

# Hudson:

система неперервної інтеграції  
(Continuous Integration).

## Використання JUnit в Hudson.

Виконали

студенти групи ФІ-92 ФТІ

Степанян Ігор та Лосінець Юрій

# Build & deployment

- Компіляція
- Тестування
- Збірка
- Копіювання на віддалений сервер
- Запуск

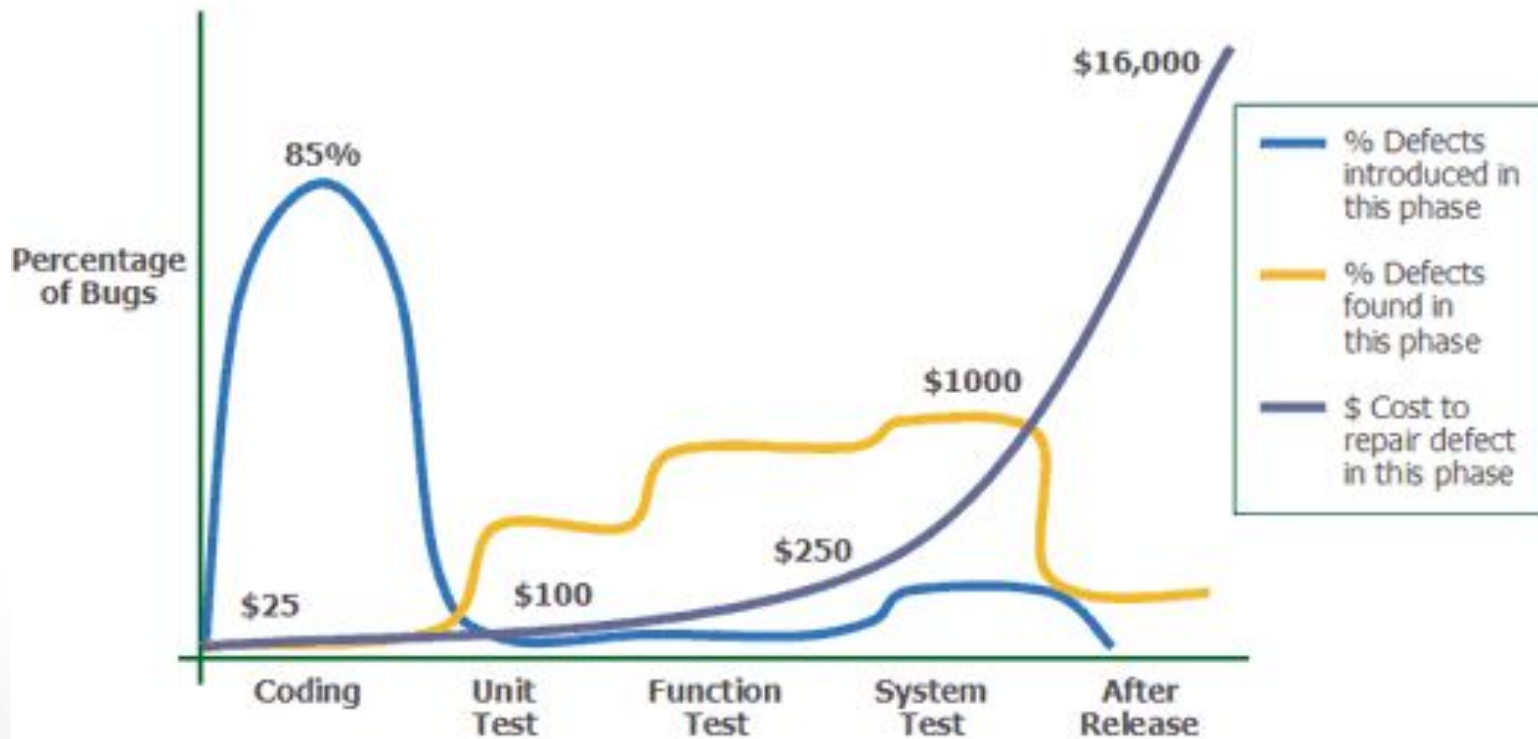
# Build & deployment

- Компіляція
- Тестування
- Збірка
- Копіювання на віддалений сервер
- Запуск



автоматизація  
?

# Тестування



# Системи контролю версій

- Зберігання версій файлів, причому зазвичай зберігаються тільки зміни між попередньою і поточною версією і таким чином сховище не росте дуже швидко
- Можливість отримати будь-які попередні версії збережених файлів
- Перегляд змін внесених між заданими у запиті версіями
- Збереження та перегляд коментарів та авторів щодо внесених змін.

# Continuous Integration

*Continuous Integration is a software development practice where members of a team integrate their work frequently, usually each person integrates at least daily - leading to multiple integrations per day. Each integration is verified by an automated build (including test) to detect integration errors as quickly as possible. Many teams find that this approach leads to significantly reduced integration problems and allows a team to develop cohesive software more rapidly.*

*... taken from Martin Fowlers article on CI*

# Практика

**Complex\_Stepanyan**

Инструменти:

**Language:** Java

**IDE:** NetBeans

**Testing engine:** JUnit

**CVS:** Mercurial

**CI Server:** Hudson

# Mercurial

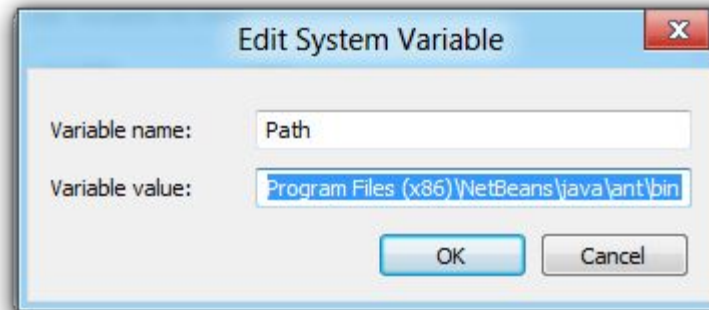
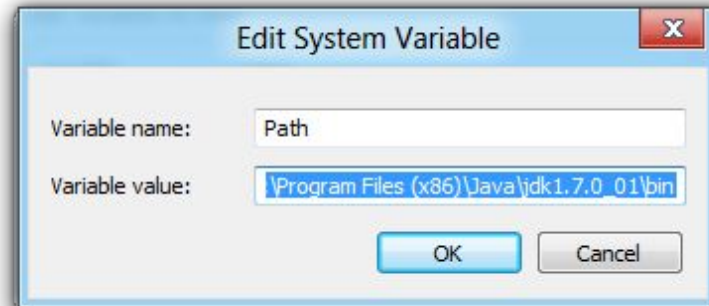
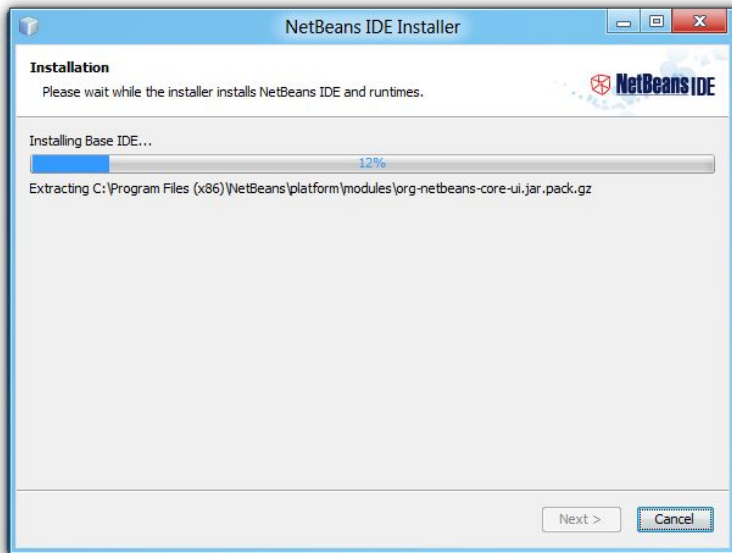
**Mercurial** — вільна розподілена система керування версій файлів та спільної роботи.

## Переваги:

- незалежна від об'єму коду висока швидкодія;
- компактне зберігання даних в проіндексованому і стислому виді;
- розподілена модель розробки, що допускає довільне злиття окремих децентралізованих репозиторіїв;
- вбудовані засоби резервного копіювання і перевірки цілісності;
- звичний CVS-подібний набір команд;
- великий вибір GUI- і веб-інтерфейсів;
- підтримка кількох моделей організації репозиторія: централізована cvs-подібна, розподілена пів-ієрархічна і ієрархічна.



# Інсталяція: NetBeans



# Інсталяція: TortoiseHg



# Інсталяція: Hudson

```
C:\Windows\system32\cmd.exe - java -jar hudson.war
C:\>java -version
java version "1.7.0_01"
Java(TM) SE Runtime Environment (build 1.7.0_01-b08)
Java HotSpot(TM) Client VM (build 21.1-b02, mixed mode, sharing)

C:\>java -jar hudson.war
Running from: C:\CI\hudson.war
[Winstone 2011/12/21 00:08:09] - Beginning extraction from war file
gEe 21, 2011 12:08:21 AM hudson.WebAppMain contextInitialized
INFO: Home directory: C:\Users\Root\hudson
gEe 21, 2011 12:08:21 AM hudson.util.CharacterEncodingFilter init
INFO: CharacterEncodingFilter initialized. DISABLE_FILTER: false FORCE_ENCODING:
false
[Winstone 2011/12/21 00:08:22] - AJPI3 Listener started: port=8009
[Winstone 2011/12/21 00:08:22] - HTTP Listener started: port=8080
Using one-time self-signed certificate
[Winstone 2011/12/21 00:08:23] - Winstone Servlet Engine v0.9.10 running: contro
lPort=disabled
gEe 21, 2011 12:08:24 AM org.hudsonci.inject.internal.SmoothieContainerBootstra
p bootstrap
INFO: Bootstrapping Smoothie
gEe 21, 2011 12:08:28 AM hudson.PluginManager createPluginStrategy
INFO: Plugin strategy: org.hudsonci.inject.internal.plugin.DelegatingPluginStrat
egy
```

```
C:\Windows\system32\cmd.exe - java -jar hudson.war
gEe 21, 2011 12:09:08 AM com.sun.jersey.server.impl.application.WebApplicationI
mpl_initiate
INFO: Initiating Jersey application, version 'Jersey: 1.5 01/14/2011 12:36 PM'
gEe 21, 2011 12:09:09 AM com.sun.jersey.server.impl.application.WebApplicationI
mpl_initiate
INFO: Initiating Jersey application, version 'Jersey: 1.5 01/14/2011 12:36 PM'
gEe 21, 2011 12:09:09 AM com.sun.jersey.server.impl.application.WebApplicationI
mpl_initiate
INFO: Initiating Jersey application, version 'Jersey: 1.5 01/14/2011 12:36 PM'
gEe 21, 2011 12:09:10 AM com.sun.jersey.server.impl.application.WebApplicationI
mpl_initiate
INFO: Initiating Jersey application, version 'Jersey: 1.5 01/14/2011 12:36 PM'
gEe 21, 2011 12:09:10 AM com.sun.jersey.server.impl.application.WebApplicationI
mpl_initiate
INFO: Initiating Jersey application, version 'Jersey: 1.5 01/14/2011 12:36 PM'
gEe 21, 2011 12:09:11 AM org.hudsonci.maven.plugin.install.SlaveBundleInstaller
consume
INFO: Maven slave bundle installed
gEe 21, 2011 12:09:11 AM org.hudsonci.rest.plugin.RestPlugin enable
INFO: API provider JAX-RS (Jersey) enabled
gEe 21, 2011 12:09:11 AM org.hudsonci.rest.plugin.RestPlugin enable
INFO: API provider Bayeux (CometD) enabled
gEe 21, 2011 12:09:11 AM org.hudsonci.events.ready.ReadyDetector run
INFO: Hudson is ready.
```

http://localhost:8080/install/?

