**Google Cloud Platform Team 1**

Google offers three options for deploying and running your Ruby on Rails applications on the Google Cloud Platform. However, if you want to have full control over the underlying virtual machines you can opt for a Google App Engine, a Google cloud platform for the development and hosting of web applications. Google App Engine is accessible via free-tier and comes with pre-installed Ruby Gems, system libraries, custom Docker runtimes, and other useful tools for Rapid Application Development (RAD) with Ruby on Rails. It is easy to get started developing Ruby on Rails apps that run on Google Cloud.

**Google Cloud Platform and Databases**

**!SQL on Google Cloud Platform**

A !SQL database is a database that is stored in blocks and not specific in structure, schema easily changed. It is good for big data and real time web app (data too large to be processed traditionally). It has no predefined model of data structure or schema, rather is stored as blocks in JSON format. It is easy to manage since it requires less manipulation once the blocks are in place. However, all joins and changes are manual.

Google cloud’s Bigtable is ideal for such a data type with low latency, only pay for what you use. It allows you to create back up against catastrophic events. All data is encrypted. It is great for very large amount of data over long periods of time

Importantly, data is never stored in Cloud Bigtable nodes themselves. Each node has pointers to a set of tablets that are stored. For this reason, rebalancing tablets from one node to another is very fast, because the actual data is not copied. Cloud Bigtable simply updates the pointers for each node. Recovery from the failure of a Cloud Bigtable node is very fast, because only metadata needs to be migrated to the replacement node. If a Cloud Bigtable node fails, no data is lost.

**Relational databases:** [Rails Active Record](http://guides.rubyonrails.org/active_record_basics.html) works best with a traditional SQL database. With  [Cloud SQL](https://cloud.google.com/sql), in just a few clicks you can create a MySQL or PostgreSQL database that is fully managed and scaled by Google, with no management on your end.

**Data Warehouse on the Google Cloud platform**

A data Mart is a condensed and more focused version of a data warehouse that reflects the regulations and process specifications of each business unit within an organization. Each data mart is dedicated to a specific business function or region. This subset of data may span across many or all of an enterprise’s functional areas.

A data lake is a vast pool of raw data, the purpose for which is not yet defined.

On the other hand, if you need to create a data warehouse with data Marts and data Lakes, BigQuery is a serverless and completely managed SQL data warehouse. It is capable of rapid SQL queries and interactive analysis of massive datasets (order of terabytes/petabytes). The platform utilizes a columnar storage paradigm that allows for much faster data scanning plus a tree architecture model that makes querying and aggregating results significantly more manageable and efficient. The BigQuery Data Transfer Service automates data movement into BigQuery on a scheduled, managed basis. The analytics team can lay the foundation for a BigQuery data warehouse without writing a single line of code

**Application Programming Interface or API**

**Application Programming Interface or API** is a computing interface which defines interactions between multiple software intermediaries. It defines the kinds of calls or requests that can be made, how to make them, the data formats that should be used, the conventions to follow, etc. It allows different systems and apps to interact between each other to share data or services. Authentication usually done through API keys. The main types of APIs are REST and SOAP. Most if not all of Google Cloud products are integrated with each other. There is also extensive documentation on how to work with each product as well as plenty of code examples. Most of their products also have SDKs (I.E. Firebase) making it very easy to integrate their products with custom APIs.

**Mobile App**

Since this is an employee only app, it should not be available through the usual channels. Fortunately, both Apple and Google offer solutions for businesses that wish to publish apps for use within their organizations.

It would also be ideal to have a straightforward system in place to distribute the app to the employees so that IT doesn’t end up having to call every single employee to have them install the app. [Google Workspace](https://workspace.google.ca/intl/en_ca/products/admin/endpoint/) (unrelated to Google Cloud) can provide such services. There are plenty of other companies that provide similar products (MDM) but it might be easier to get a complete solution from the same company. Google Workspace also comes with the suite of Google products which is a nice plus.

Employees may sometimes require the ability to access both the website’s back office and the app. Having them create two separate accounts for that purpose wouldn’t be ideal so using Google’s Firebase service seems to be a good solution as it provides secure authentication for both scenarios and only requires a single account. Firebase is a fully managed platform for building iOS, Android, and web apps that provides automatic data synchronization, authentication services, messaging, file storage, analytics, and more. It works along with GCP to provide a smooth integration for mobile and web apps to manage user generated content.

