

Министерство образования Республики Беларусь
Учреждение образования
Белорусский государственный университет
информатики и радиоэлектроники
Кафедра инженерной психологии и эргономики

На правах рукописи
УДК 004.418 :378

Дубовский
Максим Александрович

ПРОГРАММНОЕ СРЕДСТВО МОНИТОРИНГА БАНКОВСКИХ
УСТРОЙСТВ САМООБСЛУЖИВАНИЯ: ИНЖЕНЕРНО-
ПСИХОЛОГИЧЕСКИЙ ПОДХОД К РАЗВИТИЮ СИСТЕМЫ

Автореферат диссертации на соискание академической степени
магистра технических наук

1-23 80 08 – Психология труда, инженерная психология, эргономика

Магистрант М.А. Дубовский

Научный руководитель
Д.А. Степаненко, кандидат
технических наук, доцент

Заведующий кафедрой ИПиЭ
К.Д. Яшин, кандидат
технических наук, доцент

Минск 2019

КРАТКОЕ ВВЕДЕНИЕ

В диссертационной работе изложены результаты исследований разработок, выполненных магистрантом лично или в которых магистрант принимал участие в обработке результатов, их обобщении и трактовке, написании статей. Участие соавторов заключалось как в проведении совместных исследований, так и в обсуждении полученных результатов.

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Цель и задачи исследования. Цель исследования – разработать программное средство мониторинга банковских устройств самообслуживания: инженерно психологический подход к развитию системы

Для достижения поставленной цели работа проводилась в несколько этапов, на каждом из которых решались следующие задачи:

1. Аналитический обзор аналогов и технологий разработки программного средства мониторинга банковских устройств самообслуживания

2. Инженерно психологическое проектирование программного средства мониторинга банковских устройств самообслуживания

3. Программная реализация, тестирование и внедрение Программного средства мониторинга банковских устройств самообслуживания.

Объект и предмет исследования. Объектом исследования является разрабатываемое программное средство мониторинга банковских устройств самообслуживания инженерно психологический подход к развитию системы. **Предмет исследования:** инструменты инженерной психологии проектирования, разработка и тестирование программного средства.

Научная новизна и практическая значимость полученных результатов.

1. Продукт является узкоспециализированным, в связи с этим были изучены потребности пользователей и под эти потребности разработано программное средство

2. Программное средство внедрено и работает в ООО «БайТехСолюшн».

3. Разработанное программное средство позволяет руководству компании следить за работой устройств самообслуживания, а отделу мониторинга получать информацию по устройствам самообслуживания в реальном времени и управлять ими.

Апробация результатов диссертации. Опубликованность результатов. Материалы диссертационной работы докладывались и обсуждались на следующих конференциях:

1. Международная научно-практическая конференция «Евразийский союз ученых», 25-31 мая 2019 г

2. 55-я научно-техническая конференция аспирантов, магистрантов и студентов БГУИР г.Минск, БГУИР, 25-30 апреля 2019 г

Опубликованы статьи в материалах 1-й и 2-й конференций [1,2]

Личный вклад магистранта. В диссертационной работе изложены результаты исследований разработок, выполненных магистрантом лично или в которых магистрант принимал участие в обработке результатов, их обобщении и трактовке, написании статей. Участие соавторов заключалось как в проведении совместных исследований, так и в обсуждении полученных результатов.

Структура и объем диссертационной работы. Диссертация состоит из общей характеристики работы, введения, трех глав и заключения. Полный объем диссертации – 67 страницы, из них 11 страниц, занимаемых иллюстрациями. Диссертационная работа содержит 3 таблицы на 5 страницах, ссылки на 51 литературный источник на 4 страницах.

КРАТКОЕ СОДЕРЖАНИЕ РАБОТЫ

ГЛАВА 1 ТЕХНОЛОГИИ РАЗРАБОТКИ ПРОГРАММНЫХ СРЕДСТВ МОНИТОРИНГА БАНКОВСКИХ УСТРОЙСТВ САМООБСЛУЖИВАНИЯ

Область ИТ технологий является инновационным развивающимся пространством, которое не ограничивается территориальными рамками и приобретает все больший глобальный масштаб. Она активно предоставляет решения для задач как обывателей, так и целых направлений отраслей, являясь одним из важнейших факторов, которые оказывают сильное влияние реальность.

