

# Обучающие материалы

АС Сенат

Exported on 10/15/2020

## Table of Contents

<b>1</b>	<b>Навыки для стажеров Backend и АТ .....</b>	<b>3</b>
1.1	Тестовые задачи: .....	3
1.2	Программа обучения .....	5
<b>2</b>	<b>Специальные навыки стажеров АТ(автоматизации тестирования) .....</b>	<b>8</b>
2.1	Тестовые задания .....	8

# 1 Навыки для стажеров Backend и AT

1. Основы ООП, принципы, примеры. Вступление из Б Эккель про ООП из Java
2. Концепты ООП в C#
3. Что такое C#, .Net Framework, .Net Core, CLR, как выполняется (посоветовать Рихтера, но лучше рассказать наглядно)
4. Visual Studio, nuget
5. Базовый синтаксис основных конструкций в C#: циклы, условия, события, делегаты (Троелсен)
6. Основы проектирования в C#: абстракции, проектирование через интерфейсы, почему так, какие проблемы решаются
7. Основы SQL: простые запросы, задачи из инета
8. Основы SQL, продолжение: просто о сложном: индексы, планы запросов
9. Коллекции в .Net Framework и как пример generic-ов: типизированные и нетипизированные коллекции
10. Основы мультипоточности и асинхронности, краткая история, зачем это нужно, синхронизации
11. Основы проектирования: базовые паттерны, классификация (singleton, abstract fabric, builder (StringBuilder, Custom Logger Builder), декоратор (логирование), visitor), посоветовать книгу Теплякова
12. Linq для работы с коллекциями, lambda expression, понятие сигнатуры
13. ORM, зачем нужны и что делают
14. Сериализация, что это и зачем
15. Основы проектирования: паттерны корпоративных приложений: repository, unit of work
16. Основы сетевого стека: модель ISO OSI, стек TCP/IP, механизмы COMET
17. Основы серверов: верхнеуровневая работа, виды, особенности путешествия запроса в IIS и как он попадает в ASP.NET приложение, OWIN и Katana (Дать видео из Тенопарка про web)
18. Базовая архитектура любого приложения: эволюция от Transaction Script к Service + Repository к DDD и CQRS (агрегаты, анемия доменной модели, POCO и DTO)
19. Архитектурные компоненты: мапперы, валидаторы, разделение на слои
20. Основы DI: декомпозиция, контейнеры, pure DI (без контейнеров), пример из блога Симана
21. Принципы SOLID
22. Основы проектирования API: RPC, REST, json формат, передача параметров
23. Основы тестов: виды тестов, framework-и, mock-и и stub-ы
24. Основы DevOps: CI, CD, CDP

## 1.1 Тестовые задачи:

1. Написать программу на работу с файлами, консолью, сериализацией, коллекциями - основы C#.
  - Назначение: создавать сущности из файлов через api
  - Функционал: приложение подтягивает файлы JSON или XML, читает из них и дергает api на создание вопросов, заседаний, пишет в лог что сделали
  - Цель: научиться взаимодействовать с api, сериализации, логированию, обработка ошибок
  - Дополнительно: рассказать про основные понятия: вопросы, комитет, заседания, повестка, проект решения
  - Как полезное развитие: приложение для создания тестовых данных в определенных статусах
  - Технологии: HttpClient или RestSharp, NLog, сериализация.
  - Ответственный за формулировку задания: [Анищенко Дмитрий Борисович](#)<sup>1</sup>
2. Написать приложение на LINQ и EF.
  - Назначение: получать данные из хранилища

