Health Checking

MIHÁLY ÁRVAI

Introduction

- Mihály Árvai
- Senior Sitecore Developer Mirum
- ➤ ~ 10 years experience in Sitecore
- ► Hackathon winer 2019
- ► Twitter
 - ▶ @Mitya_1988
- ► Slack
 - ► Mitya88
- Medium
 - @mitya_1988
- ► GitHub
 - ► Mitya88

Agenda

- ► Health checking
 - ► Mhàs Homs
- Advanced Sitecore Health check module
 - ► How is it working?
 - ► Features
- Writing your own custom health check
- ► Further improvements



What is a health check?

- An external application or a built-in endpoint/service
- Checks if an application or component is working
- Can determine an actual or upcoming issue
- Can notify the IT or Devs about the issue
- Helps in the investigation
- Tests your application

Component states

Healthy

Warning

Error

Importance of health checking

- avoid or minimize service/application downtimes
 - ▶ Renew certificate/licence in time
- Faster resolution for an issue
 - Losing time with log files investigation
- Catch random or periodic issues
 - Something happens around the reguler application recycle time
- ► Test the latest deployment

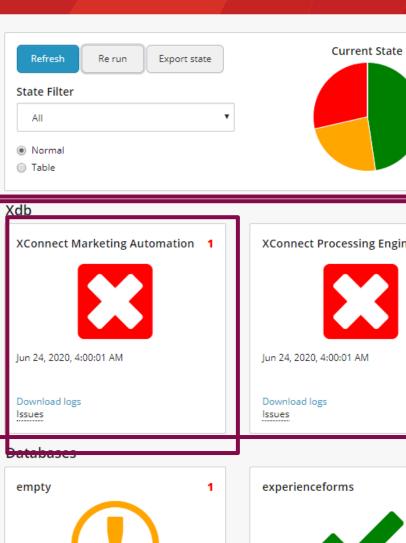
Advanced Sitecore Healthcheck module

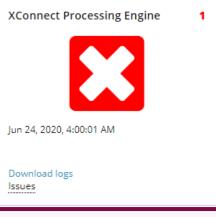
https://github.com/Mitya88/SitecoreHealthcheck

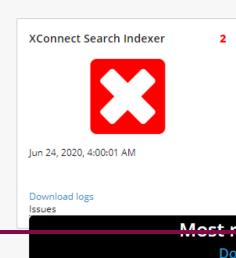
Contributors

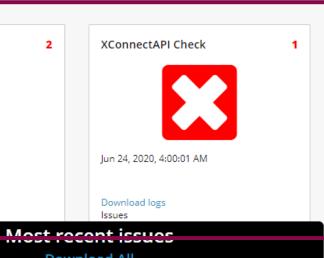
- Andrei Ardel Sitecore Developer Mirum
- Attila Horváth Lead Sitecore Developer Mirum

