



ВИСОКА ТЕХНИЧКА ШКОЛА
СТРУКОВНИХ СТУДИЈА
у Новом Саду

ИНТЕРНЕТ ЈЕЗИЦИ И АЛАТИ 1

ДОКУМЕНТАЦИЈА КОДА И БЕСПЛАТАН ХОСТИНГ КОРИШЋЕЊЕМ GIT И GITHUB ПЛАТФОРМЕ

Нови Сад, 2022/2023.

САДРЖАЈ

1.	УВОД	3
2.	КРЕИРАЊЕ РЕПОЗИТОРИЈУМА	4
2.1.	Отпремање фајлова „drag and drop“ системом	6
2.2.	Бесплатан хостинг „Github Pages“	9
2.3.	Отпремање фајлова преко CLI интерфејса	10

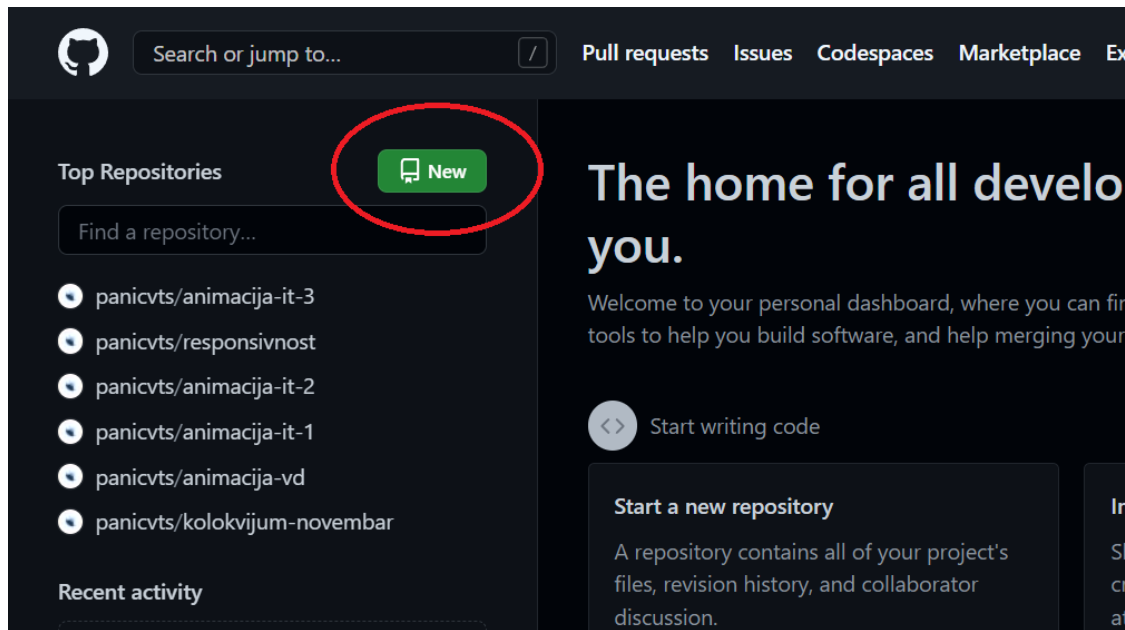
1. УВОД

Како би пројекат поставили на Github и имали приступ коду било када и било где, под условом да имамо приступ интернету неопходно је креирати налог на Github платформи путем линка: <https://github.com/>

2. КРЕИРАЊЕ РЕПОЗИТОРИЈУМА

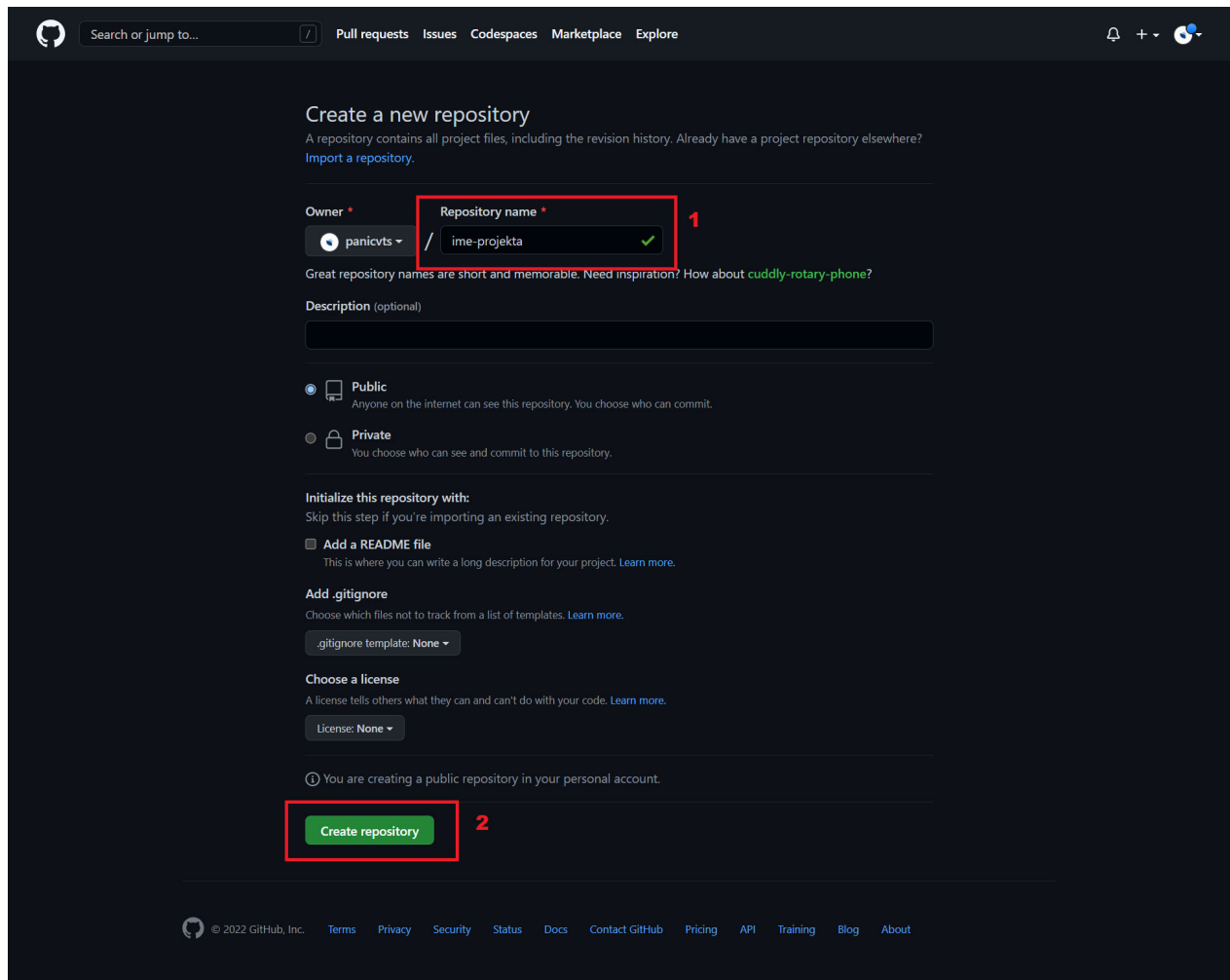
Након креираног налога, неопходно је да се улогујемо на платформу како би наше пројекте поставили на cloud.

