Cloud and service models

This morning you probably played a song on Spotify, opened Classroom and saved something to Google Drive. None of it ran on your phone, not really. Where was the song? Where is your homework right now? And who keeps the machine that stores it switched on at three in the morning?

The answer is “in the cloud”, and it is less mysterious than it sounds. A hotel in Calvià that receives ten times more bookings in August than in January cannot buy ten servers for August and throw nine away in September. It rents. As a DAM developer, almost everything you build will end up running on somebody else’s computer, so you need to know exactly what you are renting.

What you will learn

- Define **cloud computing** with its five characteristics, and tell it apart from **on-premise**
- Tell the three **service models** apart: **IaaS**, **PaaS** and **SaaS**
- Say who manages what in each model, and which part is **always** yours
- Classify real services you use, with proof from the provider’s own website
- Publish your own web page with **GitHub Pages**, and say which model it is
- Avoid the trap: “the cloud” is not in the sky, it is a building full of somebody else’s servers

The cloud is renting from the utility

> **Cloud computing** is on-demand access, over the network, to computing resources (servers, storage, databases, applications) that a provider shares among many customers, that can grow or shrink quickly, and that you pay for according to use.

In the city-builder game of this module, the cloud is **renting water and electricity from a utility** instead of building your own plant. You open the tap and pay the bill. The alternative is called **on-premise**: the company buys its own servers and keeps them in its own building.

The US standards institute NIST wrote the definition everyone quotes (SP 800-145, 2011). It lists five characteristics. A service that lacks them is not really cloud:

Characteristic	What it means	Example
On-demand self-service	You sign up and create resources yourself, without waiting for anyone to install anything.	Creating a server from a web form in a couple of minutes
Network access	You reach it from any place and device with a connection.	Opening Classroom at school, at home or on your phone
Shared resources	The same hardware serves many customers, kept separate logically.	Thousands of companies in the same data centres

Characteristic	What it means	Example
Elasticity	Resources can be added or removed quickly as demand changes.	The hotel's website in August and in January
Pay per use	Use is measured, and you pay for what you consume.	Like the electricity bill

Back to the hotel. On-premise, it buys a server, puts it in an air-conditioned room and pays somebody to look after it: a big investment on day one, and if it falls short in August, bad luck. In the cloud there is no initial investment, it pays monthly according to use, and in August it asks for more capacity.

> **Careful: "the cloud is somewhere up there"**
> The cloud is physical. It is a **data centre**: a building full of servers, cables, power supplies and cooling, owned by a provider such as Amazon, Microsoft, Google or Hetzner. An external hard disk or a server in the office cupboard is not cloud: nobody else shares it, it does not grow on demand and nobody bills you by use.

Three service models: IaaS, PaaS and SaaS

You can rent at three different heights. The higher you go, the more the provider does for you.

> A **service model** says how much of the system the provider gives you ready-made, and how much is left for you to manage.

	IaaS · infrastructure	PaaS · platform	SaaS · software
What you get	Virtual machines, network and storage	A ready environment where you deploy your code	A finished application, ready to use
What you manage	Operating system, software, application and data	The application and the data	Only your data and settings
Who it is for	System administrators	Developers	End users
Examples	Amazon EC2, Azure Virtual Machines, Google Compute Engine, Hetzner Cloud	Google App Engine, Heroku, Azure App Service, GitHub Pages	Gmail, Google Classroom, Microsoft 365, Spotify, Kahoot

- **IaaS** (*Infrastructure as a Service*): you rent a **virtual machine**, a computer simulated by software on the provider's hardware. It is like renting an empty plot with water and power connected: you build the house.
- **PaaS** (*Platform as a Service*): you bring your code and the platform runs it. Like renting a furnished flat: you bring your things.
- **SaaS** (*Software as a Service*): you just use the application. Like staying in a hotel.

