

Automated Health Check Report Generation Process for Software Applications using Azure

Sanjay P¹ Sreenivasa T.N²

^{1,2}Department of Mechanical Engineering

^{1,2}AMC Engineering College, Bangalore- 560083

Abstract— Azure is a growing collection of integrated cloud services—analytics, computing, database, mobile, networking, storage and web for moving faster. Azure is a cloud computing platform and infrastructure created by Microsoft for building, deploying, and managing applications and services through a global network of datacenters [1]. Application health check is done in software industries to minimize the risk of issues arising in software applications and to ensure that the system is completely utilized to gain the maximum benefit. This is to ensure that software is running properly without any issues and to make sure that all potential issues are proactively identified and remedied, and that application is being used to its fullest potential value [2]. System health check will help in eliminating risk of system failure, advice on system usage, methods of recording recovery and expert recommendations for possible enhancements to avoid issues in the software system applications [2]. In the present investigation, an application module was developed using .Net technologies (C#, SQL Server, Visual Studio and Azure) to automate the process of health check report generation for software applications and to send the mail to respective recipients along with the health check report. This is the MIS (Management Information System) report which provides information to the Management about health check of the applications, services running and databases of software application system.

Key words: Azure, Cloud Platform, MIS Report, Health check report

I. INTRODUCTION

A. Health Check Of The Software Systems

This health check service will focus on what the customer needs and determine the operating condition (health) and current performance of the applications, database, or server system after a set of following analyses [3]:

- Server Analysis: Examines and assesses the utilization of the hardware configuration including, for example, the CPU(s), memory and disks in relation to the usage of the software environment or application.
- Environment Analysis: Reviews how well the software environment is being used to advantage, including its architecture, background processes, start-up parameters, application layouts and built-in functionality.
- Application Analysis: Modern business applications include a layered architecture with distinct presentation, business logic and data access. So the health check should be done for running the application and its databases efficiently.
- Database Condition: Determines how well or poorly the data is structured and segmented in the database and the implications for system performance. It also tells about the usage of database systems so that corrective actions

can be taken if the database performance is slow or it is full.

- Disaster Recovery: Evaluates the status and potential effectiveness of your disaster recovery plan, backup strategy and redundancy.

The health check service provides the following benefits: ,,

- Finds improvement opportunities by precisely measuring and evaluating current application performance
- Minimizes or eliminates costly surprises like performance degradation, bottlenecks disaster preparation and memory/size related issues.
- Prolongs the life of current hardware ,,
- Provides guidance on how the system is likely to perform in the future and the impact on the plans

B. Health Check Output

The output of health check service for software applications includes a detailed and specific report on the condition of software at any recommendations for improvements in:

- Server hardware
- Software environment
- Database
- Disaster Recovery plans
- Reduced risk of surprise problems in the near future
- A standard report format you can compare with future health checks to identify changes

II. OVERVIEW

In order to make the process of generating the health check report automated, there has been a plan to write a computer program which reads the status of each resource available in Azure Platform. This can be achieved with the help of following technologies:

- PowerShell Cmdlets
- SQL Queries
- Azure Resource management APIs.

III. SOLUTION IMPLEMENTATION PROCEDURE

Health Check report developed for system application will provide the information on the following aspects:

- Health check of applications and services.

This provides the following information.

- Statuses: This will provide the status of each application and services. The status of application shown are:
 - Stopped (represented by Red color)
 - Running (Green)
 - New services (Blue)
 - Queues that are having any concerns (Yellow)
 - Environment (either production or development)
 - Solution Name present in code repository (like Git Hub)
 - Description of the application/services.
 - Any remarks provided (ex: for stopping the services)

- CPU usage of the system
- Scaling and Instance status
- Queue Status: This provides the information on the following queues
- Provides information on status, current size, max size, queue length and last accessed time of the queues.
- SQL Database status
- Live databases and its consumption: Like DB Name, available size, used size and percentage used.
- Tables and its consumption/usage
- Read the status of all available resources in the Azure and send Health Check Report to the email id's of the recipients that are configured in the system.
- Get statuses of following
 - Cloud Services using Powershell Cmdlets.
 - Databases using queries.
 - Queues using Azure Management APIs

IV. SPECIFICATIONS

- Execute Powershell Cmdlets to start/stop resources.
- In this case, track remarks for each user action (start/stop) on cloud services. If user doesn't want to track it, then avoid using the Powershell Cmdlets to control the Cloud Services.
- For tracking the database properties use queries.

- Queues status properties will be tracked using Azure Management API's.

V. MILESTONE PROCEDURES

- Automated notifications to the team for Azure Resources Health Check
- Read status of each cloud services, queues, databases and create the report with separate sheets for each kind of health check sheets programmatically.
- Remarks for each resource are not possible in automated process. So get remarks from the user upon either starting or stopping the cloud service.
- Provide Control Panel by which the user can control the Cloud Services with remarks. With last remarks provided by the user and based on status of azure resources, system will generate the Health Check Report and send this report to respective recipients (email ids configured in the system).
- Resource Management using APIs and Powershell Control Azure Resources from the Control Panel connected to the Resources available in Azure.

