

УДК 631.152:330.131.7

UDC 631.152:330.131.7

**ПРИНЦИПЫ ПОСТРОЕНИЯ И ЭТАПЫ
СОЗДАНИЯ ПРОГРАММНОГО КОМПЛЕКСА
ПО АНАЛИЗУ И ОЦЕНКЕ РИСКОВ НА
ПРЕДПРИЯТИЯХ АПК**

**PRINCIPLES OF CONSTRUCTION AND
STAGES OF CREATION OF PROGRAMMING
COMPLEX ON ANALYSIS AND ASSESSMENT
OF RISKS AT AGRO-INDUSTRIAL COMPLEX
ENTERPRISES**

Ефанова Наталья Владимировна
старший преподаватель

Efanova Natalia Vladimirovna
senior lecturer

*Кубанский государственный аграрный
университет, Краснодар, Россия*

Kuban State Agrarian University, Krasnodar, Russia

В статье рассмотрены принципы построения и этапы создания программного комплекса по анализу и оценке рисков на предприятиях АПК, включающие разработку общей архитектуры программного комплекса, логической структуры программных модулей, входящих в состав программного комплекса, физической структуры программного комплекса на уровне файловой системы и алгоритма функционирования программного обеспечения комплекса.

Principles of construction and stages of programming complex on analysis and risks assessment at the enterprises of agro-industrial complex, including design of general architecture of programming complex, logical structure of programming models, being the part of programming complex, physical structure of programming complex on the level of file system and algorithm of functioning of complex software are considered in the article.

Ключевые слова: ПРОГРАММНЫЙ КОМПЛЕКС, АНАЛИЗ И ОЦЕНКА РИСКОВ, ПРЕДПРИЯТИЯ АПК, АРХИТЕКТУРА, ЛОГИЧЕСКАЯ СТРУКТУРА ПРОГРАММНЫХ МОДУЛЕЙ, АЛГОРИТМ ФУНКЦИОНИРОВАНИЯ.

Key words: PROGRAMMING COMPLEX, ANALYSIS AND ASSESSMENT OF RISKS, ENTERPRISES OF AIC, LOGICAL STRUCTURE OF PROGRAMMING MODULES, ALGORITHM OF FUNCTIONING.

В статьях [1, 2, 3, 4] представлен ряд разработанных автором математических моделей для анализа и оценки рисков на предприятиях АПК. Модели отличаются сложностью проведения расчетов вручную. Поэтому была поставлена задача проектирования и разработки программного комплекса для автоматизации процесса моделирования и проведения экспериментов на моделях.

Базовыми принципами построения программного комплекса для анализа и количественной оценки риска на предприятиях АПК (далее ПК) являются следующие:

- *принцип системного подхода*, базирующийся на проведенной систематизации накопленных знаний и опыта;
- *принцип концентрации информационных ресурсов* для обеспечения максимальной защиты данных и их достоверности;

- *принцип независимости* создаваемого программного продукта от разработчика на стадии эксплуатации;

- *принцип открытости системы* для наращивания и модернизации путем последовательной разработки, внедрения или замены отдельных модулей.

Принцип системного подхода подразумевает разработку единой концепции и архитектуры программного комплекса, обеспечивающего интеграцию разработанных автором моделей и методик количественного анализа и оценки рисков АПК в единый программный комплекс.

Под принципом концентрации информационных ресурсов в данном контексте понимается выбор платформы, интерфейса и СУБД для организации хранения и доступа к данным.

Принцип независимости подразумевает создание такого программного продукта, после внедрения которого для его нормальной работы не требуется постоянного контакта с разработчиком со стороны пользователя.

Под принципом открытости системы понимается создание модульной архитектуры программного комплекса, позволяющей в случае необходимости с легкостью добавлять или модернизировать его отдельные части, при этом производимые действия не отражаются на работоспособности всей системы в целом.

Перечисленные принципы гарантируют создание эффективного, надежного и перспективного программного комплекса для количественной оценки и анализа рисков для предприятий агропромышленного комплекса.

Для разработки ПК необходимо выполнение следующих этапов:

1) Выбор платформы, интерфейса и СУБД для организации процесса хранения, обработки и доступа к информации.

2) Разработка архитектуры программного комплекса по модульному принципу, что обеспечит достижение этапности внедрения, оптимальности

в комплектации и последующем его развитии в соответствии с возрастающими потребностями предприятия.

3) Разработка соответствующего программного обеспечения, обладающего простым, удобным, легко осваиваемым интерфейсом, который должен предоставить конечному пользователю все необходимые для его работы функции, но в то же время не дать ему возможности выполнять какие-либо лишние действия.

Этап 1. Организация процесса доступа к данным

Программный комплекс рассчитан для работы с относительно небольшими объемами информации и применения в однопользовательском режиме. Поэтому оптимальным является выбор файл-серверной платформы для организации хранения данных, основное преимущество которой заключается в простоте. Интерфейс прикладного программирования ODBC API предоставляет общие методы доступа на основе языка баз данных SQL. Стандарт ANSI SQL включает спецификацию интерфейса на уровне вызовов, на которую опирается ODBC для обеспечения доступа и работы с данными во многих системах управления базами данных. Технологию ODBC разрабатывали как общий, независимый от источников данных способ доступа к данным. Применение технологии должно было также обеспечить переносимость приложений в среду различных баз данных без потребности переработки самих приложений. В этом смысле технология ODBC стала промышленным стандартом, ее поддерживают практически все производители СУБД и средств разработки.

Анализ инструментальных средств разработки файл-серверных приложений позволяет определить и рекомендовать их области применения. СУБД для персональных компьютеров в среде MS-Access могут быть использованы для создания масштабируемых одиночных и групповых информационных приложений и для разработки клиентской

части приложений клиент-сервер, а также как средство автоматизации делопроизводства в составе MS-Office. Инструментальное средство MS Access положительно зарекомендовало себя в разработке файл-серверных приложений с возможностью масштабирования, так как оно имеет удобные средства визуального конструирования, отладки и возможности использования как Access Basic, так и SQL.

Язык С++ применяют для расширения системного программного обеспечения, разработки крупных проектов, специальных приложений, создания библиотек и классов для предметной области, разработки динамических библиотек DLL, создания программного обеспечения для серверов приложений, разработки ОСХ, использования совместно с CASE-системами, обеспечения многоплатформенности и переносимости (по стандарту ANSI). Система визуального программирования Visual Studio 2005 позволяет писать программы на языке С++, а также предоставляет возможность использовать протокол ODBC для связи с базами данных, спроектированными в среде MS Access.

