ПРОГРАММА И МЕТОДИКА КОМПЛЕКСНЫХ ИСПЫТАНИЙ

**«Планировщик выполнения заказов для производственных систем»**

2016 г.

Оглавление

[1.Объект испытаний 3](#_Toc445857552)

[2. Цель испытаний 3](#_Toc445857553)

[3. ОРГАНИЗАЦИЯ ПРОВЕДЕНИЯ ИСПЫТАНИЙ 3](#_Toc445857554)

[4. ТРЕБОВАНИЯ К ПРОГРАММНОМУ ОБЕСПЕЧЕНИЮ 4](#_Toc445857555)

[5. Требования к программной документации 5](#_Toc445857556)

[6. Средства и ПОРЯДОК ИСПЫТАНИЙ 5](#_Toc445857557)

[7. МетодИКА испытаний 6](#_Toc445857558)

**7.1. Построение расписания………………………………………………….…………...6**

**7.2. Проверка построенного расписания с помощью отладчика-профилировщика на наличие ошибок…………………………………..…………………………………………….7**

**7.3. Графическое отображение построенного расписания……………………………8**

[8. ОТЧЕТНОСТЬ 10](#_Toc445857559)

[ПЕРЕЧЕНЬ ССЫЛОЧНЫХ ДОКУМЕНТОВ 10](#_Toc445857560)

Настоящая программа и методика приемочных испытаний определяет порядок проведения комплексных испытаний программного инструмента построения расписаний, разработанных согласно техническому заданию.

# 

**Объект испытаний**

Испытанию подлежит программный инструмент построения расписаний.

**Цель испытаний**

Испытания проводятся с целью проверки программного обеспечения *«Планировщик выполнения заказов для производственных систем»* на соответствие требованиям технического задания для построения оптимального расписания работы оборудований (станков) для выполнения некоторого заказа.

# 

**Организация проведения испытаний**

Испытания проводятся силами и средствами ННГУ и отдела 97100. Научным руководителем проекта является доцент кафедры ИАНИ Старостин Николай Владимирович.

Порядок проведения проверок и отдельные пункты программы могут изменяться или уточняться в процессе испытаний.

**Требования к программному обеспечению**

Испытания проводятся в соответствии с пунктами методики испытаний приведенными в *табл. 1*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № п/п | Наименование проверки | | Пункт ТЗ | Пункт методики |
| 1. | Функции ПО «Планировщик выполнения заказов для производственных систем» | Построение расписания | 3.2.2 | 7.1 |
| Проверка построенного расписания с помощью отладчика-профилировщика на наличие ошибок | 3.3.2 | 7.2 |
| Графическое отображение построенного расписания | 3.4.2 | 7.3 |
|
|
|
|
|
|
|

*Таблица 1.*

# 

**Требования к программной документации**

# 

Программная документация ПО *«Планировщик выполнения заказов для производственных систем»* включает в себя: Техническое задание, Сценарий работы системы, Руководство пользователя.

Методы испытаний включают в себя процедуры проверок каждого из пунктов раздела «Состав и порядок испытаний».

**Средства и порядок испытаний**

Программные средства испытаний ПО «Планировщик выполнения заказов для производственных систем» включают в себя:

* общее программное обеспечение – операционную систему Windows 7 и выше.
* IВМ-совместимый персональный компьютер (ПЭВМ), включающий в себя: процессор Pentium-2.0Hz, не менее; материнскую плату с FSB, ГГц - 5, не менее; оперативную память объемом, 1Гигабайт, не менее.
* установленные программы – Visual Studio 2012 с версией .NET Framework 3.5
* тестовые файлы в формате xml

# 

**Методика испытаний**

В данном пункте описывается схема тестирования функционала согласно требованиям, указанным в ТЗ.

**Построение расписания**

* Запустить командную строку Windows 7
* Перейти в директорию, где хранится файл *BuilderConsole.exe*
* Ввести в командную строку следующую команду:

*.\BuilderConsole.exe name\_of\_input\_dir [name\_of\_output\_dir],* где

*name\_of\_input\_dir* – входная директория, содержащая xml-документы определенного формата *(см. Приложение 1, Приложение 2);*

*name\_of\_output\_dir* – директория, куда необходимо поместить файл-решение (если директория не была указана, то подсистема сохраняет выходной xml-документ в директорию, в которой находится файл программы.).

