

المقدمة

ما هو نظام التحكم في الإصدار؟

بشكل أساسي، هو نظام يسمح لك بتسجيل التغييرات على الملفات بمرور الوقت، وبالتالي، يمكنك عرض إصدارات معينة من هذه الملفات لاحقاً.

يتتبع نظام التحكم في الإصدار (VCS)، تاريخ التغييرات حيث يتعاون الأشخاص والفرق في المشاريع معاً.

مع تطور المشروع، يمكن للفرق إجراء الاختبارات وإصلاح الأخطاء والمساهمة في رمز جديد مع الثقة بإمكانية استرداد أي إصدار في أي وقت.

يمكن للمطورين مراجعة محفوظات المشروع لمعرفة:

التغييرات التي تم إجراؤها.

من قام بالتغييرات.

متى تم إجراء التغييرات.

لماذا كانت هناك حاجة للتغييرات.

Git و GitHub ، إذا كنت تعمل في مجال البرمجة، لا بد أنك سمعت بهذا المكان من قبل. المكان الذي يمكنك من خلاله أن تحفظ مشروعك وتشاركه مع الآخرين، كما يمكنك مشاهدة مشاريع الآخرين.

في مناقشة Git vs. GitHub ، يمكن اعتبار أن GitHub هي Git مثل Facebook في الحقيقة.

ماذا يعني ذلك؟

هذا يعني أنه على الرغم من أن Facebook يشبه إلى حد ما قاعدة البيانات المرئية التي تظهر لك عبر الإنترنت..

تم تصميم GitHub كخدمة استضافة مستودع Git.

وما هي بالضبط خدمة استضافة مستودع Git؟

إنها قاعدة بيانات عبر الإنترنت تتيح لك تتبع ومشاركة مشاريع التحكم في إصدار Git خارج الكمبيوتر / الخادم المحلي.

على عكس Git ، فإن GitHub يعتمد على السحابة بشكل حصري.

على عكس Git أيضاً، فإن GitHub هي خدمة ربحية.

يتم تثبيت Git وصيانته على نظامك المحلي local system – (بدلاً من السحابة) ويمنحك سجلاً قائماً بذاته لإصدارات البرمجة المستمرة.

لماذا هي ربحية؟

هذا لأنه، بالإضافة إلى تقديم جميع ميزات ومزايا Git ، يوسع GitHub وظائف Git الأساسية.

إنه يقدم واجهة مستخدم بديهية للغاية وممتلئة بيانياً، ويوفر للمبرمجين أدوات تحكم وإدارة مهام مدمجة.

يمكن تنفيذ ميزات إضافية عبر خدمة GitHub Marketplace.

نظراً لأن GitHub يستند إلى مجموعة النظراء (Git repositories) ، يمكن الوصول إلى مستودعات Git الخاصة بالفرد عن بُعد بواسطة أي شخص مخول له، من أي جهاز كمبيوتر، فقط أنت بحاجة إلى إتصال بالإنترنت.

ببساطة، Git هو نظام تحكم في الإصدار يتيح لك إدارة محفوظات شفرة المصدر وتتبعها.

GitHub، يمكن اعتبارها استضافة على السحابة تقوم بإدارة مستودعات Git.

إذا كان لديك مشاريع مفتوحة المصدر تستخدم Git ، فإن GitHub مصمم لمساعدتك على إدارتها بشكل أفضل.

يأخذ GitHub الأمور أبعد قليلاً من Git ، حيث يقدم المزيد من الوظائف والموارد، بالإضافة إلى مكان أو بمعنى آخر مستودع على الإنترنت لتخزين المشاريع والتعامل معها.

ما هو GitHub ؟

GitHub كما قلت سابقاً git هي تقنية تُستخدم مع بعض أوامر، أي أن git ليس لديه واجهة مستخدم

GitHub هو مستودع عبر الإنترنت يستخدم تقنية git في عملياتهم وتحقيق ضوابط الإصدار مع وظائف أخرى مثل تتبع الأخطاء , وإدارة المشاريع • وإدارة دعم التذاكر .. إلخ.

