**جامعة دمشق**

**كُلية الهندسة المعلوماتية**

GitHub

2023

مَشروع مَهارات التّواصُل

إعداد الطُّلاب:

علي صلاح حمدان 20610 علي أحمد يزبك 20621

علي محمد زيتون 20613 عمر مصطفى حميدي 20638

علي سليمان سليمان 20614 مهران مُهيب أبو دقة 21036

علي إبراهيم قاسم 20618 المعتصم بالله خالد عليان 20069

**الدكتور: علاء**

الفهرس

[المُقدمة 3](#_Toc136176074)

[الأهداف الرئيسية 3](#_Toc136176075)

[خطوات إنشاء حساب 4](#_Toc136176076)

[الخدمات التي يُقدِّمُها 6](#_Toc136176079)

[الميّزات 8](#_Toc136176080)

[الأمان 9](#_Toc136176081)

[الفرق بين Git وSourceForge 10](#_Toc136176082)

[التكاليف 10](#_Toc136176083)

# المُقدمة

: GitHub‏ هي شركة تابعة لمايكروسوفت توفر استضافة لتطوير البرامج والتحكم في الإصدار باستخدام غيت. توفر الشركة وظائف التحكم في الإصدار الموزع وإدارة كود المصدر (SCM) لـ «غيت»، بالإضافة إلى ميزاته الخاصة. يوفر النظام التحكم في الوصول والعديد من ميزات التعاون مثل تتبع الأخطاء وطلبات الميزات البرمجية وإدارة المهام والتركيب المتواصل ومواقع ويكي لكل مشروع غيت هاب. يقع مقرها الرئيسي في كاليفورنيا، وهي شركة تابعة لشركة مايكروسوفت منذ عام 2018 ..

يعد موقع Github من المواقع المهمة للمبرمجين حول العالم , فهو يحتوي على كمية مشاريع مفتوحة المصدر ومنها مشاريع ضخمة ومهمة في عالم التقنية والبرمجة بشكل عام

تأسس GitHub على يد أربعة أشخاص هم:  
Tom Preston-Werner, Chris Wanstrath, Scott Chacon, P. J. Hyett

لنستطيع فهم موقع Github علينا أولاً فهم الاسم , فالاسم ينقسم لجزأين Git-Hub و جيت Git هي تقنية متبعة لادارة الاصدارات و العمل المشترك على مشاريع البرمجة او الأكواد بشكل عام , اما Hub فترجمتها الحرفية المركز الرئيسي , و يقصد بها هنا مركز تجميع هذه الاصدارات .

وفي عودة لجيتهاب , ستجد ان مشاريعك التي قمت بمشاركتها باستخدام تقنية Git ستحتاج الى مكان لتبقى فيه , وهنا يأتي دور جيت هاب ,جيتهاب هو موقع لتجميع المشاركات والمشاريع التي يتم عملها باستخدام تقنية Git , وفي واقع الأمر , لا يمكنك رفع مشروع مباشرة الى موقع Github بل عليك ان تكون قد قمت بانشاء المشروع على جهازك الخاص ومن ثم رفعه - عبر سطر الأوامر او برامج مخصصة - الى حسابك في جيت هاب .

تقدم غيت هاب خدماتها الأساسية مجانًا. خدماتها المهنية والمشاريع الأكثر تقدمًا تكون تجارية. تُستخدم حسابات غيت هاب المجانية بشكل شائع لاستضافة مشاريع مفتوحة المصدر. اعتبارًا من يناير 2019، قدمت غيت هاب مستودعات خاصة غير محدودة لجميع الخطط، بما في ذلك الحسابات المجانية، ولكنها سمحت بما يصل إلى ثلاثة مطورين فقط لكل مستودع مجانًا.بدءًا من 15 أبريل 2020، سمحت الخطة المجانية لعدد غير محدود من المطورين، ولكنها قيدت المستودعات الخاصة بـ 2,000 دقيقة من إجراءات غيت هاب لكل شهر. اعتبارًا من يناير 2020، أعلن غيت هاب عن وجود أكثر من 40 مليون مستخدم وأكثر من 190 مليون مستودع (بما في ذلك ما لا يقل عن 28 مليون مستودع عام)،مما يجعله أكبر مضيف للكود المصدري في العالم.

