

^{1,3}Т.С. Картбаев, ²В.А. Лахно, ^{3,4}А.А. Тургынбаева *,
^{3,4}Ж.Т. Айтуганова, ³Б.А. Талпакова

¹Алматинская Академия Министерства Внутренних Дел РК, Казахстан

²Национальный университет биоресурсов и природопользования Украины, Киев, Украина

³Алматинский технологический университет, Алматы, Казахстан

⁴Казахский национальный университет им. аль-Фараби, Алматы, Казахстан

*e-mail: aliza1979@mail.ru

ВЫБОР И СРЕД ПРОЕКТИРОВАНИЯ ДЛЯ СППР С ИСПОЛЬЗОВАНИЕМ ПАРАДИГМЫ СБОРОЧНОГО ПРОГРАММИРОВАНИЯ

Аннотация. В статье рассмотрены основные инструментальные среды проектирования представленных на рынке ЭС и СППР (СППР Project Expert) и (СППР «Выбор»), а также учитывая типовые спецификации современных компьютерных систем, в качестве среды разработки для СППР были выбраны среда программирования RAD Studio 10 и Microsoft (Visual Studio), язык программирования С# для организации работы и модуля, который реализует методы и модели.

В статье анализируются основные подходы выбора стратегий инвесторов в СППР в ходе процедуры оценивания инвестиционных проектов в сфере цифровизации предприятий с учетом многофакторности, проект реализован на языке С#. При выборе языка программирования С# для данного модуля были поставлены две основные задачи - скорость разработки и возможность узнавать об ошибках еще на этапе компиляции программы.

Ключевые слова: система поддержки принятия решений, инвестиционный проект, программный продукт, экспертная система, объекты.

Ведение. В настоящее время, в условиях жесткой конкуренции на рынке инвестиций, стремительно растет "цена" ошибки при принятии решений. Для обеспечения высокого профессионального уровня решений используют СППР или экспертные системы, результатом работы которых являются рекомендации для ЛПР. Качество таких решений и соответствующих рекомендаций непосредственно влияет на эффективность выбранной инвесторами стратегии финансирования. Это особенно важно для проектов, связанных с риском потери капитала. Постоянное повышение сложности моделей для СППР в слабо структурированных предметных областях, к которым можно отнести и оценку стратегий инвестора, требует адекватного и детального отображения совокупности факторов и их взаимосвязей, в частности, в БЗ СППР.

Стремительное развитие в последние десятилетия различных видов компьютерной техники поставило перед пользователями различных программных продуктов (ПП), в том числе СППР, а также перед специалистами разработчиками вопросы и проблемы, связанные с применением новых парадигм программирования, и прежде всего сборочного и объектно-ориентированного (ООП). Такие подходы позволяют обеспечить высокоэффективную разработку больших проектов, когда над них воплощением в жизнь работает сразу несколько программистов или даже целых команд. И как показал опыт создания многих коммерческих ПП, одним из путей решения подобной задачи становятся решения, основанные на совершенствовании технологий модульного программирования. Когда во главу угла ставится переход на промышленные методики разработки ПП.

Это позволит, в случае необходимости, используя промышленный подход, перейти к изготовлению сложных программных систем (ПС), которые состоят из более простых ПП, написанных в разных средах программирования и на разных языках.

