

BERRITECH





NOR GARA ETA ZER ESKATU DIGUTE?	4
WEB ORRIA	5
Helburua	5
Irismena eta funtzionalitateak	5
Funtzionalitate nagusiak	5
Baldintzak eta mugak	6
Erabilitako teknologiak	6
Proiektuaren egitura	7
Diseinuaren eta UXaren azalpena	7
Hasierako pantaila (Splash)	7
Layout nagusia	8
Funtzionamendu teknikoa (labur)	8
Karpeta aukeraketa	8
Ticketen identifikazioa	9
Zerrenda eta xehetasuna	9
Probak eta egiatzapena	10
Probak	10
Arazo nagusiak eta konponbideak	10
Ondorioak eta hobekuntza posibleak	11
Ondorioak	11
Etorkizuneko hobekuntzak	11
PROGRAMAZIOA	12
PARTEKATUTAKO KARPETA	12
Helburua	12
Karpeta partekatuaren deskribapena	12
Partekatzeko konfigurazioa	13
Karpeta partekatuen erabilera web aplikazioan	14
Abantailak	14
Mugak eta etorkizuneko hobekuntzak	15
Mugak	15
Etorkizuneko hobekuntzak	15
DATU BASEA	16
Diseinu Kontzeptuala (E-R Diagrama)	16
Erlazioak eta Kardinalitateak	16
Eedu Logikoa (Eedu Erlazionala)	16
Implementazio Fisikoa (SQL)	17
3.1. Taulen Sorkuntza	17
Datuen Adibideak eta Normalizazioa	17
Ondorioak	18
DIGITALIZAZIOA	19
HERRIKO SUPERMERKATU TXIKIRAKO DIGITALIZAZIOA	19
Digitalizazio-prozesua honela antolatuko litzateke:	20
Digitalizaziorako erabiliko genituzkeen ERP funtzionalitateak hauek dira:	21





NOR GARA ETA ZER ESKATU DIGUTE?

Gu digitalizazio-proiektuak garatzen dituen enpresa txiki bat gara, enpresa txikien kudeaketa-prozesuak hobetzea helburu duena. Kasu honetan, herriko supermerkatu txiki baten ticketen kudeaketa digitalizatzea eskatu digute: baskuletan .txt formatuan gordetzen diren ticketak automatikoki prozesatu, Ogasunera bidali eta datu-base batean gordetzea. Horrez gain, ticketen zerrenda eta bakoitzaren informazioa ikusteko web aplikazio bat garatu dugu. Azkenik, etorkizunean ERP sistema bat ezarri eta datuak hodeira migratzeko aukerak aztertu ditugu, proiektuaren bideragarritasuna eta jasangarritasuna ebaluatzeko.



WEB ORRIA

Helburua

Proiektu honen helburua da **ticketak gordetzen diren karpeta bat aukeratzea** eta karpeta horretan dauden **fitxategi guztiak (ticketak)** web interfaze batean automatikoki erakustea. Erabiltzaileak ticket bat aukeratzean, **ticketaren edukia** pantailan ikusteko aukera izanGO DU..

Sistema hau bereziki egokia da **TicketBAI erronkan** erabiltzeko, ticketak .txt edo "Documento de texto" moduan gordetzen direlako, askotan **luzapenik gabe** (adib.: 20260122090727.txt).

Funtzionalitateak

Funtzionalitate nagusiak

- **Hasierako pantaila:** "Sartu" izena duen botoi bat dago sartzeko.
- **Karpeta aukeraketa:** Erabiltzaileak ticketak dituen karpeta aukeratzen du.
- **Ticket zerrenda:** Aukeratutako karpetak ticketen fitxategi-izenak zerrendan agertzen dira.
- **Bilaketa:** Ticketen izenaren edo edukien arabera filtratzen da.
- **Ordena:** Berrienetik zarrenera dago jarrita eta (A-Z) edo (Z-A) formatuan.
- **Xehetasun panela:** Ticket bat klikatzean, edukia panel batean erakusten da guztia.
- **Ekintza osagarriak:**
 - **Kopiatu:** Ticketaren edukia clipboard-era.
 - **Deskargatu:** Ticketaren edukia .txt gisa deskargatu.