<sup>1</sup> <https://sbtatlas.sigma.sbrf.ru/wiki/display/~16699962>

- Функционал: приложение коннектится к тестовой БД и выполняет Query, реализовать два варианта : через ADO.NET и через EF, также тесты на Query, mock-и, посмотреть на планы запросов
  - Цель: научиться взаимодействовать с БД, писать запросы SQL, понимать проблемы, использовать EF, основы тестирования
  - Как полезное развитие: дать Query и тесты написать
  - Дополнительно: рассказать про основные понятия: участники вопросов, участники заседания, роли, про изолированность слоя DAL и доменного слоя(это к AutoMapper)
  - Технологии: EF, LINQ, основы тестирования, nunit, mock-и, AutoMapper, ADO.NET, SQL Server, план запросов
  - Ответственный за формулировку задания: [Сыромятников Дмитрий Сергеевич](#)<sup>2</sup>
3. Написать простое приложение ASP.NET WebApi2 и развернуть его на IIS, выполняет действия из пунктов 1 и 2
- Назначение: прокси для взаимодействия с API АС сенат
  - Функционал: предоставить API для выполнения функционала из задач 1 и 2, предоставляет аутентификацию и авторизацию, фоновый job для обработки файлов
  - Цель: познакомиться с ASP.NET WebApi2 и разворачиванием приложения на IIS, понятиями аутентификации, авторизации, OWIN
  - Как полезное развитие: улучшить работу механизмов аутентификации и заполнения информации по пользователям, имперсонации
  - Дополнительно: рассказать про процесс более подробно, как есть и к чему стремимся
  - Технологии: ASP.NET WebApi2, IIS, HangFire, Owin, Identity
  - Ответственный за формулировку задания: [Филиппов Антон Сергеевич](#)<sup>3</sup>
4. Написать приложение для многопоточности.
- Назначение: производите манипуляции с админкой посредством параллельной обработки разных источников
  - Функционал: какой-нибудь producer-consumer, читать из какой-нибудь очереди в несколько потоков, заполнять внутреннюю коллекцию буффер(два варианта: concurrent коллекции и lock-и), и на основе этого делать отправки в АС Сенат запросы на взаимодействие с админкой: блокировки, заведения пользователей, перемещения по комитетам
  - Цель: познакомиться с основами многопоточности, проблемами, блокировками, асинхронной моделью
  - Как полезное развитие: улучшить и исправить API админки, перевести все репозитории на async/await для повышения нагрузочности
  - Дополнительно: рассказать про орг. структуру, роли администраторов
  - Технологии: ConcurrentCollection-ы, Threading, Tasks, async/await
  - Ответственный за формулировку задания: [Фёдоров Дмитрий Владиславович](#)<sup>4</sup>
5. Спроектировать приложение для материалов и эволюционировать его по паттернам, покрытие тестами.
- Назначение: создание материалов и их редактирование
  - Функционал: создавать, редактировать возврат пагинированной коллекции материалов
  - Цель: познакомиться с основами проектирования и эволюционирования приложений, видами архитектур
  - Как полезное развитие: улучшения CQRS стека, добавив параллельные обработки и асинхронность событий, улучшить сохранение материалов
  - Дополнительно: рассказать про материалы, обработки документов, подписания
  - Технологии: WebApi, DI, паттерны, EF
  - Ответственный за формулировку задания: [Филиппов Антон Сергеевич](#)<sup>5</sup>

