Prepared by: kennedia IT Solution

14th of April'2019

PROPOSAL

INFRASTRUCTURE ARCHITECTURE & DESIGN

VERSION 1.1

Disease Awareness & Access to Healthcare

1. INTRODUCTION

This document provides a high level overview and explains the architecture of the Disease awareness and access to healthcare system.

The document defines goals of the architecture, the use cases supported by the system, architectural styles and components that have been selected. The document provides a rationale for the architecture and design decisions made from the conceptual idea to its implementation.

1.1 PURPOSE

The Web and Mobile app Architecture Document(WMAD) provides a comprehensive architectural overview of the Disease awareness and access to healthcare . It presents a number of different architectural views to depict the different aspects of the system.

1.2 SCOPE

The scope of this WMAD is to explain the architecture of the Disease awareness and access to healthcare system.

This document describes the various aspects of the DAAH system design that are considered to be architecturally significant. These elements and behaviours are fundamental for guiding the construction of the DAAH system and for understanding this project as a whole. Stakeholders who require a technical understanding of the DAAH system are encouraged to start by reading the Project Proposal, Concept of Operations and Software Requirements Specification documents developed for this system.

1.3 DEFINITIONS, AND ABBREVIATIONS

- Apache – Web Server
- CodeIgnitor - Php framework
- Cloud - AWS (Amazon cloud)

- HTTP - Hypertext Transfer Protocol
- WWW - World Wide Web

2. ARCHITECTURE REPRESENTATION

This document details the architecture using the user role defined in the App's. The views used to document the DAAH system are:

Use Role

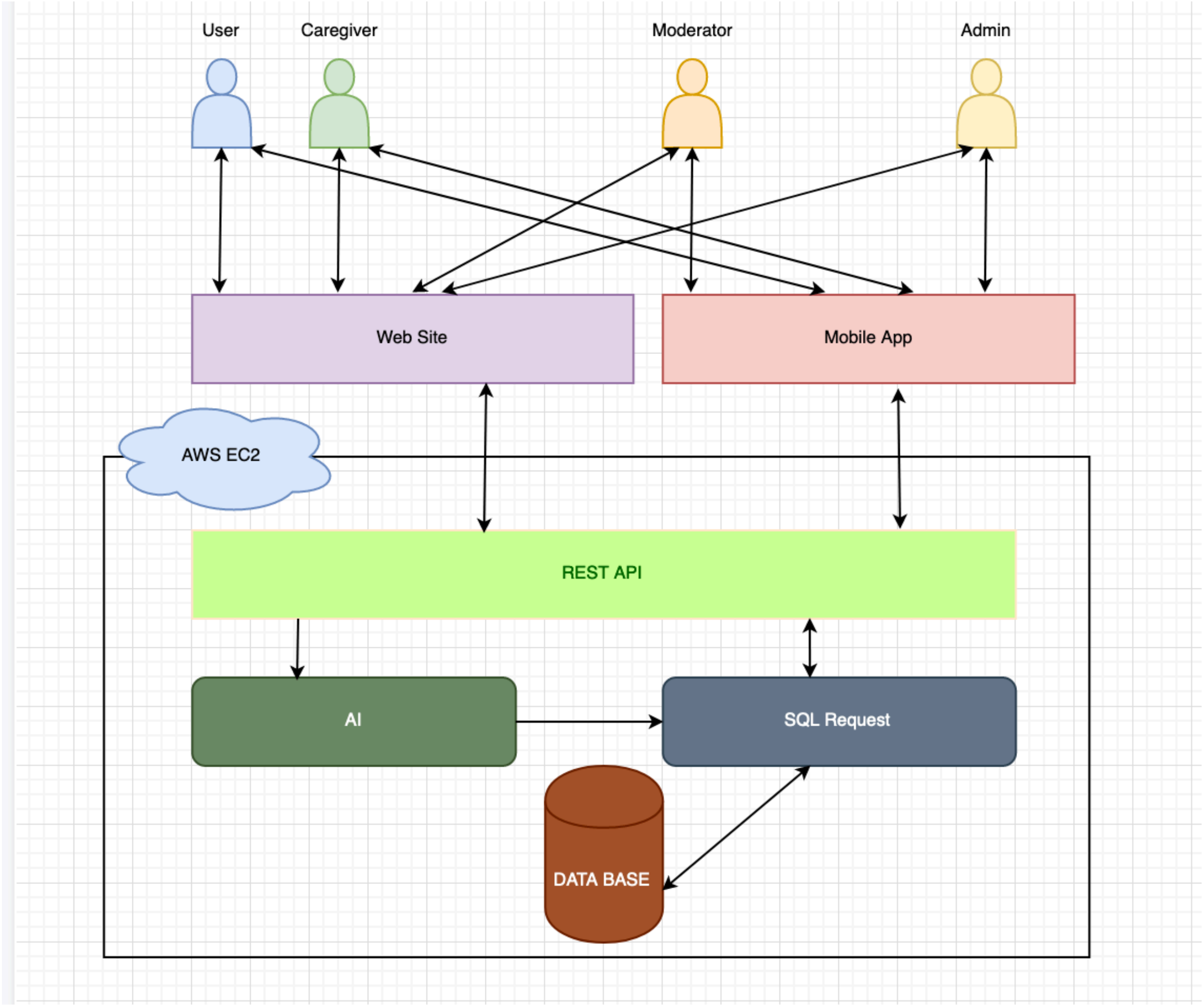
- User: user enters his health issue.
- Moderator: Moderator are monitor to user and provides connectivity User to Caregiver .
- Caregiver : it's provides a user solution
- Admin : Administration of site, like approving users, advertisements, moderators etc.

2.1 ARCHITECTURAL GOALS AND CONSTRAINTS

There are some key requirements and system constraints that have a significant bearing on the architecture. They are:

1. The system is provides awareness of disease and access to healthcare and health product offer. This system is user monitor for health update.
2. The system will be written using Codeigniter technologies but will use an AWS cloud (RDBMS system MySQL) for data persistence and will be deployed to a Linux webserver running AWS cloud. These special deployment requirements require additional consideration in the development of the architecture.
3. The system user AI for data intelligence for get app as per location and search data to assist moderator .
4. Advertisement handling for local and global users.
5. API handling

2.2 ARCHITECTURAL-MOBILE AND WEB



User Types:

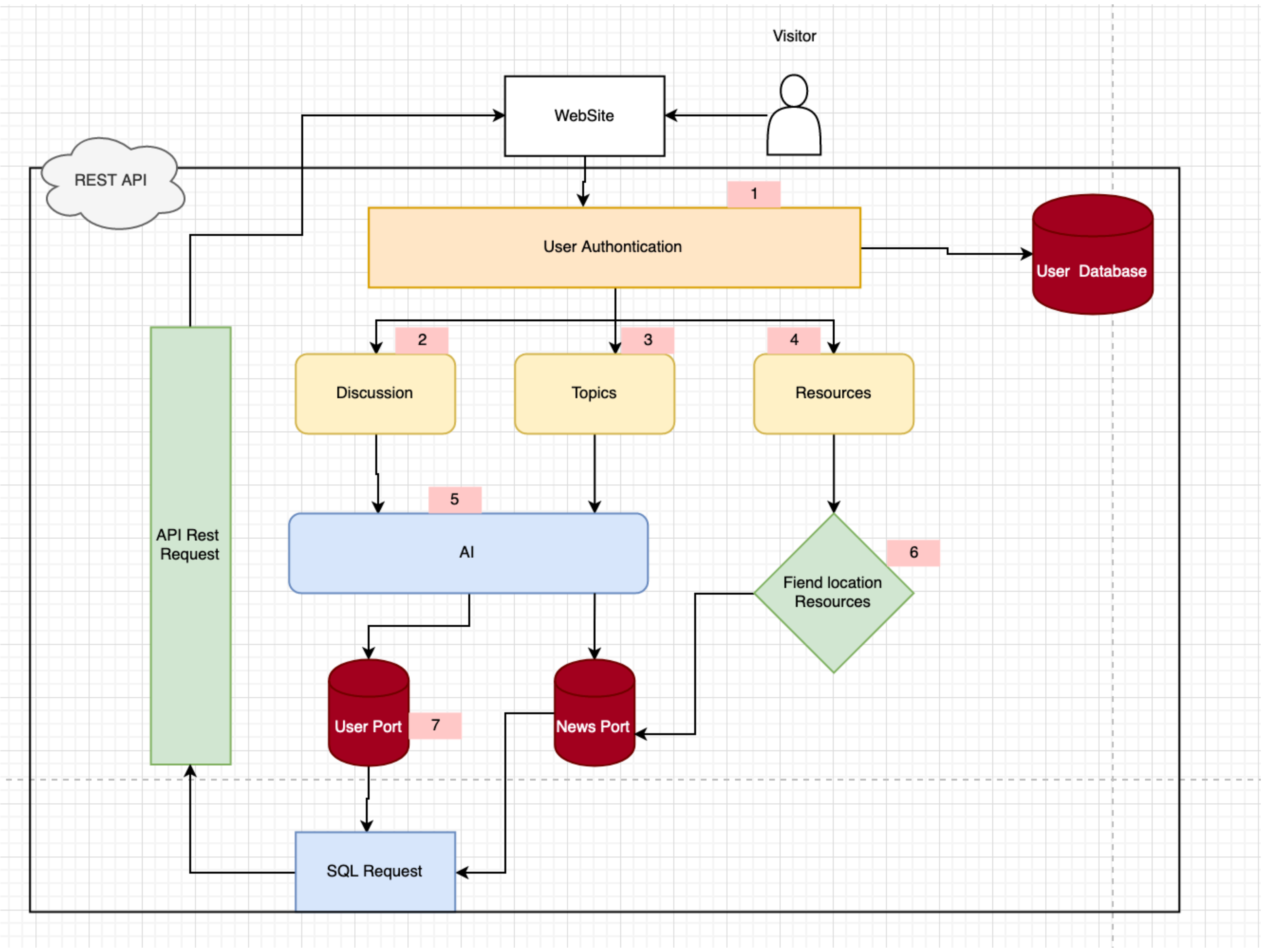
- Four for web. (patient , caregiver , Moderator , admin)
- Three for mobile (patient , caregiver , Moderator)

Communication:

User will send requests to web or mobile which enables REST API and get data from Cloud . After fetching data web and mobile shall display and process requests for platforms (web & mobile). kennedia will code the REST API in such a way that it will communicate with cloud and platforms.

- REST API store will be stored on AWS server/EC2.
- REST API will execute request in SQL server or will call AI component.
- SQL server connects to database and return data to API and API send data in JSON format to platforms.
- AI will decide the kind of data to furnish as per user request and send information to sql server . Sql server will return the data to reset api.

