|  |
| --- |
| Институт информационных технологий |
| Наименование института (факультета) |
|  |
| Кафедра математического и программного обеспечения ЭВМ |
| Наименование кафедры |

**ОТЧЕТ**

|  |  |  |
| --- | --- | --- |
| по | производственной (преддипломной) | практике |

|  |  |
| --- | --- |
| Листов |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Студента | Белова Антона Сергеевича | группы | 1ИСб-00-41оп |
|  | Ф.И.О. |  |  |

|  |
| --- |
| Место прохождения практики |
| АО «Сбербанк – Технологии» |
| в г. Череповце |

|  |  |
| --- | --- |
| Студент |  |
|  | (подпись) |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Руководитель практики от предприятия |  | старший руководитель направления по развитию IT-систем |  | Григоров Антон Сергеевич |
|  | (должность) |  | (ФИО, подпись) |
| Руководитель практики от предприятия |  | Доцент, к.т.н. |  | Варфоломеев Игорь Андреевич |
|  | (должность) |  | (ФИО, подпись) |

2018 год

Оглавление

[Введение 3](#_Toc516122723)

[Раздел 1. Общая характеристика базы практики и организация информационно-производственной деятельности на предприятии 6](#_Toc516122724)

[Раздел 2. Систематизация и обобщение научной и производственно-технологической информации по теме исследований 8](#_Toc516122725)

[Раздел 3. Разработка технического задания по теме ВКР (оформляется в виде приложения к отчету) 9](#_Toc516122726)

[Раздел 4. Сравнительный анализ возможных вариантов реализации научно- технической информации по теме исследования 10](#_Toc516122727)

[Раздел 5. Реализация некоторых из возможных путей решения поставленной в техническом задании на ВКР задачи 14](#_Toc516122728)

[Раздел 6. Технико-экономическое обоснование выполняемой разработки 52](#_Toc516122736)

[Заключение 61](#_Toc516122737)

[Приложение 1. Техническое задание 62](#_Toc516122738)

[Приложение 2. Текст программы 77](#_Toc516122739)

[Приложение 3. Тестирование 88](#_Toc516122740)

Введение

Производственная практика — практическая часть учебного процесса подготовки квалифицированных рабочих и специалистов, проходящая, как правило, на различных предприятиях в условиях реального производства. Является заключительной частью учебной практики, проходящей в учебном заведении. Во время производственной практики происходит закрепление и конкретизация результатов теоретического учебно-практического обучения, приобретение студентами умения и навыков практической работы по присваиваемой квалификации и избранной специальности или профессии.

Целью преддипломной производственной практики является доработка расчетно-пояснительной записки и приложений, а также формирования профессиональных компетенций (ПК), приведенных в табл.1.

Задачами практики является:

* поиск информации об имеющемся на данном предприятии оборудовании, применение полученных знаний на практике;
* разработать логические модели и спецификации на проектируемую информационную систему;
* реализовать отдельные компоненты информационной системы, выполнить их сборку;
* обосновать выбор метода и инструментов разработки программного обеспечения;
* обосновать выбор модели жизненного цикла программного обеспечения
* выполнить моделирование работы отдельных компонентов информационной системы;
* рассчитать технико-экономическое обоснование выполняемой разработки;
* разработать и согласовать проектную документацию;
* поиск, систематизация и обобщение необходимой для выполнения задания литературы и научно-технической информации;
* провести экспериментальную проверку предложенных технических решений;
* проанализировать результаты экспериментальной проверки предложенных технических решений.

Таблица 1

Перечень компетенций

|  |  |
| --- | --- |
| Код | Описание |
| 1 | 2 |
| ПК-1 | Способность проводить предпроектное обследование объекта проектирования, системный анализ предметной области, их взаимосвязей |
| ПК-2 | Способность проводить техническое проектирование |
| ПК-3 | Способность проводить рабочее проектирование |
| ПК-4 | Способность проводить выбор исходных данных для проектирования |
| ПК-5 | Способность проводить моделирование процессов и систем |
| ПК-9 | Способность проводить расчет экономической эффективности |
| ПК-10 | Способность разрабатывать, согласовывать и выпускать все виды проектной документации |
| ПК-22 | Способность проводить сбор, анализ научно-технической информации, отечественного и зарубежного опыта по тематике исследования |
| ПК-23 | Готовность участвовать в постановке и проведении экспериментальных исследований |

Таблица 1. Продолжение

|  |  |
| --- | --- |
| 1 | 2 |
| ПК-24 | Способность обосновывать правильность выбранной модели, сопоставляя результаты экспериментальных данных и полученных решений |

# Раздел 1. Общая характеристика базы практики и организация информационно-производственной деятельности на предприятии

Сбербанк – крупнейший российский универсальный коммерческий банк. Данный банк ведет собственные разработки программного обеспечения (ПО) для стационарных машин, находящихся в отделениях Сбербанка, мобильное ПО («Сбербанк-Онлайн»), а так же браузерная версия Личного кабинета клиента. Созданием этих приложений занимается подразделение Сбербанка – АО Сбербанк-Технологии (АО «Сбертех»).

На 01.07.2017 в численность сотрудников АО «Сбертех» составляет 9800 человек. Головной офис располагается в Москве. Также АО «Сбертех» имеет филиалы в 17 городах России, в том числе и в Череповце, расположенного на Советском проспекте, 8а.

Разработка ПО в АО «Сбертех» ведется по методологии Scrum, а также используется непрерывная интеграция (через Jenkins), подробнее о которых в пункте Организация производства.

Проектирование и кодирование ведется на личных персональных компьютерах и ноутбуках, один из которых подключен к локальной сети офиса АО «Сбертех» в Череповце, а второй к локальной сети Сбербанка, что позволяет программистам проводить тестирование своих программ с реальными банковскими данными.

Разработка ПО в АО «Сбертех» ведется по методологии Scrum, а также используется непрерывная интеграция (через Jenkins).

Scrum (методология) – это универсальная система управления проектами, которая позволяет при минимальном затрачивании ресурсов получать необходимый эффект. Данная технология применяется во время разработки информационных систем управления или во время создания программного обеспечения, также данная методология используется во время разработки крупных игровых проектов.

Для работы по данной методологии используют Scrum-доски, которые выглядят следующим образом:

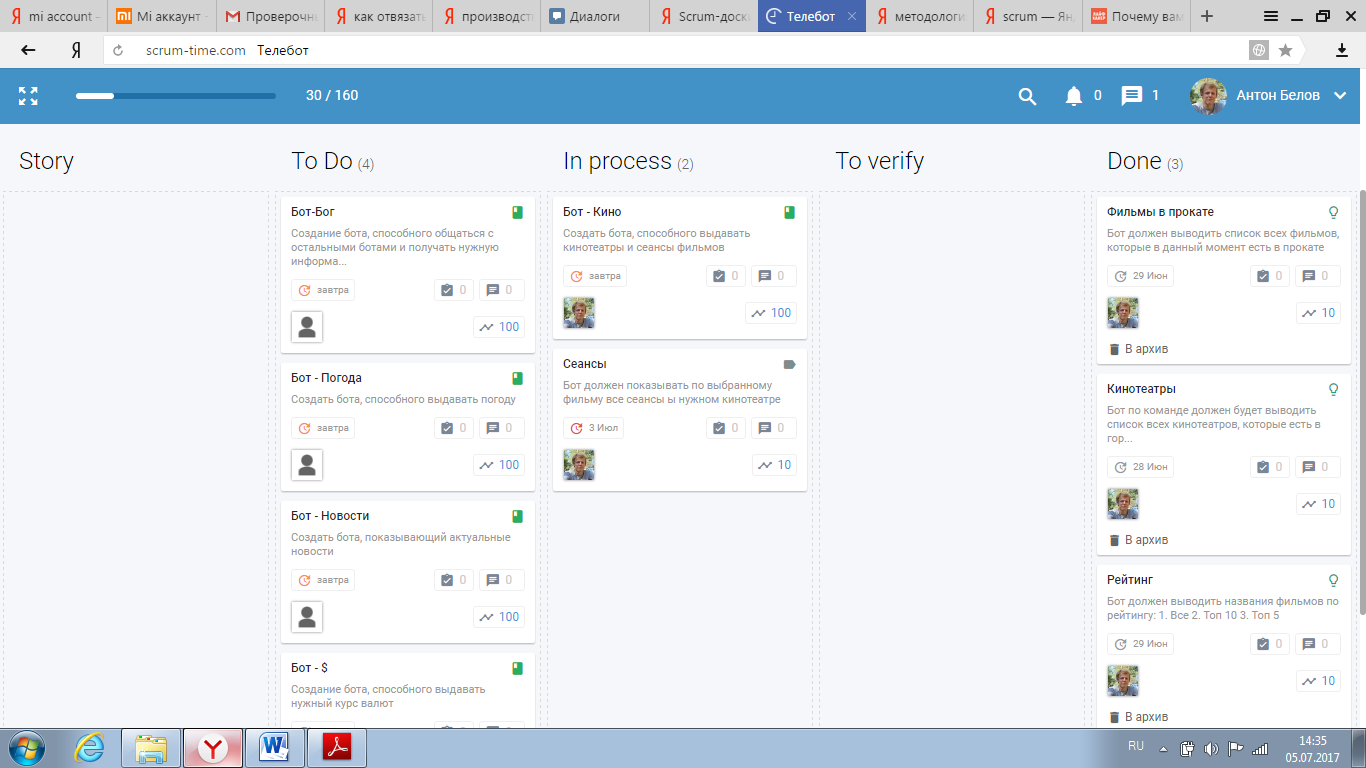


Рис.1. Scrum-доска

Любая Scrum-доска имеет 4 столбца: «что надо делать», «что делается», «что тестируется» и «что выполнено и готово использоваться». Лучше всего использовать онлайн сервисы scrum-досок, так как они позволяют в реальном времени передвигать карточки с заданиями, создавать новые, и вся деятельность наглядна и легко контролируется ответственным по разработке.

Основное направление деятельности череповецкого офиса – это разработка автоматизированных систем для Единой фронтальной системы Сбербанка (ЕФС).

Процесс разработки программного обеспечения построен на гибких методологиях разработки – Agile (Scrum, XP).

* среда разработки – Intelij IDEA, WebStorm;
* система управления версиями программного кода: git;
* центральный репозиторий: BitBucket;
* автоматизация непрерывной интеграции и непрерывной поставки: Jenkins;
* репозиторий артефактов: Nexus.

# Раздел 2. Систематизация и обобщение научной и производственно-технологической информации по теме исследований

Основная задача информационной системы (ИС) HRSaveTime – это фиксирование времени в момент прихода и ухода сотрудников в офис обособленного подразделения АО «СберТех».

Каждый раз, при поднесении пропуска к считывающему устройству, система, считав RFID – индивидуальный ID-номер, хранящийся в пропускной карте, создает временную пару – запись, содержащую «Дата с», «Время с», «Местоположение». При повторном поднесении пропуска к датчику на выходе из помещения, система добавит во временную пару данные: «Дата по», «Время по». В итоге, в базе данных (БД) будет храниться подобная запись, приведённая в табл.2.

Таблица 2

Пример временных пар в БД

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| п\п | RFID | Дата с | Дата по | Время с | Время по | Местоположение |
| 4 | 48RT 96 ER | 03.06.2018 | 03.06.2018 | 8:24 | 10:49 | Главное помещение офиса |
| 5 | 4R 3498 FR | 03.06.2018 | 03.06.2018 | 9:37 | 12:05 | Главное помещение офиса |
| 6 | 48RT 96 ER | 03.06.2018 | 03.06.2018 | 10:49 | 11:30 | Переговорная |

На основе данной информации HR Администратор может сформировать отчёт, в котором будет отражено общее количество времени проведенного в офисе, прогулы, опоздания, отгулы, больничные и командировки.

Все отсутствия заносятся HR Администратором, либо сотрудником, через создание заявки.

Для получения информации о переработках, прогулах, опозданий HR Администратору необходимо в системе заводить графики рабочего времени (ГРВ) по мере необходимости. Каждый график состоит из набора однодневных графиков рабочего времени (ОГРВ), которые присваиваются каждому дню на неделю. Нормативные ГРВ состоят из 5 одинаковых ОГРВ, а ненормативные – из разных. В свою очередь ОГРВ содержит данные о перерыве. По итогу всю информацию о графиках можно представить в виде диаграммы, изображенной на рис.2.



Рис.2. Диаграмма состава ГРВ

* Z40Q1 – Код ГРВ;
* N8RT3, N11RT9 – Код ОГРВ;
* P30N9 – Код перерыва.

Коды нормируются внутренними правилами АО «СберТех».

# Раздел 3. Разработка технического задания по теме ВКР (оформляется в виде приложения к отчету)

Техническое задание (ТЗ) — исходный документ на проектирование технического объекта (изделия). ТЗ устанавливает основное назначение разрабатываемого объекта, его технические характеристики, показатели качества и технико-экономические требования, предписание по выполнению необходимых стадий создания документации (конструкторской, технологической, программной и т. д.) и её состав, а также специальные требования.

В прил.1 «Техническое задание» отражена следующая информация:

* Основания разработки и область применения;
* Назначение разработки;
* Требования к программе;
* Требования к надежности;
* Условия эксплуатации;
* Требования к составу и параметрам технических средств;
* Требования к информационной и программной совместимости;
* Требования к программной документации;
* Стадии и этапы разработки;
* Порядок контроля и приемки.

# Раздел 4. Сравнительный анализ возможных вариантов реализации научно- технической информации по теме исследования

Время – это самый важный ресурс на любом предприятии не только для руководителя, но и для самих сотрудников. На современном рынке представлено не малое количество систем, позволяющих контролировать рабочее время сотрудников.

Все системы можно разделит на две большие категории:

* СКУД, которые фиксируют приход и уход с предприятия или конкретного офиса при помощи датчика и пропускной карты (RFID- карта или брелок), датчика отпечатку пальца, ввода персонального код-пароля и т.д.;
* системы, установленные на персональные компьютеры, производящие мониторинг работы сотрудников в реальном времени, фиксируя каждое перемещение курсора мыши на экране, каждую нажатую клавишу на клавиатуры.

В череповецком отделе ОА «СберТех» на входе установлена СКУД, работающая с RFID-картами, которые закреплены за индивидуальными сотрудниками. При поднесении карты, СКУД заносит время прихода или ухода в базу данных, после чего сам сотрудник или линейный руководитель, может просматривать небольшой отчёт в виде таблице, в которой отражается время прихода и ухода за каждый день, а так же общее отработанное и нормативное время. Пример таблице приведен на рис.3.

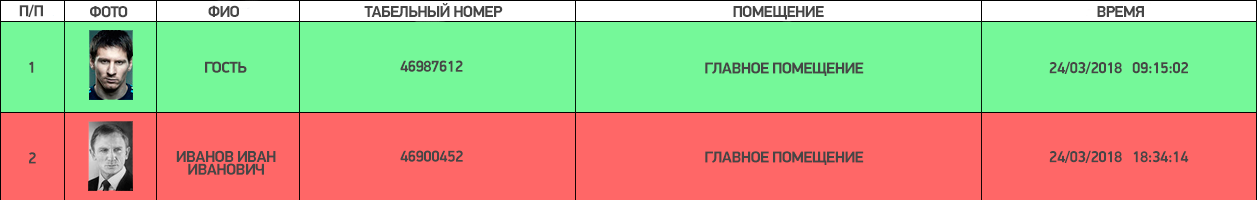


Рис.3. Пример отчётной таблице по работе СКУД

Так как функционал СКУД достаточно мал, а все отсутствия HR Администратор ведет вручную, было принято решение автоматизировать его работу. Для этого необходимо проанализировать рынок на наличие необходимого ПО (табл.3) с учётом следующих критериев:

* оформление заявок на отсутствия (отгул, отпуск, больничный или иное отсутствие);
* ведение графиков рабочего времени для автоматического расчета опозданий и переработки;
* работа с собственными серверами баз данных, так как необходимо соблюдать требования безопасности относительно персональных данных сотрудников;
* контроль метаположения для быстрого поиска сотрудника внутри офиса;
* формирование отчётов, как по отдельному сотруднику, так и по подразделениям в целом, в которых будет указано отработанное и переработанное время, общее число опозданий и отсутствий;
* взаимодействие с персональными пропускными картами;
* масштабируемость – возможность подключения дополнительного оборудования, внесения новых данных, индивидуальная настройка системы;
* цена.