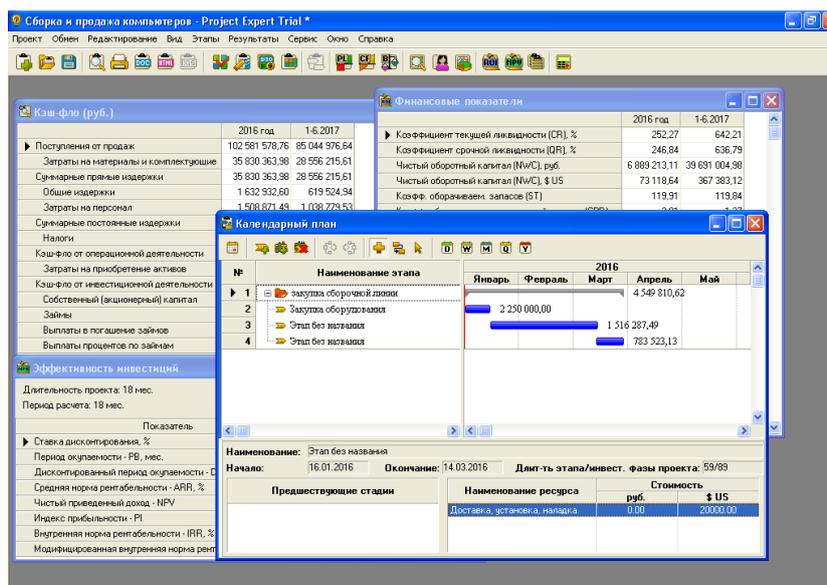
Основная часть. На основании анализа опыта проектирования представленных на рынке ЭС и СППР, см. рис. 1.1 (СППР Project Expert) 1.2 (СППР «Выбор»), а также учитывая типовые спецификации современных компьютерных систем, в качестве среды разработки для СППР были выбраны:

• Физико-математические науки

1) среда программирования RAD Studio 10 [1–5]. В данной среде программирования использовался продукт Delphi 10. Такой выбор, прежде всего продиктован, теми широкими возможностями, которые данная среда программирования предоставляет разработчикам для реализации сложных проектов под операционную систему Windows. Программный код, также достаточно легко может быть адаптирован для ОС Linux, поскольку код можно трансформировать, используя возможности среды разработки Lazarus.

2) Microsoft (Visual Studio) язык программирования C# для организации работы, модуля, который реализует методы и модели.

а)



б)

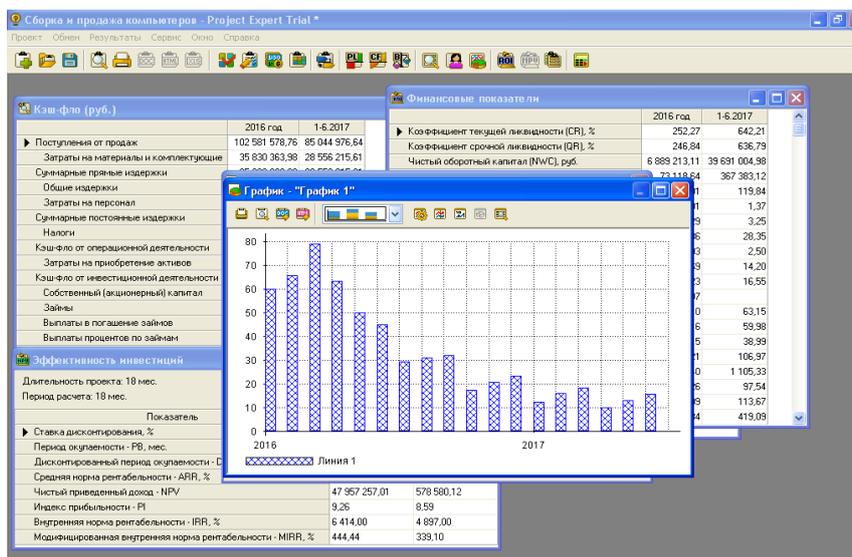
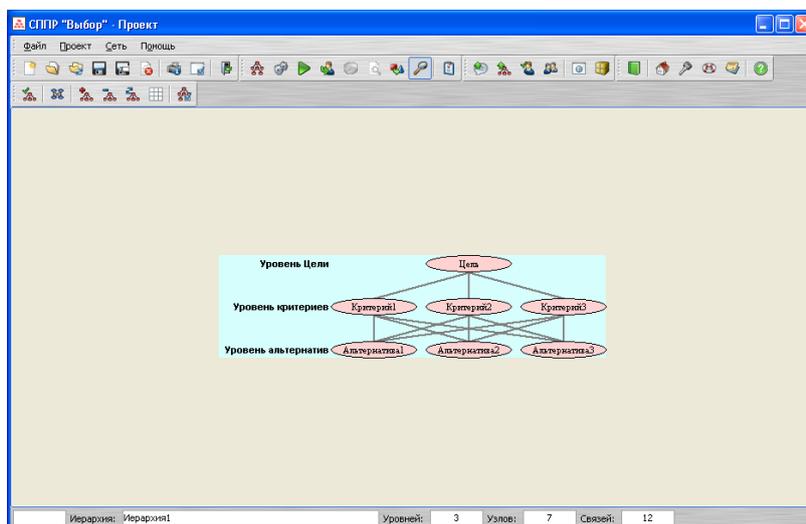


