**Development operations**

**git / what is git**

* version control system for tracking changes in computer files from small to very large projects with speed and efficiency.
* Distributed Version Control System
* cordinates work between multiple developers
* who made what changes and when
* revert back any time
* local & remote apps

**concepts of git**

* keep tracks of code history
* takes snapshots
* you decide when to take the snapshot = commit
* you can visit any snapshots any time
* you can stage files before commiting / add /

So, Git is a version control system, but what does that mean? When developers create something (an app, for example), they make constant changes to the code, releasing new versions up to and after the first official (non-beta) release.

Version control systems keep these revisions straight, storing the modifications in a central repository. This allows developers to easily collaborate, as they can download a new version of the software, make changes, and upload the newest revision. Every developer can see these new changes, download them, and contribute. Similarly, people who have nothing to do with the development of a project can still download the files and use them.

**GitHub** is a web-based [hosting service](https://en.wikipedia.org/wiki/Internet_hosting_service) for [version control](https://en.wikipedia.org/wiki/Version_control) using [Git](https://en.wikipedia.org/wiki/Git).

[**https://www.youtube.com/watch?v=SWYqp7iY\_Tc**](https://www.youtube.com/watch?v=SWYqp7iY_Tc)

[**https://www.youtube.com/watch?v=Y9XZQO1n\_7c**](https://www.youtube.com/watch?v=Y9XZQO1n_7c)

**Hosting**

Web hosting is a service that allows organizations and individuals to post a website or web page onto the Internet. A web host, or web hosting service provider, is a business that provides the technologies and services needed for the website or webpage to be viewed in the Internet. Websites are hosted, or stored, on special computers called servers. When Internet users want to view your website, all they need to do is type your website address or domain into their browser. Their computer will then connect to your server and your webpages will be delivered to them through the browser.

https://www.youtube.com/watch?v=0hGK7qiQ6WA

**Deploy**

Software deployment is all of the activities that make a [software system](https://en.wikipedia.org/wiki/Software_system) available for use. Software deployment includes all the process required for preparing a software application to run and operate in a specific environment. It involves installation, configuration, testing and making changes to optimize the performance of the software. It can either be carried out manually or through automated systems. The following are the big advantages of Software deployment: For instance, in the world of web development, deployment does not mean being released to the public. Rather, it means moving the software to a server where it is placed into action. This can be done prior to final testing before a release to the public or the client for whom the software is being developed.

<https://www.youtube.com/watch?v=nZ3BaTY8c9M>

**Building ( vs deploy ? )**

Building software is an end-to-end process that involves many distinct functions. Some of these functions are described below or The term build may refer to the process by which source code is converted into a stand-alone form that can be run on a computer or to the form itself. most important compile or "to create from source code *identifiable* software *assembly* which someone or something can use".

<https://angular.io/guide/build>

**parts of building process**

**Version control**

The [version control](https://en.wikipedia.org/wiki/Version_control) function carries out activities such as workspace creation and updating, baselining and reporting. It creates an environment for the build process to run in and captures metadata about the inputs and outputs of the build process to ensure repeatability and reliability.

### Code quality

Also known as [static program analysis/static code analysis](https://en.wikipedia.org/wiki/Static_program_analysis) this function is responsible for checking developers have adhered to the seven axes of code quality: comments, unit tests, duplication, complexity, coding rules, potential bugs and architecture & design.[[2]](https://en.wikipedia.org/wiki/Software_build#cite_note-2)

Ensuring a project has high-quality code results in fewer bugs and influences nonfunctional requirements such as maintainability, extensibility and readability, which have a direct impact on the [ROI](https://en.wikipedia.org/wiki/Return_on_investment) for a business.[[3]](https://en.wikipedia.org/wiki/Software_build#cite_note-3)

node js 🡪 c

angular 🡪 javascript

### Compilation

This is only a small feature of managing the build process. The compilation function turns source files into directly executable or intermediate objects. Not every project will require this function.

