基于Flask框架的分析与应用

项目计划书

李坤浩、张竹君、张利鹏

崔昕宇、张文斌、许京爽、聂磊

2020年3月18日

目录

[1 引言 1](#_Toc35425054)

[1.1 背景 1](#_Toc35425055)

[1.2 定义 1](#_Toc35425056)

[1.3 标准、条约和约定 1](#_Toc35425057)

[2 项目团队组织 2](#_Toc35425058)

[2.1 人员组成 2](#_Toc35425059)

[2.2 协作方式 2](#_Toc35425060)

[3 项目概述 2](#_Toc35425061)

[3.1 项目目标 2](#_Toc35425062)

[3.2 应交付成果 2](#_Toc35425063)

[4 实施计划 3](#_Toc35425064)

[4.1 软件开发构想 3](#_Toc35425065)

[4.2 软件测试构想 3](#_Toc35425066)

[4.3 项目时间计划安排表 3](#_Toc35425067)

[5 参考资料 4](#_Toc35425068)

# 1 引言

## 1.1 背景

Flask是python常用的web开发框架，其具有很多优点因此受到广泛的使用，它非常的灵活、轻便、安全且容易上手，很好地结合MVC模式进行开发，具有很强的定制性，用户可以根据自己的需求来添加相应的功能，在保持核心功能简单的同时实现功能的丰富与扩展，同时，其具有强大的插件库，可以让用户实现个性化的网站定制，开发出功能强大的网站。

## 1.2 定义

关于Flask框架的专业术语与缩略语如表1所示。

**表1 专业术语及缩略语**

|  |  |
| --- | --- |
| 术语/缩略语 | 描述 |
| Web应用框架 | Web应用框架（Web application framework）是一种开发框架，用来支持动态网站、网络应用程序及网络服务的开发，有助于减轻网页开发时共通性活动的工作负荷。例如，许多框架提供数据库访问接口、标准样板以及会话管理等，可提升代码的可再用性。[1] |
| Python | Python是一种跨平台的计算机程序设计语言，具有面向对象和动态类型的特性。Python最初被设计用于编写自动化脚本，随着版本的不断更新和语言新功能的添加，越多被用于独立的、大型项目的开发。[2] |
| WSGI | Web服务器网关接口（Python Web Server Gateway Interface，缩写为WSGI）是为Python语言定义的Web服务器和Web应用程序或框架之间的一种简单而通用的接口。[3] |

## 1.3 标准、条约和约定

参照如下标准：

GB/T 13702-1992 计算机软件分类与代码

GB/T 19003-2008 软件工程

GB/T 5538-1995 软件工程标准分类法

GB/T 9385-2008 计算机软件需求规格说明

GB/T 5532-2008 计算机软件测试规范

GB/T 18221-2000 信息技术程序设计语言

GB/T 8567-2006 计算机软件文档编制规范

# 2 项目团队组织

## 2.1 人员组成

Flask开发项目的团队人员组成如表2所示。

**表2 人员组成**

|  |  |  |  |
| --- | --- | --- | --- |
| **序号** | **学号** | **姓名** | **职责** |
| 1 | SY1906122 | 崔昕宇 | 文档、代码、测试 |
| 2 | SY1906418 | 李坤浩 | 文档、代码、测试 |
| 3 | SY1906419 | 聂磊 | 人事协调、文档、代码、测试 |
| 4 | SY1906422 | 许京爽 | 文档、代码、测试 |
| 5 | SY1906424 | 张利鹏 | 文档、代码、测试 |
| 6 | SY1906425 | 张文斌 | 架构设计、文档、代码、测试 |
| 7 | SY1906610 | 张竹君 | 文档、代码、测试 |

## 2.2 协作方式

返校前的线上协作方式：通过github、微信进行文件共享，通过微信进行文字讨论和语音交流。

待返校后，以线下交流代替微信语音交流。

# 3 项目概述

## 3.1 项目目标

需求分析：了解软件需求，编写需求规格说明书。

源码阅读：掌握项目细节，进行反向UML建模。

软件开发：构建一套深度学习模型的自动化部署系统。

软件测试：撰写测试需求分析文档，对软件进行充分测试。

## 3.2 应交付成果

项目应交付的产品如表3所示。

**表3 应交付成果列表**

|  |  |  |  |
| --- | --- | --- | --- |
| **编号** | **名称** | **交付时间** | **备注** |
| 1 | 项目计划书 | 2020/3/18 | 暂无 |
| 2 | 软件需求规格说明书 | 待定 | 暂无 |
| 3 | 软件评审报告 | 待定 | 暂无 |
| 4 | 软件设计模型 | 待定 | 暂无 |
| 5 | 测试需求规格说明书 | 待定 | 暂无 |
| 6 | 软件测试分析报告 | 待定 | 暂无 |
| 7 | 软件测试评审报告 | 待定 | 暂无 |
| 8 | 项目方法总结报告 | 待定 | 暂无 |

# 4 实施计划

## 4.1 软件开发构想

高效的深度学习模型需要通过实际部署才能在生产生活中发挥作用。常用的部署方式是将模型包装成Web服务，向外界提供基于HTTP协议的接口，从而提供各类预测功能和服务。对于算法研究人员而言，将深度学习模型部署为Web服务的学习成本较高，通常需要工程开发人员协助部署，这一定程度上降低了算法研发和实际部署的效率。本项目旨在通过自动化部署深度模型，降低算法研究人员实际部署模型的门槛，提高深度学习模型迭代研发和部署的效率。

考虑到语言和部署效率的因素，项目将采用基于Python语言的轻量级Web框架flask和容器技术作为实际的模型部署工具，同时构建一套自动化部署系统供用户使用，包括了前端交互页面，后端系统和数据库三部分，用户上传模型文件并通过界面操作即可完成模型的部署，无需使用者具备Web开发和部署的相关知识和经验。本系统将为算法研究人员提供高效可扩展的深度学习模型部署服务。

## 4.2 软件测试构想

Flask框架包含了会话、视图、日志、并发部分。目前初步的测试思路是先围绕request和response会话流程进行测试，再针对Session板块以及数据库扩展、文件上传、后台管理进行测试，在完成了以上基本测试后，可以对项目进行并发访问的压力测试。

## 4.3 项目时间计划安排表

Flask项目的时间计划安排如表4所示。

**表4 项目时间计划安排表**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 当前状态 | 编号 | 阶段名称 | 工期 | 预计开始时间 | 预计结束时间 |
| 未开始 | 1 | 软件需求分析 | 2周 | 2020/3/14 | 2020/3/28 |
| 未开始 | 2 | 软件需求评审 | 2周 | 2020/3/28 | 2020/4/11 |
| 未开始 | 3 | 软件设计与实现 | 3周 | 2020/4/18 | 2020/5/9 |
| 未开始 | 4 | 软件测试需求分析 | 1周 | 2020/5/9 | 2020/5/16 |
| 未开始 | 5 | 软件测试评审 | 3周 | 2020/5/16 | 2020/6/6 |
| 未开始 | 6 | 软件项目计划与监控 | 15周 | 2020/3/14 | 2020/6/25 |
| 未开始 | 7 | 配置管理 | 15周 | 2020/3/14 | 2020/6/25 |
| 未开始 | 8 | 软件工程实验追踪与分析 | 15周 | 2020/3/14 | 2020/6/25 |

# 5 参考资料

[1] 百度百科：Web应用框架

[2] 百度百科：Python

[3] 维基百科：WSGI

[4] Flask官方文档 <https://flask.palletsprojects.com>