

静态分析工具需求说明

(第二版)



撰写者：南京大学 程珂

南京大学 陈昊东

南京大学 施超烜

南京大学 吴晓阳

南京大学 周若衡

目录

- 1. 引言3
 - 1.1 编写目的3
 - 1.2 需求说明(第二版)概述3
- 2 改进4
 - 2.1 概述4
 - 2.2 缺陷检测项变更.....4
 - 2.3 细化用户用例图.....4
 - 2.4 事件列表变更5
 - 2.5 细化软件质量属性需求7
- 3 新的需求7
 - 3.1 新需求概述7
 - 3.2 新的需求7
 - 3.3 命令集设计8
 - 3.4 GUI 界面设计9
 - 3.5 缺陷检测报告设计.....10
- 4 结语10
 - 4.1 第二周期预期目标.....10
 - 4.2 项目未来发展方向.....11

1. 引言

1.1 编写目的

本项目基于演化模型进行开发，第二版需求说明书是对项目第一周期(2020//7~2020/5/7)需求说明的补充和改进，同时也包括了基于项目第一周期的阶段性成果提出的新的需求。第二版需求说明的编写目的在于对第一周期的后续工作进行补充以及对第二迭代周期的工作进行指导。帮助编码人员更好的理解第二周期的目标从而准确高效的完成第二周期的任务。与此同时还可以辅助我们与用户的交流，从而从用户的角度得到更有效的反馈。

1.2 需求说明(第二版)概述

需求说明(第二版)在项目背景、预期的读者和建议、项目前景、项目功能概述、用户类及其特征等内容上与第一版无区别，因此不再重复编写。

本版需求说明第2节介绍了对于第一版需求说明的改进和补充，旨在记录在项目的进展过程中，随着开发人员对项目难点、重点和用户需求的理解加深，从而对第一版需求中的部分内容进行的更改和细化。在第2节中，我们详细列出了更新的原因和修改/添加的位置及更新后的新内容。

本版需求说明第3节介绍了在项目的第二个迭代周期中需要完成的新的功能及其与用户交互的效果，旨在第二个迭代周期中对开发人员做出指导，使其更加容易理解各个功能的实现效果，从而高效准确的完成开发任务。同时也向用户和上级检查人员和用户展示我们对需求的理解，以便从上级和用户得到有效的反馈和修改意见。

本版需求说明第4节介绍了项目第二周期的预期目标和未来的开发发展方向。

2 改进

2.1 概述

在第 2 节中，我们对第一版需求说明中的部分内容进行了改进，主要包括缺陷检测项的变更，用户用例图的细化，对事件列表的修改以及细化了软件质量属性需求的部分条目。

2.2 缺陷检测项变更

在开发的过程中，我们变更了部分原先预定的缺陷检测项。将继承类数组指针问题和整数溢出缺陷检测删去，添加对重复释放内存缺陷的检测。

之前预计完成的缺陷检测项：内存泄露、内存释放后使用、除 0 模 0、使用结构体中未初始化成员、空指针解引用、继承类数组指针问题、整数溢出共 7 个缺陷

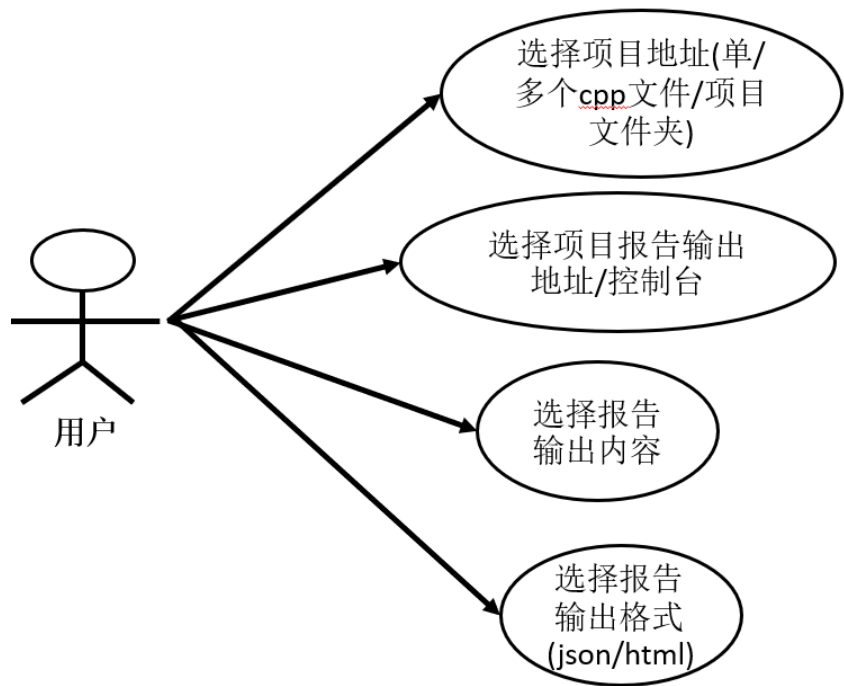
更改后的缺陷检测项：内存泄露、内存释放后使用、除 0 模 0、使用结构体中未初始化成员、空指针解引用、重复释放内存共 6 个缺陷。

