



СОФИЙСКИ УНИВЕРСИТЕТ "СВ. КЛИМЕНТ ОХРИДСКИ"
ФАКУЛТЕТ ПО МАТЕМАТИКА И ИНФОРМАТИКА

УЧЕБНА ПРОГРАМА

ОКС „бакалавър”

Избираема дисциплина

Утвърдил:

/декан/

Утвърдена с решение на ФС с протокол:

№ от

редовна форма на обучение	
Специалност:	(код и наименование)
Софтуерно инженерство, Компютърни науки, Информационни системи, Информатика, Приложна математика, Анализ на данни	

Дисциплина:	(код и наименование)
DevOps автоматизация в облачните технологии	
DevOps Automation in Cloud Technologies	

Учебната програма е разработена и предложена за утвърждаване от катедра:	
Информационни Технологии (ИТ)	
от:	Георги Георгиев

Преподавателските екипи се утвърждават ежегодно от Факултетен съвет.

Заетост и кредити		
Обща заетост:		150
Кредити:		5
Учебна заетост	Форма	Хорариум
Аудиторна заетост	Лекции	30
	Семинарни упражнения	30
	Практически упражнения (хоспитиране)	
Обща аудиторна заетост:		60
Кредити аудиторна заетост:		2
Извънаудиторна заетост	Подготовка на домашни работи (малки проекти)	
	Контролни работи и подготовка за тях	20
	Учебен проект	
	Самосотятелна работа в библиотека или с интернет ресурси	30
	Доклад/Презентация	
	Друг вид извънаудиторна заетост	
Подготовка за изпит		40
Обща извънаудиторна заетост:		90
Кредити извънаудиторна заетост:		3

Предвидена форма на оценяване:	И
--------------------------------	---

Формиране на оценката по дисциплината

№	Показател	%
	Контролни работи	
	Участие в час	
	Домашни работи	
	Учебен проект (разработване и защита)	
	Тестова проверка	33
	Текуща самостоятелна работа/контролна работа	
	Демострационни занятия	
	Участие в тематични дискусии	
	Решаване на казуси	
	Изпит - практика (решаване на задачи)	
	Изпит - теория (финален тест в края на курса)	67

Анотация на учебната дисциплина

Курсът запознава студентите с DevOps концепцията, която обединява разработката на софтуер (Dev) и ИТ операциите (Ops), насърчавайки култура на сътрудничество и споделена отговорност.

Основната цел на DevOps е да съкрати жизнения цикъл на разработка на ИТ системи, да осигури непрекъсната доставка на приложения с високо качество на софтуера и да отговори по-бързо на нуждите на клиентите. Участниците ще придобият основно разбиране за това как практиките и философиите на DevOps революционизират връзката между разработката на софтуер и ИТ операциите. Чрез съпоставяне на историята на разработването на софтуер и ИТ операциите преди появата на DevOps, студентите ще оценят еволюцията към по-интегриран подход.

В курса се разглеждат съвременни операционни системи, сървърна виртуализация, контейнери, както и работата в мрежа. Разглежда се и разработването на приложения, и принципите за внедряване и управлението им в облачни инфраструктури.

Предварителни изисквания

От кандидатите за този курс се изисква да имат базови компютърни умения. Необходимо е и свободно ползване на технически английски език.

Очаквани резултати

След успешното завършване на този курс, студентите ще могат:

- да работят в компютърни мрежи;
- да създават и управляват виртуални машини и контейнери;
- да разработват приложения за работа в контейнери;
- да инсталират и управляват приложения във виртуални машини или контейнери;
- да извършват мониторинг за нормалната работа на изградените информационни системи;
- да усвоят добри практики за решаване на проблеми.

