

北京理工大学

结课作业报告

《Java 程序设计》结课项目文档

Final Report of Java Programming

学 院： 计算机学院

专 业： 数据科学与大数据技术（全英文教学专业）

2025 年 12 月 23 日

目 录

第 1 章 程序的运行环境、安装步骤 1

第 2 章 程序开发平台 2

第 3 章 程序需求分析 3

 3.1 动因描述 3

 3.2 竞品分析 3

 3.3 功能描述 5

 3.4 UI 界面与交互设计 11

第 4 章 程序架构设计与技术实现方案 13

第 1 章 程序的运行环境、安装步骤

本项目使用 JDK 21 进行编译，程序主体为由源码编译、包装得到的 *ClinicAssistant.exe*，需要配合安装包附带的 *ExternalTools*、*app* 和 *runtime* 文件夹使用。*ExternalTools* 文件夹包含了程序运行需要的 *LibreHardwareMonitorWrapper*、*Furmark2*、*cpuburn*、*CLINIC_OP* 工具。

本项目无需安装，在确保上述 *ExternalTools*、*app* 和 *runtime* 文件夹与 *ClinicAssistant.exe* 在同级目录时，使用管理员权限开启 *ClinicAssistant.exe* 即可。

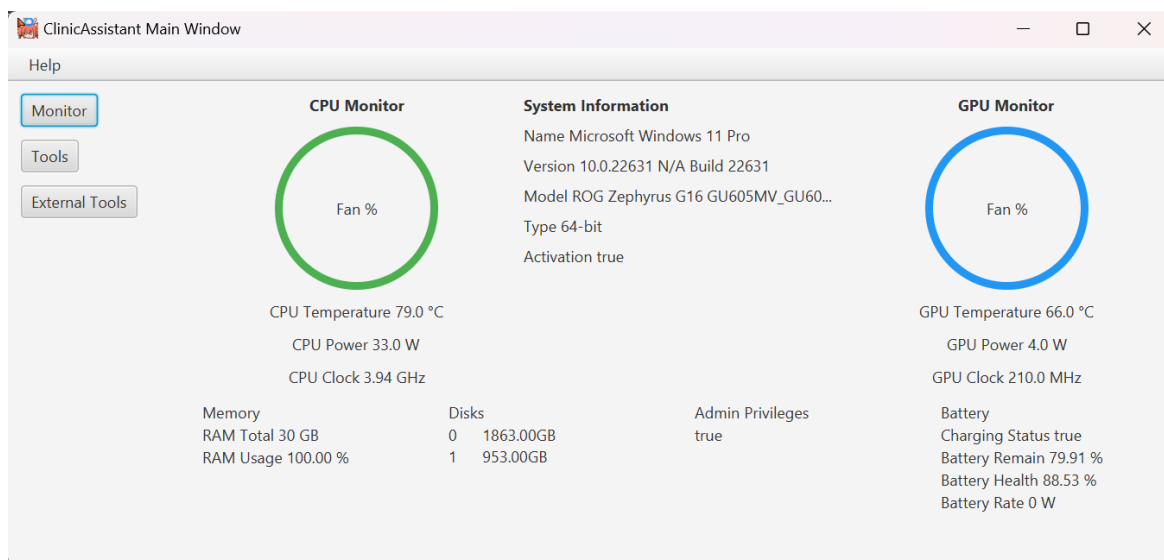


图 1-1 项目预览图

第 2 章 程序开发平台

本项目有 Java 代码 4214 行，Python 代码 84 行，C# 代码 803 行，共计 5101 行代码。程序开发主要使用以下平台和工具：

语言	开发平台	语言版本
Java	IntelliJ IDEA 2025.3	Azul 21.0.9
Python	PyCharm 2025.3	Python 3.13.5
C#	Rider 2025.3	.NET 9.0
L ^A T _E X	VS Code 1.107.1	3.141592653

