

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ**

Кафедра _____ Комп'ютеризованих систем управління _____

ДИПЛОМНИЙ ПРОЕКТ
(ПОЯСНЮВАЛЬНА ЗАПИСКА)

**ВИПУСКНИКА ОСВІТНЬО-КВАЛІФІКАЦІЙНОГО РІВНЯ
"БАКАЛАВР"**

Тема: «Програмний модуль
реалізації алгоритму набору тексту
“Клавіатурний тренажер”»

Виконавець: Таран І.А.

Керівник: Марченко Н.Б.

Нормоконтролер: Тупота Є.В.

Рецензент: _____

Київ 2021

6. Календарний план

| <i>№ п/п</i> | <i>Етапи виконання дипломного проекту</i> | <i>Термін виконання етапів</i> | <i>Примітка</i> |
|------------------|---|--|-----------------|
| 1. | <i>Аналіз предметної області та постановка завдання проектування.</i> | | |
| 2. | <i>Розробка структури програмного модуля</i> | | |
| 3. | <i>Розробка інтерфейсу програмного модуля</i> | | |
| 4. | <i>Реалізація функціоналу програмного модуля</i> | | |
| 5. | <i>Тестування програмного модуля</i> | | |
| 6. | <i>Написання пояснювальної записки</i> | | |
| 7. | <i>Підготовка доповіді та презентації</i> | | |

7. Дата видачі завдання: “ _____ ” _____ 2021 р.

Керівник дипломної роботи (проекту) _____ Марченко Н.Б.
(підпис керівника) (П.І.Б.)

Завдання прийняв до виконання _____ Таран І.А.
(підпис студента)

РЕФЕРАТ

Пояснювальна записка до дипломного проекту «Програмний модуль «Клавіатурний тренажер»»: 60 с., 53 рис., 7 табл., 27 джерел.

КЛАВІАТУРНИЙ ТРЕНАЖЕР, КОД МОРЗЕ, СЛІПІЙ ДРУК, СХЕМА, ПРОГРАМНИЙ МОДУЛЬ, ФОРМА, ІНТЕРФЕЙС, С++

Об'єкт дослідження – процес навчання методиці сліпого друку.

Предмет дослідження – програмний модуль «клавіатурний тренажер».

Мета дипломного проекту – розробити програмний модуль «клавіатурний тренажер» для можливості навчання сліпому методу друку.

Методи проектування – середовище розробки Microsoft Visual Studio Express 2013 для Windows Desktop, мова програмування С++.

Прогнозні припущення щодо розвитку об'єкта дослідження – використання робочого зразка програмного модуля для оволодіння методом сліпого друку, ознайомлення з розкладкою клавіатури та вивчення азбуки Морзе на персональних комп'ютерах.

Результати проектування – програмний модуль для навчання сліпому методу друку, який розроблено для операційної системи Windows.

ЗМІСТ

| | |
|--|----|
| ВСТУП..... | 6 |
| РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАВДАННЯ ПРОЕКТУВАННЯ | 8 |
| 1.1 Аналіз програм типу «Клавіатурний тренажер» | 8 |
| 1.2 Код Морзе | 12 |
| 1.3 Постановка завдання проектування | 14 |
| 1.4 Висновки до розділу..... | 15 |
| РОЗДІЛ 2. СТРУКТУРА ПРОГРАМНОГО МОДУЛЯ | 16 |
| 2.1 Функціональні вимоги | 16 |
| 2.2 Нефункціональні вимоги | 17 |
| 2.3 Структура програмного модуля..... | 18 |
| 2.4 Висновки до розділу..... | 23 |
| РОЗДІЛ 3. РОЗРОБКА ПРОГРАМНОГО МОДУЛЯ..... | 24 |
| 3.1 Покрокова розробка | 24 |
| 3.2 Використовувані методи | 37 |
| 3.3 Тестування програмного модуля | 40 |
| 3.4 Висновки до розділу..... | 44 |
| РОЗДІЛ 4. КЕРІВНИЦТВО КОРИСТУВАЧА | 45 |
| 4.1 Технічні вимоги..... | 45 |
| 4.2 Керівництво користувача..... | 45 |
| 4.3 Висновки до розділу..... | 56 |
| ВИСНОВКИ | 57 |
| ПЕРЕЛІК ПОСИЛАНЬ | 59 |

ВСТУП

Уміння людини працювати на комп'ютері багато в чому визначається її здатністю швидко і грамотно набирати тексти. Для розвитку таких умінь застосовують спеціальне програмне забезпечення навчального призначення – клавіатурні тренажери. Вони орієнтовані як на новачків, так і на досвідчених користувачів. При чому програми-тренажери враховують спеціалізацію людини (наприклад, на роботу з числами або текстом) і залежно від цього моделюють вправи.

Десятипальцевий метод набору має значні переваги перед звичайним. Оскільки під час друкування десятипальцевим методом використовуються всі десять пальців, це зумовлює високу швидкість друку у порівнянні зі звичайним методом.

Люди які володіють методом "сліпого" машинопису можуть друкувати тексти зовсім не підглядаючи на клавіатуру. Саме розташування клавіш на будь-якій клавіатурі спроектоване спеціально для десятипальцевого методу набору і воно (розташування клавіш) не змінювалось уже протягом десятків років. Тож вивчивши один раз цей метод людині не потрібно буде хвилюватись, що цей метод застаріє.

Об'єкт дослідження – процес навчання методиці сліпого друку.

Предмет дослідження – програмний модуль «клавіатурний тренажер».

Мета дипломного проекту – розробити програмний модуль «клавіатурний тренажер» для можливості навчання сліпому методу друку.

Методи проектування – середовище розробки Microsoft Visual Studio Express 2013 для Windows Desktop, мова програмування C++.

Прогнозні припущення щодо розвитку об'єкта дослідження – використання робочого зразка програмного модуля для оволодіння методом сліпого друку, ознайомлення з розкладкою клавіатури та вивчення азбуки Морзе на персональних комп'ютерах.

Результати проектування – програмний модуль для навчання сліпому методу друку, який розроблено для операційної системи Windows.

Завдання проекту:

1. Проаналізувати програми типу «Клавіатурний тренажер».
2. Обрати мову та середовище для реалізації програмного модуля.
3. Визначити функціональні та нефункціональні вимоги до модуля.
4. Розробити структуру програмного модуля.
5. Розробити доповнюючі схеми та діаграми..
6. Розробити інтерфейс програмного модуля.
7. Реалізувати 3 режими навчання набору тексту.
8. Реалізувати додатковий функціонал інтерфейсу.
9. Виконати тестування програмного модуля.
10. Визначити технічні вимоги програмного модуля.
11. Розробити керівництво користувача.
12. Викласти вимоги до робочого місця техника-програміста.
13. Провести розрахунок освітлення виробничого приміщення.
14. Провести розрахунок рівня шуму на робочому місці.

Дипломний проект складається з п'яти розділів.

У першому розділі «Аналіз предметної області та постановка завдання проектування» проведено, відповідно, аналіз програм типу «Клавіатурний тренажер», розглянуто код Морзе та поставлено завдання проектування;

У другому розділі «Структура програмного модуля» описано функціональні та нефункціональні вимоги, а також основні складові частини програмного модуля;

У третьому розділі «Розробка програмного модуля» розглянуто покрокову розробку, методи програмної реалізації функціоналу та тестування програмного модуля;

У четвертому розділі «Керівництво користувача» описано програмне та апаратне забезпечення програмного модуля, а також керівництво користувача;

У п'ятому розділі описано охорону праці.

РОЗДІЛ 1.
АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА
ЗАВДАННЯ ПРОЕКТУВАННЯ

1.1 Аналіз програм типу «Клавіатурний тренажер»

Клавіатурний тренажер – це програма, призначена для навчання сліпому друку, придбання навичок десятипальцевого методу набору та збільшення швидкості друку на клавіатурі комп'ютера.

Клавіатурний тренажер представляє собою навчальний текстовий редактор для набору певного текстового фрагмента, з системою підказок під час набору на клавіатурі і функцією підрахунку різних метрик друку (швидкість друку, ритмічність набору, підрахунок допущених помилок тощо).

Найбільшу популярність отримали онлайн-версії клавіатурних тренажерів, оскільки це самий практичний і зручний спосіб навчання сліпому друку і вивчення розкладки комп'ютерної клавіатури.

Тренажер друку, як онлайн-сервіс, не вимагає установки на комп'ютер користувача. Доступ до навчання можливий з будь-якої операційної системи за допомогою сучасного браузера. Займатися на клавіатурному тренажері можна з будь-якого комп'ютера, що має вихід в мережу Інтернет.

Результати виконання завдань зберігаються в хмарі і ніколи не загубляться.

Соло на клавіатурі (див. рисунок 1.1) – це один з найвідоміших і популярних клавіатурних тренажерів. Програма платна. Навчання, крім російської та англійської мов, є також німецькою. З безперечних переваг відзначимо методологічний супровід протягом усього курсу (саме курсу, це не просто тренажер). Це і рекомендації по поставі, по правильності постановки кистей і пальців, мотивуючі введення до кожного уроку з порадами, листами від клієнтів і анекдотами.

| | | | | | | | |
|---------------------|-----------------|--|--|--|---------------|--------------|----------------|
| Кафедра КСУ | | | | НАУ 21 11 30 000 ПЗ | | | |
| <i>Виконав</i> | Таран І.А.. | | | АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАВДАННЯ ПРОЕКТУВАННЯ | <i>Літера</i> | <i>Аркуш</i> | <i>Аркушів</i> |
| <i>Керівник</i> | Марченко Н.Б. | | | | | 8 | 60 |
| <i>Консульт.</i> | | | | | 123 СП-437 | | |
| <i>Норм. контр.</i> | Тупота Є.В. | | | | | | |
| <i>Зав.каф</i> | Литвиненко О.Є. | | | | | | |

При бажанні це все можна відключити. В цілому програма має багатий функціонал.

Для внесення елементу розважальності в рутину навчання з «Соло» автори створили дочірній сайт, на якому можна перевіряти швидкість набору і змагатися в цьому показнику з іншими людьми. З недоліків хотілося б відзначити дуже довгий процес навчання. Для людей з недоліком терпіння і посидючості пройти курс повністю буде нелегко. В Інтернеті часто жартують, що ця програма є не тільки тренінгом сліпого десятипальцевого друку, але і сили волі. Незручним також здається заборона на звичне друкування під час роботи з тренажером, особливо для людей, які кожен день працюють з набором величезних текстів, а також неможливість перейти до наступного уроку, поки не пройдений попередній.

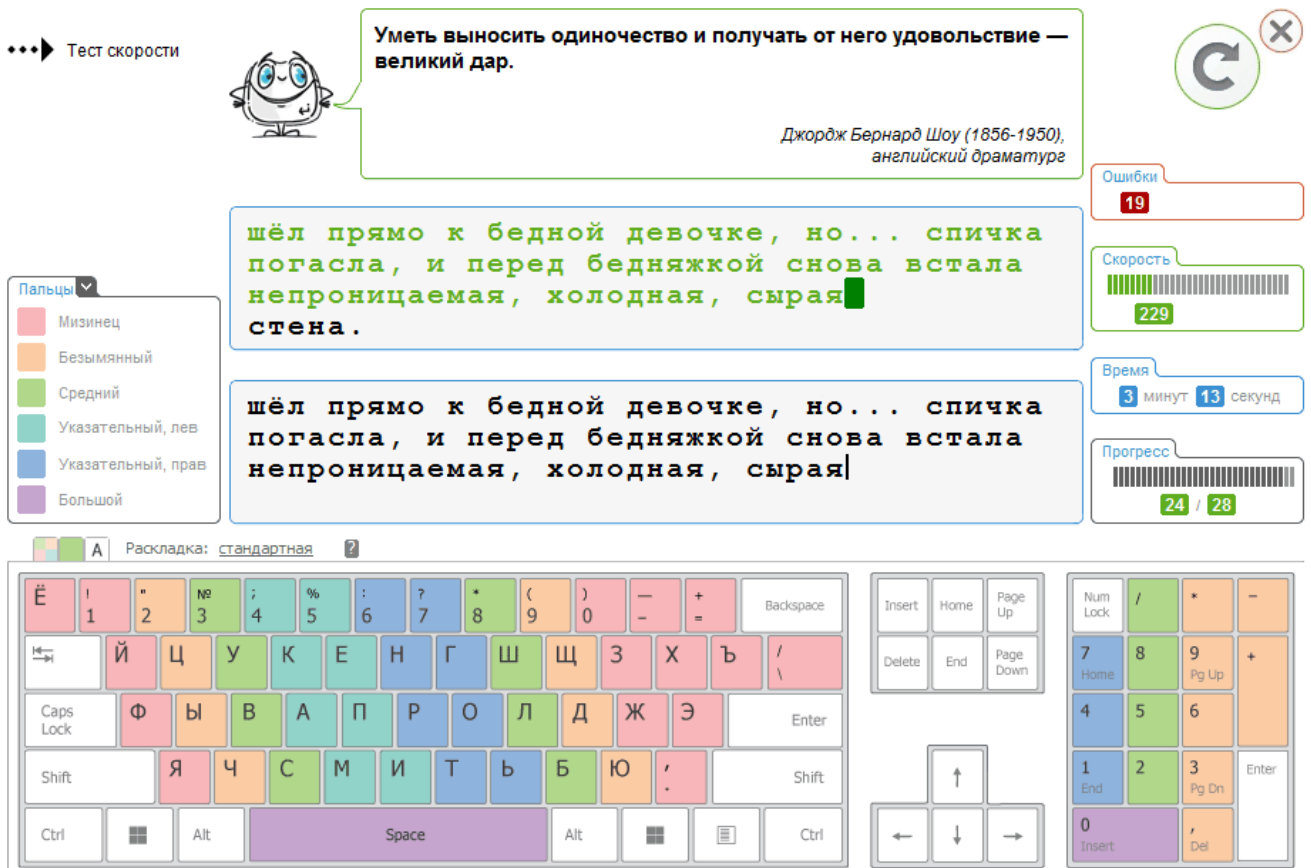


Рисунок 1.1. – Интерфейс «Соло на клавиатуре»

«VerseQ» (див. рисунок 1.2) – новое поколение колесь відомого клавиатурного тренажера «KeyTO». Програма платна, з можливістю безкоштовного тесту демо-версії. Режими - російська, англійська, німецька розкладки. Особливість її полягає в тому, що використовується нехарактерний для інших алгоритм навчання.

Розробники обіцяють, що вже через годину після початку занять ви зможете набирати текст наосліп, адже «VerseQ» вчить не поетапно освоювати пари клавіш, а орієнтації одночасно по всій клавіатурі. Перевагою тренажера є те, що під час набору не повторюються буквосполучення і фрази. Також вони побудовані за принципом реальних складів, що існують в російській мові, а не простого набору по типу «шгшгш». Плюсом є також те, що програма «запам'ятовує» зроблені вами помилки і частіше видає комбінації з тими ж поєднаннями для більшого тренування. Інтерфейс нейтральний, не відволікає. З мінусів можна відзначити тільки відсутність повної довідкової інформації по роботі з програмою, що робить її освоєння, в деякій мірі, інтуїтивним.

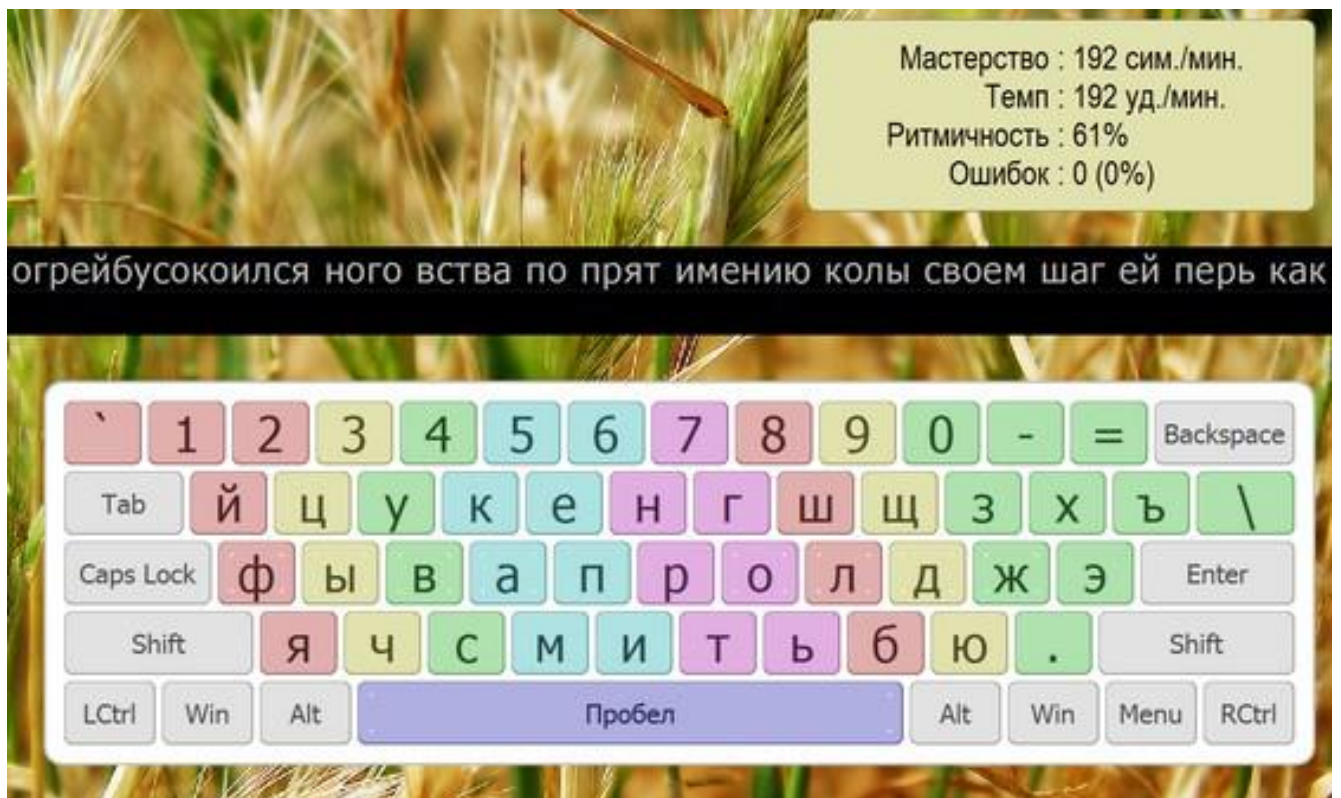


Рисунок 1.2. – Інтерфейс «VerseQ»

«Stamina» (див. рисунок 1.3) – безкоштовний клавіатурний тренажер, розроблений А. Казанцевим. Має підтримку англійської, російської та української розкладок. Інтерфейс налаштований максимально зручно для користувача. Навчання поетапне: спочатку уроки на заучування розташування і набору основних комбінацій, потім фрази. Користувач сам вибирає з чого починати і вільно

переміщається між режимами і уроками. Озвучування комусь здається смішним, комусь – безглуздом. Це справа смаку, тим більше опціонально настроюється. Плюси: зручність, легкість використання тренажера, зрозумілість, вбудована статистика, безкоштовність, можливість налаштувати програму під себе (завантажити фоновий малюнок і музику). З мінусів обивателі вказують на «терпимість» «Stamina» до помилок. виправлення не означає, що подібна помилка не буде зроблена в майбутньому.

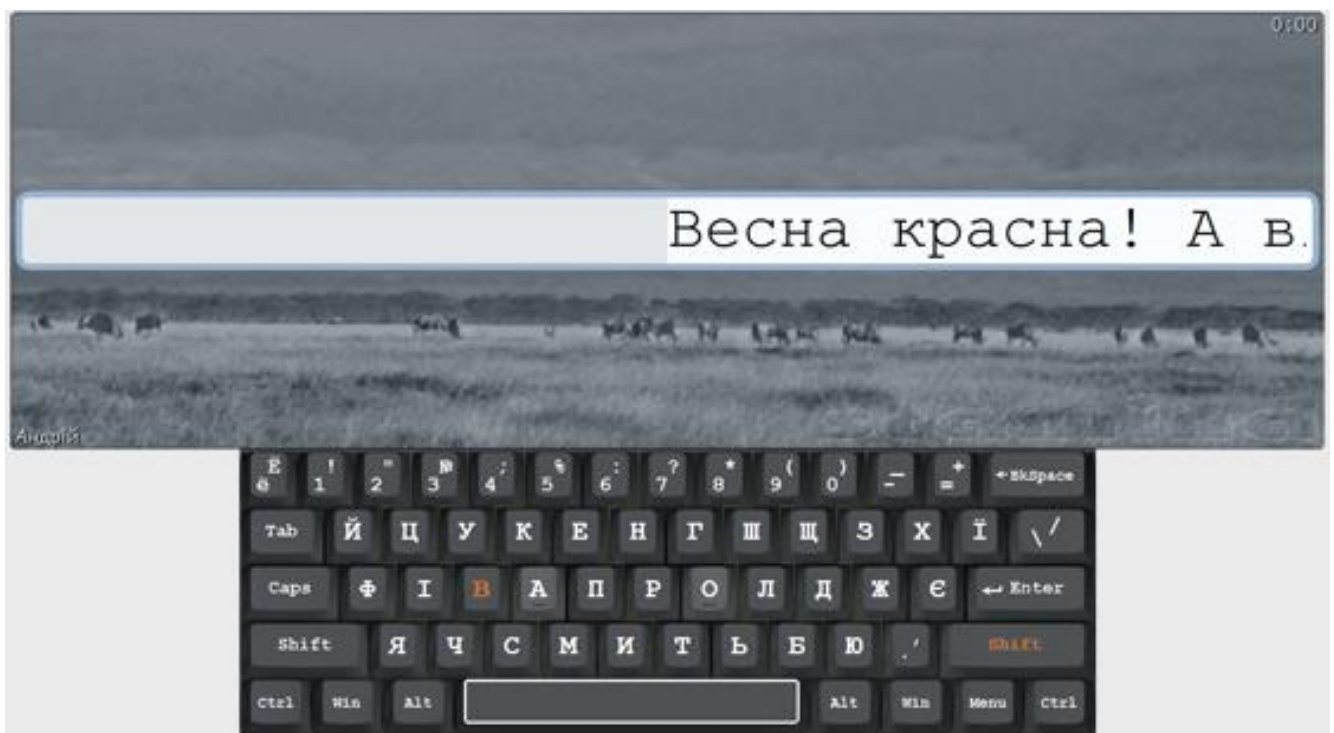


Рисунок 1.3. – Інтерфейс «Stamina»

«Typing Master» (див. рисунок 1.4) не має підтримки російської мови, але підходить як відмінний тренажер для людей бажаючих навчитися набору англійською, італійською, іспанською, німецькою та французькою мовами. Програма платна з можливістю безкоштовного тестування. Навчання структуровано, можливий вибір режиму: буквосполучення, речення, абзаци. З переваг відзначають автоматичне повторення матеріалу, якщо в ньому були допущені помилки, детальна статистика з аналізом клавіш, які викликають найбільші труднощі. Детально зроблений вступний урок, де пояснюється, яким пальцем натискати і на яку клавішу. З недоліків відзначимо строгість програми:

неможливо вибирати уроки, що робить тренажер схожим на підручник, адже попередній матеріал, спирається на вже пройдений.