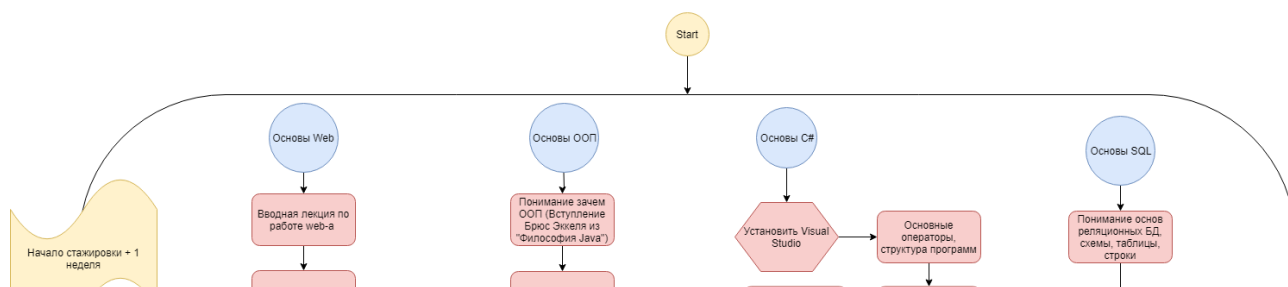
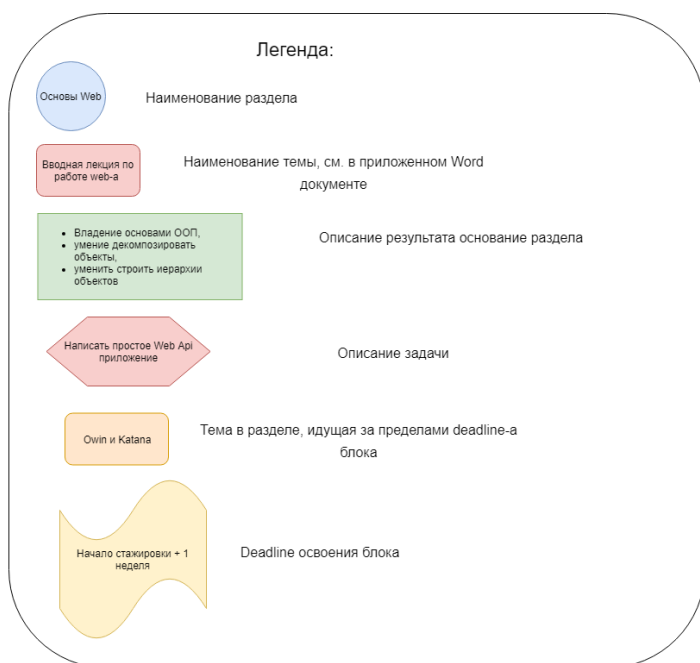
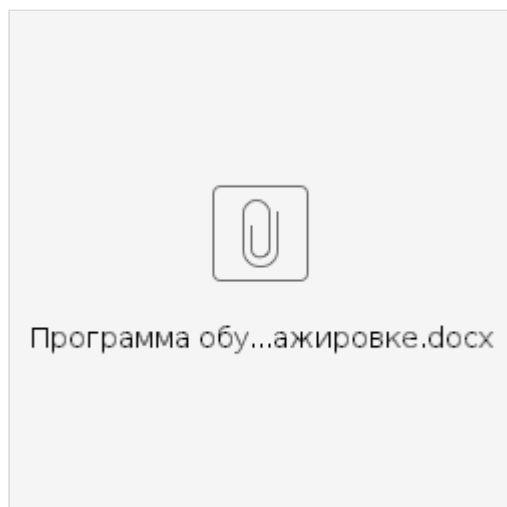
<sup>2</sup> <https://sbtatlas.sigma.sbrf.ru/wiki/display/~16715910>

