

УДК 004.9

РАЗРАБОТКА СИСТЕМЫ ТЕСТИРОВАНИЯ ДЛЯ ОТРАБОТКИ ДЕЙСТВИЙ В НЕШТАТНЫХ СИТУАЦИЯХ СОТРУДНИКАМИ ИРКУТСКОЙ НЕФТЯНОЙ КОМПАНИИ

Попков Сергей Сергеевич¹, Сычёв Пётр Павлович²

¹Магистрант;

Государственный университет «Дубна»;

141980, Московская обл., г. Дубна, ул. Университетская, 19;

e-mail: sspopkov@list.ru.

²Доцент;

Государственный университет «Дубна»;

141980, Московская обл., г. Дубна, ул. Университетская, 19;

e-mail: sychov@dubna.ru.

Статья посвящена анализу существующих инженерно-технологических процессов в нефтедобывающих комплексах Ярактинского нефтегазоконденсатного месторождения (ЯНГКМ), принадлежащего Иркутской нефтяной компании (ИНК). Рассматривается математическая модель системы сбора и контроля данных показателей нефтедобывающего оборудования. На основе математической модели предметной области и функциональных требований предлагается проектное решение для системы тестирования. В качестве инструмента разработки frontend-части системы был выбран веб-фреймворк Nuxt.js, а для backend-части – фреймворк Node.js. Приводятся сценарии использования и тестирования системы, а также результаты внедрения в компании. Сценарий нештатной ситуации создается на базе выбранной мнемосхемы (или нескольких мнемосхем), либо на базе существующего сценария. В итоге была разработана уникальная система, целью использования которой ставится повышение количества успешно ликвидированных нештатных ситуаций. Система позиционируется как веб-приложение. Уникальность исследования заключается в новом способе проведения тестирования специалистов по ликвидации аварий в среде веб-приложения, моделирующего поведение реального оборудования ЯНГКМ. Разработанное веб-приложение, отвечающее требованиям заказчика, было протестировано, задокументировано и введено в эксплуатацию в Иркутской нефтяной компании.

Ключевые слова: моделирование процессов, тестирование, веб-приложение.

Для цитирования:

Попков С. С., Сычёв П. П. Разработка системы тестирования для отработки действий в нештатных ситуациях сотрудниками иркутской нефтяной компании // Системный анализ в науке и образовании: сетевое научное издание. 2021. № 2. С. 9–17. URL : <http://sanse.ru/download/432>.

DEVELOPMENT OF A TESTING SYSTEM FOR PROCESSING ACTIONS IN EMERGENCY SITUATIONS BY EMPLOYEES OF THE IRKUTSK OIL COMPANY

Popkov Sergey S.¹, Sychov Peter P.²

¹Master's Degree student;

Dubna State University;

19 Universitetskaya Str., Dubna, Moscow region, 141980, Russia;

e-mail: sspopkov@list.ru.

²Associate professor;

Dubna State University;

19 Universitetskaya Str., Dubna, Moscow region, 141980, Russia;

e-mail: sychov@dubna.ru.

The article is devoted to the analysis of existing engineering and technological processes in the oil-producing complexes of the Yarakinskoye oil and gas condensate field (YOGCF), owned by the Irkutsk Oil Company (Inc). A mathematical model of the system for collecting and monitoring data indicators of oil-producing equipment is considered. Based on the mathematical model of the subject area and functional requirements, a design solution for the testing system is proposed. The Nuxt.js web framework was chosen as a development tool for the frontend part of the system, and for the backend part-the framework Node.js. The scenarios of using and testing the system, as well as the results of implementation in the company are presented. The emergency scenario is created on the basis of the selected mnemonic scheme (or several mnemonic schemes), or on the basis of an existing scenario. As a result, a unique system was developed, the purpose of which is to increase the number of successfully eliminated emergency situations. The system is positioned as a web application. The uniqueness of the study lies in a new way of testing accident response specialists in the environment of a web application that simulates the behavior of real YOGCM equipment. The developed web application that meets the customer's requirements, was tested, documented and put into operation at the Irkutsk Oil Company.