Рисунок 1.3. – Інтерфейс «Typing Master»

Сліпий метод друку має безліч переваг: зниження психічної стомлюваності, збереження правильної постави та зору, а також набагато збільшується продуктивність набору тексту.

1.2 Код Морзе

Код Морзе (див. рисунок 1.4), «морзянка» (Азбукою Морзе код почав називатися тільки з першої світової війни) спосіб знакового кодування (представлення букв алфавіту, цифр, знаків пунктуації та інших символів послідовністю сигналів, наприклад, довгих і коротких: «тире» і «точок»). За одиницю часу приймається тривалість однієї точки. Тривалість тире дорівнює трьом точкам. Пауза між елементами одного знака - одна точка, між знаками в слові – 3 точки, між словами – 7 точок. Названий на честь американського винахідника і художника Семюеля Морзе.

International Morse Code

1. The length of a dot is one unit.
2. A dash is three units.
3. The space between parts of the same letter is one unit.
4. The space between letters is three units.
5. The space between words is seven units.

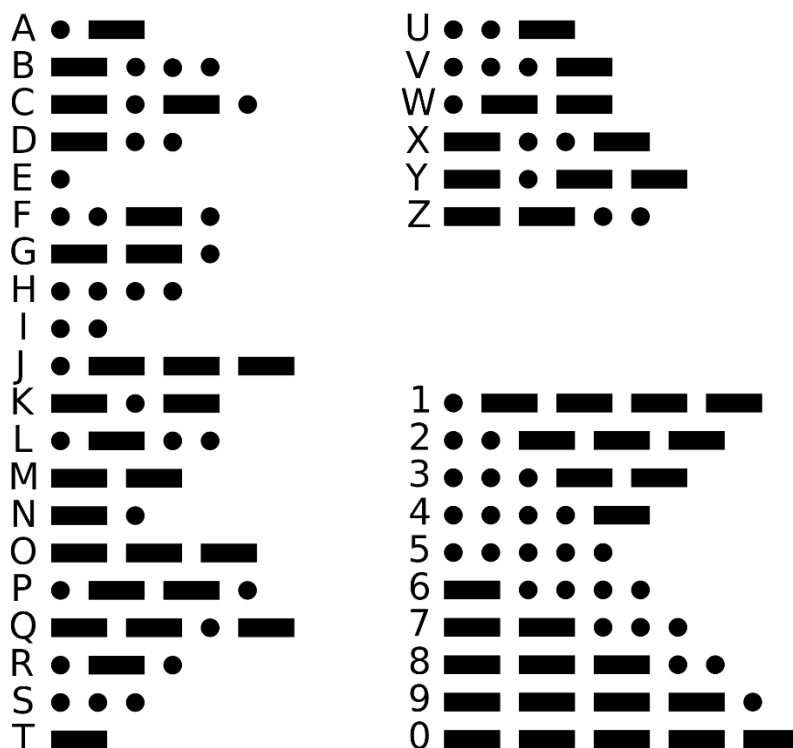


Рисунок 1.4. – Азбука Морзе

Літерні коди (власне «азбука») були додані колегою Морзе, Альфредом Вейлем – факт, який Морзе згодом всіляко заперечував (а заодно приписував собі винахід телеграфу як такого). Вейлем ж, можливо, була придумана і цифрова частина коду. А в 1848 році код Вейля/Морзе був вдосконалений німцем Фрідріхом Герке. Код, вдосконалений Герке, використовується до теперішнього часу [11].

SOS. Сигнал SOS заборонено подавати, якщо немає загрози для життя людей або судна на морі. SOS подається без пауз: між буквами: «... --- ...» (три точки, три тире, три крапки), тобто як одна довга буква. Хоча часто вважається, що SOS є аббревіатурою від «Save our souls» (врятуйте наші душі) або «Save our ship» (врятуйте наш корабель), насправді він був обраний через простоту передачі, до того ж передається не так як всі аббревіатури (окремими буквами), а єдиною буквою.

Для прискорення радіообміну широко використовуються абрєвіатури, спеціальні «Q -коди» і багаточисленні жаргонні висловлювання.

Принцип кодування азбуки Морзе виходить з того, що букви, які найчастіше вживаються в англійській мові, кодуються простішими поєднаннями точок і тире. Це робить освоєння абетки Морзе простіше, а передачі – компактніше.

У наш час, як правило, використовують більш сучасні засоби зв'язку. Азбуку Морзе іноді застосовують на флоті і в МНС. Вона дуже популярна серед радіоаматорів.

«Морзянка», швидше за все, завжди буде актуальною, адже це найдоступніший і простий спосіб зв'язку. Приймати сигнал можна на далеких відстанях і в умовах сильних радіоперешкод, кодувати послання можна вручну, а запис і відтворення відбувається за допомогою найпростіших пристроїв. Таким чином, азбука Морзе не підведе в надзвичайній ситуації, якщо з ладу вийди більш складне обладнання.

1.3 Постановка завдання проектування

Метою проекту є розробка програмного модуля, що реалізує алгоритм вводу тексту, перевірку його на помилки, підрахунок помилок, обчислення швидкості набору тексту та має інтуїтивно зрозумілий інтерфейс. Програмний модуль повинен бути написаний на одній з найбільш розповсюджених мов програмування, а його код має бути максимально зрозумілий та розбірливий.

Відповідно до мети, в проєкті поставлені наступні завдання:

1. Проаналізувати програми типу «Клавіатурний тренажер».
2. Обрати мову та середовище для реалізації програмного модуля.
3. Визначити функціональні та нефункціональні вимоги до модуля.
4. Розробити структуру програмного модуля.
5. Розробити доповнюючі схеми та діаграми..
6. Розробити інтерфейс програмного модуля.
7. Реалізувати 3 режими навчання набору тексту.
8. Реалізувати додатковий функціонал інтерфейсу.
9. Виконати тестування програмного модуля.

10. Визначити технічні вимоги програмного модуля.
11. Розробити керівництво користувача.
12. Викласти вимоги до робочого місця техника-програміста.
13. Провести розрахунок освітлення виробничого приміщення.
14. Провести розрахунок рівня шуму на робочому місці.

1.4 Висновки до розділу

1. Проаналізовано програми типу «Клавіатурний тренажер». Виявлено переваги та недоліки існуючих клавіатурних тренажерів. Головними недоліками визнано те, що переважна більшість з них платні, мають лише по одному режиму навчання, в деяких неможливо обирати уроки та завантажувати користувацькі тексти для набору. Перевагами проаналізованих клавіатурних тренажерів є: наявність декількох клавіатурних розкладок, функція переважної генерації комбінацій символів, в яких було допущено більше всього помилок, детальний вступний урок, де пояснюється який палець повинен натискати кожен клавішу. На основі аналізу сформульовано попередній список вимог до готового продукту.

2. Для реалізації програмного модуля обрано мову програмування C++ та середовище розробки Microsoft Visual Studio Express 2013.

РОЗДІЛ 2. СТРУКТУРА ПРОГРАМНОГО МОДУЛЯ

2.1 Функціональні вимоги

До програмного модуля поставлено наступні функціональні вимоги:

1. У модулі повинно бути реалізовано три окремих режими навчання:
 - Набір тексту посимвольно з ігноруванням неправильно набраних символів;
 - Набір тексту по слову з можливістю виправляти помилки, допущені при наборі окремого слова;
 - Навчання коду Морзе, шляхом набору попередньо закодованого тексту.
2. Модуль повинен мати можливість відображення статистичної інформації користувача. Такої як:
 - Підрахунок кількості символів, які залишилось набрати і всього в тексті;
 - Підрахунок кількості помилок, допущених в процесі набору;
 - Відсоток помилок по відношенню до всього тексту;
 - Скільки символів набирається в хвилину;
 - Скільки символів набирається в секунду;
 - Скільки пройшло часу з моменту початку набору тексту.
3. Модуль повинен мати функцію, що буде забезпечувати зчитування тексту для набору з будь-якого текстового файлу, в тому числі з файлу створеного користувачем.
4. Програмний модуль повинен забезпечувати коректну перевірку введених користувачем символів у відповідності до поточного тексту для набору.
5. Програмний модуль повинен мати функцію для забезпечення поступового зсуву тексту, що набирається, по мірі вводу користувачем правильних символів.
6. Програмний модуль повинен мати функцію для перевірки окремо набраного слова на відповідність тексту, з неможливістю продовжувати подальший

| | | | | | | | |
|---------------------|-----------------|--|--|---|---------------|--------------|----------------|
| Кафедра КСУ | | | | НАУ 21 11 30 000 ПЗ | | | |
| <i>Виконав</i> | Таран І.А. | | | СТРУКТУРА ПРОГРАМНОГО МОДУЛЯ | <i>Літера</i> | <i>Аркуш</i> | <i>Аркушів</i> |
| <i>Керівник</i> | Марченко Н.Б. | | | | | 16 | 60 |
| <i>Консульт.</i> | | | | | 123 СП-437 | | |
| <i>Норм. контр.</i> | Тупога Є.В. | | | | | | |
| <i>Зав.каф.</i> | Литвиненко О.Є. | | | | | | |

набір тексту, якщо введене слово було написано з помилкою.

7. Програмний модуль повинен забезпечувати коректне відтворення спеціального фонового звукового файлу.

8. Програмний модуль повинен мати спеціальну функцію для підрахунку помилок, допущених користувачем при наборі тексту.

9. Програмний модуль повинен мати функцію що буде фіксувати час, затрачений користувачем на набір всього тексту для набору та окремих його символів.

10. Програмний модуль повинен забезпечувати коректне завершення програми при натисненні кнопки «Exit» у головному меню або при натисненні кнопки у правому верхньому куті вікна.

2.2 Нефункціональні вимоги

Вимоги до інтерфейсу:

1. Інтерфейс має бути інтуїтивно зрозумілим, приємним на око та не відволікати користувача від набору тексту.

2. Інтерфейс повинен мати кнопку виклику «Info» в якій міститься інформація про програмний модуль та інструкцію користування.

3. Інтерфейс програмного модуля повинен мати кнопку виклику довідкового зображення щодо розташування пальців на клавіатурі, коду Морзе.

4. У головному меню програмного модуля має міститися пункт «Open», який містить такі підпункти як «Open file» та «Open lesson».

5. У підпункті «Open lesson» має міститися каталог уроків та вправ, які входять в урок.

6. У головному меню програмного модуля має міститися пункт «Sound», в якому описані такі підпункти як «On» та «Off».

7. У головному меню програмного модуля має міститися пункт «Save statistics»

Системні вимоги:

1. Програмний модуль повинен коректно працювати на операційній системі Windows 8.0/8.1/10.

2. Готовий програмний модуль з вихідними кодами повинен займати не більше 250 МБ дискового простору.

3. Модуль повинен бути портативним, тобто не потребувати встановлення (не стосується встановлення додаткових програм).

4. Модуль повинен коректно працювати на апаратному забезпеченні з обмеженими ресурсами.

2.2 Структура програмного модуля

Структура програмного модуля відображує наступні вимоги:

1. Вибір режиму навчання на головному вікні програми.
2. Вибір тексту для набору: вбудований чи користувачський.
3. Відображення статистичної інформації.
4. Збереження статистичної інформації.
5. Відтворення звуку.

Структурна схема програмного модуля зображена на наступному малюнку (див. рисунок 2.1):

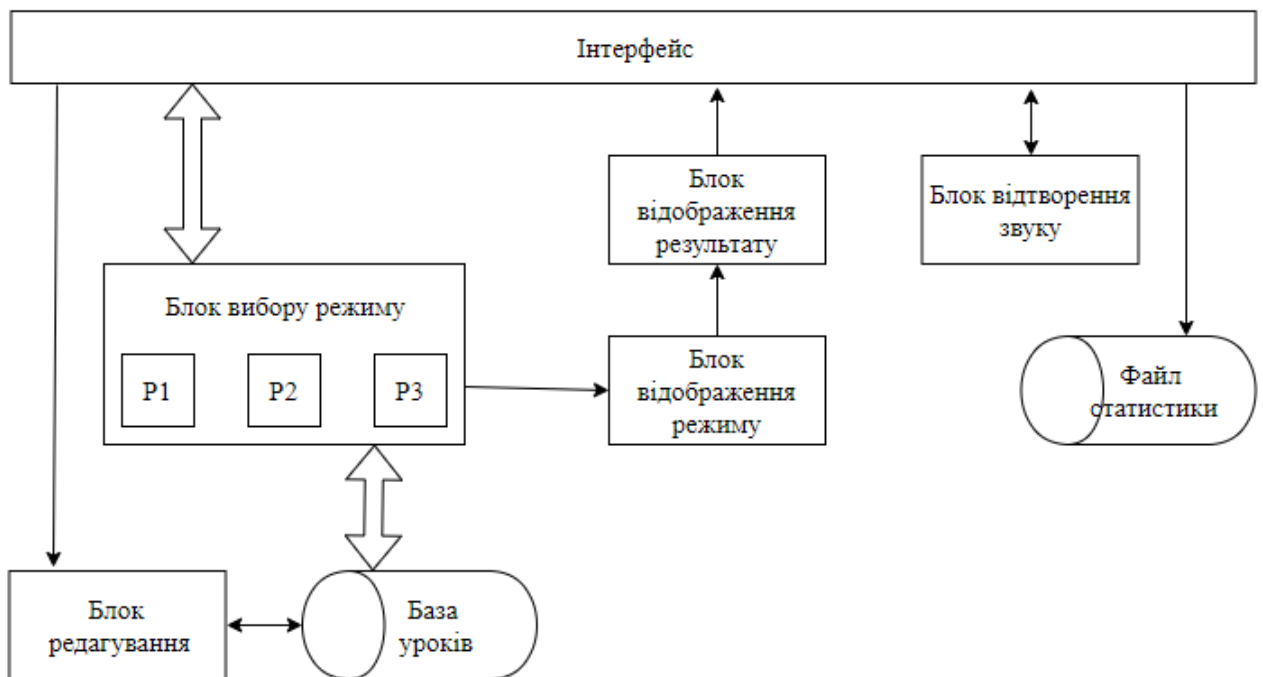


Рис. 2.1 – Структурна схема програмного модуля

Користувачу надається вибір режиму навчання, але за замовчуванням

вибирається перший режим (Mode 1). Можна вибрати вбудований урок, або завантажити власний текст. Є можливість вимкнути, а потім увімкнути фонову мелодію. Наявно безліч вікон з довідковою інформацією, такою як: «Про програму», «Як користуватися програмою», «Позиція рук під час друку», «Азбука Морзе», «Спеціальні символи азбуки Морзе». Користувач може вільно пересуватися між вправами та уроками за допомогою кнопок «Наступна вправа», «Наступний урок», «Попередня вправа» та «Попередній урок». Щоб зупинити відлік часу на таймері посередині вправи потрібно натиснути кнопку «Зупинити вправу». Щоб зберегти статистику потрібно натиснути на кнопку «Зберегти статистику». Для виходу з програми користувач може натиснути кнопку «Вихід».

Діаграма варіантів використання зображена на наступному малюнку (див. рисунок 2.2):

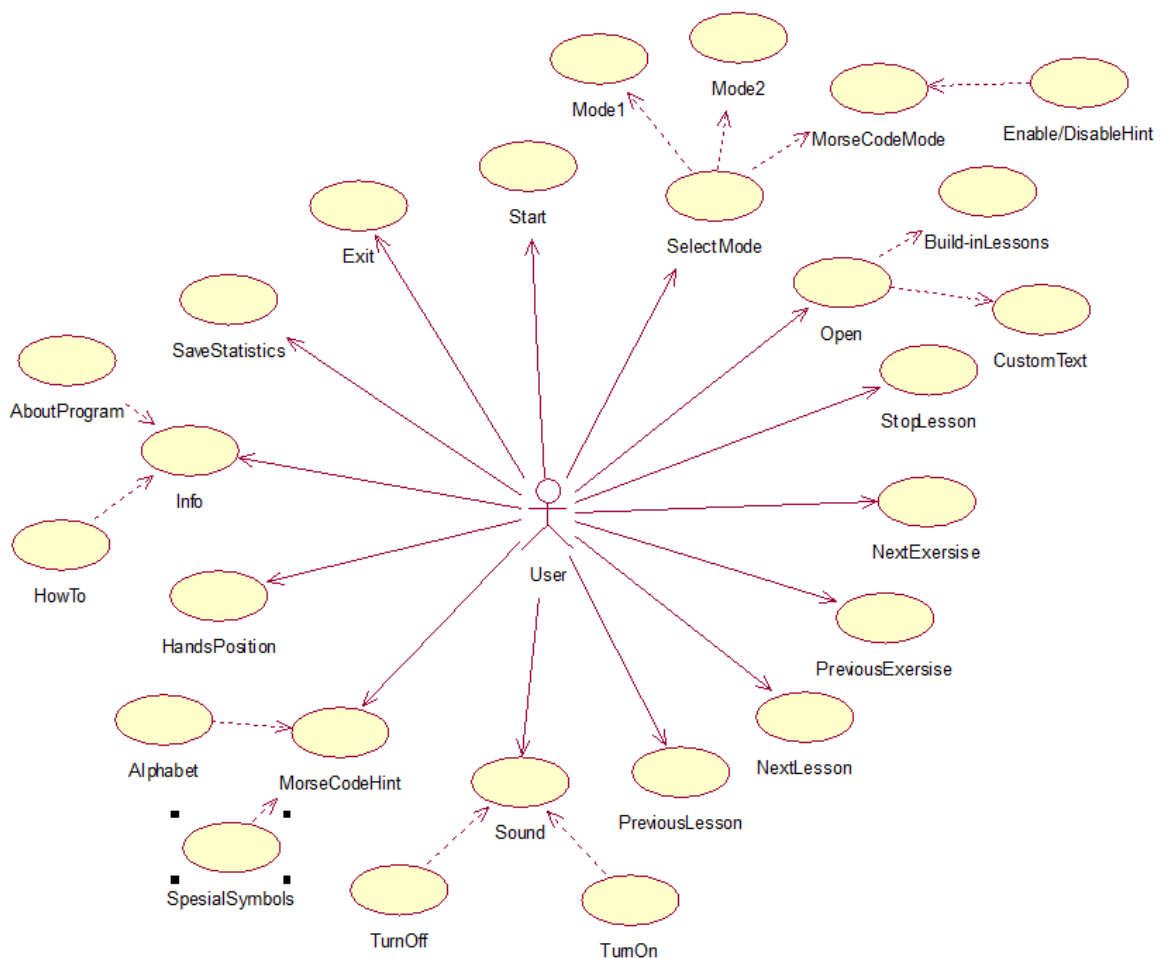


Рис. 2.2 – Діаграма варіантів використання

Функціональна схема першого режиму навчання зображена на рисунку 2.3. Після того як обрано звідки завантажити вправу, рядок завантажується до змінної та в TextBox. Очікується поки користувач натисне клавішу. Зчитується перший символ рядка та порівнюється з натиснутою клавішею. Якщо натиснуто неправильну клавішу, вона не відображується на екрані і змінна, яка уособлює помилки, зростає на одиницю. Якщо натиснуто правильну клавішу, символ відображується на екрані та очікується на введення наступний символ.

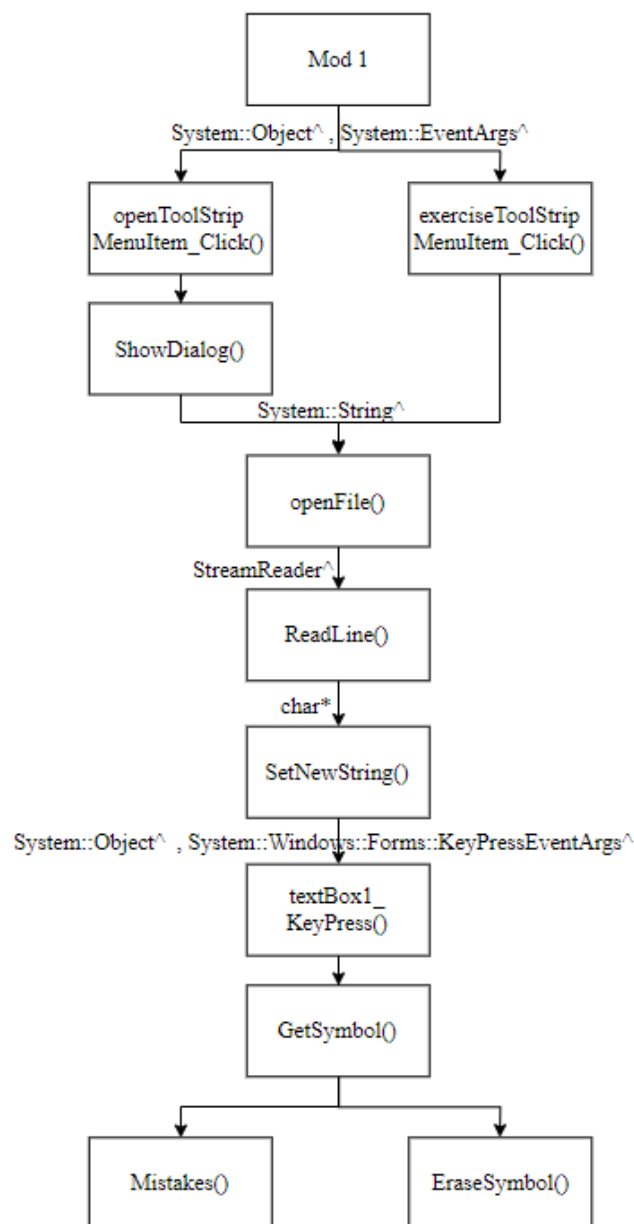


Рисунок 2.3 – Функціональна схема першого режиму навчання

Функціональна схема другого режиму навчання зображена на рисунку 2.4.

Після того як обрано звідки завантажити вправу, рядок завантажується до масиву рядків окремими словами та в один TextBox текстом в повному об'ємі, а в інший слово, яке потрібно набирати в даний момент. Очікується поки користувач введе все слово разом, а потім натисне «Space». Зчитується перше слово рядка та порівнюється з набраним словом. Якщо набрано неправильне слово, воно змінює колір на червоне і очікує коли користувач виправить помилку та натисне «Space» і змінна, яка уособлює помилки, зростає на одиницю. Якщо набрано неправильне слово, воно переноситься до наступного TextBox, який уособлює вже набраний текст. До TextBox завантажується наступне слово для набору.

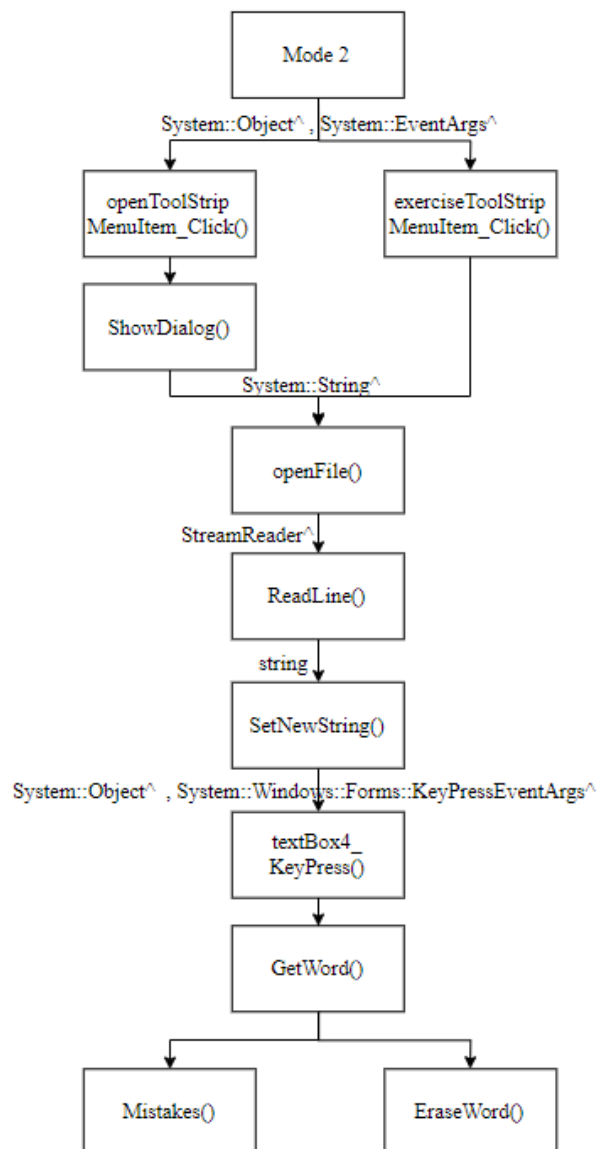


Рисунок 2.4 – Функціональна схема другого режиму навчання

Функціональна схема режиму навчання коду Морзе зображена на рисунку 2.5. Після того як обрано звідки завантажити вправу, рядок завантажується до змінної та в TextBox, цей же рядок кодується азбукою Морзе та записується в інший TextBox. Очікується поки користувач натисне одну з трьох клавіш, а саме: «.», «-» або «Space». Зчитується перший символ рядка та порівнюється з натиснутою клавішею. Якщо натиснуто неправильну клавішу, вона не відображується на екрані і змінна, яка уособлює помилки, зростає на одиницю. Якщо натиснуто правильну клавішу, символ відображується на екрані та очікується на введення наступний символ. Коли введено три символи «Space» поспіль, раніше набрані символи декодуються і відбувається зсув тексту на один символ.

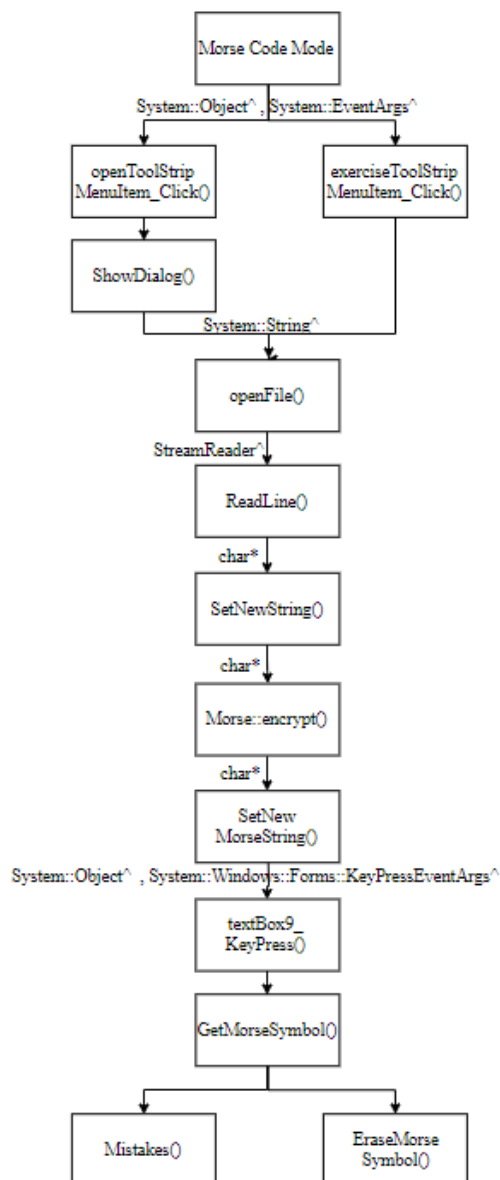


Рисунок 2.5 – Функціональна схема режиму навчання коду Морзе

2.3 Висновки до розділу

1. Визначено функціональні вимоги до модуля, основними з яких є реалізація режимів навчання, можливість відображення статистичної інформації користувача, коректне відтворення спеціального фонового звукового файлу, можливість підрахунку помилок, фіксування часу, затраченого користувачем на набір всього тексту та окремих його символів, перевірка окремого набраного слова на відповідність тексту. Визначено нефункціональні вимоги, які представляють собою вимоги до інтерфейсу та системні вимоги для коректної роботи програмного модуля.

2. На основі сформульованих вимог розроблено структуру, що відображає основні функціональні можливості програмного модуля.

3. Розроблено ряд доповнюючих схем та діаграм, а саме діаграму варіантів використання, яка є вихідним концептуальним представленням системи в процесі її проектування та розробки, а також функціональні схеми, у вигляді яких представлені всі режими навчання, реалізовані в програмному модулі.

РОЗДІЛ 3. РОЗРОБКА ПРОГРАМНОГО МОДУЛЯ

3.1 Покрокова розробка

В процесі розробки програмного модуля використано середовище розробки Microsoft Visual Studio Express 2013 для Windows Desktop.

У першу чергу створюється пустий проект CLR, який називається «FastFingers» (див. рисунок 3.1):

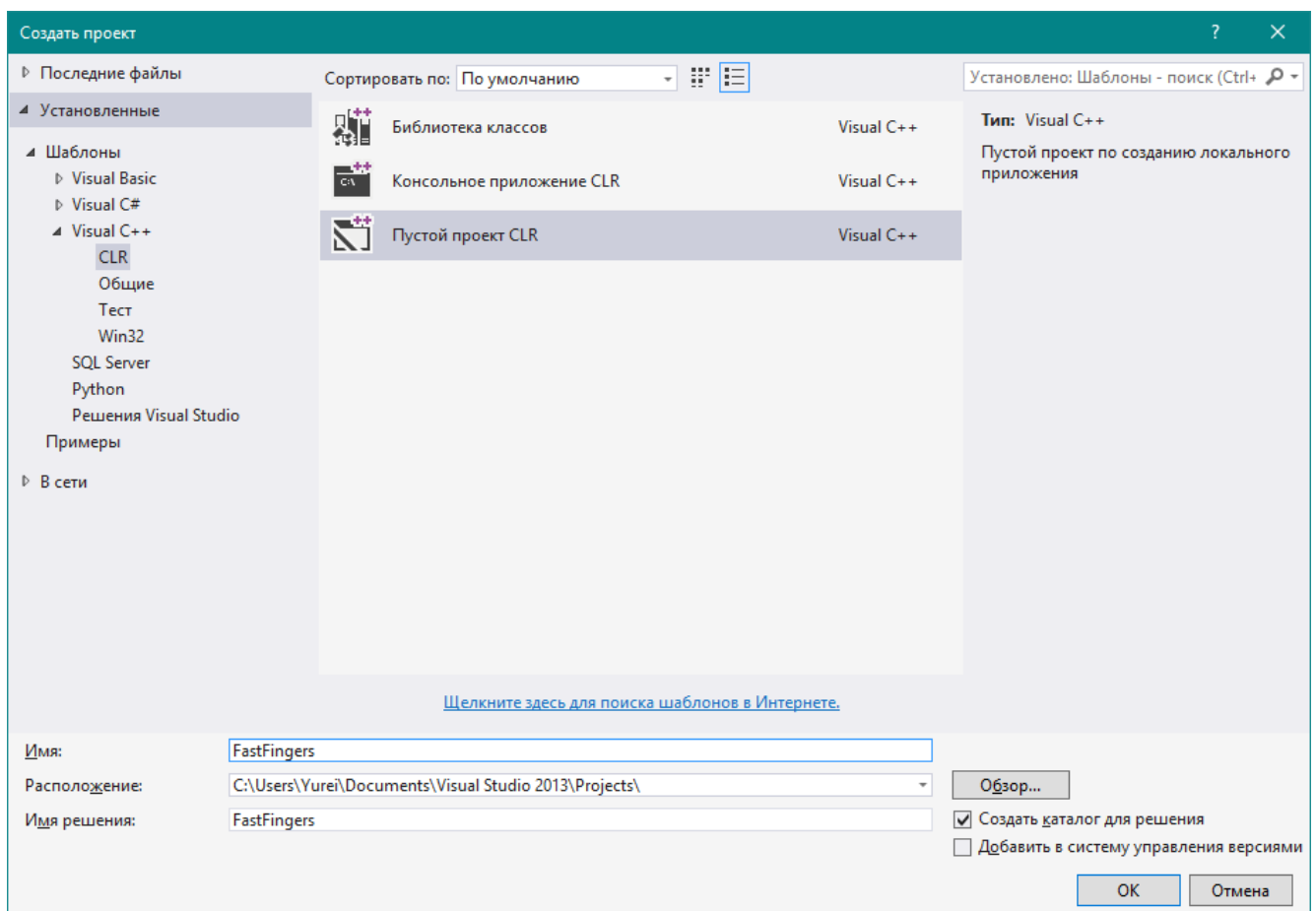


Рисунок 3.1 – Створення нового проекту

| | | | | | | | |
|---------------------|----------------|--|--|--|---------------|--------------|----------------|
| Кафедра КСУ | | | | НАУ 21 11 30 000 ПЗ | | | |
| <i>Виконав</i> | Таран І.А. | | | РОЗРОБКА ПРОГРАМНОГО МОДУЛЯ | <i>Літера</i> | <i>Аркуш</i> | <i>Аркушів</i> |
| <i>Керівник</i> | Марченко Н.Б. | | | | | 24 | 60 |
| <i>Консульт.</i> | | | | | 123 СП-437 | | |
| <i>Норм. контр.</i> | Тупота Є.В. | | | | | | |
| <i>Зав.каф</i> | Литвиненко О.Є | | | | | | |

Далі до порожнього проекту через «Solution Explorer» додається елемент користувацького інтерфейсу «Форма Windows Forms». За замовчуванням, ім'я елемента «MyForm.h» (див. рисунок 3.2):

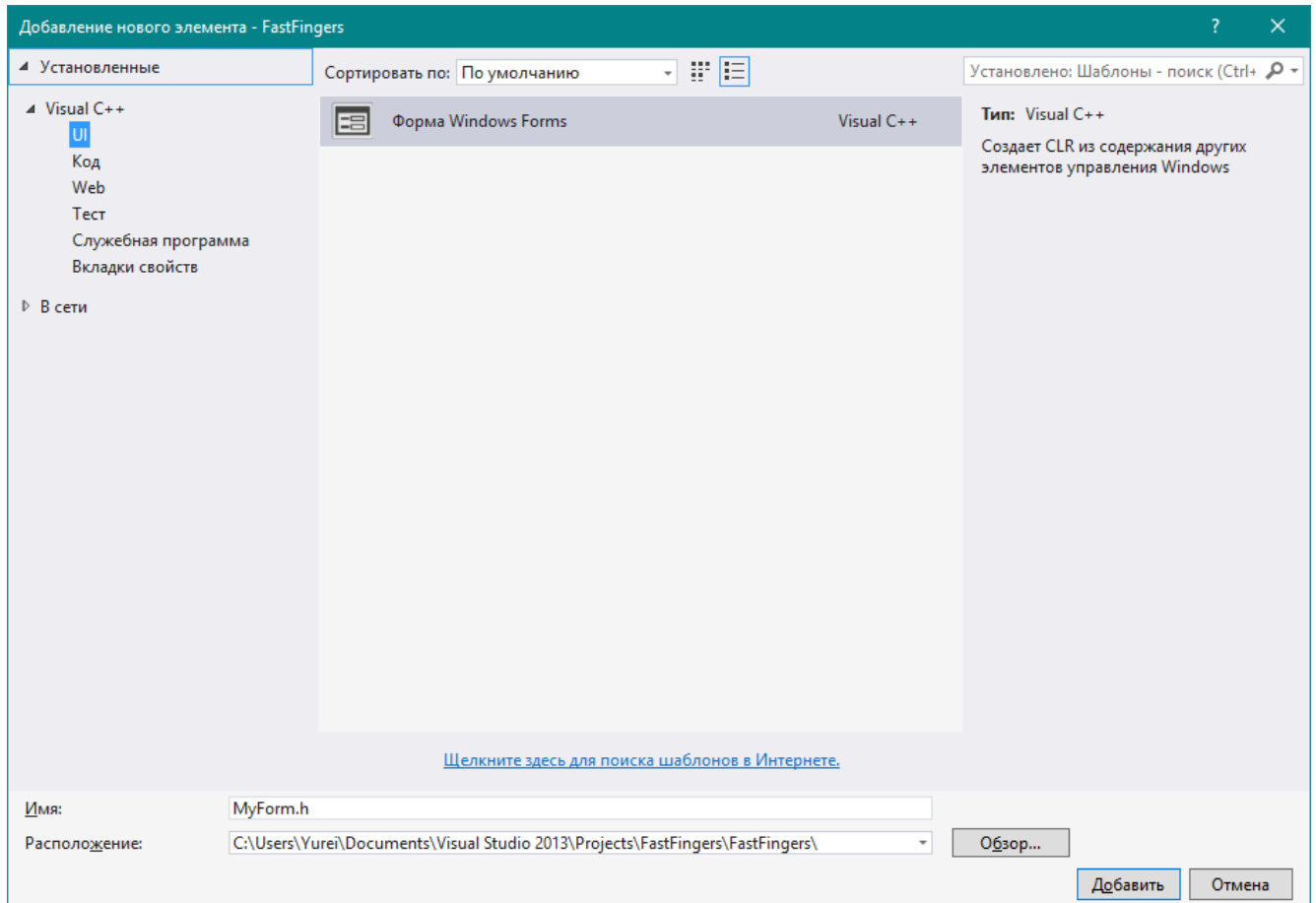


Рисунок 3.2 – Створення елементу користувацького інтерфейсу

Щоб форма працювала коректно, її необхідно налаштувати. Для цього в головному вікні натиснути «Отладка» та «Свойства FastFingers». Відкривається нове вікно «Страницы свойств FastFingers» (див. рисунок 3.3):

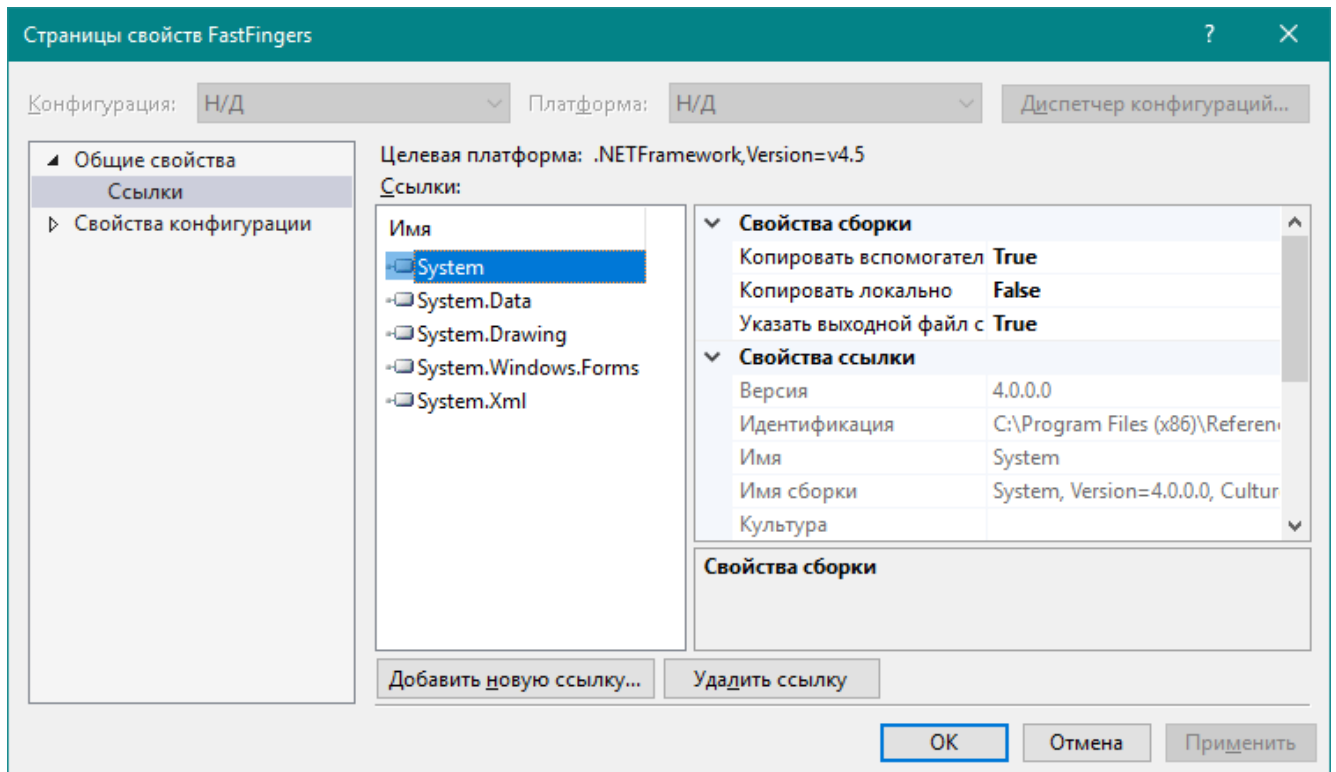


Рисунок 3.3 – Вікно властивостей проекту FastFingers

Далі обирається підсистема «Windows (/SUBSYSTEM:WINDOWS)» на вкладці «Компоновщик», натискається кнопка «Применить» (див. рисунок 3.4):

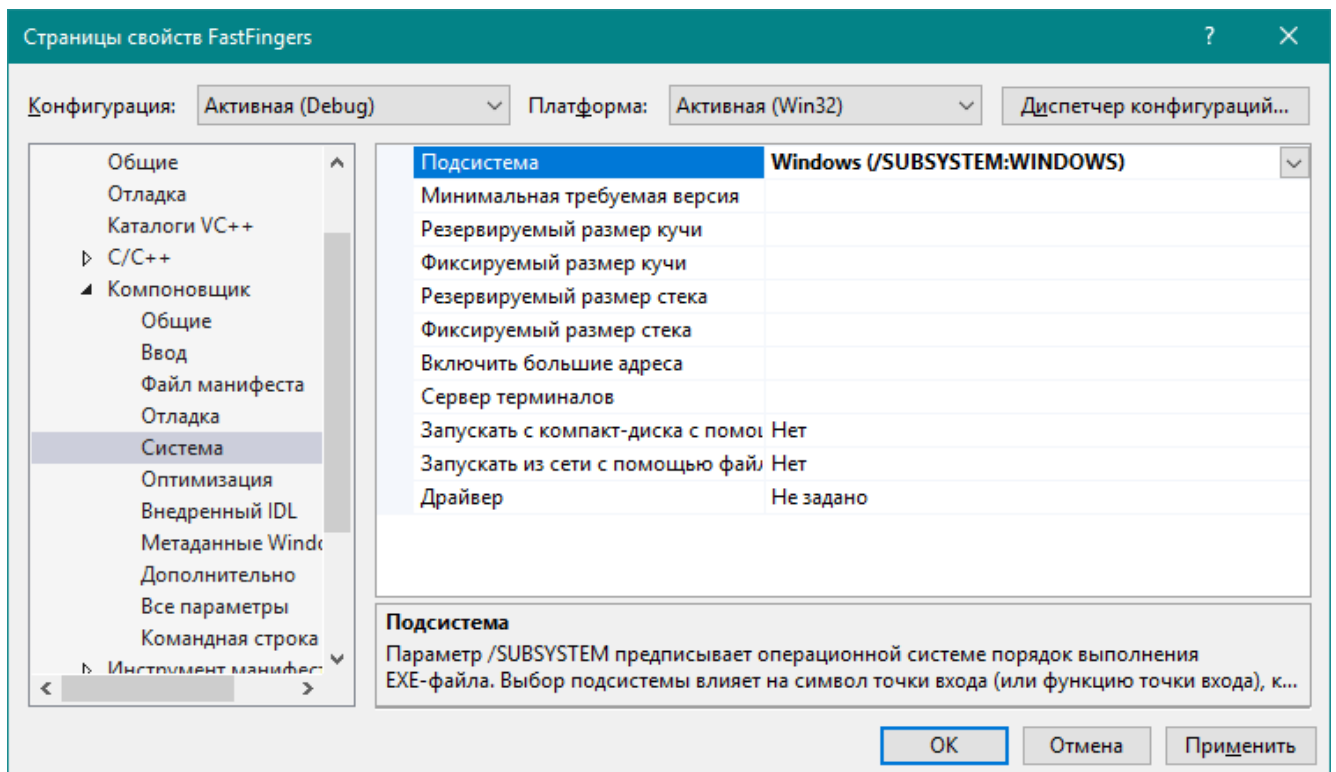


Рисунок 3.4 – Обрана підсистема

Далі визначається точка входу. Для цього потрібно перейти до пункту «Дополнительно» і вибрати в якості точки входу «Main» (див. рисунок 3.5):

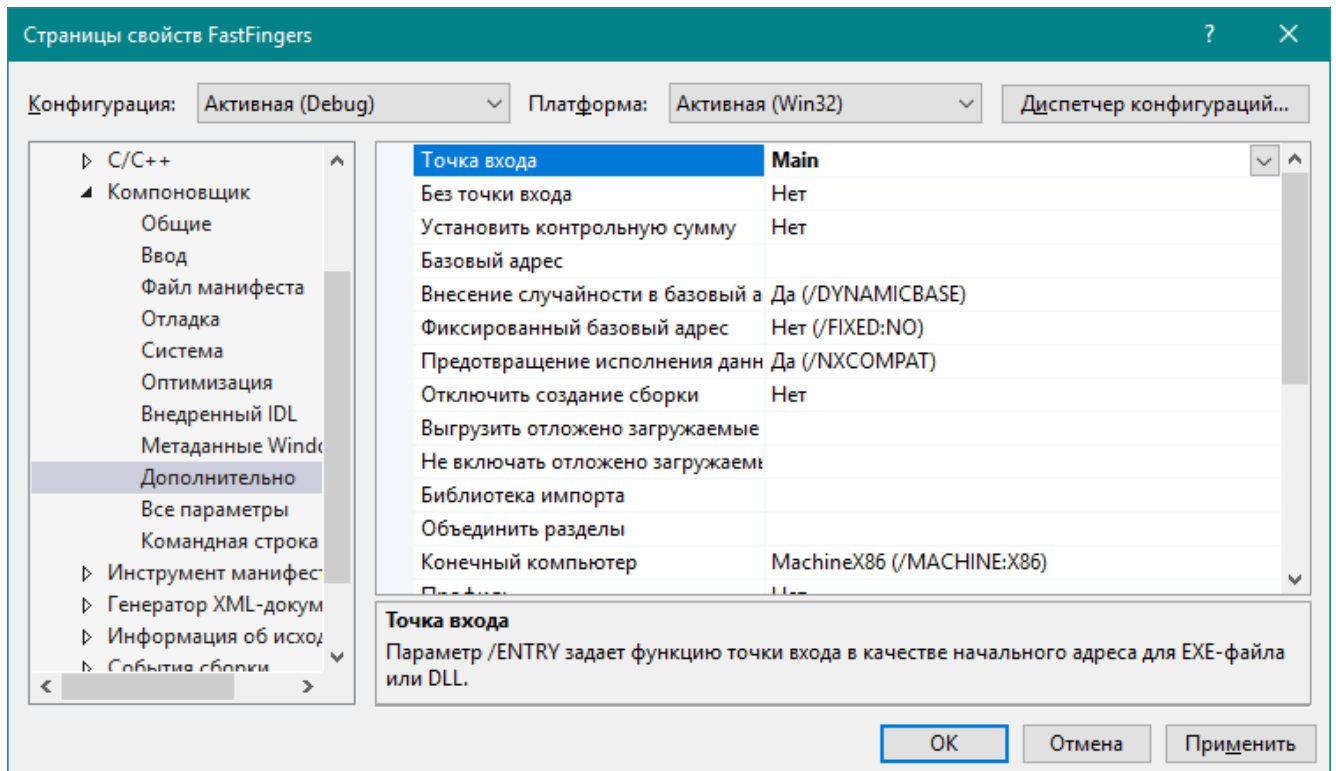


Рисунок 3.5 – Обрана точка входу

Наступним кроком в «Solution Explorer» обирається файл вихідного коду «MyForm.cpp». До нього записуються такі рядки коду:

```
#include "MyForm.h"

using namespace System;

using namespace System::Windows::Forms;

[STAThreadAttribute]
void Main(array<String^>^ args)
{
    Application::EnableVisualStyles();
    Application::SetCompatibleTextRenderingDefault(false);
    FastFingers::MyForm form;
    Application::Run(%form);
}
```

Це головна функція форми та за сумісництвом точка входу, тобто після запуску програми ця функція виконується у першу чергу.

На цьому базові налаштування закінчені.

В «Solution Explorer» обирається заголовочний файл «MyForm.h». Відкривається «MyForm.h [Конструктор]»(див. рисунок 3.6):

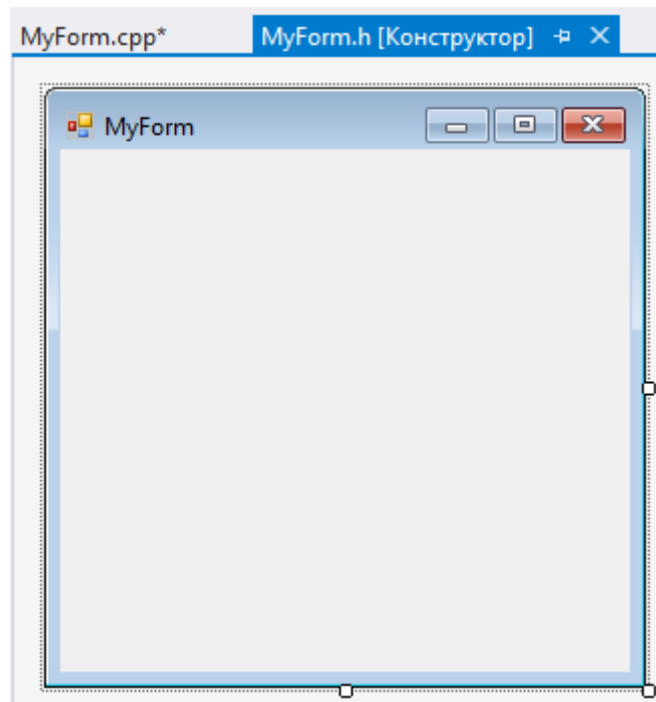


Рисунок 3.6 – Первинний вигляд форми

На формі розміщуються всі елементи, а саме: «MenuStrip», «OpenFileDialog», «Timer», «TabControl», «TextBox», «GroupBox», «Label», «Button», «CheckBox» (див. рисунок 3.7).

«MenuStrip» - для параметрів, які згруповано по функціональності.

«OpenFileDialog» - для надання можливості користувачу обирати та завантажувати власні вправи.

«Timer» - для відліку часу та оновлення статистичної інформації на екрані через деякий проміжок часу.

«TabControl» - для розміщення на сторінках трьох різних режимів навчання.

«TextBox» - для роботи з вправами.

«GroupBox» - для групування статистичної інформації.

«Label» - для відображення статистичної інформації.

«Button» - для маневрування між уроками та вправами.

«CheckBox» - для відображення підказки при навчанні коду морзе.

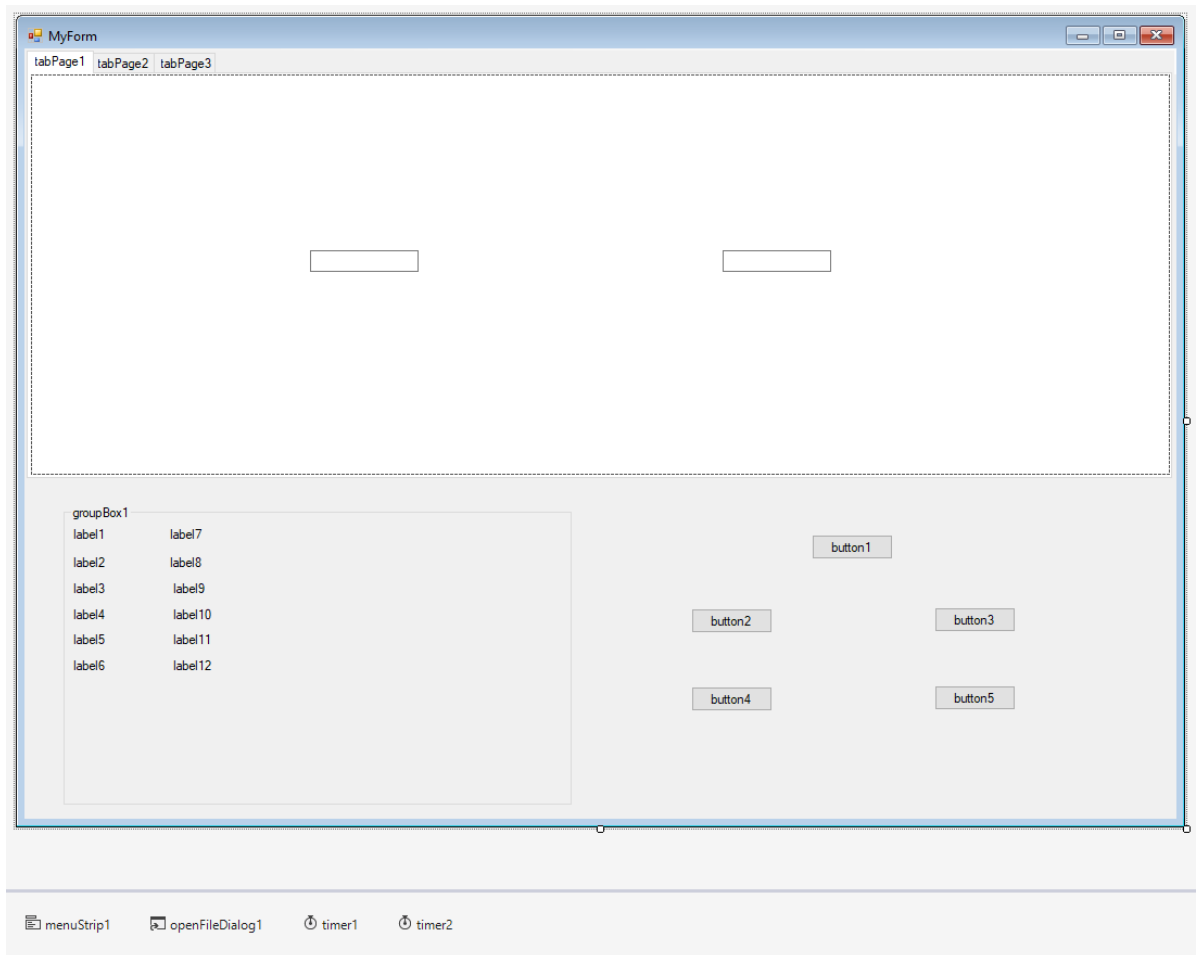


Рисунок 3.7 – Розміщення елементів на формі

Далі змінюється властивість форми «Text» з «MyForm» на «FastFingers» (див. рисунок 3.8).

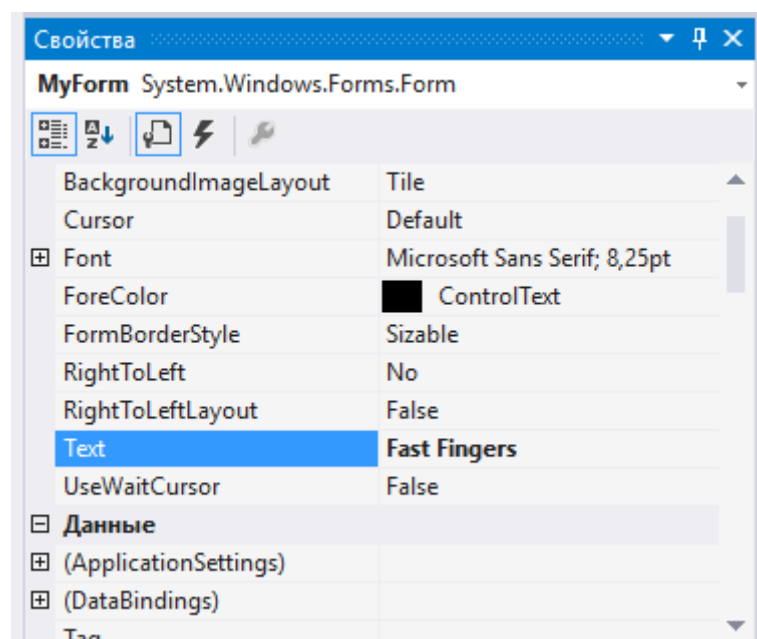


Рисунок 3.8 – Властивість форми «Text»

До форми під'єднується елемент «menuStrip1» (див. рисунок 3.9). Для цього він вибирається у випадному списку навпроти властивості форми «MainMenuStrip». Цей елемент потрібен для оформлення головного меню програми.

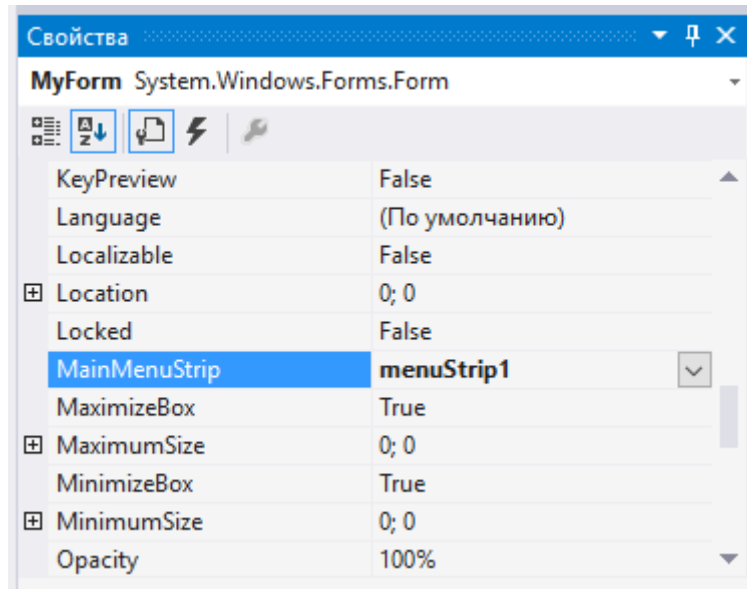


Рисунок 3.9 – Властивість форми «MainMenuStrip»

Наступним кроком оформлюється головне меню програмного модуля (див. рисунок 3.10). Додано такі пункти як «Open», «Hands position», «Morse code hint», «Sound», «Info», «Save statistics» та «Exit».

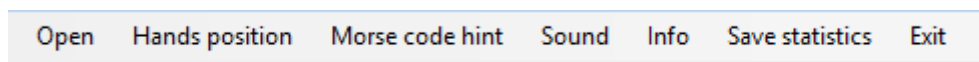


Рисунок 3.10 – Пункти головного меню програми

Підпунктами «Open» записуються «Open File» та «Open Lesson». «Open File» для обрання користувацького тексту для навчання. В свою чергу, «Open Lesson» для обрання користувачем уроку та вправи для виконання (див. рисунок 3.11):

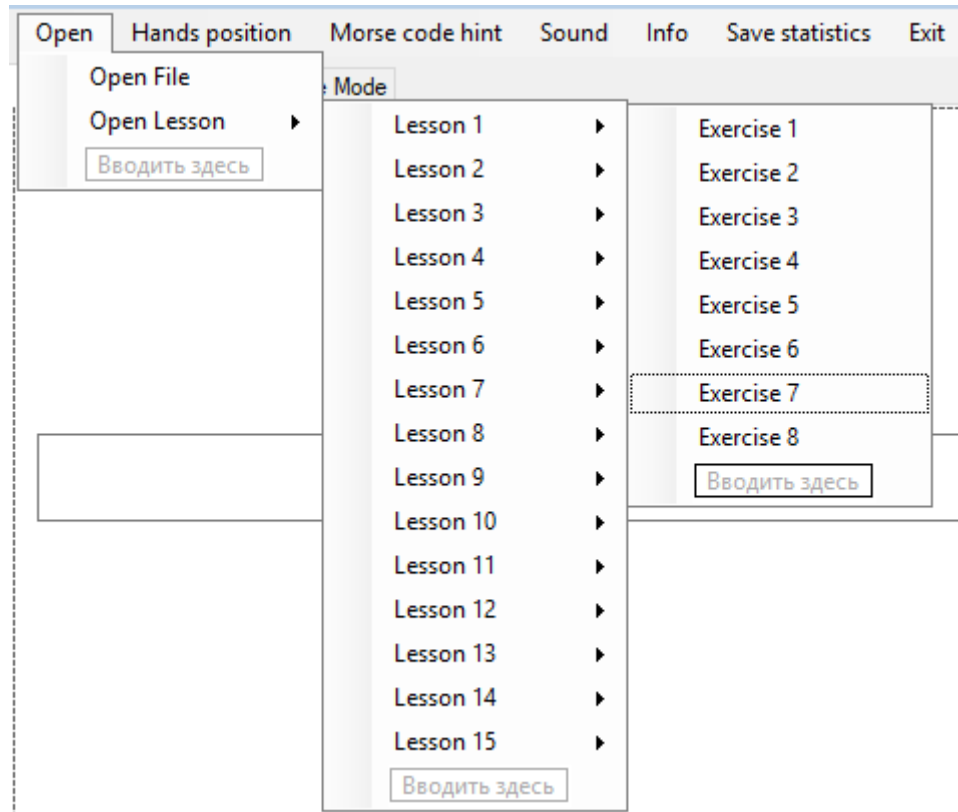


Рисунок 3.11 – Підпункти елемента головного меню «Open»

Підпунктами «Morse code hint» є «Alphabet» та «Special symbols» (див. рисунок 3.12):

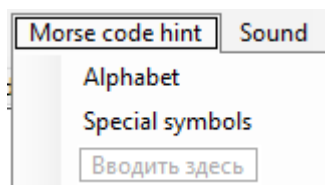


Рисунок 3.12 – Підпункти елемента головного меню «Morse code hint»

Підпунктами «Sound» є «On» та «Off» для увімкнення та вимкнення фонового аудіозапису (див. рисунок 3.13):

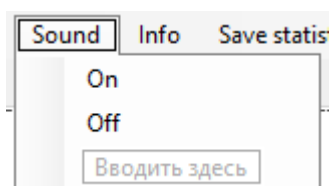


Рисунок 3.13 – Підпункти елемента головного меню «Sound»

Підпунктами «Info» є «How to» та «About program» для довідкової інформації як користуватися програмою та про програму (див. рисунок 3.14):

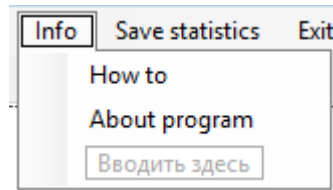


Рисунок 3.14 – Підпункти елемента головного меню «Info»

На кожному «TextBox» встановлюється «Microsoft Sans Serif» звичайний 24-го розміру (див. рисунок 3.15):