بعبارة أخرى , بعد GitHub مجتمعا مبنياً على تقنية git مع واجهة مستخدم ووظائف أخرى من قبل شركة خارجية أخرى , وهو في الواقع منتج معملوك من

قبل شخص ما أو مجموعة معينة استناداً إلى تقنية git حيث يكون git مفتوح المصدر.

يسمح GitHub للمطورين باستضافة ملفاتهم في Git repository حتى يتمكن الآخرون من المشاركة أو التعاون في المشاريع معهم , سواء كانت مشاريع

مفتوحة للمساهمة العامة (مفتوحة المصدر) أو محصورة ل team محدد للعمل في مشروع خاص.

بما أن GitHub قائم على السحابة , يمكن الوصول إلى مستودعات git الفردية عن بعد من قبل أي شخص لديه الصلاحيات Permissions من أي جهاز كمبيوتر

, في أي مكان في العالم (بشرط أن يكون لديه اتصال بالإنترنت).

وللحديث عن الاختلافات والتشابهات بين GitLab و GitHub , نحتاج إلى النظر إلى ما وراء مستودعات الكود وإلقاء نظرة على العملية بأكملها , كلا من GitLab و GitHub عبارة عن مستودعات Git على شبكة الإنترنت

ما هو مستودع Git ؟ الهدف من Git هو إدارة مشاريع تطوير البرمجيات وملفاتها , لأنها تتغير بمرور الوقت. يخزن Git هذه المعلومات في بنية بيانات تسمى المستودع. حيث يعتبر المستودع مكان مركزي يقوم المطورون بتخزين مشاريع الويب ومشاركتها واختبارها والتعاون فيها , أما بالنسبة للاختلافات الرئيسية وأوجه التشابه بين GitLab و GitHub , باستخدام GitLab , يمكنك تعيين أذونات الأشخاص وتعديلها وفقاً لأدوارهم. أما في GitHub , يمكنك تحديد ما إذا كان شخص ما سيحصل على حق الوصول للقراءة أو الكتابة إلى المستودع. يوفر GitLab , بالإضافة إلى GitHub , أداة تعقب بسيطة للمشكلات تتيح لك تغيير الحالة والمسؤول عن مشكلات متعددة في نفس الوقت. أيضاً عندما يتعلق الأمر بتصدير البيانات , يبدو أن GitLab يقوم بعمل قوي جداً , حيث يوفر لك القدرة على تصدير مشاريعك بالإضافة إلى المزيد من البيانات عن المشروع , أما بالنسبة ل GitHub فهو أكثر تقييداً عندما يتعلق الأمر بميزات التصدير لمستودعات GitHub الحالية , بناءً على هذه التشابهات والاختلافات يمكنك اختيار أحدهما , ولكن أنصحك ب GitHub لأنه لديه مجتمع كبير جداً وبالتالي من السهل حل المشاكل التي ربما تواجهك

ملخص للفروقات :1. Open source : غيت لاب مفتوحة المصدر , أما غيتهب ليست كذلك. 2. Private Repository : غيت لاب يسمح للمستخدمين بإنشاء مستودع خاص مجاني , وغيتهب يسمح بذلك أيضاً لكن بحد أقصى 3 متعاونين. 3. Navigation : لاب يوفر ميزة التنقل في المستودع , وهب كذلك. 4. Project Analysis : يوفر للمستخدم رؤية مخططات تطوير المشروع , أما في هب فهي غير متوفرة. 5. Advantages : لاب هو تطبيق سحابي آمن جداً ومفتوح المصدر , هب يتم استخدامه لمشاركة العمل أمام الجمهور , ويساعدنا في إنشاء توثيق منظم للمشروع. 6. Disadvantages : لديه العديد من الأخطاء , هب لديه مستودع محدود ويدعم فقط git وبشكل أساسي يتمثل الاختلاف الرئيسي بين GitHub و GitLab في النظام الذي تقدمه كل فلسفة. حيث يتمتع GitHub بانتشار أكبر ويركز بشكل أكبر على أداء البنية التحتية , بينما يركز GitLab بشكل أكبر على تقديم نظام قائم على الميزات مع نظام أساسي مركزي ومتكامل لمطوري الويب.