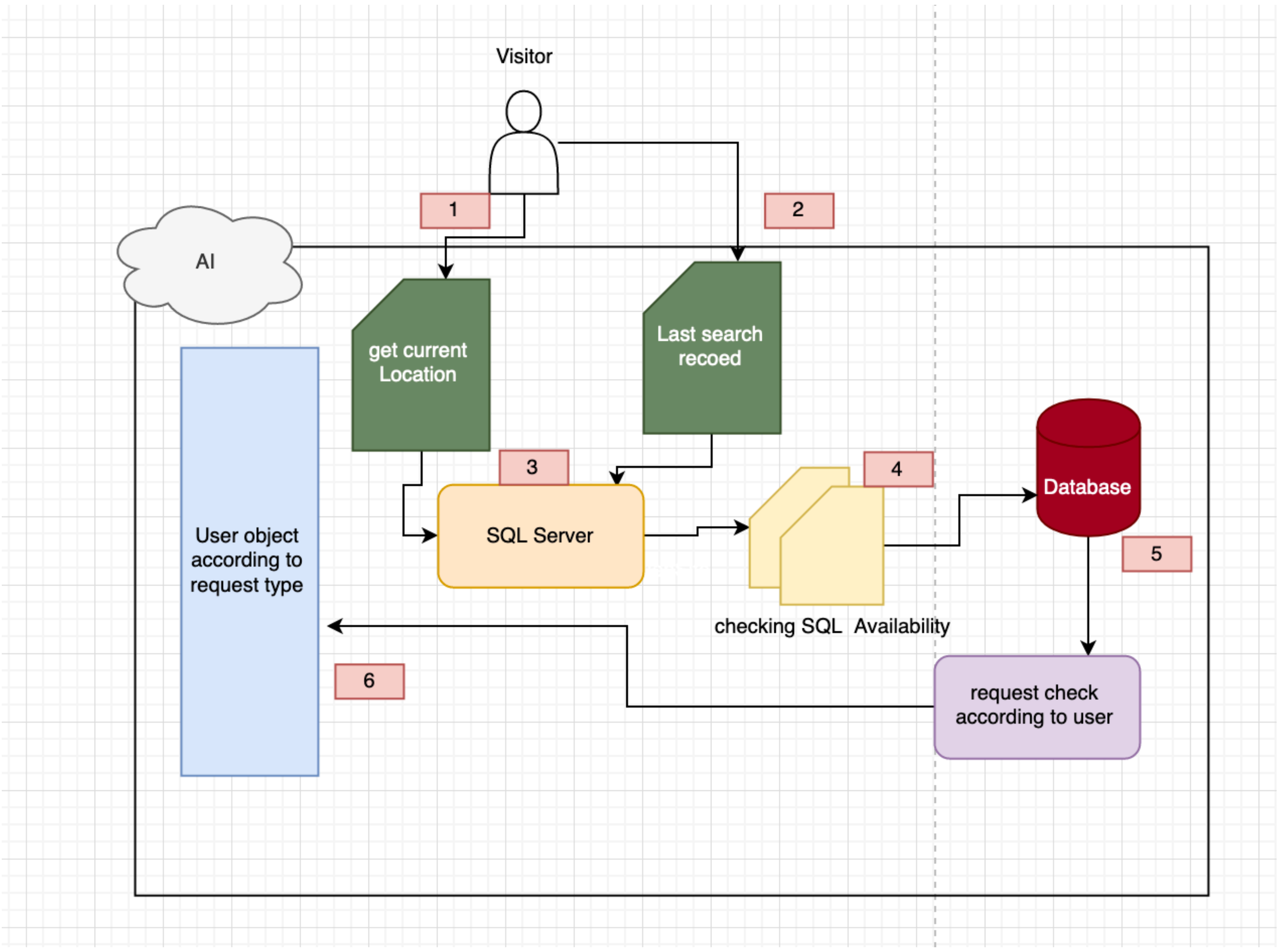
2.3 ARCHITECTURAL- REST API



User will request (Any type of request from web or Mobile) that will activate/call API to do the following

- REST API first checks user authentication by user Database then call component (Discussion, topics and resources) as per request
- Upon activating discussion component, it will check last posts as per the discussion and will fetch all those posts via AI.
- In case of Topics it will fetch it's location and last search result and send request to AI
- In case of resources the system will check whether it's local or global and display data accordingly.
- AI get location and last activity and will suggest data accordingly .
- The data will be returned to REST API to server the request of user.

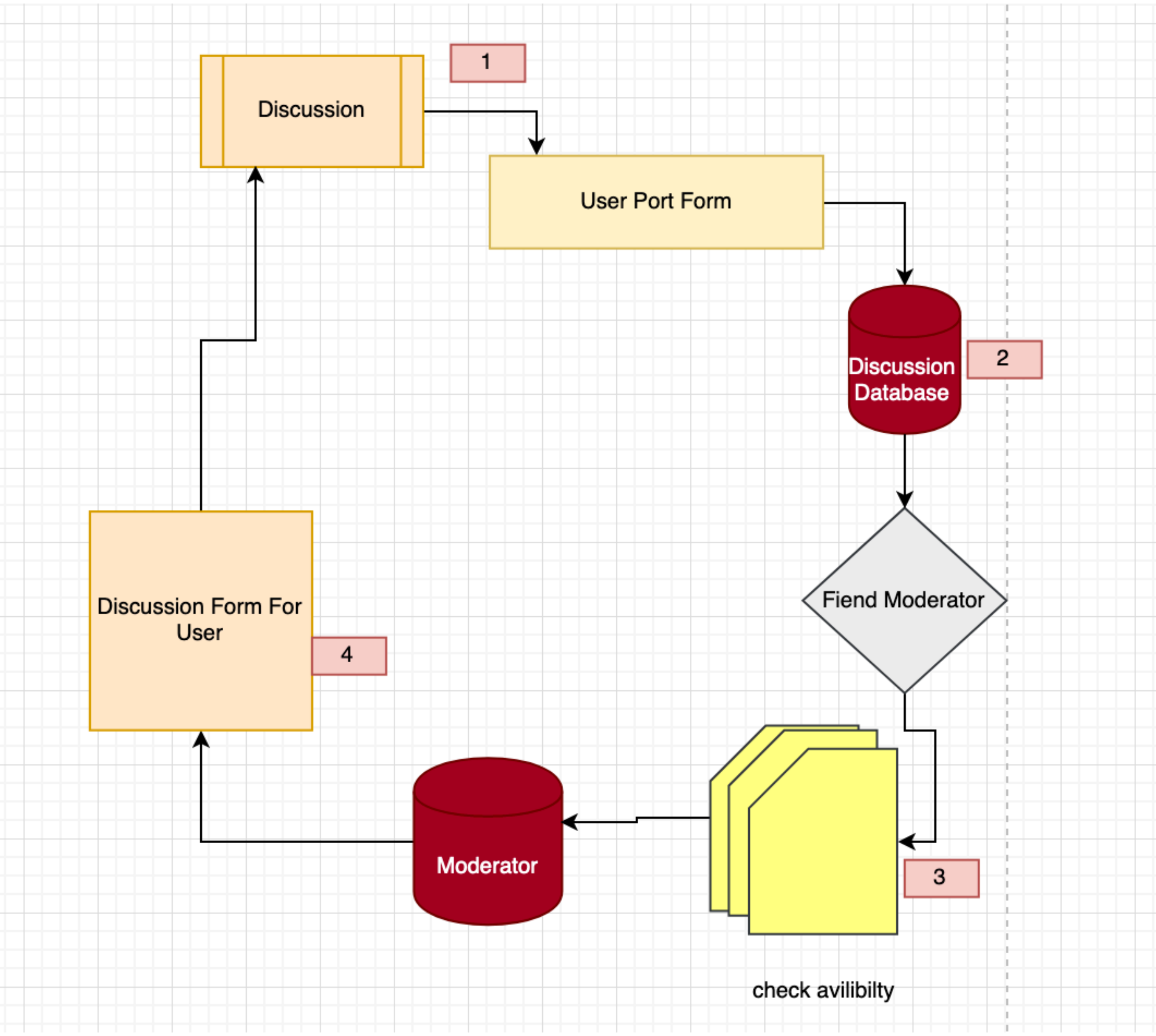
2.4 ARCHITECTURAL- ARTIFICIAL INTELLIGENCE


Visitor
1
2
AI
get current Location
Last search recoed
3
4
SQL Server
Database
5
User object according to request type
checking SQL Availability
6
request check according to user

User request will be sent in two way’s to trigger AI/ML module.

- Get current location for discussion of health issue
- User need information for any issue.
- After getting request type SQL server will find record (above Four points). The data is verified for record information
- Database will verify the data based on user type.
- The data fetched will be sent to User.

