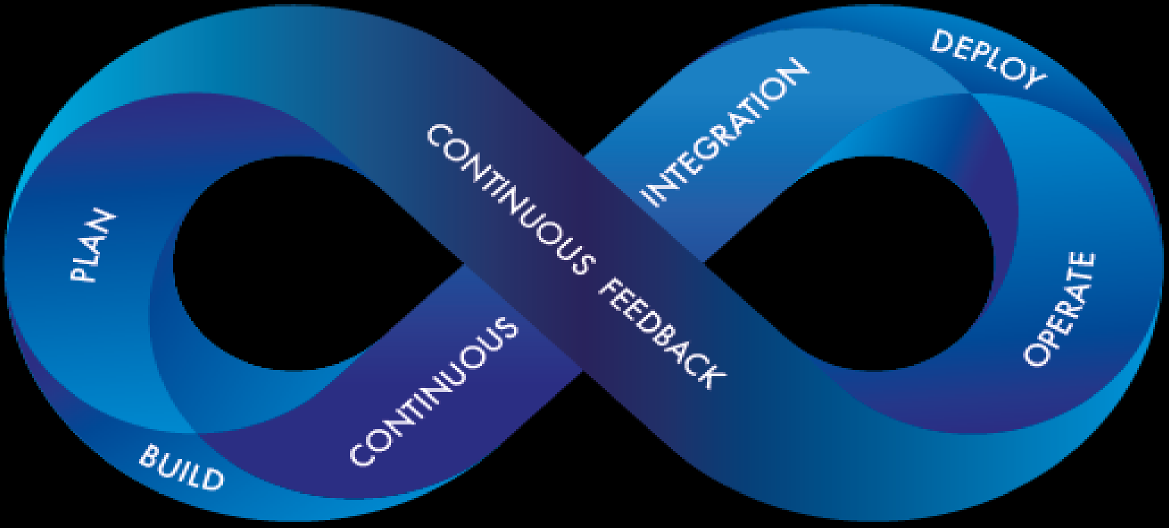
BDAYA
DEVELOPMENT



# DevOps Engineering

## Agile practices within

(Microservices, Mobile, Web, IOT, Data Science, BlockChain)

## Question Answers Ch1

Ahmed Khalil
Ahmed Fwela

Version 0.0.1

# Questions

## 1- What is DevOps?

DevOps is shorten for Development-Operations.

DevOps is a set of practices that merge tow concepts software development and operations, It aims to shorten the systems development life cycle and provide continuous delivery with high software quality.

## 2- What is the features that DevOps pushed to the market?

- Accelerating time to market
- Adapting to the market and competition
- Maintaining system stability and reliability
- Improving the mean time to recovery

## 3- The number tells the story, describe the numbers that puppet collected for the high performance of DevOps?

- 200 X more frequent deploys
- 24 X faster recovery times
- 3 X Lower change failure rates

## 4- At the planning phase of DevOps, describe how it can be agile-driven?

it's agile-driven in planning it shares the agile practices of Creating backlogs, tracking bugs, managing agile software development with Scrum, using Kanban boards and visualizing progress with dashboards are some of the ways DevOps teams plan with agility and visibility.

## 5- Mention Briefly how can you practice DevOps in Development phase?

Working for this phase include everything about developing, testing, writing, reviewing and the integration of code between teams, it's relative more about using git and other VCSs to fork, merge, commit, pull, push, for code and project components.

## 6- This phase is like to deliver orders to customers in their homes, How can that?

the deliver phase is about sending the tested code as running app in production, and it ready to consume from users. Deploy to production environment means that app is stable enough and reliable to consume from customers.

## 7- What is the mission of Operate phase ?

To make the production more stable, resilience. The Operate phase work for maintaining, monitoring, troubleshooting application in production environment. This phase aims to zero downtime while reinforcing security and governance.

## 8- The DevOps is not a man job, describe.

the environment is shared, the team is align so take the ownership and become involved in additional lifecycle phases – not just the ones central to their roles, for example the developers become accountable in all phases not only the plan and develop but also the deploy and operate because the performance and stability is related to the code and the entire production environment.

| **9- The healthy DevOps environment come from _________.** | | | |
|---|---|---|---|
| a- Isolation | b- speed | c- Agile process | d- visibility |

Answer d.

<table>
<tr><td colspan="4"><b>10-__________ is a software development practice in which developer merge code changes frequently into the main code branch.</b></td></tr>
<tr><td>a- CI</td><td>b- CD</td><td>c- IAC</td><td>d- VC</td></tr>
</table>

Answer a.

