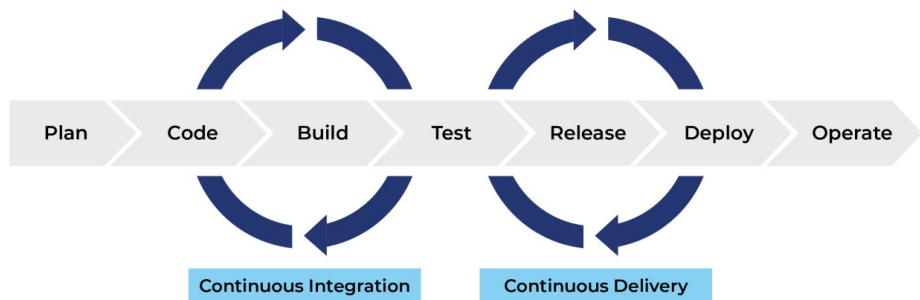


CI/CD

CI/CD





Лекція 5. CI/CD

План

1. Неперервна інтеграція
2. Безперервна доставка
3. Jenkins

1. Неперервна інтеграція

Неперервна інтеграція (Continuous Integration, CI) – це практика розробки програмного забезпечення, яка полягає в автоматичному та регулярному об'єднанні (інтеграції) у спільному репозиторії коду, написаного різними членами команди розробників.

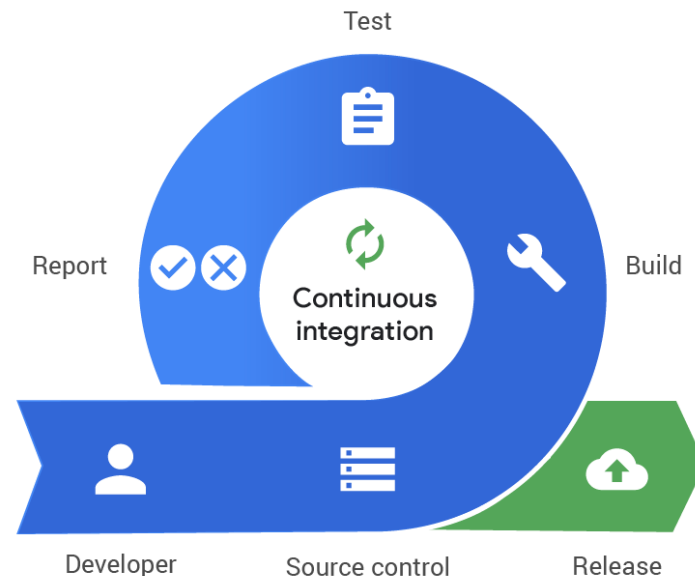
Основною метою неперервної інтеграції є раннє виявлення та усунення конфліктів, помилок і проблем інтеграції, що можуть виникати в процесі розробки.

Процес неперервної інтеграція включає:

- 1) автоматичну збірку** бінарних файлів з початкового коду після кожного коміту в репозиторії;
- 2) автоматичне тестування** для перевірки працездатності додатків після інтеграції нового коду;
- 3) інтеграція збірок і тестів** у автоматичному процесі CI;
- 4) виявлення конфліктів** між комітами різних розробників, які можуть викликати помилки інтеграції;

1. Неперервна інтеграція

- 5) **автоматичне розгортання** зібраного та протестованого додатку на тестових або проміжних серверах;
- 6) **моніторинг результатів** збірок і цілей тестів із швидким реагуванням на помилки;
- 7) **автоматичне повідомлення** команди розробників про результати збірки та тестування;
- 8) **відновлення** процесу CI після збоїв і автоматичне повідомлення про збої;
- 9) **використання спеціальних інструментів** (і платформ) для реалізації процесів безперервної інтеграції та доставки.

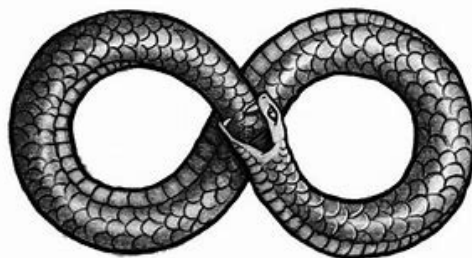


1. Неперервна інтеграція

Таким чином, неперервна інтеграція – це сучасний метод командної розробки програмного забезпечення з регулярної інтеграцією результатів роботи окремих розробників. При цьому кожна інтеграція перевіряється автоматичною збіркою для виявлення помилок.

