

---

# 项目开发文档

**BigBang 团队**

二〇一三年十一月

---

# 目 录

第 1 章	软件需求分析 .....	1
1.1	构思的创意和价值.....	1
1.1.1	项目背景：选题的动机与目的.....	1
1.1.2	研究现状：市场调查.....	2
1.2	创新性分析 .....	2
1.2.1	创新的功能点 .....	2
1.3	可行性分析 .....	3
1.3.1	主要技术路线 .....	3
1.3.2	核心技术关键与实现可行性.....	3
1.4	用例图 .....	3
1.7	运行环境与支持平台.....	4
1.7.1	运行环境 .....	4
1.7.2	支持平台 .....	4
第 2 章	软件综合设计 .....	5
2.1	程序模块划分 .....	5
2.1.1	UI 界面模块.....	5
2.1.2	模型加载模块 .....	5
2.1.3	模型着色模块 .....	5
2.1.4	模型编辑模块 .....	5
2.2	算法设计 .....	5
2.2.1	并行加载模型 .....	5
2.2.2	并行拓扑算法 .....	5
	该算法通过面包含的顶点——顶点周围的面——面包含的顶点——顶点周围的面，递归搜索并缓存拓扑数据，允许可对模型进行区域化而非顶点化操作，保证了模型的圆滑度和美观性，达到了某种雕刻效果。由于该算法所需计算资源较大，固采用了并行计算技术，一切拓扑计算均在后台进行，用户无所察觉，使用户更专注于创作本身。	
2.2.3	变形算法 .....	6
2.2.4	颜色混合算法 .....	6
2.2.5	一键特效算法 .....	6
第 3 章	工作安排 .....	7
3.1	团队工作安排 .....	7
3.2	项目时间进度表 .....	7
第 4 章	引用说明及代码统计.....	9
4.1	引用说明 .....	9
4.2	代码统计 .....	9

---

# 第 1 章 软件需求分析

## 1.1 构思的创意和价值

### 1.1.1 项目背景：选题的动机与目的

21 世纪，计算机成为了我们描绘世界、表达自我的最重要工具之一，要使计算机能精确直观地描绘周围的物体，我们必须能在二维屏幕中虚拟出一个三维空间。通过三维技术，机械工程师不需要再去抽象还原不直观的二维平面图，通过三维透视视角可以直观地设计自己的机械零件；医生不必再面对冷冰冰的 X 光片，通过三维扫描图像可以清晰地分析病人的病灶；军事指挥员可以抛弃脱离现实的二维指挥图，通过三维 GIS 技术，生成地形和影像数据的全景视图，进行直观的仿真指挥训练和实时战场分析。三维技术已逐步应用于生活中的各个角落。互联网作为人们交互的重要平台，其形态因技术问题长期被局限在 2D 模式中，随着 3D 技术的不断进步，互联网产品逐步走向 3D 化，越来越多的互联网应用开始以 3D 的形式呈现给用户，包括网络视讯、电子阅读、网络游戏、虚拟社区、电子商务、远程教育等等。世界本来就是立体的，我们需要还原一个真实的世界。

随着移动平台的发展，我们进入了移动互联时代，移动平台上的多媒体应用开发成为了当今热门的研究话题。曾经移动互联平台上的 3D 展现被长期束缚在 flash 的影响之下，而 html5 的出现带来了一场技术革新，将网页前端显示领域重新洗牌。在 3D 方面，WebGL 将 JavaScript 和 OpenGL ES2.0 结合在一起，不仅摆脱了插件束缚，同时利用 OpenGL 的跨平台特性大大提高了应用的利用率，硬件 3D 加速渲染技术更是能够使 3D 显示更加流畅自然。在 2013 年之前，手机用户上网浏览多媒体网页仍受手机硬件、浏览器以及网速等方面限制导致体验不佳，WebGL 标准在手机上的逐步普及，必将使手机网络服务得到快速地发展。

我们团队认为，一项新科技的出现其最终目标便是普及大众，造福大众，当达到这个目标后，不论是企业的利益或是科技本身的价值都将得打最大化。正如一提到 Intel 这个词汇，即便是不懂计算机的门外汉，也能细说一二，因为大部分

---

人都已把这个词汇和计算机的大脑 CPU 联系在了一起，这便是科技的魅力和最终目的:让普通大众都切身感受到科技的光芒。

3D 技术应用价值巨大，例如现在很热门的 3D 打印机，对工业和日常生活都带来了巨大的革新，但是这个潜力无限的技术在大多数人看来还是显得高深莫测。其原因便是 3D 建模软件不是每一个人都会用的，大多需要专业知识才可以掌握并且熟练操作，这对于从事工业设计，建筑，医疗等很多领域的人员都是一个现实的问题。大家都想“画”出立体模型，都想自行创建 3D 设计图，但是却无从下手。

因此我们团队的选题初衷便是为了把大家从高深的专业技术中解放出来。我们想要实现一款简单易行的 3D 建模软件，通过 HTML5 技术，将其部署在移动端上，提供基础模型库，用户只需在手机屏幕上拖拉旋转点击，轻松画出其想要的 3D 模型，随后便可以提供相应的接口将其进行 3D 打印。本创意的主要目的便是帮助人们记录下一瞬间的灵感，创造属于自己的 3D 模型或是 3D 设计稿，让他们切身体验到 3D 技术和 HTML5 技术融合后的实用和魅力!

