

کمی درباره شبکه

پروژه مبانی برنامه‌سازی

مدل کارخواه - کارگزار

مدل کارخواه-کارگزار (client-server) یک ساختار توزیع‌شده در پیاده‌سازی نرم‌افزار است که در آن حجم کارهایی که نرم‌افزار بر دوش می‌کشد بین بخشی از برنامه که فراهم‌کننده‌ی خدمات و منابع است و بخش دیگر که موظف به دریافت درخواست‌های کاربر است، تقسیم می‌شود.

معمولاً این دویخش در دو رایانه جداگانه که ارتباط میان آن‌ها از طریق شبکه تأمین می‌شود قرار دارند، اما ممکن است که هر دو بخش روی یک رایانه نیز اجرا شوند.

رایانه‌ی کارگزار برنامه‌های سطح خود را اجرا می‌کند و منابع خود را به صورت اشتراکی در اختیار رایانه‌های کارخواه قرار می‌دهد، در حالی که رایانه‌های کارخواه منابعی به اشتراک نمی‌گذارند و صرفاً درخواست‌هایشان را به کارگزار ارسال می‌کنند.

کارخواه و کارگزار برای ارتباط با یکدیگر از قراردادهایی استفاده می‌کنند که بتوانند حرف یکدیگر را متوجه شوند. در مورد یکی از قراردادهای معمول در همین مستند توضیحاتی داده خواهد شد. از جمله برنامه‌هایی که از این مدل استفاده می‌کنند، رایانامه‌ها و سیستم‌های مبتنی بر وب را می‌توان نام برد.

هر کارگزار بسته به نوع خدماتی که ارائه می‌کند می‌بایست دارای قابلیت‌های لازم برای اجرای خدمات مربوطه باشد. برای مثال کارگزار خدمات وب باید قابلیت ارائه‌ی داده‌های لازم برای ایجاد صفحات وب و ارسال آن به رایانه‌ی کارخواه را داشته‌باشد.

نحوه‌ی ارتباط کارخواه و کارگزار

برنامه‌ی اجراشده در رایانه‌ی کارخواه نیازی به دانستن عملیات مختلفی که در رایانه کارگزار صورت می‌گیرد ندارد، در عوض باید بتواند پاسخ‌هایی که از جانب کارگزار به درخواست‌هایش می‌شود را درک کند و آن‌را برای کاربر به نمایش بگذارد.

از سوی دیگر کارگزار نیز باید بداند که منتظر چگونه درخواست‌هایی باید باشد و برای هر درخواست چه کارهایی باید صورت بگیرد و چه پاسخ‌هایی باید به سمت کارخواه فرستاده شود.

به این روش ایجاد ارتباط میان کارخواه و کارگزار درخواست-پاسخ (request-response) می‌گویند.

قرارداد HTTP

HTTP قراردادی برای پیاده‌سازی سیستم‌های اطلاعاتی است که دارای اطلاعات توزیع‌شده، مشترک و ابررسانه‌ای هستند. HTTP در حقیقت بنیان تبادل اطلاعات در شبکه‌ی جهانی وب است. ابرمتن (hypertext) متن ساختاریافته‌ای است که از پیوندهای منطقی بین خود و ابرمتن‌های دیگر استفاده می‌کند.

نشست HTTP

برای آن‌که کارخواه (مثلاً رایانه‌ی کاربری که در وب جست‌وجو می‌کند) به کارگزار متصل شود باید در ابتدا درخواست ایجاد نشست میان خود و کارگزار HTTP مقصد را باید از طریق درگاهی در رایانه‌ی مقصد که کارگزار روی آن در حال شنیدن است، به کارگزار برساند.

پس از دریافت درخواست، در صورت موفقیت‌آمیز بودن ایجاد نشست، کارگزار پیغام:

HTTP/1.1 200 OK

را به سمت کارخواه برمی‌گرداند. در غیر این صورت پیامی مبنی بر خطای موجود برای ایجاد نشست را به کارخواه باز می‌گرداند.

برخی پیغام‌های خطایی که ممکن است پاسخ دهد در زیر آمده‌است. این درخواست‌ها قالب 4xx دارند:

- کد 400 Bad Request : پیغام در پاسخ به درخواست بد و نامعتبر است که کارگزار قادر به فهم آن نیست.