Ця концепція була вперше представлена більше двох десятиліть тому, щоб уникнути **інтеграційного пекла** (Integration Hell), коли прикінцева інтеграція проєкту, як правило, приводила до великої кількості проблем і багатоденного повторного виправлення коду. Що, у свою чергу, спричиняло виникнення великої кількості помилок у релізі.

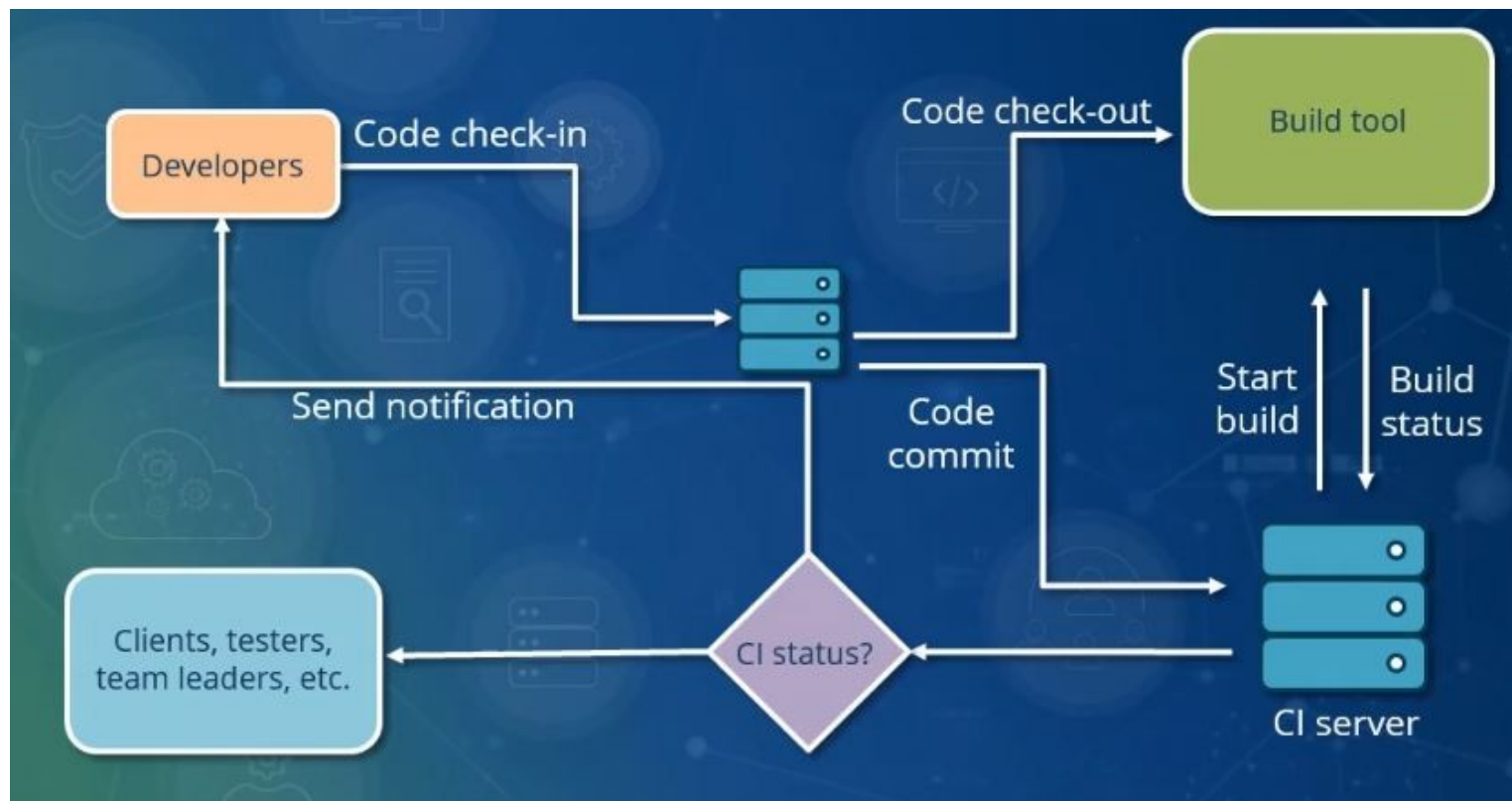
При неперервній інтеграції після фіксації коду програмне забезпечення відразу ж збирається і тестується. У великому проєкті коміти можуть виконуватися багато разів протягом робочого дня. При кожному коміті код створюється і тестується. Якщо тест пройдений, збірка перевіряється на предмет розгортання. Якщо розгортання пройшло успішно, код передається замовнику. Отже, **фіксація, збірка, тестування та розгортання** – це неперервний процес, звідки й походить його назва – CI.



