

Кафедра **Програмного**
Забезпечення
автоматизованих систем

Лабораторна робота № 2

Тема : Робота з системою керування версіями Git

Виконав:
ст. групи ПІ-14-1
Депутат Андрій
Перевірив:
Бандура В. В.

Мета: здобути необхідні знання та навички для роботи з системою керування версіями Git.

Завдання:

1. Зареєструватися на GitHub;
2. Завантажити ПЗ для роботи з Git;
3. Встановити на робочий комп'ютер;
4. Створити репозиторій на GitHub;
5. Створити локальний репозиторій;
6. Додати в локальний репозиторій файл;
7. Закомітити даний файл в локальний репозиторій;
8. Закинути зміни з локального репозиторію який ви створили в пункті 4;
9. Додати два файли в локальний репозиторій. Закомітайте і запускайте на GitHub;
10. Відновіться до стану, який був в пункті 8;
11. Зробіть клон репозиторія з GitHub в іншу папку на вашому комп'ютері;
12. Створіть нову гілку і додайте в неї новий файл;
13. Закомітайте зміни в новій branch. Синхронізуйте з репозиторієм на GitHub;
14. Об'єднайте створену гілку з основною;
15. Змініть файл, який ви додали в пункті 12 в основній branch;
16. Переключіться на гілку, яку ви створили в пункті 12 і також зробіть зміни в файлі, що додали в пункті 12.
17. Об'єднайте гілку з основною. Конфлікт файлу вирішіть в ручну.

Короткі теоретичні відомості:

Git — розподілена система керування версіями файлів та спільної роботи. Проект створив Лінус Торвальдс для управління розробкою ядра Linux, а сьогодні підтримується Джуніо Хамано (англ. *Junio C. Hamano*). Git є однією з найефективніших, надійних і високопродуктивних систем керування версіями, що надає гнучкі засоби нелінійної розробки, що базуються на відгалуженні і злитті гілок. Для забезпечення цілісності історії та стійкості до змін заднім числом використовуються криптографічні методи, також можлива прив'язка цифрових підписів розробників до тегів і комітів.

Прикладами проектів, що використовують Git, є ядро Linux, Android, LibreOffice, Cairo, GNU Core Utilities^[2], Mesa 3D^[3], Wine, багато проектів з X.org^[4], XMMS2^[5], GStreamer, Debian DragonFly BSD, Perl, Eclipse, GNOME, KDE, Qt, Ruby on Rails, PostgreSQL, VideoLAN, PHP, Beryl^[6], One Laptop Per Child (OLPC), АБІС Koha, GNU LilyPond^[7] та ELinks^[8] і деякі дистрибутиви GNU/Linux (див. нижче).

Програма є вільною і випущена під ліцензією GNU GPL версії 2.