While for simple programs the process consists of a single file being [compiled](https://en.wikipedia.org/wiki/Compiled), for complex software the source code may consist of many files and may be combined in different ways to produce many different versions.

A compiler is a special program that processes statements written in a particular programming language and turns them into machine language or "code" that a computer's [processor](https://whatis.techtarget.com/definition/processor) uses.

**CI**

Continuous integration is the practice of routinely integrating code changes into the main branch of a repository, and testing the changes, as early and often as possible. Ideally, developers will integrate their code daily, if not multiple times a day. Investing in CI results in fast feedback on code changes. Fast as in "within minutes" fast. A team that relies primarily on manual testing may get feedback in a couple hours, but in reality, comprehensive test feedback comes a day–or several days–after the code gets changed. And by that time more changes have occurred, making bug-fixing an archeological expedition with developers digging through several layers of code to get at the root of the problem.

**Bash**

Every program on your computer has the ability to do a vast amount of different things. Read files, start other programs, do math, control devices. The main difference between bash and most other programs is that unlike them, bash was not programmed to perform a certain task. Bash was programmed to take commands from you, the user. To do so efficiently, a "language" was created which allows users to "speak" to the bash program and tell it what to do. This language is the bash shell language and you are about to become intimately familiar with it.

**environment variables**

A **környezeti változók** (angolul *environment variables*) a [számítógép](https://hu.wikipedia.org/wiki/Sz%C3%A1m%C3%ADt%C3%B3g%C3%A9p) működésére nézve fontos információkat tartalmazó szöveges (sztring) szimbólumok, amelyeket az operációs rendszer tárol (A környezeti változók nem tévesztendőek össze a programok belső változóival). A környezeti változók mindig a programokon kívül léteznek, de a programok számára elérhetőek: a programok „közelében”, környezetében fordulnak elő. A környezeti változók értékeit a [programok](https://hu.wikipedia.org/wiki/Sz%C3%A1m%C3%ADt%C3%B3g%C3%A9pes_program) lekérdezhetik és felhasználhatják működésükhöz.

(circle) CI + angular + node / .env /

**Examples:**

* + Environment based API base url
  + Create secrets or database config based on environment
  + Using different services in test env / mocha chai jasmine karma/

**Expected skills (4/5)**

* **Able to recognize and resolve merge conflicts.**

A conflict arises when two separate branches have made edits to the same line in a file, or when a file has been deleted in one branch but edited in the other. Conflicts will most likely happen when working in a team environment.

## Git commands that can help resolve merge conflicts

**General tools**

git status

The status command is in frequent use when a working with Git and during a merge it will help identify conflicted files.

git log --merge

Passing the --merge argument to the git log command will produce a log with a list of commits that conflict between the merging branches.

git diff

diff helps find differences between states of a repository/files. This is useful in predicting and preventing merge conflicts.

**Tools for when git fails to start a merge**

git checkout

checkout can be used for *undoing* changes to files, or for changing branches

git reset --mixed

reset can be used to undo changes to the working directory and staging area.

**Tools for when git conflicts arise during a merge**

git merge --abort

Executing git merge with the --abort option will exit from the merge process and return the branch to the state before the merge began.

git reset

Git reset can be used during a merge conflict to reset conflicted files to a know good state

* **Able to explain workflow details on collaborating with version control.**

https://www.git-tower.com/learn/cheat-sheets/vcs-workflow

legyen ennyi elég

* **Able to explain the flow of releasing code to production servers**

<http://guides.beanstalkapp.com/deployments/best-practices.html>

ne keverjük bele, nem használtuk

* **Able to show how the application is configured for different environments**

Most web applications require to run in different environments before they make their way to production. You might need a build for your QA team to perform some tests, or a specific build to run on your continuous integration server for instance.

All of these builds will likely require a different config: Different server URLs, different logging options, etc.

"environments": {  
 "dev": "environments/environment.ts",  
 "prod": "environments/environment.prod.ts",}

https://alligator.io/angular/environment-variables/

* **Able to build the app from the command line interface**

ng serve + npm start( nodemone) + ts node

ng build