# Predix UI概述

Predix UI平台为创建运行在Predix数据和服务上的现代网页应用程序提供了有力支持。

## 现代网页应用程序

网页应用程序已经发展到能实现许多协同用户功能和任务，而这些功能和任务传统上只有桌面软件里才提供（例如，Google Docs和Microsoft Office）。另外，现代的网页浏览器内置了对[HTML5](https://en.wikipedia.org/wiki/HTML5)、[级联样式表（CSS）](https://en.wikipedia.org/wiki/Cascading_Style_Sheets)和[JavaScript](https://en.wikipedia.org/wiki/JavaScript)等开源标准的支持功能，这样能够减少对Adobe Flash或Microsoft Silverlight等专用网页浏览器插件的依赖

作为一名开发人员，您可以按照[模型视图控制器（MVC）](https://csis.pace.edu/~bergin/mvc/mvcgui.html)架构创建在多种网页浏览器和操作系统上运行的现代网页应用程序，这种架构能够将构建和维护的功能以独立模块形式分隔开：

* **模型**– 模型在[数据库](https://en.wikipedia.org/wiki/Database)或[数据服务](https://en.wikipedia.org/wiki/Data_as_a_service)中存储并检索信息。它定义了传递至控制器进行处理以及传递至视图为用户显示的信息的数据结构。
* **视图**– 视图是一个网页应用程序中可用信息的可视摘要。它是能够将任务和结果转换成用户可以理解并执行的信息的用户界面（使用HTML、CSS和JavaScript实现）。
* **控制器**– 控制器（使用JavaScript或[Java EE](https://en.wikipedia.org/wiki/Java_Platform%2C_Enterprise_Edition)、[Node.js](https://en.wikipedia.org/wiki/Node.js)或[Python](https://en.wikipedia.org/wiki/Python_%28programming_language%29)等其他网页技术实现）包含用于处理命令、作出逻辑决策和执行计算的应用程序控制逻辑。它还能够在模型和视图模块之间移动和处理数据。

**图1模型视图控制器架构**



更多信息请参见以下资源：

* [现代网页应用程序架构](https://developer.mozilla.org/en-US/Apps/Fundamentals/Modern_web_app_architecture)
* [MVC架构和框架](https://developer.chrome.com/apps/app_frameworks)

## Predix UI优点

## Predix UI的优势

Predix UI平台（<http://predixdev.github.io/predix-ui/>）包括：

* 一系列允许您按照一致的UX惯例、标准和规范添加功能或访问服务的可重用网页组件。
* 一系列建立一致的UI设计的可组合、可重用的CSS模块库。

获取更多信息请参见以下主题：

* [Predix UI仓库概述](https://www.predix.io/docs/?r=82016#cXtuQ3M)。
* [Predix UI CSS概述](https://www.predix.io/docs/?r=82015#EsrfONNT)。

作为一名开发人员，您可以使用Predix UI平台创建具有下列优势的网页应用程序：

**模块化架构**

Predix UI平台在github的PredixDev公共组里提供了一系列代码库，这些代码库包含了以下元素：

* **组件 - 允许您按照一致的UX惯例、标准和规范添加功能或访问服务的用户界面元素。**
* **卡片 – 驻留在用户界面内容区域的组件。您可以在个别卡片内的组件之间创建交互，也可以根据用户上下文与其他卡片之间创建交互。**
* **卡片组 – 为以特定顺序排列的卡片和组件提供简单的一维布局控件的组件。该控件可来自卡片组自身。**
* **视图 – 一个网页应用程序中所显示信息的可视摘要。它包括一个或多个卡片组，应用程序用户可以在运行时进行选择，以便快速获取信息并采取适当操作。**

Predix UI平台采用模块化设计，允许您只增加您需要的特性和功能。模块化架构利用可定义的变量和默认的变体，为更改和配置现有应用程序提供灵活性，而不使用大型、复杂的单一组件库。

**与框架无关**

存在多个JavaScript MVC框架（例如[AngularJS](https://angularjs.org/)、[Backbone.js](https://angularjs.org/)和[Ember.js](http://emberjs.com/)），以鼓励开发人员为HTML页面的[客户端侧绘制](https://technologyconversations.com/2014/07/10/server-vs-client-side-rendering-angularjs-vs-server-side-mvc/)编写更多结构化代码。每个框架有其独特的优点，同时它们均可以使用Predix UI库快速创建网页应用程序。更多信息，请参见[在网页应用程序中使用Predix UI库](https://www.predix.io/docs/?r=57800#UP8oJ8sa)。

## Predix UI的特性

作为一名开发人员，您可以使用以下Predix UI平台特性创建一个网页应用程序：

**基础功能**

Predix UI平台提供以下基础功能：

* 专注于使信息对工业用户来说有意义而且有用的数据优先可视化设计语言。
* 在笔记本、平板、手机或大屏幕设备上提供一致的查看和交互体验的响应式网页设计。

**简单**

 Predix UI平台使用网页组件的开源[Polymer库](https://www.predix.io/docs/?r=82008#DP6qOXF)支持以下特性：

