Сабельников ПКС-41

User\_010

Модуль 1

Была разработана краткая спецификация разрабатываемого модуля:

|  |  |
| --- | --- |
| Обозначение | Назначение |
| GetAllInvoices | Возможность поулченяи спсика всех созданных заявок |
| AddInvoice | Возможность добавления новой заявки |
| EditInvoice | Возможность редактирования заявки |
| SetExecutorToInvoice | Возможность назщначеняи исполнителя к заявке |
| AcceptInvoiceToWork | Возможность принятия заявки в рабоут исполнителем |
| CompleteInvoice | Возможность завершения заявки |
| GetStatistics | Возомжность полученяи статистики |
| Authorizeuser | Возможнотсь авторизации пользователя по логину и пароли |
|  |  |
|  |  |

Были определены входные данные:

информация о пользователе:

- Логин;

- Проль;

Информация о создаваемой заявке:

- Номер заявки;

- Дата добавления;

- Оборудование, которое требует ремонта;

- Тип неисправности;

- Описание проблемы;

- Клиент, который подал заявку;

- Статус заявки (в ожидании, в работе, выполнено).

Инфомрация о редактируемой заявке:

- Изменение этапа выполнения (выполнено, в работе, не выполнено);

- Изменение описания проблемы;

- Изменение, ответственного за выполнение работ.

Были определены выходные данные:

Список всех заявок, включающий в себя следующую информацию:

- Номер заявки;

- Дата добавления;

- Оборудование, которое требует ремонта;

- Тип неисправности;

- Описание проблемы;

- Клиент, который подал заявку;

- Статус заявки (в ожидании, в работе, выполнено).

Статистическая инфомрцяи:

- Количество выполненных заявок;

- Среднее время выполнения заявки;

- Статистика по типам неисправностей.

В формате текстового описания был сформирован основной алгоритм решения учета заявок на ремонт оборудования:

При запуске программы учета заявок пользователь проходит авторизацию, вводя логин и пароль. При нажатии кнопки «Войти» ведется поиск учетной записи с такими данными, и если она найдена, то узнается роль данного пользователя. Если пользователь обладает ролью «Исполнитель», то осуществляется переход на страницу исполнения заявки, на которой пользователь может принять заявку в работу, а также завершить заявку и оставить комментарий.

Если пользователь обладает ролью «Пользователь», то осуществляется переход на страницу главного меню, с которой пользователь может перпейти на старницу создания заявки, редактирования заявки, страницу списка заявок страницу статистики или выйти из приложения. На старнцие создания заявки полдьзователь может создать заявку с указанием следующей итнформации:

- Номер заявки;

- Дата добавления;

- Оборудование, которое требует ремонта;

- Тип неисправности;

- Описание проблемы;

- Клиент, который подал заявку;

- Статус заявки (в ожидании, в работе, выполнено).

После нажатия кнопки добавленя заявки, заявка добавляется в базу данных и она становится доступной для редактирования.

На странице редактирования заявки пользователь может отредактировать заявку с указанием следующей информации:

- Изменение этапа выполнения (выполнено, в работе, не выполнено);

- Изменение описания проблемы;

- Изменение, ответственного за выполнение работ.

При назначении исполнителя на заявке, данная заявка становится доступной для обработки у исполнителя, который может принять ее в работу, а затем завершить ее.

На странцие списка заявок пользователь может получить информацию о всех заявках в базе данных. При переходже на данную страницу происходжит запрос к базе данных, полученная инфомрация из которого отображается в виде таблицы.

