

0. Розгортання Друпала - базові дії

- розгорнути та встановити Drupal 7 останньої версії
- у встановленому Друпалі не повинно бути файлів install.php, *.txt (крім robots.txt)
- покласти під ГІТ з налаштованими вітками (детально див. пункт 1).
- налаштувати файл robots.txt (детально див. пункт 2).
- логін/пароль адміністратора сайту (він же супер-адмін, він же юзер з id = 1) повинен бути такий же як і логін/пароль до бази даних сайту. У випадку зміни логіна/пароля до бази даних- логін/пароль супер-адміна має змінитись відповідно. Логін/пароль супер-адміна можна знайти в вкладці "Info" проекту, якщо його там немає, потрібно повідомити дев-ліда/менеджера про це.

Пункт від 30 липня 2013.

- Інсталювати базові модулі (детально див. пункт 3).
- Вимкнути не потрібні модулі.
- Налаштувати файлову систему (детально див. пункт 11.2).
- Перевірити статус репорт на наявність проблем.
- за попередньою домовленістю із замовником і при необхідності сайт можна покласти під .htpasswd.

1. GIT

1.1 Кожен сайт повинен бути під ГІТом;

1.2 Усе на сайті повинно лежати під ГІТом (окрім відведеної файлової теки);

1.3 Усі зміни на живому та/або дев сайті повинні бути закомічені;

1.4 У ГІТ репозиторії гілка *master* повинна містити файл README з назвою проекту;

1.5 Також повинна бути гілка з назвою *dev* - тут повинен бути власне сайт;

1.6 Після виконання задачі, якщо є необхідність, завжди комітити та пушити зміни;

1.7 У кінці робочого дня на проекті у статусі ГІТа не повинно бути не закомічених змін.

Якщо зміни у файлах які були не закомічені магічним чином пропадуть на наступний день - ваші проблеми.

1.8 Писати адекватні та розгорнуті коментарі до комітів.

1.9 Якщо в якогось користувача немає доступу до репозиторію (а він має бути), то Development Lead має скинути bohart'у в скайп у приватну гілку переписки:

- нік користувача;

- посилання на репозиторій в лабі.

1.10 Відповідальний за те щоб адекватно ввівся версійний контроль - Development Lead.

1.11 Якщо потрібно видати доступ до нашого гіт репозиторію замовнику, то DevLead має:

а) отримати апрув від менеджера про перманентний доступ замовника до актуальної версії нашого коду;

б) змиритися з думкою що наш гавнокод буде бачити замовник;

в) скинути bohart'у в скайп у приватну гілку переписки:

- емейл замовника (важливо: цей емейл має відповідати емейлу, що буде вказаний у параметрі "автор" коміта);

- публічний ключ замовника;

- посилання на репозиторій в лабі.

1.12 Вітки ніколи не видаляються (право на видалення може надати тільки ДЛ проекту у виключно індивідуальних нюансах).

Перенесено з "Положення для Development Lead" 2 вересня 2013 року

Перенесено з "Робота з ГІТом на проектах" 2 вересня 2013 року

1.13 Ядро Drupal потрібно встановлювати в якості сабмодуля GIT-а. Рекомендації по встановленню можна знайти в блозі [Drupal Core Installation As A Submodule](#),

репозиторій з Drupal'ом знаходиться тут - [Drupal core](#)

1.14 Опис комітів (Commit messages)

1.14.1 ВАЖЛИВО: не починайте опис коміту зі знаку #. Це може призвести до значних незручностей при роботі з вашим комітом, в першу чергу при використанні rebase -і.

1.14.2 Опис коміту пишеться виключно англійською мовою.

1.14.2 Повідомлення коміту повинно виглядати наступним чином: Task #[номер задачі]: [короткий опис змін].

1.14.3 Опис коміту повинен починатись з дієслова в минулому часі, відповідно до типу задачі і характеру змін:

- **Added** (новий функціонал, New Feature)
- **Fixed** (виправлено помилку, Bug)
- **Changed** (зміна існуючого функціоналу, Task)
- **Updated** (зміна коду для відповідності змінам сторонньої бібліотеки чи АПІ, Task)

Загальний по роботі “dev-live site”

Передбачені такі основні branch-і:

- live - гілка живого сайту;
- dev - гілка дев сайту;
- одна гілка для кожного тікета.

Початок роботи з задачею:

git checkout live - переключитися на лайв бранч.

git checkout -b [ticket:id] - створити та одразу переключитися у новий бранч задачі.