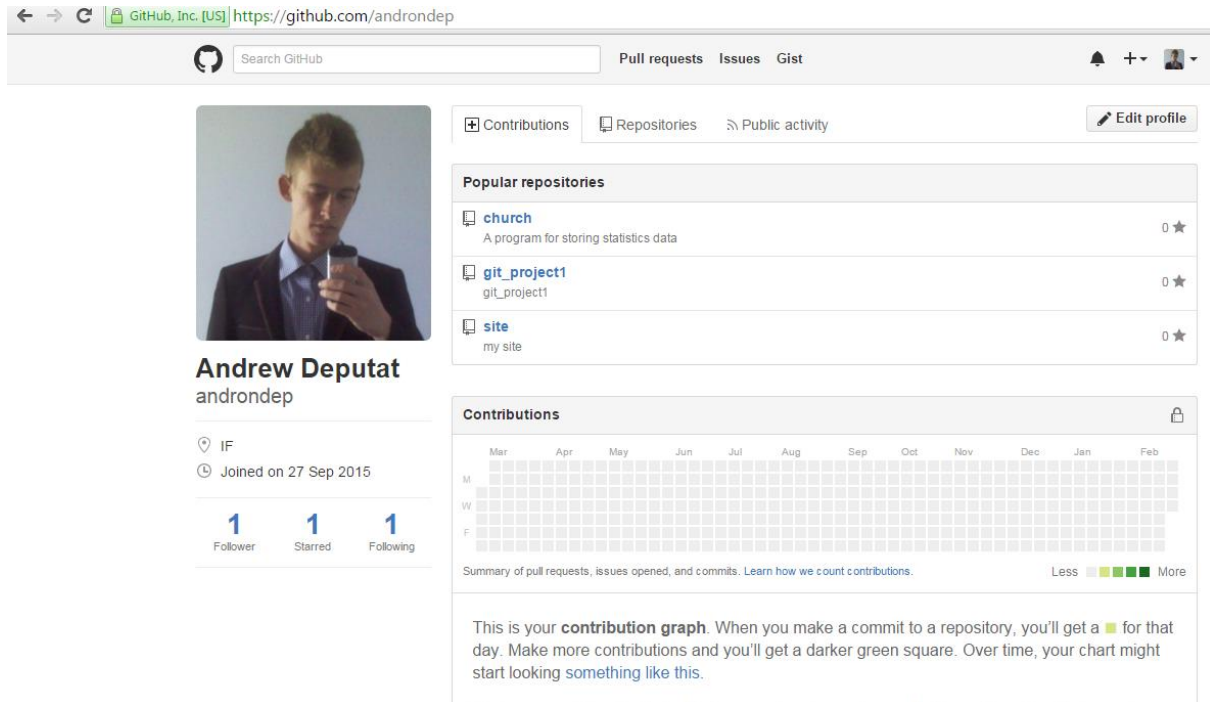
Система спроектована як набір програм, спеціально розроблених з врахуванням їхнього використання у скриптах. Це дозволяє зручно створювати спеціалізовані системи управління версіями на базі Git або користувацькі інтерфейси. Наприклад, Cogito^[9] є саме таким прикладом фронтенда до репозиторіїв Git. А StGit^[10] використовує Git для управління колекцією латок.

Система має ряд користувацьких інтерфейсів: наприклад, gitk та git-gui розповсюджуються з самим Git.

Віддалений доступ до репозиторіїв Git забезпечується git-демоном, SSH або HTTP сервером. TCP-сервіс git-daemon входить у дистрибутив Git і є разом з SSH найпоширенішим і надійним методом доступу. Метод доступу HTTP, незважаючи на ряд обмежень, дуже популярний в контрольованих мережах, тому що дозволяє використання існуючих конфігурацій мережеских фільтрів.

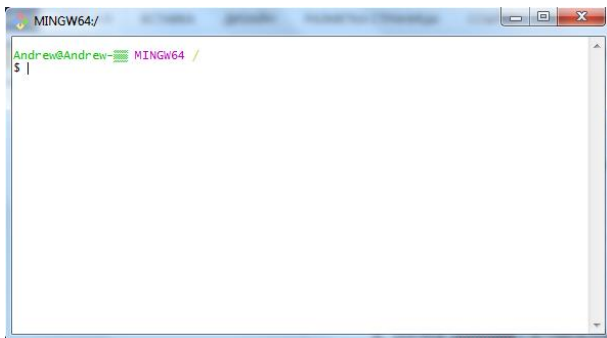
Виконання:

1. Вже зареєстрований на GitHub:

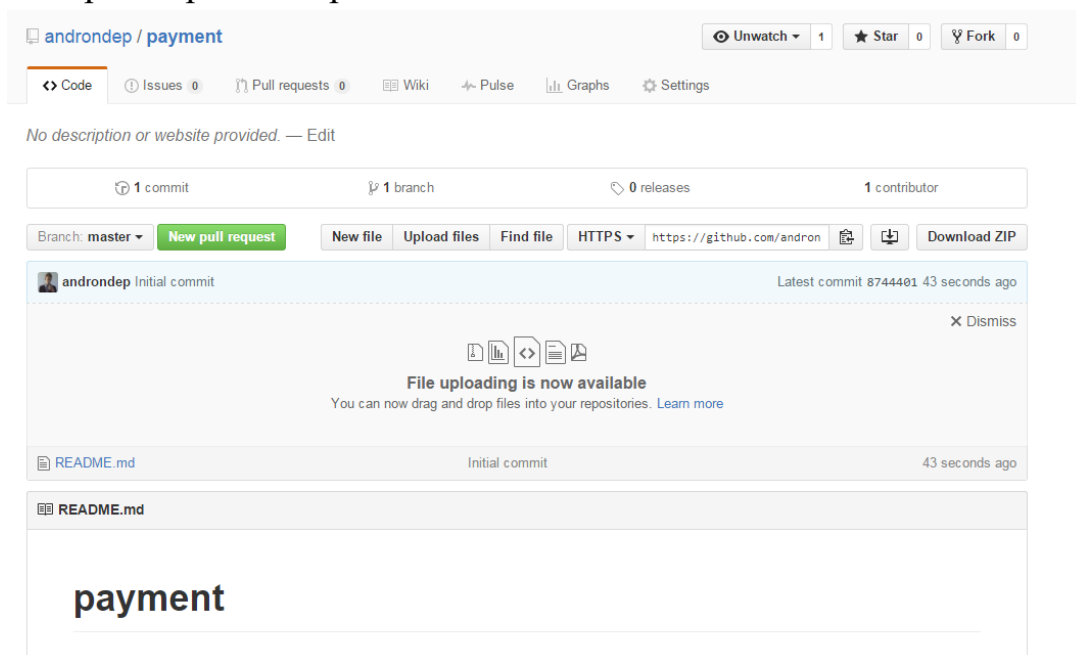


2. ПЗ був скачений і встановлений:

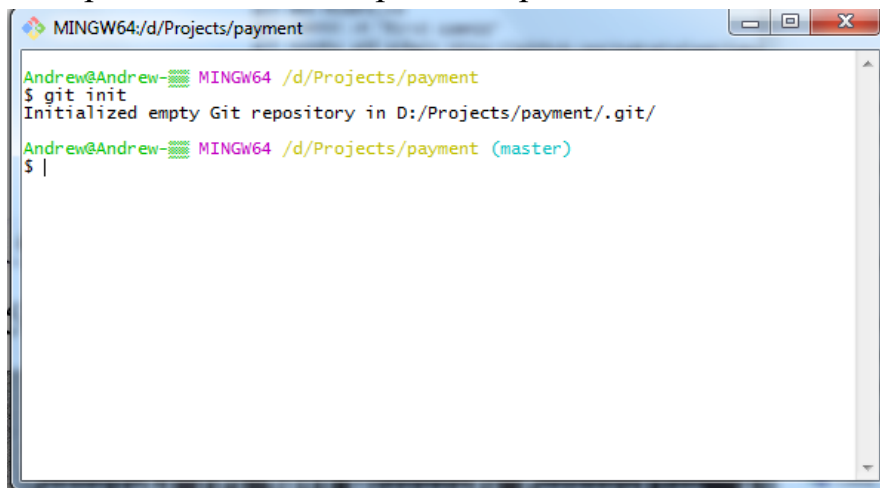
3.



4. Створення репозиторія на GitHub:

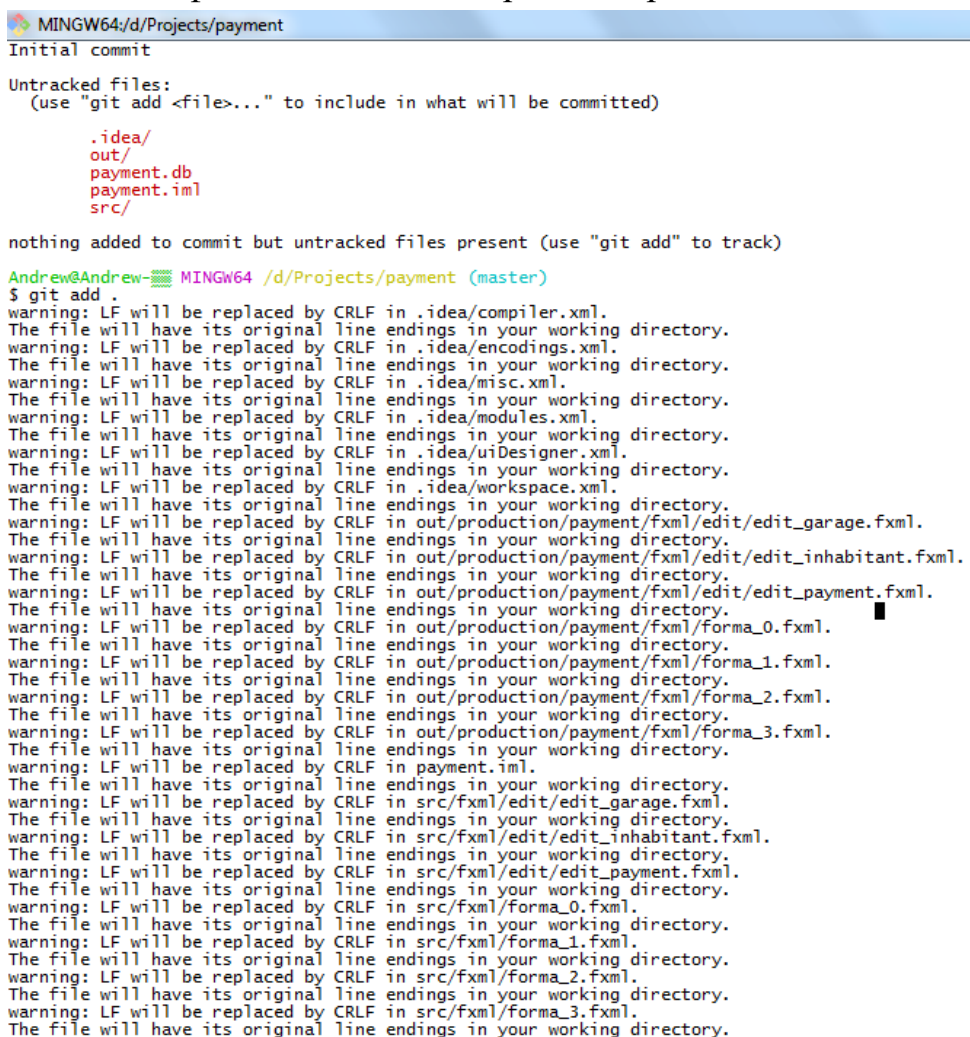


5. Створення локального репозиторія:



```
MINGW64:/d/Projects/payment
Andrew@Andrew-MINGW64 /d/Projects/payment
$ git init
Initialized empty Git repository in D:/Projects/payment/.git/
Andrew@Andrew-MINGW64 /d/Projects/payment (master)
$ |
```

6. Індексція файлів в локальний репозиторій:



```
MINGW64:/d/Projects/payment
Initial commit

Untracked files:
  (use "git add <file>..." to include in what will be committed)

        .idea/
        out/
        payment.db
        payment.iml
        src/

nothing added to commit but untracked files present (use "git add" to track)
Andrew@Andrew-MINGW64 /d/Projects/payment (master)
$ git add .
warning: LF will be replaced by CRLF in .idea/compiler.xml.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in .idea/encodings.xml.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in .idea/misc.xml.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in .idea/modules.xml.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in .idea/uiDesigner.xml.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in .idea/workspace.xml.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in out/production/payment/fxml/edit/edit_garage.fxml.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in out/production/payment/fxml/edit/edit_inhabitant.fxml.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in out/production/payment/fxml/edit/edit_payment.fxml.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in out/production/payment/fxml/forma_0.fxml.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in out/production/payment/fxml/forma_1.fxml.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in out/production/payment/fxml/forma_2.fxml.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in out/production/payment/fxml/forma_3.fxml.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in payment.iml.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in src/fxml/edit/edit_garage.fxml.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in src/fxml/edit/edit_inhabitant.fxml.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in src/fxml/edit/edit_payment.fxml.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in src/fxml/forma_0.fxml.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in src/fxml/forma_1.fxml.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in src/fxml/forma_2.fxml.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in src/fxml/forma_3.fxml.
The file will have its original line endings in your working directory.
```