<table>
<tr><td colspan="4"><b>11-_____________ is the frequent, automated deployment of application into production environment, this accelerate the deployment strategies and reduce issues.</b></td></tr>
<tr><td>a- CI</td><td>b- CD</td><td>c- IAC</td><td>d- VC</td></tr>
</table>

Answer b.

<table>
<tr><td colspan="4"><b>12-_________ is a practice to manage the code into series of versions, as a practice, tracking revisions and change history to make code easy to review and recover.</b></td></tr>
<tr><td>a- CI</td><td>b- CD</td><td>c- IAC</td><td>d- VC</td></tr>
</table>

Answer d.

<table>
<tr><td colspan="4"><b>13-_________ defines system resources and topologies in a descriptive manner that allows the team to manage these resources as they code.</b></td></tr>
<tr><td>a- CI</td><td>b- CD</td><td>c- IAC</td><td>d- VC</td></tr>
</table>

Answer c.

| **14-________ is a practice usually implemented using GIT** | | | |
|---|---|---|---|
| a- CI | b- CD | c- IAC | d- VC |

Answer d.

| **15-________ means to write just little files to manage the infrastructure within your project.** | | | |
|---|---|---|---|
| a- CI | b- CD | c- IAC | d- VC |

Answer c.

## 16- What is the Configuration Management?

The CM concept is working to find a stable way and tools to manage resources state, including servers, virtual machines and databases, using Configuration Management help the teams to roll out the changes in the environment once it happened.

## 17 – What is Continuous Monitoring?

Continuous Monitoring means to have the real time visibility into the performance and health of the application stack or pool, that delivered to users from the infrastructure running to serve the application to the higher level software component.

## 18 – What is the shared point between Microservices and DevOps?

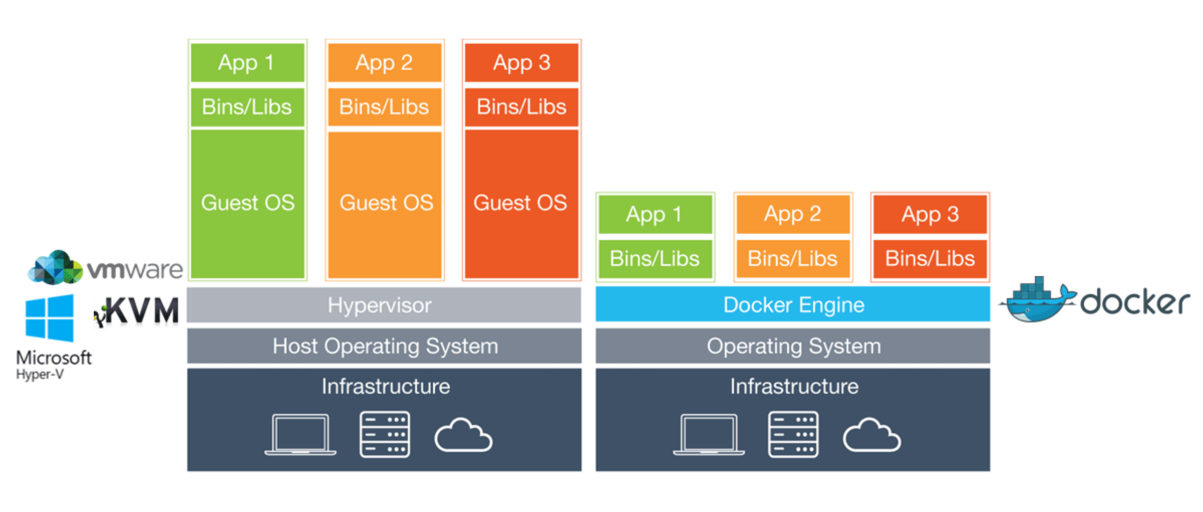
DevOps practices reinforce the idea of breaking large problems into smaller pieces and tackling them one at a time as a team.

Microservices fits perfectly into the DevOps ideals of utilizing small teams to create functional changes to the enterprise's services one step at a time. Microservices empower the implementation of small teams collaborating together in an environment of increased developer freedom.

## 19 – What is the difference between Virtualization and Containerization?

**Virtualization**: is to have many virtual machines running on the same infrastructure using VM tool like VMware or KVM or Microsoft Hyper-V, every VM now has its Operating System and the libraries for each app to run it and the app itself.

**Containerization**: is to has a Containerization support app like docker or AWS Fargate or others, to run its software then it can cluster all developer apps and libraries without using any additional overheads to install so the deployment can be automated easily, fast shippable,  and more.