gitlab يتم منح الوصول وفقاً للدور github . وفقاً لمهمة القراءة أو الكتابة , يتم منح حق الوصول

gitlab لا توجد تعليقات اما github يمكن عرض وتحرير جميع التعليقات اي ان github يوفر بيئة تفاعلية ضخمة

تتبع التعليقات

بخلاف GitLab , يوفر GitHub دعمًا كاملاً لتاريخ تحديثات التعليقات. يمكنك عرض وتحرير تاريخ التعليق. بالإضافة إلى حذف أو إزالة التفاصيل الحساسة من سجل تعديل التعليق.

مجتمع

صنع GitHub اسمًا لنفسه في عالم المطورين. يستخدم المطورون من جميع أنحاء العالم GitHub ومجموعة GitHub النشطة للغاية والمكونة من ملايين المطورين المشاركين والراغبين في التحدث عن القضايا هي المسؤولة بشكل أساسي عن نجاحها. من ناحية أخرى ، قام GitLab ببعض الأشياء الرائعة مثل استضافة الأحداث والتجمعات المجتمعية التي تجمع المساهمين مفتوحين المصدر معًا.

Advantages and Disadvantages of GitLab

Advantages

GitLab is a free and open-source license.

Self-hosting is available on all plans.

It provides package distribution services.

The CI/CD lifecycle is supported.

Here, Codes are simple to maintain.

Disadvantages

The interface is relatively slow.

There are a lot of bugs in it.

Upgrade procedure that is problematic

Some enterprise-level features are missing.

Repositories have a number of common issues

Advantages and Disadvantages of GitHub

Advantages

GitHub is Easy-to-use and also makes sharing easy.

The open-source codebase has an amazing structure.

In the repository framework, a quick search is possible.

It is easy to integrate with third-party providers.

TFS, HG, and SVN are also supported.

GitHub has both free and paid services.

Disadvantages

It doesn't have a great API development.

For those looking for a private repository, GitHub is a little expensive.

GitHub does not contain many functions.

It isn't completely free.

And GitHub also has space limitations, as single files cannot reach 100MB, and repositories are restricted to 1GB in the free version.

على الرغم من وجود منافسين قويين، يعتبر GitHub منصة إدارة ومشاركة ملفات الشفرة المصدرية الأشهر والأكثر شعبية، حيث يستخدمها ملايين المطورين حول العالم. بالإضافة إلى ذلك، تتميز GitHub بالعديد من الميزات المميزة، مثل إمكانية العمل على المشاريع بشكل مشترك، والتعليق على الشفرة المصدرية، وإنشاء المشاريع الخاصة والعامة، وتوفير الدعم الفني الجيد. كما يتميز GitHub بالتكامل مع العديد من أدوات التطوير الأخرى، مثل Travis CI و CircleCI وغيرها، مما يجعله الخيار الأمثل للمطورين المحترفين. لذلك، يمكن القول إن GitHub هي الأفضل بين التقنيات المنافسة لها.

1- GitLab:

يعد GitLab منافسًا قويًا لـ GitHub، حيث يتميز بالعديد من الميزات الجيدة التي تجعله يتفوق في بعض الجوانب على GitHub. على سبيل المثال، يوفر GitLab خدمات CI/CD المدمجة، وهو ما يعني أنه يمكنك إنشاء واختبار تطبيقاتك بسهولة، وبدون الحاجة إلى الاعتماد على أدوات خارجية.

