**Pràctica 1.- Models de servei**

Una PyME desenvolupa un portal web amb:

* Correu corporatiu i suite ofimàtica.
* Web pública amb pics d’alta demanda.
* API de reserves (BD).
* Backoffice intern, login d’usuaris, ERP i CRM propi.
* Dashboard d’analítica setmanal.

A realitzar:

Per a cada component, tria el model de servei més adequat (IaaS, PaaS o SaaS) i justifica-ho en unes línies.

1. Cerca software concret per a cada component
2. Indica què manté el proveïdor i què manté l’empresa component.

* Correu corporatiu i suite ofimàtica.

**Model Recomant: SAAS**

Per què: és un servei madur, amb gestió d’actualitzacions, seguretat, còpies i alta disponibilitat; una PyME hi guanya en agilitat i cost predictable. (SaaS = aplicació centralitzada, el proveïdor gestiona infra i aplicació).

**Software concret:**

Google Workspace (Gmail, Drive, Docs, Calendar).

Microsoft 365 (Exchange/Outlook + Word/Excel/PowerPoint + OneDrive).

**Manteniment:**

Proveïdor (Google / Microsoft): infra física, servidors, patches, actualitzacions d’aplicació, antivirus/antispam, SLA, còpies i redundància. [Google Workspace+1](https://workspace.google.com/?utm_source=chatgpt.com)

Empresa**:** gestió d’usuaris (creació/baixa), polítiques d’accés, configuració del domini i DNS, backup addicional si vol (política interna), formació i control d’accés.

* Web pública amb pics d’alta demanda.

**Model recomanat:** **PAAS**  
Per què: per a webs estàtiques o generades per un framework Jamstack, PaaS (Vercel/Netlify) + CDN cobreix desplegament, escalar i cache; si necessites control total sobre la capa d’object storage i CDN (S3 + CloudFront) agafues IaaS. Els PaaS ofereixen deploy contínuo + CDN integrat; IaaS dóna més control i funcions avançades.

**Software concret:**

Vercel (PaaS / edge CDN per Next.js i estàtic). [Vercel](https://vercel.com/?utm_source=chatgpt.com)

Netlify (PaaS / CDN per Jamstack).

AWS S3 + CloudFront (object storage + CDN) si vols IaaS amb màxim control.

**Manteniment:**

Proveïdor (Vercel/Netlify/AWS)**:** infraestructura, CDN, rèpliques, escalat automàtic, patches, latència i disponibilitat. [Vercel+2netlify.com+2](https://vercel.com/?utm_source=chatgpt.com)

Empresa: optimització d’imatges (formats, compressió), estratègia de cache/headers, CI/CD (configurar builds), controlar domini i SSL, logica aplicativa (si hi ha), i contingut (imatges, textos).

* API de reserves (BD).

**Model recomanat:** **PaaS**   
Per què: una API necessita desplegament flexible i escalat; un PaaS elimina la major part de l’operativa (gestió SO, parches, backups DB) i permet centrar-se en el codi i la integració amb la BD gestionada. Si la PyME vol control extrem, pot optar per-IaaS (VMs + BD).

**Software concret / serveis:**

Heroku (PaaS) per desplegar l’API + addons de Postgres gestionat.

Azure App Service + Azure Database for PostgreSQL/MySQL (PaaS).

AWS ECS / EKS + RDS (més IaaS / container) si prefereixes containerització i control.

**Manteniment — qui fa què:**

Proveïdor (Heroku/Azure/AWS): infra, containar/servici d’aplicacions (si PaaS), sistema de fitxers base, backups gestionats de BD (si s’escull BD gestionada), patches i SLA.

Empresa: codi de l’API, migracions DB (schema), testing, secrets (gestió), escalat lògic (configurar plans), esquemes de backup addicionals, monitoratge de l’aplicació i política de recuperació.

* Backoffice intern, login d’usuaris, ERP i CRM propi.

**Model recomanat:** **híbrid: IaaS + SaaS/PaaS**.  
Per què: si ERP/CRM és propi (custom) convé IaaS o contenidors sobre PaaS perquè cal control sobre base de dades, integracions i backups; però per l’autenticació (login) sovint és eficient utilitzar un SaaS d’identitat (Auth0) o un sistema self-hosted (Keycloak) segons requisits de privacitat.

**Software concret / serveis:**

ERP/CRM**:** *si es vol llavors llista d’opcions* — Odoo (pot ser self-hosted o a Odoo.sh), o ERP intern desplegat a VMs/containers.

Login (SaaS): Auth0 (IdP com a servei).

Login (self-hosted): Keycloak (open-source SSO/IdM).

**Manteniment — qui fa què:**

Si ERP/CRM desplegat PaaS / Odoo.sh**:** el proveïdor/PaaS gestiona la plataforma i backups segons pla; l’empresa manté la personalització, dades i configuració. (veure Odoo per opcions de hosting).

Autenticació (Auth0 SaaS): el proveïdor manté la plataforma d’identitat; l’empresa gestiona configuració de clients, regles, permisos i usuaris.

* Dashboard d’analítica setmanal.

**Model recomanat:** **SaaS**

Per què: dashboards setmanals no solen necessitar infra dedicada: eines SaaS s’integren amb la BD i ofereixen compartició i programació d’informes; si necessites dades sensibles localment o embedding, Metabase (self-hosted) és una opció.

**Software concret:**

Looker Studio (Google) gratuït / Pro per organitzacions; ràpid per informes compartits

Power BI (Microsoft) suite BI amb servei cloud i Desktop.

Metabase (open-source, self-hosted o cloud) — fàcil, programaciód’informes, embed.

**Manteniment:**

* SaaS (Looker Studio / Power BI Service): proveïdor manté infra, connectors, availability i updates; empresa manté definicions de dades, connexions (credentials), accessos i assegura la qualitat de dades.
* Self-hosted (Metabase): l’empresa desplega i manté el servidor Metabase i la infra (o contracta Metabase Cloud); el proveïdor només manté el codi i docs (si self-hosted).