Этап 2. Проектирование и разработка архитектуры ПК

После того, как выбраны средства доступа к данным, необходимо определиться с архитектурой программного комплекса. Существует множество определений понятия «архитектура». По мнению автора, наиболее полно суть данного понятия отражает определение из стандарта IEEE 1471 «Рекомендуемые методы описания архитектуры преимущественно-программных систем IEEE».

Архитектура – это базовая организация системы, воплощенная в ее компонентах, их отношениях между собой и с окружением, а также принципы, определяющие проектирование и развитие системы. В этом стандарте также определяются следующие термины, связанные с данным определением:

1) Система – это набор компонентов, объединенных для выполнения определенной функции или набора функций. Термин "система" охватывает отдельные приложения, системы в традиционном смысле, подсистемы, системы систем, линейки продуктов, семейства продуктов, целые корпорации и другие агрегации, имеющие отношение к данной теме. Система существует для выполнения одной или более миссий в своем окружении.

2) Окружение, или контекст определяет ход и обстоятельства экономических, эксплуатационных, политических и других влияний на систему.

3) Миссия – это применение или действие, для которого одно или несколько заинтересованных лиц планируют использовать систему в соответствии с некоторым набором условий.

4) Заинтересованное лицо – это физическое лицо, группа или организация (или ее категории), которые заинтересованы в системе или имеют связанные с ней задачи.

Для сравнения можно привести еще одно, более сжатое определение архитектуры. Архитектура – это набор значимых решений по поводу организации системы программного обеспечения, набор структурных элементов и их интерфейсов, при помощи которых компонуется система, вместе с их поведением, определяемым во взаимодействии этих элементов, компоновка элементов в постепенно укрупняющиеся подсистемы.

Несмотря на то, что определения архитектуры несколько отличаются, можно увидеть немалую степень сходства. В частности, оба определения указывают на то, что архитектура связана со структурой и поведением. Структура является важнейшей характеристикой архитектуры. Структурный элемент может быть подсистемой, процессом, библиотекой, базой данных, вычислительным узлом, системой в традиционном смысле, готовым продуктом и т. д. Наряду с определением

структурных элементов, любая архитектура определяет взаимодействия этих структурных элементов, которые обеспечивают желаемое поведение системы.

Выделение слоев. Концепция слоев – одна из общеупотребительных моделей, используемых разработчиками программного обеспечения для разделения сложных систем на более простые части. Выделение слоев является одним из ключевых моментов процесса проектирования и анализа архитектуры. Слои призваны упорядочить архитектурную модель и, следовательно, упрощать ее реализацию и анализ. В архитектуре программной системы слои представляют собой уровни абстракции. Слой $n+1$ использует слой n , следовательно, абстракция понятий слоя $n+1$, по меньшей мере, не ниже чем у слоя n , а в идеале – если архитектура системы эффективна, то его уровень абстракции должен быть выше. Соответственно, слой n скрывает (инкапсулирует) логику работы с понятиями, определенными на этом слое, позволяя, таким образом, слою $n+1$ реализовать работу с более сложными понятиями, организовать более сложную логику, используя выразительные средства нижележащего слоя. Таким образом, интуитивно понятно, что если система разбита на ряд слоев, то слой n – это компонент или набор компонентов системы, которые применяют только компоненты слоя $n-1$ и могут быть использованы только компонентами слоя $n+1$.

На рисунке 1 представлены слои архитектуры разрабатываемого программного комплекса.

Наибольший интерес, с точки зрения дальнейшей разработки, представляет архитектурный слой №3, который включает в себя программное обеспечение для количественного анализа и оценки риска. Внутренняя архитектура данного слоя строится по модульному принципу, где каждый отдельный блок является самостоятельной структурной единицей.

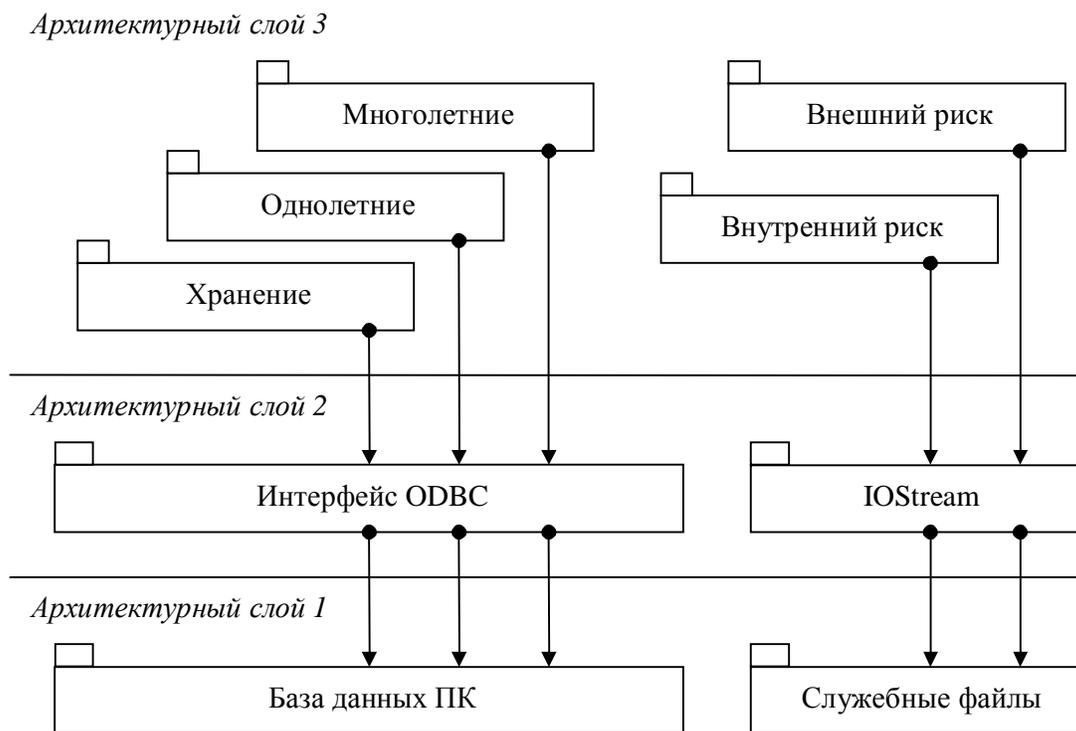


Рисунок 1 – Трехслойная архитектура разрабатываемого ПК

На рисунке 2 представлена обобщенная архитектура программного комплекса для количественного анализа и оценки риска. Из рисунков 1 и 2 видно, что в состав программного комплекса входят пять блоков-модулей программного обеспечения, соответствующих разработанным моделям и методикам анализа и оценки риска. Ряд блоков логически объединен в соответствующие подсистемы. В свою очередь, каждый блок также является системой, обладающей своей уникальной внутренней структурой, элементы которой взаимодействуют друг с другом по некоторым алгоритмам, обеспечивающим текущее поведение системы и ее конечное состояние.

