**1. Обоснование выбора технологий**

* **HTML, CSS, JavaScript**  
  Эти стандарты поддерживаются всеми браузерами и не требуют установки дополнительных плагинов. Благодаря им можно обеспечить работу сайта на любых устройствах и легко подключать интерактивные элементы (таймер, динамическая подгрузка вопросов).
* **WordPress**  
  Самая популярная система управления контентом (CMS) с огромным сообществом и множеством готовых расширений (плагинов). Это сокращает время разработки, позволяет без дополнительного кода настроить разные типы контента (тесты, вопросы, материалы) и распределить права доступа между ролями.
* **PHP и MySQL**  
  Бесплатные и широко используемые технологии серверной части. PHP хорошо интегрируется с WordPress, а MySQL обеспечивает надёжное хранение структурированных данных (пользователи, тесты, ответы, результаты). Большое число хостингов автоматически поддерживает этот стек, что упрощает развёртывание и сопровождение.
* **Облачный хостинг с HTTPS и резервным копированием**  
  Размещение на облачном сервере гарантирует высокую доступность (сайт «не упадёт» при росте нагрузки) и автоматические бэкапы. Поддержка HTTPS необходима для защиты личных данных пользователей и соответствует современным требованиям безопасности.

2. Процесс разработки

1. Сбор требований  
   Определили, какая функциональность нужна: регистрация и вход, создание тестов, проведение тестирования, просмотр результатов и материалов для разбора ошибок.
2. Проектирование базы данных и интерфейсов  
   Спроектировали таблицы для хранения сущностей — пользователей, тестов, вопросов, ответов и результатов. Нарисовали схемы экранов: страницы с тестами, результаты, админка.
3. Настройка WordPress  
   Создали кастомные типы записей и таксономии для тестов, вопросов и справочных материалов. Определили пользовательские роли и права доступа.
4. Верстка и стилизация  
   Реализовали адаптивный дизайн страниц на HTML и CSS, чтобы интерфейс корректно выглядел на разных устройствах.
5. Клиентская логика на JavaScript  
   Добавили таймер, навигацию по вопросам и динамическое обновление контента без перезагрузки страницы.
6. Серверная логика на PHP  
   — Реализовали регистрацию, вход, распределение прав по ролям.  
   — Реализовали генерацию теста и сохранение результатов в базе.  
   — Настроили получение и отображение разборов ошибок и ссылок на дополнительные материалы.
7. Тестирование  
   — Проверили работу на разных браузерах и устройствах.  
   — Прогнали нагрузочные тесты для оценки стабильности при большом числе пользователей.  
   — Проверили защиту от распространённых уязвимостей (SQL-инъекции, XSS).
8. Развёртывание и поддержка  
   Разместили сайт на хостинге, настроили систему резервного копирования и мониторинг логов. Обеспечили возможность быстрого исправления ошибок и добавления новых функций.

**Вывод:** весь стек выбран исходя из доступности, надёжности и минимальных затрат времени на разработку и поддержку. Это позволяет быстро развивать проект, легко масштабировать его и гарантировать стабильную работу для всех пользователей.