Примітки:

- Назва Бранчу у ГІТі прямолінійно залежить від номеру задачі, для прикладу, якщо номер задачі 150, тоді бранч потрібно називати наступним чином: "150" (без лапок).

Кінець роботи із задачею:

git checkout [ticket:id] - переключитися на бранч, який ви завершили.

git push origin [ticket:id] - запустити бранч з задачею (звісно все повинно бути закомічено).

git pull origin live - підтягнути усі зміни на лайф.

git merge live - змерджити усі зміни з лайфу у бранч тікета.

git checkout dev - переключитися на дев гілку.

git pull origin dev - запулити дев гілку.

git merge [ticket:id] - змерджити зміни гілки тікету у дев бранч.

git push origin dev - запустити дев бранч.

Примітки:

- Не забувати про оновлення сабмодулів (*git submodule init* && *git submodule update*).
- Перед пушем у дев бранч варто перевірити працездатність сайту.

ВАЖЛИВО:

- Ніхто не комітить у dev бранч, туди тільки мерджити тікети.
- Ніхто не комітить і не мерджить у live бранч (під виключення підпадають непередбачуванні обставини відповідальність за які несе ДЛ проекту).
- Після переносу (зміна лайв гілки) потрібно змерджити робочі бранчі з актуалізованою (!) live гілкою.

- Не варто робити 2 коміти у 1 бранч. Виключенням може послужити той випадок, якщо ви задачу виконуєте на протязі 2 днів або більше. Але намагатися робити тільки 1 коміт (це слугує для надійного мерджу тікетів).

Будь-які правила, по роботі “dev-live site”, можуть бути зміненими, але не виключеними ДевЛідом проекту.

^ Перенесено з “Робота з ГІТом на проектах” 2 вересня 2013 року

2. robots.txt

2.1 Сайт не повинен індексуватися пошуковими системами, якщо він знаходиться на дев сайті (на drupalcompu.com все вже налаштовано, тому змінювати файл не потрібно, лише перевірити);

2.2 Файл повинен знаходитися під гітігнором.

2.3 По замовчуванню живий сайт повинен бути відкритий для індексування пошуковими системами.

Перенесено з “Положення для Development Lead” 2 вересня 2013 року

3. Інсталяція модулів

3.1 усі модулі повинні бути встановленні у якості ГІТ сабмодулів;

3.2 якщо необхідний вам модуль потрібно ставити не у якості сабмодуля (наприклад nodejs), встановлювати такий модуль дозволяється лише з дозволу старшого програміста проекту;

3.3 для базової збірки обов'язкова інсталяція та налаштування наступних модулів/плагінів:

- GlobalRedirect
- Administration menu
- Views
- Pathauto
- GoogleAnalytics
- XML Sitemap
- Elysia Cron
- Metatags
- WYSIWYG
- IMCEm2
- редактор TinyMCE (по замовчуванню, якщо потрібен інший текстовий редактор, ставимо інший)
- WYSIWYG filter
- Transliteration
- Subpathauto

3.4 Не варто ставити усі модулі, що є на Д.орг. Якщо вам необхідно до сайту задіяти незначний функціонал, і є можливість написати цей функціонал у власному модулі - зверніться за рекомендацією до старшого програміста проекту.

3.5 Усі модулі, які ви не знаєте дозволено встановлювати лише після дозволу старшого програміста проекту.

4. Тема та Темізація

4.1 Тема, яку необхідно встановлювати повинна узгоджуватися з старшим програмістом та з верстальщиком, або менеджером проекту. По замовчуванню ставити тему Ninesixty;

4.2 Усі тпл-файли (*.tpl.php) повинні знаходитися у папці templates;

4.3 Усі тпл-файли, які знаходяться у папці templates у свою чергу повинні знаходитися у папках по модулях. Наприклад, якщо у нас є views-view.tpl.php, тоді цей файл повинен бути у папці THEME/templates/views/views-view.tpl.php.

5. Процес роботи над сайтом

5.1 ні в якому разі не “хачити” модулі та ядро друпала;

5.2 функціонал повинен нести максимально динамічний характер (тобто максимально уникнути проблем з переписуванням функціоналу, тому що був погано продуманий);

5.3 написаний код повинен бути зрозумілим іншим програмістам (бажано писати коментарі), давати адекватну назву модуля та його опис, уникати створення змінних типу \$query1 \$query2 або \$a \$b \$c;

5.4 Дотримуватися відповідності розміщення коду та цілісності його функціональності. (Наприклад, не допускати наступних речей: коли виключаємо кастомний модуль - сайт не працює, або частина функціоналу, яка відноситься до модуля, ще до сих пір в роботі.);

5.5 Розробку сайту вести на локальному сайті (розгорнутому на вашому комп'ютері, на комп'ютері старшого програміста або програміста);

5.6 На сайті не повинні бути включені модулі, які не беруть участі у роботі сайту;

5.7 У папці з завантаженими файлами (sites/default/files) тримати порядок, не скидувати усі завантажені файли у єдину папку.