Who manages what

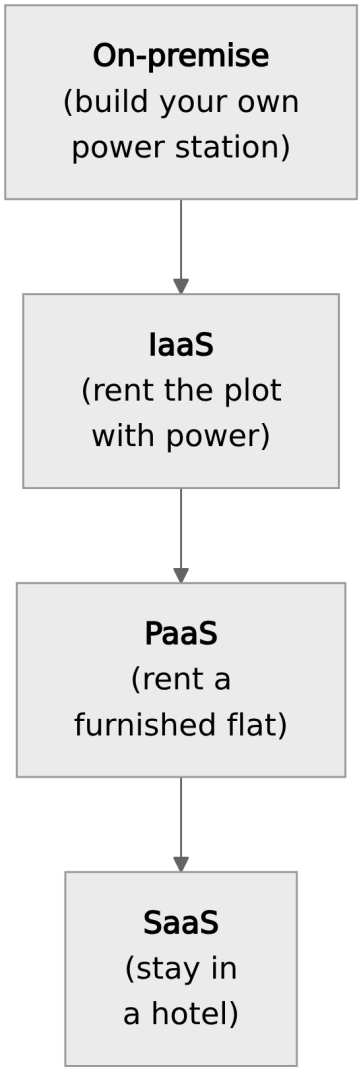
Stack the layers of any system from the building up to your data. For each model, mark what is the **customer's** job (C) and what is the **provider's** (P):

Layer	On-premise	IaaS	PaaS	SaaS
Data	C	C	C	C

Layer	On-premise	IaaS	PaaS	SaaS
Application	C	C	C	P
Runtime (Java, Node, database…)	C	C	P	P
Operating system	C	C	P	P
Virtualisation	C	P	P	P
Servers, disks and network	C	P	P	P
Building, power and cooling	C	P	P	P

Look at the first row. **Your data are always your responsibility**, whatever the model. The higher you climb, the less administration you do, but the less control you have. Providers call this the **shared responsibility model**.

> **Careful: the borders are not walls**
> Some services sit between two models. **Firebase**, for example, is a platform you build on (PaaS), but it also offers finished pieces such as user login. What is assessed is the



> **reason**: say what the customer manages and what the provider manages, and the label follows.

GitHub Pages: a platform you can use today

> **GitHub Pages** is a free GitHub service that publishes a **static website** (HTML, CSS and images that do not change per visitor) straight from a code repository.

You put the files in a repository; GitHub stores them, serves them to the world and keeps the servers running. You never see a server. That is the definition of **PaaS**, and you will use it all unit: in Deliverable 3 your group publishes its dossier as a GitHub Pages site.

Three facts to know before you start (checked on 25/10/2026 in GitHub's documentation):

Fact	What it means for you
GitHub's terms require you to be **13 or older**	Everyone in this course can open an account; it is free and needs no card
With a free account, the repository must be **public**	Everything you upload can be seen by anybody: no personal data, no phone numbers; your first name is enough
Pages is not for shops or for collecting passwords	A dossier, a portfolio or a project page is fine

In class: what level does it play at?

Part 1 · Classify real services. In the Classroom template:

Step	Do this	You should get
1	Write down **four services you used this week** (phone or computer) and add these four: Amazon EC2, Heroku, Google App Engine, Hetzner Cloud	A list of eight services
2	Classify each one as **IaaS, PaaS or SaaS** and write one sentence on why: what does the customer manage, and what does the provider?	Eight labels with a reason
3	For each, paste a link to the **official** page (product or pricing page, not a blog or an AI) and quote the sentence that proves it	Eight pieces of evidence
4	For your project company, list **three systems** it has or probably has (website, bookings, ERP, email, plant control…) and, for each, the model you would propose and a **real provider** that offers it	The draft of Section A of D3

Part 2 · Your first site on GitHub Pages.

Step	Do this	You should see
1	Sign in to GitHub, or create a free account at github.com/signup (email, password, username, a check and a code sent to your email)	Your GitHub home page
2	Click **+** (top right) → **New repository**. Name it exactly `yourusername.github.io`. Visibility **Public**. Turn on **Add README**. **Create repository**	Your new repository with a `README.md`
3	**Add file** → **Create new file**. Name it `index.html`. Write a minimal page: a heading with your first name and the sentence "This website is a … service because …". **Commit changes**	`index.html` in the repository
4	**Settings** → **Pages**. Under **Build and deployment**, **Source**: **Deploy from a branch**; **Branch**: `main` and `/ (root)`. **Save**	A message that your site is being published

Step	Do this	You should see
5	Wait (it can take up to 10 minutes) and open `https://yourusername.github.io`	Your page, visible from any computer in the world

If you cannot or do not want to open a GitHub account, make the page with **Google Sites** and your school account, and say in the sentence that Google Sites is SaaS, not PaaS. It does not affect your individual mark: in the deliverable, the website is group work.

