Setup Azure for Student Projects

This document is an attempt to describe the installation and configuration of everything that is needed for “Azure for Student Projects” (A4SP). Using A4SP it is possible to provide access to a group of students within an azure environment in which costs are controlled, access to resources are limited and administration tasks are reduced to a minimum.

As a starting point for the installation

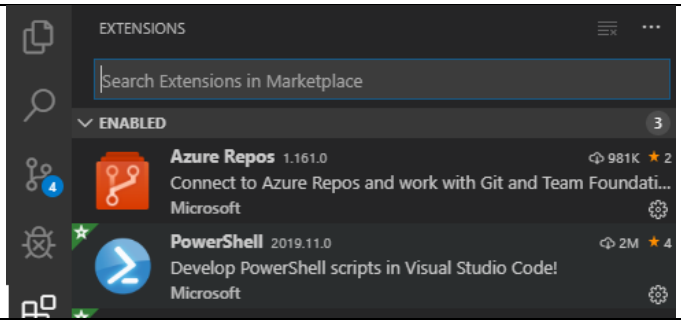
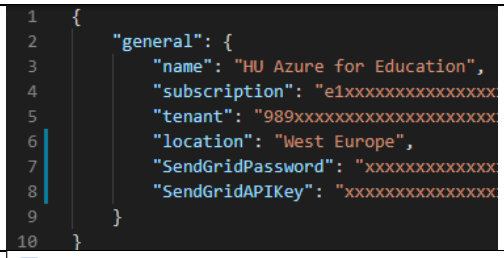
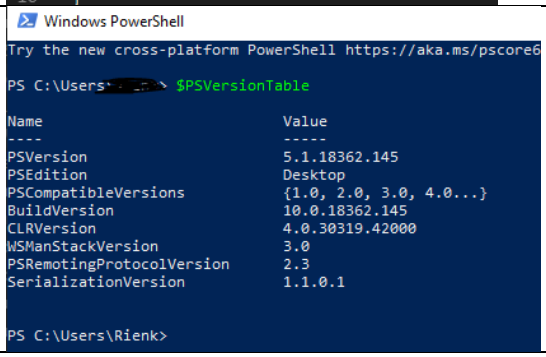
- This document is based upon an Azure customer with an Enterprise Agreement (EA).
- Everything that must be done just once will be done using the Azure Portal. Rational is that automation would take to much time and moreover, it would be less clear for an administrator what is being accomplished by running the script.
- Everything that runs on a regular basis is scripted using PowerShell (Az-modules only where possible).

Setup

To do the Setup, follow all steps in this document and in this order using the exact names as provided. It is assumed that you are the owner of a new Dev/Test subscription in your Azure account having a trust relationship with an Azure Active Directory containing the Users of your organization.

Prerequisites

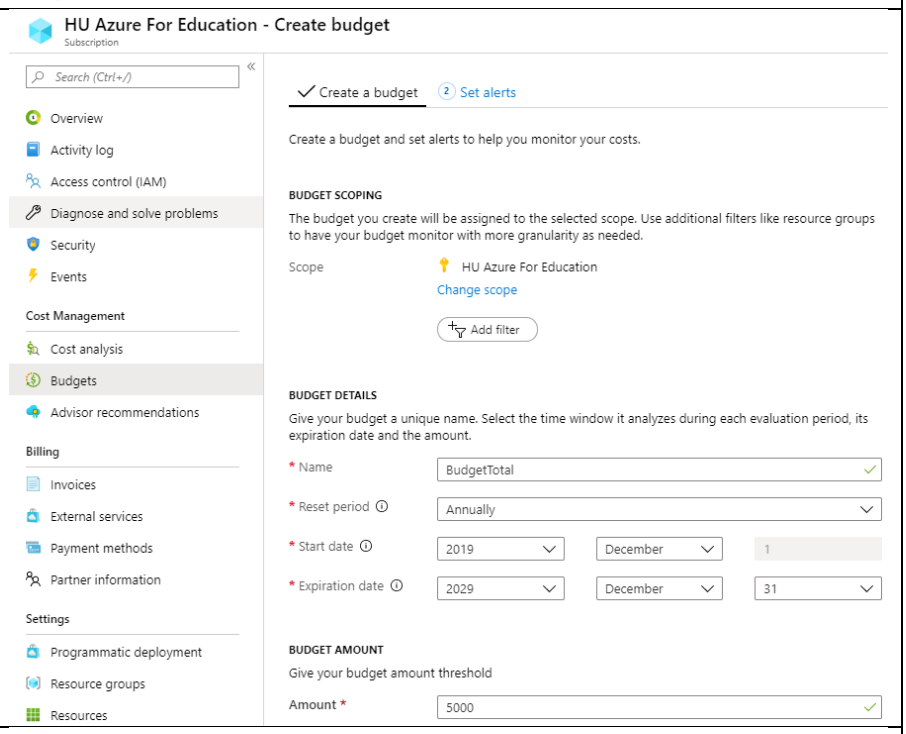
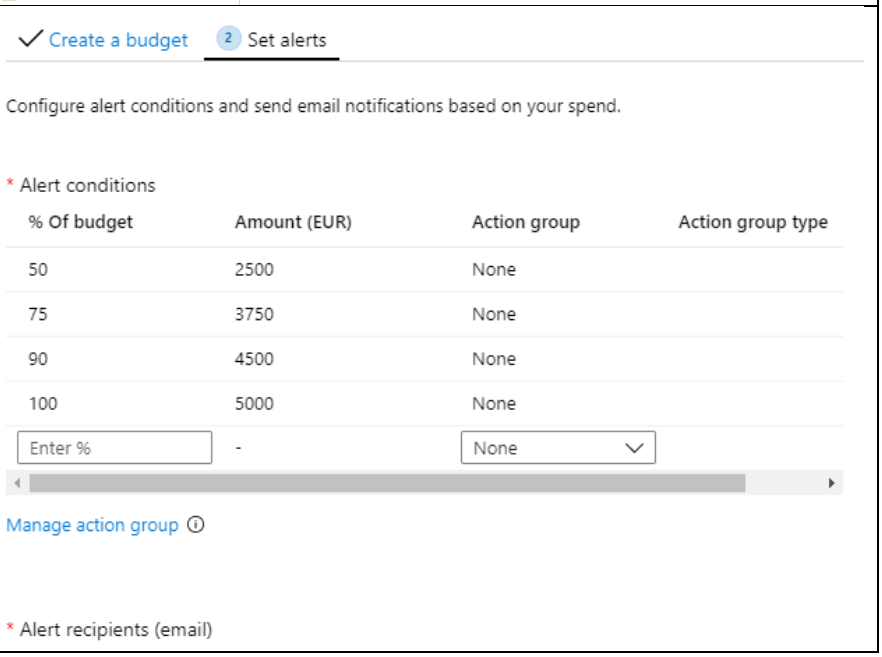
Setup your environment, first steps.

#	Description	Image
1	Install Visual Studio Code using these extensions.	
2	Download / Clone / Whatever: Get the files from this url.	https://dev.azure.com/rienkvanderploeg0902/Azure4StudentProjects/_git/AzureForStudentProjects
3	Rename .\Settings_Settings.json to .\Settings\Settings.json and provide the information that is appropriate for your organization. Note: SendGrid info is part of a later step, don't worry.	
4	Install Powershell for Azure. Check your PowerShell version first: ;# versie >= 5.1	
5	Install Az –PowerShell module. Accept NuGet provider and PSGallery as being trusted repositories if requested.	Install-Module -Name Az -AllowClobber -Scope CurrentUser
6	From a PowerShell console, run the Setup script	.\Setup\setup.ps1

