Краткое содержание

4.1 Создание проекта

Запишем основные действия пользователя будущего приложения:

- Пользователь заходит на главную страницу впервые, программа проверит:
  1. если приложение не установленно, то отправит пользователя в панель установки приложеня.
  2. если приложение установленно, то приложение проверит регистрацию пользователя.
  3. если пользователь не зарегистрирован, перенаправит на регистрацию.
  4. если пользователь зарегистрирован, перенаправит на авторизацию.
  5. если авторизация не прошла успешно, заблокирует на время этого пользователя.
  6. если авторизация прошла успешно, откроет панель администратора.
  7. если открыта панель администратора, но не установлена конфигурация (видеохостинг, блог или социальная сеть), то приложение придложит сделать выбор и установить.
  8. если открыта панель администратора и установлена конфигурация (видеохостинг), то приложение придложит сделать выбор между закрытой (вход по логину и паролю) или открытой (общедоступной).
     1. если открытая (общедоступная), то можно ли регистрироваться и т.д.. (Видеохостинг описан в руководстве по написанию UCMS Video, что является продолжение данного руководства). А в этом, мы остановимся на выборе конфигурации, окне настройки, установке дополнительных модулей (загрузка файлов, почта, редактор текста, двуфакторная авторизация и т.д.), добавление шаблонов, языков и прочее, а пока...

В наш маленький проект внесем дополнения:

- Пользователь заходит на главную страницу **index.php** впервые, программа проверит:
  1. если приложение не установленно (файлы **config.ini**, **version.txt** не существуют), то отправит пользователя в панель установки приложеня (файл **install.php**) в директорию **/install**.
  2. панель установки приложеня (файл **install.php**) предложит выбрать язык (английский/русский)
  3. по умолчанию предложен английский (файл **lang_en.php**) в дирректории **lang** установщика
  4. при выборе русский (файл **lang_ru.php**) в дирректории **lang** файл установщика **install.php** переключит установщик на русский язык и будет ожидать выбора часового пояса (форма выбора **timezone** -12, 0, +12)
  5. после выбора станет активной кнопка «Продолжить», при нажатии на которую, значения переменных **lang_select** и **timezone_select** будут записаны в файл конфигурации **config.ini**
  6. пользователь будет перенаправлен к окну проверки совместимости сервера (файл **check_server.php**)

Пока остановимся. Откуда названия файлов и директорий (папок)? Назначаем имена файлам сами, чтобы нам проще было понять, например:

- **config** - в переводе с английского **конфигурация**
- **install** - **установка**
- **lang** - **язык**
- **check_server** - **проверить сервер**
- **timezone** - **часовой пояс** и т.д.

Таким образом, мы всегда будем понимать, что директория **/install** - это папка установки, а переменная **lang_select** - выбор языка, стало быть **lang_select_russian** - выбор языка русский.

Подумаем еще немного… А где будет распологаться наше приложение? В корне сайте **" / "** или в директории **«/ucms»**? Мы не знаем что хочет пользователь, т.е. создаем приложение в директории **/ucms** без привязки к директории, а за это будет отвечать файл, где будут постоянные данные - константы (стало быть **cons.php**). Т.к. **cons.php** и **config.ini** являются конфигурационными, то целесообразно использовать отдельную директорию **/config**

Продолжим наш план-проект…

- Пользователь заходит в директорию **/ucms** на главную страницу файл **index.php** впервые, программа проверит:
    1. если приложение не установленно (файлы **config.ini**, **version.txt** не существуют), то перенаправит пользователя в панель установки приложеня (файл **install.php**) в директорию **/install**.
    2. панель установки приложения (файл **install.php**) так-же проверит наличие файлов **config.ini** и **version.txt**
    3. если файл **config.ini** не существует, то перенаправит к файлу выбора языка часовой зоны **lang_and_time_zone.php**
    4. окно выбора языка и часовой зоны (файл **lang_and_time_zone.php**) предложит выбрать язык (английский/русский) и часовую зону (от -12 до +12) и в конце попросит подтвердить выбор кнопкой «Продолжить», что создасть файл **config.ini** и запишет в него значения выбора.
    5. по умолчанию предложен английский (файл **lang_en.php**) в дирректории **lang** установщика
    6. при выборе языка - русский (файл **lang_ru.php**) в дирректории **lang** файл установщика (файл выбора языка часовой зоны **lang_and_time_zone.php**) переключит установщик на русский язык и будет ожидать выбора часового пояса (форма выбора **timezone** -12, 0, +12)
    7. после выбора станет активной кнопка «Продолжить», при нажатии на которую, данные **lang_select** и **timezone_select** будут записаны в файл конфигурации **config.ini** в дирректории **/config**
    8. пользователь будет перенаправлен к окну проверки совместимости сервера (файл **check_server.php**), который запросит наличие файла **version.txt**
    9. если **version.txt** не существует (т.е. проверка сервера не проводилась), то приложение перейдет к файлу **check_server.php** и проверит сервер….
    10. если **version.txt** существует (т.е. проверка сервера ранее проводилась), то приложение перейдет к запросу данных для подключения к приложению баз данных (файл **db_config.php**)

По мере написания проекта мы видим, что мы пишем «логику» работы приложения, на «научном» языке это называется - «алгоритм», и мы уже видим за какие действия отвечает тот или иной файл и где находится. Мы можем это отобразить формой записи алгоритма и это

будет более наглядно для понимания.

Карточки

Проще все визиулизировать, написав/нарисовав карточки (соотвествуют отображению на экране), они же файлы и директории и три кружочка: пользователь, зарегистрированный пользователь и администратор. На карточке название: **version.txt**, **db_config.php** и т.д. и в карточках пишем заметки: что делает эта карточка и зарисовки экрана пользователя, а на обратной стороне наброски кода, предварительные таблицы массивов и т.д.

Надеемся, что прочитав данную главу стало понятно, откуда берутся названия файлов, директорий, переменных и т.д. (пример: наименование файла этой страницы creating_project.txt в переводе - 4.1 создание проекта) И главное откуда мы знаем, что будет дальше! Потому что у нас есть - план! При этом мы не будем в этом руководстве писать полный проект, а поверхносто будем коментировать, что и для чего мы создаем и что в итоге получим.

Дополнения и Файлы

см. также:

4.0. Планирование 5.0. Структура директорий

From:
https://wwoss.ru/ - **worldwide open-source software**

Permanent link:
https://wwoss.ru/doku.php?id=software:development:demo:cms:ucms:creating_project

Last update: **2026/02/08 10:21**