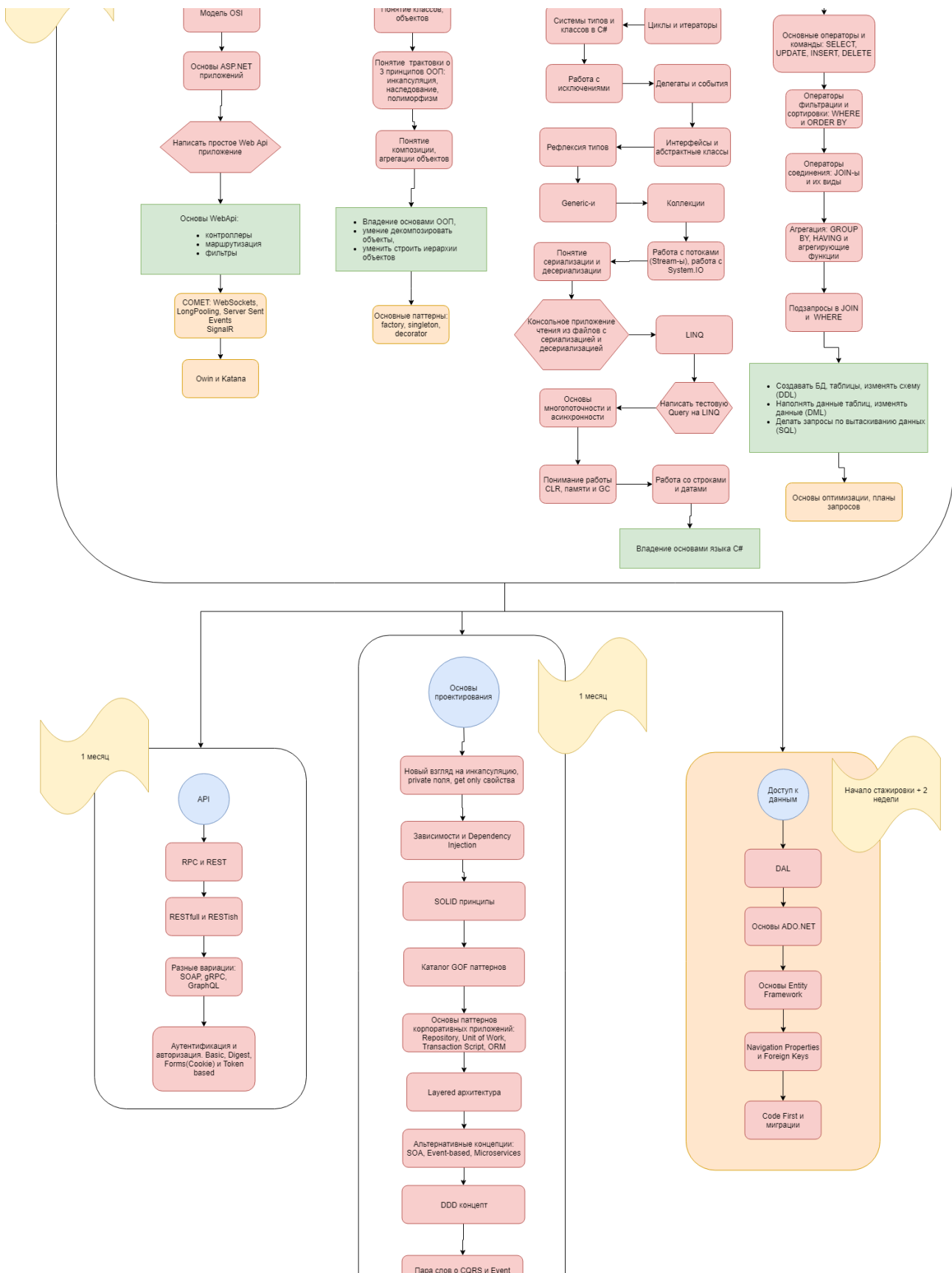
<sup>3</sup> <https://sbtatlas.sigma.sbrf.ru/wiki/display/~16726932>