На рынке программных средств существуют готовые разработки для решения поставленной нами задачи автоматизации учета товарно-денежного оборота. Наиболее популярными являются:

комплекс организации «Системные-Технологии» под названием «ST-Cash» версии 1.0 и комплекс организации «Soft-Club» под названием «SC-Cash» версии 4.07. Данные системы обладают всеми необходимыми функциями для успешной работы сотрудника банка, но все же с учётом современного развития программного и аппаратного обеспечения информационных технологий можно выделить и недостатки наличие которых может влиять на производительность работы банковской системы

Основными недостатками являются:

- использование устаревших технологий СУБД;
- довольно жёсткая поддержка операционных систем;
- закрытая база данных от администраторов и персонала сопровождающего данную систему, что усложняет восстановление системы при сбое;
- сложный для восприятия и освоения интерфейс (в частности «ST-Cash»).

Вышеперечисленные недостатки позволяют определить направления совершенствования современных программных средств.

Таким образом, разработка мониторинга устройств самообслуживания, в которой будут реализованы основные направления совершенствования этого класса программных средств, является актуальной задачей.

Выводы

Разрабатываемое средство должно функционировать в среде операционной системы Windows, работать в среде терминальных приложений. Программный продукт необходимо снабдить отказоустойчивой системой, защищенной от несанкционированного доступа, интуитивно-понятным интерфейсом и интерактивным руководством, помогающим пользователю разобраться с особенностями работы с программой.

Автоматизированная система будет внедрена вместе с системой терминальных приложений: пользователь на рабочем месте имеет не полноценный персональный компьютер, а только терминальный клиент (тонкий клиент), который посредством комплекса программно-аппаратных средств подключается к терминальному серверу, где и происходит запуск программного продукта. Данный способ позволяет улучшить защиту базы данных и самой программы от несанкционированного доступа, упростить администрирование рабочего процесса, ускорить работу с базой данных, так как локальная сеть (ЛВС) не будет загружена множеством запросов-ответов при работе с базой данных, а только лишь передавать данные-пакеты протокола RDP (протокол удаленного рабочего стола). Так же увеличивается отказоустойчивость системы в целом: сведен к минимуму вариант сбоя автоматизированной системы при неисправности рабочей станции или ЛВС, в случае с терминальным приложением, происходит только отсоединение текущего сеанса, без некорректного выхода из приложения. Так же уменьшаются затраты на электроснабжение рабочих станций: тонкие клиенты потребляют значительно меньше электроэнергии, и уменьшаются затраты на формирование новых рабочих мест: стоимость терминального клиента ниже и он не требует установки мощных источников бесперебойного питания.

ГЛАВА 2. ИНЖЕНЕРНО – ПСИХОЛОГИЧЕСКОЕ ИССЛЕДОВАНИЕ ПРОГРАММНОГО СРЕДСТВА БАНКОВСКИХ УСТРОЙСТВ САМООБСЛУЖИВАНИЯ

Автоматизированные системы, используемые в процессе учета данных экономических операций, обеспечивают передачу и быструю обработку больших объемов данных. При этом значительные объемы информации доступны в оперативном режиме для анализа, прогнозирования и контроля. Это предъявляет жесткие требования к производительности операционной системы, системы управления базами данных и средств передачи данных. Поэтому средства системы должны быть в состоянии поддерживать доступ к большим (и постоянно возрастающим) объемам данных без потери производительности.

На современном этапе развития автоматизированных систем все большее распространение получает распределенная обработка информации. Этому способствует бурное развитие компьютерной техники, снижение ее стоимости, простота в обслуживании и эксплуатации.

Структурно такие автоматизированные системы реализуются как некоторая сеть (вычислительная система), объединяющая посредством каналов передачи данных персональных электронно-вычислительных машин терминалы, другие периферийные устройства.

Функциональное моделирование является важнейшим элементом концептуального анализа, который выполняется на начальном этапе проектирования любого ПС. Разработка и анализ функциональной модели позволяет достаточно глубоко погрузиться в предметную область, выявить функции, которые должно выполнять ПС и определить связи и взаимодействия между ними.

Физическое проектирование является последним этапом создания проекта базы данных, при выполнении которого проектировщик принимает решения о способах реализации разрабатываемой базы данных. Во время предыдущего этапа проектирования была определена логическая структура базы данных (которая описывает отношения и ограничения в рассматриваемой прикладной области). Хотя эта структура не зависит от конкретной целевой СУБД, она создается с учетом выбранной модели хранения данных, например, реляционной, сетевой или иерархической. Однако, приступая к физическому проектированию базы данных, прежде всего необходимо выбрать конкретную целевую СУБД. Поэтому физическое проектирование неразрывно связано с конкретной СУБД. Между логическим и физическим проектированием существует постоянная обратная связь, так как решения, принимаемые на этапе физического проектирования с целью повышения производительности системы, способны повлиять на структуру

логической модели данных.