2- Bitbucket:

تعتبر Bitbucket منافسًا آخر لـ GitHub، والذي يتميز بعدد من الميزات المميزة، مثل إمكانية إنشاء مستودعات خاصة مجانية، والتي يمكن استخدامها لتخزين الشفرة المصدرية الخاصة بك بشكل آمن. كما يتميز Bitbucket أيضًا بدعمه للغات المتعددة، بما في ذلك Python و Ruby و Java وغيرها.

3- SourceForge:

يتفوق SourceForge في بعض الميزات، مثل إمكانية استضافة المشاريع الحرة والمفتوحة المصدر، والتي يمكن استخدامها لتخزين ملفات الشفرة المصدرية الخاصة بك. كما يتميز SourceForge أيضًا بدعمه لعدد كبير من اللغات البرمجية المختلفة، بما في ذلك Java و C++ و Python و Ruby وغيرها.

SourceForge و Github هما منصتان شهيرتان لإدارة وتخزين رموز البرمجيات المصدريّة. وهما يقدمان مجموعة متنوعة من الخدمات المتعلقة بإدارة التطوير والتعاون والنسخ الاحتياطي للبرمجيات. ولكن هناك بعض الاختلافات المهمة بينهما، وفيما يلي مقارنة مفصلة بين SourceForge و Github:

1- الملكية:

يملك Github من قبل Microsoft ، بينما يملك SourceForge من قبل Slashdot Media.

2- السعر:

يتيح SourceForge تخزين الرموز المصدريّة مجاناً لكل المستخدمين، ولكن يوجد تكلفة للحصول على ميزات إضافية. أما Github فيوجد اشتراكات مدفوعة للمستخدمين للحصول على ميزات إضافية. ويتم تحديد السعر بناءً على عدد المستخدمين وحجم التخزين.

3- اللغات والتكنولوجيا:

يدعم Github مجموعة واسعة من اللغات والتقنيات مثل جافا ، C ++ ، Ruby ، Python ، Node.js ، PHP، وغيرها الكثير. بينما يدعم SourceForge أيضاً العديد من اللغات والتقنيات مثل Java و C++ و Python وغيرها، ولكنها لا تدعم لغات البرمجة الحديثة مثل Go و Rust وغيرها.

4- الأمان:

يوفر Github خيارات الأمان الأساسية المعروفة مثل HTTPS ، ويوفر SourceForge خيار الوصول إلى الرموز المصدريّة بواسطة مفتاح SSH أو كلمة مرور.

5- الدعم:

يتميز SourceForge بدعم قوي للمشاريع ذات الحجم الكبير، ويوفر خيارات إدارة المشروع والصلاحيات بشكل أفضل من Github. يدعم Github أيضاً فرق العمل الكبيرة، لكن SourceForge يتميز بدعمه للمشاريع ذات الحجم الكبير.

6- الأدوات:

توفر Github العديد من الأدوات لإدارة البرمجيات مثل Travis CI للتكامل المستمر و CircleCI وغيرها. يوفر SourceForge أيضاً مجموعة متنوعة من الأدوات المتكاملة، مثل Apache Allura للتكامل المستمر.

7- الجماعة:

يتميز Github بجماعة مستخدمين واسعة ونشطة، حيث يستخدمها مطورون ومستخدمون في جميع أنحاء العالم. يعد SourceForge أقل شهرة من Github ، ولكنه يتمتع بجماعة مستخدمين نشطة ومتنامية.

8-التكامل:

يتميز Github بإمكانية التكامل السهل مع العديد من الأدوات والخدمات الأخرى، مثل Slack وIRA وTrello وغيرها. بينما يوفر SourceForge أيضًا تكامل مع العديد من الأدوات والخدمات الأخرى، ولكنه يمكن أن يكون أقل قدرة على التكامل مع بعض الأدوات.