У горњем левом углу почетне странице налази се листа наших репозиторијума уколико их имамо односно листа је празна уколико их немамо. У оба случаја у ток секцији се налази дугме за креирање новог репозиторијума које је неопходно кликнути како би ушли у даљи процес.



Слика 1. Почетна страница Github платформе

Након клика на зелено дугме отвориће нам се нова страница у којој постоји неколико поља које је неопходно попунити како би направили репозиторијум. Прво од њих је назив репозиторијума (пројекта). Уколико пројекат има назив дужи од једне речи, неопходно је те речи одвојити средњом цртом. После назива долази необавезно поље са описом пројекта, затим подешавање приватности пројекта где је врло важно чекирати дугме „Public“ како би нам пројекат био јаван и како би могли поставити исти на бесплатан хостинг. Поља „Read Me“, „git ignore“, „licence“ За сада можемо занемарити. Након што смо подесили све информације о пројекту кликнућемо дугме на дну странице „Create repository“ и наш фолдер ће бити креиран а затим ће нас платформа преусмерити на садржај тог фолдера (репозиторијума).



Слика 2. Креирање новог репозиторијума

Када смо преусмерени на репозиторијум биће нам понуђено неколико начина за отпремање кода нашег локалног пројекта. Кроз ову кратку скрипту објаснићемо оба начина.

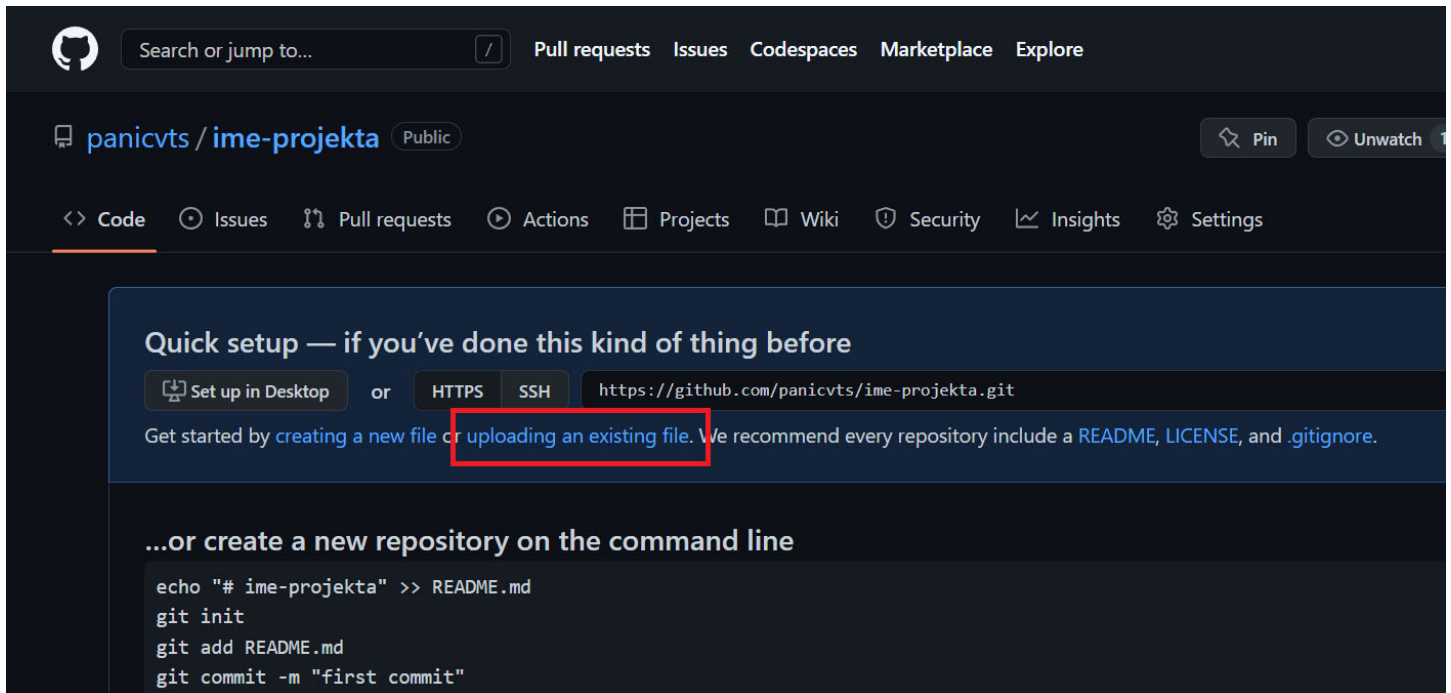
Први и најлакши начин је да отпремимо фајлове нашег локалног пројекта „Drag and drop“ системом. За то је потребно пронаћи плави правоугаоник и у њему дугме „upload an existing file“. Ту ће нам се отворити нови прозор где ћемо имати прилику да отпремимо код преко „Drag and drop“ системом или да селекујемо фајлове путем новог прозора за одабир фајлова.