Анализируемые системы:

* TimeControl Office Lite Card - данная система учета рабочего времени позволяет контролировать работу сотрудников с помощью RFID-карт и датчика считывания, подключенного к персональному компьютеру (ПК);
* Kickidler – система, устанавливаемая на ПК сотрудников, позволяющая в реальном времени HR Администратору наблюдать то, чем заняты работники офиса;
* Битрикс24 – система, устанавливаемая на ПК сотрудников, в которой каждый может спланировать свой рабочий день в пределах заданного графика;
* APOLLO – СКУД с возможностью не только фиксировать приход и уход сотрудников, но и вести видеонаблюдение;
* Casi Rusco – СКУД, позволяющая фиксировать не только проходы сотрудников, но и автомобилей, с использованием IP-камер.

Таблица 3

Сравнение известных технических решений

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Критерии | TimeControl Office Lite Card | Kickidler | Битрикс24 | APOLLO | Casi Rusco |
| 1 | 2 | 3 | 4 | 5 | 6 |
| Оформление заявок на отсутствия | - | - | + | - | - |
| Ведение графиков рабочего времени | - | - | + | - | + |
| Формирование отчётов | + | - | +/- | +/- | +/- |

Таблица 3.Продолжение

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 | 6 |
| Взаимодействие с персональными пропускными картами | + | - | - | + | + |
| Работа с собственными серверами баз данных | + | - | - | + | + |
| Контроль метаположения | +/- | - | +/- | - | - |
| Масштабируемость | + | + | + | - | - |
| Цена | + | - | + | - | - |

Из табл.1 можно выделить следующие общие недостатки готовых технических решений:

* из-за отсутствия возможности создавать персональные графики рабочего времени, отсутствует автоматического ведения такой информации, как переработки, задержки и ранние уходы сотрудников, которая влияет в дальнейшем на выплаты заработной платы сотрудникам;
* отсутствие возможности оформление заявок на отгул, командировку или отпуск самими сотрудниками, из-за чего HR Администратору придётся вводить данную информацию вручную;
* не все системы позволяют независимо работать от серверов компании, поставляющих ИС, что может повлечь за собой потерю персональных данных сотрудников и клиентов банка;
* неинформативные отчёты, предлагаемые системами, что влечет за собой дополнительные затраты рабочего времени на доработку полученной информации;
* недостаточная масштабируемость;
* высокая цена за более функциональные системы.

Поэтому было принято решение разработать собственную ИС, под названием HRSaveTime, состоящую из двух подпрограмм:

* серверное приложение HRSaveTimeServer, способное принимать информацию от считывающих устройств пропускных карт и передавать её в БД;
* клиентское приложение HRSaveTimeClient, позволяющее HR Администратору в полуавтоматическом режиме вести временные данные по сотрудникам отдела, а работникам – оформлять временные отсутствия.

Считывающие устройства будут выполнены на базе микроконтроллера Arduino Nano v3.0 и RFID-считывателя пропускных карт RC522C.

Раздел 5. Реализация некоторых из возможных путей решения поставленной в техническом задании на ВКР задачи

5.1. Разработка спецификаций проектируемой системы

Модель системы (или какого-либо другого объекта) – это формальное описание системы, в котором выделены основные объекты, составляющие систему, и отношения между этими объектами.

Спецификация требований программного обеспечения – это законченное описание поведения разрабатываемой программы. Включает ряд пользовательских сценариев (англ. use cases), которые описывают все варианты взаимодействия между пользователями и программным обеспечением.

Пользовательские сценарии являются средством представления функциональных требований. В дополнение к пользовательским сценариям, спецификация также содержит нефункциональные требования, которые налагают ограничения на дизайн или реализацию (такие как требования производительности, стандарты качества, или проектные ограничения).

5.1.1. Построение диаграмм вариантов использования

Диаграмма прецедентов (диаграмма вариантов использования) в UML – диаграмма, отражающая отношения между актерами и прецедентов и являющаяся составной частью модели прецедентов, позволяющей описать систему на концептуальном уровне [1].

Для отражения модели прецедентов на диаграмме используются:

* рамки системы (англ. systemboundary) — прямоугольник с названием в верхней части и эллипсами (прецедентами) внутри. Часто может быть опущен без потери полезной информации;
* актёр (англ. actor) — стилизованный человечек, обозначающий набор ролей пользователя (понимается в широком смысле: человек, внешняя сущность, класс, другая система), взаимодействующего с некоторой сущностью (системой, подсистемой, классом). Актёры не могут быть связаны друг с другом (за исключением отношений обобщения/наследования);
* прецедент — эллипс с надписью, обозначающий выполняемые системой действия (могут включать возможные варианты), приводящие к наблюдаемым актёрами результатам. Надпись может быть именем или описанием (с точки зрения актёров) того, «что» делает система (а не «как»). Имя прецедента связано с непрерываемым (атомарным) сценарием — конкретной последовательностью действий, иллюстрирующей поведение. В ходе сценария актёры обмениваются с системой сообщениями. С одним прецедентом может быть связано несколько различных сценариев.

Диаграмма вариантов использования подсистемы учета рабочего времени и местонахождения сотрудников ЗАО «СберТех» представлена на рис. 4.

С подсистемой могут взаимодействовать три пользователя:

1. Системный администратор (Сисадмин), которому доступен следующий функционал:
   1. настройка системы (подключение к БД, настройка считывающих устройств пропускных карт, проверка статуса подключения);
   2. мониторинг данных, идущих от считывающих устройств. В окне мониторинга данных Сисадмин может вводить запросы для получения информации о сотруднике по т.н., о его местоположении и т.п.
2. HR Администратор, котором доступен следующий функционал:
   1. ведение профилей (изменение информации о сотрудниках, временных данных, присвоение ГРВ);
   2. создание профилей (включая регистрацию пропускной карты и присваивание её к сотруднику);
   3. работа с отчетами.
3. Сотрудник ЗАО «СберТех», которому доступен следующий функционал:
   1. просмотр данных сотрудника (ФИО, контактные данные, должность и т.п.);
   2. создание запроса на отсутствие (отгул, отпуск);
   3. создание временной пары.

Вариант использования «Настройка системы» описывает взаимодействие системного администратора с настройкой системе через приложение HRSaveTimeServer.

Данный вариант использования начинает выполняться, когда системный администратор открывает экранную форму с настройкой системы (табл.4).



Рис. 4. Диаграмма вариантов использования подсистемы учета рабочего времени и местонахождения сотрудников ЗАО «СберТех»

Таблица 4

Типичный ход событий

|  |  |
| --- | --- |
| Действие исполнителя | Отклик системы |
| 1. Сисадмин открывает экранную форму с настройкой системы | 2. Открывается экранная форма с возможными настройки: подключение к БД, создание и редактирование наименований комнат, создание связи комнаты и считывающего устройства. |
| 3. Сисадмин вносит изменения в систему и сохраняет их. | 4. Система сохраняет настройки в файл Settings.txt. |

Вариант использования «Мониторинг потока данных» описывает взаимодействие системного администратора с экранной формой просмотра принятой информации от считывающих устройств пропускных карт в приложение HRSaveTimeServer.

Данный вариант использования начинает выполняться, когда системный администратор открывает экранную форму с настройкой системы (табл.5).

Таблица 5

Типичный ход событий

|  |  |
| --- | --- |
| Действие исполнителя | Отклик системы |
| 1. Сисадмин открывает экранную форму мониторинга потока данных | 2. Открывается экранная форма мониторинга потока данных, в котором в реальном времени отображается информация, получаемая от считывающих устройств пропускных карт. |

Вариант использования «Работа с запросами» описывает взаимодействие системного администратора с экранной формой мониторинга потока данных в момент написания запроса в приложения HRSaveTimeServer.

Данный вариант использования начинает выполняться, когда системный администратор открывает экранную форму с настройкой системы (табл.6).

Таблица 6

Типичный ход событий

|  |  |
| --- | --- |
| Действие исполнителя | Отклик системы |
| 1. Сисадмин открывает экранную форму мониторинга потока данных | 2. Открывается экранная форма мониторинга системы |
| 3. Сисадмин вводит параметрический запрос | 4. Обработка запроса и вывод результата на экран |

Альтернативные потоки:

При неверном вводе параметрического запроса, система выдаст ошибку.

Вариант использования «Проверка подключения датчиков» описывает взаимодействие системного администратора с экранной формой настройки системы в момент проверки статуса подключения датчиков в приложении HRSaveTimeServer.

Данный вариант использования начинает выполняться, когда системный администратор нажимает кнопку проверки (табл.7).

Таблица 7

Типичный ход событий

|  |  |
| --- | --- |
| Действие исполнителя | Отклик системы |
| 1. Сисадмин открывает экранную форму с настройкой системы | 2. Открывается экранная форма с настройками системы |
| 3. Сисадмин выбирает датчик | 4. Система отравляет запрос на датчик и получает ответ, который выдаёт Сисадмину |

Вариант использования «Ведение ролей» описывает взаимодействие системного администратора с экранной формой ведения ролей в приложении HRSaveTimeServer и HRSaveTimeClient для и HR Администратора.

Данный вариант использования начинает выполняться, когда системный администратор или HR Администратора открывает экранную форму ведения ролей (табл.7).

Таблица 8

Типичный ход событий

|  |  |
| --- | --- |
| Действие исполнителя | Отклик системы |
| 1. Сисадмин или HR Администратор открывает экранную форму ведения ролей | 2. Система формирует таблицу со списком ролей |
| 3. Сисадмин или HR Администратор выбирает роль | 4. Система выдает информацию по данной роли (доступный функционал) |

Альтернативные потоки:

Для создания роли системный администратор или HR Администратор добавляет новую запись в таблицу с указанием доступного функционала.

Вариант использования «Работа с отчетами» описывает взаимодействие HR Администратора с экранной формой ведения отчётов в приложениях HRSaveTimeClient.

Данный вариант использования начинает выполняться, когда HR Администратора формирует отчёт с выбранными данными (табл.9).

Таблица 9

Типичный ход событий

|  |  |
| --- | --- |
| Действие исполнителя | Отклик системы |
| 1. HR Администратор открывает экранную форму формирования отчётов | 2. Открывается экранная форма формирования отчётов |
| 2. HR Администратор вносит данные | 3. Система предоставляет сформированный отчет |

Альтернативные потоки

Если сотруднику не нужен новый отчет, он может просмотреть список ранее создаваемых отчётов, нажав на соответствующую кнопку.

Вариант использования «Создание профилей» описывает взаимодействие HR Администратора с экранной формой создания профиля для сотрудника в приложениях HRSaveTimeClient.

Данный вариант использования начинает выполняться, когда HR Администратора нажимает на соответствующую кнопку (табл.10).

Таблица 10

Типичный ход событий

|  |  |
| --- | --- |
| Действие исполнителя | Отклик системы |
| 1. HR Администратор открывает экранную форму для создания профиля сотрудника | 2. Открывается экранная форма создания профиля сотрудника |
| 3. HR Администратор вносит данные | 4. Система проверяет введенные данные |
| 5. HR Администратор сохраняет введенные данные | 5. Система заносит информацию в БД |

Альтернативные потоки

При проверке введенных данных система предупреждает HR Администратора о неверно введенных данных.

Вариант использования «Ведение профилей» описывает взаимодействие HR Администратора с экранной формой ведения профилей в приложениях HRSaveTimeClient.

Данный вариант использования начинает выполняться, когда HR Администратора открывает список сотрудников (табл.11).

Таблица 11

Типичный ход событий

|  |  |
| --- | --- |
| Действие исполнителя | Отклик системы |
| 1 | 2 |
| 1. HR Администратор открывает экранную форму ведения профилей | 2. Открывается экранная форма со списком сотрудников |
| 2. HR Администратор выбирает нужного сотрудника | 3. Открывается профиль сотрудника |

Таблица 11. Продолжение

|  |  |
| --- | --- |
| 1 | 2 |
| 3. HR Администратор, находясь в профиле сотрудника, запускает процесс редактирования | 4. Открывается экранна форма для редактирования профиля |
| 5. HR Администратор вносит изменения | 6. Система проверяет введенные данные |
| 7. HR Администратор сохраняет изменения | 8. Система заносит информацию в БД |

Альтернативные потоки

При проверке введенных данных система предупреждает HR Администратора о неверно введенных данных.

Вариант использования «Ведение ГРВ» описывает взаимодействие HR Администратора с экранной формой м ведения ГРВ в приложениях HRSaveTimeClient.

Данный вариант использования начинает выполняться, когда HR Администратора нажимает на кнопку добавления ГРВ (табл.12).

Таблица 12

Типичный ход событий

|  |  |
| --- | --- |
| Действие исполнителя | Отклик системы |
| 1 | 2 |
| 1. HR Администратор открывает экранную форму создания рабочих графиков | 2. Система открывает экранную форму со списком всех персональных графиков рабочего времени (ПГРВ) |
| 3. HR Администратор запускает процесс создания нового графика | 4. Система открывает экранную форму для создания перерыва |
| 5. HR Администратор вносит данные | 6. Система проверяет введенные данные |
| 7. HR Администратор сохраняет изменения | 8. Система записывает в БД данные о новом перерыве и открывает экранную форму для создания однодневного графика рабочего времени (ОГРВ) |
| 9. HR Администратор вносит данные | 10. Система проверяет введенные данные |

Таблица 12. Продолжение

|  |  |
| --- | --- |
| 1 | 2 |
| 11. HR Администратор сохраняет изменения | 12. Система записывает в БД данные о новом ОГРВ и открывает экранную форму для создания персонального графика рабочего времени (ПГРВ) |
| 13. HR Администратор вносит данные | 14. Система проверяет введенные данные |
| 15. HR Администратор сохраняет изменения | 16. Система записывает в БД данные о новом ПГРВ |

Альтернативные потоки

При проверке введенных данных система предупреждает HR Администратора о неверно введенных данных или о уже существующем графике.

Вариант использования «Создание временных пар» описывает взаимодействие Сотрудника с датчиком считывания пропускных карт.

Данный вариант использования начинает выполняться, когда Сотрудника подносит свою RFID карту к считывающему устройству (табл.13).

Таблица 13

Типичный ход событий

|  |  |
| --- | --- |
| Действие исполнителя | Отклик системы |
| 1. Пользователь подносит пропуск к датчику на пропускном пункте | 2. В Базу данных (БД) системы записывается фактическое время прихода на рабочее место, если пользователь подносит пропуск к датчику, который расположен на входе в кабинет, либо ухода, если датчик расположен на выходе из кабинета, а так же наименование кабинета |

Альтернативные потоки

Пропуск не зарегистрирован. В этом случае на пропускном пункте загорится красная лампочка, символизируя ошибку. Доступ в помещение будет ограничено. Вследствие чего в БД вместо временной пары запишется ошибка.

Вариант использования «Создания запроса на отсутствие» описывает взаимодействие Сотрудника с экранной формой м создания запроса на отсутствие в приложении HRSaveTimeClient.

Данный вариант использования начинает выполняться, когда Сотрудника открывает экранную форму создания запроса (табл.14).

