

# Инструкция по использованию приложения DialogBuilder

## Оглавление

Оглавление.....	1
Установка.....	2
Запуск скрипта .....	2
Предназначение и возможности .....	3
Совместимость:.....	3
Реализованные возможности в текущей версии: .....	3
Планируется в следующих версиях: .....	3
Известные проблемы и недостатки:.....	4
Работа с программой .....	5
Общие принципы .....	5
Добавление элементов в диалог .....	5
Редактирование свойств элементов.....	6
Редактирование имён переменных .....	6
Сохранение и контроль результата .....	6
Работа с настройками программы.....	6
Особенности работы с настройками: .....	7
Методика и рекомендации разработки диалога в приложении Dialog Builder .....	7
Дополнительные сведения .....	9
Обновления в новой версии документа: .....	9

**Обновлено 20.06.2014**

## Установка

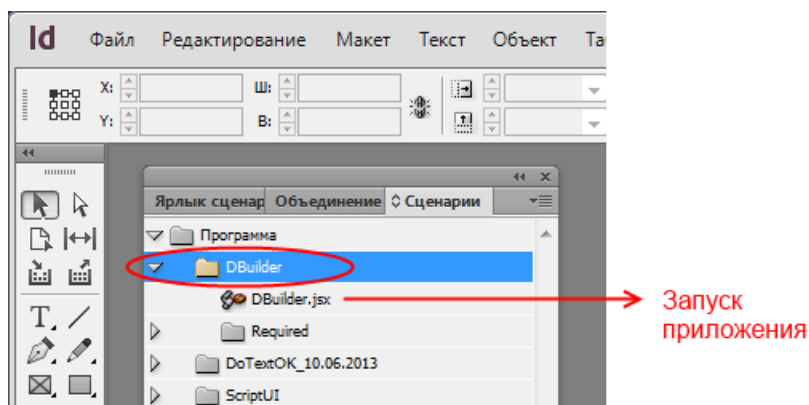
1. Скачайте архив, содержащий последнюю версию программы, из репозитория на сайте **GitHub** — <https://github.com/SlavaBuck/DialogBuilder> (прямая ссылка на архив: <https://github.com/SlavaBuck/DialogBuilder/archive/master.zip>);
2. Из скачанного архива (DialogBuilder-master.zip) распакуйте содержимое папки `contrib` в папку со скриптами Adobe InDesign. (На своей машине с установленной Windows 7 и Adobe InDesign CS6 я использую путь `C:\Program Files (x86)\Adobe\Adobe InDesign CS6\Scripts\Scripts Panel`).

Папка `src` и корневые файлы архива DialogBuilder-master.zip содержат исходный код программы. Для работы с ними требуются библиотеки из репозитория <https://github.com/SlavaBuck/Includes>. Скачайте библиотеки, распакуйте их в отдельную папку со скриптами и в файле `./src/02application.jsx` в строке 12: `#includepath "../Include/"` пропишите к ним корректный относительный путь. По умолчанию предполагается, что все библиотеки находятся в папке `Include`, которая расположена в папке, на уровень выше по отношению к папке с программой (в архиве это папка `contrib`). Если соблюсти данное условие — редактирование путей не потребуется.

## Запуск скрипта

Для того, что бы файл скрипта был виден в палитре скриптов Adobe InDesign, его следует располагать по одному из predetermined путей: содержимое папки `C:\Program Files (x86)\Adobe\Adobe InDesign CS6\Scripts\Scripts Panel` отображается в палитре скриптов под группой «Программа» — её можно назвать «системной» папкой скриптов InDesign; содержимое папки `C:\Users\Buck\AppData\Roaming\Adobe\InDesign\Version 8.0\ru_RU\Scripts\Scripts Panel` отображается под группой «Пользователь» — её можно назвать «пользовательской» папкой скриптов. Соответственно, всё то, что расположено в данных папках, будет автоматически видно в соответствующих группах (пути приведены для системы Windows).

Если в «системной» папке скриптов создать папку **Dbuilder** и распаковать в неё содержимое папки `contrib` из архива с программой — в палитре скриптов InDesign у вас должна появиться подобная картина:



Вы можете запустить программу прямо из палитры, выполнив файл файл `Dbuilder.jsx` из папки `Dbuilder`, или открыть `Dbuilder.jsx` в ExtendScript Toolkit (ESTK) и запустить его из самого редактора.

## Предназначение и возможности

Программа предназначена для максимально быстрой и простой визуальной разработки диалогов на языке JavaScript в среде Adobe ExtendScript. На основании разработанной модели диалога формируется «ресурсный» javascript-код, который прямо из программы может быть сохранён в отдельном файле и использован в качестве основы для разработки диалога в целевом рабочем скрипте.

### Совместимость:

Программа разрабатывается в среде Adobe ExtendScript CS6 и полностью совместима с версиями Adobe InDesign CS 5.5 и 6 для Windows 7. Предположительно, программа будет нормально работать и с более ранними версиями продуктов Adobe в разных версиях Windows.

В связи со значительными отличиями в системе управления событиями ScriptUI в версии Adobe InDesign CC возможны проблемы (не тестировалось).

Судя по опыту пользователей с операционной системой MacOS — программа не работает для версий Adobe InDesign CS, но частично работает с версией Adobe InDesign CC (при этом инвертируются функции нажатий левой и правой клавиши мыши). Вопрос совместимости с MacOS в текущей версии программы (актуально для версии 1.50) ещё не рассматривался.

### Реализованные возможности в текущей версии:

- Формирование, сохранение и передача в ESTK корректного javascript-кода для объекта диалога, разрабатываемого в рамках приложения;
- Многодокументный интерфейс — позволяет одновременную работу с несколькими документами-диалогами;
- Полная локализация интерфейса — обеспечивается отдельным файлом locales.jsxinc, расположенном в папке *Requires* и содержащим доступные в пользовательском интерфейсе языки для всех строковых ресурсов приложения (для текущей версии поддерживается русский и английский);
- Поддерживаются широкие возможности по управлению настройками приложения и внешним видом как интерфейса самой программы, так и внешним видом документов-диалогов;
- Поддерживается немедленная реакция на редактирование свойств и синхронизация внешнего вида объектов пользовательского интерфейса, составляющих пользовательский диалог, таким образом, поддерживается практически 100% графическое соответствие внешнего вида между редактируемым документом и конечным объектом-диалогом;
- Реализованы исчерпывающие возможности по настройке внешнего вида (настройки цветов и шрифтов) объектов пользовательского интерфейса, составляющих пользовательский диалог;
- Поддерживается возможность управления пользовательскими наборами шрифтов и цветов, которые могут быть сохранены в настройках приложения и использованы повторно;
- Наряду со стандартными ScriptUI-объектами, поддерживается работа с произвольными пользовательскими элементами графического интерфейса, которые могут быть дополнительно включены в файл controls.jsxinc, расположенном в папке *Requires* и, при необходимости, дополнены собственными свойствами (например, так реализована работа с элементом **Separator**).