5.8 Забороняється писати кирилицю у коді.

5.9 Код має відповідати coding-стандартам (php / js / css / comments).

5.10 Всі резолви в проєкті повинні йти через Development Lead-a.

5.11 За всі задачі в проєкті відповідає Development Lead (верстка, тестування, програмування, адміністрування).

Перенесено з “Положення для Development Lead” 2 вересня 2013 року

6. Правила іменування і розміщення коду

6.1 Custom модулі власного приготування

6.1.1 Основний модуль (або якщо він єдиний) має розміщуватися в директорії: “sites/all/modules/custom/internetdevels.com”. Name в info файлі - “idevels”.

6.1.2 Інші custom-модулі мають називатися по патерну idevels_* і розміщуватися в директорії: “sites/all/modules/custom/%module_name%”.

6.1.4 Вийняток відносно такого наіменування або розміщення модулів можуть бути лише ті випадки де старишй програміст, менеджер або bohart прямим текстом це вказали.

6.2 Contrib модулі (модулі з Д.орг) мають знаходитися в директорії:

“sites/all/modules/contrib/%module_name%”.

6.3 Модулі типу “feature” мають знаходитися в директорії:

“sites/all/modules/features/%feature_name%”.

6.4 Пропатчені контріб модулі повинні лежати в папці

sites/all/modules/modified/%module_name% разом з патчем. Пропатчений модуль видаляється з сабмодулів і вставляється як кастомний, але в вищенаведену папку.

Пункт від 16 липня 2013 року

6.5 Основна тема має розміщуватися в директорії “sites/all/themes” і мати однозначне наіменування з назвою проєкта.

7. Що робити, якщо сайт не адекватно працює, або взагалі не працює

7.1 при виникненні проблем з сайтом потрібно згадати, після чого ці проблеми почалися, спробувати повернути ті зміни на сайті, які були застосовані перед виникненням проблеми.

7.2 запам'ятайте, що усі роботи щодо відновлення роботи сайту потрібно проводити на локальному сайті.

7.3 якщо Ви знайшли баг у модулі/ядрі потрібно:

1) сокристуватися гуглом, пошукати патч, або рішення проблеми для даної версії модуля. Якщо знайдений необхідний патч, тоді читати пункт **5)**, якщо ні, читаємо наступний пункт.

2) спробувати встановити іншу версію модуля (новішу або старішу), якщо не допомогло, читаємо наступний пункт.

3) розібратися максимально детально у чому проблема, подебажити (можна використати xdebug-ом), переконавшись чи не створює цю помилку версія ядра сайту. Якщо версія ядра сайту застаріла, спробувати встановити новішу версію. Якщо не допомогло, читаємо наступний пункт.

4) якщо ви впевнені, у тому, що можна самому написати патч, пишемо, та тестуємо. Якщо патч працює, тоді читати пункт

5), інкаше звернутися за допомогою старшого програміста проекту, або до колег по роботі, які мали справу з подібними речами.

5) перед встановленням патчу обов'язково потрібно повідомити про це старшого програміста та менеджера проекту. Встановлювати патч тільки після апруву та попередніх рекомендацій (якщо є) цих осіб.

7.4 Не допускати безрезультатне “заліпання” над проблемою! Користуйтеся допомогою старшого програміста, або колег по роботі.

7.5 Якщо причина помилки виникла не по нашій вині, і ви її виправили, обов'язково про це необхідно повідомити старшого програміста та менеджера проекту.

8. Доступи.

8.1 Ні в якому разі без дозволу старшого програміста та менеджера не надавати доступи замовнику до глобального користувача(uid == 1).

8.2 Як зазвичай замовнику створюється роль редактора, де він зможе заповнювати сайт та/або керуватися його функціональними частинами. Усі доступи потрібно узгоджувати власне з страшим програмістом / менеджером / замовником.

8.3 Не надавати замовнику зайвих доступів до функціональних частин сайту.

