**استراتژی تست جعبه سفید (White-Box Testing Strategy):**  
در ابتدا این سوال را مطرح کنیم که تست جعبه سفید چیست؟ نام جعبه سفید این استراتژی به نوعی نمادین میباشد، فقط در جهت هماهنگی با جعبه سیاه و حس تضاد به چنین اسمی معروف است، در غیر اینصورت باید آن را جعبه شفاف یا شیشه ای نامید (Transparent-Box Testing Strategy). این استراتژی یکی از روش های برجسته طراحی موارد تست یا آزمون نرم افزار است که هدف اصلی آن بررسی منطق درونی نرم افزار است. در واقع طی این تست روال منطقی برنامه دنبال خواهد شد. بیاییم کمی ساده تر صحبت کنیم! در استراتژی جعبه سفید انگار جزئیات نرم افزار مانند کدهای منبع، مستندات طراحی و... را درون یک جعبه شیشه ای گذاشته ایم، بدینصورت که میتوان محتویات داخل آن را مشاهده و از نحوه عملکرد آن آگاه شد.

افرادی که این استراتژی را پیاده سازی میکنند معمولا اعضای تیم توسعه و تیم مستقل تست هستند. این افراد توسط این استراتژی به منطق درونی و ساختار طراحی و کدنویسی نرم افزار احاطه خواهند داشت. به زبان ساده عرض کنم که وقتی از مستندات و جزئیات دقیق یک محصول نرم افزاری آگاه باشیم، قاعدتا از نحوه پیاده سازی ان هم آگاه خواهیم بود. بنابراین فرد یا تیم مورد نظر با آگاهی کامل از ریز جزئیات نرم افزار به بررسی آن میپردازد. از جمله اعمالی که در این استراتژی انجام میشود:

1. بررسی خطوط کد منبع، بصورت خط به خط و جزء به جزء، بصورتیکه خطوط کد و مسیرهای مستقل داخل یک پیمانه حداقل یکبار اجرا و تست شوند.  
2. چک کردن تصمیمات منطقی برنامه، برای مثال تمامی شرط ها را، حتی else هایی که شاید هیچگاه اجرا نشوند را تست کنیم. (چیزی شبیه کنترل نوع ایستا در طراحی و پیاده سازی زبان های برنامه سازی)  
3. همه حلقه ها باید تست شوند، زیرا میدانیم که حلقه ها نقش تاثیرگذاری در برنامه، خصوصا در بخش حافظه ایفا میکنند. بنابراین بررسی روند کارکرد و فضای مصرفی آنها لازم بنظر میرسد.  
4. از تمامی ساختارهای اطلاعاتی داخلی در جهت تضمین اعتبار نرم افزار استفاده کنیم.

راجر پرسمن (Roger S. Pressman) در کتاب مهندسی نرم افزار خود مینویسد: "استراتژی تست جعبه سفید باید پس از مرحله برنامه نویسی طراحی شود، زیرا باید جزئیات منطقی برنامه را در اختیار داشته باشیم."

**چه نیازی به تست جعبه سفید داریم؟**  
در اواسط بحث به این اشاره کردیم که هدف اصلی استراتژی تست جعبه سفید، بررسی و پیگیری منطق درونی برنامه است. راجر پرسمن در کتاب خود سوالی را طراحی میکند، مبنی بر اینکه "چرا زمان و انرژی خود را صرف نگرانی در مورد جزئیات منطقی کنیم؟" پاسخ این مورد را به آینده موکول میکنیم.

**استراتژی تست جعبه سفید (White-Box Testing Strategy):**

در ابتدا این سوال را مطرح کنیم که تست جعبه سفید چیست؟

نام جعبه سفید این استراتژی به نوعی نمادین می‌باشد، فقط در جهت هماهنگی با جعبه سیاه و حس تضاد به چنین اسمی معروف است، در غیر این‌صورت باید آن را جعبه شفاف یا شیشه‌ای نامید (Transparent-Box Testing Strategy). این استراتژی یکی از روش‌های برجسته طراحی موارد تست یا آزمون نرم‌افزار است که هدف اصلی آن بررسی منطق درونی نرم‌افزار است. در واقع طی این تست روال منطقی برنامه دنبال خواهد شد. در استراتژی جعبه سفید جزئیات نرم‌افزار مانند کدهای منبع، مستندات طراحی و… را درون یک جعبه شیشه‌ای گذاشته‌ایم، بدین‌صورت که می‌توان محتویات داخل آن را مشاهده و از نحوه عملکرد آن آگاه شد.

افرادی که این استراتژی را پیاده سازی می‌کنند معمولا اعضای تیم توسعه و تیم مستقل تست هستند. این افراد توسط این استراتژی به منطق درونی و ساختار طراحی و کدنویسی نرم‌افزار احاطه خواهند داشت. وقتی از مستندات و جزئیات دقیق یک محصول نرم‌افزاری آگاه باشیم، قاعدتا از نحوه پیاده سازی آن هم آگاه خواهیم بود. بنابراین فرد یا تیم مورد نظر با آگاهی کامل از ریز جزئیات نرم‌افزار به بررسی آن می‌پردازد.

**هدف از تست جعبه سفید:**

اهدافی که از این استراتژی دنبال می‌شوند عبارتند از:

* بررسی حفره‌های امنیتی
* بررسی مسیرهای مسدود یا ناقص
* بررسی ساختار پیاده‌سازی شده با مستندات تهیه شده
* بررسی خروجی‌های مورد انتظار
* بررسی تمامی حلقه‌ها و شروط آن‌ها
* بررسی خط به خط کد برنامه و یا هر بخشی از آن و پوشش تست ۱۰۰ درصد
* و بالاتر و برتر از همه این اهداف، می‌توان گفت بررسی مراحل در STLC (چرخه توسعه تست نرم افزار)  که اشاره به تست واحد(Unit Testing)، تست یکپارچه‌سازی (Integration Testing) و تست سیستم (System Testing) داشت.