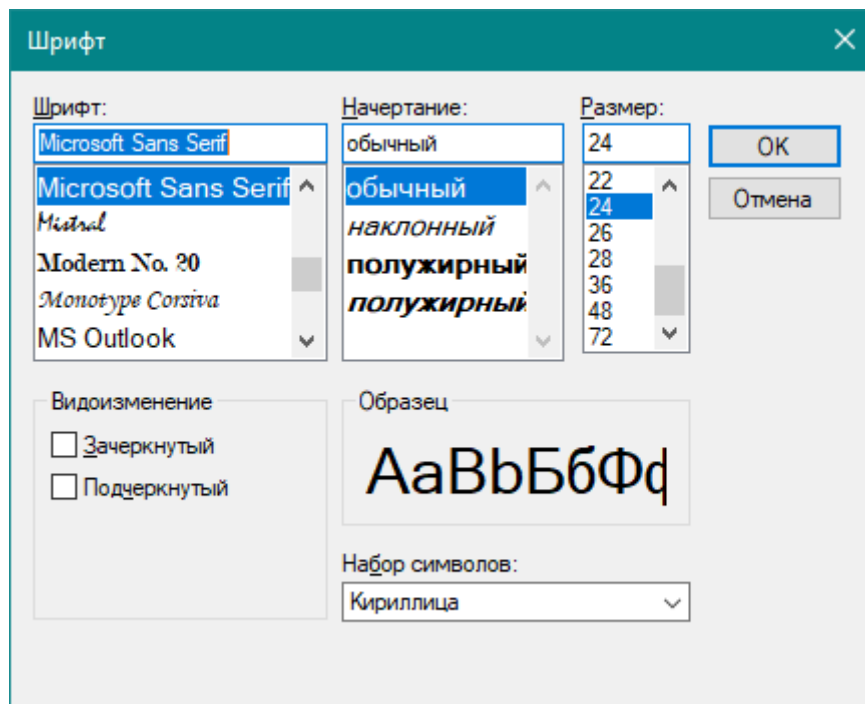


Рисунок 3.15 – Вікно налаштування шрифту

Називаються три вкладки елемента «TabControl» – «Mode 1», «Mode 2» та «Morse code mode» відповідно. П'ять кнопок – «Stop lesson», «Previous exercise», «Next exercise», «Previous lesson» та «Next lesson». Підписуються дванадцять «Label» для відображення статистичної інформації. Вигляд форми після називання усіх елементів (див. рисунок 3.16):

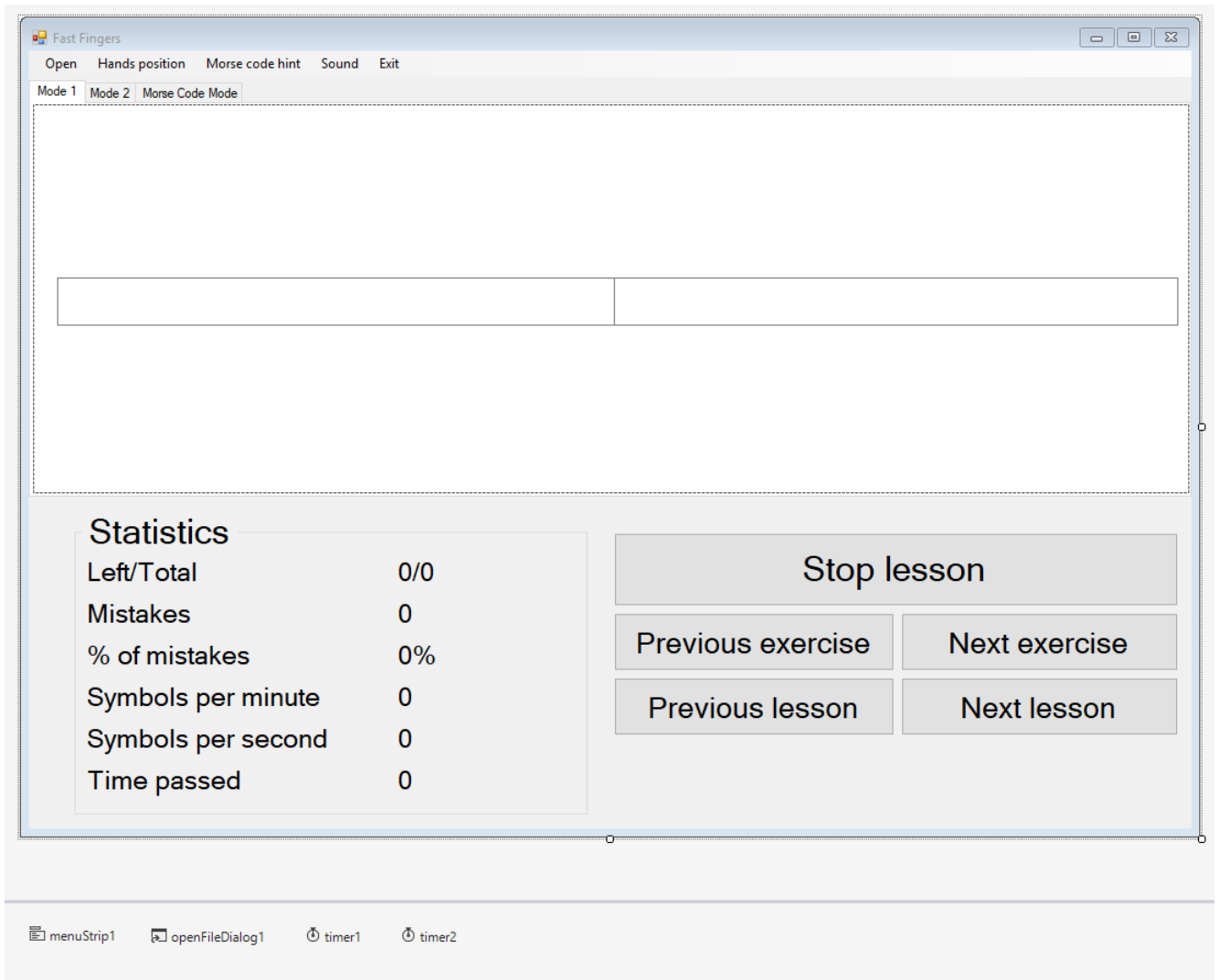


Рисунок 3.16 – Головне вікно форми

Далі для першої та другої вкладок задається фонове зображення. Завдяки цьому користувачу буде приємніше працювати в інтерфейсі програмного модуля. Для цього встановлюються властивості «BackgroundImageLayout» та «BackgroundImage» для компонентів «tabPage1» та «tabPage2» (див. рисунок 3.17). Властивість «BackgroundImageLayout» задано як «Stretch», щоб фонове зображення розтягувалося.

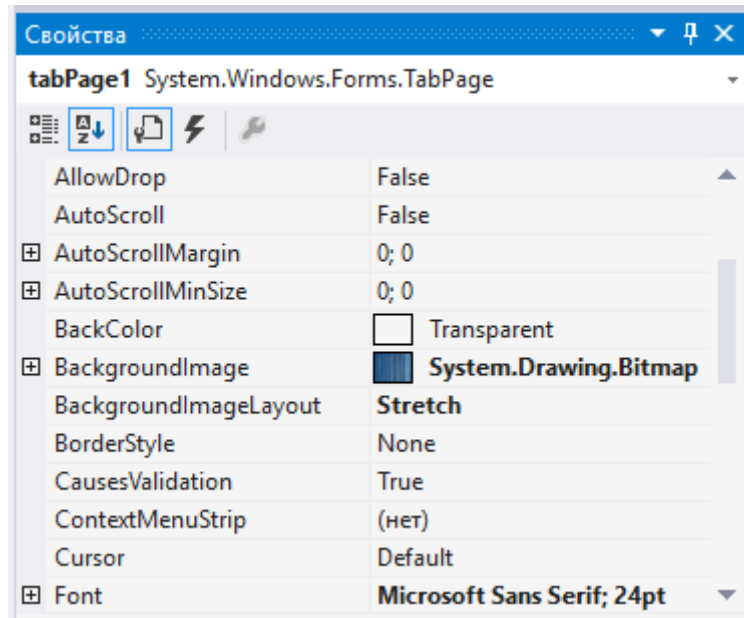


Рисунок 3.17 – Властивості «tabPage1»

Для компонента «tabPage3» встановлюються інакші властивості «BackgroundImageLayout» та «BackgroundImage» (див. рисунок 3.18). Властивість «BackgroundImageLayout» задано як «Zoom», щоб залишилася в оригінальних пропорціях.

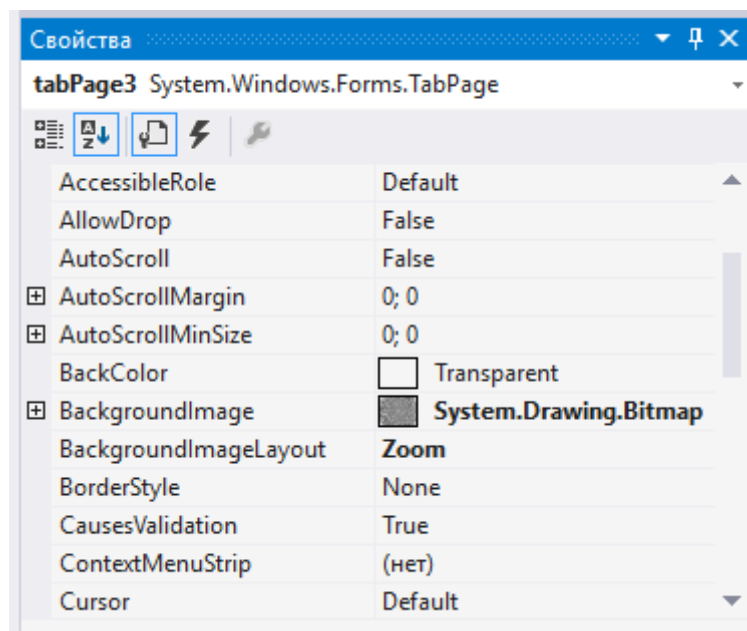


Рисунок 3.18 – Властивості «tabPage3»

Задається властивість форми «MaximizeBox» як «False» для того, щоб вікно мало фіксовану форму (див. рисунок 3.19):

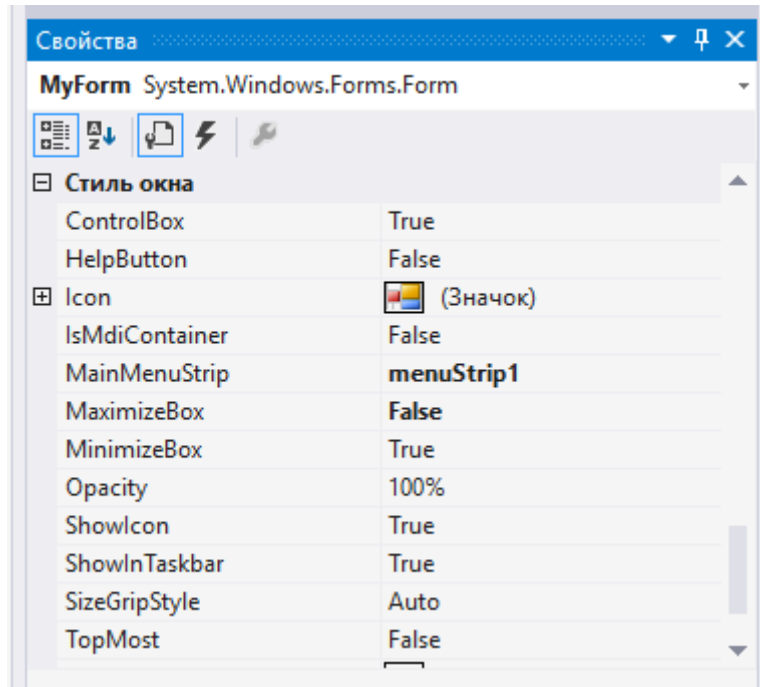


Рисунок 3.19 – Властивості «MyForm»

Кінцевий вигляд головного вікна програмного модуля представлений на наступному рисунку (див. рисунок 3.20):

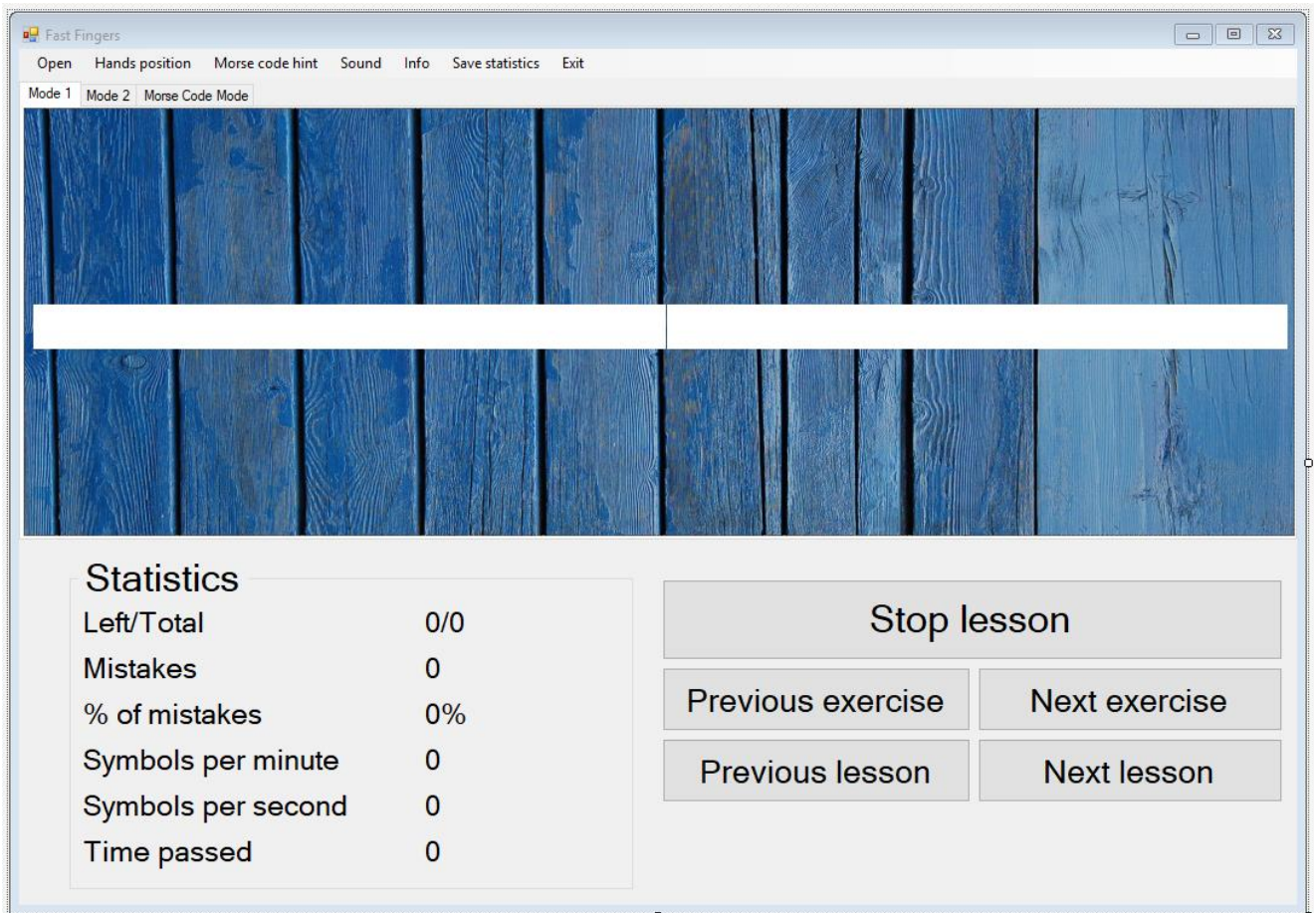


Рисунок 3.20 – Головне вікно форми

Наступним кроком створюється «MyForm1» та задаються властивості «Text» як «Hands position» (див. рисунок 3.21). Завантажується зображення, а властивість «BackgroundImageLayout» задається як «Zoom», щоб картинка залишилася в оригінальних пропорціях.

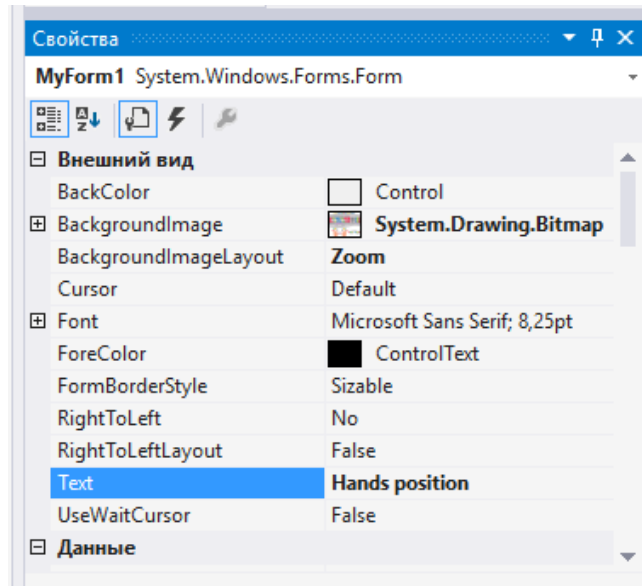


Рисунок 3.21 – Властивості форми «MyForm1»

Задається властивість форми «MaximizeBox» як «False» для того, щоб вікно мало фіксовану форму та «MinimizeBox» як «False» для того, щоб вікно не можна було згорнути. Кінцевий вигляд форми є наступним (див. рисунок 3.22):

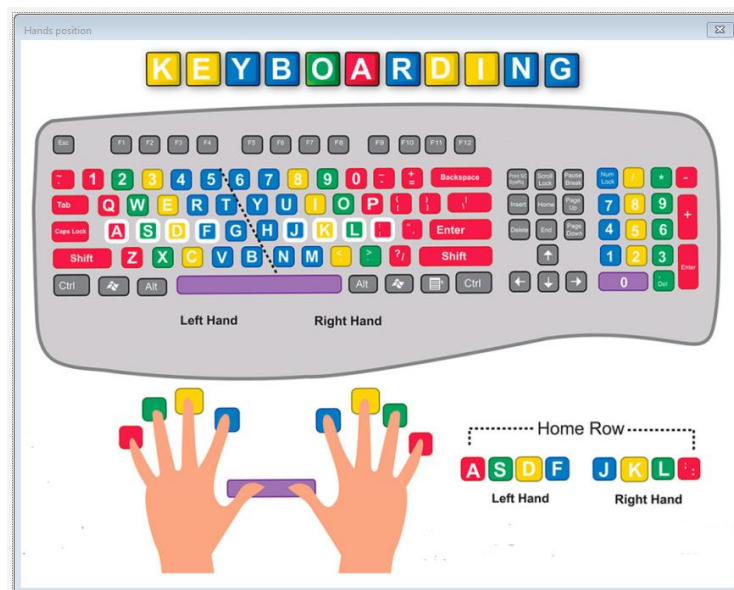


Рисунок 3.22 – Форма «Hands position»

Аналогічно створюються форми «MyForm2» та «MyForm3» (див. рисунок 3.23):

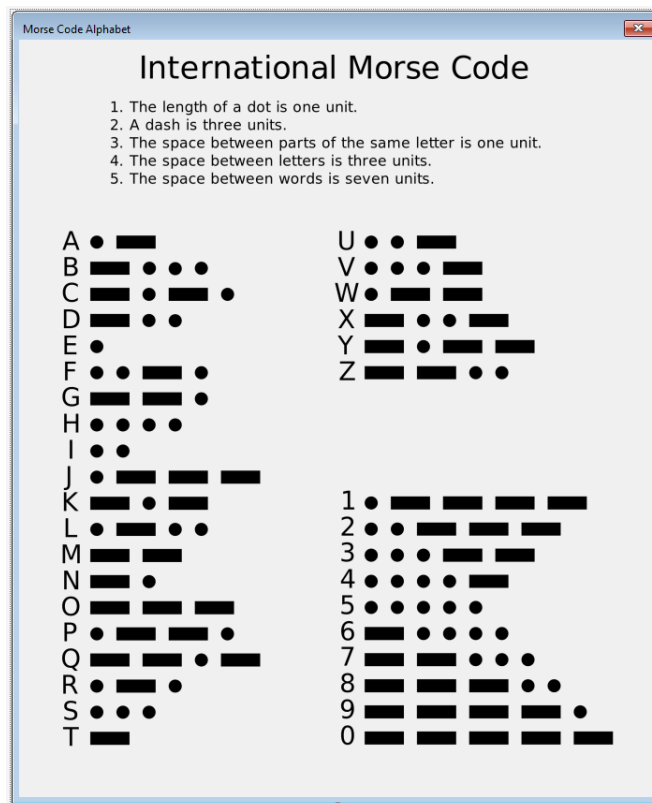


Рисунок 3.23 – Форма «Morse Code Alphabet»

3.2 Використовувані методи

В процесі розробки проекту використано наступні методи.

При завантаженні форми до неї підвантажується музичний файл, йому надається псевдонім та він відтворюється:

```
private: System::Void MyForm_Load(System::Object^ sender,
System::EventArgs^ e)
{ mciSendString("open 1.mp3 type mpegvideo alias background_sound",
NULL, 0, 0);
mciSendString("play background_sound repeat", NULL, 0, 0);
onToolStripMenuItem->Enabled = false; }
```

Для того, щоб при натисненні кнопки «Open File» користувачу відкривалося діалогове вікно для вибору користувацького тексту використано функцію «openToolStripMenuItem_Click()»:

```
private: System::Void openToolStripMenuItem_Click(System::Object^ sender,
```

```
System::EventArgs^ e)
```

```
{    openFileDialog1->ShowDialog();  
    openFile(openFileDialog1->FileName);    }
```

Для відкриття користувацького чи вбудованого тексту для навчання використовується функція «openFile()»:

```
private: void openFile(System::String^ fileName)  
{    StreamReader ^ UF = gcnew StreamReader(fileName);  
    textBox1->Text = "";  
  
    ...  
    textBox10->Text = "";  
    char *epic = SSTC(UF->ReadLine());  
    textBox2->Text = gcnew System::String(epic);  
    textBox5->Text = gcnew System::String(epic);  
    textBox8->Text = gcnew System::String(epic);  
    UF->Close();  
    string sEpic(epic);  
    dataObject.SetNewString(epic);  
    dataObject.SetNewString(sEpic);  
    dataObject.SetMistakes();  
    textBox3->Text = gcnew String(dataObject.GetWord().c_str());  
    textBox10->Text = Morse::encrypt(SSTC(textBox8->Text));  
    dataObject.SetNewMorseString(SSTC(textBox10->Text));    }
```

Відкривається файл, очищуються всі поля «TextBox», зчитується рядок з документа, записується до деяких «TextBox».