7. Закомічення змін в локальний репозиторій:

```
MINGW64/d/Projects/payment

Andrew@Andrew-MINGW64 /d/Projects/payment (master)
$ git commit -m "first commit"
[master (root-commit) 9e599e2] first commit
warning: LF will be replaced by CRLF in .idea/compiler.xml.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in .idea/encodings.xml.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in .idea/misc.xml.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in .idea/modules.xml.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in .idea/uiDesigner.xml.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in .idea/workspace.xml.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in out/production/payment/fxml/edit_garage.fxml.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in out/production/payment/fxml/edit_inhabitant.fxml.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in out/production/payment/fxml/edit_payment.fxml.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in out/production/payment/fxml/forma_0.fxml.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in out/production/payment/fxml/forma_1.fxml.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in out/production/payment/fxml/forma_2.fxml.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in out/production/payment/fxml/forma_3.fxml.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in payment.iml.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in src/fxml/edit/edit_garage.fxml.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in src/fxml/edit/edit_inhabitant.fxml.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in src/fxml/edit/edit_payment.fxml.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in src/fxml/forma_0.fxml.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in src/fxml/forma_1.fxml.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in src/fxml/forma_2.fxml.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in src/fxml/forma_3.fxml.
The file will have its original line endings in your working directory.
75 files changed, 4127 insertions(+)
create mode 100644 .idea/.name
create mode 100644 .idea/artifacts/payment_jar.xml
create mode 100644 .idea/compiler.xml
create mode 100644 .idea/copyright/profiles_settings.xml
create mode 100644 .idea/encodings.xml
create mode 100644 .idea/libraries/sqlite-jdbc_3_8_11_2.xml
create mode 100644 .idea/misc.xml
create mode 100644 .idea/modules.xml
create mode 100644 .idea/uiDesigner.xml
create mode 100644 .idea/workspace.xml
create mode 100644 out/META-INF/MANIFEST.MF
```

8. Закидання змін з локального репозиторію на репозиторій в GitHub:

```
MINGW64/d/Projects/payment

create mode 100644 src/tableClasses/Payment.java
create mode 100644 src/utis/DialogManager.java

Andrew@Andrew-MINGW64 /d/Projects/payment (master)
$ git status
On branch master
Your branch is ahead of 'origin/master' by 1 commit.
(use "git push" to publish your local commits)
nothing to commit, working directory clean

Andrew@Andrew-MINGW64 /d/Projects/payment (master)
$ git push -u origin master
Counting objects: 94, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (84/84), done.
Writing objects: 100% (94/94), 5.46 MiB | 109.00 KiB/s, done.
Total 94 (delta 16), reused 0 (delta 0)
To https://github.com/androndep/payment.git
 8744401..8264574 master -> master
Branch master set up to track remote branch master from origin.
```

9. Додавання двох файлів комічення і пушання на GitHub:

```
MINGW64:/d/Projects/payment
Andrew@Andrew-MINGW64 /d/Projects/payment (master)
$ git status
On branch master
Your branch is up-to-date with 'origin/master'.
Untracked files:
  (use "git add <file>..." to include in what will be committed)

        src/images/diagram.png
        src/images/options-delete.png

nothing added to commit but untracked files present (use "git add" to track)
Andrew@Andrew-MINGW64 /d/Projects/payment (master)
$ git add *.png
Andrew@Andrew-MINGW64 /d/Projects/payment (master)
$ git commit -m "added two new icons"
[master 4417d45] added two new icons
2 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 src/images/diagram.png
create mode 100644 src/images/options-delete.png
Andrew@Andrew-MINGW64 /d/Projects/payment (master)
$ git push -u origin master
Counting objects: 6, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (6/6), done.
Writing objects: 100% (6/6), 34.21 KiB | 0 bytes/s, done.
Total 6 (delta 2), reused 0 (delta 0)
To https://github.com/androndep/payment.git
8264574..4417d45  master -> master
Branch master set up to track remote branch master from origin.
Andrew@Andrew-MINGW64 /d/Projects/payment (master)
$
```

10. Відновлення до стану, який був у 8-му пункті:

```
MINGW64:/d/Projects/payment
$ git checkout 8264
Note: checking out '8264'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by performing another checkout.

If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -b with the checkout command again. Example:

  git checkout -b <new-branch-name>

HEAD is now at 8264574... first commit
Andrew@Andrew-MINGW64 /d/Projects/payment ((8264574...))
$ git log
commit 82645742d69cc3a71edd774b0a4615cbcd8ba2d89
Author: Andrew <androndeputat@gmail.com>
Date: Thu Feb 25 00:30:15 2016 +0200

    first commit

commit 87444010734f286e142da17b4813a6ba95df3ae1
Author: Andrew Deputat <androndeputat@ro.ru>
Date: Thu Feb 25 00:11:48 2016 +0200

    Initial commit
Andrew@Andrew-MINGW64 /d/Projects/payment ((8264574...))
$
```

