МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ   
РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное автономное образовательное учреждение высшего образования «Самарский национальный исследовательский университет имени академика С.П. Королева (Самарский университет)»

Институт информатики и кибернетики

Кафедра программных систем

**ОТЧЕТ**  
  
 по лабораторному практикуму по дисциплине

«Разработка человеко-машинного интерфейса

в системах реального времени»

Вариант № 1

Обучающийся группы 6131-020402D В.Д. Гижевская

Обучающийся группы 6131-020402D Д.А. Кремнёв

Руководитель Л.С. Зеленко

Самара 2023

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ   
РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное автономное образовательное учреждение высшего образования «Самарский национальный исследовательский университет имени академика С.П. Королева (Самарский университет)»

Институт информатики и кибернетики

Кафедра программных систем

ЗАДАНИЕ

на лабораторный практикум по дисциплине

«Разработка человеко-машинного интерфейса   
в системах реального времени»

обучающимся в группе № 6131-020402D

В.Д. Гижевской

Д.А. Кремнёву

1. Задание**:** С помощью SCADA-системы WinCC OA 3.17 разработать программное обеспечение, моделирующее процесс восстановления и пастеризации сока, предусмотреть работу купажных чанов и нагревательного элемента при возникновении нештатных ситуаций
2. Исходные данные к проекту**:** см. приложение к заданию
3. Перечень вопросов, подлежащих разработке:
   1. Произвести анализ предметной области: изучить основные принципы процесса восстановления и пастеризации сока, состав оборудования и его характеристики, а также поведение оборудования в режиме реального времени
   2. Разработать информационную модель реального объекта, определить точки данных, команды управления, аварийные ситуации
   3. Разработать прототипы экранных форм (мнемосхем)
   4. Разработать алгоритмы управления процессом восстановления и пастеризации сока
   5. Реализовать скрипты для элементов управления, провести тестирование и отладку
   6. Оформить документацию
4. Перечень графических разработок:
   1. Мнемосхема системы
5. Календарный план выполнения работ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № п/п | Содержание работы по этапам | Объем этапа в % к общему объему проекта | Срок  окончания | Фактическое выполнение |
| 1 | Оформление технического задания и его утверждение | 5 | 13.03.2023 |  |
| 2 | Описание и анализ предметной области | 15 | 13.03.2023 |  |
| 3 | Проектирование системы | 25 |  |  |
| 3.1 | Разработка информационной модели объекта | 10 | 20.03.2023 |  |
| 3.2 | Разработка прототипа интерфейса пользователя | 10 | 27.03.2023 |  |
| 4 | Реализация проекта, разработка контрольных примеров. Предъявление реализации руководителю | 45 | 20.05.2023 |  |
| 5 | Корректировка проекта и оформление документации проекта. Защита проекта с представлением презентации. | 10 | 05.06.2023 |  |

Задание принял  
 к исполнению 03.03.2023 В.Д. Гижевская

03.03.2023 Д.А. Кремнёв

ПРИЛОЖЕНИЕ  
к заданию на лабораторный практикум  
обучающимся в группе № 6131-020402D

В.Д. Гижевской

Д.А. Кремнёву

Исходные данные к проекту:

1. Характеристика объекта автоматизации:

## объект автоматизации: оборудование для купажирования и пастеризации сока;

## виды автоматизируемой деятельности:

* + процесс авторизации пользователей системы;
  + процесс моделирования работы оборудования при купажировании и пастеризации сока;
  + процесс управления оборудованием купажирования и пастеризации сока при возникновении нештатных ситуаций;
  + процесс визуализации работы оборудования купажирования и пастеризации;
  + процесс обработки аварийных сообщений;

## количество ролей пользователей – 3;

## минимальная длина пароля – 4 (символа);

## максимальная длина пароля – 18 (символов);

## минимальная длина логина – 4 (символа);

## максимальная длина логина – 20 (символов);

## количество режимов работы оборудования – 2;

## количество элементов оборудования – 6;

## количество купажных чанов – 3;

## количество датчиков уровня – 3;

## количество датчиков температуры – 1;

## количество нагревательных элементов – 1;

## количество запорных клапанов – 3;

## количество регулируемых клапанов – 2;

## минимальная температура жидкости в купажном чане – 15°C;

## максимальная температура жидкости в купажном чане – 95°C;

## минимальный угол наклона регулируемого клапана – 0°;

## максимальный угол наклона регулируемого клапана – 180°;

## минимальная скорость мешалки – 3 об/мин;

## максимальная скорость мешалки – 120 об/мин.

1. Требования к информационному обеспечению:
2. информационное обеспечение разрабатывается на основе следующих источников:
   * описание процесса восстановления и пастеризации сока [Электронный ресурс]. URL: https://ru.wikipedia.org/wiki/%D0% A1%D0%BE%D0%BA (дата обращения: 12.03.2023);
3. структуры точек данных определяются в процессе проектирования;
4. структура мнемосхемы определяется в процессе проектирования.
5. Требования к техническому обеспечению:
6. тип ЭВМ – IBM PC совместимый;
7. монитор с разрешающей способностью не ниже 800 х 600;
8. манипулятор – мышь;
9. технические характеристики определяются в процессе выполнения проекта.
10. Требования к программному обеспечению:
11. тип операционной системы – Windows 7 и выше;
12. SCADA-система – WinCC OA 3.17;
13. язык программирования – CTRL;
14. среда программирования – Vision.
15. Общие требования к проектируемой системе:

5.1 Функции, реализуемые системой:

1. функции системы:
   * аутентификация пользователя в системе, настройка интерфейса пользователя на заданную роль;
   * моделирование работы оборудования в автоматическом режиме:
   * запуск оборудования;
   * обеспечение подачи продукта в ёмкости;
   * регулирование температуры нагревательного элемента;
   * регулирование скорости мешалки;
   * моделирование возникновения нештатных ситуаций (отказов) в режиме реального времени и выдача аварийных сообщений;
   * визуализация процесса работы оборудования:
   * отображение штатной работы оборудования;
   * отображение аварийных сообщений;
   * выдача справочной информации;
2. функции администратора:
   * авторизация пользователя в системе (ввод логина/пароля);
   * настройка параметров оборудования (точек данных):
   * задать угол открытия регулируемого клапана;
   * задать количество оборотов для мешалки;
   * задать температуру нагревательного элемента;
   * работа с оборудованием (ручной режим):
   * открыть клапан;
   * закрыть клапан;
   * запустить мешалку;
   * остановить мешалку;
   * запустить нагревательный элемент;
   * остановить нагревательный элемент;
   * квитировать аварийные сообщения;
   * просмотр архивных параметров за требуемый интервал времени;
   * просмотр справочной информации;
3. функции оператора:
   * авторизация пользователя в системе (ввод логина/пароля);
   * работа с оборудованием (ручной режим):
   * открыть клапан;
   * закрыть клапан;
   * запустить мешалку;
   * остановить мешалку;
   * запустить нагревательный элемент;
   * остановить нагревательный элемент;
   * квитировать аварийные сообщения;
   * просмотр архивных параметров за требуемый интервал времени;
   * просмотр справочной информации;
4. функции гостя:
   * наблюдение за ходом технологического процесса в автоматическом режиме.

5.2 Технические требования к системе:

1. режим работы ‑ диалоговый;
2. система должна удовлетворять санитарным правилам и нормам  
    СанПин 2.2.2./2.4.2198-07;
3. условия работы средств вычислительной техники (содержание вредных веществ, пыли и подвижность воздуха) должны соответствовать ГОСТ 12.1.005, 12.01.007;
4. температура окружающего воздуха – 15-35°С;
5. влажность воздуха – 45-75%.

Руководитель   
проекта Л.С. Зеленко

Задание принял  
к исполнению 03.03.2023 В.Д. Гижевская

03.03.2023 Д.А. Кремнёв

РЕФЕРАТ

Отчет 58 с, 33 рисунков, 7 таблиц, 13 источников, 2 приложения.

СОК, ПАСТЕРИЗАЦИЯ, КОНЦЕНТРАТ, ВОДА, ПАСТЕРИЗАЦИЯ, КУПАЖНЫЙ ЧАН, ПЕРЕМЕШИВАНИЕ, НАПИТОК, ЁМКОСТЬ, МЕШАЛКА, ПРОДУКТ

Объектом автоматизации является оборудование для купажирования и пастеризации сока.

Во время лабораторного практикума разработана система моделирования работы процесса восстановления и пастеризации сока с помощью SCADA-системы WinCC OA 3.17. В системе реализованы три роли пользователей: администратор, оператор и гость. Администратор выбирает режим управления производственным процессом: ручной или автоматический, настраивает параметры оборудования. Оператор наблюдает за ходом процесса и при возникновении аварийных ситуаций включается в процесс работы оборудования. Гость только наблюдает за ходом процесса.

В системе имеется возможность задать параметры для всех видов оборудования, наблюдать за процессом восстановления и пастеризации сока, налаживание работы при возникновении нештатных ситуаций.

Программное обеспечение разработано на языке CTRL в среде WinCC OA 3.17 и функционирует под управлением операционной системы Windows 7 и выше.