Таблица 14

Типичный ход событий

|  |  |
| --- | --- |
| Действие исполнителя | Отклик системы |
| 1 | 2 |
| 1. Пользователь открывает экранную форму для создания отсутствия | 2. Открывается экранна форма для заполнения заявки на отсутствие |
| 3. Пользователь вносит данные | 4. Система проверяет введенные данные |
| 5. Пользователь отправляет заявку на согласование | 6. Система сохраняет заявку в БД и оповещает HR Администратора о новой заявке |
| 7. HR Администратор утверждает или отклоняет заявку | 8. Система сохраняет ответ в БД и оповещает сотрудника о решении HR Администратора |

Вариант использования «Просмотр собственного профиля» описывает взаимодействие Сотрудника с экранной формой м просмотра профиля в приложениях HRSaveTimeClient.

Данный вариант использования начинает выполняться, когда Сотрудника открывает свой профиль (табл.15).

Таблица 15

Типичный ход событий

|  |  |
| --- | --- |
| Действие исполнителя | Отклик системы |
| 1. Сотрудник открывает свой профиль | 2. Система выводит информацию по сотруднику, в том числе все временные данные (приходы/уходы, отсутствия и т.п.) |



5.1.2. Построение диаграмм деятельностей сценариев вариантов использования

На этапе анализа требований и уточнения спецификаций диаграммы деятельностей позволяют конкретизировать основные функции разрабатываемого программного обеспечения. Под деятельностью в данном случае понимают задачу (операцию), которую необходимо выполнить вручную или с помощью средств автоматизации. Каждому варианту использования соответствует своя последовательность задач. В теоретическом плане диаграммы деятельности являются обобщенным представлением алгоритма, реализующего анализируемый вариант использования.

Учитывая описание предметной области в виде контекстной диаграммы классов, анализируем описание сценария варианта использования Управление временными данными. Производим разбиение процесса на отдельные операции. Полученные операции показаны на диаграммах деятельностей (рис.5- рис.9). Описание деятельностей представлены в табл.16 – табл.20.



Рис. 5. Диаграмма деятельностей для сценария варианта использования «Создание временной пары»

Таблица 16

Описание деятельностей для сценария варианта использования Управление временными данными

|  |  |  |
| --- | --- | --- |
| № | Деятельность | Описание |
| 1 | 2 | 3 |
| 1 | Создание временной пары | Подпрограмма, позволяющая сотрудникам, поднося пропуск к считывающему устройству, фиксировать свой приход и уход на рабочее место и в конкретный кабинет. |
| 2 | Запись данных в БД | Если ID RFID-карты привязан к табельному номеру сотрудника, то в БД создается запись вида «дата, время, сотрудник, помещение, направление». |
| 3 | Вывод информации в HRSaveTimeServer | При удачном чтении RFID ID-карты в приложении HRSaveTimeServer в окне мониторинга потока данных отобразится информация о сотруднике. |

Таблица 16. Продолжение

|  |  |  |
| --- | --- | --- |
| 1 | 2 | 3 |
| 4 | Зеленая лампочка на считывающем устройстве | Если связь сотрудника и ID RFID-карты найдена, то на считывающем устройстве загорается зеленая лампочка, сигнализируя о доступе в помещение. |
| 5 | Вывод ошибки в HRSaveTimeServer | Если ID RFID-карты не привязан к табельному номеру сотрудника, то в приложении HRSaveTimeServer в окне мониторинга потока данных отобразится ошибка. |
| 6 | Красная лампочка на считывающем устройстве | Если связь сотрудника и ID RFID-карты не найдена, то на считывающем устройстве загорается красная лампочка, сигнализируя об ошибке. |



Рис. 6. Диаграмма деятельностей для сценария варианта использования «Ведение профилей»

Таблица 17

Описание деятельностей для сценария варианта использования Ведение профилей

|  |  |  |
| --- | --- | --- |
| № | Деятельность | Описание |
| 1 | Экранная форма ведения профилей | Открытие окна со списком сотрудников |
| 2 | Просмотр профиля | Открытие профиля выбранного сотрудника, в котором отображается его контактная информация, временные данные, отсутствия. |
| 3 | Ведение ролей | Открытие окна со списком ролей |
| 4 | Экранная форма добавление ролей | Открытие окна для создания новой роли |
| 5 | Экранная форма просмотра выбранной роли | Открытие окна с подробной информации о выбранной роли |
| 6 | Вывод ошибки | При попытке создать роль с созданным идентификатором или аналогичным функционалом, система выдаст ошибку |
| 7 | Ведение ГРВ | Открытие окна со списком доступных ГРВ |
| 8 | Просмотр ГРВ | Открытие окна с подробным описанием выбранного ГРВ |
| 9 | Создание перерыва | Открытие окна для создания нового перерыва |
| 10 | Создание ОГРВ | Открытие окна для создания нового ОГРВ |
| 11 | Создание ПГРВ | Открытие окна для создания нового ПГРВ |



Рис. 7. Диаграмма деятельностей для сценария варианта использования «Создание запросов на отсутствие»

Таблица 18

Описание деятельностей для сценария варианта использования Создание запросов на отсутствие

|  |  |  |
| --- | --- | --- |
| № | Деятельность | Описание |
| 1 | Заполнение информации | В открывшемся окне сотрудник заносит необходимую информацию по предстоящему отсутствию |
| 2 | Отправка заявки HR Администратору | Процесс передачи заявки на отсутствие от сотрудника к HR Администратору |
| 3 | Отправка принятого решения сотруднику | После того, как HR Администратор ответит на заявку на отсутствие, принятое решение отправляется сотруднику |



Рис. 8. Диаграмма деятельностей для сценария варианта использования «Мониторинг потока данных»

Таблица 19

Описание деятельностей для сценария варианта использования Мониторинг потока данных

|  |  |  |
| --- | --- | --- |
| № | Деятельность | Описание |
| 1 | Экранная форма мониторинга потока данных | Открытие окна мониторинга потока данных |
| 2 | Проверка ведения запроса | Если был введен параметрический запрос, то он проходит проверку на корректность |
| 3 | Вывод ошибки | Если параметрический запрос введен не верно, то система выдаст текст ошибки |
| 4 | Выполнение запроса и вывод результатов | Система выполняет запрос, согласно введенным параметрам |
| 5 | Просмотр потока данных | Обновление окна потока данных в реальном времени |

|  |  |
| --- | --- |
|  | Рис. 9. Диаграмма деятельностей для сценария варианта использования «Настройка системы» |

Таблица 20

Описание деятельностей для сценария варианта использования Настройка системы

|  |  |  |
| --- | --- | --- |
| № | Деятельность | Описание |
| 1 | Экранная форма настройки системы | Открытие окна с категориями настроек |
| 2 | Ведение ролей | Открытие окна со списком ролей |
| 3 | Экранная форма добавление ролей | Открытие окна для создания новой роли |
| 4 | Экранная форма просмотра выбранной роли | Открытие окна с подробной информации о выбранной роли |
| 5 | Вывод ошибки | При попытке создать роль с созданным идентификатором или аналогичным функционалом, система выдаст ошибку |
| 6 | Экранная форма просмотра выбранной роли | Вывод информации по выбранной роли |
| 7 | Изменить роль | Процесс изменения роли |
| 8 | Настройка комнат | Открытие окна для создания и изменения комнат |
| 9 | Экранная форма добавление комнаты | Открытие окна для создания новой комнаты |
| 10 | Экранная форма просмотра выбранной комнате | Открытие окна с подробной информации о выбранной комнате |
| 11 | Вывод ошибки | При попытке создать аналогичную комнату, система выдаст ошибку |
| 12 | Экранная форма просмотра выбранной комнаты | Вывод информации по выбранной комнате |
| 13 | Изменить комнату | Процесс изменения комнаты |
| 14 | Настройка датчиков | Создание связи «ID комнаты – ID датчика» |
| 15 | Настройка БД | Процесс подключения к БД |
| 16 | Проверка подключения датчиков | При выборе проверяемого датчика, система посылает данные на него и получает ответ |
| 17 | Сохранение | Сохранение изменений в БД |

* + 1. Проектирование пакета «БД»

|  |  |
| --- | --- |
|  | Рис.10. Физическая модель |

После создания физической модели данных была спроектирована БД в СУБД Oracle XE.

Таблица 21

Идентификаторы таблицы «Pers\_info»

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Атрибут | Идентификатор | Тип | Размерность | Описание |
| ID | IDPers | NUMBER | 10 | ID сотрудника |
| Фамилия | LName | VARCHAR | 40 | Фамилия сотрудника |
| Имя | Name | VARCHAR | 40 | Имя сотрудника |
| Отчество | Patr | VARCHAR | 40 | Отчество сотрудника |
| Дата рождения | Birth | Date | 10 | Дата рождения сотрудника |
| Должность | Position | VARCHAR | 50 | Должность сотрудника |
| Организационное присвоение | OrgID | NUMBER | 10 | Организационное присвоение |
| График | PRGVID | VARCHAR | 5 | График рабочего времени сотурднкиа |
| Роль | Rule | VARCHAR | 5 | Доступный набор полномочий |
| Личный мобильный телефон | LMmobile | VARCHAR | 20 | Личный мобильный телефон сотрудника |
| Рабочий мобильный телефон | WMmobile | VARCHAR | 20 | Рабочий мобильный телефон сотрудника |
| Рабочий телефон | Wphone | VARCHAR | 10 | Рабочий телефон сотрудника |
| Личный Email | Lemail | VARCHAR | 30 | Личный Email сотрудника |
| Рабочий Email | Wemail | VARCHAR | 30 | Рабочий Email сотрудника |

Таблица 22

Табельные номер

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Атрибут | Идентификатор | Тип | Размерность | Описание |
| Табельный номер | IDPernr | NUMBER | 8 | Табельный номер сотрудника |
| ID сотрудника | PersID | NUMBER | 10 | ID сотрудника |

Таблица 23

Пропускные карты

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Атрибут | Идентификатор | Тип | Размерность | Описание |
| RFID ID-карты | IDRFID | NUMBER | 10 | ID пропускной карты |
| Табельный номер | Pernr | NUMBER | 8 | Табельный номер сотрудника |

Таблица 24

Временные пары

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Атрибут | Идентификатор | Тип | Размерность | Описание |
| 1 | 2 | 3 | 4 | 5 |
| ID | ID | NUMBER | 10 | ID записи |
| RFID ID-карты | RFID | VARCHAR | 10 | ID пропускной карты |
| Дата с | DateFrom | Date | 10 | Дата, с которой сотрудник находится в том или ином помещении |
| Дата по | DateBy | Date | 10 | Дата, по которую сотрудник находится в том или ином помещении |
| Время с | TimeFrom | Date | 10 | Время, с которого сотрудник находится в том или ином помещении |

Таблица 24. Продолжение

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 |
| Время по | TimeBy | Date | 10 | Время, по которое сотрудник находится в том или ином помещении |
| Помещение | Room | NUMBER | 10 | Помещение, в котором находился сотрудник |

Таблица 25

Комнаты

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Атрибут | Идентификатор | Тип | Размерность | Описание |
| ID | ID | NUMBER | 10 | ID записи |
| Наименование | RoomName | VARCHAR | 50 | Наименование помещения |

Таблица 26

Авторизация

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Атрибут | Идентификатор | Тип | Размерность | Описание |
| ID | ID | NUMBER | 10 | ID записи |
| Табельный номер | Pernr | NUMBER | 8 | Табельный номер сотрудника |
| Логин | Login | VARCHAR | 50 | Логин сотрудника для входа в систему |
| Пароль | Password | VARCHAR | 50 | Пароль сотрудника для входа в систему |

Таблица 27

Отсутствия

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Атрибут | Идентификатор | Тип | Размерность | Описание |
| 1 | 2 | 3 | 4 | 5 |
| ID | ID | NUMBER | 10 | ID отсутствия |

Таблица 27. Продолжить

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 |
| Табельный номер | Pernr | NUMBER | 8 | Табельный номер сотрудника |
| Вид отсутствия | IDabsen | NUMBER | 10 | ID отсутствия |
| Дата с | DateFrom | Date | 10 | Дата, с которой сотрудник отсутствовал |
| Дата по | DateBy | Date | 10 | Дата, по которое сотрудник отсутствовал |
| Документ | Doc | VARCHAR | 200 | Ссылка на расположение локальной копии документа |

Таблица 28

Виды отсутствий

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Атрибут | Идентификатор | Тип | Размерность | Описание |
| ID | ID | NUMBER | 10 | ID вида отсутствия |
| Наименование | Name | VARCHAR | 50 | Наименование вида отсутствия |

Таблица 29

ПГРВ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Атрибут | Идентификатор | Тип | Размерность | Описание |
| 1 | 2 | 3 | 4 | 5 |
| ID | ID | NUMBER | 10 | ID ПГРВ |
| Дата с | DateFrom | Date | 10 | Год, с которого может использоваться ПГРВ |
| Дата по | DateBy | Date | 10 | Год, по который может использоваться ПГРВ |

Таблица 29. Продолжение

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 |
| ОГРВ1 | OGRV1 | VARCHAR | 10 | ОГРВ для понедельника |
| ОГРВ2 | OGRV2 | VARCHAR | 10 | ОГРВ для вторника |
| ОГРВ3 | OGRV3 | VARCHAR | 10 | ОГРВ для среды |
| ОГРВ4 | OGRV4 | VARCHAR | 10 | ОГРВ для четверга |
| ОГРВ5 | OGRV5 | VARCHAR | 10 | ОГРВ для пятницы |
| ОГРВ6 | OGRV6 | VARCHAR | 10 | ОГРВ для субботы |
| ОГРВ7 | OGRV7 | VARCHAR | 10 | ОГРВ для воскресенья |

Таблица 30

ОГРВ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Атрибут | Идентификатор | Тип | Размерность | Описание |
| ID | ID | NUMBER | 10 | ID ОГРВ |
| Норма время с | NormFrom | Date | 10 | Время, с которого начинается нормативный рабочий день |
| Норма время по | NormBy | Date | 10 | Время, по которое длится нормативный рабочий день |
| Сокращенный время с | AdFrom | Date | 10 | Время, с которого начинается сокращенный рабочий день |
| Сокращенный время по | AdBy | Date | 10 | Время, по которое длится сокращенный рабочий день |
| Перерыв | BreakID | VARCHAR | 10 | ID перерыва |

Таблица 31

Перерыв

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Атрибут | Идентификатор | Тип | Размерность | Описание |
| ID | ID | NUMBER | 10 | ID перерыва |
| Время с | TimeFrom | Date | 10 | Время начала перерыва |
| Время по | TimeBy | Date | 10 | Время окончания перерыва |

Таблица 32

Роли

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Атрибут | Идентификатор | Тип | Размерность | Описание |
| 1 | 2 | 3 | 4 | 5 |
| ID | ID | NUMBER | 10 | ID роли |
| Описание | Descript | VARCHAR | 50 | Описание роли |
| Профиль | Profile | VARCHAR | 50 | Полномочия на ведение профилей |
| Временный пары | TimePairs | VARCHAR | 50 | Полномочия на ведение временных пар |
| Помещения | Rooms | VARCHAR | 50 | Полномочия на ведение помещений |
| Отсутствия | Absence | VARCHAR | 50 | Полномочия на ведение отсутствий |
| Вид отсутствия | ViewAbsence | VARCHAR | 50 | Полномочия на ведение видов отсутствий |
| ПГРВ | PGRV | VARCHAR | 50 | Полномочия на ведение ПГРВ |
| ОГРВ | OGRV | VARCHAR | 50 | Полномочия на ведение ОГРВ |
| Перерыв | Break | VARCHAR | 50 | Полномочия на ведение перерывов |

Таблица 32. Продолжение

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 |
| Орг.уровни | ORGLevel | VARCHAR | 50 | Полномочия на ведение орг.уровней |
| Роли | Rules | VARCHAR | 50 | Полномочия на ведение ролей |
| Мониторинг данных | Monitor | VARCHAR | 50 | Полномочия на просмотр входных данных и работы с запросами |
| Настройка системы | Settings | VARCHAR | 50 | Полномочия на настройку системы |

Таблица 33

Организационные уровни

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Атрибут | Идентификатор | Тип | Размерность | Описание |
| ID | ID | NUMBER | 10 | ID орг.уровня |
| Наименование | Name | VARCHAR | 50 | Наименование орг.уровня |
| Линейный руководитель | LPernr | NUMBER | 8 | Табельный номер сотрудника, являющийся линейным руководителем орг.уровня |

5.2. Проектирование интерфейса пользователя

5.2.1. Построение графа диалога

Пользовательский интерфейс – комплекс программных и аппаратных средств для обеспечения информационного взаимодействия компьютера и пользователя. Основу такого взаимодействия составляют диалоги. Под диалогом понимают регламентированный обмен информацией между человеком и компьютером, направленный на решение конкретной задачи. При проектировании диалогов используют графы диалога.

