


МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний аерокосмічний університет ім. М. Є. Жуковського
«Харківський авіаційний інститут»

Кафедра 105 «Інформаційних технологій проектування»

ЗАТВЕРДЖУЮ

Голова НМК 2


Д.М. Крицький
« 31 » 08 2021 року

**РОБОЧА ПРОГРАМА ВИБІРКОВОЇ
НАВЧАЛЬНОЇ ДИСЦИПЛІНИ**

СИСТЕМИ КОНТРОЛЮ ВЕРСІЙ

(шифр і назва навчальної дисципліни)

Галузь знань: _____ 10 «Природничі науки»,
_____ 11 «Математика та статистика»,
_____ 12 «Інформаційні технології»,
_____ 15 «Автоматизація та приладобудування»,
_____ 16 «Хімічна та біоінженерія»,
_____ 17 «Електроніка та телекомунікації»,
_____ 19 «Архітектура та будівництво»
(шифр і назва галузі знань)

Спеціальність: _____ для усіх спеціальностей зазначених галузей
(шифр і назва спеціальностей)

Освітня програма: _____ для усіх освітніх програм зазначених галузей
(найменування освітньої програми)

Форма навчання: денна

Рівень вищої освіти: другий (магістерський)

Харків 2021

Робоча програма «Системи контролю версій»

(назва дисципліни)

для студентів галузі знань: 10 «Природничі науки»,

11 «Математика та статистика»,

12 «Інформаційні технології»,

15 «Автоматизація та приладобудування»,

16 «Хімічна та біоінженерія»,

17 «Електроніка та телекомунікації»,

19 «Архітектура та будівництво»

(шифр і назва галузі знань)

спеціальностей:

для усіх спеціальностей зазначених галузей

(шифр і назва спеціальностей)

освітня програма:

для усіх освітніх програм зазначених галузей

(найменування освітньої програми)

31.08.2021 р., – 8 с.

Розробник: Каратанов О. В., доцент кафедри 105 «Інформаційних

(прізвище та ініціали, посада, наукова ступінь та вчене звання)

технологій проектування», к.т.н.


(підпис)

Робочу програму розглянуто на засіданні кафедри інформаційних технологій проектування

(назва кафедри)

Протокол № 1 від 31.08.2021 року

Завідувач кафедри 105, к.т.н., доцент

(наукова ступінь та вчене звання)


(підпис)

Д.М. Крицький

(ініціали та прізвище)

1 Опис навчальної дисципліни

Найменування показників	Галузь знань, спеціальність, освітня програма, рівень вищої освіти	Характеристика навчальної дисципліни
		денна форма навчання
Кількість кредитів – 5	<p style="text-align: center;">Галузь знань</p> 12 «Інформаційні технології», 10 «Природничі науки», 11 «Математика та статистика», 12 «Інформаційні технології», 15 «Автоматизація та приладобудування», 16 «Хімічна та біоінженерія», 17 «Електроніка та телекомунікації», 19 «Архітектура та будівництво»	<i>Вибіркова</i>
Модулів – 2		Навчальний рік
Змістових модулів – 4		2021/2022
Загальна кількість годин – 150		Семестр
		1-й
Тижневих годин для денної форми навчання: аудиторних – 3 самостійної роботи студента – 5,6		Лекції*
	32 год.	
	Практичні, семінарські	
	0 год.	
	Лабораторні*	
	32 год.	
	Самостійна робота	
	86 год.	
Вид контролю:		
	іспит	

Співвідношення кількості годин ауд. занять до сам. роботи становить для денної форми $64 / 86 = 0,74$.

*Аудиторне навантаження може бути зменшеним або збільшеним на одну годину в залежності від розкладу занять.

2 Мета та завдання навчальної дисципліни

Об'єкт вивчення – процеси функціонування розподілених технологій автоматизованого проектування та створення систем, що їх використовують.

Предмет вивчення – розподілені технології автоматизованого проектування.

Мета навчання – надання слухачам знань, умінь, навичок та методичних прийомів, що необхідні для використання розподілених технологій автоматизованого проектування.