Рисунок 1.1. Пример оконного интерфейса СППР Project Expert

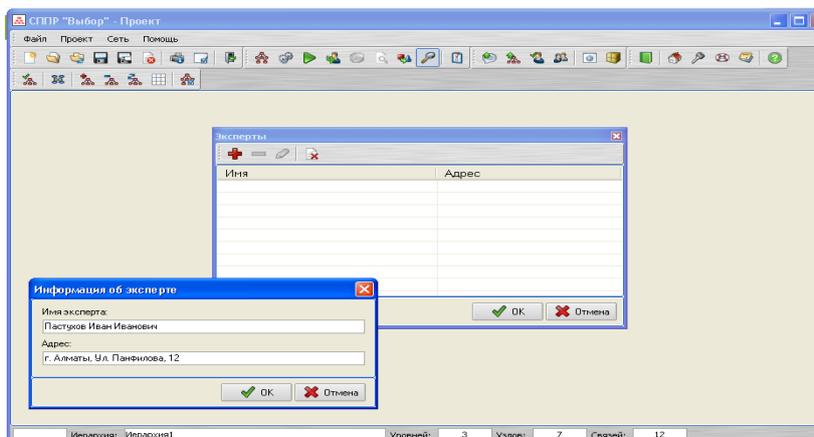
Delphi 10 – это продукт, предназначенный для быстрого создания приложений, и поддерживающий концепцию RAD (Rapid Application Development – быстрая разработка приложений). Обращает на себя внимание, что процессы, связанные с созданием интерфейсов будущей СППР, являются интуитивно понятными и достаточно простыми даже для рядового программиста, который способен поддерживать процесс разработки на

протяжении всего жизненного цикла ПП [2–6]. Именно поэтому Delphi 10, относящийся к RAD-средам проектирования, можно считать наиболее подходящей платформой для визуальной среды разработки. Это позволяет уже на стадии проектирования непосредственно видеть рабочие и диалоговые окна будущего ПП. То есть проектировщик видит уже фактический готовый интерфейс СППР, а процедуры и функции, отвечающие за вывод решений, синтезируемых СППР.

а)



б)



в)

Рисунок 1.2. Пример оконного интерфейса СППР «Выбор»

• Физико-математические науки

Delphi 10 является достаточно высокопроизводительным инструментом, который включает в себя компилятор кода и средства визуального проектирования. Аналогичный способ используется в Microsoft Visual Studio.

В основе Delphi 10 лежит язык Object Pascal. Заметим, что Object Pascal стал расширением распространенного в 80-90-е годы XX-го века объектно-ориентированного языка Pascal.

В Delphi 10 также вошли:

улучшенные библиотеки с десятками визуальных компонентов;

новые встроенные генераторы отчетов;

разнообразный инструментарий для профессиональной разработки информационных систем, экспертных систем и СППР или же традиционных ПП для операционных систем семейства Windows.