في النهاية، تعتبر Github أكثر شهرة وانتشارًا من SourceForge، ولكن يمكن اختيار المنصة المناسبة بناءً على متطلبات المشروع والميزات المطلوبة وسعر الاشتراك والمنصة المفضلة للفريق. يجب النظر في الجوانب المختلفة التي تم توضيحها في المقارنة ومقارنتها مع احتياجات المشروع واختيار المنصة التي تلبي متطلبات المشروع بشكل أفضل.

مقارنة مفصلة بين github , sourceforge مع اعطاء الافضلية لgithub

SourceForge و Github هما منصتان شهيرتان لإدارة وتخزين رموز البرمجيات المصدريّة، ولكن هناك بعض الاختلافات المهمة بينهما، وفيما يلي مقارنة مفصلة بين SourceForge و Github مع إعطاء الأفضلية لـ Github:

1-الملكية:

يملك Github من قبل Microsoft ، ولديها تاريخ طويل في تطوير البرمجيات، وقد اكتسبت الشعبية بسرعة في مجتمع المطورين. بينما يملك SourceForge من قبل Slashdot Media ، ولكنها لم تحقق نفس المستوى من الانتشار والشعبية.

2-السعر:

يتيح SourceForge تخزين الرموز المصدريّة مجانًا لكل المستخدمين، ولكن يوجد تكلفة للحصول على ميزات إضافية. أما Github فيوجد اشتراكات مدفوعة للمستخدمين للحصول على ميزات إضافية، ولكنه يوفر خيارات أكثر وأفضل فيما يتعلق بخدمات الاستضافة والتكامل.

3-اللغات والتكنولوجيا:

يدعم Github مجموعة واسعة من اللغات والتقنيات مثل جافا ، ++C ، Ruby ، Python ، Node.js ، PHP، وغيرها الكثير. بينما يدعم SourceForge أيضًا العديد من اللغات والتقنيات مثل Java و ++C و Python وغيرها، ولكنها لا تدعم لغات البرمجة الحديثة مثل Go و Rust وغيرها.

4-الأمان:

يوفر Github خيارات الأمان الأساسية المعروفة مثل HTTPS ، ويوفر SourceForge خيار الوصول إلى الرموز المصدريّة بواسطة مفتاح SSH أو كلمة مرور، ولكن Github تتمتع بمستويات أفضل من الأمان وأدوات الحماية.

-5الدعم:

يتميز Github بدعم قوي للمشاريع ذات الحجم الكبير، ويوفر خيارات إدارة المشروع والصلاحيات بشكل أفضل من SourceForge. ويتميز Github بمجتمع مطورين نشط وداعم، ويوفر دعماً للمستخدمين بشكل أفضل من SourceForge.

-6الادوات:

توفر Github العديد من الأدوات لإدارة البرمجيات مثل Travis CI للتكامل المستمر و CircleCI وغيرها، ويتمتع Github بمجموعة كبيرة من الأدوات المتكاملة والخدمات التي تتيح للمطورين العمل بكفاءة أكبر. بينما يوفر SourceForge أيضاً مجموعة متنوعة من الأدوات المتكاملة، مثل Apache Allura للتكامل المستمر، ولكن Github توفر مزيداً من الأدوات والخدمات المتكاملة.

-7الجماعة:

يتميز Github بجماعة مستخدمين واسعة ونشطة، حيث يستخدمها مطورون ومستخدمون في جميع أنحاء العالم. ويتمتع Github بشعبية عالية في مجتمع المطورين، ويتم تحديثها بانتظام وتطويرها بشكل دائم. بينما يعد SourceForge أقل شعبية وانتشاراً، ولكنه يتمتع بجماعة مستخدمين نشطة ومتنامية.

-8التكامل:

يتميز Github بإمكانية التكامل السهل مع العديد من الأدوات والخدمات الأخرى، مثل Slack و JIRA و Trello وغيرها، ويتمتع Github بمزيد من الدعم والمرونة في التكامل مع الأدوات الخارجية. بينما يوفر SourceForge أيضاً تكامل مع العديد من الأدوات والخدمات الأخرى، ولكنه يمكن أن يكون أقل قدرة على التكامل مع بعض الأدوات.

