

Termeni de Referință
Front End & Python Developer (junior)

1. Informații despre poziție

Oficiu / Unitate / Proiect	Ministerul Agriculturii și Industriei Alimentare (MAIA)
Titlul poziției	Front End & Python Developer (junior)
Locația (oraș și țară)	Chișinău / Republica Moldova
Tip (regular sau pe termen scurt)	Pe termen scurt
Regim de lucru	Hibrid (birou + de la distanță)
Data estimată de începere	17 august 2026
Durata estimată	4,5 luni

2. Descrierea proiectului (context)

Ministerul Agriculturii și Industriei Alimentare (MAIA) anunță selectarea unui Front End & Python Developer (junior) care va contribui la dezvoltarea și modernizarea ecosistemului digital pentru agricultură (IACS/dNFR) al Republicii Moldova, ce include Registrul Fermierului (RF) și platforme conexe de interacțiune cu fermierii și operatorii.

Poziția este una de nivel junior, potrivită pentru un dezvoltator la început de carieră, dornic să învețe și să se dezvolte într-o echipă tehnică ce lucrează la sisteme de importanță națională. Dezvoltatorul selectat va contribui la realizarea interfețelor de utilizator ale subsistemelor IACS/dNFR, la integrarea acestora cu serviciile back-end și registrele guvernamentale (MPass, MConnect, MTender, cadastru, ortofoto), precum și la dezvoltarea de componente de procesare și validare a datelor în Python.

3. Obiective

Principalele obiective ale misiunii de dezvoltare sunt:

- Dezvoltarea și întreținerea interfețelor web (React.js / Next.js) și mobile (React Native) pentru subsistemele IACS/dNFR, conform cerințelor funcționale (BRD/FSD) și a standardelor de utilizabilitate.
- Implementarea componentelor web-GIS interactive (vizualizare straturi, parcele agricole, instrumente de măsurare și editare) în colaborare cu echipa.
- Dezvoltarea de servicii și scripturi Python pentru procesarea, validarea și transformarea datelor spațiale și tabelare.
- Integrarea interfețelor cu API-urile REST/GraphQL ale back-end-ului și cu serviciile de e-guvernare (MPass, MConnect).
- Asigurarea calității codului prin testare automată, respectarea bunelor practici și a arhitecturii stabilite (ex. Feature-Sliced Design).

4. Domeniul de activitate (Scope)

Dezvoltatorul Front End & Python (Junior) va contribui la următoarele activități:

4.1. Dezvoltare front-end web și mobil

- Implementarea interfețelor de utilizator web folosind React.js / Next.js și TypeScript, conform machetelor de design și a specificațiilor funcționale.
- Dezvoltarea aplicației mobile (React Native) pentru acces de teren al operatorilor și fermierilor, cu suport pentru funcționalități offline acolo unde este necesar.
- Construirea de componente reutilizabile și a unei biblioteci de UI consistente (design system), cu respectarea principiilor responsive și mobile-first.
- Implementarea state management-ului (Redux Toolkit / React Query / Context API) și a fluxurilor de date între componente.
- Asigurarea accesibilității (a11y) și a internaționalizării (i18n) interfețelor, cu suport pentru limba română și, după caz, alte limbi.

4.2. Componente web-GIS

- Integrarea și randarea hărților interactive (ex. Leaflet, OpenLayers, MapLibre sau echivalent) pentru vizualizarea parcelelor agricole și a straturilor GIS.
- Implementarea instrumentelor de interacțiune cu parcelele: vizualizare, selectare, măsurare, evidențierea suprapunerilor și a erorilor topologice (în colaborare cu echipa GIS).
- Consumul și afișarea datelor spațiale din servicii (WMS/WFS, GeoJSON, vector tiles) și sincronizarea cu back-end-ul.
- Optimizarea performanței de randare pentru seturi mari de date spațiale (lazy loading, clustering, virtualizare).

4.3. Dezvoltare și integrare Python

- Dezvoltarea de servicii și endpoint-uri în Python (ex. FastAPI / Flask / Django) pentru suportul funcționalităților front-end, acolo unde este necesar.
- Implementarea scripturilor de procesare, validare și transformare a datelor (inclusiv date spațiale, ex. cu GeoPandas/Shapely), conform regulilor de business definite.
- Integrarea cu bazele de date relaționale (PostgreSQL/PostGIS, MySQL, MsSQL) și cu API-urile back-end existente.
- Dezvoltarea de utilitare pentru import/export de date și interoperabilitate cu registrele guvernamentale (cadastru, RF, ortofoto).

