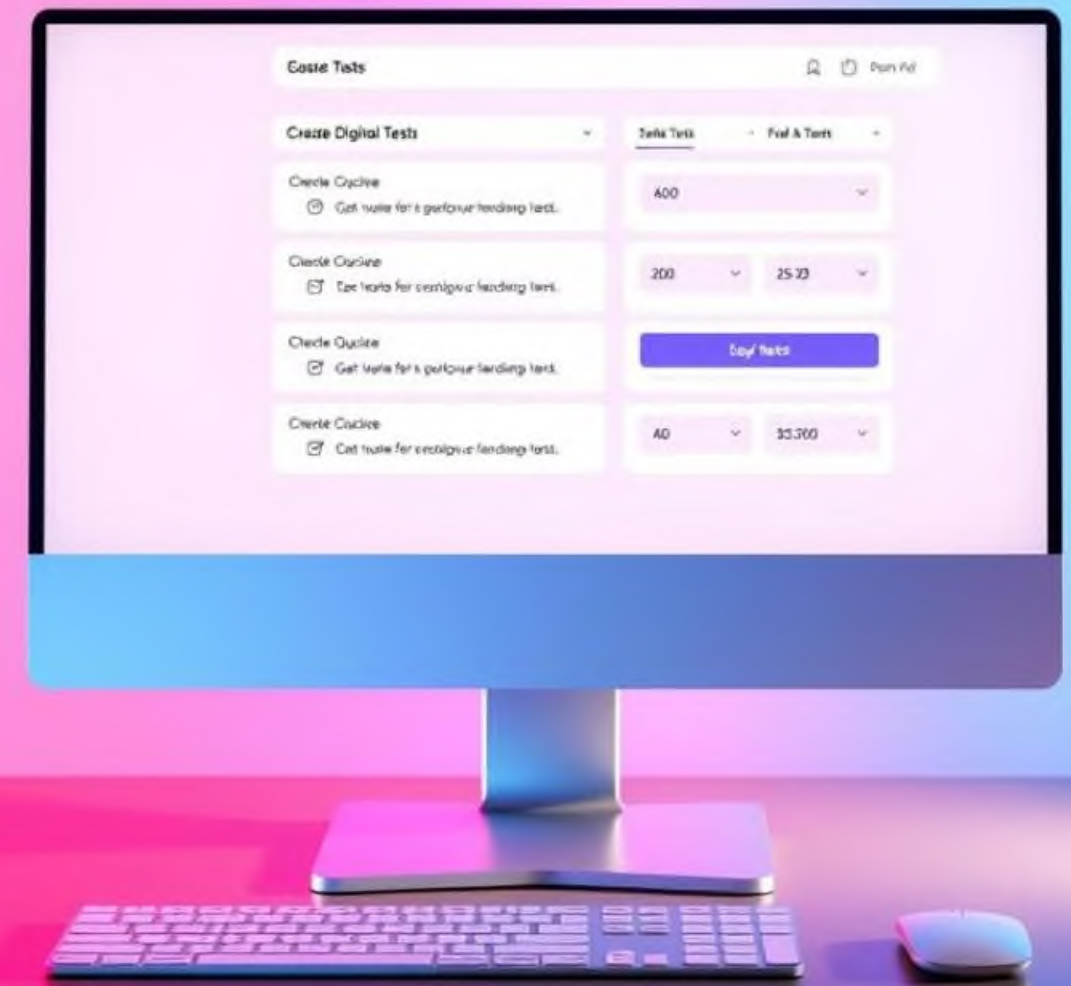


Десктопний застосунок для створення та управління тестуванням

Метою проєкту є розробка десктопного застосунку для створення, проведення та управління тестуванням у навчальних закладах, на курсах або для індивідуальних занять. Застосунок дозволить викладачам створювати тести з різних предметів, призначати їх студентам, автоматично оцінювати результати і формувати звіти. Це оптимізує процес контролю знань, спрощує проведення тестування і допомагає в управлінні навчальним процесом.





Проблеми, які вирішує аплікація

1 Організація тестування

Створення та проведення тестів для великої кількості студентів може бути складним і вимагати багато часу.

