Bee介于Native与Web两者之间, 致力于解决App开发的效率问题, 区别于Hybrid开发思想，我们为开发者提供了更好的与Native开发结合的方式，是一种保留Web前端开发优势的混合型开发技术，一边采用高集成度的Native框架完成逻辑及存储部分，一边用优化后的Web前端框架来完成界面部分。采用MIT协议.

Xml: <https://github.com/gavinkwoe/BeeFramework/wiki/Bee-Templates-Manual>

1. View模版化
   1. 具有语意的简化的HTML标签
   2. 针对移动端优化的CSS样式
   3. 将出现开源的模版样式标准库
2. Model对象化
   1. 本地存储对象化（ActiveRecord的特性引入）
   2. 存储关系对象化（通过类继承或嵌套直接表示）
3. Controller接口化
   1. 协议解析对象化（JSON与NSObject相互转换）
   2. 提供自动生成工具（基于JSON Schema的代码生成）

谁应该使用 {Bee} ？

1. 我想快速完成开发

2. 我讨厌编写冗长的代码

3. 我总是喜欢找点乐子

4. 我的项目需求大的吓人，不知从何入手

名词解释：

1. BeeUIBoard， 视图白板，控件的容器。
2. BeeUIStack， 视图堆栈，Board的容器。

BeeUIRouter， 视图路由，Stack的容器。

通过引用以下两个宏，{Bee}自动完成XML的加载和计算坐标：

1. SUPPORT\_AUTOMATIC\_LAYOUT( YES ) - 是否自动布局

2. SUPPORT\_RESOURCE\_LOADING( YES ) - 是否自动加载xml

想要解决的问题:

1. 不同屏幕的适配问题,约束
2. 事件的传递与处理(signal)
3. 快速生成界面xml代码工具未找到
4. 对语法的熟悉.