8.4 Якщо вам необхідно дати один з доступів і даний доступ дозволяє ще керуватися іншим функціоналом сайту, до якого замовник не повинен мати доступ потрібно:

а) Попередити замовника про це, роказати про можливі наслідки неусвідомленого користування тих функціональних частин сайту, які Друпал фізично не може обмежити.

б) Запропонувати змовнику зробити унікальні налаштування доступів до функціональних частин сайту (“захист від дурня”).

8.5 Не надавати замовнику доступи до *Full html* без попереднього узгодження цієї дії з старшим програмістом та менеджером проекту. Без узгодження текст обов'язково повинен проходити фільтрацію.

9. Живий сайт

9.1 На живому сайті має бути присутнім і налаштований модуль Google Analytics.

9.2 По мірі можливості має бути налаштований модуль boost або налаштоване

кешування у Друпалі, увімкнуте об'єднання CSS та JS.

9.3 В залежності від типу проекту(за виключенням NDA) повинен бути доданий наш копірайт згідно правил [СЕО-методички](#) пункт "Авторская ссылка". Якщо ж виникли сумніви, або необхідні уточнення звертатися до id.seoshkin.

Пункт від 24 березня 2014 року.

10. Development Lead:

10.1 Відповідає за ВСІ питання технічного і архітектурного характеру.

10.2 Відповідає за наявність і адекватне налаштування всіх обов'язкових модулів на сайті.

10.3 Відповідає за роботу всіх модулів, що використовуються на сайті. Їх версії, релізи, сумісність з іншими модулями, тощо.

10.4 Здійснює оцінку затратності (aka estimate) нових задач, звертаючись, в разі необхідності, до колег (верстальщиків чи програмістів в залежності від своєї позиції). За адекватність всіх оцінок повністю відповідає Development Lead (навіть якщо їх давав не він).

10.5 Не відповідає за пріоритетність задач, це завдання менеджера проекту.

10.6 Не виставляє ефективні години в проекті - це робить менеджер проекту, який може звернутися за консультацією по цьому питанню.

10.7 В проектах типу Dedicated повинен до 3-го числа кожного місяця надати менеджеру розподіл заздалегіть затвердженого бюджету між учасниками проекту..

Перенесено з "Положення для Development Lead" 2 вересня 2013 року.

11. Додатки:

11.1 Налаштування кнопок для текстового редактору

У налаштуваннях повинні бути включені наступні кнопки:

- Робота із шрифтом: Bold, Italic, Underline, Strike-through.
- Посилання: Link, Unlink.
- Робота з текстом: Align right, Align left, Justify, Align center, Bullet list, Numbered list.
- Додаткові кнопки: Image (з підтримкою IMCE), Cut, Copy, Paste, Paste from Word, Paste text.

Включення інших кнопок, або виключення існуючих, обговорюється із старшим програмістом / менеджером / замовником.

11.2 Налаштування файлової системи

- tmp - повинна знаходитись за шляхом sites/default/files/tmp;
- sites/default/files і всі її внутрішні папки повинні мати права 777;
- imce профілі - профіль звичайного користувача повинен мати власну папку, вийте за межі якої він не повинен мати можливості.

11.3 sites/default/files/

- Вивантаження файлів необхідно налаштовувати по папкам. Для прикладу: і нас є поле field_image і тип контенту acticle, відповідно усі файли, які завантажуються до матеріалів цього типу контенту потрібно зберігати у окремій папці.
- Назва папки повинна машину назву типу контенту і назву поля. Якщо файлів у данному полі буде дуже багато, тоді розділяти папку ще на підпапки, наприклад по даті.
- Ім'я файлу по можливості робити у форматі: [file-fid].[original-extension].

11.4 Аліаси

- На кожному сайті, по замовчуванню повинен бути налаштований модуль Pathauto.

- Аліаси задавати відповідно до розділів. Наприклад, якщо у нас є список новин за посиланням news, тоді кожна новина повинна мати аліас news/test-news.
- Аліаси повинні бути транслітеровані.

11.5 Тип контенту

- Машина назва типу контенту у Друпал повинна бути описана одним словом.
- Одні філди можна/треба юзати в різних типах контенту (При можливості потрібно юзати один філд в багатьох типах контенту, щоб уникати ситуацій, коли на сайті є філди field_image, field_news_image, field_blog_image з однаковими налаштуваннями.).

Пункт від 24 липня 2013 року.

12. Контент:

12.1 Тестовий контент(зображення, відео, текст) повинен максимально відповідати тематиці сайту та бути адекватним.

12.2 В якості тестового тексту можна використовувати [Lorem ipsum](#).

12.3 Забороняється використовувати політичну чи релігійну тематику.