

03 - 模块 3: Mendix Studio Pro

模块 3: Mendix Studio Pro

模块简介（模块图块）：在此模块中，您将了解 Mendix IDE，即 Mendix Studio Pro。

讲座 3.1: 简介

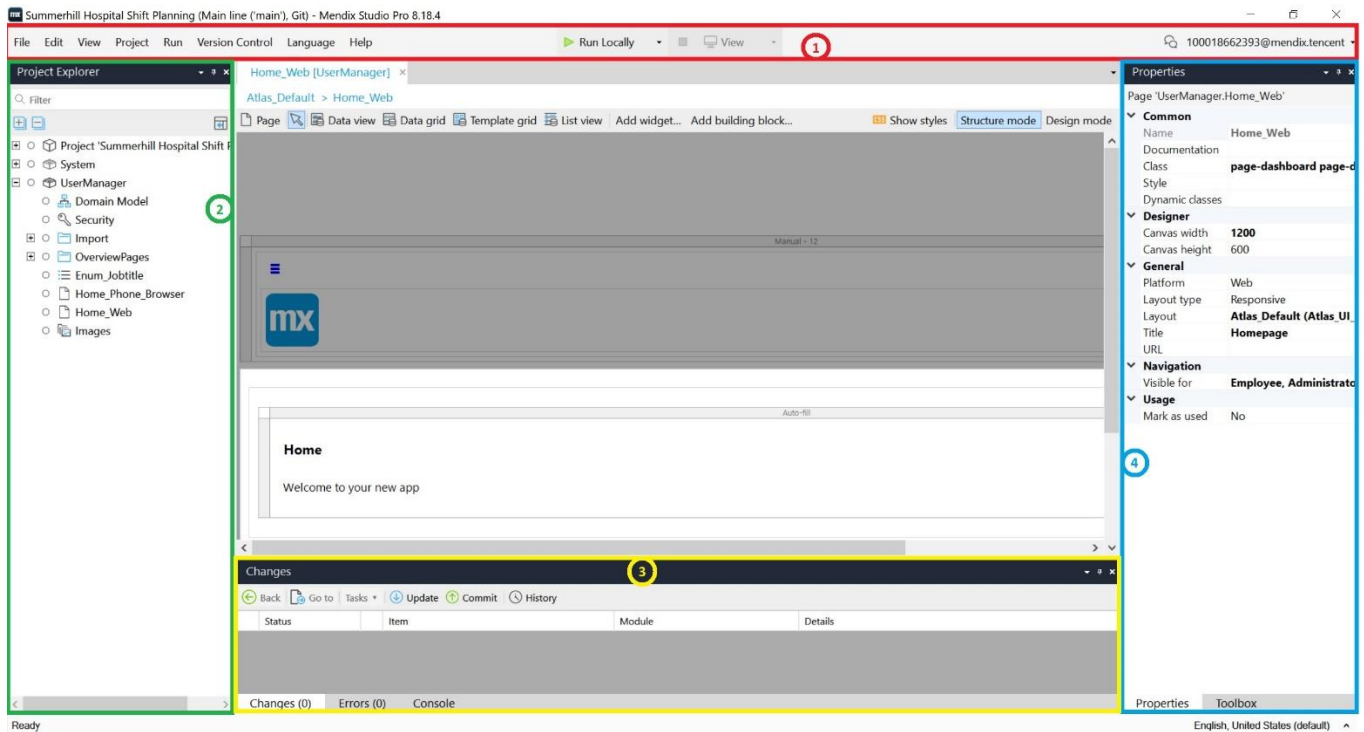
在此模块中，您将了解 Mendix IDE，即 Mendix Studio Pro。我们将介绍用户界面并说明各种组件及功能。涵盖主题如下：

- 运行 VS 本地运行
- 版本控制
- 在 Studio Pro 中创建第一个应用程序
- 导入用户案例
- 使用 Mendix 作为模拟服务的方式

在此模块中，您将设置开发环境，开始为 Summerhill 医院数字化创新团队构建应用程序。此外，您将使用的是 MockHrService 而非通用 HR 系统。如此即可不直接使用生产数据，也能测试应用程序并部署到生产，做到零故障。