2 Автоматизація перевірки

Відсутність автоматизації процесу перевірки знань перекладає навантаження на викладачів, що може призвести до виснаження та помилок.

3 Індивідуальні тести

Створення індивідуальних тестів для кожного студента або групи студентів, що вимагає додаткових зусиль та ресурсів.

4 Управління результатами

Відстеження прогресу студентів та аналіз результатів тестів може бути складним без відповідних інструментів.

Типи користувачів аплікації



Викладачі

Створення тестових завдань, призначення тестів, керування процесом тестування та аналіз результатів.



Студенти

Проходження тестів, перегляд результатів та відстеження прогресу.



Адміністратори

Технічна підтримка системи, створення акаунтів користувачів, доступ до загальних результатів.

Ключові вимоги

Інтерфейс користувача

Інтерфейс має бути інтуїтивно зрозумілим і простим у використанні. Він повинен бути розроблений за допомогою WPF для забезпечення сучасного та візуально привабливого дизайну.

Створення тестів

Аплікація повинна дозволяти створювати тести з різними типами питань, включаючи множинний вибір, коротку відповідь, правильно/неправильно та заповнення пропусків.

Оцінювання, звіти та прогрес

Аплікація має автоматично оцінювати тести з об'єктивними відповідями, а також надавати можливість ручної перевірки відкритих питань. Вона повинна генерувати детальні звіти про результати тестування та прогрес студентів.

Збереження даних

Інформація про тести, результати та статистику повинна зберігатися в базі даних. Для цієї мети можна використовувати Entity Framework або ADO.Net.

Ключові вимоги

Аутентифікація

Підтримка ролей для викладачів, студентів та адміністраторів, кожен з яких має свій рівень доступу до функцій.

Офлайн-режим

Можливість роботи без підключення до інтернету з подальшою синхронізацією результатів.

Технології, що використовуються в проекті

.NET Framework

Основна платформа для розробки, що забезпечує всі необхідні інструменти для створення десктопних додатків. .NET Framework дозволяє використовувати інтеграцію баз даних, багатофункціональні інтерфейси та обробку бізнес-логіки, що забезпечує стабільність і масштабованість проекту.

WPF (Windows Presentation Foundation)

WPF відповідає за створення графічного інтерфейсу. Дозволяє розробити сучасний та інтуїтивно зрозумілий інтерфейс для викладачів, студентів та адміністраторів. WPF підтримує анімацію та графіку, що покращує візуальне сприйняття програми.

Entity Framework (EF)

EF забезпечує спрощений доступ до бази даних, що дозволяє викладачам створювати тести та зберігати їх, а також отримувати інформацію про результати тестування студентів. EF автоматизує роботу з запитам SQL, знижуючи ризики помилок.

ADO.NET

ADO.NET дозволяє взаємодіяти з базою даних на низькому рівні, що забезпечує більший контроль над транзакціями та запитам SQL. Це корисно для оптимізації окремих операцій доступу до бази даних, наприклад, при роботі з великими обсягами даних.

Аналіз конкурентів

ExamView:

Функціонал:

- **Створення тестів:** ExamView дозволяє викладачам створювати тести на основі заздалегідь створених баз даних питань. Є підтримка різних типів питань: Multiple choice, True/False, Fill gap.
- **Банк питань:** система має вбудовану функцію управління банком питань, що дозволяє користувачам організовувати питання за темами, розділами чи рівнем складності.
- **Оцінювання:** автоматичне оцінювання тестів з можливістю детальної аналітики результатів.
- **Звіти:** викладачі можуть отримувати детальні звіти про результати студентів, прогрес у навчанні, кількість правильно виконаних завдань і помилок.
- **Проходження тестів** — студенти можуть виконувати тести на комп'ютерах без Інтернету.
- **Збереження результатів** — результати зберігаються локально і можуть бути експортовані в різні формати (PDF, друк тощо).



