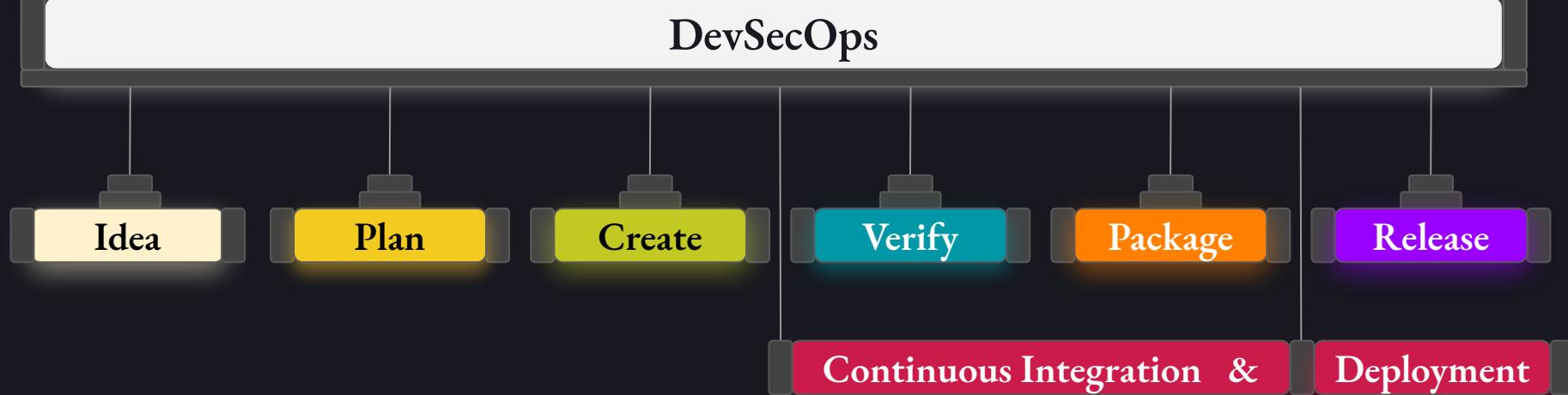
Refer MILESTONE 06 - Water Utility Release

water-utility-release      Created 5 months ago

- ➡ Checkpoint in source code
- ➡ Snapshot/build artifacts/metadata
- ➡ Associate Milestones



