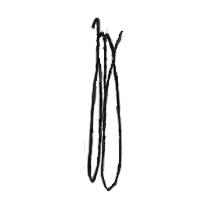
|  |
| --- |
| **Форма № Н-6.01**  Міністерство освіти і науки України  Вінницький національний технічний університет  Кафедра програмного забезпечення |
| КУРСОВА РОБОТА з дисципліни ”Комп’ютерна графіка ” на тему: Розробка статичних та динамічних зображень для вебсайту ”Зельман Ваксман” |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | |  | Студента 3 курсу 6ПІ-22б групи  спеціальності 121 - “Інженерія програмного  забезпечення” В.І.Сороколіт   Керівник: зав. кафедри ПЗ, д.т.н., О.Н. Романюк  Кількість балів: \_\_ Оцінка: ECTS \_\_\_\_ | | | Члени комісії | романюк_о_н\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Романюк О. Н. | | \_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Завальнюк Є. К. | |  |  |     м. Вінниця - 2024 рік |

Міністерство освіти і науки України

Вінницький національний технічний університет

Факультет інформаційних технологій та комп’ютерної інженерії

 ЗАТВЕРДЖУЮ

Зав. кафедри ПЗ, проф., д.т.н.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ О.Н. Романюк

*(підпис)*

Затверджено на засіданні каф. ПЗ протокол №\_1 від ”12” вересня  2024 р.

ІНДИВІДУАЛЬНЕ ЗАВДАННЯ

на курсову роботу з дисципліни ”Комп’ютерна графіка”

студенту групи 6ПІ-22б

Сороколіту Владиславу Ігоровичу

Тема роботи: Розробка статичних та динамічних зображень для вебсайту ”Зельман Ваксман”

Вихідні дані:

1) мова сайту – українська;

2) графічне забезпечення сайту виконати із використанням сучасних технологій (css3. Html та ін.);

3) анімаційне забезпечення сайту виконати із використанням відповідних засобів анімації (Inkscape, Visual Studio Code, Inteliji IDEA та ін.);

4) обгрунтувати вибір колірної гами;

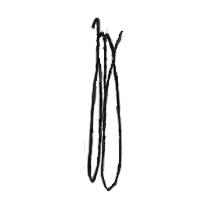
5) обгрунтувати розробку логотипу сайту;

6) багатовимірний аналіз на всіх стадіях розробки;

7) обгрунтувати стиль викладення матеріалу пояснюувальної записки.

Основні складові пояснювальної записки:

* Титульний лист
* Індивідуальне завдання
* Анотація
* Зміст
* Вступ
* 1 Багатовимірний аналіз розв’язку основної задачі
* 2 Вибір кольорової гами
* 3 Розробка логотипу сайту
* 4 Розробка статичних зображень
* 5 Розробка динамічних зображень
* 6 Оцінка обсягів зображень
* 7 Тестування роботи сайту
* 8 Розробка програмного застосунку
* Висновки
* Список використаних джерел

Назва програмного застосунку : «Неітеративна сферично-кутова інтерполяція векторів вздовж рядка растеризації».

Дата видачі «12» вересня 2024 р. Керівник \_\_\_\_ Романюк О. Н.



Завдання отримав \_\_\_ Сороколіт В. І.

*)*

АНОТАЦІЯ

В даній курсовій роботі основним завданням була розробка статичних та динамічних зображень. Створено вебсайт із докладним описом предметної області. Також був проведений аналіз актуальності та важливості виконаної роботи.

Під час виконання курсової роботи було використано програмне забезпечення із відкритим вихідним кодом – Inkscape. Для розробки частин динамічного зображення також було використано програмний продукт Inkscape. Усі графічні матеріали та інформаційне навантаження відповідають заданій тематиці.

Додатково, в рамках окремої частини курсової роботи було розроблено програмний застосунок для реалізації неітеративної сферично-кутової інтерполяції векторів вздовж рядка растеризації за відповідним алгоритмом. Реалізація була виконанна із застосуванням мови програмування Java в інтегрованому середовищі розробки Inteliji IDEA

|  |  |
| --- | --- |
| ЗМІСТ  ВСТУП | 5 |
| 1 БАГАТОВИМІРНИЙ АНАЛІЗ РОЗВ’ЯЗКУ ОСНОВНОЇ ЗАДАЧІ | 7 |
| 2 ВИБІР КОЛЬОРОВОЇ ГАМИ | 10 |
| 3 РОЗРОБКА ЛОГОТИПУ САЙТУ | 13 |
| 4 РОЗРОБКА СТАТИЧНИХ ЗОБРАЖЕНЬ | 15 |
| 5 РОЗРОБКА ДИНАМІЧНИХ ЗОБРАЖЕНЬ | 22 |
| 6 ОЦІНКА ОБСЯГІВ ЗОБРАЖЕНЬ | 27 |
| 7 ТЕСТУВАННЯ РОБОТИ САЙТУ | 29 |
| 8 РОЗРОБКА ПРОГРАМНОГО ЗАСТОСУНКУ | 32 |
| ВИСНОВКИ | 36 |
| СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ | 37 |
| Додаток А. СТАТИЧНІ ЗОБРАЖЕННЯ | 38 |
| Додаток Б. ДИНАМІЧНІ ЗОБРАЖЕННЯ | 45 |
| Додаток В. ЛІСТИНГ ВЕБСТОРІНКИ INDEX.HTML | 48 |
| Додаток Г. ЛІСТИНГ ПРОГРАМНОГО ЗАСТОСУНКУ | 53 |

ВСТУП

Комп’ютерна графіка є однією із найбільш важливих складових сучасної IT індустрії. Вона охоплює та впливає на дуже велику кількість аспектів сучасного людського життя, адже із розвитком сучасних технологій у людства з’явилося більше можливостей для використання різних електронно обчислювальних машин різного характеру та ступеня важкості.

Однією із таких сфер де комп’ютерна графіка зіграла ключову роль, є сфера розваг. Вона є однією із найбільш перспективних напрямків IT індустрії насамперед через стрімкий розвиток комп’ютерної графіки та її можливостей. Ця сфера перестала бути сфокусованою та знайшла способи використовувати новітні досягнення не тільки для створення ігор, а й для створення високореалістичних анімацій та спецефектів, інтерактивних навчальних програм та посібників направлених не тільки на дітей, а й на дорослих людей для навчання дуже специфічних навичок які можуть бути небезпечними у реальному житті. Такими можна вважати програми для навчання лікарів та пілотів де одна невелика помилка вчинена недосвідченою людиною може коштувати їй, або її пацієнту життя та здоров’я.

Комп’ютерна графіка також може бути застосована на практиці в реальній роботі деяких професій. Наприклад інженери та архітектори можуть створювати точні тривимірні моделі їх проектів які будуть автоматично змінюватись відповідно до заданих ними параметрів.

Наукова сфера також отримала користь від розвитку комп’ютерної графіки. З її допомогою вчені та програмісти змогли зі значною точністю отримати моделі небесних тіл та об’єктів які раніше могли бути описані лише математично. За прикладом астрономії, медицина також отримала можливість створювати високоточні моделі мікроорганізмів задля їх подальшого дослідження. На ряду із мікроорганізмами можуть бути створені та описані моделі нового медичного обладнання яке в майбутньому зможе допомагати людям.

Метою розробки сайту є створення легкого та швидкого доступу до структурованої інформації до відповідного розділу теми, що цікавить користувача.

Мета полягає у створенні статичних та динамічних зображень для сайту «Зельман Ваксман». Також необхідно створити програмне забезпечення для реалізації неітеративної сферично-кутової інтерполяції векторів вздовж рядка растеризації за допомогою відповідного алгоритму.

Процес створення статичних та динамічних візуальних матеріалів із застосуванням фундаментальних принципів та методів комп’ютерної графіки є об’єктом дослідження курсової роботи.

Предметом дослідження курсової роботи є методи та засоби, які використовуються для створення статичних і динамічних зображень сайту, його інтерфейсу та програмного забезпечення.

Відповідного до цього, темою програмного застосунку є неітеративна сферично-кутовоа інтерполяція векторів вздовж рядка растеризації.

1 БАГАТОВИМІРНИЙ АНАЛІЗ РОЗВ’ЯЗКУ ОСНОВНОЇ ЗАДАЧІ

Основною задачею курсової роботи є створення статичних та динамічних зображень для вебсайту із візуально привабливим інтерфейсом який зможе детально розкрити основні етапи життя всесвітньо відомого вченого із українським корінням та його відкриттів. Для цього важливо правильно зробити перший крок, а саме визначити типу комп’ютерної графіки [1] яка буде використана у майбутньому. Основними претендентами на цю роль є векторна та растрова графіки. Щоб визначитись яка із них краще задовольняє потреби необхідно провести їх порівняльний аналіз та надати перевагу одній із них. Порівняльний аналіз двох типів графіки наведено у таблиці 1.

Таблиця 1.1 – Порівняльний аналіз типів графіки

|  |  |  |
| --- | --- | --- |
| Аспект | Векторна графіка | Растрова графіка |
| Масштабування | Якість зберігається  при будь-якому масштабі. | Втрата чіткості під час збільшення ("пікселізація"). |
| Розмір файлу | Зазвичай менший завдяки математичному опису об’єктів. | Великий розмір файлів, особливо з високою роздільною здатністю. |
| Легкість редагування | Легко змінювати розміри, стиль і кольори об’єктів. | Зміни можуть погіршувати якість та вимагати значної обчислювальної потужності. |
| Підтримка фотографій | Погано підходить для фотографій та реалістичних текстур. | Ідеально підходить для фотографій і градієнтів. |
| Якість на різних пристроях | Забезпечує чітке відображення елементів, наприклад, молекулярних структур, на будь-якому екрані. | Може мати розмитий вигляд при масштабуванні на великих або маленьких екранах. |

Проаналізувавши ці два види графіки можна дійти висновку, що найбільш доцільно буде використовувати векторну графіку для переважної більшості зображень на сайті, та растрову графіку у разі, якщо потрібно буде розмістити фотографію реальної людини чи іншого об’єкта.

Отже, для розробки графічного наповнення вебсайту, а саме статичних та динамічних зображень було застосовано векторний стиль комп’ютерної графіки. Це дозволило створити якісні зображення які легко масштабувати відповідно до вимог користувачів сайту. Щоб створювати ці зображення було обрано популярний графічний редактор Inkscape [2]. Inkscape – це потужний графічний редактор із відкритим кодом, який часто обирають для створення та редагування векторної графіки. Ось основні переваги та недоліки його використання:

Переваги:

1. Безкоштовність та відкритий код – Inkscape є абсолютно безкоштовним, а також з відкритим кодом, що дозволяє користувачам вносити зміни, вдосконалювати функції або налаштовувати програму для власних потреб.

2. Сучасні інструменти для роботи з векторною графікою – Inkscape надає повний набір інструментів для роботи з векторними об’єктами: інструменти для малювання, фігур, тексту, шляхи, контури, трансформації, обрізання тощо.

3. Розширення формату SVG – редактор зберігає файли у форматі SVG, який є стандартом для вебдизайну та забезпечує високу якість зображення незалежно від масштабу.

4. Інтеграція з іншими інструментами та форматами – Inkscape підтримує імпорт та експорт у багато форматів (EPS, PDF, PNG та інші), що полегшує інтеграцію з іншими графічними редакторами та обробку різних форматів зображень.

5. Багатофункціональність та налаштування інтерфейсу – Inkscape пропонує безліч додаткових функцій, включаючи можливість роботи з шарами, кривими Безьє, фільтрами, а також підтримує скрипти на Python для автоматизації роботи.

6. Кросплатформеність – програма доступна на Windows, macOS та Linux, що робить її універсальною для використання на різних операційних системах.

Недоліки:

1. Менша продуктивність при роботі з великими файлами – Inkscape може сповільнюватися під час обробки складних або великих проектів з великою кількістю елементів, через що робота стає менш продуктивною.

2. Відсутність деяких професійних функцій – у порівнянні з професійними графічними редакторами, такими як Adobe Illustrator, Inkscape має обмеження щодо специфічних інструментів, таких як розширені ефекти, додаткові опції для друку та підтримка CMYK для друкованих матеріалів.

3. Проблеми з сумісністю форматів – у деяких випадках Inkscape може не зовсім коректно імпортувати файли з інших форматів, особливо тих, що мають власні ефекти або фільтри, наприклад, AI-файли з Adobe Illustrator.

Програмний застосунок, що реалізуватиме неітеративну сферично-кутову інтерполяцію векторів вздовж рядка растеризації розроблятиметься у інтегрованому середовищі розробки Inteliji IDEA[3].

Отже, для розробки статичних та динамічних зображень було обрано графічний редактор Inkscape. Для розробки програмної компоненти було обрано мову програмування Java та інтегроване середовище розробки Inteliji IDEA.

2 ВИБІР КОЛЬОРОВОЇ ГАМИ

Одним із найбільш важливих елементів дизайну вебсторінки є колір [4]. Він значним чином впливає на зручність використання, користувацький досвід, а також на популярність та можливий прибуток від комерційного сайту. Задля покращення цих показників слід визначитись із кольоровою гамою сайту. Вона може значним чином впливати на вищезазначені характеристики, а також підкреслювати необхідні елементи сторінки або додати яскравості та динамічності. У зв’язку із цим існує набір певних практик які зарекомендували себе як стабільні та перевірені часом заходи які задовільняють користувачів. Наприклад слід використовувати контрастні кольори для тексту та фону сторінки, щоб зробити текст більш читабельним.

Також кольори можуть допомогти у випадку якщо компанія має намір стати популярною. Прикладами таких комбінацій кольорів є червоний та білий кольори логотипа компанії Coca-Cola (рисунок 2.1),



Рисунок 2.1 – Логотип компанії Coca-Cola

червоний та жовтий кольори у компаній Shell (рисунок 2.2) та McDonald`s, а також



Рисунок 2.2 – Логотип компанії Shell

комбінація зеленого та білого у мережі всесвітньо відомих кав’ярень Starbucks (рисунок 2.3).



Рисунок 2.3 – Логотип компанії Starbucks

Отже, наступним важливим кроком буде визначення кольорової гами. Існує декілька основних видів формування кольорової гами.

1. Монохроматична гамма — використовує один колір в різних відтінках і насиченостях. Ця гамма створює спокійний та елегантний ефект, зберігаючи цілісність дизайну, і часто застосовується, щоб передати простоту та стриманість.

2. Аналогова гамма — складається з трьох кольорів, розташованих поруч на колірному колі. Вона створює м'який та природний вигляд, і часто зустрічається в природі, де кольори поступово переходять один в одного.

3. Комплементарна гамма — базується на поєднанні кольорів, розташованих напроти один одного на колірному колі (наприклад, синій і помаранчевий). Вона додає контрастності, динамізму та привертає увагу, що добре підходить для акцентів та виразних елементів.

4. Тріадна гамма [5] — використовує три кольори, розташовані рівномірно на колірному колі, наприклад, червоний, синій і жовтий. Така гамма виглядає яскраво та збалансовано, додаючи життєрадісності та енергії.

В нашому випадку буде використано тріадну гаму через її очевидні переваги (баланс кольорів, контрастність та гнучкість у поєднанні кольорів). Використовуючи цю систему було обрано три кольори (рисунок 2.4).

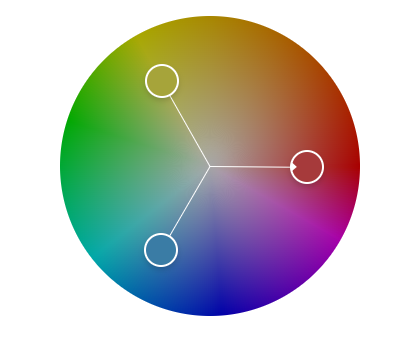


Рисунок 2.4 – Основні кольори вебсайту

#A63A3B –"брунатний" або "червона охра" .

#3A7CA5 –"кобальтовий синій".

#A6A43A –"гірчичний" або "зеленувато-жовтий".

Отже, для розробки сайту було обрано такі кольори: гірчичний, кобальтовий синій та брунатний.

3 РОЗРОБКА ЛОГОТИПУ САЙТУ

Як вже було зазначено у попередньому розділі логотип [6] є важливим елементом не тільки бренду, а й вебсторінки. Саме тому важливо підібрати логотип який буде підходити не тільки за своїм значенням, а й буде зрозумілим звичайному користувачу. У зв’язку з цим слід обрати те, що є простим і зрозумілим зображенням. Для такої мети добре підходить графічний логотип. Графічний (символьний) логотип — це тип логотипа, в якому основним елементом є графічний символ або піктограма, а не текст. Такий логотип зазвичай складається з простої і легко впізнаваної графічної форми, яка передає ідею або сутність бренду. При цьому символ або піктограма може бути незалежною від назви компанії, що дозволяє використовувати його автономно.

Враховуючи тематику сайту, а саме “Зельман Ваксман”, слід підібрати зображення на медичну тематику. Оскільки Зельман Ваксман був не тільки лікарем, а й біохіміком та дослідником, то логотип має символізувати дослідження, експерименти та здоров’я. Таким зображенням буде виступати колба, заповнена хімічним реактивом. Контур логотипа вебсайту наведено на рисунку 3.1.

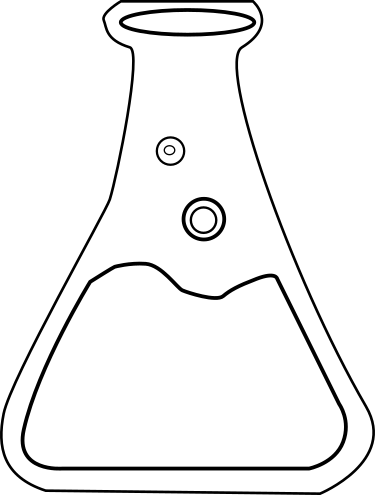


Рисунок 3.1 – Контур логотипа вебсайту

Отримавши готовий контур зображення, можна розпочати процес додавання кольору. Використовуючи кольору які зазвичай використовуються для зображення скла та рідин було обрано синій та блакитні кольори. Результат готового логотипу зображено на рисунку 3.2.

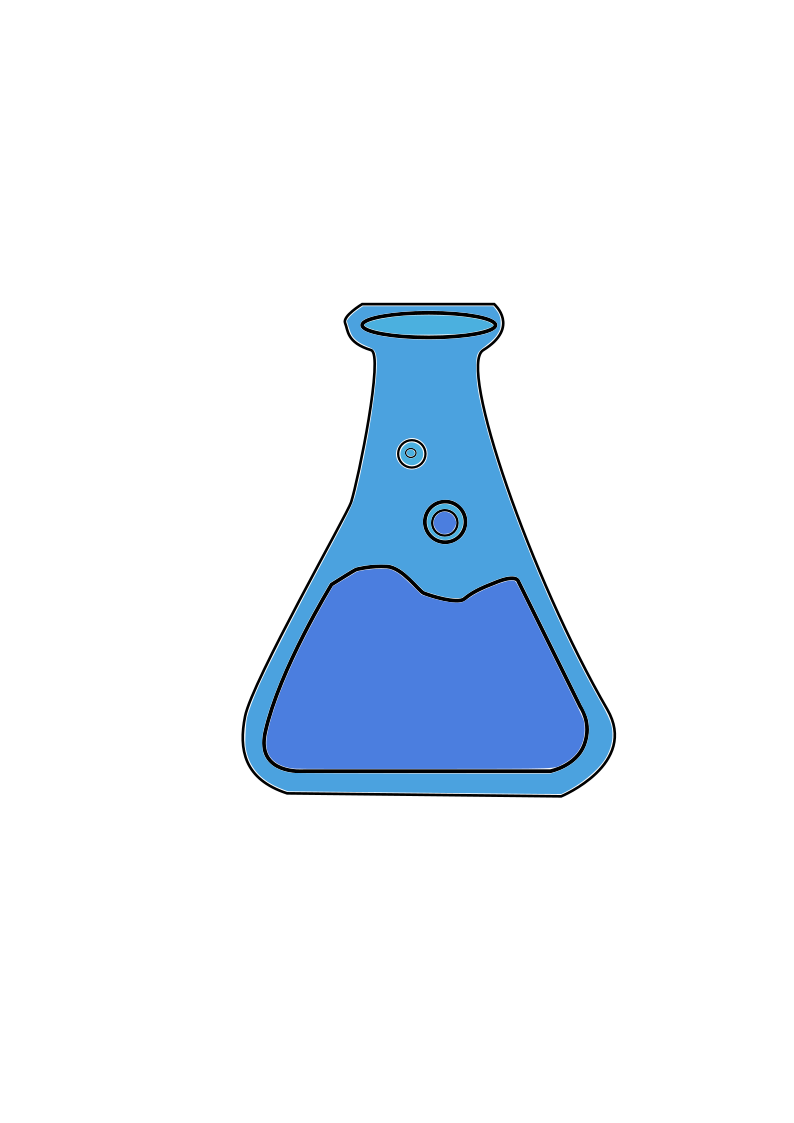


Рисунок 3.2 – Готовийлоготипу вебсайту

Отже, на основі обраної для вебсайту тематики, було обрано колбу з реагентом у якості логотипу.

4 РОЗРОБКА СТАТИЧНИХ ЗОБРАЖЕНЬ

Одним із найважливіших методів донесення інформації є супровід текстової інформації якісними зображеннями. У зв’язку із цим необхідно потрібно коректно підібрати тип, використаної на сайті графіки. Хоча основні відмінності растрової та векторної графіки було розглянуто у першому розділі, слід більш детально поглинути у їх відмінності.

Векторна графіка — це графіка, яка використовує математичні формули для зображення таких об’єктів, як сегменти ліній, криві та фігури. Основними форматами векторних зображень є SVG, AI та EPS.

Одною із переваг є невелика розмірність файлів. Це досягається тим, що для опису об’єктів використовуються математичні формули. Також завдяки цьому підвищується компактність що напряму впливає на швидкість завантаження сторінкою вебсайту.

Другим важливим фактором є відсутність втрат якості при збільшені або зменшені зображень. Найбільш яскравим прикладом цієї переваги є використання векторних зображення на невеликих екранах телефонів чи ноутбуків.

Іншим важливим пунктом є інтерактивність. Векторні зображення легко змінюються. До них можна застосовувати різні стилі, змінювати їх характеристики, а також можливість застосовувати CSS-анімацію і JavaScript, що робить їх інтерактивними й гнучкими у вебдизайні.

У цей час є лише один значний недолік. Він полягає у тому, що векторні зображення погано підходять для фотографій. Цей формат погано передає реалістичні текстуру та кольорові градієнти.

Растрові зображення складаються з пікселів, що утворюють детальну картину. Найбільш характерними форматами для растрових зображень є JPEG, PNG, GIF, BMP.

Порівняно із векторною графікою, растровій характерна висока деталізація. Піксельна структура дозволяє створювати дуже деталізовані зображення, які необхідні для фотографій і складних кольорових градієнтів.

Сумісність з різними форматами є додатковим корисним функціоналом. Формати, як-от JPEG і PNG, підтримуються більшістю пристроїв та програм, ідеальні для фотографій і зображень з великою кількістю кольорів.

Для растрової графіки характерні недоліки які є оберненою стороною переваг векторного типу. Растрова графіка є більш об’ємною та займає більше місця. Також її дуже складно масштабувати зберігаючи при цьому якість картинки.

Саме тому, детально проаналізувавши ці види графіки було прийнято остаточно рішення розробляти статичні зображення у векторному стилі. Усі зображення буде створено у безкоштовному графічному редакторі Inkscape та розміщено на сайті.

Тепер розглянемо приклад створення основних зображень на сайті. Першим зображенням окрім логотипа стали спори Streptomyces griseus**.** Для початку необхідно створити контур зображення (рисунок 4.1) який, у майбутньому буде зафарбовано.

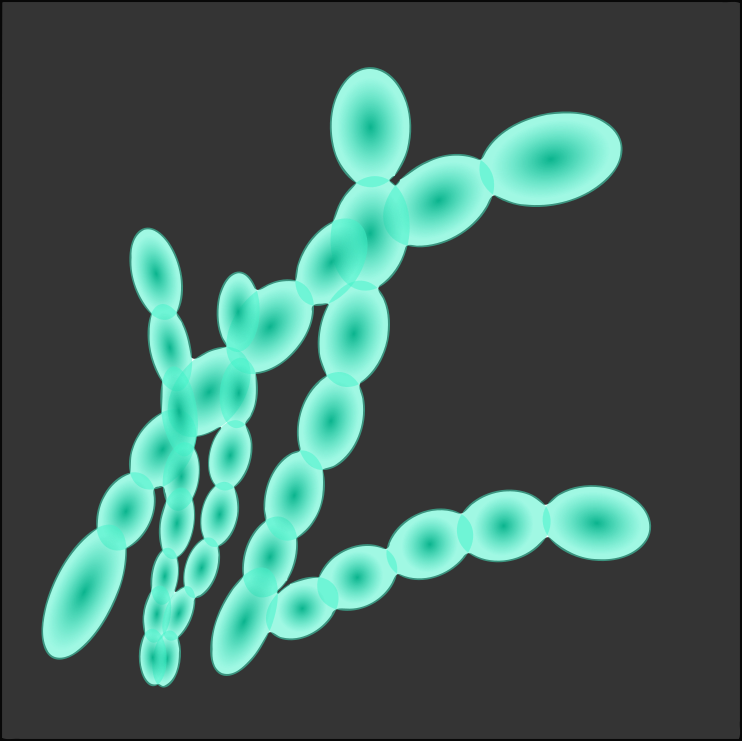


Рисунок 4.1 –Зображення спор Streptomyces griseus

Наступним на черзі є зображення актиноміцетів. Вони були основним предметом досліджень вченого протягом десятків років його наукової діяльності. Їх зображення переважно складається із великої кількості довгастих об'єктів різної довжини та форми. Для передачі різноманіття їх форм було використано інструмент – крива Безьє. Вона дозволяє будувати різні форми із можливістю одразу залити зображення кольором. Зображення актиноміцетів наведено на рисунку 4.2

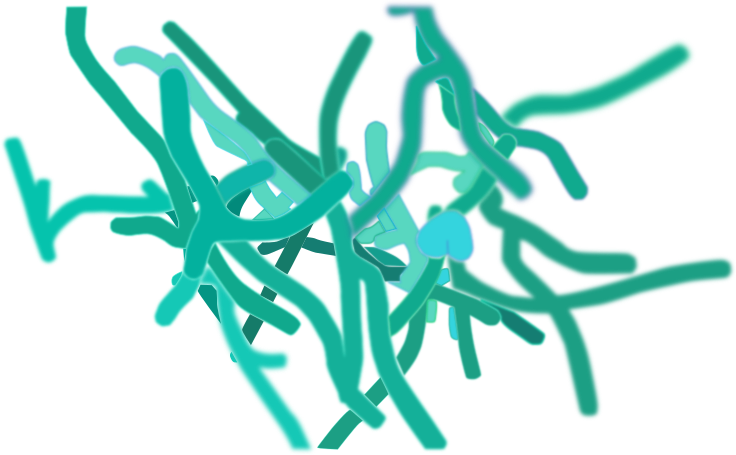


Рисунок 4.2 – Зображення актиноміцетів

Наслідком наукової роботи та дослідів Ваксмана став препарат який стане першим дієвим засобом від туберкульозу. Оскільки це зображення (рисунок 4.3) являє собою хімічною структурою тому важливо зберегти правильні кольори заливки, оскільки вони позначають різні атоми.

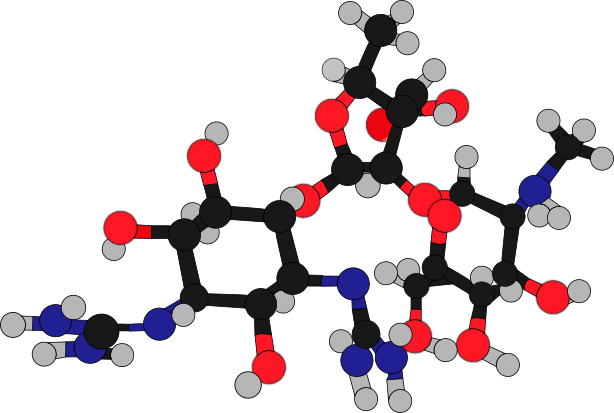


Рисунок 4.3 – Зображення кольорової структури стрептоміцину

Наступним зображенням є шприц - один із основних інструментів дослідника-мікробіолога. Зображення шприца наведено на рисунку 4.4.

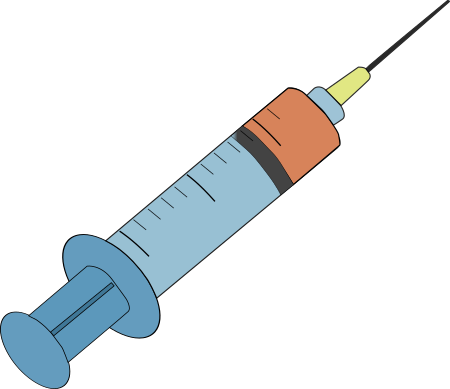


Рисунок 4.4 – Зображення шприца із поділками

Наступним зображенням буде Нобелівська премія (риснуок 4.5). Цю нагороду Зельмана Ваксман отримав за винахід ліків від туберкульозу.

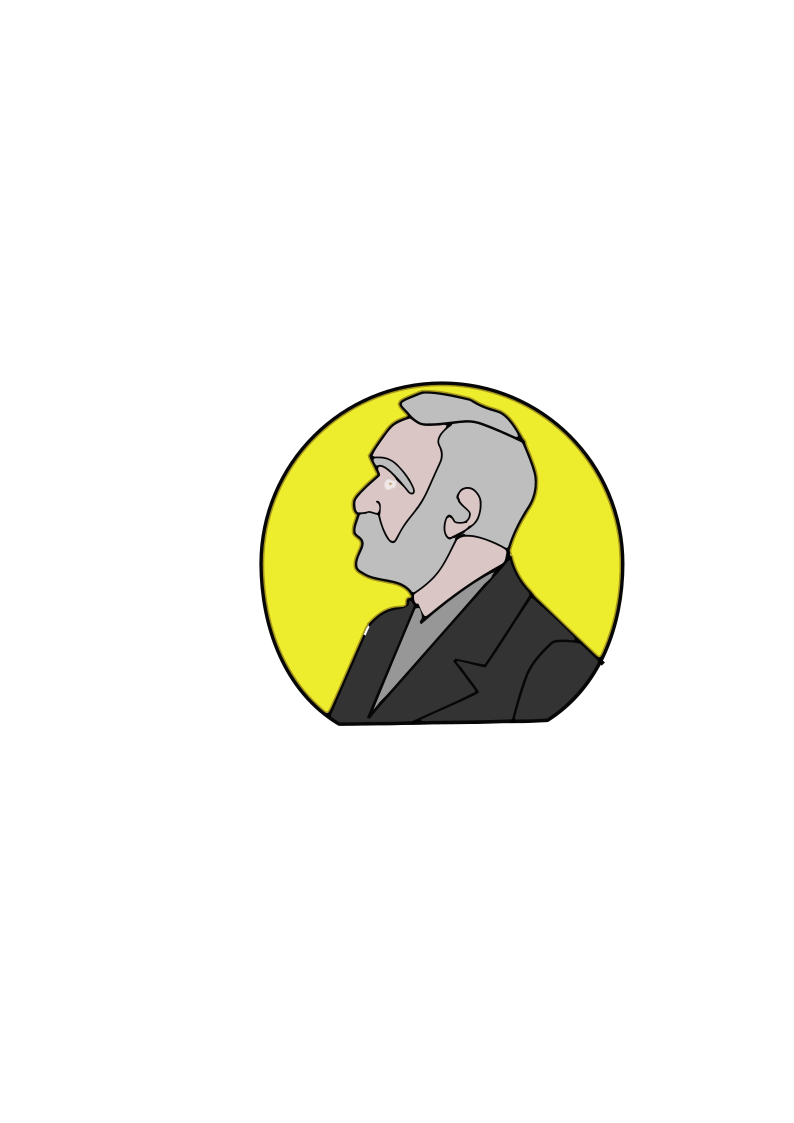


Рисунок 4.5 – Зображення Нобелівскьої премії

Одним із інструментів дослідника-мікробіолога є мікроскоп. Зображення його кольорової версії (рисунок 4.6) яка була створена методом зафарбовування контуру.

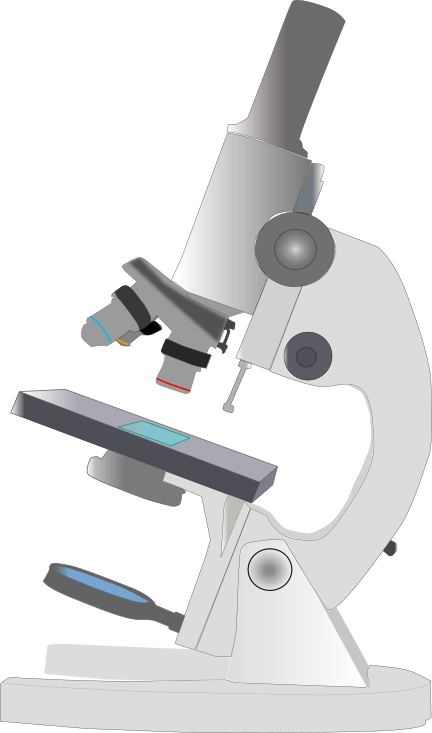


Рисунок 4.6 – Контур зображення мікроскопа

Наступним зображенням буде чаша Петрі яка безпосередньо пов’язна із мікроскопом, адже вона використовується для культивації певних видів бактерій які також присутні на зображенні(рисунок 4.7).



Рисунок 4.7 –Зображення чашки Петрі із бактеріями

Восьмим зображенням буде препарат «Стрептоміцин» (рисунок 4.8), який є дуже відомим антибіотиком широкого спектру..



Рисунок 4.8 –Зображення антибіотика «Стрептоміцин»

Передостаннім зображення буде обкладинка журналу Time (рисунок 4.9) на яку попав Зельман Ваксман за свої наукові відкриття.



Рисунок 4.9 –Зображення обкладинки журналу Time

Останнім за номером, та не за значенням є бактерія (рисунок 4.10), зображення якої лягло в основу GIF анімації.

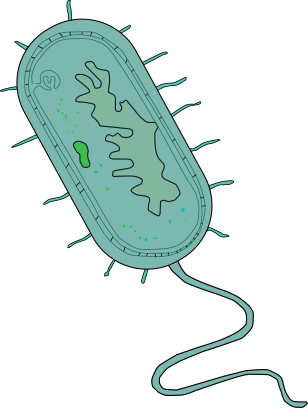


Рисунок 4.10 –Зображення бактерії

Отже, було проведено глибокий аналіз видів графіки, а також продемонстровано створення статичних зображень у застосунку Inkscape. Всі створені зображення наведені в додатку А.

5 РОЗРОБКА ДИНАМІЧНИХ ЗОБРАЖЕНЬ

Динамічні зображення, або анімації, є важливою складовою сучасних вебсайтів. Із розвитком сучасних IT технологій динамічні зображення стали більш якісними та знаходять все більше способів інтеграції для задоволення потреб користувачів. Це пов’язано насамперед із розвитком апаратних можливостей у графічного обладнання нового покоління. Сучасні графічні чіпи стають все більш доступними для широкого спектра людей. Це означає що все більше людей можуть дозволити собі насолоджуватись високоякісними анімаціями. Саме через це важливо додавати анімації, щоб зробити вебсайт більш сучасним та привабливим для користувачів.

Тепер важливо вирішити який вид анімацій слід обрати для нашого вебсайту.

Анімація поділяється на двовимірну та тривимірну анімацію. Двовимірна анімація – це така форма анімації, яка використовує векторну графіку для створення двовимірних об’єктів. Важливою функцією цього виду анімацій є використання векторів, що дозволяє з легкістю маніпулювати створеними об’єктами.

В свою чергу тривимірна анімація представляє більш сучасний та високотехнічний підхід у сфері комп’ютерної графіки. Цей метод анімації використовує тривимірну графіку для створення реалістичних сцен, які широко використовуються у багатьох галузях від розваг до медицини та авіації.

В нашому випадку буде використана одна із видів двовимірної анімації, а саме GIF[7].

GIF-анімація (Graphics Interchange Format) — це тип цифрової графіки, який дозволяє створювати короткі, зациклені анімації, зазвичай без звуку. На відміну від відеофайлів, GIF зберігає послідовність кадрів у стислому форматі з обмеженою палітрою (до 256 кольорів). Це робить їх легшими за розміром і відповідним для швидкого завантаження на вебсайтах. GIF-анімація складається з послідовності зображень або кадрів, які змінюються один за одним, створюючи ефект руху. Кожен кадр може відображатися певний час, що дозволяє налаштовувати швидкість анімації. Коли всі кадри програні, анімація може автоматично повторюватися без зупинки. Саме через переваги такого виду анімацій GIF буде обрано у якості анімації для вебсайту.

Щоб створити GIF анімацію достатньо створити певну послідовність кадрів за допомогою таких інструментів як Adobe Photoshop, GIMP, або в онлайн-інструментах, як-от Canva чи Ezgif. У нашому випадку було використано онлайн застосунок Canva для створення анімації. У якості динамічного зображення було обрано зображення бактерії що рухається. Перший кадр анімації наведено на рисунку 5.1.

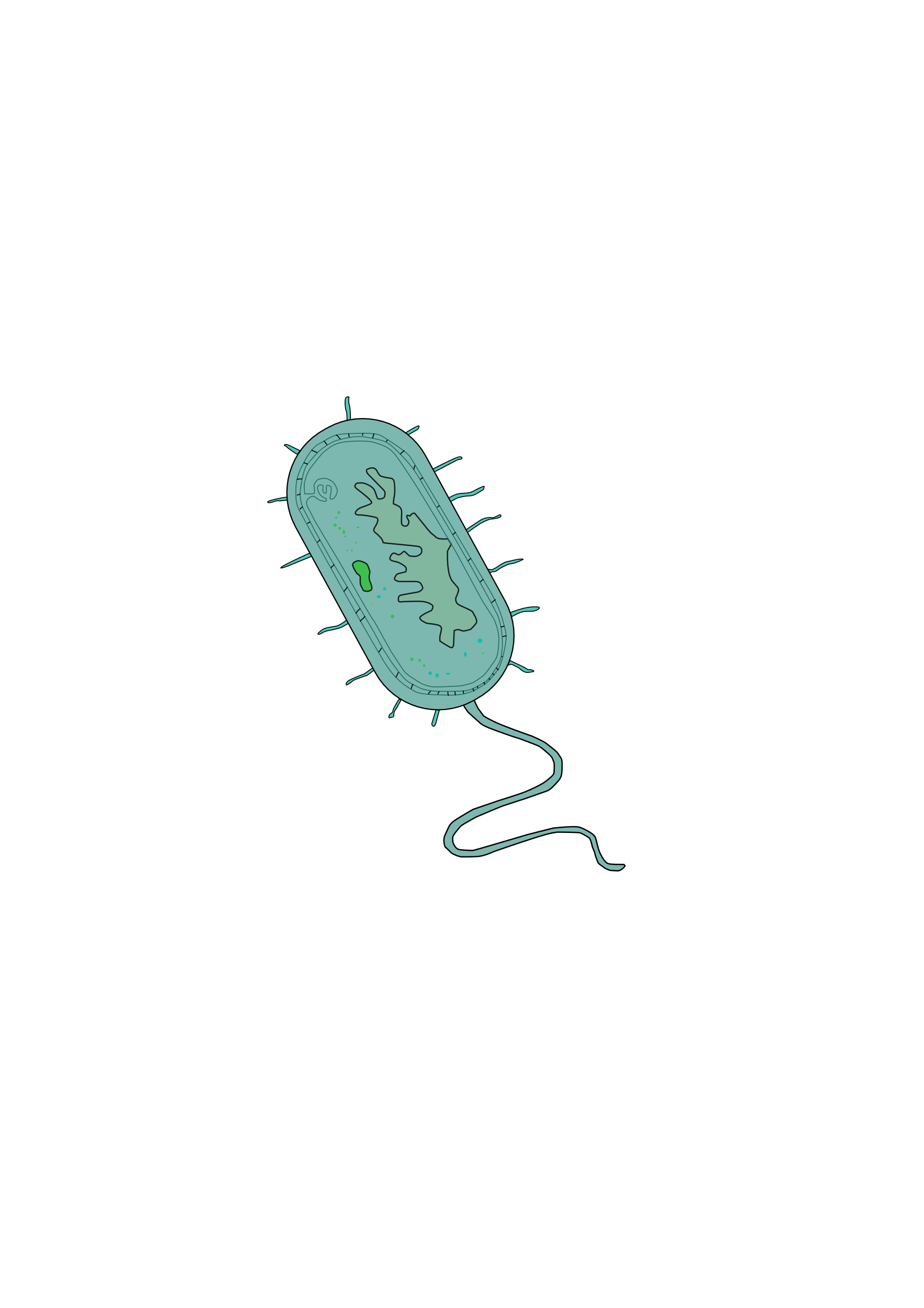


Рисунок 5.1. – Перший кадр анімації бактерії

Для подальшого створення анімації необхідно додати певні зміни до зображення, щоб при використанні правильної їх послідовності створити вигляд того, що об’єкт знаходиться у стані постійного руху. Наступним кадром анімації буде вигляд зображення бактерії яка починає свій рух. Для цього буде змінено “волосинки” або фімбрії що знаходяться по периметру тіла бактерії. Відповідне зображення бактерії на початку руху наведено на рисунку 5.2.

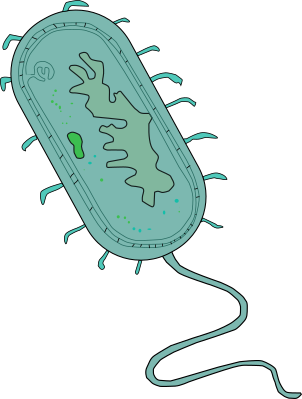


Рисунок 5.2. – Другий кадр анімації бактерії

На наступному зображені (рисунок 5.3) видно що джгутик починає випрямлятись що свідчить про продовження руху мікроорганізму.

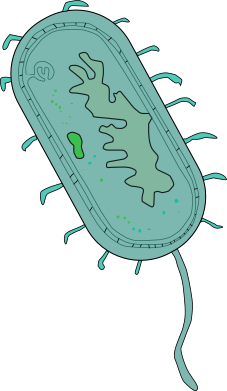


Рисунок 5.3. – Третій кадр анімації бактерії

На наступному зображені (рисунок 5.4) відображення фазу стрімкого руху мікроорганізму за якої він досягає найбільшої швидкості.

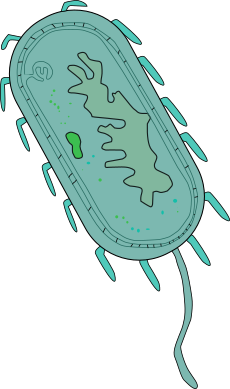


Рисунок 5.4. – Четвертий кадр анімації бактерії

На п’ятому кадрі зображенгя (рисунок 5.5) відображається процес “гальмування” руху мікроорганізму

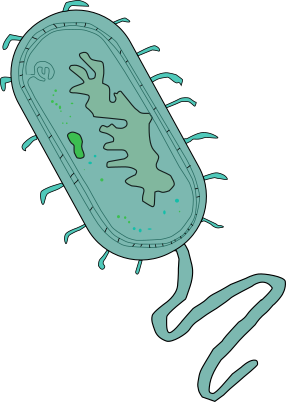


Рисунок 5.5. – П’ятий кадр анімації бактерії

На останньому кадрі (рисунок 5.6) анімації бактерія повертається до свого початкового стану та завершує цикл свого руху

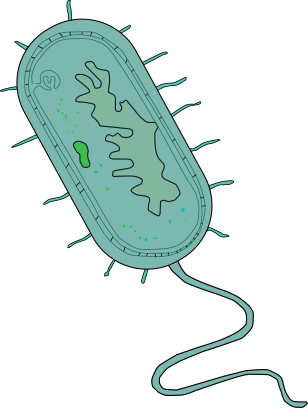


Рисунок 5.6. – Шостий кадр анімації бактерії

Отже, для розробки і перевірки роботи динамічних зображень було обрано застосунок Canva. Усі кадри розробленої анімації наведено в додатку Б.

6 ОЦІНКА ОБСЯГІВ ЗОБРАЖЕНЬ

Чим швидше розвивається наш світ, тим актуальнішим стає питання інформації та її обсягів. Правильна оцінка обсягу інформації та оптимізація зображень мають ключове значення для створення ефективних та швидких вебсайтів. Невеликі, оптимізовані зображення зменшують навантаження на сервер і покращують швидкість завантаження сторінок, що позитивно впливає на досвід користувачів і SEO[8].

Для фотографій та зображень з багатою кольоровою гамою підходить формат JPEG (JPG), який дозволяє добре стискати файли з мінімальною втратою якості. PNG ідеально підходить для зображень з прозорістю, наприклад, для логотипів та іконок. SVG рекомендується використовувати для векторної графіки, такої як ілюстрації або логотипи, оскільки він не втрачає якості при масштабуванні та дозволяє зображенням залишатися чіткими на екранах різних розмірів. Новітній формат WebP, створений спеціально для вебсайтів, пропонує високу якість стиснення та підтримку прозорості, поєднуючи переваги JPEG і PNG в одному форматі.

Тому для формату зображень вебсайту варто обрати стабільний формат PNG оскільки більшість статичних зображень мають прозорий фон. Додатковою перевагою цього формату над іншим є те, що векторне зображення при конвертуванні у формат PNG не втратить своєї якості. Після конвертування зображень у новий формат було оцінено їх об’єм та занесено в таблицю 6.1.

Таблиця 6.1 – Оцінка розмірів зображень

|  |  |
| --- | --- |
| Назва зображення | Розмір зображення, кб |
| Logo.png | 44,3 |
| Actinomyces.png | 217 |
| Bacteria.png | 49,5 |
| Microscope.png | 83,7 |
| Nobelprize.png | 60,7 |
| Petri.png | 104 |
| Spores.png | 197 |

Продовдження таблиці 6.1

|  |  |
| --- | --- |
| Назва зображення | Розмір зображення, кб |
| Struct.png | 80,1 |
| Syringe.png | 27,2 |
| Waksman.png | 40,8 |
| Bacteria-gif.gif | 82,9 |
| Streptomycinum\_farm.png | 62,7 |

Для визначення кінцевого навантаження на окремі сторінки вебсайту спричиненого зображеннями можна оцінити сумарний об’єм для кожної їх сторінки. Відповідні результати занесені у таблицю 6.2.

Таблиця 6.2 – Розміри зображення на сторінках сайту

|  |  |
| --- | --- |
| Назва сторінки | Загальний об’єм зображень, кб |
| Формування особистості | 284 |
| Дослідження | 735,3 |
| Стрептоміцин | 351,6 |
| Нагороди | 187 |

Отже, було проаналізовано та обрано формат зображень який буде використовуватись на сайті. Після експортування зображення у обраний формат було сформовано таблицю значень розміру зображень, а також таблицю об’єму зображень на кожній сторінці.

7 ТЕСТУВАННЯ РОБОТИ САЙТУ

У нашому світі доступність сайту є не менш важливою ніж його наповнення. Саме тому важливо підтримувати якість вебсайту вчасним тестуванням для виявлення недоліків не тільки програмної реалізації, а й ефективності. Щоб виявляти такі недоліки слід проводити такі типу тестування як: тестування доступності , тестування ефективності, тестування функціональності.

Тестування доступності є важливою складовою первірки вебсайту на зручність його використання різними категоріями людей. Таке тестування охоплює перевірку на відповідність стандартам та забезпечення взаємодії х допоміжними технологіями, як екранні зчитувачі.

Тестування ефективності – це загальна первірка продуктивності роботи вебсайту під різним навантаженням, щоб оцінити його параметри. Таке тестування може включати стрес-тестування та об’ємне тестування.

Тестування функціональності являє собою перевірку того, наскільки коректно працює вебсайт та чи відповідає він зазначеним вимогам. Цей вид тестування включає перевірку роботи кнопок, форм, навігації, а також інтеграції з іншими важливими складовими вебсайту (API). Окремо можна виділити таку складовку як SEO або search engine optimization. Вона відповідає за те, як часто користувачам в інтернеті буде рекомендовано ваш вебсайт, а значить наскільки він може бути прибутковим.

Саме для тестування цих параметрів існує Lighthouse[9].

Lighthouse — це інструмент, розроблений компанією Google, який допомагає покращити якість вебсайтів шляхом проведення всебічного тестування різних його аспектів. Він оцінює такі важливі характеристики, як продуктивність, доступність, SEO-оптимізація, відповідність найкращим практикам безпеки та підтримка прогресивних вебдодатків (PWA).

Основна перевага використання Lighthouse полягає в тому, що він дозволяє побачити сайт очима користувача та пошукових систем. Інструмент оцінює продуктивність сайту, зокрема швидкість завантаження сторінок, що є ключовим для утримання відвідувачів і зниження показника відмов.

Запустивши тестування готового вебсайту за допомогою цього інструмента були отримані такі результати(рисунок 7.1).

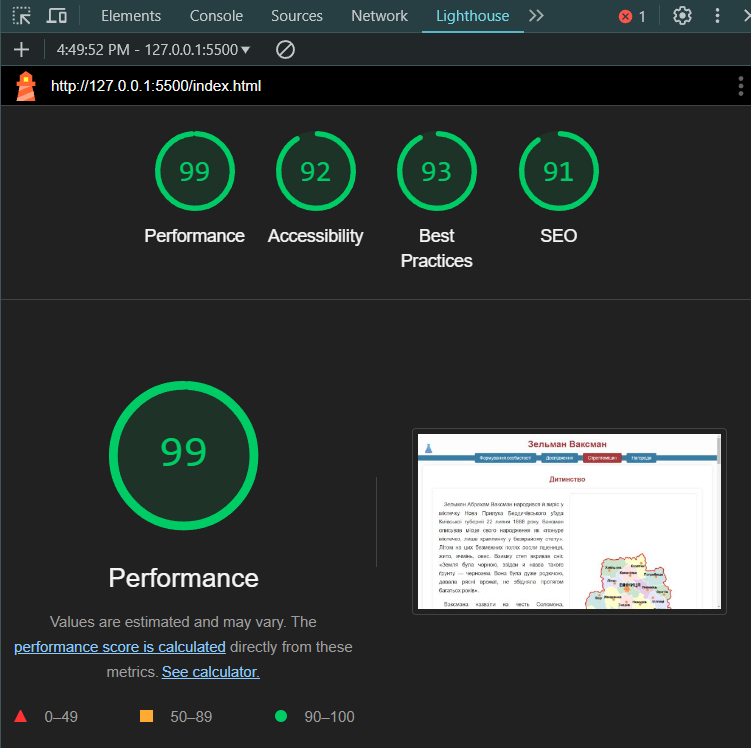


Рисунок 7.1 – Результат тестування вебсайту

Аналізуючи результати тестування можна дійти висновоку, що вебсайт відповідає високим стандартам продуктивності та є доступним для користувачів на різноманітних пристроях. Вебсайт застосовує найкращі практики із своєї галузі для покращення його роботи. Висока оцінка параметру SEO (див. рисунок 7.1) зазначає той факт, що пошукові машини (Google, Yahoo!) будуть часто рекомендувати його користувачам в мережі інтернет.

Після цього слід дослідити контент сайту на відповідність темі курсового проекту. Зображення тестованого сайту наведено на рисунку 7.2.



Рисунок 7.2 – Зображення тестованого вебсайту

Отже, проаналізувавши тестування, можна дійти висновку, що сайт задовольняє технічним вимогам. Також було проаналізовано його вміст, що був визначений як відповідний до теми проекту. Лістинг програмного коду наведено в додатку В.

8 РОЗРОБКА ПРОГРАМНОГО ЗАСТОСУНКУ

Технічне завдання :

Неітеративна сферично-кутова інтерполяція вздовж рядка растеризації

Розмір координатного простору — 1024\*768

Кольоровий режим — ture.color

Кількість векторів — від 10 до 200

Під час роботи над курсовим проектом було розроблено програмну реалізацію неітеративної сферично-кутової інтерполяції векторів. Застосунок було розроблено мовою програмування Java із застосування бібліотеки для розробки користувацьких інтерфейсів JavaFX[10]. Код програми був написаний в інтегрованому середовищі розробки Inteliji IDEA.

Неітеративна сферично-кутова інтерполяція векторів — це метод інтерполяції, який дозволяє знаходити проміжні вектори на поверхні сфери між двома заданими векторами. Основною характеристикою такого методу є те, що він забезпечує плавний та мінімально викривлений перехід між векторами, використовуючи геометрію сфери. Формула для інтерполяції виглядає так:



(8.1)

Де:

* N(t) – інтерпольований вектор у момент часу t;
* N(a) та N(k) – початковий і кінцевий одиничний вектори;
* Φ — кут між векторами N(a) та N(k).

Сферична інтерполяція може бути застосована у різних галузях комп’ютерної графіки і не тільки.

У робототехніці, для планування руху маніпуляторів та інших рухомих частин робота.

Інтерполювання орієнтацій використовується в тривимірному просторі коли важливо забезпечити плавне обертання об’єкта без ривків.

Сферично-кутова інтерполяція також широко застосовується у комп’ютерній графіці, а саме для забезпечення плавного руху камери, об’єктів чи персонажів за допомогою своїх алгоритмів. Приклад сферично-кутової інтерполяції векторів наведено на рисунку 8.1.



Рисунок 8.1 – Приклад сферично-кутової інтерполяції

Вхідні дані програмного застосунку:

– кількість векторів між нормалями;

– координати для одиничних нормалей.

Ці параметри не є константними, що дає змогу користувачеві змінювати відповідні параметри для більшої наочності можна змінювати у застосунку:

1. Текстові поля для зміни координат нормальних векторів.

2. Спінер дає можливість обрати кількість векторів розташованих посеред 2 нормальних векторів.

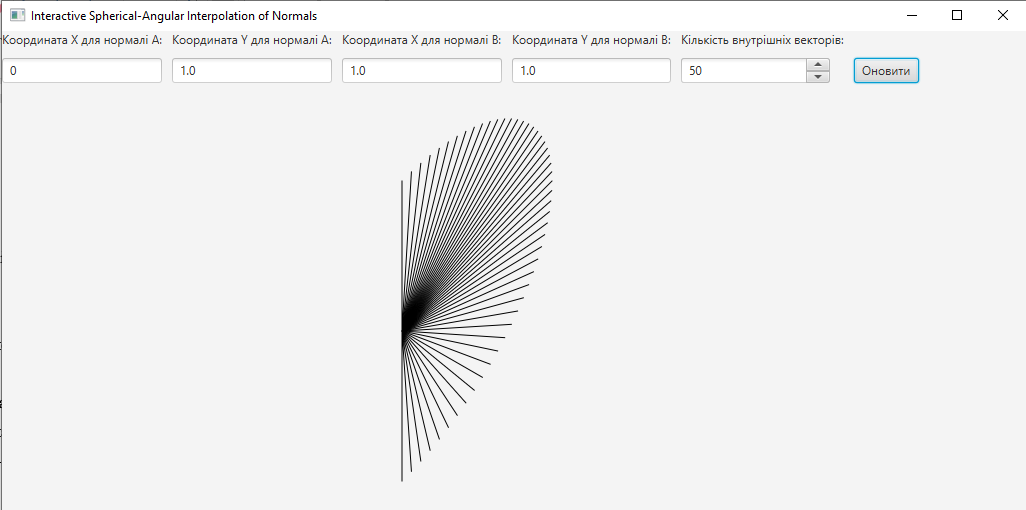
3. Кнопка «Намалювати», після натискання створює зображення відповідно до заданих параметрів. Приклад сферично-кутової інтерполяції із зміненими параметрами наведено на рисунку 8.2.

Рисунок 8.2 – Приклад інтерполяції із зміненими параметрами

Вихідні дані:

– сформоване зображення сферично-кутової інтерполяції векторів.

Малювання зображення відбувається в процесі обрахування програмою проміжних векторів для значень t від 0 до 1 із заданим кроком, які будуть розташовані між початковими одиничними нормалями. Після їх обрахування вони будуть зображені на полотні. Також до них буде застосовано параметр scale який являє собою масштаб. Алгоритм сферично-кутової інтерполяції продемонстровано у вигляді блок схеми та наведено на рисунку 8.3.