ExamView

Переваги

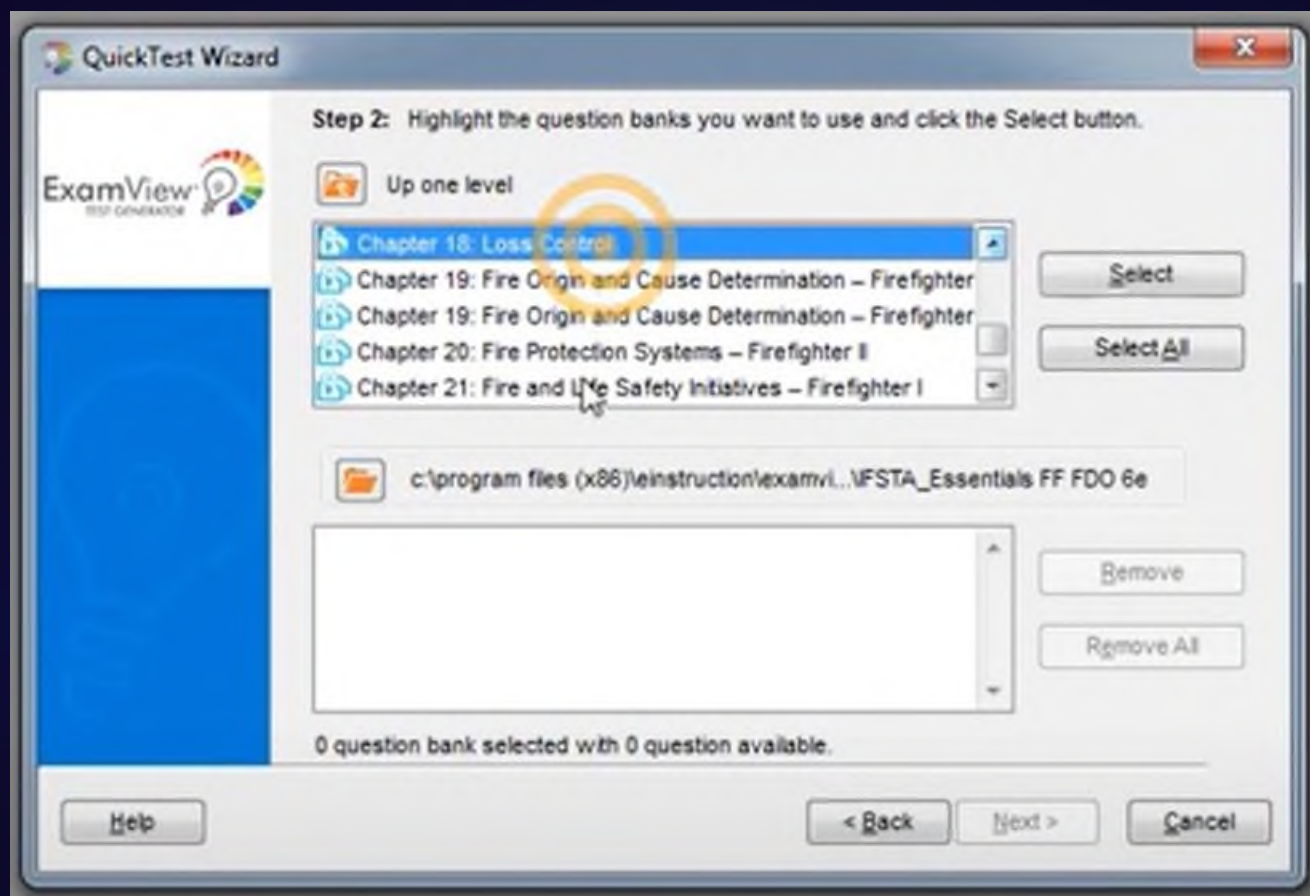
Викладачі можуть легко створювати тести за допомогою "QuickTest Wizard". Програма має інтуїтивний інтерфейс, що спрощує використання для нових користувачів.

ExamView відомий своєю простотою використання та гнучкістю. Викладачі можуть налаштовувати складність тестів, змінювати порядок питань та додавати мультимедійні матеріали.

Недоліки

Інтерфейс ExamView може здаватися застарілим у порівнянні з новими рішеннями. Доступні лише базові функції аналізу результатів та управління тестами.

Відсутність хмарних функцій може ускладнити спільну роботу з тестами для кількох викладачів.



