1. **Etapele Software development life cycle (SDLC)**

**În timpul dezvoltării software, software-ul trece printr-o serie de etape.**

**Astfel pentru fiecare etapă, intervin o serie de activităţi:**

1. **Requirement analysis (Planning)**

**În primă fază, se vor stabili personalul dar şi costurile aferente dezvoltării produsului, urmând ca mai apoi să fie stabilite posibilităţile şi soluţiile aferente, astfel încat să se creeze cel mai bun produs, în special pentru client, dar şi în ceea ce priveşte competitorii.**

1. **Design**

**În aceasta etapă se vor analiza informaţiile despre publicul ţintă , dar şi crearea unei interfeţe utilizator plăcute și ușor de înțeles, bazată pe tendințe reale; urmând apoi designul interfață (utilizator și grafică), dezvoltarea stilului, selecția fonturilor și a culorilor produsului care urmează să fie dezvoltat, astfel încat să se realizeze desingul propriu-zis al produsului.**

1. **Development**

**După cum sugerează și numele, aici are loc dezvoltarea software-ului. Este primul mediu în care dezvoltatorii scriu primele linii de cod și fac toate actualizările de cod. Acest mediu constă de obicei dintr-un server care este partajat de mai mulți dezvoltatori care lucrează împreună la același proiect.**

**O serie de teste preliminare au loc în această etapă înainte de a trece la etapele următoare.**

1. **Testing**

**Mediul de testare permite inginerilor QA să testeze codul nou și modificat, fie prin tehnici automate fie prin tehnici manuale.**

**Aici se pune accentul pe testarea componentelor individuale și nu a întregii aplicații, pentru a verifica compatibilitatea dintre codul vechi și cel nou, performanța sistemului și așa mai departe.**

1. **Deployment**

**Înainte de a trece la producţia propriu-zisă, se testează într-un mediu non-producție pentru a verifica dacă aplicația se va comporta corect după implementare. Astfel, reprezintă un spațiu sigur departe de ochii publicului pentru a se asigura că totul funcționează conform așteptărilor. Chiar dacă produsul se află în faza propriu-zisă de producţie nu înseamnă că testele trebuie oprite, este recomandată şi ar trebui sa se facă testarea , acest lucru se poate face cu ajutorul semnalizatoarelor de caracteristici, care permit testarea în siguranță la nivel de producție, pe un număr preselectat de utilizatori.**

1. **Pentru fiecare etapă din SDLC explică ce activități întreprinde QA-ul.**

**Asigurarea calității software (SQA) este un proces crucial care trebuie implementat în timpul ciclului de viață al dezvoltării software. Scopul principal al QA este de a se asigura că software-ul îndeplinește funcționalitatea planificată, fără erori și că va funcționa impecabil în diferite circumstanțe.**

**QA este integrat în toate etapele de dezvoltare și este utilizat chiar și după etapa de post-lansare. Specialiștii QA creează și implementează diverse strategii pentru îmbunătățirea calității software-ului. Ei aplică tipurile de testare necesare pentru a se asigura că software-ul va funcționa corect.**

**Requirement analysis**

**Echipa QA colaborează cu analiștii de afaceri pentru a efectua un studiu de fezabilitate al cerințelor, pentru a analiza posibilele riscuri, a crea planuri de testare și pentru a construi o strategie pentru o abordare de testare a asigurării calității .**

**Design**

**Este necesar să fie revizuit designul, să se verifice testabilitatea acestuia, să se verifice dacă arhitectura de proiectare îndeplinește toate cerințele funcționale și nefuncționale. De asemenea, specialiștii QA realizează Diagrama fluxului de date (DFD) împreună cu designerii UI/UX și o documentează. În cele din urmă, inginerii QA testează designul după finalizarea acestuia pentru a imita comportamentul utilizatorului final.**

**Development**

**Asigurarea calității în dezvoltarea de software poate fi efectuată odată ce software-ul este creat sau este posibil să se utilizeze abordarea TDD - dezvoltare bazată pe teste. TDD înseamnă că software-ul trece printr-un proces de testare în timpul dezvoltării, după fiecare iterație.**