Since the app will mostly just be making calls to the Google Cloud APIs we’ll be creating, performance within the app will not be the top priority. The app will simply be a convenient way to get the relevant information to the employees quickly. For that reason, a technology like React Native would be a good fit as it allows developers to create cross platform apps with relative ease. Better yet, React Native comes with libraries that make integrating Firebase with React Native easier such as [React Native Firebase](https://rnfirebase.io/).

Deploying new releases of the app should look a lot like the process we will be using for the website. The app code will live in a Git repository and Cloud Build will be used to build and test the app.

**IoT**

Simple definition: A system of interrelated, internet-connected objects that are able to collect and transfer data over a wireless network without human intervention.

**Key benefits:**

New data that can be analyzed to streamline business processes, increasing productivity and reducing costs.

The new IoT data can also be used to build historical trends that help predict issues before they occur. For example, manufacturer warranty and maintenance information can be paired with IoT-collected data to predict maintenance incidents.

* 1. Increased customer satisfaction as a result of the added efficiencies.
* There are several ways to connect “things” to a network but it is usually done through a cellular network or through Wi-Fi.
* Every device needs a unique IPv6 address.
* Typically, the “things” are low power, low cost sensors.

# **Why use Google for IoT?**

* They offer a full managed service through Google IoT Core that requires a minimal setup.
* Protocol bridge to connect devices using industry standard protocols. Secured with TLS 1.2.
* The data is sent through Cloud Pub/Sub, allowing it to be used by analytic systems.
* Integrates with most of Google’s data analytics options.
* Devices can be managed, registered or deployed through REST APIs.
* Real-time metrics are included in the product through Google’s Stackdriver Monitoring.
* Centralized logging.
* Offline support.

**AI and Machine Learning / Google Cloud service**

The different type of learning

1. Supervised Learning – The AI is presented with a set of inputs and there corresponding output, its goal would be to find an algorithm that maps correctly the inputs to some outputs.

2. Unsupervised Learning – The AI will be on its own to try and find some patterns or structure in the inputs, it is commonly used to find patterns in a data set.

3. Reinforcement Learning – The AI will be interacting with a dynamic environment and is giving a goal. Ex: drive a car, play against an opponent, etc. The AI is provided with some feedback and based on the feedback that it received (good or bad), it will try to improve from it.

**Google Cloud AI**

The Cloud of AI computing (the service mentioned are used on the cloud by the companies so that they don’t have to have the infrastructure to process the model(s) they want to train).

• Cloud TPUs – Is a Tensor Processing Unit (custom build chips made for computing deep neural network task), more powerful training and more accurate models, it is also cost-effective and with faster speed and scale

• Cloud GPUs – Is a range of NVIDIA GPUs (Graphics processing unit, has in the term graphics cards), to be used by the company for training their AI. It is also a cost-effective process in which you can scale-up (it is refer as to a migration of the computation of the learning model to a bigger and faster processing server, and this allow the company to keep running the workload the way they always have) or scale-out (which refer to expanding the computation to other/multiple servers rather than a big server. It allow the possibility to move one workload to an other server or even to combine them into a single computing resource, if needed) the workload process.

**Tensor-Flow**

• The AI Cloud computing of Google has support for Tenser-Flow model which to choose from a variety of models. The support for more custom model is also supported by the cloud service.

**Google Cloud TPUs**

• We suggest the use of cloud TPUs for the neural network for the fast and reliable result

for training a custom model that fits your need.

**DevOps in Plain English**

The DevOps is a term and not a tool. DevOps is generally viewed as a state of mind. On one side, we have the developers team and on the other we have the IT team. Both play an important role for a successful information system. The development team assuring the implementation of new feature and working on testing those feature and the operation team assuring the success of the deployment and the maintenance of the servers. In a big company like Rocket-Elevators the deployment are done sequentially, once per month, and if the developers are not keeping a good communication with the operation team they might not be able to release their new feature at time. We can think of DevOps as a good communication between both party but also having a base understanding of the work that the other team do, and even switch side.

