



When you develop projects in Microsoft Azure Cloud Services, you typically want to begin small, utilizing the least resources possible. Then, as your project grows, scale the available processing power and space to accommodate the performance and size requirements as necessary. This topic presents main **levels of development** on Microsoft Azure and provides links to related configuration tasks, which you need to perform when ascending to a higher level.

Level 0 - Local development

You can choose to begin developing your Kentico Azure project locally in an emulator before deploying it to the cloud environment. In this case, set up a database and Azure Storage service and configure the web role's settings. See [Developing Kentico Azure projects locally](#).

Level 1 - Development

For the duration of project development, **one instance** of CMSApp web role is usually enough. To configure a project to use one web role instance, perform the [basic configuration](#) tasks.

Level 2 - Production with SLA

When you deploy your project and switch to the production environment, you may want to qualify for [Microsoft SLA](#). In such case, your project must use at least **two instances** of the CMSApp web role. When you increase the number of used web role instances, you need to adjust your project to synchronize the data between the instances and to store session state information.

For data synchronization, you can use the default web farm synchronization tasks, in which case you do not need to configure anything.

For storing session state information, you can use **Microsoft Azure SQL Database** – see [Storing cache and session state data in Azure environment](#).

Level 3 - Performance

When the performance of the level 2 environment is not sufficient, you can configure the **Azure Redis Cache** to store session state information – see [Storing session state information in Azure Cache Service](#).

Level 4 - Scalability

When you need even more power, you can further scale your project using the following approaches:

- Utilize larger cloud services
- Use more web role instances
- Configure autoscaling