Напомена: Уколико фајлове одаберемо преко прозора за одабир фајлова неће нам се отпремити фолдери већ ћемо морати ручно да их касније креирамо што и није препорука уколико не познајемо github.

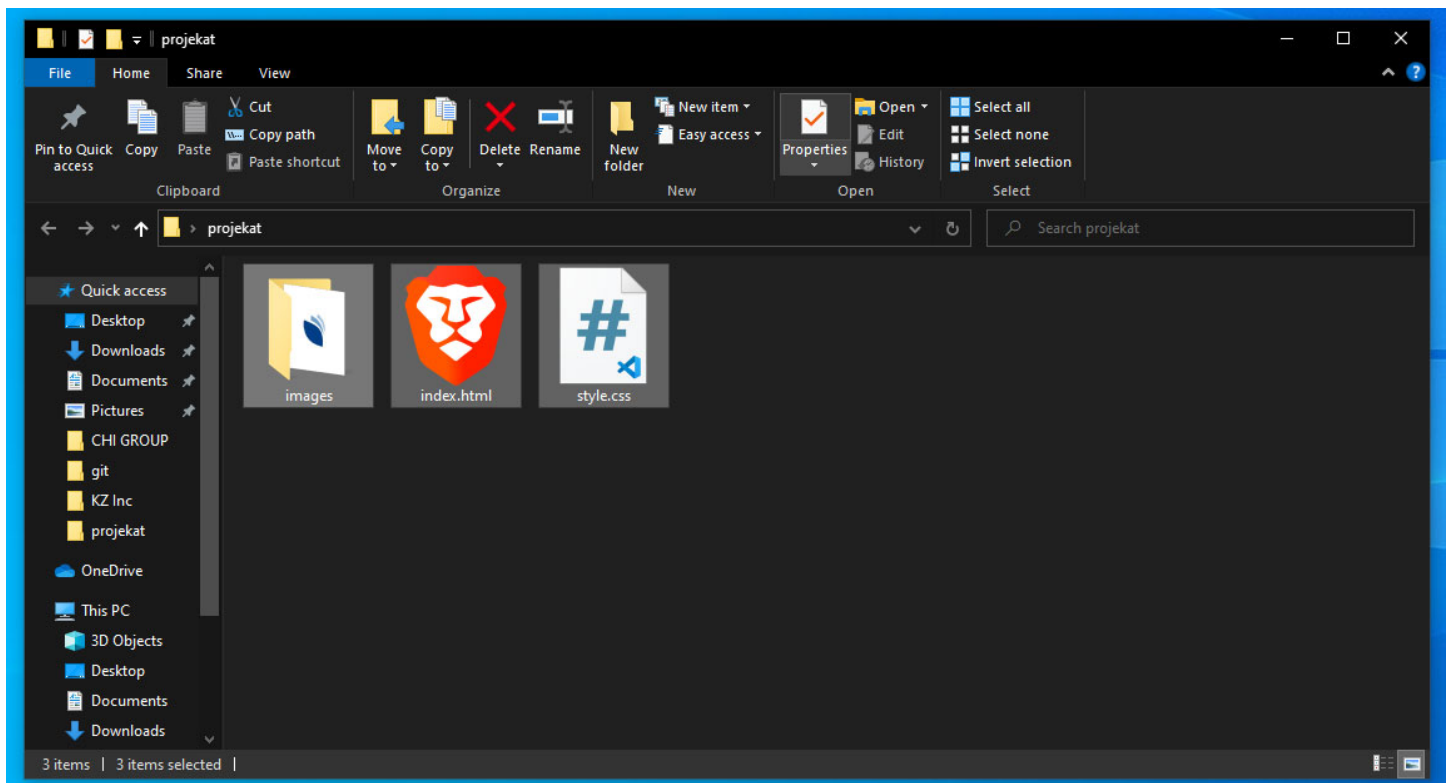
Дакле, препорука је да померимо главни прозор где нам се налази платформа, затим да отворимо фолдер локалног пројекта на рачунару, мишем селекујемо све фајлове и фолдере, затим их пренесемо и пустимо у правоугаоник / квадрат на платформи у другом прозору.

Напомена: Не пребацујемо цео фолдер локалног пројекта већ искључиво садржај.

2.1. Отпремање фајлова „drag and drop“ системом



Слика 3. Отпремање већ постојећих фајлова



Слика 4. Селекција свих фајлова и фолдера унутар локалног пројекта

Када смо завршили селекцију и одрадили отпремање, у дну странице видећемо листу свих фајлова и фолдера које смо отпремили. Приметно је шта су фолдери и који су фајлови у њима.