Baldintzak eta mugak

- Weba **frontend hutsean** egin da (HTML/CSS/JS).
- Karpeta irakurtzeko erabiltzen da **File System Access API** (showDirectoryPicker()), eta horrek baldintza hauek ditu:
 - **Chrome/Edge** gomendatzen da.
 - **Live Server (localhost)** erabilita funtzionamendu egokia.
 - Beste nabigatzaile batzuetan (adib. Firefox) muga gehiago egon daitezke.

Erabilitako teknologiak

- **HTML5**: egitura eta interfazearen osagaiak.
- **CSS3**: diseinu moderno eta responsive-a (panelak, txartelak, splash).
- **JavaScript**: logika nagusia:
 - Karpeta aukeraketa
 - Fitxategien irakurketa
 - Zerrendaren karga eta filtrazioa
 - Ticketen xehetasunak erakustea
- **VS Code + Live Server**: garapen-ingurunea eta probak.



Proiektuaren egitura

Proiektuak egitura simple eta egonkorra du:

```
berritech-ticketak/  
    index.html  
    styles.css  
    app.js
```

- index.html: splash pantaila + aplikazioaren layout-a.
- styles.css: estiloak (koloreak, tipografia, panelak, responsive).
- app.js: logika (karpeta aukeratu, ticketak kargatu, bilatu, ordenatu, xehetasuna, kopiatu, deskargatu).

Diseinuaren eta UXaren azalpena

Hasierako pantaila (Splash)

Aplikazioak hasierako pantaila bat du "Sartu" botoiarekin.
Helburua:

- Erabiltzaileari ongietorria ematea.
- Aplikazioaren funtzioa azaltzea.
- Eta interfazea "garbi" mantentzea (app-era sartu aurretik).



Layout nagusia

Aplikazioak bi panel nagusi ditu:

- **Ezkerrean:** Ticket zerrenda (fitxategi-izenak + preview laburra).
- **Eskuinean:** Ticket xehetasuna (eduki osoa).

Horrek aukera ematen du:

- Ticketen artean azkar nabigatzeko,
- Eta edukia modu eroosoan konsultatzeko.

Funtzionamendu teknikoa (labur)

Karpeta aukeraketa

Erabiltzaileak showDirectoryPicker() erabiliz karpeta hautatzen du. Behin hautatuta:

- Aplikazioak dirHandle.entries() bidez karpetako fitxategiak enumeratzen ditu.



Ticketen identifikazioa

Ticketak .txt izan daitezke edo **luzapenik gabeko testu-fitxategiak** (Windows-ek “Documento de texto” gisa erakusten ditu). Horregatik, sistemak irizpide hauek onartzen ditu:

- .txt edo .TXT luzapena dutenak,
- Edo luzapenik gabe baina testu-motakoak (fitxategiaren edukia testua denean).

Zerrenda eta xehetasuna

- Kargatzean, ticket bakoitzaren edukia irakurtzen da eta **preview** bat sortzen da (lehen 2-3 lerroak).
- Zerrendan ticket bat aukeratzean, edukia eskuineko panelean erakusten da.
- Edukia textCache moduan gordetzen da berriro irakurtzea saihesteko.



Probak eta egiaztapena

Probak

- Karpeta aukeratu → “Karpeta aukeratuta: ...” mezua agertzen dela egiaztatu.
- Fitxategi kopurua (“Ticketak: X”) eguneratzen dela egiaztatu.
- Ticket bat aukeratu → edukia panelean agertzen dela egiaztatu.
- Bilaketa → zerrenda filtratzen dela egiaztatu.
- Ordenazioa → zerrenda behar bezala ordenatzen dela egiaztatu.