Capability to send Alerts related Costs on the Subscription Level

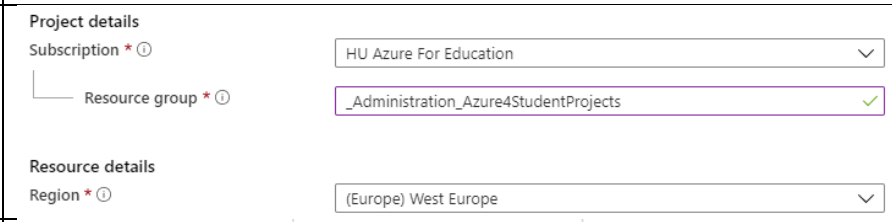
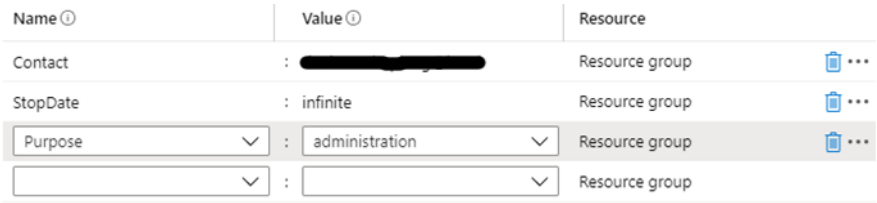
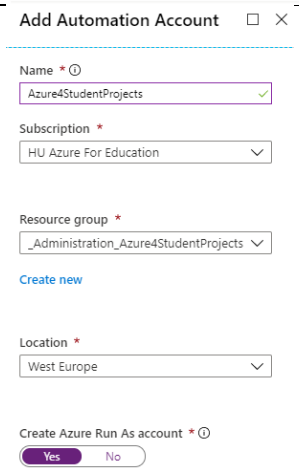
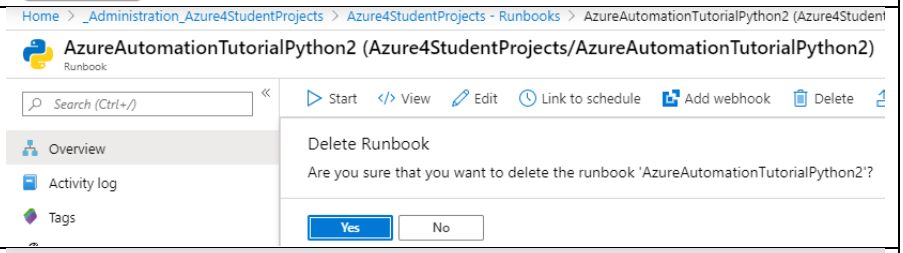
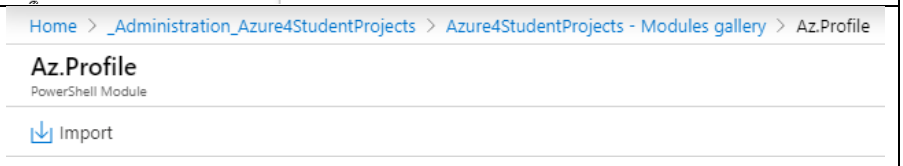
First things first:
You should be warned about high expenses. This can be done by creating a budget on the subscription level.

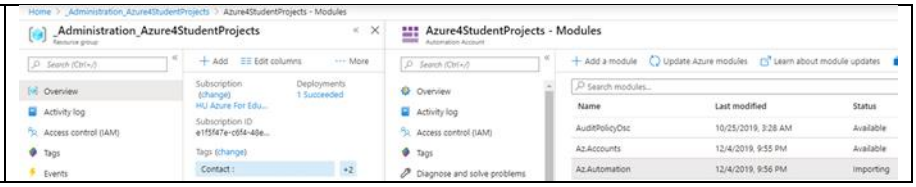
Note: not described, but if you like, create budgets for administrative resource groups like "_Administration_Azure4StudentProjects", "_AppServicePlans" and "NetworkWatcherRG" (standard: name=rg-name, budget=500 euro, alert=50%, action=sendAlertToAdministrator)

#	Description	Image
1	Goto Subscription and "Create budget" Scope: Subscription! Name: BudgetTotal Reset periode: Annually Start date: First of this month Expiration date: Last day of Last available year in the drop down box. Budget: Depending on your budget / calendar year, for instance 5000 (euro).	
2	Set appropriate alerts, for instance at 50% / 75% / 90% and 100%. Fill in alert recipients, more than one of course!	

Create an Automation Account

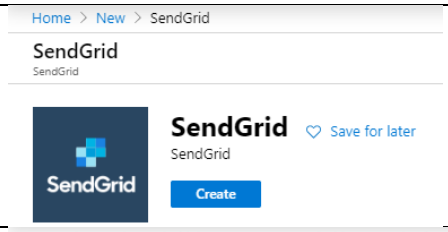
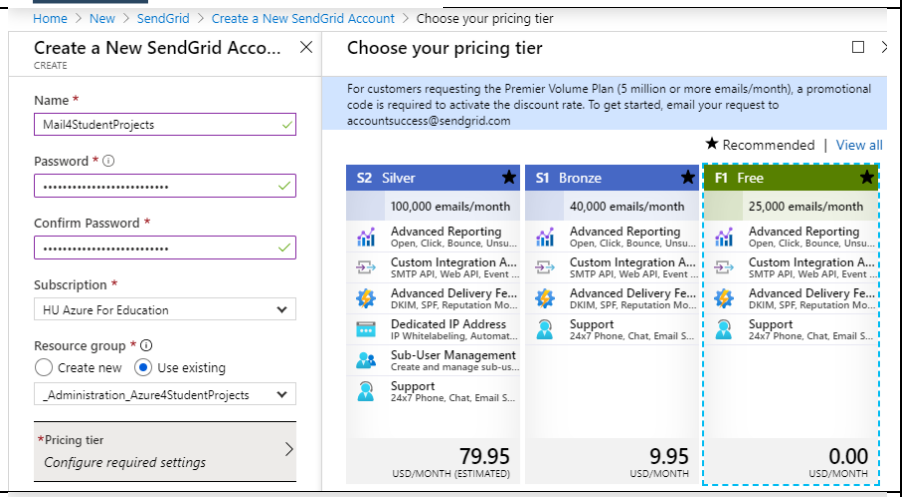
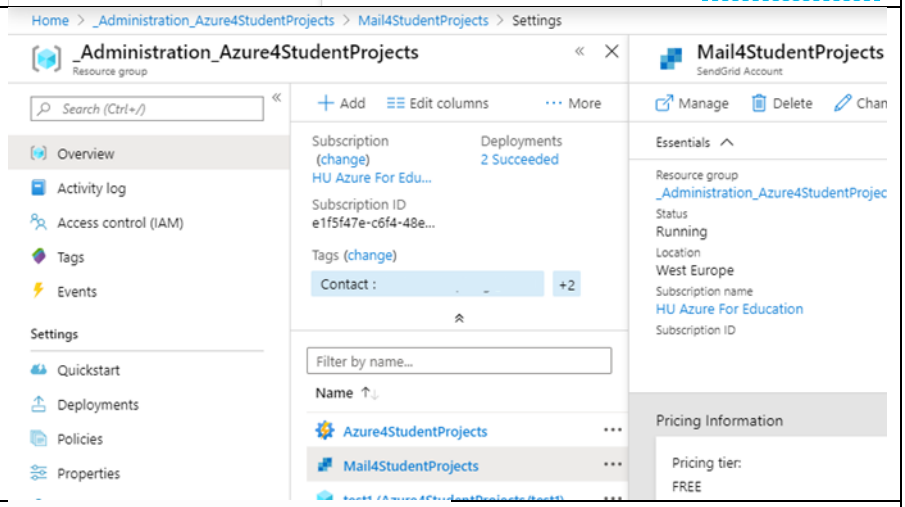
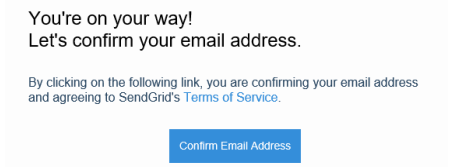
In order to be able to run scripts automatically, for instance to send reports or check budgets and take action if budgets are exceeded you must create an automation account within your subscription.
This should be done by the owner of the subscription.

