

Exploration on Software Development System Design under Human-computer Interaction

Xiao Chen

China Mobile (Hangzhou) Information Technology Co., Ltd., Hangzhou, Zhejiang, 310000, China

Abstract

With the rapid development of information technology, the complexity of software development is increasing, and the current user experience requirements for software products, software development is no longer only writing code, more involves how to design intuitive, easy to use and meet the needs of users interactive interface. In today's digital age, human-computer interaction has become the core element of software development system design. The design of the software development system under human-computer interaction emphasizes the effective communication between users and computer systems, which requires designers to deeply understand the user's behavior, needs and psychological models, so as to create software products that meet the technical standards and are highly user-friendly. In this regard, this paper explores the design of the software development system under human-computer interaction for its reference.

Keywords

human-computer interaction; software development system; design; key points; inquiry

人机交互下的软件开发系统设计探究

陈晓

中移（杭州）信息技术有限公司，中国·浙江 杭州 310000

摘要

随着信息技术的迅猛发展，软件开发的复杂性不断增加，加之当前用户对软件产品的体验要求越来越高，软件开发不再仅是编写代码，更涉及到如何设计出直观、易用且满足用户需求的交互界面。在当今数字化时代，人机交互已成为软件开发系统的核心要素。人机交互下的软件开发系统设计，强调的是用户与计算机系统之间的有效沟通，它要求设计师深入理解用户的行为、需求和心理模型，从而创造出既符合技术标准又具有高度用户友好性的软件产品。对此，论文针对人机交互下的软件开发系统设计展开探究，以供参考。

关键词

人机交互；软件开发系统；设计；要点；探究

1 引言

软件开发作为当今社会信息化建设的核心，已经进入一个全新的阶段。在传统的软件开发中，开发工具的复杂性和用户与系统之间的隔阂常常导致开发效率低下，且难以充分满足开发人员的需求。人机交互技术的发展，为解决这一问题提供了新的思路。人机交互关注的是人类与计算机系统之间的互动方式，其目标在于设计出既能提高开发效率，又能优化开发体验的开发工具和环境。随着技术的不断进步，人机交互的研究成果逐渐应用到软件开发系统设计中，不仅改善了用户界面的交互性，还提高了开发过程的可控性和灵活性。

2 人机交互概述

人机交互是一门跨学科的学科，涉及计算机科学、认知心理学、社会学、设计学等多个领域。其核心目标是研究如何提高人类用户与计算机系统之间的互动质量。随着信息技术的飞速发展，传统的硬件设备和软件系统逐渐无法满足用户日益复杂的需求，尤其是在软件开发过程中，开发人员不仅要关注系统的功能实现，还需要注重如何优化与系统的互动，提高开发效率与软件质量。人机交互的研究涵盖了用户界面设计、交互模式、可用性测试、认知负担分析等方面，并通过一系列方法与技术，如可视化编程环境、智能化开发工具等，帮助开发人员实现更高效的开发体验。在软件开发系统设计中，人机交互的应用不仅优化了开发工具的设计，还为开发人员提供了更为直观、易用、个性化的开发环境。

【作者简介】陈晓（1983-），女，中国浙江东阳人，本科，工程师，从事产品设计、产品测试、用户体验、交互设计等研究。

3 人机交互下的软件开发系统设计要点

3.1 用户界面与交互设计优化

提升开发人员使用体验的关键在于软件开发系统设计中用户对用户界面优化。首先要模块化设计开发工具的界面，清晰划分各个功能区域，避免操作过程中出现混淆的情况。具体地说，项目管理、代码编辑、调试控制台等功能区域独立成模块，开发人员可以根据需要对模块的显示和布局进行调整，甚至将不常用的功能隐藏起来，这样让用户的界面视觉负担就会减轻。同时，采用自适应布局方式，并支持多种显示装置及解析度，让界面可自动调整不同荧幕尺寸，提升界面的应变能力。其次，交互性设计中，针对按键、菜单、输入框等交互要素的设计要遵循一致性原则，即处于不同开发场景下相同功能控件样式与位置都保持一致，如此一来不但在使用的时候要减少开发人员的学习代价。在操作过程中，为了保证开发人员能够很快的获得操作结果，并做出相应的调整，系统应该提供如操作成功的提示或错误警告在内的实时反馈信息。同时反馈方式除了文字资料之外，还可以通过多种形式传递出来，比如色彩、图标、声音等，以达到增强用户感知的目的。另外，在代码调试、项目编译等复杂操作场景下，系统要提供动态进度及状态指示，使开发人员对目前操作的进度做到心中有数，严禁出现长时间等待而没有任何反馈的情况。最后，用户界面设计另外一个重要环节就是图形化界面和命令行操作二者无缝衔接。开发工具要同时支持图形化接口和命令行的操作，并且能够实现两者之间无缝切换，这样才能保证开发人员在复杂的编程任务中选择最便捷的操作方式，同时也可以让他们在不同的工作流中对合适的操作方式进行灵活地选择。

