

4-30 实现考勤成功页面（Web层）

140.143.132.225:8000/project-1/doc-78

一、查询用户的入职日期

因为在Emos系统中，当天没有考勤的员工就算做缺勤。有一种特殊请款我们必须引起注意，假设A员工周三入职的，从周三开始执行人脸考勤。那么在考勤成功页面，我们看到周一和周二的考勤结果都是缺勤，这个是不对的。所以我们要先查询出来员工的入职日期，查询本周考勤的时候，应该从员工入职日期开始算起。

编写 `TbUserDao.xml` 文件，查询员工的入职日期。

```
1. <select id="searchUserHiredate" parameterType="int" resultType="String">
2.     SELECT hiredate FROM tb_user WHERE id=#{userId} AND status = 1
3. </select>
```

编写 `TbUserDao.java` 接口，定义抽象方法。

```
1. public interface TbUserDao {
2.     .....
3.     public String searchUserHiredate(int userId);
4. }
```

在 `UserService.java` 接口中，声明抽象方法。

```
1. public interface UserService {
2.     .....
3.     public String searchUserHiredate(int userId);
4. }
```

在 `UserServiceImpl.java` 中，实现抽象方法。

```
1. public class UserServiceImpl implements UserService {
2.     .....
3.     @Override
4.     public String searchUserHiredate(int userId) {
5.         String hiredate = userDao.searchUserHiredate(userId);
6.         return hiredate;
7.     }
8. }
```

二、实现查询考勤结果的Web方法

编写 `CheckinController.java` 中的Web方法，查询用户签到的结果。

```
1. public class CheckinController {
2.     .....
3.     @Autowired
4.     private UserService userService;
5.     @Autowired
6.     private SystemConstants constants;
7.     @GetMapping("/searchTodayCheckin")
8.     @ApiOperation("查询用户当日签到数据")
9.     public R searchTodayCheckin(@RequestHeader("token") String token) {
10.         int userId = jwtUtil.getUserId(token);
11.         HashMap map = checkinService.searchTodayCheckin(userId);
12.         map.put("attendanceTime", constants.attendanceTime);
13.         map.put("closingTime", constants.closingTime);
14.         long days = checkinService.searchCheckinDays(userId);
15.         map.put("checkinDays", days);
16.         //判断日期是否在用户入职之前
17.         DateTime hiredate = DateUtil.parse(userService.searchUserHiredate(userId));
18.         DateTime startDate = DateUtil.beginOfWeek(DateUtil.date());
19.         if (startDate.isBefore(hiredate)) {
20.             startDate = hiredate;
21.         }
22.         DateTime endDate = DateUtil.endOfWeek(DateUtil.date());
23.         HashMap param = new HashMap();
24.         param.put("startDate", startDate.toString());
25.         param.put("endDate", endDate.toString());
26.         param.put("userId", userId);
27.         ArrayList<HashMap> list = checkinService.searchWeekCheckin(param);
28.         map.put("weekCheckin", list);
29.         return R.ok().put("result", map);
30.     }
31. }
```