讲座 3.2: Mendix Studio Pro 用户界面

使用 Studio Pro 开发 Mendix 应用程序，它提供受 Visual Studio 启发的 IDE，您会从中发现许多常用的 IDE 元素。



- 控制栏 (1) 设有标准的文件、编辑、查看和帮助选项。它还设有项目资源管理器，便于您更细致地管理项目的版本控制，查找 Mendix 的特定功能。运行菜单包含常见的并查集，另加以 Mendix 特有的调整。最后，我们提供语言菜单选项，允许您配置国际化和本地化 (i18n) 选项。
- Studio Pro 界面左侧设有项目资源管理器 (2)，可在此找到分布于模块中的所有项目元素。在列表顶部，您会发现默认的“项目”模块，可在此查找并设置您的项目属性，诸如导航、安全性和您从 Marketplace 上下载的任何模块。在其下方设有第二个默认模块“系统”，此处可配置每款 Mendix 应用程序所需的基本功能。以下所有模块均为用户自行创建。
- 在底部 (3)，可找到版本管理、用户案例管理和调试选项卡。
- 在右侧 (4)，可找到上下文相关的属性和工具箱选项卡等元素。

关于控制栏要特别注意的是，**运行与本地运行**的区别。这两个选项都将启动应用程序的构建，但构建位置截然不同。

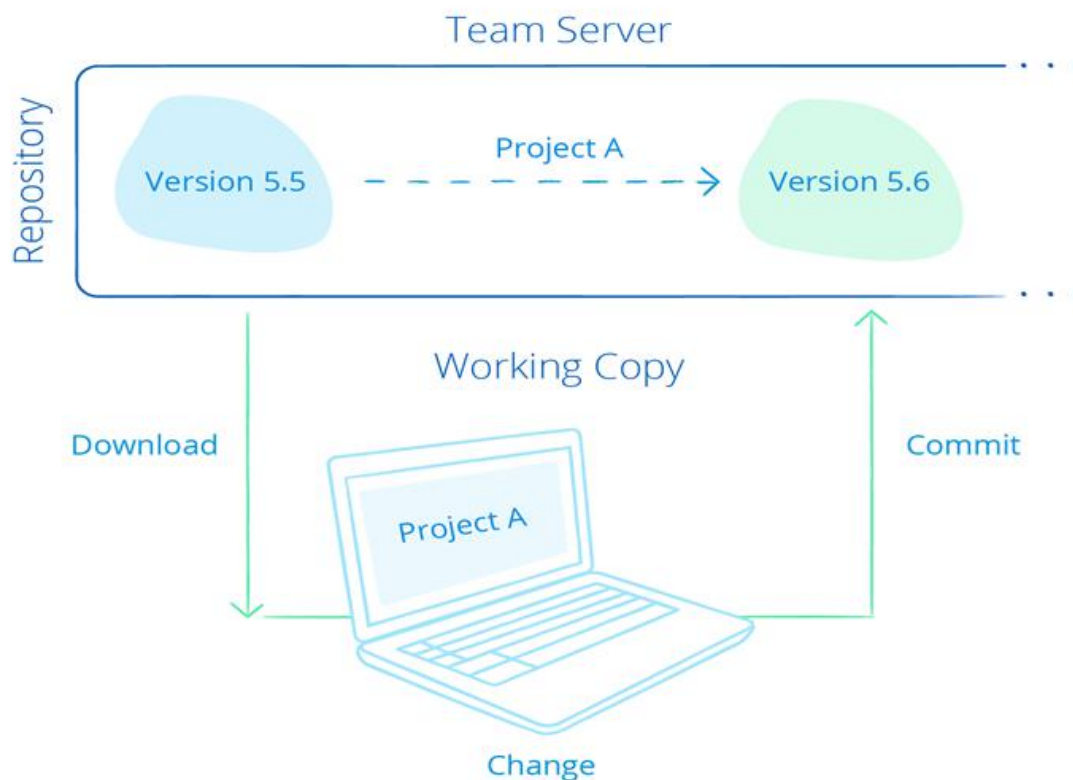
本地运行将在本地计算机上构建应用程序并启动 Mendix 本地实例，以便与应用程序的实时版本进行交互。