* 定制元素 — 在HTML中定义新元素。
* HTML导入 — 载入HTML、CSS和JavaScript资源，用作一个聚合的HTML文件。

这简化了构建网页应用程序的工作，使应用程序能够在各种网页浏览器上提供桌面软件的许多特性。

**可扩展**

Predix UI平台可[扩展](http://www.ibm.com/developerworks/library/wa-polymer/)，以支持以下特性：

* 将多个网页组件构成一个单一实例。
* 通过一个网页组件添加图形。
* 通过CSS设计一个网页组件的风格。
* 处理来自一个网页组件的事件。
* 修改一个定制网页组件的属性。

## Predix UI特性

**连接器和适配器**

Predix UI平台包括默认的连接器和数据适配器，用于快速连接资产、时间序列和键值存储等Predix服务。

参见[Predix UI Seed概述](https://www.predix.io/docs/?r=82006#Je2y7sCv)来了解一个[初学者样例](https://www.predix.io/services/starter-pack.html?id=1225)，通过该样例您能学会部署一个网页应用程序到本地开发环境和Cloud Foundry。Predix UI Seed为您提供以下特性和服务：

* 应用程序导航
* 响应式布局
* 网页路由
* 数据绑定
* 视图切换
* 数据可视化
* 日期和时间输入
* 服务集成

## 使用Predix UI开发应用程序

作为一名开发人员，在使用Predix UI前您应该了解以下网页技术：

**开源标准**

作为一名开发人员，在使用Predix UI平台前您应该了解以下开源标准：

* HTML5 – 参见以下资源获取更多信息：
	+ [HTML5参考](https://dev.w3.org/html5/html-author/)
	+ [HTML5开发者指南](https://developer.mozilla.org/en-US/docs/Web/Guide/HTML/HTML5)
* CSS 级联样式表 - 参见以下资源获取更多信息：
	+ [级联样式表参考](https://www.w3.org/Style/CSS/Overview.en.html)
	+ [CSS3开发者指南](https://developer.mozilla.org/en-US/docs/Web/CSS/CSS3)
* Javascript- 参见以下资源获取更多信息：
	+ [Javascrip参考](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Language_Resources)
	+ [Javascript开发者指南](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Language_Resources)

**开发者工具**

作为一名开发人员，在使用Predix UI平台前您应该了解以下开源工具：

* GitHub 代码库
* Node.js 运行时环境
* Bower 包管理
* Gulp 构建系统
* CSS 框架
* SASS (Syntactically Awesome Stylesheets) 语言

**GitHub 代码库**

作为一名开发人员，您使用GitHub作为源代码库和版本控制系统。Predix UI平台使用模块化设计，允许您从GitHub上的PredixDev公共组里可用的单独库中只增加您需要的特性和功能。

关于Github代码库，参见以下资源获得更多信息：

* [GitHub 概述](https://guides.github.com/activities/hello-world/)
* [GitHub 文档](https://guides.github.com/)

**Node.js 运行时环境**

作为一名开发人员，您使用Node.js作为一个跨平台的运行时环境来创建网页应用程序。Node.js包括了npm工具来管理[node模块](https://docs.npmjs.com/getting-started/creating-node-modules)，诸如Bower包管理和Gulp构建系统。

关于Node.js运行时环境，参见以下资源获得更多信息：

* [Node.js 概述](https://nodejs.org/en/about/)
* [Node.js 文档](https://nodejs.org/en/docs/)

**Bower 包管理**

作为一名开发人员，您使用Bower包管理来安装正确的包版本，用来配置一个网页应用程序（诸如HTML, CSS或Javascript文件）和它们的依赖项。Bower抓取和安装所需的包，然后在bower.json清单文件里记录包的依赖项。

关于Bower包管理，参见以下资源获得更多信息：

* [Bower 概述](https://www.npmjs.com/package/bower)
* [Bower 文档](https://bower.io/)

**Gulp 构建系统**

作为一名开发人员，您使用Gulp构建系统来启动一个网页服务器，用于运行和管理构建一个网页应用程序的任务。这个网页服务器能通过在网页浏览器中输入<http://localhost:9000>进行访问。

关于Gulp构建工具，参见以下资源获得更多信息：

* [Gulp 概述](https://www.npmjs.com/package/gulp)
* [Gulp 文档](https://github.com/gulpjs/gulp/blob/master/docs/README.md)

**相关概念**

[Predix UI 概述](https://www.predix.io/docs/?r=81988#YB7xIwtq)

[Predix UI 库概述](https://www.predix.io/docs/?r=81987#cXtuQ3M)

[Predix UI CSS 概述](https://www.predix.io/docs/?r=81986#EsrfONNT)

**相关任务**

[在一个网页应用程序里使用一个 Predix UI 库](https://www.predix.io/docs/?r=81985#UP8oJ8sa)

[设置Predix UI 开发环境](https://www.predix.io/docs/?r=81984#UnGpptt2)

[用Predix UI定制网页应用程序](https://www.predix.io/docs/?r=81983#HIGIb2K9)