Arazo nagusiak eta konponbideak

- **Fitxategi-izenak luzapenik gabe** zeuden → .endsWith(".txt") filtroa ez zen nahikoa.
 - Konponbidea: luzapenik gabeko testu-fitxategiak ere onartzea.



Ondorioak eta hobekuntza posibileak

Ondorioak

Sistema egonkorra eta erabilerraza lortu da: karpeta aukeratu eta ticketak web bidez ikusi.

Etorkizuneko hobekuntzak

- Azpikarpeten irakurketa errekurtsiboa (walk).
- Ticketen edukia “parseatzea” eta datu estrukturatuak erakustea:
 - data/ordua, guztira, BEZa, produktuak...
- Multi-erabiltzaile modua (server bidez), taldeko erabilera errazteko.



PROGRAMAZIOA

PARTEKATUTAKO KARPETA

Helburua

Atal honen helburua da gure C# kontsola-aplikazioak partekatutako karpetatik ticketak automatikoki irakurri, prozesatu eta ondoren XML sortu + balidatu + bidali + backup + DBra sartu nola egiten duen azaltzea.

Sistema hau TicketBAI erronkarako egokia da, ticketak .txt fitxategi moduan sortzen direlako eta prozesua zentralizatzea beharrezkoa delako.

Karpeta partekatuaren deskribapena

Programa honetan partekatutako karpeta nagusia hau da:

- \\MSIDEANDONI\CentralTicketBAI

Karpeta horren barruan aplikazioak egitura hau erabiltzen du:

- Sarrera (ticketak):
 - \\MSIDEANDONI\CentralTicketBAI\frutategia\Tiketak\
 - \\MSIDEANDONI\CentralTicketBAI\harategia\Tiketak\
 - \\MSIDEANDONI\CentralTicketBAI\okindegia\Tiketak\
 - \\MSIDEANDONI\CentralTicketBAI\txarkutegia\Tiketak\
- Irteerak (XML):
 - \\MSIDEANDONI\CentralTicketBAI\XML\
- Backup-ak:
 - Ticket backup: \\MSIDEANDONI\CentralTicketBAI\BackUp\TICKET\
 - XML backup: \\MSIDEANDONI\CentralTicketBAI\BackUp\XML\



- Erregistroa (Excel log):

- \\MSIDEANDONI\\CentralTicketBAI\\erregistroa.xlsx

Partekatutako konfigurazioa

Partekatutako karpeta erabiltzeko beharrezko da:

1. UNC bidea erabiltzea

Programak zuzenean erabiltzen du sareko bidea (\\MSIDEANDONI\\...), eta azpibideak Path.Combine() bidez osatzen ditu (bideak ondo lotzeko).

2. Baimenak

- Ticketak irakurtzeko: Read
- Ticketak backup-era mugitzeko: Move/Write
- XML karpetan sortzeko: Write
- BackUp karpetetan mugitzeko: Write

3. Sistema/ingurune baldintzak (praktikoak)

- Sareko konexioa aktibo egotea (zerbitzaria eskuragarri)
- Erabiltzaileak karpeta partekatuan baimen egokiak izatea

Karpeta partekatuen erabilera web aplikazioan

Web aplikazioan karpeta irakurtzea normalean nabigatzailearen baimenekin egiten da (showDirectoryPicker()), eta erabiltzaileak eskuz hautatu behar du karpeta.

Gure C# programan, ordea:

- Karpeta partekatua aurrez definituta dago (\\MSIDEANDONI\\CentralTicketBAI)
- Irakurketa/mugimendua sistematik zuzenean egiten da (Directory.GetFiles, File.OpenText, File.Move...)



- Hau da: web-aren aldean, hemen gakoa Windows baimenak + sareko bidea dira.

