**虚拟组装项目架构**

Issue Date – 2017/09/14

|  |  |
| --- | --- |
| **DOCUMENT ID:** | |
| **PROJECT: 虚拟装配** | **RELEASE: 1.0.0** |
| **FEATURE:** | **SUBSYSTEM:** |
| **DISTRIBUTE TO:** | |

Revision History

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Version** | **Author** | **Reviewed By** | **A. Description Of Change**  **B. Summary of Review** | **Issued by / Date** |
| v1.0.0 | 张琦 |  | 创建文档 | 2017/09/14 |
|  |  |  |  |  |
|  |  |  |  |  |

目录

[1 概述 4](#_Toc493357048)

[1.1 概述 4](#_Toc493357049)

[1.2 编写目的 4](#_Toc493357050)

[1.3 关键术语 4](#_Toc493357051)

[2 项目架构 4](#_Toc493357052)

[2.1 整体架构图 4](#_Toc493357053)

[2.2 架构说明 4](#_Toc493357054)

[2.2.1项目准备模块 5](#_Toc493357055)

[2.2.2数据定义模块 5](#_Toc493357056)

[2.2.3辅助逻辑模块 5](#_Toc493357057)

[2.2.4底层数据模块 5](#_Toc493357058)

[2.2.5数据处理模块 6](#_Toc493357059)

[2.2.6UI逻辑模块 6](#_Toc493357060)

[3 相关文档 6](#_Toc493357061)

[4 附件 7](#_Toc493357062)