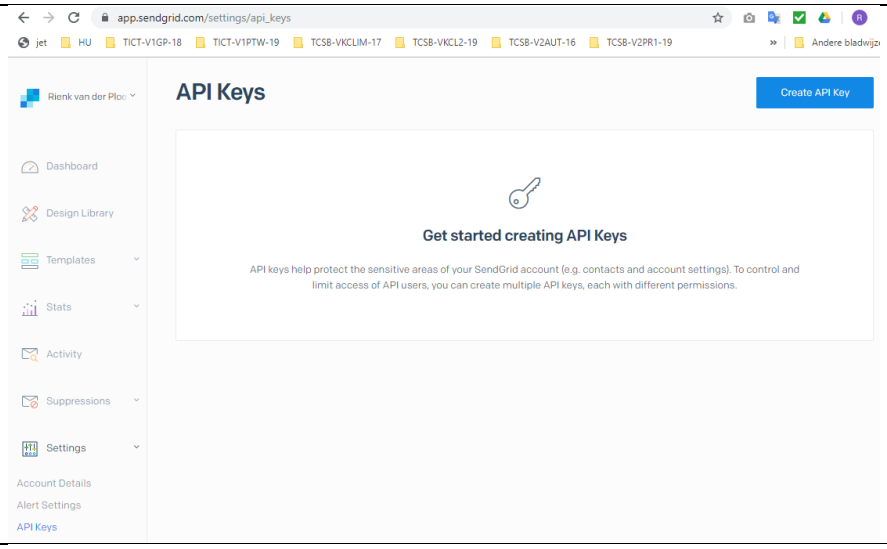
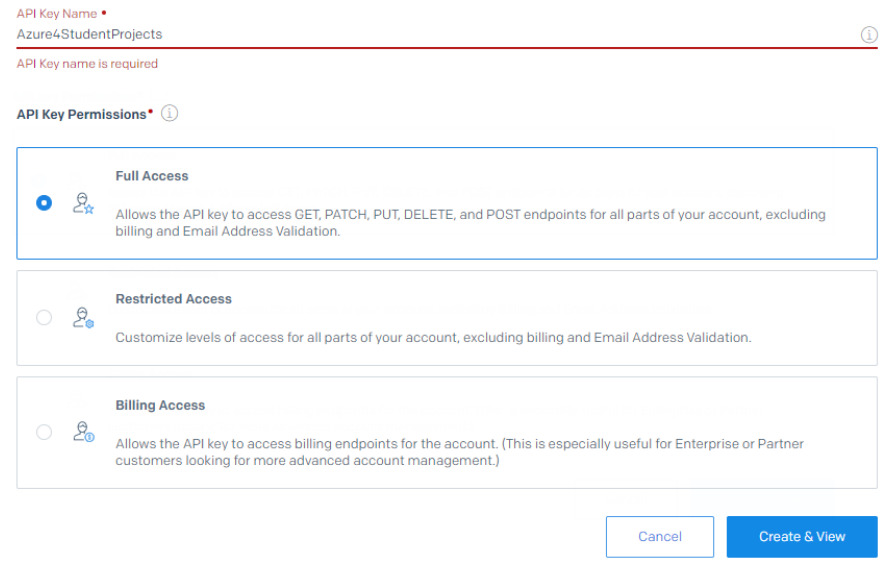
#	Description	Image
1	Create a new ResourceGroup (RG) starting with underscore to distinguish from StudentProjects. Name is: `_Administration_Azure4StudentProjects`	
2	Provide it with a few tags: Contact, StopDate and Purpose. This is important , for instance, this will prevent being deleted later.	
3	Add an automation account in this RG. Name must be: `Azure4StudentProjects` Type is: “Run As Account”	
4	Remove all (3) Tutorial scripts from the Resource Group.	
5	Import the Az.Profile module from the Modules Gallery in this automation account. Import more Az.modules in this order: Az.Accounts Az.Consumption Az.Automation Az.Resources Az.Profile Az.KeyVault Az.Monitor Az.Network Az.Storage Az.CosmosDB	

6	Check if import succeeded (might take a long time!) if you like.	

Capability to Send E-mail from an automation runbook Using SendGrid in Azure

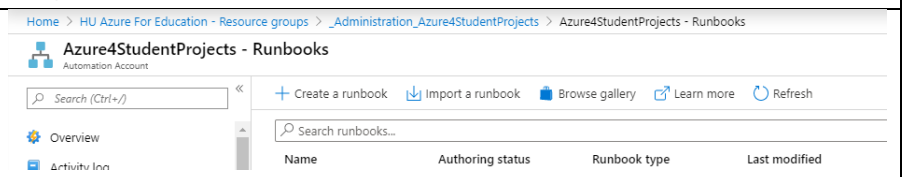
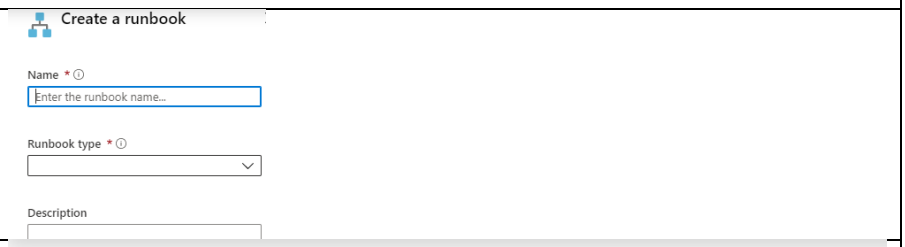
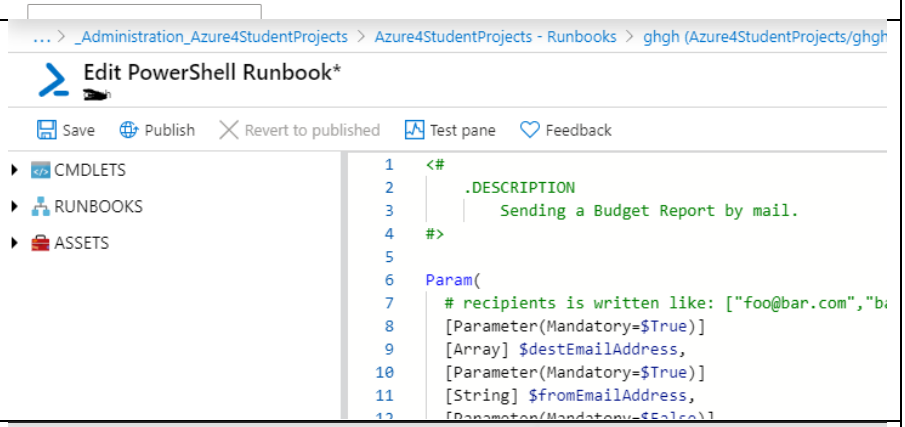
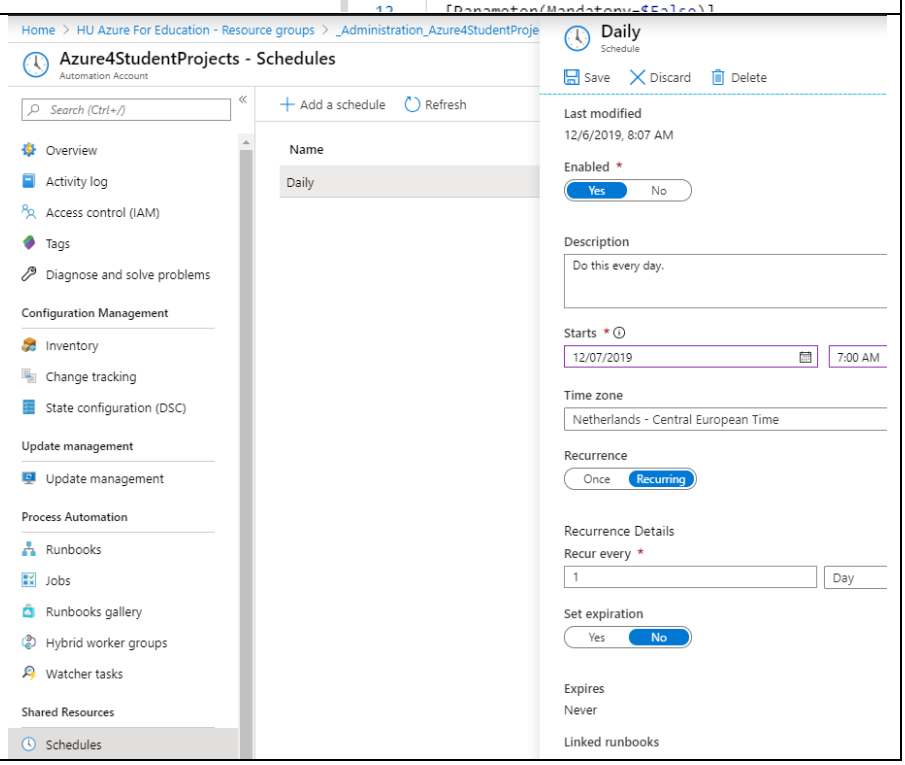
In order to send e-mails originating from your runbooks, for instance to send usage reports, you must create a SendGrid account within your subscription. SendGrid is a cloud-based email service that provides reliable transactional email delivery, scalability, and real-time analytics along with flexible APIs that make custom integration easy. Azure customers can unlock 25,000 free emails each month.
https://docs.microsoft.com/nl-nl/azure/sendgrid-dotnet-how-to-send-email#create-a-sendgrid-account