Граф диалога – ориентированный взвешенный граф, каждой вершине которого сопоставлена конкретная картинка на экране (кадр) или определенное состояние диалога, характеризующееся набором доступных пользователю действий. Дуги, исходящие из вершин, показывают возможные изменения состояний при выполнении пользователем указанных действий. В качестве весов дуг указывают условия переходов из состояния в состояние и операции, выполняемые во время перехода. Каждый маршрут на графе соответствует возможному варианту диалога [13].

Граф диалога пользователя с HRSaveTimeServer изображен на рис. 43. Описание вершин графа представлено в табл. 100. Описание дуг графа представлено в табл.101.

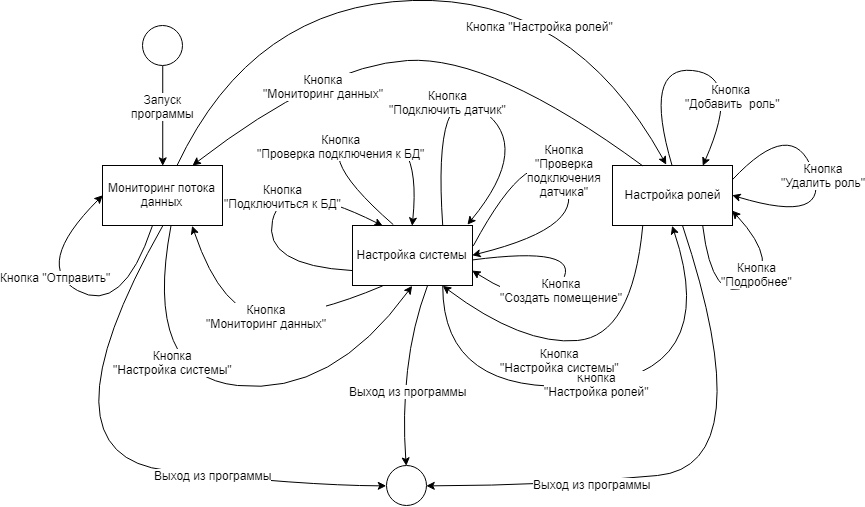


Рис.11. Граф диалога пользователя с HRSaveTimeServer

Таблица 34

Описание вершин графа

|  |  |
| --- | --- |
| Вершина графа | Описание |
| Мониторинг потока данных | Экранная форма с набором элементов, позволяющее наблюдать за входящей информацией со считывающих устройств пропускных карт |
| Настройка системы | Экранная форма с набором элементов, позволяющие провести базовую настройку системы |
| Настройка ролей | Экранная форма с набором элементов , позволяющие создавать роли |

Таблица 35

Описание дуг графа

|  |  |
| --- | --- |
| Дуга графа | Описание |
| 1 | 2 |
| Запуск программы | При запуске HRSaveTimeServer Открывается экранна форма «Мониторинг потока данных» |
| Кнопка «Отправить» | При нажатии на кнопку «Отправить» выполняется обработка запроса, введенного в поле ввода |
| Кнопка «Настройка системы» | При нажатии на кнопку «Настройка системы» Открывается экранна форма с настройкой системы |
| Кнопка «Мониторинг данных» | При нажатии на кнопку «Мониторинг данных» Открывается экранна форма для мониторинга данных и ввода запросов |
| Кнопка «Настройка ролей» | При нажатии на кнопку «Настройка ролей» Открывается экранна форма с настройкой ролей |

Таблица 35. Продолжение

|  |  |
| --- | --- |
| 1 | 2 |
| Выход из программы | При нажатии на кнопку закрытия окна происходит закрытие программы |
| Кнопка «Подключение к БД» | При нажатии на кнопку «Подключение к БД» выполняется функция подключения к базе данных, при которой система запоминает данные для последующей работы с БД |
| Кнопка «Проверка подключения к БД» | При нажатии на кнопку «Проверка подключения к БД» выполняется функция попытка подключения к БД и вывод статуса подключения |
| Кнопка «Подключить датчик» | При нажатии на кнопку «Подключить датчик» система записывает связь кабинета и датчика в БД |
| Кнопка «Проверка подключения датчика» | При нажатии на кнопку «Проверка подключения датчика» система отправит запрос на выбранный датчик и будет ждать ответа, после чего выдаст статус подключения |
| Кнопка «Создать помещение» | При нажатии на кнопку «Создать помещение» откроется экранная форма для ввода называния помещения |
| Кнопка «Добавить роль» | При нажатии на кнопку «Добавить роль» откроется экранная форма для ввода названия роли и присваивания полномочий |
| Кнопка «Удалить роль» | При нажатии на кнопку «Удалить роль» из таблицы удаляется выбранная роль |
| Кнопка «Подробнее» | При нажатии на кнопку «Подробнее» Открывается экранная форма с подробностями по выбранной роли |

Граф диалога пользователя с HRSaveTimeClient изображен на рис. 44. Описание вершин графа представлено в табл. 102. Описание дуг графа представлено в табл. 103.

Таблица 36

Описание вершин графа

|  |  |  |
| --- | --- | --- |
| Вершина графа | Описание | |
| 1 | 2 | |
| Мой профиль | Экранная форма с набором элементов, позволяющая отображать информацию о сотруднике, вошедший в систему | |
| Список сотрудников | Экранная форма с набором элементов, позволяющая отображать список сотрудников | |
| Создание профиля | Экранная форма с набором элементов, позволяющая создать профиль сотрудника | |
| Просмотр профиля | Экранная форма с набором элементов, позволяющая отображать подробную информацию о сотруднике и редактировать её | |
| Список созданных запросов | Экранная форма с набором элементов, позволяющая отображать список созданных запросов на отсутствие | |
| Создание запросов | Экранная форма с набором элементов, позволяющие создать запрос на отсутствие | |
| Просмотр запросов | Экранная форма с набором элементов, позволяющая отобразить подробную информацию об отсутствии | |
| Список сохраненных отчётов | Экранная форма с набором элементов, позволяющая отобразить список сохраненных шаблонов отчётов | |
| Формирование отчёта | Экранная форма с набором элементов, позволяющая ввести необходимую информацию для формирования отчёта | |
|  | | Рис. 12. Граф диалога пользователя с HRSaveTimeClient | |

Таблица 36. Продолжение

|  |  |
| --- | --- |
| 1 | 2 |
| Результат | Экранная форма с набором элементов, позволяющая просматривать сформированный отчёт перед печатью |
| Список ГРВ | Экранная форма с набором элементов, позволяющая просматривать список созданных графиков |
| Подробная информация о ГРВ | Экранная форма с набором элементов, позволяющая отобразить подробную информацию о графике |
| Создание ГРВ | Экранная форма с набором элементов, позволяющая создать новый график |
| Окно перерывы | Экранная форма с набором элементов, позволяющая просматривать все перерывы и производить их редактирование |
| Окно ОГРВ | Экранная форма с набором элементов, позволяющая просматривать все ОГРВ и производить их редактирование |
| Список ролей | Экранная форма с набором элементов, позволяющая просматривать все созданные роли |
| Подробная информация о роли | Экранная форма с набором элементов, позволяющая получить подробную информацию о роли |
| Создание роли | Экранная форма с набором элементов, позволяющая создать новую роль |
| Мониторинг данных | Экранная форма с набором элементов, позволяющая просматривать входящие данные с устройств считывания пропускных карт |

Таблица 37

Описание дуг графа

|  |  |
| --- | --- |
| Дуга графа | Описание |
| 1 | 2 |
| Запуск системы | При запуске HRSaveTimeClient открывается экранная форма входа |
| Кнопка «Вход» | При нажатии на кнопку «Вход» система проводит аутентификацию пользователя по введенным данным |
| Кнопка «Выход» | При нажатии на кнопку «Выход» происходит выход из системы |
| Кнопка «Сотрудники» | При нажатии на кнопку «Сотрудники» отображается список всех сотрудников, зарегистрированных в системе |
| Кнопка «Создать профиль» | При нажатии на кнопку «Создать профиль» отображается экранная форма для создания нового профиля сотрудника |
| Кнопка «Подробнее» | При нажатии на кнопку «Подробнее» отображается подробная информация |
| Кнопка «Запрос на отсутствие» | При нажатии на кнопку «Запрос на отсутствие» отображается весь перечень созданных запросов |
| Кнопка «Создать запрос» | При нажатии на кнопку «Создать запрос» отображается экранная форма для создания запроса |
| Кнопка «Отправить запрос» | При нажатии на кнопку «Отправить запрос» запрос на отсутствие отправляется HR Администратору для согласования |
| Кнопка «Отчеты» | При нажатии на кнопку «Отчеты» отображается список созданных шаблонов для отчётов |
| Кнопка «Создать отчёт» | При нажатии на кнопку «Создать отчёт» отображается экранная форма для создания отчёта |

Таблица 37. Продолжение

|  |  |
| --- | --- |
| 1 | 2 |
| Кнопка «Сохранить отчёт» | При нажатии на кнопку «Сохранить отчёт» сохраняется шаблон для быстрого формирования отчёта в будущем |
| Кнопка «Сформировать отчёт» | При нажатии на кнопку «Сформировать отчёт» формируется отчёт на основе введенной информации и отображается в окне предварительного просмотра |
| Кнопка «Выгрузить» | При нажатии на кнопку «Выгрузить» отчёт в виде Excel-файла выгружается на локальный жёсткий диск сотрудника |
| Кнопка «Графики» | При нажатии на кнопку «Графики» отображается экранная форма со списком созданных графиков |
| Кнопка «Создать график» | При нажатии на кнопку «Создать график» отображается экранная форма для создания графика |
| Кнопка «Поиск перерыва» | При нажатии на кнопку «Поиск перерыва» отображается экранная форма со списком созданных перерывов с возможностью их редактирования |
| Кнопка «Поиск ОГРВ» | При нажатии на кнопку «Поиск ОГРВ» отображается экранная форма со списком созданных ОГРВ с возможностью их редактирования |
| Кнопка «Роли» | При нажатии на кнопку «Роли» отображается экранная форма со списком созданных ролей |
| Кнопка «Создать роль» | При нажатии на кнопку «Создать роль» отображается экранная форма для создания роли |

Таблица 37. Продолжение

|  |  |
| --- | --- |
| 1 | 2 |
| Кнопка «Мониторинг данных» | При нажатии на кнопку «Мониторинг данных» отображается экранная форма с возможностями просмотра входной информации и работы с запросами |
| Кнопка «Отправить» | При нажатии на кнопку «Отправить» обрабатывается введенный в поле ввода запрос. Результат выводится в окне мониторинга потока данных |
| Кнопка «Редактировать» | При нажатии на кнопку «Редактировать» отображается экранная форма для редактирования данных |
| Кнопка «Сохранить» | При нажатии на кнопку «Сохранить» введенная информация сохраняется в БД в соответствующих таблицах |
| Кнопка «Отмена» | При нажатии на кнопку «Отмена» все совершенные действия отменяются |
| Выход из системы | Завершение всех процессов HRSaveTimeClient |

5.2.2. Разработка форм ввода-вывода информации

Для подсистемы учёта рабочего времени и местоположения сотрудников АО «СберТех» был выбран интерфейс со свободной навигацией. Он поддерживает концепцию интерактивного взаимодействия с системой, визуальную обратную связь с пользователем и возможность прямого манипулирования объектом.

При запуске HRSaveTimeServer появляется экранная форма мониторинга потока данных, в котором системный администратор может наблюдать в реальном времени информацию, которую передают датчики считывания пропускных карт.

В окне мониторинга потока данных системный администратор может вводить параметрические запросы (команды) для получения актуальной информации, например, по табельному номеру сотрудника.

Из данного окна можно перейти в «Настройки системы» и «Управление ролями».

При нажатии на «Настройки системы», системному администратору предлагается предварительно настроить систему перед эксплуатацией:

* настройка подключения к базе данных;
* настройка помещений;
* подключение датчиков.

В окне «Управление ролями» сотруднику предлагается добавить необходимые роли, которые в дальнейшем будут присвоены сотрудникам.

При запуске HRSaveTimeClient открывается экранная форма авторизации в системе. После успешного ввода логина и пароля, открывается экранная форма с личным профилем сотрудника. Используя боковое меню с левой стороны, сотрудник может попасть в одну из экранных форм, в зависимости от потребностей и роли, которая присвоена сотруднику.

В экранной форме «Сотрудники» отображается список профилей, созданные в системе. При необходимости, можно создать новый, заполнив соответствующие поля.

В экранной форме «Запросы» отображаются все созданные сотрудником запросы. При необходимости, можно создать новую, заполнив соответствующие поля и отправить её на согласование. В свою очередь, HR Администратор видит весь список запросов, у каждого из которых способен менять статус на «Подтверждено» или «Отклонено».

В экранной форме «Отчёты» сотруднику доступен список всех сохраненных ранее им отчётов, с возможностью повторного формирования. При необходимости, можно создать новый, заполнив соответствующие поля.

В экранной форме «Графики» HR Администратору доступен перечень созданных ранее ГРВ, с возможностью их редактирования. При необходимости, можно создать новый, заполнив соответствующие поля.

В экранной форме «Мониторинг» СисАдмину доступна потоковая информация, получаемая со считывающих устройств. Для получения информации по табельным номерам, по RFID меткам, сотрудник может в поле ввода вносить параметрические запросы.

Раздел 6. Технико-экономическое обоснование выполняемой разработки

6.1. Расчёт трудоёмкости разработки

* предполагаемое число операторов q = 5000;
* коэффициент сложности разработки с = 1,5;
* коэффициент коррекции программы в ходе ее разработки р = 0,1;
* коэффициент квалификации разработчиков k = 0,5;
* коэффициент увеличения затрат труда вследствие недостатков описания задачи B = 1.

Трудоемкость разработки программного обеспечения решения задачи можно рассчитать по формуле:

где – затраты труда на изучение описания задачи; – затраты труда на разработку алгоритма решения задачи; – затраты труда на программирование по готовой блок-схеме; – затраты труда на отладку программы на ЭВМ; – затраты труда на подготовку документации.

Результат:

чел-час

Условное число операторов в программе определяется по формуле:

Результат:

чел-час

Затраты труда на изучение описания задачи (с учетом квалификации программиста):

чел-час

Затраты труда на разработку алгоритма решения задачи :

чел-час

Затраты труда на программирование по готовой блок-схеме:

чел-час

Затраты труда на отладку программы на ЭВМ при автономной отладке одной задачи:

чел-час

При комплексной отладке задачи:

чел-час

Затраты труда на подготовку документации по задаче:

чел-час

Затраты труда на подготовку материалов в рукописи:

чел-час

Затраты труда на редактирование, печать и оформление документации:

чел-час

* 1. Расчет себестоимости разработки

В себестоимость разработки входят: основная заработная плата разработчиков, дополнительная заработная плата, единый социальный налог, затраты на использование машинного времени, затраты на носители информации, затраты на текущий и профилактический ремонт вычислительной техники, прочие эксплуатационные расходы.

Себестоимость разработки рассчитывается по формуле:

где – себестоимость программного продукта, руб.; – затраты на оплату труда, руб.; – затраты на использование машинного времени, руб.; – затраты на носители информации, руб.; – затраты на текущий и профилактический ремонт вычислительной техники, руб.; – прочие эксплуатационные расходы, руб.