Keywords: process modeling, testing, web application.

For citation:

Popkov S., Syrov P. Development of a testing system for processing actions in an emergency situations by employees of the Irkutsk Oil Company. System Analysis in Science and Education, 2021;(2):9–17(In Russ). Available from: <http://sanse.ru/download/432>.

Введение

Основным месторождением Иркутской нефтяной компании (ИНК) является Ярактинское нефтегазоконденсатное месторождение, расположенное в Иркутской области. В этом месторождении добывается примерно 80 % углеводородного сырья компании [1]. Технологический комплекс данного месторождения состоит из дожимной насосной станции (ДНС), установки подготовки нефти (УПН) и системы очистки нефти (СОН). Управление оборудованием осуществляется системой на базе концепции SCADA. Рассматриваемая система тестирования предназначена для сотрудников Ярактинского месторождения [2].

Система тестирования сотрудников ИНК представляет собой веб-приложение, которое предоставляет возможность обучить персонал компании последовательности действий в различных чрезвычайных ситуациях.

В основе системы лежит такая концепция программных продуктов, как SCADA. Принципиальным решением в рамках данной концепции является сбор, обработка, отображение и архивирование информации в режиме реального времени [3]. Такой подход к управлению информацией применяется во всех отраслях хозяйства, где требуется обеспечить операторский контроль за технологическими процессами в реальном времени. SCADA-система может являться частью автоматизированной системы управления технологическими процессами (АСУТП), системы экологического мониторинга, научного эксперимента, автоматизации задания и т. д.

Применительно к нефтегазовой отрасли промышленности, SCADA-система выполняет функции мониторинга оборудования в режиме реального времени, управления оборудованием, отображения текущего состояния ресурсов в виде мнемосхемы, регистрирования нештатных ситуаций и записи журнала произошедших событий [4].

Структура системы разделена на два крупных блока – блок администрирования и блок тестирования. Акторами системы являются Администратор и Пользователь. Функционирование системы осуществляется на базе мнемосхем SCADA для упомянутых технологических комплексов – ДНС, УПН и СОН. Нештатные ситуации представляются в виде сценариев поведения оборудования на мнемосхемах – изменении показателей различных датчиков, цветовой индикации, отображении событий в консоли оператора и т. д. Для удобства администрирования представлен также блок статистики результатов прохождений сценариев пользователями системы.

Математическая модель

Сценарий нештатной ситуации создается на базе выбранной мнемосхемы (или нескольких мнемосхем), либо на базе существующего сценария. Прогресс изменений всех показателей на схеме фиксируется с помощью графа состояний. В рамках одного сценария Администратор задает несколько «узловых точек» на данном графе, а также ребра, которые связывают эти узловые точки **[Ошибка! Источник ссылки не найден.]**. Каждый узел графа представляет собой совокупность параметров всех элементов, расположенных на мнемосхеме (значение, цвет, анимация), а также уведомление на панели статусов в определенный временной «срез». Каждое ребро представляет собой промежуток времени, проходящий между «срезами». Администратор добавляет/редактирует/удаляет элементы графа состояний и параметры элементы мнемосхемы для каждого узла.

Для сценария можно задать текстовый тест, прикрепив его к какому-либо узлу графа состояний:

- Указать количество возможных вариантов причины аварии, заполнить поля ответов (свои или из справочников) и указать правильный вариант.
- Указать количество возможных действий при аварии, заполнить поля ответов и указать правильную очередность ответов. (Предлагать наполнять поля из справочников, либо свои). Очередность ответов указывается с помощью графа, где нумерованный узел – действие, ориентированное ребро – верный порядок перехода от одного действия к другому.
- Указать критерии оценивания (таблица: количество ошибок – оценка).