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 11](#_Toc136802553)

[1 Описание предметной области 13](#_Toc136802554)

[1.1 Основные понятия и определения 13](#_Toc136802555)

[1.2 Описание процесса восстановления сока 14](#_Toc136802556)

[1.3 Описание оборудования, задействованного в процессе восстановления сока 14](#_Toc136802557)

[1.3.1 Описание купажных ёмкостей 15](#_Toc136802558)

[1.3.2 Описание нагревательного элемента 16](#_Toc136802559)

[1.3.3 Описание клапанов 19](#_Toc136802560)

[1.4 Постановка задачи 21](#_Toc136802561)

[2 Проектирование системы 25](#_Toc136802562)

[2.1 Разработка информационной модели объекта 25](#_Toc136802563)

[2.1.1 Описание точек данных (Data Points) 25](#_Toc136802564)

[2.2 Проектирование интерфейса пользователя 26](#_Toc136802565)

[3 Реализация системы 33](#_Toc136802566)

[3.1 Настройка элементов точек данных 33](#_Toc136802567)

[3.2 Настройка исходных значений 34](#_Toc136802568)

[3.3 Общие настройки объекта 34](#_Toc136802569)

[3.4 Обработка сообщений (alert) 35](#_Toc136802570)

[3.5 Разработка и описание графического интерфейса 35](#_Toc136802571)

[3.5.1 Описание графических элементов мнемосхемы 35](#_Toc136802572)

[3.5.2 Начало работы 37](#_Toc136802573)

[3.5.3 Режим администратора 38](#_Toc136802574)

[3.5.4 Режим оператора 39](#_Toc136802575)

[3.5.5 Режим гостя 43](#_Toc136802576)

[3.5.6 Обработка внештатных ситуаций 43](#_Toc136802577)

[3.5.7 Просмотр параметров оборудования 44](#_Toc136802578)

[Заключение 45](#_Toc136802579)

[Список использованных источников 46](#_Toc136802580)

[ПРИЛОЖЕНИЕ Б Сценарии (скрипты управления) 51](#_Toc136802581)

ВВЕДЕНИЕ

SCADA (Supervisory Control And Data Acquisition – диспетчерское управление и сбор данных) – программная система, предназначенная для разработки или обеспечения работы в реальном времени систем сбора, обработки, отображения и архивирования информации об объекте мониторинга или управления [1].

Задачи SCADA-систем:

* обмен данными с «устройствами связи с объектом» в реальном времени через драйверы;
* обработка информации в реальном времени;
* логическое управление;
* отображение информации на экране монитора в удобной и понятной для человека форме;
* ведение базы данных реального времени с технологической информацией;
* аварийная сигнализация и управление тревожными сообщениями;
* подготовка и генерирование отчетов о ходе технологического процесса;
* осуществление сетевого взаимодействия между SCADA-станциями;
* обеспечение связи с внешними приложениями.

Использование SCADA-систем в системах автоматизации производственных процессов позволяет увеличить производительность операторов существенно уменьшить процедурные ошибки операторов, а также свести к нули критические (некорректируемые) ошибки операторов [1].

Во время лабораторного практикума необходимо разработать систему моделирования процесса восстановления и пастеризации сока, которая позволит симулировать данный процесс и смоделировать его поведение в реальном производстве, управлять различными параметрами, влияющими на работу, а также найти узкие места, которые могут привести к ошибкам операторов.

Система будет разрабатываться с помощью SCADA-системы SIMATIC WinCC OA 3.17, которая позволяет провести эффективный инжиниринг и обеспечить максимальный уровень надёжности и безопасности.

1. Описание предметной области
   1. Основные понятия и определения

Сок – жидкий пищевой продукт, являющийся одним из главных ингредиентов любой диеты. Соки и нектары являются незаменимым средством для восстановления химической структуры поврежденных органов и тканей, содержат в себе витамины, минералы, энзимы и другим химические элементы, необходимые человеку для сбалансированного питания [2].

Натуральные соки производят из овощей и фруктов при помощи механического воздействия (отжима измельченной массы-фруктового или овощного пюре) и консервирования физическими способами.

Концентрированные соки – соки, произведенные путем физического удаления из сока прямого отжима части содержащейся в нем воды в целях увеличения содержания растворимых сухих веществ не менее чем в два раза по отношению к исходному соку прямого отжима. Такой сок является сырьем для дальнейшего производства продуктов питания и не предназначен для употребления в пищу и прямых продаж. Благодаря избавлению от большей части влаги увеличивается срок хранения и упрощается перевозка.

Восстановленный сок – сок, произведенный из концентрированного сока и питьевой воды. В нем не могут содержаться консерванты, красители, ароматизаторы и подсластители [2].

На рисунке 1 представлены этапы производства восстановленного сока [2].

  
Рисунок 1 – Этапы производства восстановленного сока

В данной работе мы рассмотрим процесс восстановления и пастеризации сока.

* 1. Описание процесса восстановления сока

В процессе восстановления нескольких бочек концентрированного сока закачивают в чан, перемешивают и отправляют в следующий танк – купажный. Там смесь концентрированных соков восстанавливают до необходимой консистенции: добавляют специально подготовленную воду, которую выпарили на предприятии по первичной переработке фруктов и овощей, тем самым получая первоначальный вариант. Воду добавляют постепенно: сначала 70%, потом 15%, затем оставшиеся 15% [3].

После тщательного перемешивания начинается процесс пастеризации. Для удаления из напитка кислорода и микроорганизмов восстановленный сок нагревается до 88-95°C, а затем резко охлаждается до 25°C. Такой способ пастеризации позволяет сохранить максимум пользы (вкусовые качества, витамины, аромат), а также уничтожить все вредные микроорганизмы.

Так получается 100% сок, который по вкусовым характеристикам и содержанию полезных веществ почти соответствует свежевыжатому [3].

* 1. Описание оборудования, задействованного в процессе восстановления сока

Промышленное оборудование – это обширный спектр различных механизмов, машин и устройств, которые используют в разнообразных производственных процессах. Перед данными устройствами стоят важные задачи, которые направлены на исполнения технологических операций [4].

Каждое предприятие, не зависимо от его специализации и уровня, остро нуждается в современном оборудовании промышленной классификации. Оборудование для производства увеличивает производительность труда, улучшает условия для сотрудников и снижает расходы ресурсов [4].

* + 1. Описание купажных ёмкостей

Купажная емкость – это одностенная емкость из нержавеющей пищевой стали цилиндрической формы с коническим или торосферическим дном и плоской крышкой. Ёмкости такого вида устанавливаются на раму или опоры, которые могут регулироваться по высоте [5]. В данной работе все ёмкости будут представлять собой купажные чаны.

Купажная емкость предназначена для быстрого перемешивания двух и более компонентных жидкостей. Купажные емкости используются при производстве алкогольных и безалкогольных напитков.

  
Рисунок 2 – Примеры купажных ёмкостей

Купажная емкость оборудована следующим [5]:

* герметичный люк со смотровым окном для технологического доступа во внутреннее пространство емкости;
* мотор-редуктор с мешалкой особой конструкции;
* патрубки наполнения и опорожнения;
* моющая головка;
* регулируемые по высоте опоры;
* датчики нижнего и верхнего уровня;
* датчик объема;
* лестница, приборы КИП и А.

В таблице 1 описаны технические характеристики купажных ёмкостей разных моделей.

Таблица 1 ‒ Технические характеристики купажных ёмкостей

|  |  |  |
| --- | --- | --- |
| Наименование показателей | КН-600 | СПК-600 |
| Нaпряжeниe питaния, (В) | 380/220 | 380/220 |
| Габариты (мм) | 2300/900/2000 | 3500/2000/1800 |
| Масса (кг) | 250 | 600 |
| Установленная мощность (кВт) | до 1,1 | до 1,5 |
| Производительность (л/час) | 600 | 600 |

На рисунке 3 представлены виды мешалок в купажных чанах.

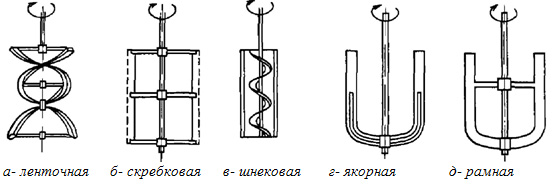
   
Рисунок 3 – Виды мешалок

Таблица 2 ‒ Технические характеристики мешалок

|  |  |
| --- | --- |
| Тип мешалки | Скорость, об/мин |
| Ленточная | от 3 до 120 |
| Скребковая | от 3 до 80 |
| Шнековая | от 3 до 200 |
| Якорная | от 3 до 120 |
| Рамная | от 3 до 120 |

В данной работе в качестве мешалки был выбран рамный тип.

* + 1. Описание нагревательного элемента

Электронагреватели – приборы, способные преобразовывать электрическую энергию в тепловую. Они могут иметь разную форму и принцип действия, но при этом всегда выполняют одну и ту же функцию [6].

Классификация нагревателей по виду [7]:

* патронные нагреватели. Патронный нагреватель содержит электрическую резистивную спираль, окруженную изолирующим порошком (обычно оксид магния) и упакованную в оболочку конической формы. Выводы прикреплены с одного конца. Этот тип нагревателя обычно вставляется в цилиндрическое отверстие. Чрезвычайно важны размер и форма отверстия, а также размер и форма нагревательного элемента;
* плоский нагреватель. В основном изготавливается плоский нагреватель в прямоугольной форме. В качестве изолятора часто может использоваться слюда. Резистивный элемент наматывается на миканит и зажимается двумя кусками слюды и затем заключен в металлическую оболочку. В некоторых случаях может использоваться без металлической оболочки;
* кольцевые нагреватели. Элементы, входящие в конструкцию кольцевого ТЭНа идентичны плоским нагревателям. Главным отличием является форма. Кольцевые нагреватели используются для нагрева жидкостей и для плавления твердых тел. Последний вид применения очень распространен в индустрии переработки пластмасс, где пластиковые гранулы необходимо нагреть до достаточной температуры. Такой нагрев сам по себе не плавит пластик, а подготавливает материал к механическому процессу обработки. Устанавливают кольцевой ТЭН на цилиндрическую поверхность оборудования, внутри которой работает шнек;
* трубчатые нагреватели. Трубчатые нагреватели имеют резистивную спираль, окруженную изолирующим порошком, заключенную в металлическую оболочку. Клеммы выходят из обоих концов нагревателя. Этот тип нагревателя обычно имеет круглое поперечное сечение, хотя может быть изготовлен и другой формы, например, квадратной или треугольной. Трубчатые нагреватели часто изготавливаются с изгибами и закруглениями для наилучшей поддержки заданной температуры [7].