بناءً على ما سبق، يتمتع Github بمزايا أكثر وأفضل من SourceForge ، وبالتالي يعتبر الخيار الأفضل لمعظم المطورين والفرق البرمجية. توفر Github مجموعة واسعة من الأدوات والخدمات المتكاملة والأمان والدعم الفني والتكامل السهل مع الأدوات الخارجية والجماعة النشطة. وبالتالي، يمكن القول إن Github هي الخيار الأفضل لإدارة وتخزين رموز البرمجيات المصدرية.

Github و Bitbucket هما منصتان شهيرتان لإدارة وتخزين رموز البرمجيات المصدرية، ولكن هناك بعض الاختلافات المهمة بينهما، وفيما يلي مقارنة مفصلة بين Github و Bitbucket مع إعطاء الأفضلية لـ: Github

-1الملكية:

يملك Github من قبل Microsoft ، ولديها تاريخ طويل في تطوير البرمجيات، وقد اكتسبت الشعبية بسرعة في مجتمع المطورين. بينما يملك Bitbucket من قبل Atlassian ، وهي شركة تقنية معروفة، ولكنها لم تحقق نفس مستوى الانتشار والشعبية.

-2السعر:

يُتيح Github تخزين الرموز المصدريّة مجاناً للمستخدمين العامّة، ولكن يوجد تكلفة للحصول على ميزات إضافية. أما Bitbucket فيوجد اشتراكات مدفوعة للمستخدمين للحصول على ميزات إضافية، ولكنها توفر خيارات أقل فيما يتعلق بخدمات الاستضافة والتكامل.

-3 اللغات والتكنولوجيا:

يدعم Github مجموعة واسعة من اللغات والتقنيات مثل جافا ، ++C ، Ruby ، Python ، Node.js ، PHP ، وغيرها الكثير. بينما يدعم Bitbucket أيضا العديد من اللغات والتقنيات مثل Java و ++C و Python وغيرها، ولكنها لا تدعم لغات البرمجة الحديثة مثل Go و Rust وغيرها.

-4الأمان:

يوفر Github خيارات الأمان الأساسية المعروفة مثل HTTPS ، ويوفر Bitbucket خيار الوصول إلى الرموز المصدريّة بواسطة مفتاح SSH أو كلمة مرور، ولكن Github تتمتع بمستويات أفضل من الأمان وأدوات الحماية.

-5الدعم:

يتميز Github بدعم قوي للمشاريع ذات الحجم الكبير، ويوفر خيارات إدارة المشروع والصلاحيات بشكل أفضل من Bitbucket. ويتميز Github بمجتمع مطورين نشط وداعم، ويوفر دعماً للمستخدمين بشكل أفضل من Bitbucket.

-6الأدوات:

توفر Github العديد من الأدوات لإدارة البرمجيات مثل Travis CI للتكامل المستمر و CircleCI وغيرها، ويتمتع Github بمجموعة كبيرة من الأدوات المتكاملة والخدمات التي تتيح للمطورين العمل بكفاءة أكبر. بينما يوفر Bitbucket أيضاً مجموعة متنوعة من الأدوات المتكاملة، مثل Bamboo للتكامل المستمر، ولكن Github توفر مزيد أفضل في هذا الجانب.

بناءً على هذه المقارنة، يمكن القول إن Github يتفوق على Bitbucket في العديد من الجوانب، مثل السعر والأمان والدعم والأدوات، ويدعم مجموعة واسعة من اللغات والتقنيات. وبالإضافة إلى ذلك، فإن Github يتمتع بشعبية كبيرة في مجتمع المطورين ويوفر خيارات إدارة المشروع والصلاحيات بشكل أفضل. لذلك، يمكن القول إن Github هي المنصة المفضلة لإدارة وتخزين رموز البرمجيات المصدريّة.