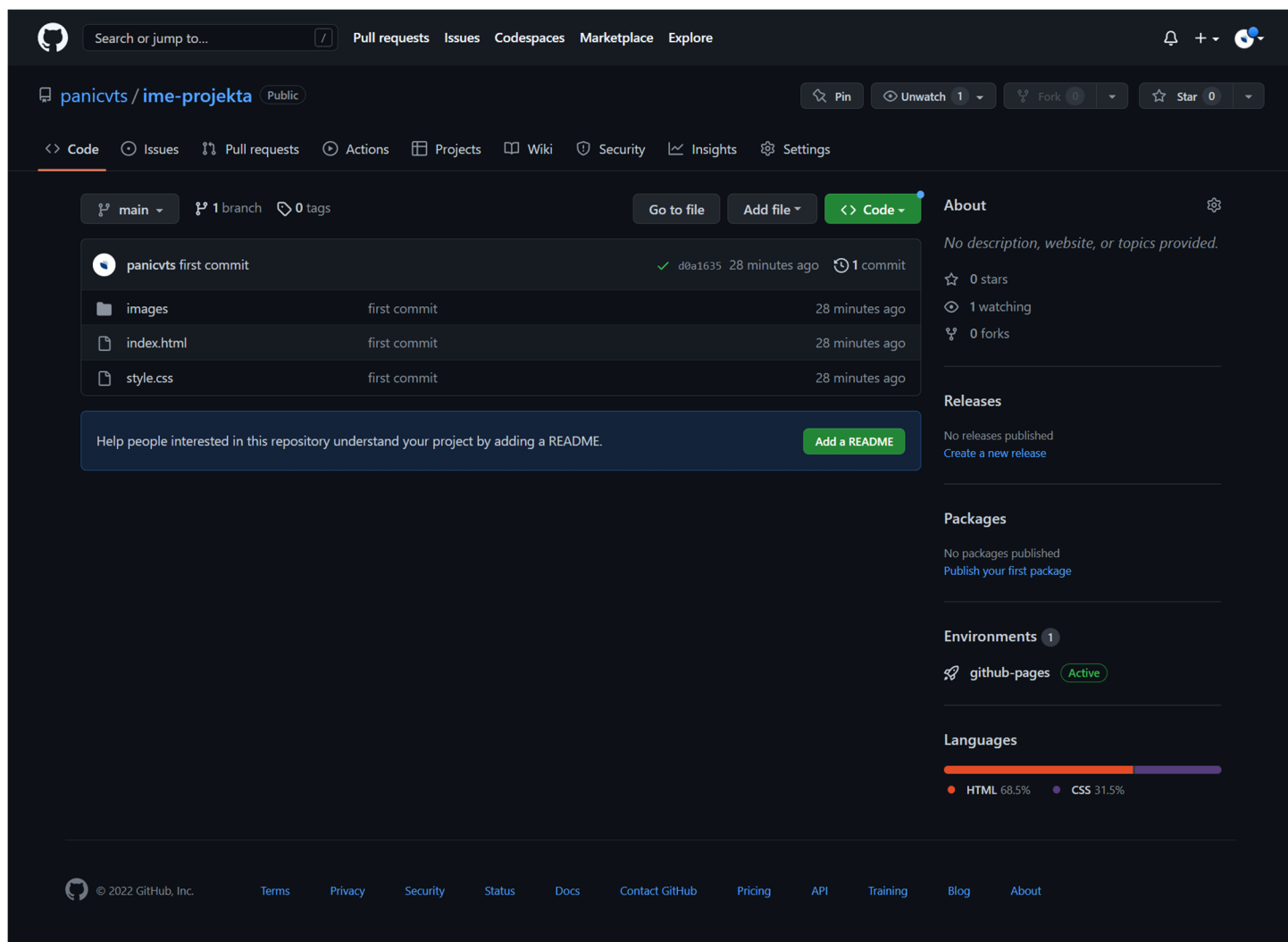
Након провере да ли је све уреду, није обавезно али не би било лоше да оставимо коментар. Углавном су први коментари када се пројекат убацује на платформи: „First commit“, „Prvi commit“, „Initial Commit“, „Init“ и слично. У овом случају први коментар је „first commit“ да се зна касније у пројекту кад је и шта рађено.

Коментари су врло важни при документацији кода како би се у будућности знало све о некој промени. У корпоративном свету, апсолутно свака промена мора да се документује. Није важно да ли је промењена позадинска боја или нешто комплексније, све мора да се документује. Такође код коментара је врло важно да буду смислени и да читалац без потешкоћа схвати каква је промена била у питању и где.



Слика 5. Писање коментара

Када је коментар написан и када смо кликнули дугме „Commit“, платформа ће у наш репозиторијум убацити фајлове и фолдере а затим ће нас преусмерити на тај репозиторијум где можемо видети информације као што су наши фајлови, фолдери, време када су убачени, колико пута је код мењан (commit), ко је убацио последњу промену и слично.