Расчёт дискретного времени

Так как рассмотренное понятие «узловых» точек на графе состояний системы не имеет привязки ко времени, необходимо ввести понятие расчета дискретного времени. С помощью введения этой формулы будет возможно проведение математического моделирования. Укажем минимальный возможный момент времени t_{min} , количество кадров в секунду fps и запишем формулу расчёта интервала обновления (1):

$$I = \frac{t_{min}}{fps}, \quad (1)$$

Для t_{min} , равного 1000 мс (1 секунда), интервал обновления с шагом в 1 кадр в секунду будет равен 1000 мс. Интервал обновления является шагом моделирования. Введем также понятие скорости воспроизведения моделирования – $speed$. Запишем итерационную формулу расчета времени воспроизведения (2):

$$t_i = t_{i-1} + I \times speed, \quad (2)$$

где требуется указание начального момента времени t_0 . Примем его равным 0. Общее модельное время обозначим через t_{max} (3).

$$t_i, t_{min} \leq t_i \leq t_{max}. \quad (3)$$

Введем понятие количества датчиков системы S (4):

$$s_j, j \leq S. \quad (4)$$

Для датчика $s_j, j \in S$ возникает событие $s_j(t_i)$, при котором меняются величины показателей датчика s_j . Таким образом, состояние системы M в момент времени t_i , с учетом рассмотренных ранее формул 1 и 2 может быть рассчитано по формуле 5:

$$M(t_i) = \sum_{i=0}^{\frac{t_{max}}{I}} \sum_{j=0}^S s_j(t_i). \quad (5)$$

Полученная формула представляет собой искомый процесс моделирования показателей датчиков оборудования во время сценария. Рассмотрим далее более подробно механизм расчета показателя датчика s_j в момент времени t_i .

Алгоритм обновления показателей датчиков

Для расчета текущего состояния показателей датчика $s_j(t_i)$ введем понятия предыдущего состояния датчика: $s_j(t_{i-1})$, и последующего: $s_j(t_{i+1})$. Таким образом, требуется также задать граничные значения показателей. Будем считать нулевое (стандартное) значение показателей начальным условием. Обозначим его как $s_j(t_{min})$. Рассмотрим на формуле 6 случай, если для показателей датчика больше не задано событий, а модельное время еще не закончилось.

$$s_j(t_i) = s_j(t_{max}). \quad (6)$$

Так, даже если для показателей датчика не задано ни одного события, то по формуле 6 для каждого временного отрезка оно будет считаться как значение в момент завершения моделирования, которое в таком случае равно нулевому значению датчика (7):

$$s_j(t_i) = s_j(t_{min}) = s_j(t_{max}), \quad (7)$$

если $\forall t_i s_j$ – не задано.

Рассмотренные математические выкладки определяют значения показателей датчиков только в тех случаях, если для $t_i \exists s_j$. Однако по такой логике необходимо для каждого момента времени задать значение показателей датчика. Например, моделирование линейного увеличения показателей датчика на большом объеме модельного времени не представляется возможным. Решением данной проблемы может быть введение линейной зависимости значений показателей датчиков между соседними событиями на модельном времени. Введем понятия времени предыдущего события t_{prev} и времени последующего события t_{next} , которые были заданы как начальные условия для определенного датчика s_j . То есть, для этих величин были заданы $s_j(t_{prev})$ и $s_j(t_{next})$. В том случае, если для датчика задано всего одного события в момент времени t_{next} , то $t_{prev} = t_{min}$. Если следующее событие не определено, $t_{next} = t_{max}$. Рассмотрим измененное представление события на формуле 8, учитывающее возможность линейного изменения показателей:

$$s_j(t_i, \Delta t), \quad (8)$$

где Δt вычисляется по формуле 9.

$$\Delta t = \frac{t_i - t_{prev}}{t_{next} - t_{prev}}. \quad (9)$$

Таким образом, Δt представляет собой процент изменения величины показателя датчика от предыдущего события к последующему. Кроме вышеуказанного, датчик может иметь накопительный тип, а также в системе должен учитываться сбой работы АСУТП. В первом случае показатель датчика инкрементируется единицей, а во втором показатель не изменяется во времени, эмулируя ошибку.

