

微信小程序赋予图书馆内部 办公自动化的新可能性及实践重点

——以厦门大学为例

陈俊杰 吴明杰 张晓静 李 剑

【摘要】在对微信小程序“核心能力”分析的基础上,以现实需求为导向,立体对比了三种主要技术路线,阐述了微信小程序在助推图书馆内部办公从“自动化”向“移动化”全面演进中的无限潜力。通过厦门大学具体实践中的几个应用场景,真实展现小程序在内部管理效率提升方面的优势,不仅是抛砖引玉,更是为了探讨实践中的开发思路和技术难点。

【关键词】微信小程序 OA 移动化

Abstract: On the basis of the analysis of “core feature” of WeChat mini program together with the guiding of actual demand, the article compares the pros and cons of three main technical routes, and discusses the infinite potentiality of WeChat mini program in boosting the library OA from automation to mobility. By means of several application scenarios of Xiamen University, the advantages of WeChat mini program in improving internal management efficiency are fully demonstrated. This article offers not only a good starting point, but also discusses the technical challenges in actual development.

Key words: WeChat mini program OA mobility

DOI:10.15941/j.cnki.issn1001-0424.2018.04.005

0 引言

微信小程序无需安装、用完即走的构建思路是“移动化理念”的极致诠释;它内嵌于微信,拥有移动端完整硬件能力;在开发方面,又因为近似 Web 开发的技术栈而无需过高的成本。这些特点恰恰能解决目前图书馆内部办公自动化系统“移动化”发展乏力的痛点。前雅虎用户体验总监 Luke Wroblewski 在其作品《Mobile First》^[1]中首次提出“移动优先”的设计理念,正式宣告互联网移动时代全面到来。“移动优先”的核心价值在于其便利性和对用户的无缝触达。移动化的春风同样也吹向了一向对新技术敏感的图书馆界。笔者在《移动图书馆 APP 的构建策略和关键技术——以 iOS 为例》^[2]中提及的厦门大学 iOS 移动图书馆客户端就是国内图书馆行业中发布的第一个移动应用。在此之后其他高校、公共图书馆也陆续发布了自己的移动 APP,但声量明显越来越小。此外,图书馆对于移动技术的应用也大多局限于将 OPAC 的主要功能迁移到移动端以及部分馆藏移动阅读的实现。对于“内部办公自动化系统(下称 OA)”的移动化却鲜有研究者涉及。究其根本原因,在于移动应用开发、维护的门槛和成本过高。而如前所述,微信小程序(下简称小程序)的推出,无疑为图书馆 OA 的移动化带来了新的可能性。

1 文献综述

1.1 图书馆 OA 相关研究

国内部分以中国知网的学术期刊数据库作为文献来源,分别以不限时间和 2012~2017 年为时间限定,以“主题=图书馆 and 主题=办公自动化”为检索式进行检索,得到记录分别为 197 条和 56 条,剔除不符合主题的记录后共计 115 条和 56 条。限定核心期刊的记录分别为 10 条和 2 条,其中以阐述新技术的应用为主的有 13 篇,近 5 年只有 1 篇。国内图书馆的办公自动化始于 20 世纪 90 年代初,2000 年以后图书馆办公自动化建设和研究走向深入^[3]。近 5 年来学术界对图书馆 OA 关注较少,而已有的涉及技术路线方向的研究则集中于 JSP^[4]、ASP^[5]等相关

传统 Web 技术。

国外部分,在 ProQuest 中的 Library & Information Science Abstracts (LISA) 数据库,限定在同行评审的学术期刊中使用检索策略 ab (library) AND ab (office automation),共得到 41 篇文章(截至 2017 年 12 月 6 日)。其中与本文主题相关的研究主要集中于工作评估体系^{[6][7][8]},主要从成本效益的角度讨论内部管理系统在图书馆中的重要性,特别是在数据收集、具体服务应用等方面,而对于具体所采用的建设技术路线尚未有文章提及。

1.2 微信小程序相关研究

小程序推出时间不长,且主要用户群在国内,因此国外文献尚未有提及。而国内核心期刊中只有两篇文章,即《当图书馆遇上微信小程序》^[9]与《微信小程序在图书馆移动服务中的应用实践——以排架游戏为例》^[10],前者着重从宏观上分析小程序与图书馆服务模式的结合,偏向于小程序应用于图书馆“对外服务”的探讨,后者是以排架游戏为例的小程序的应用实例,方向上同样偏向“对外服务”。