Слика 6. Репозиторијум

2.2. Бесплатан хостинг „Github Pages“

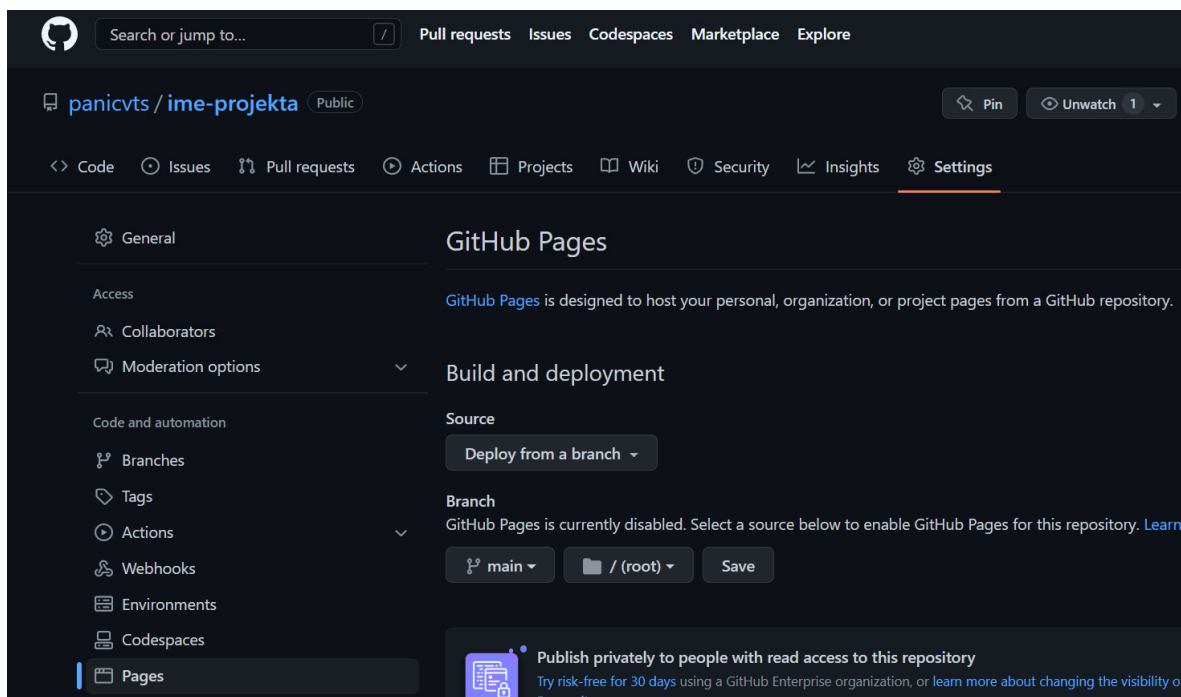
Уколико желимо да наш пројекат поставимо на бесплатан хостинг, потребно је отићи у подешавање репозиторијума, затим са леве стране изабрати „Pages“, те код „Build and deployment“ изабрати извор „Deploy from branch“, и пошто имамо само једну грану која је уједно и главна потребно је под „Branch“ изабрати „main“, затим кликнути на дугме „save“ да сачувамо промене.

Овај процес може потрајати од пар минута до пар сати и то све зависи од платформе и сервера.

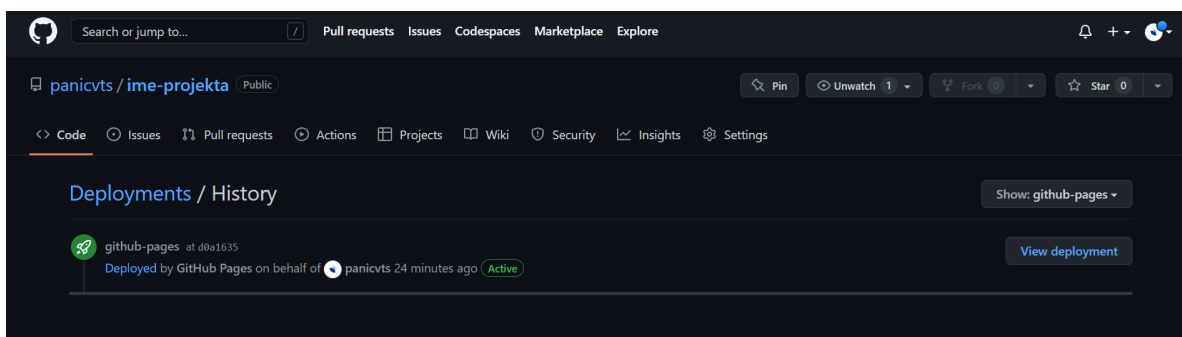
Напомена: Почетна страница нашег сајта мора имати назив index.html и не сме биит ни у каквом фолдеру. У супротном платформа ће јавити грешку и наш сајт неће бити окачен на хостинг.

Уколико је наш пројекат окачен на хостингу, унутар репозиторијума у доњем углу десног менија ће нам се појавити поље „Enviroments“ и дугме „github-pages“ са статусом. То се може видети на слици 6.

Када кликнемо на дугме „github-pages“ отвориће нам се нови прозор са историјом наших развијања пројекта. Кликном на дугме „View deployment“ платформа ће нас преусмерити на линк сајта. Тај линк можемо копирати и послати било где како би више људи видело наш сајт.