其中，Java 负责程序主要逻辑和界面的实现，Python 用于生成大量简单重复的样板代码，C# 用于与 LibreHardwareMonitor 通信。

第3章 程序需求分析

3.1 动因描述

我是我校校级学生组织网络开拓者协会下属电脑诊所的一员。在修电脑的过程中，我们经常需要进行烤机测试、BitLocker 解锁、硬件状态查看等操作。这需要在诊所的工具箱 *CLINIC_OP* 中打开位于不同目录的若干软件、在多个软件之间切换，非常不方便。有时，一些常见的维修需要使用命令行进行，这就要求诊所的同学记住不少 Powershell 命令，无疑增大了维修的精力成本。另外，诊所常做的烤机测试¹需要人工全程监视硬件状态，以判断电脑的散热能力，这理应被自动化的工具代替，作为一个自动进行的工作。

因此，我开发了 *ClinicAssistant* 实用功能工具箱，提供硬件信息监视、一键激活 Windows、一键进入 BIOS、解锁 BitLocker、重置代理、修复网卡代码 56 错误、快捷进入 *CLINIC_OP* 工具等诊所常用的实用功能。另外，借助数值分析的知识，我对诊所判断烤机结果的经验方法进行数学建模，从而实现了自动烤机测试功能。本工具可以降低电脑维修的技术门槛、规范维修的操作流程，从而极大地便利诊所同学维修电脑，使诊所的服务效率更高、更规范。

3.2 竞品分析

目前，最常见的集成了若干维修工具的工具箱程序为“图吧工具箱”。

图吧工具箱类似于一个导航页，收集、罗列出了电脑维修各个方面的若干软件，并为每个软件提供了简单介绍。它允许直接从程序内启动这些软件，而本身并不提供任何维修工具。除此以外，图吧工具箱无其它自动化功能。此外，图吧工具箱缺少常用命令的一键执行脚本。对于电脑维修的新手而言，图吧工具箱并不能起到太大帮助；对于经验丰富的高手来说，又没必要从图吧工具箱去启动工具。

¹烤机测试：指借助特殊软件使电脑的 CPU、GPU 处于最大负荷运行状态一段时间，通过对电脑温度、功率等指标的观察判断其最大性能和散热能力。

结课作业报告



图 3-1 图吧工具箱

诊所最常使用的工具莫过于电脑硬件指标（温度、功率、频率等）的监测和烤机测试工具。指标监测常用软件是 *AIDA64* 或 *HWiNFO*；烤机测试常用软件是 *cpuburner*（集成于 *AIDA64*）和 *Furmark*。每次进行烤机操作时，都需要打开 *AIDA64* 和 *Furmark* 两个软件并在软件内操作。*AIDA64* 可以同时进行 CPU 的烤机测试和硬件指标检测，但是其没有指标－时间图象的功能，无法直观判断指标变化趋势。因此有时需要额外开启 *HWiNFO* 以获取图象。这样，为了完成烤机测试就需要开启三个程序，操作上非常不便。

此外，上述的两款检测软件中 CPU 和 GPU 数据间被其它硬件信息分隔开，并且软件给出的信息太多，而大多对于烤机测试无用。因此烤机时必须在二者间上下滑动翻找需要的信息，也带来了不必要的繁琐操作和学习成本。

