

GitLab i GitLHub



GitLab i GitHub su jedna drugoj konkurencija kada je u pitanju arena platformi za kontrolu verzija. Iako su po mnogo čemu slične, postoje i razlike. Za donošenje odluke koja će najbolje odgovarati vašim potrebama, neophodno je da se upoznate sa ključnim osobinama koje ove platforme nude.

Sličnosti

GitLab i GitHub dele ključne sličnosti koje ih čine moćnim alatima za kontrolu verzija i kolaboraciju u razvoju softvera. Zbog toga, važno je razumeti te sličnosti kako biste mogli efikasnije koristiti ove platforme:

1. Osnovna funkcionalnost: Centralizovano skladište koda i praćenje promena

- **Repozitorijum:** Obe platforme nude centralizovano skladište koda poznato kao repozitorijum. Ovo omogućava svim članovima tima da imaju centralni pristup i pregled nad svim kodovima projekta.
- **Praćenje promena:** Obe platforme omogućavaju praćenje promena koda tokom vremena. Možete videti ko je napravio koje

promene, kada su ih napravili, i vratiti se na prethodne verzije koda ako je potrebno.

- **Zajednički rad:** Obe platforme podržavaju zajednički rad na projektima. Više developera može istovremeno raditi na različitim delovima koda, a platforme integrišu alate za spajanje i koordinaciju tima.

2. Git: Zajednički jezik za kontrolu verzija

- **Jedinstveni koncept:** Obe platforme se oslanjaju na Git, najpopularniji sistem kontrole verzija. Git koristi koncept "snimki" koda za svaku izmenu, omogućavajući vam da vidite istoriju promena i vratite se na prethodne verzije ako je potrebno.
- **Git komande:** Oba GitLab i GitHub koriste iste osnovne Git komande. To znači da ako ste već naučili Git komande na jednoj platformi, možete ih lako koristiti i na drugoj.

3. Kolaboracija: Alati za timski rad

- **Zahtevi za spajanje (pull requests):** Obe platforme omogućavaju programerima da predlože promene koda (pull requests) na pregled drugim članovima tima pre nego što se spoje sa glavnim kodom. Ovo doprinosi poboljšanju kvaliteta koda i otkrivanju potencijalnih problema.
- **Kopije (forkovi):** Obe platforme podržavaju pravljenje kopija (forkovanje), što programerima omogućava da naprave svoju verziju repozitorijuma na kojoj mogu eksperimentisati i razvijati nove funkcije, bez opasnosti da direktno utiču na glavni kod. Ove kopije (fork) se kasnije mogu spojiti sa glavnim repozitorijumom kroz zahteve za spajanje (pull requests).
- **Praćenje problema (issue tracking):** Obe platforme nude sistem za praćenje problema (issue tracking). To omogućava developerima i timu da prijavljuju, prate i rešavaju probleme i zadatke povezane sa projektom.
- **Diskusije oko koda:** Obe platforme omogućavaju diskusije oko koda, što omogućava developerima da razgovaraju o specifikacijama koda, rešavanju problema i koordinaciji razvoja.

4. Zajednica: Veliki bazen developera za pomoć i resurse

- **Aktivne zajednice:** Obe platforme imaju velike i aktivne zajednice developera. To znači da možete lako pronaći pomoć i resurse online, uključujući dokumentaciju, tutorijale, i forume za rasprave.
- **Resursi u okviru platforme:** Obe platforme nude i svoje vlastite resurse za učenje i podršku, kao što su dokumentacija, vodiči, i video tutorijali.

Iako ove sličnosti omogućavaju developerima i timovima efikasan rad na obe platforme, treba imati na umu i razlike između GitLab-a i GitHub-a kako biste mogli doneti informisanu odluku o tome koja platforma najbolje odgovara vašim specifičnim potrebama.

Razlike

Svrha

Iako oba imaju centralnu ulogu u kontroli verzija, GitLab i GitHub imaju različite primarne svrhe, što dovodi do značajnih razlika u njihovim funkcionalnostima i pogodnosti za različite tipove projekata i timova.