# الأهداف الرئيسية

1- التعاون والمشاركة: يتيح GitHub للمطورين والمبرمجين التعاون والمشاركة في مشاريعهم البرمجية بسهولة، كما يساعد على تحسين جودة الأكواد وتطويرها بشكل مستمر.

2- الإدارة والمراقبة: يمكن استخدام GitHub لإدارة ومراقبة المشاريع البرمجية، حيث يتيح الموقع أدوات لإدارة الإصدارات والتحكم في عمليات الدمج والتعديلات والإصلاحات في الأكواد.



**3**

3- التوثيق والتعلم: يمكن استخدام GitHub لتوثيق المشاريع البرمجية وتوفير تعليمات وشروحات للمستخدمين، كما يمكن استخدامه للتعلم والتدريب على البرمجة عن طريق الاطلاع على مشاريع مفتوحة المصدر والمشاركة فيها.

4- البحث: يتيح GitHub البحث عن المشاريع البرمجية المفتوحة المصدر والمساهمة فيها، كما يمكن استخدامه للبحث عن حلول للمشاكل التقنية والتعرف على أحدث التقنيات والأدوات في مجال البرمجة.

5- الاستخدام التجاري: يمكن استخدام GitHub في مجال الأعمال والتجارة، حيث يمكن للشركات والمؤسسات استخدامه لإدارة ومراقبة مشاريعهم البرمجية والتعاون مع فرق العمل.

6- التواصل والتبادل: يمكن استخدام GitHub للتواصل والتبادل بين المطورين والمبرمجين، حيث يمكن للمستخدمين الاطلاع على مشاريع الآخرين وتعلم منها وتبادل المعلومات والخبرات.

بشكل عام، يمكن القول أن موقع GitHub يهدف إلى تسهيل عملية التعاون والمشاركة بين المطورين والمبرمجين وتعزيز ثقافة المفتوحة المصدر، كما يهدف إلى تحسين جودة الأكواد وتطويرها بشكل مستمر وتوفير أدوات وموارد للمستخدمين لتسهيل عملية الإدارة والمراقبة والتوثيق والتعلم. كما يعمل على تعزيز التواصل والتبادل بين المطورين والمبرمجين وتشجيع المجتمع البرمجي على الابتكار والتطوير.

# خطوات إنشاء حساب

على محرك البحث جوجل، ثم نختار النتيجة الأولى [GitHub](https://github.com/) أولًا:نبحث عن موقع

ثانيًا: بعد الدخول إلى النتيجة الأولى، تظهر لنا الصفحة الرئيسية للموقع GitHub, ننقر على الزر في أعلى يمين الصفحة الذي يحمل عبارة "sign up" , أي انشاء حساب جيت هاب جديد.

بعد ذلك سيطلب منك إدخال عدد من المعلومات الواحدة تلو الأخرى وهي:

1-البريد الالكتروني الخاص بك: حيث سيصلك بعد إدخاله رسالة لتأكيد الحساب

2-انشاء كلمة مرور خاصة بك: ويفضل الاعتماد على كلمة مرور قوية، كما أن الموقع يشترط بأن تكون كلمة المرور على الأقل 8 حرف لكن بشرط أن تتضمن حروف صغيرة وأرقام.

3-عمل اسم مستخدم مميز: يطلب منك الموقع انشاء اسم مستخدم مميز عن أي حساب آخر، حيث أنه سيظهر عند مشاركة مشاريعك كبداية للرابط الثابت، حيث أن الوسيلة الوحيدة للفصل بين الكلمات هي الفاصلة أو علامة الداش –

بعد ادخال جميع المعلومات السابقة، سيطرح لك سؤال فيما إذا كنت تريد تلقي التحديثات للمنتجات



**4**

عبر الايميل أم لا وذلك بإدخال الحرف y للقبول أو n للرفض .

رابعًا: إثبات أنك إنسان، حيث يعرض لك لغز بسيط،هذا الإجراء يساعد النظام على تجنب البوتات التي تعمل باستمرار على انشاء حسابات GitHub .

خامسًا: سيصلك كود تحقق مكون من 8 أحرف على بريدك الالكتروني، أدخله في حقل التحقق .