imo

File Edit View Favorites Tools Help

## Hudson

Hudson + Install as Windows Service

ENABLE AUTO REFRESH

- New Job
- Manage Hudson
- People
- Build History
- New View

### Build Queue

No builds in the queue.

### Build Executor Status


Status 0/2

Idle

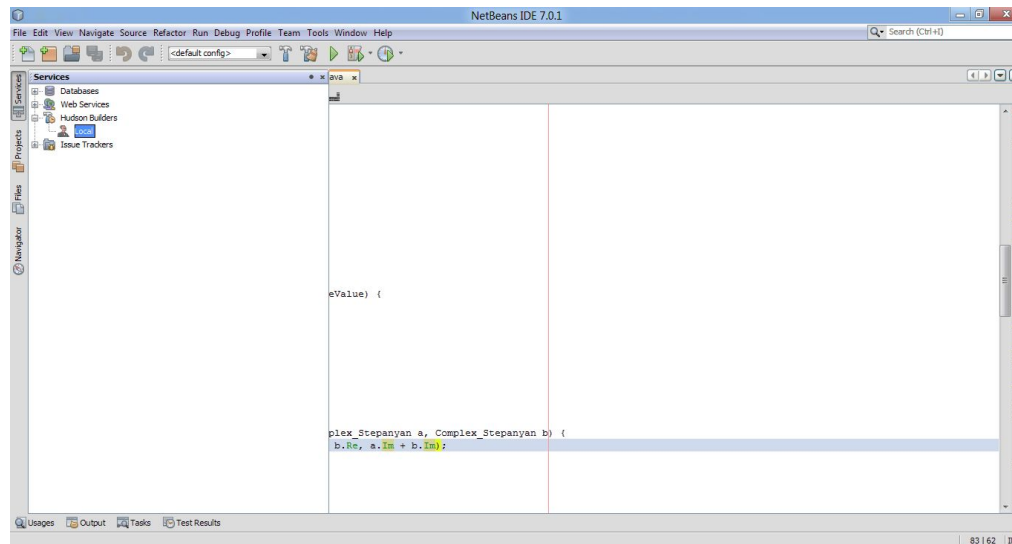
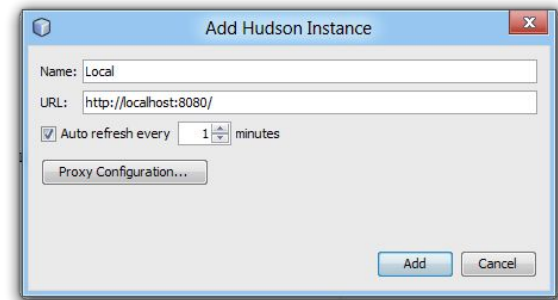
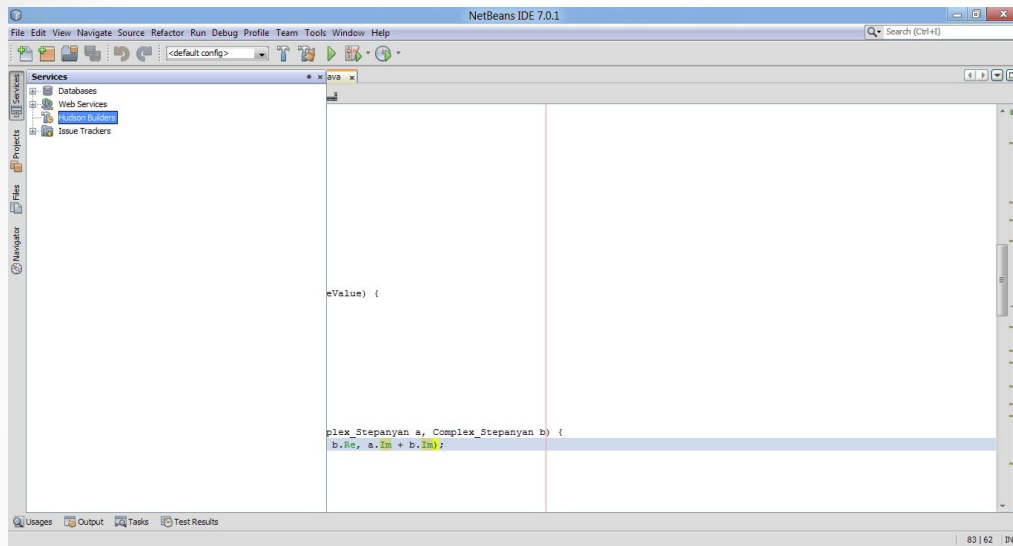
## Install as Windows Service

Installing Hudson as a Windows service allows you to start Hudson as soon as the machine starts, and regardless of who is interactively using Hudson.

