Exercise: DevOps Planning & Research

### It’s better not to use waterfall model because if you don’t finish one phase you can’t continue and you lose more time on the project. That is the reason why is better to use microservices who breaks it down into a collection of smaller independent units that have separate environment instead of using monolith which is a is a single unified unit. As first step is considered to be planning, than code, build, test, deploy and monitor application performance.

You should show the DevOps initiative and develop your DevOps strategy. Also it’s good to define roadmap’s objective and to share the roadmap with your engineering and operations teams and to keep the roadmap up to date at all times

You have to set focused, short-term goals and plans because it is easier to fix them if error occurred. It’s important to prioritize customer satisfaction, to use Agile methodologies, select an application automation toolset that is compatible with our IT environment.

1. DevOps is a culture that allows the development and the operations team to work together. DevOps requires collaboration, transparency. This results in continuous development, testing, integration, deployment, and monitoring of the software throughout the lifecycle. That’s why is faster and easier to work in a team and to help each other and to use all the steps from above.
2. First I will set my priorities and plan well, plan my day according to my most productive times and assign a fixed time during the day to deal with interruptions such as emails, phone calls. It is good to set reminders for all your task. Research plan helps define your focus, method, and goals. I alwaysdo the task that fills me with the most anxiety first, because after that the rest is easier.
3. Tools like Mural and Miro empower the entire software team to gather ideas and conduct research. [Jira Product Discovery](https://www.atlassian.com/software/jira/product-discovery) organizes this information into actionable inputs and prioritizes actions for development teams. Jira provides sprint planning, issue tracking, and allow collaboration. That I will use for planning phase. For building- Kubernetes and Docker. [Infrastructure as code](https://www.atlassian.com/continuous-delivery/principles/infrastructure-as-code?utm_campaign=service-desk_devops16-blog) means re-provisioning is faster than repairing and more consistent and reproducible. Here we can use Puppet, Docker or Terraform. To have a source control and collaborative coding the best for using are Git/Bitbucket, GitHub. These tools help store the code in different chains so you can see every change and collaborate more easily by sharing those changes. Continuous integration is the practice of checking in code to a shared repository several times a day, and testing it each time. Tools that automatically apply your tests to development branches for example Jenkins (tool for monitoring execution of repeated tasks), AWS, Circleci. Test automation can increase software quality and reduce risk by doing it early and often. Here we can use Selenium, HRay, mabl. For deployment is good to find something that gives you full visibility on branches, builds, pull requests in one place like Jira Software.

### 

### DevOps requires cultural changes, automated processes, and investments in the right technologies, improve communication and collaboration among development and operations teams, in order to increase the speed and quality of software deployment.

Agile teams focus on delivering work in smaller increments, instead of waiting for a single massive release date.

1. It is important to follow the steps and to work as a team in order to deliver fast, secure solution to the client.  Team members rely on each other as they all work together towards a common goal. Banking institutions are racing to introduce software solutions into their business processes. Customers these days are unwilling to wait in line at banks to pay their bills or make transactions to other accounts.