## DevSecOps

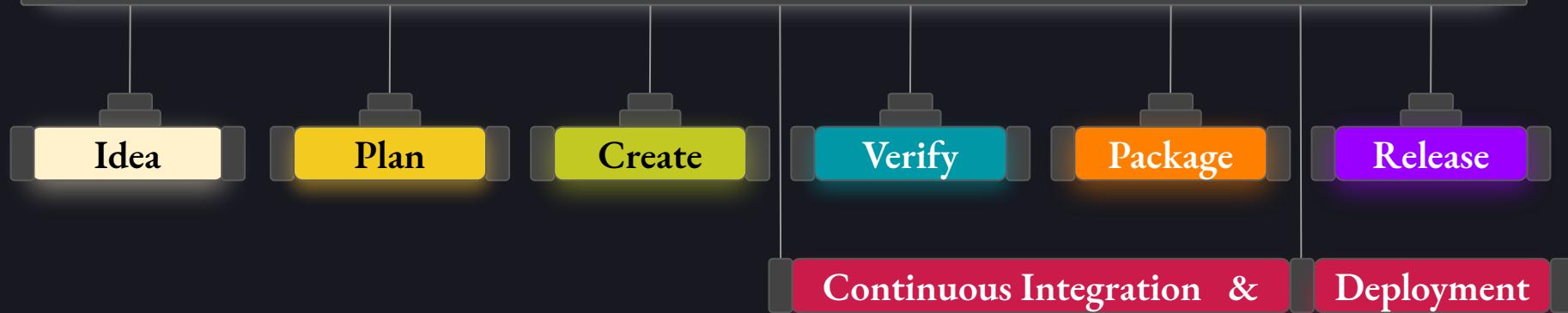


There Is A SDLC Traceability

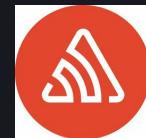




## DevSecOps

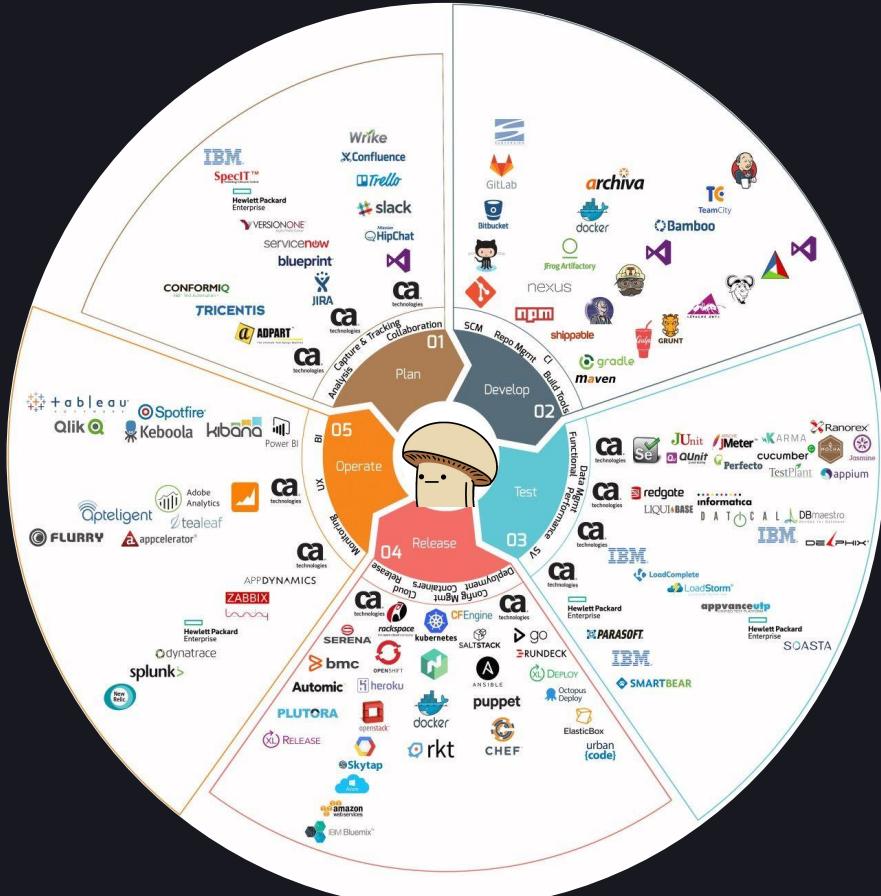


## Continuous Observability





# The ToolChain Explosion



Development

Operations

DBA

Networking

Standards & Compliance

Security Specialist

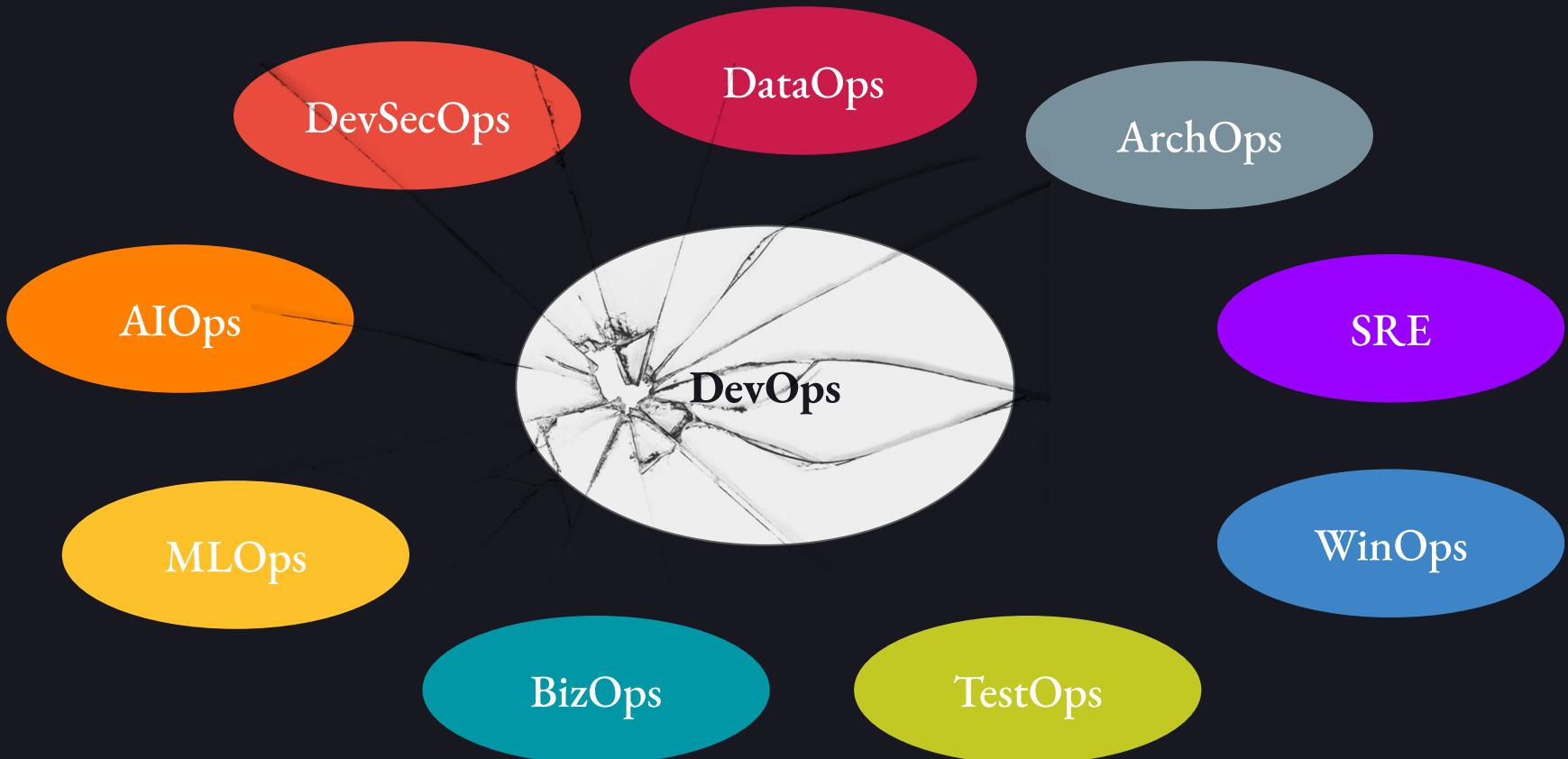
Quality Control

Containers

Cloud Technology

Business Goals

Cross Collaboration





# Automation -> DevOps -> NoOps -> AI

The collage includes the following logos:

- Docker (blue whale icon)
- Ship (blue steering wheel icon)
- Apache Kafka (white hexagon with blue grid icon)
- Apache Flink (orange hexagon with orange grid icon)
- serverless (black square with red server icon)
- AWS Lambda (orange block icon)
- CloudWatch Metrics (blue hexagon with white icon)
- CloudWatch Logs (blue circle with white icon)
- pythonanywhere (Python logo with text)
- Heroku (purple h icon)
- DigitalOcean App Platform (green gear icon)
- Cloud (blue cloud icon)
- DuploCloud (white background with a robot icon and text)
- NoOps (Red circle with a diagonal slash and the word "NoOps")
- DigitalOcean App Platform (blue background with a person in a boat icon and text)

Text elements:

- What suits us?
- Where we are?