На зображенні видно вікно програми **ExamView**, яке дозволяє створювати тести. Програма має "**QuickTest Wizard**", що спрощує процес створення тестів. У поточному вікні користувач обирає запитання з банку запитань, розділених на глави (наприклад, безпека пожежників або комунікації в пожежній частині). Інтерфейс дає змогу обрати потрібні глави або весь банк запитань за допомогою кнопок "**Select**" і "**Select All**".

Загалом, **ExamView** відомий своєю зручністю для викладачів, які хочуть швидко створювати тести, використовуючи готові питання з банку. Загалом, **ExamView** відомий своєю зручністю для викладачів, які хочуть швидко створювати тести, використовуючи готові питання з банку.

Quiz Faber



Q Create a question

Out. Cettion

Get are question, choicer, Qulswar.
Yot wile are queve nfomulzslple aromering potive.

Create Question:

☐ Multiple chize

☐ Multiple choice

Question >

☐ Multiple answer

Question >

☐ True/talse

10 Cuestion >

☐ True answer

Q Matching >

☐ Open answer

C Custom >

☐ Fill gap

11 Cusstighton >

Cancel

QuizFaber

Створення тестів

QuizFaber дозволяє створювати різні типи тестів, від простих вікторин до складних оцінювальних завдань.

Робота офлайн

QuizFaber може працювати без доступу до Інтернету, що робить його зручним для викладачів, які працюють в умовах обмеженого підключення.

Звіти та аналітика

QuizFaber надає базові звіти та аналітику, що дозволяють викладачам отримати загальне уявлення про результати студентів.