Слика 7. Github Pages подешавања

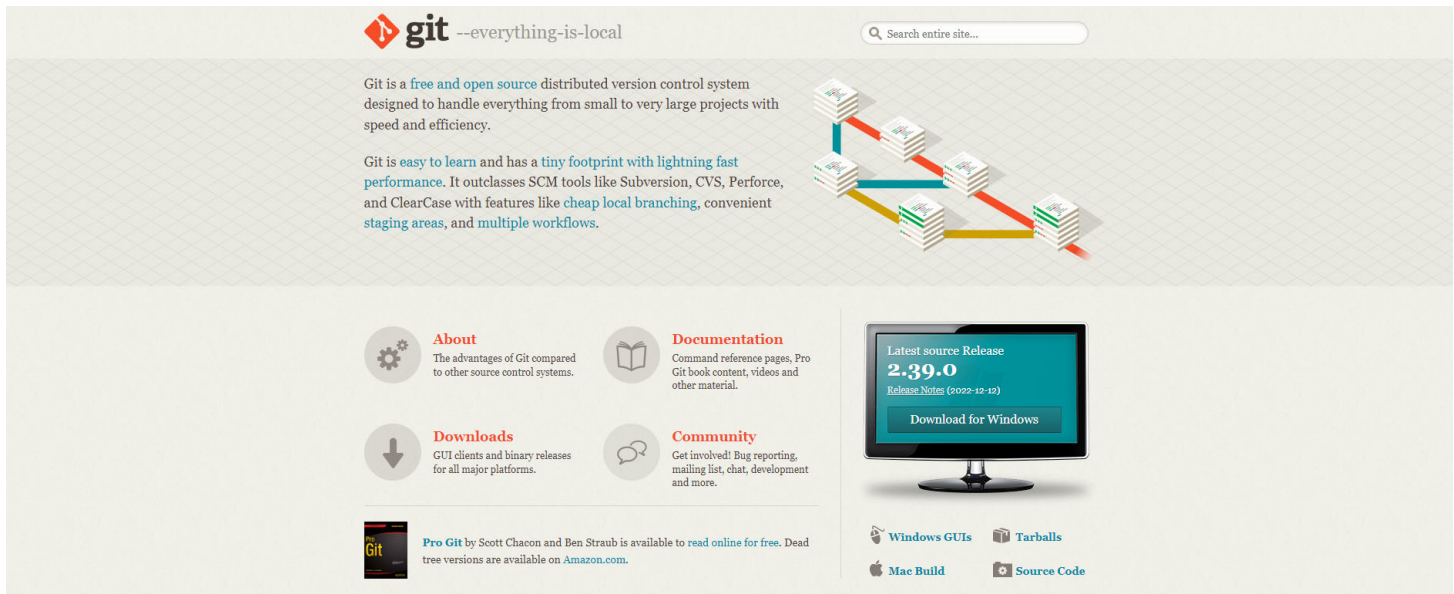


Слика 8. Историја развијања сајта

2.3. Отпремање фајлова преко CLI интерфејса

Како све више градимо наш пројекат и како он има све више и више промена које је потребно отпремити на репозиторијум, постаје напорно сваки секунд додавати кодове и фајлове „drag and drop“ системом те нам је потребна бржа и ефикаснија метода.

Решење за то је git програм који је потребно инсталирати на рачунар путем линка: <https://git-scm.com/downloads>



Слика 9. Сајт за преузимање git програма

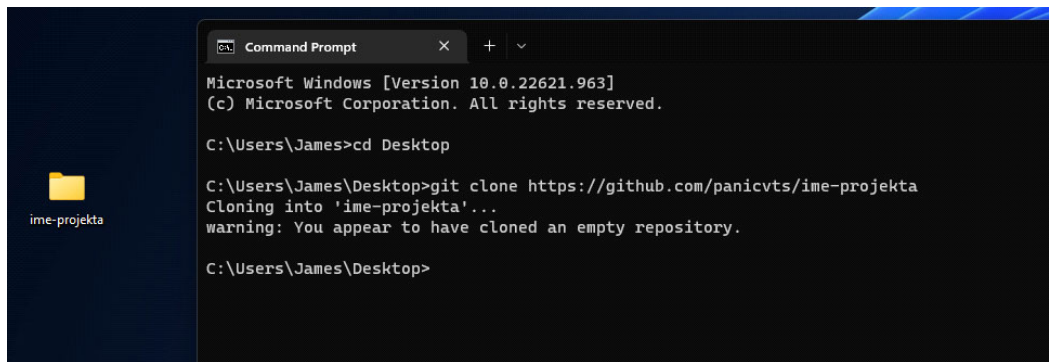
Када је програм преузет и инсталиран можемо почети са локалним отпремањем кода на репозиторијум. Једини проблем је како ће наш локални пројекат да зна где треба да отпреми фајлове. Постоје два решења за то. Први и тежи начин је отворимо локални пројекат у „command prompt-у” и откуцамо пар линија кода да подесимо параметер. Овај начин је тежи јер често не можемо успоставити везу са репозиторијумом тако лако. Други начин који је уједно и препорука за почетак је да клонирамо репозиторијум на локални рачунар и да одатле радимо све измене будући да одмах имамо повезан фолдер са репозиторијумом. Потребно је само да копирамо линк репозиторијума (не линк сајта), отворимо „command prompt“ на рачунару и путем DOS команди пронађемо место за клонирање.

Када смо пронашли место где желимо да клонирамо репозиторијум потребно је откуцати следећу коадну:

Git clone *линк до репозиторијума*

Затим ћемо приметити да се у овом случају створио фолдер на десктопу са именом „ime-projekta“ што је и назив удаљеног репозиторијума. На слици 10 се види да је git избацио упозорење са речима да је клониран празан репозиторијум.

Када је све то одрађено можемо копирати све фајлове нашег пројекта у тај новонастали фолдер.



Слика 10. Клонирање репозиторијума

Клонирањем смо само довели удаљени репозиторијум на локални рачунар и остварена је веза међутим још увек не можемо отпремити фајлове јер git не зна ко је корисник који шаље промене. Дакле потребно је улоговати се на рачунару на github платформу. То ћемо постићи променом конфигурације имејла и корисничког имена као што је приказано на слици 11.

```
C:\Users\James\Desktop\ime-projekta>git config --global user.email "panic@vtsns.edu.rs"
C:\Users\James\Desktop\ime-projekta>git config --global user.name "panicvts"
C:\Users\James\Desktop\ime-projekta>
```