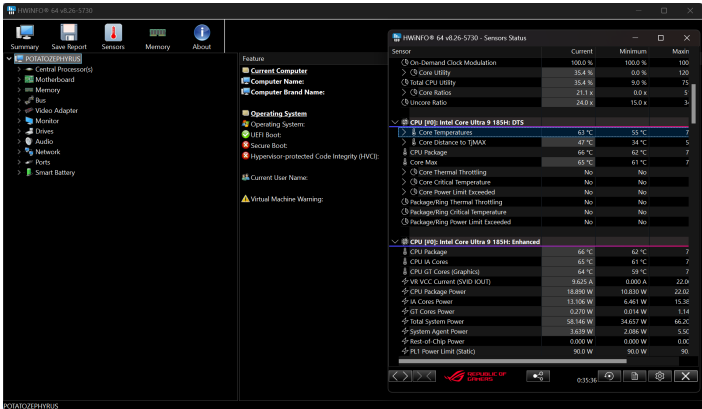


图 3-2 HWINFO

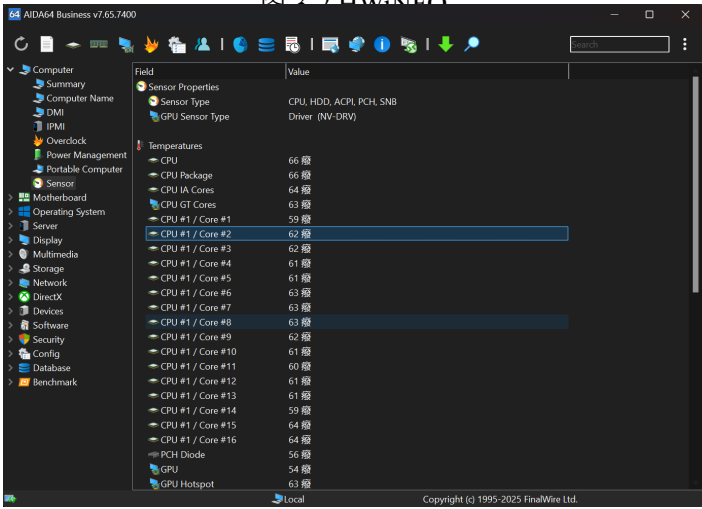


图 3-3 AIDA64

图 3-4 两款常用硬件监测软件

因此，*ClinicAssistant* 提供的常用指标监视；自动化、一键式操作可以极大地方便电脑维修的各种操作。

3.3 功能描述

本程序作为维修电脑的实用工具箱，主要面向有电脑维护维修需求的同学。通过本程序可以直观地观察到电脑的各项硬件信息和指标、方便地进行各类维护诊断操作。具体功能清单如下：

硬件信息监视：实时监视电脑的 CPU 温度、功率、频率；GPU 温度、功率、频率；内存大小与占用率；物理硬盘数量及大小；电池充电状态、剩余电量、电池健康和充放电功率。系统信息监视：监视系统的名称、版本号、主板模具、位数和激活

状态。如图3-5所示。

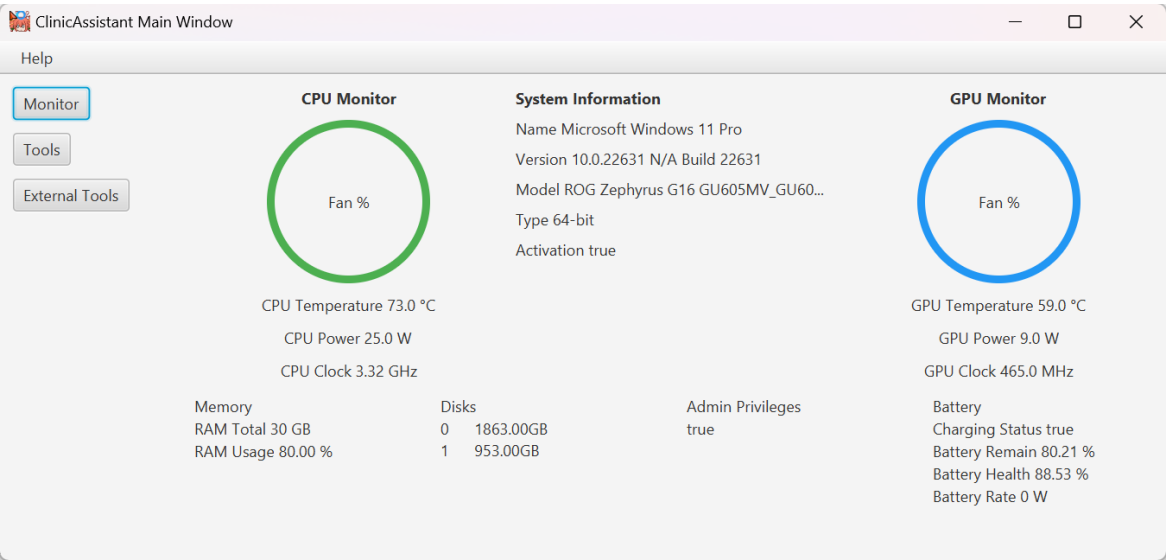


图 3-5 监视器页面

实用功能菜单：提供了一键激活 Windows、一键重启并进入 BIOS、一键解锁 BitLocker、一键代理重置、一键网卡 Code56 修复和自动烤机测试功能。如图3-6所示。