Проектное решение

Рабочая область системы включает в себя главный экран, на котором размещена мнемосхема, панель с перечнем мнемосхем, панель с перечнем событий, панель управления/настройки сценария.

Мнемосхема отображает все задействованные объекты, показания их датчиков и цветовую индикацию. Цветовая индикация изменяется в зависимости от показателей датчиков, относительно настроек, указанных при настройке сценария. Цветовая индикация клапанов/арматуры отображает статус элемента:

- Серый цвет – используется для статических изображений и отображения параметров, значение которых в данный момент несущественно, не используется или недостоверно, а также для индикации выключенного состояния технологических агрегатов (механизмов);

- Зеленый цвет – нормальное значение параметра; технический агрегат исправен и включен;
- Желтый цвет – используется для предупредительной сигнализации значений параметров, вывода предупреждающих сообщений;
- Красный цвет – используется для аварийной сигнализации значений параметров, вывода сообщений о неисправностях или авариях в системе, а также для отображения неисправных механизмов.

Значения уровней в Системе отображаются в виде столбчатых диаграмм, высота которых соответствует текущему значению уровня в диапазоне 0-100%.

Мнемосхема представляет собой подложку с размещенными в ней вырезами, в которых размещаются виджеты, отображающие симуляцию значений соответствующих датчиков. Показатели виджетов изменяются в соответствии со значениями, вводимыми Администратором при создании сценария. Если показатели с течением времени изменяются, их значение изменяется линейно. Если показатель виджета неизменен, его значение колеблется по синусоиде с амплитудой в % от значения.

Панель «Перечень событий» отображает происходящие с течением эмуляции события в виде списка с цветовой индикацией (цвета изменяются в зависимости от важности событий. Важность событий указывается при настройке соответствующего справочника).

Все сообщения разбиты на несколько классов:

- Аварийные – превышение параметром аварийной границы, неисправность механизма и т.д. Цвет фона сообщения – красный.
- Предупредительные – превышение параметром предупредительной границы. Цвет фона – желтый.
- Диагностические – сообщения о неисправностях модулей, датчиков, ошибках работы программного обеспечения и пр. Цвет фона – черный.
- Каждое сообщение состоит из нескольких текстовых полей:
 - дата – дата фиксирования сообщения системой;
 - время – время фиксирования сообщения системой;
 - элемент – элемент, к которому относится зафиксированное событие;
 - событие – описание события (текст сообщения);
 - класс – класс сообщения (аварийное, предупредительное и т.д.).

Показания датчиков, список событий и цветовая индикация задаются Администратором при создании/настройке сценария мнемосхемы.

Диаграммы вариантов использования

Рассмотрим диаграмму вариантов использования для Администратора (см. рис. 1). Выполнение прецедентов: «Пройти тестирование», «Редактировать сценарий», «Редактировать датчик» и «Редактировать пользователя» требует обязательного выполнения прецедентов: «Выбрать сценарий», «Выбрать датчик» и «Выбрать пользователя», соответственно.