Слика 11. Подешавање креденцијала

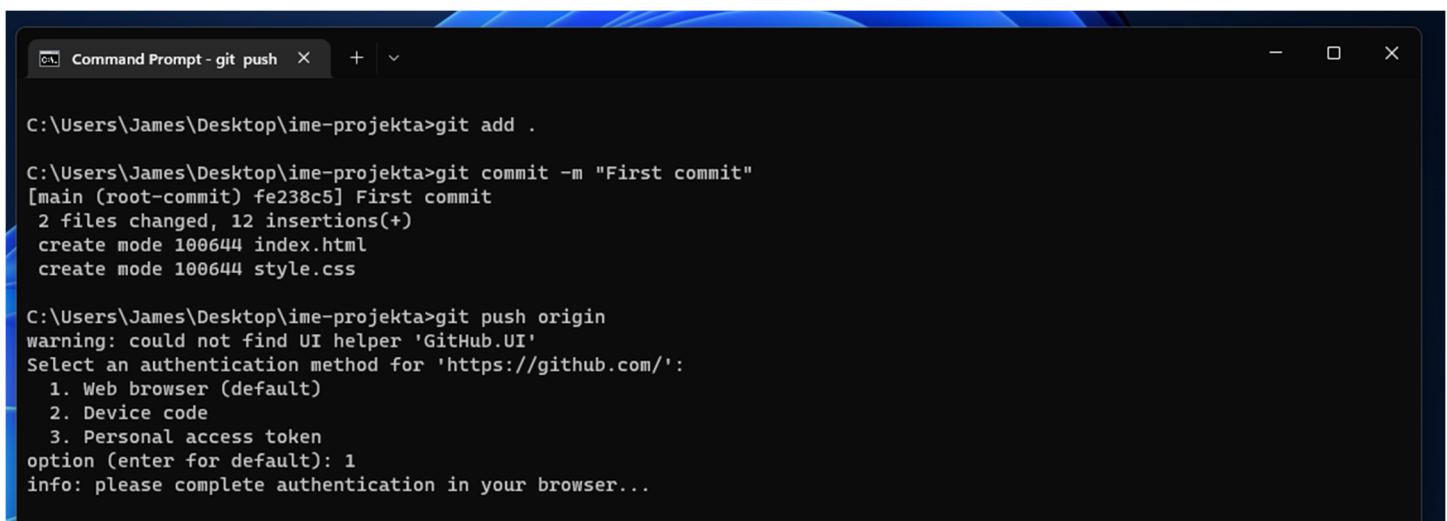
Када је то готово, можемо откуцати следеће линије кода како би успешно отпремили фајлове на github:

Git add . – Служи нам да спремимо фајлове. Тачка представља све фајлове који се налазе у фолдеру

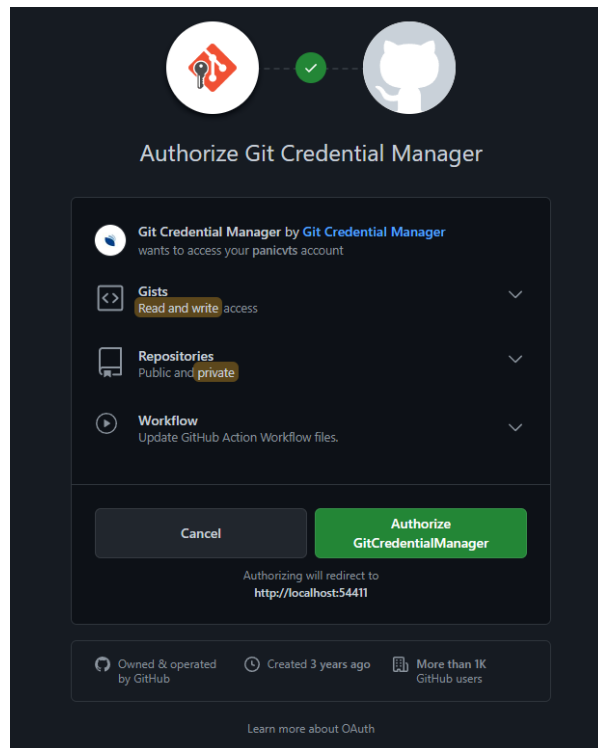
Git commit -m „Назив коментара“ – Служи нам да припремимо коментар који ћемо отпремити са фајловима

Git push origin – Служи нам да отпремимо фајлове на удаљени репозиторијум

Када то одрадимо git ће нас питати који систем аутентификације желимо да користимо.



Слика 12. Отпремање фајлова и аутентификација



Слика 12. Ауторизација кренџијала

Када извршимо ауторизацију приметићемо у CMD-у да смо успешно отпремили фајлове и неке детаље о њима као и грану на којој се налазе измене. Сада све што је потребно да радимо приликом писања кода је да документујемо сваку промену са горе наведене три линије кода.