1. Неперервна інтеграція

Впровадження неперервної інтеграції дозволяє:

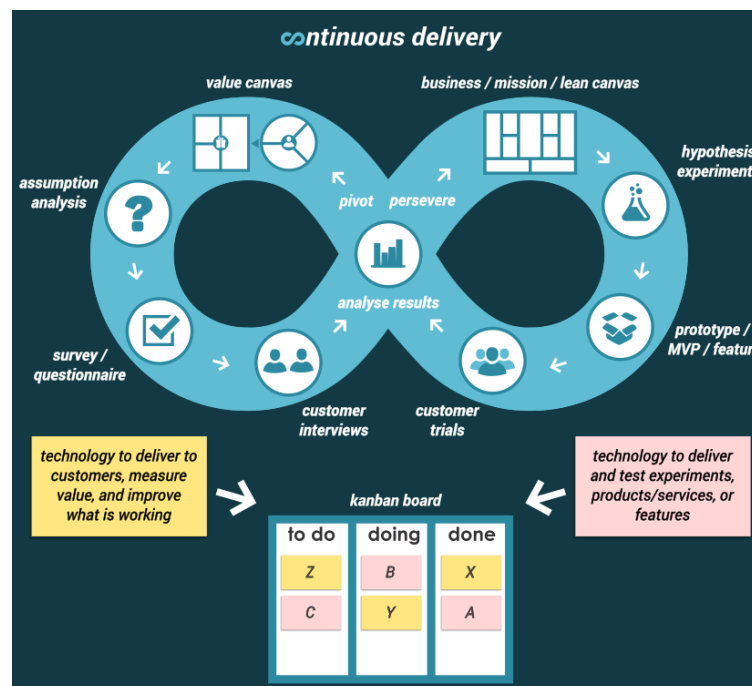
- прискорити процес розробки;
- підвищити якість програмного коду;
- зменшити ризики, пов'язані з інтеграцією та злиттям коду;
- автоматизувати технологічні процеси;
- покращити співпрацю між членами команди розробки.



2. Безперервна доставка

Безперервна доставка (Continuous Delivery, CD) – це практика розробки програмного забезпечення, яка розширює принцип неперервної інтеграції і включає автоматичне розгортання (доставку) програми в цільове середовище (тестовий сервер, виробничий сервер і тому подібне) після успішного завершення процесу неперервної інтеграції та тестування.

Основна мета безперервної доставки – забезпечити готовність до випуску нових версій програми у будь-який момент часу.



2. Безперервна доставка

Практика безперервної доставки включає:

- 1) **автоматичне розгортання** програми на цільових серверах або платформах після успішного проходження тестів у процесі неперервної інтеграції;
- 2) **створення автоматизованого ланцюга доставки** (pipeline), який включає етапи збирання, тестування й розгортання;
- 3) **управління конфігураціями** цільових середовищ та ресурсів для забезпечення цілісності та відтворюваності розгортання;
- 4) **використання систем управління версіями** для контролю змін у коді, конфігурації та ресурсів програми;
- 5) **автоматичне сповіщення** розробників про статус процесу доставки програмного продукту;
- 6) **встановлення моніторингу продуктивності та доступності** програми в реальному часі, а також отримання зворотного зв'язку для швидкого реагування на збої та проблеми;
- 7) **розробка процедур резервного копіювання даних** та можливості відновлення за потреби;

2. Безперервна доставка

- 8) **використання етапу попереднього розгортання (staging)** для перевірки програми перед фінальним розгортанням у виробництві;
- 9) **оперативне розгортання** – впровадження нової версії програми шляхом поетапного перемикавання користувачів на нову версію без простоїв та перерв у роботі (**rolling deployment**);
- 10) **використання мікросервісної архітектури** для більш гнучкого та швидкого постачання.

Метою безперервної доставки є створення процесу розробки, який дозволяє командам постачати нові версії програм у будь-який момент часу з мінімальними ризиками та максимальною автоматизацією.

