Server Technology Evaluation

Tertúlias

# Evaluation Strategy

Given the goal of this phase is to select a specific server technology that we will propose to use as a project backend, we opted to take an RFP-like approach, which we believe is a well‑known paradigm that anyone can easily relate to, thus providing clear grounds for the understanding of our decision making process.

First, we will try to list the most critical internal and external factors to evaluate, regarding each selected technology provider. Internal factors being essentially those related to the provider’s technology and external factors being those related to the business and financial environment where the provider moves.

Along with the above, we shall narrow the choices by defining some architecture aspects that we believe make sense, given both our understanding of the technology trends and other aspects we find relevant for the project development, namely, aspects that will save us time in the development, that contributes to a robust, performant and scalable application architecture and that have a costs curve in line with future business growth.

With this approach in mind, we will build a weighted criteria list, against which we will score all short-listed candidates and from there we shall pick our winner.

# Evaluation Criteria

“Mobile back end as a service has become a critical component of enterprise mobility. Mobile strategists and architects can choose from products that offer a variety of required mobile app services, including data hosting, integration and orchestration, location, and identity and access management”

(Richard Marshall, 2015, p. 1)

Clearly, what we need to have as a backend is the functionality that it is normally provided by the mBaaS. We can achieve this either by selecting directly an mBaaS to provide the service, or by selecting discrete parts of infrastructures and services and assemble our own infrastructure and make the required developments to provide the mBaaS functionality we need.

* In respect to suppliers:
  + We shall only short-list specific technologies linked to the core business of the respective supplier.
  + We shall short-list only technologies from suppliers with a solid track record and a large installed base of Clients of relevant Clients whose operation is largely dependent on those technologies.
  + We would prefer Cloud based solutions to minimize the setup, management and specific skills (ex. security) required to have a production solution deployed in the Internet.
* In respect to the technology:
  + Mandatory:
    - Data hosting, SQL or NoSQL (JSON);
    - Identity and access management, namely “OAuth 2.0” - integration with Facebook, Google and Twitter login;
    - REST API;
    - Low entry cost footprint;
  + Valuable:
    - Cloud based;
    - Location based services;
    - A comprehensive management console;
    - SDK for Android, IOs, and JavaScript;
    - Push notifications;
    - Free-tier services;

# Selection Short-List

As a result of our research[[1]](#footnote-1) we’ve short-listed the following vendor’s offers:

1. Amazon – AWS Mobile Services; <<https://aws.amazon.com/mobile/>>
2. Appcelerator – Appcelerator; <<http://www.appcelerator.com/>>
3. Apache – BaasBox; <<http://www.baasbox.com/>>
4. built.io – built.io; <<https://www.built.io/>>
5. Google – Firebase; <<https://www.firebase.com/>>
6. Microsoft – Microsoft Azure Mobile Services; <<https://azure.microsoft.com/en-us/services/app-service/>>
7. MongoDb – MongoLab; <<https://mlab.com/>>
8. Oracle – Oracle Mobile Cloud Service; <<https://cloud.oracle.com/mobile>>
9. Red Hat – FeedHenry; <<http://www.feedhenry.com/>>
10. IBM – StrongLoop; <<https://strongloop.com/>>

# Comparison Matrix

## The Matrix

| Requirement | Weigth | AWS | Appcelerator | BaasBox | built.io | Firebase | Azure | Oracle | FeedHenry | StrongLoop | | MongoLab |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Offer is part of core business | M | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y | |
| Service maturity (> 5 yrs) | M | Y | Y | N | Y | Y | Y | N | Y | Y | N | |
| Customer base size | M | Y | Y | N | N | Y | Y | N | Y | Y | N | |
| Relevant mobile Apps using it | M | Y | Y | N | N | Y | Y | N | Y | Y | N | |
| DB Backend (SQL/NoSQL) | M | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y | |
| IAM | M | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y | |
| REST API | M | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y | |
| Low entry level costs | 8 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | |
| Cloud Free-tier | 5 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | |
| Server Side Code | 8 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | |
| GeoSpacial services | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | |
| Management console | 5 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | |
| SDKs | 4 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | |
| Push notifications/sync services | 8 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | |
| Cloud & Hosted & OnPermises | 3 | 2 | 1 | 3 | 1 | 1 | 2 | 1 | 3 | 2 | 3 | |
| Setup easiness | 3 | 1 | 5 | 5 | 0 | 5 | 5 | 1 | 2 | 3 | 5 | |
| Overall personal impressssion | 5 | 4 | 3 | 2 | 1 | 3 | 5 | 3 | 3 | 5 | 3 | |
| Score |  | 68 | 72 | - | - | 56 | 85 | - | 68 | 79 | - | |