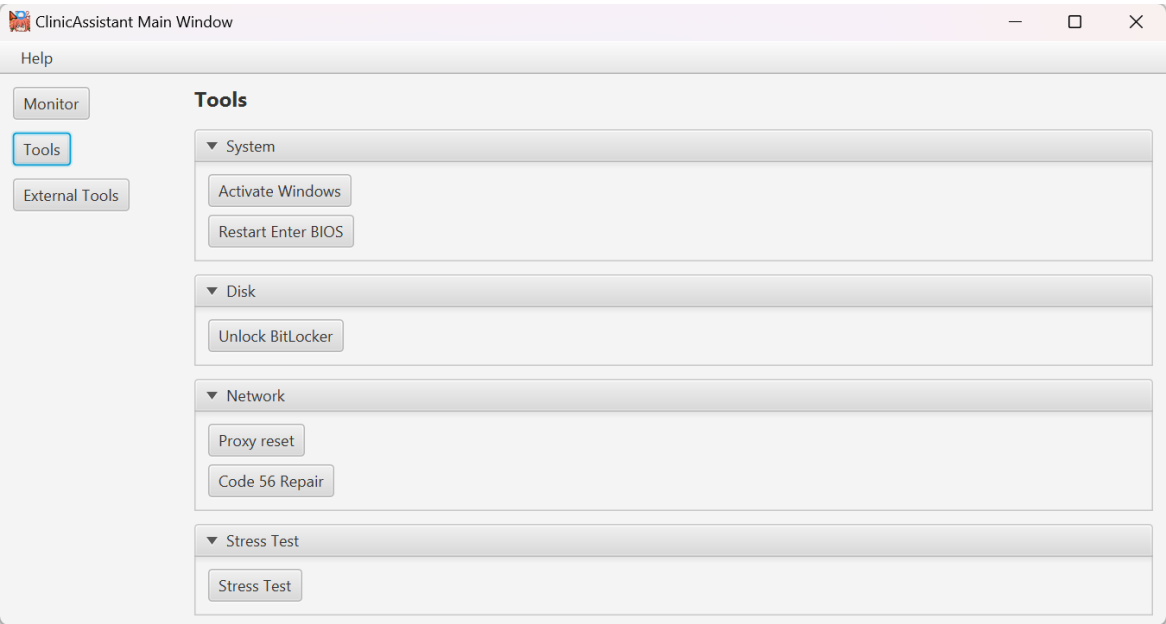


图 3-6 实用工具页面

结课作业报告

外部功能导航：提供了快速打开外部工具的导航页，点击按钮即可打开对应工具。如图3-7所示。

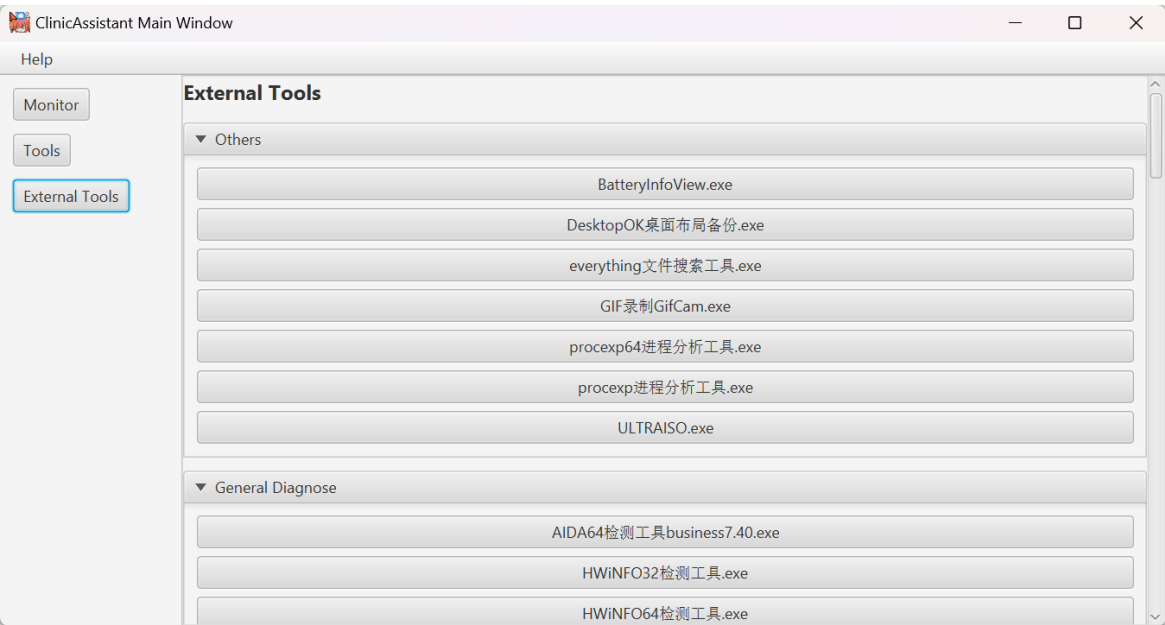


图 3-7 外部工具导航页面

帮助页面：帮助界面附上了网协 wiki 的链接和各咨询群的群号。如图3-8所示。

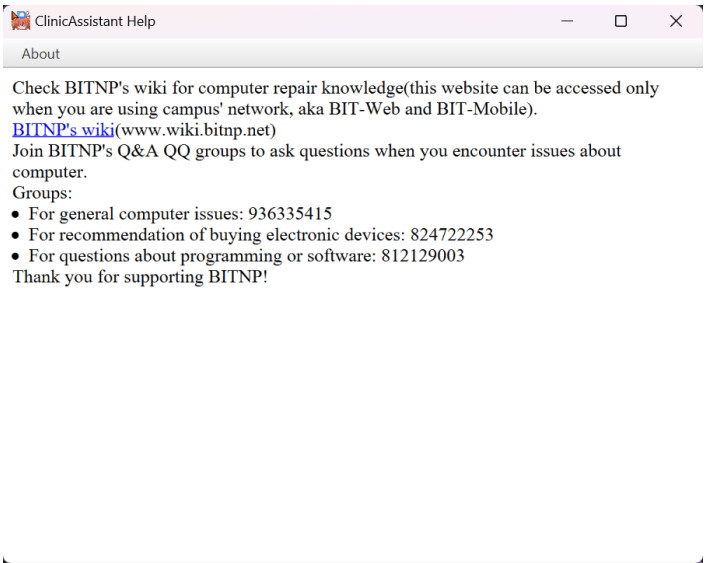


图 3-8 帮助界面

结课作业报告

关于页面：关于界面附上了网协首页与本项目 GitHub 仓库的链接。如图3-9所示。



图 3-9 关于界面

版本页面：关于界面附上了当前 *ClinicAssistant* 的图形化界面和内核版本。如图3-10所示。