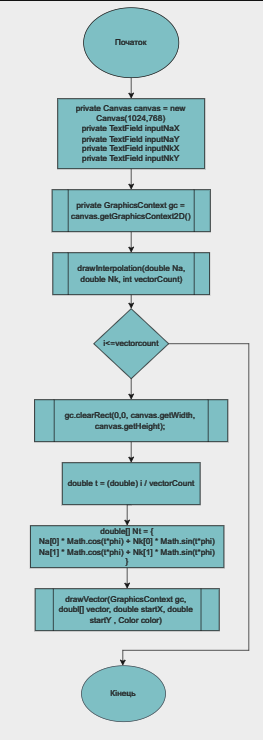


Рисунок 8.3 – Блок схема алгоритму програми

Отже, програмна реалізація неітеративної сферично-кутової інтерполяції векторів вздовж рядка растеризації була реалізована на мові Java із використанням бібліотеки для створення користувацьких інтерфейсів JavaFX. Застосунок дозволяє користувачеві задати параметри для створення зображення, що дає змогу користувачеві переглянути роботу алгоритму із різними параметрами.

В процесі розробки програмної реалізації програми було вивчено та реалізовано на практиці алгоритм сферично-кутової інтерполяції векторів. Вхідні параметри, такі як координати нормальних векторів та кількість проміжкових векторів між ними задаються користувачем через інтерфейс користувача.

Лістинг коду наведено в додатку Г.

ВИСНОВКИ

У ході виконання курсової роботи було детально проаналізовано важливість комп’ютерної графіки у сучасному світі, а також її численні сфери використання. Для цього здійснено всебічний і багатовимірний аналіз методів розв’язку задач, результати якого стали основою для подальшого обґрунтування вибору найбільш доцільного виду графіки. Згідно з результатами проведеного аналізу, було визначено пріоритетний вид графіки, який виявився найбільш відповідним для проєкту, а саме векторна графіка, що відзначається високою якістю та гнучкістю використання.

Після вибору виду графіки було проведено аналіз кольорових моделей, де розглянуто різні типи кольорових гам. У результаті обрано тріадну модель як основну кольорову гаму для вебсайту, що дозволило забезпечити гармонійне поєднання кольорів і привабливий візуальний стиль. Далі розроблено значну кількість векторних зображень, як статичних, так і динамічних, що відповідають тематиці курсового проекту «Зельман Ваксман». Медіаконтент було створено з використанням сучасних технологій комп’ютерної графіки та інструментів графічного дизайну, таких як Inkscape та Canva.

Як результат, було успішно розроблено вебсайт за темою «Зельман Ваксман», який задовольняє всі технічні та творчі вимоги. Окрім цього, у процесі виконання курсової роботи створено програмний застосунок мовою Java із використанням бібліотеки JavaFX для реалізації графічного інтерфейсу. У цьому застосунку здійснюється програмна реалізація неітеративної сферично-кутової інтерполяції векторів вздовж лінії растеризації, що підвищує точність і продуктивність процесу.

Для програмного застосунку було розроблено блок-схему його алгоритму роботи.

Сайт було розроблено з метою донесення звичайним користувачам історії життя та наукової роботи видатного вченого, що народився у Вінницький області, а саме Зельмана Ваксмана.

# ПЕРЕЛІК ПОСИЛАНЬ

1. Романюк О.Н., Романюк О.В., Чехместрук Р. Ю. Комп’ютерна графіка:  
навчальний посібник. Вінниця : ВНТУ, 2023. 147 с.

2. Inkscape – Inkscape overview. URL : <https://inkscape.org/about/> (дата звернення 30.09.2024).

3. Inteliji IDEA – The Leading Java and Kotlin IDE. URL: <https://www.jetbrains.com/idea/> (дата звернення 02.10.2024).

4. Колір — це найшвидший спосіб створити хороше перше враження про продукт. URL : <https://dou.ua/forums/topic/39339/> <https://inkscape.org/about/> (дата звернення 05.10.2024).

5. 7 колірних схем для гармонійного дизайну. URL: <https://blog.depositphotos.com/ua/kolirni-shemy.html> (дата звернення 15.10.2024).

6. Логотип це? види, значення та функції. URL: <https://it-rating.ua/logotip-tse-vidi-znachennya-ta-funktsii> (дата звернення 16.10.2024).

7.

[GIF-файли: створення, редагування й відкриття. URL: https://www.adobe.com/ua/creativecloud/file-types/image/raster/gif-file.html (дата звернення 16.10.2024).](GIF-файли: створення, редагування й відкриття. URL: https://www.adobe.com/ua/creativecloud/file-types/image/raster/gif-file.html  (дата звернення 16.10.2024).)

8. Що таке SEO. URL: <https://www.serp-hub.org/ua?gad_source=1&gclid=Cj0KCQjwvpy5BhDTARIsAHSilylYPUAIa842AmU8SJlYnChcO8fSNWI69m8_LIYWdCts80yclU1epm8aApxNEALw_wcB> (дата звернення 19.10.2024).

9.

[Overview | Lighthouse - Chrome for Developers. URL: https://developer.chrome.com/docs/lighthouse/overview (дата звернення 21.10.2024).](Overview | Lighthouse - Chrome for Developers. URL: https://developer.chrome.com/docs/lighthouse/overview  (дата звернення 21.10.2024).)

10. JavaFX - JavaFX is an open source, next generation client application platform for desktop. URL: https://openjfx.io/ (дата звернення 05.11.2024).