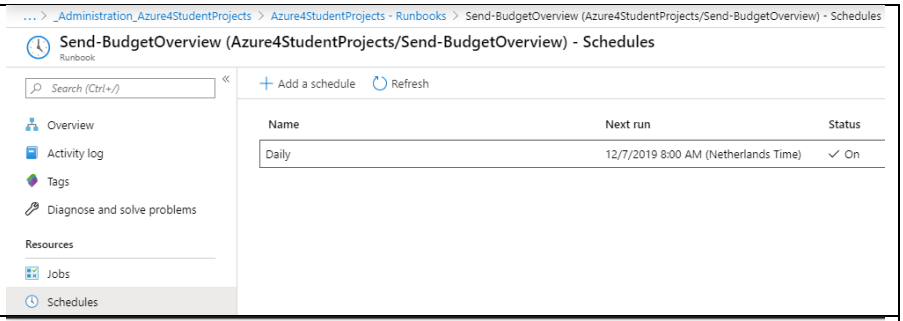
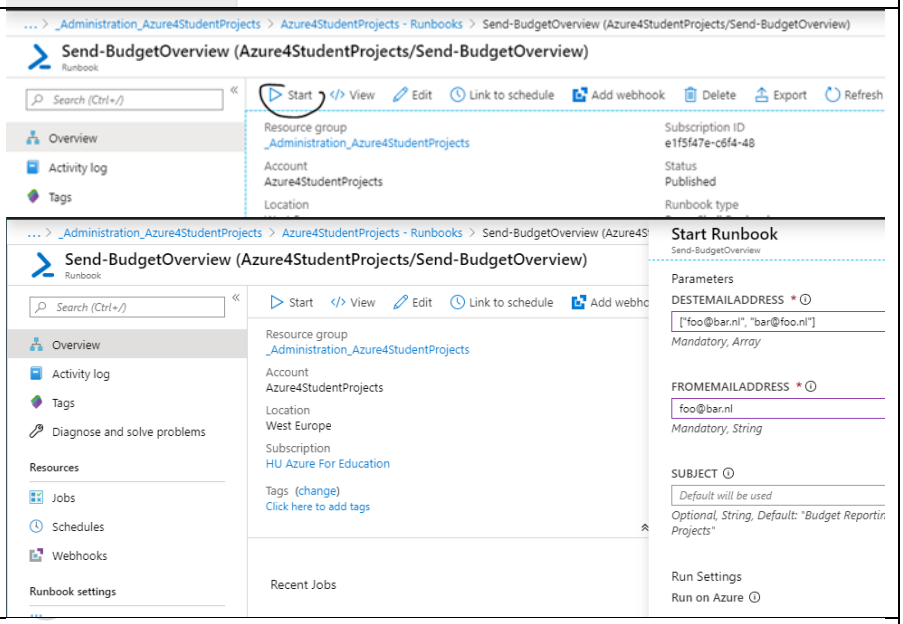
#	Description	Image
1	Logon to the azure portal as subscription owner and Create (Add) SendGrid.	
2	Complete the signup form. Select the F1 Free version. Name: Mail4StudentProjects Password: to be found in Settings\Settings.json ResourceGroup: _Administration_Azure4StudentProjects Contact Information: Owner of the subscription. Click “Create”.	
3	After deployment of the SendGrid account succeeded, go to the Resource Group and click on the SendGrid Account Resource. Click on “Manage”. You’ll be redirected to the app.sendgrid.com for e-mail confirmation.	
4	Click on the button in the mail received. You’ll redirected to: https://app.sendgrid.com/	

5	To send an email using SendGrid, you must supply your API Key. Go to your dashboard → Settings → API Keys → Create API Key. Note: If you perform this step at a later moment, goto the website via “Manage”, see step number “7”.	
6	Provide the name:“Azure4StudentProjects” Select “Full Access” and click “Create&View”.	
7	Next step is to Create an Azure KeyVault and store your SendGridAPIKey in that vault. The API key is needed every time you will send an e-mail.	This is done by executing the script: `.\Setup\Scripts\Create-AzureAutomationVault.ps1` References: https://docs.microsoft.com/nl-nl/azure/key-vault/quick-create-portal https://docs.microsoft.com/nl-nl/azure/automation/automation-send-email

Capability to Send Budget Reports on a daily basis to one or more administrators using a runbook

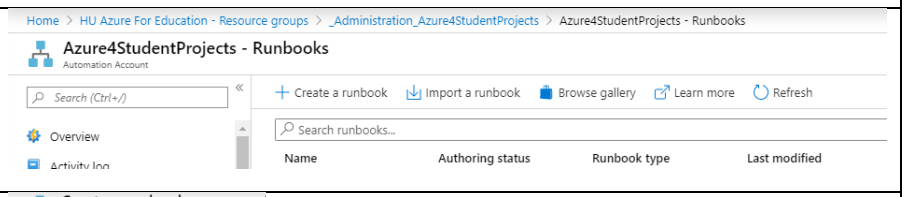
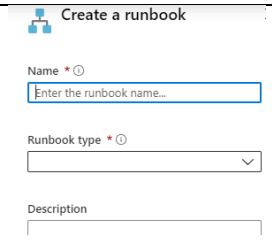
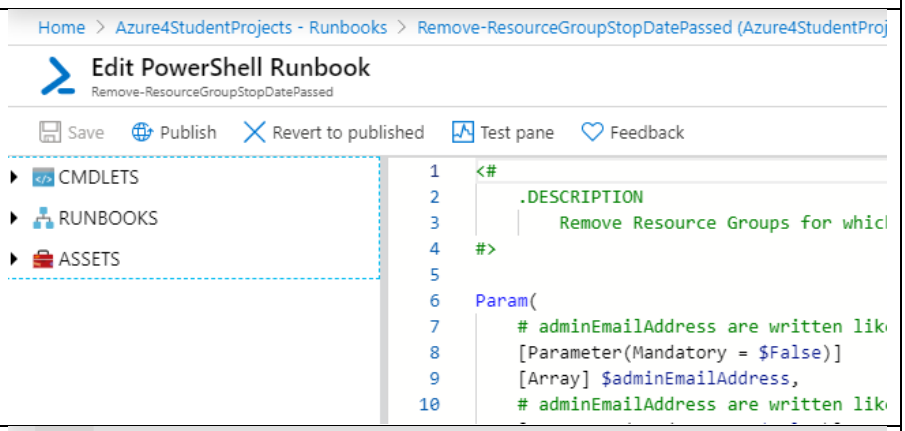
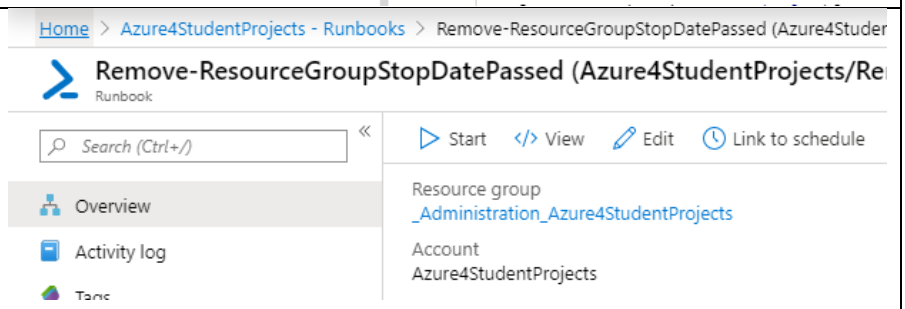
Next step is to create a runbook that sends a budget report every day by mail. It provides an overview of the subscription and budget reports of every resource group sorted by budget that is left and a color of red when budget that is left is < 10 euro.