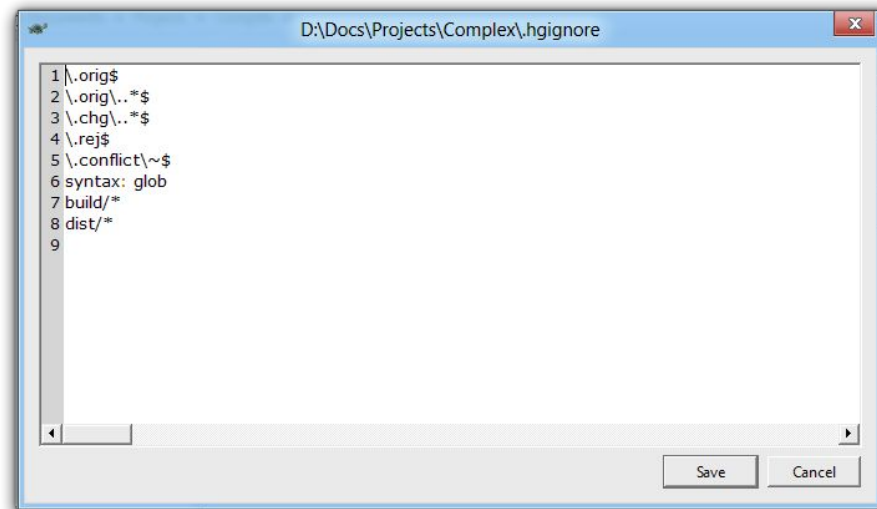
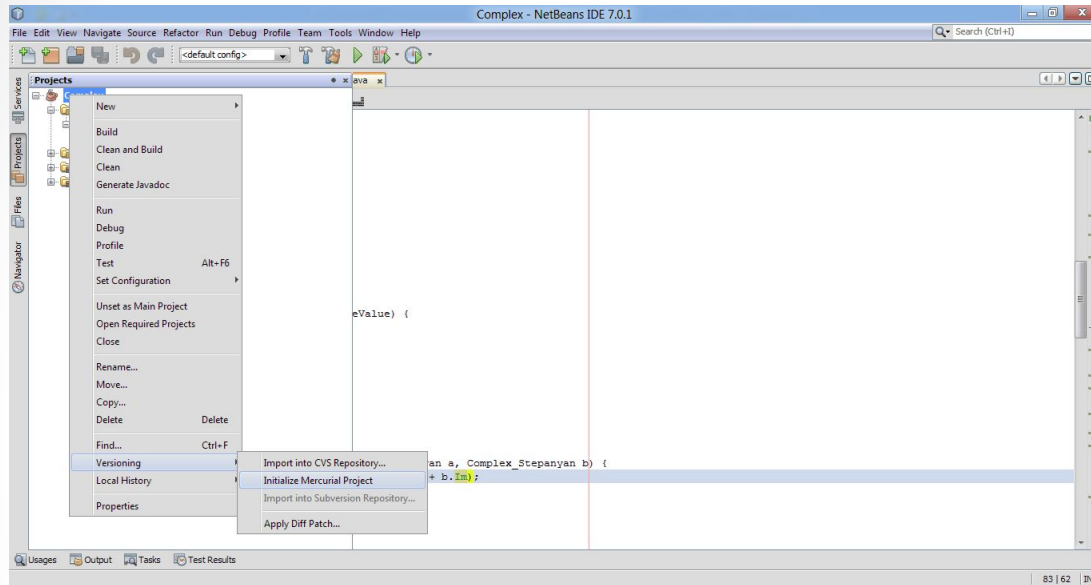
Installation Directory:



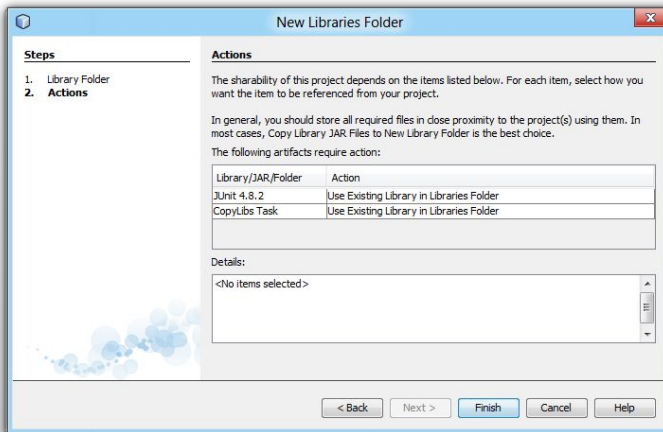
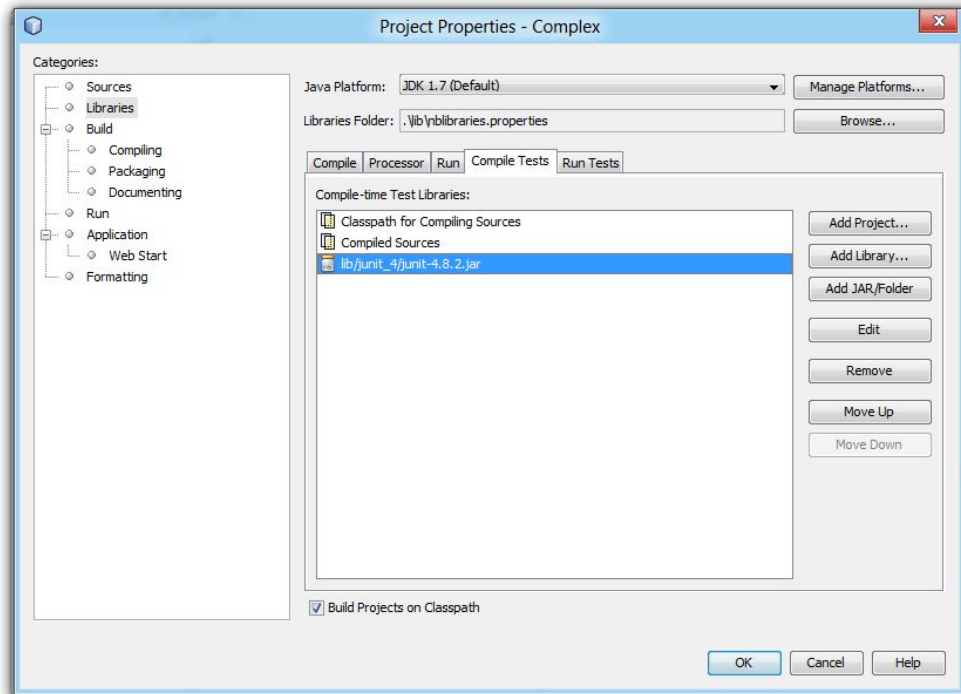
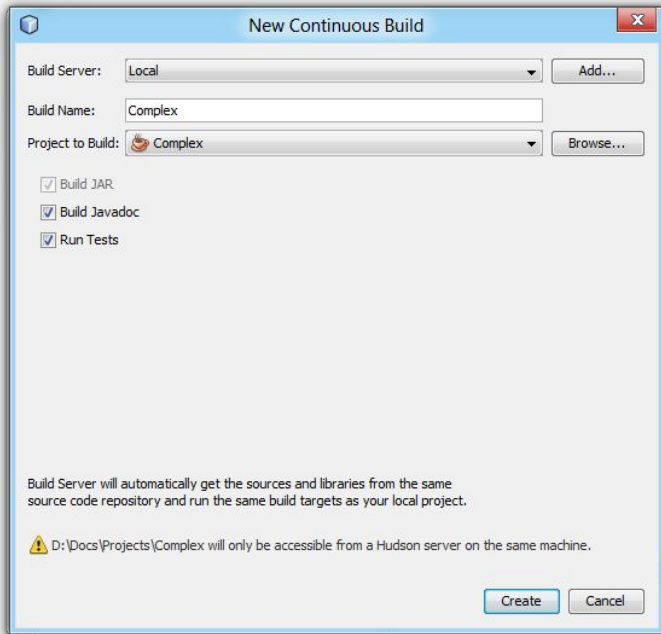
# Інсталяція: NetBeans+Hudson