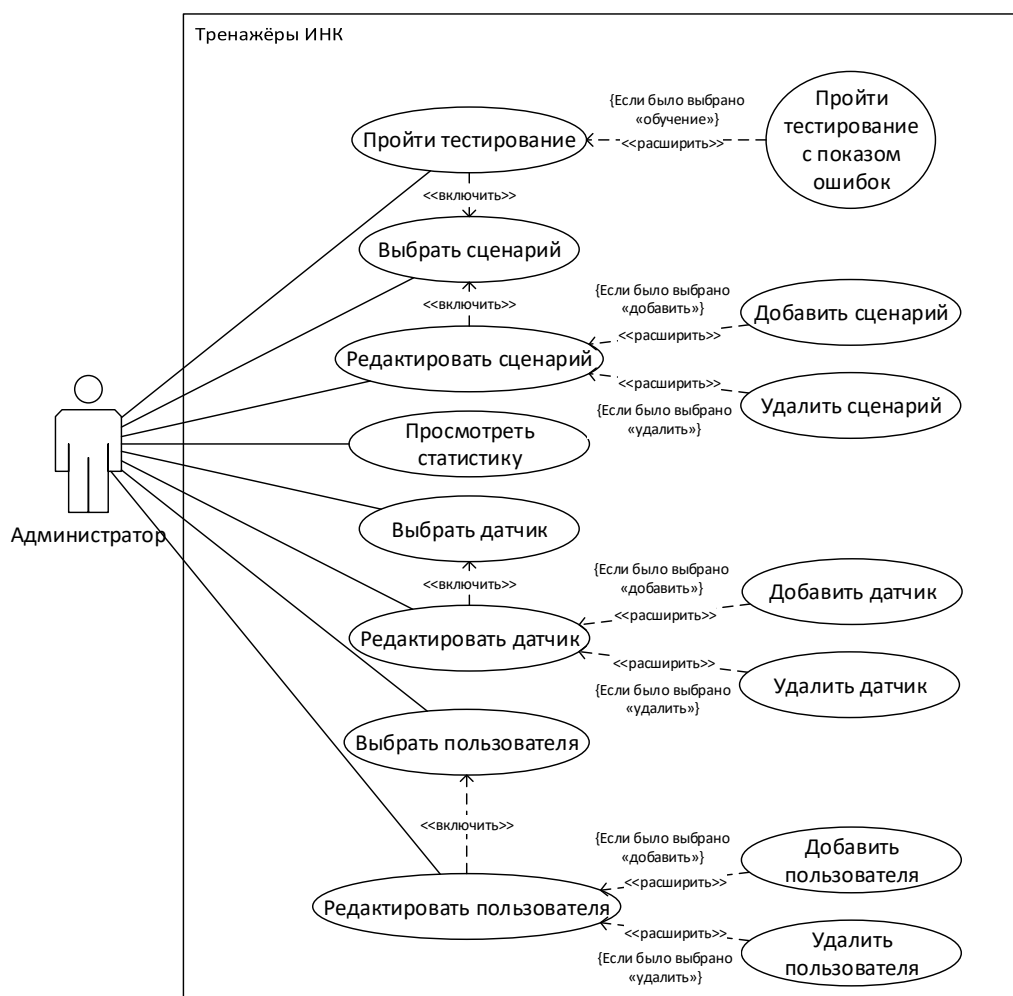


Рис. 1. Диаграмма вариантов использования для Администратора

Рассмотрим диаграмму вариантов использования для Пользователя (см. рис. 2).