В данной работе в качестве нагревательного элемента будет использоваться трубчатый нагреватель, установленный в купажный чан.

Трубчатые ТЭНы электрические применяются в промышленных и бытовых приборах для нагревания различных жидкостей или газов: вода, масло, растворы кислот (слабые), спокойный или подвижный воздух, газы [8].

Мощность ТЭНов зависит от многих параметров, самыми главными являются длина и диаметр ТЭНа. Длиной в трубчатом нагревателе считается размер греющей части трубки, без шпилек и изолятора.

Внутри металлической трубки (2) устанавливается нагревательная высокорезистивная спираль (3,4), которая сильно нагревается при прохождении через нее электрического тока. По бокам спираль намотана на контактные металлические стержни (5). Внутрь трубки со спиралью засыпается прессованный электроизолятор. На концах нагревательного элемента производится полная герметизация выводов (6) и изоляция контактных стержней диэлектриком (7), таким образом ТЭН получается полностью защищен от воздействия жидкостей или газов на нагреватель.

Трубка электротэна изготавливается из трех типов материалов: для менее мощных нагревателей чаще всего применяется углеродистая сталь, для более мощных – нержавеющая жаропрочная сталь, для создания оболочки для ТЭНов для воды может быть также использована медь [8].

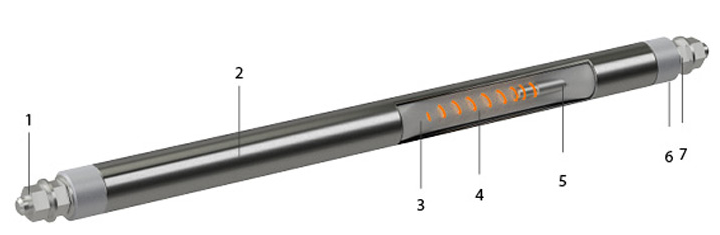
   
Рисунок 4 – Конструкция ТЭНа

Таблица 3 ‒ Технические параметры ТЭНов для воды

|  |  |
| --- | --- |
| Параметр | Показатель |
| Максимальная длина, мм | 6000 |
| Максимальный диаметр, мм | 18 |
| Минимальный диаметр, мм | 6 |
| Материал | Нержавейка, чёрная сталь, медь |
| Напряжения, В | 12, 24, 36, 42, 48, 60, 127, 220, 380 |

* + 1. Описание клапанов

В данной работе будут использоваться 2 вида клапанов: запорные и регулирующие.

Основное применение запорного клапана состоит в том, чтобы глухо перекрывать поток в трубопроводе. Как правило, для этого требуется небольшое усилие при повороте затвора. Качественный клапан будет плотно закрываться, исключая даже мельчайшие щели, для полной герметизации [9].

Регулирующие клапана являются сложным элементом, состоящим из электронной и механической частей. Электронная часть отслеживает различные параметры в трубопроводе – температуру, давление, плотность. На основании полученных данных изменяется положение задвижки. Такие клапана применяются в механизмах, где требуется создание конкретных условий для протекания технологического процесса [9].

По типу затворного механизма регулирующие клапаны разделяются на следующие устройства [10]:

* седельные. Рабочим элементом седельного клапана является плунжер. Плунжер проходит через одно или два седла клапана, изменяя проходное сечение регулирующего клапана. Односедельные клапаны используются в системах трубопроводов малого диаметра, а устройства с двумя седлами отличаются более точной регулировкой давления и устанавливаются на трубопроводах большего размера и с большем давлением рабочей среды;
* клеточные. В таких регулирующих клапанах запорный элемент имеет форму полого цилиндра с радиальной перфорацией. При помощи такого затвора осуществляется контроль и регулировка давления рабочей среды в системе трубопроводов. Принцип действия таков: затвор перемещается внутри «клетки», которая служит направляющим элементом и изменяет ее пропускную способность;
* мембранные. В мембранных клапанах регулирование, перекрытие потока рабочей среды в трубопроводе происходит с помощью мембраны, которая служит запорным элементом. Изготавливается мембрана из эластичной резины или фторопласта. В процессе регулирования потока рабочей среды в мембранных клапанах возникают погрешности. Для устранения этих погрешностей клапаны комплектуются специальными устройствами (элементами), которые контролируют положения штока, что позволяет более точно регулировать процесс транспортировки рабочей среды по трубопроводу.
* золотниковые. Регулирующие клапана золотникового типа имеют запорно-регулирующий элемент в виде золотника, который в зависимости от ситуации поворачивается на определенный угол. При управлении клапаном с золотниковым типом затвора не требуется больших усилий. Рабочая среда в виде жидких и газообразных веществ почти не оказывает сопротивления при перемещении затвора клапана [10].

В данной работе будут использоваться клапаны золотникового типа. Внешний вид золотникового клапана представлен на рисунке 5.

   
Рисунок 5 – Золотниковый клапан

* 1. Постановка задачи

Во время лабораторного практикума необходимо разработать систему моделирования процесса восстановления и пастеризации сока с помощью SCADA-системы WinCC OA 3.17, которая будет управлять процессом подачи жидкостей в купажные чаны, перемешиванием и нагревом раствора в заданных режимах и с помощью которой оператор сможет наблюдать за данным процессом посредством мнемосхемы, отображающей текущее состояние оборудования на разных этапах.

В системе должно быть реализовано три роли пользователей: администратор, оператор и гость.

Каждый пользователь должен авторизоваться в системе (ввести логин (от 4 до 20 символов) и пароль (от 4 до 18 символов)), система должна проверить ученые данные пользователей и настроить интерфейс пользователя на заданную роль.

В режиме оператора пользователь сможет наблюдать за работой оборудования и управлять им при возникновении нештатных ситуаций (открывать и закрывать клапаны, запускать и останавливать мешалку, включать и выключать нагревательный элемент). Так же оператор должен будет иметь возможность квитировать аварийные сообщения, просматривать архивные параметры за требуемый интервал времени и справочную информацию.

В режиме администратора пользователь должен иметь возможность выполнять все функции оператора, а также осуществлять настройку параметров оборудования, а именно: угол открытия регулируемого клапана, количество оборотов мешалки температуру нагревательного элемента.

Необходимо обеспечить проверку корректности введенных администратором данных: угол открытия регулируемого клапана должен быть в пределах 0-180°, температура жидкости должна быть в пределах 15-20°C, мощность двигателя купажного чана должна быть в пределах 1,1-1,5 кВТ, максимальная температура нагрева смеси должна быть в пределах 90-95°C.

В режиме гостя пользователь сможет только наблюдать за ходом технологического процесса в автоматическом режиме.

В системе должно быть предусмотрено два режима работы: ручной и автоматический. В автоматическом режиме параметры оборудования (соотношение и скорость подачи жидкостей в купажный чан, температуры жидкостей, скорость мешалки, максимальная температура нагрева) являются значениями по умолчанию. В ручном режиме администратор должен вводить данные параметры самостоятельно. Также должна иметься возможность выбора времени перемешивания.

Необходимо предусмотреть возможность экстренного отключения двигателя купажного чана, нагревательного элемента и закрытия всех клапанов при возникновении аварийных ситуаций.

Все процессы, реализуемые в системе должны быть визуализированы:

* работа компонентов системы: подача жидкостей в купажный чан, регулирование клапанов, перемешивание смеси, нагрев смеси;
* параметры, состояния оборудования и объёмы жидкостей в купажных чанах;
* отказы оборудования и выдача аварийных сообщений.

Так же в системе должна быть справочная информация, содержащая информацию о системе и работе в ней, а также информацию о разработчиках.

Таким образом, в системе должны быть реализованы следующие функции:

1. функции системы:
   * аутентификация пользователя в системе, настройка интерфейса пользователя на заданную роль;
   * моделирование работы оборудования в автоматическом режиме:
   * запуск оборудования;
   * обеспечение подачи продукта в ёмкости;
   * регулирование температуры нагревательного элемента;
   * регулирование скорости мешалки;
   * моделирование возникновения нештатных ситуаций (отказов) в режиме реального времени и выдача аварийных сообщений;
   * визуализация процесса работы оборудования:
   * отображение штатной работы оборудования;
   * отображение аварийных сообщений;
   * выдача справочной информации;
2. функции администратора:
   * авторизация пользователя в системе (ввод логина/пароля);
   * настройка параметров оборудования (точек данных):
   * задать угол открытия регулируемого клапана;
   * задать количество оборотов для мешалки;
   * задать температуру нагревательного элемента;
   * работа с оборудованием (ручной режим):
   * открыть клапан;
   * закрыть клапан;
   * запустить мешалку;
   * остановить мешалку;
   * запустить нагревательный элемент;
   * остановить нагревательный элемент;
   * квитировать аварийные сообщения;
   * просмотр архивных параметров за требуемый интервал времени;
   * просмотр справочной информации;
3. функции оператора:
   * авторизация пользователя в системе (ввод логина/пароля);
   * работа с оборудованием (ручной режим):
   * открыть клапан;
   * закрыть клапан;
   * запустить мешалку;
   * остановить мешалку;
   * запустить нагревательный элемент;
   * остановить нагревательный элемент;
   * квитирование аварийных сообщений;
   * просмотр архивных параметров за требуемый интервал времени;
   * просмотр справочной информации;
4. функции гостя:
   * наблюдение за ходом технологического процесса в автоматическом режиме.
5. Проектирование системы
   1. Разработка информационной модели объекта

Отправной точкой при создании проекта в WinCC OA является разработка информационной модели будущей системы, включающей иерархию и структуры данных, алармов и их свойств. Основой для формирования информационной модели может служить (в зависимости от назначения и масштаба системы) техническая структура объектов, включенных в SCADA-систему (аналогия «элемент – устройство – агрегат – …»), организационная модель объекта, например иерархия диспетчерских пунктов (местный уровень – региональный уровень – центральный уровень), или их сочетание. Информационная модель выражается в виде совокупности типов точек данных и созданных на их основе структур для хранения конкретных переменных процесса [11].