更改原因：在开发的过程中，考虑到开发人员的经历和学习成本较高，以及部分缺陷之间关联程度较低，因此**暂时**先取消对于部分缺陷的检测，添加对于与其他检测项关联程度较高的内存重复释放缺陷的检测。在后续工作中，随着开发人员对相关知识的了解更加深入和对 API 的使用更加熟练，我们也许会重新添加新的缺陷检测项。但是总缺陷项数**不会少于**在本版需求说明中确定的 6 个缺陷检测项。

2.3 细化用户用例图

随着项目进展的深入，我们对用户交互过程的理解也更加深刻。因此在第一版需求说明的基础上对用户用例图进行了更改和更加细致的划分。在当前阶段，本项目拟开发命令行版

本与图形界面(GUI)版本，在这两个版本中均支持下图中的功能。



2.4 事件列表变更

在命令行版本中，进行如下更改

操作	事件	触发	事件源	行为	响应	事件宿
原： (1)	用户查看自己的项目列表	用户输入查看项目列表的命令	用户	从数据库中根据用户账号信息读取用户的项目列表，整理成项目列表展示给用户	用户项目列表	用户
改： (1)	用户查看检测历史记录	用户输入查看历史记录的命令	用户	从数据库中读取用户检测历史，将历史记录中每一次检测的检测时间，各项检测选项，项目所在地址，报告输出地址等信息整理成历史记录条目，依次输出	用户检测历史记录	用户

删：	用户重新检测	用户输入重新检测的命令,提交项目文件重新并选择需要检测的缺陷	用户	将检测项信息和项目文件地址发送给程序的 Processor 组件进行预处理,展示缺陷检测的进度	缺陷检测的进度	用户
增：	用户删除历史记录	用户输入删除历史记录的命令和选择需要删除的命令范围	用户	从数据库中删除用户相应的检测历史信息	删除后的历史记录	用户
增：	用户选择输出格式	用户输入选择输出格式的命令和选择的输出格式 (json/html)	用户	在内存中载入用户选择的输出格式	用户选择的输出格式	用户
增：	用户选择输出内容	用户输入选择输出内容的命令和选择的输出内容(检测时	用户	在内存中载入用户选择的输出内容	用户选择的输出内容	用户

		间, 各种缺陷 占比, 缺陷所 在位置等)				
--	--	-----------------------------	--	--	--	--

2.5 细化软件质量属性需求

对于第一版需求说明中的**健壮性需求**，我们提出了更加具体的指标：项目测试的语句覆盖率不低于 90%，分支覆盖率不低于 60%。

3 新的需求

3.1 新需求概述

在本节中，我们对于在第二个迭代周期(2020/5/7~2020/6/7)中预计要新增的功能进行需求说明，在第一版本周期中，我们只简略的描述了是用命令行进行交互以及可以使用的功能，在第二个周期中，我们将明确命令行版本所支持的命令集，从而为命令解析模块的开发提供参考；明确缺陷检测输出报告的 HTML 界面模板，从而对于输出(Printer)模块的开发进行参考，并且明确可输出的格式为 json 和 HTML 两种，便于用户使用 json 文件自行进行输出渲染，也方便用户将本项目集成进自己的工作流中；我们还将进行图形界面(GUI)版本的开发，因此在本节中我们还将详细介绍 GUI 的设计，为相应模块的开发者提供参考。