1.3 总结

综上,现有相关研究思路是将小程序仅作为略优于传统 Web 开发的一个技术选项,将其应用于“对外服务”,而未对小程序本身的特点作深入分析、比对,将其优势特点与图书馆 OA 建设结合。鉴于此研究目前少有人涉及,本文将试着从理论模型到实践架构做一个阐述。

2 需求导向的技术选型

技术路线大多难以绝对化判断孰优孰劣,而只能通过对具体应用领域、场景的针对性分析来选择“最合适”的路线。本文以厦门大学图书馆(下称我馆)的一个内部管理需求为例,分析现实工作流程中的痛点,对比几种技术路线的优缺点。最终展示小程序所提供的核心功能在满足该需求上的优势,并辅以开发实践的技术细节,以期为图书馆 OA 建设的技术选型提供一个思路及实践的参考。

2.1 图书馆内部管理的现实需求

图书馆流通部门除了最基本的图书借还工作外仍有很多诸如委托、预约图书的查找、取消、接收、更改状态;期刊报纸的登到、上架;新书的接收、贴标转换处理;座位系统的巡查;检索机及自助文印机的维护管理;整理书架;消防门禁巡查等事务流程性的工作。据其特点,规范化显得十分必要,直接决定了工作能否按时按质保量地完成。我馆长期以来的工作方式是使用纸质表格列明任务、时限、效果等,由当班工作人员根据工作情况逐一进行手工记录及确认,这种方式其实存在诸多问题:

(1) 时限、位置无法确认。需严格要求特定时间点完成的任务难以核实,如循环定时的座位巡查、新到报纸及时上架、下班前检查消防门禁等。最常见的情况就是下班前一次性勾选表格中所有项目。

(2) 责任、效果难以匹配。难以核实当班人员是否真正完成既定工作,从表格中获取的信息只能是“某人是否签名”。

(3) 存档、统计成本较高。工作质量、工作量、完成情况难以统计汇总,用纸质表格勾选、签名的方式,平均每天 3 张的纸质表格,随着时间的推移,资料堆积如山,难以查找、统计。

2.2 技术选择(微信小程序)分析

应对以上问题,信息化手段的介入无疑是必要的。下文将从技术选型开始,逐步分析阐述小程序的特点、优势,尝试以小程序构建“Lib 小助手”为例,着重介绍几个应用实践场景。

目前,信息化的技术路线可以总结为传统 Web 技术、移动端 APP 和小程序三种。它们的优劣比较如表 1。

表 1 三种技术路线对比

技术路线	开发学习曲线	开发维护成本	移动端硬件能力调用	安装成本	使用便利性	移动端使用流畅度
传统 Web	平滑	低	弱(基于 HTML5)	低	差	低
移动端 APP	陡峭	高	强	高	好	高
小程序	平滑	低	高	低	极好	高

传统 Web 技术是最为常用的路线,发展相对较成熟,是大多数信息化工作的默认选择。其能够实现数据的记录、沉淀、统计。每个工作人员使用自己的账号登陆系统,在特定时间进行特定的操作,很好地解决了“身份认

证”、“时间监管”、“后续统计”等问题。而且在开发维护成本方面也是最低的。当前我馆的所有内部管理系统都采用该路线。

移动端 APP 则算是传统 Web 技术的进化版,服务器后端仍沿用 Web 技术。而负责用户交互的前端则从浏览器迁移到移动端的“原生应用”。正是其高度的易用性和不受空间限制的灵活性促成了“移动化浪潮”的到来。它能够以更好的用户体验(应用载入速度、操作顺畅度、响应速度等)来实现传统 Web 技术所能够实现的功能。在此基础上,移动终端的附加硬件能力,如 GPS、陀螺仪、摄像头等,能够更优地适配实际需求。例如闭馆前的消防门禁排查可以使用摄像头拍摄照片来核实消防通道是否正常,照片中提取的拍摄时间和位置则可以作为有效的确认信息。但是,该路线存在一个难以回避的高成本问题,针对 Android、iOS 两大主流平台的适配则更进一步推高开发成本。这也是该技术路线未在图书馆 OA 领域得到应用的最重要原因。

图书馆 OA 有两个主要特点:相对于 OPAC 等外部系统使用用户较少、系统功能点较多。用户较少使得系统能够获得平均收益就偏低,而功能点多又造成开发成本的上升。这些原因导致图书馆管理层无法、也没有动力采用这个路线,进而难以享受到“移动化浪潮”带来的便利。