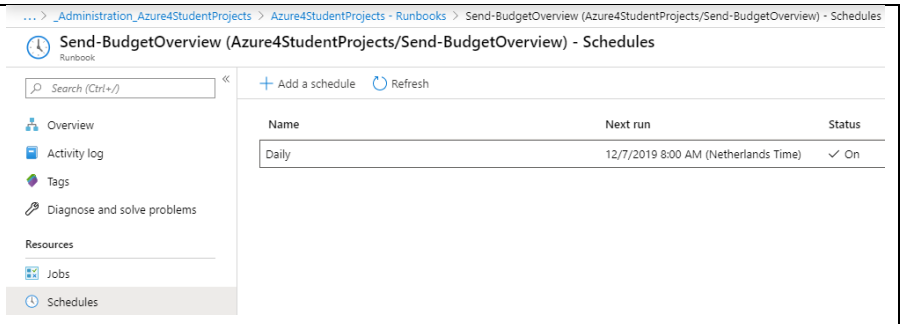
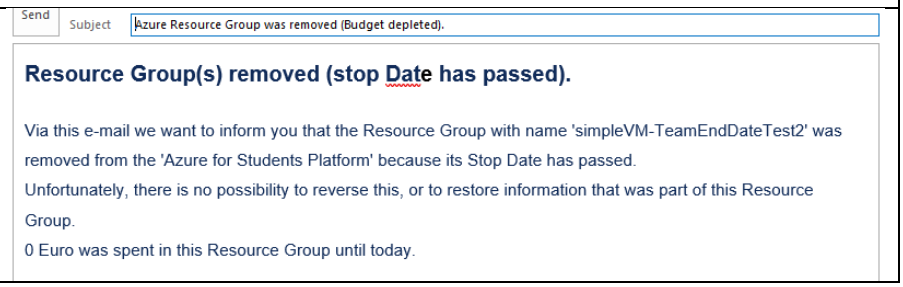
#	Description	Image
1	Create a new Runbook by clicking “Create a runbook” from RG=_Administration_Azure4StudentProjects→ Automation account=Azure4StudentProjects → “Runbooks”	
2	Click “Create a runbook” Name: “Send-BudgetOverview” Type: PowerShell Description: short description if you like. Click: “Create”.	
3	Contents of the runbook can be found in the repository with files: .\Setup\Runbooks\Send-BudgetOverview.ps1 Paste it in the editor and press “Save”. Press “Publish”.	
4	Add a schedule called “Daily” with these properties: Starts: choose the day of tomorrow.	

5	Link (Add) the schedule to your Runbook.	
6	Don't forget to test the runbook.	
7	Result should be something like this in your mailbox:	If there are problems with how this message is displayed, click here to view it in a web browser. Click here to download pictures. To help protect your privacy, Outlook prevented automatic download of some pictures in this message. **Report: Azure for Student Projects Spending** Total Cost this year: 1.85 Euro (from a budget of 5000 Euro in total). Note, this table is sorted: Resource Groups with the least remaining budget are shown at the top of the list. | Resource Group | Budget | Spent | Budget Left | | --- | --- | --- | --- | | simpleVM-Team1 | 10 | | 10 | | simpleVM-Team2 | 10 | | 10 |

Capability to Remove the Resource Group when the Stop Date has passed using a runbook

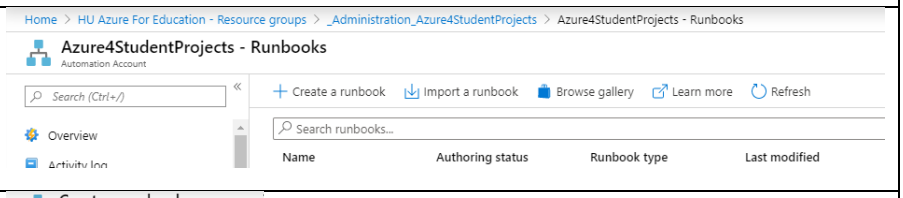
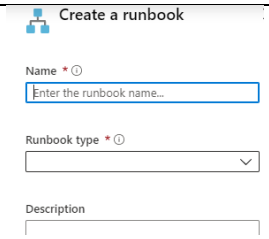
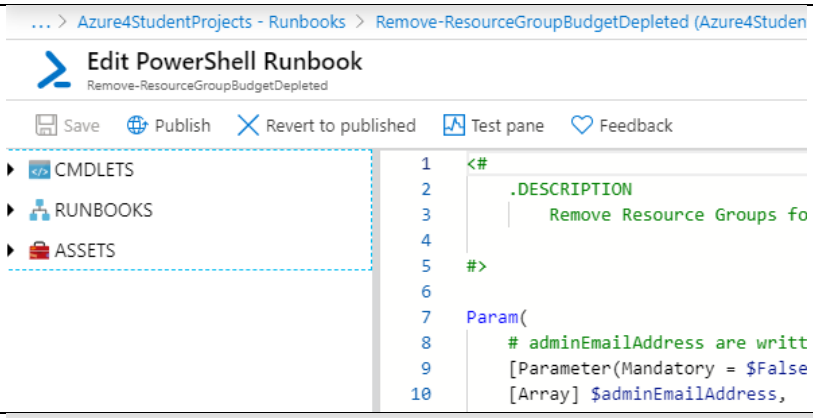
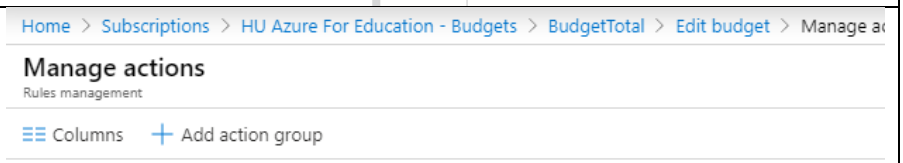
Next step is to create a runbook that removes a resource group when the stop date, that was provided during onboarding, has passed. A notification will be send to its users and adminstrator.

#	Description	Image
1	Create a new Runbook by clicking “Create a runbook” from RG=_Administration_Azure4StudentProjects→ Automation account=Azure4StudentProjects → “Runbooks”	
2	Click “Create a runbook” Name: “Remove-ResourceGroupStopDatePassed” Type: PowerShell Description: short description if you like. Click: “Create”.	
3	Contents of the runbook can be found in the repository with files: .\Setup\Runbooks\Remove-ResourceGroupStopDatePassed.ps1 Paste it in the editor and press “Save”. Press “Publish”.	
4	Link this runbook to the previous created daily schedule, by clicking “Link to Schedule”. Starts: choose the day of tomorrow.	

5	Link (Add) the schedule to your Runbook: “Add a schedule”. Link a schedule to your runbook → Select the “Daily” schedule. Provide the adminemailaddress for recipient and sender and leave the other two fields empty. Press twice “Ok”→ “Ok”.	
6	Result should be something like this in your mailbox when the stop date has passed:	