运行将应用程序的最新版本推送到云端并在那里运行。为此，我们使用 Mendix 自由云，允许您在限定时间内运行应用程序。一旦过期，应用程序将进入休眠状态。如果有人尝试访问它，应用程序将恢复，此过程需要几秒钟。需要注意的是，此服务为免费提供，因此

不提供与许可云节点相同的正常运行时间和稳定性保障。完美适用于测试应用程序或提供演示链接，但不应将此功能用于生产应用程序。

讲座 3.3：版本控制与合并

Mendix 内置版本控制功能。默认版本控制基于 Subversion 集中式版本控制系统。最近，您见识到分布式版本控制系统的出现，例如 Git 和 Mercurial。两个系统都有其优缺点，最为明显的是，集中式系统提供了更多的控制功能，而分散式系统则提供了优越的分支。无论哪种方式，Mendix Team Server 令确保管理和控制轻松执行。

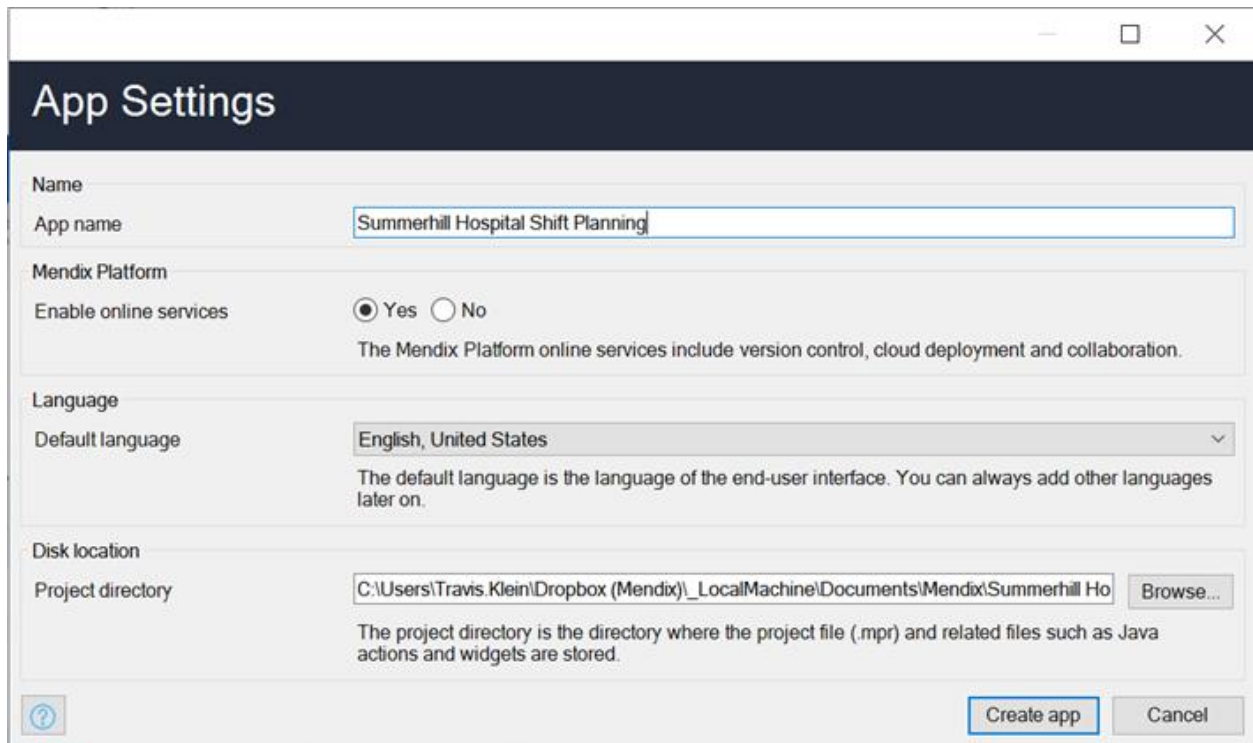


Mendix 允许您从团队服务器下载最新版本。这样它就成为了您的工作副本。您所作的任何更改都将与这个工作副本进行比较。如果团队成员向团队服务器提交某些内容，可通过更新工作副本获取更改。一旦开始提交，此流程将自动进行。与任何版本控制系统一样，如果您所作的更改与存储库中的更改发生冲突，您需要解决冲突后才能提交成功。该方法确实会产生一定的影响；Subversion 将始终与中央存储库进行比较，因此只有在提交时才会出现冲突。对于分布式版本控制系统，在存储库之间交换变更集时也可能出现冲突，因此，在这方面，Mendix 团队服务器可帮助您避免这些问题。