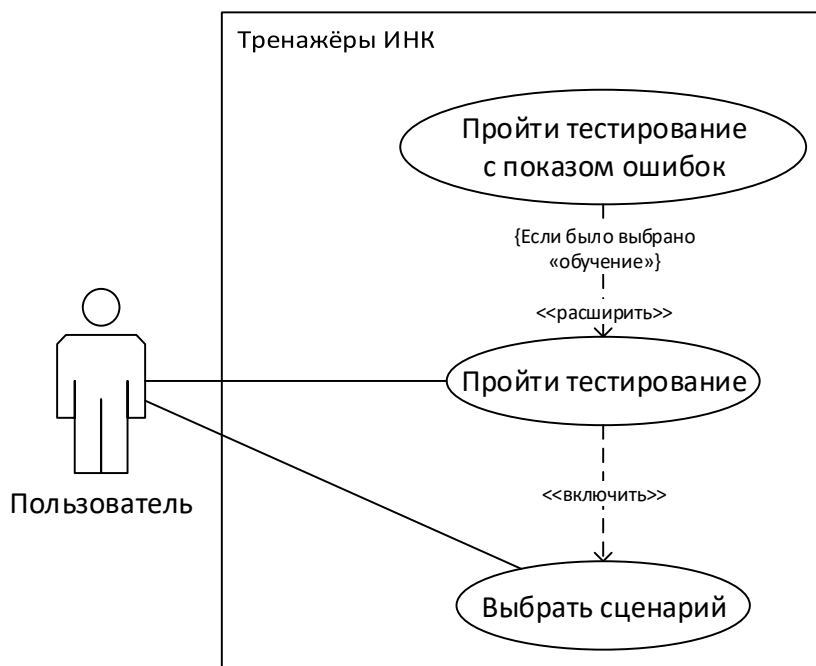


Рис. 2. Диаграмма вариантов использования для Пользователя

На диаграмме (рис. 2) показаны следующие прецеденты использования: «Пройти тестирование», «Выбрать сценарий» и «Пройти тестирование с показом ошибок». Выполнение прецедента «Пройти тестирование» требует обязательного выполнения прецедента «Выбрать сценарий». Прецедент «Пройти тестирование с показом ошибок» расширяет прецедент «Пройти тестирование» при выполнении логического условия «Если было выбрано “обучение”».

Ролевая модель программной системы

Доступность функций системы в ролевой структуре представлена на таблице 1.

Табл. 1. Ролевая модель системы

Сценарий	Пользователь	Администратор
Пройти тестирование	+	+
Пройти экзамен	+	+
Пройти сценарий в режиме обучения	+	+
Редактировать сценарий		+
Редактировать сценарий		+
Редактировать датчик		+
Редактировать мнемосхему		+
Просмотреть статистику прохождений		+

Внедрение системы

Система позиционируется как веб-приложение. В качестве инструмента разработки *frontend*-части системы был выбран веб-фреймворк *Nuxt.js*, а для *backend*-части – фреймворк *Node.js*.

Ввод в эксплуатацию

Во время реализации поставленных задач система своевременно тестировалась на предмет соответствия заявленным и согласованным требованиям. После того, как весь функционал, описанный в функциональных требованиях, был завершен и протестирован, состоялся запуск системы для проверки работоспособности, и удовлетворению этих требований. Для этого была поднята инфраструктура, обеспечивающая непрерывную разработку и выкладку приложения.

Блок работы со сценариями позволяет задавать поведение размещенных на мнемосхемах элементов с течением времени, вывод предупреждающих сообщений, настройку теста для проверки экзаменуемого (см. рис. 3).