На странице статистики пользователь может получить статистическую информацию, включающую в себя:

- Количество выполненных заявок;

- Среднее время выполнения заявки;

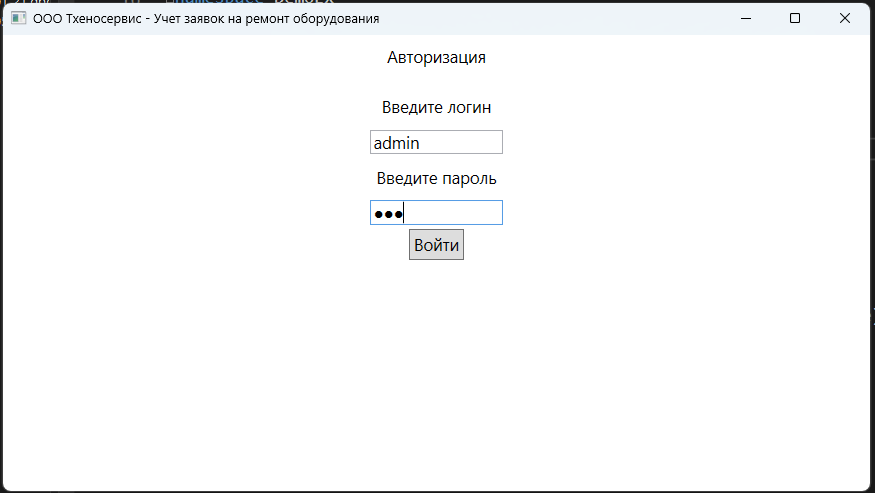
- Статистика по типам неисправностей.

В формате текстового описания был детализирован алгоритм функции расчета вкол-ва выполненных заявок:

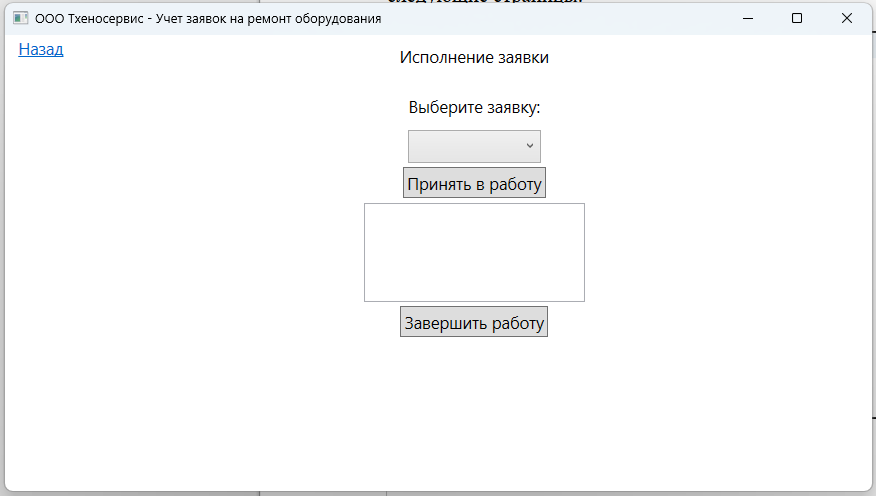
При переходе на страницу статистики происходит запрос к базе данных с использованием стандартной функции SQL, которая подсчитывает кол-во записей в таблице заявок, удовлетворяющих условию «Статус = Выполнено». Полученное кол-во отображается в строке «Кол-во выполнгенных заявок».

Был разработан пользовательский интерфейс, включающий в себя следующие страницы:

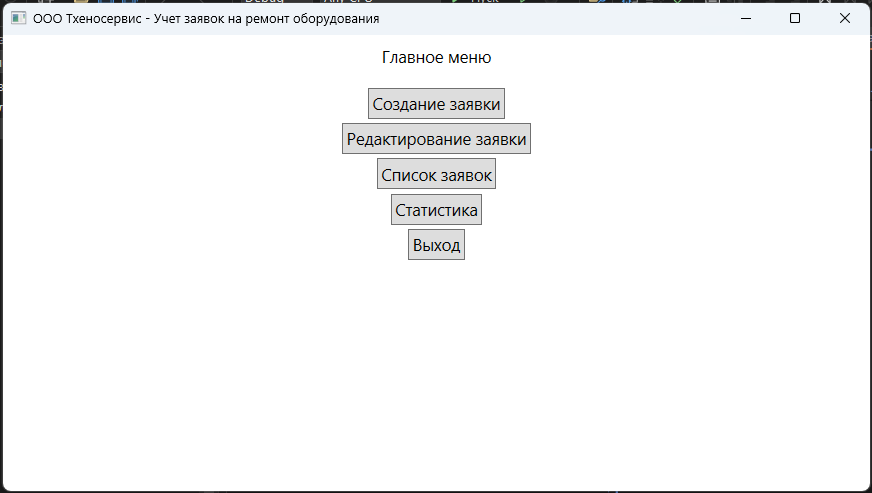
Страница авторизации:



