

Web 全栈开发 职业技能等级标准

标准代码：510081

（2021 年 2.0 版）

腾讯科技（深圳）有限公司 制定

2021 年 12 月 发布

目 次

前言.....	1
1 范围.....	2
2 规范性引用文件.....	2
3 术语和定义.....	2
4 适用院校专业.....	4
5 面向职业岗位（群）	5
6 职业技能要求.....	6
参考文献.....	13

前 言

本标准按照 GB/T 1.1-2020《标准化工作导则 第1部分：标准化文件的结构和起草规则》的规定起草。

本标准起草单位：腾讯科技（深圳）有限公司、北京工业职业技术学院、长沙民政职业技术学院、武汉职业技术学院、广东工程职业技术学院。

本标准主要起草人：王辉、江源、魏蒙、郭亮、胡菡、陈金朋、赵秉强、张梓雄、张鹏、郭蕊、赵元苏、朱珍、朱贺新、刘业辉、郭广军、付朝晖、刘亮、邓小飞、项阳阳（排名不分先后）。

声明：本标准的知识产权归属于腾讯科技（深圳）有限公司，未经腾讯科技（深圳）有限公司同意，不得印刷、销售。

1 范围

本标准以客观反应现阶段软件开发和互联网行业技术发展水平对Web全栈从业人员的能力要求为目标，明确了具有Web全栈开发职业技能的人员的工作领域、工作任务、知识要求及职业技能要求。

本标准适用于Web全栈开发职业技能培训、考核与评价，相关用人单位的人员聘用、培训与考核可参照使用。

2 规范性引用文件

下列文件对于本标准的应用是必不可少的。凡是注日期的引用文件，仅注日期的版本适用于本标准。凡是不注日期的引用文件，其最新版本适用于本标准。

GB-T 5271.1-2000 信息技术 词汇 第1部分：基本术语

GB/T 18792-2002 信息技术 文件描述和处理语言 超文本置标语言(HTML)

GB_T 37668-2019 信息技术 互联网内容无障碍可访问性技术要求与测试方法

3 术语和定义

GB-T 5271.1-2000、GB_T 37668-2019、GB/T 18792-2002 界定的以及下列术语和定义适用于本标准。

3.1 网页 web page

包含超文本标记语言标签的纯文本文件。

注：网页是构成网站的基本要素。

[GB/T 37668-2019, 定义 2.1]

3.2 超文本标记语言第五版 hyper text markup language 5

通过标记符号来标记要显示的网页中各个部分的一种规范。

注：2014年超文本标记语言完成了第五次重大修改(亦被称为HTML5),用以取代1999年制定的HTML 4.01、XHTML 1.0标准。

[GB/T 37668-2019, 定义 2.9]

3.3 用户界面 user interface

泛指用户的操作界面，包含于移动APP、网页、智能穿戴设备等。

注：用户界面，是指对软件的人机交互操作逻辑、界面美观的整体设计。

3.4 浏览器 browser

一个用户代理，其主要功能是把文件呈现给用户。

[GB/T 18792-2002, 定义 4.1]

3.5 超链接 Hyperlink

称为源锚点和宿锚点的两那个锚点间的一种链接关系。该链接从源锚点走向宿锚点。源锚点也叫做链尾，宿锚点也叫做目的地和链头。

[GB/T 18792-2002, 定义 4.12]

3.6 网络 network

结点和互连分支的一种安排。

[GB-T 5271.1-2000, 定义 01.01.44]

3.7 数据库database

按照概念结构组织的数据的汇集，它描述这些数据的特征及与数据对应的实体间的关系，并支持一个或多个应用领域。

[GB-T 5271.1-2000, 定义 01.08.05]

3.8 信息安全information security

保护、维持信息的保密性、完整性和可用性，也可包括真是性、可核查性、抗抵赖性、可靠性等性质。

3.9 布局layout

应用程序内容中所有对组件进行规划和安排的有关事项。

[GB/T 37668-2019, 定义 2.18]

3.10 组件 component

软件系统中具有相对独立功能、可独立部署、可组装的软件实体。

[GB/T 37668-2019, 定义 2.17]