Таким образом, прежде всего Delphi 10 предназначен для разработчиков, занимающихся написанием ПП для компаний любого размера от небольших предприятий до трансконтинентальных корпораций. Одним из главных преимуществ данной среды проектирования стала возможность оперативно разрабатывать ПП, рассчитанные под архитектуру клиент-сервер, во взаимодействии с такими СУБД как MySQL и Interbase.

Среда проектирования Delphi позволяет компилировать достаточно компактные и высокоэффективные выполняемые модули (.exe), а также динамически присоединяемые библиотеки (.dll), что особенно актуально при разработке модульных систем, к которым можно отнести и разрабатываемую нами СППР. В свою очередь компактные и быстро работающие модули сложных ПП, существенно снижают нагрузку на клиентские рабочие места (или клиентские рабочие станции), что в конечном счете также имеет важное значение для пользователей.

Преимущества Delphi 10 в сравнении с сопоставимыми ПП, например, таких компаний как: Microsoft (Visual Studio); Eclipse CDT, можно сформулировать так:

быстрая разработка приложения (RAD);

хорошая производительность разработанных ПП;

невысокие требования разработанных ПП к ресурсам компьютеров или клиентских рабочих станций;

возможность масштабировать приложение за счет добавления или имплементации новых компонент;

возможность разрабатывать собственные новые компоненты или инструменты, например, когда потенциального заказчика не устраивает классический компонент, установленный по умолчанию. Заметим, что доступ к существующим компонентам и инструментам Delphi 10 можно просмотреть и организовать в исходных кодах;

хорошая читабельность иерархии объектов в проекте.

Для того чтобы окончательно обосновать, почему наш выбор остановился на RAD Studio 10, кратко перечислим некоторые недостатки языка C++ в сравнении с Object Pascal:

1. Надо делать много инициализаций (например, выполнять регистрацию классов окон, организацию циклов обработки сообщений, выполнять процедуры, связанные с созданием оконных функций, выбором пиктограмм для окон и прочие задачи, возникающие в ходе реализации проекта). В крупных проектах таких как СППР и ЭС необходимы навыки системного программиста, а также, знание в совершенстве функций WinApi. Заметим, что в Delphi 10 процессы, связанные с системным программированием уже имплементированы и непосредственно встроены в среду разработки RAD Studio 10. При этом инициализация по умолчанию уже работает. Таким образом у программиста имеется возможность непосредственно сосредоточиться на алгоритмах или сделать акцент в проекте на организации взаимодействия главного файла (.exe) с динамически подсоединяемыми библиотеками или другими *.exe модулями.

2. По признанию многих практикующих разработчиков [2, 4, 6, 7], имеющих опыт проектирования программных продуктов как в Delphi 10, так и в C++ (Visual Studio), синтаксис последнего сложнее, а также необходимо постоянно учитывать нюансы меняющихся стандартов языка C++. Хотя код C++ и выглядит более компактно и для опытного программиста более понятен при восприятии.

3. Существенной сложностью написания кода на C++ остается необходимость постоянно отслеживать регистр символов. Так как C++ чувствителен к регистрам, в которых набраны символы.

4. В Delphi 10 классы (объекты) можно располагать только в динамической памяти. Тогда как в C++ это можно делать как в статической или динамической памяти, так и в стеке. Это делает приложения на Delphi более безопасным с точки зрения кибербезопасности и защиты от взлома.

5. Как уже указывалось выше, достаточно просто приложения, разработанные в среде Delphi под ОС Windows, можно трансформировать под среду программирования Lazarus. Данная среда программирования внешне похожа на Delphi, а основой Lazarus стал FreePascal.

Однако пока размеры разрабатываемых на Lazarus приложений, все еще велики, хотя среда стремительно совершенствуется и по мере развития, этот недостаток, очевидно будет устранен.

Также пока Lazarus не поддерживает работу динамических библиотек. Потому программные продукты Lazarus должны включать в себя все используемые пакеты.