1. **概述**
   1. **概述**

虚拟装配的项目架构。

* 1. **编写目的**

描述开发架构，为开发提供参考，作为团队开发的标准。

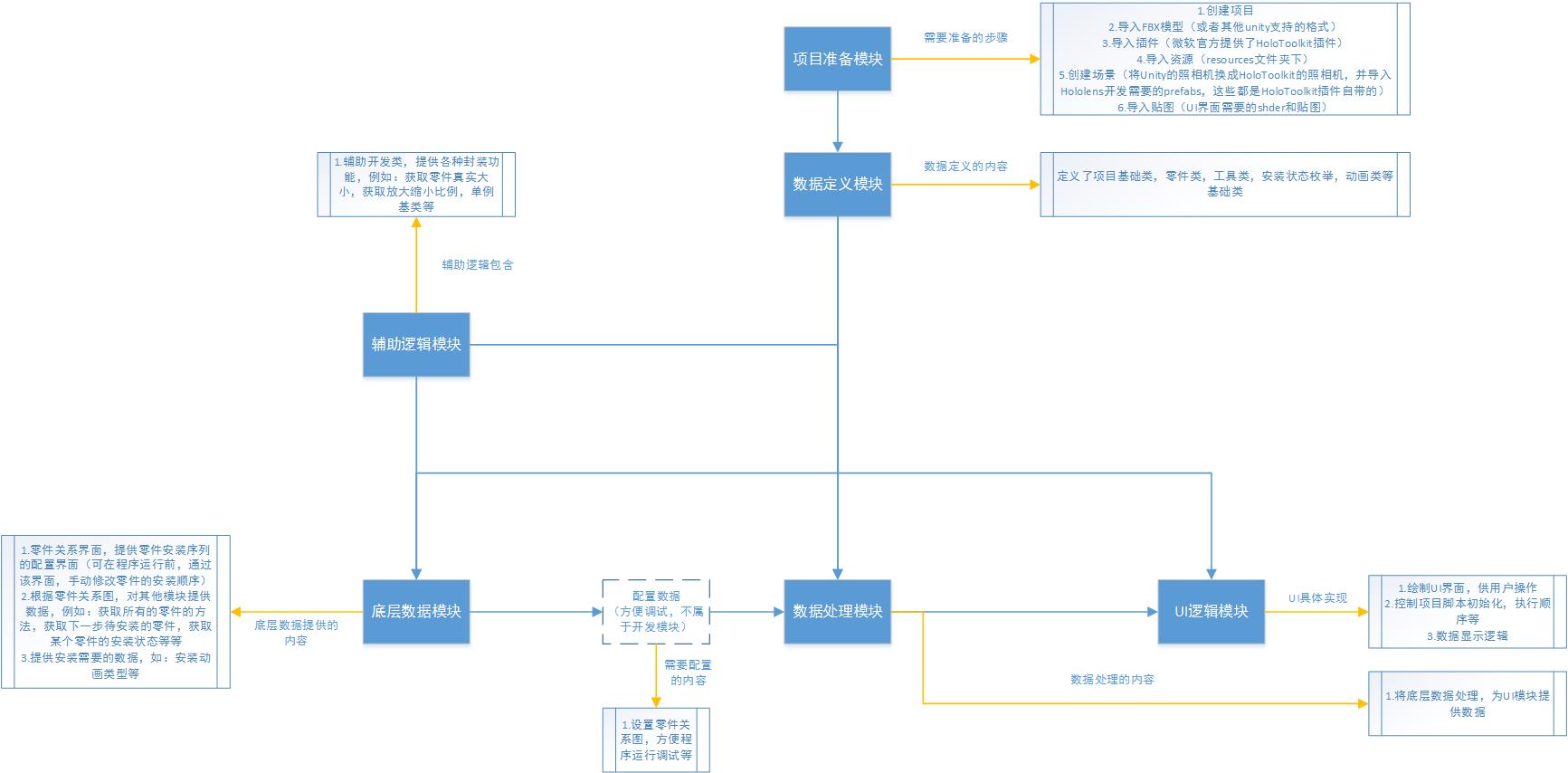
**预期读者**

项目开发团队成员

* 1. **关键术语**

|  |  |
| --- | --- |
| **缩写** | **描述** |
| Holocaust | MR混合现实智能眼镜 |

1. **项目架构**
   1. **整体架构图**



* 1. **架构说明**

### 2.2.1项目准备模块

1. 创建项目
2. 导入FBX模型（或者其他unity支持的格式）
3. 导入插件（微软官方提供了HoloToolkit插件）
4. 导入资源（resources文件夹下）
5. 创建场景（将Unity的照相机换成HoloToolkit的照相机，并导入Hololens开发需要的prefabs，这些都是HoloToolkit插件自带的）
6. 导入贴图（UI界面需要的shder和贴图）

### 2.2.2数据定义模块

**Part命名空间，定义基础类**

* + - 1. AnimationCollection.cs~动画集合类（零件动画），包含动画播放等方法
      2. AnimationPlayer.cs类~动画类（节点图画线部分动画），包含动画名称等属性
      3. ConnectorBase.cs~~创建文本框类（节点图中零件的文本框）
      4. IFlowNode.cs类~~节点图接口，包含节点图的一些规范，如：获取名称
      5. InstallationState.cs类~~安装状态枚举，列举零件的安装状态
      6. ModelType.cs类~~模型类别枚举，列举模型的类别
      7. Node.cs类(重点/基础)~~零件类，贯穿整个项目，最重要的类
      8. Part.cs类~~零件类，已经并入上述Node类中
      9. UtilityConnection.cs类~~节点图文本框，创建文本框类
      10. WorkSpace类~~定义工作区的一些属性
      11. 螺丝/螺纹孔/螺纹通孔/双层通孔/双层长孔/通孔/长孔.cs等类，定义相应的螺丝类

### 2.2.3辅助逻辑模块

**Utils命名空间，提供辅助工具类**

1.EventTriggerListener类，重写按钮点击事件，辅助按钮统一管理，减少代码逻辑

2.IdCounter.cs文件~~零件ID逻辑

3.Singleton.cs文件~~单例类

### 2.2.4底层数据模块

**Assemble命名空间，底层数据操作以及接口**

1.AssembleController.cs文件~~扫描整个节点图

2.AssembleFlowParser.cs文件~~节点图具体的联线

3.DependencyGraph.cs文件~~提供节点的很多方法（获取所有节点，获取下一个节点，获取上一个节点等）

### 2.2.5数据处理模块

**Cache命名空间，文件读写/Common命名空间，数据操作类/Control命名空间，数据封装类/Manager命名空间，逻辑类**

1.DataHelper.cs文件~~提供数据读写，后续存档功能的依靠

2.CommonPartsCommon.cs文件~~常用零件的数据操作，提供获取常用零件，获取常用零件类型等方法

3.NodesCommon.cs文件~~提供零件的数据操作，提供获取所有零件，获取所有零件类型等操作方法

4．ToolsCommon.cs文件~~提供工具类的数据操作方法，获取所有工具方法，获取所有工具类型等方法

5.CommonPartsController.cs文件~~常用零件的封装类，封装常用零件集合，只对外提供常用零件的获取，添加等操作方法

6.NodesController.cs文件~~零件的封装类，封装零件的集合，只对外提供获取零件集合，添加零件的操作方法

7.ToolsController.cs文件~~工具的封装类，封装工具的集合，只对外提供获取工具集合，添加工具等操作方法

8.WorkSpaceConTroller.cs文件~~工作区的封装类，封装工作区集合，只对外提供获取工作区集合，添加工作区等操作方法

9.AddCommonPartsManager.cs文件，添加常用零件的逻辑类，程序初始化，往CommonPartsController的集合中添加常用零件

10.CommonPartsController的集合中添加常用零件

11.AddPartsManager.cs文件，程序初始化，添加零件的逻辑类，往NodesController.cs零件集合中添加零件

12AddToolsManager.cs文件，程序初始化，添加工具逻辑类，往ToolsController.cs工具集合中添加工具

13.AssembleManager.cs文件，安装逻辑类，控制整个安装流程

14.NodeManager.cs文件，单个零件安装逻辑

15.WorkSpaceManager.cs文件，工作区逻辑

### 2.2.6UI逻辑模块

**Utils命名空间**

1.EventTriggerListener类，重写按钮点击事件，辅助按钮统一管理，减少代码逻辑

2.IdCounter.cs文件~~零件ID逻辑

3.Singleton.cs文件~~单例类

1. **相关文档**

无

1. **附件**