* + 1. Описание точек данных (Data Points)

Ключевым элементом при построении информационной модели в WinCC OA является понятие точки данных, связывающей переменные (теги), относящиеся к некоторому устройству или объекту, в единую древовидную структуру с возможностью создания произвольных подуровней. На практике в точку данных, относящуюся к одному устройству, объединяются примерно от 4 до 30 переменных [12].

В соответствии с идеологией WinCC OA для создания точек данных должен быть определен соответствующий тип точки данных, который будет выполнять роль шаблона при формировании множества точек данных с такой структурой.

Тип точки данных (DPT – data point type) – это описание структуры разнородных типизированных элементов, объединенных по признаку принадлежности к одному физическому объекту. Тип точки данных определяет формат представления сущности и показывает, в каком виде информация о процессе будет храниться в точке данных, какими элементами и какого типа будет описан реальный объект. Тип не хранит значение сущностей, конкретные значения будут храниться в соответствующей точке данных – экземпляре типа ТД. Точка данных (ТД, DP – data point) – это переменная, характеризующая состояние объекта, состоящая из элементов ТД, строго определенных структурой ТТД.

Элемент точка данных (ЭТД, DPE – data point element) – это динамическая переменная, обозначающая состояние объекта или внутренние состояния системы строго определенного типа данных [11].

В таблицах 4 и 5 представлено описание параметров оборудования, участвующих в построении информационной модели системы.

В системе должно быть предусмотрено:

* выключение двигателя купажного чана при неисправностях мешалки;
* выключение нагревательного элемента при неисправностях купажного чана;
* выдача ошибок при неисправностях оборудования;
* аварийное ручное отключение.
  1. Проектирование интерфейса пользователя

Проектирование пользовательского интерфейса (ПИ) сегодня является очень важным аспектом разработки и требует от разработчиков специальных знаний о том, как разработать хороший интерфейс. Качество пользовательского интерфейса является самостоятельной характеристикой программного продукта, сопоставимо по значимости с такими его показателями, как надежность и эффективность использования вычислительных ресурсов.

Таблица 4 – Описание параметров оборудования

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Параметр | Диапазон значений | Тип параметра |
| Купажный чан | Рабочий объем | 600 л | фиксированный |
| Объём | [100, 600] л | рассчитываемый |
| Скорость мешалки | [3, 120] об/мин | настраиваемый |
| Статус | [включен; выключен] | логический |
| Датчик уровня | Текущий уровень заполнения | [0; 100] % | измеряемый |
| Состояние | [работает; не работает] | измеряемый |
| Датчик температуры | Текущий уровень температуры | [0; 100] °C | измеряемый |
| Состояние | [работает; не работает] | измеряемый |
| Регулируемый клапан | Текущее состояние | [закрыт, открыт] | логический |
| Угол открытия клапана | [0; 180] ° | настраиваемый |
| Нагревательный элемент | Уровень температуры | [15; 95] °C | настраиваемый |
| Статус | [включен; выключен] | логический |

Таблица 5 – Описание точек данных

| Точка данных | Команды | Состояния | Неисправности/Аварийные сообщения |
| --- | --- | --- | --- |
| 1 | 2 | 3 | 4 |
| Купажный чан | запустить, остановить, установить скорость мешалки, отправить в ремонт. | в работе,  выключен,  отказал,  в ремонте. | * не включился; * не выключился; * заклинило мешалку; * износ мешалки. |
| Датчик уровня | включить, выключить, отправить в ремонт. | включен, выключен,  отказал,  в ремонте. | * не включился; * не выключился. |
| Датчик температуры | включить, выключить, отправить в ремонт. | включен, выключен,  отказал,  в ремонте. | * не включился; * не выключился. |
| Запорный клапан | закрыть, открыть, отправить в ремонт. | закрыт, открыт,  отказал,  в ремонте. | * не закрылся; * не открылся. |
| Регулируемый клапан | закрыть, открыть, установить угол открытия, отправить в ремонт. | закрыт, открыт,  отказал,  в ремонте. | * не закрылся; * не открылся; * не установился нужный угол открытия. |



Продолжение таблицы 5

| 1 | 2 | 3 | 4 |
| --- | --- | --- | --- |
| Нагревательный элемент | запустить, остановить, установить температуру, отправить в ремонт. | в работе,  выключен,  отказал,  в ремонте. | * не включился; * не выключился; * не установилась нужная температура. |

Эффективный ПИ должен обеспечивать бесперебойное предоставление информации о системе пользователю, явное и своевременное информирование об ошибках.

Отличительной особенностью интерфейса систем реального времени является наличие ограничения времени его реакции на внешние воздействия. В человеко-машинных системах реального времени в процессе управления участвует человек. Он может просто следить за тем, чтобы не случилось каких-нибудь аварийных ситуаций. Если они происходят, он должен предпринимать меры, помогая системе.

На рисунке 6 изображен прототип мнемосхемы разрабатываемой системы.

В системе должен быть предусмотрен контроль управления, который позволит определять, в каком состоянии находится каждая точка данных. Так же в системе можно будет включить ручной режим управления, который позволит оператору вносить изменения в процесс пастеризации.

На рисунке 7 представлен прототип панели настройки оборудования. Здесь можно будет задать параметры температуры нагревательного элемента, скорости вращения мешалки, состояния клапанов.

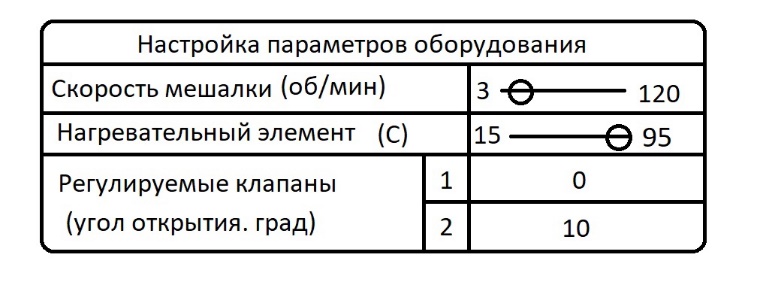
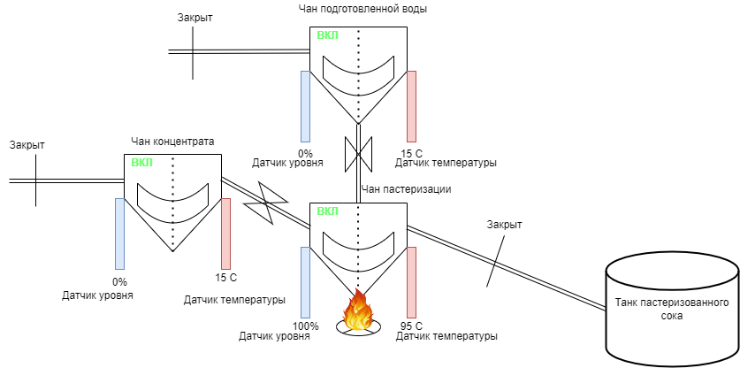
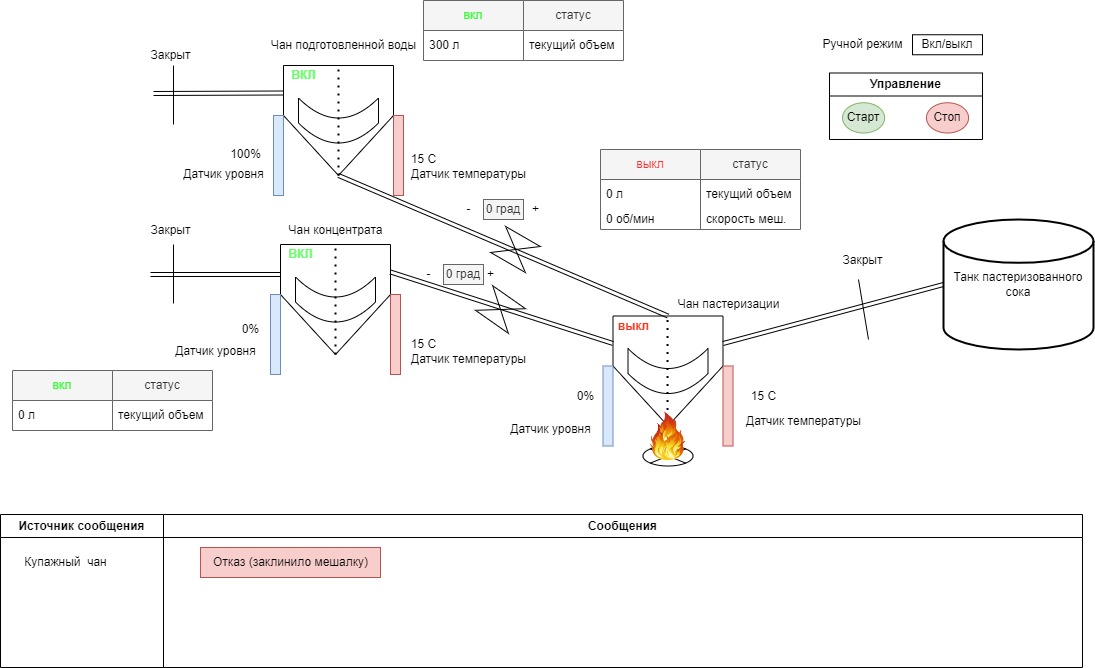