Abantailak

- Iturburu bakarra: taldekide guztiekin ticket berdinak prozesatzen dituzte.
- Automatizazioa: 1 sakatu eta pipeline osoa martxan jartzen da (irakurri → XML → XSD → bidali → backup → DB).
- Backup segurua: prozesatutako .txt-ak eta XML-ak backup karpetetara mugitzen dira.
- Erregistroa: XML bidalketak Excel batean log-eatzen dira (erregistroa.xlsx).
- Akats gutxiago: lokaleko bideen arazoak saihesten dira (dena zentralizatua).

Mugak eta etorkizuneko hobekuntzak

Mugak

- Sarearen menpekotasuna: \\MSIDEANDONI ez badago eskuragarri, prozesua ezin da egin.
- Baimen arazoak: erabiltzaile batek Write/Move ez badu, ezin du backup egin edo XML sortu.
- Fitxategien filtroa: momentuz .txt bakarrik hartzen dira (*.txt), luzapenik gabeko testu-fitxategiak ez.
- XML bidea hartzea: HartuXmlBidea()-k karpetako lehen .xml fitxategia hartzen du; karpetan XML asko badaude, ez du beti "azkena" hartuko.
- Errore-mezuak: badira kasu batzuk non mezua "ez dago tiketik prozesatzeko" ager daitekeen fitxategi formatu desegokiagatik (lerroak ez badu 5 zatiko \$ egitura).



Etorkizuneko hobekuntzak

- Konfigurazio-fitxategia: \\MSIDEANDONI\CentralTicketBAI bezalako bideak appsettings.json edo antzeko batean gordetzea (kodea ukitu gabe).
- Azpikarpeten irakurketa errekurtsiboa: baskula gehiago edo egitura konplexuagoa onartzeko.
- Fitxategi mota gehiago: luzapenik gabeko “text” fitxategiak ere onartzea (web proiektuan egin zen bezala).
- XML “azkena” hartza: sortu berriena aukeratzea (timestamp edo GetCreationTime erabiliz).

Menu nagusia eta aukeren funtzionamendua

Menuaren egitura orokorra

Programa abiatzean Main() metodoak while begizta batean mantentzen du aplikazioa, erabiltzaileak “Ateria” aukeratu arte. Menuak 4 aukera ditu:

1. Prozesatu
2. Estadistikak ikusi
3. Datu basean dagoen saltzailea aldatu
4. Ateria

Erabiltzaileak zenbaki bat sartzen du eta switch bidez dagokion funtzioa exekutatzen da. Balio okerra sartuz gero (edo testua sartuz gero), errorea harrapatzen da eta mezua erakutsi ondoren Enter sakatu arte ez da menura bueltatzen (EnterItxaron()).

1. Aukera: Prozesatu (1 zapalduta)

Helburua

Partekatutako karpetan dauden 4 baskuletako ticketak irakurri, objektu bihurtu, XML sortu, XSD bidez balidatu, email bidez bidali, backup egin, eta amaieran MySQL datu-basean gorde.



Prozesuaren pausuak

1. Datu-baseko mapak kargatu (urretiko prestaketa)
 - BaskulakKargatu() → baskula izena → baskula_id
 - ProduktuakKargatu() → (produktu izena + baskula_id) → id_produktua
 - AutoSalmentaKargatu() → autosalmentaren izena (id_saltzailea = 0)
2. Ticketak irakurri (partekatutako karpetatik)
 - \\MSIDEANDONI\\CentralTicketBAI\\<baskula>\\Tiketak\\ karpetetan *.txt bilatzen ditu.
 - Fitxategi bakoitzean:
 - Fitxategiaren sorrera-data hartzen da:
File.GetCreationTime() → DateTime (minutu zehaztasunera biribilduta)
 - Lehen lerroa irakurtzen da eta \$ bidez zatitzen da
Formatu zuzena: 5 zati (produktua, saltzailea, prezioa, kopurua, guztira)
 - Zenbakiak CultureInfo("es-ES") erabilita parseatzen dira.
 - Datu-baseko mapekin produktua/baskula lotzen da eta Tiketa objektua sortzen da.
 - Bukatutakoan, baskula horretako .txt guztiak backup-era mugitzen dira:
 - ...\\Tiketak\\ → \\MSIDEANDONI\\CentralTicketBAI\\BackUp\\TICKET\\
3. (Aukerazkoa) Saltzailea aldatzea listan
 - Erabiltzaileari galdetzen zaio: "Ticketen bateko saltzailea aldatu nahi duzu?"