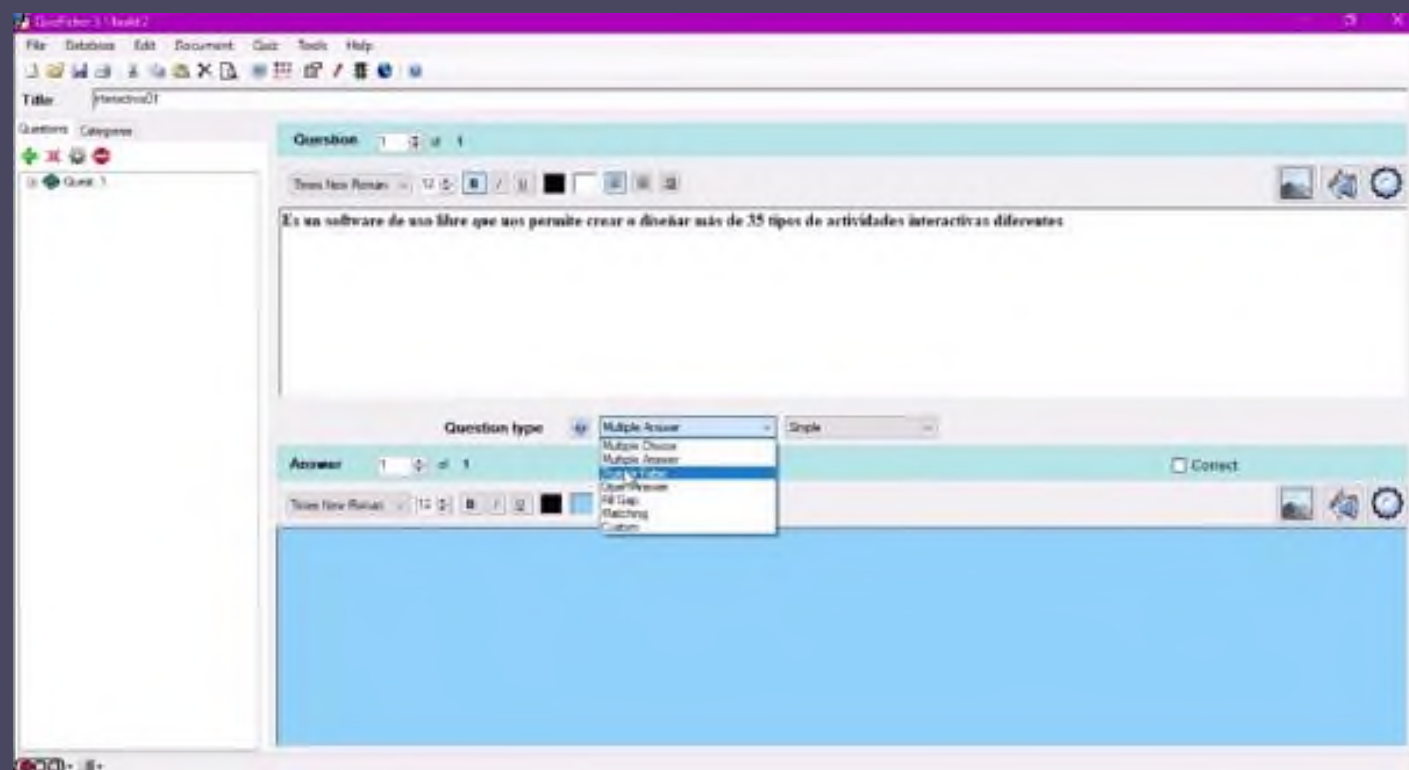
QuizFaber

Переваги

- Легкий у використанні і швидкий у встановленні.
- Підтримує локальне виконання тестів, що зручно для викладачів, які не мають доступу до постійного підключення до Інтернету.
- Добре підходить для невеликих навчальних програм або індивідуальних викладачів.

Недоліки

- Відсутність розширених можливостей для аналізу результатів та керування великими базами запитань.
- Обмежений сучасний інтерфейс, що може здатися застарілим для молодших користувачів.
- Менша кількість налаштувань для управління великими тестовими програмами або багатетапними тестами.



На зображенні видно інтерфейс програми, що дозволяє користувачеві створювати нові запитання з різними типами відповідей: Multiple choice, Multiple answer, True/False, Open answer, Fill gap, Matching, Custom.Є можливість форматовувати текст запитання і відповідей (шрифти, кольори), додавати зображення та інші мультимедійні елементи (значки фото та аудіо праворуч від питання). Зліва видно структуру тесту — користувач може додавати кілька запитань і організовувати їх за категоріями.

Покращення та Доповнення

1 Сучасний Інтерфейс

Інтерфейс обох програм виглядає застарілим. Для нашого проєкту варто реалізувати більш сучасний і зручний інтерфейс з використанням технології **WPF (Windows Presentation Foundation)**. Це дозволить створити інтуїтивніший і більш візуально привабливий дизайн. Наприклад, можна використовувати сучасні стилі кнопок, таблиць і анімації для полегшення навігації користувачів між різними функціями програми.

2 Мультикористувацький Режим

Реалізація підтримки декількох користувачів (викладачів, учнів та адміністраторів) зі своїми рівнями доступу. Наприклад, викладачі можуть створювати тести і призначати їх учням, а учні можуть проходити ці тести. Адміністратори ж можуть керувати системою, керувати обліковими записами та базами даних. Викладачі матимуть доступ до створення та редагування тестів, а також перегляду аналітики та звітів. Учні матимуть можливість лише проходити тести та переглядати свої результати.

3 База Даних та Аналітика

Зберігання результатів тестів у базі даних (наприклад, за допомогою ADO.Net або Entity Framework) дозволить користувачам переглядати історію тестувань, результати за певний період і відстежувати прогрес. Можна додати просту аналітику, яка дозволить викладачам бачити загальний прогрес учнів, кількість правильних і неправильних відповідей, середній час на виконання тестів і результати в різних категоріях.

Функції, які варто запозичити



Типи запитань

З обох застосунків варто взяти різноманітні типи запитань: Multiple choice, True/False, Fill gap, Matching. Однак варто також додати підтримку Open answer як це є в **QuizFaber**. Це забезпечить більшу гнучкість при створенні тестів.



Мультимедійні можливості

Як **ExamView**, так і **QuizFaber** підтримують додавання зображень та інших мультимедійних елементів. Це варто реалізувати і в нашому проєкті, щоб викладачі могли додавати візуальні або аудіо матеріали для складніших завдань.



Організація питань

Можливість категоризації питань за темами чи розділами, як у обох застосунках, допоможе викладачам швидко знаходити потрібні запитання і використовувати їх для тестів. Це важливо реалізувати у нашій системі для зручності.

Висновок

Ми детально розглянули можливості для створення десктопного застосунку для управління тестуванням. Ми виявили, що існуючі рішення, такі як ExamView та QuizFaber, мають багато цінних функцій, які ми можемо адаптувати до наших потреб.

