Template from www.lineupx.com/job-descriptions/full-stack-developer. Replace every highlighted [field] and keep the version you need: the full-time job for freshers or the internship.

Full stack developer (fresher)

Company: [Company name]
Location: [Location] | Work mode: [Work mode]
Employment type: Full-time | CTC: [CTC for the full-time role]
Joining: [Joining date] | Apply by: [Apply by]

About [Company name]

[About your company]

About the role

Join the [Team] team at [Company name] as a fresher full stack developer. You will build features across the whole product: the screens people use, the APIs behind them and the database that stores the data. You will start with small, well-defined tasks, work with a senior developer who reviews your code, and take on more of the stack as your confidence grows. It suits a graduate who enjoys seeing a feature work from browser to database.

What you will do

- Build and update pages and components in the team's front-end framework, such as React or Angular, from designs or wireframes, including forms, validation and error messages
- Create and change REST API endpoints on the server, covering routes, input validation and responses, with a senior reviewing each change
- Design or change simple database tables or collections, write the queries behind them and apply schema changes through the team's migration process
- Connect screens to the API and handle loading, empty and failure states, so users see a clear message instead of a blank or broken page
- Add login checks and permission checks to new endpoints using the team's existing token or session approach
- Trace a bug across the layers: reproduce it in the browser, inspect the network request, read the server log and check the stored data, then fix it at its source
- Write unit tests and API tests for the code you change, and run both the front-end and back-end suites before raising a pull request
- Keep secrets out of source control and browser code, and use the team's approved server-side configuration
- Deploy changes to the test environment through the team's pipeline and confirm they work there before release
- Show finished features to the product owner or tester in a demo and act on the feedback

What we are looking for

- HTML, CSS and modern JavaScript, including calling an API and handling the response with promises or async functions
- One front-end framework or library, such as React or Angular, used in a project you can show
- One back-end language and framework, such as Node.js with Express, Spring Boot or Django, used to build REST endpoints
- Working knowledge of one database, either SQL (MySQL or PostgreSQL) or NoSQL (MongoDB), including designing a simple schema

- Understanding of how the web works: HTTP methods and status codes, JSON, and how cookies or tokens keep a user logged in
- Git and the pull request workflow
- One end-to-end project with a front end, an API and a database that you can run or open live and explain layer by layer

Good to have

- Basic deployment experience, such as hosting a project on a cloud service or a Linux server
- TypeScript
- API testing with Postman and unit testing with Jest or a similar tool
- Basic Docker
- Experience with an ORM or an authentication library

Who can apply

Final-year students and graduates of B.E./B.Tech in computer science, IT or related streams fit well, as do BCA, B.Sc (computer science or IT), MCA and M.Sc candidates. Self-taught developers and candidates from other streams are welcome when they have a working end-to-end project.

How to apply

Apply through this job post by [Apply by]. Shortlisted candidates will hear from us about the next steps.

Full stack developer intern

Company: [Company name]
Location: [Location] | Work mode: [Work mode]
Employment type: Internship | Duration: [Internship duration] | Stipend: [Stipend for the internship]
Joining: [Joining date] | Apply by: [Apply by]

About [Company name]

[About your company]

About the role

[Company name] is looking for a full stack developer intern to work with a mentor on real product features. You will build small pieces across screens, APIs and the database, learn how the team reviews and tests changes, and finish with an end-to-end feature you can demonstrate.

What you will do

- Build a small page or form from a design and connect it to an existing API
- Add one simple API endpoint with input validation and a matching test
- Write database queries or a small schema change, reviewed by your mentor
- Fix bugs from the issue list and note which layer each one came from
- Write down how to run your feature locally, so another intern can follow the steps
- Demonstrate the finished feature in the final week

What we are looking for

- Studying computer science, IT or a related subject, or a recent graduate
- HTML, CSS and JavaScript, plus one back-end language
- Has built at least one small project with a login or data entry form that stores data
- Knows basic Git commands
- Free to work through the whole internship without clashes with exams

Who can apply

Final-year students and graduates of B.E./B.Tech in computer science, IT or related streams fit well, as do BCA, B.Sc (computer science or IT), MCA and M.Sc candidates. Self-taught developers and candidates from other streams are welcome when they have a working end-to-end project.

How to apply

Apply through this job post by [Apply by]. Shortlisted candidates will hear from us about the next steps.

Screening questions to add to your job post

1. **Describe one feature you built from the screen to the database. Which parts did you write yourself?** Shows whether the candidate has worked across layers and which layer is strongest. A vague answer about ‘the whole project’ needs a follow-up.
2. **Your page shows an empty list, but the same API call returns data in Postman. What do you check?** Tests debugging across layers. Good answers mention the browser network tab, the shape of the response, CORS and how the front end stores the data.
3. **How does your project keep a user logged in, and how are passwords stored?** Reveals awareness of hashing and of tokens or sessions. Plain-text passwords in a college project are a teaching point, not an automatic rejection.
4. **How did you design the tables or collections for your project, and what would you change now?** Shows data modelling judgement and honest reflection on past choices.
5. **Share a live link or repository for your project, or describe it if the work is private. How would someone run it?** A working link or a clear README is the strongest evidence. If the work is private, accept a walkthrough and ask the candidate to show how it runs.
6. **Which layer is your weakest, and what are you doing about it?** Honest self-assessment helps you place the fresher and plan coaching. A claim of equal strength everywhere is a warning sign.