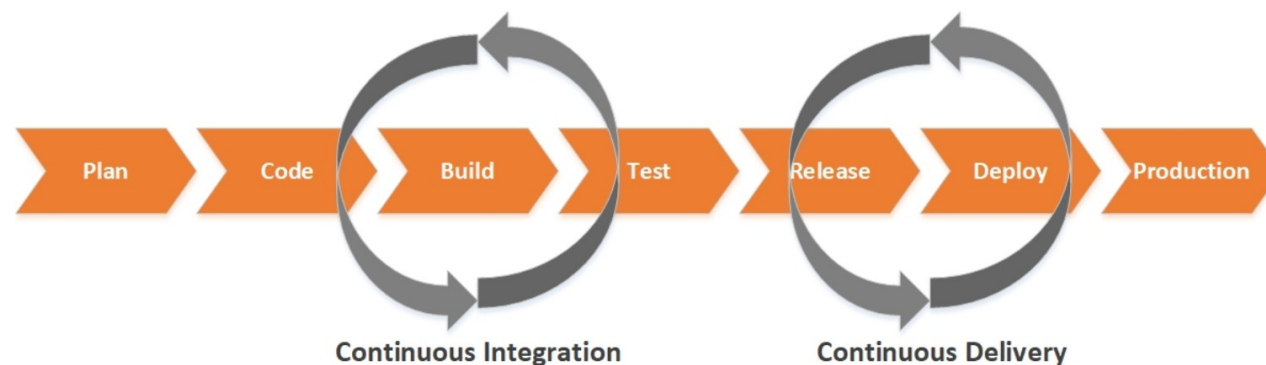
Використання безперервної доставки дозволяє компаніям швидко реагувати на зміни у вимогах клієнтів та ринку, покращувати якість та стабільність продукту та збільшувати ефективність розробки.

2. Безперервна доставка

CI/CD потребують **безперервного тестування**, оскільки кінцева мета — розробка якісного програмного забезпечення.

Безперервне тестування часто реалізується у вигляді набору різноманітних автоматизованих тестів, які виконуються в **CI/CD-конвеєрі (pipeline)**.

Ефективна практика CI/CD дозволяє реалізувати безперервне розгортання програм: при успішному проходженні коду через CI/CD-конвеєр, збірки автоматично розгортаються у продакшн.



2. Безперервна доставка

Типовий CD-конвеєр складається з наступних основних етапів:

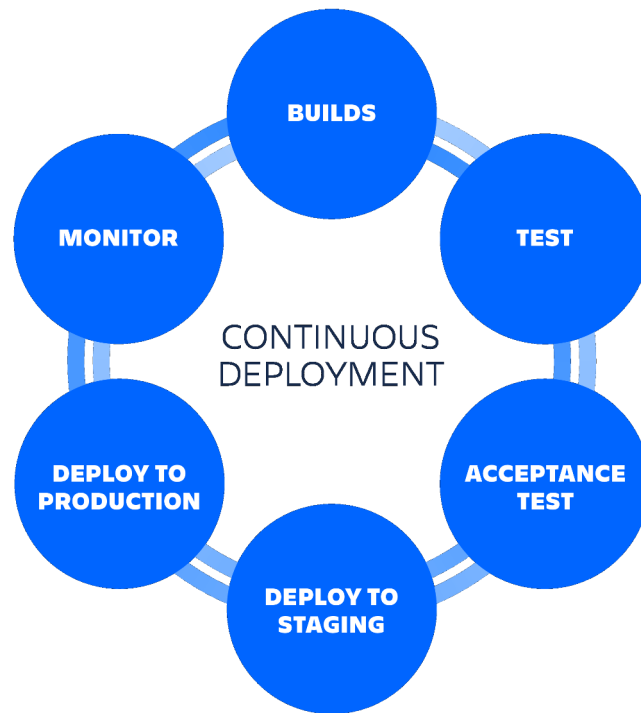
- збирання;
- тестування;
- розгортання.

Більш складні конвеєри можуть включати наступні етапи:

- отримання коду із системи управління версіями та виконання збирання;
- «інфраструктура як код» (автоматизоване налаштування обчислювальних та мережевих ресурсів методом їх опису у вигляді програмного коду);
- копіювання коду у цільове середовище;
- налаштування змінних оточення для цільового середовища;
- розгортання компонентів програми (вебсервери, бази даних тощо);
- виконання додаткових дій (наприклад, перезапуск або виклик сервісів), необхідних для працездатності внесених змін;
- автоматичне тестування та відкат змін у разі провалу тестів;
- логування та надсилання сповіщень про стан доставки.

