What is the REX SDK ?

### What is REX ?

The primary goal for the REX concept is to help Revit API developers to concentrate on essential development aspects when creating various Extensions or add-ins for Revit, by providing support for typical, commonly used functionalities.

The outcomes coming from using this approach are:

* + Acceleration of add-ins development
  + Better consistency across add-ins
  + Seamless integration within Revit

REX is a technology, a Framework supporting development of Extensions (add-ins) for Revit and making them consistent and aligned with the way Revit interacts with users.

The primary goals for the REX **Framework** can be defined as follows:

* + Provide a Higher level of API interaction with Revit
  + Provide components / tools for typical functionalities
  + Enable seamless add-ins’ behavior within Revit environment
  + Enable easy add-ins activation and registration

### What is the REX SDK?

REX SDK is development environment for Rapid Application Development purposes that helps to create, deploy and activate add-ins based on the REX technology.

The core part of the REX SDK is implemented as a form of Microsoft Visual Studio C# template. Using the template provided, you can quickly build an add-in that has a similar look & feel to Autodesk Revit Extensions.

The REX SDK is composed of:

* Project Template (C#)
  + UI definition
  + Interactions
  + Localization support
  + Deployment
  + Microsoft Installation project
* Documentation
  + Getting Started
  + User manual and Design guidelines
  + API documentation
  + Samples

### To whom it is addressed?

Advantages coming from using this technology can be visible by all Revit API developers making add-ins for Revit, but it’s mostly efficient for those who find below presented aspects applicable:

* Developing multiple add-ins
* Advanced UI creation for add-ins
* Create commercial add-ins for further distribution
* Developing multi-language add-ins
* Developing links with other products

### What are the most interesting features for API Developer?

* C# project templates, ready to build, distribution and activation in Revit
* Common UI controls as EditBox, ComboBox,IndexLabel,…
* Components for HTML Reports generation
* Units conversions and units based parameters display and editing consistency with Revit
* Automated Class Data serialization and storage within BIM models

### What are the benefits for Revit Users?

The primary advantages for end-users are:

* Consistent look&feel across various add-ins
* Consistent behavior aligned with Revit product (e.g. units sensitive values edit and display)
* Could be used on previous developed add-ins for Revit to take advantages of dialogs, controls, units, and other utilities.

### What is the difference between Revit API and REX API?

Functionalities of REX SDK themselves do not provide an additional access to internal Revit. REX API extends Revit API functionalities by a set of new components.

### What products this technology can be applied to?

The REX based approach is applicable for Revit Architecture, Revit Structure and Revit MEP products.

### How do I get started?

Following steps will help to run first a REX based add-in for Revit:

- Make sure that Microsoft Visual Studio 2010 is installed

- Make sure that Revit 2012 is installed

- Open the Getting Started Manual and follow the steps described steps

- Create a project template

- develop necessary code

- Build the application

- Open Revit and run the Add-in