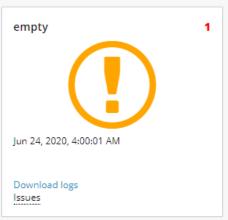
Advanced Sitecore Healthcheck

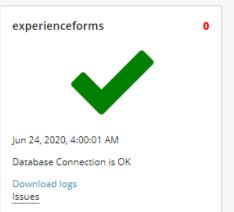


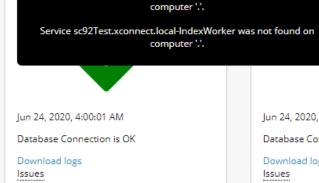






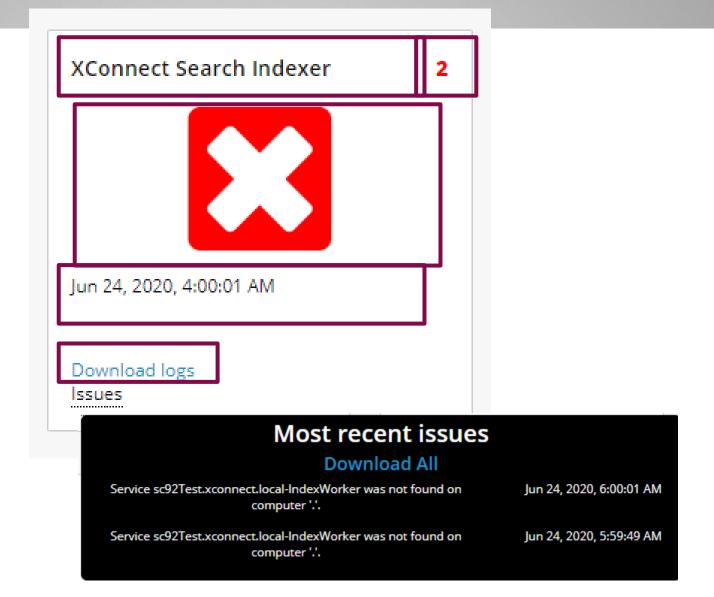




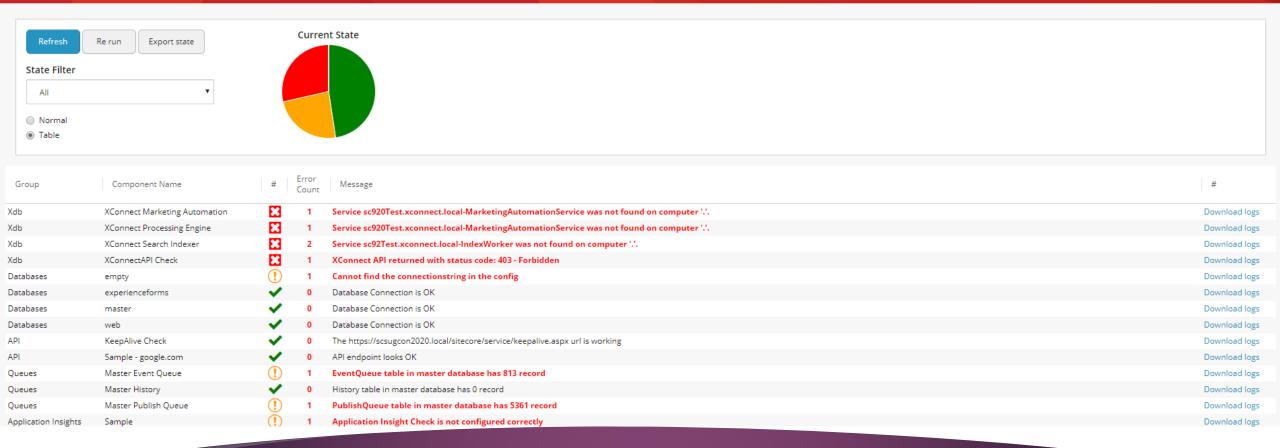


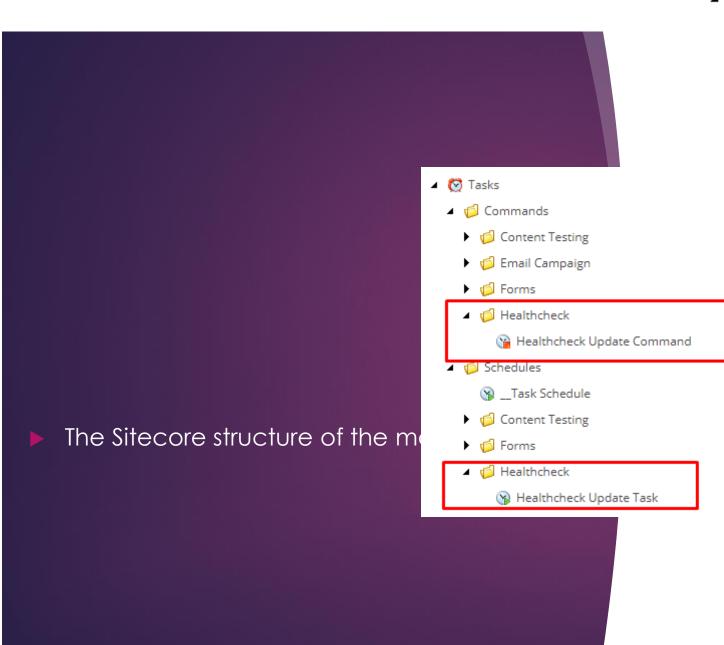


https://twitter.com/mitya 1988



Advanced Sitecore Healthcheck





- Modules
 - Healthcheck Module
 - API Methods
 - Certificate Lookups
 - Cleanup Lookups
 - - ▶ 🔞 Xdb
 - @ Databases
 - API
 - Queues
 - & Master Event Queue
 - Master History
 - & Master Publish Queue
 - ▶ (Application Insights
 - ▶ (Certificates
 - ▶ 🔞 ltems
 - ▲ 【 License
 - License Check
 - ▶ (Logfiles
 - Levels
 - Settings \$

Days - Defines the number of days after the error messages will be delet	ted:	
1		
Mail Settings		
Sender Name:	✓ Masks ✓ Gommands	
mihaly.arvai@mirumagency.com	► Content Testing	
Sender Email:	►	
mihaly.arvai@mirumagency.com		
Recipient Emails:	✓ Schedules	
mihaly.arvai@mirumagency.com		
Subject:	Forms Healthcheck	
Healthcheck	(§) Healthcheck Update Task	
Body:		
Error Level - Defines the error level when automatic email should be sen	nd:	
Healthy		

. 🥯 🗚 1 1

This is a summary of components after Healtcheck:

Component name	Last check	Status	Message		
Databases					
core	5/22/2020 7:12:57 AM	Healthy	Database Connection is OK		
experienceforms	5/22/2020 7:12:57 AM	Healthy	Database Connection is OK		
master	5/22/2020 7:12:57 AM	Healthy	Database Connection is OK		
security	5/22/2020 7:12:57 AM	Healthy	Database Connection is OK		
web	5/22/2020 7:12:57 AM	Healthy	Database Connection is OK		
API					
KeepAlive Check	5/22/2020 7:12:57 AM	Error	The remote server returned an error: (403) Forbidden.		
Sitecore Identity Server	5/22/2020 7:12:57 AM	Error	API returned with status code: 502 - Bad Gateway		

Component Templates

Data

- Name
- Status
- Last Check Time
- Error Messages
- Healthy Message

Configuration

 Different fields based on the component

- MSSQL Databases
 - ▶ Not for only Sitecore related databases, for external databases
 - MongoDB is added as a Custom Check
- APIs
 - ► GET/POST
 - Request Body
 - Response Code and Body Verification
 - Supports Authentication (JWT, Basic, Certificate)

- Windows Services
 - Check state of the windows service (Solr or XDB related services)
- WebJobs
 - Check state of the WebJob (XDB related or custom webjobs)

- Application Insights
 - Checks and reads errors/warnings from the configured ApplicationInsight ina given time period
- Log Files
 - Checks and reads errors/warnings from the Sitecore log files in a given period

- XConnectAPI Check
 - Calls XConnect APIs with client certificate
- Keep Alive Check
 - Calls the keepalive.aspx
 - Uses WebUtil.GetFullUrl

- EventQueue, PublishQueue, History table record count check
 - Checks the number of records in these tables in every database
- Item Check
 - ► Check if the given item exists in the configured db/language

- ► Licence Check
 - ► Checks when the licence will expire
- Certificate Check
 - ► Checks when the configured certificate will expire

- Search Check
 - Checks if index contains the configured number of documents
 - Query can be defined
- Custom Check
 - Implement your own check
 - Samples included

Implementing custom check

- ▶ Reference the Healthcheck.Service.Customization binary
 - (It will be published on nuget.org soon)
- Create a class, derive it from CustomHealthcheckRun
- Override DoHealthcheck method
- Create Custom Healthcheck item in Sitecore and configure the fields

Implementing custom check

```
0 references | 0 changes | 0 authors, 0 changes
public class MongoSample : CustomHealthcheckRun
{
    2 references | 0 changes | 0 authors, 0 changes
    public override CustomHealthcheckResult DoHealthcheck(NameValueCollection parameters)
    {
        throw new NotImplementedException();
    }
}
```

Implementing custom check

```
6 references | Árvai Mihály, 43 days ago | 1 author, 1 change public class CustomHealthcheckResult {
5 references | Árvai Mihály, 43 days ago | 1 author, 1 change public HealthcheckStatus Status { get; set; }
4 references | Árvai Mihály, 43 days ago | 1 author, 1 change public string ErrorMessage { get; set; }
2 references | Árvai Mihály, 43 days ago | 1 author, 1 change public string HealthyMessage { get; set; }
2 references | Árvai Mihály, 43 days ago | 1 author, 1 change public Exception Exception { get; set; }
}
```

```
public override CustomHealthcheckResult DoHealthcheck(NameValueCollection parameters)
   var connectionString = parameters["ConnectionString"];
   var result = new CustomHealthcheckResult
       HealthyMessage = "Connection is ok",
       Status = HealthcheckStatus.Healthy
   };
   try
       if (string.IsNullOrEmpty(connectionString))
           result.Status = HealthcheckStatus.Warning;
           result.ErrorMessage = "Connectionstring is not provided in 'ConnectionString' param";
           return result;
       var client = new MongoClient(connectionString);
       if (client.Cluster.Description.State != MongoDB.Driver.Core.Clusters.ClusterState.Connected)
           result.Status = HealthcheckStatus.Error;
           result.ErrorMessage = "State is disconnected";
   catch (Exception ex)
       result.Status = HealthcheckStatus.Error;
       result.Exception = ex;
       result.ErrorMessage = ex.Message;
   return result;
```

Modules Healthcheck Module Configuration API Methods Type: Certificate Lookups Cleanup Lookups Healthcheck. Service. Samples. Mongo Db Component Health, Healthcheck. Service. SamplesComponents ▶ 🔞 Xdb Parameters: ■ (Databases ConnectionString mongodb://localhost:27017/myproject Custom Healthcheck empty master 🜠 web Data API Queues Name: Application Insights MongoDB Check Certificates 🕨 ز ltems Status: ▶ (License Warning Logfiles

DEMO

Azure PaaS XP Step by Step setup: https://www.youtube.com/watch?v=J_qk7jT_Y-U

Further improvements

- Disk space check
- More support on XConnect and on on-premise installation.
- Check Reporting Processing
- Page Speed Tests