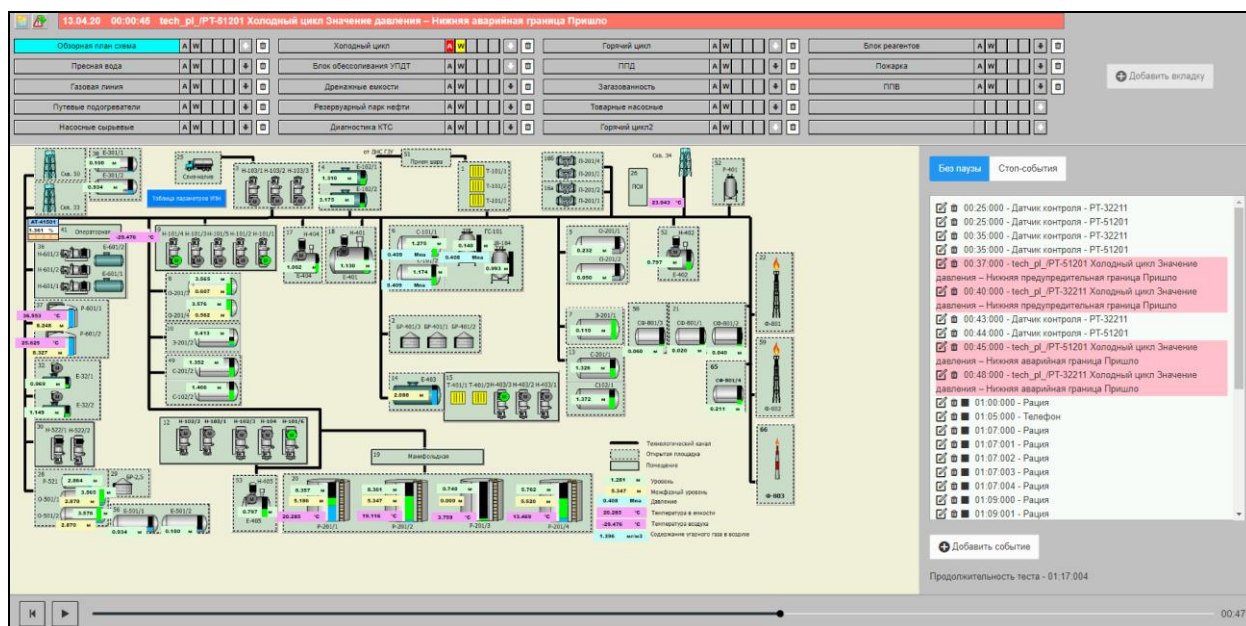


Рис. 3. Настройка сценария

В зависимости от выбранного режима прохождения система отображает соответствующий вид страницы сценария (см. рис. 4, 5).