3.2 新的需求

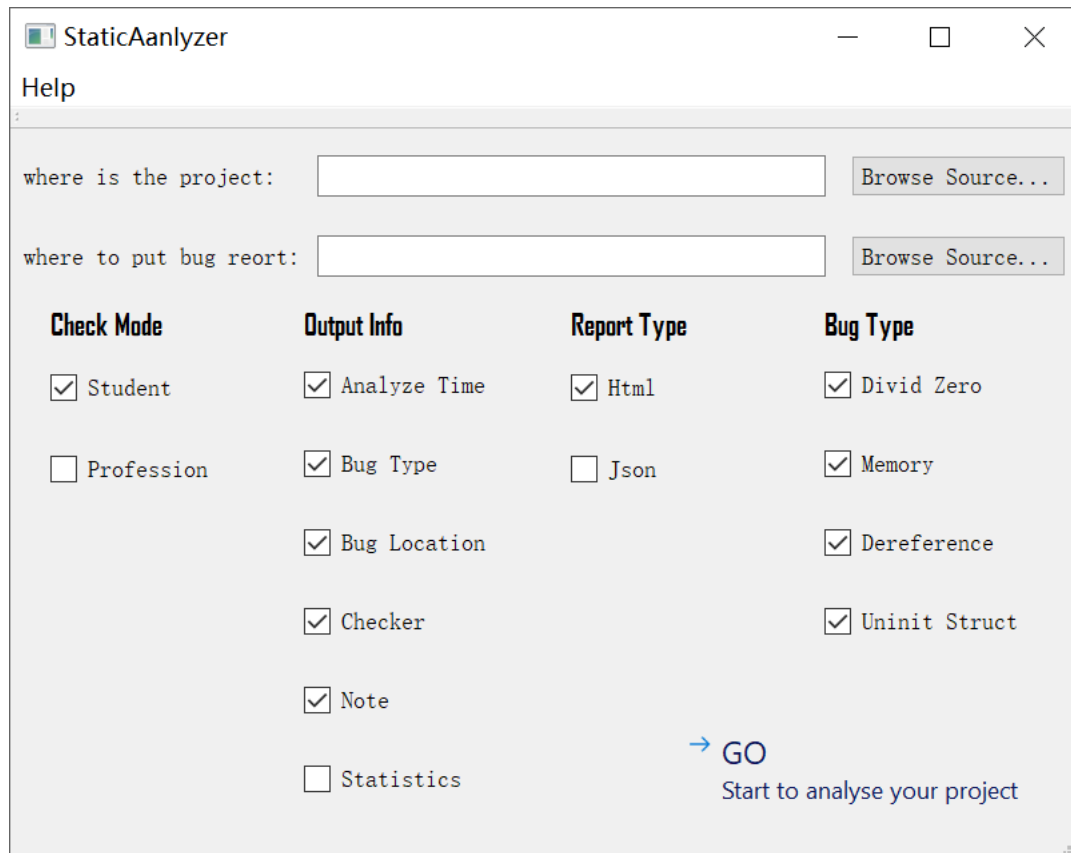
在第二个周期中，新增的需求主要在用户与图形界面的交互过程，用户所能完成的功能是不变的，只是操作形式和输出方式发生了改变。新增需求的事件名、事件源、时间宿、与

第一版需求命令行交互的过程都相同,唯有触发由“输入相应命令”变更为“点击相应控件”,
响应由命令行展示变更为图形界面展示。具体的控价设计及其功能在 3.4 会详细介绍,所以在此不再绘制事件列表。

3.3 命令集设计

命令	参数	实例	解释	类型
-i addr1 addr2 ...	被检测项目文件所在地址, 支持单个 <code>cpp</code> 文件, 多个 <code>cpp</code> 文件和项目文件夹的输入形式	-i helloAst.cpp hellowlr.cpp	输入	必须
-o addr type	指定缺陷检测报告的输出地址和输出类型, 默认为命令行输出, <code>type</code> 参数支持 <code>html</code> : 输出 HTML 版本 和 <code>json</code> : 输出 json 版本	-o D:\test html	输出	可选
-s choice1 choice2 ...	制定检测项, 默认为全选, <code>choice</code> 参数支持 <code>dz</code> (除 0,模 0), <code>mem</code> (内存重复释放, 内存释放后使用, 内存泄露), <code>uo</code> (结构体成员未初始化), <code>der</code> (空指针解引用)	-s dz mem	需要检测的选项	可选
-p	静态分析工具默认调用 <code>student</code> 版本, 调用项目成员自行编写的检测模块, 在 <code>-p</code> 模式下设定为 <code>profession</code> 版本, 将会调用 <code>clang-static-analyzer</code> 进行静态分析, 无参数, 在 <code>-p</code> 模式下, <code>-s</code> 命令无效	-p	Profession 模式	可选
-c choice1 choice2 ...	输出内容, 默认输出检测时间(<code>time</code>), 缺陷种类(<code>type</code>), 缺陷所在位置(<code>loc</code>), 检测出该缺陷的模块(<code>chekcer</code>) 和其他相关信息(<code>note</code>), <code>choice</code> 参数支持以上所有选项, 还支持 <code>statis</code> 选项, 输出每一种缺陷在项目中的个数和在缺陷总数中的比例	-c type loc note statis time	输出内容	可选

