

Appendix VII: Configure CORS Settings in SDE with Aras.DevOpsFramework.SaaS3 package

Cross-Origin Resource Sharing (CORS) allows web applications hosted on different domains to connect safely to instances deployed through SDE with the Aras.DevOpsFramework.SaaS3 package. This is useful when third-party web apps need to reach APIs or other resources.

To set up CORS, the user must follow these steps:

1. Open the correct default variable group for the environment (CI, SIT, UAT, STG, or PROD), or a custom group if needed:
 - o CustomConfigValues-CI
 - o CustomConfigValues-SIT
 - o CustomConfigValues-UAT
 - o CustomConfigValues-STG
 - o CustomConfigValues-PROD
2. Add the CORS settings to the variable named corsAllowOrigins in the variable group used for “Environment Specific Configuration”. (See the “Environment Specific Configuration” chapter for more details.)

Properties

Variable group name

CustomConfigValues-UAT

Description

Link secrets from an Azure key vault as variables ⓘ

Variables

Name ↑	Value	🔒
<u>corsAllowOrigins</u>	https://supplier.example.com, https://partner.example.com	🔒



The value must be a text string that lists domains separated by commas. Each entry should follow this format:

```
protocol://domain[:port]
```

For example:

“ <https://example.one.com> , <https://example.two.com> ”.

Important

* symbol is not allowed in values.

Important

If this setting is not needed, the user should remove the corsAllowOrigins variable from the corresponding variable group.

3. Run the CI and Deploy pipelines when needed.

If the user needs to update the corsAllowOrigins value, the user should change the setting and run the Deploy pipeline using the **Update** strategy.

Important

Do not configure CORS using transformations in case of SDE with Aras.DevOpsFramework.SaaS3 package.

