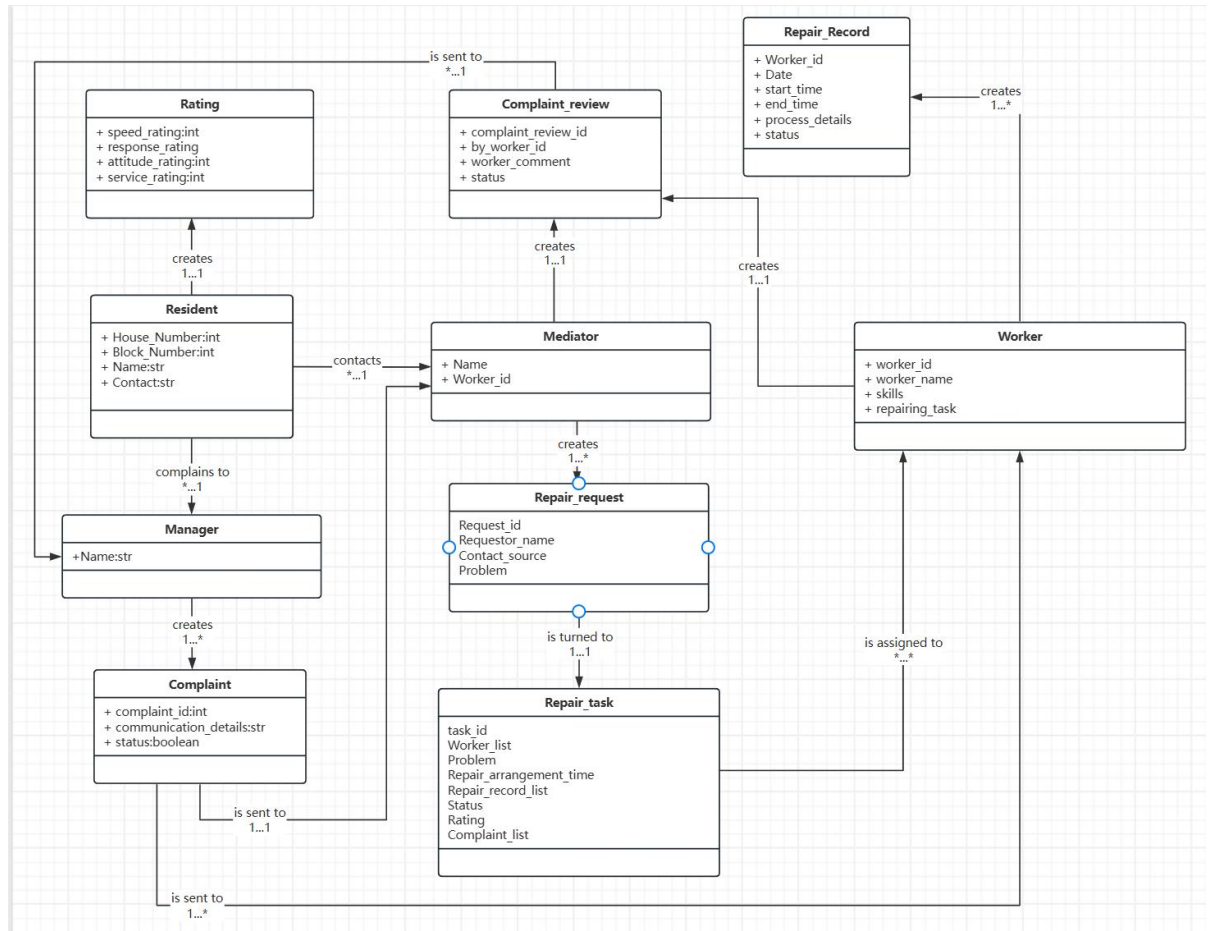
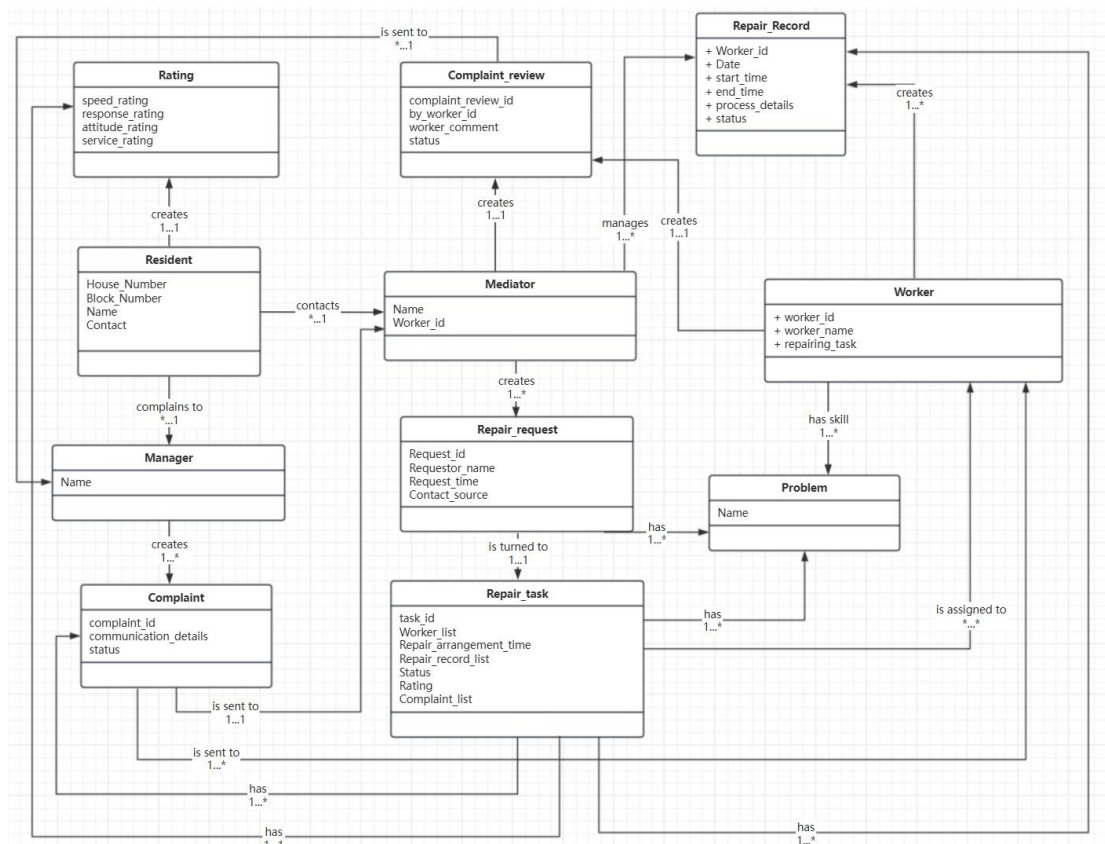