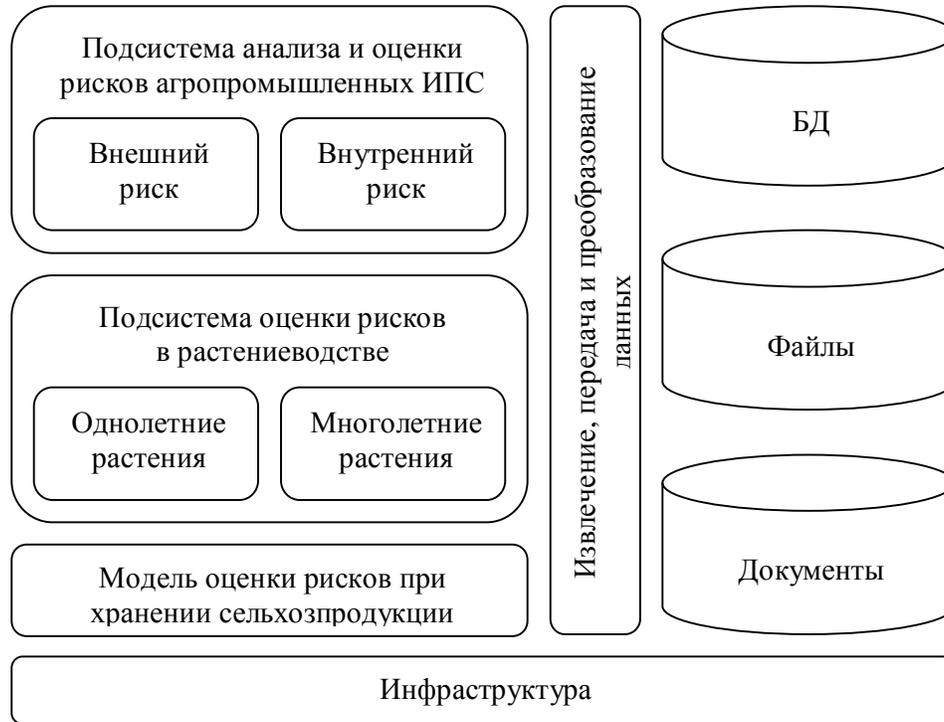


Рисунок 2 – Обобщенная архитектура программного комплекса

Структура модулей

На рисунке 3 представлена блочная структура модуля оценки интегрального показателя внешней среды воздействия ИПС.

На рисунке 4 показана блочная структура модуля анализа эффективности ИПС с учетом рисков составляющей на этапе создания материального потока.

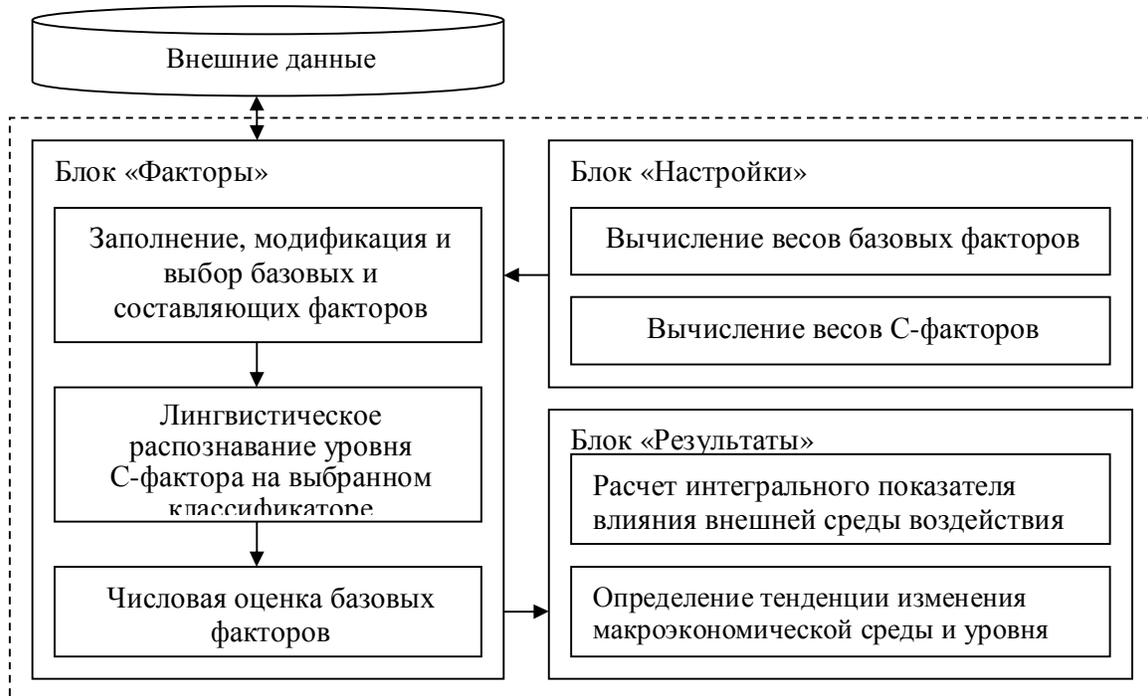


Рисунок 3 – Блочная структура модуля оценки интегрального показателя внешней среды воздействия ИПС

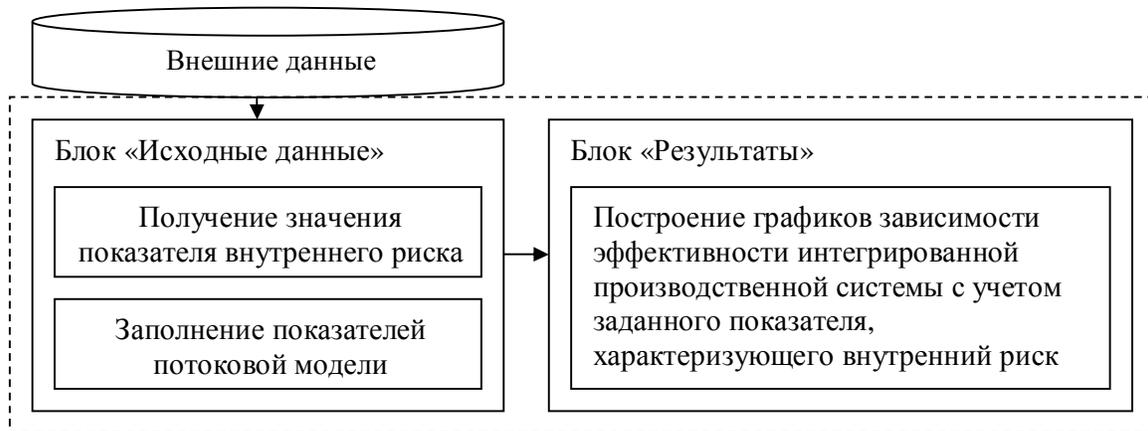


Рисунок 4 – Блочная структура модуля анализа эффективности ИПС с учетом рисков составляющей на этапе создания материального потока

На рисунке 5 представлена блочная структура модуля оценки рисков при посадке многолетних растений, а на рисунке 6 – блочная структура модуля оценки рисков при выращивании однолетних растений.