№	Тема	Хорариум л./с.упр./пр.
1	Модул 1: Въведение в DevOps 1.1. История на разработката на софтуер и ИТ операциите 1.2. Преди DevOps 1.3. Какво е DevOps 1.4. DevOps Toolchain 1.5. Демо и дискусия: Настройване на среда за курса	2 2 0
2	Модул 2: Операционни системи и работа в мрежа 2.1. Какво е операционна система 2.2. История и еволюция на операционните системи 2.3. Видове операционни системи 2.4. Архитектури на операционните системи 2.5. Сървърна срещу настолна ОС 2.6. Съвременни парадигми в сървърните ОС 2.7. Основи на работата в мрежа 2.8. Мрежови инструменти и техники	2 2 0
3	Модул 3: Основи на програмирането и приложенията 3.1. Програмни езици 3.2. Езици за програмиране на ниско ниво срещу езици за програмиране на високо ниво 3.3. Интерпретатори срещу компилатори при езиците за програмиране 3.4. Библиотеки 3.5. Frameworks 3.6. Паралелизъм и асинхронни парадигми 3.7. Настолни приложения 3.8. Мобилни приложения 3.9. Уеб приложения 3.10. Уеб услуги 3.11. Данни и устойчивост на приложения - RDBMS, NoSQL кеширане, опашки за съобщения 3.12. Демонстрация и дискусия: Разработване на приложение помощта на python	3 3 0
4	Модул 4: Системи за контрол на версиите 4.1. История на системите за контрол на версиите 4.2. Централизирано срещу разпределено управление 4.3. Git 4.4. GitHub, GitLab, BitBucket и др. 4.5. Самостоятелно хостване срещу облак 4.6. Работа в екип 4.7. Клонове (Branch) 4.8. Git History 4.9. Git Blame 4.10. Pull Request / Merge Request 4.11. Стратегии за разклоняване (Branching Strategy) 4.12. Демо и дискусия: GitHub	2 2 0

5	Модул 5: Внедряване на приложения 5.1. Bare Metal 5.1.1. Разбиране на Bare Metal 5.1.2. Bare Metal сървъри 5.1.3. Стратегии за внедряване 5.1.4. Производителност и сигурност 5.1.5. Дискусия: Предимства и недостатъци от използването на Bare Metal реализации 5.2. Виртуализация 5.2.1. Разбиране на виртуализацията 5.2.2. Как работи хардуерната виртуализация 5.2.3. Виртуални машини (VM) 5.2.4. Хипервайзори 5.2.5. Пара- срещу пълна виртуализация 5.2.6. Синтетична виртуализация 5.2.7. Самостоятелно хоствани срещу облачни виртуални машини 5.2.8. Демонстрация и дискусия: Създаване на VM и инсталиране на софтуер 5.3. Контейнери 5.3.1. Разбиране на контейнерите 5.3.2. Как Docker става де факто стандарт 5.3.3. Работа с Docker 5.3.4. Docker файл 5.3.5. Демонстрация и дискусия: Настройка на контейнерна среда на VM. Изграждане на имидж. Стартиране на контейнер. 5.4. Оркестрация на контейнери 5.4.1. Необходимост от оркестрация 5.4.2. Платформи за оркестриране на контейнери 5.4.3. Въведение в Kubernetes 5.4.4. Демонстрация и дискусия: Настройване на kubernetes и инсталиране на pods	6	6	0
6	Модул 6: Облачни технологии и DevOps 6.1. Въведение в облачните технологии 6.2. Облачни модели 6.3. Облачни доставчици 6.4. Публични, частни и хибридни облачни решения 6.5. Управлявани услуги (Managed Services) 6.6. Големите 3 - AWS, Azure, Google 6.7. Други облачни доставчици - Alibaba, Oracle, IBM и др. 6.8. Други доставчици на PaaS (Mongo Atlas и др.) 6.9. Multi-Cloud (Комбиниране на различни услуги от различни облачни доставчици) 6.10. Облачни изчисления и DevOps 6.11. Дискусии: Как да прецените кой доставчик е подходящ за вашите нужди? Дали мулти-облакът е добър вариант за вашите работни натоварвания?	2	2	0