Рисунок 7 – Прототип панели настройки оборудования

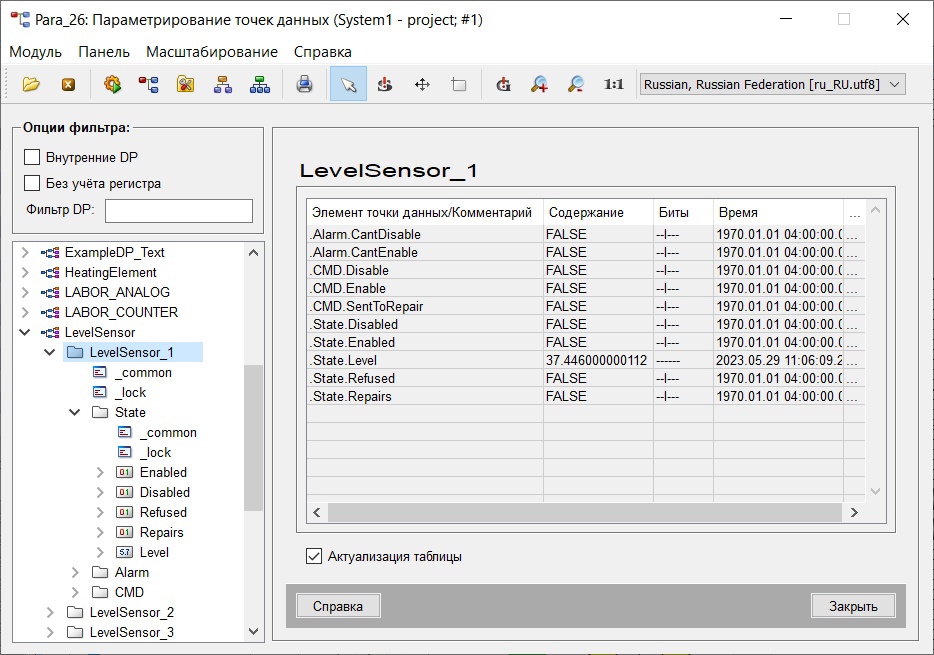
На рисунке 8 представлен прототип панели управления, с помощью которой можно будет осуществлять контроль процесса восстановления и пастеризации сока.

  
Рисунок 6 – Прототип мнемосхемы системы

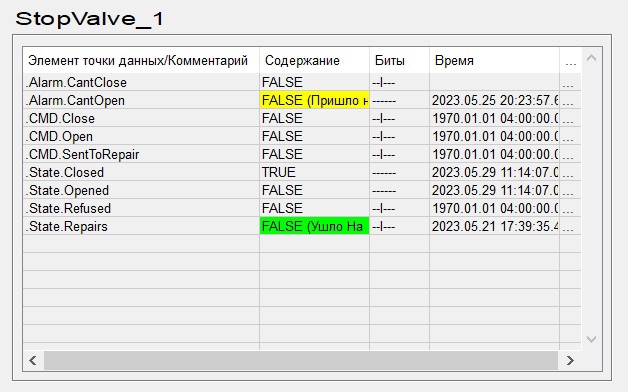
  
Рисунок 8 – Прототип главной панели приложения

1. Реализация системы
   1. **Настройка элементов точек данных**

Для точной работы системы требуется организовать точки данных для элементов интерфейса. В WinCC OA точки данных представлены в виде дерева (рисунок 9).

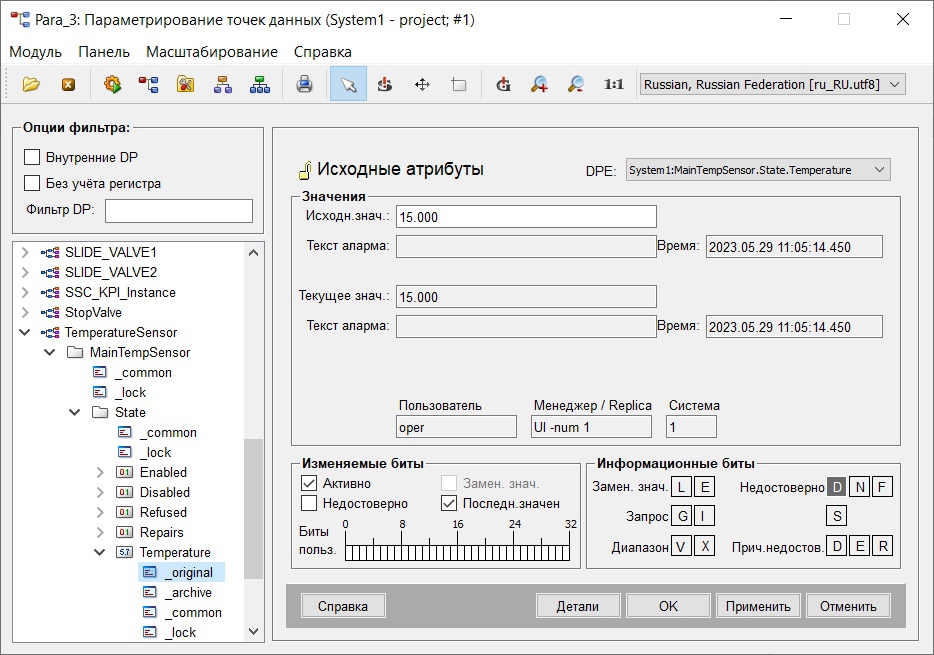
Рисунок 9 – Общая структура точек данных (дерево ТД)

В соответствии с информационной моделью системы, разработанной в п. 2.1.1, были внесены с систему необходимые точки данных (описание приведено в приложении А), на рисунке 10 приведен пример ТД «Запорный клапан».

 **Рисунок 10 – Описание ТД «Запорный клапан»**

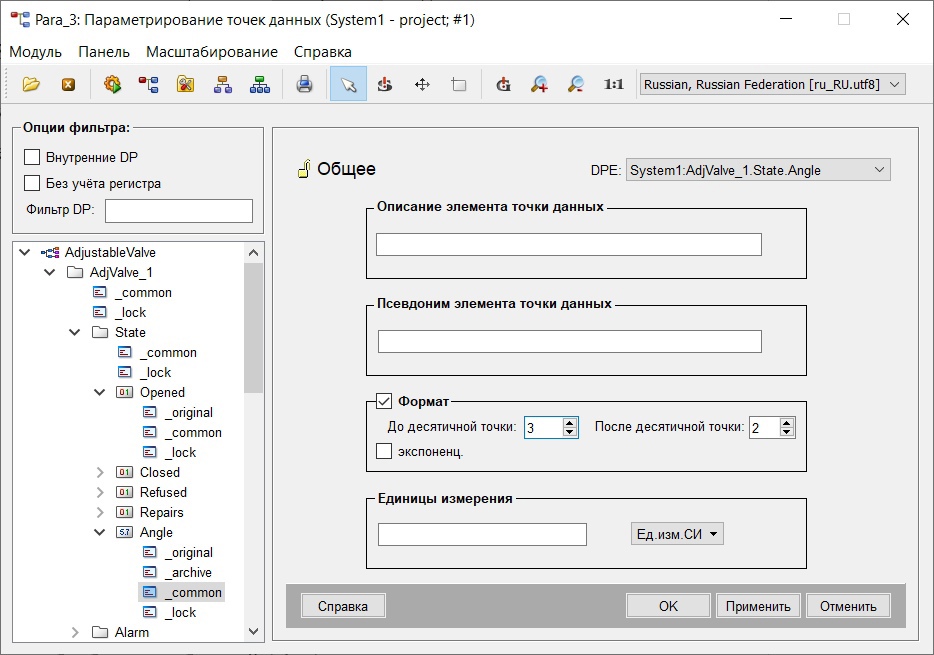
* 1. **Настройка исходных значений**

В зависимости от типа, настраивается начальное значение точки (рисунок 11).

