**Task 4**

**الگو هاي ساختاری پروژه**

Adapter

این الگو به دو کلاس ناسازگار اجازه می‌دهد تا با یکدیگر ارتباط برقرار کنند. به عنوان مثال، فرض کنید یک سایت کاریابی از یک سیستم مدیریت محتوا (CMS) برای مدیریت محتوای خود استفاده می‌کند CMS. از یک فرمت خاص برای ذخیره فایل‌های رزومه استفاده می‌کند، اما سایت کاریابی از یک فرمت دیگر برای ذخیره فایل‌های رزومه استفاده می‌کند. در این صورت، می‌توان از الگو Adapter برای سازگار کردن فرمت‌های مختلف فایل‌های رزومه استفاده کرد.

Decorator

این الگو به شما امکان می‌دهد تا ویژگی‌های جدیدی را به یک کلاس بدون تغییر کد اصلی آن کلاس اضافه کنید. به عنوان مثال، فرض کنید یک سایت کاریابی به دنبال اضافه کردن فیلترهای جدید به صفحه نتایج جستجو خود است. در این صورت، می‌توان از الگو Decorator برای اضافه کردن فیلترهای جدید به صفحه نتایج جستجو بدون تغییر کد اصلی صفحه استفاده کرد.

Façade

این الگو به شما امکان می‌دهد تا ویژگی‌های جدیدی را به یک کلاس بدون تغییر کد اصلی آن کلاس اضافه کنید. به عنوان مثال، فرض کنید یک سایت کاریابی به دنبال اضافه کردن فیلترهای جدید به صفحه نتایج جستجو خود است. در این صورت، می‌توان از الگو Decorator برای اضافه کردن فیلترهای جدید به صفحه نتایج جستجو بدون تغییر کد اصلی صفحه استفاده کرد.

Flyweight

این الگو به شما امکان می‌دهد تا اشیا را به صورت کارآمدتر ذخیره و مدیریت کنید. به عنوان مثال، فرض کنید یک سایت کاریابی از تصاویر پروفایل کارجویان استفاده می‌کند. در این صورت، می‌توان از الگو Flyweight برای ذخیره و مدیریت کارآمدتر تصاویر پروفایل کارجویان استفاده کرد.

Proxy

این الگو به شما امکان می‌دهد تا به طور غیر مستقیم به یک شی دسترسی پیدا کنید. به عنوان مثال، فرض کنید یک سایت کاریابی می‌خواهد دسترسی به اطلاعات شخصی کارجویان را محدود کند. در این صورت، می‌توان از الگو Proxy برای کنترل دسترسی به اطلاعات شخصی کارجویان استفاده کرد.

**الگوهای رفتاری**

**Chain of Responsibility**

این الگو به شما امکان می‌دهد تا درخواست‌ها را به ترتیب به چندین عنصر منتقل کنید. به عنوان مثال، فرض کنید یک سایت کاریابی از یک سیستم مدیریت ایمیل برای ارسال ایمیل‌های اطلاع‌رسانی استفاده می‌کند. در این صورت، می‌توان از الگو Chain of Responsibility برای پردازش درخواست‌های ارسال

ایمیل‌های اطلاع‌رسانی استفاده کرد.

Command

این الگو به شما امکان می‌دهد تا درخواست‌ها را به صورت کارآمدتر مدیریت کنید. به عنوان مثال، فرض کنید یک سایت کاریابی از یک سیستم مدیریت آگهی‌های استخدام استفاده می‌کند. در این صورت، می‌توان از الگو Command برای مدیریت درخواست‌های حذف آگهی‌های استخدام استفاده کرد.

**Iterator**

این الگو به شما امکان می‌دهد تا به صورت کارآمدتر از یک مجموعه داده تکرار کنید. به عنوان مثال، فرض کنید یک سایت کاریابی از یک مجموعه داده برای ذخیره آگهی‌های استخدام استفاده می‌کند. در این صورت، می‌توان از الگو Iterator برای تکرار کارآمدتر از مجموعه آگهی‌های استخدام استفاده کرد.

**Mediator**

این الگو ارتباط بین اجزای نرم‌افزار را تسهیل می‌کند. به عنوان مثال، فرض کنید یک سایت کاریابی از چندین بخش مختلف مانند بخش استخدام، بخش بازاریابی و بخش پشتیبانی استفاده می‌کند. در این صورت، می‌توان از الگو Mediator برای تسهیل ارتباط بین بخش‌های مختلف سایت استفاده کرد.

Observer

این الگو به اجزای نرم‌افزار اجازه می‌دهد تا در صورت تغییر وضعیت یکدیگر مطلع شوند. به عنوان مثال، فرض کنید یک سایت کاریابی از یک سیستم مدیریت آگهی‌های استخدام استفاده می‌کند. در این صورت، می‌توان از الگو Observer برای اطلاع‌رسانی کاربران از تغییرات آگهی‌های استخدام استفاده کرد.

**الگو های ایجادی**

**Abstract Factory**

این الگو نحوه ایجاد اشیا را با استفاده از یک رابط مشترک مشخص می‌کند. به عنوان مثال، فرض کنید یک سایت کاریابی از اشیا مختلفی مانند آگهی‌های استخدام، کارجویان و کارفرمایان استفاده می‌کند. در این صورت، می‌توان از الگو Abstract Factory برای ایجاد اشیا مختلف به صورت کارآمدتر استفاده کرد.

**Builder**

این الگو به شما امکان می‌دهد تا اشیا را به صورت کارآمدتر ایجاد کنید. به عنوان مثال، فرض کنید یک سایت کاریابی از رزومه‌های پیچیده استفاده می‌کند. در این صورت، می‌توان از الگو Builder برای ایجاد رزومه‌های پیچیده به صورت کارآمدتر استفاده کرد.