小程序的推出恰好可作为图书馆 OA 移动化的助推器,其主要优势总结如下:

(1) 普及率高。微信庞大的装机量及使用率(8.89 亿的月活跃用户数)^[11],已使其成为一种生活方式。得益于此,依托于微信的小程序无需用户另行安装,具有移动端 APP 不可比拟的超低使用成本。

(2) 辨识度高。微信号本身已经成为了个人核心的身份标识,能够保证操作人员身份的真实性。通过微信号的绑定,用户无需记录繁多的系统账号,在小程序实现无缝自动登录使用系统功能。

(3) 硬件能力强。丰富的接口使其具备移动终端几乎所有硬件调用能力。

(4) 体验友好。小程序拥有媲美移动 APP 原生应用的用户体验,用户使用小程序的流畅度与原生应用并无二致,大大优于 Web 技术。同时其离线可用的特性也使得在网络信号差的区域使用成为可能。

(5) 使用路径短。最便捷的使用方式就是“扫码即用”。在自习室入口放置一个小程序二维码,工作人员巡查座位时走到该处,扫码即可使用巡座小程序。整个操作路径远比传统 Web 技术(打开浏览器、输入网址)、移动端 APP(找到 APP、打开、进入特定栏目)更短、更便捷。

(6) 触达率高。小程序能够在特定时间,给特定用户发送通知,以此就能实现在某个工作流程前提醒工作人员该做某事。而且,该通知是经由微信推送,因此被用户禁用的可能性远小于普通 APP,拥有更好的触达性。

(7) 成本较低。小程序所用技术栈接近 Web 技术(下一部分实践部分将深入介绍),具备 Web 开发技术背景的开发人员能够在很短的时间内上手开发,开发成本大大低于 APP 开发。同时,其内嵌于微信应用,因此无需另外处理跨平台的适配问题。

基于以上分析,小程序的核心优势可以归结为 3 点:用户体验良好,硬件能力强,开发成本较低。它的出现为图书馆 OA 开发带来了新的可能性,也是当前最优技术选项。此外,国内外具备类似小程序能力的技术路线其实也有其他厂商推出,如 Facebook 的 Bots^[12],支付宝的小程序^[13]。但由于网络、普及程度、接口限制等原因,综合评估还是“微信小程序”占优。

3 应用场景及实践重点

小程序框架提供了自己的视图层描述语言 WXML 和 WXSS,以及基于 JavaScript 的逻辑层框架,并在视图层与逻辑层间提供了数据传输和事件系统,可以让开发者方便地聚焦于数据与逻辑上^[14]。其中 WXML、WXSS 脱胎于 HTML 及 CSS,而逻辑层所使用的 JavaScript 则只是在原生 JavaScript 基础上增加了小部分小程序专属特性。因此,具备 Web 开发经验的工程师都能够迅速上手开发。

下文将配合实际应用场景,阐述小程序实际开发实践的几个重点。

3.1 服务端要求

小程序的整体架构类似于 APP,可归为 Client/Server 架构,其本身负责用户交互,基础的数据提供及复杂处理则是交由服务端来完成。同时,微信官方规定,服务端需使用更为安全的 HTTPS 进行数据传输。因此服务端的部署需申请 HTTPS key,否则无法通过腾讯官方的审核。

3.2 身份认证

目前,个人微信号已经相当于个人第二身份证,将其作为认证标识是整个系统的核心特性。但小程序内部无法获取用户微信号,只能另辟蹊径。每个用户使用任意小程序时都会由微信分配一个永久的 openid,不同用户在

同一个小程序内所获得的 openid 是不重复的,而且一经生成就与微信号永远绑定。基于此,openid 十分适合作为身份标识使用。将系统用户信息与小程序 openid 绑定后,用户每次使用小程序时就无需进行繁琐的登录操作。具体逻辑如图 1。