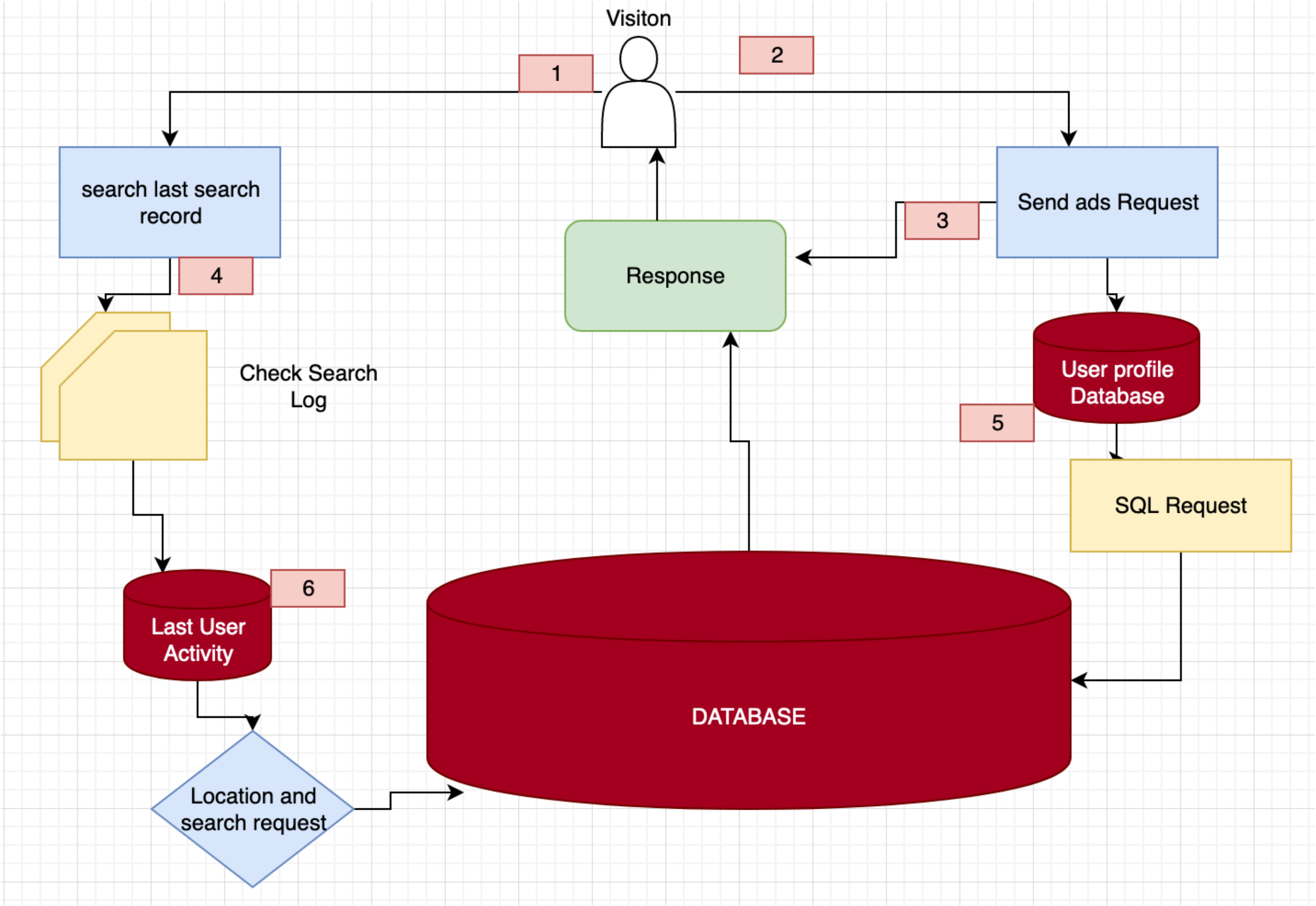
2.5 ARCHITECTURAL- DISCUSSION



In Discussion :-

1. When user will send any post (health issue) AI will find similar type issue in database.
2. Suggestions will be shared with Moderator enabling them to propose suggestions to users.
3. Moderator will be assigned as per their availability.

2.6 ARCHITECTURAL- ADVERTISEMENT



The advertisements will get displayed in two following ways

A) according to location (Local/Global)

B) according to search adv. by user

- If user open resource page he is get Advertisements by location.
- Only active advertisements will be fetched from database.
- The advertisements will be displayed as per profile and location of user.

