

آزمایشگاه نرم افزار

گزارش کار شماره 7

گروه 6

مهران حدادی، سیدمعین نوری

نجفی، محمد رستمیان

ویژگی های توصیفی

1. کارایی (Performance/Efficiency) :

طبق فرمولی که برای کارایی داریم :

$$Performance = \frac{P_{Output}}{P_{Input}}$$

سیستم ما باید طوری Process ها را انجام دهد که P_{Output} (توان خروجی) نسبت به P_{Input} (توان ورودی) افزایش یابد تا درنهایت $Performance$ ما بالا رود.

2. امنیت (Privacy/Policy) :

سیستم طوری طراحی شود که نسبت به خطرات خارجی نظیر هکر ها مقاومت داشته باشد و بتواند از خود و داده های سیستم و همچنین داده های کاربران دفاع کند. به عنوان مثال از پروتکل های SSH در شبکه و از HTTPS در محیط اینترنت استفاده کنیم.

3. سرعت :

سیستمی طراحی شود که بتواند در کمترین زمان ممکن، به نیاز کاربران جواب دهد و اخلاقی در عملکرد سیستم بوجود نیاید. نظیر استفاده از سرورهای بیشتر در سایت هایی که کاربران زیادی دارند و همچنین Balancer های مناسب برای اینکار و ...

4. قابلیت استفاده مجدد (Reusability) :

سیستم طوری طراحی شود که تا حد امکان از مؤلفه های آماده استفاده نماید تا علاوه بر اینکه بتوانیم از سیستم در جاهای دیگر استفاده کنیم، بتوانیم از قطعه های کوچکتر آن به طور مازولار استفاده نماییم.

5. قابلیت جابجایی (Portability) :

اصطلاحاً Cross Platform نیز گفته می شود. یعنی به پلتفرم خاصی وابسته (Dependence) نباشد و بتوانیم روی سیستم عامل ها و دستگاه های مختلف بدون مشکل اجرا کنیم. مانند استفاده از زبان های برنامه نویسی نظیر جاوا.

6. نگهداری (Maintenance) :

عامل های مختلفی برای نگهداری مطرح می شود مانند عواملی نظیر اندازه نرم افزار، پیچیدگی نرم افزار و ...

در سیستم دانشگاه :

• امنیت :

برای احراز هویت از کاربر Username و Password می خواهد و این کار را برای دانشجو، استاد و... انجام می دهیم و مقادیر وارد شده را با دیتابیس تراز می دهیم و اگر درست بود اجازه ورود می دهد. برای ارسال داده های ارسالی از کاربر، از متد Post در پروتکل HTTP به جای متد Get استفاده می شود تا داده ها در Body صفحه HTML به سمت Server ارسال شود و داده ها نتوانند سرقت شوند.

از Validation Component برای تست کردن استفاده می شود، این تست Unit Testing تست را به صورت واحد مورد تست Component یک از Use Case قرار می دهد. فرض کنید در یک چند مؤلفه استفاده کردیم، هر کدام از مؤلفه ها را به مورد بررسی قرار می Unit Test طور مجزا در دهیم. اگر از لحاظ امنیت موردی نداشت، همه مؤلفه ها را در یکجا جمع می کنیم و سپس به ماژولار تبدیل می کنیم.

• Reusability :

احراز هویت به صورت یک مؤلفه طراحی شده که این شرایط را برای ما فراهم می کند که از این مؤلفه در سیستم های دیگر استفاده شود. این مورد را می توانیم در مورد مؤلفه های دیگر سیستم استفاده کنیم. در خصوص Testing مراحل زیر وجود دارد که در هر مورد می بایست از این ها به ترتیب استفاده کنیم :

1. Unit
2. integration
3. Functional
4. System
5. Stress Test
6. Performance
7. Usability
8. Acceptance
9. Regression
10. Data Test



برای تست نرم افزار، گام اول unit test است. فرض کنید در سیستم دانشگاهی، برای عملیات LOGIN ابتدا می بایست زیر برنامه ها و مولفه ها تست شوند.

1. تست مولفه احراز هویت:

- نام کاربری

- رمز عبور

2. تست یکپارچگی دومورد بالا تحت عنوان validation

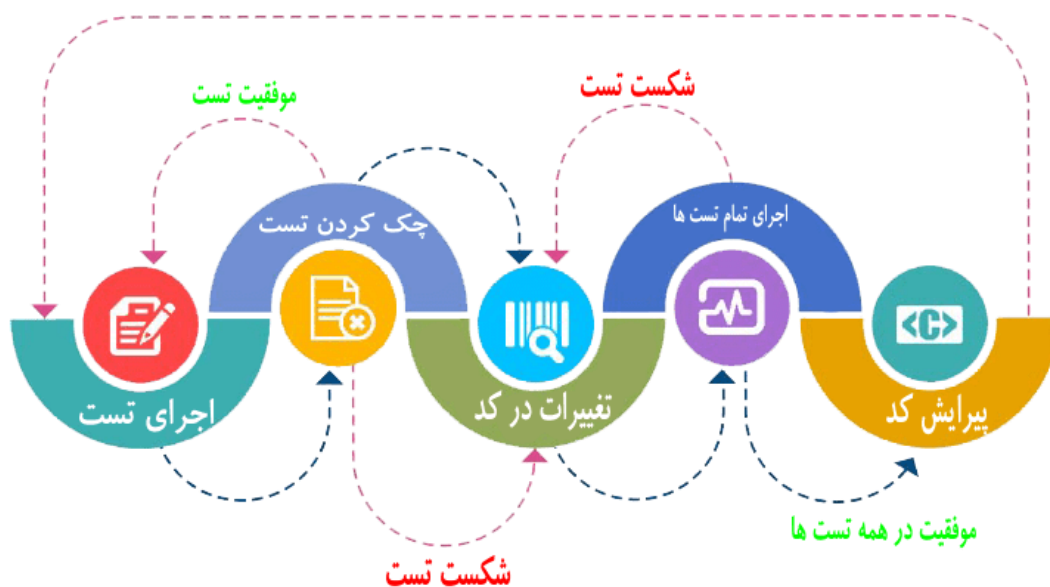
نکته: برای تمام زیر سیستم های سیستم دانشگاه، همین روال طی می شود.

3. تست سیستم (بعد از اتمام تست تمام واحدها و زیر سیستم ها)

4. نوبت به تست داده می رسد و اینکه ببینیم، سیستم با انواع داده های مجاز سازگار است یا خیر.

5.....

چرخه تست:



هر مرحله ای که با شکست مواجه شود، به ابتدای چرخه باز می گردیم.