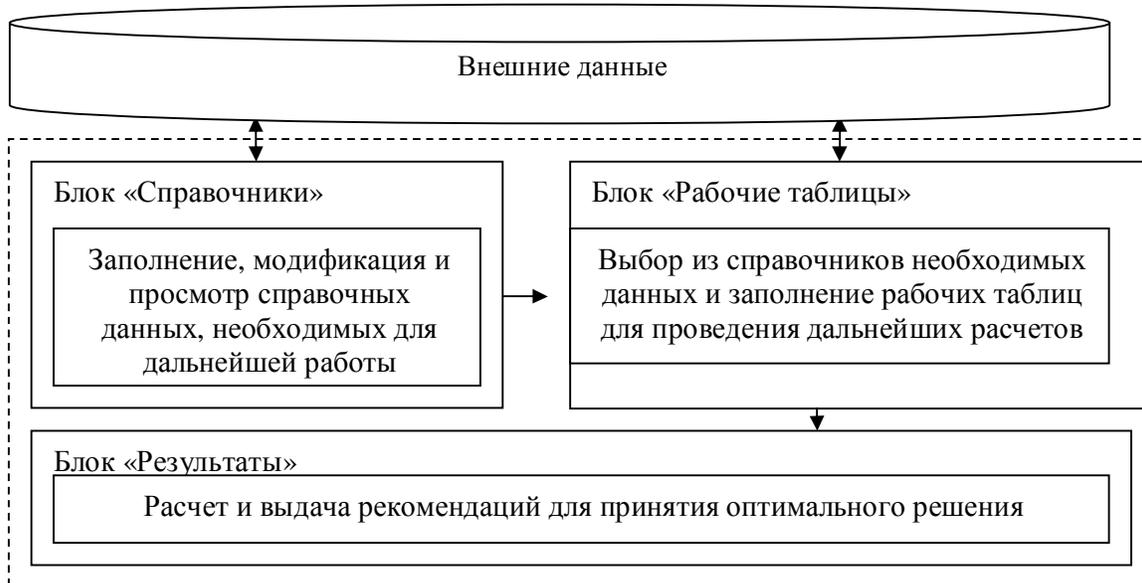


Рисунок 5 – Блочная структура модуля оценки рисков при посадке многолетних растений

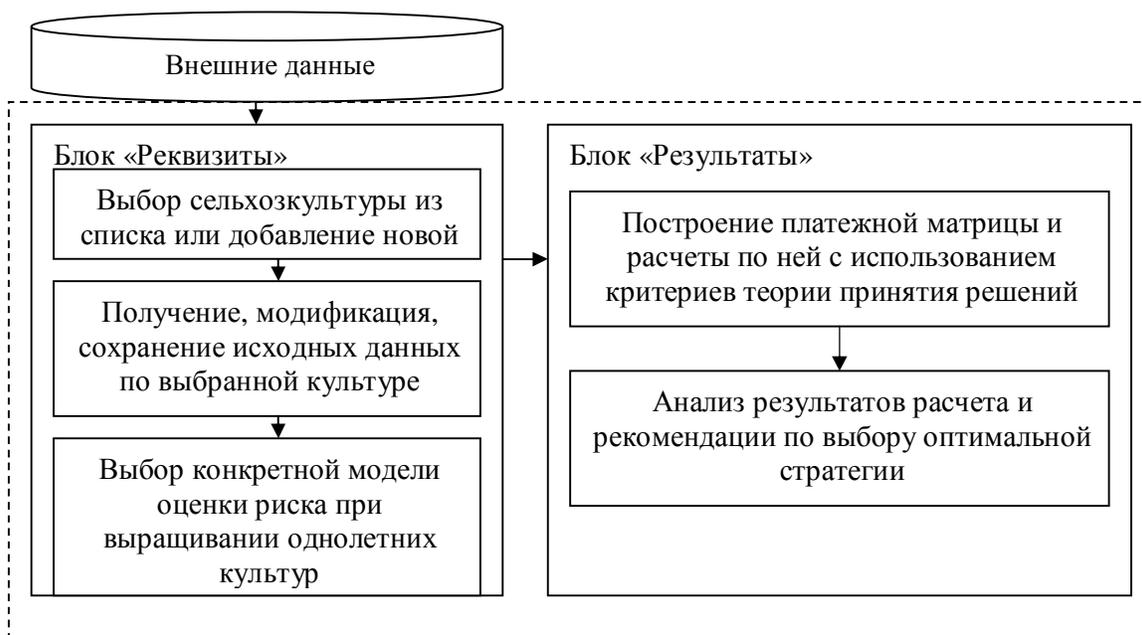


Рисунок 6 – Блочная структура модуля оценки рисков при выращивании однолетних растений

Блочная структура модуля выбора оптимальной стратегии в управлении хранилищем сельскохозяйственной продукции показана на рисунке 7.