Рисунок 11 – Настройка \_origin датчик температуры

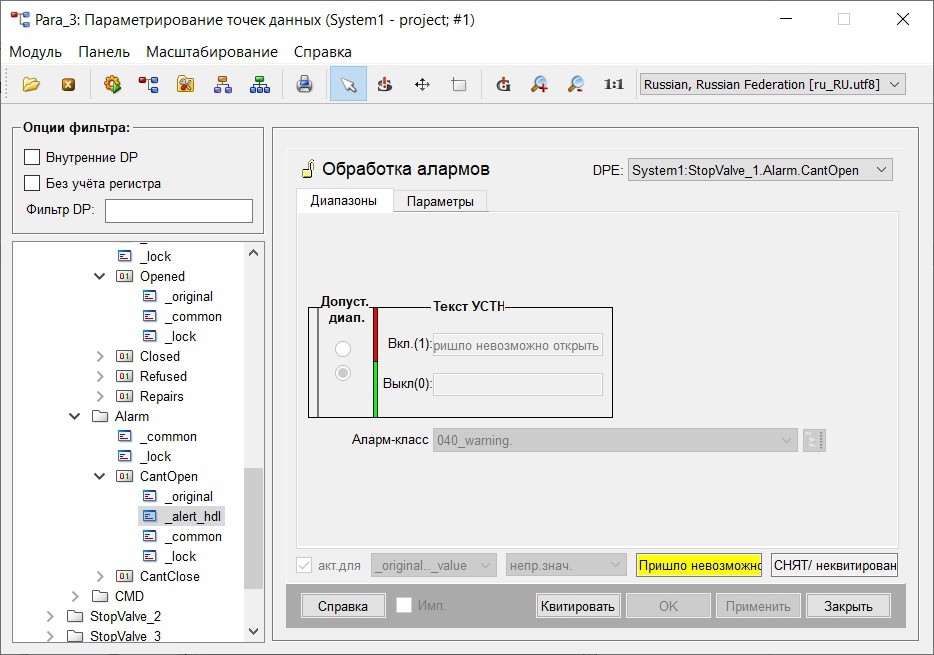
* 1. **Общие настройки объекта**

**Существует возможность также совершить общие настройки. Например, добавить описание или ограничить количество знаков после запятой для соответствующих типов точек данных (рисунок 12).**

  
Рисунок 12 – Настройка \_common для регулируемого клапана

* 1. **Обработка сообщений (alert)**

Если значение точки данных превышает некое заданное значение, то можно настроить вывод соответствующего сообщения (рисунок 13).

  
Рисунок 13 – Настройка \_alert\_hdl для запорного клапана

* 1. **Разработка и описание графического интерфейса**

Графический пользовательский интерфейс представляет собой форму пользовательского интерфейса, которая позволяет пользователям взаимодействуйте с электронными устройствами с помощью графических значков и звуковых индикаторов, таких как первичная нотация, вместо текстовых пользовательских интерфейсов, печатных меток команд или текстовой навигации [13].

* + 1. Описание графических элементов мнемосхемы

В разрабатываемую систему входят элементы, помогающих оператору эффективно использовать возможности пресса, их описание приведено в таблицах 6 и 7.

Таблица 6 – Описание пиктограмм

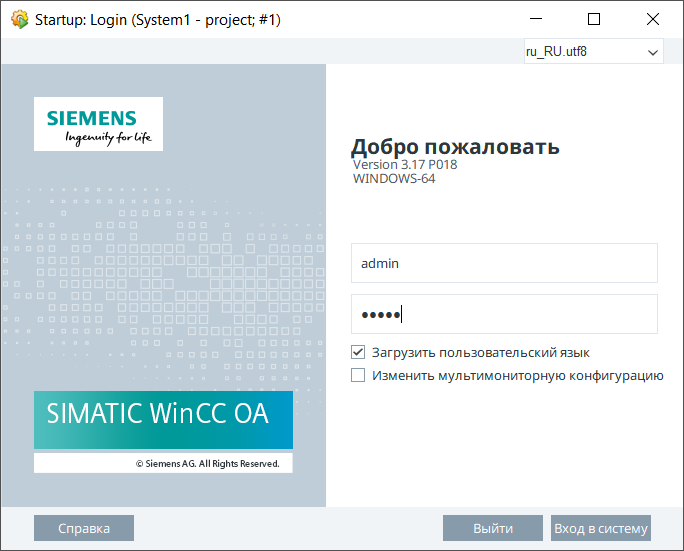
|  |  |
| --- | --- |
| Пиктограмма | Назначение |
|  | Переключатель, изменяющий состояние оборудования (автоматический режим) |
|  | Переключатель, изменяющий состояние оборудования (задвижки) |
|  | Отображает значение характеристики оборудования |
|  | Показывает уровень жидкости/температуры в резервуаре |
|  | Индикатор задвижки (красный цвет – задвижка закрыта; зеленый цвет – задвижка открыта) |
|  | Индикатор скорости мешалки (чем быстрее скорость мешалки, тем быстрее моргает) |

Таблица 7 – Описание кнопок для управления

|  |  |
| --- | --- |
| Кнопка | Назначение |
|  | Кнопка наладки (ремонта) мешалки при возникновении нештатной ситуации. |
|  | Открывает панель настроек для администратора |

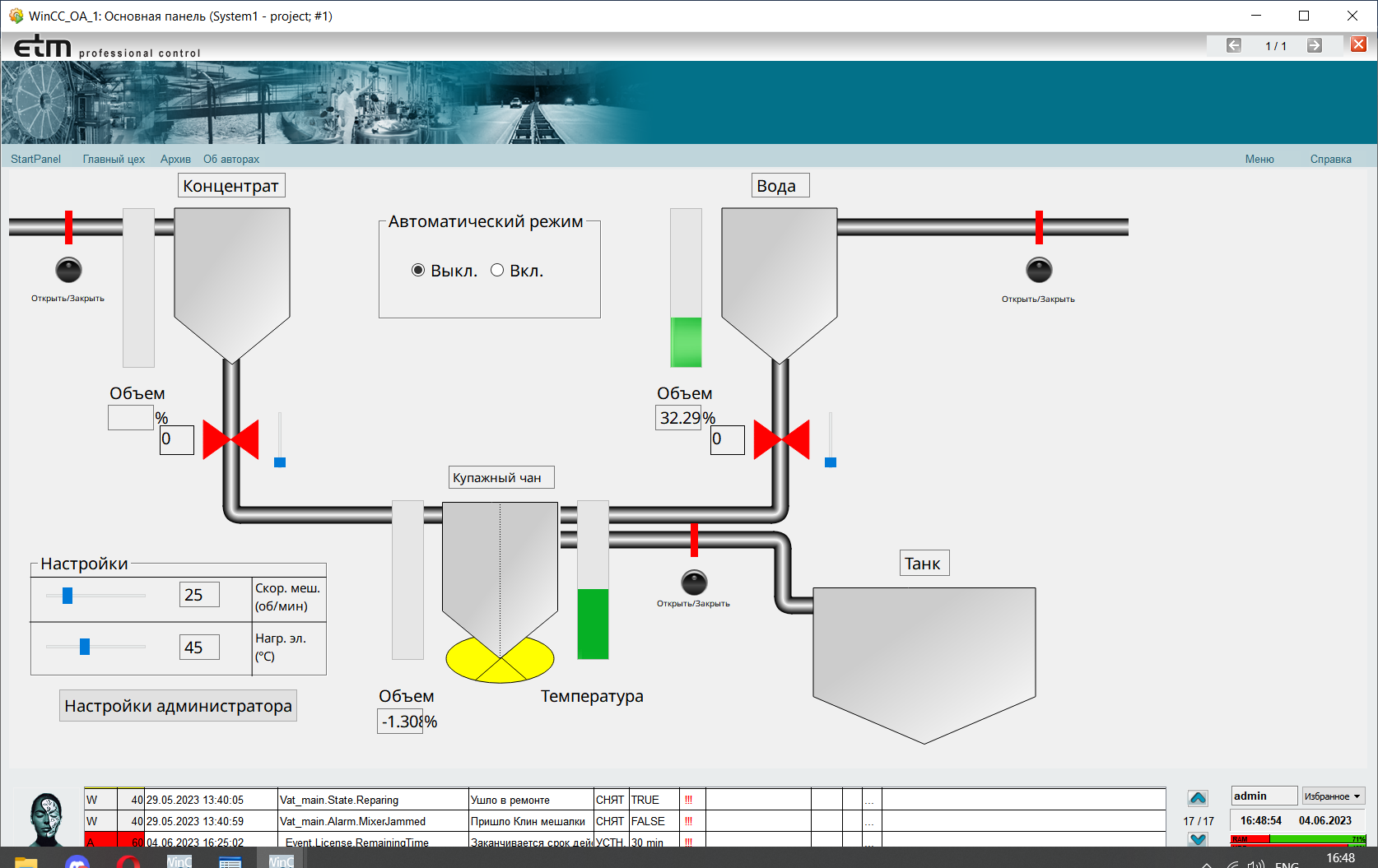
* + 1. Начало работы

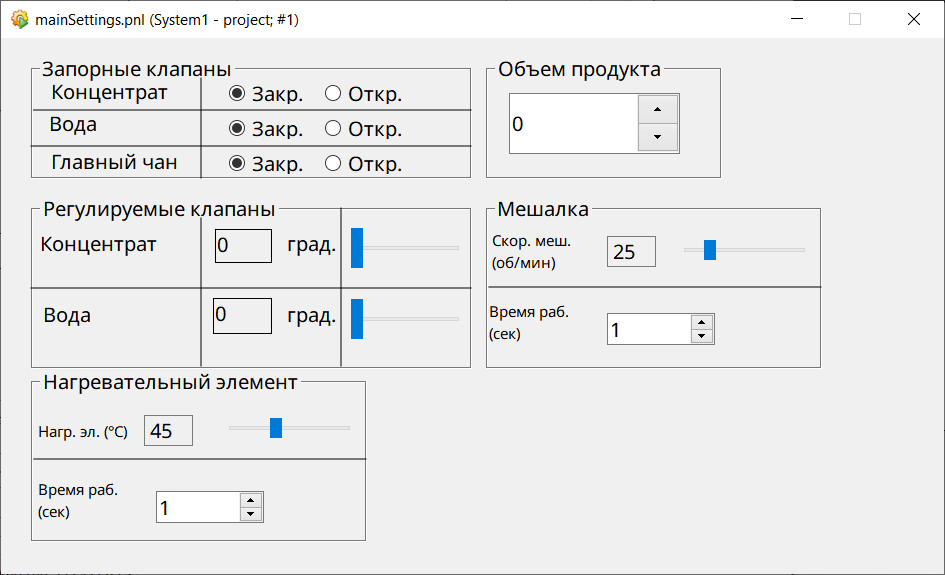
В системе предусмотрено три роли пользователей: администратор, оператор и гость. Перед началом работы с системой пользователи должны авторизоваться. На рисунке 14 приведено окно авторизации в системе, которое отображается при запуске системы. В центральной части окна расположены поля для ввода логина и пароля. В нижней части окна расположены кнопки «ОК» и «Отменить». При нажатии кнопки «Отменить» окно авторизации закрывается. При нажатии кнопки «ОК», если логин и пароль введены верно, окно для входа в систему закроется и откроется новое окно для работы с системой.