Результат:

Состав команда проекта: программист-дипломник, руководитель ВКР.

Затраты на оплату труда при разработке программного продукта вычисляются по формуле:

где – общая зарплата работника за час; – отчисления с зарплаты, %; – время написания программы. Время написания программы совпадает с временем работы компьютера.

Результат:

Заработная плата программиста за час определяется по следующей формуле:

где – ставка программиста; – фонд рабочего времени в месяц, ч.

Результат, с учётом :

Заработная плата дополнительная определяется по следующей формуле:

где – заработная плата; – норма отчислений на дополнительную зарплату (10 %).

Результат:

Зарплата общая вычисляется по следующей формуле:

Результат:

Отчисления на соцстрах, фонд занятости и пенсионный фонд вычисляются по следующей формуле:

где – отчисления на соцстрах (0,5 % от Зобщ); – отчисления в фонд занятости (0,5 % от Зобщ); – отчисления в Пенсионныйфонд (2 % от ).

Результат:

Затраты на использование машинного времени вычисляются по формуле:

где Зм.вр – затраты на использование машинного времени, руб.; См.вр – стоимость одного часа машинного времени, руб./ч; Врв.т – время использования вычислительной техники, ч.

Результат:

Таблица 38

Данные по заработной плате

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Должность  разработчика | Разряд | Время работы, мес. | Стпр,  руб. | Зпр,  руб. | Здоп,  руб. | Зобщ,  руб. | Отч,  руб. | Зтр, руб. |
| Программист-дипломник | 1 | 3 | 30000 | 312,5 | 97,7 | 410,2 | 49,2 | 132 307,2 |
| Научный руководитель ВКР | 5 | 3 | 19400 | 202,1 | 40,8 | 242,9 | 29,2 | 80 380,8 |

Стоимость одного часа машинного времени рассчитывается по формуле:

где – стоимость одного часа машинного времени, руб./ч; – покупная цена компьютера, руб.; – срок службы компьютера, год; – количество рабочих дней в году; – время работы компьютера в течение суток, ч; – стоимость одного кВт\*ч электроэнергии, руб.; – мощность вычислительной системы, кВт.

Результат:

Время использования вычислительной техники () рассчитывается по следующей формуле:

где Кд.р – количество дней разработки ПО; Врс – время работы компьютера в течение суток, ч.

Результат:

Затраты на носители информации принимаются в размере 2 % от цены вычислительной техники.

Затраты на текущий и профилактический ремонт принимаются в размере 4 % от цены вычислительной техники. Прочие эксплуатационные расходы включают в себя затраты на освещение, отопление, охрану, уборку и текущий ремонт помещений. Они принимаются в размере 10 % от стоимости помещения (или его аренды), где происходит разработка программного продукта.

* 1. Расчет цены разработки

Определение цены является одной из труднейших задач, стоящих перед любым предприятием. И именно цена предопределяет успехи предприятия – объемы продаж, доходы, получаемую прибыль.

Установление определенной цены на программный продукт служит для последующей его продажи и получения прибыли. Очень важно назначить цену таким образом, чтобы она не оказалась слишком высокой или слишком низкой.

Для определения минимальной цены, ниже которой разработчику будет невыгодно продавать программный продукт, используется следующая формула:

где – цена программного продукта, руб.; – себестоимость программного продукта, руб.; – норматив прибыли (20 %, в формуле = 0,2).

Результат:

* 1. Расчёт экономической эффективности

В расчет экономической эффективности входит определение эксплуатационных расходов и капитальных затрат потребителя, годовой экономии эксплуатационных расходов у одного потребителя, срока окупаемости программного продукта, годового экономического эффекта.

Расходы потребителя, связанные с эксплуатацией программы, определяются по следующей формуле:

где – эксплуатационные расходы потребителя, руб.; – объем машинного времени в течение года, необходимый для решения данной задачи с использованием программы, ч; – стоимость одного часа машинного времени, руб./ч; – цена программного продукта, руб.; – срок службы программного продукта, год. Обычно составляет 1 – 2 года, затем выпускается новая версия программного продукта.

Результат:

Если на момент внедрения программного продукта у потребителя все работы выполнялись вручную, тогда капитальные затраты рассчитываются по формуле:

где – капитальные расходы потребителя, руб.; – объем машинного времени в течение года, необходимый для решения данной задачи с использованием программы, ч; – полезный годовой фонд времени работы вычислительной техники, принимается условно 2000 ч в год; – капитальные затраты на вычислительную технику, для которой предназначена программа, руб.

Результат:

Капитальные затраты на вычислительную технику рассчитываются по формуле:

где ЦЭВМ – цена вычислительной техники, руб.; Рп.п – прочие расходы потребителя, связанные с помещением (отопление, освещение, уборка и т.д.), принимаются в размере 10 % от стоимости помещения потребителя (или его аренды), руб.

Результат:

Для расчета годовой экономии эксплуатационных расходов потребителя вычисляются эксплуатационные затраты потребителя при решении задачи вручную:

где – эксплуатационные расходы потребителя при решении задачи вручную, руб.; – фонд заработной платы персонала, обслуживающего решение задачи вручную, руб.; 12 – количество месяцев в году; 1,21 – поправочный коэффициент.

Результат:

Тогда годовая экономия эксплуатационных расходов у одного потребителя рассчитывается по формуле

Результат:

Срок окупаемости программного продукта рассчитывается по формуле

Результат:

Годовой экономический эффект, получаемый одним пот­ре­бителем, рассчитывается по формуле

где – нормативный коэффициент эффективности дополнительных капитальных вложений, равный 0,15.

Заключение

В ходе выполнения заданий на преддипломную практику были получены расчетно-пояснительная записка с приложениями для выпускной квалификационной работы, дизайн экранных форм информационной системы, а также основной набор классов, для функционирования программы.

На базе практики были закреплены и получены такие профессиональные компетенции, как способность проводить предпроектное обследование объекта проектирования, системный анализ предметной области, их взаимосвязей (ПК-1), способность проводить техническое проектирование (ПК-2), способность проводить рабочее проектирование (ПК-3), способность проводить выбор исходных данных для проектирования (ПК-4), способность проводить моделирование процессов и систем (ПК-5), способность проводить расчет экономической эффективности (ПК-9), способность разрабатывать, согласовывать и выпускать все виды проектной документации (ПК-10), способность проводить сбор, анализ научно-технической информации, отечественного и зарубежного опыта по тематике исследования (ПК-22), готовность участвовать в постановке и проведении экспериментальных исследований (ПК-23), способность обосновывать правильность выбранной модели, сопоставляя результаты экспериментальных данных и полученных решений (ПК-24).

# Приложение 1

МИНОБРНАУКИ РОССИИ

федеральное государственное бюджетное образовательное учреждение

высшего образования

«ЧЕРЕПОВЕЦКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Институт информационных технологий

Кафедра математического и программного обеспечения ЭВМ

Дисциплина: «Администрирование информационных систем»

|  |
| --- |
| УТВЕРЖДАЮ |
| Зав. кафедрой МПО ЭВМ |
| д.т.н., профессор \_\_\_\_\_\_\_\_ Ершов Е.В. |
| «\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_ 2018 г. |

РАЗРАБОТКА ПОДСИСТЕМЫ УЧАТА РАБОЧЕГО ВРЕМЕНИ И МЕСТОПОЛОЖЕНИЯ СОТРУДНИКОВ АО «СБЕРБАНК-ТЕХНОЛОГИИ»

Техническое задание на курсовую работу

Листов \_\_\_

|  |  |  |
| --- | --- | --- |
|  | Руководитель: | доцент Юдина О.В. |
|  | Исполнитель: | студент гр. 1ИСб-00-41оп |
|  |  | Белов А.С. |

2018 г.

Введение

В обособленном подразделении АО «Сбербанк-Технологии» в г. Череповце (ОП АО «СберТех») установлена система контроля учета доступа (СКУД), предназначенная для автоматического управления входом и выходом сотрудников в офис и внутренние помещения.

СКУД необходимо снабдить дополнительным функционалом, который позволит на основе данных со считывающих устройств пропускных карт составлять подробный отчет о работе сотрудников ОП, тем самым сократив ведение табеля учёта рабочего времени старшим руководителем направления по развитию IT-систем (HR-Администратор).

1. Основания разработки и область применения

Основанием для разработки служит задание на выпускную квалификационную работу, выданное в ОП АО «СберТех» в г.Череповце.

Дата утверждения: 12 февраля 2018 года.

Наименование темы разработки: разработка подсистемы учета рабочего времени и местоположения сотрудников АО «Сбербанк-Технологии».

1. Назначение разработки

Разрабатываемое ПО должно автоматически фиксировать с помощью пластиковых карт точное время прихода и ухода каждого сотрудника, занося информацию в соответствующую таблицу в базе данных (БД).

Помимо автоматической фиксации времени, системе необходимо предоставить сотрудникам возможность самостоятельно оформлять заявки на отпуск, отгулы или командировки и отправлять их на согласование HR-Администратору.

Система должна самостоятельно производить вычисления времени, такие как:

* фактическое время работы сотрудника за период (разница между временем с момента ухода и моментом прихода в офис в течение суток);
* фактическое время переработки за период (разница между фактическим и плановым временем ухода);
* количество дней, которое сотрудник отсутствовал по какой-либо причине за период;
* количество дней накопленного отпуска (из расчета 28 дней, в течении года).

Вся данная информация должна попадать в отчёты за период по одному или нескольким сотрудникам (табл.П1.1. – табл.П1.5).

1. Требования к программе

Подсистема учета рабочего времени и местоположения сотрудников АО «СберТех» должна состоять из двух подпрограмм:

* клиентское приложение, содержащее функционал для удобного ведения данных по сотрудникам. С данной программой будут работать сотрудники отдела.
* серверное приложение, содержащее функционал для работы с пропускными картами и БД. С данной программой будут работать системный администратор (СисАдмин).
  1. Требования к функциональным характеристикам клиентского приложения

С клиентским приложением будет работать HR-Администратор и другие сотрудники ОП АО «СберТех».

Для выполнения должностных обязанностей HR-Администратора ПО должно обладать следующим функционалом:

* возможность создания персонального профиля в программе для каждого сотрудника, в котором будет отражена следующая информация:
  + фамилия, имя, отчество;
  + дата рождения в формате ДД.ММ.ГГГГ;
  + текущая должность;
  + организационная единица;
  + локальное местоположение на территории офиса в формате наименование кабинета;
  + контактная информация (рабочий телефон, мобильный телефон, e-mail адрес);
  + код персонального графика рабочего времени;
  + таблица «Присутствия», содержащая поля, приведенные в табл.П1.6, позволяющая хранить точное время прихода и ухода сотрудника с рабочего места;
  + таблица «Отсутствия», содержащая поля, приведенные в табл.П1.7, позволяющая хранить данные по временным отсутствиям, а также нетрудоспособности сотрудника.
* возможность редактирования и удаления персонального профиля сотрудника;
* возможность создание персонального графика рабочего времени, отражающего следующую информацию:
  + перерыв – время обеденного перерыва;
  + ОГРВ – однодневный график рабочего времени, который отражает суточную часовую норму, которую должен отработать сотрудник (включая время на перерыв);
  + вид графика – 5 через 2 (5/2), 2 через 2 (2/2) или индивидуальный;
  + описание, в котором указывается краткая информация о графике;
* возможность редактирование и удаления графика рабочего времени;
* возможность оформления отсутствий:
  + прогул – беспричинное отсутствие на рабочем месте;
  + отгул – отсутствие на рабочем месте по договоренности с линейным руководителем;
  + больничный - отсутствие на рабочем месте в связи с болезнью;
  + отпуск:
    - основной – за год работы накапливается 28 календарный дней;
    - декрет – в связи с рождением и воспитанием ребенка до 1,5 лет.
  + командировка – отсутствие на рабочем месте в связи с выполнением рабочего задания вне офиса.
* регистрация пропускной карты;
* формирование отчетов:
  + отчет по временным данным у одного или нескольких сотрудников за период;
  + отчет по больничным у одного или нескольких сотрудников за период;
  + отчет по отгулам у одного или нескольких сотрудников за период;
  + отчет по прогулов у одного или нескольких сотрудников за период;
  + отчет по командировкам у одного или нескольких сотрудников за период;
  + отчет по одному или нескольким сотрудникам;



В ПО должны быть предусмотрены роли для разграничения полномочий сотрудников:

* sysAdmin – доступ к настройке системы и мониторингу потока данных;
* HRAdmin – полный доступ к управлению сотрудниками и временными данными;
* wMan – доступ к просмотру собственного профиля и создания заявки на отсутствие;
* возможность создавать свои роли с настраиваемым функционалом.

Роль sysAdmin присваивается системному администратору для возможности настраивать систему и просматривать мониторинг потока данных.

Роль HRAdmin присваивается HR Администратору системным администратором.

Роль Wman должна присваиваться HR Администратором новым сотрудникам.

Таблица П1.6

Описание полей таблицы «Присутствия»

|  |  |
| --- | --- |
| Имя поля | Предполагаемый тип поля |
| Дата | Дата |
| Время с | Время |
| Время до | Время |
| Полный день | Болевая переменная |

Таблица П1.7

Описание полей таблицы «Отсутствия»

|  |  |
| --- | --- |
| Имя поля | Предполагаемый тип поля |
| Дата с | Дата |
| Дата по | Дата |
| Вид отсутствия | Текст |
| Документ | Ссылка на документ |

Для запуска нужного профиля, в системе должен быть предусмотрен процесс авторизации, в котором сотруднику предлагается ввести свой логин и пароль, который генерируется HR-Администратором случайным образом в момент создания профиля сотрудника. Впоследствии, HR-Администратор может изменить пароль у любого сотрудника лично, либо сгенерировать повторно.

После ввода данных, системе необходимо обратиться на сервер к БД, в которой хранятся табельный номер сотрудника, пароль и роль. В зависимости от последнего, программа отображает только те элементы интерфейса, которые доступны для данной роли.

* 1. Требования к функциональным характеристикам серверного приложения

Серверное приложение должно обладать следующим функционалом:

* получение от считывающего устройства пропускных карт данных, в виде: id-адрес и номер датчика считывания;
* на основе полученного id-адреса, внести запись в БД о приходе/уходе и местоположении по конкретному сотруднику;
* при считывании id-адреса выводить информации в реальном времени на экран в виде:

[ДД:ММ:ГГГ] [ЧЧ:ММ:СС] [ID-адрес карты] [ФИО сотрудника] [Местоположение];

* с использованием параметрических запросов реализовать возможность запуска, остановки и паузы получения данных со считывающего устройства;
* с использованием параметрических запросов реализовать возможность выводить информацию по заданному id-адресу на экран в виде:

[ID-адрес карты] [ФИО сотрудника] [Статус Местоположение/«Вне офиса»].

* 1. Требования к надежности

Для безопасного функционирования программ необходимо выполнить следующие требования:

* программа будет работать на рабочих станциях сотрудников, находящихся в локальной сети ОП «СберТех». Поэтому, передача информации от персональных компьютеров (ПК) к серверу будет осуществлена по защищённой локальной сети;
* сервер должен быть подключён к сети через источник бесперебойного питания для возможности сохранения последних транзакций;
* в программе должны быть предусмотрены всевозможные ошибки, текст которых должен выводиться пользователю в доступной форме. Все ошибки должны записываться в log-файл, хранящийся в папке с серверным приложением;
  1. Требования к составу и параметрам технических средств

Для эксплуатации клиентского ПО необходимо иметь:

* персональный компьютер или ноутбук со схожими характеристиками:
  + операционная система Windows 10;
  + процессор Intel Core i3;
  + свободное место на жёстком диске 300 Мбайт;
  + объём оперативной памяти 4 Гбайта;
  + клавиатура;
  + мышь;
  + монитор с разрешением 1280 на 720 пикселей.