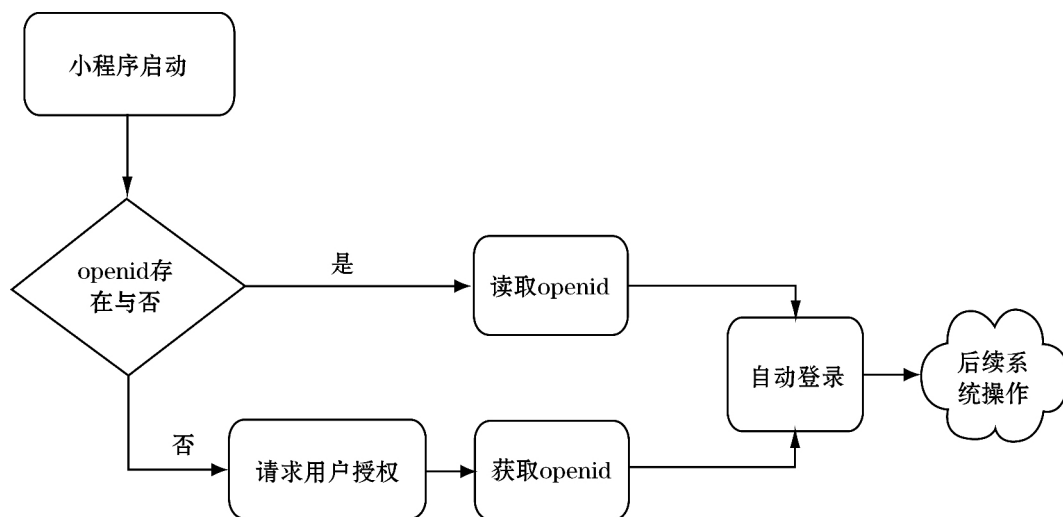


图 1 openid 获取流程图

针对 openid 的获取,官方提供了 wx.checkSession、wx.login、wx.getUserInfo 三个相关接口。需通过三个接口配合使用的多次往复才能够安全地获取用户 openid。首先通过 wx.checkSession 获取当前用户登录态,登录态有效则直接调用 wx.getUserInfo 请求用户信息,获取 encryptedData,进而使用与登录态关联的登录凭证 (code) 解密 encryptedData,获得用户详细个人信息。如果登录态无效,则需重新申请用户授权,以获取登录凭证。

最终解密后信息如下,其中 openid 即为我们所需,其他为附带信息。

```

{
  "openId": " OPENID" ,
  "nickName": " NICKNAME" ,
  "gender": GENDER ,
  "city": " CITY" ,
  "province": " PROVINCE" ,
  "country": " COUNTRY" ,
  "avatarUrl": " AVATARURL" ,
  "unionId": " UNIONID" ,
  "watermark":
  {
    "appid": " APPID" ,
    "timestamp": TIMESTAMP
  }
}

```

每次调用 wx.getUserInfo 会弹出授权请求窗口。因此,建议每次获取 openid 后保存到本地,后续使用直接读取存储内容即可。直到登录态过期再重新请求授权,避免重复频繁的向用户申请授权,影响用户体验。此外,由于获取用户信息是整个小程序使用的起点,如用户拒绝授权则无法使用,小程序内部需对此做相应的提示处理。

3.3 拍照功能

如前文分析,照片及照片拍摄时间是检验任务是否完成、何时完成的依据。小程序的 wx.chooseImage 能够调用手机拍照接口。在获取到照片之后还需将其发送服务端进行存储及分析。存储便于日后查询,而分析则主要是提取照片拍摄时间。精简后实现代码如下:

```

wx.chooseImage ( {

```

```

    count: 9, // 允许选择照片数量
    sizeType: ['compressed'], // 指定是原图还是压缩图, 默认二者都有。基于我们的需求, 我们选择压缩图以加快上传进度
    sourceType: ['album', 'camera'], // 指定来源是相册还是相机
    success: (res) => {
      that.selectedImages = that.selectedImages.concat(res.tempFilePaths)
      that.flag.uploadedImageCount = 0
      that.$apply()
      for (let i = 0; i < res.tempFilePaths.length; i++) {
        let imagePath = res.tempFilePaths[i]
        //调用上传文件 api 将图片传到服务器 wx.uploadFile ( {
        url: app.globalData.apiHost + 'jobs.json? async=1',
        filePath: imagePath,
        name: 'origin_file',
        success: (resp) => {
          //请求成功后回调
        },
        complete: (res) => {
          //请求失败后回调
        }
      })
    }
  })
}
})

```

3.4 地理位置信息获取

GPS 信息有助于了解任务执行的位置, 避免弄虚作假。位置相关的验证都可通获取 GPS 信息进行验证。简单调用方式如下:

```

wx.getLocation ( {
  type: 'wgs84',
  success: function (res) {
    var latitude = res.latitude
    var longitude = res.longitude
    //完成之后需将位置信息发送到服务器端进行保存
  }
})