GitLab: Široka DevOps platforma

Primarna svrha: GitLab je šira DevOps platforma koja nudi kontrolu verzija kao jednu od svojih osnovnih funkcija. Pored upravljanja kodom, GitLab nudi i niz dodatnih alata i funkcionalnosti za ceo DevOps životni ciklus, uključujući:

- **Planiranje projekata:** Definisanje i upravljanje zadacima, planiranje sprintova, kanban table za vizuelni prikaz toka posla.
- **CI/CD pipeline-ovi:** Automatizacija procesa gradnje, testiranja i implementacije koda.
- **Upravljanje infrastrukturom:** Provizionisanje i upravljanje serverima, kontejnerima i drugim resursima.
- **Monitoring i bezbednost:** Praćenje performansi aplikacija, identifikacija i rešavanje bezbednosnih problema.

- **Wiki i dokumentacija:** Platforma za zajedničko kreiranje i održavanje dokumentacije projekta.

Prednosti:

- **Centralizovana platforma:** Omogućava organizacijama i timovima da centralizuju sve svoje DevOps alate i procese na jednom mestu.
- **Povećana efikasnost:** Automatizacija i integracija DevOps alata poboljšavaju efikasnost i brzinu razvoja i implementacije.
- **Smanjeni troškovi:** Konvergencija alata na jednoj platformi može smanjiti troškove licenciranja i održavanja.

Mane:

- **Učenje i usvajanje:** Širok spektar funkcionalnosti može zahtevati veće ulaganje u učenje i usvajanje od strane timova.
- **Može biti prevelik za pojedinačne projekte:** Obim funkcionalnosti može biti prekomeran i nepotreban za manje ili jednostavnije projekte.

GitHub: Fokus na kontrolu verzija i hosting koda

Primarna svrha: GitHub je primarno fokusiran na kontrolu verzija i hosting koda. Pruža osnovne funkcionalnosti za:

- **Skladištenje koda:** Hostovanje izvornog koda projekta i praćenje promena.
- **Kolaboracija:** Omogućavanje pull requestova, forkovanja, praćenja problema (issue tracking) i diskusija oko koda za timski rad.
- **Automatizacija:** GitHub Actions za automatizaciju gradnje, testiranja i implementacije koda.
- **Bezbednost:** Funkcije za skeniranje bezbednosnih nedostataka u kodu i zaštitu od zlonamernog koda.
- **Registry paketa:** Platforma za skladištenje i deljenje privatnih paketa koda.

Prednosti:

- Jednostavnost i intuitivnost: Lako za učenje i korišćenje, idealno za individualne programere i male timove.
- Velika i aktivna zajednica: Široka baza korisnika i razvijen ekosistem omogućava lako pronalaženje pomoći i resursa.
- Fokus na razvoj softvera: Glavni fokus na razvoj softvera čini ga pogodnim za projekte fokusirane na kod.

Mane:

- Ograničene DevOps funkcionalnosti: Ne nudi istu širinu DevOps alata i integracija kao GitLab.
- Visoki troškovi za privatne repozitorijume: Besplatni plan ima ograničene funkcionalnosti, a privatni repozitorijumi mogu biti skupi za veće timove i organizacije.

Razlike u funkcionalnostima: GitLab i GitHub

Iako oba imaju zajedničke osnovne funkcionalnosti za kontrolu verzija i saradnju, GitLab i GitHub se razlikuju u dodatnim funkcionalnostima koje nude, što ih čini prikladnim za različite potrebe.

GitLab: Širok spektar DevOps alata

GitLab se, pored kontrole verzija, fokusira i na druge aspekte razvoja softvera, nudeći širok spektar DevOps alata:

- Kanban tabla: Vizuelni prikaz toka zadataka, omogućavajući timu da prati i upravlja zadacima projekta, prioritetima i statusom.
- Wiki: Platforma za zajedničko kreiranje i održavanje dokumentacije projekta, omogućavajući centralizovano skladište za informacije, tutorijale i druge relevantne sadržaje.
- CI/CD pipeline: Automatizacija procesa gradnje, testiranja i implementacije koda. Definišete korake i automatski ih pokrećete prilikom svakog izvršenog pull requesta ili promene koda.

- Korisničke grupe i kontrola pristupa: Omogućava definisanje grupa korisnika sa različitim nivoima pristupa i permisijama za projekte i resurse.