  
Рисунок 14 – Окно «Вход в систему»

* + 1. Режим администратора

После входа в систему администратору открывается рабочее окно. Администратор начинает с главной мнемосхемы (рисунок 15). Для начальной настройки оборудования администратор должен нажать на кнопку «Настройки администратора». Система открывает вкладку «Настройки» для настройки параметров системы (рисунок 16).

  
Рисунок 15 – Основная панель приложения в начальном состоянии

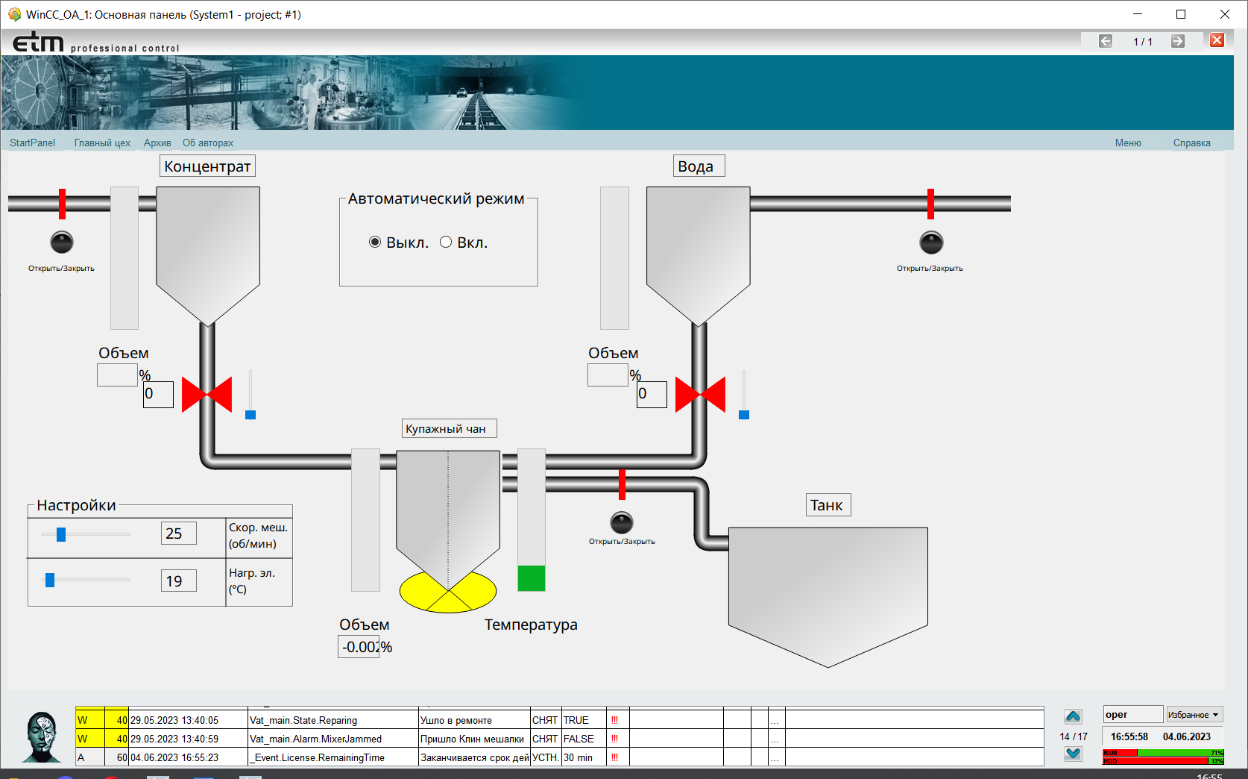
  
Рисунок 16 – Панель настроек для администратора

В окне настроек администратор может изменить состояние запорных клапанов, угол открытия регулируемых клапанов, возможный объём жидкости, скорость и время работы мешалки, температуру и время работы нагревательного элемента.

Введённые параметры сохраняются автоматически для соответствующих точек данных.

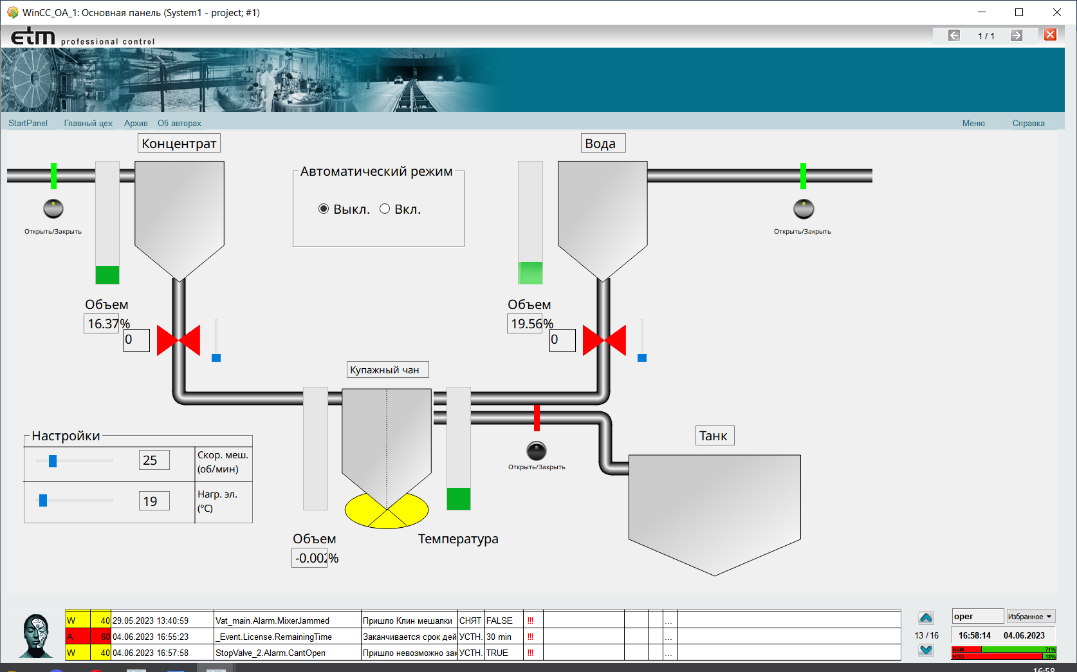
* + 1. Режим оператора

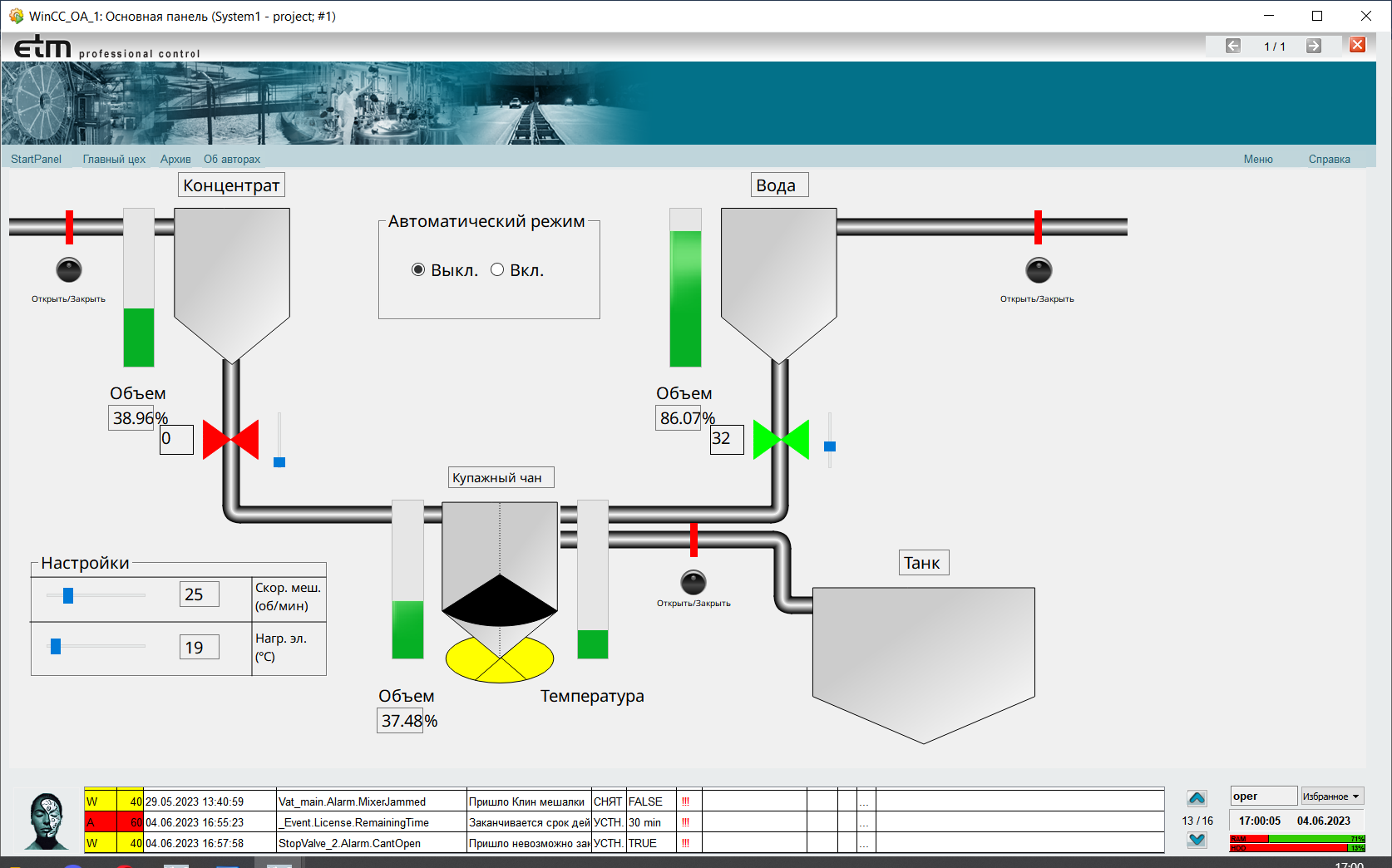
Оператор начинает работу с вкладки «Главный цех». На рисунке 17 система приведена в начальном состоянии. Здесь оператор в ручном режиме может задать все необходимые параметры для настройки оборудования.