### 1.1.2 研究现状：市场调查

目前市场上主流的 3D 建模与编辑软件或者应用主要分为两个方向。

第一类是以 3Dsmax，Maya 等为代表的专业桌面 3D 建模软件。这一类软件拥有强大的功能，多样的工具以及高效的性能与出色的表现。但由于这一类软件的目标用户群体大多定位为专业技术人员或者拥有较高的门槛，所以在易用性方面表现并不出色。与此同时桌面软件的特性以及对于硬件资源的依赖决定了这一类软件在移动性方面存在着巨大的不足。

第二类则是以 Sunlass，3DTin 等为代表的在线 3D 建模应用。这一类应用全部都是在 HTML5 出现之后随之兴起的 Web 应用。它们比较之第一类传统的桌面软件在移动性上有了很大的改进。但是由于这些应用的目标人群基本也定位在有一定 3D 建模基础的人群身上，所以在易用性方面还是有很大的改进空间。

## 1.2 创新性分析

### 1.2.1 创新的功能点

1.全色域取色板：在传统色域图去色斑基础上加入三原饱和色和黑白色，方便用户取特殊颜色。

- 2.全新手势设计，使得用户进行传统桌面三维界面操作成为可能。
- 3.独特的位移组合，兼顾建模自由度和手势操作便捷性。

### 1.3 可行性分析

经过我们团队研究，WebGL 技术通过浏览器调用显卡资源进行大量顶点着色运算，性能远超以往单 cpu 处理集合体顶点，使得在便携设备上渲染大量几何数据满足普通用户的需求。

#### 1.3.1 主要技术路线

- 应用架构: B/S 架构
- 前台技术: HTML5
- 3D 展现层: WebGL,Three.js 3D 引擎
- 后台技术: JSP 动态网页开发技术
- 数据存储: 服务器端 Oracle 数据库, 客户端 HTML5 本地存储技术
- 相关知识: 3D 计算机图形学、离散数学、拓扑学