3.11 HTML

超文本置标语言 (Hyper Text Markup Language)。

3.12 HTML5

超文本置标语言第五版 (Hyper Text Markup Language 5)。

3.13 URL

统一资源定位符, 也称网页地址 (Universal Resource Locator)。

3.14 HTTP

超文本传输协议 (HyperText Transfer Protocol)。

3.15 HTTPS

安全套接字层的超文本传输协议 (HyperText Transfer Protocol over Secure Socket Layer)。

3.16 XSS

跨站脚本 (Cross Site Scripting)。

3.17 CSRF

跨站请求伪造 (Cross Site Request Forgery)。

3.18 API

应用程序接口 (Application Programming Interface)。

4 适用院校专业

4.1 原版职业教育专业目录

中等职业学校：计算机应用、计算机网络技术、网站建设与管理、软件与信息服务、电子与信息技术、电子技术应用、移动应用技术与服务等相关专业。

高等职业学校：计算机应用技术、计算机网络技术、软件技术、软件与信息服务、电子商务技术、计算机信息管理、大数据技术与应用、人工智能技术服务、信息安全

与管理、云计算技术与应用、电子信息工程技术、智能终端技术与应用、智能产品开发、移动互联应用技术、移动应用开发、物联网应用技术等相关专业。

应用型本科学校：计算机科学与技术、软件工程、网络工程、物联网工程、电子科学与技术、电子信息工程等相关专业。

4.2 新版职业教育专业目录

中等职业学校：计算机应用、计算机网络技术、网站建设与管理、软件与信息服务、电子信息技术、电子技术应用、物联网技术应用、移动应用技术与服务、数字媒体技术应用、大数据技术应用、网络信息安全、网络安防系统安装与维护、计算机平面设计、计算机与数码设备维修、物联网技术应用、现代通信技术应用、通信系统工程安装与维护、通信运营服务等相关专业。

高等职业学校：计算机应用技术、计算机网络技术、软件技术、大数据技术、人工智能技术应用、信息安全技术应用、云计算技术应用、电子信息工程技术、智能产品开发与应用、移动互联应用技术、移动应用开发、物联网应用技术、数字媒体技术、虚拟现实技术应用、人工智能技术应用、嵌入式技术应用、工业互联网技术、区块链技术应用、工业软件开发技术、动漫制作技术、密码技术应用、现代通信技术、现代移动通信技术、通信软件技术、通信系统运行管理、智能互联网络技术、网络规划与优化技术、电信服务与管理等相关专业。

高等职业教育本科学校：计算机应用工程、软件工程技术、网络工程技术、物联网工程技术、电子信息工程技术、大数据工程技术、人工智能工程技术、信息安全与管理、软件工程技术、数字媒体技、云计算技术、信息安全与管理、虚拟现实技术、嵌入式技术、工业互联网技术、区块链技术、电子信息工程技术等相关专业。

应用型本科学校：计算机科学与技术、软件工程、网络工程、物联网工程、电子科学与技术、电子信息工程等相关专业。

5 面向职业岗位（群）

【Web全栈开发】（初级）：主要面向IT互联网行业，从事各企事业单位、政府信息化数字化部门的PC端网页、移动端网页、小程序、应用软件开发等前端开发岗位，主要完成各类前端开发、可重用的界面组件编写，还原展示界面等工作，高效完成产品动态信息展现。

【Web全栈开发】（中级）：主要面向IT互联网行业，从事各企事业单位、政府信息化数字化部门的PC端网页、移动端网页、小程序、应用软件开发等前端开发岗位，主要完成技术选型、架构设计、PC端和移动端、小程序等可视化开发、产品核心模块、核心代码的编写、页面功能和交互开发Bug修复等工作，不断提升用户体验。

【Web全栈开发】（高级）：主要面向IT互联网行业，从事各企事业单位、政府信息化数字化部门PC端网页、移动端网页、小程序、全栈开发、应用软件开发、架构设计等前端开发岗位，主要完成相关API开发、关系型数据库和NoSQL数据库相关开发、Web安全问题解决、全栈开发、移动端开发、架构设计、性能优化等工作，持续优化代码和性能。

6 职业技能要求

6.1 职业技能等级划分

Web全栈开发职业技能等级分为三个等级：初级、中级、高级，三个级别依次递进，高级别涵盖低级别职业技能要求。

【Web全栈开发】（初级）：主要面向IT互联网行业，从事各企事业单位、政府信息化数字化部门的PC端网页、移动端网页、小程序、应用软件开发等前端开发岗位，主要完成各类前端开发、可重用的界面组件编写，还原展示界面等工作，高效完成产品动态信息展现。