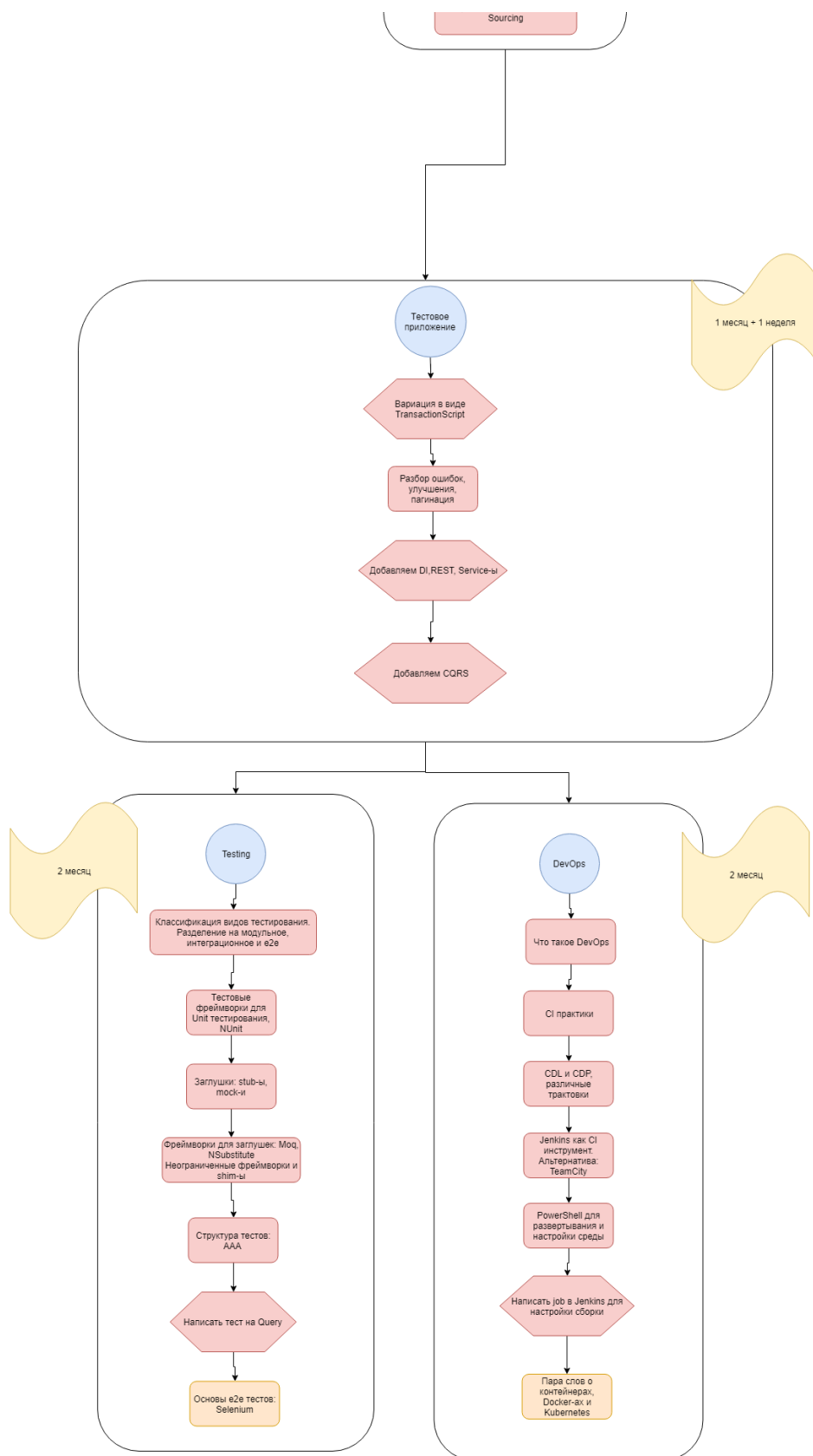
<sup>4</sup> <https://sbtatlas.sigma.sbrf.ru/wiki/display/~16773996>

<sup>5</sup> <https://sbtatlas.sigma.sbrf.ru/wiki/display/~16726932>

## 1.2 Программа обучения







## 2 Специальные навыки стажеров АТ(автоматизации тестирования)

1. Знание основ верстки (HTML, CSS)
2. Знание принципа работы Selenium WebDriver
3. Понимание и умение строить основные Selenium локаторов (id, class, css, XPath)
4. Понимание видов тестирования (модульное, интеграционное, end-to-end)
5. Знание паттерна PageObject и его вариаций (Void, Static, Fluent)
6. Умение работать с тестовым фреймворком (JUnit)
7. Навыки написания генераторов тестовых данных
8. Умение формировать результаты тестового прогона (Allure)
9. Навыки написания тестов под Android
10. Навыки написания тестов под iOS
11. Умение писать API и HT тесты (JMeter)

### 2.1 Тестовые задания

1. Написание простого e2e теста на страницу логина
2. Написание e2e теста на сценарий с несколькими пользователями. Например, создание вопроса и проверка недоступности вопроса для других ролей
3. Написание простого unit-test-a
4. Написание сложного e2e сценария с синхронизацией действий пользователей по websocket-ам
5. Написание API теста

План стажировки

Лекция	Практика	Длительность	Ответственные
Основы C#	Клиент для генерации вопросов Сенат	1 неделя	
ООП	Добавить генерацию очных / заочных заседаний	1 неделя	
Шаблоны проектирования	Добавить стратегию и декоратор для отправки запросов		Анищенко д.?
Веб-сервисы	Создать веб-сервис		
БД	Добавить Persists слой		
Многопоточность			
Проектирование	Адаптировать приложение под DDD, CQRS	1 неделя	Анищенко Д