Щоб відкрити довідку про положення рук під час друку використовується наступна функція:

```
private: System::Void handsPositionToolStripMenuItem_Click(System::Object^  
sender, System::EventArgs^ e)
```

```
{    MyForm1 form1;  
    form1.ShowDialog();    }
```

Створюється об'єкт форми «MyForm1» та відкривається у вигляді вікна.

Аналогічно відбувається з азбукою Морзе та спеціальними символами.

Коли вимикається фонові мелодія, стає доступна кнопка, щоб її знову увімкнути, а кнопка «Off» стає неактивною.

```
private: System::Void offToolStripMenuItem_Click(System::Object^ sender,
System::EventArgs^ e)
{
    onToolStripMenuItem->Enabled = true;
    offToolStripMenuItem->Enabled = false;
    mciSendString("stop background_sound", NULL, 0, 0);
    mciSendString("seek background_sound to start", NULL, 0, 0);
}
```

Коли вмикається фонові мелодія стає доступна кнопка, щоб її знову вимкнути, а кнопка «On» стає неактивною.

```
private: System::Void onToolStripMenuItem_Click(System::Object^ sender,
System::EventArgs^ e)
{
    onToolStripMenuItem->Enabled = false;
    offToolStripMenuItem->Enabled = true;
    mciSendString("play background_sound repeat", NULL, 0, 0);
}
```

Для відображення довідкової інформації про використання програми використовується наступна функція:

```
private: System::Void howToolStripMenuItem_Click(System::Object^ sender,
System::EventArgs^ e)
{
    MessageBox::Show("1. Select the mode\n2. Select a file(custom / built -
in)\n3. Start typing\n4. Optionally, you can turn on / off the sound\n5. Optionally, you
can save statistics", "How To", MessageBoxButtons::OK, MessageBoxIcon::None);
}
```

Для відображення інформації про версію програми та її розробника використовується наступна функція:

```
private: System::Void aboutProgramToolStripMenuItem_Click(System::Object^
sender, System::EventArgs^ e)
{
    MessageBox::Show("Version: 1.0\n\nCreator: Yulia Vergeles\n\nVrite to
the developer: grapefruit777@outlook.com", "About Program FastFingers",
MessageBoxButtons::OK, MessageBoxIcon::None);
}
```

Щоб зберегти статистичну інформацію використовується функція

«saveStatisticsToolStripMenuItem_Click()»:

```
private: System::Void saveStatisticsToolStripMenuItem_Click(System::Object^
sender, System::EventArgs^ e)
{
    dataObject.GetTime();
    ofstream fout("statistics.txt", ios_base::app);
    fout << dataObject.GetDay() << "." << dataObject.GetMon() << "." <<
    dataObject.GetYear() << " " << dataObject.GetHour() << ":" <<
    dataObject.GetMin() << ":" << dataObject.GetSec() << " symbols per
    minute: " << dataObject.GetSPM() << " symbols per second: " <<
    dataObject.GetSPS() << " % of mistakes: " << dataObject.GetPOM() <<
    "\n";
    fout.close();
}
```

Виконується функція, яка визначає поточну дату та час. Відкривається файл «statistics.txt» для дозапису в кінець файлу. Записується потрібна інформація. Файл закривається.

3.3 Тестування програмного модуля

Мета: координація зусиль учасників проекту в частині контролю якості і проведення тестування програмного модуля.

Рівень плану: Тест План (Test Plan).

Склад документа: опис загальних підходів і видів тестів програмного модуля «Клавіатурний тренажер»; опис проведення тестування з точки зору кінцевого користувача.

План проекту: тестування програмного модуля як самостійного продукту.

Опис тестованого модуля: являє собою написаний на C++ програмний модуль, який реалізує три режими навчання і має власний інтерфейс.

Відповідний стандарт: IEEE 829-1998 Format.

Тестові завдання. Контролю якості повинен бути підданий програмний модуль в цілому, а також його окремі частини.

Повинно бути проведено тестування таких частин:

1. Функціональні можливості (правильність, здатність до взаємодії).

2. Надійність (стабільність, стійкість до помилки).
3. Практичність (зрозумілість, простота використання).
4. Ефективність (характер зміни в часі, характер зміни ресурсів).
5. Супроводжуваність (аналізованість).
6. Мобільність (адаптованість, простота впровадження).

Аспекти, що тестуються. Функції, які повинні бути перевірені з точки зору користувачів, що робить модуль і як він функціонує:

1. Програмний модуль в цілому (Н).
2. Коректність введення і виведення інформації (Н).
3. Три режими навчання (Mode 1, Mode 2, Morse Code Mode) (Н).
4. Додаткові функції (відтворення звуку і функція збереження в текстовий файл статистичної інформації) (L).
5. Коректність функціонування інтерфейсу (М).

Нетестовані аспекти. Елементи, які не повинні бути перевірені як з точки зору користувачів, так і системою управління конфігурацією/версією управління версіями з огляду на не включення таких елементів в цей випуск програмного модуля:

1. Узгодженість.
2. Захищеність.
3. Відновлюваність.
4. Стійкість.
5. Взаємозамінність.

Підхід до тестування. Рівень тестування: системне, з точки зору кінцевого користувача.

Спеціальні засоби тестування: відсутні, тестування буде проводитися вручну.

Метрики: в рамках даного плану передбачається створити комплект тестів, повний щодо метрики за тестовими випадками (Test Cases) (відповідно до міжнародного стандарту ISO 14598).

Особливі вимоги до тестування: відсутні, тестування проводиться в звичайному режимі.

Сегмент компонентів: певний сегмент компонентів повинен бути

протестований разом.

Обмеження для тестування: істотним обмеженням є проведення тестування вручну.

Рекомендована методика тестування: ручна, так як зменшує матеріальні та програмні витрати на проведення тестування.

Критерії успішності та припинення тестування. Критерії успішності тестування:

1. Система передається в експлуатацію, коли розроблений повний комплект тестів і всі розроблені тести виконуються без помилок.

2. Вдале тестування на виправлену помилку при повторній передачі на тестування.

Критерії припинення тестування:

1. Система повертається на доопрацювання, якщо хоча б один з розроблених тестів виявляє помилку.

2. Після виправлення помилки система знову передається на тестування.

Тест кейси. Тест кейси зображені на таблицях 3.1, 3.2, 3.3.

Таблиця 3.1 – Тест кейс «Функція 1.1»

| | | |
|--|--|--|
| Назва: | Функція 1.1 | |
| Функція: | Набір тексту в першому режимі | |
| Дія | Очікуваний результат | Результат теста: <ul style="list-style-type: none">• пройдено• провалено• заблоковано |
| Передумова: | | |
| Відкрийте програмний модуль | Програмний модуль відкритий і доступний | пройдено |
| Відкрийте через головне меню «Орен» та виберіть вправо | Відкриття тексту | пройдено |
| Кроки тесту: | | |
| Набір тексту по одній букві | Перехід тексту до лівої частини рядку введення | пройдено |
| Набір останнього елемента тексту | Праве вікно з текстом пуста | пройдено |
| Післяумова: | | |
| | | |
| | | |

Таблиця 3.2 – Тест кейс «Функція 1.2»

| | | |
|--|---|---|
| Назва: | Функція 1.2 | |
| Функція: | Набір тексту в другому режимі | |
| Дія | Очікуваний результат | Результат теста: • пройдено • провалено • заблоковано |
| Передумова: | | |
| Відкрийте програмний модуль | Програмний модуль відкритий і доступний | пройдено |
| Відкрийте через головне меню «Орен» та виберіть вправо | Відкриття тексту | пройдено |
| Виберіть другий режим | Відкриття другого режиму | пройдено |
| Кроки тесту: | | |
| Набір тексту по одному слову + «Space» | Перехід тексту до іншого текстового | пройдено |
| Набір слова з помилкою + «Space» | Виділення слова червоним кольором | пройдено |
| Набір останнього елемента тексту | Вікно з текстом пуста | пройдено |
| Післяумова: | | |
| | | |
| | | |

Таблиця 3.3 – Тест кейс «Функція 1.3»

| | | |
|--|--|---|
| Назва: | Функція 1.3 | |
| Функція: | Набір тексту в режимі навчання коду Морзе | |
| Дія | Очікуваний результат | Результат теста: • пройдено • провалено • заблоковано |
| Передумова: | | |
| Відкрийте програмний модуль | Програмний модуль відкритий і доступний | пройдено |
| Відкрийте через головне меню «Орен» та виберіть вправо | Відкриття тексту | пройдено |
| Виберіть режим навчання коду Морзе | Відкриття режиму навчання коду Морзе | пройдено |
| Кроки тесту: | | |
| Набір тексту по одному символу | Перехід тексту до лівої частини рядку введення | пройдено |
| Набір неправильного символу | Блокування введення | пройдено |
| Набір останнього елемента тексту | Вікно з текстом пуста | пройдено |
| Післяумова: | | |
| | | |
| | | |

3.4 Висновки до розділу

1. Розроблено інтерфейс програмного модуля, що відповідає вимогам та дозволяє користувачу комфортно користуватись всіма функціями модуля.

2. Реалізовані 3 режими навчання набору тексту, а саме: «Mode 1» (звичайний режим), «Mode 2» (режим вводу слів) та «Morse code mode» (режим навчання коду Морзе).

3. Реалізовано додатковий функціонал інтерфейсу, а саме: відтворення звуку, відображення та збереження статистичної інформації, відображення довідкової інформації.

4. Розроблено тест план, згідно якому протестовано програмний модуль. В процесі виконання тестування не було виявлено критичних дефектів.

РОЗДІЛ 4. КЕРІВНИЦТВО КОРИСТУВАЧА

4.1 Технічні вимоги

Програмний модуль надається у вигляді папки з вихідними кодами та скомпільованим програмним файлом формату .exe. Для працездатності необхідний персональний комп'ютер з операційною системою Windows 10, а також встановлений .NET Framework версії не нижче 4.5.1.

Розмір папки з програмним модулем та вихідними кодами – 120 МБ.

4.2 Керівництво користувача

Після відкриття програмного модуля користувач потрапляє в головне меню (див. рисунок 4.1).

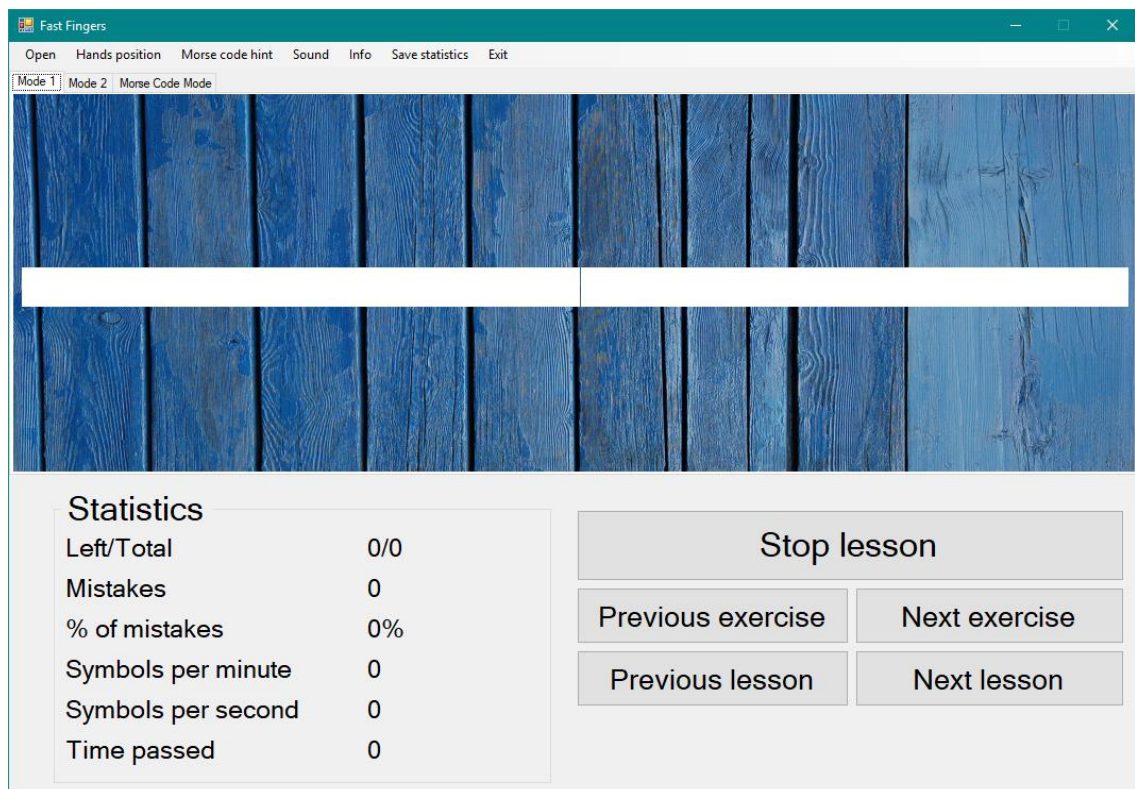


Рисунок 4.1 – Головна форма програмного модуля

| | | | | | | | |
|---------------------|-----------------|--|--|--------------------------------|---------------|--------------|----------------|
| Кафедра КСУ | | | | НАУ 21 11 30 000 ПЗ | | | |
| <i>Виконав</i> | Таран І.А. | | | КЕРІВНИЦТВО КОРИСТУВАЧА | <i>Літера</i> | <i>Аркуш</i> | <i>Аркушів</i> |
| <i>Керівник</i> | Марченко Н.Б. | | | | | 45 | 60 |
| <i>Консульт.</i> | | | | | 123 СП-437 | | |
| <i>Норм. контр.</i> | Тупота Є.В. | | | | | | |
| <i>Зав.каф.</i> | Литвиненко О.Є. | | | | | | |

За змовчуванням програмний модуль відкривається на першому режимі навчання. Присутнє головне меню програмного модуля, вкладки з режимами навчання, область відображення статистичної інформації та п'ять кнопок для маніпуляцій з вбудованими вправами.

При натисненні на кнопку «Info» та на його підпункт «How to» з'являється вікно з загальною інформацією як користуватися програмним модулем (див. рисунок 4.2). Потрібно вибрати режим, далі потрібно вибрати файл та почати урок. Можна вмикати/вимикати фоновий аудіозапис. Можна зберігати статистику.

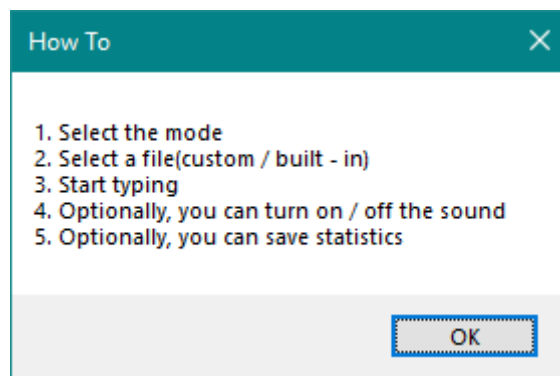


Рисунок 4.2 – Вікно «How to»

На кнопці «Info» також є підпункт «About program», який надає інформацію про поточну версію програмного модуля, хто його створив та як зв'язатися з розробником (див. рисунок 4.3).

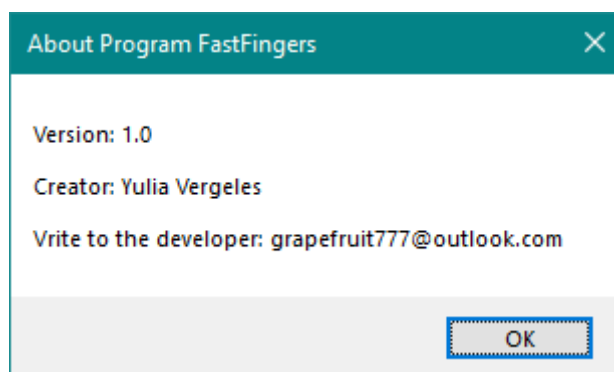


Рисунок 4.3 – Вікно «About program»

На кнопці «Hands position» знаходиться картинка, яка ілюструє як потрібно розташовувати пальці на клавіатурі (див. рисунок 4.4).

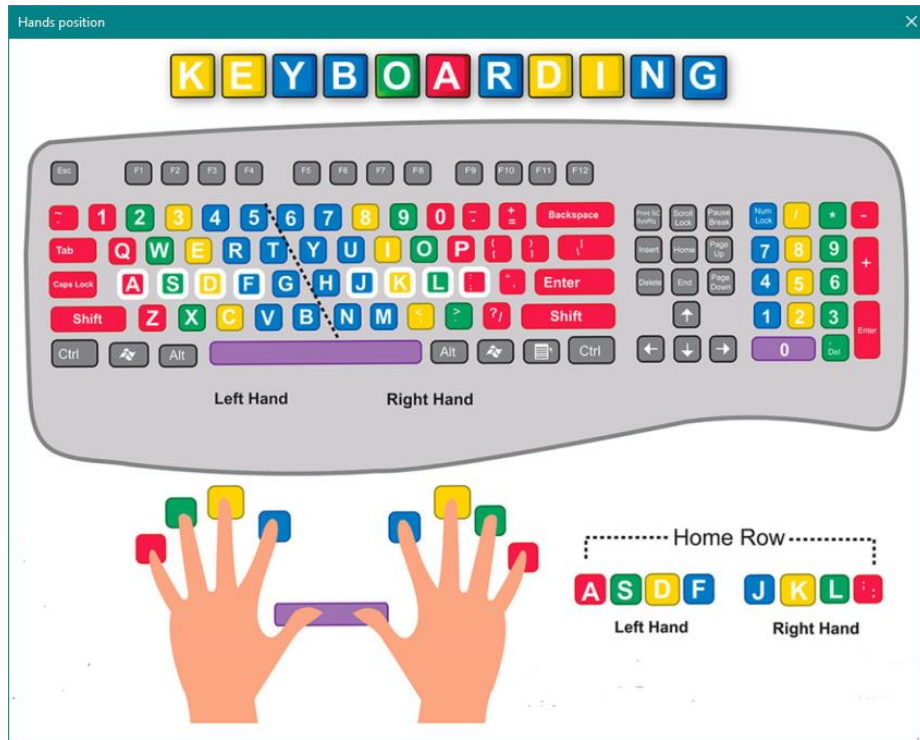


Рисунок 4.4 – Вікно «Hands position»

На кнопці «Morse code hint» розташовано два підпункти «Alphabet» та «Special Symbols». У вікні «Alphabet» демонструється англійський алфавіт та цифри в інтерпретації коду Морзе (див. рисунок 4.5).

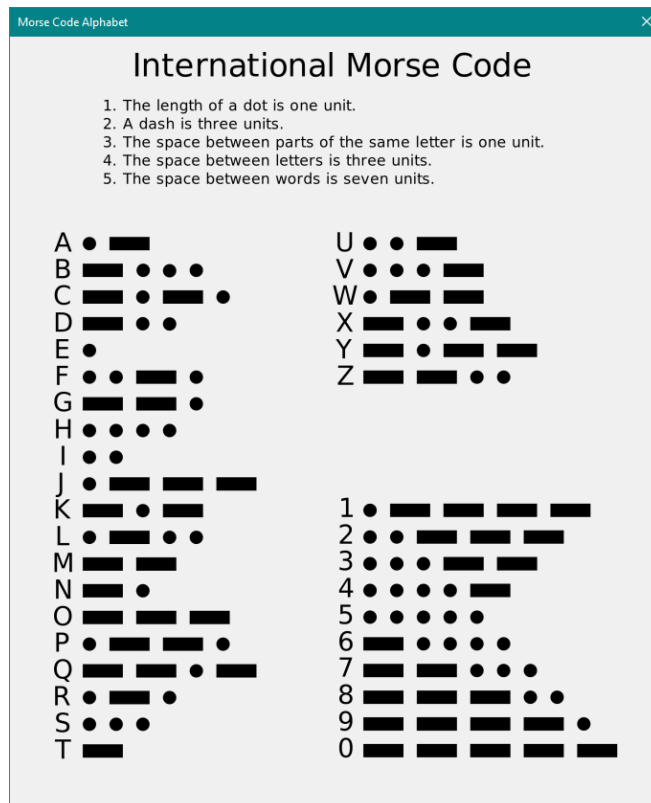


Рисунок 4.5 – Вікно «Morse Code Alphabet»

У вікні «Special Symbols» демонструються спеціальні символи коду Морзе (див. рисунок 4.6).

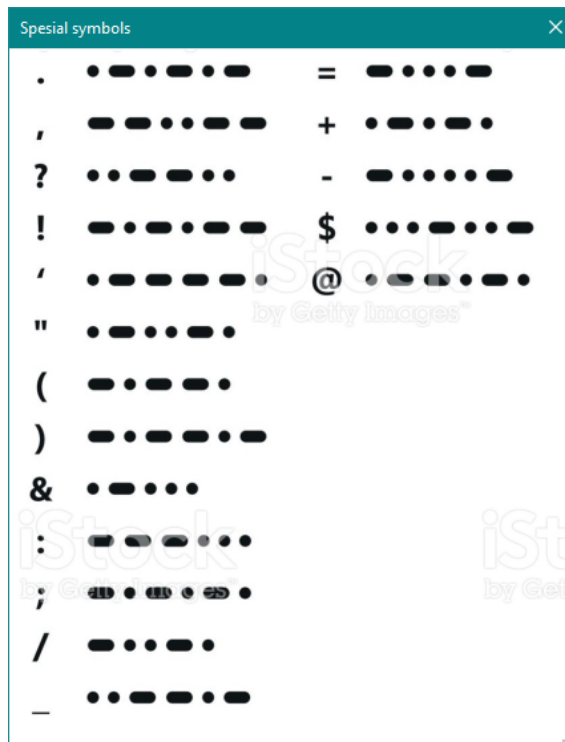


Рисунок 4.6 – Вікно «Special Symbols»

На кнопці «Sound» можна вмикати та вимикати фоновий аудіозапис (див. рисунок 4.7).