【Web全栈开发】（中级）：主要面向IT互联网行业，从事各企事业单位、政府信息化数字化部门的PC端网页、移动端网页、小程序、应用软件开发等前端开发岗位，

主要完成技术选型、架构设计、PC端和移动端、小程序等可视化开发、产品核心模块、核心代码的编写、页面功能和交互开发Bug修复等工作，不断提升用户体验。

【Web全栈开发】（高级）：主要面向IT互联网行业，从事各企事业单位、政府信息化数字化部门PC端网页、移动端网页、小程序、全栈开发、应用软件开发、架构设计等前端开发岗位，主要完成相关API开发、关系型数据库和NoSQL数据库相关开发、Web安全问题解决、全栈开发、移动端开发、架构设计、性能优化等工作，持续优化代码和性能。

6.2 职业技能等级要求描述

表 1 Web 全栈开发职业技能等级要求（初级）

工作领域	工作任务	职业技能要求
1. 网页开发基础	1.1 网页结构搭建	<p>1.1.1 能够根据网页效果，应用 HTML 中常用的文本标签、图像标签、超链接标签、列表标签、表格标签、表单标签、音频标签、视频标签、div 标签。</p> <p>1.1.2 能够运用所掌握的各种标签，实现完成页面结构搭建。</p>
	1.2 网页样式设置与布局	<p>1.2.1 能够根据页面效果需求，依据网页引入 CSS 语法的方式和 CSS 的常用选择器，实现 CSS 样式设置。</p> <p>1.2.2 能够根据盒子模型的元素组成，应用 CSS 属性设置，实现盒子模型宽高、边框、边距和背景属性的设置。</p> <p>1.2.3 能够根据标签元素的分类，依据 display 属性的设置，实现标签类型的变换。</p> <p>1.2.4 能够根据网页布局的需求，依据网页中常见布局的方法，应用 CSS 定位、CSS 浮动和弹性布局，实现网页布局。</p> <p>1.2.5 能够根据 CSS 的继承性、CSS 的层叠性和优先级，应用 CSS 样式设置，实现在样式冲突或相同优先级的情况下，CSS 样式的优化和正确设置。</p>
	1.3 网页特效和交互效果设计	<p>1.3.1 能够根据网页过渡效果需求，应用 CSS3 中的 transition 属性的设置，实现网页中指定的过渡动画效果。</p> <p>1.3.2 能够根据网页变形动画效果要求，依据 CSS3 中的 transform 的功能，应用 rotate（旋转）、scale（缩放）、translate（移动）、skew（倾斜）等属性设置，实现文字或图像的变形动画效果。</p> <p>1.3.3 能够根据网页动画的需要，依据 animation 属性，应用关键帧来控制动画的每一步，实现动画效果。</p> <p>1.3.4 能够根据网页的功能需求，依据网页添加 JavaScript 语句的方式、JavaScript 数据类型、语法、流程控制、函数和对象，实现网页的控制和交互效果。</p>
2. 组件化网页开发	2.1 js 库动态效果设计	<p>2.1.1 能够根据网页的动态交互要求，依据 JS 库的特点，应用 JS 主流库 jQuery 的选择器、DOM 操作、事件和动画等功能进行交互效果的开发，实现网页的动态效果。</p> <p>2.1.2 能够根据 jQuery 扩展要求，依据 jQuery 的扩展原理，应用 jQuery 的扩展方法，熟练地利用常用插件或者自定义插件进行网页的开发，提高开发效率。</p>
	2.2 AJAX 动态网页开发	<p>2.2.1 能够依据浏览器的事件机制和渲染原理，应用网页开发工具和浏览器调试工具，进行网页的开发和调试。</p> <p>2.2.2 能够根据网络通信的请求，依据网络的体系结构和 HTTP 协议，应用原生的 AJAX 请求和常用的跨域技巧，获取网络数据，达到异步通信的效果。</p> <p>2.2.3 能够根据数据请求任务要求，依据 jQuery 的操作 AJAX 的方法，利用 jQuery 提供的 AJAX 相关 API，异步获取数据，实现网页的异步刷新，达到创建动态网页的效果。</p>