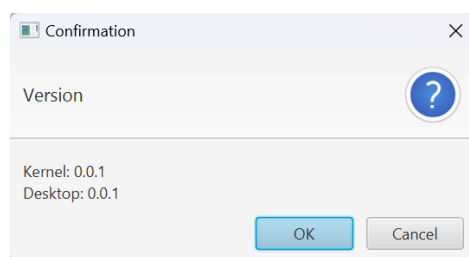


图 3-10 版本界面

一键激活 Windows：使用激活脚本激活 Windows 系统，点击按钮即可激活 Windows 系统，若已激活系统则无法按下激活按钮。如图3-11所示。

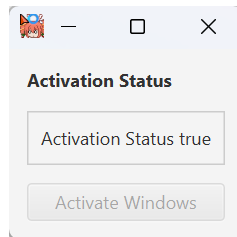


图 3-11 Windows 激活工具

一键进入 BIOS：重启电脑并进入 BIOS 设置界面。如图3-12所示。

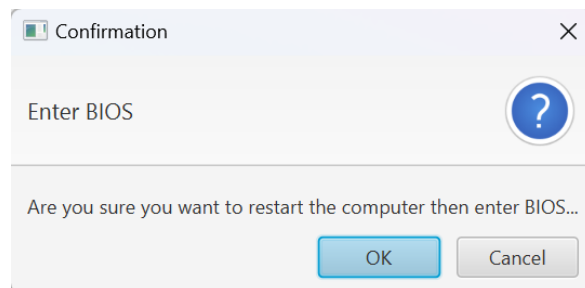


图 3-12 重启并进入 BIOS 工具

一键解锁 BitLocker：实时识别电脑上所有磁盘分区的 BitLocker 加密情况，在没有完全解密的磁盘上按下左键即可招出确认菜单，确认后自动解锁 BitLocker。如图3-13所示。

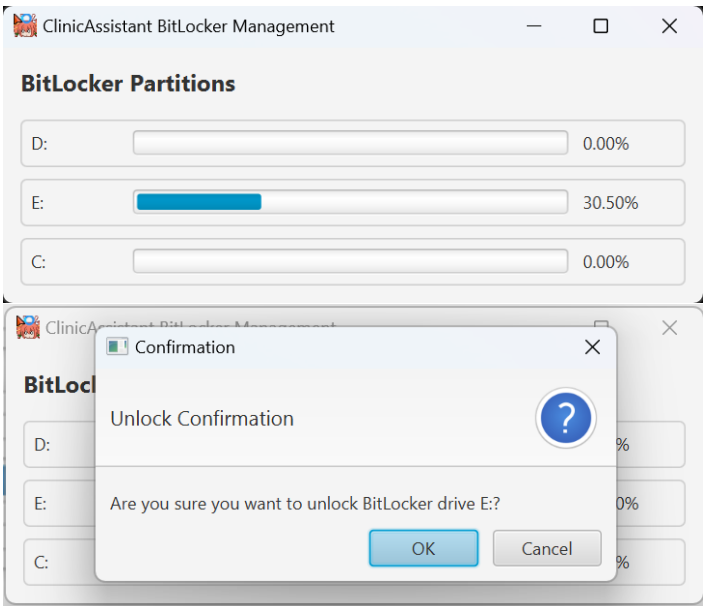