4.4. Integrare cu servicii și autentificare

- Integrarea autentificării și autorizării cu platformele de e-guvernare (MPass) și cu fluxurile de schimb de date (MConnect), conform standardelor aplicabile.
- Implementarea fluxurilor de autentificare securizată (JWT cu reîmprospătare automată a token-ului, OAuth după caz) și a gestionării sesiunilor.
- Consumul și abstractizarea serviciilor REST/GraphQL printr-un strat tipizat de acces la date.
- Integrarea notificărilor (push/email) și a comunicării în timp real (WebSockets/SSE) unde este aplicabil.

4.5. Calitate, testare și livrare

- Scrierea testelor unitare și de integrare (Jest, React Testing Library) și a testelor E2E de bază (Cypress/Playwright).
- Respectarea standardelor de cod (ESLint, Prettier, convenții de proiect) și participarea la code review.

- Configurarea și menținerea pipeline-urilor CI/CD pentru build, testare și deployment automatizat.
- Documentarea componentelor, a API-urilor consumate și a deciziilor tehnice relevante.

4.6. Colaborare și suport

- Colaborarea cu analistul de business pentru clarificarea cerințelor și a criteriilor de acceptanță.
- Participarea la ceremoniile Agile (sprint planning, grooming, review, retrospective).
- Suport pentru fazele de UAT: remedierea defectelor identificate și ajustarea funcționalităților.
- Transfer de cunoștințe către echipa MAIA și predarea documentației tehnice.

5. Livrabile (Deliverables)

5.1. Componente front-end

- Interfețe web (React.js/Next.js) funcționale pentru modulele IACS/dNFR atribuite, conform BRD/FSD.
- Aplicație mobilă (React Native) cu funcționalitățile solicitate, testată pe iOS și Android.
- Bibliotecă de componente UI reutilizabile și documentația aferentă (design system).

5.2. Componente web-GIS

- Module de vizualizare și interacțiune cu hărțile și parcelele agricole, integrate cu serviciile GIS.
- Instrumente de măsurare și evidențiere a erorilor topologice, conform cerințelor.

5.3. Servicii și scripturi Python

- Servicii/endpoint-uri Python pentru suportul funcționalităților, cu documentație API (OpenAPI/Swagger).
- Scripturi de procesare, validare și transformare a datelor spațiale și tabelare.
- Utilitare de import/export și integrare cu registrele guvernamentale.

5.4. Calitate și livrare

- Suite de teste automate (unitare, integrare, E2E de bază) cu acoperire pentru funcționalitățile critice.
- Pipeline CI/CD configurat și funcțional (build, test, deployment).
- Codul sursă versionat în repository-ul proiectului, cu istoric curat și convenții respectate.

5.5. Documentație și transfer de cunoștințe

- Documentație tehnică: arhitectura front-end, componente, fluxuri de date, instrucțiuni de build și deployment.
- Raport final de predare-primire, incluzând sumarul activităților, livrabilele predate și recomandările pentru etapele următoare.

6. Cerințe pentru candidat (calificări și experiență)

Studii și certificări

- Studii superioare (în curs sau finalizate) în informatică, tehnologii informaționale, ingineria software sau un domeniu conex.
- Certificări relevante în dezvoltare web/front-end sau cloud – constituie avantaj.

Experiență și cunoștințe (nivel junior)

- Cunoștințe de dezvoltare front-end cu React.js; familiaritatea cu React Native constituie avantaj. Se acceptă proiecte academice, personale, de tip internship sau open-source ca dovadă a competențelor.
- Cunoștințe de bază în programare Python (scripturi, procesarea datelor).
- Noțiuni privind integrarea cu API-uri REST/GraphQL și principiile de autentificare (JWT/OAuth) – constituie avantaj.
- Orice experiență cu componente GIS/spațiale (hărți interactive, date geospațiale) constituie un avantaj major.
- Experiența în proiecte guvernamentale, registre publice sau sisteme de agricultură constituie avantaj.