Завданнями дисципліни є:

- опанувати базові знання щодо основ контролю версій за допомогою Git,
- освоїти сучасні підходи до розподіленої розробки,
- отримати знання з сучасних технологій контролю версій,
- ознайомитись з архітектурою розподілених систем контролю версій,
- вивчити основні команди Git та опанувати графічні клієнти Git

Студенти повинні досягти таких компетентностей:

- здатність до абстрактного мислення, аналізу та синтезу.
- здатність застосовувати знання у практичних ситуаціях.
- навички використання інформаційних і комунікаційних технологій.
- навички роботи з інформацією (уміння знаходити та аналізувати інформацію з різних джерел).

- здатність генерувати нові ідеї (креативність).

- здатність приймати обґрунтовані рішення.

- здатність оцінювати та забезпечувати якість виконуваних робіт.

- визначеність і наполегливість щодо поставлених завдань і взятих обов'язків.

В результаті засвоєння курсу студенти повинні

знати:

- область використання систем контролю версій;
- класифікацію систем контролю версій;
- типові підходи при роботі з такими системами;
- правила та методи роботи та використання систем контролю версій при реалізації IT-проектів.

вміти:

- використовувати на практиці теоретичні знання з питань використання систем контролю версій, вибору найбільш ефективних рішень щодо їх використання;
- створювати та використовувати локальні та віддалені репозиторії;
- взаємодіяти у командній розробці програмного забезпечення.

мати навички:

- використання Git.

3 Програма навчальної дисципліни

Змістовний модуль 1. Загальні принципи роботи систем контролю версій

Тема 1. Вступ до курсу «Системи контролю версій»

Знайомство з курсом та викладачем. Вимоги до успішного складання курсу. Вакансій на ринку праці, які потребують знання систем контролю версій. Термінологія систем контролю версій. Класифікація систем контролю версій.

Тема 2. Огляд систем контролю версій

Історія систем контролю версій. RCS. CVS. Subversion. Git. Mercurial.

Тема 3. Git. Встановлення та налаштування. Робочий процес

Налаштування операційної системи – *git config* (Ubuntu/MacOS/Windows, примітка – CRLF), встановлення Git, створення облікового запису на Github. Знайомство з основами Git. Перші команди git. Докладно розбираємо процес від початку роботи до фіксації результату в Git: як створити репозиторій (*git init*), додати файл (*git add*) і зробити комміт (*git commit*).

Тема 4. Інтеграція з Github.

Вчимося налаштовувати GitHub, створювати в ньому репозиторій та з'єднувати його з локальним репозиторієм. А також клонувати репозиторій, створений на GitHub на свій комп'ютер. GitHub Remote – Push & Pull

Тема 5. Робоча директорія (Working Directory). Індекс

Що таке робоча директорія і чим вона відрізняється від репозиторію, вчимося відновлювати файли. Перевірка статусу файлів. Методи роботи з індексом.

Тема 6. Аналіз змін та скасування коммітів

Аналіз зроблених змін. Знайомимось із командою *git diff*, яку потрібно обов'язково запускати перед кожним коммітом. Аналіз історії змін (коммітів). Вчимося отримувати різноманітну інформацію про минулі комміти: хто, коли і як змінював код. Вивчаємо команди, які дозволяють вирішити це завдання: *log*, *show*, *blame*, *grep*. Скасування змін у робочій директорії. Відкаг змін, які зроблено в робочій директорії, але ще не потрапили до комміту. Вчимося повертати змінені файли до початкового стану. Скасування коммітів. Що робити, якщо коміт уже зроблено, але з якихось причин нас не влаштовує? Вивчаємо спеціальні команди, що дозволяють спростити скасування або зміну комміту: *revert*, *reset*. Зміна останнього комміту. Зробили коміт, але забули додати частину файлів? Розбираємось, як додати зміни до поточного комміту, не створюючи новий. Переміщення з історії. Git дозволяє не тільки переглядати історію, а й переміщатися нею, завантажуючи в робочу директорію стан коду на момент виконання будь-якого комміту. Розбираємось, як це робити.