图 3-13 BitLocker 解锁工具

一键重置网络代理：重置网络代理。如图3-14所示。

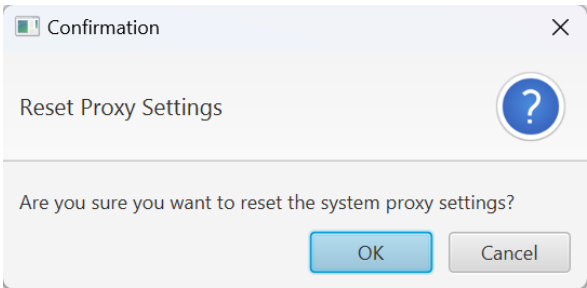


图 3-14 重置网络代理工具

一键修复网卡代码 56：修复网卡驱动报错“代码 56”。如图3-15所示。

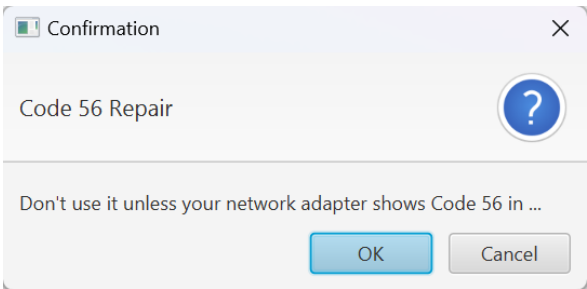


图 3-15 重置网络代理工具

结课作业报告

自动烤机：勾选需要烤 CPU 或/和 GPU，输入被烤电脑电源适配器的功率（可选），自动进行烤鸡测试。在测试时可以查看已测试时间、温度 — 时间图象和功率 — 时间图象、算法对指标的实时评估。默认执行自动模式：当算法判定烤机效果已经可以说明散热良好时烤机自动结束。若 CPU 或 GPU 温度超出安全上限则烤机自动结束。如图3-16所示。