讲座 3.3.1：创建您的第一个应用程序

现在，您已准备好创建第一个 Mendix 应用程序。您会发现，其操作方法与大多数常用的 IDE 类似。

1. 打开 Studio Pro。
2. 单击绿色的**新应用程序**按钮。
3. 从**启动应用程序**选项卡选择**空白应用程序**模板。
4. 单击**使用此起点**创建应用程序。
5. 给您的应用程序起一个描述性的名字，例如，**Summerhill Hospital**。
6. 确保启用**在线服务**设置为**是**。
7. 单击**创建应用程序**。Mendix 将花一两分钟时间为您完成整个创建过程。



App Settings

Name

App name: Summerhill Hospital Shift Planning

Mendix Platform

Enable online services: ☒ Yes ☐ No

The Mendix Platform online services include version control, cloud deployment and collaboration.

Language

Default language: English, United States

The default language is the language of the end-user interface. You can always add other languages later on.

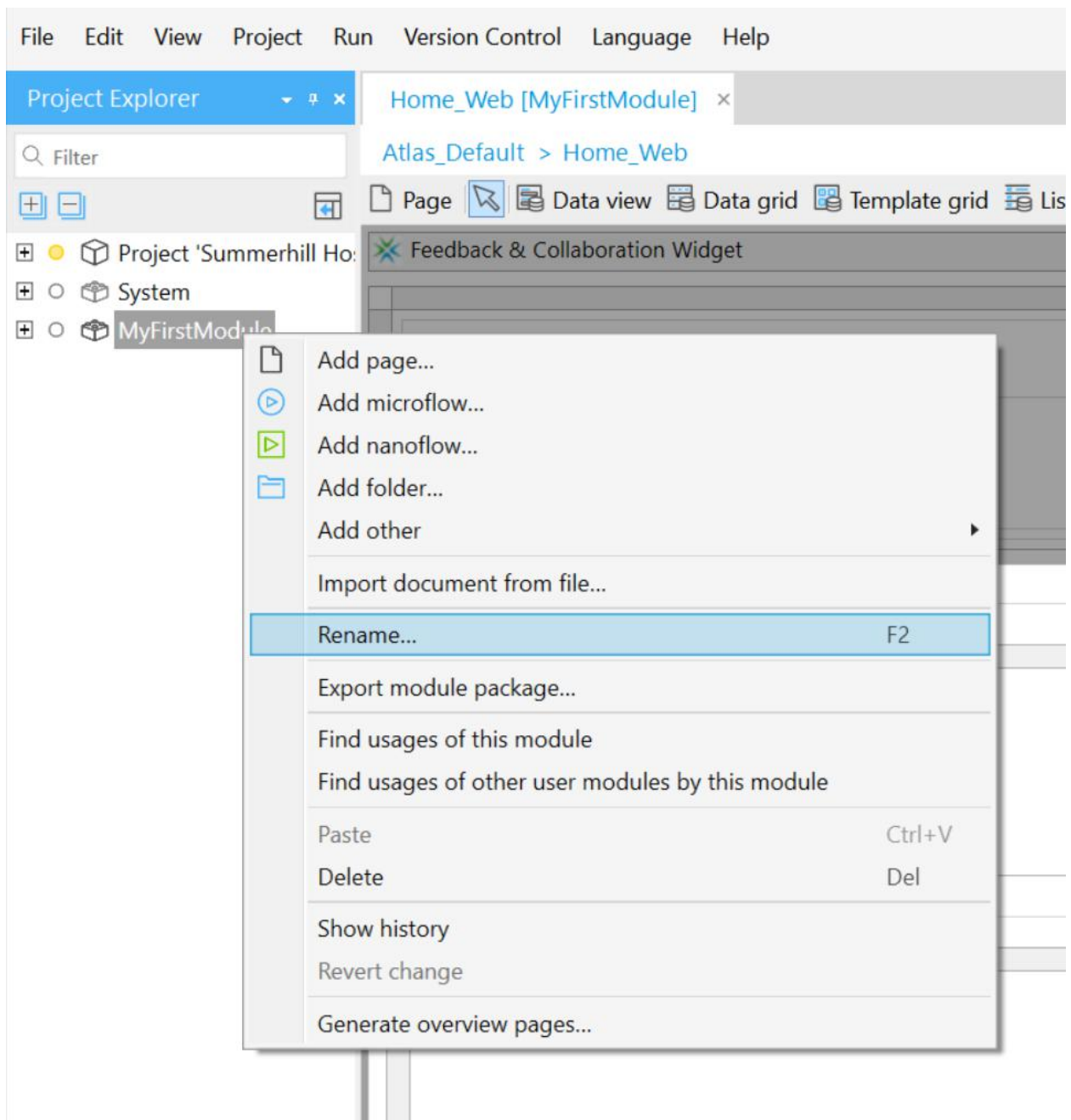
Disk location

Project directory: C:\Users\Travis Klein\Dropbox (Mendix)_LocalMachine\Documents\Mendix\Summerhill Ho

The project directory is the directory where the project file (.mpr) and related files such as Java actions and widgets are stored.

[?](#) [Create app](#) [Cancel](#)

8. 在 Studio Pro 中打开您的应用程序，右键单击项目资源管理器中的 **MyFirstModule**，然后选择**重命名**。将此模块的名称更改为 **UserManager**。



您即将处理的模块现已命名为 **UserManager**，这样每个人都知晓它的功能。在下一个讲座中，您将了解更多有关在 **Mendix** 中命名的知识，以便您项目中其他的开发人员轻松识别项目中的各个元素。

讲座 3.4：命名规则

为了确保项目组织有序且易于理解，**Mendix** 应用程序遵循一套命名规则。这样有助于您创建描述性名称，轻松组织项目中的元素。在本讲座中，您将了解本课程中使用的命名规则以及在哪里可以找到有关 **Mendix** 推荐规则的描述类文档。如此即可提高应用程序的可维护性，使之易于阅读和理解。

即使 Mendix Studio Pro 提供了强大的导航和搜索支持，我们也建议您遵循命名和建模规则，原因在于：

- 应用程序创建完成后，通常会移交给不同的人员进行维护，因此即使多年之后，他们也需要了解您所做的操作以及如何改进。
- 人人都应能够快速了解现有的应用程序，知晓功能设置位置，以及不同组件的关联方式。
- 清晰的结构有助于标识可重用代码。

在上一场讲座中，您对模块进行了重命名。在 Mendix 中，模块应视为可替换。它们应尽可能作为独立组件运行，以便替换为提供相同功能的其他模块。因此，模块名称应指明模块功能，例如，**CustomerManagement** 或 **SharePointIntegration**。如您所见，名称应采用帕斯卡命名法 (Pascal Case) 书写，其中每个单词的第一个字母应大写。

整个课程中，您将有机会认识各种不同的命名规则。每次遇到这些规则，您将了解到其他相关的信息。

在下一个学习任务中，您将向项目添加用户案例。

讲座 3.4.1：导入用户案例

还记得您之前创建的在线环境吗？现在我们来查看并填写信息。

1. 前往 <https://home.mendix.com>。
2. 打开您的项目。
3. 在左侧的**协同**下单击**案例**。
4. 单击**更多**按钮，然后选择**导入/导出**选项。
5. 从列表中选择**从 Excel 更新案例**，使用**浏览**按钮选择提供给您 Excel 文件，然后单击**导入**。
6. 滚动到页面底部，然后单击**下一步**。

