

Schema Waterfall a aplicației FiscontExpert Consult

Versiune document: 2 iulie 2026

Schema Waterfall a aplicației FiscontExpert Consult

Versiune document: 2 iulie 2026 Aplicație: FiscontExpert Consult - site public, portal client și administrare Scop: model clasic Waterfall pentru proiectare, implementare, testare, lansare și mentenanță.

1. Ce este abordarea Waterfall pentru această aplicație

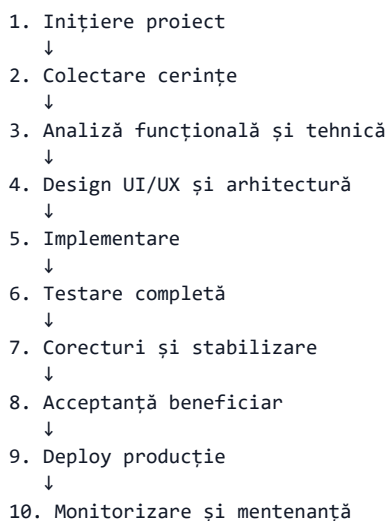
Waterfall este o metodologie secvențială. Fiecare fază trebuie finalizată și aprobată înainte de trecerea la faza următoare.

În acest proiect, Waterfall este util mai ales pentru:

- documentație;
- conformitate GDPR;
- cerințe contractuale;
- validare client;
- lansare pe hosting;
- proiecte cu cerințe clare și livrabile formale.

Waterfall nu este la fel de flexibil ca Scrum, dar oferă trasabilitate bună: cerință → analiză → design → implementare → testare → livrare → mentenanță.

2. Schema generală Waterfall



3. Fazele Waterfall

Faza 1 - Inițiere proiect

Scop:

- stabilirea obiectivului general;
- definirea aplicației;
- stabilirea persoanelor implicate;
- stabilirea livrabilelor.

Intrări:

- cerința beneficiarului;
- site-ul original;
- acces local/hosting;
- scopul comercial și administrativ.

Ieșiri:

- scop proiect;
- listă module;
- listă riscuri inițiale;
- plan de lucru.

Gate de aprobare:

Beneficiarul confirmă direcția proiectului și modulele principale.

Faza 2 - Colectare cerințe

Scop:

- documentarea completă a ce trebuie să facă aplicația.

Cerințe colectate:

- reproducere site public;
- backend de administrare;
- portal client;
- utilizatori creați de admin;
- firme asigurate utilizatorilor;
- documente;
- analiză financiară;
- analiză economică;
- analiză SWOT;
- analiză GDPR;
- integrare Oblio, SmartBill, TDec, WinMentor, Nexus, Saga;
- SEO avansat;
- Google Analytics;
- Google Ads/AdSense;

- GDPR/cookie;
- plăți;
- chat intern;
- email marketing;
- Cloudflare;
- retenție legală;
- backup/restore;
- template-uri;
- mentenanță;
- responsive.

leșiri:

- backlog funcțional;
- backlog tehnic;
- criterii de acceptare;
- manuale inițiale.

Gate de aprobare:

Cerințele sunt suficient de clare pentru proiectare.

Faza 3 - Analiză funcțională și tehnică

Scop:

- transformarea cerințelor în specificații tehnice.

Activități:

- analiză pagini publice;
- analiză module admin;
- analiză portal client;
- analiză fluxuri autentificare;
- analiză structură bază de date;
- analiză securitate;
- analiză GDPR/DPIA/GAP;
- analiză integrări externe;
- analiză hosting cPanel;
- analiză limitări.

leșiri:

- schema bazei de date;
- structura rutelor;
- fluxuri de user/admin;
- lista funcționalităților complete/parțiale;

- lista riscurilor tehnice.

Gate de aprobare:

Arhitectura și fluxurile sunt acceptate înainte de implementare.

Faza 4 - Design UI/UX și arhitectură

Scop:

- stabilirea aspectului și structurii aplicației.

Design public:

- header;
- meniu;
- steaguri limbi;
- buton portal clienți;
- hero;
- secțiuni homepage;
- carduri servicii;
- noutăți;
- footer;
- pagini interne;
- contact;
- pagini legale.

Design admin:

- sidebar;
- conținut scrollabil;
- module;
- formulare;
- carduri statistici;
- editor pagini;
- tabele;
- acțiuni.

Design portal:

- dashboard client;
- documente;
- analiză;
- cont propriu;
- avertismente plată.

Arhitectură:

- Next.js;
- Prisma;
- MySQL/MariaDB;
- server custom cPanel;
- storage documente;
- setări globale în DB;
- build precompilat.

Gate de aprobare:

Beneficiarul validează aspectul și structura înainte de implementare masivă.