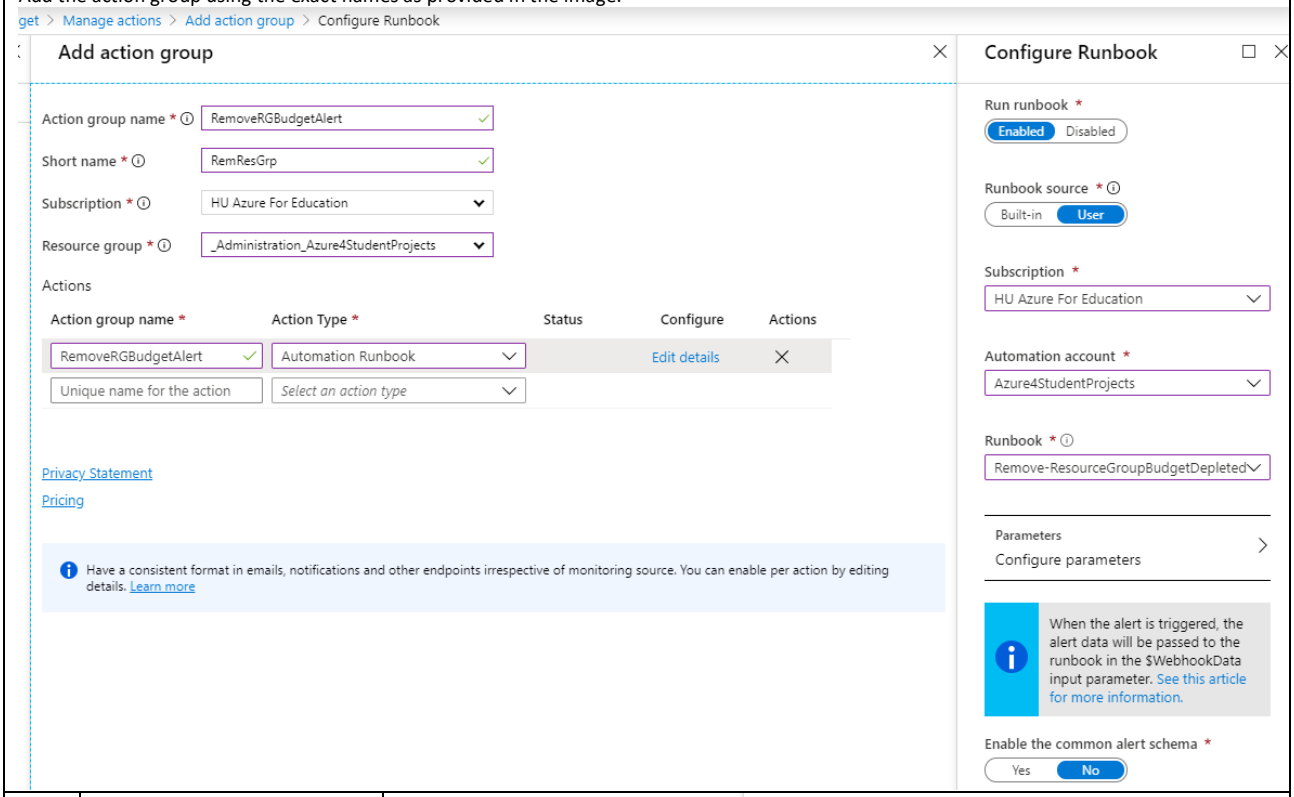
Capability to Remove the Resource Group when budget is depleted using a runbook

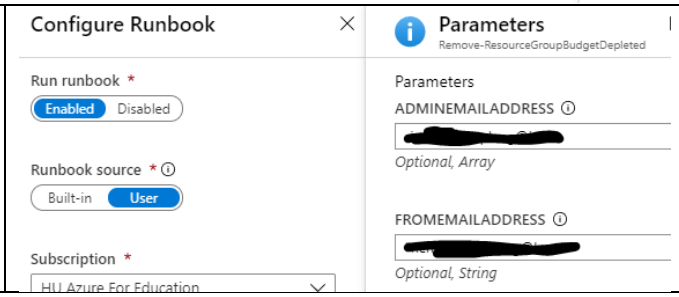
Next step is to create a runbook that removes the Resource Group and send an e-mail when budget is depleted.

#	Description	Image
1	Create a new Runbook by clicking “Create a runbook” from RG=_Administration_Azure4StudentProjects→ Automation account=Azure4StudentProjects → “Runbooks”	
2	Click “Create a runbook” Name: “Remove-ResourceGroupBudgetDepleted” Type: PowerShell Description: short description if you like. Click: “Create”.	
3	Contents of the runbook can be found in the repository with files: .\Setup\Runbooks\Remove-ResourceGroupBudgetDepleted.ps1 Paste it in the editor and press “Save”. Press “Publish”.	
3	Go to your subscription → Budgets → BudgetTotal → Edit budget → “Set alerts”. Click “Manage action group” Click “add action group”	

4

Add the action group using the exact names as provided in the image:

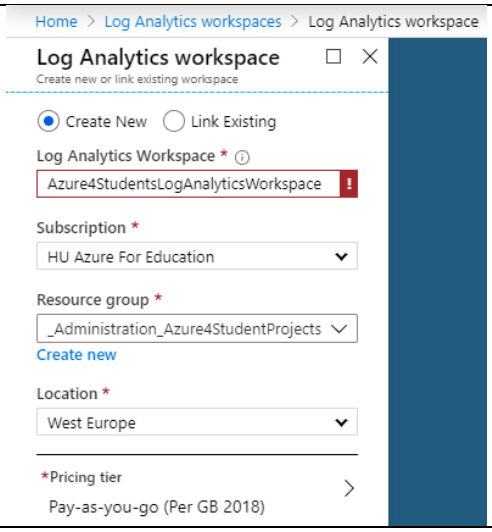


5	Click “Configure parameters” in the same screen and choose the right e-mail addresses for the sender and the recipients of the removal (end users will be informed as well). Click “Ok”, “Ok”.	

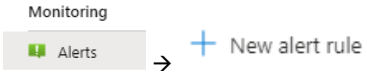
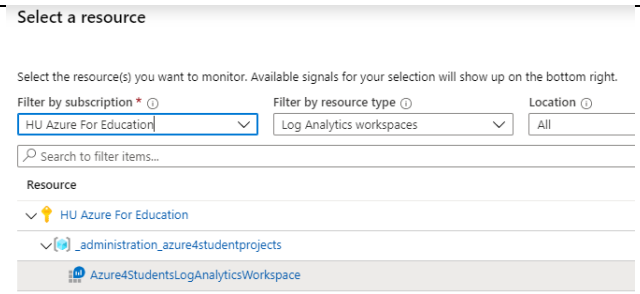
Capability to get an alert when a runbook fails

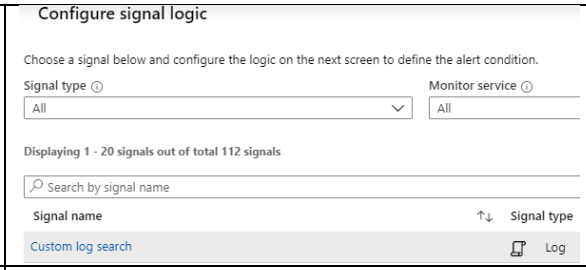
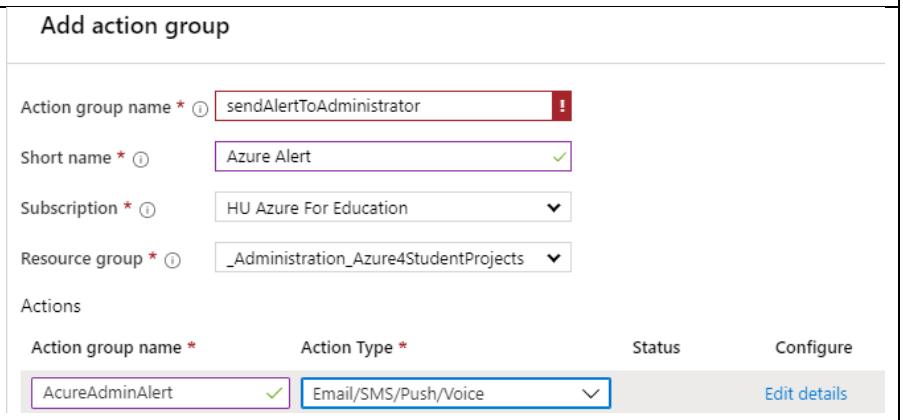
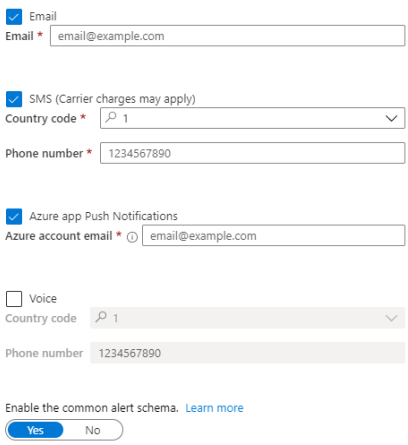
Runbooks are essential in this process. Therefore, it makes sense to forward Azure Automation job data to Azure Monitor logs. This provides insight on the automation jobs, gives the possibility to trigger an email or alert based on the runbook job status etcetera.