1997, 1998, 1999, 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 2019, 2020, 2021, 2022, 2023, 2024, 2025, 2026, 2027, 2028, 2029, 2030, 2031, 2032, 2033, 2034, 2035, 2036, 2037, 2038, 2039, 2040, 2041, 2042, 2043, 2044, 2045, 2046, 2047, 2048, 2049, 2050, 2051, 2052, 2053, 2054, 2055, 2056, 2057, 2058, 2059, 2060, 2061, 2062, 2063, 2064, 2065, 2066, 2067, 2068, 2069, 2070, 2071, 2072, 2073, 2074, 2075, 2076, 2077, 2078, 2079, 2080, 2081, 2082, 2083, 2084, 2085, 2086, 2087, 2088, 2089, 2090, 2091, 2092, 2093, 2094, 2095, 2096, 2097, 2098, 2099, 2100, 2101, 2102, 2103, 2104, 2105, 2106, 2107, 2108, 2109, 2110, 2111, 2112, 2113, 2114, 2115, 2116, 2117, 2118, 2119, 2120, 2121, 2122, 2123, 2124, 2125, 2126, 2127, 2128, 2129, 2130, 2131, 2132, 2133, 2134, 2135, 2136, 2137, 2138, 2139, 2140, 2141, 2142, 2143, 2144, 2145, 2146, 2147, 2148, 2149, 2150, 2151, 2152, 2153, 2154, 2155, 2156, 2157, 2158, 2159, 2160, 2161, 2162, 2163, 2164, 2165, 2166, 2167, 2168, 2169, 2170, 2171, 2172, 2173, 2174, 2175, 2176, 2177, 2178, 2179, 2180, 2181, 2182, 2183, 2184, 2185, 2186, 2187, 2188, 2189, 2190, 2191, 2192, 2193, 2194, 2195, 2196, 2197, 2198, 2199, 2200, 2201, 2202, 2203, 2204, 2205, 2206, 2207, 2208, 2209, 2210, 2211, 2212, 2213, 2214, 2215, 2216, 2217, 2218, 2219, 2220, 2221, 2222, 2223, 2224, 2225, 2226, 2227, 2228, 2229, 2230, 2231, 2232, 2233, 2234, 2235, 2236, 2237, 2238, 2239, 2240, 2241, 2242, 2243, 2244, 2245, 2246, 2247, 2248, 2249, 2250, 2251, 2252, 2253, 2254, 2255, 2256, 2257, 2258, 2259, 2260, 2261, 2262, 2263, 2264, 2265, 2266, 2267, 2268, 2269, 2270, 2271, 2272, 2273, 2274, 2275, 2276, 2277, 2278, 2279, 2280, 2281, 2282, 2283, 2284, 2285, 2286, 2287, 2288, 2289, 2290, 2291, 2292, 2293, 2294, 2295, 2296, 2297, 2298, 2299, 2300, 2301, 2302, 2303, 2304, 2305, 2306, 2307, 2308, 2309, 2310, 2311, 2312, 2313, 2314, 2315, 2316, 2317, 2318, 2319, 2320, 2321, 2322, 2323, 2324, 2325, 2326, 2327, 2328, 2329, 2330, 2331, 2332, 2333, 2334, 2335, 2336, 2337, 2338, 2339, 2340, 2341, 2342, 2343, 2344, 2345, 2346, 2347, 2348, 2349, 2350, 2351, 2352, 2353, 2354, 2355, 2356, 2357, 2358, 2359, 2360, 2361, 2362, 2363, 2364, 2365, 2366, 2367, 2368, 2369, 2370, 2371, 2372, 2373, 2374, 2375, 2376, 2377, 2378, 2379, 2380, 2381, 2382, 2383, 2384, 2385, 2386, 2387, 2388, 2389, 2390, 2391, 2392, 2393, 2394, 2395, 2396, 2397, 2398, 2399, 2400, 2401, 2402, 2403, 2404, 2405, 2406, 2407, 2408, 2409, 2410, 2411, 2412, 2413, 2414, 2415, 2416, 2417, 2418, 2419, 2420, 2421, 2422, 2423, 2424, 2425, 2426, 2427, 2428, 2429, 2430, 2431, 2432, 2433, 2434, 2435, 2436, 2437, 2438, 2439, 2440, 2441, 2442, 2443, 2444, 2445, 2446, 2447, 2448, 2449, 2450, 2451, 2452, 2453, 2454, 2455, 2456, 2457, 2458, 2459, 2460, 2461, 2462, 2463, 2464, 2465, 2466, 2467, 2468, 2469, 2470, 2471, 2472, 2473, 2474, 2475, 2476, 2477, 2478, 2479, 2480, 2481, 2482, 2483, 2484, 2485, 2486, 2487, 2488, 2489, 2490, 2491, 2492, 2493, 2494, 2495, 2496, 2497, 2498, 2499, 2500, 2501, 2502, 2503, 2504, 2505, 2506, 2507, 2508, 2509, 2510, 2511, 2512, 2513, 2514, 2515, 2516, 2517, 2518, 2519, 2520, 2521, 2522, 2523, 2524, 2525, 2526, 2527, 2528, 2529, 2530, 2531, 2532, 2533, 2534, 2535, 2536, 2537, 2538, 2539, 2540, 2541, 2542, 2543, 2544, 2545, 2546, 2547, 2548, 2549, 2550, 2551, 2552, 2553, 2554, 2555, 2556, 2557, 2558, 2559, 2560, 2561, 2562, 2563, 2564, 2565, 2566, 2567, 2568, 2569, 2570, 2571, 2572, 2573, 2574, 2575, 2576, 2577, 2578, 2579, 2580, 2581, 2582, 2583, 2584, 2585, 2586, 2587, 2588, 2589, 2590, 2591, 2592, 2593, 2594, 2595, 2596, 2597, 2598, 2599, 2600, 2601, 2602, 2603, 2604, 2605, 2606, 2607, 2608, 2609, 2610, 2611, 2612, 2613, 2614, 2615, 2616, 2617, 2618, 2619, 2620, 2621, 2622, 2623, 2624, 2625, 2626, 2627, 2628, 2629, 2630, 2631, 2632, 2633, 2634, 2635, 2636, 2637, 2638, 2639, 2640, 2641, 2642, 2643, 2644, 2645, 2646, 2647, 2648, 2649, 2650, 2651, 2652, 2653, 2654, 2655, 2656, 2657, 2658, 2659, 2660, 2661, 2662, 2663, 2664, 2665, 2666, 2667, 2668, 2669, 2670, 2671, 2672, 2673, 2674, 2675, 2676, 2677, 2678, 26