3.2 代码补全和智能化提示功能

提升开发人员编程效率的关键在于代码补全和智能化提示功能的设计。首先，代码补全功能设计中可根据开发人员输入上下文信息基础上予以智能推荐。举例来说，当开发人员输入函数名几个字符后，代码补全功能按照函数的命名规则、调用频率等信息将函数名字自动补全函数，同时在该过程中同步展示出这一函数的相关文档或参数提示，这样一来有助于开发人员快速了解函数的使用方法。开发工具可以整合机器学习算法，分析开发人员的编码习惯和项目的具体需求，提供个性化的补充建议，以提高智能程度。例如，如果某一开发人员经常使用某些具体的代码库或框架，软件开发系统会将这些偏好记录下来，当其在输入代码时系统会对这些代码段给予优先推荐。另外，代码补全不仅限于简单的单词或函数名称，系统还能对诸如自动填充条件判断语句、循环结构、异常处理块等常见编程结构这些复杂性更高的情景进行处理。其次，智能提示功能设计出了自动完成代码，还要包括实时的错误侦测和纠偏建议两个方面。在开发过程中，系统可以对代码进行实时扫描，在开发人员输入代码的同时，识别潜在的语法错误、逻辑错误和性能问题，并提供

即时的反馈。例如，系统会及时提示开发人员，并在开发人员写出可能导致空指针异常的一段代码时，为减少程序错误地出现提供纠偏建议。此外，智能提示功能还具备跨语言的代码补全。例如，在多语言开发环境下，开发人员可能会同时使用不同的编程语言，如 Java、Python、JavaScript 等，而开发工具应该可以自动识别当前正在使用的编程语言，并在代码补充和错误提示方面提供相应语言的支持。最后，随着开发工具的进化，代码补全功能应该可以处理更复杂的编码场景，如动态生成的代码，利用关系复杂程度更高的模块等，这样一来可以即便针对困难性更强项目中开发人员都可以实现高效编码工作。

3.3 协同工作与多用户支持

现代软件开发往往是团队协作的产物，因此必须将协同与多用户支持的需要考虑到开发工具的设计之中。第一，一体化版本控制系统（VCS）是协同工作支撑的根本。版本控制系统可以追踪团队成员修改项目代码的记录，确保每一位开发人员的修改不会产生冲突，一旦出现问题，可以在之前的版本中进行回溯。通常采取将如 Git 等常用版本控制系统予以整合，同时支持分支管理、合并、冲突解决等功能，以此使得多位开发人员协作时能够对代码版本进行高效管理。此外，系统应该支持实时的代码同步和合并，让多个开发人员在不用担心数据丢失或冲突的情况下，可以在各自机器上同时编辑相同的文件。开发工具有即时通信与讨论功能，让开发人员在开发过程中随时沟通与合作，以加强协作工作的便捷性。例如，通过集成的即时聊天工具与团队成员讨论，当某个开发人员在某个模块中遇到技术问题时，就能迅速得到解决问题的办法。在功能实现上，设计出基于项目的聊天室，这样可在不受项目其他部分干扰的情况下，开发人员可以针对特定模块或任务进行集中讨论。组员之间借助留言、文件分享等方式进行交流，保证信息的及时传递，减少沟通不畅而耽误工作的情况。另外，系统设计中还应整合任务管理工具，帮助团队下达任务，设定开发进度，并对任务完成情况进行实时跟踪。例如，开发人员可以将每个人的工作内容通过任务板进行分配，并在完成任务时进行标记，团队负责人可以对项目进展情况进行实时监控，确保每项任务都能按时完成。最后，开发环境应支持多层级的权限管理，这样可以针对具体开发人员在项目中的角色赋予相应操作权限，以防止敏感资料外泄或误操作给项目进度带来负面影响。

