14

**Тестовое задание**

**Для международной олимпиады**

Постановка задачи

Общее описание задачи

В рамках данного задания необходимо разработать систему учета ошибок. Учет ошибок ведется по проектам. В системе должна быть предусмотрена возможность ведения и просмотра списка ошибок, фильтрации ошибок в разрезе проектов, исполнителей, отделов, ведение статусов по зарегистрированным ошибкам, ведение и просмотр списка исполнителей, построение отчета по ошибкам в разрезе проектов, исполнителей, отделов, статусов.

Сведения о необходимом для выполнения объеме задач, смотрите в конце документа.

Описание предметной области

Одним из важных процессов создания программных продуктов является контроль качества. Одним из традиционных подходов к обеспечению контроля качества является тестирование программных продуктов выделенными для этих целей специалистами. В ходе тестирования специалисты-тестировщики обнаруженные ошибки регистрируют в системе учета ошибок. Каждой зарегистрированной ошибке может быть назначен специалист-разработчик, который будет заниматься отработкой данной ошибки. По результатам отработки запись в системе учета ошибок может быть закрыта, отклонена или переназначена на другого специалиста. Закрытие, отклонение ошибок может производиться путем смены статусов ошибок. Статусы ошибок по каждой ошибке накапливаются (сохраняются).

Функции системы

* Регистрация, редактирование, удаление, просмотр, фильтрация, сортировка информации о сотрудниках
* Регистрация, редактирование ошибок (назначение сотрудников, ввод описания, назначение и смена статусов), просмотр ошибок, фильтрация, сортировка информации об ошибках
* Необязательное требование - Формирование отчета по ошибкам в разрезе проектов и отделов

Технологические требования

1. Архитектура системы

Система должна иметь стандартную трёхуровневую архитектуру – уровень хранения данных (СУБД), уровень бизнес-логики (сервер приложений), уровень представления (клиентский уровень).

1. Платформа

В качестве платформы следует использовать любой язык программирования высокого уровня.

1. СУБД

В качестве СУБД разрешается использовать любую доступную СУБД, кроме внедрённых в сервера приложений.

1. Сервер приложений

Возможно использовать любой промышленный сервер приложений.

1. Реализация клиента

Уровень представления необходимо реализовать в виде веб.

1. Клиент-серверное взаимодействие

Клиент-серверное взаимодействие зависит от выбранных технологий реализации клиента и сервера. К примеру, взаимодействие может осуществляться как по http-протоколу, так и другими средствами (к примеру, XML-сообщениями установленного формата).

1. Описание БД
   1. Начальное наполнение БД

В БД должны иметься справочники, приведенные в таблице 1. Наполнение справочников представлено в таблицах 2, 3, 4 и 5.

Таблица 1. Справочники

|  |  |
| --- | --- |
| **Наименование** | **Описание** |
| Справочник отделов | Кодовый справочник. Содержит названия отделов компании, разрабатывающей ПО |
| Справочник статусов ошибок | Содержит статусы, назначаемые ошибкам в течение их жизненного цикла |
| Справочник степеней критичности ошибок | Содержит наименования степеней критичности ошибок |
| Справочник проектов | Содержит список проектов, разрабатываемых в организации |

Таблица 2. Наполнение справочника отделов

|  |  |
| --- | --- |
| **Код отдела** | **Наименование отдела** |
| 001 | Отдел сбора и анализа требований |
| 002 | Отдел проектирования программных продуктов |
| 003 | Отдел разработки программных продуктов |
| 004 | Отдел обеспечения качества программных продуктов |
| 005 | Отдел внедрения программных продуктов |

Таблица 3. Наполнение справочника статусов ошибок

|  |
| --- |
| **Наименование статуса ошибки** |
| Зарегистрирована |
| Решена |
| Отклонена |

Таблица 4. Наполнение справочника степеней критичности ошибок

|  |
| --- |
| **Наименование степени критичности ошибки** |
| Тривиальная |
| Минорная |
| Мажорная |
| Критическая |
| Блокирующая |

Таблица 5. Наполнение справочника проектов

|  |
| --- |
| **Наименование проекта** |
| АИС МЧС |
| АИС МВД РК |
| АИС Минздрав |

1. Описание серверной части

На серверной стороне слой доступа к данным должен быть абстрагирован и использоваться для всех типов записей (сотрудники, записи об ошибках и их статусах, справочные данные).

