**7.1. Диалоговый интерфейс**

Прототип программного компонента диалогового интерфейса представляет собой визуальную оболочку, обеспечивающую возможность удобной работы пользователя с системой интеллектуальной поддержки.

Главной функцией компонента диалогового интерфейса является предоставление доступа к функциональным возможностям системы интеллектуальной поддержки. Графический интерфейс пользователя имеет особое значение среди компонентов системы, так как именно с его помощью происходит управление программным комплексом в целом. Для этого принципы его работы должны лежать на поверхности, а детали взаимодействия с нижележащими подсистемами должны быть скрыты от пользователя.

Компонент диалогового интерфейса работает в качестве сетевого web-клиента, агрегирующего функциональность распределенных компонентов, предоставляющих интерфейсы в виде web-сервисов. Прототип компонента разработан в виде «тонкого клиента» с использованием языка программирования C# версии 3.0 с применением ряда специализированных библиотек.

Для удобной организации работы пользователя в web-интерфейсе используется модель проектов, задач и запусков. Эта логика в основном отражена в области дерева проектов (см. рисунок 7.1.1).

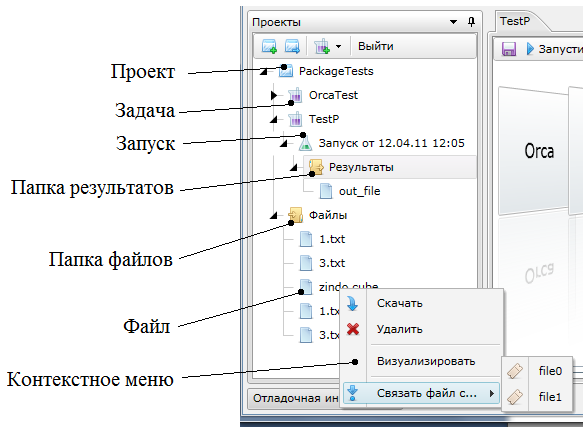


Рисунок 7.1.1 − Элементы дерева проектов

*Проект* – это объединение нескольких вычислительных задач и загруженных пользователем входных файлов. В один момент в web-клиенте может быть загружен только один корневой проект.

*Задача* – это описание вычислительной задачи на языке EasyFlow с возможностью неоднократного запуска. Задача может быть создана как прямым редактированием скрипта, так и выбором из существующих шаблонов или прохождения опроса интеллектуальной системы (см. ниже). Задачи группируются как дочерние элементы корневого проекта в дереве проектов.

*Запуск* – это сущность, фиксирующая факт запуска задачи. Каждая задача может быть запущена несколько раз (например, после изменения некоторых параметров запуска). Запуски группируются как дочерние элементы задачи в дереве проектов.

*Файл* – сущность, представляющая реальный файл, находящийся в удаленном хранилище. Файлы группируются в папки. Проект содержит папку «Файлы», в которую пользователь может закачивать собственные входные данные. Любой запуск содержит папку «Результаты», в которую собираются выходные файлы данного запуска.

Каждый элемент дерева проектов имеет собственное контекстное меню, вызываемое нажатием на нем правой клавишей мыши. В нем содержатся действия, присущие тому или иному элементу (создание, удаление, запуск, открытие и т.д.).

Помимо главного окна web-интерфейс может создавать дочерние всплывающие окна для выполнения различных действий (например, прохождение опроса интеллектуальной системы для уточнения проектного решения). Для взаимодействия с пользователем web-интерфейс использует стандартные компоненты: кнопки, контекстные меню, поля ввода, панели докинга и т.д.

Пример web-интерфейса с деревом проекта для сервиса моделирования качки судна на нерегулярном волнении был показан на рисунке 6.1.5. В ходе работы сервиса экземпляры пакета ShipXDS запускаются на исполнение на выбранных вычислительных ресурсах. В процессе исполнения осуществляется мониторинг их состояния и по мере освобождения ресурсов на них автоматически запускаются новые экземпляры в соответствии с построенным планом.

На рисунке 7.1.2 приведен образ WF исполнения композитного приложения для сервиса моделирования качки судна на нерегулярном волнении.

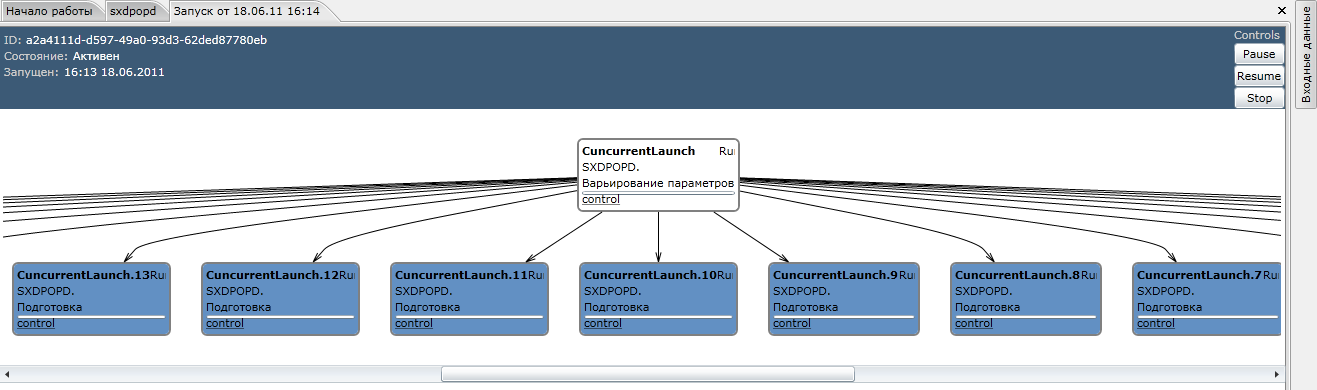


Рисунок 7.1.2 − Отображение образа WF исполнения композитного приложения моделирования качки судна на нерегулярном волнении в диалоговом интерфейсе