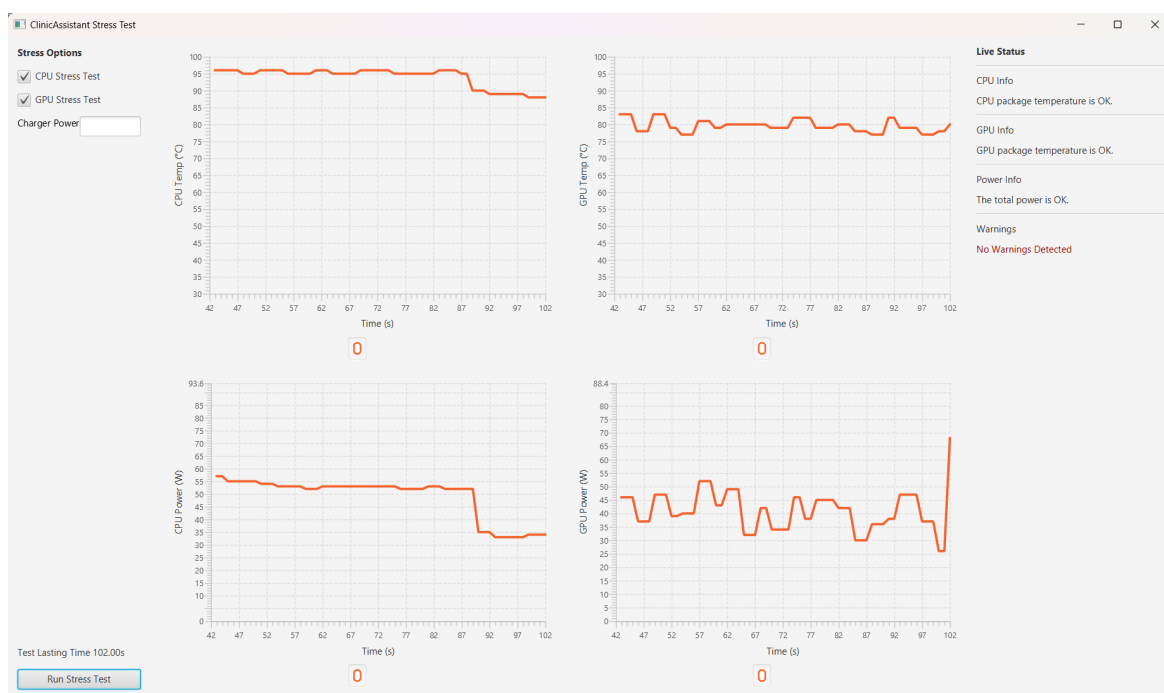


图 3-16 自动烤机测试工具

3.4 UI 界面与交互设计

本项目从诊所工作的实际情况出发进行设计，目标在于使得常用参数指标易于观察、常用操作可被“一键”执行，从而方便诊所的日常工作。因此，我将软件设计成了“监视器”、“工具”和“外部工具”三部分，使用左侧栏进行切换。如图3-5、3-6和3-7所示。

为了这个目标，整个软件都遵循着简单清晰的设计原则：仅使用按钮、文本和图表来设计交互、展示信息，以牺牲部分可定制性和更加详细的信息，换来更快速易于上手的交互逻辑。如 BitLocker 解锁工具（见图3-13）和自动烤机工具（见图3-16）就是这种设计的典型例子：BitLocker 解锁工具只设计了解锁而无加锁、自动烤机工具只设计了自动判断而无定时功能，就是因为这些需求对诊所工作而言相当罕见，因

此无需进行额外设计。

本项目支持国际化，图形化界面的默认语言是英语。这是为了避免系统编码配置有误导导致的界面乱码，这常见于使用英文作为显示语言或开启了全局 UTF8 编码实验功能的系统。

第 4 章 程序架构设计与技术实现方案