



Demo request paths showcasing the application's functionality:

A client requests `www.avouris.com` on the DNS and receives an IP from the A records resolving to the load balancer. The load balancer then asks the most available web server for the static content.

---User requests READ---

A user clicks on "Search Jobs" and enters "Patras" as location, a query is made through the read API to the backend to get all the results. The results are first cached in redis and then served to the user.

---User requests WRITE---

A company requests a job listing be posted on the site or an employee lists an already requested:

1) The employee logs into the application by sending a request to the authentication API, which takes his credentials and asks AuthDB for a user with that username/password combination. On success, it fetches the user info with his ID as the key from the PostgreSQL Backend and handles the privilege escalation. Then receives all job listings available from the backend database and once he creates a new job listing through the write API, the `IsActive` flag switches on (opposite for deletes obviously).

2) The company XYZ logs into their business account (see above for logging in) asks for a job listing under "Full-stack Developer" and creates an issue (similar to a gh pr) with all the information on the job. The request is then cached with Redis and is available for the employee to accept (without need for comms with the backend) and then written to the PostgreSQL Backend with the `IsActive` flag unset by default.