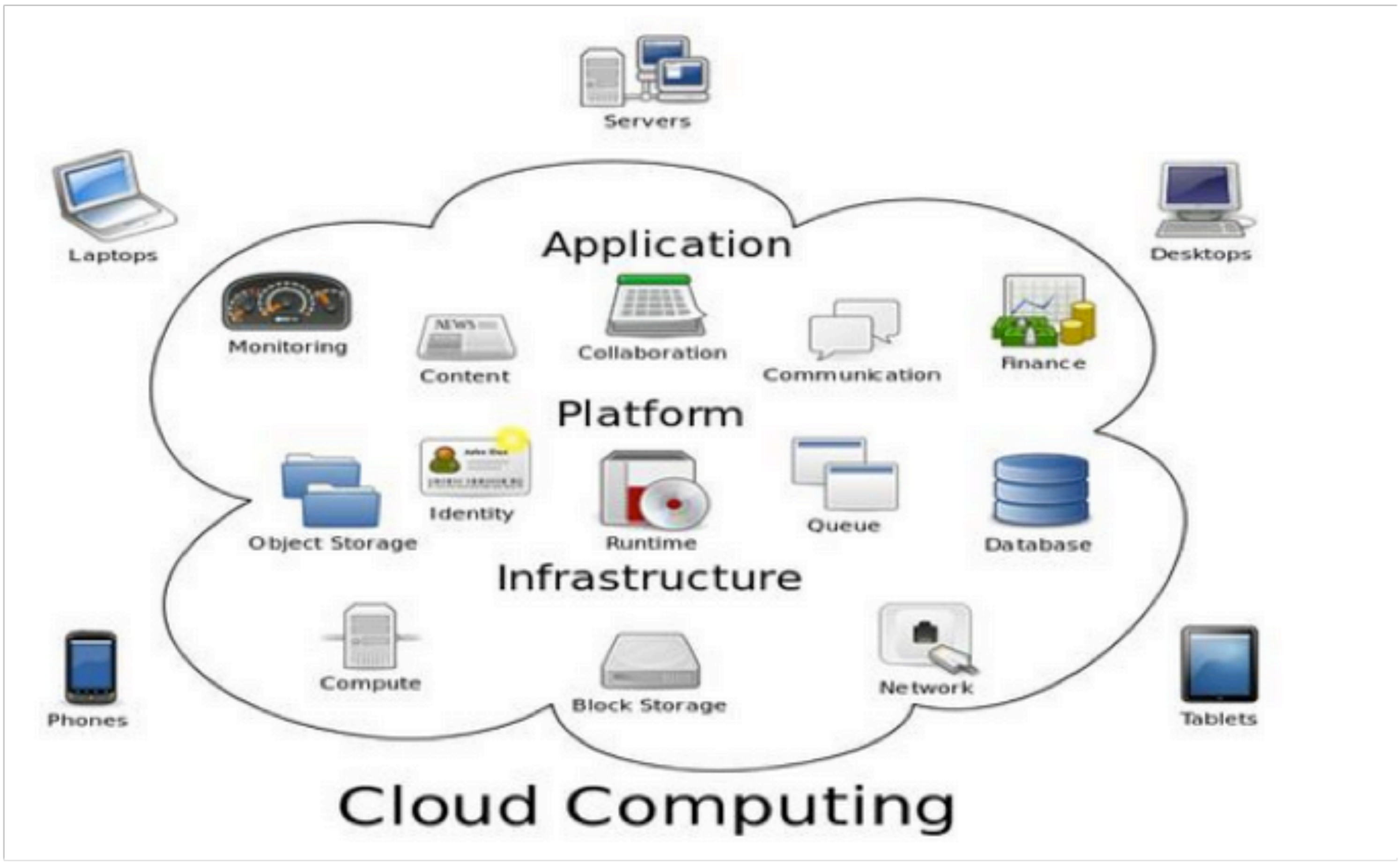
3. CLOUD ANALYSIS

This section details the assessment of best possible option for DAAH application from current cloud offerings in the market.

3.1 PURPOSE

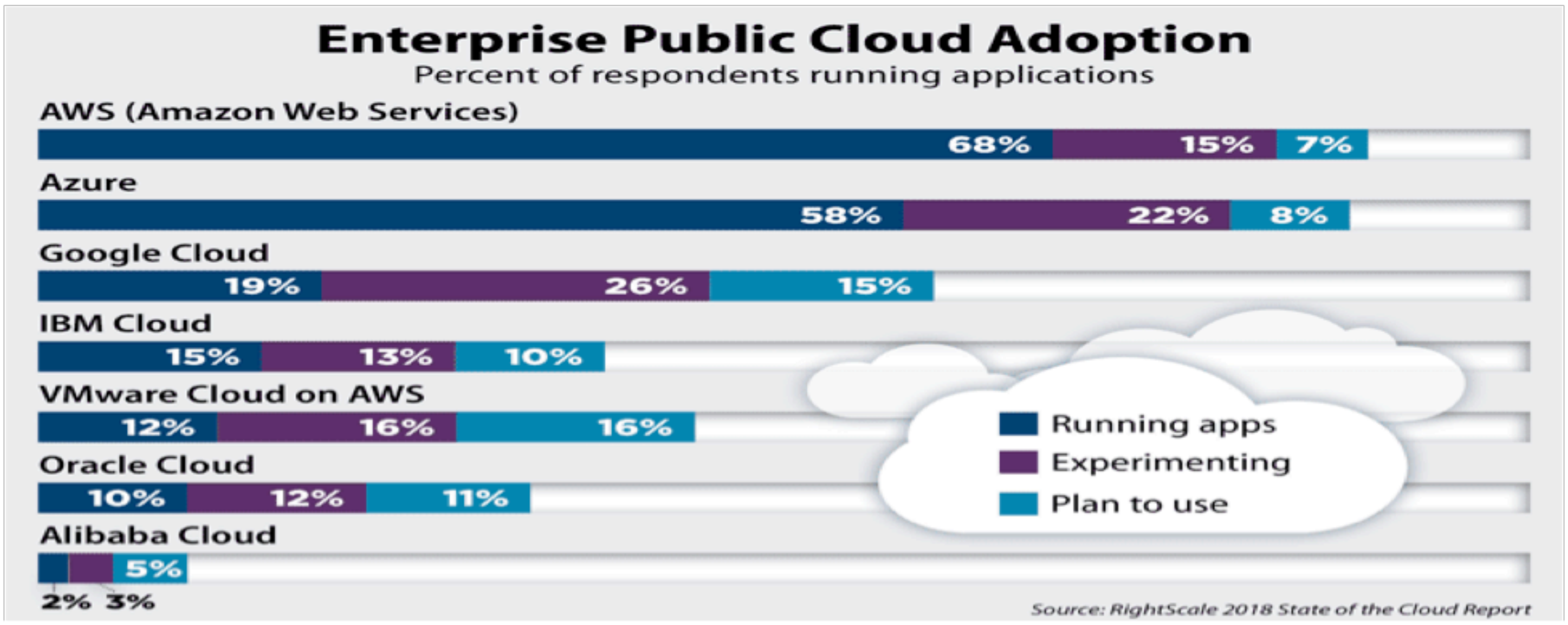
Cloud computing is *not* about is your hard drive. When you store data on–or run programs from the hard drive, that's called local storage and computing and is also *not* about having a dedicated hardware server in residence.

you need to access your data or your programs over the Internet, or at the very least, have that data synchronised with other information over the Net. In a big business, you may know all there is to know about what's on the other side of the connection; as an individual user, you may never have any idea what kind of massive data-processing is happening on the other end. The end result is the same: with an online connection, cloud computing can be done anywhere, anytime.