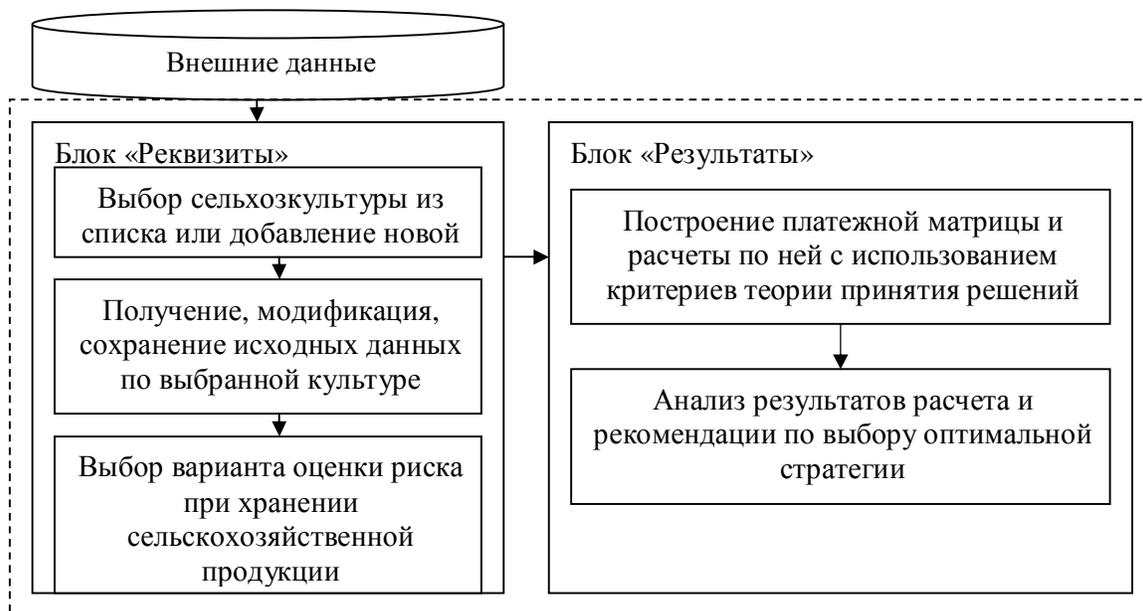


Рисунок 7 – Блочная структура модуля выбора оптимальной стратегии в управлении хранилищем сельскохозяйственной

Разработанная архитектура программного комплекса положена в основу написания соответствующего программного обеспечения. Каждый отдельный модуль является самостоятельной единицей, его модификация никак не отражается на работоспособности всей системы в целом. Еще раз стоит подчеркнуть, что модульность проектирования и построения программного комплекса обеспечивает выполнение принципа открытости системы для наращивания и модернизации путем последовательной разработки, внедрения или замены отдельных модулей.

Этап 3. Физическая реализация и алгоритм функционирования

На рисунке 8 показана общая физическая структура программного комплекса на уровне файловой системы, которая является своего рода обобщением разработанной архитектуры.

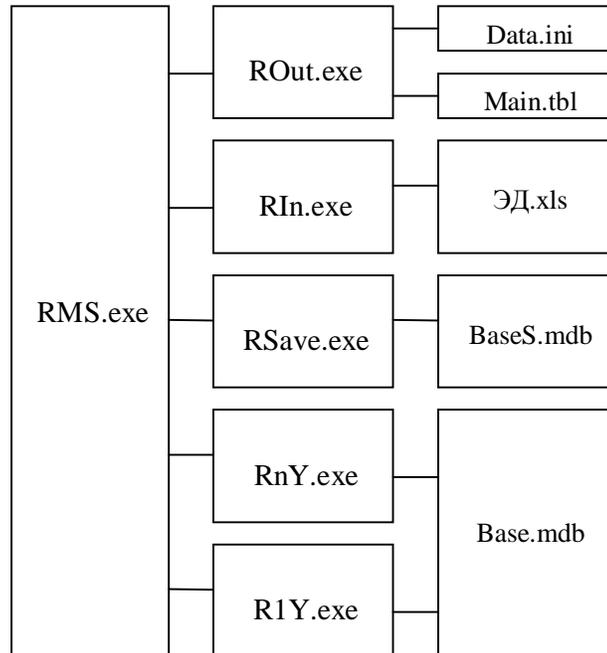


Рисунок 8 – Общая физическая структура программного комплекса на уровне файлов

Расшифровка обозначений рисунка 8:

1) RMS.exe – главный запускающий файл программного комплекса, посредством которого осуществляется доступ к уже подключенным модулям, подключение новых или удаление старых модулей.

2) Rout.exe – запускающий файл модуля оценки интегрального показателя воздействия внешней среды. Необходимые для работы этой программы данные хранятся в служебных файлах data.ini и main.tbl.

3) RIn.exe – запускающий файл модуля оценки эффективности агропромышленной интегрированной производственной системы с учетом рисков составляющей на этапе создания материального потока. Данные, необходимые для работы этой программы, хранятся в служебном файле ЭД.xls.

4) RSave.exe – запускающий файл модуля выбора оптимальной стратегии в управлении хранилищем сельскохозяйственной продукции. Данные хранятся в файле bases.mdb.

5) RnY.exe и R1Y.exe – запускающие файлы модулей оценки рисков в растениеводстве для многолетних и однолетних культур, соответственно. Данные хранятся в файле base.mdb.

Алгоритм функционирования программного обеспечения комплекса можно рассмотреть на примере модуля для оценки интегрального показателя воздействия внешней среды (рисунок 9).