ДОДАТОК А. СТАТИЧНІ ЗОБРАЖЕННЯ

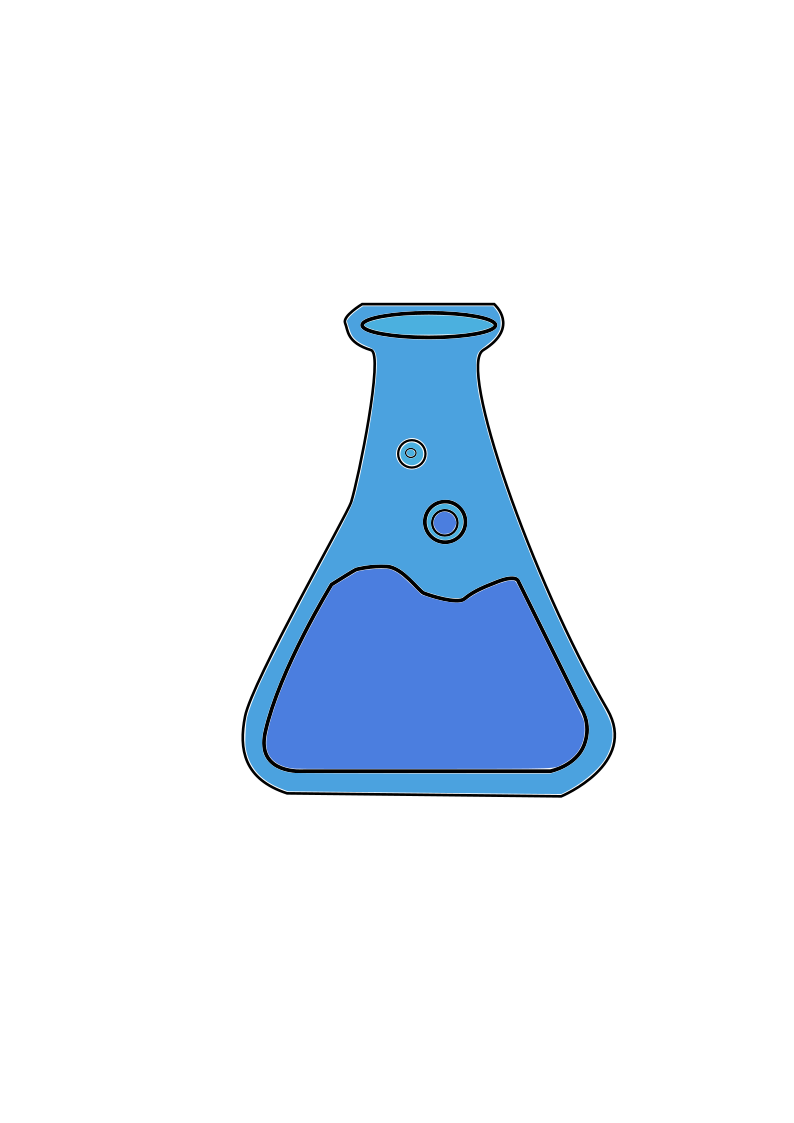


Рисунок А. 1 – Векторне зображення колби

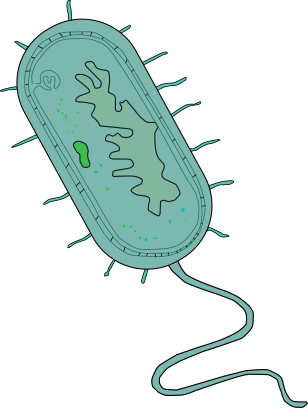


Рисунок А. 2 – Векторне зображення бактерії

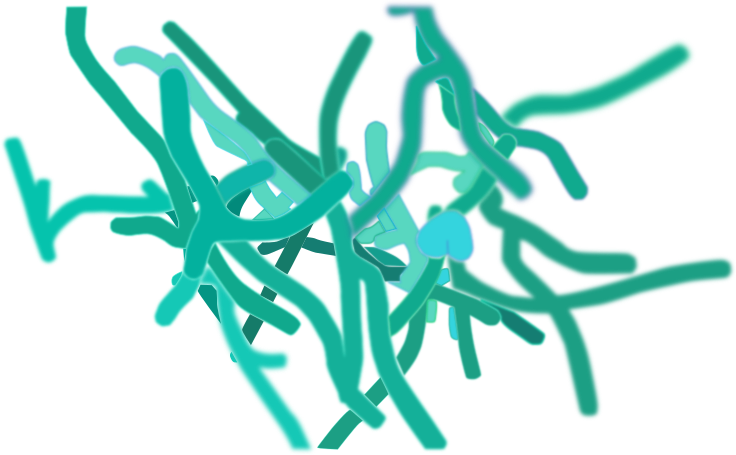


Рисунок А. 3 – Векторне зображення актиноміцетів

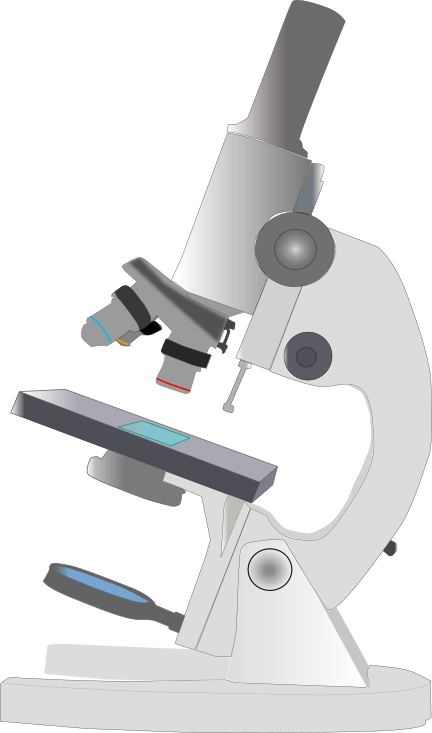
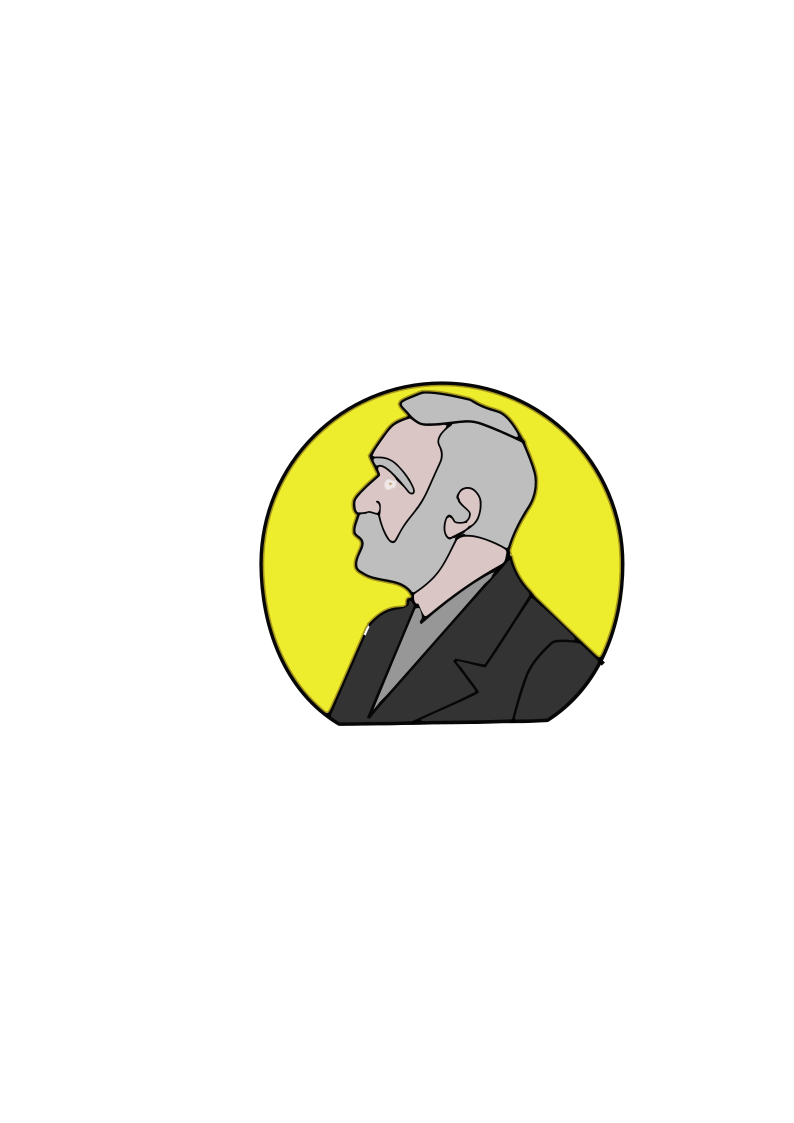


Рисунок А. 4 – Векторне зображення мікроскопа



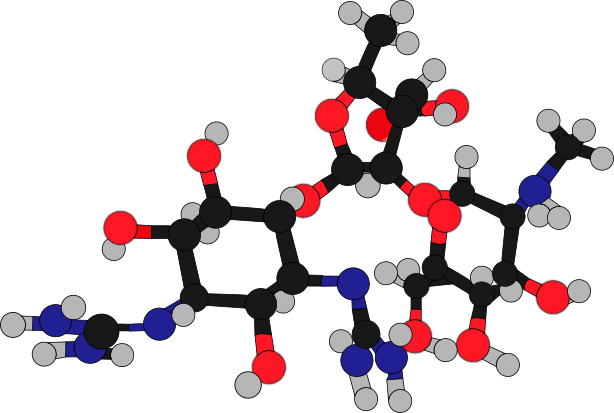
Рисунок А. 5 – Векторне зображення Нобелівської премії

Рисунок А. 6 – Векторне зображення структури стрептоміцину

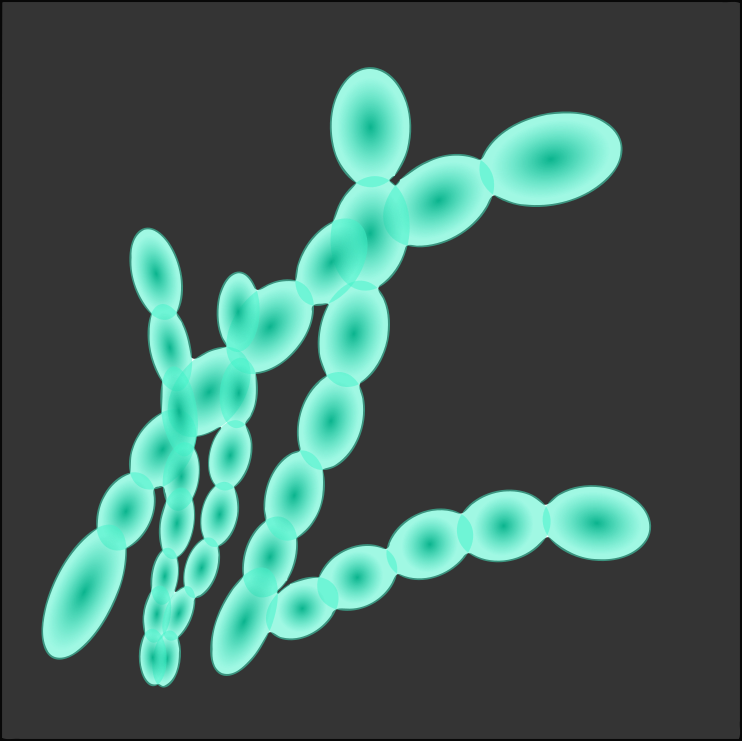


Рисунок А. 7 – Векторне зображення спор грибів



Рисунок А. 8 – Векторне зображення чашки Петрі

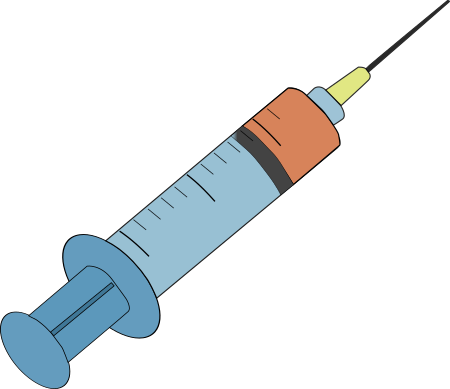


Рисунок А. 9 – Векторне зображення шприци



Рисунок А. 10 – Векторне зображення журналу TIME



Рисунок А. 11 – Векторне зображення антибіотика «Стрептоміцин»

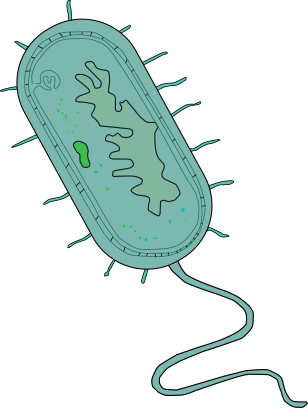
ДОДАТОК Б. ДИНАМІЧНЕ ЗОБРАЖЕННЯ

Рисунок Б. 1 – 1 кадр анімації

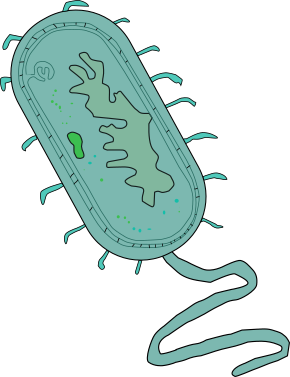


Рисунок Б. 2 – 2 кадр анімації

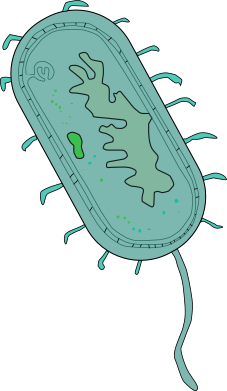


Рисунок Б. 3 – 3 кадр анімації

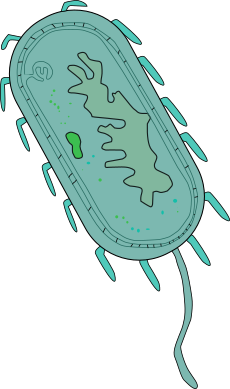


Рисунок Б. 4 – 4 кадр анімації

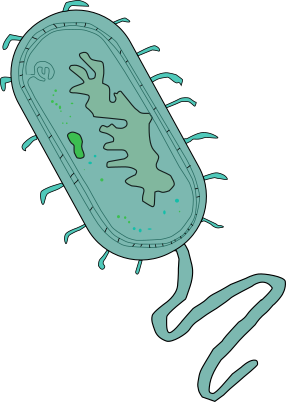


Рисунок Б. 5 – 5 кадр анімації

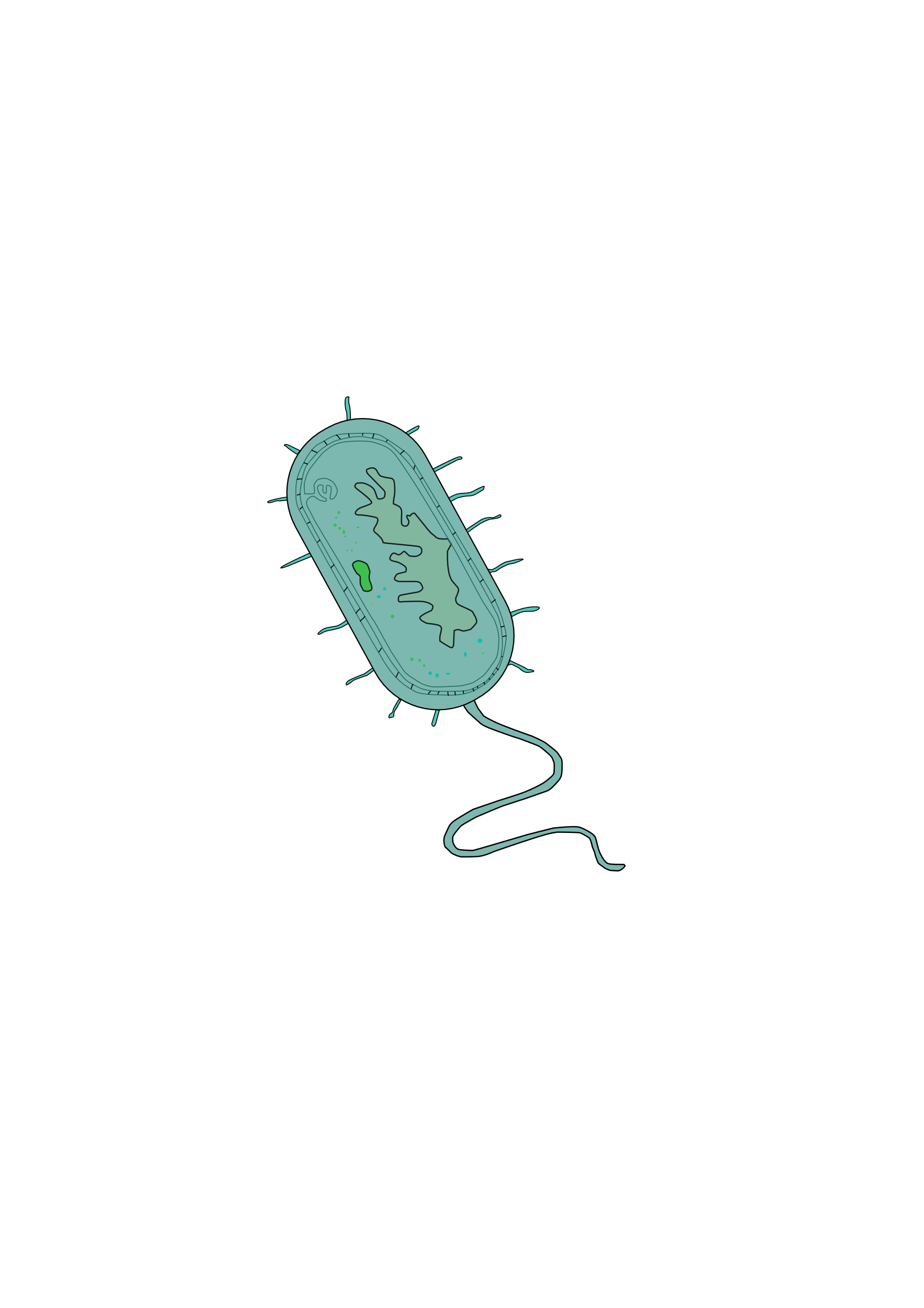


Рисунок Б. 6 – 6 кадр анімації

ДОДАТОК В. ЛІСТИНГ ВЕБСТОРІНКИ INDEX.HTML

<!DOCTYPE html>

<html lang="uk">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Зельман Ваксман - Формування особистості </title>

    <link rel="stylesheet" href="styles.css">

    <script src="script.js"></script>

</head>

<body>

    <header>

        <div class="logo">

            <img src="photos/logo.WebP" alt="Website Logo">

        </div>

        <h1>Зельман Ваксман</h1>

        <nav>

            <ul>

                <li><a href="index.html">Формування особистості</a></li>

                <li><a href="methods.html">Дослідження</a></li>

                <li><a href="streptomyces.html">Стрептоміцин</a></li>

                <li><a href="awards.html">Нагороди</a></li>

            </ul>

        </nav>

    </header>

    <main>

        <h2>Дитинство</h2>

        <section  class="content">

            <div class="left-content-alt">

              <p>

                Зельман Абрахам Ваксман народився й вирiс у мiстечку Нова Прилука Бердичiвського уїзда Київської губернiї 22 липня 1888 року.

