PRASANNA NATARAJAN

prasanna@npras.in | https://npras.in | +91 7708087998 | Tamilnadu, India

PROFESSIONAL SUMMARY

To benefit the business I'm working with by increasing their revenue, and/or decreasing their operational cost by leveraging my technical, people (and AI) skills.

SKILLS

I'm a Ruby on Rails developer with 17 years of total and 12+ years of relevant experience.

I can design cloud-scale infrastructure, develop great web apps with Rails, and deploy them in the cloud with modern tools.

I'm a full-stack developer with expertise not only in Ruby, Rails and Javascript, but also in the other technologies surrounding this ecosystem like SQL, Git, Shell Programming, Unix, Linux server Configuration/Provisioning, HTML, CSS, AWS, Docker etc.

I also have deep hands-on expertise in several AWS services, and devops skills and tools.

I'm aware of agile approach to software development and have been part of many sprint planning and scrum calls.

I'm a constant learner and keep myself updated with new technologies as well as revisiting foundational skills to keep myself sharp. I've been learning Golang, advanced postgresql and OpenBSD this past few years. My current side project is rubyexamples.com built with Go.

My current role is the equivalent of Senior Staff Engineer, but I've led teams in the past and have worked in the capacity of a Lead Engineer or a Product Owner.

WORK HISTORY

Lookout Inc | Staff Software Engineer IC4 (*Nov 2019 - Dec 2021, Jun 2023 - Current)*

This is the same client and product mentioned below (Altran). After quitting Altran, I worked for Lookout directly. My earlier responsibilities continued in addition to new ones.

I own the push notification and billing services fully. Besides, as the only experienced ruby dev in our team, I also handle work for many related ruby services that have been recently orphaned due to internal re-org. My work scope includes several technical areas, of which some highlights:

- Did upgrades on services at multiple levels: rails/ruby/jruby version upgrades, OS versions upgrades, database engine upgrades, EC2 instance type changes (x86 to graviton), paid third-party library (sidekiq) upgrades.
- Routinely did security related critical tasks that involve both designing and implementing ideas involving tls certs, api keys, customer data management, application secrets encoding, government compliance (fedramp, fips), restricting server traffic/access
- Experience with multiple debugging tools to find and squash any kind of bugs - language specific tools (pry, byebug, monkey-patching), OS/unix/linux specific tools (/etc/proc, strace, journalctl), git specific tools (git bisect/blame/logs), db tools (explain analyze, vacuum) and the typical “read code and available documentation”
- Did several re-architecting for the sole purpose of reducing AWS costs (created postgresql views for high-traffic api calls, de-coupled web and worker processes into separate ec2 instances, simplified several rounds of api calls to just 2, merged a smaller java service into the ruby monolith
- Created the entire infrastructure necessary to deploy my application in a brand new AWS region (involving, dns, elbs, declarative deployment pipelines, security groups, server sysadmin, routing traffic through proxies etc)
- Write adhoc scripts to cleanup production DB and run arbitrary tasks
- Besides these, I teach my team members advanced ruby and coding design patterns on a regular basis

Altran | Software Engineer *(Jan 2018 – Nov 2019)*

I worked here for a client who is a US-based mobile security app (Lookout), one of the industry’s top. I’m part of the team that handles the billing part of the app. So far, I’ve worked on various feature development, bug fixes, infrastructure redesign etc. The bug fixes and feature enhancements easily account for millions in revenue.

- I was exposed to various billing mechanisms like Google InApp, Apple Pay, Braintree credit-card processing mechanisms.
- The main app depends on a lot of gems built in-house that all talk with various telecom and billing providers' api. I've worked on all of them.
- Implemented a discount mechanism from scratch
- Coded new pages and functionalities for customer-facing APIs, as well as the admin backend for internal use
- Fixed lots of bugs doing archeology in codebase and the company's wiki, and reaching out to other departments, talking to senior devs who're like librarians keeping alive some ancient lore.
- Used many tools like ELK Kibana (for log spelunking), Datadog (tracking metrics, creating dashboards/charts), Athena (querying prod data), Sentry (error management)
- Upgraded config of our jenkins and spinnaker pipeline to deploy our rails app to newer Ubuntu version. Used docker to test it locally and wrote handful of bash scripts.
- Built debian package for downloading and installing rvm offline in the server.
- Contributed to the company's wiki by documenting things I thought might be useful for others.
- As part of a remote team, I interact with people from US and Ukraine on a day to day basis via slack/email/zoom.
- Used and configured Spinnaker to deploy the rails project to production servers.
- SPOC for the service mentioned above (billing related app). So I'm involved in conversations with PM, Sales team and other team members and stakeholders.