## 20 – How to Practice CI/CD on IOT?

CI ensures that the IoT applications that are a part of DevOps, and perhaps some that are not, are all considered a holistic application or system. These component parts are dependent upon one another, even components that are not under the direct control, such as sensors embedded in machines (e.g., a jet engine). This is a bit of a leap from traditional approaches to DevOps, where all components are under direct control.

CD is the most important component of DevOps and IoT. This is due to the fact that the application needs to live on a platform, say a IaaS cloud provider, as well as work and play with remote devices. This complicates the processes of deployment, in that there could be updates to the application that reside on a centralized cloud platform, and perhaps even firmware updates that reside on a remote sensor or device.

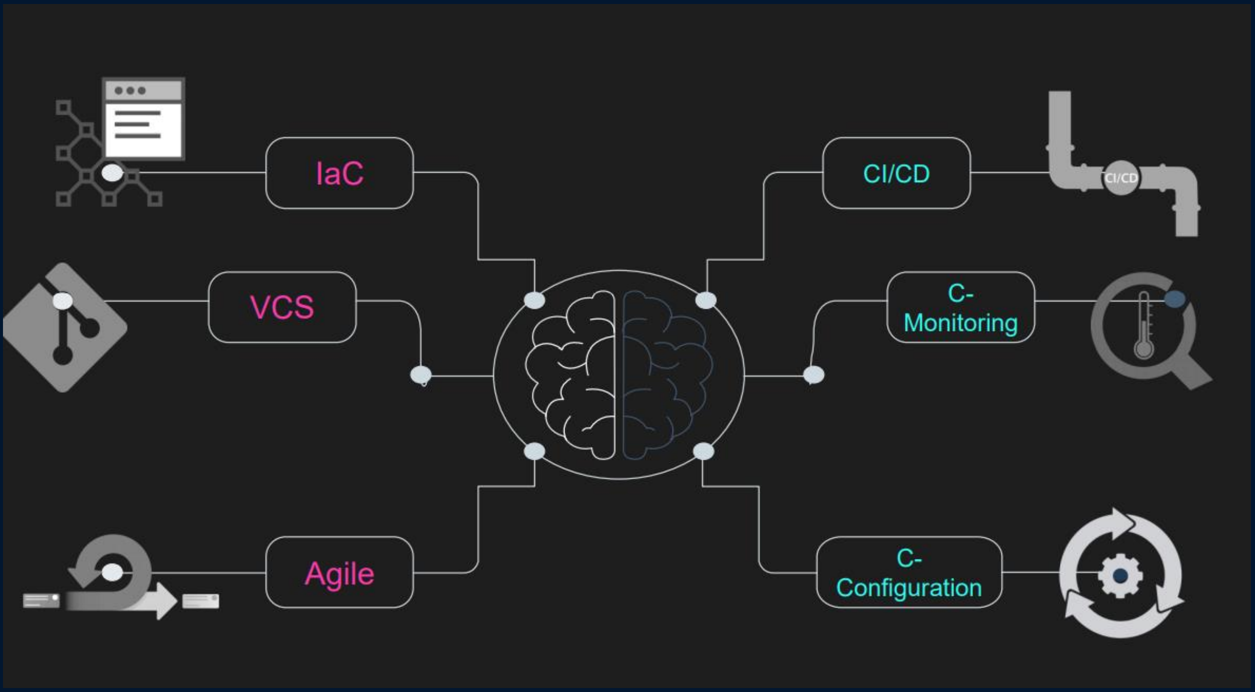
## 21- List some applications working as DevOps environment?

- Azure DevOps
- GitHub
- GitLab
- Gera Bit Bucket
- Beanstalck

## 22 – List the advantages of shortening life cycle of SDLC?

- these cycles make it easier to risk management and planning.
- Allows organizations to adapt and react to evolving customer needs and competitive pressure.

