

Міністерство освіти і науки України
Харківський національний економічний університет
імені Семена Кузнеця

Кафедра комп'ютерних систем і технологій

Міждисциплінарний курсовий проект
студентки 2-го курсу
спеціальності 186 «Видавництво та поліграфія»
першого (бакалаврського) рівня всіх форм навчання
на тему «Розрахунок і візуальне моделювання механічного
пристрою засобами комп'ютерної графіки»

Виконала: студентка
2 курсу, ф-т ІТ
групи 6.04.186.010.22.2
Андрющенко Є.В.

Керівник проекту:
Євсєєв О.С.

Харків, 2024

ЗМІСТ

Вступ	4
1. Розрахунок параметрів механічного пристрою одноступінчастого циліндричного редуктора	5
1.1 Опис пристрою редуктора	5
1.2 Кінематичний і силовий розрахунок приводу	5
1.2.1 Розрахунок тихохідного валу редуктора	5
1.2.2 Розрахунок конструктивних розмірів зубчатого колеса	9
1.3 Ескізне компонування редуктора	11
1.3.1 Визначення діаметрів шестерні	11
1.3.2 Розрахунок елементів корпусу редуктора	12
1.4 Опис складання й розбирання редуктора	14
2. Візуальне моделювання механічного пристрою у статичі й динаміці	14
2.1 Моделювання механічного пристрою за допомогою інструментального засобу інженерної й комп'ютерної графіки FreeCAD	14
2.1.1 Подання зовнішнього вигляду пристрою у вигляді фотографій з різних ракурсів та перелік деталей пристрою, та опис їх взаємодії	14
2.1.2 Фізичне або віртуальне «розбирання» пристрою, для уявлення форми деталей, що входять до його складу	16
2.1.3 Моделювання окремих компонентів пристрою	16
2.1.4 Моделювання зібраного пристрою	21
2.2 Моделювання механічного пристрою в динаміці за допомогою інструментального засобу тривимірного моделювання Blender	22
2.2.1 Складання й візуалізація пристрою в середовищі Blender	22
2.2.2 Моделювання роботи пристрою в динаміці	23
2.2.3 Обґрунтування параметрів рендерінгу проекту й вибору програмного засобу для відтворення експортованої моделі	23

2.3 Створення мультимедійної презентації за допомоги засобів Adobe Animate	24
2.4 Розробка дизайну електронного та друкованого складників комплексного курсового проекту	33
2.4.1 Дизайн програмної оболонки комплексного курсового проекту	29
2.4.2 Дизайн друкованого складника комплексного курсового проекту	33
2.5 Розробка мультимедійної презентації з використанням мови програмування C#	34
2.5.1 Вступ	34
2.5.2 Розробка організаційних і технологічних блок-схем сценарію мультимедійної презентації	35
2.5.3 Розробка технологічної схеми сценарію	35
2.5.4 Розробка функціональних описів елементів організаційних і технологічних блок-схем	37
2.5.5 Розробка програмного забезпечення мультимедійної презентації	39
2.5.6 Складання й публікація комплексного курсового проекту	39
2.5.7 Аналіз роботи мультимедійної презентації	39
Висновок	40
Джерела	41
Додатки	42

ВСТУП

Мета та завдання курсового проектування: розрахунок параметрів механічного пристрою з наступним візуальним моделюванням у статиці й динаміці для використання в мультимедійній презентації. Уміння виконувати розрахунок пристроїв технічної механіки; створення зображень об'єктів з використанням векторної й растрової графіки; створення композиції з використанням інструментів для створення растрової графіки; моделювання роботи пристроїв з використанням інструментів для створення анімації; створення динамічного web-сайту з елементами анімації; формування презентації з використанням мови програмування.

Курсове проектування покликане сприяти систематизації, поглибленню й закріпленню знань, отриманих студентом при вивченні дисциплін «Інженерна і комп'ютерна графіка», «Теоретична механіка», «Технічна механіка», «Комп'ютерна анімація», «Основи дизайну і композиції», «Технологія комп'ютерного дизайну», «Об'єктно-орієнтоване програмування» а також розвитку навичок практичного застосування отриманих знань при розробці моделі механічного пристрою засобами комп'ютерної графіки.

Актуальність тема:

Наразі, світ все більше віддає перевагу використанню мультимедійних засобів для навчання, розваг та інформаційної комунікації. Розрахунок параметрів механічного пристрою з візуальним моделюванням у статиці й динаміці дозволяє створювати ілюстрації та анімації для пояснення роботи пристроїв, що є важливим у процесі навчання та підвищення рівня розуміння матеріалу. Розробка механічних пристроїв та їх візуальне моделювання має велике значення для різних галузей, таких як інженерія, дизайн, архітектура та інші. Використання векторної та растрової графіки, анімації та web-технологій дозволяє створювати інтерактивні презентації, які ефективно передають інформацію та сприяють кращому розумінню матеріалу. Саме тому, проект систематизує та закріплює знання з теми, що є актуальною у контексті сучасних вимог до фахівців.

Розділ 1. «Розрахунок параметрів механічного пристрою одноступінчастого циліндричного редуктора»

Варіант 1

Підрозділ 1.1. «Опис пристрою редуктора»

Проектований одноступінчастий редуктор складається з наступних частин:

- 1) вхідний та вихідний вали
- 2) корпус (блок)
- 3) зубчаста пара
- 4) підшипники
- 5) сальники
- 6) мастило

Одноступінчастий циліндричний редуктор використовується для зменшення швидкості обертання вихідного вала і збільшення обертового моменту. Він складається з двох циліндричних зубчастих коліс - великого зубчастого колеса і малого зубчастого колеса, які взаємодіють між собою. Цей тип редуктора широко використовується у промисловості для передачі потужності від одного механізму до іншого зі зменшенням швидкості обертання. Циліндричні прямозубі редуктори можуть виготовлятися з різних матеріалів, в залежності від конкретних умов експлуатації і вимог до них.

Підрозділ 1.2. «Розрахунки».

1.2.1. Розрахунок тихохідного валу редуктора.

Вхідні дані:

- 1) Номінальний обертальний момент на вихідному валу редуктора
 $T_2 = 120000 \text{ Н}\cdot\text{м}$
- 2) Синхронна частота обертання валу електродвигуна $n_1 = 530 \text{ мін}^{-1}$.
- 3) Частота обертання $n_2 = 265 \text{ мін}^{-1}$.

Для розрахунку валів необхідно знати розміри їх ділянок по довжині, сили, що діють, і крутні моменти, що допускаються напруги матеріалу.

Розрахунок I ділянки валу за формулою: $d_1 = \sqrt[3]{\frac{T_2}{0.2[\tau]}}$

$$d_1 = \sqrt[3]{\frac{120000}{0.2 * 25}} = \sqrt[3]{\frac{120000}{5}} = \sqrt[3]{24000} = 29$$

де d_1 - діаметр вихідного кінця (хвостовика) валу, мм; T_2 - крутний момент на валу, що розраховується, Н·м; $[\tau]$ - знижена допустима напруга на кручення, $[\tau] = 25$ МПа для валу зі сталі 45.

Довжину хвостовика вибирають у діапазоні $l_1 = (1,5 \dots 2)d_1$.

$$l_1 = 2 * 29 = 58.$$

Розрахунок II ділянки валу визначається за формулою: $d_2 = d_1 + 5$.

$$d_2 = 29 + 5 = 34.$$

Довжина ділянки l_2 визначається за формулою: $l_2 = L_k + y$

$$l_2 = 18 + 10 = 28$$

L_k - величина, яка залежить від товщин опорної поверхні кришки підшипника, пружинної шайби і висоти головки болта кріплення кришки до корпусу (при зовнішньому діаметрі підшипника тихохідного валу $D < 105$ мм можна прийняти $L_k = 18$ мм, при $D \geq 105$ мм - $L_k = 22$ мм); y - приймається у діапазоні 5...10 мм.

Попередній діаметр ділянки III визначають за формулою: $d_3 = d_1 + 10$.

$$d_3 = 29 + 10 = 40.$$

Отримане значення необхідно узгодити із внутрішнім діаметром підшипника (табл. 1.1). Приймаємо більше значення.

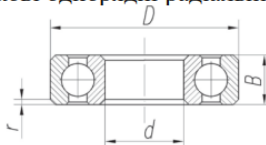
Довжина ділянки l_3 визначається за формулою: $l_3 = (10 \dots 15) + B$

$$l_3 = 15 + 18 = 33$$

де B - ширина підшипника тихохідного валу (табл. 1.1).

Таблиця 1.1

Підшипники кулькові однорядні радіальні (ГОСТ 8338)



Позначення	Розміри, мм				
	d	D	B	r	d _{зн}
204	20	47	14	1.5	25...26
205	25	52	15	1.5	30...30.5
206	30	62	16	1.5	35...37
207	35	72	17	2	42
208	40	80	18	2	47...48
209	45	85	19	2	52...53
210	50	90	20	2	57...58
211	55	100	21	2.5	62...65
212	60	110	22	2.5	67...71
213	65	120	23	2.5	72...77
214	70	125	24	2.5	77...82
215	75	130	25	2.5	82...85
216	80	140	26	3.0	90...92
217	85	150	28	3.0	95...99
218	90	160	30	3.0	100...105
219	95	170	32	3.5	107...111
220	100	180	34	3.5	112...117

Діаметр ділянки IV визначають за формулою: $d_4 = d_3 + 5$.

$$d_4 = 40 + 5 = 45.$$

Довжина ділянки l₄ визначається за формулою: $l_4 = L_c - 2$

$$l_4 = 24 - 2 = 22$$

де L_c - довжина маточини. Для визначення значення L_c потрібно:

- 1) Визначити передаточне числа: $u_{12} = \frac{n_1}{n_2}$, де n_2 та n_1 – частоти обертання відповідно вихідного та вхідного валів.

$$u_{12} = \frac{530}{265} = 2$$

- 2) Визначити міжосьову відстані передачі: $a = (u + 1)^3 \sqrt{\left(\frac{K}{\sigma_H \cdot u}\right)^2 \cdot \frac{T \cdot K_H}{\psi_a}}$, де K - коефіцієнт, для прямозубих коліс він дорівнює 315; σ_H – нормальна напруга у матеріалах, Мпа (420); K_H - коефіцієнт концентрації навантаження (1,17); ψ_a – коефіцієнт ширини зубчастого колеса (0,15).

- 3) Визначити ширину зубчастого колеса: $b_2 = 0.15 \cdot a$

$$b_2 = 0.15 \cdot 152 = 22.8$$

- 4) Визначити довжину маточини: $L_c = 1.2 \cdot b_2$

$$L_c = 1.2 \cdot 22.8 = 27$$

Діаметр ділянки V визначають за формулою: $d_5 = d_4 + 3$.

$$d_5 = 45 + 3 = 48.$$

Довжина ділянки 15 визначається за формулою: $l_5 = 10 \dots 15$ мм.

$$l_5 = 15 \text{ мм.}$$

Діаметр ділянки VI визначають за формулою: $d_6 = d_{зп}$

$$d_7 = 40$$

де $d_{зп}$ - діаметр заплечика підшипника (табл. 1.1)

Довжина ділянки 16 визначається за формулою: $l_6 = 10 \dots 15$ мм.

$$l_6 = 10 \text{ мм.}$$

Діаметр ділянки VII визначають за формулою: $d_7 = d_3$

$$d_7 = 40$$

Довжина ділянки 17 визначається за формулою: $l_7 = B + (2 \dots 3)$.

$$l_7 = 18 + 2 = 20$$

Визначаємо розміри шпонкового пазу. Розміри призматичних шпонок та пазів для них наведені в табл. 1.2, де прийняті такі позначення: t_1 - глибина шпонкового паза на валу; t_2 - глибина шпонкового паза в маточині (рис. 1.1)

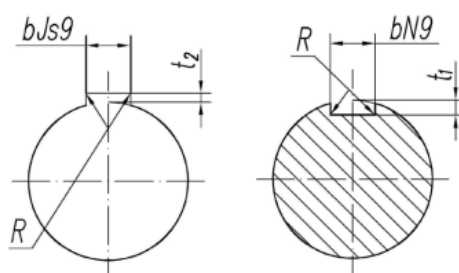


Рис. 1.1 «Пази шпонкові»

D1

Св. 22 до 30	8	7	4	3.3	18...90
--------------	---	---	---	-----	---------

$t_1 = 4$

$t_2 = 3,3$

довжина-18

R-0,25

D4

Св. 44 до 50	14	9	5.5	3.8	36...160
--------------	----	---	-----	-----	----------

 $t_1=5,5$ $t_2=3,8$

довжина-36

 $R=0,4$

Таблиця 1.2

Розміри шпонок призматичних та пазів, мм, за ГОСТ 23360

Діаметр валу d	Перетин шпонки		Глибина паза		Довжина шпонок	Найбільший радіус R
	b	h	t ₁	t ₂		
Св. 12 до 17	5	5	3	2.3	10...56	0.25
Св. 17 до 22	6	6	3.5	2.8	14...70	
Св. 22 до 30	8	7	4	3.3	18...90	
Св. 30 до 38	10	8	5	3.3	22...110	0.4
Св. 38 до 44	12	8	5	3.3	28...140	
Св. 44 до 50	14	9	5.5	3.8	36...160	
Св. 50 до 58	16	10	6	4.3	45...180	0.6
Св. 58 до 65	18	11	7	4.4	50...200	
Св. 65 до 75	20	12	7.5	4.9	56...220	
Св. 75 до 85	22	14	9	5.4	63...250	1.0
Св. 85 до 95	25	14	9	5.4	70...280	
Св. 95 до 110	28	16	10	6.4	80...320	
Св. 110 до 130	32	18	11	7.4	90...360	
Св. 130 до 150	36	20	12	8.4	100...400	
Св. 150 до 170	40	22	13	9.4	100...400	

1.2.2. Розрахунок конструктивних розмірів зубчатого колеса

Визначення передаточного числа, міжосьової відстані передачі, ширини зубчатого колеса залишити з розрахунку валу.

Визначення модуля зубчастих коліс. Рекомендується вибирати модуль m для прямозубих коліс, користуючись такими співвідношеннями: $m=0.015*a$

$$m = 0.015 * 152 = 2,5$$

Визначення числа зубів у коліс. Сумарне число зубів ($z_1 + z_2$) передачі з прямими зубами визначають за виразом: $z_{\Sigma} = \frac{2a}{m}$.

$$z_{\Sigma} = \frac{2 * 152}{2.5} = 122$$

Число зубів шестерні дорівнює: $z_1 = \frac{z_{\Sigma}}{u+1} \geq 17$

$$z_1 = \frac{122}{2 + 1} = 41$$

Число зубів колеса визначають як: $z_2 = z - z_1$.

$$z_2 = 122 - 41 = 81$$

Визначення діаметрів зубчастих коліс.

Ділильний діаметр колеса, мм: $d_2 = 2a - m \cdot z_1$.

$$d_2 = 2 * 152 - 2.5 * 41 = 304 - 102.5 = 201.5$$

Діаметри (мм) кіл вершин і западин зубчастих коліс знаходять як $d_{a2} = d_2 + 2m$; $d_{f1} = d_2 - 2.5m$.

$$d_{a2} = 201.5 + 2 * 2.5 = 206.5$$

$$d_{f1} = 34 - 2.5 * 2.5 = 27.75$$

Основні розміри зубчастих коліс (рис.1.2)

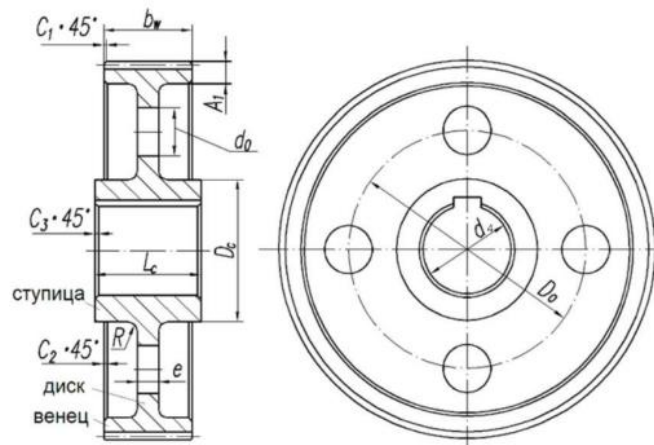


Рис. 1.2 «Основні розміри зубчастих коліс»

Діаметр маточини $D_c = 1.6 \cdot d_4$, де d_4 - діаметр посадкового отвору зубчастого колеса.

$$D_c = 1.6 * 45 = 72$$

Товщина ободку $A_1 = (5 \dots 6) m$, де m - модуль передачі.

$$A_1 = 5 * 2.5 = 12.5$$

Товщина диска $e = 0.5 b_2$.

$$e = 0.5 * 22.8 = 11.4$$

Діаметр центрального кола $D_0 = 0.5 (d_{a2} - 2A_1 + D_c)$, де d_{a2} - діаметр кола вершин зубів колеса

$$D_0 = 0.5(206,5 - 25 + 72) = 126,75$$

Діаметр отворів $d_0 = 0,25 (d_{a2} - 2A_1 - D_c)$

$$d_0 = 0.25(206.5 - 25 - 72) = 109.5$$

Підрозділ 1.3 Ескізне компоновання редуктора

Ескізна компоновка швидкохідного валу редуктора представлена на рис. 1.3.

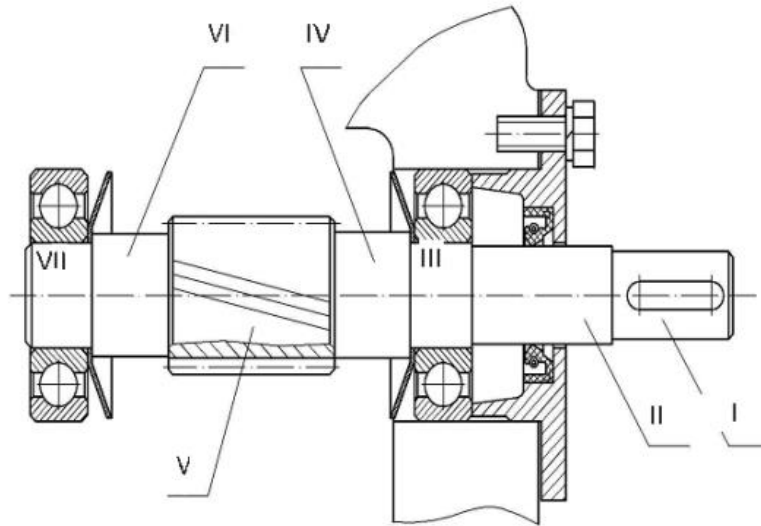


Рис.1.3 «Ескізна компоновка швидкохідного валу редуктора»

Формування зображення швидкохідного валу починають із зубчастого вінця, основні розміри якого визначаються під час розрахунку зубчастої передачі.

1.3.1 Визначення діаметрів шестерні

Ділильний діаметр шестерні, мм: $d_1 = m \cdot z_1$.

$$d_1 = 2 \cdot 41 = 82.$$

Діаметри (мм) кіл вершин і западин зубчастих коліс знаходять як $d_{a1} = d_1 + 2m$; $d_{f1} = d_1 - 2,5m$.

$$d_{a1} = 82 + 2 \cdot 2,5 = 87$$

$$d_{f1} = 82 - 2,5 \cdot 2 = 73$$

Ширину шестерні b_1 задають більше ширини колеса на величину від 3 до 7 мм і округлюють до найближчого більшого значення за формулою: $b_1 = b_2 + 5$.

$$b_1 = 23 + 6 = 29$$

1.3.2 Розрахунок елементів корпусу редуктора

При розрахунку визначають товщину стінки корпусу редуктора $\delta = 1.12 \cdot \sqrt[4]{T_2}$

$$\delta = 1.12 \cdot \sqrt[4]{120000} = 20,8 = 21$$

Діаметр фундаментного болта розраховують за формулою $d_{61} = \sqrt[3]{4T_2} \geq 12$ мм (табл.1.3)

$$d_{61} = \sqrt[3]{4 \cdot 120000} = 78 = 30 \geq 12 \text{ мм}$$

Таблиця 1.3

Параметр	Діаметр різьблення болта						
	M8	M10	M12	M16	M20	M24	M30
a_j	13	15	18	21	25	28	35
b_j	24	28	33	40	48	55	68
d_0	9	11	13	17	22	26	32
D_0	17	20	25	30	38	45	56

Ширина фланців у підшипників $L_2 = 3 + \delta + B$

$$L_2 = 3 + 21 + 18 = 42$$

При проектуванні одноступеневих редукторів на першому етапі визначається загальний КПД і необхідна потужність приводу, вибирається стандартний електродвигун і виконується кінематичний розрахунок приводу. У рамках цього розрахунку знаходять загальне передатне число приводу, розподіляють його за механічними передачами і визначають частоти обертання валів приводу, у тому числі швидкохідного та тихохідного валів редуктора. Далі знаходять потужності та крутні моменти, що передаються валами редуктора. За цими даними розраховують зубчасту або черв'ячну передачу та приступають до проектування редуктора. Загальний вигляд редуктора показано на рис. 1.4.

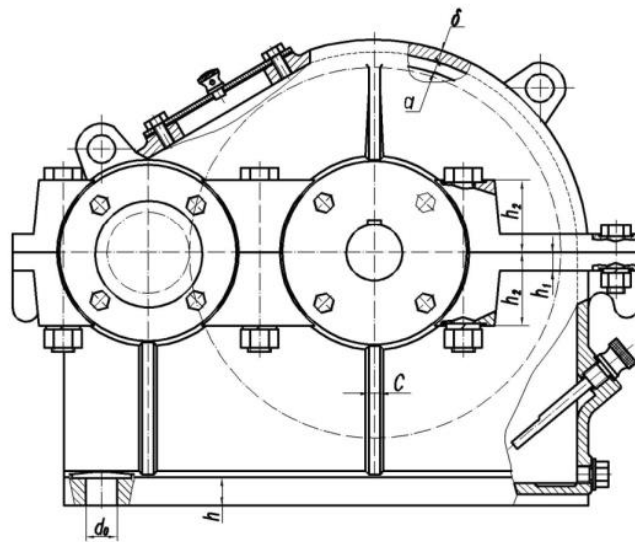


Рис 1.4 аркуш 13 «Загальний вигляд редуктора»

Підрозділ 1.4 Опис складання та розбирання редуктора

Складання одноступінчастого циліндричного редуктора є досить складним та важливим процесом, який включає кілька етапів. Перш ніж почати складання редуктора, необхідно підготувати всі необхідні інструменти, деталі та робоче місце. Перевірити деталі та компоненти на чистоту та якість. Робота починається з встановлення підшипників у корпус редуктора. Вони забезпечують гладку роботу валів. Після встановлення підшипників у корпус, установіть вали у підшипники. Важливо впевнитися, що вони правильно вирівняні та встановлені на потрібну глибину. На валах встановлюються зубчасті колеса - підйомне та провідне. Вони повинні встановлюватися так, щоб зубці правильно взаємодіяли один з одним. Шестерня, яка з'єднує підйомне колесо та провідне колесо, монтується на відповідний вал. Перед закриттям корпусу перевіряються всі з'єднання та впевніться, що всі деталі правильно встановлені та функціонують. Закривається корпус редуктора за допомогою кришки, яка фіксується болтами або гвинтами. Після складання редуктора він повинен бути підданий тестуванню для перевірки правильності складання та його ефективності. При необхідності вносяться корективи та налаштування редуктора для оптимальної роботи. Неправильне складання редуктора може призвести до його несправності або навіть поломки.

Розбирання редуктора відбувається у такій послідовності:

- 1) Відкриття корпусу. Корпус редуктора відривається, відкрутивши болти або гвинти, які утримують корпус.

- 2) Видалення редукторної пари. Виймаються редукторні пари разом зі шестернею та підшипниками з корпусу.
- 3) Роз'єднання коліс. Роз'єднується підйомне колесо від провідного колеса, викрутивши болти або гвинти, які їх з'єднують.
- 4) Огляд деталей. Оглядаються деталі на наявність пошкоджень та зносу. При потребі замінюються пошкоджені або зношені деталі.

РОЗДІЛ 2. Візуальне моделювання механічного пристрою одноступінчастого циліндричного редуктора у статиці й динаміці

Підрозділ 2.1. «Моделювання механічного пристрою за допомогою інструментального засобу інженерної й комп'ютерної графіки FreeCAD»

2.1.1. Подання зовнішнього вигляду пристрою у вигляді фотографій з різних ракурсів, перелік деталей пристрою, та опис їх взаємодії

Пристрій представляє собою компактний циліндричний корпус, який зазвичай має циліндричну форму зі згладженими краями. Загальний вигляд пристрою (рис.2.1) видає враження професійної конструкції, з урахуванням ергономічних та функціональних вимог, що забезпечують ефективну роботу редуктора в різних умовах експлуатації.



Рис.2.1 «Загальний вигляд редуктора»

Вхідний вал (рис.2.2) циліндричного редуктора є одним з ключових елементів, через який передається обертовий рух від джерела потужності до редуктора. Він знаходиться з одного з боків корпусу редуктора. Розташовується в центрі корпусу або може зсуватися ближче до верхнього або нижнього краю.

Шестерні (рис.2.3) в одноступінчастому циліндричному редукторі відіграють ключову роль у передачі обертового моменту та забезпеченні відповідного передаткового числа. Деталь має зубчасту форму з рівними.

Зазвичай у одноступінчастого циліндричного редуктора використовуються кульові або роликові підшипники. Кульові підшипники (рис.2.4) забезпечують плавне обертання при невеликих навантаженнях, тоді як роликові підшипники використовуються для більших навантажень.



Рис.2.2 аркуш 15 «Вхідний вал»



Рис.2.3 «Шестерня»



Рис.2.4 «Кульковий підшипник»

Перелік деталей пристрою: корпус, зубчасте колесо, вал ведений, підшипник та ведучий вал-шестерня.

Опис взаємодії: обертовий рух, що передається на вхідний вал, приводить у рух встановлені на ньому вхідні шестерні. Вхідні шестерні взаємодіють з вихідними шестернями, які встановлені на вихідному валу. Ця взаємодія забезпечує

передачу обертового руху від вхідного валу до вихідного валу. Обертовий рух, який передається вихідним валом, утримується і підтримується за допомогою підшипників. Підшипники забезпечують плавне обертання валу без зайвих тертя і зносу. Гвинти, гайки та інші кріпильні елементи забезпечують надійне з'єднання різних деталей редуктора і утримують їх в правильному положенні.

2.1.2. Фізичне або віртуальне «розбирання» пристрою, для уявлення форми деталей, що входять до його складу

Під час відкручування кріплень та зняття кришок редуктора обережно виконуємо кожен крок, щоб уникнути пошкодження ущільнюючих елементів. Після зняття кришок видаляємо з корпусу вхідний та вихідні вали. Таким чином отримуємо доступ до внутрішніх компонентів. Далі виймаємо підшипники з редуктора. Цей розбір пристрою допомагає очистити від забруднень та змащуючого матеріалу всі деталі.

Опис форми:

- 1) Зубці усіх шестерень прямі, вони мають форму трапеції. У кожної шестерні є допоміжні отвори для накладання на вал і тільки одна має декілька отворів, які розміщуються паралельно масивом.
- 2) Верхній корпус має циліндричну форму з плоскою або деякою вигином верхньою поверхнею. Він може мати отвори для кріплення кришок, підшипників та інших деталей. У верхньому корпусі можуть бути вирізи або виступи для кріплення додаткових компонентів, таких як підшипники або кріпильні елементи.
- 3) Нижній корпус також може мати циліндричну форму, але зазвичай він має більш плоску форму з плоскою або слабко вигнутою нижньою поверхнею. На нижньому корпусі можуть бути відводи для відведення тепла або отвори для дренажу змащуючого матеріалу.
- 4) Кожен вал містить підкладку для шестерень з випуклим отвором для чіткого попадання.
- 5) Пластикові прокладки трошки випирають з під основи й містять в собі отвори для вставки валів.

2.1.3. Моделювання окремих компонентів пристрою

Перед побудовою замірюємо всі деталі нашого редуктора та записуємо їх. Весь процес моделювання будемо проводити в реальному масштабі, а одиниці виміру, які будемо використовувати – міліметри.

Побудова валу:

За допомогою інструменту «Відрізок» малюємо товщину валу, поділену навпіл. Довжина кожної частини залишається не змінною. Таким чином з відрізків створюємо контур валу (рис.2.5) та з'єднуємо кінець з початком. Переходимо у робочий простір Part Design. Надалі створюємо тіло та вибираємо дію «Фігура обертання». При підтвердженні дії налаштовуємо горизонтально вісь ескізу. Таким чином отримуємо 3D-модель валу (рис.2.6). Наступним кроком є створення шпонкового пазу на четвертій ступені. Для цього створюємо опорну площину на циліндричній поверхні чотири. На створенній площині створюємо ескіз, де малюємо контур шпонкового пазу. Далі робимо виріз по контуру шпонкового пазу та встановлюємо глибину вирізу. Аналогічним способом створюємо шпонковий паз на першій поверхні, але з відповідними розмірами. Останнім кроком для завершення моделі – фаски на гранях. Для цього обираємо потрібну грань та застосовуємо інструмент «Фаска». Далі встановлюємо розмір фаски розраховані до цього етапу. Пристосовуємо фаски для всіх необхідних граней.

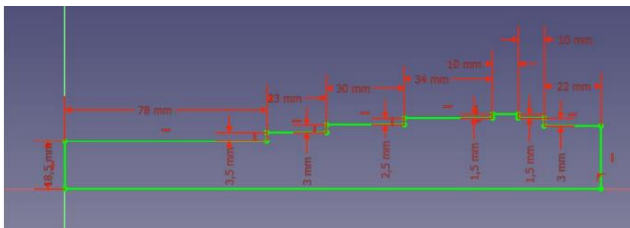


Рис.2.5 «Контур валу»

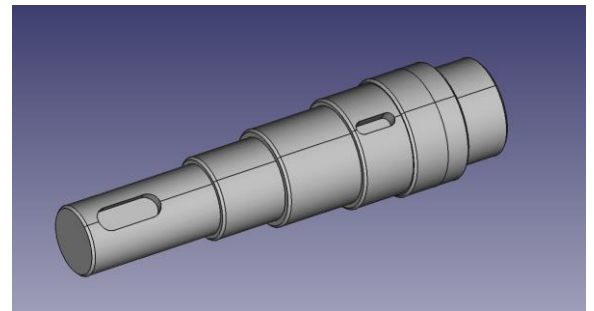


Рис.2.6 «3D-модель валу»

Побудова зубчастого колеса:

Для створення 3D-моделі зубчастого колеса відразу працюємо у робочому просторі, що б була можливість створювати нове тіло. За допомогою пункту «Евольвентна шестерня» встановлюємо параметри: кількість зубці та модуль, які було вираховані у минулому розділі. Для створення першої поверхні створюємо накладку, де можна ввести значення ширини зубчастого вінця та обрати налаштування «Симетрично до площині». Для другої поверхні необхідно за допомогою ескізу малюємо коло необхідного розміру та вирізаємо з певною глибиною вирізу. На поверхні малюємо коло з діаметром для третьої поверхні, та накладаємо з певною величиною виступу. Таким чином була створена третя поверхня.

З протилежної сторони зубчастого колеса повторюємо дії для другої та третьої поверхні (рис 2.7) Після цих кроків потрібно зробити центральний наскрізний отвір. Для цього вирізаємо наскрізь коло, яке завчасно було намальовано за наданими розмірами. На шестерні потрібно зробити чотири отвори на стінці. Для цього FreeCAD має інструмент «Круговий масив». Саме так за малий проміжок часу можна зробити відразу певну кількість отворів з певним розміром. Так як і з валом, додаємо фаски на необхідних гранях. Для шестерні потрібно закруглити стики площин. Для цього обираємо бічну площину та використовуємо інструмент «Заокруглення» та встановлюємо необхідний нам радіус закруглення. Формування шпонкового пазу. Активуємо третю площину, де створюємо ескіз з розмірами шпонкового пазу. Розміри повинні відповідати шпонковим пазам з моделі валу, для правильного складання редуктора. Ескіз вирізаємо наскрізь. 3D-модель евольвентної шестерні за заданими розмірами готова. (рис.2.7)

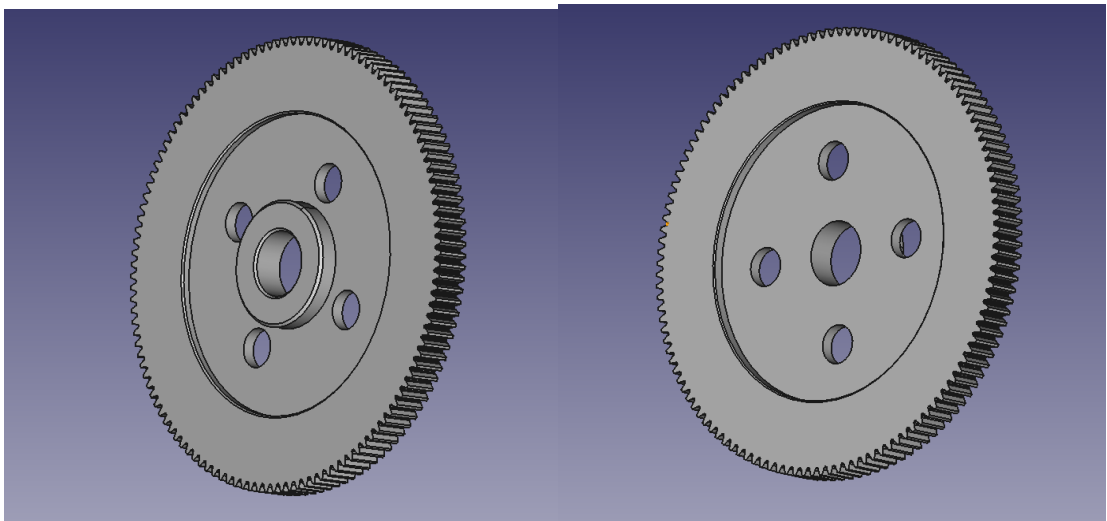


Рис 2.7 «3D -модель шестерні з обох сторін»

Побудова вала-шестерні:

3D-деталь повторює всі минулі кроки, тому моделювання не дуже важно виконується. Першим кроком створюємо Евольвентну шестерню, яка буде займати четверту ступень валу. Створюючи, налаштовуємо кількість зубців та модуль. Використовуємо функцію «Накладка» та задаємо значення ширини зубчастого вінця. Наступним кроком створення третьої ступені валу або третю поверхню. Створюємо ескіз кола з необхідним діаметром та створюємо накладку з заданою шириною. Саме за таким алгоритмом створюються друга та перша ступені та з протилежного боку шестерні п'ята то шоста ступені. Так як і при створенні валу, додаємо шпонковий паз лише на першу поверхню, оскільки на

четвертій поверхні знаходиться шестерня. Для пазу створюємо опорну площину, аби розмістити ескіз контуру та вирізати його на задану глибину. Кінцевою дією є встановити фаски на необхідних гранях. Зберігаємо готову модель валу-шестерні (рис.2.9).

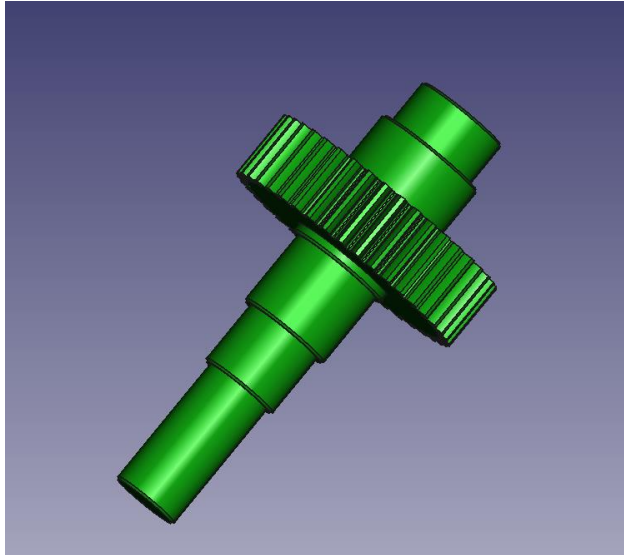


Рис.2.9 «3D-модель валу-шестерні»

Побудова нижнього корпусу:

Першим кроком є створення ескізу нижньої частини за розмірами розрахунків. Створюємо накладку з потрібним значення довжини. Закруглюємо кути з певним радіусом. Для кріплення створюємо чотири отвори на однаковій відстані від країв основи. Створюємо корпус за допомогою ескізу контуру та видавлюємо на потрібну висоту. Таким же чином створюємо контур, але для внутрішнього простору корпусу та вирізаємо до поверхні основи. Закруглюємо гострі кути всередині та зовні корпусу. Створюємо верхній пояс. Для цього малюємо ескіз, товщина поясу така ж як і основа. Закруглюємо зовнішні кути. Створюємо отвори для розміщення валів. Створюємо ескіз з діаметром підшипника та враховувати товщину бортика. Ці дії повторюємо з протилежної сторони, звертаємо увагу, що б кола знаходилися чітко посередині. Через те, що зверху з'являється непотрібна частина, тому обрізаємо її. На верхньому поясі створюємо отвори для кріплення. Обов'язково створюємо ребра жорсткості. Сполучаємо ребра жорсткості з іншими поверхнями плавними переходами. Завдяки цій послідовності будується нижній корпус (рис.2.10).

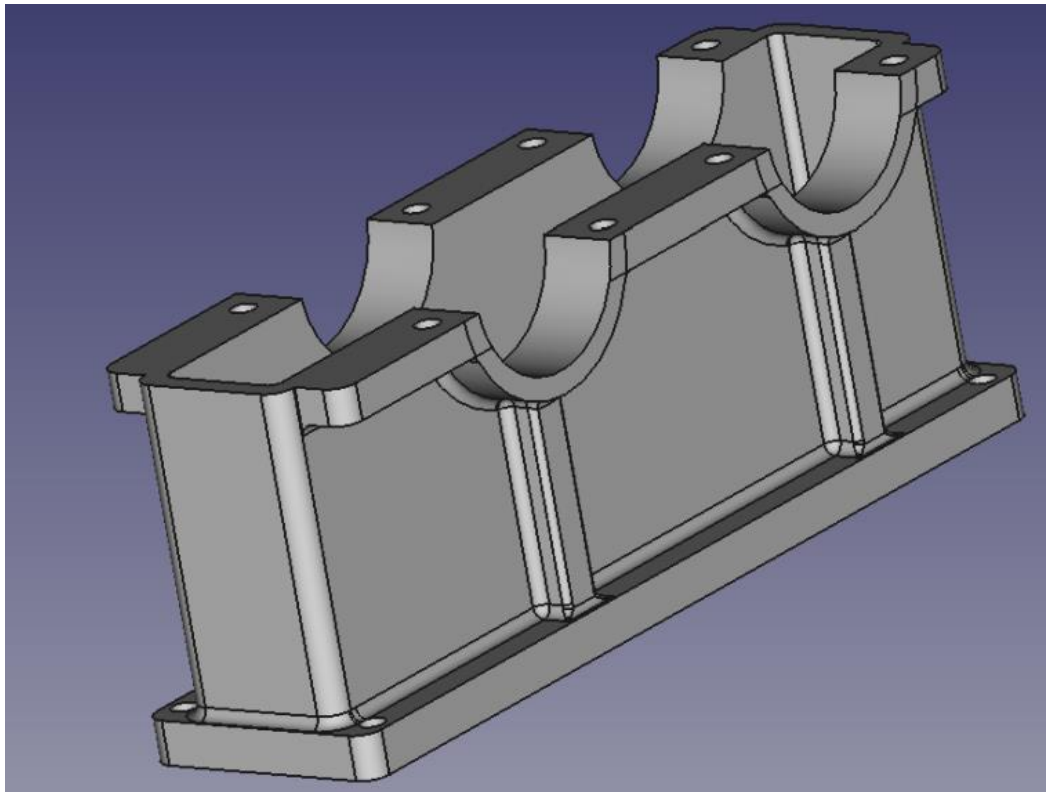


Рис 2.10 аркуш 20 «3D- модель нижнього корпусу редуктора»

Побудова підшипника:

Перш ніж перейти до побудов, необхідно визначити діаметр кульок у підшипнику. В FreeCAD встановлюємо робочий простір Part. Створюємо циліндр, де вказуємо параметри радіусу та висоту. Додаємо другий циліндр з другим радіусом. Обираємо дві моделі та вирізаємо таким чином, аби всередині великого циліндру з'явився отвір розміром з другим циліндром. Отримуємо фігуру «кільце». Таким же чином створюємо ще два циліндри всередині нашої першої фігури. Наступним кроком потрібно вирізати отвір, аби знову вийшло кільце, але з декількома областями. Обираємо середню область та вирізаємо її з циліндру. Завдяки цим крокам отримуємо зовнішню та внутрішню обойми підшипника. Інструментом «Тор» отримуємо внутрішні канавки, завчасно помістив його всередину до обойм та виконати виріз. Додаємо кульки підшипника потрібним радіусом, та засовуємо по всій канавці підшипника. Останнім кроком є фаски. За короткий час отримуємо модель підшипника (рис.2.11)

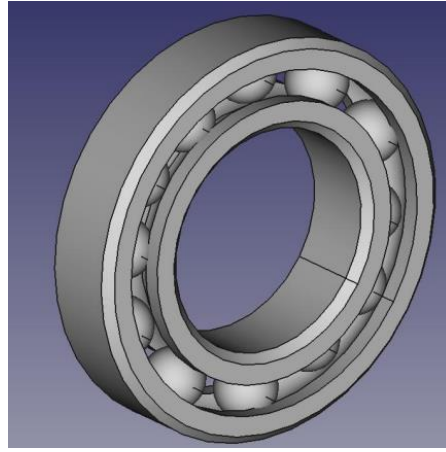


Рис 2.11 «3D-модель підшипника»

2.1.4. Моделювання зібраного пристрою

Ми побудували всі деталі, тому приступаємо до їх об'єднання. Беремо зубчасте колесо та нанизуємо його на тихохідний вал, чітко розмістивши його як воно є в оригіналі. Налаштовуємо тихохідний та швидкохідний вал, щоб чітко та правильно встали до їх відділів. Перевіряємо чи збігаються зубці, вони повинні співпасти. Ні один зубець не повинен заходити один на одного і між двома шестернями повинна бути відстань, яке забезпечує правильну роботу зацеплення й самого редуктора. Наступними кроком слід надягати підшипники з обох боків кожного валу. Вони допомагають зменшити тертя та забезпечують оптимальну передачу обертового моменту від вихідного вала до вихідного шестерні редуктора (рис. 2.12). Всі перенесення елементів виконуємо за допомогою інструменту 3D перенесення.

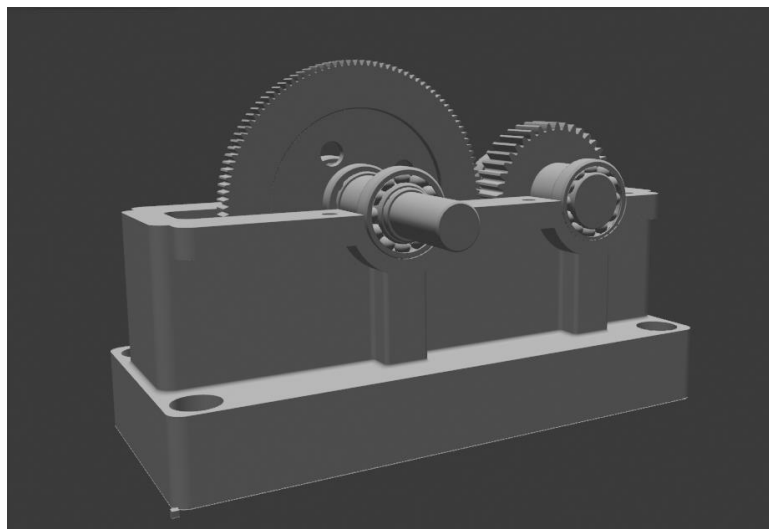


Рис.2.12 «Готова збірка»

Підрозді 2.2 «Моделювання механічного пристрою в динаміці за допомогою інструментального засобу тривимірного моделювання Blender»

2.2.1 Складання й візуалізація пристрою в середовищі Blender

Важливим етапом у виконанні курсового проекту є створення анімації складання редуктора, адже ми повинні не тільки вміти створювати 3D модель, а й розуміти з чого складається редуктор і як він збирається. Для цього необхідно відкрити програмний продукт Blender.

Для створення анімації необхідно імпортувати деталі в робочий простір програми, це можна зробити за допомогою команди Файл – Імпорт. Обираємо файл зі збереженим форматом .obj (Alias Mesh) у FreeCAD. Для правильного виконання завдання слідкуємо таким правилам:

- 1) Кожна частина деталей повинна знаходитись на окремих шарах;
- 2) Розміри зубців шестерень повинні чітко відповідати один одній і сходиться у розмірах.

З лівої сторони знаходяться основні елементи керування для складання редуктора (рис.2.13)

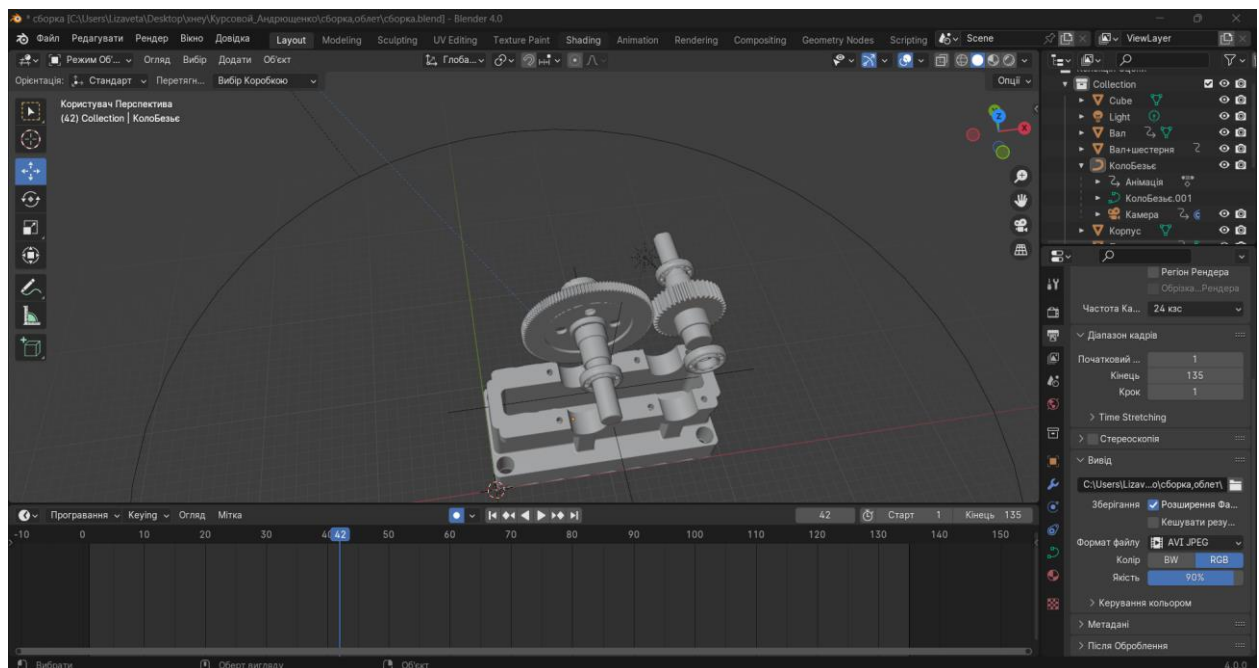


Рис.2.13 «Вікно програми»

2.2.2. Моделювання роботи пристрою в динаміці

Головною деталлю для роботи в програмі потрібно включити інструмент AutoKey, це для того щоб ключові кадри встановлювались автоматично після будь-якої дії. Далі переходимо до створення анімації. Спочатку з'єднуємо наш редуктор для розуміння кінцевого результату. Роботу починаємо з останнього кадру на часовій шкалі, оскільки редуктор вже має правильне розміщення деталей. Таким чином маємо послідовність виконання «розбору» редуктора. Тому наступним кроком перетягуємо повзунок на потрібну кількість кадрів та налаштовуємо параметри кута нахилу (рис.2.14). Відповідні вікна знаходяться у розділі інструментів у лівому кутку програми. Таким чином виходить анімація роботи шестерень в приладі. Звертаю увагу, що б був обраний потрібна деталь.

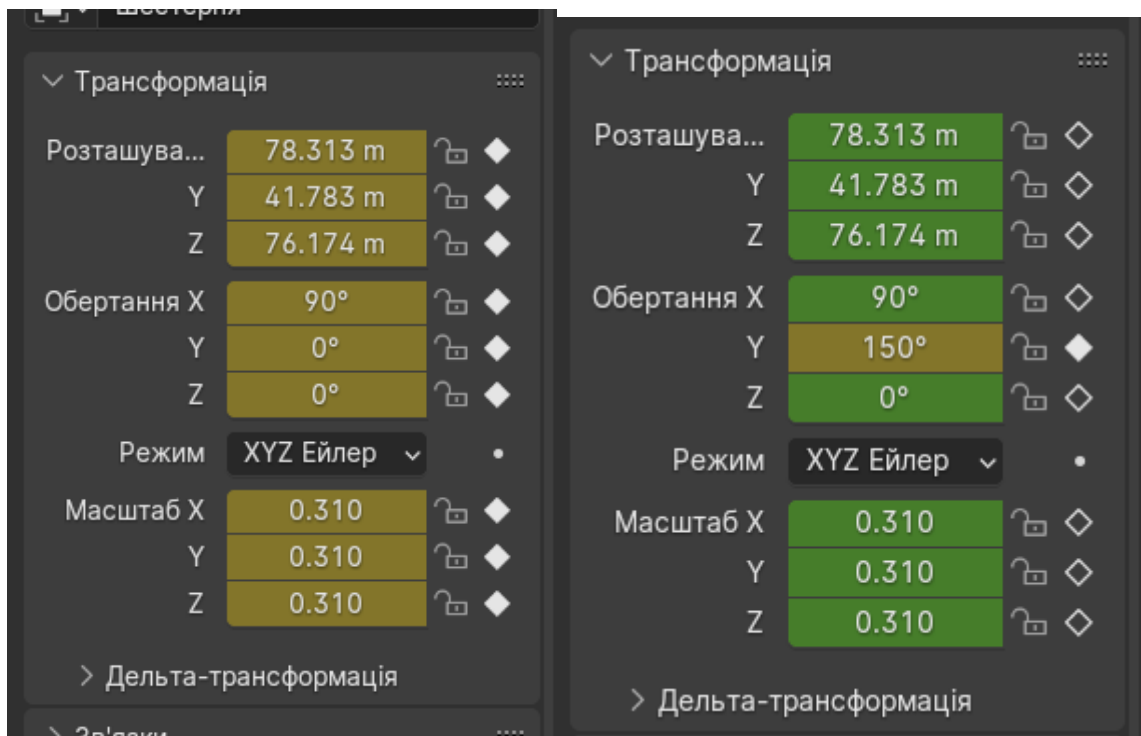


Рис.2.14 «Приклад налаштування на першому та останньому кадрі»

2.2.3 Обґрунтування параметрів рендерингу проекту й вибору програмного засобу для відтворення експортованої моделі

Для експортування результату роботи необхідно виконати наступні дії. В правому нижньому куту знаходимо налаштування OutPut (рис.2.15). У частині формату файлу обираємо AVI JPEG. За допомогою цього формату зберігається відео з покадровою анімацією. Після проробленого кроку у верхній частині робочого простору відкриваємо Рендирінг – Рендерінг анімацією.

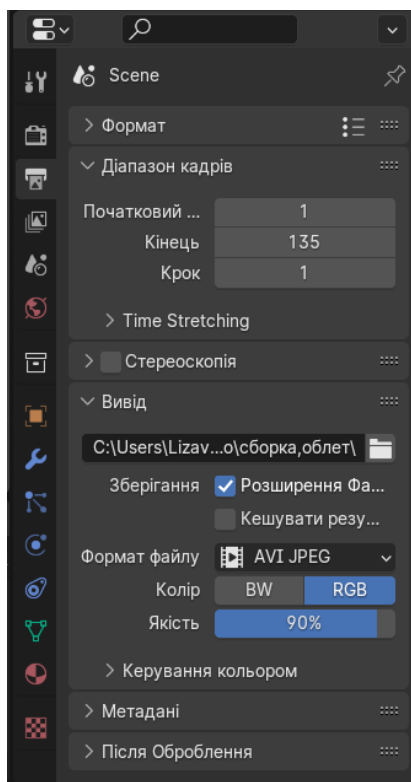


Рис.2.15 аркуш 23 «Вікно OutPut»

Підрозділ 2.3 «Створення мультимедійної презентації за допомоги засобів Adobe Animate»

Для створення мультимедійної презентації необхідно задати розмір робочого простору, в моєму випадку це 1000x600 пікселів. Переміщення між кадрами – метод переходу між сторінками.

Для компактного розміщення та навігації по кліпу прийняла рішення створила шари для фону, анімації, кнопок та коду. В презентації використовувала такі елементи: Movie Clip, відео, аудіо, кнопки, що були зроблені в інших програмах.

Презентація відкривається у вікні з головним меню, де розміщені чотири головні кнопки для переходу між розділами. Елементи створювала в програмі Photoshop, для створення таким чином кнопок.

На головному меню, кнопки робила анімованими при наведенні та натисканні. В символі кнопок змінювала налаштування (рис.2.16). При наведенні на кнопку вона змінюється у розмірі, тобто збільшується, при натисканні – зменшується зі звуком.



Рис.2.16 аркуш 24 «Головне меню»

Для того щоб анімація зупинялась на кадрах з кнопками я використовувала такий код: stop(); Для переходу між кадрами, використовувала такий код (рис.2.17).

```

1  stop();
2  btn1.addEventListener(MouseEvent.CLICK, fl_ClickToGoToAndStopAtFrame_17);
3
4  function fl_ClickToGoToAndStopAtFrame_17(event:MouseEvent):void
5  {
6      gotoAndStop(2);
7  }
8
9
10 btn2.addEventListener(MouseEvent.CLICK, fl_ClickToGoToAndStopAtFrame_18);
11
12 function fl_ClickToGoToAndStopAtFrame_18(event:MouseEvent):void
13 {
14     gotoAndStop(3);
15 }
16
17 btn3.addEventListener(MouseEvent.CLICK, fl_ClickToGoToAndStopAtFrame_19);
18
19 function fl_ClickToGoToAndStopAtFrame_19(event:MouseEvent):void
20 {
21     gotoAndStop(4);
22 }

```

Рис.2.17 «Код для переходу»

Для кожної кнопки використовувала код «stop();», для того щоб анімація зупинялась на потрібних кадрах. До певних кнопок додавала певні код. Останній рядок забезпечує перехід на необхідний нам кадр (рис.2.18):

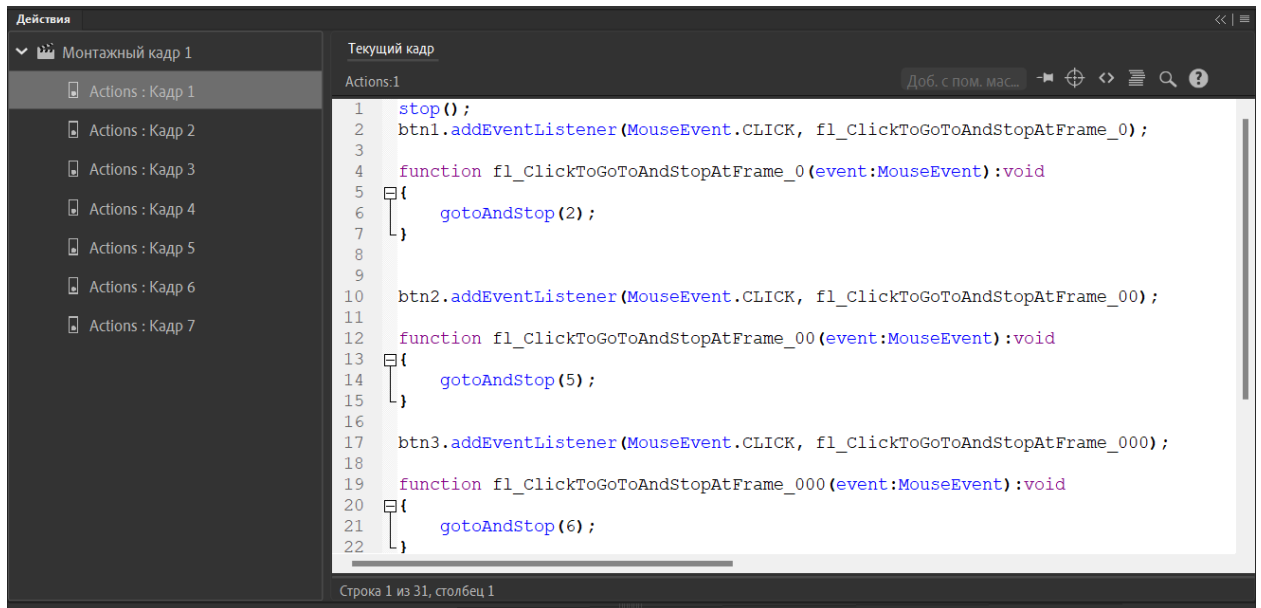


Рис.2.18 аркуш 26 «Код для перехода між кадрами»

При імпортуванні всіх відео в бібліотеку обрала формат .mp4, оскільки при використанні інших форматів виникає помилка. Виконувала команду Файл-Імпорт-Імпортувати в робочу бібліотеку. При імпортуванні обирала такі варіанти: завантажити відео з компонентом відтворення, імпортувати без обкладинки. Для зупинки та запуску відео додала відповідні кнопки та використала певні обробники подій (рис.2.19). Кінцевий результат виглядає наступним чином (2.20).

```

7  this.play_btn3.addEventListener(MouseEvent.CLICK, eventstartObl);
8  function eventstartObl (event:MouseEvent){
9      this.myVideoPlayer3.play();
10 }
11
12 this.stop_btn3.addEventListener(MouseEvent.CLICK, eventstopObl);
13 function eventstopObl (event:MouseEvent){
14     this.myVideoPlayer3.stop();
15 }

```

Рис.2.19 «Обробник подій для керування відео»



Збирання механізму у Blender

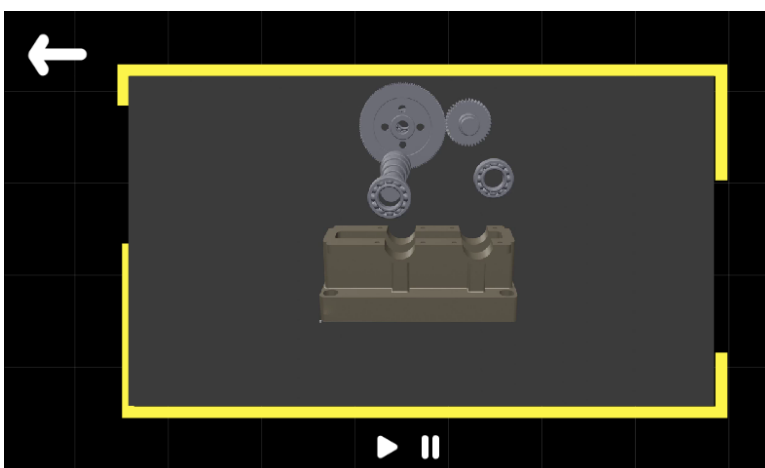
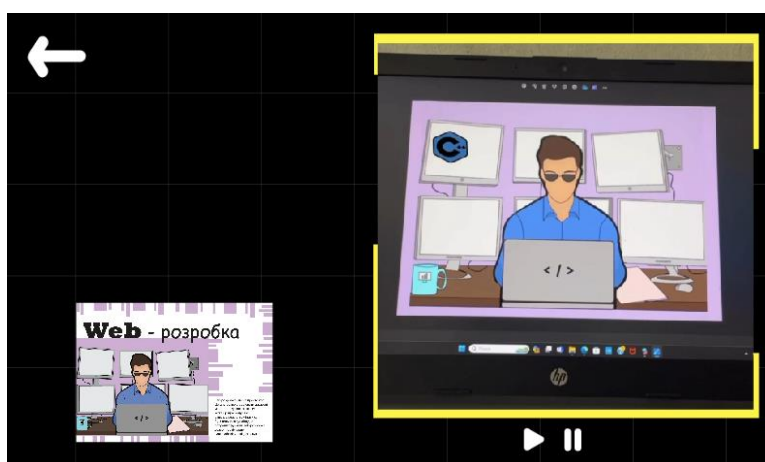


Рис.2.20 аркуш 28 «Кінцевий результат»

Для анімації зображень використовувала фрагмент ролику, де створювала анімацію збільшення (рис.2.21). Для того що б анімація починалась лише при наведенні використовувала символ-кнопку.

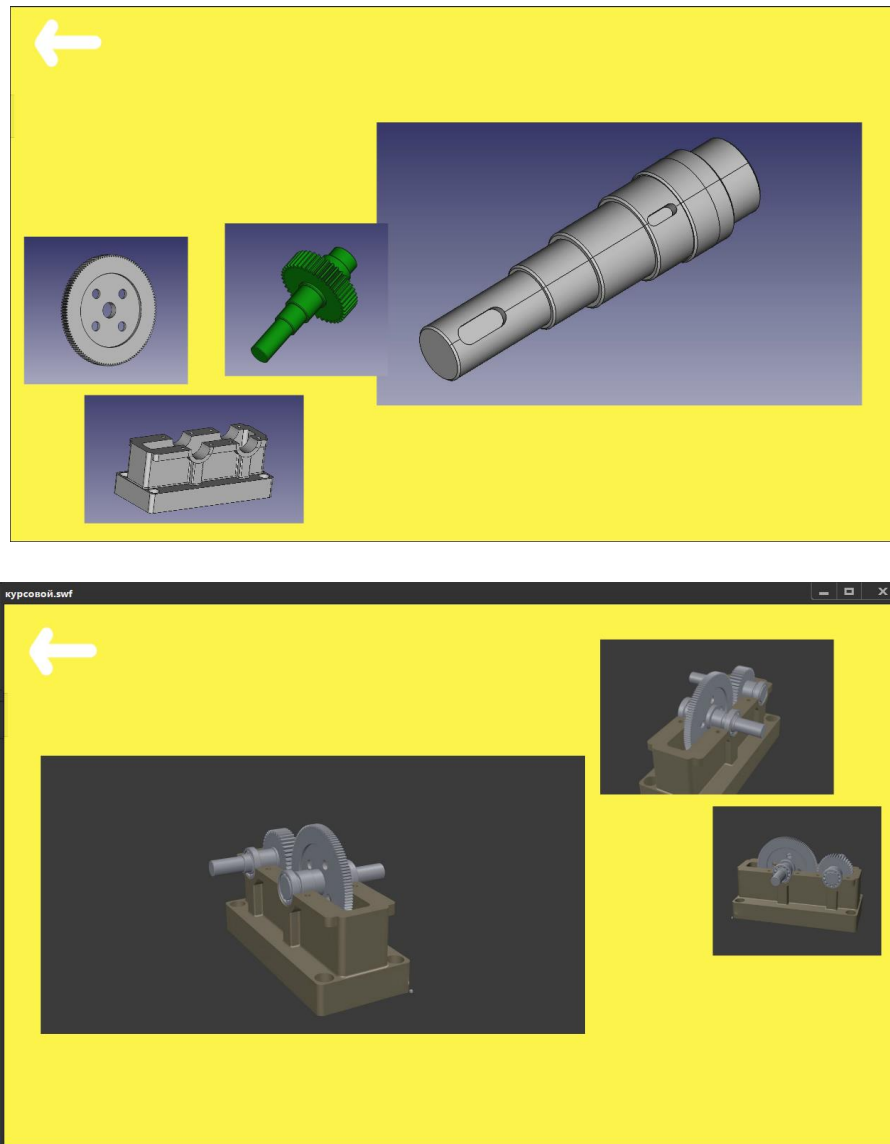


Рис.2.21 «Кадри з фотографіями»

Після повного створення презентації, її потрібно правильно зберегти у вигляді додатку формату .exe для коректної прив'язки до презентації на С#. Для цього виконуємо команду Файл - Параметри публікації і у відкритшомуся вікні ставимо галочку навпроти Проектор Win, тиснемо Опублікувати і зберігаємо файл (рис.2.22).

```
//кнопка результати
Ссылка 1
private void pictureBox5_Click(object sender, EventArgs e)
{
    System.Diagnostics.Process.Start(@"C:\Users\Lizaveta\Desktop\курсовой2.exe");
    this.Close();
}
```

Рис. 2.22 «Код для відкриття презентації»

Подивитись на всі сторінки презентації можна у Додатку 1.

Підрозділ 2.4. «Розробка дизайну електронного та друкованого складників комплексного курсового проекту»

2.4.1 Розробка дизайну програмної оболонки комплексного курсового проекту відповідно до обраного напрямку.

Першим кроком необхідно обрати стильовий напрямок для курсового проекту. Головна мета – впізнавання стилю на всіх сторінках програми.

Мною було обрано швейцарський стиль, який вважається одним з найвпливовіших та універсальних напрямків у сучасному дизайні, завдяки своїй простоті, чіткості та універсальності. Швейцарський дизайн, відзначається своєю простотою, чистотою і функціональністю. Він виник в Швейцарії в середині 20 століття і став важливим напрямком у графічному дизайні та архітектурі. У швейцарському стилі дуже важлива раціональність і лаконічність. Дизайнери цього напрямку відмовляються від зайвих деталей і використовують мінімальну кількість кольорів, шрифтів та графічних елементів. Можна виділити декілька головних характеристик:

- типографіка: шрифт повинен доносити повідомлення чітко та точно. Для легкого читання використовують зазвичай шрифти без засічок. Текст повинен мати ієрархію, тобто заголовок використовує жирний шрифт, а підзаголовок – основний.
- асиметричне розташування елементів: прийом використовується для створення цікавих і динамічних композицій, в яких елементи розташовані несиметрично відносно центральної осі та привертають увагу глядача і збільшують емоційну виразність дизайну.
- вільний простору: вільний простір дозволяє підтримувати баланс сторінки.
- акцент: використання нейтральних кольорів з акцентним кольором.

У сучасному світі швейцарський стиль популярний у графічному дизайні при створенні постерів, етикеток, друкованої реклами, верстки друкованих видань. У всіх ілюстраціях, виконаних у цьому стилі, насамперед в очі впадає основна інформація, а вже потім все інше. При цьому якщо виникає необхідність задіяти ілюстраторський ресурс, перед малюнками віддається перевага фотографіям. Вся робота була проведена у програмі Figma, де було додатково завантажений плагін Photorea для мінімального використання додаткових додатків.

Використала базові білий (FFFFFF) та чорний (000000) кольори. Для акцентного кольору був вибір між червоним та жовтим. Свій вибір зупинила на останньому, оскільки жовтий (FCF44A) передає більшу динамічність при цьому не перенавантажувати композицію.

Для заголовку було використано шрифт Impact. Він підходить до вимогів швейцарського стилю, бо не має засічок, підкреслює різкість та лаконічність. Для підзаголовку використовувала шрифт Imprima. Гарнітур має прості, чисті лінії та добре виражені форми, що робить його зручним для читання.

Потрібно проаналізувати моє втілення цього стилю. Запускаючи програму, відкривається головна сторінка (рис.2.13). На ній міститься головна інформація про курсовий проект та автора, кнопка для переходу на нове вікно програми та закрита додатку. Композицією на головному екрані відобразила асиметрію та динаміку. Це передається хаотичним розміщенням прямокутників та незвичним розміщенням тексту. Опис про автора розмістила за сіткою по лівому боку вирівняний текст. Притримувалась основних характеристик концепції. Основним місцем для розміщення кнопки для закриття програми – верхній правий кут. Продовження додатку розмістила у нижньому кутку. Це на інтуїтивному рівні зрозуміло та звично, оскільки більшість програм мають схоже розташування.



Рис.2.23 «Дизайн головної сторінки»

Після продовження, у вікні зображене меню з чотирьох кнопок (рис.2.16). Останні зображені з текстом та геометричної фігури. Відображають динаміку, при цьому мають вільний простір. У правому верхньому куту залишається кнопка про закриття програми, в протилежному, лівому куті – кнопка повернення в минуле вікно програми.

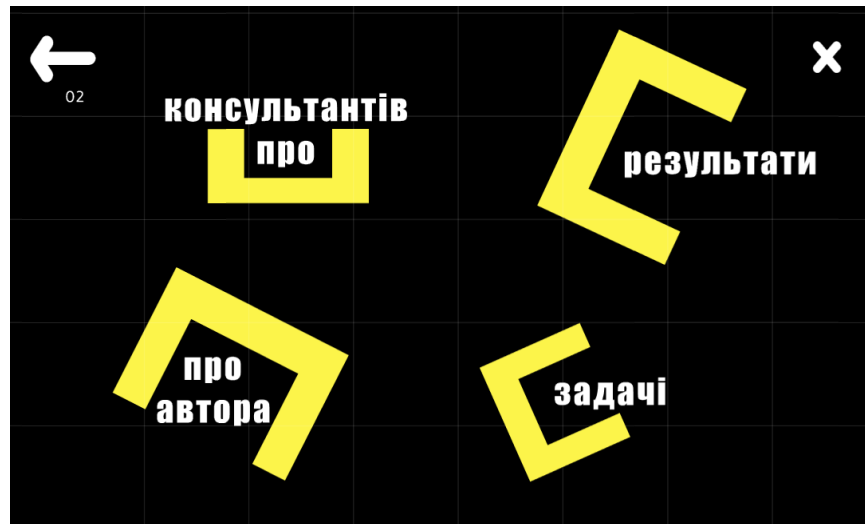


Рис.2.24 аркуш 30 «Вікно меню»

Перша кнопка «Про автора» змінює вікно на інформацію про мене та моєї фотографією (рис.2.17). Для вирізання зображення використовувала плагін з Figma. Додала тінь до всіх зображень, аби поєднати сторінку зі стилем. Градієнт створений для м'якого переходу між фоном та зображенням. Назва сторінки та оздоблення виконані в такому ж стилі, як головна. Кнопки залишила в такому ж розміщенні.



Рис.2.25 «Сторінка про автора»

Повернувшись на сторінку з меню, можна обрати кнопку «Про консультантів», де зібрана основна інформація про викладачів (рис.2.18). Зліва розміщена панель кнопок, де при виборі змінюється інформація про консультанта. Розміщення кнопок навігації залишилось незмінним.



Рис.2.26 аркуш 31 «Про консультантів»

Кнопка «Задачі» відкриває вікно з цілями кожної дисципліни (рис.2.19). При обиранні дисципліни відкривається інформація навпроти кожної кнопки.



Рис.2.27 «Задачі»

Кнопка «Результат» закриває презентацію розроблену за допомогою C#, та відкривається презентація в Adobe Animate. Кнопка закриття програми відкриває вікно з підтвердженням про вихід.

Після втілення швейцарського стилю в додаток, можна сказати, що дизайн відповідає вимогам обраного стилю. Під час створення презентацій керувалась деякими принципами композиції:

- принцип рівноваги (стан композиції, при якій елементи, що входять до її складу, збалансовані відносно друг другу);
- принцип цілісності (дотримання єдиних форм та елементів дизайну, повторення кольорів);
- принцип підпорядкованості елементів;
- принцип доцільності (передача змісту, ідеї та концепції дизайну).

2.4.2 Розробка дизайну друкованого складника комплексного курсового проекту.

Мета дизайну – створення постеру та додавання доповненої реальності за допомогою Meta Spark Studio. Основними вимогами до плакату є: формат – A2; роздільна здатність – 300 dpi; шрифт основного тексту ≈ 24 -26 пт. Розробка плакату в форматі RGB і CMYK. Роботу з постером буду продовжувати зі зображенням лабораторної роботи 1 (рис.2.20).



Рис.2.28 «Створений постер»

Постер створений на тему «Веб-розробник». Основні ознаки швейцарського стилю в дизайні поліграфічної продукції: чітка модульна сітка; мінімалізм; гротескні шрифти; модерністська виключка; модерністський абзац (нульовий відступ або "сліпий" рядок); прості геометричні фігури; фотознімок як ілюстрація; асиметрія; динаміка; вільний простір; чітка візуальна ієрархія.

Оскільки постер має головну ілюстрацію, то до неї розробимо доповнену реальність. Ілюстрація була створена на лабораторній роботі №1 з дисципліни «Основи композиції та дизайну». Створити ефект доповненої реальності допоможе Meta Spark Studio. Ціль кінцевого результату – розробити ефект, який буде з’являтися на екрані при наведенні об’єктиву камери на зображення.

Анімація ілюстрації містить 12 кадрів у форматі - png; кольорова модель – RGB (рис.2.21).

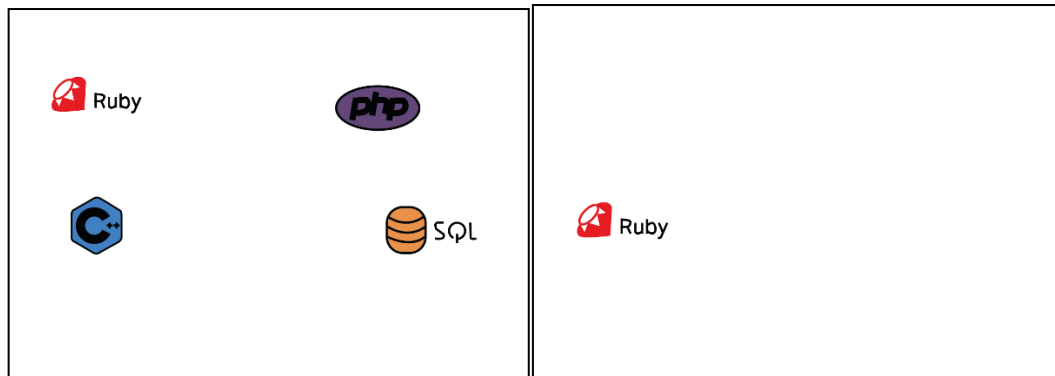


Рис.2.29 «Кадри анімації»

Для перевірки роботи або використання маски надаю посилання:

<https://www.instagram.com/ar/313410875143770/?ch=OWZlMmEyZDNiOTVlMmMzNTcwMWYzZGNjYjI3MWZjNWl%3D>

Підрозділ 2.5. «Розробка мультимедійної презентації з використанням мови програмування C#»

2.5.1 Вступ

Завдання, які розв’язуються в проєкті: розробка програмної оболонки. Вона дозволить об’єднати усі раніше створені компоненти в одне єдине.

Потрібно:

- закріпити навички практичної роботи (що отримані в процесі раніше вивченого курсу «Основи об’єктно-орієнтованого програмування») з керуючими елементами Windows Form середовища Visual Studio .NET;
- навчитися застосовувати програмні засоби підтримки багатозадачності в багатомодульних C# - проєктах; 40
- закріпити практичні навички розробки й налагодження багатомодульних графічних додатків. Як правило, технологія проєктування будь-якої мультимедійної презентації містить у собі такі етапи:
- розробка організаційних і технологічних блок-схем сценарію мультимедійної презентації;

- розробка функціональних описів елементів організаційних і технологічних блок-схем;
- розробка програмного забезпечення, яке підтримує реалізацію сценарію, що пропонується в заданому середовищі виконання (у нашому випадку ОС Windows).

2.5.2. Розробка організаційної блок-схеми сценарію мультимедійної презентації

Організаційна блок-схема відображає структуру проекту, це допомагає логічно впорядкувати розділи. Перед створенням презентації необхідно створити схему (схема 1)

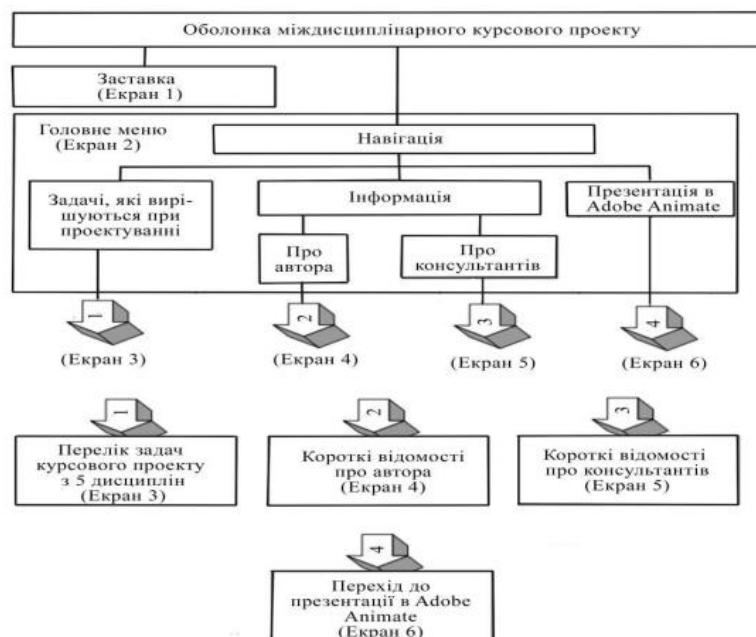


Схема 1. Організаційна блок-схема

2.5.3. Розробка технологічної блок-схеми сценарію мультимедійної презентації

Технологічна блок-схема (схема 2) докладно розкриває інтерактивні можливості проекту. Вона використовується як відправна точка для функціонального опису інтерфейсу (екранних форм), які потім повинні бути реалізовані об'єктно-орієнтованими засобами мови C#

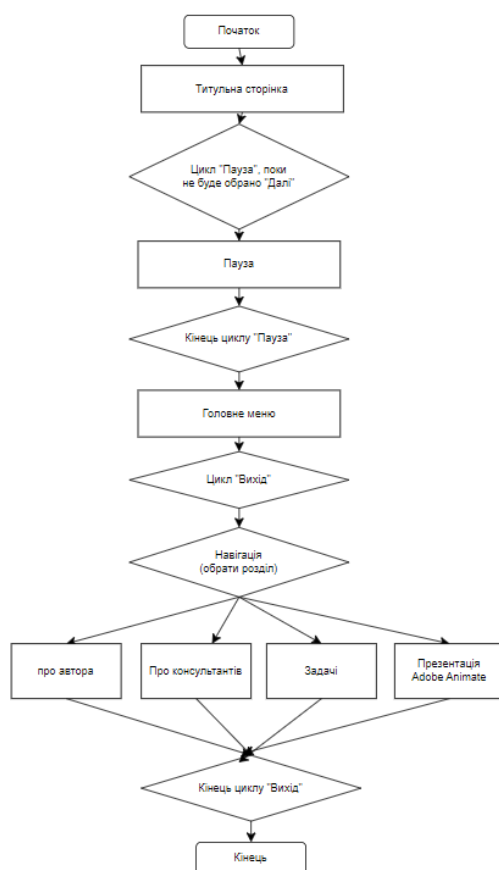


Схема 2. Технологічна карта

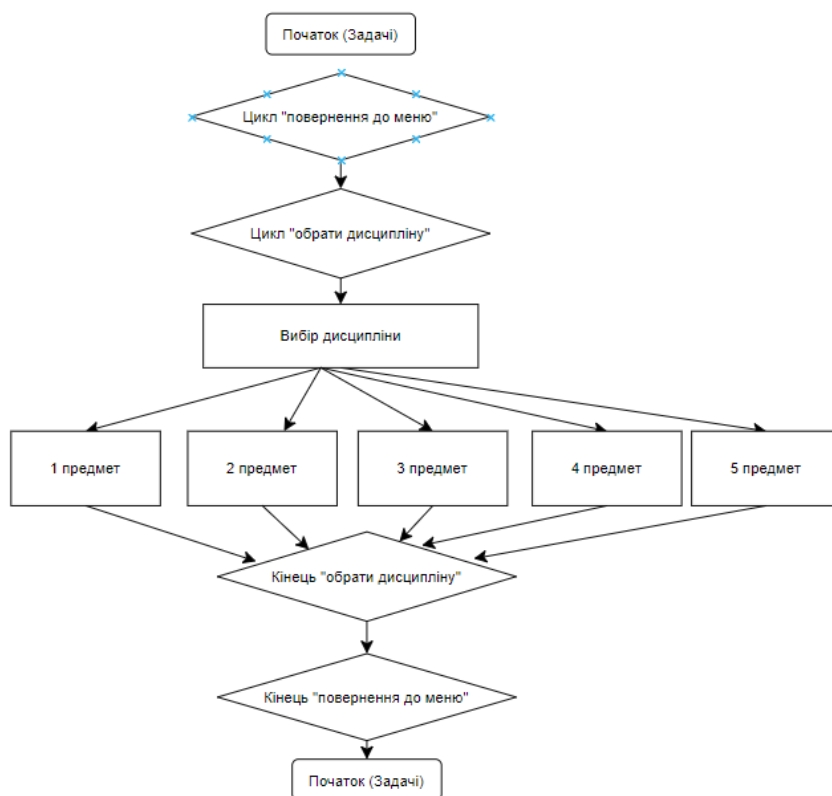


Схема 2.1 Деталізація технологічної блок-схеми

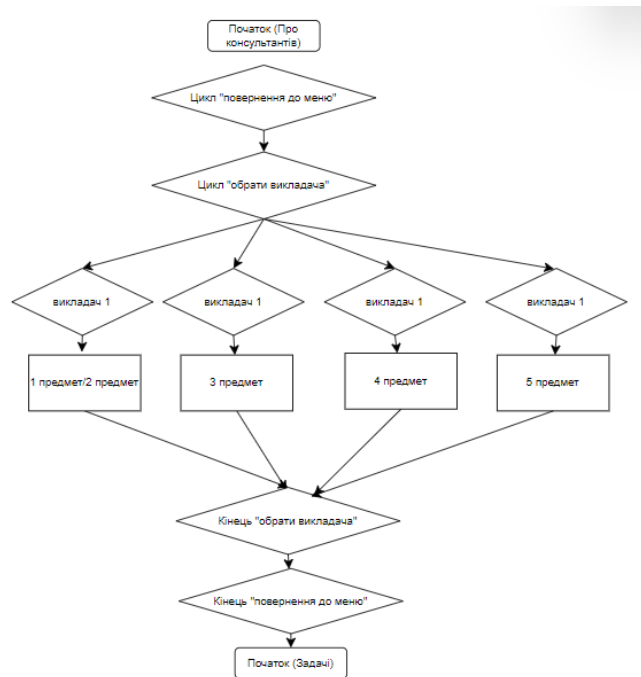


Схема 2.2 Деталізація технологічної блок-схеми

2.5.4. Розробка функціональних описів елементів організаційних і технологічних блок-схем

Перша форма є титульною сторінкою і з неї можна перейти в головне меню, або закрити презентацію. На рис. 2.30. наведений ескіз заставки (екран 1) комплексного курсового проекту.

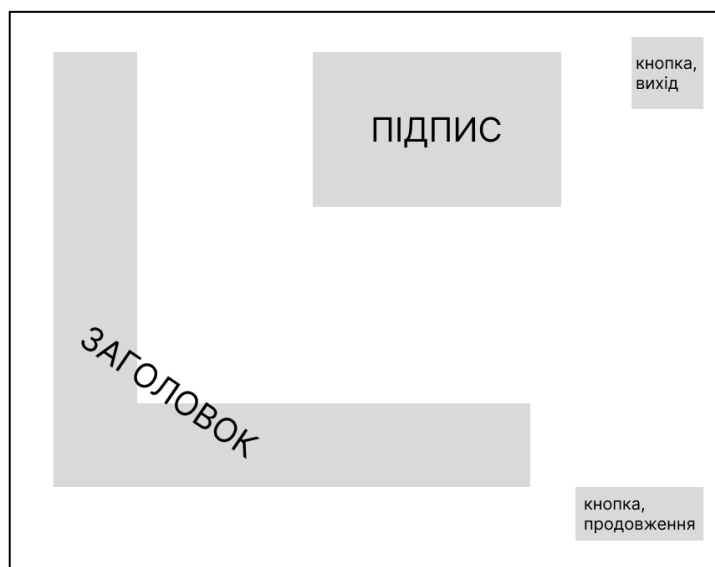


Рис. 2.30 «Ескіз заставки»

На рис. 2.31. наведений ескіз головного меню (екран 2)



Рис. 2.32 аркуш 37 «Ескіз головного меню»

На рис. 2.33. наведений ескіз сторінки про автора (екран 3)



Рис. 2.33 «Ескіз сторінки про автора»

На рис. 2.34. наведений ескіз сторінки про консультантів (екран 4)



Рис. 2.34 «Ескіз сторінки про консультантів»

На рис. 2.35. наведений ескіз сторінки про задачі проекту



Рис. 2.35 аркуш 38 «Ескіз сторінки про задачі»

2.5.5. Розробка програмного забезпечення мультимедійної презентації

Кроки створювання презентації:

- 1) Розробка екранних форм відповідно до раніше спроектованих (див. попередній розділ) ескізів сторінок;
- 2) Додавання в проект оброблювачів подій (див. Додаток 2)

2.5.6. Складання й публікація комплексного курсового проекту

Після підготовки форм і включення до складу відповідних класів необхідних оброблювачів подій здійснюємо складання проекту. Всі файли мультимедійних компонентів перебувають у тій же папці (... \bin\Debug\...), де формується кінцевий файл проекту.

2.5.7. Аналіз роботи мультимедійної презентації

У Додатку 3 наведені всі екранні форми й показана їхня відповідність ескізам відповідних сторінок сценарію, а також всі коди.

ВИСНОВОК

При виконанні курсового проекту були закріплені знання з різних дисциплін.

По дисципліні «Технічна механіка» я навчилася розраховувати параметри механічного пристрою з наступним візуальним моделюванням в статиці і динаміці для використання в мультимедійної презентації.

По дисципліні «Інженерна та комп'ютерна графіка» я візуально змодельовала механічний пристрій редуктор в статиці та динаміці (обліт) за допомогою програмного засобу FreeCAD.

Розділ «Основи композиції та дизайну» допоміг мені розробити дизайн мультимедійної презентації та буклету, які я створила у Photoshop.

По дисципліні «Комп'ютерна анімація» було створено збірку редуктора у Blender та анімовану презентацію у Adobe Animate.

З дисципліни «Програмування засобів мультимедіа» я розробила оболонку мультимедійної презентації, за допомогою якої продемонструвала всі створені раніше частини курсового проекту.

В результаті виконання усіх частин міждисциплінарного курсового проекту була отримана папка з усіма файлами, які містять в собі презентації, фотографії, відео. Створення всіх елементів проекту закріпило наші знання з певних предметів та дало змогу на все життя запам'ятати процес їх реалізації.

ДЖЕРЕЛА:

1. Методичні рекомендації до виконання междисциплінарного курсового проекту для студентів 2-го курсу спеціальності 186 "Видавництво та поліграфія" першого (бакалаврського) рівня всіх форм навчання / Укл. О.І. Пушкар, В.В. Браткевич, О.С. Євсєєв, В.Є. Климнюк, Л.В. Потрашкова, – Харків: Вид. ХНЕУ, 2019. – 64 с. (Укр. мов.)
2. Швейцарський стиль: точність і функціональність до найдрібніших деталей. URL: <https://welovebrands.com.ua/ua/blogs/style-swiss/> (дата звернення: 24.04.2024)
3. Швейцарский стиль дизайн). URL: [https://ru.wikipedia.org/wiki/Швейцарский_стиль_\(дизайн\)#Общая_характеристика](https://ru.wikipedia.org/wiki/Швейцарский_стиль_(дизайн)#Общая_характеристика) (дата звернення: 18.04.2024)
4. <https://www.arthuss.com.ua/books-blog/kompozic> (10.05.2024)
5. Підтема2_1 «Принципи та засоби к»
6. Підтема 2_1. «Принципи та засоби композиції
- 7.

ДОДАТКИ

Додаток 1.

Лістинінг Екрану 1

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

namespace курсовой
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void Form1_Load(object sender, EventArgs e)
        {
        }

        private void pictureBox3_Click(object sender, EventArgs e)
        {
            DialogResult result = MessageBox.Show("Ви впевнені, що хочете вийти?",
"Підтвердження виходу", MessageBoxButtons.YesNo, MessageBoxIcon.Question);
            if (result == DialogResult.Yes)
            {
                Application.Exit(); // Вихід з програми
            }
        }

        private void pictureBox2_Click(object sender, EventArgs e)
        {
            // Створіть новий екземпляр форми, яку ви хочете відкрити
            Form2 form2 = new Form2();

            // Прикажіть нову форму показати вікно
            form2.Show();

            // Закрийте поточну форму
            this.Hide();
        }
    }
}

```

Лістинінг Екрану 2

```

using Microsoft.VisualBasic.ApplicationServices;
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Diagnostics;
using System.Drawing;
using System.Linq;

```

```

using System.Reflection.PortableExecutable;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace курсовой
{
    public partial class Form2 : Form
    {
        public Form2()
        {
            InitializeComponent();
        }

        private void Form2_Load(object sender, EventArgs e)
        {
        }

        private void pictureBox2_Click(object sender, EventArgs e)
        {
            // Створіть новий екземпляр форми, яку ви хочете відкрити
            Form1 form1 = new Form1();

            // Прикажіть нову форму показати вікно
            form1.Show();

            // Закрийте поточну форму
            this.Close();
        }

        private void pictureBox3_Click(object sender, EventArgs e)
        {
            DialogResult result = MessageBox.Show("Ви впевнені, що хочете вийти?",
"Підтвердження виходу", MessageBoxButtons.YesNo, MessageBoxIcon.Question);
            if (result == DialogResult.Yes)
            {
                Application.Exit(); // Вихід з програми
            }
        }

        //кнопка результати
        private void pictureBox5_Click(object sender, EventArgs e)
        {
        }

        System.Diagnostics.Process.Start(@"C:\Users\Lizaveta\Desktop\курсовой2.exe");
        this.Close();
    }

    private void pictureBox6_Click(object sender, EventArgs e)
    {
        // Створіть новий екземпляр форми, яку ви хочете відкрити
        Form3 form3 = new Form3();

        // Прикажіть нову форму показати вікно
        form3.Show();

        // Закрийте поточну форму
        this.Close();
    }

    private void pictureBox4_Click(object sender, EventArgs e)
    {
        // Створіть новий екземпляр форми, яку ви хочете відкрити
        Form4 form4 = new Form4();
    }
}

```

```

        // Прикажіть нову форму показати вікно
        form4.Show();