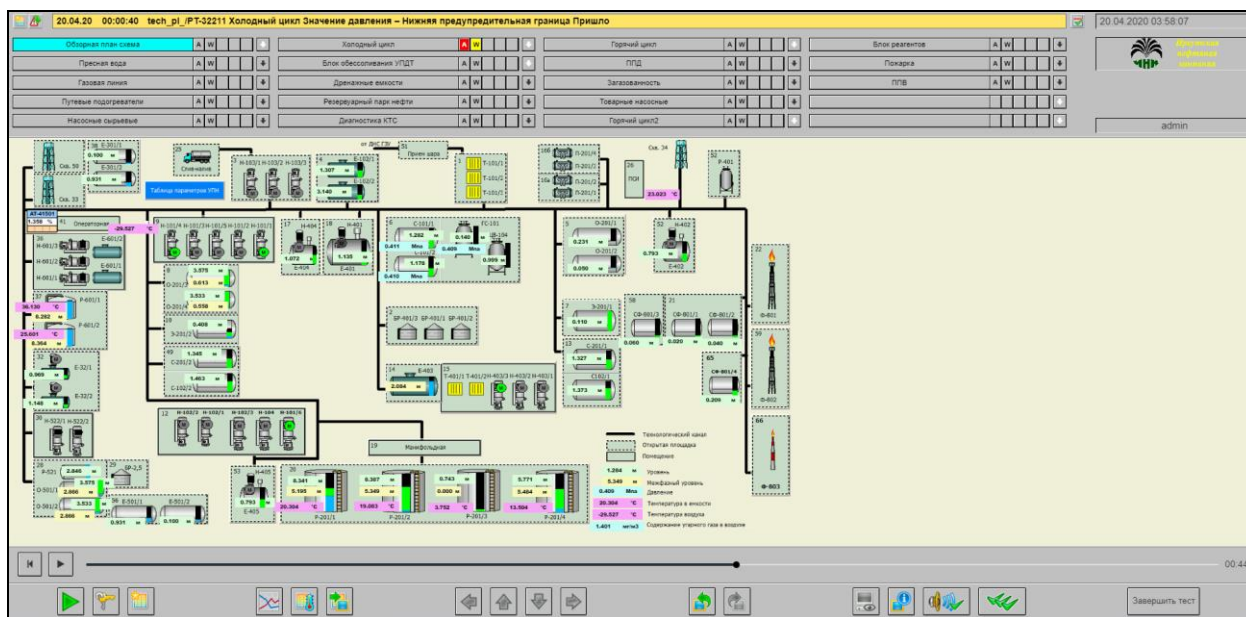


Рис. 4. Прохождение сценария в режиме обучения

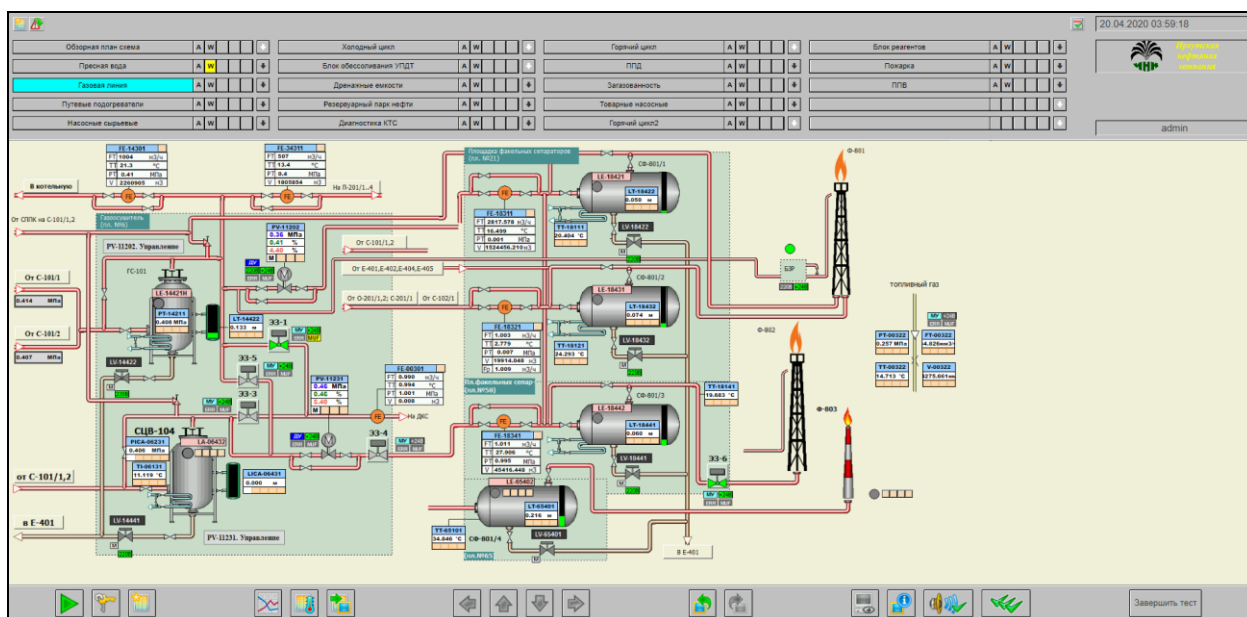


Рис. 5. Прохождение сценария в режиме экзамена

Заключение

Разработанное веб-приложение отвечает требованиям заказчика, протестировано, задокументировано и введено в эксплуатацию в Иркутской нефтяной компании. На данный момент ведется фаза активного использования приложения с его сопровождением.

Уникальность исследования заключается в новом способе проведения тестирования специалистов по ликвидации аварий в среде веб-приложения, моделирующего поведение реального оборудования ЯНГКМ. В итоге была разработана уникальная система, целью использования которой ставится повышение количества успешно ликвидированных нештатных ситуаций.

Список литературы

1. История – ИНК // ИНК. 20 Лет. Сибирская история / ООО «ИНК», 2021. — URL : <https://irkutskoil.ru/about/history/>.
2. Бурцев С. А., Черданцев А. Оценка воздействия на окружающую природную и социальную среду при разработке Ярактинского месторождения (нефтяная и газовая части) : Исполнительное резюме. — Иркутск : ERM Eurasia : Иркутская нефтяная компания, 2008. — 23 с.
3. Sailawi, W. SCADA System and Industrial Network for Petroleum Industry. — Grenoble Alpes University, 2017. — 21 с.
4. Елизаров И. А., Третьяков А. А., Пчелинцев А. Н. и др. Интегрированные системы проектирования и управления: SCADA-системы: учебное пособие. — Тамбов: Изд-во ФГБОУ ВПО «ТГТУ», 2015. — 160 с. — ISBN 978-5-8265-1469-6.
5. Boyer S. A. SCADA Supervisory Control and Data Acquisition. — International Society of Automation, 2010. — 179 с.