## 23 – List the DevOps Practices?



## 24 – What is the Version Control?

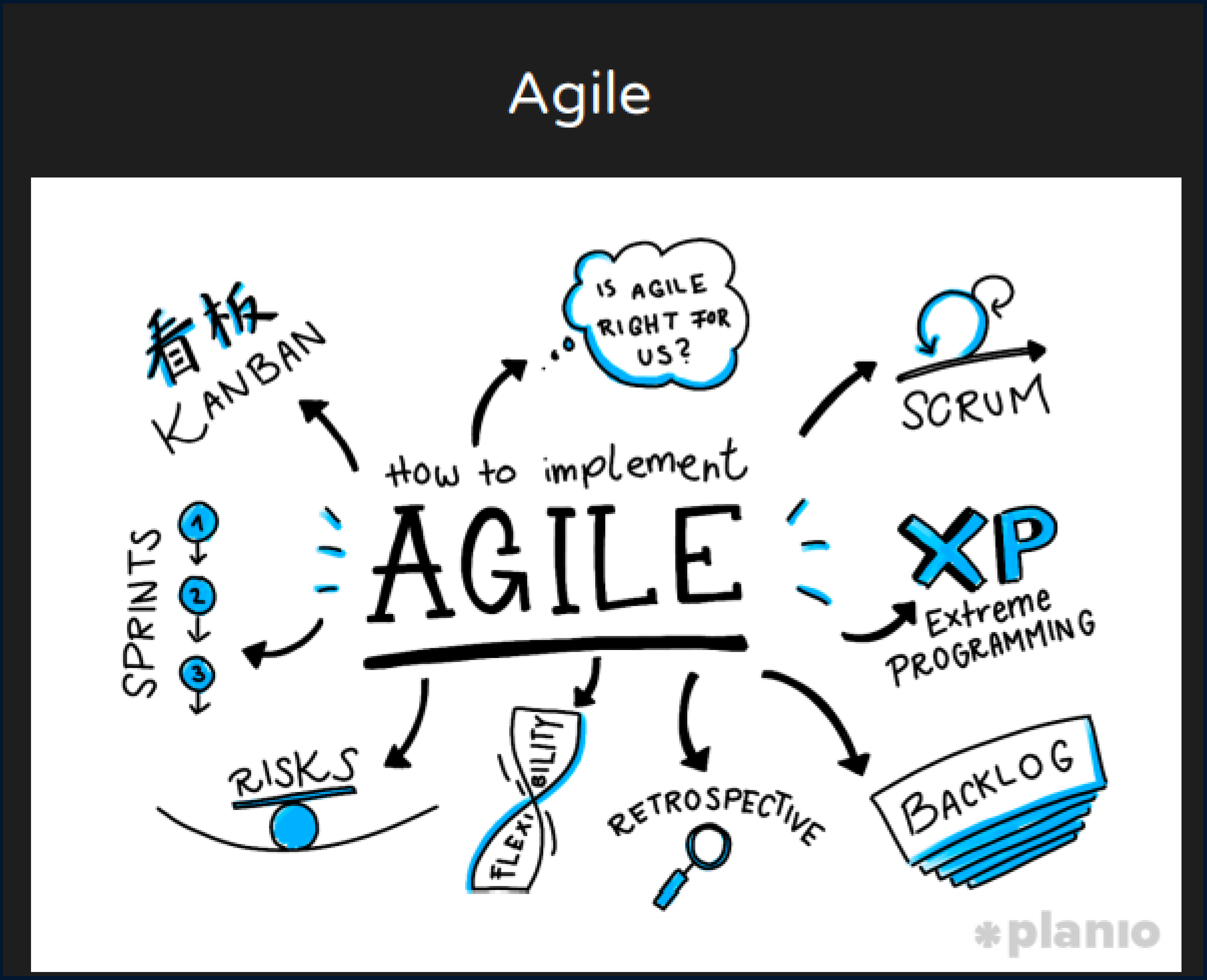
Version control is a practice to manage the code into series of versions, as a practice, tracking revisions and change history to make code easy to review and recover. Like any system the user consume, it have many version of release and each number has its own features.

## 25 – List Some of Version Control Systems?

- GIT
- Mercurial
- Microsoft TFSV

## 26 – What is Agile?

Agile is a modern approach for software development that emphasizes team collaboration, customer and user continuous feedback, and high adaptability to change through short release cycles. The feedback from users is the main change director adjusted from the (need and want) of business today engines.