#### 1.3.2 核心技术关键与实现可行性

基于现有模式以及 ThreeJS 框架特性设计适合本应用的鼠标以及手势操作模式。

基于 ThreeJS 框架构建 3D 模型加载以及编辑功能。

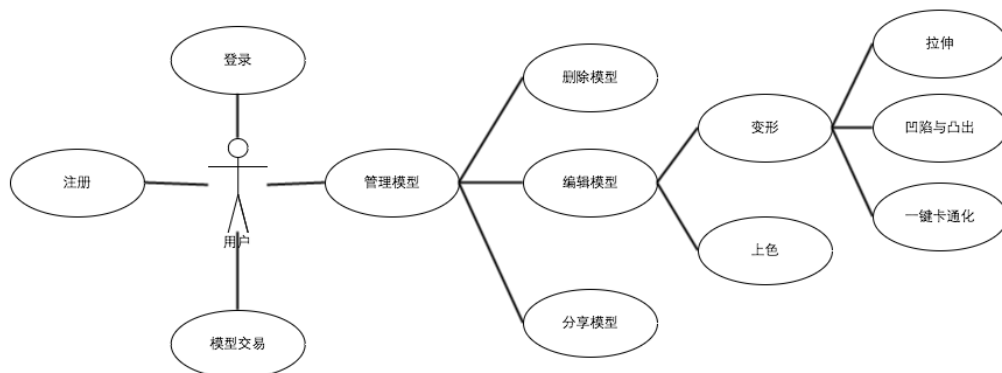
利用拓扑学以及计算机图形学知识实现 3D 模型变形。

利用第三方 API 实现页面中分享功能。

对于用户信息以及模型相关信息进行存储时使用的 MySQL 数据库操作的相关技术。

以上提到的技术或思想，包含了难度较大的 3D 建模的部分，在考虑了我们队员的能力以及其他因素之后，我们认为项目可行性较高。

### 1.4 用例图



---

图 1 用例图

## 1.7 运行环境与支持平台

### 1.7.1 运行环境

支持 Html5 与 WebGL 的浏览器环境（必要）。

### 1.7.2 支持平台

桌面浏览器版的 FingerStudio 支持满足运行环境的 Windows 系统、OS X 系统以及 Linux 系统。

移动平台版的 Fingerstudio 现已支持安装 Android2.3 环境以上的手机以及平板。

---

## 第 2 章 软件综合设计

### 2.1 程序模块划分

#### 2.1.1 UI 界面模块

UI 界面模块以 bootstrap 3 为框架，遵从了响应式设计的原则。

#### 2.1.2 模型加载模块

解析顶点坐标：从 ply 文件中获取顶点坐标。

解析顶点颜色：从 ply 文件中获取顶点颜色。

解析面索引：获取面索引。

#### 2.1.3 模型着色模块

取色器模块：从取色器中获取颜色信息。

着色器模块：对模型选定位置进行着色。

#### 2.1.4 模型编辑模块

凹陷与凸起模块：对模型进行凹陷或凸起操作。

推挤模块：对模型进行推挤操作。

一键特效模块：一键对模型进行特效形变。

### 2.2 算法设计

#### 2.2.1 并行加载模型

由于 Web UI 元素为单线程处理，而 3D 模型普遍容量较大，一般网络环境中加载模型素材会导致 UI 停滞。本应用采用了 html5 多线程技术，即需要加载模型时，产生第二线程解析模型数据，可保证 UI 元素流畅运行，同时充分利用了各硬件平台的 cpu 多核心。

#### 2.2.2 并行拓扑算法

该算法通过面包含的顶点——顶点周围的面——面包含的顶点——顶点周围的面，

---

递归搜索并缓存拓扑数据，允许可对模型进行区域化而非顶点化操作，保证了模型的圆滑度和美观性，达到了某种雕刻效果。由于该算法所需计算资源较大，固采用了并行计算技术，一切拓扑计算均在后台进行，用户无所察觉，使用户更专注于创作本身。

### 2.2.3 变形算法

拓扑运算结束之后，形变操作将对拓扑缓存中的顶点进行非线性位移，其中运用了多种三角函数的结合，使得形变结果符合日常物体的构造属性和物理性质，大大增加了建模的成功率。

### 2.2.4 颜色混合算法

在着色模块中，用户从取色板取色并对模型着色时，颜色混合算法会对之前完成的拓扑结果中的颜色进行统计，并以非线性的方式在模型原有顶点之上渐变式的叠加用户选择的颜色，使得上色达到水彩笔的效果。

### 2.2.5 一键特效算法

为了满足用户对建模速效性和趣味性的要求，该应用特殊加入了类似一键视频滤镜的功能。其算法对整个模型遍历并根据原始数据相对预设关键点的便宜量进行形变，能够瞬间使得模型整体夸张化，卡通化。



## 第 3 章 工作安排

### 3.1 团队工作安排

成员	工作
成员 A	界面开发 (html5+javascript), 快速建模技术设计, 项目统筹安排
成员 B	界面设计 (html5+javascript), 基于 three.js 引擎的图形编程
成员 C	数据库构建 (oracle), 基于 three.js 引擎的图形编程
成员 D	基于 three.js 引擎的图形编程, 服务器架设, 人机交互编程

### 3.2 项目时间进度表

项目重要里程碑	预计完成日期
功能需求分析	2013.04.18 (已完成)
界面原型设计	2013.04.25 (已完成)
程序架构、组员任务分配	2013.04.30 (已完成)
前台初稿定型和搭建完毕	2013.05.20 (已完成)
第一阶段项目整合	2013.06.08 (已完成)
利用 Three.js 完成应用框架搭建和 3D 层搭建	2013.07.20 (已完成)
快速建模功能搭建完毕	2013.07.30 (已完成)
数据库与素材库搭建完毕	2013.08.07 (已完成)
服务器功能搭建完毕	2013.08.13 (已完成)
功能集成测试与优化	2013.08.22 (已完成)
前台 UI 跨平台设计与优化	2013.08.29 (已完成)
Beta 版公众测试与功能调整	2013.09.06 (已完成)
第二阶段项目整合	2013.09.10 (已完成)
快速建模功能完善	2013.09.27 (已完成)
选择已有模型编辑功能	2013.10.11 (已完成)
导出模型	2013.10.16 (已完成)

---

第三方 API 实现分享功能	2013.10.19 (已完成)
前台 UI 跨平台设计与再优化	2013.10.25 (已完成)
Kinect 录入模型	2013.10.29 (已完成)
上色功能	2013.11.03 (已完成)
一键特效	2013.11.11 (已完成)
优化版测试与功能调整	2013.11.13 (已完成)
第三阶段项目整合	2013.11.15 (已完成)
文档整理与展示材料准备	2013.11.18 (已完成)

---

## 第 4 章 引用说明及代码统计

### 4.1 引用说明

引用内容	引用来源	如何使用
Jquery1.8 库	Jquery 官网	用于项目中 ajax、多个动画特效
Bootstrap 3	Bootstrap 官网	UI 设计
Three.js	Three.js 官网	3D 引擎
Parallel.js	Github 开源项目	并行加载与计算

### 4.2 代码统计

语言	JavaScript	Html/CSS	C/C++/Java	PHP/其他 服务器端 脚本	其 他
原创的代码 行数 ● 不含注释 ● 不含第三 方库	7286	1532	403		
第三方库的 代码行数 ● 不含注释	92024	13613			