Lab2

根据 md 要求，我选择了先不使用大模型，手工构造一个版本，然后使用大模型进行对比验证的方式。上图为未使用大模型前所做出的领域图。



接着，把 md 里面的要求拿去询问大模型（本次选择了使用免费版 chatgpt 模型），在其回答上进行了如下的补充和改进：



和原本手工做出的图相比，具体补充了 **Repair_task** 和 **complaint**、**rating** 之间的关联，以及 **repair_record** 和 **mediator** 之间进行关联，方便其进行工人的管理（平时做系统时候会习惯性地知道需要关联起来，但画图时候遗漏了），以及把 **problem** 拉出来单独当类，这样可以减少数据库的重复记录（因为 **worker** 的维修能力本质上就是会需要进行报修的 **problem**）。

从上图可见，上图满足了下方基础流程：

业主 **Resident** 通过报修电话或者微信群报修，描述故障信息——>调度员 **Mediator** 将报修信息录入系统，作为待调度的任务 **Repair_request**（包括报修时间、故障内容、报修人、报修来源（比如微信、电话））——>调度员在合适的时间尽快对报修进行任务调度 **Repair_task**，将任务分配给合适的维修工人 **Worker**——>维修工人收到任务后会去现场维修，并在系统中留下维修记录 **Repair_record**（包括完成时间，处理过程）——>任务完成后，报修人给出评价 **Rating**（内容包括响应及时度、服务态度评分、对结果的满意度）

用户通过系统能够

1. 调度员可通过 **repair_task** 中的 **status** 和 **worker** 的 **repairing_task** 了解报修的状态，包括解调度的过程以及维修工的活动记录
2. 维修工人可了解当前分配给他的各种任务已经完成情况，包括调度给他的报修 **repairing_task** 和**需要他处理的投诉****complaint**。
3. 对于处理中的报修，通过 **repair_task** 的 **status** 知道当前活动的调度。
4. 可通过 **worker** 的 **repairing_task** 知道一个维修工人当前是否空闲，以便合理调度
5. **mediator** 可通过一个 **repair_task** 的所有 **repair_records** 统计处理一次报修已经用掉的工时
6. **mediator** 可通过一个 **repair_task** 的一个 **worker** 所生产的所有 **repair_records** 统计维修工人在一个时间段内的工作时间

拓展流程方面，上图满足了下方部分：

1. 多次调度

有时候派去的维修工考察了现场后，发现他的工种无法处理该故障，比如用户报告的故障是停电，但实际的原因是管道漏水。这时候需要完成当前的调度，重新调度，派另外的维修工人。系统中只能有最多一个活动的调度。

Mediator 可以通过对 **Repair_task** 的员工名单 **Worker_list** 的增删查改去操作这个部分，并在修改完成后通知相关被更动的工作人员。

2. 需要多次执行的活动

有些复杂的故障，维修工人一次无法完成，需要分多次，甚至在几天内完成。维修工人每次开始一次活动都需要单独记录本次处理过程的开始时间，结束后需要记录结束时间以及处理的内容，便于计算工时，统计工作量。

Worker 可以通过为一个 **Repair_task** 创建多个 **Repair_record** 来实现这部分。计算工时部分则由 **mediator** 进行。

3. 故障类型与维修工人工种的匹配

报修人报修后，调度员需要根据报修内容确定故障的类别，而每个维修工都登记了他能处理的故障分类，一个维修工可以处理一个或多个故障类别。这样，在调度时，调度员在选择了故障类别后，系统可以推荐相关的维修工人。

此部分可由具有对 **worker** 的详情（维修能力）进行查找功能的 **mediator** 来完成调度。

4. 投诉

用户如果不满意报修的处理，可以随时对此发起投诉（可以多次），投诉的内容为一段描述投诉内容的文字。每个到当前为止参与了该报修处理的维修工人和该报修的调度员可以收到该投诉并需要提交情况说明，物业经理在所有相关维修工人都提交了情况说明后，进行投诉处理，与客户沟通并将情况记录下来，关闭该投诉。

此部分的流程为 **Resident** 进行电话或微信投诉于物业经理 **Manager**，**Manager** 负责将电话或微信内容与对应的 **Repair_task** 进行关联，生成一个或多个 **Complaint** 并将其发给调度员 **Mediator** 和相关的 **Worker**。**Mediator** 和 **Worker** 需要对收到的 **Complaint** 生成 **Complaint_review** 进行情况说明，并将其发送回给 **Manager**。**Manager** 可以通过更改 **Complaint** 中的 **status** 来关闭投诉。