## Closing Notes

# DevOps Survey Results 2021

*Only the Key Takeaways*

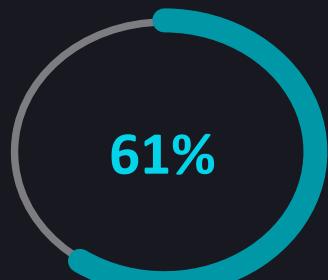


## Deliverables

**61%** say it helped them produce high quality deliverables

**49%** say they see a faster time to market

**49%** say it improved their deployment frequency



Produce high quality deliverables

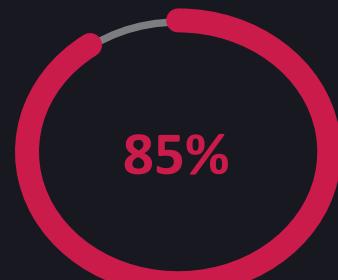


## Barriers

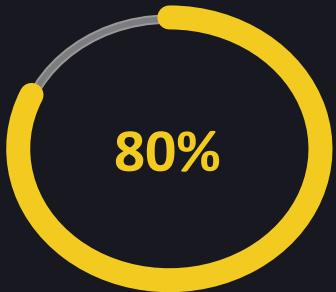
**37%** say lack of skills in employees

**36%** say legacy infrastructure

**35%** Culture and Rigid process

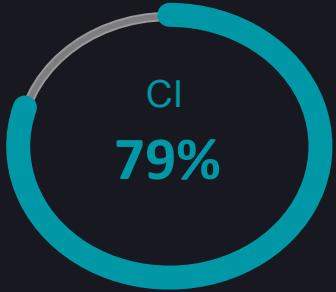


Some form of obstacles



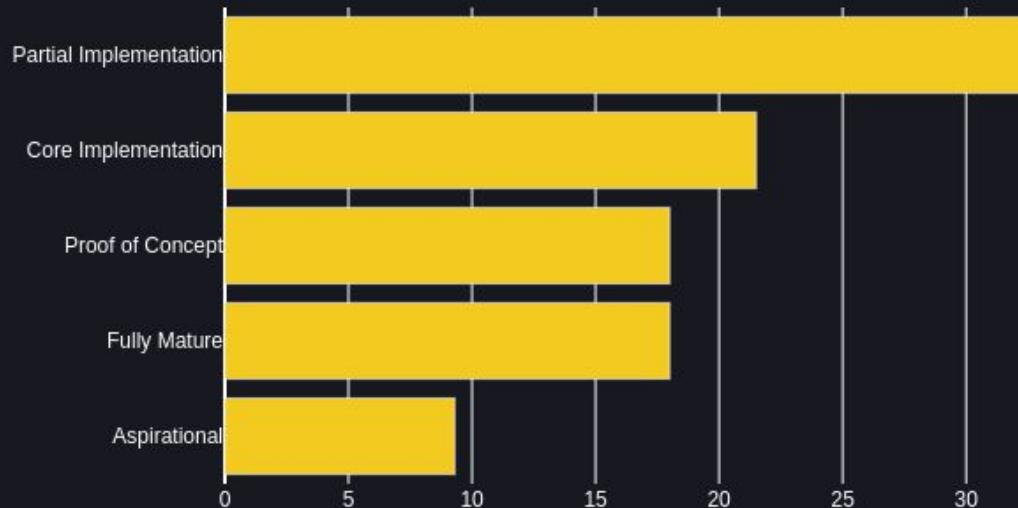
80%

Source code Management and code development are the most widely spread toolchains used



CI  
79%

Best describe your teams CI/CD practices



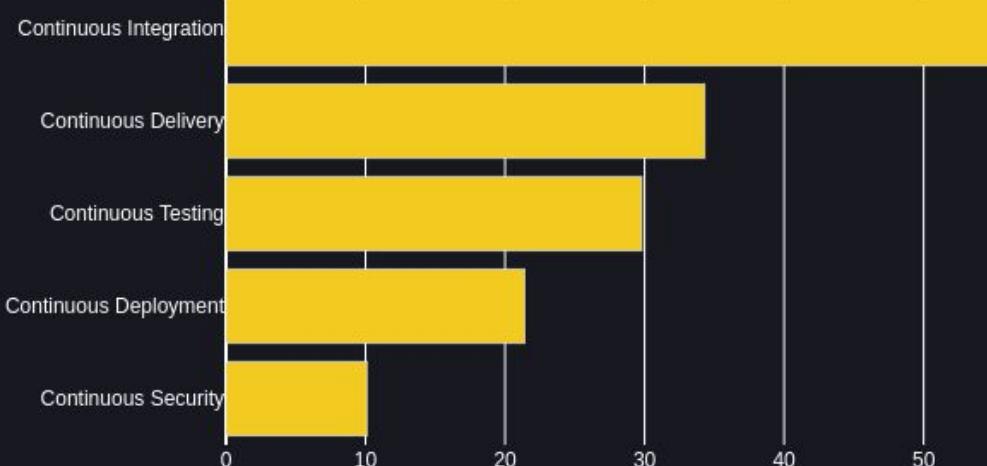
Delivery  
76%

80%

Source code Management and code development are the most widely spread toolchains used

CI  
79%

Best practices implemented



Delivery  
76%

Soft Skills & Collaboration



# What has been Done So Far

It All started with enman...