Тема 7. Розуміння Git

Основна «робота» Git – формування безлічі однозв'язних списків, що складаються з коммітів. Знайомимось з ключовим поняттям Git та терміном «гілка». Ігнорування файлів (*Gitignore*). Вивчаємо поняття "ігнорування файлів" і вчимося правильно визначати, що додавати до репозиторій, а що ні. *Stash*. Як «ховати» зміни у робочій директорії та відновлювати їх за необхідності? Знайомимось із командою *stash*.

Змістовний модуль 2. Розподілена розробка програмного забезпечення

Тема 8. Відкриті проекти (Open Source)

Як влитися у розробку відкритих проектів, навчитися командній роботі та покращити своє портфоліо.

Тема 9. Стили найменування коммітів

Домовленість щодо найменування коммітів. Типи комміту.

Тема 10. Віртуальна команда

Що таке віртуальна команда? Як підвищити ефективність роботи віртуальної команди?

4 Структура навчальної дисципліни

Назви змістових модулів і тем	Кількість годин					
	денна форма					
	усього	у тому числі				
л.		п. р.	лаб.	к. р.	с. р.	
Змістовий модуль 1						
Тема 1	4	2	–	–	–	2
Тема 2	4	2	–	–	–	2
Тема 3	14	2	–	4	–	8
Тема 4	16	4	–	4	–	8
Тема 5	16	2	–	4	–	10
Тема 6	18	4	–	4	–	10
Тема 7	22	4	–	8	–	10
Разом за змістовим модулем 1	94	20	–	24	–	50
Змістовий модуль 2						
Тема 8	20	4	–	4	–	12
Тема 9	16	4	–	–	–	12
Тема 10	20	4	–	4	–	12
Разом за змістовим модулем 2	56	12	–	8	–	36
Усього годин за дисципліною	150	32	–	32	–	86

5 Теми семінарських занять

Семінарські заняття навчальним планом не передбачені.

6 Теми практичних занять

Практичні заняття навчальним планом не передбачені.

7 Теми лабораторних занять

№ з/п	Назва теми	Кількість годин
1	SVN	4
2	Git. SmartGit	4
3	Git. Qt	4
4	Git. Visual Studio	4
5	Git. Консоль	4
6	SCRUM. Product Backlog	4
7	Розподілена розробка проектної документацій	4
8	Модульна контрольна робота	4
	Разом	32

8 Самостійна робота

№ з/п	Назва теми	Кількість годин
1	<u>Поточна самостійна робота</u> : робота з літературними джерелами та програмною документацією, робота над конспектом та тестовими завданнями, підготовка до участі в олімпіаді з програмування	86
	Разом	86

9 Курсова робота

Курсова робота навчальним планом не передбачена.

10 Методи навчання

Лекції проводяться з використанням основних розділів конспекту лекцій в електронній формі, елементів мультимедійної підтримки курсу (презентації), демонстрацій окремих прийомів роботи з інструментальним середовищем та/або роздаточного матеріалу у вигляді схем та діаграм.

Лабораторні роботи виконуються з використанням навчальних (демонстраційних) та ліцензованих робочих версій функціональних модулів СУБД.

Самостійна робота включає підготовку до лабораторних робіт, модульного контролю та заліку, виконання розрахунково-графічної роботи і вивчення вказаних вище тем за конспектом, літературними джерелами та програмною документацією.

11 Методи контролю

Поточний контроль здійснюється відповідно до повноти, своєчасності та якості виконання лабораторних робіт та домашнього завдання (розрахунково-графічної роботи).

Проміжний (модульний) контроль – письмова контрольна робота на 7 та 14-му тижні.

Підсумковий контроль – письмовий залік та усні відповіді на запитання викладача за курсом.

12 Критерії оцінювання та розподіл балів, які отримують студенти

12.1 Розподіл балів, які отримують студенти

Складові навчальної роботи	Бали за одне заняття (завдання)	Кількість занять (завдань)	Сумарна кількість балів
Модульні контрольні роботи	0–50	2	0–50
Лабораторні роботи	1–5	8	8–40
Робота на лекціях	0–1	10	0–10
Усього за семестр			8–100

Семестровий контроль (іспит/залік) проводиться у разі відмови студента від балів поточного тестування та за наявності допуску до іспиту/заліку. При складанні семестрового іспиту/заліку студент має можливість отримати максимум 100 балів.