Prednosti GitLab funkcionalnosti:

- Povećana efikasnost: Automatizacija i integracija alata omogućava brži i efikasniji razvoj i implementaciju.
- Centralizacija: Smanjuje potrebu za korišćenjem više zasebnih alata za različite DevOps faze.
- Bolja organizacija i vidljivost: Kanban tabla i Wiki poboljšavaju organizaciju i vidljivost toka rada i projekta.

GitHub: Fokus na saradnju oko koda i dodatne funkcije

GitHub, iako primarno fokusiran na kontrolu verzija, nudi i dodatne funkcionalnosti vezane za razvoj softvera:

- GitHub Actions: Automatizacija procesa gradnje, testiranja i implementacije koda, slično GitLab CI/CD pipeline-u, ali sa manje opcija i fleksibilnosti.
- Security features: Funkcije za skeniranje sigurnosnih nedostataka u kodu i zaštitu od malicioznog koda, što pomaže u otkrivanju i prevenciji potencijalnih sigurnosnih problema.
- Package Registry: Platforma za privatno skladištenje i deljenje paketa koda, omogućava developerima da dele i ponovno koriste kod unutar organizacije na siguran način.

Prednosti GitHub funkcionalnosti:

- Pojednostavljenje razvoja: Automatizacija gradnje i testiranja koda poboljšava efikasnost.
- Sigurnost: Skeniranje sigurnosnih nedostataka pomaže u zaštiti aplikacija od sigurnosnih rizika.
- Kod ponovljivosti: Package Registry omogućava efikasno ponovno korišćenje koda u više projekata.

Zaključak:

Izbor platforme zavisi od vaših potreba. GitLab je bolji izbor za DevOps timove i kompleksnije projekte koji zahtevaju širok spektar alata i automatizacije. GitHub je bolji izbor za manje timove i projekte fokusirane na razvoj softvera, gde je primarna potreba kontrola verzija, saradnja i dodatne funkcionalnosti za automatizaciju i sigurnost.

Razlike u dostupnosti: GitLab-a naspram GitHub-a

Takođe, i u pogledu dostupnosti, GitLab i GitHub imaju različite modele i mogućnosti, što utiče na njihovu prilagođenost različitim budžetima i potrebama:

GitLab:

- **Besplatna verzija sa mogućnošću samo-hostovanja:** GitLab nudi besplatnu verziju sa otvorenim kodom koju kompanije mogu instalirati i hostovati na sopstvenim serverima. Ovo omogućava organizacijama potpunu kontrolu nad platformom i podacima, ali zahteva dodatne resurse za implementaciju, održavanje i administraciju.
- **Plaćena cloud verzija:** GitLab takođe nudi plaćene cloud verzije sa različitim nivoima planova i funkcionalnosti. Ove verzije se hostuju na GitLab infrastrukturi, pa organizacije ne moraju brinuti o održavanju servera, ali su u obavezi da plaćaju mesečnu pretplatu.

Prednosti GitLab dostupnosti:

- **Fleksibilnost:** Kompanije mogu birati između verzije sa samo-hostovanjem i cloud verzije, u skladu sa svojim potrebama i resursima.
- **Smanjeni troškovi:** Besplatna verzija je dobra opcija za manje timove i projekte sa ograničenim budžetom.
- **Puna kontrola:** Verzija sa samo-hostovanjem pruža organizacijama potpunu kontrolu nad platformom i podacima.

Mane GitLab dostupnosti:

- **Složenost verzije sa samo-hostovanjem:** Implementacija i održavanje verzije GitLab-a sa samo-hostovanjem zahteva stručno znanje i dodatne resurse.

- Ograničene funkcionalnosti u besplatnoj verziji: Besplatna verzija ima manje funkcionalnosti u poređenju sa plaćenim cloud verzijama.

GitHub:

- Besplatni plan: GitHub nudi besplatni plan sa ograničenim funkcionalnostima, uključujući javne repozitorijume (ne privatne), ograničen prostor za skladištenje i ograničene funkcije saradnje.
- Plaćeni planovi: GitHub nudi više plaćenih planova sa povećanim prostorom za skladištenje, privatnim repozitorijumima, naprednim alatima za saradnju i dodatnim funkcionalnostima poput analize koda i skeniranja sigurnosnih nedostataka.