	2.3 组件化开发	<p>2.3.1 能够根据组件化开发任务要求，应用面向对象的思想，利用 JavaScript 变量、作用域、原型、构造函数等知识进行 DOM 编程，实现代码的封装，达到前端页面的模块化开发目的。</p> <p>2.3.2 能够依据组件化原理，分析组件化的应用场景和思路，进行组件化的基础开发，实现代码的复用。</p> <p>2.3.3 能够依据前端开发需求，结合页面布局、用户界面、数据、功能等，利用抽象和封装的知识，实现前端页面的组件化开发，提高开发效率。</p>
3. 移动端网页开发	3.1 移动端页面开发	<p>3.1.1 能够根据移动端网页效果，应用 HTML 中常用的元素标签，实现完成页面结构搭建。</p> <p>3.1.2 能够根据移动端页面效果需求，合理设置 CSS 样式，实现移动端网页布局。</p> <p>3.1.3 能够通过使用媒体查询实现响应式布局效果，实现适配不同终端的页面。</p>
	3.2 小程序开发	<p>3.2.1 能够独立完成微信小程序开发环境搭建和配置任务。</p> <p>3.2.2 能够根据界面效果图，使用微信开发者工具，选择小程序基础组件、合理配置，实现小程序界面布局。</p> <p>3.2.3 能够根据任务要求，利用小程序基础组件与基础 API，实现简单的交互效果。</p> <p>3.2.4 能够综合应用网页开发技巧和小程序基础知识，独立开发简单小程序。</p>

表 2 Web 全栈开发职业技能等级要求（中级）

工作领域	工作任务	职业技能要求
1. 性能和工程化	1.1 版本控制管理	<p>1.1.1 能够使用 Git 进行简单的版本管理。</p> <p>1.1.2 能够使用 Git 进行 pull/push 等版本控制基本操作。</p> <p>1.1.3 能够使用 Git 进行 commit 等提交操作。</p> <p>1.1.4 能够使用 Git 进行分支管理。</p>
	1.2 模块化和包管理器应用	<p>1.2.1 理解作用域、命名空间、模块化之间的联系，了解模块化的优点。</p> <p>1.2.2 了解模块化方案的进化史和主流的模块化规范。</p> <p>1.2.3 了解 Node.js 的作用和特点，掌握 Node.js 的安装，能运行简单的 Node.js 程序。</p> <p>1.2.4 能够运用包管理器，并熟练使用常用命令，能实现 NPM 源切换。</p>
	1.3 网页性能优化	<p>1.3.1 理解网页性能的重要性。</p> <p>1.3.2 了解网页性能衡量标准和主要的指标。</p> <p>1.3.3 能够使用主流工具对相应网页指标进行分析，并发现影响网页性能的问题点。</p> <p>1.3.4 能够运用常用手段优化网页性能。</p>

	1.4 打包机制和工程化应用	<p>1.4.1 了解 Webpack 的产生背景，理解其解决的核心问题。</p> <p>1.4.2 掌握 Webpack 的安装和基本使用。</p> <p>1.4.3 掌握 Webpack 的核心概念和常用配置。</p> <p>1.4.4 理解工程化的概念和必然性。</p> <p>1.4.5 能够基于 Webpack 的制定工程化方案，实现打包。</p>
2. 复杂网页开发	2.1 ECMAScript2015 (ES6) 应用	<p>2.1.1 了解 ES6 和 ES5 的区别，了解 E6+ 的相关拓展。</p> <p>2.1.2 掌握 ES6 引入的新特性，掌握 ES6 的核心语法。</p> <p>2.1.3 理解 ES6 Module 的关键点。</p> <p>2.1.4 能够运用 ES6 Promise 实现异步编程。</p>
	2.2 TypeScript 应用	<p>2.2.1 理解 JavaScript 的不足，理解 TypeScript 产生的背景和优点。</p> <p>2.2.2 能够搭建 TypeScript 开发环境，能使用第三方库简化流程。</p> <p>2.2.3 理解 TypeScript 中的各种静态类型，掌握 TypeScript 语法和使用方式。</p> <p>2.2.4 能够在主流框架中整合 TypeScript，实现核心功能。</p>
	2.3 主流框架应用	<p>2.3.1 了解常见的前端主流框架的种类，特点和优缺点。</p> <p>2.3.2 了解 MVVM 模式与 MVP 模式与 MVC 模式的区别，掌握 MVVM 原理中生命周期、虚拟 DOM、内容渲染等。</p> <p>2.3.3 能够至少熟练应用一个主流前端框架，掌握相关实现原理，熟悉其语法和核心 API。</p> <p>2.3.4 能够配合主流打包、编译、构建工具，并实现优化组合。</p>
3. 移动端网页开发	3.1 移动端基础应用	<p>3.1.1 了解移动端的概念以及与 PC 端的区别。</p> <p>3.1.2 了解移动端屏幕、移动端浏览器、操作系统的不同。</p> <p>3.1.3 掌握主流移动端页面开发技术选型与常用适配方案。</p> <p>3.1.4 能够解决移动端固定定位、字体单位选择、光标颜色等常见问题。</p> <p>3.1.5 能够解决主流前端框架应用中 PC 端和移动端页面开发中的适配和兼容性问题。</p>
	3.2 响应式设计实现	<p>3.2.1 掌握 HTML5 多媒体技术在移动端页面上的应用。</p> <p>3.2.2 掌握响应式布局开发与 Flex 布局，掌握 CSS 响应式方案。</p> <p>3.2.3 能够通过网页布局、样式和业务逻辑处理，实现适配不同终端的页面。</p> <p>3.2.4 掌握媒体查询中的媒体类型以及媒体特性的概念与用法，能使用媒体查询实现响应式布局效果。</p> <p>3.2.5 能够熟练掌握主流前端框架中响应式设计及其原理，并能根据业务需求设计实现相关功能。</p>
	3.3 小程序开发	<p>3.3.1 了解小程序的概念、开发环境搭建和配置。</p> <p>3.3.2 掌握小程序基础组件与操作。</p> <p>3.3.3 掌握小程序常用 API。</p> <p>3.3.4 能够运用主流前端框架开发小程序应用的相关功能。</p> <p>3.3.5 能够综合应用网页开发技巧和小程序基础知识，独立开发小程序。</p>

