Приложение «Estimates 0.2.0»

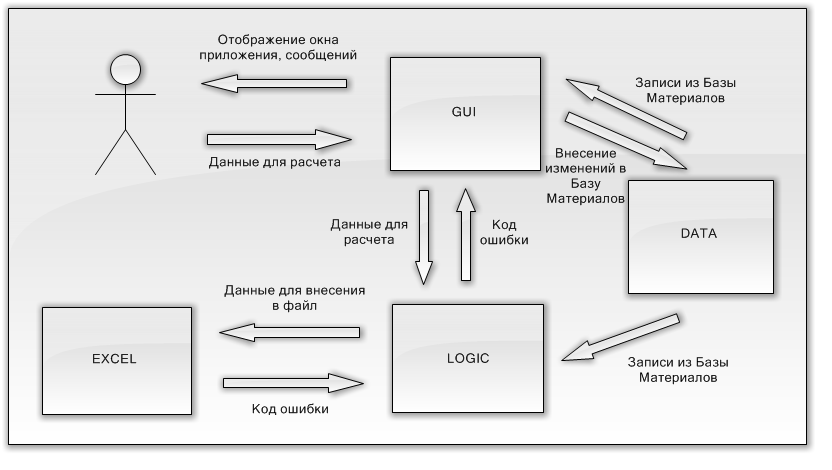
Описание общей структуры

Отличия от версии 0.1.0 обозначены красным.

Приложение состоит из четырех основных модулей:

1. GUI: модуль, отвечающий за отображение графического интерфейса приложения, за реакцию на действия пользователя, за проверку вводимых пользователем данных.
2. LOGIC: модуль, реализующий логику расчета смет из введенных пользователем данных.
3. EXCEL: модуль, ответственный за создание итогового файла формата xls.
4. DATA: модуль, выполняющий обращения к базе данных SQLite.

Взаимодействие между модулями и пользователем показано на схеме:

Приложение пишется на языке Java. Исходный код приложения разбит на пять корневых пакетов:

1. *guipackage* — содержит реализацию модуля GUI.
2. *logicpackage* — содержит реализацию модуля LOGIC.
3. *excelpackage* — содержит реализацию модуля EXCEL.
4. *datapackage* — содержит реализацию модуля DATA.
5. *supportpackage* — вспомогательный пакет.

Общие положения

Имена пакетов, подпакетов, классов, методов, полей пишутся на английском языке.

Имена методов являются глаголами, отражающими суть задачи, выполняемой методом: *CalculateEstimates().*

Имена классов и полей являются существительными, отражающими моделируемую сущность либо роль в структуре приложения: *Material, ExcelHelper.*

Имена пакетов и подпакетов пишутся строчными буквами.

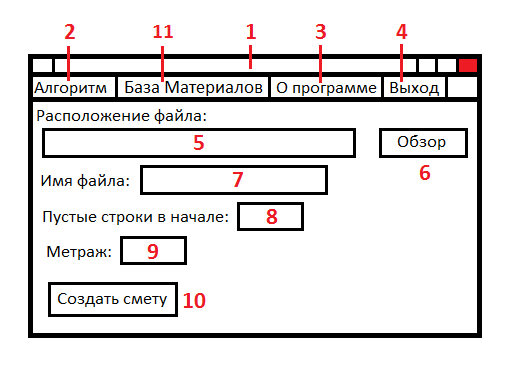
Имена классов, методов, полей пишутся с заглавной буквы. Если имя содержит несколько слов, каждое из них пишется с заглавной буквы.

Методы (кроме конструкторов и методов обратного вызова) и поля классов объявляются приватными, если в задании не указано обратное.

Задание на разработку модуля GUI.

За основу разработки берется модуль GUI из версии 0.1.0.

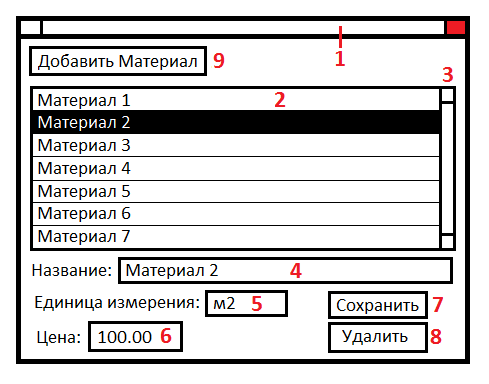
Новый вид главного окна приложения:



Описание новых элементов главного окна приложения:

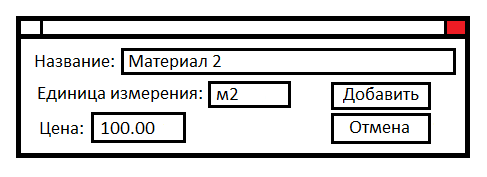
11. При выборе пункта меню «База Материалов», поверх основного окна пользователь видит окно редактирования Базы Материалов.