VI. ARCHITECTURE/FUNCTIONAL FLOW

The Architecture for automating the health check report is as shown below.

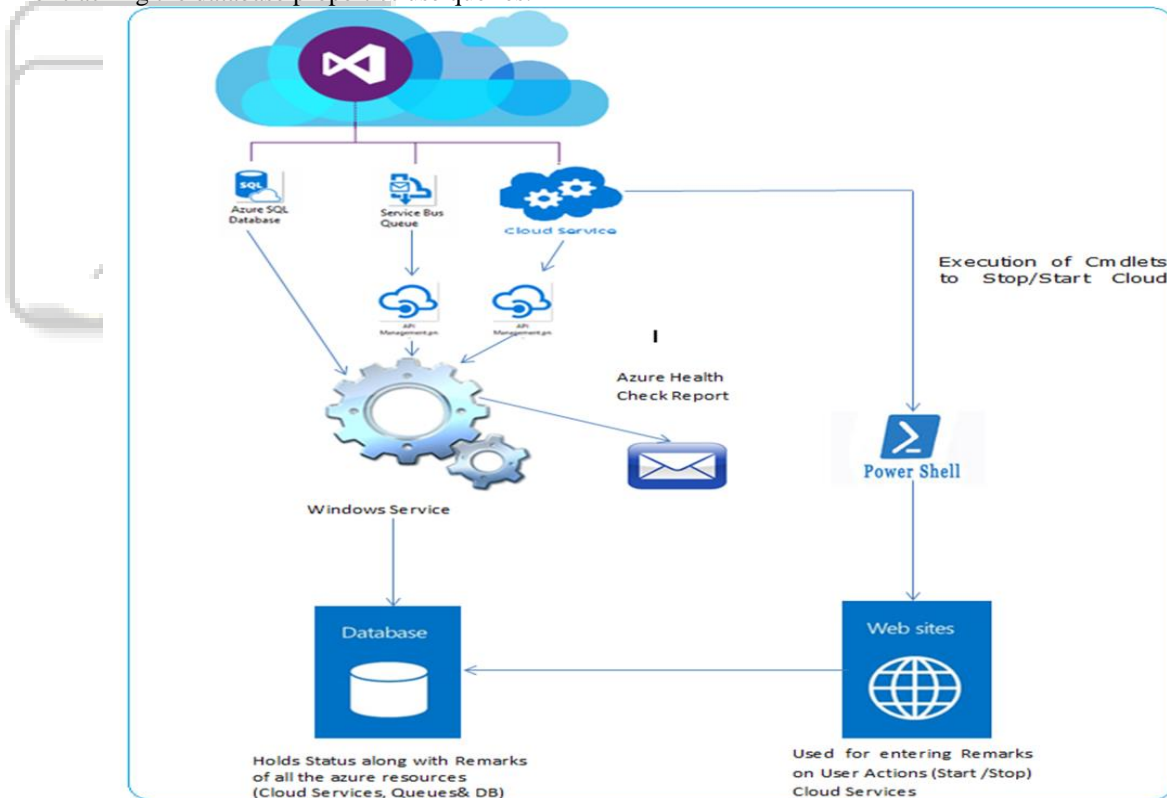


Fig. 1: Architecture/Functional Flow

VII. SOLUTION IMPLEMENTATION DETAILS

The Windows service will check availability of resources like (Cloud Services, Databases, Queues) in azure before checking health of each resource in Azure.

Windows Service will also read status of Cloud Services and Queues using Azure Management APIs. These APIs are provided by Azure. Then it will read the database status properties using database queries.

For entering remarks for the actions on Cloud Services like stop/start, provide a Web interface page which lists all Cloud Services with the status, remarks, actions (start/stop). If the users want to start/stop the Cloud Service, they should provide remarks providing reasons. These remarks will be saved in the database table along with the cloud service status information.

After processing all above steps, windows service will send an email with Azure Health Check Report to recipients configured.

VIII. USER INTERFACE PROVIDED

The user interface developed is shown in the below mentioned figure for control panel.

Custom Azure Control Panel, v1

S.No	Cloud Service	Status	Scaling / Instance Status	Description	Remarks	Action
1	3060Messages	Stopped	Manual/ 1 Instance Stopped(Max 1)		This service has been stopped temporarily	Start Submit
2	Backupmessage	Stopped	Auto/6 Instance Stopped(Max 6)	Archive retrieval process into the development database	This service has been stopped temporarily	Start Submit
3	Backup3060	Running	Auto/1 Instance Running(Max 8)	Archive retrieval process into the development database		Stop Submit