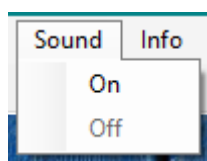


Рисунок 4.7 – Кнопка «Sound»

В головному меню програмного модуля можна вибрати пункт «Open» підпункт «Open File». Відкриється діалогове вікно (див. рисунок 4.8). Обирається користувацький текст та натискається кнопка «Open».

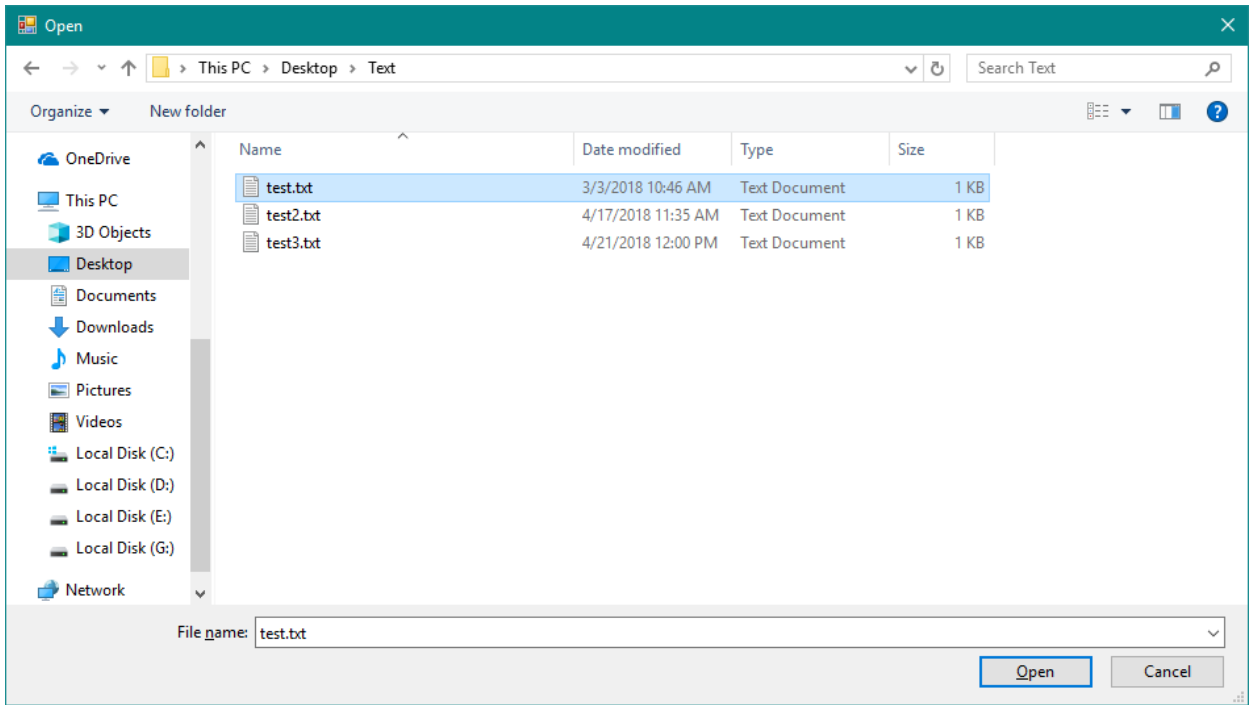


Рисунок 4.8 – Діалогове вікно «Open File»

Користувацький текст відкривається у першому режимі (але завантажується у всі три) (див. рисунок 4.9).

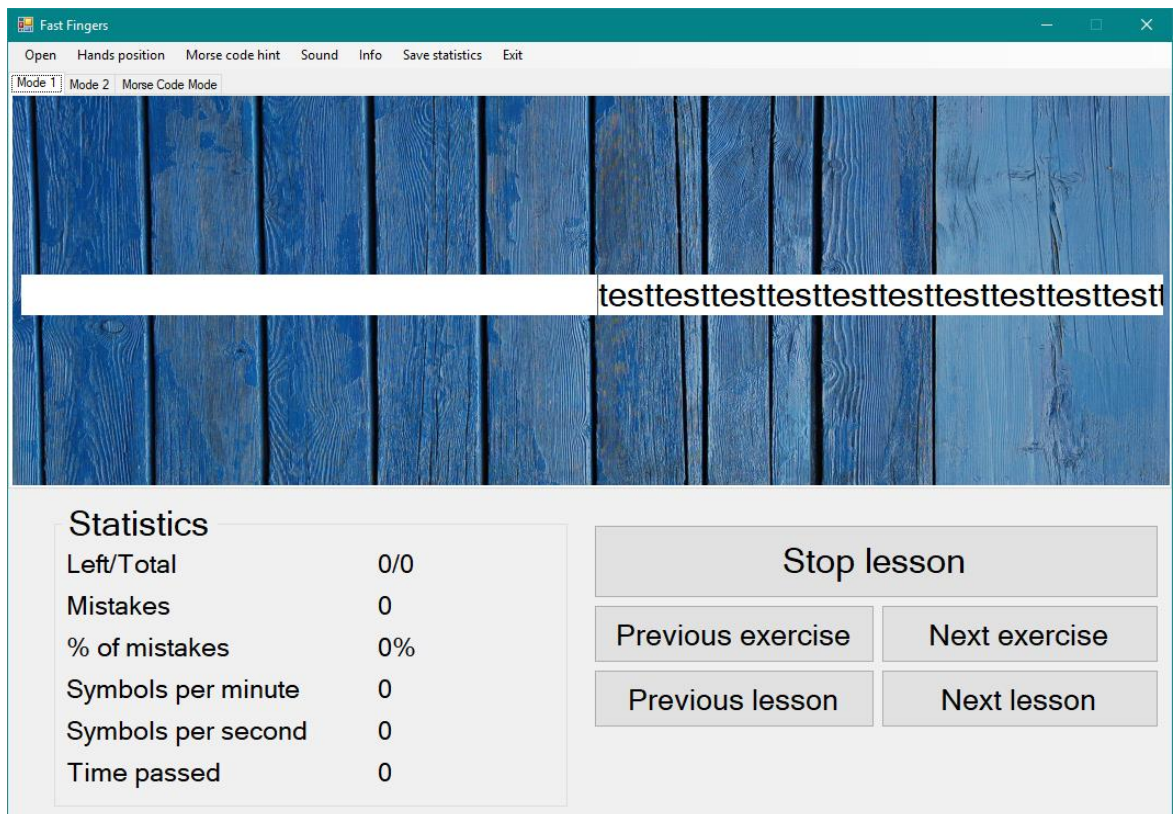


Рисунок 4.9 – Завантаження користувацького тексту

Щоб обрати вбудований урок потрібно натиснути на кнопку «Open» в головному меню, далі на «Open Lesson» та обрати номер уроку та вправи (див. рисунок 4.10).

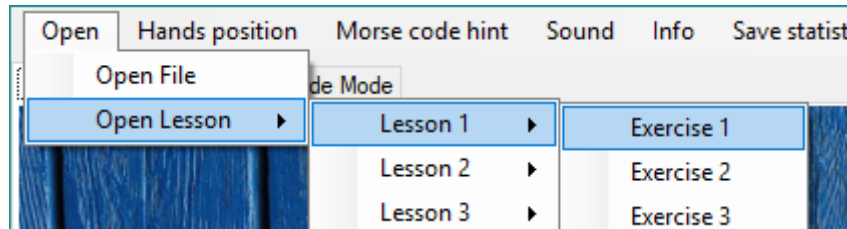


Рисунок 4.10 – Вибір вбудованого уроку

Вбудований текст відображується на екрані (див. рисунок 4.11).

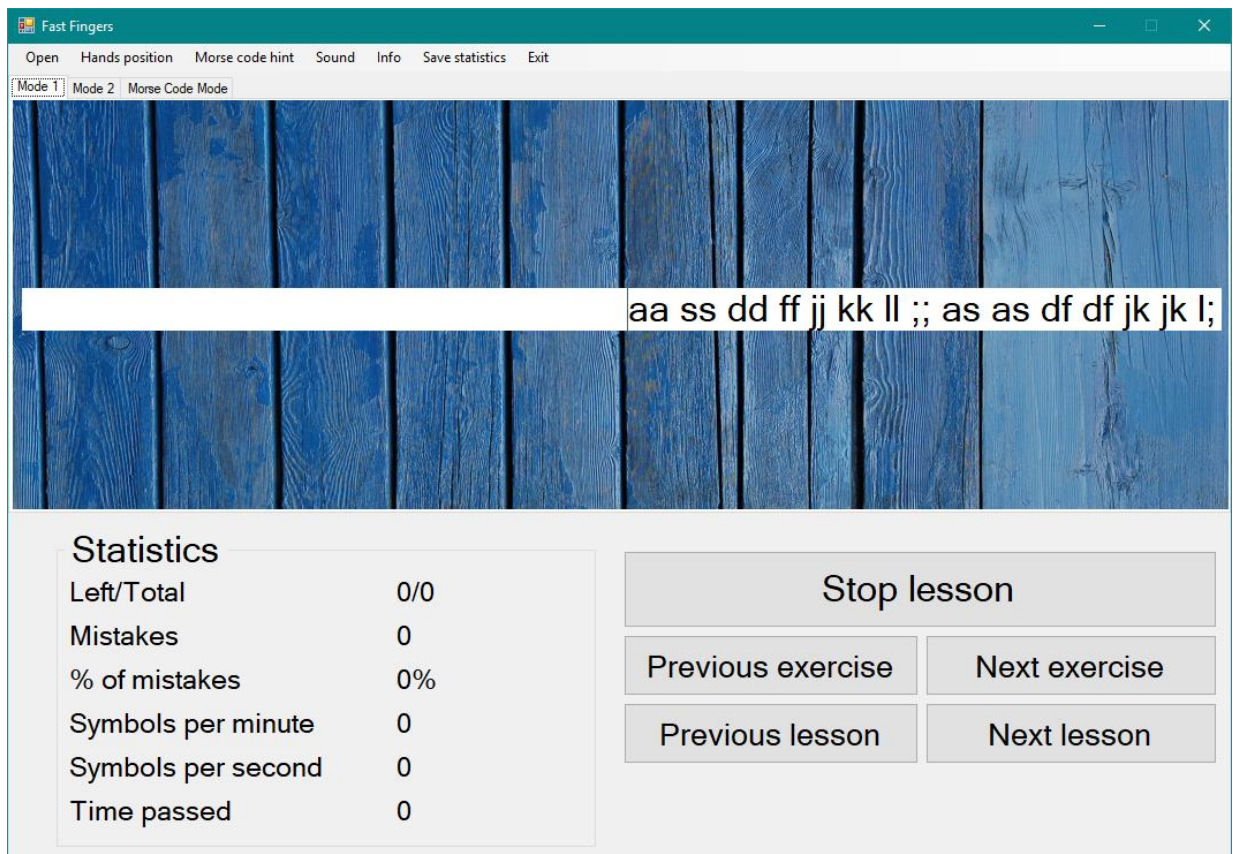


Рисунок 4.11 – Завантаження вбудованого тексту

Тепер можна розпочати навчання. По мірі того як буде набиратися текст, він буде зсуватися ліворуч (див. рисунок 4.12). Якщо натиснута неправильна клавіша (та, яка не відповідає поточному символу для набору), на екрані це не відобразиться, але кількість помилок зросте на одиницю.

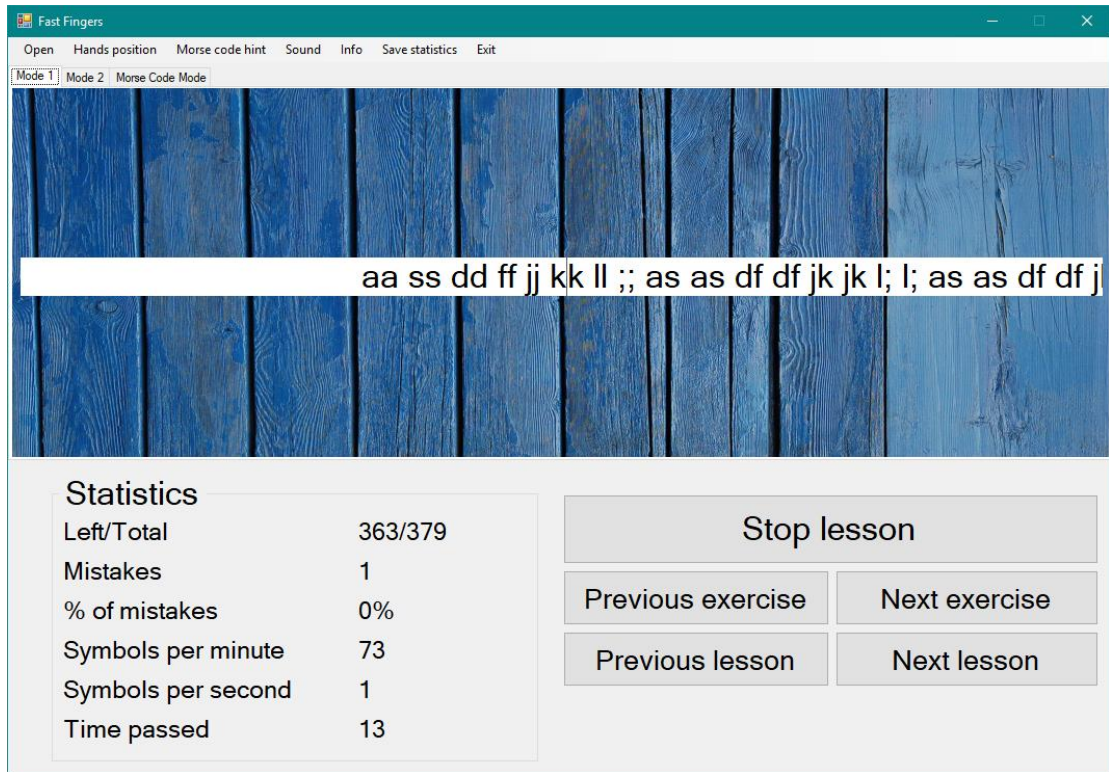


Рисунок 4.12 – Навчання в першому режимі

Другий режим навчання виглядає наступним чином (див. рисунок 4.13):

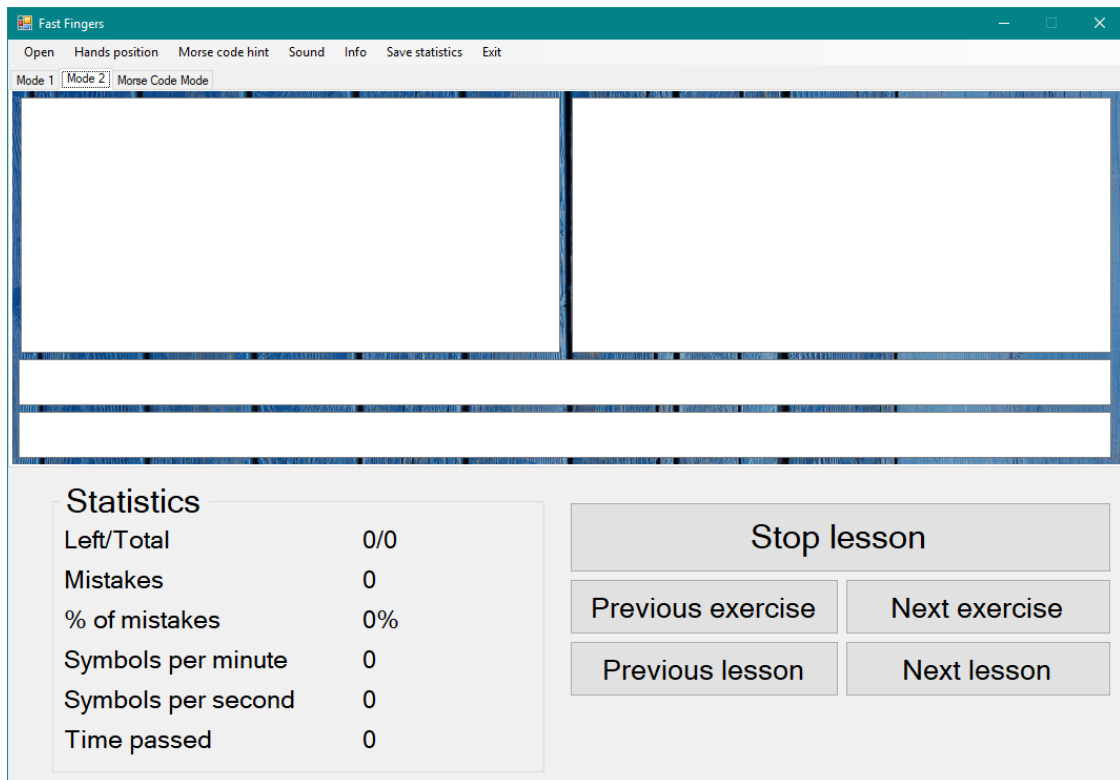


Рисунок 4.13 – Інтерфейс другого режиму навчання

Після завантаження тексту вікно програми в другому режимі навчання має наступний вигляд (див. рисунок 4.14):

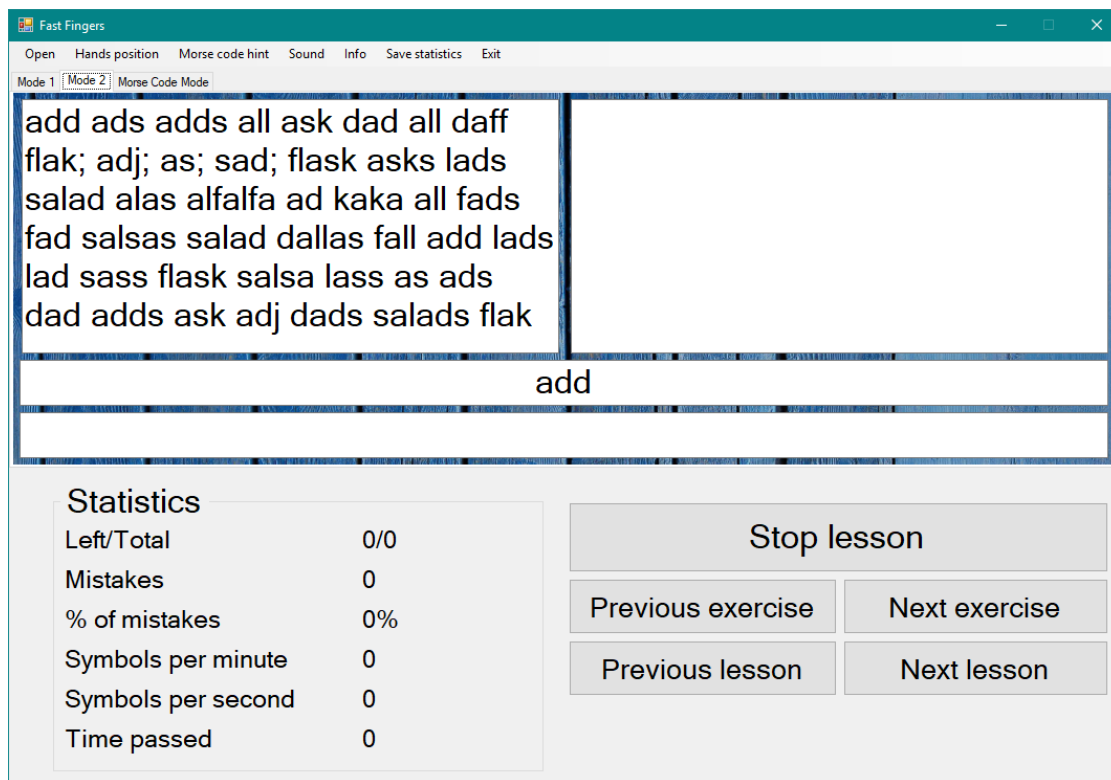


Рисунок 4.14 – Завантаження вбудованого тексту

В другому режимі навчання очікується поки користувач надрукує все слово разом та натисне «Space». Якщо слово, яке набране користувачем, відповідає слову, яке очікується, то слово переноситься до правого текстового поля. Якщо слово введено невірно та було натиснуто клавішу «Space», то воно виділяється червоним кольором, та очікується доки користувач його виправить (див. рисунок 4.15). До кількості помилок додається одиниця.