Hand in on Classroom: the table of part 1 with its evidence, your three company systems and the link to your site. **No generative AI**: this activity is assessed (class work, 30%).

For your project

This lesson starts Deliverable 3 · Cloud architecture, due **Monday 8 February 2027 at 23:59**, with individual defences in session 14 that same day.

Today's work	Becomes	Criterion
Three company systems with a model and a real provider	Section A: service models	a
Your GitHub account and first site	The group's website, published by the Technician	Group part

Key takeaways

- **Cloud computing** is renting computing resources on demand, over the network, shared, elastic and paid by use. It runs in real **data centres**. The opposite is **on-premise**.
- **IaaS** gives you machines, **PaaS** gives you a place to deploy your code, **SaaS** gives you the finished application.
- The higher the model, the less you administer and the less you control. **Your data are always your responsibility** (the **shared responsibility model**).
- End users mostly meet **SaaS**; developers live mostly in **PaaS** and **IaaS**.
- **GitHub Pages** publishes a **static website** from a repository: a PaaS you can use for free, with a public repository.

Quick check

1 Which of these is cloud computing?

A An external hard disk where the company copies its files every Friday
B A server bought by the company and kept in its own office
C Storage rented from a provider, reached over the internet and billed by the GB used
D A USB stick shared by the whole office

2 A company rents a virtual machine and installs the operating system, the database and its application on it. Which model is it?

A IaaS
B PaaS
C SaaS
D On-premise

3 In which model does the customer manage only their data and settings?

- **A** IaaS
- **B** PaaS
- **C** SaaS
- **D** In none: the customer always manages the operating system

4 You upload the files of your website to a repository and GitHub Pages publishes it without you managing any server. Which model is it?

- **A** IaaS
- **B** PaaS
- **C** SaaS
- **D** It is not cloud, because it is free

5 Which statement about the shared responsibility model is true?

- **A** In SaaS, the provider is responsible for the customer's data
- **B** In IaaS, the provider updates the operating system of your virtual machines
- **C** Whatever the model, the customer is responsible for its data
- **D** On-premise, the provider manages the building and the power

6 A classmate says: "The cloud is just the internet; my files are in the air." What is the best answer?

- **A** True: the cloud has no physical location
- **B** The files are in real data centres owned by a provider; the internet is only the road to them
- **C** True, but only for SaaS services
- **D** False: cloud files are always stored on your own device

7 A bakery in Inca wants online ordering without hiring a programmer, and a start-up in Palma wants to deploy its own app without managing servers. Which models fit each one best?

- **A** IaaS for the bakery, SaaS for the start-up
- **B** SaaS for the bakery, PaaS for the start-up
- **C** PaaS for both
- **D** On-premise for both, to save money

You now know what you can rent. Next you will see who owns the power station, what the cloud is used for and how much the bill is: Deployment and functions.

Solutions

1. C · The cloud is **on demand**, **over the network**, **shared**, **elastic** and **paid by use**. The disk, the office server and the stick have none of those features; the office server is **on-premise**.

2. A · The provider gives only the **infrastructure** (the virtual machine, network and storage). Everything from the operating system up is the customer's job: that is **IaaS**.

3. C · In **SaaS** the application is finished and the provider runs everything below it. The customer still owns the **data**, which is their responsibility in every model.

4. B · You bring the code; the platform stores and serves it. That is **PaaS**. Being free does not stop it being cloud: the provider still runs the data centre.

5. C · The **data** row is always the customer's. In IaaS the operating system is also yours; on-premise there is no provider at all.

6. B · The cloud is physical: buildings full of servers, cables and cooling. The **network** is how you reach them. That is why, in lesson 4, it will matter **in which country** they are.

7. B · The bakery needs a **finished application** (SaaS). The start-up has its own code and wants somebody else to run the servers: that is **PaaS**.