6. Для запуска Delphi 10 не требуется мощная аппаратная платформа. Достаточно будет 4 Гб ОЗУ и процессор с одним, двумя ядрами.

7. Небольшие программные продукты, разработанные с помощью в Delphi 10, смогут работать на любых даже маломощных ПК, а для их повседневной эксплуатации не требуется ни большой объем оперативной памяти, ни производительный процессор.

Еще одним преимуществом данной среды разработки именно для информационных систем или систем поддержки принятия решений, стала возможность использования представления ПП как MDI (Multiple Document Interface). То есть приложение можно представить, как несколько дочерних окон, располагающихся внутри главного окна программы.

Модуль выбора стратегий инвесторов в СППР в ходе процедуры оценивания инвестиционных проектов в сфере цифровизации предприятий с учетом многофакторности проект реализован на языке C #.

При выборе языка программирования C # для данного модуля были поставлены две основные задачи - скорость разработки и возможность узнавать об ошибках еще на этапе компиляции программы. Язык C # поддерживает одновременно функциональный и объектно-ориентированный подходы, что и предопределило выбор этого языка в качестве базового для разработки финансового модуля СППР. В качестве среды разработки была выбрана среда Visual Studio 2019.

Выводы. Сделан анализ проектирования представленных на рынке ЭС и СППР (СППР Project Expert) и (СППР «Выбор»).

Были выбраны в качестве среды разработки для СППР:

1) среда программирования RAD Studio 10. В данной среде программирования использовался продукт Delphi 10. Такой выбор, прежде всего продиктован, теми широкими возможностями, которые данная среда программирования предоставляет разработчикам для реализации сложных проектов под операционную систему Windows. Программный код, также достаточно легко может быть адаптирован для ОС Linux, поскольку код можно трансформировать, используя возможности среды разработки Lazarus.

2) Microsoft (Visual Studio) язык программирования C# для организации работы, модуля, который реализует методы и модели.

ЛИТЕРАТУРА

- [1] Дарахвелидзе, П., & Марков, Е. (2003). Программирование в Delphi 7. БХВ-петербург., 1003 с.
[2] Осипов, Д. (2006). Delphi. Профессиональное программирование. СПб.: Символ-Плюс, 856 с.
[3] Фленов, М. Е. (2012). Библия Delphi (3-е издание). БХВ-Петербург, 946 с.
[4] Осипов, Д. Л. (2011). Базы данных и Delphi. Теория и практика. БХВ-Петербург, 840 с.
[5] Корняков, В. Н. (2005). Программирование документов и приложений MS Office в Delphi. БХВ-Петербург, 826 с.
[6] Novák, I. (2011). Beginning Microsoft Visual Studio LightSwitch Development. John Wiley & Sons.
[7] Karaci, A. (2011). Performance Comparison of Managed C# and Delphi Prism in Visual Studio and Unmanaged Delphi 2009 and C++ Builder 2009 Languages. International Journal of Computer Applications, 26(1), 9-15.

REFERENCES

- [1] Darakhvelidze, P., & Markov, E. (2003). Programmirovaniye v Delphi 7. BKhV-peterburg., 1003 с.
[2] Osipov, D. (2006). Delphi. Professional'noe programmirovaniye. SPb.: Simvol-Plyus, 856 с.
[3] Flenov, M. E. (2012). Bibliya Delphi (3-e izdanie). BKhV-Peterburg, 946 с.
[4] Osipov, D. L. (2011). Bazy dannykh i Delphi. Teoriya i praktika. BKhV-Peterburg, 840 с.
[5] Korniyakov, V. N. (2005). Programmirovaniye dokumentov i prilozhenii MS Office v Delphi. BKhV-Peterburg, 826 с.
[6] Novák, I. (2011). Beginning Microsoft Visual Studio LightSwitch Development. John Wiley & Sons.
[7] Karaci, A. (2011). Performance Comparison of Managed C# and Delphi Prism in Visual Studio and Unmanaged Delphi 2009 and C++ Builder 2009 Languages. International Journal of Computer Applications, 26(1), 9-15.

