Lersus. Основные задачи и возможности

Задачи

- Создавать и поддерживать современный интерактивный веб-контент
- Избавить автора от необходимости программировать при создании контента
- Обеспечивать соответствие структуры и дизайна контента всем требованиям портала, для которого этот контент создается
- Помогать автору в создании контента
- Предоставлять удобный интерфейс создания тестов
- Импорировать и экспорировать контент
- Внедрять активный контент в интерактивные материалы

Возможности

- Создание и редактирвоание сложного интерактивного контента
- Использование в контенте Flash анимаций, Видео, Аудио, Java-апплетов
- Быстрое визуальное создание тестов и включение их в контент
- Интуитивно понятный интерфейс пользователя
- Экспорт контента в заданном формате
- Совместимость экспортируемого контента с системами дистанционного обучения
- Изменение формата, структуры экспортируемого контента в считанные минуты

Lersus. Поддерживаемые форматы

- Поддерживаемые форматы

- Функции LERSUS позволяют экспортировать контент в:
- формате XML для возможности динамически формировать внешний вид контента средствами систему управления контета
- формате HTML для размещения в сети
- формате Adobe Acrobat для печати документа
- текстовом формате
- формате Microsoft Word
- формате который требуется для решения Ваших задач

- С нашим редактором Вы сможете создавать учебные материалы в соответствии со стандартами дистанционного обучения, такими как SCORM, IMS QTi или Moodle GIFT. Если Ваша система дистанционного обучения требует реализации контента в другом формате, то модели LERSUS могут реализовать и этот формат.

- LERSUS поддерживает импорт контента из документов Microsoft Word. Поддерживается импорт текста, рисунков, таблиц, формул и списков. LERSUS импортирует контент с учетом стилей форматирования.

- Если Вы уже имеете созданный контент в этом формате Вы сможете мгновенно преобразовать этот контент в формат модулей LERSUS.

- LERSUS может создавать интерактивную помощь в формате HTML Help, HTML Help 2.0 и документов технической документации.

Пример кода манифеста SCORM-пакета:

```
<?xml version="1.0" encoding="UTF-8"?>
<manifest version="1.3" identifier="8EA33DC1" xmlns="http://www.imsglobal.org/xsd/imscp_v1p1">
  <metadata>
    <schema>ADL SCORM</schema>
    <schemaversion>2004 4th Edition</schemaversion>
  </metadata>
  <organizations default="09B4C179">
    <organization identifier="09B4C179" structure="hierarchical">
      <title>Содержание</title>
      <item identifier="7D841A9D" isvisible="true" identifierref="44D33973">
        <title>Пример объекта SCO, взаимодействующего с LMS</title>
      </item>
    </organization>
  </organizations>
  <resources xmlns:adlcp="http://www.adlnet.org/xsd/adlcp_rootv1p3">
    <resource identifier="44D33973" adlcp:scormType="sco" type="text/html" href="sco.htm">
      <file href="sco.htm" />
    </resource>
  </resources>
</manifest>
```

Пример простейшего SCO: (данная html-страница запрашивает у системы управления обучением имя учащегося, который её открыл)

```
<html>
  <head>
    <script language="javascript">
    function findAPI(win) { //ищем в родительских окнах объект с названием API.
      var findAPITries=0; //будем считать количество попыток, чтобы поиск не был бесконечным.
      while ((win.API_1484_11 == null) && (win.parent != null) && (win.parent != win)) {
        findAPITries++;
        if (findAPITries > 20) return null; //число 20 взято условно, теоретически его может и не хватить.
        win = win.parent;
      }
      return win.API_1484_11;
    }
    function getAPI() { //получаем объект API для текущего SCO.
      var theAPI = findAPI(window); //сначала пробуем искать в родителях текущего окна.
      if ((theAPI == null)) { //если не нашли в родителях текущего окна,
        if ((window.opener != null) && (typeof(window.opener) != "undefined"))
          theAPI = findAPI(window.opener); //то попробуем найти в родителях окна, открывшего текущее.
      }
      return theAPI;
    }
    function start() { //эта функция сработает в момент открытия SCO.
      var api = getAPI();
      if (api!=null) {
        api.Initialize("");
        value=api.GetValue("cmi.learner_name"); //запрашиваем у системы имя учащегося,
        document.write("Имя учащегося: "+value); //и выводим его на экран.
      }
      else document.write("Не удаётся подключиться к API системы.");
    }
    function stop() { //эта функция сработает в момент закрытия SCO.
      var api = getAPI();
      if (api!=null) api.Terminate("");
    }
    </script>
    <title>Пример объекта SCO, взаимодействующего с LMS</title>
  </head>
  <body onLoad="start()" onunload="stop()">
  </body>
</html>
```

Требования к подготовке дистанционных курсов

Рекомендуемая структура дистанционного курса, представляемого на Конкурс:

- Начало дистанционного курса
  - приветствие, знакомство, оформление личной странички обучающегося.
  - вхождение или погружение в тему (определение целей обучения на дистанционном курсе)
  - определение ожиданий обучающихся (планирование личностного смысла обучения и формирование безопасной образовательной среды)
- Работа над темой
  - интерактивная лекция (передача и объяснение педагогом новой информации)
  - механизмы мотивации к активному обучению на курсе
  - проработка содержания темы (индивидуальная или групповая работа обучающихся над темой)
  - комментарии по результатам работы и консультации преподавателя
- Завершение дистанционного курса
  - эмоциональная разрядка, разминка
  - подведение итогов (диагностика по результатам обучения, рефлексия, анализ и оценка участия в дистанционном курсе)
  - планирование применения полученных в ходе обучения на дистанционном курсе ЗУНКов, определение перспектив на продолжение обучения по теме, обратная связь с выпускниками курса.

Дистанционный курс не должен:

- содержать информации, нарушающей авторские права третьих лиц;
- содержать фактографических ошибок и неэтичных компонентов;
- содержать информации, прямо или косвенно призывающей к половой и расовой дискриминации, межнациональной и межрелигиозной розни, призывов к насилию, терроризму, нарушению демократических свобод и ценностей, а также прав граждан.