CoverLetter

To whom it may concern,

Over the last 25 years I have worked primarily as a software developer doing backend web development. Additionally, I have worked as a test automation engineer, systems administrator and technical writer. While creating REST APIs for large variety of software systems, I discovered a passion for maintaining the accuracy and usability of the documentation accompanying these APIs. I am presently searching for a new role that will allow me to leverage both my experience as a backend dev and my passion for creating highly usable documentation.

Writing Samples

Here's a quick inline sample of some engineering writing:

```
Accumulating Peak Connections Data Across Nodes into 5 Minute Buckets

Since stats data will come in at 30 second intervals and the intervals
are not guaranteed to have aligned timestamps across nodes, we must make an
approximation of overall peak connections based on available data. If we
assume that the number of concurrent connections is reasonably well distributed
across nodes by the load balancer, we can capture a reasonable peak total
connections estimate by having the worker process run at a frequency that will
cover a small but nonzero number of readings from each node in redis (e.g. 2
minute interval gets approximately 4 entries for each node). At each interval,
we find the maximum peak connections for each node over that small period and
combine those maximums for an approximate peak total connections over the period
spanned by all the timestamps currently in redis. This peak connections number
is then compared to any previously recorded peak connections already in the
relevant 5 min time bucket and updated with the maximum value.

Such an approximation could be slightly higher than the actual peak connections
but should never underestimate the peak connections. Furthermore, this estimate
provides a good analogue for the actual expected enforcement behavior since the
type of connection load that would contribute to overestimation (uneven, bursty
connection load) is the same type of connection load that could lead to
admission control limits being enforced when the total connection count is
actually under the limit (i.e. if individual node got more than its share of
connections).
```

And here are a few more historical samples:

- Product Documentation:
  - 3rd Rail
- Internal Engineering:
  - Customer Visible Optimization Status
  - Kernel Crashdumps
- Academic:
  - Thesis Excerpt
  - Peer to PCAST
  - Metavid.org