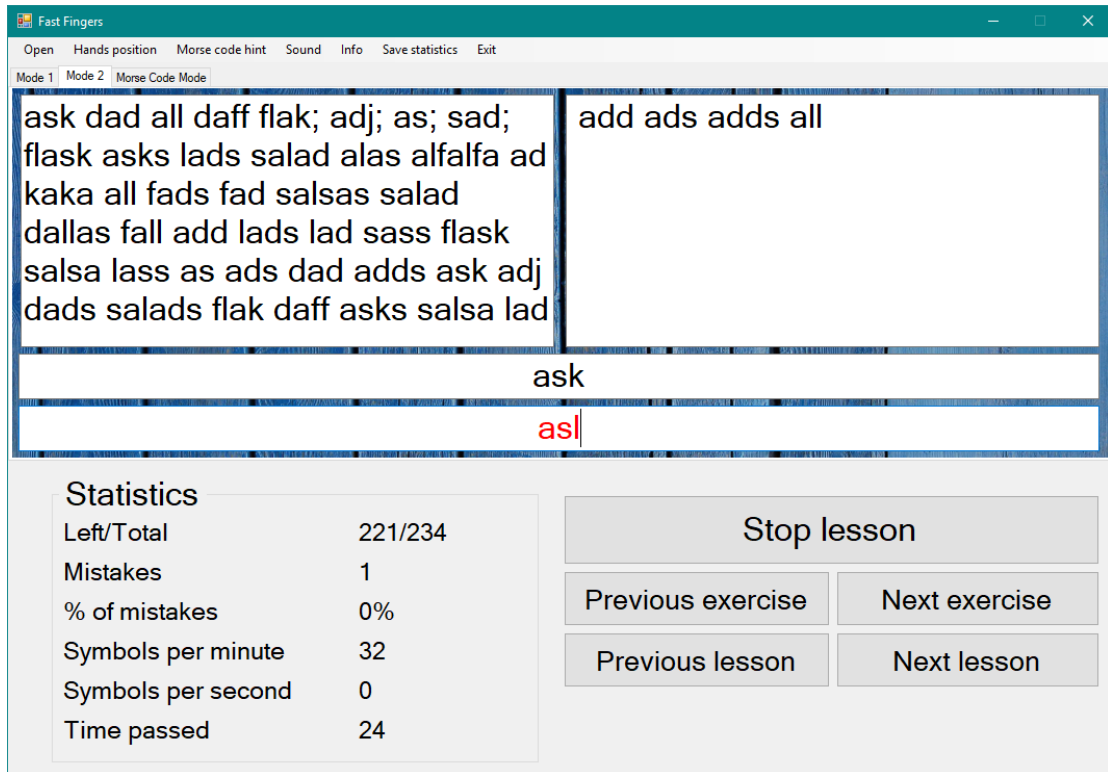


Рисунок 4.15 – Навчання в другому режимі

Третім режимом є навчання коду Морзе (див. рисунок 4.16).

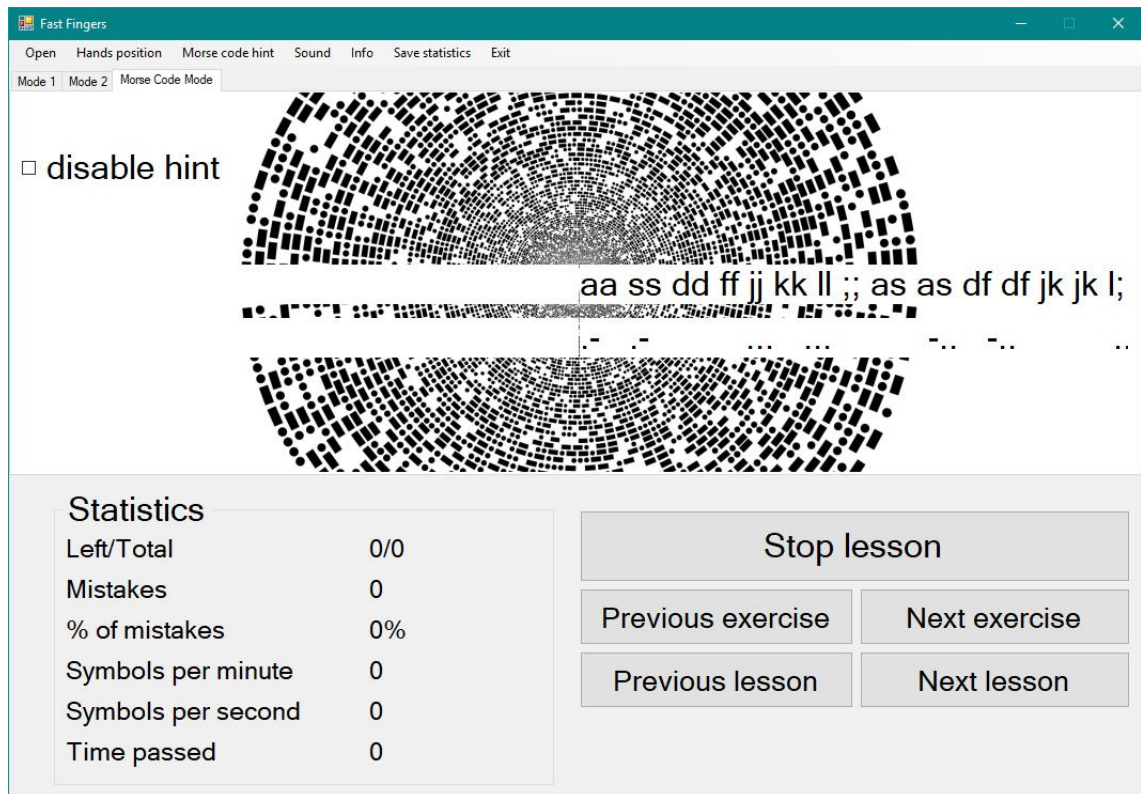


Рисунок 4.16 – Інтерфейс третього режиму навчання

Даний режим працює аналогічно з першим: символи коду Морзе вводяться до нижнього лівого поля, поступово видаляючись з правого. В той самий час текст з верхнього лівого поля переходить у праве по мірі набору цільних символів (див. рисунок 4.17).

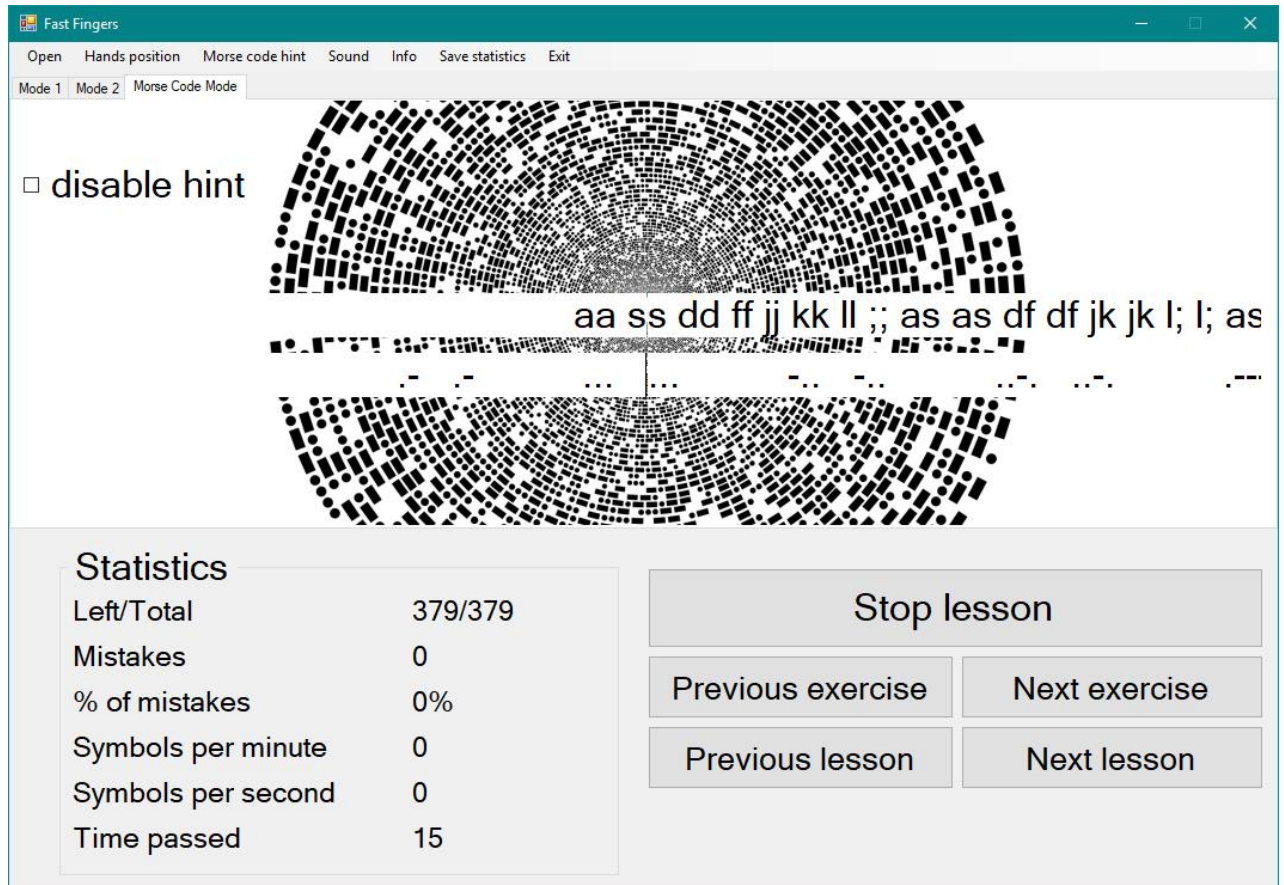


Рисунок 4.17 – Навчання в третьому режимі

Для даного режиму можливо ввімкнути функцію приховування підказки, тобто закодованого тексту. Для цього потрібно активізувати прапорець «disable hint» (див. рисунок 4.18).

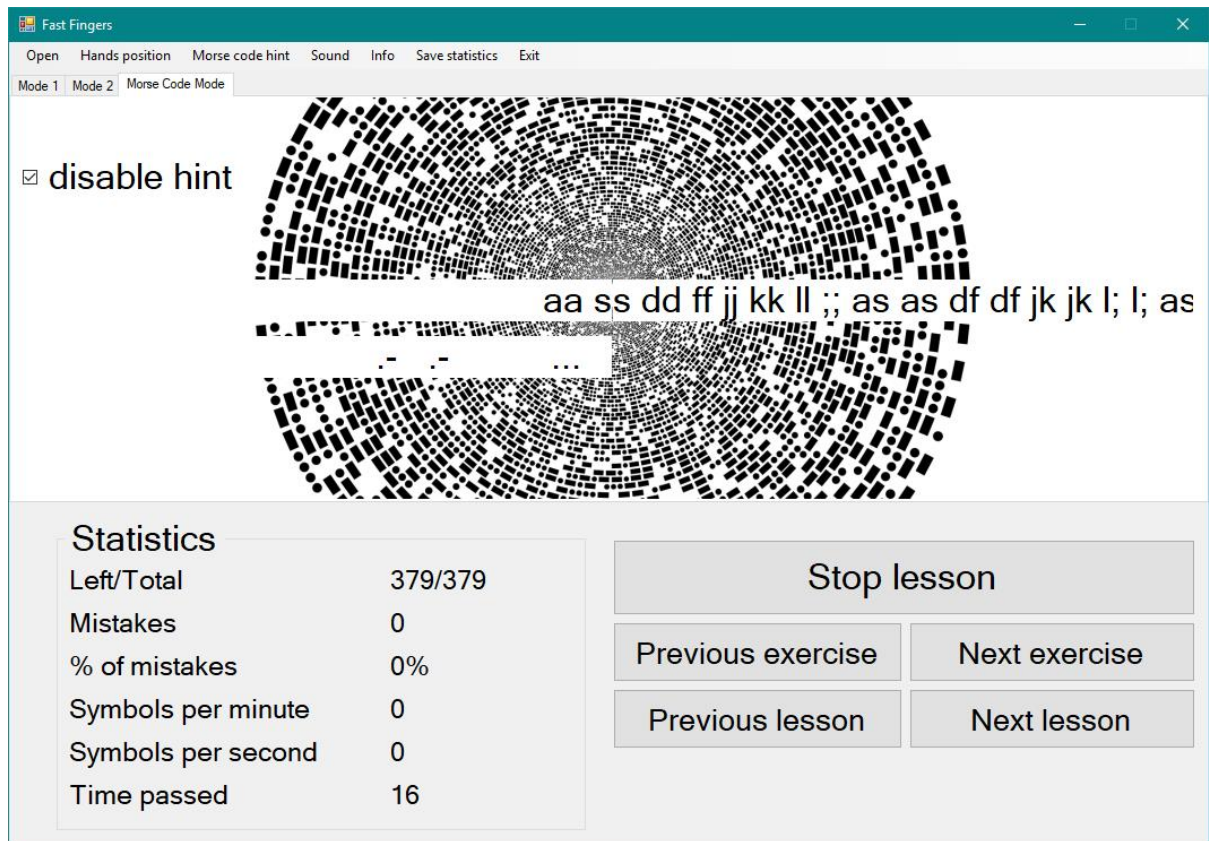


Рисунок 4.18 – Активізований прапорець «disable hint»

На екрані також відображується статистична інформація (див. рисунок 4.19). Вона описує скільки символів уроку залишилось набрати та скільки їх всього, кількість допущених помилок, відсоток допущених помилок, швидкість набору (символів в хвилину та в секунду) та скільки пройшло часу.

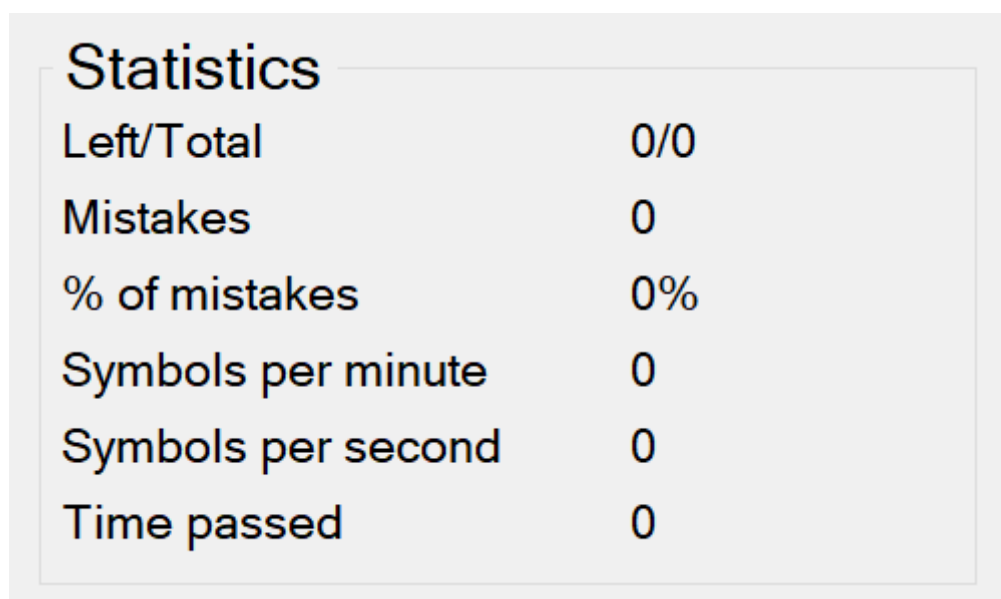
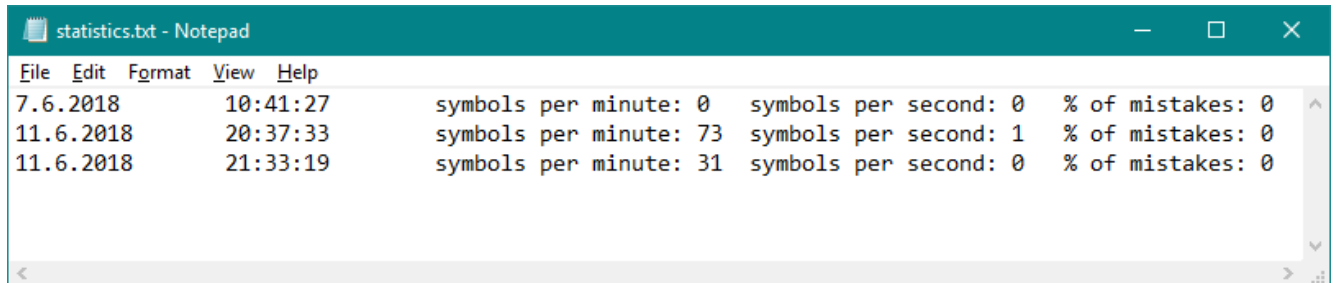


Рисунок 4.19 – Статистична інформація

За бажанням користувач може зберегти статистику до файлу натиснувши кнопку головного меню «Save statistics».

Приклад збереження статистики до файлу зображено на наступному малюнку (див. рисунок 4.20):



```
statistics.txt - Notepad
File Edit Format View Help
7.6.2018      10:41:27      symbols per minute: 0      symbols per second: 0      % of mistakes: 0
11.6.2018    20:37:33      symbols per minute: 73     symbols per second: 1      % of mistakes: 0
11.6.2018    21:33:19      symbols per minute: 31     symbols per second: 0      % of mistakes: 0
```

Рисунок 4.20 – Відображення статистики в файлі

За допомогою кнопок «Stop lesson» «Previous exercise» «Next exercise» «Previous lesson» «Next lesson» відбувається маніпуляція з уроками (див. рисунок 4.21).

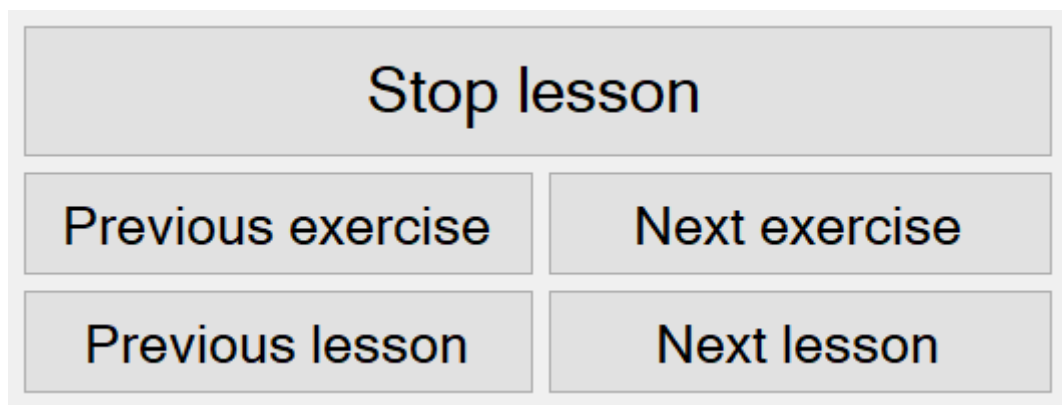


Рисунок 4.21 – Кнопки

4.3 Висновки до розділу

1. Визначено технічні вимоги, що відображають характеристики програмного та апаратного забезпечення, необхідного для коректного функціонування програмного модуля.

2. Розроблено керівництво користувача, що дає основні вказівки по експлуатації програмного модуля.

ВИСНОВКИ

1. Проаналізовано програми типу «Клавіатурний тренажер». Виявлено переваги та недоліки існуючих клавіатурних тренажерів. Головними недоліками визнано те, що переважна більшість з них платні, мають лише по одному режиму навчання, в деяких неможливо обирати уроки та завантажувати користувацькі тексти для набору. Перевагами проаналізованих клавіатурних тренажерів є: наявність декількох клавіатурних розкладок, функція переважної генерації комбінацій символів, в яких було допущено більше всього помилок, детальний вступний урок, де пояснюється який палець повинен натискати кожен клавішу. На основі аналізу сформульовано попередній список вимог до готового продукту.

2. Для реалізації програмного модуля обрано мову програмування C++ та середовище розробки Microsoft Visual Studio Express 2013.

3. Визначено функціональні вимоги до модуля, основними з яких є реалізація режимів навчання, можливість відображення статистичної інформації користувача, коректне відтворення спеціального фонового звукового файлу, можливість підрахунку помилок, фіксування часу, затраченого користувачем на набір всього тексту та окремих його символів, перевірка окремого набраного слова на відповідність тексту. Визначено нефункціональні вимоги, які представляють собою вимоги до інтерфейсу та системні вимоги для коректної роботи програмного модуля.

4. На основі сформульованих вимог розроблено структуру, що відображає основні функціональні можливості програмного модуля.

5. Розроблено ряд доповнюючих схем та діаграм, а саме діаграму варіантів використання, яка є вихідним концептуальним представленням системи в процесі її проектування та розробки, а також функціональні схеми, у вигляді яких представлені всі режими навчання, реалізовані в програмному модулі.

6. Розроблено інтерфейс програмного модуля, що відповідає вимогам та дозволяє користувачу комфортно користуватись всіма функціями модуля.

7. Реалізовані 3 режими навчання набору тексту, а саме: «Mode 1» (звичайний режим), «Mode 2» (режим вводу слів) та «Morse code mode» (режим навчання коду

Морзе).

8. Реалізовано додатковий функціонал інтерфейсу, а саме: відтворення звуку, відображення та збереження статистичної інформації, відображення довідкової інформації.

9. Розроблено тест план, згідно якому протестовано програмний модуль. В процесі виконання тестування не було виявлено критичних дефектів.

10. Визначено технічні вимоги, що відображають характеристики програмного та апаратного забезпечення, необхідного для коректного функціонування програмного модуля.

11. Розроблено керівництво користувача, що дає основні вказівки по експлуатації програмного модуля.

12. Викладено вимоги до робочого місця техника-програміста, а саме: висоту робочої поверхні, розміри простору для ніг, вимоги до розташування документів на робочому місці, характеристики робочого крісла, вимоги до поверхні робочого столу, регульованість елементів робочого місця.

13. Проведено розрахунок освітлення виробничого приміщення, в результаті якого визначено оптимальні тип та кількість освітлювальних приладів для конкретного робочого місця.

14. Проведено розрахунок рівня шуму на робочому місці та визначено, що отримане значення не перевищує допустимий рівень шуму для робочого місця оператора.

Практична цінність дипломного проекту полягає у створенні робочого зразка програмного модуля та подальшому використанні його на персональних комп'ютерах з ціллю навчання методиці сліпого друку та вивчення азбуки Морзе.