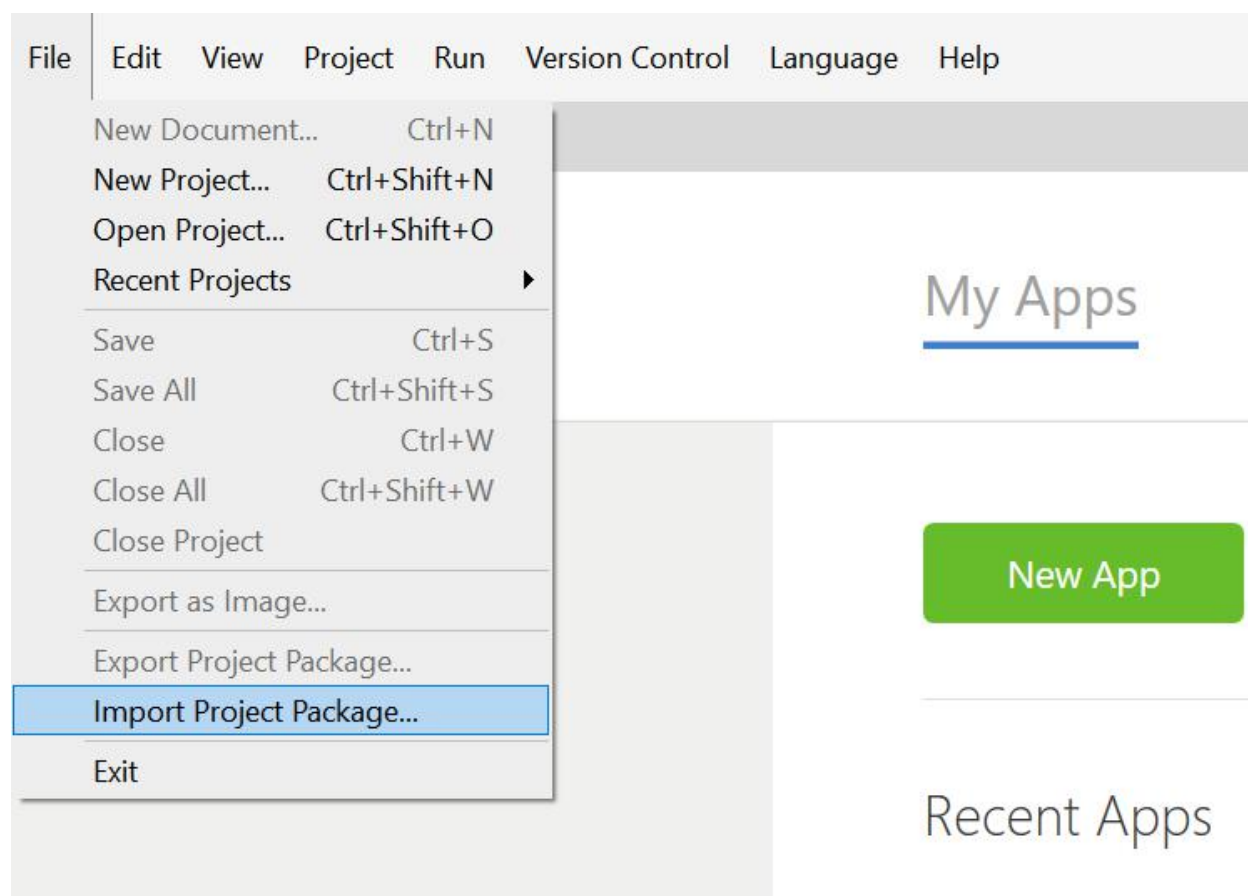
您的用户案例现已导入。

讲座 3.4.2：启动 MockHR 服务

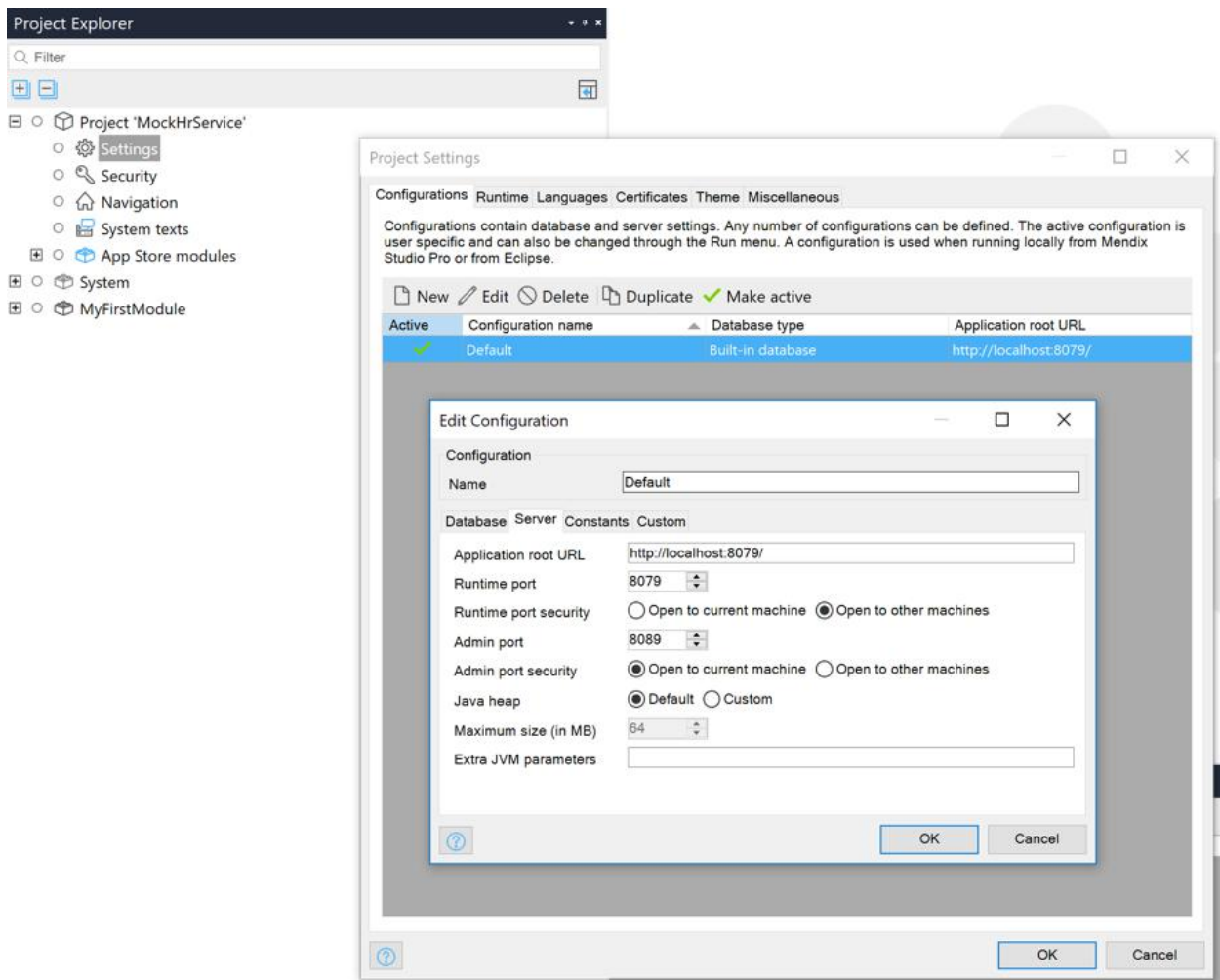
要确保构建合适的功能，在开发过程中访问外部服务非常重要。但在某些情况下，例如对于敏感个人数据，您会不希望在开发过程中使用真实信息。因此，最好使用真实信息编造的数据。这种情况下，可考虑利用系统提供的 Mendix 项目，名为 **MockHrService**，其中包含虚拟数据。数据结构与 Summerhill 医院提供的真实 **HrService** 保持一致。本练习将帮助您设置这项仿真服务，为您后续的开发过程提供支持。由于仿真服务完全模拟真实服务，

可以确定这款应用程序将运用于实际生产。您将在 Mendix Studio Pro 的第二个实例中运行 MockHrService。

1. 请从本讲座的资源中下载 **MockHrService**。
2. 打开 **Studio Pro** 第二个实例。
3. 在文件菜单下，单击**导入项目包**。
4. 选择您之前下载的 **MockHrService** 软件包。
5. 选择是启用**在线服务**以在 **New Mendix Team Server** 上存储新的应用程序，然后单击**确定**。



6. 确保此服务在端口 8079 和 8089 上运行。为此，导航至项目 "**MockHrService**"> 设置 > 配置 > 编辑 > 服务器。



7. 通过单击**本地运行**启动服务。

您已成功启动您的第一个 Mendix 应用程序。祝贺您！没有许可证，此应用程序将运行数小时。如果发现错误，请检查应用程序是否仍在运行。

总结

我们介绍了 Mendix Studio Pro 用户界面，并阐述了版本控制如何与 Mendix 团队服务器协同运行。新创建的应用程序和仿真 HR 服务在后台运行，您可以开始准备进入下一个讲座。