### Планируется в следующих версиях:

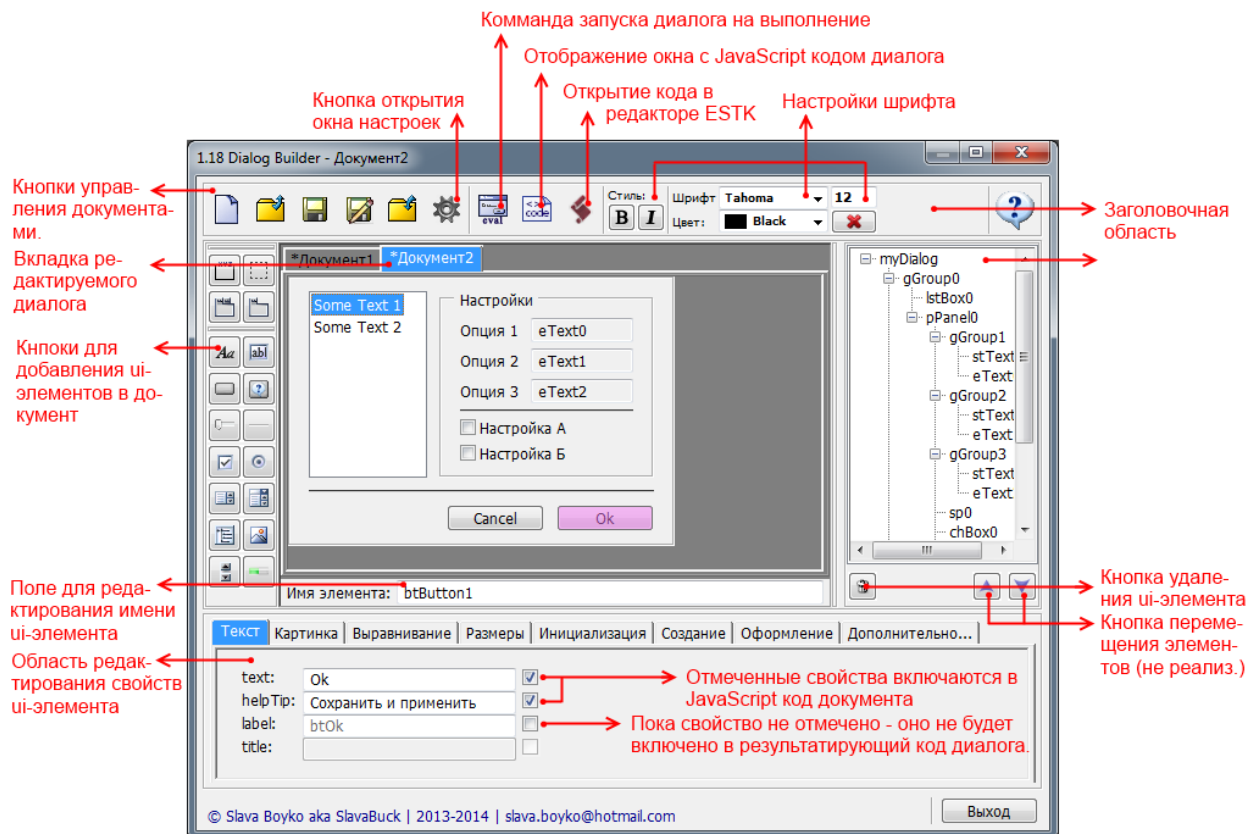
- Возможность открытия с последующим редактированием ранее сохранённых документов — объектов-диалогов;

- Групповое выделение объектов пользовательского интерфейса в документе и выполнение групповых операций: выравнивания, установка размеров и пр...;
- Расширение возможностей по настройке и работе со списками и изображениями в объекте диалога:
  - конвертация и включение изображений в тело скрипта;
  - настройка масштабирования изображений, помещаемых в пользовательские элементы управления;
  - Работа с массивами изображений для элементов, поддерживающих мультисостояния (IconButton/toolbutton и т.п.);
  - реализация работы с многоколоночными списками и включением изображений в списки;
  - привязка списков к источникам данных;
  - дополнение более широких возможностей по редактированию элемента управления TreeView;
- Использование мастера начального создания диалога с предустановленными шаблонами;
- Внедрение совместимости с MacOS и последними версиями Adobe Indesign CC.

#### Известные проблемы и недостатки:

- Работа с изображениями и списками реализована на самом базовом уровне, так, например, не реализована работа с многоколоночными списками, настройкой древовидной структуры для TreeView, массивами изображений для элементов с поддержкой мультисостояний и т.п. (подробнее см. в планах не реализацию);
- При генерации ресурсных строк диалогов некоторые цифровые значения свойств сохраняются как строчные значения, например, запись свойства margins:[10, 15, 10, 15] может быть сгенерирована в виде margins:['10', '15', '10', '15']. Эта особенность не влияет на функционирование кода, так как ScriptUI корректно обрабатывает такие случаи и конвертирует эти строчные значения в корректные числовые, поэтому пока эта проблема игнорируется (в том числе из соображений производительности);
- Не совсем корректно вычисляется размер элемента управления TabbedPanel при добавлении в него вкладок (элементов Tab). В связи с нетривиальной логикой расчёта итогового размера TabbedPanel (зависит от многих факторов: кол-ва панелей, их размеров, размеров их подписей и т.д.) исследуются более простые и быстрые возможности это делать, а пока используется грубый и упрощённый подход, который даёт небольшую погрешность в сторону увеличения фактического размера элемента;
- Не реализовано открытие ранее сохранённых документов. Реализация требует существенной работы по синтаксическому анализу JavaScript кода и планируется в ближайших короткосрочных планах;
- Длительная начальная инициализация приложения связана с необходимостью создания большого количества графических элементов для интерфейса пользователя и элементов управления, таких как списки цветов. Решение уже найдено и планируется к реализации в ближайших короткосрочных планах;
- Самым основным недостатком пока считаю отсутствие 100% совместимости программы с последней версией Adobe InDesign CC для Windows. Реализация планируется только в долгосрочных планах, как и поддержка работы в MacOS;

## с программой



## Общие принципы

Идея написания программы заключалась в предоставлении возможностей максимально простой и быстрой разработки диалогов для скриптов, работающих в среде Adobe ExtendScript и InDesign. Разработка диалога опирается на стандартные возможности компоновщика ScriptUI (AutoLayout manager) при которой размеры и положение элементов диалога вычисляется автоматически в зависимости от их настроек и содержимого. Таким образом, после добавления в документ, элементы требуют минимальной дополнительной настройки (или вообще не требуют). В то же время редактором свойств элементов предусматривается возможность «ручного» управления всеми свойствами элементов, включая их положения в контейнерах и т.п..