7

## Milestones

### MILESTONE 07 - Energy Usage Breakdown

started on Nov 18, 2020

12 Issues · 35 Merge Requests

16% complete

↑ [Close Milestone](#)

### MILESTONE 06 - Water Utility Release

started on Oct 28, 2020

Release [water-utility-release](#)

14 Issues · 50 Merge Requests

100% complete

↑ [Close Milestone](#)

### MILESTONE 06 - On hold (PDF Report and Mass Upload)

5 Issues · 1 Merge Request

0% complete

↑ [Close Milestone](#)

### BACKLOGS

11 Issues · 41 Merge Requests

90% complete

↑ [Close Milestone](#)

### MILESTONE 05

10 Issues · 38 Merge Requests

80% complete

↑ [Close Milestone](#)



# What has been Done So Far

7

Milestones

334

Issues

The board displays the following tasks across the columns:

- Open:** R&D - Extract Data generated (R&D, #270)
- To Do:** R&D - Extract Data generated (R&D, #270), Mass Upload - Upload (API, File-Upload/Download, #285)
- Active/In Progress:** PDF Report Generation (API, Backend/Dynamic/A, #293), Mass Upload - Create sheets programmatically (Backend/Dynamic/Automation, #287)
- Done:** Migrate Dev and Staging digital ocean server (Configuration, Deploy/De, #333)
- Deploy/Development:** Dashboard API(All widget response in dashboard re (Optimization, R&D, Wid, #329)
- Deploy/Staging:** Bug - Deleted wid dashboard(get\_all) (Bug, High Priority, #302)
- Deploy/Production:** Setup SSL for Production (Backend/Dynamic/Automation, #331)

**Sidebar Milestones:**

- R&D - Extract Data generated (R&D, #270)
- Periodic (Backend/Dynamic, #277)
- Mass Upload - Create sheets programmatically (Mass Upload, #284)
- Mass Upload - Create sheets programmatically (Mass Upload, #285)



# What has been Done So Far

7

Milestones

334

Issues

1844

Pipelines

880

Dev Deployments

240

UAT Deployments

## Trigger Mail - Update redirection link to dynamically send link based on environment

Related to send mail on user creation function

To upload designs, you'll need to enable LFS and have an admin enable hashed storage. [More information](#)

Linked issues 0 +

### Related merge requests 4

Admin dashboard !1313



E-Mail use TLS !1314



Dev ~ RFC email domain name issue exception handle !1315



Staging ~ RFC email domain name issue exception handle. Update Email Text dynamically based on ENV !1316

BACKLOGS



load\_test.py

+0 -20



Enman</a> site.</p>



Show 20 lines Show all unchanged lines

praveen  
@praveen

Milestone  
**BACKLOGS**

Time tracking

Spent 30m

dit

Edit

Due date

None

dit

Labels

API Changes

Backend/Dynamic/Automation

Deploy/Development

User module

dit

Confidentiality

Not confidential

dit

Edit

## Related Pipelines

## Related Commits

## Related Changes

## Related Issues

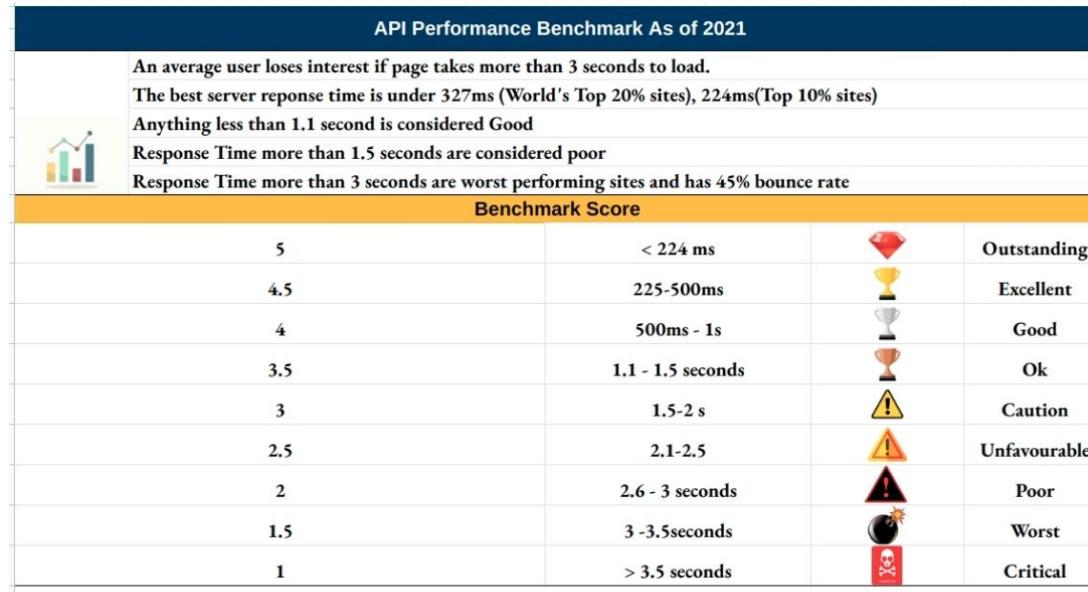
Entire Project Lifecycle Traceability. From An Idea To Value to Delivery