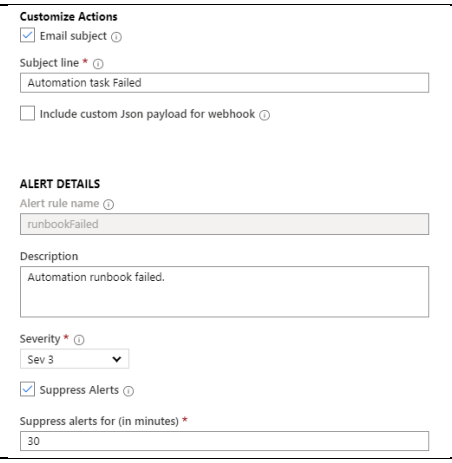
Reference information:
https://github.com/MicrosoftDocs/azure-docs/blob/master/articles/automation/automation-manage-send-joblogs-log-analytics.md

#	Description	Image
1	Create a Log Analytics workspace. Settings: Workspace name: Azure4StudentsLogAnalyticsWorkspace Resourcegroup: _administration_azure4studentprojects Pricingtier: Pay-as-you-go (Per GB 2018) Location: West Europe	
2	Find the resrouceid for your automation account	`Get-AzResource -ResourceType "Microsoft.Automation/automationAccounts"`
3	Find the resourceid for your log analytics workspace	`Get-AzResource -ResourceType "Microsoft.OperationalInsights/workspaces"`

4	After running the commands to the right, it may take an hour before you start to see records in Azure Monitor logs of new JobLogs or JobStreams being written.
	
	To see the logs, run the following query in log analytics log search:
	`AzureDiagnostics | where ResourceProvider == "MICROSOFT.AUTOMATION"`
		`$workspaceId = "[resource id of the log analytics workspace]"`
		`$automationAccountId = "[resource id of your automation account]"`
		
		`Eg:`
		`$workspaceId = "/subscriptions/<MYSUBSCRIPTIONID>/resourceGroups/_Administration_Azure4StudentProjects/providers/Microsoft.OperationalInsights/workspaces/Azure4StudentsLogAnalyticsWorkspace"`
		`$automationAccountId = "/subscriptions/<MYSUBSCRIPTIONID>/resourceGroups/_Administration_Azure4StudentProjects/providers/Microsoft.Automation/automationAccounts/Azure4StudentProjects"`
		
		`Set-AzDiagnosticSetting -ResourceId $automationAccountId -WorkspaceId $workspaceId -Enabled 1`

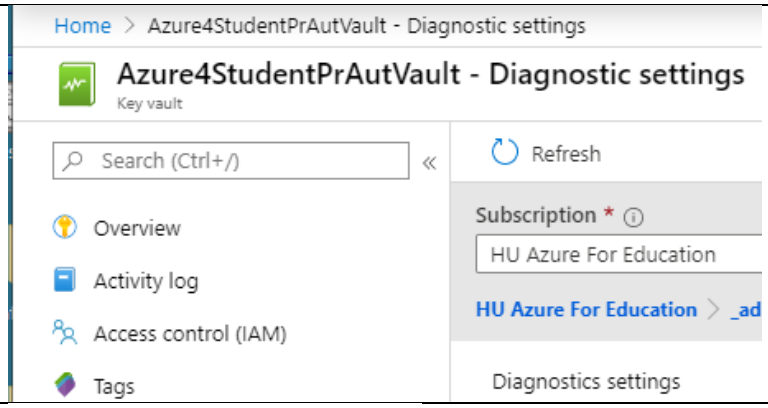
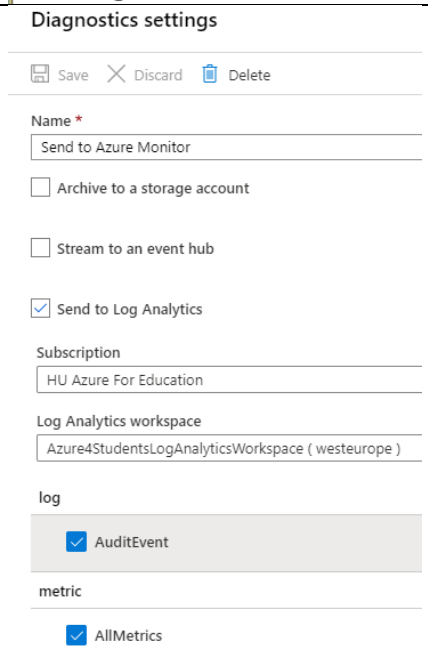
5	To Send an email when a runbook job fails or suspends: Click “Alerts” in loganalyticsworkspace. Click “New alert rule”	
6	**Select:** your LogAnalyticsWorkspace resource as the resource to monitor. Click “Done”	

7	Add a **condition**: “Custom log search” (click on this link).	
8	Add (log) **search** query: Alert logic: Threshold value=0 Resulting in logic: number of results > 0. Period (in minutes): 1440, Frequency (in minutes): 15. Click “Done”.	`AzureDiagnostics | where ResourceProvider == "MICROSOFT.AUTOMATION" and Category == "JobLogs" and (ResultType == "Failed" or ResultType == "Suspended") and (TimeGenerated > now()-35m)`
9	Create an Action (group): Click “Edit details”	
10	Configure actions at your convenience: Click “OK”.	

11	Create/modify alert details and click “Create Alert Rule”. Test the rule by failing a runbook!	

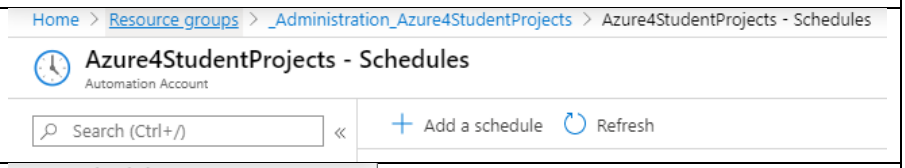
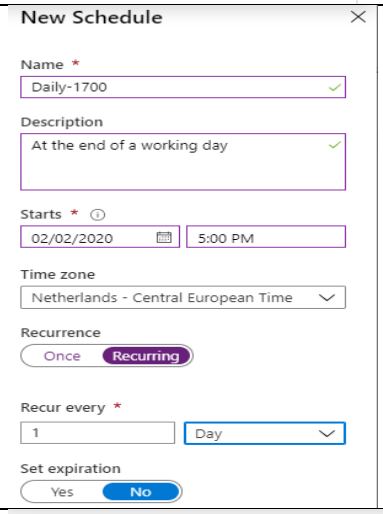
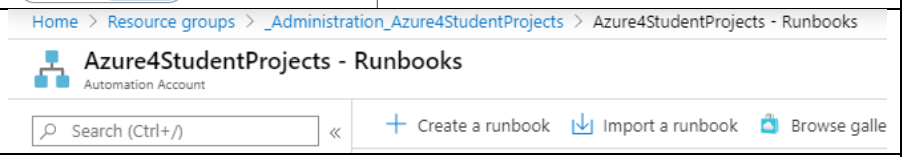
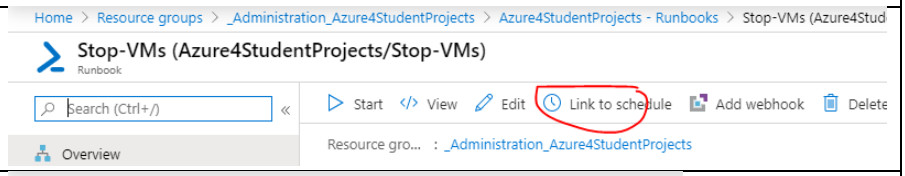
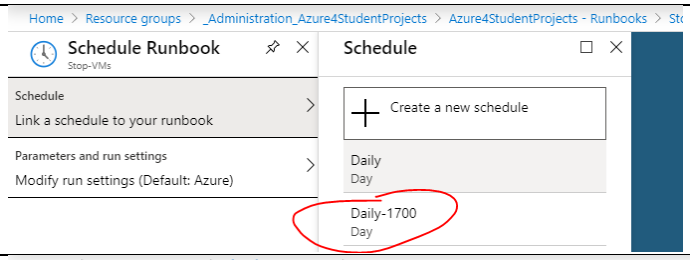
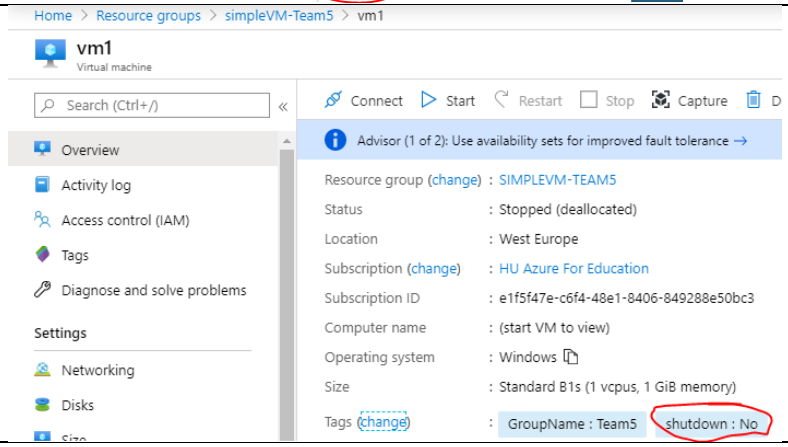
Enable Diagnostic Log for key Vault

To improve your security score in the security center, enable key vault logging.