Faza 5 - Implementare

Scop:

- dezvoltarea aplicației.

Module implementate:

- site public;
- multi-limbă;
- template-uri;
- admin;
- portal;
- autentificare;
- 2FA;
- recuperare parolă;
- recuperare cont;
- utilizatori;
- companii;
- documente;
- analiză financiară/economică/SWOT/GDPR;
- GDPR/DPIA/GAP;
- cookie consent;
- SEO/Ads;
- email marketing;
- chat intern;
- plăți configurabile;
- integrări API ca stocare credentiale;
- easter egg;
- mentenanță;
- backup/restore;
- ștergeri GDPR cu retenție.

leșiri:

- cod sursă;
- bază de date;
- build local;
- arhivă prebuilt;
- manuale.

Gate de aprobare:

Funcțiile principale sunt implementate și pot intra în testare completă.

Faza 6 - Testare completă

Scop:

- verificarea funcțională, vizuală, tehnică, securitate, GDPR și responsive.

Tipuri de test:

- testare funcțională;
- testare UI;
- testare responsive;
- testare cross-browser;
- testare autentificare;
- testare roluri;
- testare admin;
- testare portal;
- testare documente;
- testare SEO;
- testare GDPR;
- testare performanță;
- testare hosting/deploy.

Gate de aprobare:

Nu există erori critice sau blocante.

Faza 7 - Corecturi și stabilizare

Scop:

- corectarea problemelor găsite la testare.

Reguli:

- nu se modifică ce funcționează corect;
- se corectează punctual;
- fiecare fix este retestat;

- se actualizează manualele dacă se schimbă comportamentul.

Gate de aprobare:

Toate bugurile critice sunt închise sau acceptate ca limitări documentate.

Faza 8 - Acceptanță beneficiar

Scop:

- beneficiarul validează aplicația.

Checklist acceptanță:

- site public arată corect;
- admin funcționează;
- portal funcționează;
- login/reset/2FA funcționează;
- documentele funcționează;
- analiza se generează;
- raportul GDPR rulează;
- template-urile se schimbă;
- plățile sunt documentate ca parțiale dacă gateway-ul nu este implementat;
- integrările sunt documentate ca stocare credentiale dacă sync-ul nu este implementat;
- manualele există.

Gate de aprobare:

Beneficiarul aprobă livrarea sau cere corecturi finale.

Faza 9 - Deploy producție

Scop:

- urcarea aplicației pe hosting.

Pași:

- 1 backup local;
- 2 build local;
- 3 generare arhivă prebuilt;
- 4 backup live;
- 5 upload arhivă;
- 6 configurare variabile cPanel;
- 7 `npm install`;
- 8 `npm run db:generate`;
- 9 `npm run db:push` doar dacă schema s-a schimbat;
- 10 restart Node app;
- 11 verificare rute critice;

12 dezactivare mentenanță.

Gate de aprobare:

Site-ul live răspunde corect și logurile nu au erori critice.

Faza 10 - Monitorizare și mentenanță

Scop:

- menținerea aplicației funcționale.

Activități:

- backup periodic;
- verificare uptime;
- verificare loguri;
- actualizare manuale;
- rulare GDPR/DPIA/GAP periodic;
- testare după modificări;
- actualizare dependențe controlat;
- verificare securitate.

Gate periodic:

Aplicația rămâne stabilă și conformă.

4. Livrabile Waterfall

Fază	Livrabil
Inițiere	Scop proiect, module, riscuri inițiale
Cerințe	Cerințe funcționale și tehnice
Analiză	Arhitectură, fluxuri, DB, rute
Design	UI public/admin/portal, responsive
Implementare	Cod, DB, module, build
Testare	Raport testare, bug list
Stabilizare	Fixuri, retestare
Acceptanță	Confirmare beneficiar
Deploy	Site live, checklist deploy
Mentenanță	Backup, loguri, rapoarte periodice

5. Avantaje Waterfall pentru acest proiect

- documentație clară;
- bun pentru conformitate;
- bun pentru aprobări formale;

- ușor de urmărit ce s-a livrat;
- potrivit pentru audit GDPR;
- potrivit pentru hosting/deploy controlat.

6. Dezavantaje Waterfall pentru acest proiect

- schimbările târzii sunt costisitoare;
- nu se potrivește perfect când designul se ajustează vizual frecvent;
- feedbackul apare târziu dacă nu se fac demo-uri intermediare;
- riscă să blocheze dezvoltarea dacă cerințele nu sunt clare.

Recomandare:

Folosește Waterfall pentru documentație, audit, deploy și conformitate.

Folosește Scrum pentru dezvoltare iterativă și ajustări vizuale/funcționale.