3.2 POPULAR CLOUD OPTIONS AND COMPARISON

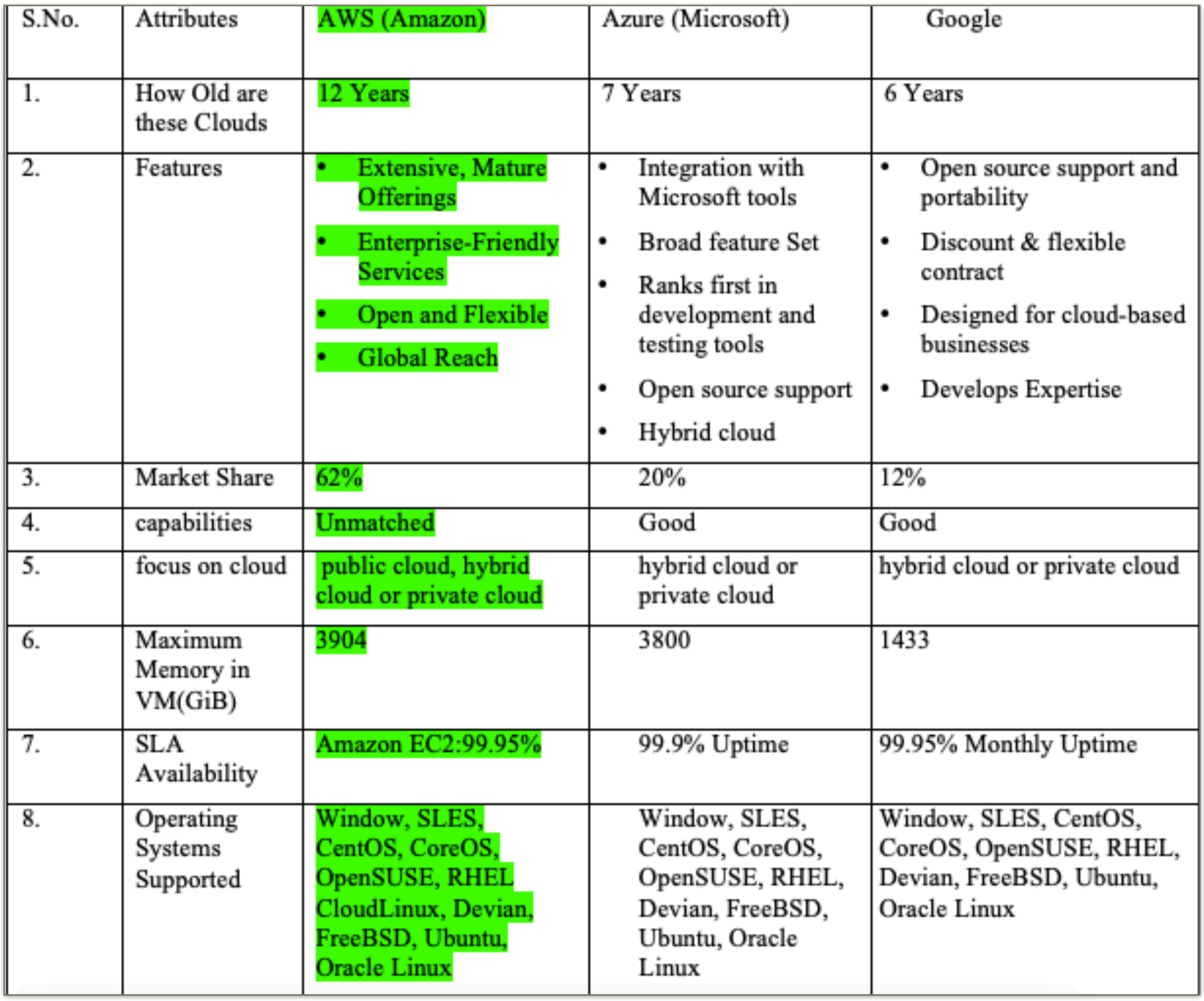
kennedia evaluated AWS cloud | Microsoft Azure | Google Cloud in detail , however below image gives adoption statistics for different providers (ref: Rightscale@2018).


Enterprise Public Cloud Adoption

Percent of respondents running applications

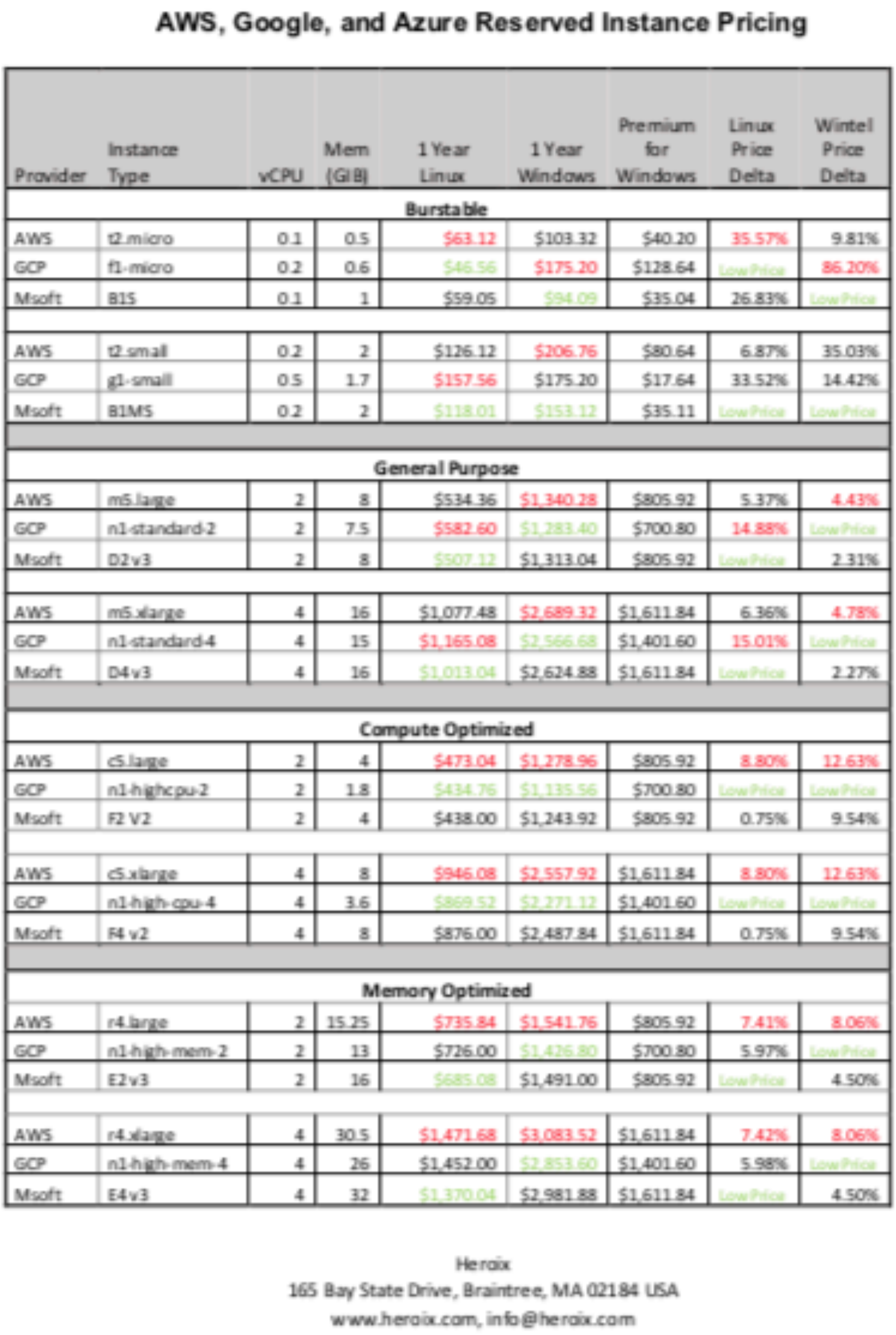
Source: RightScale 2018 State of the Cloud Report

Each offerings evaluated has its strengths and weaknesses.