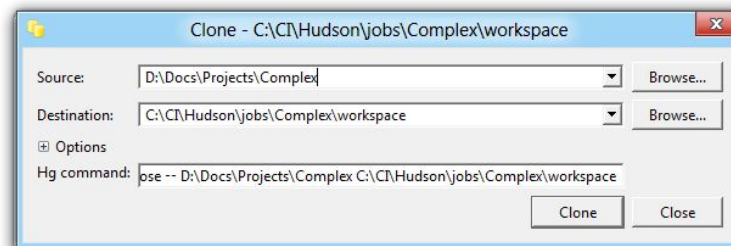
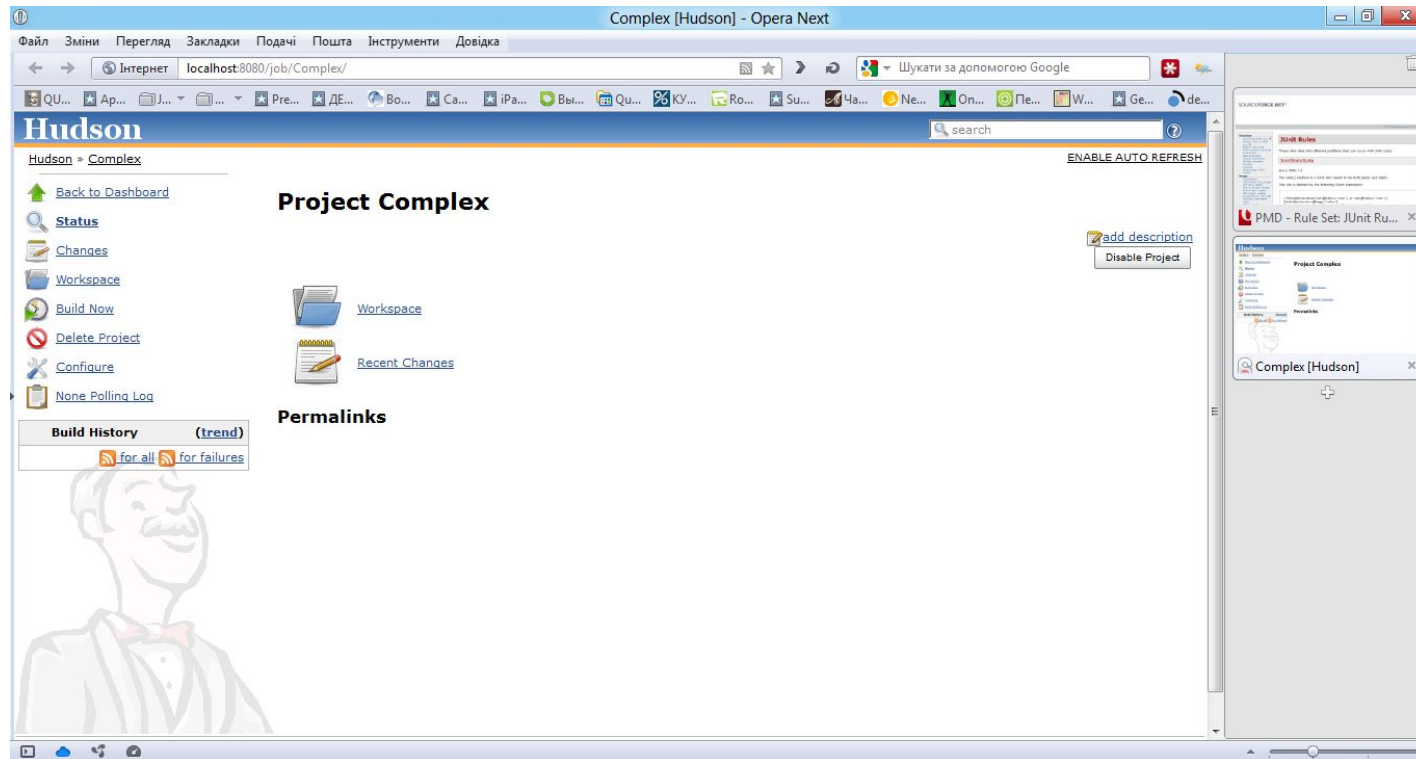
# Проект: ініціалізація