* Нажать клавишу Enter

Если в указанном пути встречаются недопустимые символы, то выдается сообщение от консоли «Файл не найден»;

Если указанный путь до файла не существует, то выдается сообщение от консоли «Файл не найден»;

Если не указывается входная директория, то выдается сообщение «Неверные параметры команды»;

Если входной директории не существует, то выдается сообщение «Файл не найден»;

Если во входной директории не находится необходимых для построения расписания файлов (или хотя бы одного), выдается сообщение «Файл не найден»;

Если во входной директории лежат xml-файлы с информацией, записанной не по оговоренному формату, то выдается сообщение «Неверный формат исходных данных»;

Если во входной директории лежат файлы не xml расширения, то выдается сообщение «Недопустимое расширение исходных файлов»;

Если указанной выходной директории не существует, то выдается сообщение «Файл не найден»;

Если путь до файла указан верно, входная директория указана верно, в ней лежат необходимые xml-документы с информацией, записанной по оговоренному формату, выходная директория указана верно либо не указана, то тест пройден.

* Ждать завершения работы программы

Если в результате работы программы не был создан выходной файл с расписанием, то тест не пройден;

Если в результате работы программы был создан пустой выходной файл с расписанием, то тест не пройден;

Если в результате работы программы был создан выходной файл с расписанием, где запись расписания не соответствует заранее оговоренному формату представления данных, то тест не пройден;

Если в результате работы программы был создан файл с расписанием, помещенный не в указанную выходную директорию, то тест не пройден.

Если в результате работы программы был создан полноценный выходной файл с расписанием с записью данных, соответствующих заранее оговоренному формату и помещенный в указанную выходную директорию, то тест пройден.

**Проверка построенного расписания с помощью отладчика-профилировщика на наличие ошибок**

* Выполнить пункт методики 7.1
* Запустить командную строку Windows 7, если она не запущена
* Перейти в директорию, где хранится файл *DebuggerConsole.exe*
* Ввести в командную строку следующую команду:

*.\DebuggerConsole.exe name\_of\_input\_dir [name\_of\_output\_dir],* где

*name\_of\_input\_dir* – входная директория, содержащая xml-документы определенного формата (см. Приложение 1, Приложение 2, Приложение 3);

*name\_of\_output\_dir* – директория, куда необходимо поместить log-файл (если директория не была указана, то подсистема сохраняет выходной xml-документ в директорию, в которой находится файл программы.).

* Нажать клавишу Enter

Если в указанном пути встречаются недопустимые символы, то выдается сообщение от консоли «Файл не найден»;

Если указанный путь до файла не существует, то выдается сообщение от консоли «Файл не найден»;

Если не указывается входная директория, то выдается сообщение «Неверные параметры команды»;

Если входной директории не существует, то выдается сообщение «Файл не найден»;

Если во входной директории не находится необходимых для анализа расписания файлов (или хотя бы одного), выдается сообщение «Файл не найден»;

Если во входной директории лежат xml-файлы с информацией, записанной не по оговоренному формату, то выдается сообщение «Неверный формат исходных данных»;

Если во входной директории лежат файлы не xml расширения, то выдается сообщение «Недопустимое расширение исходных файлов»;

Если указанной выходной директории не существует, то выдается сообщение «Файл не найден»;

Если путь до файла указан верно, входная директория указана верно, в ней лежат необходимые xml-документы с информацией, записанной по оговоренному формату, выходная директория указана верно либо не указана, то тест пройден.

* Ждать завершения работы программы

Если в результате работы программы не был создан выходной log-файл, то тест не пройден;

Если в результате работы программы был создан пустой выходной log-файл, то тест не пройден;

Если в результате работы программы был создан выходной log-файл, где запись ошибок расписания не соответствует заранее оговоренному формату представления данных, то тест не пройден;

Если в результате работы программы был создан log-файл, помещенный не в указанную выходную директорию, то тест не пройден.

Если в результате работы программы был создан полноценный выходной log-файл с анализом расписания (с перечнем содержащихся в расписании ошибок и предупреждений), записанный в соответствии заранее оговоренному формату и помещенный в указанную выходную директорию, то тест пройден.

**Графическое отображение построенного расписания**

* Выполнить пункты методики 7.1,7.2;
* Нажать на кнопку «*Выбрать директорию».* Либо файл -> «*Выбрать директорию».*
* Выбрать директорию с входными данными, удовлетворяющими характеристикам (см. Приложение 3 ТЗ), нажать кнопку “*ОК*”.
* Нажать кнопку *«Построить расписание»* Либо файл ->*«Построить расписание»*
* Нажать кнопку *«Анализ расписания»* Либо файл ->*« Анализ расписания»*
* Нажать на любую из секций построенных графиков

Результат испытания считается положительным, если будут выполнены условия:

1. После нажатия кнопки *«Построить расписание»* подсистема построила диаграмму Ганта.
2. При нажатии на любую из секций построенных графиков отобразилась детальная информация по выбранной операции.

**Отчетность**

По всем видам испытаний оформляется протокол испытаний, который подписывается членами комиссии, проводившими испытания.

Протокол испытаний утверждается председателем комиссии.

**Перечень ссылочных документов**

1. ТЗ на научно-исследовательскую работу *«Планировщик выполнения заказов для производственных систем».*
2. ПО *«Планировщик выполнения заказов для производственных систем»*