## Добавление элементов в диалог

Для создания диалога необходимо создать новый документ. С помощью группы кнопок для добавления ui-элементов в диалог добавляются все необходимые компоненты. После выделения элемента в нижней части отображаются значения всех доступных для него свойств. Для добавления элементов в контейнер (под контейнерами понимаются такие элементы как группы, панели и вкладки / group, panel, tabbedpanel, tab) его необходимо предварительно выделить мышкой (кликнуть либо по самому элементу, либо по соответствующему узлу в дереве компонентов диалога в правой части окна приложения). Чтобы фокус добавления элементов автоматически переустанавливался на контейнер сразу после его добавления в диалог, можно изменить соответствующую установку «Автофокус» на вкладке «Основные» в панели настроек приложения. По умолчанию (без автофокуса) после добавление контейнера, несмотря на то, что он выделяется, активным продолжает оставаться прежний контейнер и добавление новых элементов продолжается в нём.

### Редактирование свойств элементов

После добавление элемента в диалог, на него автоматически переустанавливается фокус редактирования, о чём свидетельствует соответствующее цветовое выделение (цвет выделения может настраиваться в настройках: кнопка «Настройки» -> «Основные» -> «Цвет подсветки») и переключаются поля свойств в нижней части экрана. При этом свойства, не доступные для данного элемента блокируются. Справа от свойства расположено поле для установки флага, который указывает на необходимость включать данное свойство в результирующий код диалога. Несмотря на редактирование свойства и наличия у него какого-либо значения — свойство не будет включено в код диалога, пока не будет отмечено. Для визуального выделения неотмеченных свойств дополнительно используется серый цвет для текста в соответствующих полях редактирования. Редактирование свойств организовано по принципу «Live update» — при котором любое редактирование свойства сразу отображается на состоянии и внешнем виде элемента. Для выделения любого элемента достаточно кликнуть по нему левой клавишей мыши или выбрать его в дереве элементов, расположенном в правой части окна программы.

### Редактирование имён переменных

При добавлении элементов в диалог, для них происходит автоматическое генерирование имён переменных. Переменные генерируются на основании шаблона, заданного в настройках на вкладке «Имена переменных». На выбор доступен один из трёх шаблонов: «small» — короткие двух- и трёх-буквенные имена; «full» — длинные имена (установлено по умолчанию); «user» — пользовательская таблица имён (для незадаанных значений используется пользовательское значение по умолчанию, отображаемое серым шрифтом в соответствующем поле элемента). Для генерации имени используется простое правило — к выбранному шаблону имени элемента добавляется его порядковый номер в структуре диалога. Сгенерированное имя переменной доступно для редактирования в соответствующем поле редактора. При редактировании имени переменной для выбранного элемента следует не допускать использование одинакового имени для нескольких элементов в пределах одного контейнера, в то же время элементы из разных контейнеров могут иметь одинаковые имена.