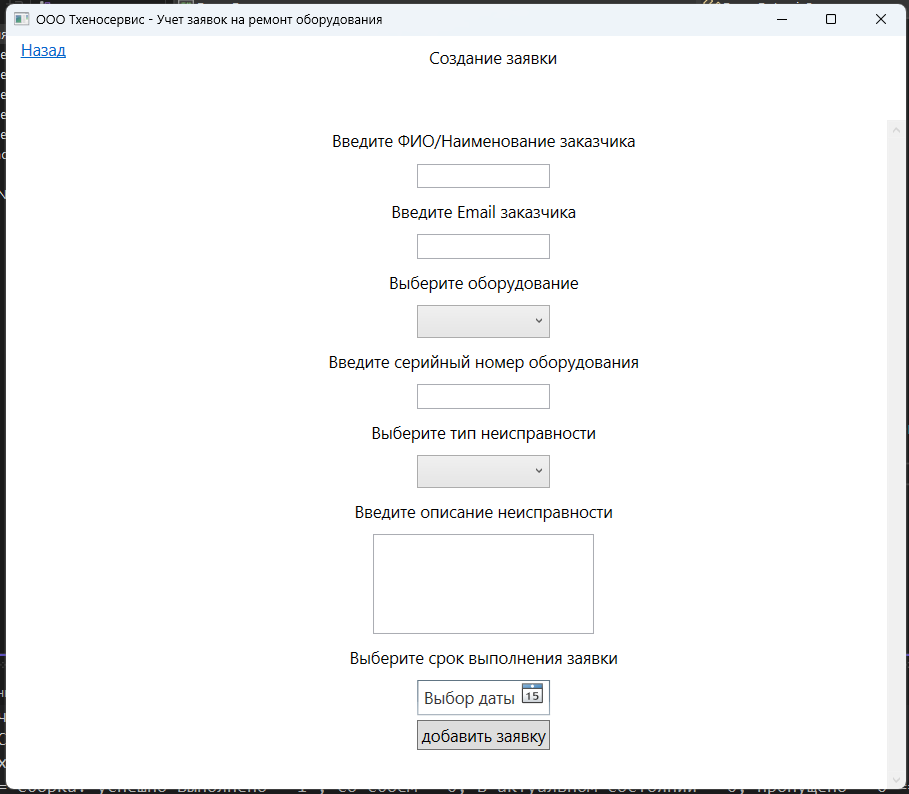
Страница исполнения заявки:



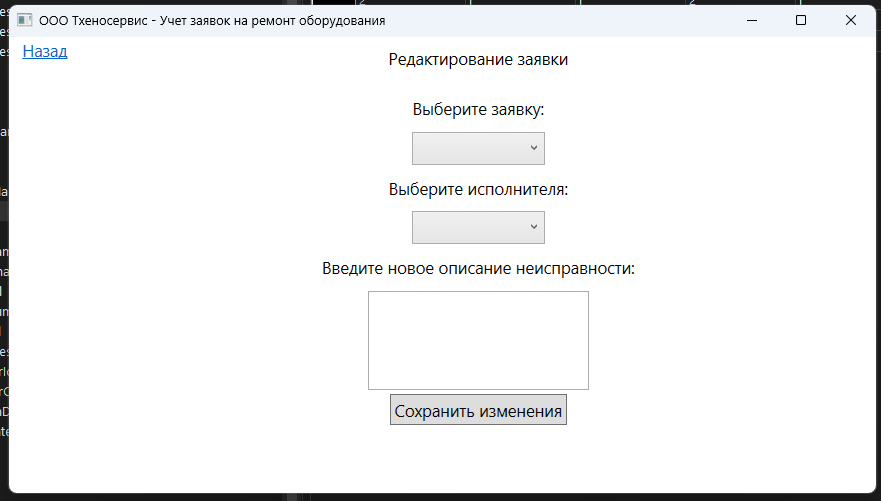
Страница главного меню:



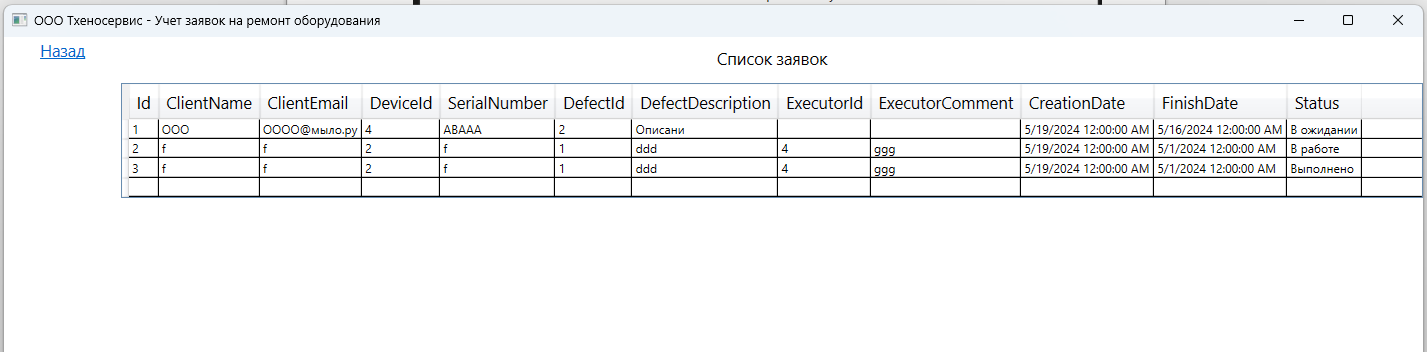
Страница создания заявки:



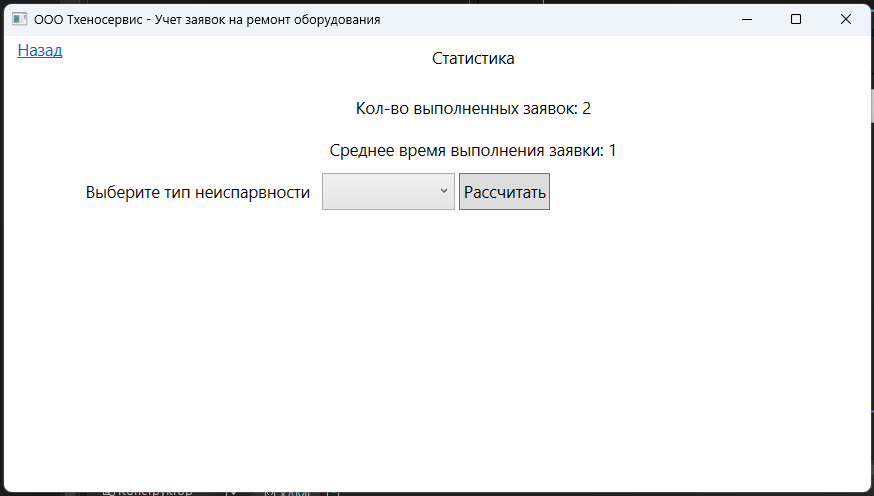
Страница редактирования заявки:



Страница просмотра списка заявок:

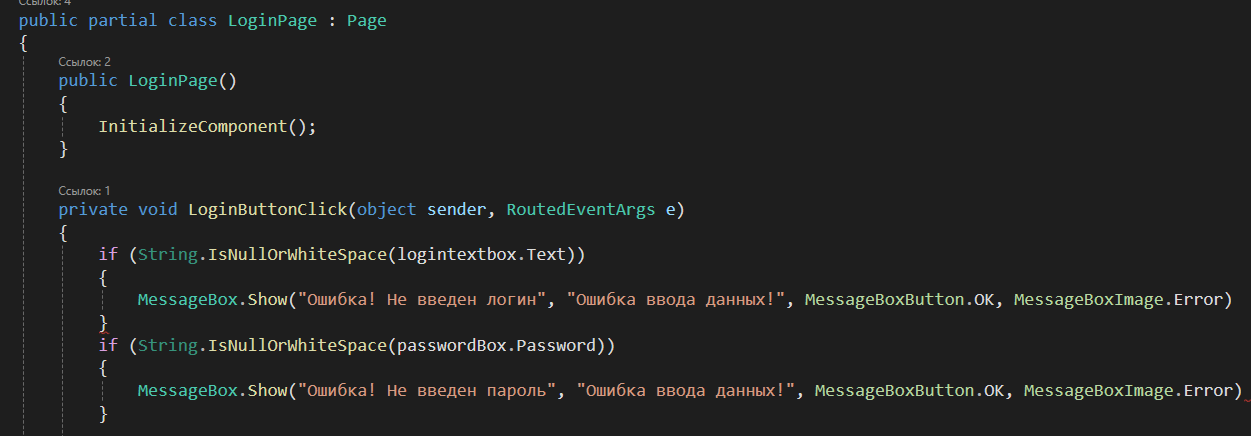


Страница статистики:

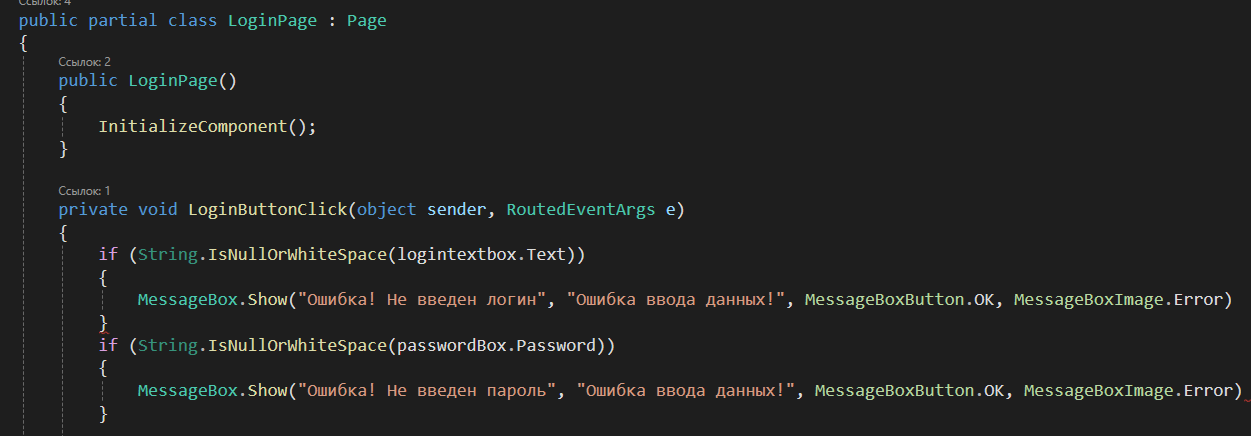


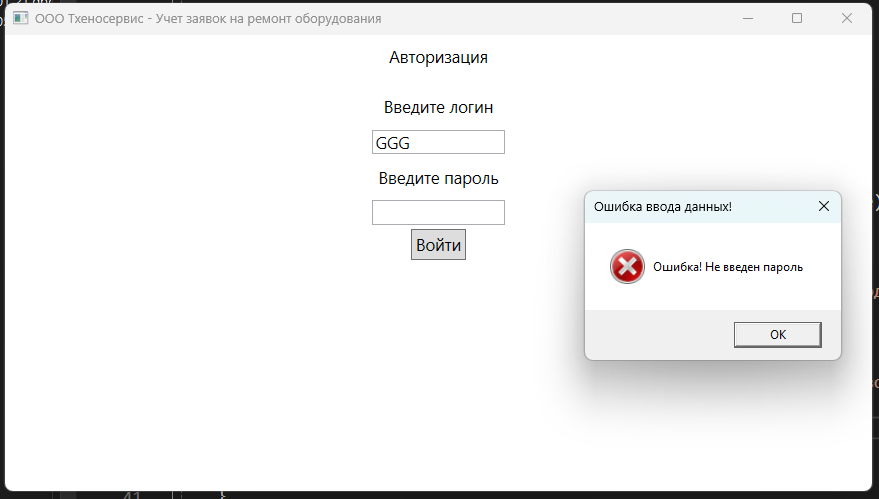
На всех разработанных страницах были созданы заголовки, описывающие функционал данных страниц. Также были реализованы переходы между страницами с помощью кнопок и возможность вернуться на предыдущую страницу там, где это предусмотрено техническим заданием.

Код разрабатываемого модуля был написан в соответствии с правилами именования в языке C#, например классы и методы именованы в стиле PascalCase, где все слова в названии написаны с заглавной буквы, а переменные и атрибуты классов именованы в стиле camelCase, где все слова в названии пишутся с заглавной буквы, кроме первой:

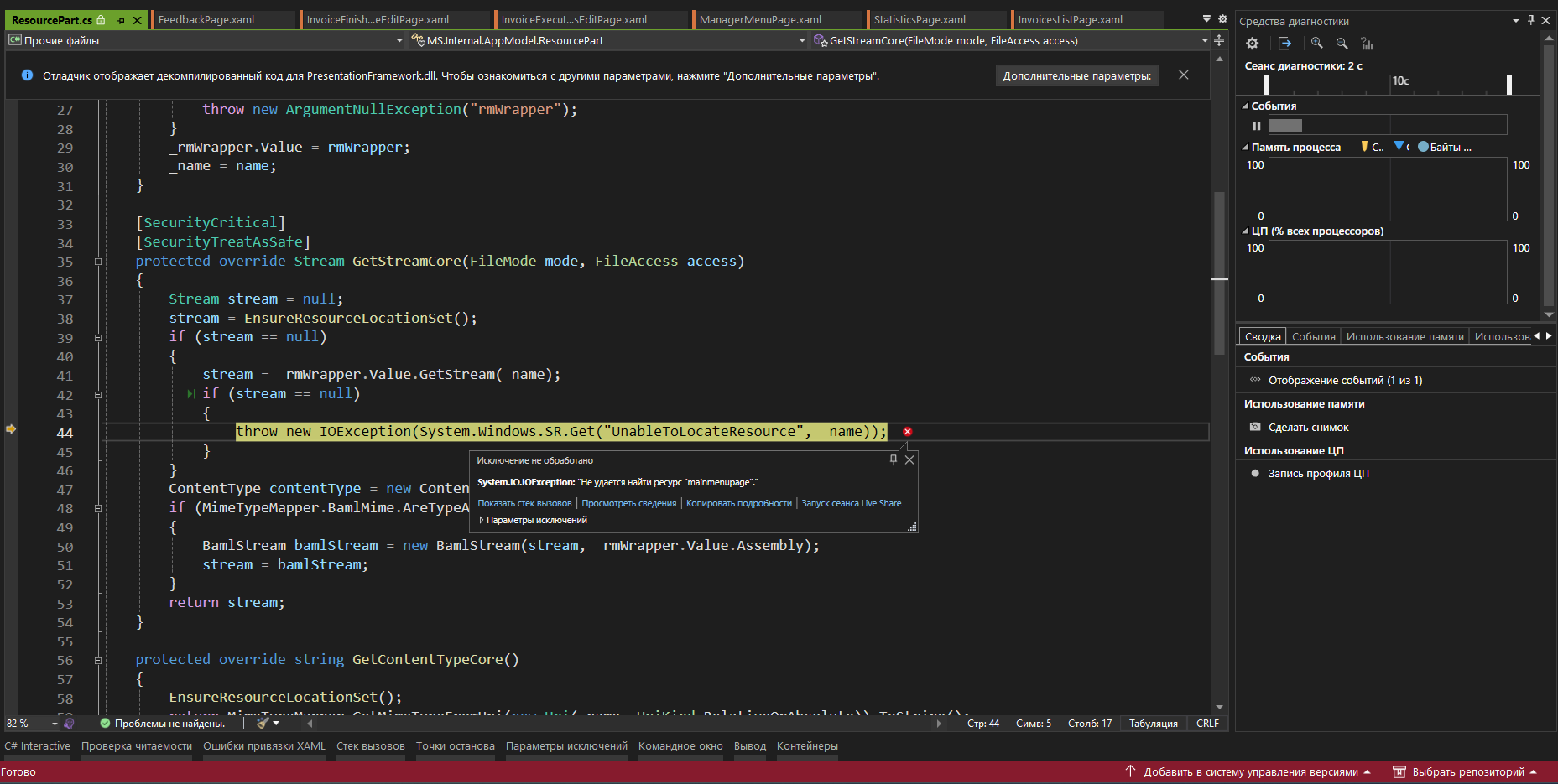


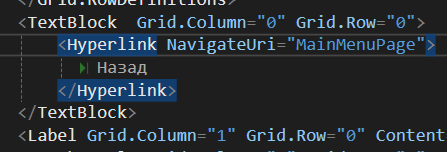
Были реализованы программные обработки исключительных ситуаций, например обработка вводимых данных при авторизации — если пользователь не ввел логин или пароль, то при нажатии кнопки «Войти» ему отобразится сообщение об ошибке с соответствующим текстом:





Была произведена отладка программного модуля и с помощью встроенных средств отладки была локализована ошибка, возникающая при переходе по кнопке Назад на страницу главного меню:



При исправлении адреса перехода на гиперссылке с MainMenuPage на MainMenuPage.xaml ошибка исчезла и отладка была проведена успешно.

Было произведено тестирование разрабатываемого модуля и были выполнены следующие тестовые случаи:

|  |  |  |  |
| --- | --- | --- | --- |
| Назначение | Порядок действий | Ожидаемый результат | Фактический результат |
| Проверка функции проверки вводимых при авторизации данных. | 1. Запустить программу «Учет заявок на ремонт оборудования»  2. Ввести в поле «Логин» текст «login»  3. Нажать кнопку «Войти» | Пользователю отобразится сообщение об ошибке с текстом «Ошибка! Не введён пароль» | Результат соответствует ожидаемому |
| 1. Запустить программу «Учет заявок на ремонт оборудования»  2. Ввести в поле «Пароль» текст «password»  3. Нажать кнопку «Войти» | Пользователю отобразится сообщение об ошибке с текстом «Ошибка! Не введён логин» | Результат соответствует ожидаемому |

|  |  |  |  |
| --- | --- | --- | --- |
| Назначение | Порядок действий | Ожидаемый результат | Фактический результат |
| Проверка функции проверки вводимых при создании заявки | 1. Запустить программу «Учет заявок на ремонт оборудования»  2. Зайти в учетную запись с ролью «user»  3. Перейти на старнциу создания заявки  4. Ввести данные во все поля, кроме поля Emal клиента | Пользователю отобразится сообщение об ошибке с текстом «Ошибка! Не введены все данные! | Результат соответствует ожидаемому |
| 1. Запустить программу «Учет заявок на ремонт оборудования»  2. Зайти в учетную запись с ролью «user»  3. Перейти на старнциу создания заявки  4. Ввести данные во все поля, кроме поля выбора даты срока выполнения | Пользователю отобразится сообщение об ошибке с текстом «Ошибка! Не введён логин» | Результат соответствует ожидаемому |