- کد 401 Unauthorized : این پاسخ زمانی ایجاد می‌شود که دسترسی به منبعی در کارگزار نیاز به احراز هویت کارخواه باشد و هنوز صورت گرفته نشده‌باشد.

- کد 403 Forbidden : این پاسخ زمانی است که کارگزار درخواست را فهمیده‌است اما از پاسخ‌گویی به آن خودداری می‌کند.

- کد 404 Not Found : این پاسخ به درخواست‌هایی است که کارگزار منبع درخواست‌شده را در رایانه‌ی خود پیدا نمی‌کند.

به این ترتیب نشستی بین کارگزار HTTP و کارخواه ایجاد می‌شود.

برخی از انواع درخواست‌ها در HTTP

پس از ایجاد نشست میان کارخواه و کارگزار، HTTP کارخواه می‌تواند با ارسال درخواست‌هایی اطلاعات لازم برای ایجاد صفحات وب را دریافت کند. توضیح برخی از آن‌ها در زیر آمده است.

درخواست GET

این درخواست از کارگزار می‌خواهد تا یک فایل مشخص در رایانه‌ی کارگزار را نمایش دهد. مثلاً مرورگر وب از کارگزار www.example.com می‌خواهد تا برایش محتویات فایل `index.html` را نمایش دهد:

GET /index.html HTTP/1.1

Host: www.example.com

و کارگزار در پاسخ به آن پاسخ زیر را ارسال می‌کند:

```
HTTP/1.1 200 OK
Date: Mon, 23 May 2005 22:38:34 GMT
Content-Type: text/html; charset=UTF-8
Content-Encoding: UTF-8
Content-Length: 138
Last-Modified: Wed, 08 Jan 2003 23:11:55 GMT
Server: Apache/1.3.3.7 (Unix) (Red-Hat/Linux)
ETag: "3f80f-1b6-3e1cb03b"
Accept-Ranges: bytes
Connection: close
```

```
<html>
<head>
<title>An Example Page</title>
/<head>
<body>
  Hello World, this is a very simple HTML document.
</body>
</html>
```

این نکته را نیز باید افزود که به وسیله‌ی این دستور می‌توان به سمت رایانه‌ی کارگزار اطلاعاتی را نیز حین درخواست خود ارسال کند.

این اطلاعات می‌تواند در انتهای نشانی درخواستی از سوی کارخواه اضافه شود. مثال زیر را برای روشن شدن این کار مطرح می‌کنیم:

`/directory/index.php?name=YourName&day=Yourday`

این درخواست به سمت رایانه‌ی کارگزار فرستاده می‌شود. برنامه‌ی مجری http در رایانه‌ی کارگزار پس از دریافت و پردازش چنین درخواستی داده‌های name و day را از آن استخراج می‌کند. با داشتن چنین داده‌هایی پاسخ‌گویی کارگزار به نیازهای کارخواه منعطف‌تر خواهد شد.

درخواست HEAD

این درخواست نیز شبیه به درخواست GET است با این تفاوت که به جای آن که محتویات فایل را دریافت کند، metadata هایی در مورد آن از کارگزار طلب می‌کند. در مثال بالا تنها بخش بالایی پاسخ دریافت می‌شود و برچسب‌های html دریافت نمی‌شود.

درخواست POST

این درخواست برای ارسال اطلاعات که معمولاً در form های html صورت می‌گیرد، کاربرد دارد. ارسال اطلاعات در این درخواست به منظور پردازش شدن داده‌ها در یک منبع مشخص است. این پردازش اطلاعات ممکن است باعث ایجاد منبع جدید در رایانه‌ی کارگزار یا تغییری در فایل‌های موجود در آن بشود. در این درخواست اطلاعاتی که فرستاده می‌شود، در بدنه‌ی درخواست قرار می‌گیرند. نکته‌ی مهم در تفاوت میان ارسال اطلاعات در GET و POST این است که در GET نباید داده‌های مهم (از قبیل گذرواژه‌ها و ...) را فرستاد چرا که به سادگی قابل مشاهده است. در عوض یکی از کاربردهای اصلی POST نیز همین است.