[在一个网页应用程序里使用一个Predix UI CSS 模块](https://www.predix.io/docs/?r=81982#IiukBBxE)

**设置Predix UI开发环境**

安装Node.js和Bower

**在您开始前**

设置Predix开发者环境。参见[设置Predix开发者环境](https://www.predix.io/docs/?r=81981#hOTKiBl-Uva9INX3)。

**步骤**

1. 为您的操作系统安装最新版的Node.js运行时环境。参见<https://nodejs.org/en/download/>。

2. 修改Node.js中npm包管理工具的文件权限。参见 <https://docs.npmjs.com/getting-started/fixing-npm-permissions>.。

3. 安装Bower浏览器包管理工具。参见<https://www.npmjs.com/package/bower>。

npm install bower -g

**相关任务**

[在一个网页应用程序里使用一个 Predix UI 库](https://www.predix.io/docs/?r=81985#UP8oJ8sa)

[安装Predix UI CSS初学者样例](https://www.predix.io/docs/?r=81976#Zyi5rVik)

[配置Predix UI CSS 初学者样例](https://www.predix.io/docs/?r=81975#5qaRhSv)

[在一个网页应用程序里使用一个Predix UI CSS 模块](https://www.predix.io/docs/?r=81982#IiukBBxE)

# Predix网页应用程序支持的浏览器

Predix UI平台已经过测试，支持这些网页浏览器和操作系统。

以下网页浏览器的最低版本已经过完整测试，以支持所有Predix UI库，这些库在GitHub里的 PredixDev公共组中提供。

| **浏览器** | **OS** |
| --- | --- |
| Chrome 48.0 | * Windows 8.1
* MacOS 10.10
 |
| Firefox 43.0 | * Windows 8.1
* MacOS 10.10
 |
| IE 10 | Windows 7 |
| IE 11 | Windows 8.1 |
| Safari 9.0 | MacOS 10.10 |

***注：如果您在通过支持的网页浏览器使用Predix UI存储库时遇到问题，请记录GitHub问题。关于记录GitHub问题的更多信息，请参见***[***https://guides.github.com/features/issues/***](https://guides.github.com/features/issues/)***。***

# Predix UI术语

作为一名开发人员，您应该了解这些Predix UI术语和定义。

**组件**

网页组件是允许用户执行功能或访问服务的用户界面元素。它定义了展示信息之间的交互，与上下文、数据绑定和服务之间的交互和行为方式。

组件遵循以下粒度级的工作流和数据聚合原理：

* 使用简单HTML句法（例如，<px-datepicker/>）来定义依赖关系、行为方式、风格和模板。
* “骨架”（最低风格）和“Predix”（完整风格）主题的绘制。
* 独立发行和版本。
* 通过扩展或合成从其他组件继承功能的能力。
* 作为能够运用其公共API的另一个组件的一部分进行嵌套。
* 暴露[AngularJS](https://docs.angularjs.org/guide/introduction)等应用程序框架使用的API的能力。包括范围和双向绑定。

**卡片**

卡片是可以在不同显示环境之间共享的组件。卡片用作驻留在网页浏览器内容区域的容器。它可以包含嵌入式数据，以便在连接受限或被切断时使用。开发人员可以根据需要显示和隐藏卡片。

卡片可以接收并传输事件，而且可以注意到其驻留位置的上下文并作出反应。组件可以在单独的卡片内交互，卡片可以与其他卡片交互。您还可以通过一系列卡片创建一个卡片组，以创建更复杂的用户交互。关于如何使用卡片和卡片组的信息，请参见[设置视图服务](https://www.predix.io/docs/?r=81971#Fe6g4Bs7)数据库。

**卡片组**

卡片组是一个使用特定布局确定卡片排列的组件。关于如何使用卡片和卡片组的信息，请参见[设置视图服务](https://www.predix.io/docs/?r=81971#Fe6g4Bs7)数据库。

**视图**

视图是一个网页应用程序中所显示信息的可视摘要。视图可以拥有用户可在运行时选择的不同卡片组（以特定顺序排列的卡片和其他组件）。

**基本组件**

基本组件是用户界面代码的构建块。基本组件很简单也很强大，可单独使用也可以组合用于创建更复杂的组件。

**模板**

模板组件是整个屏幕或屏幕区域的布局。模板组件专注于内容结构或布局，而不是实际内容本身。

**上下文**

上下文是用户或系统的其他部分使用对象或系统的环境和方式。它由其自己的当前状态和周围系统的状态进行表示。它可以考虑到意图和历史。通过符合这些要求，对象或实体可以拥有上下文，可以是上下文的一部分，也可以就是上下文。

当您组合以下因素时，上下文感知仪表板（上下文仪表板）的行为方式和外观可能会不同：

* 用户或人员。
* 可通过仪表板表示，或者仪表板可对其执行操作的基本对象。
* 时间（例如，工作日、AM/PM、年份）。
* 设备。
* 角色和权限。
* 用户组。
* 系统、设备和用户的状态。
* 个人偏好。
* 感官因素。
* 环境。
* 语言。
* 场所。
* 文字系统（例如语音、字母、象形文字或表意文字）