**考勤系统生成考勤结果 详细设计**

**功能**

考勤系统要能统计员工预先排好的班次的上下班时间，并要能统计员工各类请假，各种加班，以及出差，短时间内外出等的各种时间。

**限制条件**

本设计不支持没有预先设置班次，而通过系统智能抓取上下班的这种情况，这种情况将通过另外的方法实现。

**功能实现**

**一．首先将员工一段时间内的的排班生成一个时间段列表，**

比如某个员工8月1日到8月5日上白班，

那么第一次处理生成时间段列表就有5个

2013-8-1 9：00 --- 2013-8-1 18：00

2013-8-2 9：00 --- 2013-8-2 18：00

2013-8-3 9：00 --- 2013-8-3 18：00

2013-8-4 9：00 --- 2013-8-4 18：00

2013-8-5 9：00 --- 2013-8-5 18：00

**二． 然后从所有的这些时间段中切出员工请假和外出的时间，**

比如8月2日员工申请 10：00—14：00 外出

那么 8月2日那个时间段就会被切成两段，

2013-8-2 9：00 ---2013-8-2 10：00 和 2013-8-2 14：00 ---2013-8-2 18：00

如果员工在10点之前打卡，那么有 的公司会规定这样算早退，同样，14：00之后打卡算迟到。这种情况下，员工要多打两次卡。

还有一种情况，如果员工在8月4日请假，那么时间段列中就会删除

2013-8-4 9：00 --- 2013-8-4 18：00

这里有几点注意：

所有的请假和外出时间都是按实际单上的时间，比如 员工申请外出 10：-14：00，但实际外出打卡是在11：00,，14：00回来，这种情况下考勤结果上外出时间也是4个小时，而不是实际外出三个小时。

**三．加班时间段的合并，**

如果员工申请 8月1日 19：00 ---22：00 加班

那么时间段列表中就会增加一个时间段

2013-8-1 19：00 --- 2013-8-1 22：00

系统在这里还可以做到加班时间段与正常上班时间段合并，如

申请 8月3日 18：00--- 22：00加班，

那么时间段2013-8-3 9：00 --- 2013-8-3 18：00 就会被修改成

2013-8-3 9：00 --- 2013-8-3 22：00 同时时间段的允许加班属性被设置成真。

这里有几点注意：

1. 加班时间不能与正常上班的时间重叠，如果有重叠，则会从加班时间部分移除重叠部分。
2. 如果加班单指定了加班时间，那么实际加班时间就要根据打卡时间来确定，但最大为申请单的加班时间，如果在加班时间段内提前下班，那么不计为早退，但加班时间相应减去提前下班的时间。同样，如果加班时迟到了，也不计为迟到，只会相应扣除上班时间。

四．**将所有的要上班的时间段设置好后，系统分析员工的打卡记录，确定每个时间段的实际上下班时间。**

分析上下班时间用到如下几个准则：

1. 上班时间，尽量取与上班最早有效的打卡时间接近的打卡记录时间
2. 如果前一个时间段的下班时间和下一个时间段的上班时间中间只有一个刷卡记录，

那么这个唯一的刷卡记录用来表示前一个时间段的下班时间。

1. 上班时间和下班时间间隔不能小于1分钟，比如某一天只有9：00：11 和9：00：55两次打卡，则上班时间为9:00:11,但后面的那个刷卡时间不算成下班时间，这种情况会记录成缺勤而不是早退。
2. 下班时间，尽量取离员工下班后最大延长刷卡时间点最近的刷卡时间。

**五．对所有的时间段都附加了上下班时间后，软件最后一步是根据实际上下班时间来判断员工的考勤情况，比如迟到，早退，缺勤，正常，加班时间等。**

附：

考勤时间段结构体：

public class ShiftTimezone

{

public int StaffID { get; set; } //人员编号

public DateTime ShiftDate { get; set; } //班次日期

public string ShiftID { get; set; } //班次ID

public string ShiftName { get; set; } //班次名称

public DateTime StartTime { get; set; } //开始时间

public int AllowLate { get; set; }//允许迟到分钟数

public int BeforeStartTime { get; set; }//允许的最大提前刷卡分钟数 public DateTime EndTime { get; set; } //结束时间

public int AllowEarly { get; set; } //允许的早退分钟数

public int AfterEndTime { get; set; } //最大的延长刷卡分钟

public bool AllowOT { get; set; } //是否允许加班

public int MinOTTime { get; set; } //最小加班分钟数

public int MaxOTTime { get; set; } //最大加班分钟数

public DateTime? ArriveTime { get; set; } //实际上班打卡时间

public DateTime? LeaveTime { get; set; } //实际下班打卡时间

}