سادسًا: اختيار عدد فريق العمل، وهل أنت معلم أم طالب، بعد ذلك تحديد اهتماماتك كما هو موضح في الصورة التالية:

ثم سيطلب بعد ذلك تحديد نوع الحساب مجاني او مدفوع,وسنوضح لكم الفرق بينهما :

هناك بعض الفروقات بين حساب GitHub المجاني و المدفوع، والفرق الرئيسي هو قدرة المستخدم على إنشاء مستودعات خاصة.

في الحساب المجاني، المستخدم يمكنه إنشاء مستودعات عامة فقط، ولا يمكنه تخزين ملفات خاصة بكلمة مرور.

أما الحساب المدفوع، فالمستخدم يمكنه إنشاء مستودعات خاصة وتخزين ملفات خاصة ويمكنه أيضًا الانضمام إلى المشاريع الخاصة التي يتسنى لمالكي هذه المشاريع دعوته.

وهناك بعض الميزات الإضافية التي تحتويها الحساب المدفوع والتي لا تتوفر في الحساب المجاني. لذلك فإن اختيار الحساب يعتمد على الاحتياجات الفردية للمطور.

وفي هذه الصورة نعرض بعض الفروقات بينهما :

عند الانتهاء من جميع هذه الخطوات، ستنتقل إلى لوحة التحكم الخاصة بحسابك، حينها يمكنك تعديل البروفايل، ووضع صورة ومعلومات تعريفية عنك، أو إنشاء مستودعة جديدة ثم رفع المشروع الخاص بك.

# الخدمات التي يُقدِّمُها

1. GitHub Actions: هي خدمة تسمح للمطورين بإنشاء سلاسل عمل (Workflows) تلقائية لتشغيل اختبارات الوحدات ونشر التطبيقات وإطلاق الأحداث والتنبيهات، وتعمل بشكل متكامل مع مستودعات GitHub.

2. GitHub Packages: هي خدمة تسمح للمستخدمين بنشر وإدارة حزم البرمجيات الخاصة بهم، وتوفر خدمة استضافية للتخزين والنشر والإدارة لحزم البرمجيات.

3. GitHub Gist: هي خدمة تسمح للمستخدمين بإنشاء ملفات نصية وأكواد مصدرية صغيرة ومشاركتها مع المستخدمين الآخرين، وهي مفيدة للمطورين الذين يرغبون في مشاركة الأكواد المصدرية الصغيرة أو الحلول المؤقتة أو بعض النصوص.

**5**

4. GitHub Pages: هي خدمة تسمح للمستخدمين بإنشاء مواقع وصفحات ويب مجانية ونشرها باستخدام GitHub، وتوفر هذه الخدمة مساحة تخزين ونطاقات مجانية للمستخدمين.

5. GitHub Desktop: هو تطبيق سطح المكتب الذي يسمح للمستخدمين بإدارة مستودعات GitHub وإجراء عمليات الاستنساخ والاستدعاء والتحميل والتحكم في الإصدارات بشكل سهل وبديهي.

6. GitHub Marketplace: هي متجر يحتوي على مجموعة كبيرة من التطبيقات والخدمات والأدوات التي تساعد المطورين على تحسين عملية التطوير وإدارة المشاريع، ويمكن للمستخدمين تحميل هذه التطبيقات والخدمات والأدوات مباشرة من متجر GitHub.

7. GitHub Education: هو برنامج تعليمي يهدف إلى تعزيز التعليم في مجال تطوير البرمجيات، ويوفر للطلاب والمدرسين والمدارس والجامعات الدعمة اللازمة لتحسين مهارات التطوير والعمل مع GitHub.

8. GitHub Enterprise Server: هي خدمة تسمح للمؤسسات بإنشاء وإدارة خدمة GitHub على خوادمها الخاصة، وتوفر هذه الخدمة أمانًا واستقرارًا أكبر للبيانات والأكواد المصدرية.

9. GitHub Sponsors: هي خدمة تسمح للمستخدمين بدعم المطورين المفضلين لديهم عن طريق الاشتراك في برنامج الرعاية المالية للمطورين، وهي طريقة لتشجيع المطورين على الاستمرار في تقديم المشاريع البرمجية المجانية والمفيدة.

10. GitHub CLI: هو أداة سطر الأوامر التي تسمح للمستخدمين بإدارة مستودعات GitHub وإجراء العمليات المختلفة عليها باستخدام الأوامر المختصرة، وهذا يساعد على تسريع عملية التطوير وتحسين الإنتاجية.