3.4 GUI 界面设计



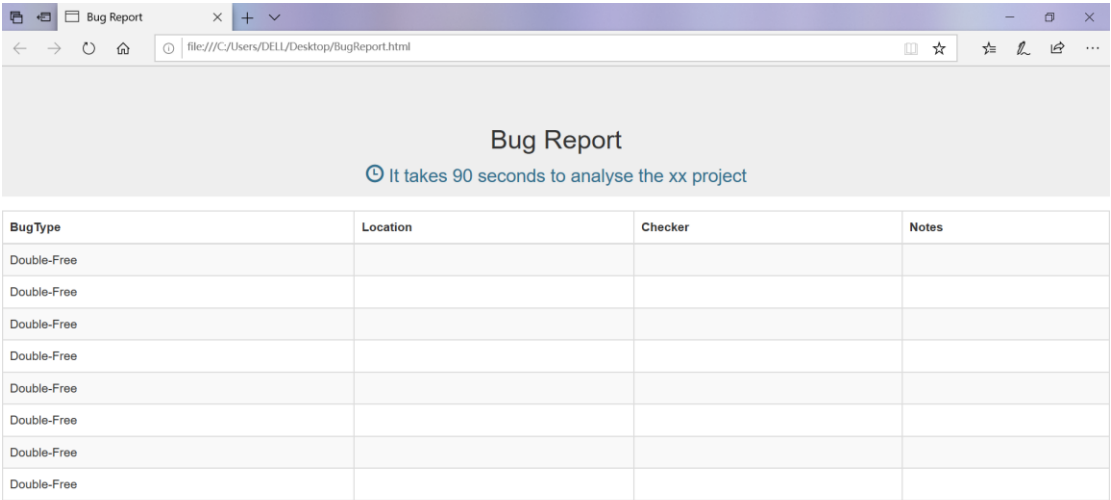
界面设计如上图所示，以下逐个介绍每个控件的作用

1. 菜单栏中的 Help 选项：用于打开与工具相关的信息，如：开发者邮箱，github 仓库以及功能简介和对部分控件的介绍。
2. 项目所输入地址栏：可以直接输入项目所在地址，也可以通过点击右侧按钮选择项目文件，支持单个 cpp 文件，多个 cpp 文件以及项目文件夹格式的输入。
3. 缺陷检测报告输出地址栏：可以直接输入报告输出地址，也可以通过点击右侧按钮选择
4. Check Mode 列：默认为 student，其中 profession 选项对应命令行中 -p 命令
5. Output Info 列：默认选项如上图所示，对应命令行中 -c 命令
6. Report Type 列：默认选项如上图所示，对应命令行中 -o 的 type 参数
7. Bug Type 列：默认选项如上图所示，对应命令行中 -s 命令

3.5 缺陷检测报告设计

以 HTML 格式输出的缺陷检测报告默认界面如下：界面设计简介明了，缺陷类型，缺陷所在位置，检测出缺陷的 Checker 以及相关的其他输出一目了然。

该输出报告支持当下主流浏览器：Chrome，Firefox 以及 Microsoft Edge



也可以根据用户在命令行/GUI 界面的不同设定，输出特定的统计信息：

Conclusion

Bug Type	Num	Percent
Use Afer Free	3	30%
Double Free	4	40%
Div Zero	3	30%

4 结语

4.1 第二周期预期目标

第二周期的预期目标为：在核心缺陷检测功能不断推进的过程中，完成图形界面模块，

命令解析模块和输出模块和编写以及测试，生成详细的设计文档和测试文档。并且在第二周期结束时能够完成选择的 6 个缺陷检测中至少 4 个项过程内的基本检测功能。

4.2 项目未来发展方向

项目当前拟定只完成命令行版本和图形界面版本，在后续开发的过程中可能会实现 Web 版本，也可能会增加新的缺陷检测项。