3.4 集成开发环境（IDE）的可定制化设计

集成开发环境（IDE）作为软件开发的核心工具，为了适应不同开发人员的需要，必须具有很高的可定制功能。第一，IDE 在开发环境中要允许开发人员对界面的布局、主题风格、快捷键设置等进行自定义，这是根据个人喜好而定的。具体的做法包括在界面的颜色、字体、窗口布局上允许开发人员根据个人偏好对它们进行调整，使之与自己的工作习惯更加契合。开发人员还可以将常用的功能如调试工具、

代码检查、编译器等置于易于访问的位置,根据不同的编程任务对工具栏的显示内容进行调整有助于提升其工作效率。开发人员在编辑器层面上要能够自由选择编码高亮、自动缩进以及编码折叠等编辑器功能,这样在编码的时候也能做到游刃有余。另外,IDE的可定制化在支持编程语言方面也有所体现。开发人员可以通过安装不同的插件来支持多种编程语言将IDE功能予以扩展。比如,开发人员在处理Web开发的时候,可以选择安装一些相关的插件,如JavaScript,CSS,HTML之类的,这样就可以提高开发效率;又比如,在处理数据科学项目的时候,开发人员为了满足不同需求,可以选择安装Python或R语言的外部插件。同时,IDE还要能够支持不同版本的编程语言或框架,这样就避免了由于工具切换而造成的时间浪费,开发人员就可以在相同的的环境下进行灵活的切换。最后,在调试和测试方面,IDE应该允许开发人员对界面的显示内容进行定制,以快速地进行切换,如单步执行、断点调试等,这样就增强了开发人员的操作灵活性。总体而言,IDE的可定制化设计,不仅可以提高开发效率,而且可以根据具体任务的需要,让开发人员进行个性化的调整,这样的工作舒适度和效率就会得到很好的提高。

3.5 错误检测与调试工具智能化

错误检测与调试是软件开发系统设计过程中必然要经历的一个环节。开发工具要整合智能的错误检测和调试功能,以减少调试时间及提高调试效率。首先错误检测功能利用静态代码分析、动态分析等技术对代码中存在的潜在错误进行自动识别。例如,当开发人员编写代码的时候,系统可以通过标记、提示的形式,实时分析代码结构,识别出潜在的语法错误、变量未定义、类型不匹配等问题,并将这些信息及时反馈给开发人员。可引入诸如深度学习之类的先进技术对大量的代码库开展学习,同时根据项目的具体要求进行针对性分析,从而提高检测潜在错误成功率。此外,错误检测智能化还应该能够识别出诸如内存泄漏、过深的循环嵌套等代码中潜在的性能瓶颈,从而帮助开发人员及早发

现这类问题并加以解决。智能调试工具不仅要支持常规的单步调试、断点调试等功能,还要能够提供动态监控、变量值追踪等功能,帮助开发人员实时查看代码执行状态。例如,当开发人员设定断点时,智能化调试工具应该可以自动记录程序的执行路径,并提供实时显示的可变量值,帮助开发人员对问题进行快速定位。同时,该工具还应该支持可视化的调试界面,将程序的执行流程和数据流以图形化的方式呈现出来,让开发人员能够更加直观地了解程序的执行过程,并且可以有效地减少调试耗时及复杂程度。总的来说,智能化错误检测和调试工具可以帮助开发人员在整个开发过程中更高效地发现问题以及开展错误修复,从而提升软件开发质量。

4 结语

综上所述,人机交互技术给软件开发系统带来了革命性的改变。借助于优化用户界面、智能化代码提示、协同工作支持、可定制化集成开发环境(IDE)以及智能化错误检测与调试工具等方面设计,使开发人员可以在更有效的环境中进行软件开发,并且具有更高的灵活性。未来在科技日益发展背景下,人机交互在软件开发系统设计中将发挥出更大的作用,特别是在开发人员的工作效率的提高、代码错误发生的减少以及团队协作流程的优化等方面。

参考文献

- [1] 吴雅茜.人机交互下的软件开发系统设计研究[J].计算机产品与流通,2023(6):38-40.
- [2] 吴卉玲.基于人机交互的软件开发系统设计探讨[J].电脑知识与技术,2021(17):72-73.
- [3] 马悦,刘佳.列车人机交互软件架构设计与研究[J].内燃机车,2022(5).
- [4] 曹崧浩.计算机自动检测控制系统软件开发与设计[J].信息产业报道,2023(8):106-108.
- [5] 曹刚,宋润泉,刘承宗,等.基于Qt开发软件的授权许可系统设计与实现[J].计算技术与自动化,2023,42(1):179-182.