                Ваксман описував мiсце свого народження як «понуре мiстечко, лише краплинку у безкрайому степу». Лiтом на цих безмежних полях росли пшениця, жито, ячмiнь, овес.

                Взимку степ вкривав снiг. «Земля була чорною, звiдси й назва такого ґрунту — чернозем. Вона була дуже родючою, давала ряснi врожаї, не збiдняла протягом багатьох рокiв».

            </p>

            <p>

                Ваксмана назвали на честь Соломона, бiблiйного царя, чиє iм’я мовою iдиш спростилося до Золмана (написання й вимова «Зельман» з’явилися вже в Америцi).

                Батько Зельмана Якiв Ваксман був набожною людиною i жив невибагливо, вiддаючи в аренду невеликi дома, що належали йому у сусiднiх селах та у Винницi.

                Вiльний час, яким вiн завдячував прибутковi, батько присвячував молитвi та вiдвiдинам мiсцевої синагоги.

                В автобiографiї Ваксман опише його як чудового оповiдача, що любив повчальнi притчi про стародавнiх мудрецiв та багатовiкову iсторiю єврейського народу.

            </p>

            </div>

            <div class="right-content-alt">

                <img src="photos/vinnitsa\_map.WebP" alt="Вінницька область">

            </div>

        </section>

        <h2>Батьки Ваксмана</h2>

        <section  class="content">

            <div class="left-content">

                <img src="photos/selo.WebP" alt="Зельман Ваксман фото">

            </div>

            <div class="right-content">

            <p>

                Коли Зельман народився, батько був абсолютно щасливий, але незабаром його вiдправили до вiйська, де вiн вiдслужив довгих п’ять рокiв. Син з батьком не були надто близькi, проте Якiв Ваксман завжди намагався допомогти сину як мiг. От тiльки мiг вiн не так вже й багато.

            </p>

            <p>

                Головною людиною дитинства й юностi майбутнього вченого стала його мати Фрейда Ваксман (дiвоче прiзвище Лондон). Освiчена, як на тогочасну жiнку: читала на iдиш, володiла iвритом у достатнiй мiрi, щоб самостiйно вивчати Тору, послуговувалася грамотною українською. Все це стало їй у нагодi в мабутньому.

            </p>

               <p>

                Як згадує Ваксман, мати пiзно вийшла замiж (майже у критичному для того часу вiцi, двадцятисемирiчною), оскiльки весь свiй час присвячувала урядженню долi кожної з молодших сестер. Вона збудувала дiм — без зайвої розкошi, аде власний, свiй. В цьому домi постiйно мешкали сестри Фрейди з її племiнниками й племiнницями, а також її мати. Фрейда була справжньою iдише маме — владною та вiдповдальною.

               </p>

            </div>

        </section>

        <h2>Ранні роки</h2>

        <section  class="content">

            <div class="left-content-alt">

            <p>

                У п’ятирiчному вiцi хлопчика вiддали до мiсцевого хедеру, де меламед навчав дiтей читати Тору. З часом освiта поширювалася, невдовзi майбутнiй мiкробiолог вже поринув у книги пророкiв й почав вивчати Талмуд. Але Фрейда не для того народила єдиного сина, щоб вiн обмежився цим знанням. Самого лише хедера для її сина замало, в цьому вона була впевнена, тому найняла для десятирiчного хлопця домашнiх вчителв, якi навчили його iвриту, iсторiї, арифметики й географiї. Ваксман зазначає, що у тринадцятирiчному вiцi непогано знав Тору i Талмуд. З десяти рокiв вiн сам починає заробляти як викладач. Спочатку вчить учнiв хедера читати й писати, згодом готує дiтей з багатих родин до вступу у свiтськi школи. Те, що вдається заробити, йде на його вчителiв.

            </p>

            <p>

                Ваксман готувався до вступних iспитiв у гiмназiю. Дiстатися до гiмназiї було нелегко. I через вже iснуючу процентну норму для євреїв: їхня кiлькiсть не могла перевищувати строго визначеного невеликого вiдсотка учнiв, що створювало неабияку конкуренцiю мiж самими євреями. I через звичайний брак гiмназiй у Бердичевi та недалекiй Вiнницi. А шлях до унiверситету лежав саме через гiмназiю. Ваксман готувався скласти iспити екстерном за першi шiсть рокiв навчання, щоб дiстатися до сьомого класу житомирської гiмназiї. I ця спроба закiнчилася невдачею.

            </p>

            </div>

            <div class="right-content-alt">

                <img src="photos/Vaksman\_photo.WebP" alt="Зельман Ваксман фото">

            </div>

        </section>

        <h2>Нью-Джерсi</h2>

        <section  class="content">

            <div class="left-content">

                <img src="photos/jersi.WebP" alt="jersi">

            </div>

            <div class="right-content">

            <p>

                Восени 1910 року корабель, яким пливли Ваксман та його друзi, прибув з Гамбурга до Фiладельфiї. Тут шляхи хлопцiв i дiвчат з Прилук мали розiйтися. Ваксмана зустрiли двоюрiднi сестри, i вiн вирушив до рiдних у Нью-Джерсi.

            </p>

            <p>

                У його двоюрiдної сестри були невеличка ферма й курник. Коли сьогоднi ми, звiсно, надто швидко йдемо шляхом, який крок по кроку, долаючи величезнi труднощi, поконував майбутнiй вчений, неможливо не думати про те, що його у буквальному розумннi провадила рука провидiння. Саме на цiй фермi Ваксман вперше працював з курами.

            </p>

            </div>

        </section>

        <div class="gif-container">

            <img src="photos/bacteria-gif.gif" alt="Animated GIF" class="animated-gif" />

        </div>

    </main>

    <footer>

        <p class="footr">&copy; 2024 Зельман Ваксман</p>

    </footer>

</body>

</html>

ДОДАТОК Г. ЛІСТИНГ ПРОГРАМНОГО ЗАСТОСУНКУ

package com.example.demo;  
  
import javafx.application.Application;  
import javafx.scene.Scene;  
import javafx.scene.canvas.Canvas;  
import javafx.scene.canvas.GraphicsContext;  
import javafx.scene.control.Button;  
import javafx.scene.control.Label;  
import javafx.scene.control.Spinner;  
import javafx.scene.control.TextField;  
import javafx.scene.layout.GridPane;  
import javafx.scene.layout.Pane;  
import javafx.stage.Stage;  
import javafx.scene.paint.Color;  
  
public class InteractiveNormalInterpolation extends Application {  
 private Canvas canvas = new Canvas(1024, 768);  
 private GraphicsContext gc = canvas.getGraphicsContext2D();  
  
 @Override  
 public void start(Stage stage) {  
 // Елементи для введення координат нормалей  
 TextField inputNaX = new TextField("1.0");  
 TextField inputNaY = new TextField("0.0");  
 TextField inputNkX = new TextField("0.0");  
 TextField inputNkY = new TextField("1.0");  
  
 // Підписи під текстовими полями  
 Label labelNaX = new Label("Координата X для нормалі A:");  
 Label labelNaY = new Label("Координата Y для нормалі A:");  
 Label labelNkX = new Label("Координата X для нормалі B:");  
 Label labelNkY = new Label("Координата Y для нормалі B:");  
  
 // Елемент для вибору кількості внутрішніх векторів  
 Spinner<Integer> vectorCountSpinner = new Spinner<>(10, 200, 20);  
 Label vectorCountLabel = new Label("Кількість внутрішніх векторів:");  
  
 // Кнопка для оновлення візуалізації  
 Button updateButton = new Button("Намалювати");  
  
 // Панель для розташування елементів  
 GridPane controls = new GridPane();  
 controls.setHgap(10);  
 controls.setVgap(10);  
 controls.add(labelNaX, 0, 0);  
 controls.add(inputNaX, 0, 1);  
 controls.add(labelNaY, 1, 0);  
 controls.add(inputNaY, 1, 1);  
 controls.add(labelNkX, 2, 0);  
 controls.add(inputNkX, 2, 1);  
 controls.add(labelNkY, 3, 0);  
 controls.add(inputNkY, 3, 1);  
 controls.add(vectorCountLabel, 4, 0);  
 controls.add(vectorCountSpinner, 4, 1);  
 controls.add(updateButton, 5, 1);  
  
 // Панель для малювання та керування  
 Pane root = new Pane();  
 root.getChildren().addAll(canvas, controls);  
  
 // Обробник для кнопки оновлення  
 updateButton.setOnAction(e -> {  
 double[] Na = {Double.parseDouble(inputNaX.getText()), Double.parseDouble(inputNaY.getText())};  
 double[] Nk = {Double.parseDouble(inputNkX.getText()), Double.parseDouble(inputNkY.getText())};  
 int vectorCount = vectorCountSpinner.getValue();  
 drawInterpolation(Na, Nk, vectorCount);  
 });  
  
  
  
 Scene scene = new Scene(root, 1024, 768);  
 stage.setScene(scene);  
 stage.setTitle("Interactive Spherical-Angular Interpolation of Normals");  
 stage.show();  
 }  
  
 private void drawInterpolation(double[] Na, double[] Nk, int vectorCount) {  
 gc.clearRect(0, 0, canvas.getWidth(), canvas.getHeight());  
 double phi = Math.PI; // Фіксований кут для інтерполяції  
 vectorCount++;  
 for (int i = 0; i <= vectorCount; i++) {  
 double t = (double) i / vectorCount;  
 double[] Nt = {  
 Na[0] \* Math.cos(t \* phi) + Nk[0] \* Math.sin(t \* phi),  
 Na[1] \* Math.cos(t \* phi) + Nk[1] \* Math.sin(t \* phi)  
 };  
 drawVector(gc, Nt, 400, 300, Color.BLACK);  
 }  
 }  
  
  
 private void drawVector(GraphicsContext gc, double[] vector, double startX, double startY, Color color) {  
 double scale = 150; // Масштаб для кращої видимості  
 gc.setStroke(color);  
 gc.strokeLine(startX, startY, startX + vector[0] \* scale, startY - vector[1] \* scale);  
 }  
  
 public static void main(String[] args) {  
 launch(args);  
 }  
}