Fig. 2: Custom Azure Control Panel,V1

IX. RESULTS

A. Health Check Report Of The Application/Services

Health check Dashboard							
Sl.no	Application / Service	Status	Environment	Solution Name in Github	Description	Remarks	CPU Percentage
1	3090MessageMove	Stopped		3090Messages	Move 3090 messages and take backup.	As per Dev team, this service is temporary stopped	10%
2	EventBackup	Stopped	Development	BackUpEventMsg	Archive retrieval process into the development database	As per Dev team, this service is temporary stopped	5%
3	MessageupRaw	Running	Development	MessageRaw	Archive retrieval process into the development database		0.30%
4	BackupRaw	Running	Development	BackupRaw	Archive retrieval process into the development database		0.49%
5	BackupTCP	Stopped	Development	BackUpTCP	Archive retrieval process into the development database	As per Dev team, this service is temporary stopped	20%
6	BackupTCP2	Stopped	Development	BackUpTCP2	Archive retrieval process into the development database	As per Dev team, this service is temporary stopped	9%
7	PolicyCostReport	Running	Production	PolicyCostReport	Generates the report about cost of the selected policy.	This in production	98.88%
8	dailymedicalpolicyreport	Running	Production	DailyLivePolicyReport	Generates the report about the medical policies that are in LIVE and its details	This in production	98.93%
9	dailymedicineconsumedreport	Running	Production	DailyMedReport	Generates report about the total medicine consumed	This is in pre-production	98.41%
10	devicemessagearchivalservice	Running	Production	DeviceMessageATS	Archives messages from polycyessage table	This is new service and in production	4.61%
11	backupWR3090	Running	Production.	MoveBackUpATS	Moves the retrieved back up data from 'updateraw' to event and position queues	Not all role instances are ready. 4 Instances: 2 Running, 2 Busy	32.79%

Fig. 3: Helth Check Dashboard

B. Health Check Report for Queues Status

Service Bus: insurancepolicyqueue						
Name	Status	Current Size	Max Size	Queue Length	Last Accessed Time	
policydata_for_import	Active	1.13 MB	16 GB	18	3/9/2016 20:57	
meddataqueue	Active	0 B	5 GB	0	1/20/2016 13:38	
consumptiondataqueue16gb	Active	17.18 MB	16 GB	274	3/11/2016 12:10	
livepolicydataqueue	Active	89.99 KB	5 GB	11	1/18/2016 10:16	
ncd10standsqueue	Active	32.8 KB	16 GB	4	3/11/2016 12:11	
Service Bus: InsQueueHist						
Name	Status	Current Size	Max Size	Queue Length	Last Accessed Time	
insurancedata	Active	0 B	16 GB	0	11/12/2015 22:14	
policydata	Active	0 B	16 GB	0	11/9/2015 20:07	
medicalstock	Active	0 B	16 GB	0	3/11/2016 12:12	
patientdetails	Active	1.47 MB	16 GB	183	3/11/2016 12:12	
tcpevent	Active	13.54 MB	16 GB	216	3/11/2016 12:13	
opdpatients	Active	12.48 MB	16 GB	199	3/11/2016 12:13	
policydetails	Active	10.16 MB	16 GB	162	3/11/2016 12:14	
insruancesstatus	Active	81.57 MB	16 GB	1301	3/11/2016 12:14	
Service Bus: 3090dbsystem						
Name	Status	Current Size	Max Size	Queue Length	Last Accessed Time	
policydatadata	Active	0 B	16 GB	0	3/11/2016 12:15	
meddata	Active	30.91 MB	16 GB	3861	3/11/2016 12:15	
policyreference	Active	0 B	16 GB	0	11/6/2015 14:30	
messagedata	Active	0 B	16 GB	0	3/11/2016 12:16	
tcpevent	Active	5.08 MB	16 GB	81	3/11/2016 12:16	
policylive	Active	276.29 MB	16 GB	4407	3/11/2016 12:16	
insstatus	Active	314.16 MB	16 GB	5011	3/11/2016 12:17	

Fig. 4:

C. Health Check Report For Databases And Tables

Databases and its consumption:				
Database Name	Available Size	Used Size	Percentage Used	
3090	300 GB	135.55 GB	45%	
3060	500 GB	361.68 GB	72%	
Backup_3090	250 GB	2.39 GB	1%	
MedicalReportApplication_db	1024 MB	3 MB	0%	
HealthCheckMaster	400 GB	11.34 GB	3%	
SendSMSandMail	500 MB	14 MB	3%	
HIPAAANCDSTD	1024 MB	8 MB	1%	
MedRiskData	500 GB	195.35 GB	39%	
MedicineStock	1024 MB	93 MB	9%	
Tables and its Consumption:				
Database Name	Available Size	Table Name	Used Size	Total Count
3090	500 GB	ServiceMessage	237.91 GB	3482861930
3060		PolicyReference	99.46 GB	1478756541
3090		Other tables	24.31 GB	

Fig. 5:

X. CONCLUSIONS

An automated health check report was developed using .Net technologies (C# and SQL server using Azure platform) in order to determine the size and performance of queues, tables and databases.

This report can be used to take the decision to improve the performance of the systems.

Automated the process of reading statuses of Azure resources and send Azure Health Check report along with

the email to the respective recipients configured in the system.

Tracked remarks for actions being taken to start/stop cloud service was be done through the web page (user interface developed, shown in section:8)

REFERENCES

- [1] azure.microsoft.com/en-in/
- [2] <http://www.wilmacco.com/system-health-check>
- [3] [https://www.progress.de/docs/default-source/default-document-library/ Progress/Documents](https://www.progress.de/docs/default-source/default-document-library/Progress/Documents).