11. Створення клона репозиторія з GitHub в іншій папці:

```
MINGW64:/d/Projects/clone
Andrew@Andrew-MINGW64 /d/Projects/clone
$ git clone https://github.com/androndep/payment.git
Cloning into 'payment'...
remote: Counting objects: 103, done.
remote: Compressing objects: 100% (73/73), done.
remote: Total 103 (delta 18), reused 100 (delta 18), pack-reused 0
Receiving objects: 100% (103/103), 5.49 MiB | 1.82 MiB/s, done.
Resolving deltas: 100% (18/18), done.
Checking connectivity... done.
Andrew@Andrew-MINGW64 /d/Projects/clone
$
```


12. Створення нової гілки і додавання нового файлу:

```
MINGW64/d/Projects/payment

Andrew@Andrew-MINGW64 /d/Projects/payment (master)
$ git checkout -b version_1.1
Switched to a new branch 'version_1.1'

Andrew@Andrew-MINGW64 /d/Projects/payment (version_1.1)
$ git status
On branch version_1.1
Untracked files:
  (use "git add <file>..." to include in what will be committed)

        payment v1.0.exe

nothing added to commit but untracked files present (use "git add" to track)

Andrew@Andrew-MINGW64 /d/Projects/payment (version_1.1)
$ git add .exe
fatal: pathspec '.exe' did not match any files

Andrew@Andrew-MINGW64 /d/Projects/payment (version_1.1)
$ git add .
```

13. Комічення змін у новій гілці і синхронізація з репозиторієм на GitHub:

```
MINGW64/d/Projects/payment

Andrew@Andrew-MINGW64 /d/Projects/payment (version_1.1)
$ git commit -m "exe file added"
[version_1.1 185b199] exe file added
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 payment v1.0.exe

Andrew@Andrew-MINGW64 /d/Projects/payment (version_1.1)
$ git push -u origin version_1.1
Counting objects: 3, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 5.38 MiB | 133.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0)
To https://github.com/androndep/payment.git
 * [new branch]      version_1.1 -> version_1.1
Branch version_1.1 set up to track remote branch version_1.1 from origin.

Andrew@Andrew-MINGW64 /d/Projects/payment (version_1.1)
$ |
```

14. Об'єднання нової гілки з основною:

```
MINGW64/d/Projects/payment

Delta compression using up to 4 threads.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 5.38 MiB | 133.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0)
To https://github.com/androndep/payment.git
 * [new branch]      version_1.1 -> version_1.1
Branch version_1.1 set up to track remote branch version_1.1 from origin.

Andrew@Andrew-MINGW64 /d/Projects/payment (version_1.1)
$ git merge master
Already up-to-date.

Andrew@Andrew-MINGW64 /d/Projects/payment (version_1.1)
$ git status
On branch version_1.1
Your branch is up-to-date with 'origin/version_1.1'.
nothing to commit, working directory clean

Andrew@Andrew-MINGW64 /d/Projects/payment (version_1.1)
$
```


15. Змінення файла який був доданий в 12-му пункті в основній branch:

```
MINGW64/d/Projects/payment
Andrew@Andrew:~$ MINGW64 /d/Projects/payment (master)
$ git status
On branch master
Your branch is up-to-date with 'origin/master'.
nothing to commit, working directory clean

Andrew@Andrew:~$ MINGW64 /d/Projects/payment (master)
$ git status
On branch master
Your branch is up-to-date with 'origin/master'.
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

        modified:   README.md

no changes added to commit (use "git add" and/or "git commit -a")

Andrew@Andrew:~$ MINGW64 /d/Projects/payment (master)
$ git add .

Andrew@Andrew:~$ MINGW64 /d/Projects/payment (master)
$ git commit -m"changes in README.md"
[master b3c970c] changes in README.md
1 file changed, 2 insertions(+), 1 deletion(-)

Andrew@Andrew:~$ MINGW64 /d/Projects/payment (master)
$ git push -u origin master
Counting objects: 3, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 281 bytes | 0 bytes/s, done.
Total 3 (delta 1), reused 0 (delta 0)
To https://github.com/androndep/payment.git
4417d45..b3c970c master -> master
Branch master set up to track remote branch master from origin.

Andrew@Andrew:~$ MINGW64 /d/Projects/payment (master)
$
```