Для эксплуатации серверного ПО необходимо иметь:

* сервер со схожими характеристиками:
  + операционная система Windows Server 2016;
  + процессор Intel Xenon E3 1200;
  + свободное место на жёстком диске 100 Мбайт;
  + объём оперативной памяти 4 Гбайта;
  + клавиатура;
  + мышь;
  + монитор с разрешением 600 на 480 пикселей.
* считывающие устройства пропускных карт;
* пропускные пластиковые карты.
  1. Требования к информационной и программной совместимости
     1. Требования к исходным кодам и языкам программирования

Клиентское приложение для сотрудников должно быть разработано с учётом следующих рекомендаций:

* код программы реализован на языке высоко уровня C# с применением объектно-ориентированного подхода;
* интерфейс ПО выполнен с использованием WPF;
* компиляция кода выполнена в среде разработки Visual Studio 17.

Серверное приложение должно быть разработано с учётом следующих рекомендаций:

* код программы реализован на языке высоко уровня C# с применением объектно-ориентированного подхода;
* компиляция кода выполнена в среде разработки Visual Studio 17.
  + 1. Требования к программным средствам, используемым программой

Для успешного функционирования программы на клиенте и сервере должна быть установлена библиотека .NET Framwork 4.5.

* + 1. Требования к защите информации и программ
* пароль сотрудников необходимо храниться в БД в хешированном виде;
* в момент ввода пароля сотрудником в соответствующее поле, каждый символ должен отображаться в виде снежинки «\*» или точки «●»;
  + 1. Специальные требования

Вместе с документацией и диском, содержащим две программы, необходимо предоставить прототип системы считывания пропускных карт с инструкцией по подключению.

1. Требования к программной документации
   1. Структурные требования к расчётно-пояснительной записке

Программная документация содержит расчетно-пояснительную записку, руководство пользователя, наборы тестовых данных и результатов тестирования, спецификации, а также текст программы.

* 1. Требования к оформлению программной документации

При оформлении текстовых и графических материалов, следует придерживаться общих требований ЕСПД.

Текстовые документы оформляются на белых листах формата А4 (210х297 мм). В соответствии с общими требованиями поля листа определяются следующим образом: левое – 30 мм, правое – 10 мм, верхнее – 20 мм, нижнее – 20 мм. Формат текста: Word for Windows, через один интервала, шрифт – Times New Roman Cyr, размер шрифта – 14, отступ первой строки абзаца – 0,75 см, выравнивание – по ширине. Количество знаков в строке, считая пробелы, не должно превышать 60 символов. Текст программы может быть расположен в две колонки, шрифт – Times New Roman Cyr, размер шрифта – 8.

Номер страницы проставляется в середине верхнего поля страницы арабской цифрой. Первая страница – титульный лист, вторая страница – аннотация, с третьей страницы начинается оглавление. Номер страницы на титульном листе не проставляется.

Наименование разделов, подразделов, пунктов должно быть кратким и соответствовать содержанию. Каждая новая глава печатается с новой страницы. Наименование разделов (основных частей) пишется прописными буквами по центру строки. Расстояние между заголовками и текстом, а также между заголовками разделов и подразделов должно быть равно двум интервалам.

Наименования подразделов и пунктов размещаются с абзацного отступа (0,75 см) и печатаются с прописной буквы, без подчеркивания и без точки в конце.

При использовании ссылок на пункты, разделы и подразделы указывается порядковый номер раздела или пункта (например: «в разд. 2», «в п. 2.3.1»).

Рисунки, таблицы и формулы нумеруются арабскими цифрами последовательно, так называемая сквозная нумерация, или в пределах раздела (относительная нумерация). В приложении нумеруются в пределах приложения. Каждый рисунок имеет порядковый номер и название, помещаемые под рисунком по центру.

Таблицы в основном применяются для оформления цифрового материала. Шрифт – Times New Roman Cyr, размер шрифта – 14. Иногда возможен 12-й размер шрифта. Обозначения единиц физических величин необходимо применять в системе СИ. Номер таблицы размещается в правом верхнем углу над заголовком.

5. Стадии и этапы разработки

Стадии и этапы разработки представлены в табл. П1.8.

Таблица П1.8

Стадии и этапы разработки

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Номер этапа разработки | Наименование этапа  разработки ПО | Сроки  Разработки | Результат  выполнения | Отметка о  выполнении |
| 1 | 2 | 3 | 4 | 5 |
| 1 | Разработка технического задания | 12.02.2018 - 16.02.2018 | Документ «Техническое задание» |  |
| 2 | Проведение сравнительного анализа отечественных и зарубежных аналогов проектируемой системы | 19.02.2018 – 7.03.2018 | Пункт 1.1. Сравнительный анализ отечественных и зарубежных аналогов проектируемой системы в РПЗ |  |
| 3 | Выбор технологии, среды и языка программирования | 8.03.2018 – 12.03.2018 | Пункт 1.2. Выбор технологии, среды и языка программирования в РПЗ |  |
| 4 | Анализ процесса обработки информации, выбор структур данных для ее хранения, выбор методов и алгоритмов решения задачи | 13.03.2018 – 20.03.2018 | Пункт 1.3. Анализ процесса обработки информации, выбор структур данных для ее хранения, выбор методов и алгоритмов решения задачи в РПЗ |  |

Таблица П1.8. Продолжить

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 |
| 5 | Разработка спецификаций проектируемой системы | 21.03.2018 – 08.04.2018 | Пункт 1.4. Разработка спецификаций проектируемой системы в РПЗ |  |
| 6 | Проектирование системы | 09.04.2018 – 26.04.2018 | Пункт 1.5. Проектирование системы в РПЗ |  |
| 7 | Проектирование интерфейса пользователя для серверного приложения | 27.04.2018 – 28.04.2018 | Пункт 1.6. Проектирование интерфейса пользователя в РПЗ |  |
| 8 | Проектирование интерфейса пользователя для клиентского приложения | 28.04.2018 – 30.04.2018 | Пункт 1.6. Проектирование интерфейса пользователя в РПЗ |  |
| 9 | Программирование микроконтроллера Arduino | 01.05.2018 – 02.05.2018 | Считывание датчиком пропускных карт RFID-код, хранимый в пропуске |  |
| 10 | Разработка серверного приложения | 03.05.2018 – 13.05.2018 | Работоспособное серверное приложение |  |
| 11 | Разработка клиентского приложения | 14.05.2018 – 03.06.2018 | Работоспособное клиентское приложение |  |

Таблица П1.8. Продолжить

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 |
| 12 | Выбор стратегии тестирования, разработка тестов, прохождение тестов | 11.06.2018 | Пункт 1.7. Выбор технологии, среды и языка программирования в РПЗ |  |
| 13 | Написание руководство пользователя | 15.06.2018 | Приложение 4. Руководство пользователя |  |

6. Порядок контроля и приемки

Порядок контроля и приёма работ представлен в табл.П1.9.

Таблица П1.9

Порядок контроля и приемки

|  |  |  |  |
| --- | --- | --- | --- |
| Наименование  контрольного этапа  выполнения  курсовой работы | Сроки  контроля | Результат выполнения | Отметка о приемке  результата  контрольного этапа |
| Выдача задания | 12.02.18 | Получено задание на выполнение курсового проекта |  |
| Техническое задание | 14.03.18 | Документ «Техническое задание» |  |
| Расчётно-пояснительная записка | 08.06.18 | Документ «Расчётно-пояснительная записка» |  |
| Приложения к расчётно-пояснительной записке | 15.06.2018 | Документы «Приложение 2»,  «Приложение 3», «Приложение 4», «Приложение 5», «Приложение 6» |  |
| Защита курсовой работы | 21.06.18 | Защита курсовой работы |  |

# Приложение 2

Код программы

Form1.cs

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Text;

using System.Windows.Forms;

using System.Data;

using System.Data.OleDb;

using System.IO;

namespace HRSaveTime\_Server

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

BDConnect con = new BDConnect();

SerialPorts sp = new SerialPorts();

public static SortedList<string, string> setting = new SortedList<string, string>();

List<string> list = new List<string>();

public void GetSetting()

{

StreamReader sr = new StreamReader("settings.txt");

try

{

while (!sr.EndOfStream)

{

list.Add(sr.ReadLine());

}

sr.Close();

if (list.Count != 0)

{

for (int i = 0; i < list.Count; i = +2)

{

setting.Add(list[i], list[i + 1]);

}

}

}

catch (Exception ex)

{

MessageBox.Show(ex.Message.ToString(), "Ошибка");

}

}

public void SetSetting(string key, string value)

{

if (!setting.ContainsKey(key))

{

setting.Add(key, value);

}

else

{

setting.Remove(key);

setting.Add(key, value);

}

StreamWriter sw = new StreamWriter("settings.txt");

try

{

foreach (string k in setting.Keys)

{

foreach (string v in setting.Values)

{

sw.WriteLine(k);

sw.WriteLine(v);

}

}

sw.Close();

}

catch (Exception ex)

{

MessageBox.Show(ex.Message.ToString(), "Ошибка");

}

}

public void GetLocationCB()

{

if (textBox5.Text != "")

{

List<string> mas = new List<string>();

mas = con.SetLocationName(textBox5.Text);

comboBox1.Items.Clear();

foreach (string m in mas)

{

comboBox1.Items.Add(m);

}

}

}

private void Form1\_Load(object sender, EventArgs e)

{

GetSetting();

var result = "";

setting.TryGetValue("BD", out result);

if (result != null)

{

textBox5.Text = result;

string value = con.ConnectStatus(result);

if (value == "OK")

{

label9.ForeColor = Color.Green;

label9.Text = "Подключено";

GetLocationCB();

}

else

{

MessageBox.Show(value, "Ошибка", MessageBoxButtons.OK, MessageBoxIcon.Error);

label9.ForeColor = Color.Red;

label9.Text = "Ошибка";

return;

}

}

string[] val = sp.SearchPorts();

foreach (string v in val)

{

comboBox2.Items.Add(v);

comboBox3.Items.Add(v);

}

}

private void button8\_Click(object sender, EventArgs e)

{

if (textBox5.Text != "")

{

string Value = con.ConnectStatus(textBox5.Text);

if (Value == "OK")

{

label9.ForeColor = Color.Green;

label9.Text = "Подключено";

SetSetting("BD", textBox5.Text);

GetLocationCB();

}

else

{

label9.ForeColor = Color.Red;

label9.Text = "Ошибка";

MessageBox.Show(Value, "Ошибка", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

else

{

MessageBox.Show("Необходимо добавить базу данных в систему.", "Внимание", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

}

private void button9\_Click(object sender, EventArgs e)

{

string Result = con.AddDataBase();

if (Result != "")

{

textBox5.Text = Result;

}

}

private void button5\_Click(object sender, EventArgs e)

{

SettingsRoom sR = new SettingsRoom();

sR.ShowDialog();

GetLocationCB();

}

}

}

SerialPorts.cs

using System;

using System.Collections.Generic;

using System.Text;

using System.IO.Ports;

namespace HRSaveTime\_Server

{

class SerialPorts

{

public String[] SearchPorts()

{

string[] ports = SerialPort.GetPortNames();

return ports;

}

}

}

SettingsRoom.cs

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Text;

using System.Windows.Forms;

using System.Data.OleDb;

namespace HRSaveTime\_Server

{

public partial class SettingsRoom : Form

{

public SettingsRoom()

{

InitializeComponent();

}

BDConnect bd = new BDConnect();

bool Edit = false;

bool newRow = false;

List<int> editIndex = new List<int>();

List<int> newIndex = new List<int>();

private void SettingsRoom\_Load(object sender, EventArgs e)

{

var result = "";

Form1.setting.TryGetValue("BD", out result);

List<string> sss = bd.SetLocationIDNAame(result);

int j = 0;

for (int i = 0; i < sss.Count / 2; i++)

{

dataGridView1.Rows.Add();

dataGridView1.Rows[i].Cells[0].Value = sss[j];

dataGridView1.Rows[i].Cells[1].Value = sss[j + 1];

j += 2;

}

}

private void SettingsRoom\_FormClosing(object sender, FormClosingEventArgs e)

{

var result = "";

Form1.setting.TryGetValue("BD", out result);

if (Edit || newRow)

{

DialogResult dr = MessageBox.Show("Вы хотите сохранить изменения?", "Внимание", MessageBoxButtons.YesNo, MessageBoxIcon.Information);

if (dr == DialogResult.Yes)

{

if (Edit)

{

foreach (int i in editIndex)

{

bd.SaveEditBDLocation(result, dataGridView1.Rows[i].Cells[1].Value.ToString(), dataGridView1.Rows[i].Cells[0].Value.ToString());

}

}

if (newRow)

{

foreach (int i in newIndex)

{

bd.SaveNewBDLocation(result, dataGridView1.Rows[i - 1].Cells[1].Value.ToString(), dataGridView1.Rows[i - 1].Cells[0].Value.ToString());

}

}

}

else if (dr == DialogResult.No)

{

this.Close();

}

}

}

private void добавитьToolStripMenuItem\_Click(object sender, EventArgs e)

{

dataGridView1.Rows.Add();

dataGridView1.Rows[dataGridView1.RowCount - 1].Cells[1].ReadOnly = false;

dataGridView1.Rows[dataGridView1.RowCount - 1].Cells[0].Value = Convert.ToInt32(dataGridView1.Rows[dataGridView1.RowCount - 2].Cells[0].Value) + 1;

dataGridView1.CurrentCell = dataGridView1.Rows[dataGridView1.RowCount - 1].Cells[1];

dataGridView1.BeginEdit(true);

newRow = true;

newIndex.Add(dataGridView1.RowCount);

}

private void dataGridView1\_CellClick(object sender, DataGridViewCellEventArgs e)

{

if (e.ColumnIndex == 2)

{

dataGridView1.Rows[e.RowIndex].Cells[1].ReadOnly = false;

dataGridView1.CurrentCell = dataGridView1.Rows[e.RowIndex].Cells[1];

dataGridView1.BeginEdit(true);

Edit = true;

editIndex.Add(e.RowIndex);

}

}

}

}

HRSaveTimeClient

MainWindow.xaml.cs

using System;

using System.Collections.Generic;

using System.Text;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Data;

using System.Windows.Documents;

using System.Windows.Input;

using System.Windows.Media;

using System.Windows.Media.Imaging;

using System.Windows.Navigation;

using System.Windows.Shapes;

namespace HRSaveTimeClient

{

/// <summary>

/// Логика взаимодействия для MainWindow.xaml

/// </summary>

public partial class MainWindow : Window

{

public MainWindow()

{

InitializeComponent();

}

private void Window\_MouseLeftButtonDown(object sender, MouseButtonEventArgs e)

{

this.DragMove();

}

private void TextBlock\_PreviewMouseDown(object sender, MouseButtonEventArgs e)

{

MessageBox.Show("Если вы забыли пароль, то обратитесь к вашему HR Администратору для его сброса!", "Инструкция", MessageBoxButton.OK, MessageBoxImage.Information);

}

private void Image\_PreviewMouseDown(object sender, MouseButtonEventArgs e)

{

this.Close();

}

private void Button\_Click(object sender, RoutedEventArgs e)

{

GlobalForm gf = new GlobalForm();

this.Visibility = Visibility.Hidden;

gf.ShowDialog();

this.Visibility = Visibility.Visible;

}

private void Image\_MouseEnter(object sender, MouseEventArgs e)

{

ExitApp.Width = 20;

ExitApp.Height = 20;

}

private void Image\_MouseLeave(object sender, MouseEventArgs e)

{

ExitApp.Width = 15;

ExitApp.Height = 15;

}

private void TextBlock\_MouseEnter(object sender, MouseEventArgs e)

{

RememberPasswordButton.TextDecorations = TextDecorations.Underline;

}

private void TextBlock\_MouseLeave(object sender, MouseEventArgs e)

{

RememberPasswordButton.TextDecorations = null;

