F-S-03-01：前端显示“WCS 通讯超时”

典型现象

- WMS 或 WCS 前端界面弹出红色提示：“WCS 通讯超时”。
- 任务无法下发，设备状态不更新。
- 刷新页面后可能短暂恢复，但很快再次出现。

可能原因

1. **网络连接问题**（占比约 40%）：WMS/WCS 服务器与 WCS 服务之间的网络中断、防火墙拦截、交换机端口故障。
2. **WCS 服务异常**（占比约 35%）：WCS 服务进程停止、假死或 CPU 占用过高无法响应请求。
3. **超时阈值设置过小**（占比约 15%）：前端或网关配置的超时时间（如 5 秒）小于实际业务处理时间。
4. **数据库或下游接口慢**（占比约 10%）：WCS 在处理请求时需要查询大量数据或调用外部接口，导致响应时间超时。

排查思路

1. **定位故障范围**：区分是所有客户端均报错，还是仅个别终端异常。若为个别终端，优先排查客户端网络；若为全局异常，聚焦服务器端。
2. **验证服务连通性**：从 WMS 服务器 ping WCS 服务器 IP，或通过 curl 调用 WCS 健康检查接口（如 /health ），确认网络与服务可达。
3. **核查 WCS 服务状态**：登录 WCS 服务器，通过 systemctl status wcs 等命令检查进程是否存活，同时监控 CPU、内存资源使用率。
4. **分析后端日志**：在 WCS 日志中搜索 timeout 或 slow 关键词，定位耗时过长的请求，针对性优化慢查询或业务逻辑。
5. **调整超时配置**：若业务实际处理耗时（如 10 秒）大于当前超时阈值（如5 秒），适当调大前端或网关的超时设置。

保养提示

- 部署 WCS 服务监控（如 Prometheus + Grafana），实时监控服务存活状态、接口响应时间，异常时自动告警。
- 定期巡检网络设备（交换机、防火墙）日志，及时处理丢包、拒绝连接等异常记录。
- 对 WCS 关键接口设置异步处理模式，避免长时间阻塞主流程，减少超时误报。
- 针对 WCS 数据库查询、下游接口调用，设置合理的超时保护，防止请求长时间挂起。

F-S-03-02：页面加载缓慢或白屏

典型现象

- 打开 WMS/WCS 页面时，长时间显示加载中（转圈）或最终白屏。
- 切换菜单或刷新数据时等待超过 10 秒。
- 其他网页访问正常，仅仓储系统慢。

可能原因

1. **后端接口响应慢**（占比约 50%）：数据库慢查询、大量数据返回、外部接口调用延迟。
2. **静态资源 CDN 故障**（占比约 20%）：前端 JS/CSS 文件托管在 CDN 上，CDN 节点异常导致资源加载失败。
3. **网络带宽不足**（占比约 15%）：办公网络与服务器之间带宽拥塞，或 Wi-Fi 信号差。
4. **浏览器内存泄漏**（占比约 10%）：长期不关闭页面导致浏览器占用内存过高，响应变慢。
5. **服务端资源耗尽**（占比约 5%）：Web 服务器（如 Tomcat、 Nginx）连接池或线程池满。

排查思路

1. **前端请求耗时定位**：

   打开浏览器开发者工具（F12）→「网络」标签：

   - 按 “Time” 列排序，找出耗时最长的请求。
   - 检查是否有请求返回 500/404 等错误状态码。

2. **数据库慢查询优化**：在数据库端开启慢查询日志，定位执行时间超过1 秒的 SQL，分析执行计划并优化索引。
3. **静态资源可用性验证**：在浏览器中直接访问报错的 JS/CSS 资源链接，确认能否正常下载；若失败，更换 CDN节点或回源至内网服务器。
4. **网络延迟与丢包排查**：从客户端 ping 服务器 IP，观察时延与丢包率；若时延 > 100ms 或丢包率 > 1%，需排查网络设备。
5. **服务器资源状态检查**：查看 Web 服务器的连接数、线程池使用率，若接近上限，考虑扩容或优化并发控制逻辑。