Неделя	Тема недели	Раздел	Лекция	Тема		Длительность (занятия)	День	Практическое задание
1	Основы	ООП	ООП	Основы ООП, принципы, примеры. Вступление из Б Эккель про ООП из Java	Филиппов А.	1	пн	Установить Visual Studio  Создать консольный проект 10 простых заданий
				Концепты ООП в C#				
		Основы C#	.Net и окружение	Что такое C#, .Net Framework, .Net Core, CLR, как выполняется (посоветовать Рихтера, но лучше рассказать наглядно)	Сыромятников Д.	1		
				Visual Studio, nuget				
			Синтаксис	Базовый синтаксис основных конструкций в C#: циклы, условия, события, делегаты (Троелсен). Аналогии с конструкциями языка C	Сыромятников Д.	1		
		Шаблоны проектирования	Основы проектирования	Основы проектирования в C#: абстракции, проектирование через интерфейсы, почему так,какие проблемы решаются	Анищенко Д.	1	ср	Задания на работу с LINQ и коллекциями
		Основы C#	Коллекции	Коллекции в .Net Framework и как пример generic-ов: типизированные и нетипизированные коллекции	Сыромятников Д.	1		
			Linq	Linq для работы с коллекциями, lambda expression, понятие сигнатуры	Сыромятников Д.	1		
Проверка заданий						2	пт	-

Неделя	Тема недели	Раздел	Лекция	Тема		Длительность (занятия)	День	Практическое задание
2	Web	Веб-сервисы	Сеть	Основы сетевого стека: модель ISO OSI, стек TCP/IP, механизмы COMET	<b>Филиппов А.</b>	1	пн	Создать web сервис и запустить на IIS (заготовка Практического задания №3)
			Веб-сервисы с .Net	Основы серверов: верхнеуровневая работа, виды, особенности путешествия запроса в IIS и как он попадает в ASP.NET приложение, OWIN и Katana (Дать видео из Тенопарка про web). История, устройство	<b>Филиппов А.</b>	3		
			API	Основы проектирование API: RPC, REST, json формат, передача параметров. Показать примеры API и контрпримеры плохого API. Познакомить с web консолью браузера	<b>Фёдоров Д.</b>	1		
		Основы C#	Сериализация	Сериализация, что это и зачем, что такое XML и т.д.	<b>Сыромятников Д.</b>	1	ср	Практическое задание №1 (обращение к API Сената)
		Многопоточность	Многопоточность	Основы мультипоточности и асинхронности, краткая история, зачем это нужно, синхронизации ( <a href="http://www.albahari.com/threading/">http://www.albahari.com/threading/</a> )	<b>Сыромятников Д.</b>	1		
		Проверка заданий				2	пт	-
3	Основы проектирования и SQL	Шаблоны проектирования	Паттерны GoF	Основы проектирования: базовые паттерны, классификация (singleton, abstract fabric, builder (StringBuilder, Custom Logger Builder), декоратор(логирование), visitor), посоветовать книгу Теплякова	<b>Анищенко Д.</b>	2	пн	Практическое задание №2

Неделя	Тема недели	Раздел	Лекция	Тема		Длительность (занятия)	День	Практическое задание
			Паттерны корп. приложений	Основы проектирования: паттерны корпоративных приложений: repository, unit of work	Анищенко Д.	1	ср	
		БД	Основы SQL	Основы SQL: что такое БД, простые запросы, задачи из инета	Фёдоров Д.	1		
		Шаблоны проектирования	Архитектура	Архитектурные компоненты: мапперы, валидаторы, разделение на слои	Анищенко Д.	2		
		БД	Основы SQL	Основы SQL, продолжение: просто о сложном: индексы, планы запросов	Фёдоров Д.	1		
		Проверка заданий					пт	
4	Проектирование	Шаблоны проектирования	Слоенная архитектура	Базовая архитектура любого приложения: эволюция от Transaction Script к Service + Repository к DDD и CQRS (агрегаты, анемия доменной модели, POCO и DTO)	Анищенко Д.	2	пн	Практическое задание №5
		Шаблоны проектирования /БД	ORM	ORM, зачем нужны и что делают	Фёдоров Д.	2		
		Шаблоны проектирования	DI	Основы DI: декомпозиция, контейнеры, pure DI (без контейнеров), пример из блога Симана	Анищенко Д.	1	ср	
		Шаблоны проектирования	SOLID	Принципы SOLID	Анищенко Д.	2		