Этапы концептуального и логического проектирования больших систем следует отделять от этапов физического проектирования. На это есть несколько причин.

- они связаны с совершенно разными аспектами системы, поскольку отвечают на вопрос, что делать, а не как делать;
- они выполняются в разное время, поскольку понять, что надо сделать, следует прежде, чем решить, как это сделать;
- они требуют совершенно разных навыков и опыта, поэтому требуют привлечения специалистов различного профиля.

Этапы физического проектирования баз данных:

1. Перенос глобальной логической модели данных в среду целевой СУБД.
2. Проектирование основных отношений;
3. Разработка способов получения производных данных;
4. Реализация ограничений предметной области;
5. Проектирование физического представления базы данных;
6. Анализ транзакций;
7. Выбор файловой структуры;
8. Определение индексов;
9. Определение требований к дисковой памяти;
10. Проектирование пользовательских представлений;
11. Разработка механизмов защиты;
12. Обоснование необходимости введения контролируемой избыточности;
13. Текущий контроль и настройка операционной системы.

Проектирование базы данных — это итерационный процесс, который имеет свое начало, но не имеет конца и состоит из бесконечного ряда уточнений. Его следует рассматривать прежде всего как процесс познания. Как только проектировщик приходит к пониманию работы предприятия и смысла обрабатываемых данных, а также выражает это понимание средствами выбранной модели данных, приобретенные знания могут показать, что требуется уточнение и в других частях проекта. Особо важную роль в общем процессе успешного создания системы играет концептуальное и логическое проектирование базы данных.

Система помощи по использованию программы на сегодняшний день является неотъемлемой частью любой программы. Справочная информация предоставляет пользователю возможность в любой момент времени получить полную информацию о назначении и использовании того или иного компонента программы. В данной программе также необходима справочная

система, которая позволит пользователю получить помощь при возникновении затруднений, связанных с использованием программы.

На этапе планирования необходимо составить перечень разделов справочной службы и нужных перекрестных ссылок. Структура разделов и количество перекрестных ссылок зависит от сложности программы, для которой создается Help-служба. Необходимо учесть следующие рекомендации:

а) любой раздел может содержать список подчиненных или связанных разделов, т. е. таким образом, создается произвольная иерархия справочной службы;

б) необходимо структурировать разделы по роду предоставляемой ими информации;

в) каждый раздел должен отображаться в отдельном окне;

г) монотонный текст (как и чрезмерно "раскрашенный") резко увеличивает утомляемость пользователя и затрудняет усвоение информации. Поэтому справка должна содержать структурированный текст, большое количество рисунков;

д) для того, чтобы у пользователя возникало меньше проблем при работе с данной программой необходимо использовать подсказки, которые бы автоматически появлялись удержании курсора мыши над элементом управления.

Система помощи пользователю должна содержать следующие основные разделы:

- работа в автоматизированной системе (перечисление функций и работа с ними);

- работа с редактором базы данных (инструкции по способам добавления, удаления и редактирования записей базы данных);

- права доступа (инструкции по созданию новых пользователей и правах доступа);

- работа с отчетами (помощь в работе с отчетами).

Доступ к отдельным разделам должен осуществляться при нажатии специальной кнопки помощи, расположенной в окне приложения. Также можно получить доступ ко всем разделам справки, нажав на кнопку «Help» в главном меню.

Защита информации от взлома должна обязательно присутствовать для того, чтобы предотвратить несанкционированный доступ к информации, хранящейся в базе данных автоматизированной системы.

Во-первых, авторизация пользователя происходит на уровне аутентификации при подключении к серверу терминалов. После включения персонального компьютера, либо тонкого клиента, пользователю необходимо ввести имя пользователя и пароль для доступа к профилю учетной записи на

сервере.

Защита данных предусмотрена так же и программой. Для этого база данных должна быть защищена паролем. Для того, чтобы ограничить редактирование базы данных, необходимо разработать систему авторизации, которая позволила бы разграничить права доступа к определенным компонентам программы. Для защиты паролей от взлома необходимо разместить все пароли в защищенной паролем базе данных – это позволит реализовать принцип многопользовательского доступа, а также легко и быстро добавлять/удалять пользователей.

Таким образом, доступ к системе смогут осуществить только определенные пользователи – неавторизованный пользователь не сможет получить доступа к программному средству, а даже если и получит доступ, то не сможет считать информацию в базе данных.