Вид окна редактирования Базы Материалов:



Описание окна редактирования Базы Материалов:

1. Стандартный заголовок окна Windows с кнопкой, позволяющей закрыть окно.
2. Список с прокруткой (класс JList), перечисляющий все содержащиеся в Базе материалы. Список материалов получается от модуля DATA при создании окна. Для этого создается объект класса *datapackage.MaterialDatabaseHelper*, конструктор которого имеет вид *public MaterialDatabaseHelper()*. После этого вызывается метод *public String[] getMaterialsList()* созданного объекта. Список, возвращаемый методом, и является требуемым списком материалов. При выборе пользователем какого-либо пункта, вызывается метод *public MaterialDatabaseRecord getRecord(String Name)* созданного объекта. При этом, *Name —* название выбранного пользователем материала. Метод возвращает объект класса *datapackage.MaterialDatabaseRecord*, содержащий полное описание материала:
   1. метод *public String getName()* возвращает имя материала, отображаемое в поле 4;
   2. метод *public String getUnit()* возвращает единицу измерения материала, отображаемую в поле 5;
   3. метод *public double getPrice()* возвращает цену материала, отображаемую в поле 6.
3. Полоса прокрутки списка.
4. Название выбранного материала. Пользователь может редактировать данное поле. Поле может содержать символы русской и латинской раскладки, цифры, знаки препинания. Длина строки не должна превышать 100 символов.
5. Единица измерения выбранного материала. Пользователь может редактировать данное поле. Поле может содержать символы русской и латинской раскладки, цифры, знаки препинания. Длина строки не должна превышать 10 символов.
6. Цена выбранного материала. Пользователь может редактировать данное поле. Цена отображается с двумя знаками после десятичной точки. Поле может содержать только вещественные неотрицательные числа в диапазоне от 0.00 до 999 999 999.99.
7. При нажатии кнопки «Сохранить»:
   1. отбрасываются возможные пробельные символы с начала и с конца каждого поля, заполненного пользователем;
   2. проверяется, правильно ли пользователь заполнил поля: все ли поля заполнены, не содержат ли поля запрещенных символов, не выходят ли значения полей за пределы допустимых диапазонов. Если обнаружена ошибка в заполнении полей, поверх основного окна пользователь видит диалоговое окно с текстом «*Ошибка в заполнении полей*» и единственной кнопкой «*ОК*», закрывающей окно;
   3. значения полей 4, 5, 6 передаются объекту класса *datapackage.MaterialDatabaseRecord,* полученному от модуля DATA. Для этого используются методы:
      1. *public void setName(String Name)* — для задания имени материала;
      2. *public void setUnit(String Unit)* — для задания единицы измерения;
      3. *public void setPrice(double Price)* — для задания цены;
   4. вызывается метод *public int setRecord(MaterialDatabaseRecord Record)* созданного объекта класса *datapackage.MaterialDatabaseHelper.* При этом, *Record —* отредактированное описание материала. В случае, если вызванный метод возвращает значение, отличное от нуля, поверх основного окна пользователь видит диалоговое окно с текстом «*Ошибка при работе с базой данных*» и единственной кнопкой «*ОК*», закрывающей окно.
   5. список 2 обновляется при помощи метода *public String[] getMaterialsList().*
8. При нажатии кнопки «Удалить», поверх основного окна пользователь видит диалоговое окно с текстом: «*Вы уверены, что хотите удалить данный материал из Базы?»* и двумя вариантами ответа: «*Да*», «*Нет*». В случае, если пользователь выбирает ответ «Да»:
   1. вызывается метод *public int deleteRecord(MaterialDatabaseRecord Record)* созданного объекта класса *datapackage.MaterialDatabaseHelper.* При этом, *Record* — описание выбранного материала, полученное от модуля DATA. В случае если вызванный метод возвращает значение, отличное от нуля, поверх основного окна пользователь видит диалоговое окно с текстом «*Ошибка при работе с базой данных*» и единственной кнопкой «*ОК*», закрывающей окно.
   2. список 2 обновляется при помощи метода *public String[] getMaterialsList().*
9. При нажатии кнопки «Добавить материал», поверх основного окна пользователь видит окно создания нового материала:



Правила заполнения полей аналогичны описанным выше, в пунктах 4, 5, 6. Нажатие кнопки «Отмена» равносильно закрытию окна. При нажатии кнопки «Добавить»:

* 1. аналогично пункту 7.a
  2. аналогично пункту 7.b
  3. вызывается метод *public int createRecord(String Name, String Unit, double Price)* созданного объекта класса *datapackage.MaterialDatabaseHelper.* При этом, *Name, Unit, Price —* значения полей «Название», «Единица измерения», «Цена» соответственно. В случае если вызванный метод возвращает значение, отличное от нуля, поверх основного окна пользователь видит диалоговое окно с текстом «*Ошибка при работе с базой данных*» и единственной кнопкой «*ОК*», закрывающей окно.
  4. список 2 обновляется при помощи метода *public String[] getMaterialsList().*