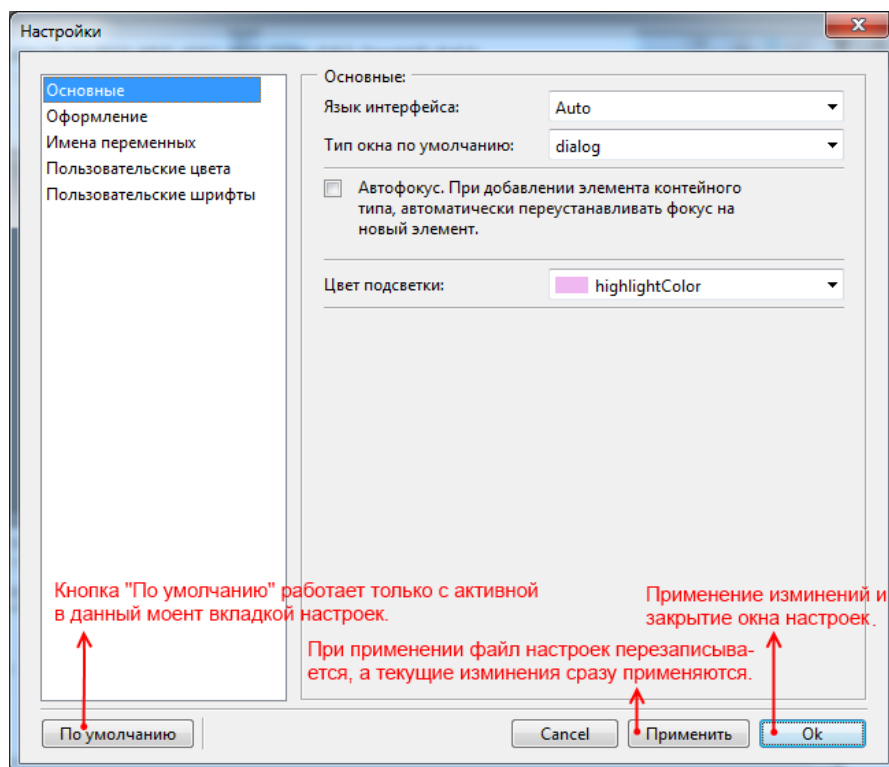
### Сохранение и контроль результата

При окончании работы над диалогом его можно сохранить в файл скрипта. В ходе работы можно контролировать его фактический внешний вид с помощью кнопки «Eval» в заголовочной области приложения. Просмотреть текущий код без предварительного сохранения можно с помощью кнопки «Code». Также, можно напрямую передать код диалога в редактор ESTK (при передаче кода происходит автоматическое сохранение и, в случае необходимости, зарос имени файла).

### Работа с настройками программы

Окно настроек доступно по нажатию кнопки «Настройки» в заголовочной области приложения. Первые три вкладки «Основные», «Оформление» и «Имена переменных» содержат все доступные опции для настройки приложения. Вкладки «Пользовательские цвета» и «Пользовательские шрифты» содержат соответствующие таблицы. Все добавленные в них значения становятся доступными в соответствующих элементах управления в рамках всего приложения. Все добавленные в данные таблицы значения сохраняются в файле настроек и остаются доступными после перезапуска приложения.

Нажатие кнопок «Ок» и «Применить» приводит к перезаписи файла настроек и применению внесённых изменений. Нажатие кнопки «По умолчанию» отражается только на открытой в данный момент вкладке.



### Особенности работы с настройками:

При запуске программы происходит попытка обнаружения файла *options.jsxinc* в папке Required, расположенной в папке с программой. В случае обнаружения файла (и наличия в нём сигнатуры <DBuilder options> ) происходит применение настроек, объявленных в файле. Если файл с настройками не обнаруживается – используются настройки по умолчанию. Файл автоматически создаётся и перезаписывается при применении и сохранении настроек в самом приложении.

Файл допустимо редактировать как с помощью графического интерфейса, предоставляемого приложением, так и в ручном режиме. Все допустимые настройки и их описание содержатся в файле *options.sample.jsxinc*, расположенном в папке Required.

При ручном редактировании файла настроек следует знать, что при сохранении, настройки сохраняются в виде JavaScript-объекта в JSON нотации. Таким образом, в файле они должны синтаксически представлять корректный JavaScript-объект (наиболее частая ошибка – пропуск запятой после определения свойства и другие подобные синтаксические ошибки, специфичные для JavaScript...).

## Методика и рекомендации разработки диалога в приложении Dialog Builder

Данный раздел предполагает наличие определённых базовых знаний и некоторого опыта создания диалогов в среде ExtendScript. При разработке и создании макета для диалога следует понимать общие принципы работы системного компоновщика элементов (AutoLayout Manager)

для графической подсистемы ScriptUI. Получить базовые навыки работы со ScriptUI можно с помощью замечательно работы **ScriptUI for dummies**, автором которой является Peter Kahrel. Последнюю версию документа можно скачать по адресу — <http://www.kahrel.plus.com/indesign/scriptui.html>, также на сайте <http://adobeindesign.ru/> на странице <http://adobeindesign.ru/2010/12/26/scriptsui/> доступен перевод (более ранней версии) этого руководства (прямая ссылка: <http://adobeindesign.ru/wp-content/uploads/Написание-скриптов.-Пользовательский-интерфейс26.12.2010.zip>).