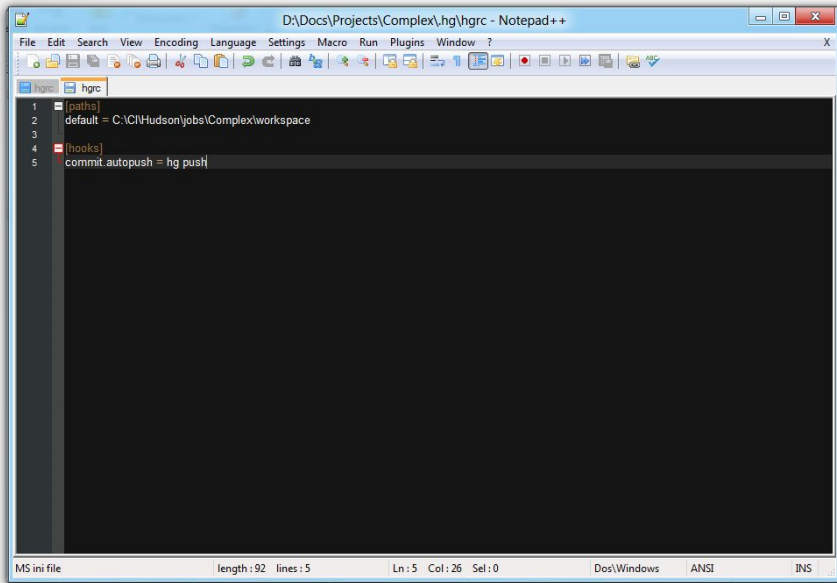
# Проект: ініціалізація



# Проект: клонування



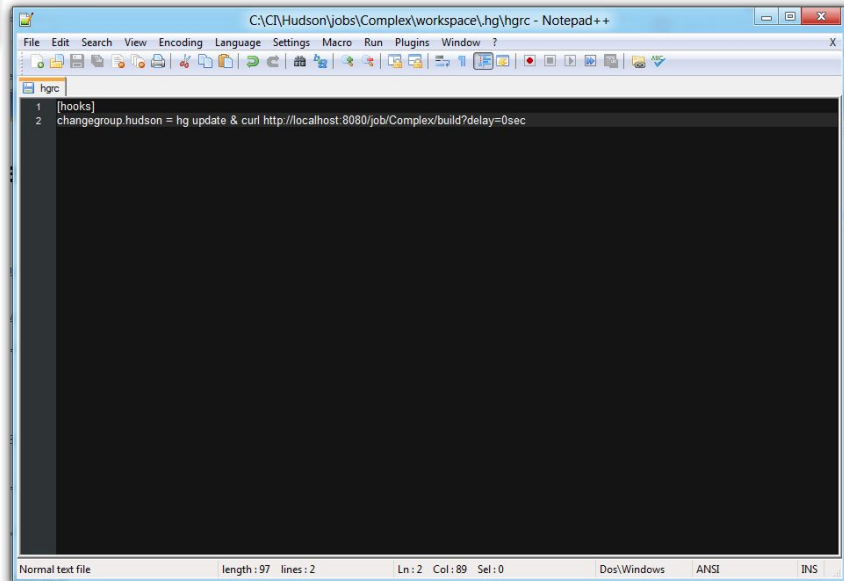
# Проект: hg-hooks



A screenshot of a Notepad++ window titled "D:\Docs\Projects\Complex\hg\hgrc - Notepad++". The window shows the content of the hgrc file with the following text:

```
1 [paths]
2 default = C:\CI\Hudson\jobs\Complex\workspace
3
4 [hooks]
5 commit.autopush = hg push
```

The status bar at the bottom indicates "MS ini file", "length : 92", "lines : 5", "Ln : 5", "Col : 26", "Sel : 0", "Dos\Windows", "ANSI", and "INS".



