**УТВЕРЖДАЮ**

Зав. кафедрой программной инженерии

В.В. Смелов

**Экзаменационные вопросы по дисциплине «Технология программирования в Интернет»**

**(2 курс, специальность ПИ)**

1. Интернет: определение, принцип устройства и организации, службы Интернет, понятие Интернет-сервера.
2. HTTP: определение, основные свойства, структура запроса, структура ответа, статусы ответа, заголовки (примеры).
3. Web-приложение: определение, структура, общая схема web-приложения.
4. ASP.NET CORE: назначение, основные свойства, кроссплатформенность, общая структура приложения, маршрутизация, типы приложений ASP.NET CORE, программный менеджер NuGet, встроенные компоненты.
5. ASP.NET CORE: проект, назначение файлов проекта: lanchsetting.json, appsettings.json, Program.cs. Основные настраиваемые (builder) компоненты. Объект HttpContext. Директорий wwwroot.
6. ASP.NET CORE: middleware: определение, назначение, принцип устройства, встроенные элементы. Принцип разработки пользовательского элемента. Встроенные компоненты платформы, подключаемые как middleware –элементы.
7. ASP.NET CORE: статические файлы: назначение, применение, скачивание статических файлов.
8. ASP.NET CORE: приложение miniAPI: структура приложения, конечные точки MapXXX, UseExceptionHandler, Fallback-обработчик, обработка параметров и входных данных, заголовки Content-Type, Accept. Маршрутизация запросов с помощью MapGroup.
9. ASP.NET CORE: приложение miniAPI: фильтры конечных точек, обработка параметров (GetArgument), делегат next.
10. ASP.NET CORE: приложение miniAPI: скачивание файлов.
11. ASP.NET CORE: приложение miniAPI: шаблоны и констрейнты маршрутов.
12. ASP.NET CORE: приложение miniAPI: применение сервиса LinkGenerator, поименованные конечные точки (WithName)переадресация с помощью RedirectToRoute.
13. ASP.NET CORE: приложение miniAPI: понятие привязки модели Model binding), атрибуты FromXXX, применение собственных типов в шаблонах маршрута, необязательные параметры, массивы параметров.
14. ASP.NET CORE: приложение miniAPI: хорошо известные типы в параметрах.
15. ASP.NET CORE: приложение miniAPI: пользовательские привязки, атрибут AsParameter.
16. ASP.NET CORE: приложение miniAPI: валидация

параметров запроса с помощью встроенного механизма (MinimaApisExtentions).

1. ASP.NET CORE: принцип разработки middleware-обработчика ошибок (Use, UseMiddleware, расширение IApplicationBuilder).
2. ASP.NET CORE: понятие и принцип применения Dependency Injection, создание Transient, Singleton b Scope-сервисы, атрибут FromServices.
3. ASP.NET CORE: приложение Razor Page: паттерн MVVM, структура приложения (View, Model), файл Program.cs (builder.Services, app.Map….), маршрутизация, порядок прохождения и обработки запроса, обработчики зпросов генерации ответа, применение Dependency Injection в обработчиках.
4. ASP.NET CORE: приложение Razor Page: Razor-разметка @{…}, @page, @using, @inject, @model, @funtions, @if, @foreach.
5. ASP.NET CORE: приложение Razor Page: Layout-макетирование, применение файла Layout.cshtml.css., применение директив @RenderBody, @RenderSection, @section.
6. ASP.NET CORE: приложение Razor Page: привязка параметров к модели - отображение (mapping).
7. ASP.NET CORE: приложение Razor Page: схемы работы с формами, обработка параметров формы, пересылка (upload) файлов с клиента на сервер с помощью формы
8. ASP.NET CORE: приложение Razor Page: понятие tag helpers, применение частичных представлений,
9. ASP.NET CORE: приложение MVC: определение, назначение, схема приложения, структура проекта.
10. ASP.NET CORE: приложение MVC: маршрутизация, таблица маршрутизации, шаблоны маршрутов.
11. ASP.NET CORE: приложение MVC: контроллеры, жизненный цикл контроллера, фильтры котроллеров, применение Dependency Injection в контроллерах, обработка параметров, способы передачи данных во View.
12. ASP.NET CORE: приложение MVC: представление, жизненный цикл представления, применение Dependency Injection в контроллерах в представлениях, макетирование представлений, @Razor-разметка.
13. ASP.NET CORE: приложение MVC: встроенные и пользовательские Html-helpers.
14. ASP.NET CORE: приложение Web API: понятие REST, назначение Web API, схема приложения, возможные реализации Web API, особенности контроллеров API.

Доцент кафедры программной инженерии В.В. Смелов