Также все действия пользователей заносятся в специальный log-файл, который впоследствии может быть использован администратором системы для выяснения неисправности системы при сбое.

Одной из важнейших задач при разработке информационной системы является проектирование взаимодействия человека с ней. Специалисты-системотехники владеют навыками проектирования пользовательских интерфейсов, умеют проводить исследование юзабилити и эргономическую экспертизу информационных систем различной сложности и назначения. Разработчик учитывает, что интерфейс пользователя информационной системы должен удовлетворять ряду критериев: минимальное время выполнения задачи пользователем; минимальное число произвольных ошибок пользователя; минимальная неоднозначность в понимании интерфейса (способствует самообучению пользователей и делает их поведение предсказуемым); высокая стандартизация интерфейса (облегчает обучение пользователей); объем вводимой пользователем информации должен стремиться к минимуму; простота и визуальная привлекательность. Поэтому проектирование конкретного вида деятельности пользователя предусматривает: исследование внутренних средств деятельности человека (его опыта, знаний, навыков, восприятия, мышления, памяти); согласование этих средств с внешними средствами (документами, алгоритмами, инструментами, органами ручного управления) в соответствии с основной целью функционирования создаваемой информационной системы; формирование на основе разработанного проекта требований к техническим средствам системы, которые используются человеком; реализацию требований при воплощении продукта.

Взаимодействие пользователей с прикладным программным обеспечением, входящим в состав системы должно осуществляться посредством визуального графического интерфейса (GUI). Интерфейс

системы должен быть понятным и удобным, не должен быть перегружен графическими элементами и должен обеспечивать быстрое отображение экранных форм. Навигационные элементы должны быть выполнены в удобной для пользователя форме. Средства редактирования информации должны удовлетворять принятым соглашениям в части использования функциональных клавиш, режимов работы, поиска, использования оконной системы. Ввод-вывод данных системы, прием управляющих команд и отображение результатов их исполнения должны выполняться в интерактивном режиме. Интерфейс должен соответствовать современным эргономическим требованиям и обеспечивать удобный доступ к основным функциям и операциям системы.

Интерфейс должен быть рассчитан на преимущественное использование манипулятора типа «мышь», то есть управление системой должно осуществляться с помощью набора экранных меню, кнопок, значков и т. п. элементов. Клавиатурный режим ввода используется при заполнении и/или редактировании текстовых и числовых полей экранных форм.

Все надписи экранных форм, а также сообщения, выдаваемые пользователю (кроме системных сообщений) должны быть на русском языке.

Система должна обеспечивать корректную обработку аварийных ситуаций, вызванных неверными действиями пользователей, неверным форматом или недопустимыми значениями входных данных. В указанных случаях система должна выдавать пользователю соответствующие сообщения, после чего возвращаться в рабочее состояние, предшествовавшее неверной (недопустимой) команде или некорректному вводу данных.

Логическое проектирование – создание схемы базы данных на основе конкретной реляционной модели данных. Для реляционной модели данных дата логическая модель – набор схем отношений, обычно с указанием первичных ключей, а также «связей» между отношениями, представляющих собой внешние ключи.

После нормализации отношений следует построение логической модели БД. Логическая модель является основой базы данных, она должна отображать взаимосвязи между реляционными таблицами.

Ключевое поле – это одно или несколько полей, комбинация значений которых однозначно определяет каждую запись в таблице. Если для таблицы определены ключевые поля.

Ключевые поля используются для быстрого поиска и связи данных из разных таблиц при помощи запросов, форм и отчетов.

В каждой таблице должно быть обязательно ключевое поле, которое будет использовать ключевое поле типа счетчик.

Связи между таблицами в базе данных используются при формировании запросов, разработке форм, при создании отчетов.

Выводы

При проектировании и разработке подсистем должны использоваться следующие базовые принципы:

- исключение дублирования ввода информации и повышение ее достоверности, за счет отождествления ранее введенной информации;
- расширяемость/масштабируемость - обеспечение возможности добавления новых функций или изменения некоторых уже имеющихся при неизменных остальных функциональных частях;
- иерархичность, реализуемая формированием многоуровневой структуры системы с учетом делегирования полномочий на соответствующий уровень управления;
- стандартизуемость - проектируются и разрабатываются на основе согласованных международных стандартов и предложений, реализация открытости осуществляется на базе функциональных стандартов (профилей) в области информационных технологий.