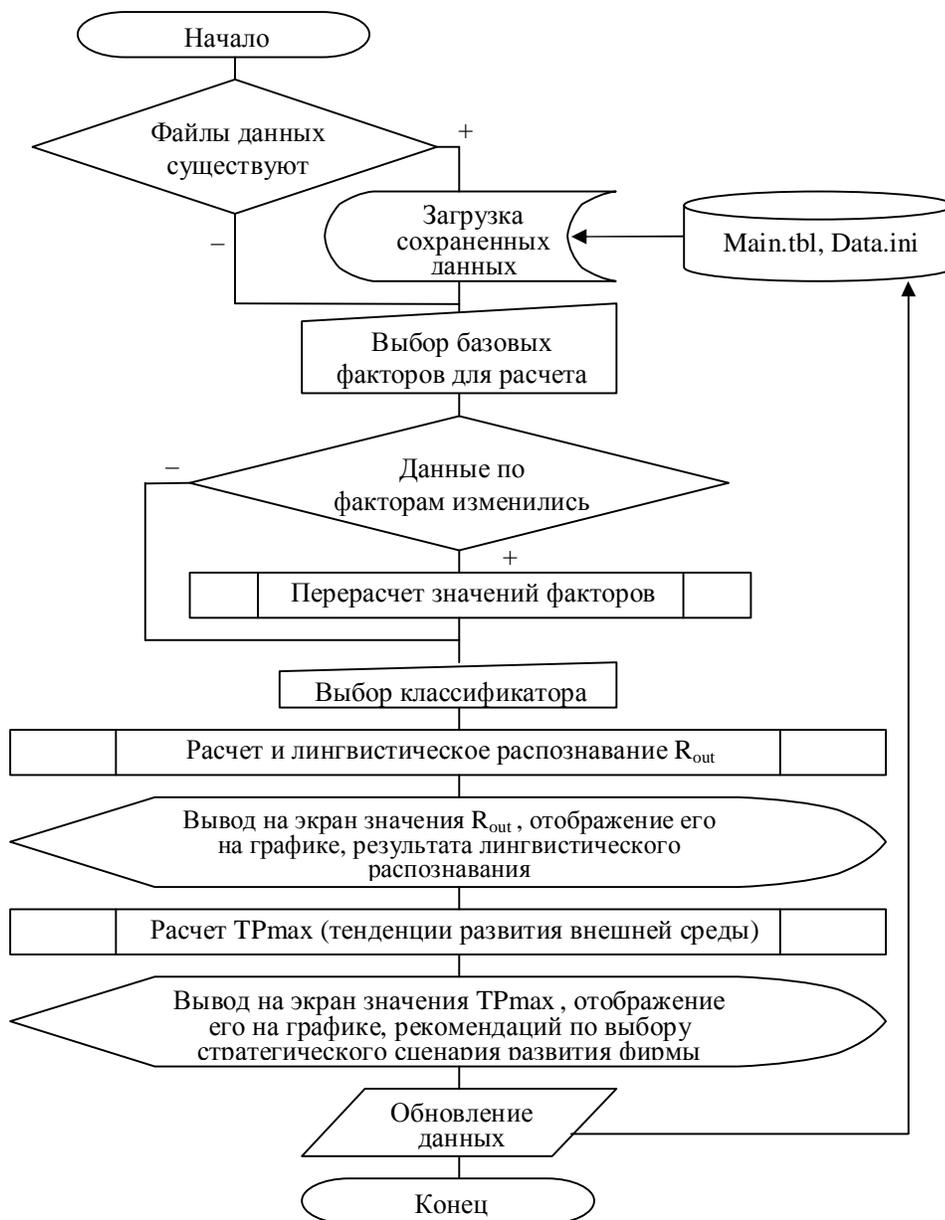


Рисунок 9 – Алгоритм функционирования модуля для оценки интегрального показателя воздействия внешней среды

Программный комплекс построен по модульному принципу: в его состав входят шесть программ-модулей. **Интерфейс пользователя** рассмотрим на примере головной программы и модуля оценки показателя воздействия внешней среды. На рисунке 10 показано главное окно программы, которое появляется на экране после запуска исполняемого файла RMS.EXE. Для запуска любого модуля необходимо дважды щелкнуть по соответствующей строке.

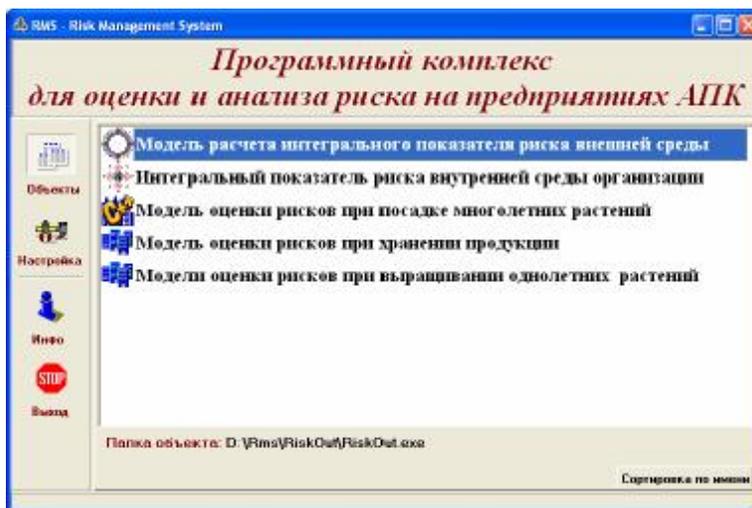


Рисунок 10 – Главное окно программного комплекса

В окне «Настройка» (рисунок 11) можно выбрать модули, которые будут отображаться в главном окне.

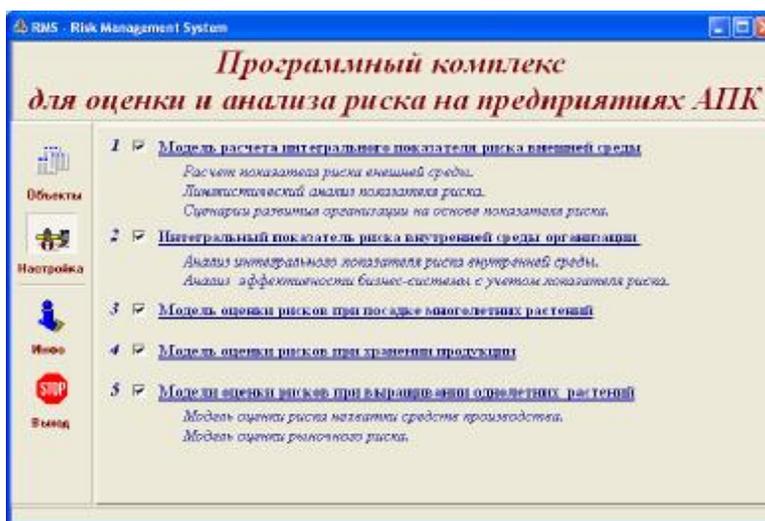


Рисунок 11 – Окно «Настройка»

На рисунке 12 показано основное окно модуля для оценки интегрального показателя воздействия внешней среды, которое содержит три вкладки: «Факторы» (активная вкладка на рисунке 12), «Расчет показателя риска внешней среды - Rout» и «Настройки».

Вкладка «Факторы» предназначена для заполнения исходных данных для расчета интегрального показателя риска.