# What has been Done So Far

2	1.0 Second	"Maximum acceptable limit. Though user won't feel interrupted, but may experience some delay"
3	>3 second	"Time when user starts to notice visible delays. This should be greatly considered on how the response affects to mobile users or users with low speed network. Bounce rate is 45%"
4	>5 second	"Vast delays and site become unusable for most users. The site's bounce rate will increase to an average 70%"

## Benchmark Stats based on World Web Technology Survey



## The Power of Wiki

- ➡ Home Page - TOC
- ➡ DB Schema Designs and .SQL
- ➡ API Endpoints
- ➡ Case Studies
- ➡ Benchmarking APIs

# What has been Done So Far



[Open](#)  Periodic PDF Report Generation

## Type 02 - Rendering all graphs in single page



Edited 7 months ago by praveen

# The Power of Wiki

- CAN
  - MicroScheduler GraphQL
  - Docker Stack
  - PDF Designs



# What has been Done So Far

## Design Management



The screenshot shows a GitHub issue comment in a "Sample report format" window. The comment is from user "praveen @praveen" just now, stating: "Can the positions of Pie and bar charts be swapped?". Below the comment is a "Reply..." text input field.

→ CAN

→ MicroScheduler GraphQL

→ Docker Stack

→ PDF Designs

→ Manage Designs

→ Discussions



# What has been Done So Far

→ Pipelines Configured for Microscheduler, Smart-meter, CAN, Portal, 5G, MLOps. Started for Frontend

→ OpenAPI Swagger Spec

→ Private PIP Packages

→ Container Registry

The screenshot shows a project overview for a 'fastapi' project. On the left, there's a sidebar with options like Project overview, Repository, Issues, Merge Requests, CI/CD, Security & Compliance, Operations, Packages & Registries (which is currently selected), Package Registry, Container Registry (also selected), Analytics, Wiki, Snippets, and Members. The main area displays a 'Root image' entry under 'Container Registry'. It shows 'Root image' with '1 tag', 'Cleanup disabled', and 'Last updated 1 month ago'. Below it, there's an 'Image tags' section with a 'latest' tag (606.67 MB). A note says it was published to the repository at 09:03 GMT+0530 on 2021-04-10. Manifest and configuration digests are also listed.

Published	Digest
1 month ago	Digest: 1158951

Manifest digest:	sha256:1158951500666388f92f016b18921969fffb87512a645afba8975efef5da0d282
Configuration digest:	sha256:265321e229e6912a2293f122ec2bcce737dd4cb43a3ab35952cb6d249e031c98

# What has been Done So Far

- ▶ Prometheus-Grafana Observability Platform
- ▶ Environment and Monitoring Config

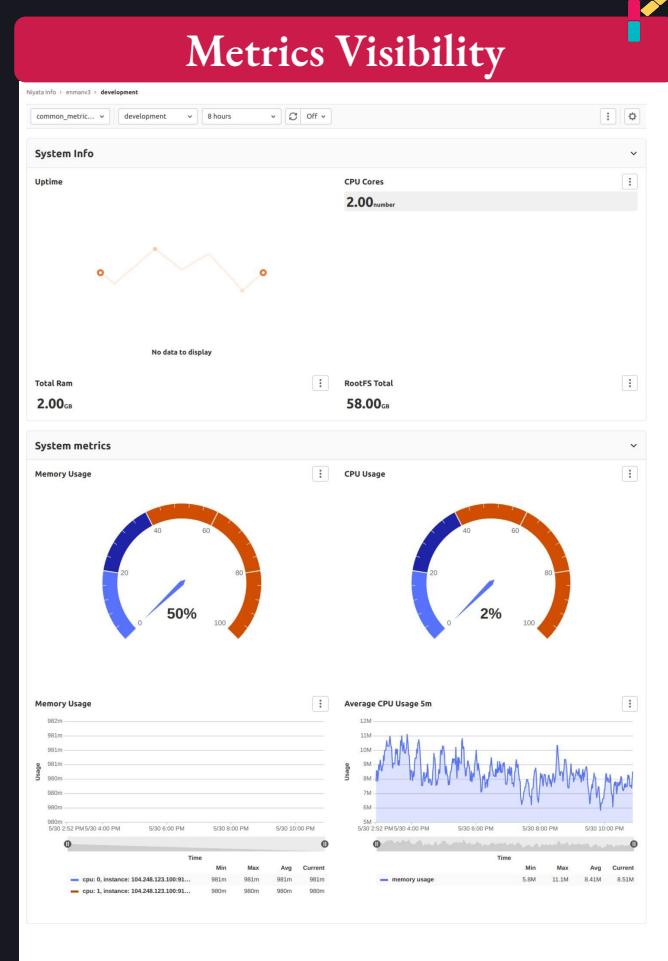
The screenshot shows a dashboard titled "Environments" with three listed environments: "staging", "development", and "production".