}

}

}

MoreORGForm.xaml.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Data;

using System.Windows.Documents;

using System.Windows.Input;

using System.Windows.Media;

using System.Windows.Media.Imaging;

using System.Windows.Shapes;

namespace HRSaveTimeClient

{

/// <summary>

/// Логика взаимодействия для MoreORGForm.xaml

/// </summary>

public partial class MoreORGForm : Window

{

public MoreORGForm()

{

InitializeComponent();

}

BrushConverter bc = new BrushConverter();

private void DataGrid\_MouseDown(object sender, MouseButtonEventArgs e)

{

this.DragMove();

}

private void OkMoreORGButton\_MouseEnter(object sender, MouseEventArgs e)

{

OkMoreORGButton.Background = (Brush)bc.ConvertFrom("#2c71fd");

}

private void OkMoreORGButton\_MouseLeave(object sender, MouseEventArgs e)

{

OkMoreORGButton.Background = (Brush)bc.ConvertFrom("#0049db");

}

private void OkMoreORGButton\_PreviewMouseLeftButtonDown(object sender, MouseEventArgs e)

{

this.Close();

}

private void CencelMoreORGButton\_MouseEnter(object sender, MouseEventArgs e)

{

CencelMoreORGButton.Background = (Brush)bc.ConvertFrom("#cbcaca");

}

private void CencelMoreORGButton\_MouseLeave(object sender, MouseEventArgs e)

{

CencelMoreORGButton.Background = (Brush)bc.ConvertFrom("#8d8d8d");

}

private void CencelMoreORGButton\_PreviewMouseLeftButtonDown(object sender, MouseEventArgs e)

{

this.Close();

}

}

}

MorePernrForm.xaml.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Data;

using System.Windows.Documents;

using System.Windows.Input;

using System.Windows.Media;

using System.Windows.Media.Imaging;

using System.Windows.Shapes;

namespace HRSaveTimeClient

{

/// <summary>

/// Логика взаимодействия для MorePernrForm.xaml

/// </summary>

public partial class MorePernrForm : Window

{

public MorePernrForm()

{

InitializeComponent();

}

BrushConverter bc = new BrushConverter();

private void DataGrid\_MouseDown(object sender, MouseButtonEventArgs e)

{

this.DragMove();

}

private void OkMorePernrReportsButton\_MouseEnter(object sender, MouseEventArgs e)

{

OkMorePernrReportsButton.Background = (Brush)bc.ConvertFrom("#2c71fd");

}

private void OkMorePernrReportsButton\_MouseLeave(object sender, MouseEventArgs e)

{

OkMorePernrReportsButton.Background = (Brush)bc.ConvertFrom("#0049db");

}

private void OkMorePernrReportsButton\_PreviewMouseLeftButtonDown(object sender, MouseButtonEventArgs e)

{

this.Close();

}

private void CencelMorePernrReportsButton\_MouseEnter(object sender, MouseEventArgs e)

{

CencelMorePernrReportsButton.Background = (Brush)bc.ConvertFrom("#cbcaca");

}

private void CencelMorePernrReportsButton\_MouseLeave(object sender, MouseEventArgs e)

{

CencelMorePernrReportsButton.Background = (Brush)bc.ConvertFrom("#8d8d8d");

}

private void CencelMorePernrReportsButton\_PreviewMouseLeftButtonDown(object sender, MouseButtonEventArgs e)

{

this.Close();

}

}

}

# Приложение 3

Тестирование программы

HRSaveTimeSever

При запуске программы открывается окно с «Лог» - монитор потока данных, которые поступают в систему со считывающих устройств пропускных карт.

Пример считывание на рис.П3.1.

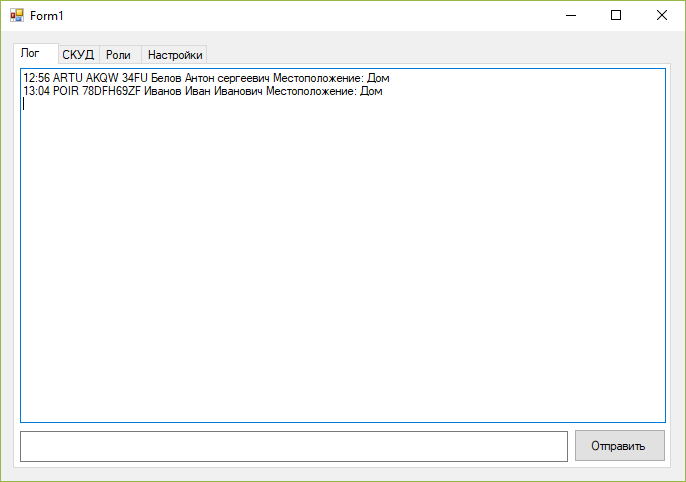


Рис.П3.1. Монитор потока данных

В данном окне предусмотрена возможность ввода параметрических запросов и команд. Например, getUser – выводит информацию о сотруднике по RFID-номеру пропуска (рис.П3.2). Результат представлен на рис.П3.3.

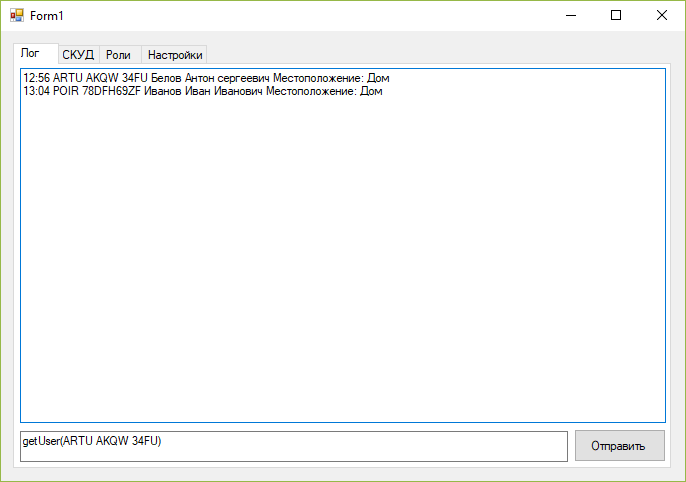


Рис.П3.2. Пример запроса

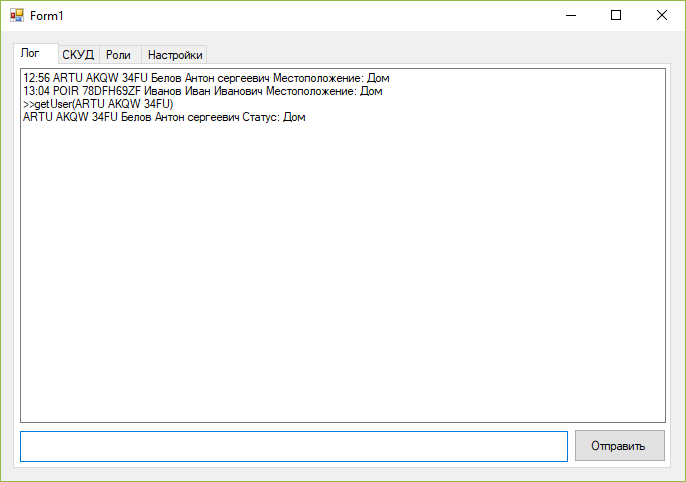


Рис.П3.3. Результат выполнения запроса

Во вкладке СКУД предусмотрена возможность настраивать комнату и соответствующие датчики (рис.П3.4).

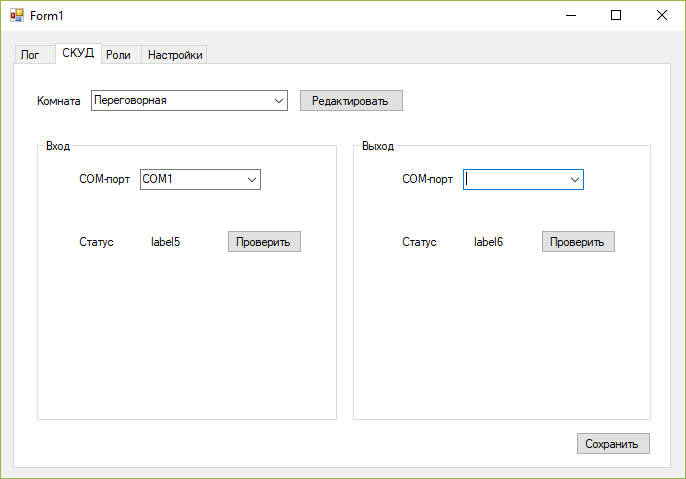


Рис.П3.4. Настройка СКУД

Для добавления или изменения помещений жмём «Редактировать». В открывшемся окне (Рис.П3.5) предусмотрена возможность редактирования записей (Рис.П3.6), добавлять новые (Рис.П3.7), либо удалять, нажав соответствующую кнопку.

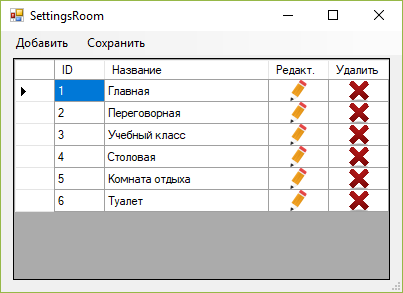


Рис.П3.5. Окно ведения помещений

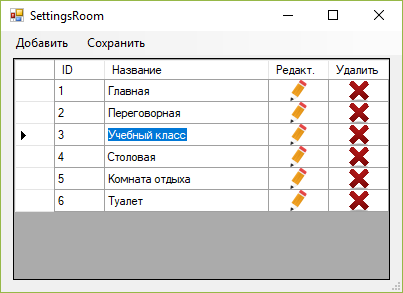


Рис.П3.6.Нажатие кнопки «Редактировать»

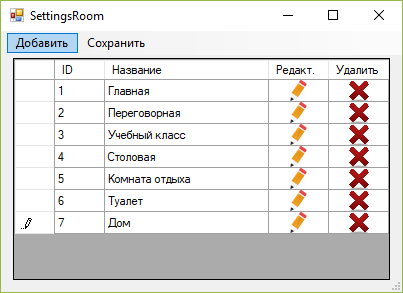


Рис.П3.7.Нажатие кнопки «Добавить»

В вкладке «Роли» можно добавить или изменить существующие роли в системе (рис.П3.8).

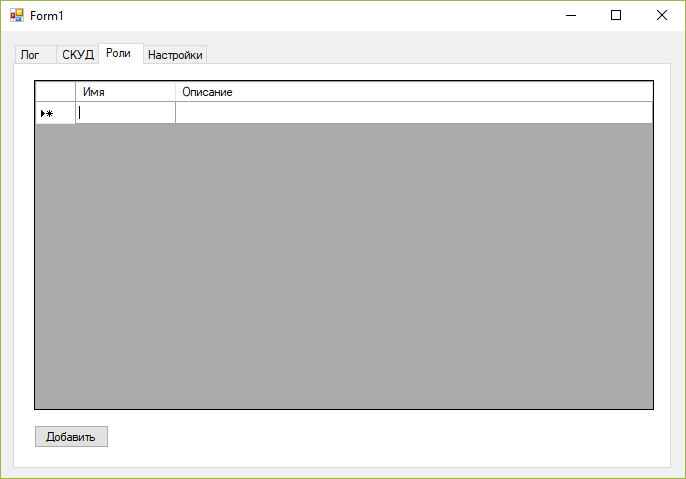


Рис.П3.8.Окно ведения ролей

В вкладке «Настройки» можно подключить БД, атак же проверить статус подключения (рис.П3.9).

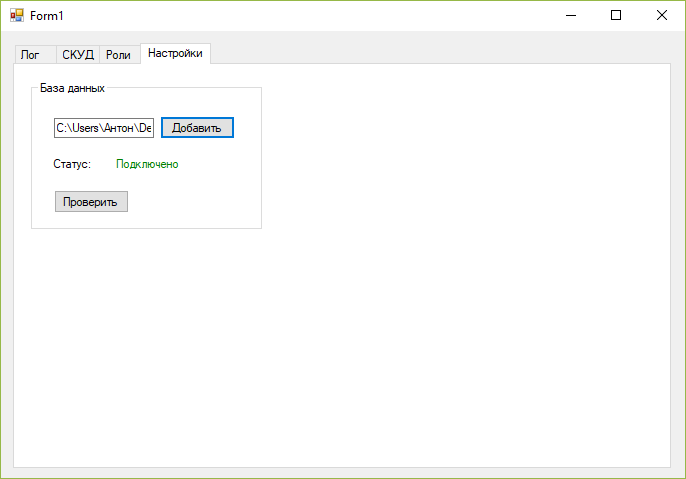


Рис.П3.9.Окно «Настройки»

HRSaveTimeClient

HRSaveTimeClient – клиент-приложение, которое в первую очередь служит для отображения информации, хранящейся в БД.

Примеры окон данной программы представлены на рис.П3.10 – П3.17.

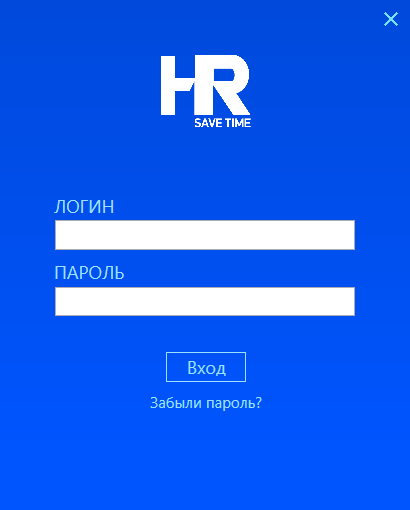


Рис.П3.10.Окно авторизации

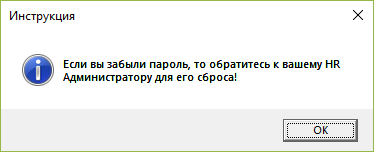


Рис.П3.11.Окно инструкции, при нажатии «Забыли пароль?»