## Short-list notes

### AWS

IAM: AWS Cognito

DB Backend (NoSQL): AWS DynamoDb

Data storage: AWS Cognito + AWS S3

Code in the Cloud: AWS Lambda

Push notifications: AWS SNS

GeoSpacial services: Geo library

### Appcelerator

Product: Formerly “Cocoafish”

REST API: Appcelerator Arrow

DB Backend (NoSQL): Appcelerator ArrowDb

Push notifications: Appcelerator Arrow

Refs: <<https://flexr.wordpress.com/2013/01/02/backends-as-a-service-appcelerator-acs-vs-parse-vs-applicasa-vs-stackmob/>>

### BaasBox

DB Backend (NoSQL): OrientDB

### Firebase

GeoSpacial services: GeoFire library

### Azure

Product Suite: Azure Mobile Services, App Service

Pricing: <<https://azure.microsoft.com/en-us/pricing/details/app-service/>>

DB Backend (NoSQL): SQL server, MongoDb, etc.

### Oracle

Product Suite: Oracle Mobile Cloud Service.

Mobile Cloud Pricing: <<https://cloud.oracle.com/en_US/mobile?resolvetemplatefordevice=true&tabID=1431452915472>>

Major concern: Platform offer integration and stability

### FeedHenry

Product: Formerly, FeedHenry Mobile Application Platform; Currently, RedHat Mobile Application Platform.

### StrongLoop

Platform: IBM MobileFirst Platform

Product: LoopBack

Major concern: Product consolidation

Reference: <<https://mobilefirstplatform.ibmcloud.com/blog/2015/09/10/getting-started-with-ibm-mobilefirst-and-node-js-apis-built-with-loopback/>>

# Technology Selection

Given the results of the analysis, our proposal is to select **Microsoft’s Azure Mobile Services** as a backend server.

The overall key points to support our proposal are:

* From a strategic standpoint:
  + The vendor is strongly committed to mobility and cloud, with a clear strategy to cover all aspects required for a B2C mobile application for startup companies – like the one that might result from this project, including standards adoption and support for multivendor components, which is crucial to prevent vendor lock-up.
  + The vendor has consistently been one of the top worldwide software editors for the last three decades, with a solid track-record in technology innovation, and has shown repeatedly its ability to recognize important technology shifts and winning market trends and the capacity to realign its offer accordingly.
  + Microsoft also has a solid track record in complete support to its own technology, including company channels to train, support, work with and promote local partners at any scale.
* From a technology standpoint:
  + For what matters to our target project, in the short-term, the vendor supports all the needs we were able to identify, both with their own technology and via integration of third party architecture blocks, as a result of its model of infrastructure services composed by a loosely coupled collection of services built on Azure that can be choreographed into a tailored solution (Hammond, 2015).
  + The vendor is long recognized for providing state of the art technology backed by clear setup and integration methods, extensive set of software libraries and profuse documentation both in online sources and books from a huge number of authors and editors.

The main risks identified for the short term (June 2016!) and respective mitigations are:

* Setup and use:
  + Risk: Lack of experience and knowledge in setup and use of the server technology.
  + Mitigation: Select a cook-book recipe that suites the project requirements, preferably involving an SQL database technology.
* Mobile devices interface:
  + Risk: Secure connection establishment of Android – Azure.
  + Mitigation: Evolve from documentation examples.
* Server side services integration:
  + Risk: Coupling services such as push notifications with database events.
  + Mitigation: Identify specific needs early and prepare Plan B’s on client side.
* Security threats:
  + Risk: Server attacks.
  + Mitigation: Early identification of best practices both on server side services and client side access.

This document includes the following annexes:

* Annex 1:

End of Document

# Annexes

## Annex 1: References

### References

Hammond, M. F. (2015, September 10). The Forrester Wave™: Mobile Infrastructure - Ten Providers That Matter And How They Stack Up. Cambridge,, MA, USA.

Richard Marshall, V. L. (2015). Market Guide for Cloud Mobile Back-End. USA: Gartner, Inc.

### Recommended Reading

Developer Economics:

* Finding the Right BaaS <<http://www.developereconomics.com/finding-right-baas/>>
* tool finder (beta) <[http://www.developereconomics.com/search/tools/#q//sector/35/license//technology//platform//app\_category/](http://www.developereconomics.com/search/tools/%23q/sector/35/license/technology/platform/app_category/)>

Flexr:

* Backends as a Service: Appcelerator ACS vs Parse vs Applicasa vs StackMob <<https://flexr.wordpress.com/2013/01/02/backends-as-a-service-appcelerator-acs-vs-parse-vs-applicasa-vs-stackmob/>>

InfoWorld:

* Review: Microsoft Azure beats Amazon and Google for mobile development <<http://www.infoworld.com/article/2890167/application-development/review-microsoft-azure-beats-amazon-and-google-for-mobile-development.html>>

1. See Annex 1: References [↑](#footnote-ref-1)