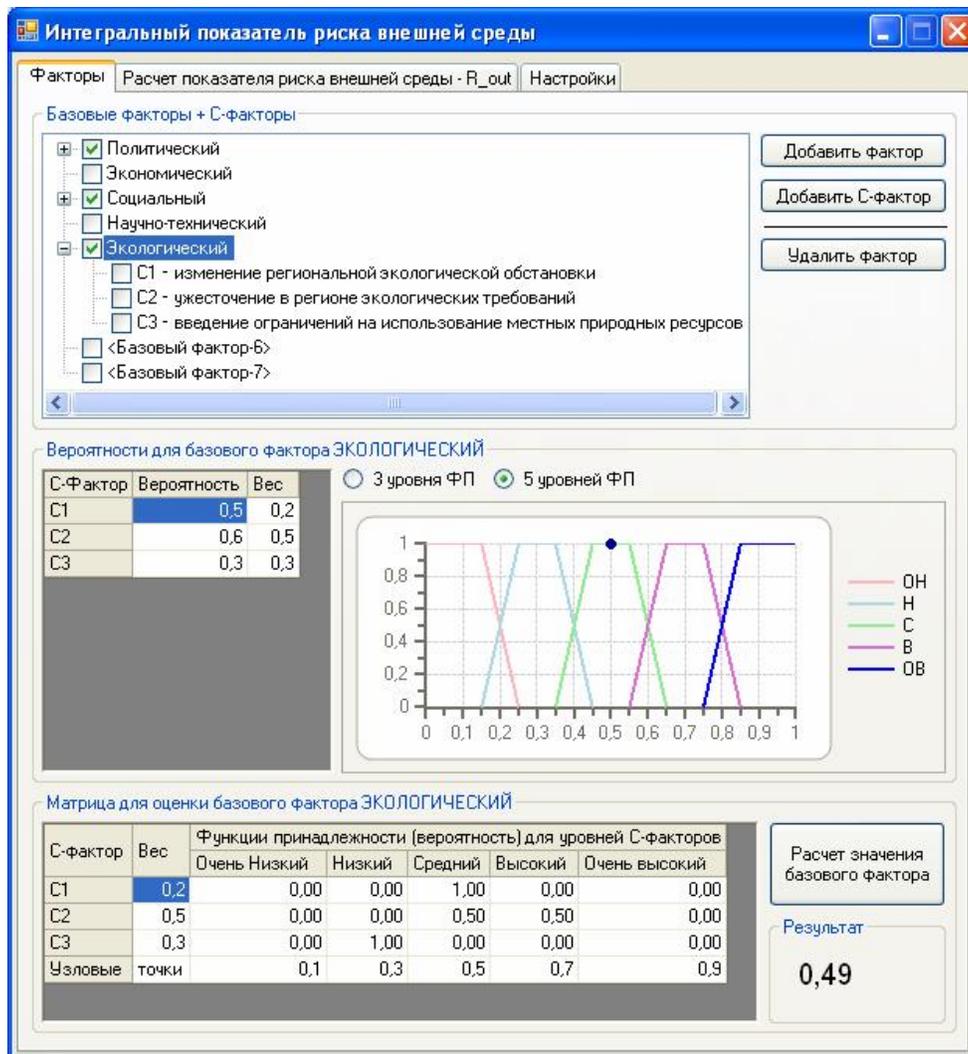


Рисунок 12 – Главное окно программы с активной вкладкой «Факторы»

Вкладка «Расчет показателя риска внешней среды - Rout» (рисунок 13) предназначена непосредственно для расчета показателя и лингвистического распознавания полученного значения с использованием трехуровневого или пятиуровневого 01-классификатора.

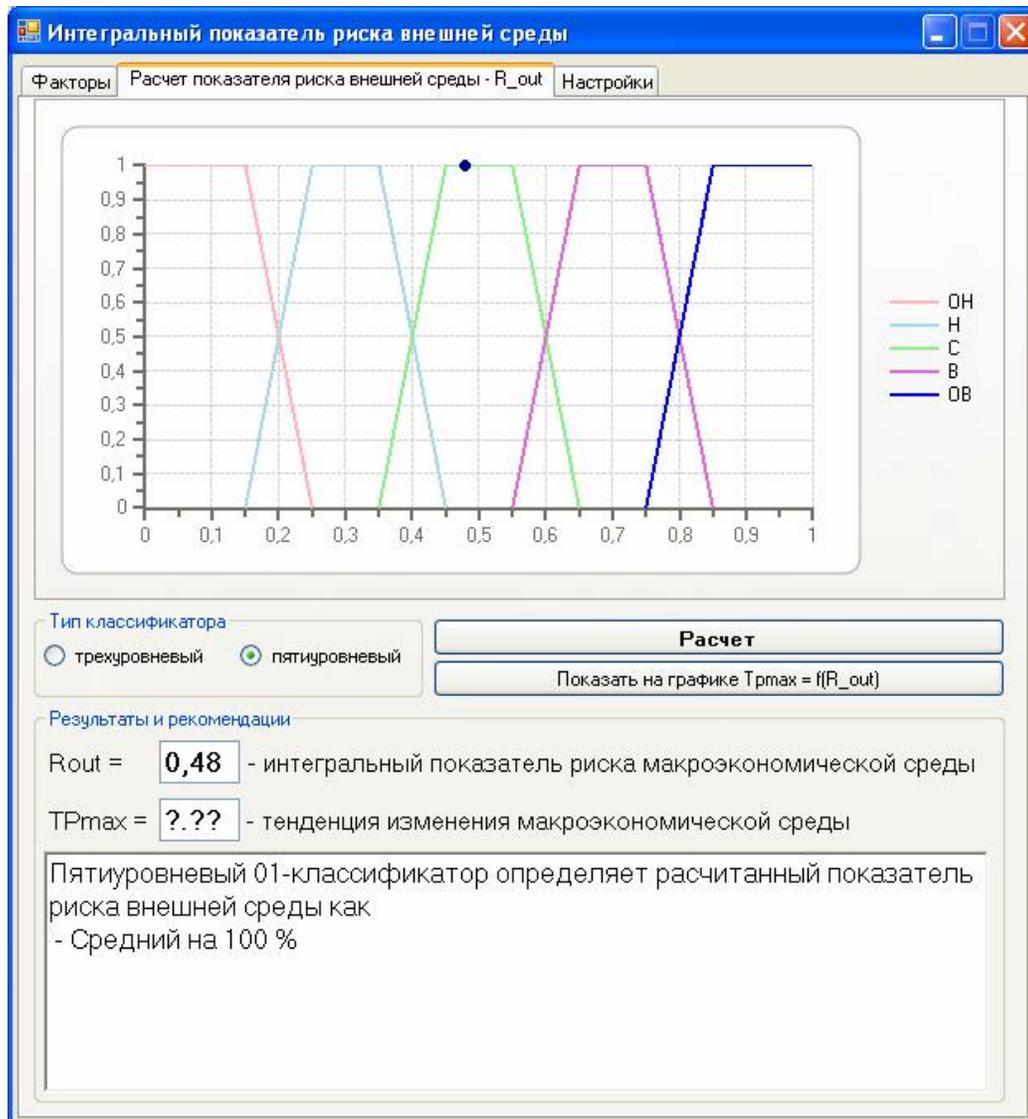


Рисунок 13 – Вкладка «Расчет показателя риска внешней среды - R_{out}»

Здесь же можно провести анализ показателя риска с позиции выбора сценария развития организации в зависимости от тенденции развития внешней среды. Для этого необходимо нажать и удерживать кнопку «Показать на графике $T_{pmax} = f(R_{out})$ ». После нажатия на эту кнопку окно будет выглядеть, как показано на рисунке 14. В окне результатов будут даны соответствующие рекомендации.