7	Модул 7: Amazon Web Services (AWS) 7.1.Основни услуги на AWS 7.2.IaaS срещу PaaS 7.3.IaaS в действие 7.4.Синергия между управляваните услуги 7.5.Работа в мрежа в AWS 7.6.Разширени услуги на AWS 7.7.Демонстрация и дискусия: Внедряване на приложение върху AWS IaaS и PaaS	2	2	0
8	Модул 8: Continuous Integration (CI), Continuous Delivery (CD), Continuous Deployment (CD) 8.1.Въведение в CI 8.2.Защо CI 8.3.Изграждане на CI pipelines 8.4.Въведение в CD 8.5.Continuous Delivery срещу Continuous Deployment 8.6.Изграждане на CD pipelines 8.7.Pipelines като код 8.8.Идеология за много среди 8.9.Идеология за управление на тайните (secrets) 8.10.Демонстрация и дискусия: Създаване на CI/CD pipelines	2	2	0
9	Модул 9: Инфраструктура като код (IaC) и Configuration Management 9.1.Разбиране на IaC и Configuration Management 9.2.IaC срещу Configuration Management 9.3.Инструменти за IaC и Configuration Management 9.4.Terraform като IaC инструмент 9.5.Ansible като инструмент за управление на конфигурациите 9.6.Демонстрация и дискусия: Създаване на Terraform решение за AWS инфраструктура, Създаване на Ansible playbook за конфигуриране на виртуални машини.	2	2	0
10	Модул 10: Наблюдаемост и мониторинг (мониторинг, регистриране и предупреждение) 10.1.Въведение в наблюдаемостта и мониторинга (Observability and Monitoring) 10.2.Ключови понятия 10.3.Осъществяване на мониторинг 10.4.Събиране и анализиране на логове 10.5.Внедряване на известия 10.6.Демонстрация и дискусия: Настройване на наблюдение и мониторинг на инфраструктурата и приложенията	2	2	0

11	Модул 11: DevOps и сигурност 11.1.Основни принципи на сигурността (Defense in Depth) 11.2.Сигурност на инфраструктурата 11.3.Сигурност на приложението 11.4.Физическа сигурност 11.5.Мрежова сигурност 11.6.Как наблюдението и мониторинга са свързани със сигурността 11.7.Интегриране на сигурността в DevOps CI/CD Pipeline 11.8.Управление на самоличността и достъпа 11.9.Управление на тайните (secrets) 11.10.Демонстрация и дискусия: Интегриране на DevSecOps в жизнения цикъл на приложението	3	3	0
12	Модул 12: Тестване на софтуер 12.1.Въведение в софтуерното тестване 12.2.Принципи и методологии на тестване 12.3.Видове тестване на софтуер 12.4.Прилагане на стратегия за тестване	2	2	0

Конспект за изпит	
№	Въпрос
1	Въведение в DevOps
2	Операционни системи и работа в мрежа
3	Основи на програмирането и приложенията
4	Системи за контрол на версиите
5	Внедряване на приложения
6	Облачни технологии и DevOps
7	Amazon Web Services (AWS)
8	Continues Integration (CI), Continues Delivery (CD), Continues Deployment (CD)
9	Инфраструктура като код (IaC) и Configuration Management
10	Наблюдаемост и мониторинг (мониторинг, регистриране и предупреждение)
11	DevOps и сигурност
12	Тестване на софтуер

Библиография
Основна
1. http://rsmt.it.fmi.uni-sofia.bg/DevOps/

Допълнителна

1. Docker Guides. <https://docs.docker.com/guides/>
2. Ansible community documentation. <https://docs.ansible.com/>
3. GitLab documentation. <https://docs.gitlab.com/>

Дата: 25.04.2024 г.

Съставил: Георги Георгиев

СПРАВКА ПРЕПОДАВАТЕЛСКИ ЕКИПИ

Софтуерно инженерство, Компютърни науки, Информационни системи, Информатика

DevOps автоматизация в облачните технологии

[illegible]