Система должна удовлетворять следующим требованиям:

- пользовательский интерфейс системы должен быть сформирован в соответствии с навыками и профилем пользователей;
- работоспособность - система должна быть доступна 99,9% времени;
- восстанавливаемость - после восстановления системы из состояния неработоспособности обработка данных должна производиться в том же режиме, что и до сбоя.

Система должна содержать защиту от ввода неверных данных.

ГЛАВА 3. ТЕСТИРОВАНИЕ И РУКОВОДСТВО ПО ЭКСПЛУАТАЦИИ ПРОГРАММНОГО СРЕДСТВА

Целью процесса разработки является реализация потребности заказчика, путем выполнения комплекса взаимосвязанных мероприятий в условиях временных и ресурсных ограничений, результатом которых является разработанное программное средство, удовлетворяющее требованиям заказчика. Для того чтобы удостовериться, что все требования соблюдены и работоспособны – проводится тестирование программного средства. Ведь именно на данном этапе можно отследить некорректность и неточность работы программного средства.

Таблица 1 – Характеристики аппаратных платформ

№	Роль	Аппаратная конфигурация	Программная конфигурация
1	Сервер	Intel® Xeon® E5-2650 (2.0GHz/8-core/20MB/8GT-s QPI/95W, DDR3-1600, Turbo2) Processor ОЗУ: 64 ГБ	Windows Server 2008 R2 SP1 64-bit
2	Рабочая станция	Intel® Core(TM) i5-7400 CPU 3.00 GHZ, ОЗУ: 16,00 ГБ	Windows 10 Pro 64-bit

При проведении тестирования были обнаружены некоторые ошибки в работе программного средства, которые были быстро устранены. После завершения успешного тестирования всех основных функций ПС, можно подвести итог о его исправном функционировании и выполнении основных требований, поставленных к разрабатываему ПО.

Результатом описанных выше этапов тестирования явились ошибки. Каждой ошибке присваивался свой уровень важности, в зависимости от влияния на работу приложения. Ошибки с более высоким приоритетом исправлялись в первую очередь.

Программа предназначена для автоматизации сбора, хранения и обработки информации об указанных операциях с целью:

- повышения эффективности работы;
- получения необходимой отчетности;
- исключения ошибок ввода информации;
- учета, дополнительного контроля и анализа.

Программа может функционировать как отдельный самостоятельный продукт, при этом для работы понадобится персональный компьютер, удовлетворяющий следующим минимальным требованиям:

- процессор с тактовой частотой не менее 1000 Гц;
- оперативная память RAM 256 Мбайта;

– объём внешней памяти 25 Мбайт (без учёта заполнения базы данных);

– свободное место на винчестере: 15Мб;

– мышь, клавиатура;

– монитор с поддержкой качества цветопередачи в 16 тыс. цветов.

Для работы с программой запустите `Monitoring.cmd`. После успешного запуска появляется консоль в которой будут отображены все подгружаемые модули, а также отображены доступные роли.

ЗАКЛЮЧЕНИЕ

Разработано программное средство для функционирования в среде операционной системы Windows, работает в среде терминальных приложений. Программный продукт снабжен отказоустойчивой системой, защищенной от несанкционированного доступа, интуитивно-понятным интерфейсом и интерактивным руководством, помогающим пользователю разобраться с особенностями работы с программой.

Автоматизированная система внедрена вместе с системой терминальных приложений: пользователь на рабочем месте имеет не полноценный персональный компьютер, а только терминальный клиент (тонкий клиент), который посредством комплекса программно-аппаратных средств подключается к терминальному серверу, где и происходит запуск программного продукта. Разработанное ПС уменьшает затраты на электроснабжение рабочих станций: тонкие клиенты потребляют значительно меньше электроэнергии, и уменьшаются затраты на формирование новых рабочих мест: стоимость терминального клиента ниже и он не требует установки мощных источников бесперебойного питания.

СПИСОК ОПУБЛИКОВАННЫХ РАБОТ

Статьи в рецензируемых научных журналах

1. Dubouski M.A. Software monitoring system of banking devices of self-service: engineering-psychological approach to development of the system / M.A. Dubouski // Proceedings of the Second International Conference of European Academy of Science, Bonn, May 25 - 31 2019. - Bonn, 2019. - P. 19 - 20.

Статьи в сборниках материалов научных конференций

2. Дубовкий М.А. Программное средство мониторинга банковских устройств самообслуживания. / М.А. Дубовский, О. М. Бакунова, // Материалы 55-я научно-техническая конференция аспирантов, магистрантов и студентов БГУИР г.Минск, БГУИР, 25-30 апреля 2019 г