11. GitHub Code spaces: هي خدمة تسمح للمستخدمين بإنشاء مساحات عمل افتراضية (Virtual Environments) للتطوير والاختبار والتعاون على مشاريع برمجية، وتوفر هذه الخدمة بيئة تطويرية موحدة وسهلة الاستخدام للمطورين.

12. GitHub Discussions: هي خدمة تسمح للمستخدمين بإنشاء منتديات (Forums) للمشاريع البرمجية، وهذا يساعد على تحسين التواصل بين المطورين والمستخدمين والتبادل الفعال للأفكار والمشكلات والحلول.

31. GitHub Security: هي خدمة تسمح للمستخدمين بتحليل الأمان لمستودعاتهم وتحديد المشاكل الأمنية والثغرات وتوفير نصائح لتحسين الحماية.

14. Dependabot: هي خدمة تسمح للمستخدمين بالحفاظ على تحديثات البرامج والمكتبات الخاصة بمشاريعهم، وتقدم Dependabot تحديثات آلية ومنتظمة للمكتبات والأدوات المستخدمة في المشروع.

15. GitHub Code spaces: هي خدمة تسمح للمستخدمين بإنشاء مساحات عمل افتراضية (Virtual Environments) للتطوير والاختبار والتعاون على مشاريع برمجية، وتوفر هذه الخدمة بيئة تطويرية موحدة وسهلة الاستخدام للمطورين.

16. GitHub Marketplace: هو متجر يحتوي على مجموعة كبيرة من التطبيقات والخدمات والأدوات التي تساعد المطورين على تحسين عملية التطوير وإدارة المشاريع، ويمكن للمستخدمين تحميل هذه التطبيقات والخدمات والأدوات مباشرة من متجر GitHub.

**6**

17. GitHub Pages: هي خدمة تسمح للمستخدمين بإنشاء مواقع وصفية ويب مجانية ونشرها باستخدام GitHub، وتوفر هذه الخدمة مساحة تخزين ونطاقات مجانية للمستخدمين.

18. GitHub Insights: هي خدمة تسمح للمستخدمين بتحليل بيانات المستودعات وفهم كيفية استخدام الأكواد المصدرية والاكتشافات الأخرى، وتوفر هذه الخدمة بيانات مفيدة لتحسين الإنتاجية وتحديد أفضل الممارسات.

19. GitHub Mobile: هو تطبيق للهواتف الذكية يسمح للمستخدمين بإدارة مستودعات GitHub وإجراء عمليات الاستنساخ والاستدعاء والتحميل والتحكم في الإصدارات بشكل سهل وبديهي.

20. GitHub Templates: هي خدمة تسمح للمستخدمين بإنشاء قوالب (Templates) لمشاريع البرمجيات، ويمكن استخدام هذه القوالب لتوفير الوقت والجهد في إنشاء مشاريع جديدة وتطويرها بسرعة.

21. GitHub Learning Lab: هي خدمة تعليمية مجانية تساعد المستخدمين على تعلم كيفية العمل مع GitHub وإدارة المشاريع والتعاون مع الآخرين، وتوفر هذه الخدمة دروسًا تفاعلية وتمارين عملية.

22. GitHub Archive Program: هي مبادرة تهدف إلى حفظ تاريخ البرمجيات والأكواد المصدرية والمشاريع على GitHub، وتسمح هذه المبادرة للمستخدمين بحفظ مشاريعهم للأبد الأوصاف.

23. GitHub REST API: هي واجهة برمجة التطبيقات (API) التي تسمح للمطورين بالتفاعل مع بيانات GitHub وإجراء العمليات المختلفة عليها من خلال البرمجة.

24. GitHub GraphQL API: هي واجهة برمجة التطبيقات (API) الأخرى التي تسمح للمطورين باستخدام لغة GraphQL للتفاعل مع بيانات GitHub وإجراء العمليات المختلفة عليها.

25. GitHub Advanced Security: هي خدمة تسمح للمستخدمين بتحليل الأمان لمستودعاتهم وتحديد المشاكل الأمنية والثغرات وتوفير نصائح لتحسين الحماية، بالإضافة إلى توفير أدوات للكشف عن البرمجيات الخبيثة وإدارة الأمان بشكل أفضل.