保养提示

- 定期分析慢查询日志，每周至少优化一条耗时最长的 SQL 语句。
- 将静态资源部署至内网服务器或使用可靠的 CDN 服务，并配置合理的缓存策略。
- 提醒用户每日下班前关闭浏览器标签页，避免长时间运行导致内存累积。
- 为 Web 服务器设置合理的连接池 / 线程池上限，配置告警阈值，避免资源耗尽。

F-S-03-03：点击按钮无反应

典型现象

- 点击 WMS/WCS 界面的按钮（如“确认”、“查询”、“导出”）后，没有任何响应，按钮不变灰，也没有提示。
- 浏览器控制台（F12）报 JavaScript 错误。

可能原因

1. **JavaScript 报错**（占比约 50%）：前端代码异常（如未定义的变量、语法错误）导致事件绑定失败或函数中断。
2. **权限不足**（占比约 25%）：用户账号没有操作该按钮的权限，但前端未弹出提示（或未正确处理权限状态）。
3. **请求被拦截**（占比约 15%）：浏览器插件（如广告拦截）、 CORS 策略或防火墙规则阻止了请求发出。
4. **Session 过期**（占比约 10%）：用户登录状态已过期，但前端未自动跳转到登录页，点击按钮时静默失败。

排查思路

1. **前端错误定位**：

   打开浏览器开发者工具（F12）→「控制台」标签：

   - 查看是否存在红色报错信息，根据错误堆栈定位代码问题。
   - 重点关注 xxx is not defined 、 Cannot read property of undefined 等常见错误。
2. **验证请求是否发出**：切换至「网络」标签，点击按钮后观察是否有新请求生成；若无请求发出，说明前端未触发调用；若有请求，检查状态码与返回结果。
3. **校验用户权限**：使用管理员账号测试同一按钮，若管理员可正常操作，则为权限配置问题，需检查角色权限配置表。
4. **排查浏览器环境干扰**：临时禁用所有浏览器插件，或使用无痕模式重试，验证是否为插件拦截导致。
5. **检查会话有效性**：在控制台执行 console.log(sessionStorage)或 localStorage ，检查存储的 Token 及登录信息是否过期。

保养提示

- 在前端代码中增加全局错误捕获（ window.onerror ），将 JS 错误上报至后端日志系统，便于问题追溯。
- 对无权限操作，前端应弹出明确提示（如 “您没有权限执行此操作”），避免静默失败影响用户判断。
- 定期清理无效 Session 缓存，设置合理的会话超时时间（如 8 小时），并在会话过期时引导用户重新登录。
- 开发环境中开启严格模式与代码校验，避免因语法错误导致按钮功能失效。

F-S-03-04：报表导出失败或乱码

典型现象

- 点击“导出 Excel/PDF”按钮后，页面无反应或提示“导出失败”。
- 导出文件打开后出现乱码（如“?????”）或空白内容。
- 小数据量可导出，大数据量导出超时或失败。

可能原因

1. **数据量过大导致超时**（占比约 45%）：导出几十万条记录时，后端生成文件时间超过网关或浏览器的超时设置。
2. **字符集问题**（占比约 25%）：数据中包含中文、特殊符号，导出时未使用 UTF-8 编码，或 Excel 打开时用错编码。
3. **临时目录无写权限**（占比约 15%）：后端生成临时文件时，服务器磁盘权限不足或磁盘空间满。
4. **浏览器安全设置**（占比约 10%）：浏览器阻止了弹窗或文件下载（如弹出窗口被拦截）。
5. **内存溢出**（占比约 5%）：后端一次性将所有数据加载到内存，导致内存溢出（OOM）。