- “bai” sartuz gero, SaltzaileaAldatu(lt) bidez listako ticket baten Langilea aldatzen da (oraindik XML sortu aurretik).

4. XML sortu

- TiketakXmlraGorde(lt) → \\...\\XML\\tiketak_YYYYMMDD_HHMMSS.xml
- Amaieran mezua: “✓ Tiketak egoki prozesatu dira”

5. XSD balidazioa

- Validatu(xml, Tiketak.xsd)
- Ondo bada: “✓ XML balidatua izan da”; bestela errore-mezua eta “✗”.

6. XML bidali (Outlook) + Excel erregistroa

- Email hartzalea finkoa da (adib. ogasuna...).
- XML eranskin gisa bidaltzen da.
- Bidalketa erregistroa.xlsx fitxategian gordetzen da (Data, Hartzalea, XML izena, bidea, tamaina...).

7. XML backup-era mugitu

- \\...\\XML\ → \\...\\BackUp\\XML\

8. MySQL datu-basean sartu

- Ticket bakoitza tiketa taulan sartzen da (id_produktua, id_saltzailea, fetxa, kantitatea_kg, prezioa_guztira).

9. Enter bidez geldialdia

- Amaieran: Enter sakatu arte ez du aurrera egiten (EnterItxaron()), eta ondoren pantaila garbitu eta menura bueltatzen da.



2. Aukera: Estadistikak ikusi (2 zapalduta)

Helburua

MySQL datu-basean gordetako ticketekin estatistika nagusiak kalkulatzea eta kontsolan modu “politean” erakustea.

Zer kalkulatzen du?

SQL kontsulten bidez, “TOP 1” lortzen du:

- Saltzaileak:
 - Produktu gehien saldu duen saltzailea (kg)
 - Diru gehien ingresatu duen saltzailea (EUR)
 - Produktu gutxien saldu duen saltzailea (kg)
 - Diru gutxien ingresatu duen saltzailea (EUR)
- Produktua:
 - Gehien saldu den produktua (kg)
 - Gutxien saldu den produktua (kg)

Nola erakusten da?

- Spectre.Console erabiliz:
 - Panel bat titulurako
 - Taula (Atala / Izena / Balioa)
 - BarChart bat konparazio azkarra ikusteko

Amaieran, EnterItxaron() erabiltzen da: erabiltzaileak irakurri arte ez da menura bueltatzen.



3. Aukera: Datu basean dagoen saltzailea aldatu (3 zapalduta)

Helburua

MySQL datu-basean dagoen ticket baten saltzailea aldatzea (UPDATE), kontsolatik aukeratuta.

Prozesua

1. Ticketak DB-tik kargatu

- SELECT ... FROM tiketa ORDER BY id_tiketa
- Lista batean gordetzen dira (List<Saltzailea>).

2. Ticketen zerrenda erakutsi

- Kontsolan ticket bakoitza agertzen da:
 - Ticket ID-a, Produktua (id), Saltzailea (id), Data, Kantitatea, Guztira...