```

地理位置信息属于个人隐私, 每个请求该接口都将弹出授权窗口, 请求用户授权。同样的, 小程序需要对用户拒绝授权的情况进行处理。

3.5 通知

小程序具备主动向特定用户发送通知的能力, 称为“模板消息”。通过模板消息接口能够在特定时间点通知特定用户进行某个操作的功能。但是该接口的调用也是有一定限制的, 它要求用户必须有表单请求后才能对其发送一条模板消息^[15]。因此, 在设计用户操作流程时需把这个因素考虑在内, 如果下一个步骤需要“通知”, 则当前的步骤应设法包装成表单请求。“Lib 小助手”中有一个“早上 10: 00 巡查座位”的小任务, 我们能够在 9: 55 分提醒当班工作人员。而在此之前的一个操作是“期刊登到”, 因此需将“期刊登到”的操作封装到 form 内, 实际设计如下:

期刊登到页面 wxml:

```

<form bindsubmit = " formSubmit" report - submit = " true" >
  <view class = " section section_ gap" >
    <view class = " section_ _ title" >switch </view >
    <switch name = " switch" />
  </view >
</form >

```

javascript 业务逻辑:

```

formSubmit: function ( e) {
//向服务器发送 post 请求
wepy. request ( {
  url: 'https: //xxxxx. com/xx',
  header: {
    'content - type': 'application/json',
    'sessionid': that. globalData. sessionid
  },
  method: 'POST',
  data: { formId: ' xxxxx' },
  success: ( resp) => {
  },
  fail: ( resp) => {
  }
})
}

```

以上代码有两个要点: 一是 form 组件必须将属性 report - submit 声明为 true, 表示将基于此表单发送模板消息。二是将相应的 formId 提交到服务器, 服务器端发送模板消息时需附带 formId, 以此表示模板消息的来源表单。

通过对“期刊登到”步骤的如此设计, 我们就获得了一个模板消息发送的“额度”, 以此“额度”提醒用户进行下一步的“巡查座位”任务。

3.6 精确的请求来源追踪

使用 GPS 可以获取操作具体位置, 但是由于 GPS 本身技术限制, 其定位精度只能达到 10 米的级别^[16]。“Lib 小助手”中座位巡查任务要求工作人员巡遍阅览室的各个角落。此类“精确位置审核”的需求需结合使用 GPS 及“带参数小程序二维码”^[17]两个接口来满足。“带参数小程序二维码”用以保证用户的操作来源于扫描特定位置的二维码。与此同时, GPS 信息的获取则可以识别用户是否通过扫描二维码照片进行“作弊”。

通过“获取二维码”的接口 B 可以批量生成带参数的小程序二维码, post 请求时附带 page 及 scene 参数。page 用于配置扫码后进入的小程序页面, 如系统中座位巡查对应的页面“pages/patrol”, scene 则用于标记来源的精确位置, 如 room_ 201_ south_ west, 表示 201 阅览室西南角。小程序内部接收请求时就可以在页面的 onLoad 方法中读取 options. scene 以获取该值并据此判断操作位置。

4 效果及展望

图 2~5, 以小程序实现的“Lib 小助手”, 彻底颠覆了原有管理方式无序、繁琐、表面、难以核实、难以统计的问题, 实现了图书馆日常工作管理流程的无纸化、移动化、实时化、图像化。对于工作人员, 一切工作由小程序精准地进行提醒、记录, 确保准时、完整。对于管理人员, 所有的日常工作任务、人员、时间、地点、效果等都一览无遗, 后期的数据统计也是水到渠成之事。小程序提供的一系列接口能力, 完美满足了图书馆内部管理工作的需求; 其移动化优势极大提升了办公效率; 其接近 Web 开发的技术栈特点大大降低了学习、开发成本。这一切都为图书馆内部管理系统的移动化带来了全新的、更优的、更具可执行性的方案。



