СЕССИЯ 1

1. разработка БД и заполнение её тестовыми данными

2. веб-интерфейс регистрация пациента

3. веб-интерфейс госпитализация

4. Настольный клиент «НАПРАВЛЕНИЕ ПАЦИЕНТОВ НА ЛЕЧЕБНО-ДИАГНОСТИЧЕСКИЕ МЕРОПРИЯТИЯ»

СЕССИЯ 2

1. ФУНКЦИЯ «ОТСЛЕЖИВАНИЕ ДВИЖЕНИЯ КЛИЕНТОВ И СОТРУДНИКОВ ПО ПОМЕЩЕНИЯМ БОЛЬНИЦЫ» - настольный или веб (API от экспертов)

2. ФУНКЦИЯ «РЕГИСТРАЦИЯ ДВИЖЕНИЯ ПАЦИЕНТОВ ПО СТАЦИОНАРУ И ЗАНЯТОСТЬ КОЕЧНОГО ФОНДА» - настольный или веб

СЕССИЯ 3

1. Wireframe и диаграммы UML. Основные: деятельнрсти, последовательности, вариантов использования; реже диаграммы классов и состояний.

2. Небольшое мобильное приложение

СЕССИЯ 4

1. Доработка БД(расширение)

2. Разработка настольного приложения

3. ФУНКЦИЯ «РЕЗЕРВИРОВАНИЕ РЕСУРСОВ» (\*только проектирование БД, не требует реализации).

!!!ВАРИАТИВ!!!

При желании работодателя эти 2 сессии могут быть изменены хоть на 100%

СЕССИЯ 5

1. Разработка мобильного приложения, а также ситуативно API

СЕССИЯ 6

1. Руководство пользователя, также может быть заменено на руководство программиста.

Смотрел вебинар проскакивала информация что в данном случае госты не требуются

2. Презентация

Чемпионат будет: 21.03.2024

**Что следует изучить:**

1. Создание диаграмм UML (изучить до 22.02):

Диаграмма деятельности

Диаграмма вариантов использования

Диаграмма последовательности

1. Создание БД (изучить до 29.02):

DBeaver

DbContext:

Microsoft.EntityFrameworkCore;

Microsoft.EntityFrameworkCore.SqlServer;

Создание моделей, правильное подключение к БД, переопределение функции OnConfiguring.

1. WebAPI (изучить до 15.03):

Настройка CORS - app.UseCors(builder => builder.AllowAnyOrigin());

Для корректной работы browser-приложения.

Атрибут <JsonSerializerIsReflectionEnabledByDefault>true</JsonSerializerIsReflectionEnabledByDefault> - для решения проблемы с JSON- сериализацией

Создание клиента (HttpClient)

1. Настольное приложение, впрочем, то же самое и для мобильного (изучить до 20.03):

HttpClient (опять же для обращения к API)

Avalonia (повторить)

(изучить до 21.03)

1. Функция drag&drop - <https://github.com/AvaloniaUI/Avalonia.Xaml.Behaviors> - тут есть пример реализации
2. Библиотека для создания QRCode (<https://github.com/MikeCodesDotNET/Avalonia-QRCode>)
3. Библиотека для создания капчи (<https://github.com/topics/captcha?l=c%23&o=desc&s=updated>) – тут список, сам выберешь, что тебе понравится
4. MessageBox для сообщений пользователю (<https://github.com/AvaloniaCommunity/MessageBox.Avalonia>) – не забывай, на андроид окон нет, тут это учтено, главное правильно настроить, но по дефолту сообщение всплывает посередине экрана на платформе с одним view, в остальном происходит создание нового окна