Неделя	Тема недели	Раздел	Лекция	Тема		Длительность (занятия)	День	Практическое задание
		Проверка заданий					пт	
5	Тестирование и DevOps	CI/CD	Автотестирование	Основы тестов: виды тестов, framework-и, mock-и и stub-ы	<b>Анищенко Д.</b>	2	пн	Добавить unit test-ы
			DevOps	Основы DevOps: CI, CDL, CDP	<b>Такжонов А.</b>	1		
		Проверка заданий						

1. Основы ООП, принципы, примеры. Вступление из Б Эккель про ООП из Java - **Филиппов А.**
2. Концепты ООП в C# - **Филиппов А.**
3. Что такое C#, .Net Framework, .Net Core, CLR, как выполняется (посоветовать Рихтера, но лучше рассказать наглядно) **Сыромятников Д.**
4. Visual Studio, nuget - **Сыромятников Д.**
5. Базовый синтаксис основных конструкций в C#: циклы, условия, события, делегаты (Троелсен) - **Сыромятников Д.**
6. Основы проектирования в C#: абстракции, проектирование через интерфейсы, почему так, какие проблемы решаются - **Анищенко Д.**
7. Основы SQL: простые запросы, задачи из инета - **Фёдоров Д.**
8. Основы SQL, продолжение: просто о сложном: индексы, планы запросов - **Фёдоров Д.**
9. Коллекции в .Net Framework и как пример generic-ов: типизированные и нетипизированные коллекции - **Сыромятников Д.**
10. Основы мультипоточности и асинхронности, краткая история, зачем это нужно, синхронизации (<http://www.albahari.com/threading/>) - **Сыромятников Д.**
11. Основы проектирования: базовые паттерны, классификация (singleton, abstract fabric, builder (StringBuilder, Custom Logger Builder), декоратор (логирование), visitor), посоветовать книгу Теплякова - **Анищенко Д.**
12. Linq для работы с коллекциями, lambda expression, понятие сигнатуры - **Сыромятников Д.**
13. ORM, зачем нужны и что делают - **Фёдоров Д.**
14. Сериализация, что это и зачем, что такое XML и т.д. - **Сыромятников Д.**
15. Основы проектирования: паттерны корпоративных приложений: repository, unit of work - **Анищенко Д.**
16. Основы сетевого стека: модель ISO OSI, стек TCP/IP, механизмы COMET - **Филиппов А.**
17. Основы серверов: верхнеуровневая работа, виды, особенности путешествия запроса в IIS и как он попадает в ASP.NET приложение, OWIN и Katana (Дать видео из Тенопарка про web) - **Филиппов А.**
18. Базовая архитектура любого приложения: эволюция от Transaction Script к Service + Repository к DDD и CQRS (агрегаты, анемия доменной модели, POCO и DTO) - **Анищенко Д.**
19. Архитектурные компоненты: мапперы, валидаторы, разделение на слои - **Анищенко Д.**
20. Основы DI: декомпозиция, контейнеры, pure DI (без контейнеров), пример из блога Симана - **Анищенко Д.**

21. Принципы SOLID - **Анищенко Д.**
22. Основы проектирование API: RPC, REST, json формат, передача параметров. Показать примеры API и контрпримеры плохого API. Познакомить с web консолью браузера - **Фёдоров Д.**
23. Основы тестов: виды тестов, framework-и, mock-и и stub-ы - **Анищенко Д.**
24. Основы DevOps: CI, CDL, CDP - **Такжанов А.**