S.No.	Attributes	AWS (Amazon)	Azure (Microsoft)	Google
1.	How Old are these Clouds	12 Years	7 Years	6 Years
2.	Features	• Extensive, Mature Offerings • Enterprise-Friendly Services • Open and Flexible • Global Reach	• Integration with Microsoft tools • Broad feature Set • Ranks first in development and testing tools • Open source support • Hybrid cloud	• Open source support and portability • Discount & flexible contract • Designed for cloud-based businesses • Develops Expertise
3.	Market Share	62%	20%	12%
4.	capabilities	Unmatched	Good	Good
5.	focus on cloud	public cloud, hybrid cloud or private cloud	hybrid cloud or private cloud	hybrid cloud or private cloud
6.	Maximum Memory in VM(GiB)	3904	3800	1433
7.	SLA Availability	Amazon EC2:99.95%	99.9% Uptime	99.95% Monthly Uptime
8.	Operating Systems Supported	Window, SLES, CentOS, CoreOS, OpenSUSE, RHEL CloudLinux, Devian, FreeBSD, Ubuntu, Oracle Linux	Window, SLES, CentOS, CoreOS, OpenSUSE, RHEL, Devian, FreeBSD, Ubuntu, Oracle Linux	Window, SLES, CentOS, CoreOS, OpenSUSE, RHEL, Devian, FreeBSD, Ubuntu, Oracle Linux

3.3 COST COMPARISON



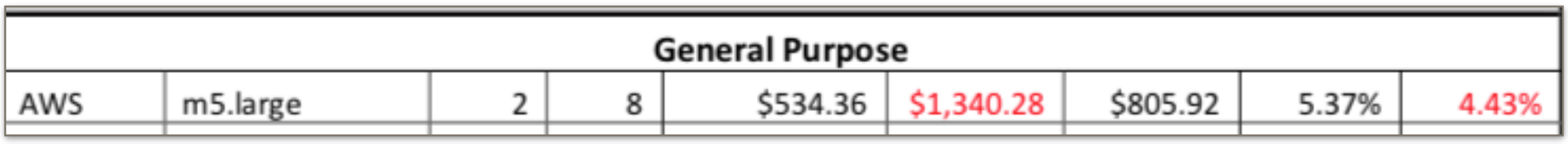
AWS, Google, and Azure Reserved Instance Pricing

Provider	Instance Type	vCPU	Mem (GIB)	1 Year Linux	1 Year Windows	Premium for Windows	Linux Price Delta	Wintel Price Delta
				Burstable				
AWS	t2.micro	0.1	0.5	$63.12	$103.32	$40.20	35.57%	9.81%
GCP	f1-micro	0.2	0.6	$46.56	$175.20	$128.64	Low Price	86.20%
Msoft	B1S	0.1	1	$59.05	$94.09	$35.04	26.83%	Low Price
AWS	t2.small	0.2	2	$126.12	$206.76	$80.64	6.87%	35.03%
GCP	g1-small	0.5	1.7	$157.56	$175.20	$17.64	33.52%	14.42%
Msoft	B1MS	0.2	2	$118.01	$153.12	$35.11	Low Price	Low Price
				General Purpose				
AWS	m5.large	2	8	$534.36	$1,340.28	$805.92	5.37%	4.43%
GCP	n1-standard-2	2	7.5	$582.60	$1,283.40	$700.80	14.88%	Low Price
Msoft	D2 v3	2	8	$507.12	$1,313.04	$805.92	Low Price	2.31%
AWS	m5.xlarge	4	16	$1,077.48	$2,689.32	$1,611.84	6.36%	4.78%
GCP	n1-standard-4	4	15	$1,165.08	$2,566.68	$1,401.60	15.01%	Low Price
Msoft	D4 v3	4	16	$1,013.04	$2,624.88	$1,611.84	Low Price	2.27%
				Compute Optimized				
AWS	c5.large	2	4	$473.04	$1,278.96	$805.92	8.80%	12.63%
GCP	n1-highcpu-2	2	1.8	$434.76	$1,135.56	$700.80	Low Price	Low Price
Msoft	F2 V2	2	4	$438.00	$1,243.92	$805.92	0.75%	9.54%
AWS	c5.xlarge	4	8	$946.08	$2,557.92	$1,611.84	8.80%	12.63%
GCP	n1-high-cpu-4	4	3.6	$869.52	$2,271.12	$1,401.60	Low Price	Low Price
Msoft	F4 v2	4	8	$876.00	$2,487.84	$1,611.84	0.75%	9.54%
				Memory Optimized				
AWS	r4.large	2	15.25	$735.84	$1,541.76	$805.92	7.41%	8.06%
GCP	n1-high-mem-2	2	13	$726.00	$1,426.80	$700.80	5.97%	Low Price
Msoft	E2 v3	2	16	$685.08	$1,491.00	$805.92	Low Price	4.50%
AWS	r4.xlarge	4	30.5	$1,471.68	$3,083.52	$1,611.84	7.42%	8.06%
GCP	n1-high-mem-4	4	26	$1,452.00	$2,853.60	$1,401.60	5.98%	Low Price
Msoft	E4 v3	4	32	$1,370.04	$2,981.88	$1,611.84	Low Price	4.50%