Mozo | Software Engineer at Zanec *(Mar 2017 – Jan 2018)*

https://mozo.com.au/ is Australia's leading financial comparison and review site. I was also responsible for:

- leading the remote team at Zanec to collaborate with Mozo's technology director and his team. We are 2 devs and 1 designer.
- managing project delivery. Mozo will usually just give specs for any new feature they want developed, and set a deadline. It's my responsibility to get my team to build and deliver the project successfully with as little help as possible.
- Writing ruby and javascript code that the daily tasks require.
- The main projects I worked on for this client were:
- A new React app (backed by Rails) for Mozo's partner Choice.com.au. I learnt react for this and developed all of the react and ruby code. Things used in react: redux, saga, multiselect, redux-form, css modules and webpack.
- Built from scratch a custom CMS tool that's used for creating marketing articles. It's designed to create articles of various pre-defined "templates". An example article created from this tool: https://mozo.com.au/personal-loans/articles/get-money-ready-for-christmas-with-a-low-cost-personal-loan

- Built another React app for Mozo's backoffice admin site. It was a complex redux form that fetched data remotely and rendered many components conditionally based on business logic.

Tata Communications' Sponsored Data Exchange *(Velankani 2016 – 8 months)*

This was an exciting project that I worked on. In a span of 4 months, I implemented 3 projects (2 rails, 1 ruby script), designed a cloud-scale infrastructure, and then deployed them to the cloud using modern devops tools.

I lead a team of 3, did over 80% of the design and development. I was responsible for client communication for requirement gathering and status reporting.

In the first phase, I built the 2 Rails apps that exposed many APIs. I used Rails 5. Based on the client's high-level model requirements, I designed the low-level database and entities with relationships. I had to take care of all query performances too. I wrote test cases for all APIs and made sure the test coverage is over 90%. I also used rubo-cop gem to enforce a coding standard among all members of the team.

For billing requirements, the client wanted to use a NoSql DB too, apart from the traditional Mysql RDBMS. We chose AWS's DynamoDB, and implemented the right partition and sort keys that would prove crucial to building the RubyScript app later on. In the second phase, I designed and deployed the infrastructure required to scale this project in the future, in AWS.

- I used Opsworks to setup these 3 app in 3 different stacks.
- I wrote custom Chef recipes, when Opsworks' in-built Chef Recipes fell short.
- I defined a secure architecture where these apps would run within AWS: I created
- VPC,
- private and public subnets,
- route tables and routes to define traffic among them,
- NAT Gateway and Internet Gateway to enable the network to talk to internet,
- Security Groups to control access to ports,
- Elastic Loadbalancers to distribute traffic among multiple instances running the same application,
- Cloudwatch metrics and alarms to watch important KPIs
- Setup Loadbased Autoscaling instances that would come to life when the load on app instances go beyond a certain threshold
- Defined IAM Users and Roles to provide/restrict access to these resources
- In the third phase, I built a Cloudformation template that codified everything I manually did above. This template was the final delivery of this project. With this template, the client will be able to deploy these 3 apps, along with the secure

infrastructure defined above, in any AWS account, in any region across the world.
- The client Tata and Velankani were impressed with the work I did here, especially about how I was able to dive deep into new areas like most of the AWS Services. My suggestions about improving/changing some already-documented specs were also noted positively.

Numerex IoT *(Velankani 2016)*

From Rubyeffect, I went to Velankani as a Contract-hire. Numerex was a big project. They were championing IoT in various domains. We were set to take over the Rails development work from the Onshore team that was managing it previously.

I lead a team of 3. During the initial phases of the project, I would gather the team together and we will together walk through any particular component of a project. I would show them many ways to navigate around the codebase quickly, explore certain classes and libraries, and share with them testing and debugging techniques.

With limited time, documentation and knowledge sharing, we had to reverse engineer and build custom internal tools to simulate the devices' behaviour, and to help our testing team verify our implementations.

It was a set of 5 Rails applications. They were using Rails 4 and Ruby 2.x.
In the Rails apps, we followed best practises and wrote elaborate ruby test cases to verify our implementation. The client was happy with the level of details, and quickly trusted our abilities.

We were tasked at building many features for these apps using the Agile methodology.

ParkMobile USA *(Rubyeffect, 2015 – 9 months)*

Parkmobile http://us.parkmobile.com/ is the leading parking solution provider in USA.
We (3 members) worked for them for 8 months.
We worked on new feature requests and fixing bugs. I remember 2 interesting tasks that I specifically worked on:

1) Fixing memory leak:
The production rails app, which was deployed in 4 servers would have its memory rise high whenever there was more of a specific activity in the site. We'll get alerted from Nagios once it gets the threshold. It was an established routine that when we get the alerts, we'll hop on to all the app servers, and restart the apps 1 at a time. This solved the issue, but we were only treating the symptoms.

I investigated the issue using tools like: rbtrace, memory_profiler, and ObjectSpace, found the offending code and fixed it.

2) Adhoc script to charge a few million dollars from customers:
The script would be run once in a while when a discount has to be applied on a few customer's purchases. Since we are charging money, I was careful to capture all possible cases where the input might not be as expected, and collect and report relevant errors, and the customer data that caused it. In just 4 days, I had to write this script. When it was deployed in production, it ran successfully without any incident, while bringing in an extra few million dollars in revenue.

OTHERS

And several other projects from companies I worked prior to the above.

EDUCATION

Anna University - Chennai (*2004 - 2008)*
Bachelor of Engineering in Computer Science