В DevOps інколи окремо виділяють таку операцію, як **безперервне розгортання (Continuous Deployment, CDP)** – постачання результатів роботи практик CI/CD у промислове середовище.



3. Jenkins

Jenkins – це кросплатформна програмна система з відкритим початковим кодом, написаним мовою програмування Java, яка призначена для забезпечення процесу CI/CD.



<https://www.jenkins.io/>

3. Jenkins

Важливою особливістю Jenkins є наявність великої кількості різноманітних плагінів, що розширюють його функціональність.

Getting Started

Getting Started

✓ Folders	✓ OWASP Markup Formatter	✓ Build Timeout	✓ Credentials Binding
✓ Timestampers	✓ Workspace Cleanup	✓ Ant	⌚ Gradle
⌚ Pipeline	⌚ GitHub Branch Source	⌚ Pipeline: GitHub Groovy Libraries	⌚ Pipeline: Stage View
⌚ Git	⌚ SSH Build Agents	⌚ Matrix Authorization Strategy	⌚ PAM Authentication
⌚ LDAP	⌚ Email Extension	✓ Mailer	⌚ Dark Theme

Jenkins 2.440.2

⌚ JavaBeans Activation Framework (JAF) API

⌚ JAXB

⌚ SnakeYAML API

⌚ JSON API

⌚ Jackson 2 API

⌚ commons-text API

⌚ Pipeline: Supporting APIs

⌚ Plugin Utilities API

⌚ Font Awesome API

⌚ Bootstrap 5 API

⌚ JQuery3 API

⌚ ECharts API

⌚ Display URL API

⌚ Checks API

⌚ JUnit

⌚ Matrix Project

⌚ Resource Disposer

Workspace Cleanup

Ant

⌚ JavaMail API

⌚ Durable Task

⌚ Pipeline: Nodes and Processes

⌚ bouncycastle API

⌚ Instance Identity

⌚ Pipeline: SCM Step

⌚ Pipeline: Groovy

⌚ Pipeline: Job

⌚ Jakarta Activation API

⌚ Jakarta Mail API

⌚ Apache HttpComponents Client 4.x API

Mailer

⌚ Pipeline: Basic Steps

Gradle

⌚ - required dependency

3. Jenkins

The screenshot shows the Jenkins dashboard. At the top is a black header with the Jenkins logo, a search bar labeled 'Пошук (CTRL+K)', and several icons including a shield with a red '1', a user profile, and a share icon. Below the header is a light gray bar with 'Dashboard >'. The main content area has a left sidebar with links: '+ Новий Item', 'Люди' (with a group icon), 'Історія побудов' (with a folder icon), 'Manage Jenkins' (with a gear icon), and 'My Views' (with a checkbox icon). The sidebar also contains two expandable sections: 'Черга побудов' (Queue) showing 'Немає збірок у черзі' and 'Статус виконавців побудов' (Executors) showing '1 Вільно' and '2 Вільно'. The main area features a large heading 'Ласкаво просимо у Дженкінс!' followed by a paragraph: 'This page is where your Jenkins jobs will be displayed. To get started, you can set up distributed builds or start building a software project.' Below this is a section 'Start building your software project' with a 'Create a job' button. Another section 'Set up a distributed build' contains three buttons: 'Set up an agent' (with a computer icon), 'Configure a cloud' (with a cloud icon), and 'Learn more about distributed builds' (with a question mark icon). A link 'додати опис' is visible in the top right of the main area.

Jenkins

Пошук (CTRL+K)

Dashboard >

+ Новий Item

Люди

Історія побудов

Manage Jenkins

My Views

Черга побудов

Немає збірок у черзі