Удаление записей о сотрудниках должно производиться «логически», без физического удаления из БД. Для этого, каждая запись о сотрудниках должна иметь статус «Удален».

1. Описание клиентской части

На уровне представления данных необходимо реализовать механизмы ввода, редактирования, просмотра, удаления, сортировки и фильтрации записей.

Интерфейсы основных форм, «Список сотрудников» и «Ошибки», в данном тестовом задании не описываются. Формы «Списка сотрудников» и «Ошибок» должны предоставлять возможность ввода, редактирования, просмотра, удаления, сортировки, фильтрации содержащихся в них данных.

На рисунках 1, 2, 3 и 4 схематически представлены диалоги редактирования/просмотра записей о сотрудниках, вкладки диалога редактирования сведений об ошибке и статусах ошибок соответственно.

Рисунок 1. Схема диалогового окна редактирования записей о сотрудниках

Рисунок 2. Схема вкладки «Главная» диалогового окна редактирования записей об ошибках

Рисунок 3. Схема вкладки «Статусы» диалогового окна редактирования записей об ошибках

Рисунок 4. Схема диалогового окна «Редактирование статуса»

Описание функций диалоговых окон по полям представлено в таблицах 6, 7, 8 и 9. Кроме описания, в данных таблицах указано, какие поля необходимо включать в фильтры; фильтры строятся произвольным образом. Форматно-логический контроль (ФЛК) осуществляется следующим образом – если в поле введено неверное значение, возможность сохранения блокируется, поле выделяется каким-либо образом как ошибочное.

Таблица 6. Описание полей диалогового окна редактирования записей о сотрудниках

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **№** | **Название поля** | **Тип** | **Описание** | **ФЛК** | **Включать в фильтр** |
| 1 | Фамилия | Текстовое поле | Фамилия сотрудника | Только символы алфавита | Да |
| 2 | Имя | Текстовое поле | Имя сотрудника | Только символы алфавита | Нет |
| 3 | Отчество | Текстовое поле | Отчество сотрудника | Только символы алфавита | Нет |
| 4 | Отдел | Кодовый справочник | Содержит справочник «Список отделов». Позволяет вводить код отдела либо выбирать его наименование из списка | Только справочные значения | Да |
| 5 | Поле подтверждения | Кнопки | Содержит кнопки подтверждения/отмены ввода данных | Нет | Нет |

Таблица 7. Описание полей вкладки «Главная» диалогового окна редактирования записей об ошибках

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **№** | **Название поля** | **Тип** | **Описание** | **ФЛК** | **Включать в фильтр** |
| 1 | Идентификатор | Текстовое поле | Содержит уникальный идентификатор ошибки в системе. Представляет собой целое число, генерируемое на сервере и передаваемое клиенту. Поле не подлежит редактированию. | Нет | Да |
| 2 | Дата регистрации | Поле даты | Содержит дату регистрации ошибки в системе. При заведении новой ошибки автоматически проставляется текущая дата. Редактирование возможно. | Дата должна быть не больше текущего дня | Нет |
| 3 | Сотрудник | Справочник | Содержит полный список сотрудников, зарегистрированных в системе | Только справочные значения | Да |
| 4 | Критичность | Справочник | Содержит справочник «Степени критичности ошибок». | Только справочные значения | Нет |
| 5 | Проект | Справочник | Содержит справочник проектов | Только справочные значения | Да |
| 6 | Описание | Текстовое поле | Содержит описание ошибки, причин и условий ее возникновения и т.д. | Длина поля не более 1000 символов | Нет |
| 7 | Поле подтверждения | Кнопки | Содержит кнопки подтверждения/отмены ввода данных | Нет | Нет |

Таблица 8. Описание полей вкладки «Статусы» диалогового окна редактирования записей об ошибках