12.2 Якісні критерії оцінювання

Проміжна атестація включає в себе теоретичні завдання, що дозволяють оцінити рівень засвоєння учнями знань, і практичні завдання, що виявляють ступінь сформованості умінь і володінь.

Засвоєні знання і освоєння вміння перевіряються за допомогою модулів, вміння і володіння перевіряються в ході вирішення завдань.

Обсяг і якість освоєння учнями дисципліни, рівень сформованості дисциплінарних компетенцій, оцінюються за результатами поточних і проміжної атестацій кількісною оцінкою, вираженої в балах, максимальна сума балів з дисципліни дорівнює 100 балам.

12.3 Критерії оцінювання роботи студента протягом семестру

Сума балів, набраних студентом з дисципліни, переводиться в оцінку відповідно до таблиці.

Сума балів з дисципліни	Оцінка за проміжної атестації	Характеристика рівня освоєння дисципліни
від 90 до 100	«зараховано» / «відмінно»	Студент демонструє сформованість дисциплінарних компетенцій на підсумковому рівні, виявляє всебічне, систематичне і глибоке знання навчального матеріалу, засвоїв основну літературу і знайомий з додатковою літературою, рекомендованою програмою, вміє вільно виконувати практичні завдання, передбачені програмою, вільно оперує набутими знаннями, вміннями, застосовує їх у ситуаціях підвищеної складності.
від 75 до 89	«зараховано» / «добре»	Студент демонструє сформованість дисциплінарних компетенцій на середньому рівні: основні знання, вміння освоєні, але допускаються незначні помилки, неточності, труднощі при аналітичних операціях, перенесення знань і умінь на нові, нестандартні ситуації.
від 60 до 74	«зараховано» / «задовільно»	Студент демонструє сформованість дисциплінарних компетенцій на базовому рівні: в ході контрольних заходів допускаються значні помилки, виявляється відсутність окремих знань, умінь, навичок за деякими дисциплінарним компетенціями, студент відчуває значні труднощі при оперуванні знаннями та вміннями при їх перенесенні на нові ситуації.
від 41 до 59	«не зараховано» / «незадовільно»	Студент демонструє сформованість дисциплінарних компетенцій на рівні нижче базового, проявляється недостатність знань, умінь, навичок.
від 0 до 40	«не зараховано» / «незадовільно»	Дисциплінарні компетенції не формувати. Виявляється повне або практично повна відсутність знань, умінь, навичок.

Шкала оцінювання: бальна і традиційна

Сума балів	Оцінка за традиційною шкалою	
	Іспит, диференційований залік	Залік
90 – 100	Відмінно	Зараховано
75 – 89	Добре	
60 – 74	Задовільно	
0 – 59	Незадовільно	Не зараховано

13 Методичне забезпечення

Навчально-методичний комплекс дисципліни у електронному вигляді знаходиться на кафедральних серверах та на гугл-диску – Каратанов О. В. Навчально-методичний комплекс дисципліни [Електронний ресурс] / О. В. Каратанов. – Х. : XAI, 2020. – Режим доступу : https://drive.google.com/file/d/1fDoiKwXdob7r-IVuKYA_Nvopk1bESPI2/

14 Рекомендована література

14.1 Базова

1. Git - Основи Git [Електронний ресурс]. – Режим доступу: <https://git-scm.com/book/ru/v2>
2. GIT HowTo [Електронний ресурс]. — Режим доступу <http://githowto.com>
3. Git Magic - Preface [Електронний ресурс]. — Режим доступу: <http://www-cs-students.stanford.edu/~blynn/gitmagic/intl/ru/>
4. Code School - Try Git [Електронний ресурс]. — Режим доступу: <http://try.github.io/levels/1/challenges/1>
5. git - the simple guide [Електронний ресурс]. — Режим доступу: <http://rogerdudler.github.io/git-guide/>

14.2 Допоміжна

1. Введение в Git: от установки до основных команд [Електронний ресурс]. — Режим доступу: <https://tproger.ru/translations/beginner-git-cheatsheet/>