3. Erabiltzaileak aukeratzen du

- Lehenik: zein ticket aldatu (zerrendako posizioa)
- Ondoren: saltzaile berriaren ID-a (0-7)

4. UPDATE egin

- UPDATE tiketa SET id_saltzailea=@aldaketa WHERE id_tiketa=@id;

Kontuan hartzeakoak (mugak)

- Hemen aldaketa DB-n zuzenean egiten da (ez listan).
- Ticketik ez badago: “Datu basean ez dago ticketik gordeta”.
- Aukeraketa okerra bada (zenbakiak ez badu balio): errore mezua eta menura buelta.



4. Aukera: Ateria (4 zapalduta)

Helburua

Programatik modu kontrolatuan irtetea.

Funtzionamendua

- Spectre.Console bidez mezu bat erakusten da:
 - "✓ Programatik egoki irten zara"
- Ondoren Environment.Exit(0) exekutatzen da eta aplikazioa amaitzen da.

Erabilgarritasuna: Enter bidez kontrola + erroreen kudeaketa

- Erabiltzaileak aukera okerra sartzen badu edo testua sartzen badu, try/catch bidez harrapatzen da.
- Mezu argia ematen da eta Enter sakatu arte ez da jarraitzen.
- Honek programa "azkar pasa" ez dadin eta erabiltzaileak pauso pauso kontrolatzeko balio du.



DATU BASEA

Diseinu Kontzeptuala (E-R Diagrama)

Datu-basearen diseinua supermerkatu baten eguneroko transakzioak jasotzeko egituratuta dago. Lau entitate nagusi identifikatu dira:

- **Baskula:** Produktuak saileka antolatzen ditu (adib. Frutategia, Harategia).
- **Produktua:** Salgai dauden artikuluak, haien prezioa eta dagokien baskula identifikatzen ditu.
- **Saltzailea:** Salmenta bakoitzaren arduraduna den langilea.
- **Tiketa:** Egindako salmenta bakoitzaren erregistro zehatza (data, produktua, kantitatea eta prezioa).

Erlazioak eta Kardinalitateak

1. **Baskula - Produktua ("dauka"):** Baskula bakoitzak (1,1) produktu bat edo gehiago izan ditzake (1,n).
2. **Produktua - Tiketa ("eduki"):** Produktu bakoitza (1,1) hainbat tiketatan ager daiteke (0,n).
3. **Saltzailea - Tiketa ("saldu"):** Saltzaile bakoitzak (1,1) hainbat tiketa kudea ditzake (1,n).

Eredu Logikoa (Eedu Erlazionala)

Normalizazio prozesuaren ondoren, taulak honela definitu dira:

- **BASKULA** (id_baskula PK, Izena).
- **PRODUKTUA** (id_produktua PK, produktuaren_izena, prezioa_kg, id_baskula FK).
- **SALTZAILEA** (Id_saltzailea PK, Izena).
- **TIKETA** (Id_tiketa PK, data, Kantitatea_kg, Prezioa_guztira, id_produktua FK, Id_saltzailea FK).



Implementazio Fisikoa (SQL)

MySQL bidezko implementazioa lau script nagusitan banatu da:

3.1. Taulen Sorkuntza

- **Baskula:** baskula_id (INT, PK) eta izena (VARCHAR) eremuak ditu.
- **Produktua:** id_produktua (PK) eta baskula_id kanpoko gakoa ditu, sailen arteko osotasuna bermatzeko.
- **Saltzailea:** id_saltzailea (PK) eta langilearen izena jasotzen ditu.
- **Tiketa:** Taula konplexuena da. AUTO_INCREMENT bidezko PK-a du eta bi kanpoko gako (CONSTRAINT bidez lotuta) produktuekin eta saltzaileekin. fetxa eremuak CURRENT_TIMESTAMP erabiltzen du automatikoki.

Datuen Adibideak eta Normalizazioa

Datuen lagin bat erabili da sistemaren funtzionamendua egiaztatzeako:

Taula	Adibideak
Produktuak	SAGARRA (20,3€/kg, Baskula 1), URDAIAZPIKOA (18,5€/kg, Baskula 4).
Saltzaileak	1: JON, 2: MIKEL, 3: UNAI.
Tiketak	202601121233: Jonek 13,4kg saldu ditu (20,54€ guztira).