16. Переключення на гілку, яку було створено в пункті 12 і також зроблені зміни в файлі, що додано в пункті 12:

```
MINGW64/d/Projects/payment
Andrew@Andrew:~$ MINGW64 /d/Projects/payment (master)
$ git checkout version_1.1
Switched to branch 'version_1.1'
Your branch is up-to-date with 'origin/version_1.1'.

Andrew@Andrew:~$ MINGW64 /d/Projects/payment (version_1.1)
$ git status
On branch version_1.1
Your branch is up-to-date with 'origin/version_1.1'.
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

        modified:   README.md

no changes added to commit (use "git add" and/or "git commit -a")

Andrew@Andrew:~$ MINGW64 /d/Projects/payment (version_1.1)
$ git add .

Andrew@Andrew:~$ MINGW64 /d/Projects/payment (version_1.1)
$ git commit -m"second changes in README.md but in version_1.1"
[version_1.1 2bd344b] second changes in README.md but in version_1.1
1 file changed, 2 insertions(+), 1 deletion(-)

Andrew@Andrew:~$ MINGW64 /d/Projects/payment (version_1.1)
$ git push -u origin version_1.1
Counting objects: 3, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 303 bytes | 0 bytes/s, done.
Total 3 (delta 1), reused 0 (delta 0)
To https://github.com/androndep/payment.git
185b199..2bd344b version_1.1 -> version_1.1
Branch version_1.1 set up to track remote branch version_1.1 from origin.

Andrew@Andrew:~$ MINGW64 /d/Projects/payment (version_1.1)
$ |
```

17. Об'єднання створеної гілки з основною. Вирішення конфлікту:

MINGW64:/d/Projects/payment

```
Andrew@Andrew- MINGW64 /d/Projects/payment (version_1.1)
$ git merge master
Auto-merging README.md
CONFLICT (content): Merge conflict in README.md
Automatic merge failed; fix conflicts and then commit the result.

Andrew@Andrew- MINGW64 /d/Projects/payment (version_1.1|MERGING)
$ git status
On branch version_1.1
Your branch is up-to-date with 'origin/version_1.1'.
You have unmerged paths.
  (fix conflicts and run "git commit")

Unmerged paths:
  (use "git add <file>..." to mark resolution)

        both modified:   README.md

no changes added to commit (use "git add" and/or "git commit -a")

Andrew@Andrew- MINGW64 /d/Projects/payment (version_1.1|MERGING)
$ git add .

Andrew@Andrew- MINGW64 /d/Projects/payment (version_1.1|MERGING)
$ git commit -m"problem fixed"
[version_1.1 bd0a5a5] problem fixed

Andrew@Andrew- MINGW64 /d/Projects/payment (version_1.1)
$ git push -u origin version_1.1
Counting objects: 3, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 328 bytes | 0 bytes/s, done.
Total 3 (delta 1), reused 0 (delta 0)
To https://github.com/androndep/payment.git
  2bd344b..bd0a5a5  version_1.1 -> version_1.1
Branch version_1.1 set up to track remote branch version_1.1 from origin.

Andrew@Andrew- MINGW64 /d/Projects/payment (version_1.1)
$ git checkout master
Switched to branch 'master'
Your branch is up-to-date with 'origin/master'.

Andrew@Andrew- MINGW64 /d/Projects/payment (master)
$ git merge version_1.1
Updating b3c970c..bd0a5a5
Fast-forward
 README.md      | 7 ++++++-
 payment v1.0.exe | Bin 0 -> 5696523 bytes
 2 files changed, 6 insertions(+), 1 deletion(-)
 create mode 100644 payment v1.0.exe

Andrew@Andrew- MINGW64 /d/Projects/payment (master)
$ git add .

Andrew@Andrew- MINGW64 /d/Projects/payment (master)
$ git commit -m"problem fixed too"
On branch master
Your branch is ahead of 'origin/master' by 3 commits.
  (use "git push" to publish your local commits)
nothing to commit, working directory clean

Andrew@Andrew- MINGW64 /d/Projects/payment (master)
$ git push origin master
Total 0 (delta 0), reused 0 (delta 0)
To https://github.com/androndep/payment.git
  b3c970c..bd0a5a5  master -> master

Andrew@Andrew- MINGW64 /d/Projects/payment (master)
$
```

Висновок: здобув необхідні знання та навички для роботи з системою керування версіями Git.