# الميّزات

يعد GitHub منصة شهيرة لإدارة ومشاركة الأكواد المصدرية والعمل بشكل تعاوني على مشاريع برمجية، ويتضمن العديد من الميزات التي تجعلها مفيدة للمطورين والمستخدمين على حد سواء، ومن أبرز هذه الميزات:

1. إدارة المشاريع: يمكن للمستخدمين إنشاء مستودعات (Repositories) للمشاريع البرمجية وإدارتها بسهولة.

2. التحكم في الإصدارات: يتيح GitHub للمستخدمين إمكانية التحكم في إصدارات المشاريع وإدارة التغييرات التي يتم إجراؤها على مستوى الأكواد المصدرية.

3. التعاون والمشاركة: يمكن للمستخدمين المشاركة والتعاون على المشاريع بين المطورين  
 والفرق المختلفة.



**7**

4. نظام إدارة الأخطاء: يمكن للمستخدمين إضافة نظام إدارة الأخطاء (Issue Tracker) لمشاريعهم، وهذا يساعد على تتبع الأخطاء والمشاكل وإصلاحها بشكل أسرع وأكثر دقة.

5. الإدارة المتقدمة للفروع: يتيح GitHub للمستخدمين إنشاء وإدارة الفروع (Branches) بشكل متقدم، وهذا يساعد على تنظيم عملية التطوير وتحسينها.

6. الاستضافة: يمكن للمستخدمين استضافة مشاريعهم على GitHub، وهذا يسمح للمستخدمين بالوصول إلى مشاريعهم من أي مكان وبأي جهاز يتصلون به.

7. التوثيق: يمكن للمستخدمين إضافة التوثيق الخاصة بالمشاريع على GitHub، وهذا يجعلها أكثر وضوحًا وسهولة في الاستخدام.

8. الاختبار الآلي: يمكن للمستخدمين إضافة اختبارات آلية (Automated Testing) لمشاريعهم على GitHub، وهذا يساعد على الكشف المبكر عن الأخطاء وتقليل الوقت المستغرق في إصلاحها.

9. العمل الجماعي: يتيح GitHub للمستخدمين العمل بشكل جماعي على مشاريع برمجية، وهذا يسمح للمطورين بالتواصل والتعاون معًا والتنسيق فيما بينهم بشكل أفضل.

10. خدمة الاستضافة الخاصة: يمكن للمستخدمين استخدام خدمة الاستضافة الخاصة (GitHub Enterprise) لإنشاء خادم خاص بهم لإدارة المشاريع والأكواد المصدرية.

11. التكامل مع أدوات أخرى: يمكن للمستخدمين تكامل GitHub مع أدوات أخرى مثل Slack وTrello وJenkins وغيرها لتسهيل العملية وتحسين الإنتاجية.

12. الأمان والحماية: يتمتع GitHub بميزات أمان عالية وحماية قوية للأكواد المصدرية، ويوفر خيارات للتحقق من صحة الهوية والإذنات والتحقق الثنائي.

هذه بعض من الميزات الرئيسية التي يوفرها GitHub، وبالإضافة إلى ذلك، يوفر GitHub العديد من الأدوات والخدمات الأخرى التي تساعد المطورين على تحسين عملية التطوير والإنتاجية، مثل خدمة التحكم في الإصدارات (Version Control), وخدمة التعليقات والمراجعات (Code Review), وخدمة النشر (Deployment), والكثير من الخدمات الأخرى التي تجعل GitHub منصة شاملة ومفيدة للمطورين.

# الأمان

يعتبر GitHub آمنًا للاستخدام العام,حيث يوفر  العديد من ميزات الأمان، بما في ذلك تشفير الاتصال باستخدام بروتوكول HTTPS وتطبيق ممارسات أمانية تضمن سلامة البيانات الخاصة بالمستخدمين. كما توجد إجراءات إضافية يمكن أن يتبعها المستخدمون لتحسين الأمان على حساباتهم على GitHub،على سبيل المثال:

-1استخدام كلمات مرور قوية ولا تشاركها مع أي شخص آخر.



**8**

-2 تفعيل المصادقة الثنائية (Two-factor authentication)، وهي طريقة لتوفير حماية إضافية لحساب الـ GitHub الخاص بك، حيث يتم إرسال رمز مؤقت إلى هاتفك المحمول أو البريد الإلكتروني لتسجيل الدخول إلى حسابك.

-3تجنب إعطاء الأذونات لأي شخص غير موثوق به مع دور "Owner" أو "Admin" في ال repositories الخاصة بك.