Ondorioak

Datu-basearen egitura honek **osotasun erlazionala** bermatzen du. Kanpoko gakoen erabilerak (FK) ez du uzten existitzen ez den produktu edo saltzaile bati lotutako tiketik sortzen. Gainera, decimal motako datuak erabiltzeak (decimal(20,3)) doitasun handia ematen die pisu eta prezioen kalkuluei.



DIGITALIZAZIOA

HERRIKO SUPERMERKATU TXIKIRAKO DIGITALIZAZIOA

Enpresaren gaur egungo helburu handiena eskuz egiten diren prozesuak digitalizatzea da. Horretarako, Odoo bezalako ERP bat ezartzeak aukera ematen du informazioa zentralizatzeko, departamentu edo atal guztien artean datuak partekatzek. Gainera, informazioa hodeira migratzeak eskuragarritasuna handitzen du (edozein lekutatik sar daiteke), segurtasun-kopiak eta eguneraketak errazten ditu, eta sistemaren eskalagarritasuna hobetzen du, enpresak hazten jarraitzen badu ere.

Digitalizazio-prozesua normalean faseka egiten da, arriskuak murritzeko eta eguneroko lanean etenik ez sortzeko. Lehenengo pausoa diagnostikoa izaten da: zein datu eta prozesu ditugun, nola sortzen diren, eta nora joan behar duten Odoo barruan. Ondoren, datuak prestatu eta garbitu behar dira (produktuen kodeak, izenak, prezioak, BEZA, stock hasierakoa, hornitzailak, bezeroak...), bestela akatsak sistemara eramango direlako. Behin Odoo-ko hodeia konfiguratuta, moduluak aktibatu eta integrazioak prestatzen dira. Azkenik, pilotu bat egiten da atal batekin (adibidez frutategia), fluxu osoa ondo dabilela egiaztatu, eta gero hedapen osoa egiten da. Martxan jarri aurretik, funtsezkoa da langileen prestakuntza eta proben fasea, sistemak benetako egoeretan ondo erantzuten duela ziurtatzeko.



Digitalizazio-prozesua honela antolatuko litzateke:

- 1) **Diagnostikoa eta plangintza:** prozesuak mapatu (baskula → salmenta → ticket/faktura → kontabilitatea), beharrak eta helburuak definitu.
- 2) **Arkitektura eta ingurunea:** Odoo hodeian (Odoo Online/Odoo.sh) aukeratu, erabiltzaileak/rolak eta segurtasuna definitu.
- 3) **Datuен prestaketa:** produktuak, prezioak, BEZak, stock hasierakoa, hornitzazoleak, bezeroak eta kontabilitate oinarria normalizatu.
- 4) **Moduluен konfigurazioa:** biltegi egitura, salmenta konfigurazioa, ordainketa-metodoak, dokumentu txantiloia etab zenbakitzeari.
- 5) **Integrazioak:** POS + baskulak + (beharrak) TicketBai, eta zuen fitxategi/erregistro fluxuen importazioa (txt/XML).
- 6) **Pilotua eta hedapena:** atal batean probatu, akatsak zuzendu, eta gero atal guztietaera zabaldu.
- 7) **Martxan jartzea eta hobekuntza:** prestakuntza, jarraipena, KPI-ak eta txostenen bidezko optimizazioa.

Behin prozesua ulertuta, irudian agertzen diren ERP funtzionalitateetatik zeintzuk erabiliko liratekeen definitzen da. Gure projektuan (supermerkatua + baskulak + tiketak) funtsezkoak dira operatiba egunerokoa eta fiskalitatea ondo lotzen dituztenak: stocka, salmentak, erosketak, kontabilitatea eta txostenak.