**Testing**

**Procesul de testare a software-ului este de obicei clasificat în două categorii: testare funcțională și testare nefuncțională.**

***Testarea funcțională* presupune testarea aspectelor software funcționale. Tipurile de testare funcțională arată comportamentul intern al software-ului și modul în care acesta poate interacționa cu alte sisteme.**

**Testarea funcțională poate fi efectuată atât manual , cât și cu ajutorul testării automate.**

***Testarea nefuncțională* implică testarea modului în care funcționează sistemul, sunt testate aspectele nefuncționale ale software-ului, cum ar fi performanța, gradul de utilizare, securitatea.**

**Testarea nefuncțională îi ajută pe inginerii QA să îmbunătățească software-ul într-o mare măsură.Testarea nefuncțională este întotdeauna efectuată folosind instrumente de automatizare, deoarece este un proces destul de complicat, iar testele manuale vor necesita prea mult timp și efort.**

**Deployment**

**Odată ce produsul este lansat, dezvoltatorii îl întrețin în continuare, iar echipa QA creează ghiduri de utilizare și manuale ale produsului pentru a-l livra utilizatorului final. De asemenea, specialiștii întocmesc documentația de testare pentru a se asigura că toate erorile au fost detectate și totul este remediat.**

1. **Avem următoarele environments: dev, test, staging, production. Descrie activitățile și tipurile de testare folosite pe aceste medii.Gândește-te unde se execută unit tests, testarea manuală funcțională, regression, smoke etc. Trebuie să explici ce fel de testare, unde și de ce se execută pe acel environment.**

**Development**

***Planificarea testelor.***

**Analiștii QA planifică procesul de testare, ce obiective de testare trebuie îndeplinite și ce abordare se recomandă să folosească.**

***Testare timpurie.***

**Inginerii QA efectuează teste timpurii pentru a detecta erori în timpul primei faze de dezvoltare și pentru a accelera procesul de dezvoltare.**

**Testing**

***Execuția testului.***

**Inginerii QA efectuează teste manuale/automatizate folosind diferite tipuri de testare în funcție de particularitățile software-ului.**

***Analiza defectelor.***

**Este necesar să se analizeze toate defectele și să se identifice motivul apariției lor.**

***Raportare.***

**Specialiștii folosesc sisteme de urmărire a erorilor pentru a crea rapoarte pentru dezvoltatori cu descrieri ale erorilor și defectelor care trebuie remediate.**

***Colaborare.***

**Echipa QA colaborează cu analiști de afaceri, manageri de proiect, dezvoltatori și clienți pentru a obține cea mai înaltă calitate a unui produs software.**

***Crearea raportului de rezumat al testului.***

**Când software-ul este testat, inginerii QA trebuie să creeze un raport de rezumat al testului care arată nivelul de calitate al software-ului.**

**Staging**

***Teste de performanţă***

**Aceste teste măsoară cât de repede o aplicație răspunde la o interacțiune.**

***Testarea integrării sistemului***

**Scopul principal al acesor teste este de a se asigura că toate modulele unei aplicații funcționează împreună fără probleme și pot comunica între ele.**

**Production**

***Testarea de acceptare a utilizatorului (UAT)***

**Această testare este efectuată pentru a se asigura că aplicația îndeplinește cerințele utilizatorilor și pentru a se asigura că funcționalitățile aplicației ating satisfacția clienților.**

***Testarea Alpha***

**În timpul testării Alpha, versiunea este testată în afara echipelor implicate în proiect.Acest test ajută la identificarea oricărei erori sau probleme înainte de a lansa software-ul pentru utilizatorul final.**

***Testare Beta***

**În acest mediu de testare, software-ul este lansat unui număr limitat de utilizatori din lumea reală din afara organizației, pentru a obține feedback-ul lor, care este apoi redirecționat dezvoltatorilor pentru a optimiza și îmbunătăți lansarea după cum este necesar, înainte de a fi lansat pentru toți utilizatorii.**