Prednosti GitHub dostupnosti:

- Jednostavnost: Besplatni plan je lako postaviti i koristiti bez dodatnih troškova za male projekte.
- Širok izbor plaćenih planova: Različiti nivoi plaćenih planova omogućavaju odabir plana koji odgovara potrebama i budžetu.
- Velika zajednica: Besplatni plan omogućava pristup velikoj zajednici razvojnih programera i resursima.

Mane GitHub dostupnosti:

- Ograničena funkcionalnost besplatnog plana: Besplatni plan može biti nedovoljan za profesionalne ili veće projekte koji zahtevaju privatnost i dodatne funkcije.
- Troškovi za privatne repozitorijume: Privatni repozitorijumi i napredne funkcije zahtevaju plaćenu pretplatu.

Zaključak:

Izbor platforme zavisi od vaših finansijskih mogućnosti, potrebe za samostalovanjem i željenih funkcionalnosti. Besplatni planovi GitLab-a i GitHub-a su odlični za individualne projekte i male timove sa ograničenim budžetom. Za veće projekte i profesionalni razvoj, plaćene verzije obe platforme nude veće mogućnosti i fleksibilnost.

Koju platformu izabrati?

Odluka između GitLab-a i GitHub-a ne mora biti strogo crno-bela. Obe platforme su moćne i imaju svoje prednosti i slabosti. Ključ je u razumevanju vaših specifičnih potreba i prioriteta za donošenje informisane odluke.

Kada je GitLab bolji izbor:

- **Veliki i kompleksni projekti:** Ako radite na velikim i složenim projektima koji zahtevaju DevOps alate kao što su CI/CD pipeline-ovi, upravljanje infrastrukturom i planiranje projekata, GitLab nudi sveobuhvatan set alata i funkcionalnosti.
- **Centralizacija:** Ako želite centralizovanu platformu za sve DevOps procese, GitLab vam omogućava da integrišete sve DevOps alate na jednom mestu, poboljšavajući efikasnost i preglednost.
- **Samo-hostovanje:** Ako vaša organizacija zahteva ili preferira potpunu kontrolu nad platformom i podacima, GitLab vam daje mogućnost samo-hostovanja na vašim sopstvenim serverima.
- **Veliki timovi:** Za velike timove sa različitim potrebama pristupa, GitLab nudi detaljnije kontrole pristupa i korisničke grupe za upravljanje dozvolama projekata.

Kada je GitHub bolji izbor:

- **Projekti fokusirani na razvoj softvera:** Ako je primarni fokus vašeg projekta razvoj softvera i tražite robustne funkcionalnosti za kontrolu verzija i saradnju oko koda, GitHub je odlična opcija.
- **Mali i srednji timovi:** Za manje timove koji traže jednostavnu i intuitivnu platformu za kontrolu verzija i saradnju, GitHub je lakši za učenje i korišćenje.
- **Projekti otvorenog koda:** Ako je vaš projekt otvorenog koda, GitHub ima veliku i aktivnu zajednicu developera, što vam omogućava lakše pronalaženje pomoći i resursa.
- **Ograničen budžet:** Besplatni plan GitHub-a je odlična ponuda za manje projekte i individualne developere sa ograničenim budžetom.

Dodatne razlike za razmatranje:

- **Cene:** GitLab i GitHub imaju različite modele cena. GitLab nudi besplatnu verziju sa mogućnošću samo-hostovanja, ali plaćene cloud verzije mogu biti skuplje od GitHub-ovih plaćenih planova.
- **Korisničko sučelje:** Iako su oba smatrana intuitivnim, neki korisnici smatraju da je GitLab-ovo sučelje malo složenije od GitHub-ovog.
- **Zajednica:** Obe platforme imaju velike i aktivne zajednice, ali GitHub ima značajno veću bazu korisnika.

Zaključak:

Izbor između GitLab-a i GitHub-a zavisi od specifičnih potreba i prioriteta. Pažljivo razmotrite faktore kao što su veličina i fokus vašeg projekta, potrebe za samo-hostovanjem, budžet i veličina tima. Uzimajući u obzir ove razlike, možete doneti informisanu odluku o tome koja platforma će najbolje podržati vaše DevOps ili razvojne potrebe.