        // Закрийте поточну форму
        this.Close();
    }

    private void pictureBox7_Click(object sender, EventArgs e)
    {
        // Створіть новий екземпляр форми, яку ви хочете відкрити
        Form5 form5 = new Form5();

        // Прикажіть нову форму показати вікно
        form5.Show();

        // Закрийте поточну форму
        this.Close();
    }
}
}

```

Лістинінг Екрану 3

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace курсовой
{
    public partial class Form3 : Form
    {
        public Form3()
        {
            InitializeComponent();
        }

        private void pictureBox2_Click(object sender, EventArgs e)
        {
        }

        private void pictureBox3_Click(object sender, EventArgs e)
        {
            // Створіть новий екземпляр форми, яку ви хочете відкрити
            Form2 form2 = new Form2();

            // Прикажіть нову форму показати вікно
            form2.Show();

            // Закрийте поточну форму
            this.Close();
        }

        private void pictureBox2_Click_1(object sender, EventArgs e)
        {
            DialogResult result = MessageBox.Show("Ви впевнені, що хочете вийти?",
            "Підтвердження виходу", MessageBoxButtons.YesNo, MessageBoxIcon.Question);
            if (result == DialogResult.Yes)
            {

```

```

        Application.Exit(); // Вихід з програми
    }
}
}
}

```

Лістинінг Екрану 4

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace курсовой
{
    public partial class Form4 : Form
    {
        public Form4()
        {
            InitializeComponent();

            private void radioButton1_CheckedChanged(object sender, EventArgs e)
            {
                pictureBox4.Visible = true;
                pictureBox6.Visible = false;
                pictureBox7.Visible = false;
                pictureBox8.Visible = false;
            }

            private void radioButton2_CheckedChanged(object sender, EventArgs e)
            {
                pictureBox4.Visible = false;
                pictureBox6.Visible = true;
                pictureBox7.Visible = false;
                pictureBox8.Visible = false;
            }

            private void radioButton3_CheckedChanged(object sender, EventArgs e)
            {
                pictureBox4.Visible = false;
                pictureBox6.Visible = false;
                pictureBox7.Visible = true;
                pictureBox8.Visible = false;
            }

            private void radioButton4_CheckedChanged(object sender, EventArgs e)
            {
                pictureBox4.Visible = false;
                pictureBox6.Visible = false;
                pictureBox7.Visible = false;
                pictureBox8.Visible = true;
            }

            private void pictureBox3_Click(object sender, EventArgs e)
            {
                // Створіть новий екземпляр форми, яку ви хочете відкрити
                Form2 form2 = new Form2();

                // Прикажіть нову форму показати вікно
            }
        }
    }
}

```

```

        form2.Show();

        // Закрийте поточну форму
        this.Close();
    }

    private void pictureBox2_Click(object sender, EventArgs e)
    {
        DialogResult result = MessageBox.Show("Ви впевнені, що хочете вийти?",
        "Підтвердження виходу", MessageBoxButtons.YesNo, MessageBoxIcon.Question);
        if (result == DialogResult.Yes)
        {
            Application.Exit(); // Вихід з програми
        }
    }
}

```

Лістинінг Екрану 5

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace курсовой
{
    public partial class Form5 : Form
    {
        public Form5()
        {
            InitializeComponent();

            private void radioButton1_CheckedChanged(object sender, EventArgs e)
            {
                pictureBox5.Visible = true;
                pictureBox6.Visible = false;
                pictureBox7.Visible = false;
                pictureBox8.Visible = false;
                pictureBox9.Visible = false;
            }

            private void radioButton2_CheckedChanged(object sender, EventArgs e)
            {
                pictureBox5.Visible = false;
                pictureBox6.Visible = true;
                pictureBox7.Visible = false;
                pictureBox8.Visible = false;
                pictureBox9.Visible = false;
            }

            private void radioButton3_CheckedChanged(object sender, EventArgs e)
            {
                pictureBox5.Visible = false;
                pictureBox6.Visible = false;
                pictureBox7.Visible = true;
                pictureBox8.Visible = false;
                pictureBox9.Visible = false;
            }
        }
    }
}

```

```

private void radioButton4_CheckedChanged(object sender, EventArgs e)
{
    pictureBox5.Visible = false;
    pictureBox6.Visible = false;
    pictureBox7.Visible = false;
    pictureBox8.Visible = true;
    pictureBox9.Visible = false;
}

private void radioButton5_CheckedChanged(object sender, EventArgs e)
{
    pictureBox5.Visible = false;
    pictureBox6.Visible = false;
    pictureBox7.Visible = false;
    pictureBox8.Visible = false;
    pictureBox9.Visible = true;
}

private void pictureBox3_Click(object sender, EventArgs e)
{
    // Створіть новий екземпляр форми, яку ви хочете відкрити
    Form2 form2 = new Form2();

    // Прикажіть нову форму показати вікно
    form2.Show();

    // Закрийте поточну форму
    this.Close();
}

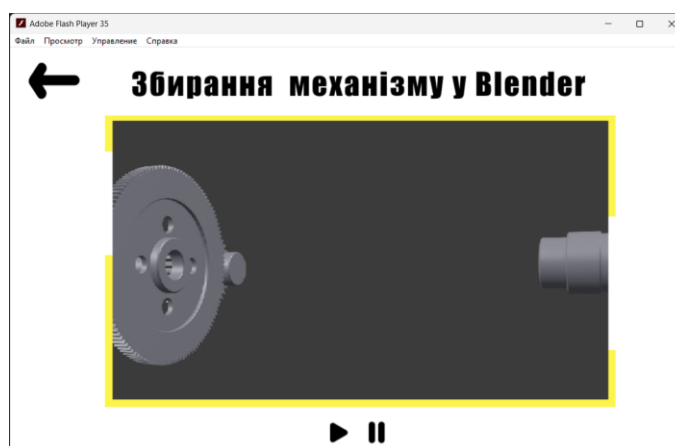
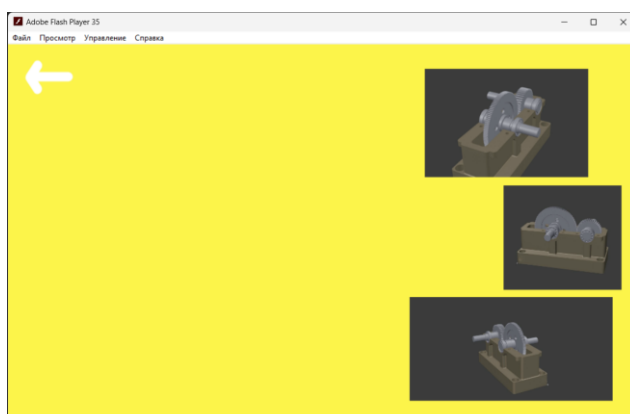
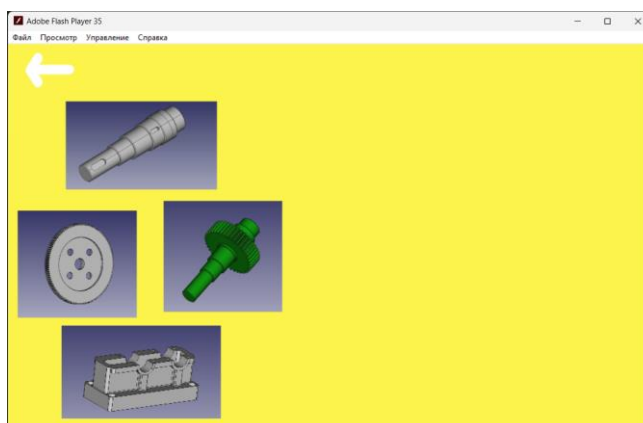
private void pictureBox2_Click(object sender, EventArgs e)
{
    DialogResult result = MessageBox.Show("Ви впевнені, що хочете вийти?",
    "Підтвердження виходу", MessageBoxButtons.YesNo, MessageBoxIcon.Question);
    if (result == DialogResult.Yes)
    {
        Application.Exit(); // Вихід з програми
    }
}
}
}

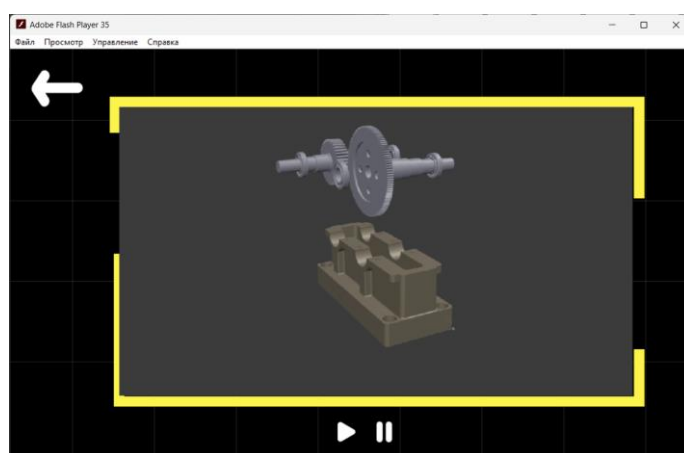
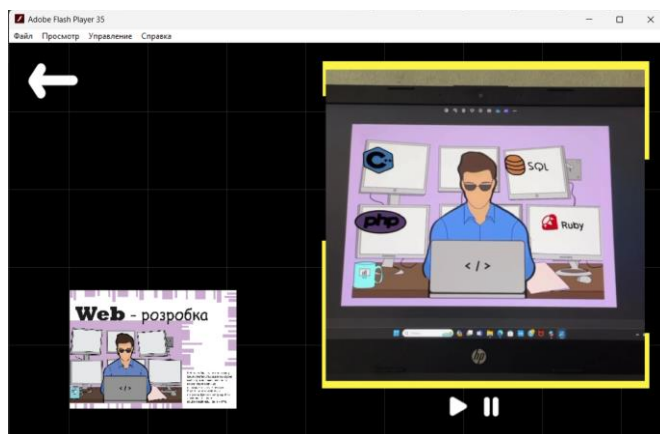
```

Додаток 2.

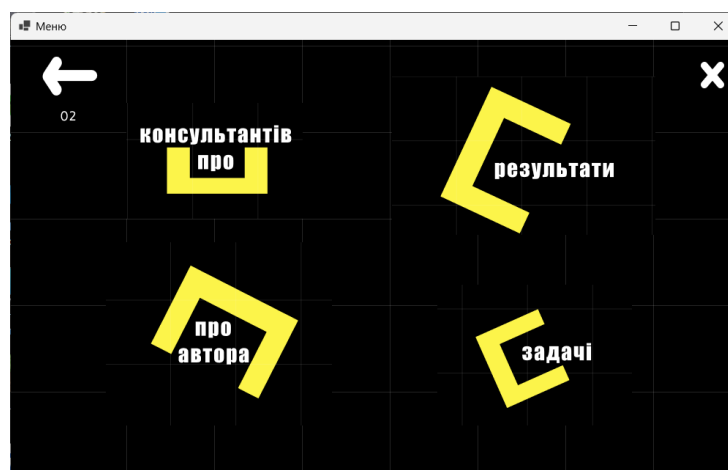
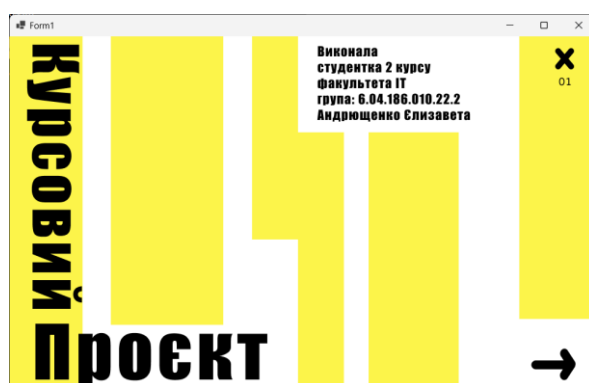
Відеокадри презентації в Adobe Animate







Додаток 3.



←

04

Про консультантів

×

Євсєєв О.С.

Гордєєв А.С.

Потрашкова Л.В.

Кобзєв І.В.



Статус
Кандидат економічних наук, доцент

Нафедра
Комп'ютерних систем і технологій

Наукові інтереси
Системний аналіз, системи підтримки прийняття рішень, штучний інтелект, управління розподіленими командами, розробка комп'ютерних ігор

Навчальні дисципліни
Комп'ютерна анімація та віртуальна реальність

Євсєєв
Олексій Сергійович

←

03

Про автора

×



Статус
Студентка 2 курсу,
групи 6.04.186.010.22.2

Нафедра
Комп'ютерних систем і технологій

Опис
Я весела та енергійна особистість з великим захопленням до мистецтва та творчості. Я люблю вивчати нові речі та постійно розвиватися.

Про автора

←

05

Задачі

×

Технічна механіка

Інформатика та комп'ютерна графіка

Програмування засобів мультимедіа

Комп'ютерна анімація

Основи композиції та дизайну

Розрахунок параметрів механічного пристрою одноступінчастого циліндричного редуктора на підставі виданих індивідуальних даних;
Створення схеми побудови редуктора;
Опис алгоритму збирання та розбирання редуктора.

Задачі