Приступая к работе над макетом диалога, следует знать, что в пределах контейнера (элемента контейнерного типа, которым выступает один из элементов `group`, `panel`, `tabbedpanel`, `tab` и сам объект диалога: `dialog` или `palette`) элементы располагаются один за другим только по вертикали или горизонтали. Свойство `orientation` в контейнере отвечает за способ размещения в нём элементов. Оно имеет текстовый формат и может обладать одним из трёх значений: `'row'` (элементы располагаются горизонтально в ряд, по умолчанию для контейнеров `group`); `'column'` (элементы располагаются вертикально, по умолчанию для всех, кроме `group`) и `'stack'` (элементы располагаются один над другим).

Для расположения нескольких различных элементов с требуемой ориентацией относительно друг-друга существует два основных способа: 1) Вычислять и задавать требуемые значения свойства `location: [x, y]` для каждого элемента управления; 2) Объединять элементы с помощью контейнеров в группы и размещать такие группы с помощью родительских контейнеров в нужной ориентации. Второй способ является гораздо более простым и наглядным, так как не требует дополнительных вычислений и легко реализуем на практике. Хотя приложение DialogBuilder позволяет использовать оба подхода, именно второй способ является основным в силу своей простоты и наглядности.

Таким образом, общий подход в разработке макета диалога с помощью конструктора диалогов DialogBuilder заключается в создании необходимой структуры вложенных контейнеров с требуемой ориентацией, в случае необходимости контейнеру можно произвольно переустанавливать его свойство `orientation` и, размещением в них требуемых элементов управления.

Проще всего понять вышеизложенный подход позволяет короткое видео на сайте YouTube <http://youtu.be/i6P0OuBvmql>, в котором продемонстрировано создание небольшого диалога, приведённого в иллюстрации на странице 5 данного руководства.

Вторым, исключительно практическим аспектом, является контроль над размерами определённых блоков диалога или всего диалога в целом, с целью придания ему и его составляющим элементам нужных пропорций, выравнивания и размещения. Тут также возможны варианты. Например, можно задавать нужное свойство `preferredSize: [x, y]` для контейнеров и использовать свойство `alignment` в элементах управления или `alignChildren` — в самих контейнерах. Однако, на практике, практически никогда не нужно использовать заранее предопределённые свойства размеров (`preferredSize` или `size`) для элементов управления контейнерного типа. Вместо этого, следует задавать требуемые размеры самим элементам управления, а расчёт и масштабирование контейнеров оставить на попечении у AutoLayout Manager-а. При этом, если элемент управления позволяет задавать собственный размер с помощью свойства `characters` (`StaticText`, `EditText`, ...) — лучше использовать именно его, такой подход избавит вас от неприятностей неправильного размера и неудачного обрезания текста в элементах, при использовании для них различных размеров шрифтов и начертаний.



## Дополнительные сведения

Текущая версия (1.50) находится в активной разработке, поэтому многие возможности ещё не включены в данную версию.

В остальном программа полностью функциональна и может быть использована для быстрого создания диалогов и импорта их в виде готового JavaScript кода, полностью готового для использования в рабочих скриптах.

## Обновления в новой версии документа:

Версия	Описание	Дата	Автор(ы)
1.18	Первая редакция	02.04.2014	Вячеслав Бойко
1.20	<ul style="list-style-type: none"><li>Слегка подкорректирована структура файла, внесены мелкие правки в текст;</li><li>В соответствии с новой функциональностью изменён раздел «Функциональные особенности текущей версии»</li></ul>	25.05.2014	Вячеслав Бойко
1.40	<ul style="list-style-type: none"><li>В связи с введением поддержки работы со списками и изображениями, отредактирован раздел «Функциональные особенности текущей версии»</li></ul>	05.06.2014	Вячеслав Бойко
1.50	<ul style="list-style-type: none"><li>Номера версий документации приведены в соответствие с версией приложения;</li><li>Существенно обновлено содержимое разделов в соответствии с новой функциональностью приложения;</li><li>Добавлен раздел «Предназначение и возможности», в разделе описаны реализованные и планируемые функциональные возможности, в том числе приведены известные недостатки и планы по их устранению;</li><li>Добавлен раздел «Методика и рекомендации разработки диалога в приложении Dialog Builder»</li></ul>	20.06.2014	Вячеслав Бойко