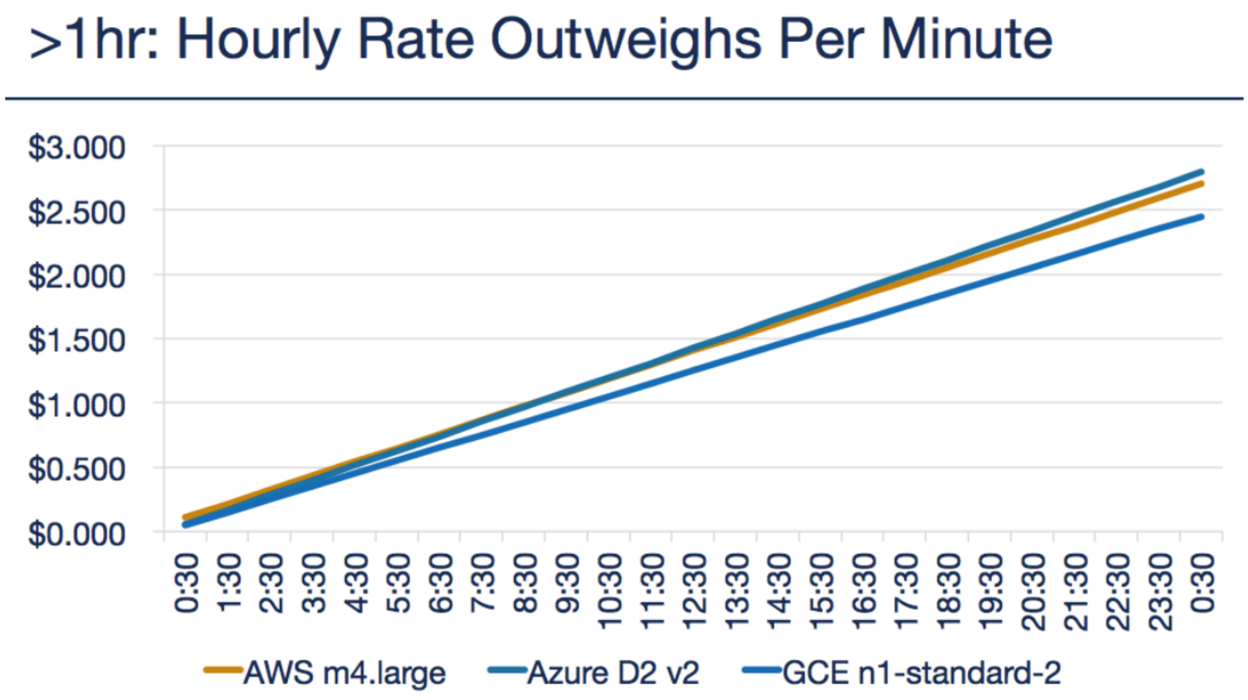
Heroix
165 Bay State Drive, Braintree, MA 02184 USA
www.heroix.com, info@heroix.com

3.4 OUR RECOMMENDATION

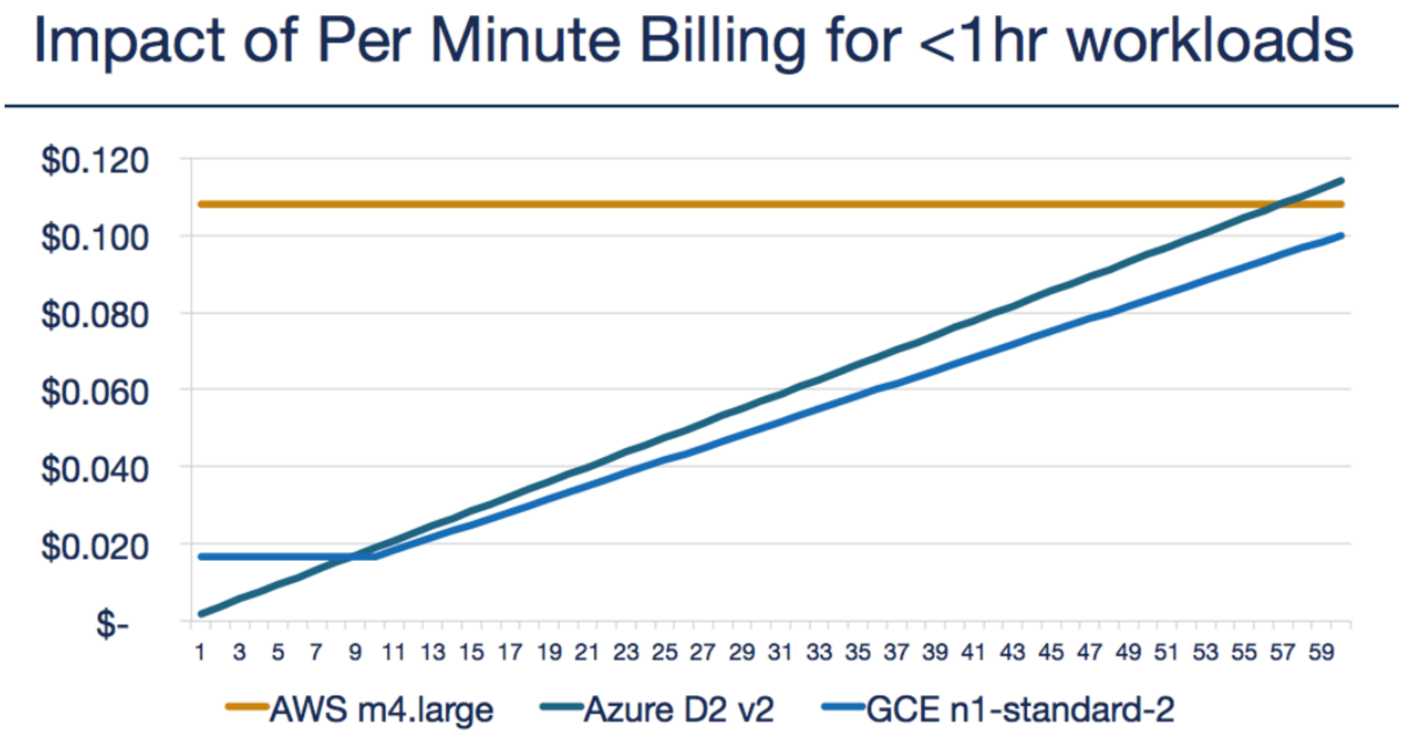
AWS is little expensive to begin than Azure but lower in cost than Google cloud. However the maintenance cost of AWS is hourly basis where as for other options those are minutes basis.



General Purpose								
AWS	m5.large	2	8	$534.36	$1,340.28	$805.92	5.37%	4.43%



>1hr: Hourly Rate Outweighs Per Minute



Impact of Per Minute Billing for <1hr workloads

4. TECHNOLOGY STACK

kennedia carefully accessed the best possible technology stack for development of WEB, Android and iOS apps with REST API (Web services)

Platform	Technology	Why to choose
iOS App - iPhone 6 onwards	Swift 5.0	SWIFT framework is better than objective c, since its light weight and support latest iOS SDK 12.2. Objective C has NOT released a new version since long.
Android App - Version 6 above.	Java, Android SDK, Android Studio	These technologies are best to use for Android development. We will be using latest versions.
Website and API	CI	CI is a powerful and open source MVC framework. Light weight and enables coder with lot of flexibility. Maintaining CI code is also less expensive.

4.1 MISC. COST ESTIMATION

Platform	Registrations	Cost	Remarks
Android	Google Play Store	USD 25.00	One time cost
iOS	Apple store developer account	USD 100	Per year
Web	Image used on Set up page	Not Known	
Cloud Costing	As per your choice	Refer 4.3	Refer 4.3

5. CONCLUSION

During the development of the proposal, the initial information was collected through our discussion and mobile design, to analyse the requirements to be able to formulate a feasible infrastructure.

This paper hopes to be handy to the design of architecture infrastructure and an important contribution for decision making I’m choosing cloud and development technologies.