Agile

## 26 – List some of agile frameworks ?

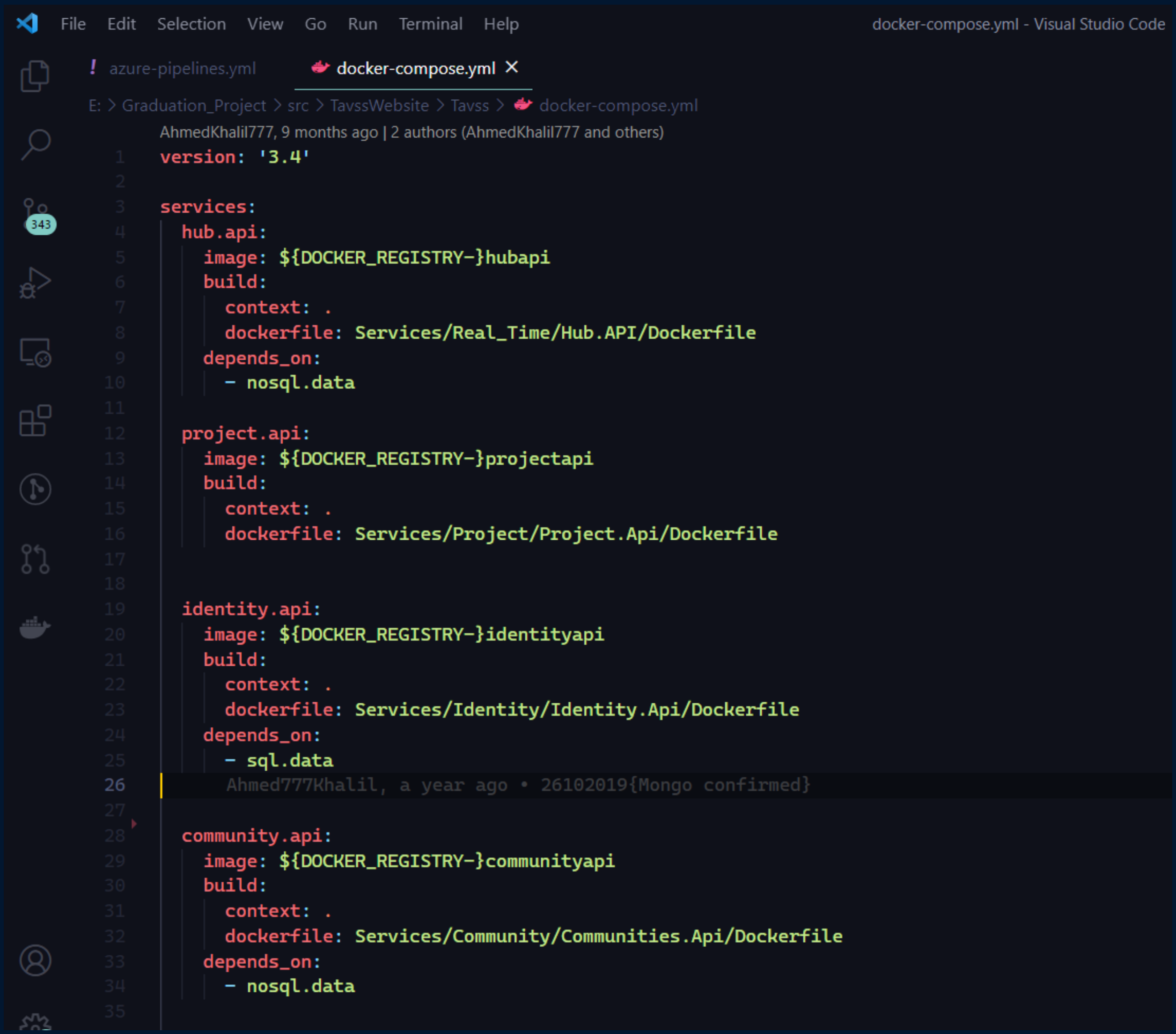
- SCRUM
- Kanban
- XP (Extreme Programming)
- FDDI
- Manifesto

## 26 – What is the mean of Infrastructure as Code?

IaC defines system resources and topologies in a descriptive manner that allows the team to manage these resources as they code.

That mean to write just little files to manage the infrastructure within your project.

Ex: docker Compose files and Kubernetes


File
Edit
Selection
View
Go
Run
Terminal
Help
docker-compose.yml - Visual Studio Code
azure-pipelines.yml
docker-compose.yml
E: > Graduation_Project > src > TavssWebsite > Tavss > docker-compose.yml
AhmedKhalil777, 9 months ago | 2 authors (AhmedKhalil777 and others)
343
Ahmed777Khalil, a year ago • 26102019{Mongo confirmed}

```yaml
version: '3.4'

services:
  hub.api:
    image: ${DOCKER_REGISTRY-}hubapi
    build:
      context: .
      dockerfile: Services/Real_Time/Hub.API/Dockerfile
    depends_on:
      - nosql.data

  project.api:
    image: ${DOCKER_REGISTRY-}projectapi
    build:
      context: .
      dockerfile: Services/Project/Project.Api/Dockerfile


  identity.api:
    image: ${DOCKER_REGISTRY-}identityapi
    build:
      context: .
      dockerfile: Services/Identity/Identity.Api/Dockerfile
    depends_on:
      - sql.data


  community.api:
    image: ${DOCKER_REGISTRY-}communityapi
    build:
      context: .
      dockerfile: Services/Community/Communities.Api/Dockerfile
    depends_on:
      - nosql.data

```