Статус виконавців побудов

1 Вільно

2 Вільно

Ласкаво просимо у Дженкінс!

This page is where your Jenkins jobs will be displayed. To get started, you can set up distributed builds or start building a software project.

Start building your software project

Create a job

Set up a distributed build

Set up an agent

Configure a cloud

Learn more about distributed builds

додати опис

3. Jenkins



+ Новий Item

Люди

Історія побудов

Manage Jenkins

My Views

Enter an item name

wfem

» Required field

Freestyle project
Classic, general-purpose job type that checks out from up to one SCM, executes build steps serially, followed by post-build steps like archiving artifacts and sending email notifications.

Pipeline
Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.

Створення нового проекту

3. Jenkins

The screenshot shows the Jenkins web interface. At the top, there's a navigation bar with the Jenkins logo, a search bar, and user information. Below it, a breadcrumb trail shows 'Dashboard > wfem > Configuration'. The main area is titled 'Configure' and has a sidebar with various configuration options: General (selected), 'Управління вихідним кодом' (Source Management), Build Triggers, Build Environment, Build Steps, and Post-build Actions. The 'General' tab is active, showing a 'General' section with an 'Enabled' toggle. Below this is a 'Plain text' field with a 'Попередній перегляд' (Preview) link. There are several checkboxes for build settings: 'GitHub project', 'Throttle builds', 'Скасувати старі білди', 'Ця будова має параметри', and 'Виконати паралельні збірки в разі необхідності'. At the bottom, there's a 'Додаткові' (Additional) dropdown menu. The 'Управління вихідним кодом' section is partially visible at the bottom, showing 'None' selected.

Jenkins

Пошук (CTRL+K)

Serhii Homeniuk

Вийти

Dashboard > wfem > Configuration

Configure

General

Enabled

Опис

Plain text [Попередній перегляд](#)

☐ GitHub project

☐ Throttle builds

☐ Скасувати старі білди

☐ Ця будова має параметри

☐ Виконати паралельні збірки в разі необхідності

Додаткові

Управління вихідним кодом

☒ None

☐ Git

Зберегти Apply

Управління вихідним кодом

☐ None

☒ Git

Repositories

Repository URL

<https://github.com/SeregaGomen/wfem.git>

Credentials

- none -

+ Add

Додаткові

Add Repository

Branches to build

Branch Specifier (blank for 'any')

*/master

Add Branch

3. Jenkins

Build Steps

Add build step ^

Filter

Execute Windows batch command

Execute shell

Invoke Ant

Invoke Gradle script

Invoke top-level Maven targets

Run with timeout

Set build status to "pending" on GitHub commit

Build Steps

Execute Windows batch command ?

Command

See [the list of available environment variables](#)

go build

Додаткові ▾

Add build step ▾

Post-build Actions

Add post-build action ▾

Зберегти

Apply

3. Jenkins

Dashboard >

+ Новий Item

👤 Люди

📁 Історія побудов

⚙️ Manage Jenkins

📌 My Views

All +

S	W	Name ↓	Востаннє успішно	Востаннє невдало	Тривалість останньої побудови
⋮	☀️	wfem	N/A	N/A	Н/Д

✎ додати опис

Інформація про наявні проєкти в Jenkins

3. Jenkins

The screenshot displays the Jenkins web interface. At the top, there's a navigation bar with the Jenkins logo, a search bar labeled 'Пошук (CTRL+K)', and user information for 'Serhii Homeniuk'. Below this, a breadcrumb trail shows 'Dashboard > wfem >'. The main content area is divided into a left sidebar and a main panel. The sidebar contains links for 'Статус', 'Зміни', 'Робочий Простір', 'Build Now', 'Налаштування', 'Видалити Project', and 'Rename'. The main panel shows the 'wfem' project status with a green checkmark icon. Below this, there's a section titled 'Постійні посилання' (Permanent links) with a list of build links: 'Last build (#1), 5 min 1 sec тому', 'Last stable build (#1), 5 min 1 sec тому', 'Last successful build (#1), 5 min 1 sec тому', and 'Last completed build (#1), 5 min 1 sec тому'. At the bottom left, there's a 'Build History' widget showing a table with one entry: '#1' with a timestamp '3 квіт. 2024 р., 16:16'. The widget also includes a filter bar and atom feed links for all and failed builds.

Jenkins

Пошук (CTRL+K)

Serhii Homeniuk

Вийти

Dashboard > wfem >

Статус

Зміни

Робочий Простір

Build Now

Налаштування

Видалити Project

Rename

✓ wfem

додати опис

Вимкнути проект

Постійні посилання

- [Last build \(#1\), 5 min 1 sec тому](#)
- [Last stable build \(#1\), 5 min 1 sec тому](#)
- [Last successful build \(#1\), 5 min 1 sec тому](#)
- [Last completed build \(#1\), 5 min 1 sec тому](#)

Build History Тренд

Filter...

✓ #1	3 квіт. 2024 р., 16:16
------	------------------------

[Atom feed для всіх](#) [Atom feed для невдач](#)

Вікно параметрів Item

3. Jenkins

 Робочий Простір

 Build Now

 Налаштування

 Видалити Project

 Rename

 Build History Тренд ▾

 #1 3 квіт. 2024 р., 16:16

 Atom feed для всіх  Atom feed для невдач

 Jenkins Пошук (CTR)

Dashboard > wfem > #1 > Консольний Вивід

 Стан

 Зміни

 Console Output

 View as plain text

 Редагувати опис

 Delete build '#1'

 Git Build Data

 Консольний Вивід

```
Started by user Serhii Homeniuk
Running as SYSTEM
Building in workspace C:\ProgramData\Jenkins\jenkins\workspace\wfem
The recommended git tool is: NONE
No credentials specified
Cloning the remote Git repository
Cloning repository https://github.com/SeregaGomen/wfem.git
> git.exe init C:\ProgramData\Jenkins\jenkins\workspace\wfem # timeout=10
Fetching upstream changes from https://github.com/SeregaGomen/wfem.git
> git.exe --version # timeout=10
> git --version # 'git version 2.39.1.windows.1'
> git.exe fetch --tags --force --progress -- https://github.com/SeregaGomen/wfem.git +refs/heads/*:refs/remotes/origin/* # timeout=10
> git.exe config remote.origin.url https://github.com/SeregaGomen/wfem.git # timeout=10
> git.exe config --add remote.origin.fetch +refs/heads/*:refs/remotes/origin/* # timeout=10
Avoid second fetch
> git.exe rev-parse "refs/remotes/origin/main^{commit}" # timeout=10
Checking out Revision 1cc5a653519e188da5a5d0056e73d07377a6a74d (refs/remotes/origin/main)
> git.exe config core.sparsecheckout # timeout=10
> git.exe checkout -f 1cc5a653519e188da5a5d0056e73d07377a6a74d # timeout=10
Commit message: "Last"
First time build. Skipping changelog.
[wfem] $ cmd /c call C:\Windows\TEMP\jenkins17559602907553256636.bat

C:\ProgramData\Jenkins\jenkins\workspace\wfem>go build
go: downloading gonum.org/v1/gonum v0.13.0
go: downloading golang.org/x/exp v0.0.0-20230425010034-47ecfdc1ba53

C:\ProgramData\Jenkins\jenkins\workspace\wfem>exit 0
Finished: SUCCESS
```

Побудова проєкту, історія збірок та детальна інформація про результати збирання

3. Jenkins

```
pipeline {
  agent any

  triggers {
    cron('H/30 * * * *')
  }

  stages {
    stage('Checkout') {
      steps {
        git 'https://gitlab.com/gserega71/pyfem.git'
      }
    }
    stage('Install Dependencies') {
      steps {
        bat "'D:/Python/Python312/Scripts/pip' install numpy scipy simplejson pyqt5 pyopengl'"
      }
    }
    stage('Run Program') {
      steps {
        bat "'D:/Python/Python312/python' fem_test.py"
      }
    }
  }
}
```