图2 主要流程截图



图3 个人每日操作记录

5 结语

本文选择“Lib 小助手”作为范例，呈现了小程序的多个核心能力及与之相适应的应用场景。笔者希望借此

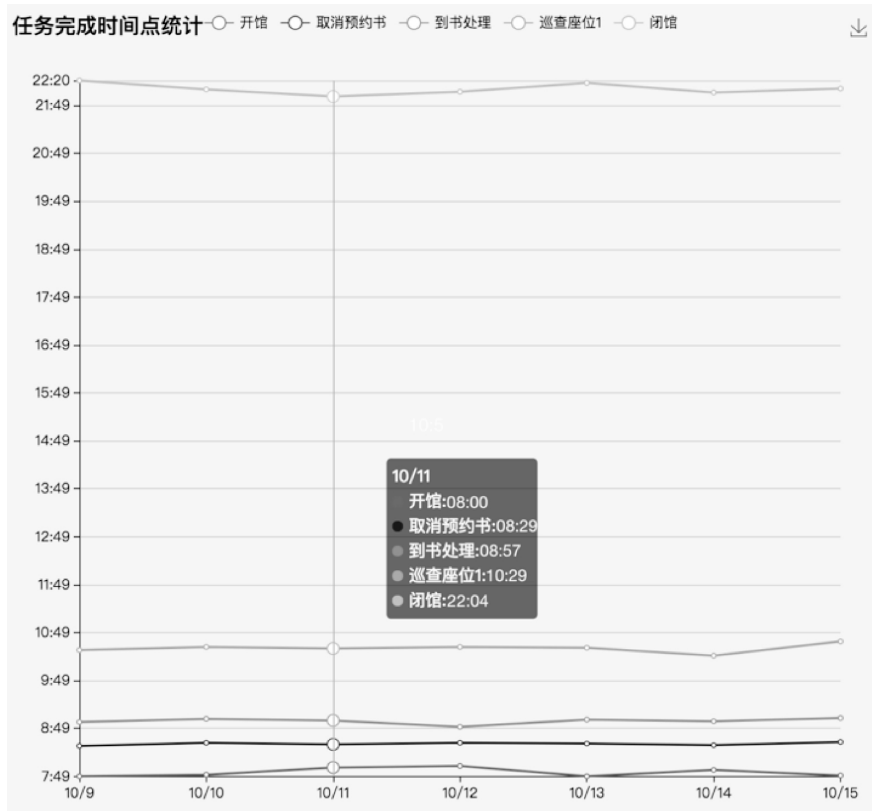


图4 管理后台任务完成时间点统计图

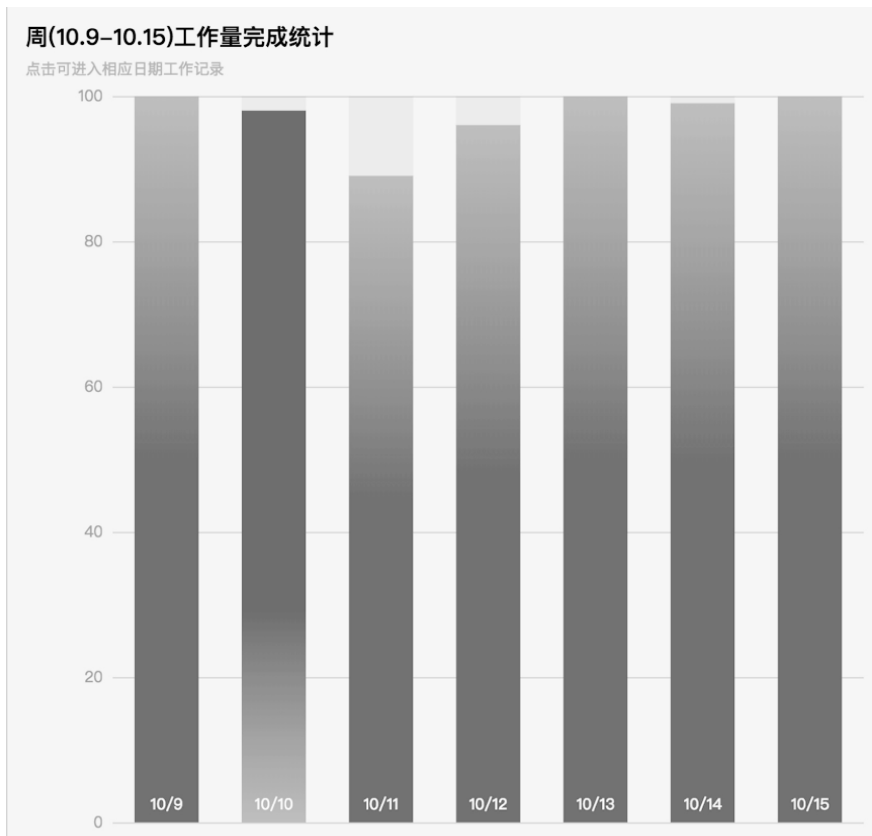


图5 每周工作量完成统计

为读者展示一些小程序能力利用的实践以抛砖引玉。限于篇幅,本文提及的应用场景还是相当有限,除上文提及的接口外,仍有大量接口可供各种线下场景中应用。如“功能模块分享到微信群”的接口可用于实现任务的指派、认领;微信身份认证用于办公物品认领等都是可以尝试的方向;GPS接口用于安保巡查等等,尽可放开思路,大胆尝试。笔者也将在其他内部管理需求中继续探索尝试更多应用场景,以期利用小程序这一强大、实惠的技术提升图书馆管理工作的效率。

注释

- [1] Luke Wroblewski. Mobile First [EB/OL]. [2017-12-06]. <https://abookapart.com/products/mobile-first>.
- [2] 陈俊杰, 黄国凡. 移动图书馆 APP 的构建策略和关键技术——以 iOS 为例[J]. 现代图书情报技术, 2012 (9): 75-80.
- [3] 徐淑娟等. 论 2000~2012 年图书馆办公自动化研究与发展[J]. 图书馆学刊, 2014 (2): 133-136.
- [4] 蒋春林, 苏林忠, 楼宏青. 用 JSP 搭建高校图书馆网络办公系统[J]. 图书馆论坛, 2006 (5): 1-3.
- [5] 薛涵, 张静芳, 赵宁. 融入知识管理的高校图书馆 OA 系统研究[J]. 情报科学, 2013 (2): 140-143.
- [6] Poll R. Performance, Processes and Costs: Managing Service Quality with the Balanced Scorecard [J]. Library Trends, 2001 (4): 709-717.
- [7] Phipps S, Franklin B, Sharma S. Striving for Excellence: Organizational Climate Matters [J]. Evidence Based Library and Information Practice, 2013 (2): 22-35.