表 3 Web 全栈开发职业技能等级要求（高级）

工作领域	工作任务	职业技能要求
1. 服务器端开发	1.1 Node.js 应用	<p>1.1.1 能够理解服务端开发和前端开发思路上的区别，了解 Node.js 产生的背景和特点。</p> <p>1.1.2 能够理解 Node.js 事件驱动机制，掌握常见模块的使用，掌握 Node.js 生态工具的运用。</p> <p>1.1.3 能够掌握 Express 等至少一种主流 Node Server 框架的使用，理解中间件实现原理。</p> <p>1.1.4 能够理解 RESTful 架构风格，能使用 Node.js 或相关框架设计和开发 RESTful API。</p> <p>1.1.5 能够理解 GraphQL 的设计理念，能够掌握 GraphQL 的 Schema 等语法，能够使用 Apollo 等框架搭建 GraphQL 服务端与客户端。</p>
	1.2 数据库应用	<p>1.2.1 能够了解 MySQL 等主流关系型数据库和 MongoDB 等 NoSQL 数据库，及其特点和应用场景。</p> <p>1.2.2 能够掌握关系型数据库常用命令以及在 Node 中的应用。</p> <p>1.2.3 能够掌握常用 NoSQL 数据库常用命令以及在 Node 中的应用。</p> <p>1.2.4 能够掌握数据驱动和 ORM，能定义和同步模型，能针对模型完成增、删、查、改。</p>
	1.3 Linux 与应用部署	<p>1.3.1 能够掌握基本的 Linux 服务器操作命令，能够进行基础的 Linux 环境下服务管理操作。</p> <p>1.3.2 能够理解 Docker/Kubernetes 中的基础概念，能够编写简单的 Dockerfile 文件并针对前端应用进行打包与部署。</p> <p>1.3.3 能够理解 DevOps 中的基础概念，能够结合 Gitlab CI/Jarvis 等工具搭建 CI/CD workflow，能够执行线上应用的滚动部署/蓝绿部署等操作。</p>
2. 架构设计	2.1 设计模式	<p>2.1.1 能够理解前端设计模式的核心思想和基本理念，掌握其抽象的设计原则。</p> <p>2.1.2 能够了解常用的设计模式在主流前端框架中的应用。</p> <p>2.1.3 能够理解前端应用场景中创建型模式、结构型模式和行为型模式等核心设计模式的场景和应用。</p> <p>2.1.4 能够掌握常用代码优化的语言特性，利用设计模式进行代码优化。</p>
	2.2 算法和数据结构	<p>2.2.1 能够掌握冒泡排序、选择排序、插入排序、归并排序、快速排序等算法。</p> <p>2.2.2 能够掌握如数组、链表、队列等常用数据结构的特点和实现。</p> <p>2.2.3 能够理解和掌握递归（深度、广度搜索）、分治策略（如二分查找法）、排序等常用算法。</p> <p>2.2.4 能够结合前端实际开发情景，在业务中运用数据结构与算法。</p>