排查思路

1. **定位问题范围**：筛选少量数据（如 10 条）尝试导出，若成功则问题与数据量相关；若仍失败，优先排查权
   限、编码或前端拦截问题。
2. **分析后端日志**：搜索日志中 export 、 OutOfMemoryError 、 timeout等关键词，确认具体错误类型。
3. **检查服务器磁盘状态**：使用 df -h命令查看磁盘空间，确认临时目录（如 /tmp ）是否已满，清理过期无用文件。
4. **验证字符集兼容性**：用记事本打开导出的 CSV 文件，另存为 UTF-8 格式后再用 Excel 打开；若显示正常，需将导出编码设置为 UTF-8-BOM。
5. **优化大数据导出策略**：大数据量导出改为异步方式 —— 用户提交导出任务后，系统后台生成文件，完成后文件映射提供下载。

保养提示

- 设置导出最大行数限制（如最多 10 万条），超出限制时提示用户分批导出。
- 定期清理临时目录（如每天凌晨删除超过 24 小时的临时文件），避免磁盘空间耗尽。
- 在导出接口中加入内存监控，当 JVM 内存剩余低于 20% 时拒绝新的导出请求，防止 OOM 故障。
- 导出文件时统一采用 UTF-8-BOM 编码，确保 Excel 打开中文无乱码。

F-S-03-05：登录失败或跳转循环

典型现象

- 输入正确用户名密码后，页面提示“登录失败”或“认证错误”。
- 登录成功后立即又跳回登录页，形成无限循环。
- 其他用户可以正常登录，仅个别账号异常。

可能原因

1. **认证服务异常**（占比约 40%）：后台认证服务（如 LDAP、OAuth、SSO）宕机或网络不通，无法验证用户信息。
2. **Session/Cookie 配置错误**（占比约 30%）：浏览器禁用了 Cookie，或 Session 的域名、路径配置不正确，导致登录后无法维持会话。
3. **用户状态异常**（占比约 15%）：账号被锁定、密码过期、未授权访问该应用。
4. **重定向 URL 错误**（占比约 10%）：登录成功后应该跳转的首页地址配置错误，或端口、上下文路径不对。

排查思路

1. **排除浏览器环境干扰**：使用浏览器无痕模式测试登录，若无痕模式正常，则清除原浏览器缓存与 Cookie，排查插件或配置问题。
2. **验证认证服务连通性**：从应用服务器 telnet 认证服务端口（如 LDAP 389、OAuth 443），若无法连通，检查防火墙策略或认证服务状态。
3. **分析后端登录日志**：按用户名检索日志，查看是否有 “密码错误”“账号锁定”“会话创建失败” 等明确错误记录。
4. **检查 Cookie 与会话配置**：在浏览器开发者工具「Application」→「Cookies」中，确认登录后是否写入了有效的 Session ID，且域名、路径配置正确。
5. **验证重定向地址有效性**：手动输入登录后的首页 URL（如 /home ）直接访问，若无法打开，排查权限配置或路由守卫规则。

保养提示

- 为认证服务部署主备或集群架构，避免单点故障导致全系统无法登录。
- 在登录页面提供明确的错误提示（如 “密码错误”“账号已锁定，请联系管理员”），而非笼统的 “登录失败”，便于用户定位问题。
- 定期清理过期账号、测试账号与僵尸账号，避免占用系统资源或引发安全风险。

F-S-03-06：前端提示任务异常

典型现象

- WMS/WCS 界面弹出提示框："任务异常: WCS通讯超时"、"任务异常: 库存不足"、"任务异常: 货位被占用"等。
- 异常信息明确指向具体原因，但用户不知道如何处理。

可能原因

1. **宽海 WMS 内置异常捕获**：系统已经自动识别异常类型并展示给用户，这是产品的正常设计。
2. **业务层面的问题**：例如通讯超时（网络问题）、库存不足（锁库或数量不够）、货位被占用（已有货物未出库）。
3. **程序未能自动恢复**：某些异常需要人工干预，如手动释放锁、强制完成任务。