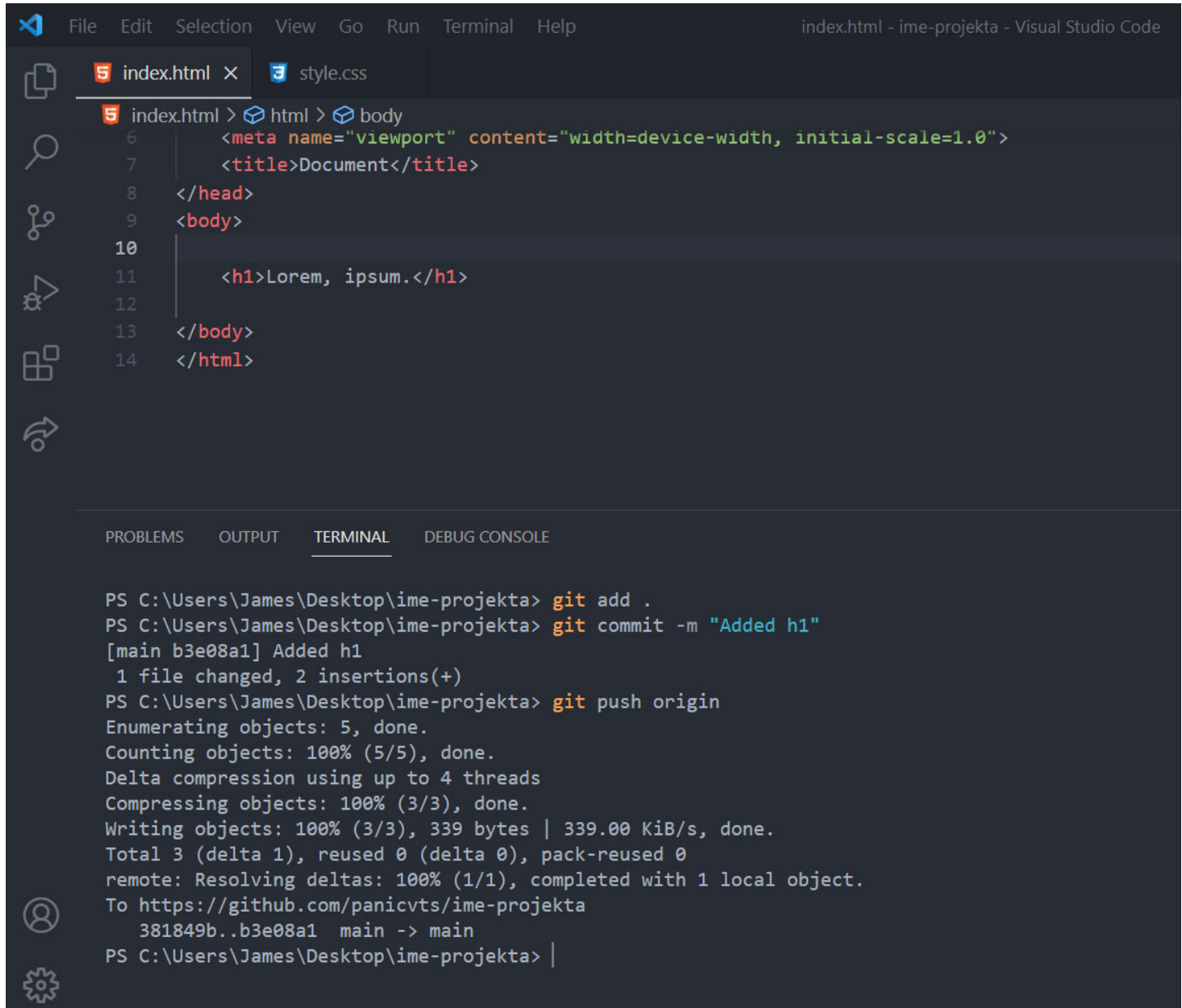
```
Command Prompt
C:\Users\James\Desktop\ime-projekta>git add .
C:\Users\James\Desktop\ime-projekta>git commit -m "First commit"
[main (root-commit) fe238c5] First commit
2 files changed, 12 insertions(+)
create mode 100644 index.html
create mode 100644 style.css
C:\Users\James\Desktop\ime-projekta>git push origin
warning: could not find UI helper 'GitHub.UI'
Select an authentication method for 'https://github.com/':
 1. Web browser (default)
 2. Device code
 3. Personal access token
option (enter for default): 1
info: please complete authentication in your browser...
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (4/4), 428 bytes | 428.00 KiB/s, done.
Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/panicvts/ime-projekta
 * [new branch]      main -> main
C:\Users\James\Desktop\ime-projekta>
```

Слика 13. Успешно отпремљени фајлови на удаљени репотиторијум

Међутим, још увек читав процес није ефикасан јер морамо стално отворати cmd како отпремали измене.

Решење лежи све популарнијем текст едитору „Visual Studio Code“. Наиме он има уграђени терминал (што је исто што и cmd, git bash, powershell итд) који нам омогућава да у оквиру текст едитора имамо отворен мали терминал који је повезан са нашим фолдером па је потребно само да куцамо горе наведене три команде како би успешно отпремали код. На слици 14 се може приметити колико је VSCode терминал прегледнији од cmd-a.

У терминал се улази кликом на „Terminal“ у горњем менију, затим кликом на „new terminal“.

The image is a screenshot of the Visual Studio Code interface. At the top, there is a menu bar with 'File', 'Edit', 'Selection', 'View', 'Go', 'Run', 'Terminal', and 'Help'. Below the menu bar, there are two tabs: 'index.html' and 'style.css'. The 'index.html' tab is active, showing an HTML document with the following content:

```
5 index.html > html > body
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Document</title>
8 </head>
9 <body>
10
11   <h1>Lorem, ipsum.</h1>
12
13 </body>
14 </html>
```

Below the code editor, there is a panel with four tabs: 'PROBLEMS', 'OUTPUT', 'TERMINAL', and 'DEBUG CONSOLE'. The 'TERMINAL' tab is active, showing a PowerShell prompt with the following commands and output:

```
PS C:\Users\James\Desktop\ime-projekta> git add .
PS C:\Users\James\Desktop\ime-projekta> git commit -m "Added h1"
[main b3e08a1] Added h1
 1 file changed, 2 insertions(+)
PS C:\Users\James\Desktop\ime-projekta> git push origin
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 339 bytes | 339.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/panicvts/ime-projekta
 381849b..b3e08a1  main -> main
PS C:\Users\James\Desktop\ime-projekta> |
```

Слика 14. Visual Studio Code терминал

Уколико користимо више један налог на више рачунара за исти пројекат, незамисливо је да сваки пут радимо клонирање пројекта. Решење за то је једна линија кода која нам довлачи последњу верзију репозиторијума:

Git pull

Дакле, не довлачимо последњу верзију са неког другог рачунара која није отпремљена већ искључиво последњу верзију репозиторијума.