- staging:** Deployment #1059 by Deploy\_Staging #4093. Commit `staging -o ee63d6ad`. Updated 2 months ago. Merge branch 'develop'... .
- development:** Deployment #1058 by Deploy\_Dev #4058. Commit `development -o 1c2c60e8`. Updated 3 months ago. Merge branch 'admin-...' .
- production:** Deployment #1054 by Deploy\_Production #4... . Commit `master -o 208d61eb`. Updated 3 months ago. Merge branch 'staging'... .

On the right side of the dashboard, there are four small icons in a grid:

- A blue square with a white gear icon.
- A light blue square with a white list icon.
- A red square with a white circle icon.
- A red square with a white square icon.

A black button labeled "Open live environment" is positioned below these icons.



# What has been Done So Far

## Sentry Integration

- Sentry Error tracking
- Performance Monitoring
- Project Related Errors
- Create Issues from Errors

The screenshot displays the Sentry interface across three main sections:

- Left Panel (Project Overview):** Shows "Empower Plant" and "Jane Schmidt" as the user. Navigation links include Projects, Issues, Discover, Performance, Alerts, Releases, and User Feedback.
- Middle Panel (Performance Monitoring):** A "Performance" dashboard with "My Projects" dropdown, "All Environments" dropdown, and a time range of "Last 14 days". It features two charts: "Apdex" (ranging from 0.7 to 1.0) and "Transactions Per Minute" (ranging from 35k to 50k). Buttons for "By Transaction" and "By Trend" are at the top right, and "Display: All Transactions" is selected.
- Right Panel (Error Details):** A detailed view of an error. At the top, it says "enterprise > ... > enterprise > ... > [REDACTED] errors > Error Details". It shows the error was reported 3 days ago by [REDACTED]. There are buttons for "Ignore", "Resolve", and a red-bordered "Create issue".
  - Report Summary:** Level: error. Sentry event: <https://sentry.io/.../2163464048>, Events: 15, Users: 1.
  - Stack Trace:** A code snippet from `django/db/backends/utils.py` at line 84:

```
    self.db.validate_no_broken_transaction()
    with self.db.wrap_database_errors:
        if params is None:
            return self.cursor.execute(sql)
        else:
            return self.cursor.execute(sql, params)

def _executemany(self, sql, param_list, *ignored_wrapper_args):
    self.db.validate_no_broken_transaction()
    with self.db.wrap_database_errors:
        return self.cursor.executemany(sql, param_list)
```

# What has been Done So Far

ChatOps



- ➡ Webhooks Integration
- ➡ Python - MSTeams
- ➡ Smart-meter stats
- ➡ Smart-meter critical alerts

**Board data Alert**  
Smart-meter bot that notifies data insertion rate on a defined interval

Board activity  
 No change in board arrival rate.

Data activity  
 Data insertion rate is increased by 9.43

Incoming Board data stats for the last 10 minutes  
 Total boards currently inserting data: 30  
Total Records inserted: 232

	name	total_records
0	NI004	8
1	NI005	8



## Next Steps...

- Consistent SDLC in Place
- Continuous Delivery & Continuous Deployment
- Better Code Review/ Peer Review
- Design Management
- Better Release Management
- Improve the Centralized Observability Stack
- Kubernetes/Serverless/Microservices



*Next Session On  
Grafana...*



*Breathe In...*

*Breathe Out...*

# Its Q&A

**Thank You**