排查思路

1. **优先按提示定位问题域**：WMS 前端会明确标注异常原因，直接根据提示定向排查：
    - 提示 "WCS 通讯超时" → 参考 S-03-01 文档排查通讯链路问题。
    - 提示 "库存不足" → 参考 S-02-03 文档核查库存锁定与分配规则。
    - 提示 "货位被占用" → 进入货位管理界面，查看该货位的锁定任务与在库状态。
2. **查看任务详情中的异常字段**：在 WMS 后台 "任务管理" 模块，找到对应任务，查看 "异常字段" 列的详细信息，该字段会记录错误代码、异常时间及问题详情。
3. **追溯接口请求日志**：按任务 ID 筛选相关 API 接口日志，查看请求与响应的完整内容，获取更详细的错误堆栈或业务报错信息。
4. **人工处理后重试任务**：根据异常类型完成对应人工操作（如释放锁定货位、补充库存、重启通讯服务），处理完成后点击 "重试" 按钮，重新执行任务。

保养提示

- 开展现场操作员专项培训，让用户能根据前端提示自主处理常见异常（如通讯超时检查网络、库存不足联系补货），减少工程师介入频次。
- 针对高频异常场景（如货位被占用），可在 WMS 中增加 "强制解锁" 等快捷操作入口，所有人工干预操作需强制记录操作日志并关联审批流程。
- 定期汇总系统异常数据，统计高频异常类型与触发场景，针对性优化业务规则或系统逻辑，从源头减少异常发生。

F-S-03-07：页面数据显示不全

典型现象

- 列表页面只显示了部分数据，滚动到底部后没有加载更多。
- 搜索或筛选后，应该返回 50 条结果，只显示了20 条。
- 详情页面某些字段为空，但数据库中有值。

可能原因

1. **分页参数错误**（占比约 45%）：前端传递的 pageSize 或 pageNum 不正确，或者后端接口默认限制了最大返回条数（如最多 100 条）。
2. **接口返回数据截断**（占比约 25%）：后端返回的 JSON 数据量过大，中间件（如 Nginx、网关）对响应体进行了截断或限流。
3. **权限过滤**（占比约 15%）：用户没有权限查看某些数据，后端静默过滤了敏感字段，但前端未做提示。
4. **前端渲染错误**（占比约 10%）：JavaScript 渲染时遇到无效数据（如 null、undefined）导致提前退出循环，后续数据未渲染。
5. **数据库查询 LIMIT 隐式限制**（占比约 5%）：SQL 语句中使用了 LIMIT 10 但业务需要更多。

排查思路

1. **核对总数与返回数**：在界面查看"总记录数"（如"共 50 条"），再数一下实际显示的条数。若实际少于总数，可能是分页问题。
2. **打开浏览器开发者工具**：查看网络请求中的接口返回体。比较返回数据的 total 字段和 rows 数组长度是否匹配。
    - 若 rows 长度小于预期，查看请求参数中 pageSize 是否正确。
    - 若返回体完整但前端显示不全，检查前端渲染逻辑。
3. **检查网关/代理配置**：如果使用 Nginx，查看 proxy_buffer_size 和 client_max_body_size 是否过小。适当增大缓冲区。
4. **验证用户权限**：用管理员账号查询相同条件，看是否显示完整。若管理员正常，则为数据权限问题。
5. **查看后端 SQL 日志**：确认 SQL 是否包含了 LIMIT 子句且值小于需要的数量。

保养提示

- 前后端约定分页规范：统一使用 pageNum 和 pageSize，后端默认 pageSize 最大不超过 500。
- 对于敏感字段的权限过滤，应在前端显示"无权限查看"占位符，而不是直接留空。
- 定期审查 API 接口的响应大小，对超过 1MB 的请求考虑分页或压缩传输。