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **№** | **Название поля** | **Тип** | **Описание** | **ФЛК** | **Включать в фильтр** |
| 1 | Панель инструментов | Панель инструментов | Содержит кнопки, позволяющие создавать новые записи о статусах, редактировать и просматривать существующие, удалять и фильтровать статусы | Нет | Нет |
| 2 | Список статусов | Таблица | Содержит колонки «Статус» (значение из справочника статусов ошибок), «Сотрудник, назначивший статус» (значение из справочника сотрудников, зарегистрированных в системе), «Дата задания статуса» | Нет | Нет |

Таблица 9. Описание полей диалогового окна «Редактирование статуса»

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **№** | **Название поля** | **Тип** | **Описание** | **ФЛК** | **Включать в фильтр** |
| 1 | Дата задания статуса | Поле даты | Содержит дату задания статуса. При задании нового статуса должно автоматически заполняться текущей датой | Дата должна быть не больше текущего дня | Да |
| 2 | Статус | Справочник | Содержит справочник «Статусы ошибок» | Только справочные значения | Да |
| 3 | Сотрудник, назначивший статус | Справочник | Содержит список значений справочника сотрудников, зарегистрированных в системе | Только справочные значения | Да |
| 4 | Поле подтверждения | Кнопки | Содержит кнопки подтверждения/отмены ввода данных | Нет | Нет |

Таким образом, иерархия форм в системе выглядит:

**3**

**1**

**2**

**5**

**4**

Рисунок 5. Иерархия вложения диалогов в системе

1 – Список сотрудников;

2 - Редактирования записей о сотрудниках;

3 - Список ошибок;

4 - Редактирования записей об ошибках;

5 - Редактирование статуса.

Контроль результатов

Контролируемые артефакты и принципы их оценки

Ниже приведен список артефактов (результатов работы), которые будут оцениваться при сдаче работы, а также вкратце очерчены принципы их оценки.

1. Код

Важнейшим артефактом является код готовой системы. Исходные файлы системы передаются для оценки. Будет оценивать работоспособность кода и его качество, понимание принципов ООП, шаблонов проектирования ПО, владение указанными технологиями.

1. Структура БД

Будут оцениваться знания принципов построения БД, владение инструментами проектирования БД.

1. Пользовательские интерфейсы

Пользовательские интерфейсы будут оцениваться в ходе демонстрации системы. Критериями будут являться работоспособность интерфейсов, их соответствие требованиям/схемам, а также практичность пользовательских интерфейсов (usability).

Объемы выполнения задачи

Необязательным в данном задании могут являться сортировка в таблицах, фильтрование,.

Однако, наличие всех вышеуказанных пунктов, а так же схема БД, спроектированная в CASE-средстве, и схема структуры классов в UML является будет являться большим плюсом при проверке выполненного задания.

Так же приветствуется реализация отчетных форм

Выходные формы

Основными назначениями выходных форм являются проверка умения пользоваться языком манипулирования данных, специфичным для выбранной СУБД (SQL, XQuery и т.п. Оцениваться будет работоспособность и качество запросов для выборки при построении отчета.

Описание выходных форм

* 1. Отчет (построение, отображение)

В качестве одной из выходных форм системы должен являться отчет «Ошибки в разрезе проектов и отделов». Отчет формируется любым удобным способом (Crystall Reports, Jasper, обычный html, выгрузка в Microsoft Word или Excel). Формат отчета представлен в таблице 10. Необходимо предусмотреть возможность формировать отчет за указанный период, в качестве входного параметра даты для отчета использовать дату регистрации ошибки.

Таблица 10. Формат отчета «Ошибки в разрезе проектов и отделов»

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Отдел1** | **…** | **Отделn** |
| **Проект1** | Количество ошибок1 | … | Количество ошибокn |
| **Критичность1** | … | … | … |
| **…** | … | … | … |
| **Критичностьi** | … | … | … |
| **Итого** | Сумма ошибок по проекту1 по отделу1 | … | Сумма ошибок по проекту1 по отделуn |
| **…** | .. | … | … |
| **Проектm** | Количество ошибок1 | … | Количество ошибокn |
| **Критичность1** | … | … | … |
| **…** | … | … | … |
| **Критичностьi** | … | … | … |
| **Итого** | Сумма ошибок по проектуm по отделу1 | … | Сумма ошибок по проектуm по отделуn |

Прочая информация

В виду ограниченности по срокам данного задания, не предусматривается реализация административных функций, а также функций аутентификации пользователей.