Competențe tehnice și personale

- Limbaje și fundamente: JavaScript (ES6+), TypeScript, HTML5, CSS3, Python; cunoștințe C/C++ – avantaj.
- Framework-uri și biblioteci: React.js, Next.js, React Native; cunoștințe Vue.js – avantaj.
- State management: Redux/Redux Toolkit, React Query, Context API; styling: Tailwind CSS, Styled Components.
- Baze de date: PostgreSQL/PostGIS, MySQL, MsSQL, Oracle (noțiuni).
- Testare și tooling: Jest, React Testing Library, Cypress/Playwright, Vite/Webpack, ESLint/Prettier, Git.
- Arhitectură și pattern-uri: Feature-Sliced Design (FSD), custom hooks, monorepo, principii SOLID.
- Abilități de comunicare în limba română; cunoașterea limbii engleze la nivel de lucru (minim B2).
- Capacitate analitică, atenție la detalii și abilitatea de a lucra în echipă în metodologii Agile (Scrum/Kanban).
- Motivație de a învăța continuu, atitudine proactivă și deschidere către feedback și mentorat (esențial pentru nivelul junior).

7. Durata și locația

- Durata: 5 luni; recomandare de etapizare: configurare mediu și înțelegere context (1–2 săptăm.), dezvoltare iterativă a componentelor front-end și GIS (continuu, pe sprinturi), dezvoltare servicii Python și integrări (continuu), testare și suport UAT (3–4 săptăm.), predare finală și transfer de cunoștințe (1–2 săptăm.).
- Locație: Chișinău (sediul MAIA) + remote (hibrid); prezență periodică la sediu pentru sincronizări și sesiuni cu echipa.

8. Raportare și supervizare

- Raportare directă către Managerul de Proiect MAIA / liderul tehnic al proiectului.
- Activitatea se desfășoară sub îndrumarea și cu suportul liderului tehnic și al dezvoltatorilor seniori (mentorat adaptat nivelului junior).
- Ședințe săptămânale de sincronizare cu echipa; participare la ceremoniile de sprint (metodologie Agile).
- Rapoarte bilunare de progres (funcționalități livrate, defecte remediate, riscuri și blocaje).
- Demonstrații (demo) la finalul fiecărui sprint/fază cheie către echipa de proiect și stakeholderi.

9. Criterii de acceptanță a livrabilelor

- Funcționalitățile livrate corespund cerințelor din BRD/FSD și criteriilor de acceptanță definite.
- Interfețele sunt responsive, accesibile și funcționează corect pe browserele și dispozitivele țintă (web + iOS/Android).
- Codul respectă standardele proiectului (ESLint/Prettier, convenții), trece testele automate și este aprobat la code review.
- Componentele web-GIS afișează și procesează corect datele spațiale, cu performanță acceptabilă pe seturi mari de date.
- Serviciile/scripturile Python procesează și validează datele conform regulilor de business, cu documentație aferentă.
- Documentația tehnică este completă, iar transferul de cunoștințe către echipa MAIA este realizat.

10. Condiții de plată

Plata se va efectua pe bază de milestones, conform următoarelor etape orientative:

- Milestone 1 – Configurarea mediului, plan de lucru detaliat și primele componente UI de bază (luna 1).
- Milestone 2 – Module front-end web funcționale și integrare cu API-urile back-end (luna 2–3).
- Milestone 3 – Componente web-GIS și aplicația mobilă (React Native) funcțională (luna 3–4).
- Milestone 4 – Servicii/scripturi Python, integrări (MPass/MConnect) și suite de teste (luna 4–5).
- Milestone 5 – Suport UAT finalizat, documentație și raport final de predare-primire (luna 5).

11. Procesul de aplicare

Candidații interesați trebuie să transmită următoarele documente:

- CV actualizat, evidențiind experiența relevantă în dezvoltare front-end (React/React Native), Python și, după caz, proiecte GIS/guvernamentale (inclusiv proiecte academice, personale sau open-source).
- Scrisoare de intenție, cu descrierea abordării tehnice propuse pentru proiect.
- Portofoliu / link-uri către proiecte anterioare relevante (ex. repository GitHub, aplicații live), dacă sunt disponibile.
- Oferta tehnică/ financiară.

Elaborat:

Aprobat: