



	<p><b>理论与实证准备</b></p> <p>软件工程作为一门综合性学科，软件工程的系统化方法可以为需求分析、设计、编码、测试与维护等阶段提供了清晰的流程，确保了平台功能的全面性与稳定性。</p> <p>领域驱动设计通过建立领域模型，将复杂业务逻辑转化为软件设计中的实体、值对象、聚合等概念，使得代码结构紧密贴合业务需求。不仅可以精准描述业务需求，对于高效响应、处理并反馈这些业务需求也有良好的保证。</p> <p>SpringBoot 框架以其轻量级、易配置的特性，成为构建高效、可扩展的微服务系统的首选。这些特性极大地提高了开发效率，确保了平台能够快速响应变化，满足高并发、大数据量的处理需求。</p> <p>Uni-App 框架为提供了统一的解决方案，支持多端适配。这一特性确保了无论用户使用何种设备，都能获得一致的使用体验。</p> <p>本研究通过阅读调研文献、用户访谈的方式了解现有救助模式的局限性及其改进方向，为新平台的功能设计提供参考。基于选定的技术栈配置相应开发环境，搭建五台中间件服务器以验证关键技术是否能满足预期效果，如数据库连接池配置、缓存机制实现等。对于数据安全与隐私保护，研究并设计的安全策略和技术手段。</p> <p><b>拟解决的问题</b></p> <ol style="list-style-type: none"><li>1. 系统数据的管理与整合</li><li>2. 用户体验优化</li><li>3. 领域划分设计</li><li>4. 信息安全保障</li><li>5. 移动应用适配</li></ol> <p><b>研究（设计）方法与技术路线</b></p> <ol style="list-style-type: none"><li>1.需求分析：根据前期调研结果明确系统功能需求，确定主要功能，如贴子发布预览、参与救援、申请领养、动物投喂等功能点。</li><li>2.系统技术选型：采用软件工程技术，结合领域驱动设计（DDD）的理念，通过前后端分离架构构建了一款专门服务于流浪动物救助的综合性平台。该平台前端使用 Uni-App 框架进行开发，配合 Httpd 作为文件资源服务器，后端则基于 Spring Boot 框架，并引入 Elasticsearch 以支持高效的全文搜索功能。</li><li>3.原型制作：利用 Figma 工具绘制高保真的产品原型</li><li>4.编码实现：按照既定计划分阶段完成各部分代码编写工作</li><li>5.测试优化：使用开源测试工具对系统进行全面测试（功能、性能、兼容性等）并邀请少量用户参与试用</li><li>6.部署上线：选择合适的云服务商进行产品发布，配置好服务器环境及相关安全设置后对外开放访问。</li></ol>
--	--

论文写作 提纲	<p>摘要</p> <p>ABSTRACT</p> <p>1 绪论</p> <p>1.1 应用背景及意义</p> <p>1.2 国内外相关应用现状</p> <p>1.3 论文结构安排</p> <p>2 流浪动物救助平台需求分析</p> <p>2.1 流浪动物救助平台功能性需求</p> <p>2.2 流浪动物救助平台非功能性需求</p> <p>2.3 流浪动物救助平台的可行性</p> <p>3 流浪动物救助平台的设计</p> <p>3.1 流浪动物救助平台设计目标</p> <p>3.2 流浪动物救助平台架构设计</p> <p>3.3 流浪动物救助平台数据库设计</p> <p>3.4 流浪动物救助平台功能设计</p> <p>3.5 流浪动物救助平台原型设计</p> <p>4 流浪动物救助平台的实现</p> <p>4.1 流浪动物信息中心实现</p> <p>4.1.1 登陆注册</p> <p>4.1.2 信息管理</p> <p>4.2 流浪动物交流论坛实现</p> <p>4.2.1 帖子浏览与发布</p> <p>4.2.2 帖子评论区</p> <p>4.2.3 聊天室</p> <p>4.3 流浪动物领养服务实现</p> <p>4.3.1 领养申请</p> <p>4.3.2 申请审核</p> <p>4.4 流浪动物救助平台后台实现</p> <p>4.4.1 用户信息管理</p> <p>4.4.2 动物信息管理</p> <p>4.4.3 帖子内容管理</p> <p>4.4.4 评论内容管理</p> <p>5 流浪动物救助平台的测试</p> <p>5.1 功能模块测试</p> <p>5.2 性能测试</p> <p>5.3 系统测试</p> <p>5.4 测试结论</p> <p>6 总结与展望</p> <p>参考文献</p>
------------	--