^{1,3}Т.С. Қартбаев, ²В.А. Лахно, ^{3,4}А.А. Тұрғынбаева*, ^{3,4}Ж.Т. Айтуғанова, ³Б.Ә. Талпақова

¹Қазақстан Республикасы ішкі істер министрлігінің Алматы академиясы, Қазақстан

⁴әл-Фараби атындағы Қазақ ұлттық университеті, Алматы, Қазақстан

³Алматы технологиялық университеті, Алматы, Қазақстан

²Украинаның биоресурстар және табиғатты пайдалану ұлттық университеті, Киев, Украина

*e-mail: aliza1979@mail.ru

**ҚҰРАСТЫРУ БАҒДАРЛАМАЛАУ ПАРАДИГМАСЫН ҚОЛДАНА ОТЫРЫП,
ШҚҚЖ ҮШІН ТАҢДАУ ЖӘНЕ ЖОБАЛАУ ОРТАСЫ**

Андатпа. Мақалада СЖ және ШҚҚЖ (ШҚҚЖ Project Expert) және (ШҚҚЖ «Выбор») нарығында ұсынылған негізгі аспаптық жобалау орталары қарастырылған, сондай-ақ, заманауи компьютерлік жүйелердің типтік сипаттамаларын ескере отырып, RAD Studio 10 бағдарламалау ортасы және Microsoft (Visual Studio), C# бағдарламалау тілі әдістер мен модельдерді жүзеге асыратын жұмыс пен модульді ұйымдастыруға арналған ШҚҚЖ үшін даму ортасы ретінде таңдалды.

Мақалада көпфакторлықты ескере отырып, кәсіпорындарды цифрландыру саласындағы инвестициялық жобаларды бағалау рәсімі барысында ШҚҚЖ-де инвесторлардың стратегияларын таңдаудың негізгі тәсілдері талданады, жоба C # тілінде іске асырылған. Осы модуль үшін C # бағдарламалау тілін таңдағанда екі негізгі міндет қойылды - даму жылдамдығы және бағдарламаны құрастыру сатысында қателер туралы білу мүмкіндігі.

Түйінді сөздер: шешімдерді қабылдауды қолдау жүйесі, инвестициялық жоба, бағдарламалық өнім, эксперттік жүйе, объектілер.

^{1,3}T.S. Kartbayev, ²V.A. Lakhno, ^{3,4}A.A. Turgynbayeva*, ^{3,4}Zh.T. Aytuganova, ³B.A. Talpakova

¹Almaty academy of the Ministry of internal affairs of the Republic of Kazakhstan, Kazakhstan

²National University of Life and Environmental Sciences of Ukraine, Kiev, Ukraine

³Almaty Technological University, Almaty, Kazakhstan

⁴al-Farabi Kazakh National University, Almaty, Kazakhstan

*e-mail: aliza1979@mail.ru

SELECTION AND DESIGN ENVIRONMENTS FOR DSS USING THE ASSEMBLY PROGRAMMING PARADIGM

Abstract. The paper considers the main tool design environments of the ES and DSS (DSS Project Expert) and (DSS «Выбор») presented on the market and taking into account the typical specifications of modern computer systems, the RAD Studio 10 programming environment was chosen as the development environment for the DSS and Microsoft (Visual Studio), C# programming language for organizing work and modules that implement methods and models.

The article analyzes the main approaches to choosing strategies for investors in the DSS during the procedure for evaluating investment projects in the field of digitalization of enterprises, taking into account multi-factorism, the project is implemented in C#. When choosing the C# programming language for this module, two main tasks were set - the speed of development and the ability to learn about errors even at the program compilation stage.

Keywords: decision support system, knowledge base, model base, expert system, keywords, objects.