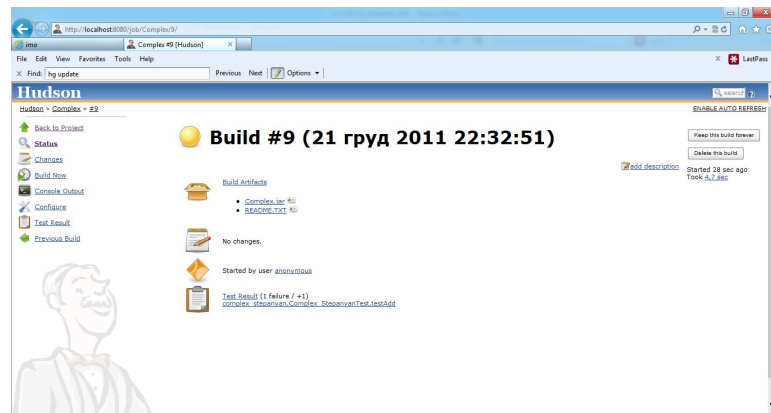
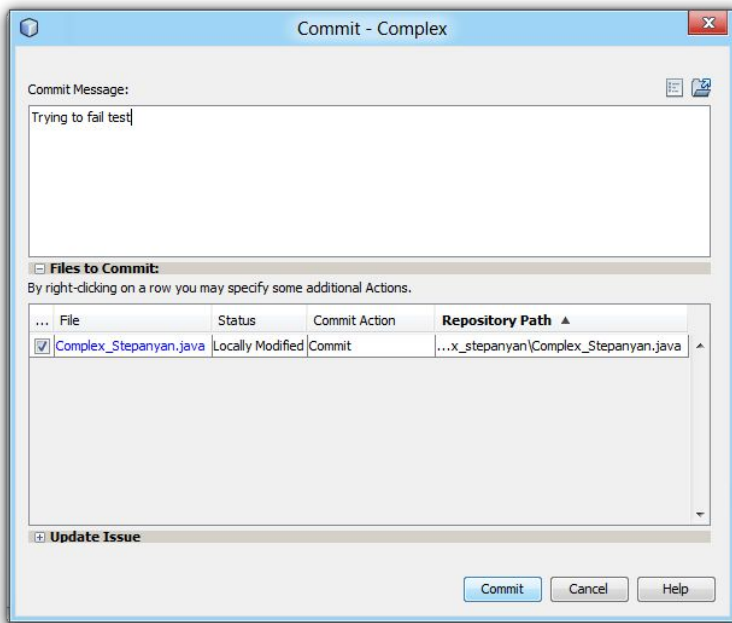
A screenshot of a Notepad++ window titled "C:\CI\Hudson\jobs\Complex\workspace\hg\hgrc - Notepad++". The window shows the content of the hooks section of the hgrc file with the following text:

```
1 [hooks]
2 changegroup.hudson = hg update & curl http://localhost:8080/job/Complex/build?delay=0sec
```

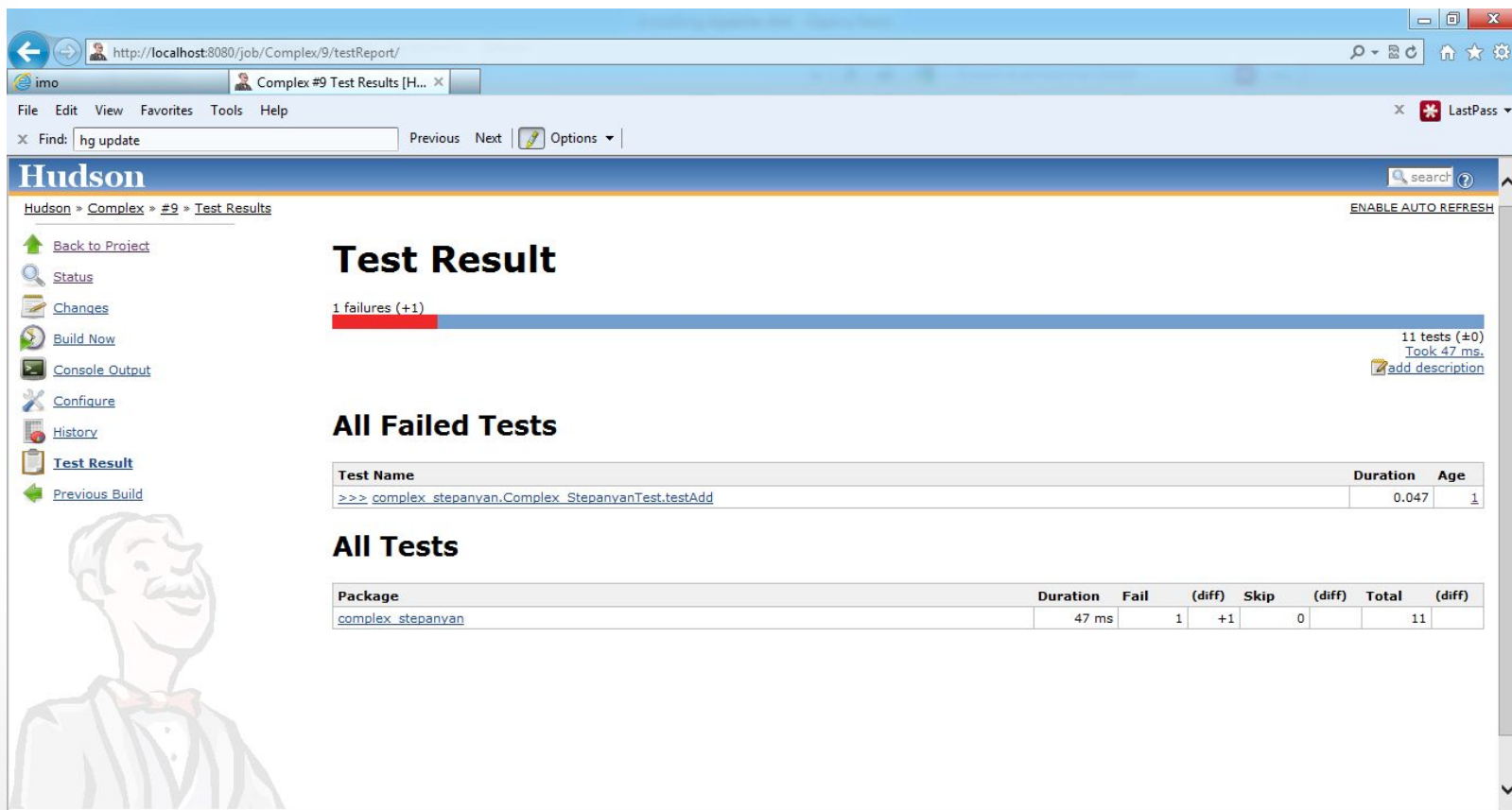
The status bar at the bottom indicates "Normal text file", "length : 97", "lines : 2", "Ln : 2", "Col : 89", "Sel : 0", "Dos\Windows", "ANSI", and "INS".