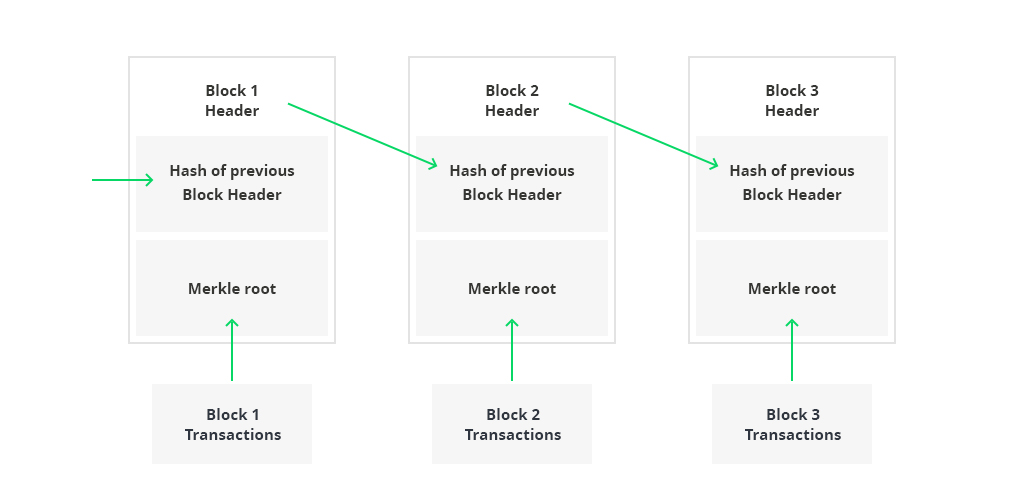
So globally DevOps regroup two team, one related to the development and the other to the operation. The team should work together as a collective. The agglomeration of their work result in the project as a whole. So as the project grow in size there should not be late release, bugs, etc.

# **DevOps Tools Offered By Google**

DevOps cannot be implemented only by using the right tools. A cultural shift must also occur within the organization. With that said, here are some tools that can be used to help with the transition:

There are many more tools for more specialized workflows but for our purposes, those tools should do the trick.

**The Blockchain**

The Blockchain is mostly used as a digital ledger (it is a digital record of transactions made from one party to an other). It consists of **Blocks** that record transactions across many computers, so that any block can not be altered retroactively (event that already happened) without altering the other subsequent blocks. The Blockchain is managed autonomously by a process called peer-to-peer (which is the process of computing a certain workload on multiple computer which allow faster results). The use of a Blockchain as for the transferring of digital goods, remove the possibility of having the problem of double-spending which can be seen as a falsified or duplicated digital goods (counterfeit money), and in which case can lead to an inflation. The blocks are linked together using cryptography (it is a process used to secure information). The blocks all contain a *cryptographic hash* (it is the outcome of a certain message and cannot be the same hash for another message, if provided the same message twice you will be getting the same hash) of the previous block, a timestamp, and some transaction data (mostly known as Merkle root).

**The usage of the Blockchain by Rocket-Elevators**

The Blockchain that Rocket Elevators want to use is called a **Smart Contract Blockchain** which is intended to automatically execute, control or document relevant information about a given contract. The objective here is to reduce the need for inter-mediators, arbitration and avoid fraud lost, as well as reduce malicious and accidental exception. That way we avoid the need for a third party to come in place to act as an intermediary between two contracting entities.

**The Blockchain and Google Cloud Service**

Google now allows the hosting of STRATO node (is a rapid deployment Blockchain for businesses called Blockapps), in their clouds. As far has I know google doesn’t directly provide a service for Blockchain but is in cooperation with Blockapps. Google provide the node (a node is one of the computer communicating to the network called peer-to-peer) and Blockapps provide the software and engineering needed so that the client doesn’t need to hired Blockchain Developers and invest in a lot of hardware. The service offer in their marketplace called “STRATO for business Network” is a good foundation for the client and is flexible enough to accommodate their business need

**Price List**

|  |  |  |
| --- | --- | --- |
| **Services** | **Price** | **Amounts** |
| App engine | $109.65 | F2, 2 instance standard |
| BigQuery | $2,100 | 10GB/month free |
| Cloud CDN | $3.00 | Traffic upload/download |
| Dataflow | $1.59 | 1 standard batch |
| Bigtable | $1,576.80 | 2 nodes |
| cloud endpoint API | $3.00 | 2million request /month |
| Cloud TPU | $986.00 | v2 8core 64G flash memory |
| Compute engine | $107.71 | 2 instance |
| Cloud storage | $70.66 | 3TB |
| Cloud SQL | $369.48 | 730 hrs/month |
| Firebase | Free for up to 1000 calls/month | then 0.01c per verification in the US |
| Cloud IAM | No Additional Cost |  |
| IOT Core | $12.00 | 3GB |
| Cloud Pub/Sub | free 10GB/month | 40 after |
|  |  |  |
| Monthly Total | $5,339.89 |  |