**1. Анализ и постановка задачи**

* Определить, что за приложение и для кого оно.
* Сформулировать основные функции приложения.(подробно прямо что должно отображатся, что должно делать,)
* Определить роли пользователей (гость, пользователь, администратор).  
  **Результат:** текстовое ТЗ (техническое задание).

**2. Дизайн интерфейса (UI/UX)**

* Нарисовать макет интерфейса (можно от руки).
* Определить список страниц (главная, регистрация, вход, список записей и т.д.).
* Решить, где будут формы, кнопки, меню.
* Сделать простые карточки страниц (wireframes).  
  **Результат:** карта страниц + эскизы интерфейса.

**3. Проектирование базы данных**

* Определить, какие данные нужно хранить.
* Выделить сущности и связи между ними.
* Нарисовать схему таблиц.  
  **Пример:**
  + User: id, login, password
  + Message: id, user\_id, text, date  
    **Результат:** схема БД (ER-диаграмма или таблицы).

**4. Архитектура приложения**

* Решить, как связаны страницы и где будет PHP-логика.
* Определить структуру файлов проекта.  
  **Пример:**
  + index.php -главная
  + login.php -вход
  + register.php -регистрация
  + db.php -подключение к БД
  + style.css -оформление  
    **Результат:** структура проекта (карта файлов).

**5. Верстка (HTML + CSS)**

* Создать каркас страниц на HTML.
* Добавить оформление с помощью CSS.
* Сначала сделать статические страницы без PHP.  
  **Результат:** статический дизайн сайта.

**6. Бэкэнд (PHP + БД)**

* Подключить PHP к базе данных.
* Реализовать работу форм (регистрация, вход, добавление записей).
* Организовать вывод данных из БД в HTML.  
  **Результат:** динамический сайт.

**7. Тестирование**

* Проверить работу форм.
* Проверить корректность отображения данных.
* Исправить ошибки и недочёты.  
  **Результат:** список багов и исправлений.

**8. Презентация / сдача проекта**

* Показать интерфейс приложения.
* Продемонстрировать работу функций.
* Объяснить структуру базы данных.
* Объяснить написанный код.