В общих чертах, веб-приложение, построенное на Yii Framework 2, состоит из двух основных компонентов: фронтенда и бэкенда.

Фронтенд (папка "views") представляет собой часть приложения, которую видит пользователь, то есть то, что отображается в веб-браузере. Он содержит файлы представлений, которые определяют, как данные должны быть отображены на страницах приложения. Файлы представлений содержат php-код, который определяет внешний вид и поведение страниц приложения.

Бэкенд (папки "controllers", "models" и "config") отвечает за обработку запросов пользователя и управление данными. Он использует концепцию модель-представление-контроллер (MVC), которая разделяет приложение на три части: модели, представления и контроллеры.

Модели (папка "models") представляют собой объекты, которые представляют данные в приложении. Они содержат логику доступа к данным и методы для взаимодействия с базой данных.

В файле "models/User.php" вы найдете модель пользователя, которая определяет структуру таблицы базы данных для хранения информации о пользователях.

Сюда небольшой туториал по подключению моделей к веб-приложению. Можно картинку и описание.

Контроллеры (папка "controllers") обрабатывают запросы пользователя, обрабатывая данные и передавая их в модели и представления. Они являются посредниками между пользователем и моделями приложения.

Файл "web/index.php" является точкой входа в приложение. Он загружает настройки и запускает приложение, обрабатывая запросы пользователя.

В папке "config" вы найдете файлы настроек приложения. Файлы содержит основные настройки приложения, такие как настройки базы данных, URL-адреса приложения и другие параметры.

Объяснить какие файлы

Желательно часть кода с комментарием.

Файлы представлений (папка "views") определяют, как данные должны быть отображены на страницах приложения. Они содержат php-код, который определяет внешний вид и поведение страниц приложения.

Сюда можно скрин интерфейса

Контроллер "SiteController.php" обрабатывает запросы пользователя и передает данные в представления. В частности, метод "actionIndex()" этого контроллера отображает домашнюю страницу приложения, а метод "actionContact()" отображает страницу контактов.

Методы в контроллерах представляют собой действия, которые выполняются при запросе определенного URL-адреса пользователем. Например, если пользователь заходит на страницу example.com/site/index, то будет выполнен метод actionIndex() в контроллере SiteController. Этот метод возвращает представление, которое отображает содержимое главной страницы сайта.

Методы в моделях представляют собой функции, которые выполняют операции с данными из соответствующих таблиц базы данных. Например, метод find() в модели User возвращает объект пользователя с определенным идентификатором, который извлекается из базы данных.

В файле web/index.php находится точка входа в приложение. Этот файл загружает все необходимые компоненты фреймворка и инициализирует приложение. После этого он обрабатывает входящий запрос от пользователя и вызывает соответствующий метод контроллера.

В файле config/web.php находятся настройки приложения для окружения веб-сервера. Этот файл определяет параметры, такие как база данных, настройки URL и другие важные параметры.

В файле config/console.php находятся настройки приложения для окружения командной строки. Этот файл определяет параметры, такие как база данных, настройки консоли и другие важные параметры.

В файле model.mwb находится диаграмма базы данных, которая была создана с помощью MySQL Workbench. Эта диаграмма содержит схему базы данных, которая определяет структуру таблиц, связи между ними и другие важные параметры.

В целом, Yii Framework 2 предоставляет мощный инструментарий для создания веб-приложений, который позволяет ускорить разработку и сократить количество кода, необходимого для создания приложения. Однако, для полного понимания работы фреймворка, необходимо изучить его документацию и освоить несколько базовых концепций, таких как MVC-архитектура, компоненты и конфигурация.

Если кратко, сейчас глава не имеет чёткой связи между своими частями, необходимо их связать. Пример: к беку относятся такие файлы, в них это (скрин), тут было настроено что-то ... Также на ваш вкус можете вставить часть кода в главу, это нужно по двум причинам (увеличит объём главы, поможет преподавателю в проверке работы). Сама глава очень хороша, однако это пока не подходит под категорию главы в курсовой работе, скорее похоже на пояснительную записку.