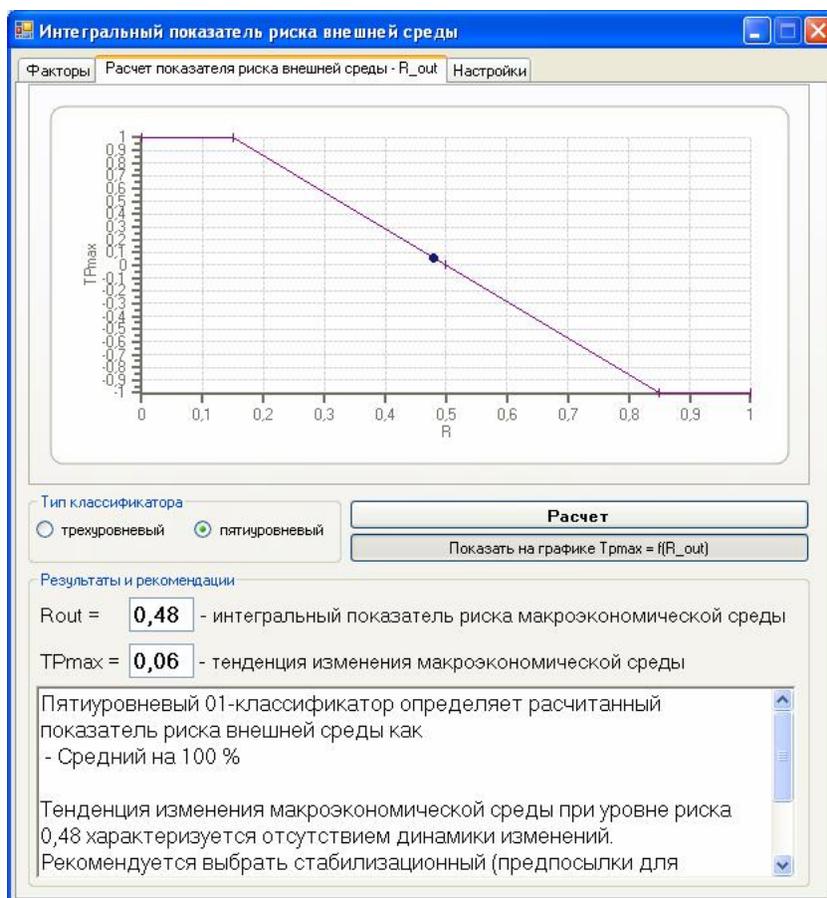


Рисунок 14 – Вкладка «Расчет показателя внешней среды» после нажатия и удержания кнопки «Показать на графике $Tr_{max} = f(R_{out})$ »

Вкладка «Настройки» (рисунок 15) предназначена для определения метода задания весов для базовых и составляющих факторов. Существует три варианта: равновероятный, вручную и методом Фишберна. По умолчанию для базовых факторов принят равновероятный вариант, для составляющих факторов подразумевается ручной ввод на странице «Факторы» в таблице «Вероятности для базового фактора». Существующий порядок можно изменить как для базовых, так и для составляющих факторов. Например, выбрав «метод Фишберна», необходимо проставить ранги факторов (1, 2, 3 и т.д. по количеству факторов – одинаковый ранг у разных факторов не допустим), затем нажать соответствующие кнопки, чтобы изменения были приняты.

Возможно применение различных методов у базовых и составляющих факторов независимо друг от друга.

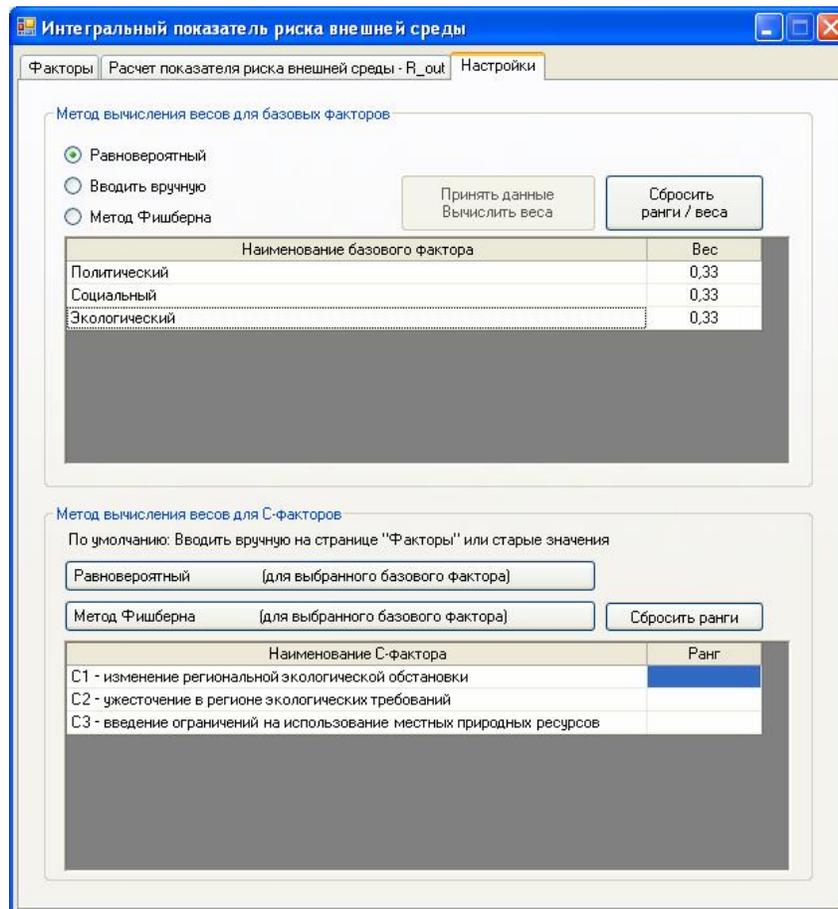


Рисунок 15 – Вкладка «Настройки»

Список литературы

1. Ефанова Н.В. Выявление и оценка рисков при выращивании сельскохозяйственных культур // Информационная среда вуза: Материалы X Междунар. науч.-техн. конф./Иван. гос. архит.-строит. акад. – Иваново, 2003. – С. 440–443.
2. Лойко, В.И. Методика оценки риска потери прибыли при хранении сельскохозяйственной продукции / В.И. Лойко, Н.В. Ефанова // Научный журнал КубГАУ [Электронный ресурс]. – Краснодар: КубГАУ, 2004. – №02(4). – Режим доступа: <http://ej.kubagro.ru/2004/02/19/p19.asp>.
3. Лойко, В.И. Методика оценки риска при посадке многолетних растений / В.И. Лойко, Н.В. Ефанова // Научный журнал КубГАУ [Электронный ресурс]. – Краснодар: КубГАУ, 2005. – №06(14). – Режим доступа: <http://ej.kubagro.ru/2005/06/09/p09.asp>.
4. Лойко, В.И. Подход к оценке интегрального показателя риска интегрированных производственных систем / В.И. Лойко, Н.В. Ефанова // Научный журнал КубГАУ [Электронный ресурс]. – Краснодар: КубГАУ, 2005. – №03(11). – Режим доступа: <http://ej.kubagro.ru/2005/03/18/p18.asp>.