# Проект: fail-commit



# Проект: fail-commit



The screenshot shows the Hudson web interface for a test report. The browser address bar shows `http://localhost:8080/job/Complex/9/testReport/`. The page title is "Complex #9 Test Results [Hudson]". The main content area displays "Test Result" with a summary of "1 failures (+1)". A progress bar shows 1 failure out of 11 tests. The "All Failed Tests" section contains a table with one entry: `>>> complex_stepanvan.Complex_StepanvanTest.testAdd` with a duration of 0.047. The "All Tests" section contains a table with one entry: `complex_stepanvan` with a duration of 47 ms, 1 failure, and 11 total tests.

**Test Result**

1 failures (+1)


11 tests (±0)  
Took 47 ms.  
[add description](#)

**All Failed Tests**

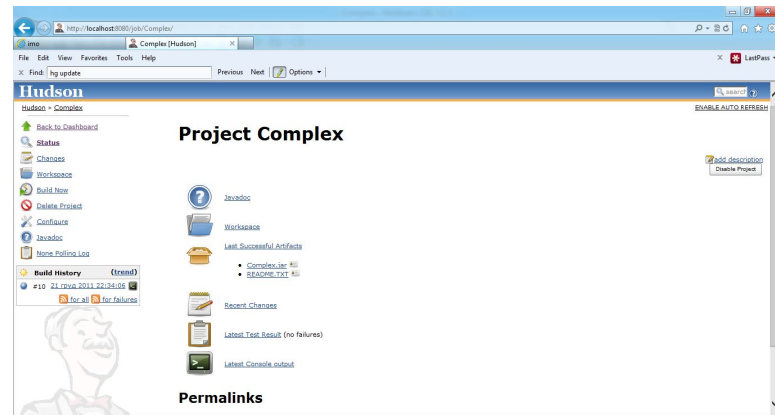
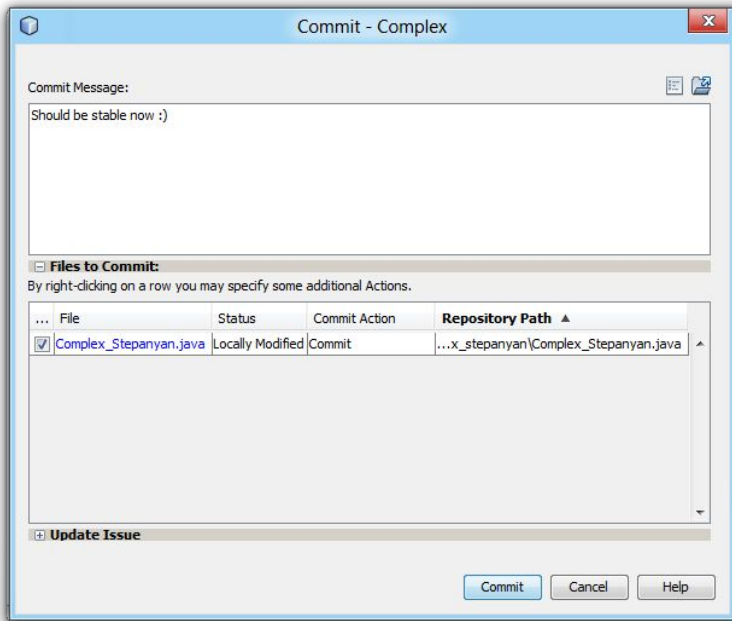
Test Name	Duration	Age
>>> <code>complex_stepanvan.Complex_StepanvanTest.testAdd</code>	0.047	<a href="#">1</a>

**All Tests**

Package	Duration	Fail	(diff)	Skip	(diff)	Total	(diff)
<code>complex_stepanvan</code>	47 ms	1	+1	0		11	



# Проект: success-commit



# Проект: success-commit

The screenshot displays a web browser window with the following details:

- Address Bar:** `http://localhost:8080/job/Complex/javadoc?`
- Page Title:** `Complex_Stepanyan`
- Navigation:** Back, Forward, Home, Star, Settings icons.
- Menu:** File, Edit, View, Favorites, Tools, Help.
- Search:** Find: `hg update` | Previous | Next | Options
- Sidebar (All Classes):** `Complex_Stepanyan`
- Constructors Section:**
  - Constructor and Description**
  - `Complex_Stepanyan ()`  
Default constructor (sets number to 0)
  - `Complex_Stepanyan (double number)`  
Constructor for setting number from real
  - `Complex_Stepanyan (double reValue, double imValue)`  
Constructor for setting both real and imag parts of complex number at once
- Method Summary Section:**
  - Methods**
  - | Modifier and Type                     | Method and Description  |
|---------------------------------------|---|
| <code>double</code>                   | <code>abs ()</code><br>Get absolute value of complex number                                 |
| <code>static Complex_Stepanyan</code> | <code>add (Complex_Stepanyan a, Complex_Stepanyan b)</code><br>Add two complex numbers      |
| <code>static Complex_Stepanyan</code> | <code>div (Complex_Stepanyan a, Complex_Stepanyan b)</code><br>Divide two complex numbers   |
| <code>boolean</code>                  | <code>equals (java.lang.Object obj)</code>  |
| <code>double</code>                   | <code>getIm ()</code><br>Get imag part of complex number                                    |
| <code>double</code>                   | <code>getRe ()</code><br>Get real part of complex number                                    |
| <code>int</code>                      | <code>hashCode ()</code>  |
| <code>static Complex_Stepanyan</code> | <code>mul (Complex_Stepanyan a, Complex_Stepanyan b)</code><br>Multiply two complex numbers |

Дякую за увагу!

Питання? Зауваження?