Digitalizaziorako erabiliko genituzkeen ERP funtzionalitateak hauek dira:

- **Biltegiratzea eta inbentarioa**
 - Biltegietako stock-a eta kokapenak (atalka: frutategia/harategia/...)
 - Inbentario-zenbaketak eta diferentzien kontrola
- **Salmentak**
 - Bezeroen kudeaketa
 - Salmenta-prozesua
 - Kanpainak / promozioak (deskontuak, eskaintza bereziak)
- **Kontabilitatea eta finantzak**
 - Fakturazioa eta tiketen erregistroa (TicketBai integratzeko oinarria)
 - Gastuak eta bankuen jarraipena
 - Finantza-txostenak (BEZ laburpenak, salmenta/diru-sarrerak)
- **Erosketak**
 - Hornitztaileen kudeaketa
 - Erosketa-eskaerak eta jasotzeak
 - Stock minimoen araberako berrerosketa (automatizazioa)
- **Txostenak eta analisiak**
 - Datuen analisia (salmenta gehien duten produktuak, atal errentagarriak...)
 - Aginte-panelak (KPI: stock hausturak, ticket kopurua, batez besteko saskia...)



JASANGARRITASUNA

Gure proiektua supermerkatu bateko ticketen kudeaketa digitalizatzera bideratuta dago: ticketak .txt formatutik automatikoki irakurri, datu-basean gorde, TicketBAI-rako XMLak sortu eta balidatu, Ogasunera bidali eta web interfazetik kontsultatu ahal izatea. Horrek prozesuak modernizatu eta eraginkorrago bihurtzen ditu, eta, aldi berean, Agenda 2030eko hainbat helbururekin bat egiten du:



7. Energia eskuragarria eta ez-kutsatzailea:

Zuen aplikazioan praktikan jar daiteke: kode eta datu-prozesuen optimizazioa, exekuzio-ordua murriztea, backupen politika adimentsuak, eta hodeiko hornitzale “green” aukeratzea (bideragarritasun txostenean sartzeko).

8. Lan duina eta hazkunde ekonomikoa:

- Enpresak langile espezializatua kontratatuko du prozesua gauzatzeko; gaitasun digitalak eta enplegarritasuna bultzatzen dira.
- Automatizazioak denbora aurrezten du eta negozio txiki baten lehiakortasuna hobetu dezake.

9. Industria, berrikuntza eta azpiegiturak:

- Proiektuak supermerkatuaren prozesu kritiko bat digitalizatu eta azpiegitura teknologiko bat (datu-basea + XML fluxua + weba) ezartzen du.
- Etorkizunerako ERP + hodeirako migrazioa aztertzea ere berrikuntza eta modernizazioa da.

12. Ekoizpen eta kontsumo arduratsuak:

- Ticketak digitalizatzeak paper-kontsumoa murrizten du eta kudeaketa eraginkorragoa egiten du (gutxiago inprimatu, gutxiago artxibo fisiko).
- Gainera, datuetan oinarritutako estatistikekin (salmenta/produktoak) hondakinak eta soberakinak murrizteko aukera dago.

13. Klimaren aldeko ekintzak:

- Papera gutxitzea + prozesuak optimizatzea = aztarna ekologiko txikiagoa (materiala, garraioa, biltegiratzea).
- Hodeira migratzeko bideragarritasuna ebaluatzean, aukera dago energia-eraginkortasuna (hosting “berdeagoa”, optimizazioa) sartzeko.

16. Bakea, justizia eta erakunde sendoak:

- TicketBAI/Ogasunarekin betetzeak gardentasuna, trazabilitatea eta araudia betetzea indartzen du (XMLak balidatuta, erregistroak, ezabatzea debekatuta, etab.).
- Datuen osotasuna, auditoretza eta segurtasun-kopiak bezalako neurriak helburu honekin oso lotuta daude.