-4 إعداد واستخدام مفاتيح SSH الآمنة لاتصال آمن بين مستخدم الـ GitHub والخادم.

-5 استخدام HTTPS بدلاً من SSH. هذا سيتيح لك استخدام اتصال مشفر للوصول إلى GitHub.

-6 تحديث GitHub إلى آخر إصداراته دائمًا، وذلك لعلاج الثغرات الأمنية التي تم اكتشافها مؤخرًا.

-7 استخدام أدوات الأمان المتاحة، مثل GitHub Security Alerts و Dependabot وتقارير المشكلات الأمنية. إذا تلقيت تنبيهًا حول ثغرة أمنية جديدة في أي repository، عليك إجراء التحديثات اللازمة في أسرع وقت ممكن.

-8 الشهادات الرقمية: توفر Git Hub شهادات SSL (Secure Sockets Layer) لضمان أن بيانات المستخدمين يتم نقلها عبر الإنترنت بشكل آمن.

-9 الاستجابة للحوادث: في حالة الكشف عن انتهاك أمني ، فإن Git Hub يتخذ إجراءات لإزالة المخاطر على الفور.

-10 التحقق من البرمجيات الضارة: تقوم Git Hub بفحص ملفات الكود عند الرفع للتأكد من عدم وجود برامج ضارة.

# الفرق بين SourceFrogeوGitHub

ٍ SourceForge وGitHub هما منصتان لإدارة ونشر مشاريع البرمجيات، ولكن هناك بعض الفروقات بينهما:

1- GitHub توفر خدمات إضافية مثل إدارة المشروع والتحكم في الإصدارات والتعاون بين المطورين، في حين أن SourceFroge تركز على إدارة المشروع فقط.

2- GitHub يستخدم نظام Git لإدارة الإصدارات، في حين أن SourceFrogeيستخدم Git و SVN.

3- GitHub توفر خططًا مجانية وبأسعار مختلفة للاستخدام الشخصي والأعمال التجارية، في حين أن SourceFroge توفر خططًا مجانية للاستخدام الشخصي فقط.

4- GitHub يستضيف المشاريع على خوادمه، في حين أن SourceFroge يسمح للمستخدم بتثبيته على خادمه الخاص.

# التكاليف

اما على التكاليف المترتبة على تحميل وتنزيل و استخدامGitHub فإنه



**9**

بشكل عام استخدام GitHub لتحميل وتنزيل المشاريع واستخدامها من قبل المستخدمين يعد مجانيًا ولا يتطلب أية تكاليف مالية مباشرة. ومع ذلك، قد يتم فرض تكاليف إضافية على المستخدمين إذا قاموا بتحميل مشروع يتطلب مصادر إضافية أو إذا استخدموا خدمات إضافية مدفوعة على GitHub كخدمات الرسائل أو النماذج .

بالإضافة إلى ذلك، يجب على المستخدمين احتساب التكاليف غير المباشرة مثل الوقت والجهد المستخدمين في تحميل وتنزيل المشاريع والتعامل مع التغييرات والتحديثات اللازمة. وقد يتطلب ذلك إجراء إعدادات وتكوينات معينة واستخدام أدوات إضافية للعمل بكفاءة أكبر.

بشكل عام، يمكن القول أن استخدام GitHub يتطلب تكاليف محدودة جدًا، إذا اقتصر استخدام المستخدم على تنزيل المشاريع والبرامج المجانية. ولكن، إذا كان المستخدم يرغب في الاستفادة من مزايا إضافية، فإنه يجب عليه النظر في الخدمات المدفوعة وتكاليفها قبل الاشتراك فيها.

تتضمن خدمات GitHub المدفوعة مزايا مثل إمكانية إضافة مستخدمين إضافيين إلى المشروع، وإنشاء قائمة بالمشاريع الخاصة، وتعيين أذونات محددة للمستخدمين، ومزايا أخرى. وتختلف تكاليف هذه الخدمات حسب الخطة التي يختارها المستخدم، حيث يتراوح سعر الاشتراك الشهري بين 4 دولارات و21 دولارًا، وقد يصل إلى 200 دولار شهريًا للشركات الكبيرة.

# المراجع:

* GitHub
* منصة Naqrah
* ويكيبيديا
* ChatGPT

**10**

**11**