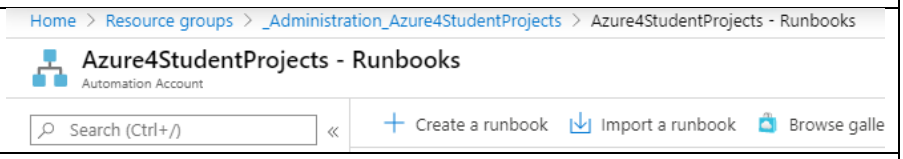
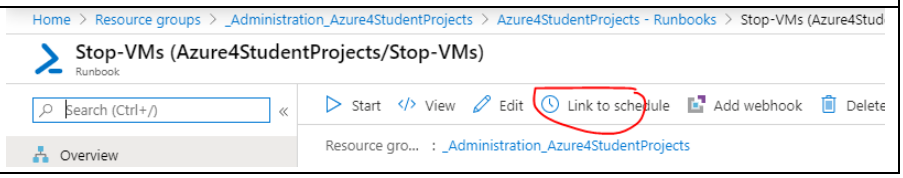
#	Description	Image
1	Goto your Key Vault and click “Diagnostic settings”	
2	Enable logging and click “Save”	

Capability to stop VMs (not tagged by shutdown=no) at the end of the working day (5:00 PM)

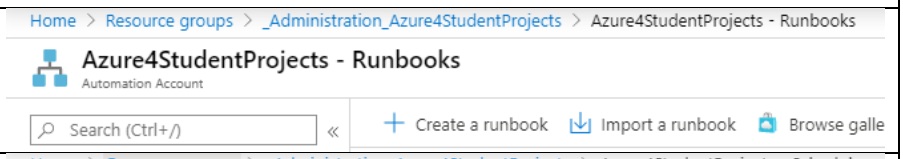
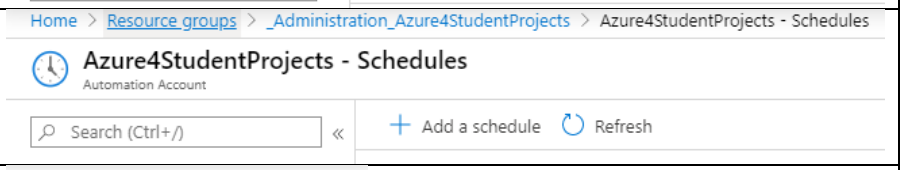
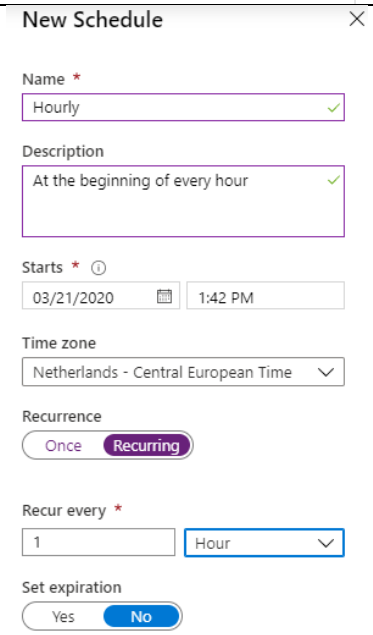
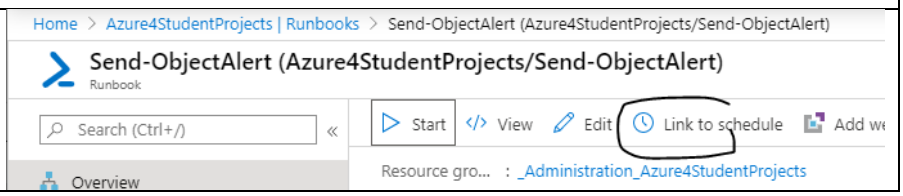
In order to decrease costs, all VMs are stopped at 17:00 hours (Netherlands CET) unless tagged by "shutdown"="no".

#	Description	Image
1	Create a new daily schedule in your automation account by clicking "Add a Schedule"	
2	Provide the details -Daily-1700 -At the end of a working day -5:00 PM -CET -Recurring -every Day -No expiration and click "Create".	
3	Create a new runbook in your automation account by clicking "Add a runbook" Name: "Stop-VMs"	
4	Add the contents of the file . Don't forget to "Test", "Save" and Publish".	`.\Setup\Runbooks\Stop-VMs.ps1`
5	Link the runbook to your newly create schedule	
	No additional run settings needed.	
6	Don't forget to test using a vm, for instance:	

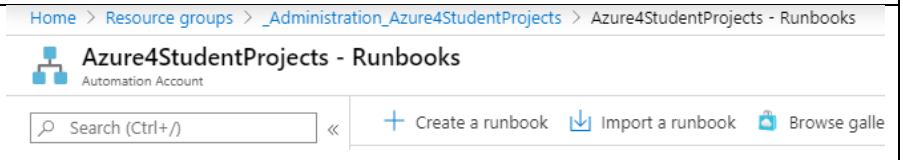
Capability get a cost report of all costs from YESTERDAY

#	Description	Image
1	Create a new runbook in your automation account by clicking “Add a runbook” Name: “Send-YesterdayCostOverview”	
2	Add the contents of the file . Don’t forget to “Test”, “Save” and Publish”.	`.\Setup\Runbooks\Send-YesterdayCostOverview.ps1`
3	Link the runbook to your Daily schedule	

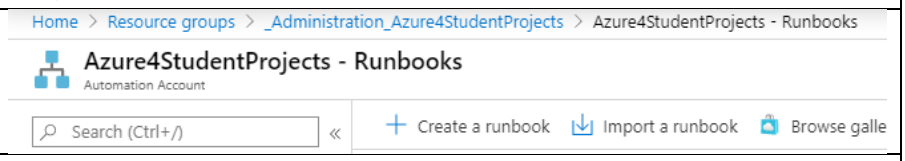
Capability to receive an Object Alert every hour if too many objects are created.

#	Description	Image
1	Create a new runbook in your automation account by clicking “Add a runbook” of PowerShell Runbook type. Name: “Send-ObjectAlert”	
2	Create a new hourly schedule in your automation account by clicking “Add a Schedule”	
3	Provide the details -Hourly -At the beginning of every hour -5:00 PM -Recurring -every 1 Hour -No expiration and click “Create”.	
4	Add the contents of the file . Don’t forget to “Test”, “Save” and Publish”.	`.\Setup\Runbooks\Send-ObjectAlert.ps1`
5	Link the runbook to your Hourly schedule and provide the right parameters.	

Capability to receive Activity Log information per Resource Group.

#	Description	Image
1	Create a new runbook in your automation account by clicking “Add a runbook” of PowerShell Runbook type. Name: “Send-AcitivyLogAnalysis”	
4	Add the contents of the file . Don’t forget to “Test”, “Save” and Publish”.	`.\Setup\Runbooks\Send-ActivityLogAnalysis.ps1`
5	Link the runbook to your Daily schedule and provide the right parameters.	

Capability to receive CosmosDB RU/s Alert and remediation on a database level. Cannot be remediated automatically on the collection level.

#	Description	Image
1	Create a new runbook in your automation account by clicking “Add a runbook” of PowerShell Runbook type. Name: “Set-CosmosRU”	
4	Add the contents of the file . Don’t forget to “Test”, “Save” and Publish”.	`.\Setup\Runbooks\Set-CosmosRU.ps1`
5	Link the runbook to your Hourly schedule and provide the right parameters.	

Capability to …. (for later usage)

Foo bar

#	Description	Image
1	Create	
2	Provide	
3		
4		
5		
6		
7		