Tehnologii si instrumente pentru dezvoltarea programelor

Introducere

In ultimii ani tehnologiile si instrumentele folosite pentru dezvoltarea programelor au evoluat foarte repede si in momentul acesta ne ofera numeroase solutii pentru gestionarea complexitatii software-ului. Cu aceste tehnologii si instrumente putem acoperi diferite necesitati cum ar fi: scrierea codului, dar si gestionarea propriu-zisa a proiectelor, testarea si lansarea programelor. Pentru siguranta unui proiect software de succes avem nevoie desigur de un set de instrumente bine ales.

Note istorice

Spre deosebire de limbajele de programare folosite astazi, la inceput, inginerii foloseau limbaje de programare simple pentru a scrie coduri. In ultimul timp, folosim solutii cu mult mai avansate care sa ne ajute in procesele de compilare, de testare si de implementare a programelor. Cel mai popular instrument folosit la acest moment este GCC (GNU Compiler Collection) care a devenit un standard in toate mediile de dezvoltare. Cu aparitia internetului, au aparut o varietate de platforme folosind o gama larga de framework-uri si IDE-uri.

Tehnologii si instrumente pentru dezvoltare

1. Medii de dezvoltare integrate (IDEs)

Pentru acestea avem: Visual Studio, IntelliJ IDEA, Eclipse

Rolul: Este un mediu controlat de programare format din mai multe programe care ajuta programatorul in scrierea codului. Software-ul folosit combina eficient mai multe instrumente necesare dezvoltatorilor de soft intr-o singura interfata grafica. Cu ajutorul unui IDE putem construi, depana si testa codul unor aplicatii.

2. Sisteme de control al versiunilor

Pentru acestea avem: platforma de Git

Rolul: Controlul versiunilor este foarte important cand vine vorba de colaborare intre echipe si atunci cand este nevoie sa urmarim modificarile facute intr-un proiect anume. Pentru acestea folosim platforme cum ar fi GitHub si GitLab care ne permit sa gestionam eficient toate versiunile proiectelor.

3. Instrumente de testare

Pentru acestea avem: Junit, TestNG, Selenium

Rolul: Testarea software este procesul prin care o aplicatie sau un sistem este evaluat pentru a se asigura ca functioneaza corect si indeplineste cerintele specificate. Scopul principal al testarii software este de a identifica si corecta erorile, asigurand astfel calitatea si fiabilitatea produsului final.

4. Platforme DevOps si CI/CD

Pentru acestea avem: Jenkins, CircleCI

Rolul: Scopul principal este sa faciliteze colaborarea si comunicarea intre echipele de dezvoltare si operatiuni, in vederea obtinerii unui ciclu de dezvoltare si implementare eficient.

5. Instrumente pentru gestionarea proiectelor

Pentru acestea avem: Jira, Trello, Asana

Rolul: Aceste instrumente sunt utilizate in mod obisnuit pentru urmarirea erorilor, managementul proiectelor si urmarirea problemelor. Se bazeaza in intregime pe aceste trei aspecte. Este utilizat pe scara larga in dezvoltarea si testarea software-ului.

Concluzii

Toate aceste tehnologii si instrumente avansate sunt esentiale in dezvoltarea programelor si asigura eficienta codului. Trebuie sa avem in vedere alegerea corecta a acestor tehnologii si platforme pentru ca proiectul software sa fie unul de succes.

# References

<https://www.guru99.com/software-development-tools.html>

<https://www.guru99.com/ro/jira-interview-questions.html>

<https://www.roweb.ro/ro/blog/inginer-devops-transformare-digitala/>

<https://www.zaptest.com/ro/ce-este-testarea-sistemului-o-scufundare-in-abordari-tipuri-instrumente-sfaturi-si-trucuri-si-multe-altele>

<https://www.simplilearn.com/software-development-tools-article>

<https://www.headspin.io/blog/top-software-testing-tools>

<https://www.geeksforgeeks.org/software-testing-tools/>