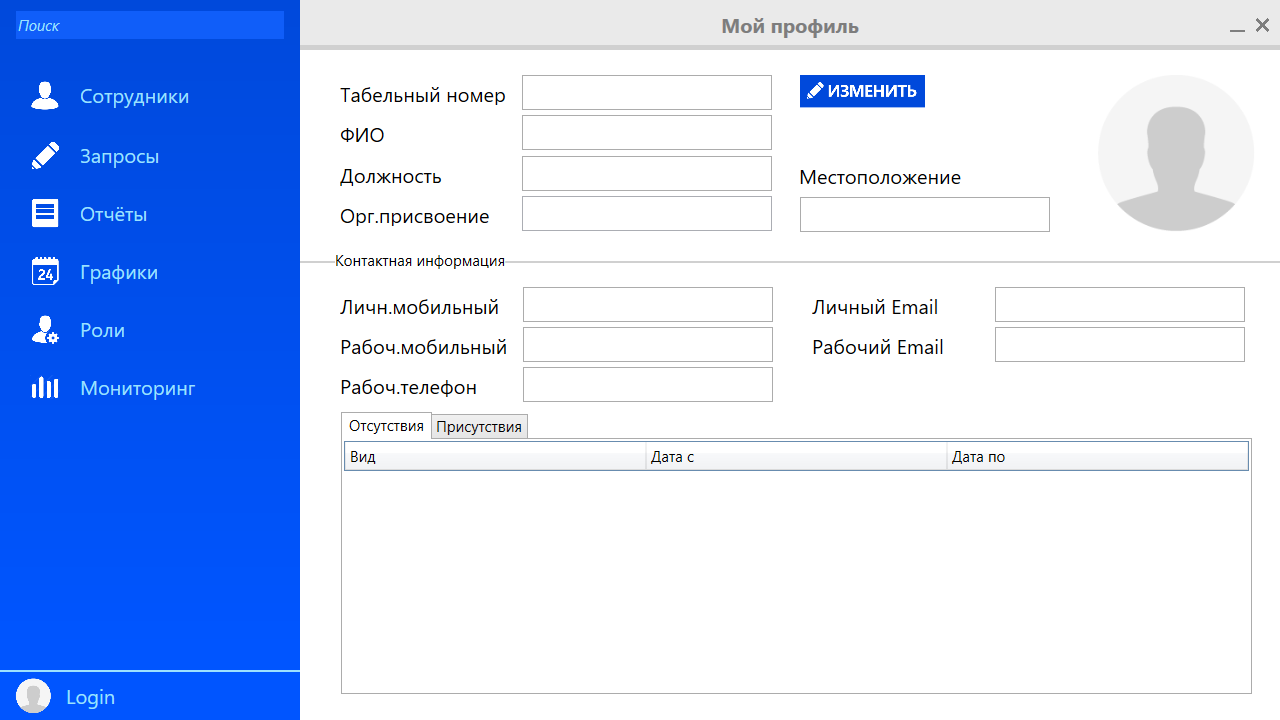


Рис.П3.12.Окно «Мой профиль»

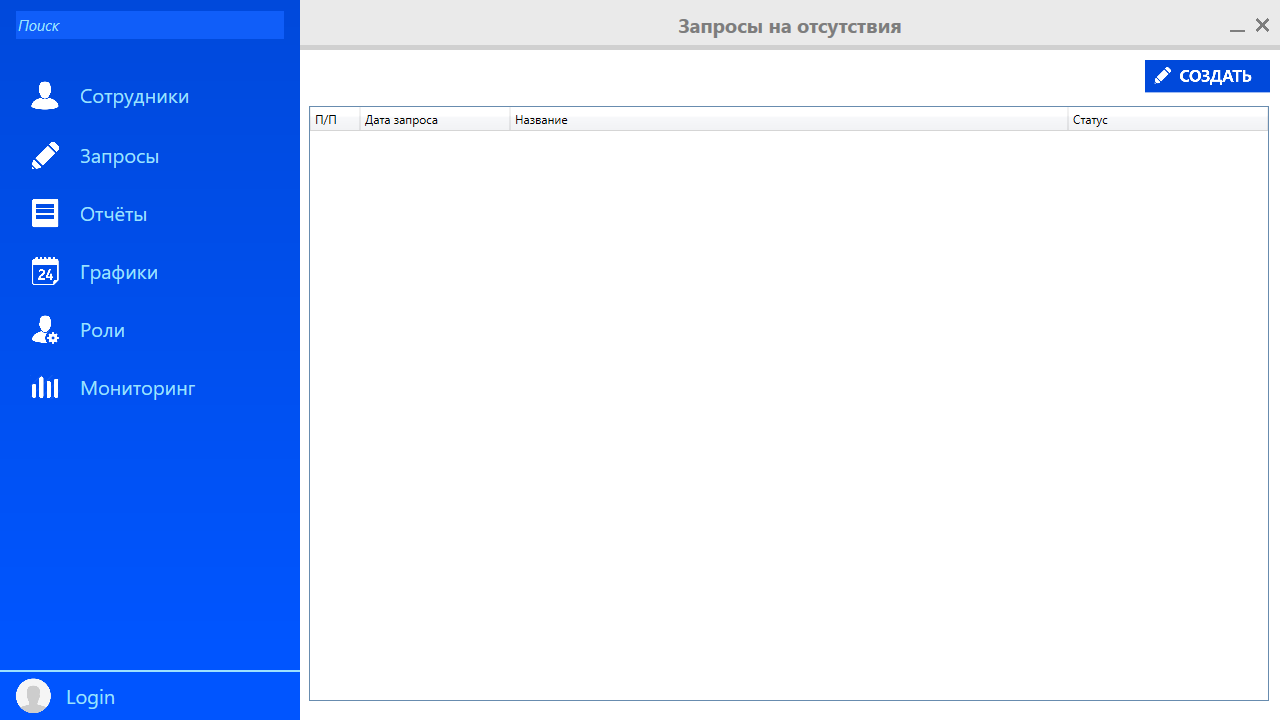


Рис.П3.13.Окно «Запросы на отсутствие»

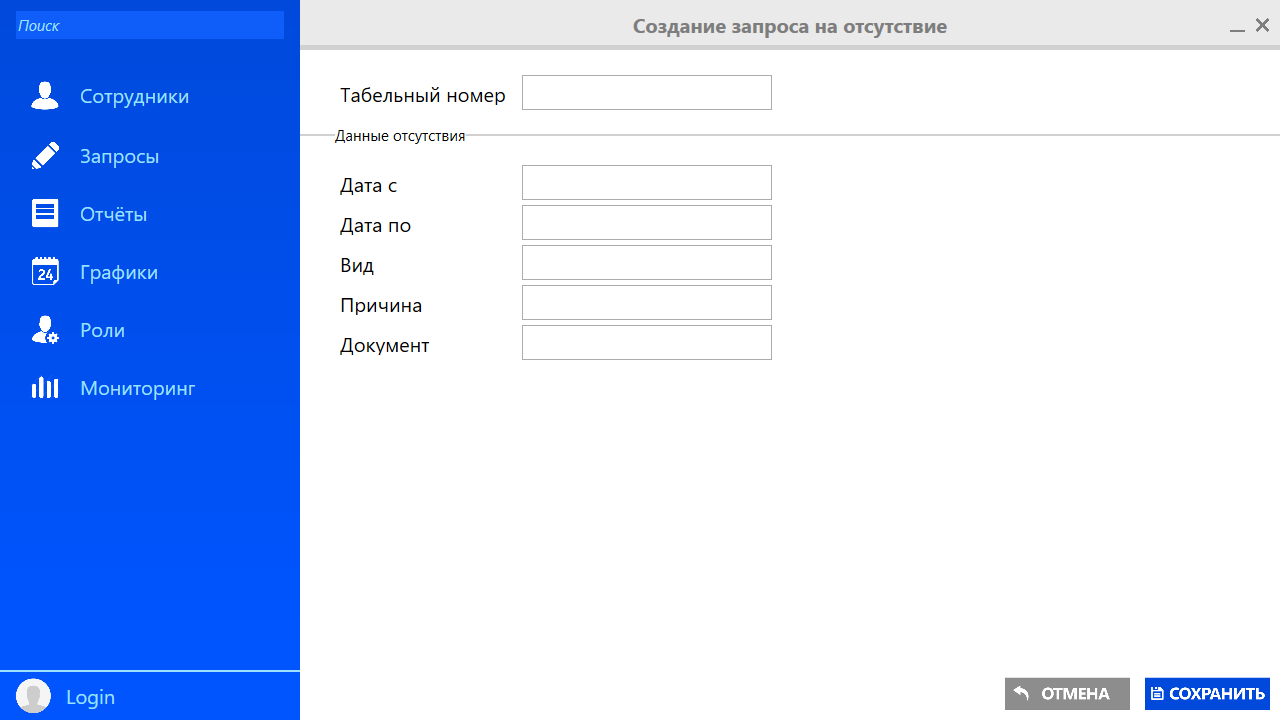


Рис.П3.14.Окно «Создание запроса на отсутствие»

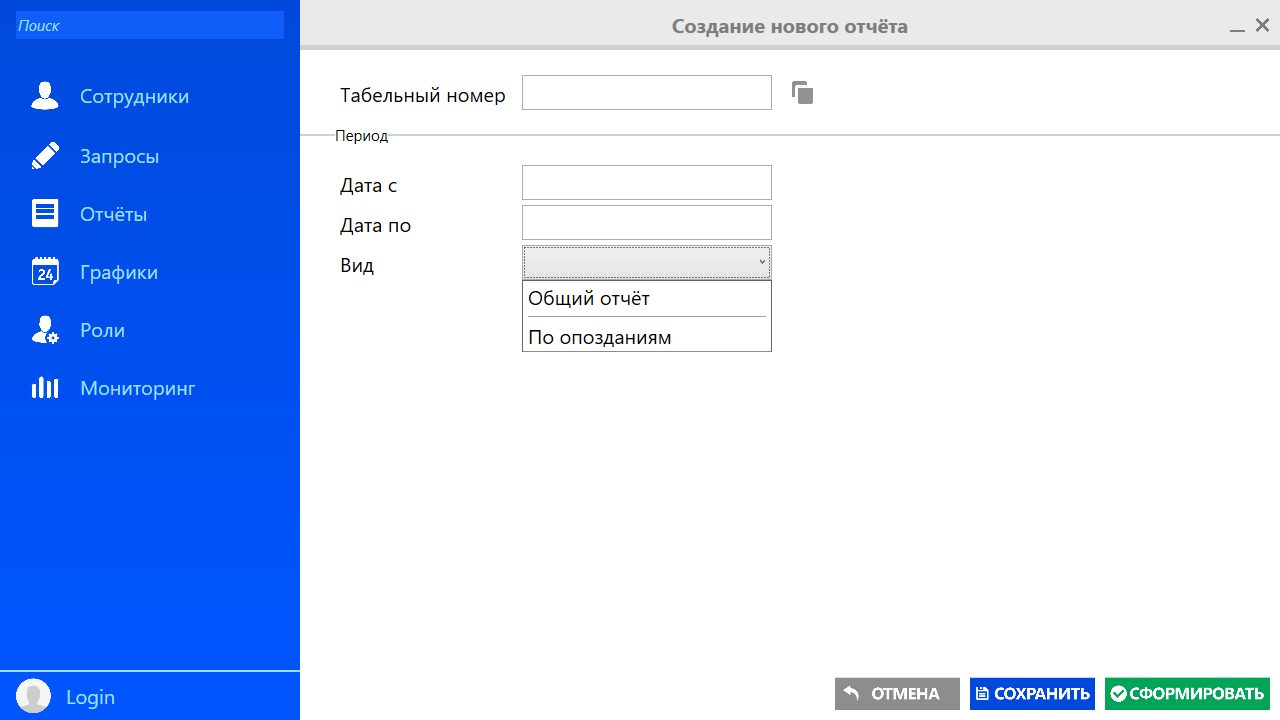


Рис.П3.15.Окно «Создание нового ответа»

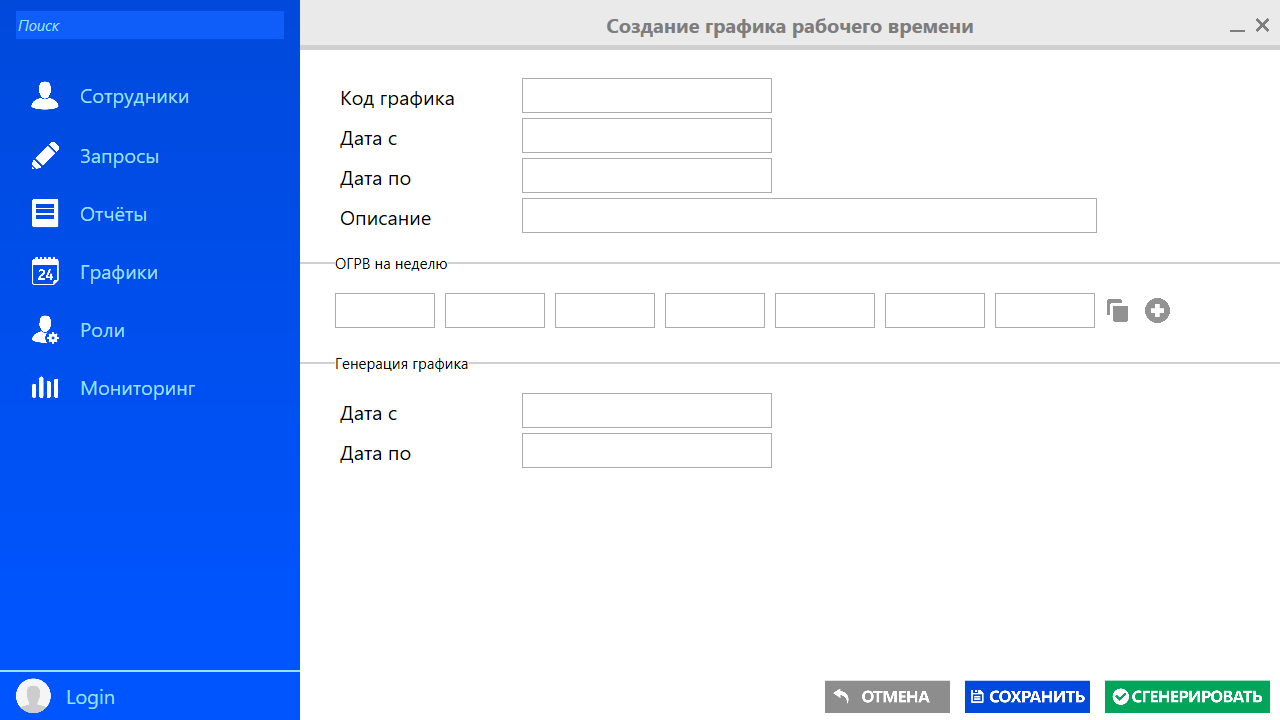


Рис.П3.16.Окно «Создание ГРВ»

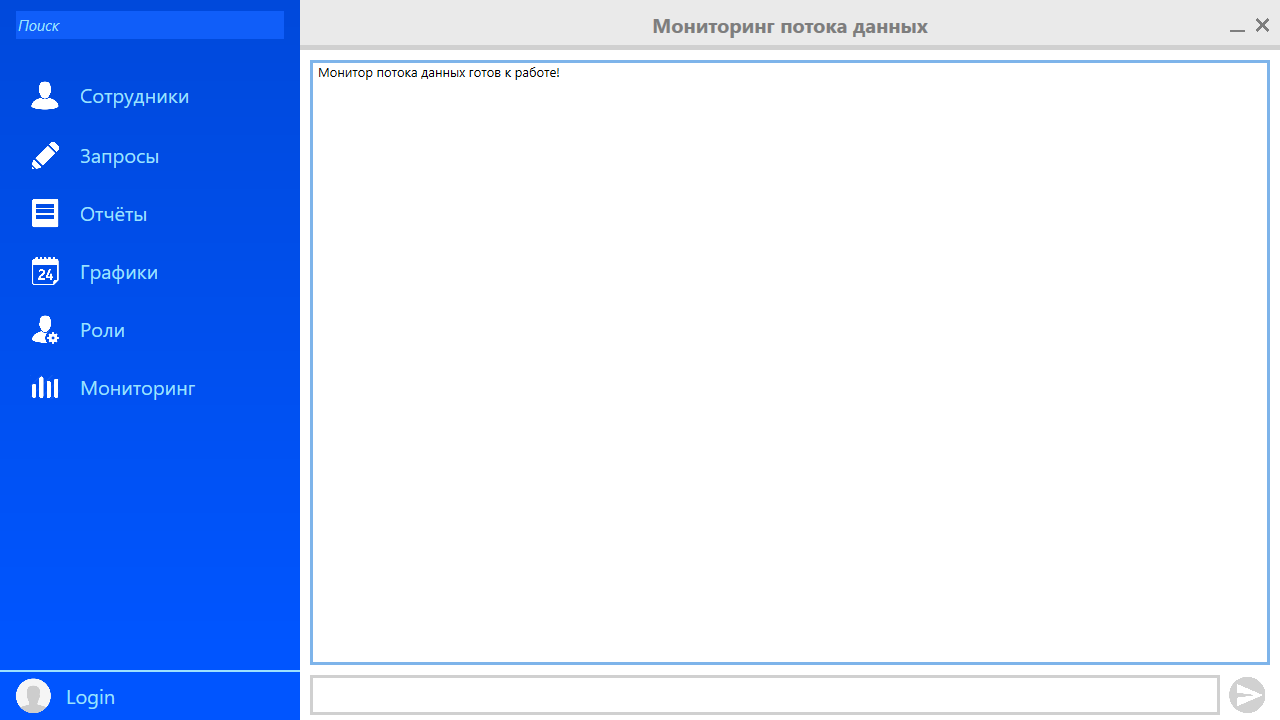


Рис.П3.17.Окно «Мониторинга потока данных»