- [8] Balague N, Duren P, Juntunen A, Saarti J. Quality Audits As a Tool for Quality Improvement in Selected European Higher Education Libraries [J]. The Journal of Academic Librarianship, 2014 (5): 529-533.
- [9] 王天泥. 当图书馆遇上微信小程序[J]. 图书与情报, 2016 (6): 83-86.
- [10] 朱玉强. 微信小程序在图书馆移动服务中的应用实践——以排架游戏为例[J]. 图书馆论坛, 2017 (7): 132-138.
- [11] 企鹅智库. 微信 2017 用户研究和商业机会洞察[EB/OL]. [2017-12-06]. <http://tech.qq.com/a/20170424/004233.htm>.
- [12] Facebook. Messenger Bots for Business & Developers [EB/OL]. [2017-12-06]. <https://messenger.fb.com/>.
- [13] 蚂蚁金服. 蚂蚁金服开放平台[EB/OL]. [2017-12-06]. <https://open.alipay.com/channel/miniIndex.htm>.
- [14] 腾讯. 小程序框架[EB/OL]. [2017-12-06]. <https://mp.weixin.qq.com/debug/wxadoc/dev/framework/MINA.html>.
- [15] 腾讯. 小程序模板消息[EB/OL]. [2017-12-06]. <https://mp.weixin.qq.com/debug/wxadoc/dev/api/notice.html#发送模板消息>.
- [16] 维基百科. GPS [EB/OL]. [2017-12-06]. <https://zh.wikipedia.org/wiki/%E5%85%A8%E7%90%83%E5%AE%9A%E4%BD%8D%E7%B3%BB%E7%BB%9F>.
- [17] 腾讯. 小程序获取二维码[EB/OL]. [2017-12-06]. <https://mp.weixin.qq.com/debug/wxadoc/dev/api/qrcode.html>.

陈俊杰 厦门大学图书馆馆员, 经济学学士, 工学硕士。研究方向: 人工智能、数据挖掘、新技术图书馆应用。

吴明杰 厦门大学图书馆馆员, 管理学学士。研究方向: 图书馆管理。通讯作者。

张晓静 厦门大学图书馆馆员, 管理学学士。研究方向: 图书馆服务。

李 剑 厦门大学图书馆馆员, 厦门大学图书馆法学分馆主任, 管理学学士, 理学硕士。研究方向: 高校图书馆学科服务、南海相关法律与政策研究。