	2.3 框架/库的编写	<p>2.3.1 能够掌握 VueJS 等主流前端框架的设计理念，能够分析主流前端框架的代码组织与模块构成，能够根据团队需求个性化改造/复刻相关框架。</p> <p>2.3.2 能够针对团队的业务需求，使用 Webpack 等构建脚手架工具，能够统一团队内部的开发环境、代码审查、部署交付等基础流程。</p> <p>2.3.3 能够了解低代码搭建的理念与实践，能够快速构建团队物料库，能够进行领域低代码搭建平台的开发。</p>
	2.4 微前端与跨端/混合开发	<p>2.4.1 能够掌握微前端的设计理念与实现方式，能够利用 Webpack 等脚手架搭建团队内微服务框架与流程。</p> <p>2.4.2 能够了解 Electron、Cordova、React Native 等混合开发框架的使用方式，能够利用这些框架开发桌面端、移动端等应用。</p> <p>2.4.3 能够了解常见的小程序跨端框架的设计理念与实现方式，能够自行设计、开发小程序框架。</p>
3. 线上业务支撑	3.1 安全预防	<p>3.1.1 能够理解和掌握前端常见的 XSS、CSRF、Cookie 等安全问题等的预防和解决。</p> <p>3.1.2 能够了解常见 HTTP 协议的安全问题和危害，掌握常见的数据加密和鉴权方案。</p> <p>3.1.3 能够理解和掌握服务器端常见的密码安全、信息泄露等信息安全问题及其预防和解决。</p> <p>3.1.4 能够理解和掌握服务器端 SQL 注入、上传漏洞等信息安全问题的预防和解决。</p>
	3.2 性能优化	<p>3.2.1 能够深刻理解输入 URL 到页面加载完成的全流程，掌握前端性能优化的切入点。</p> <p>3.2.2 能够掌握 http_load、siege 等性能检测的常用工具和指标，进行性能问题的分析和定位。</p> <p>3.2.3 能够使用预加载、懒加载、精灵图、字体图标等从网络、渲染、存储、端能力等方面优化和解决性能相关问题。</p> <p>3.2.4 能够利用构建工具如 webpack 等对各种前端资源加载进行优化。</p>
	3.3 用户体验中台	<p>3.3.1 能够掌握 APM 的核心关注指标与追踪维度，能够针对团队业务个性化定制体验度量指标。</p> <p>3.3.2 能够了解常见网页分析统计、追踪反馈平台的部署与使用方式，能够快速为线上业务集成埋点统计分析的能力。</p> <p>3.3.3 能够掌握前端错误捕获、上报与追踪的方法，能够搭建错误追踪平台。能够了解用户行为捕获与复现的方式，能够为用户提供实时反馈能力。</p>

参考文献

- [1] GB-T 5271.1-2000 信息技术 词汇 第1部分：基本术语
- [2] GB/T 18792-2002 信息技术 文件描述和处理语言 超文本置标语言(HTML)
- [3] GB_T 37668-2019 信息技术 互联网内容无障碍可访问性技术要求与测试方法
- [4] 中华人民共和国职业分类大典（2015年版）
- [5] 中华人民共和国高等教育法（2018年修正本）
- [6] 高等职业学校软件技术专业顶岗实习标准（2017年）
- [7] 国家职业教育改革实施方案（2019年）
- [8] 中等职业学校专业教学标准（2017年版）
- [9] 高等职业学校专业教学标准（2019年）
- [10] 职业技能等级标准开发指南（试行）（2020年版）
- [11] 教育部关于印发《职业教育专业目录（2021年）》的通知（教职成〔2021〕2号）
- 《教育部关于公布2019年度普通高等学校本科专业备案和审批结果的通知》
（教高函〔2020〕2号）
- [12] 《教育部关于公布2020年度普通高等学校本科专业备案和审批结果的通知》
（教高函〔2021〕1号）
- [13] 人力资源社会保障部, 国家质量监督检验检疫总局, 国家统计局. 中华人民共和国职业分类大典（2015年版）