  
Рисунок 17 – Основная панель приложения в начальном состоянии

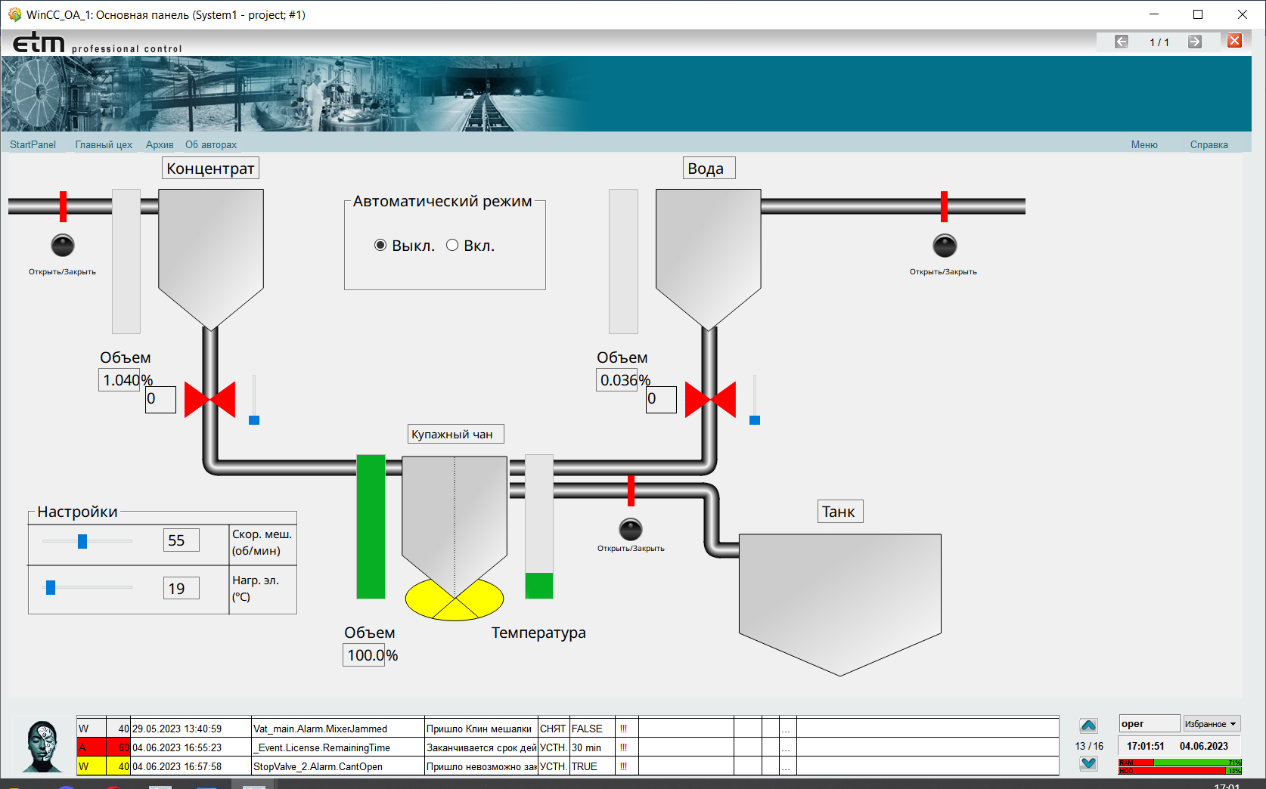
На рисунке 18 приведен процесс заполнения чанов. Для заполнения необходимо открыть запорные клапаны, после чего продукты начнут поступать в чаны. Справа от каждого чана отображается текущий уровень заполненности.

На рисунке 19 приведен процесс заполнения купажного чана. Для начала этого процесса необходимо установить угол открытия регулируемых клапанов. После чего продукты двух типов (концентрат и вода) попадают в основной (купажный) чан. Справа от купажного чана отображается текущий уровень заполненности. Слева от купажного чана отображается текущая температура жидкости.

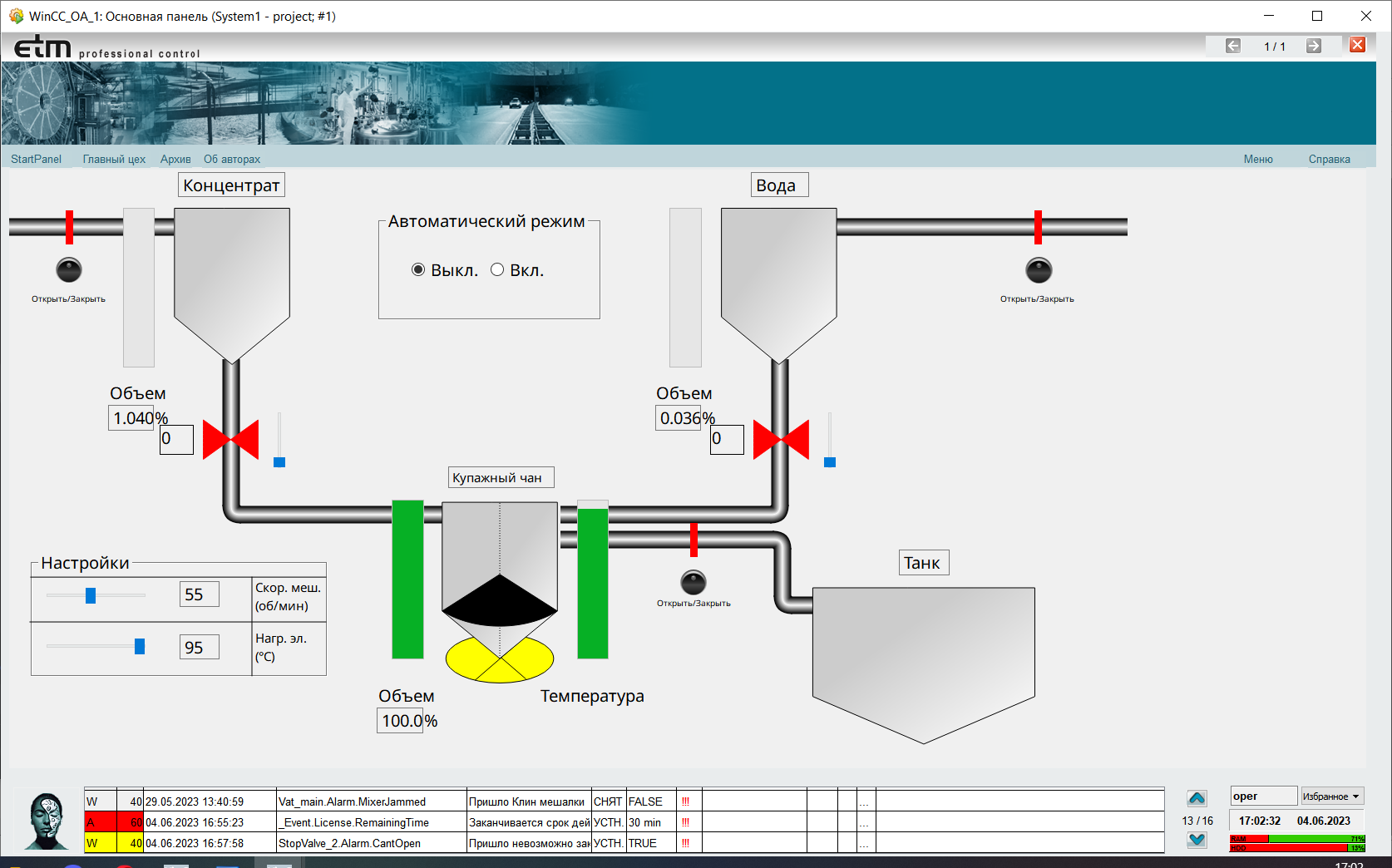
  
Рисунок 18 – Процесс заполнения чанов

  
Рисунок 19 – Процесс заполнения купажного чана

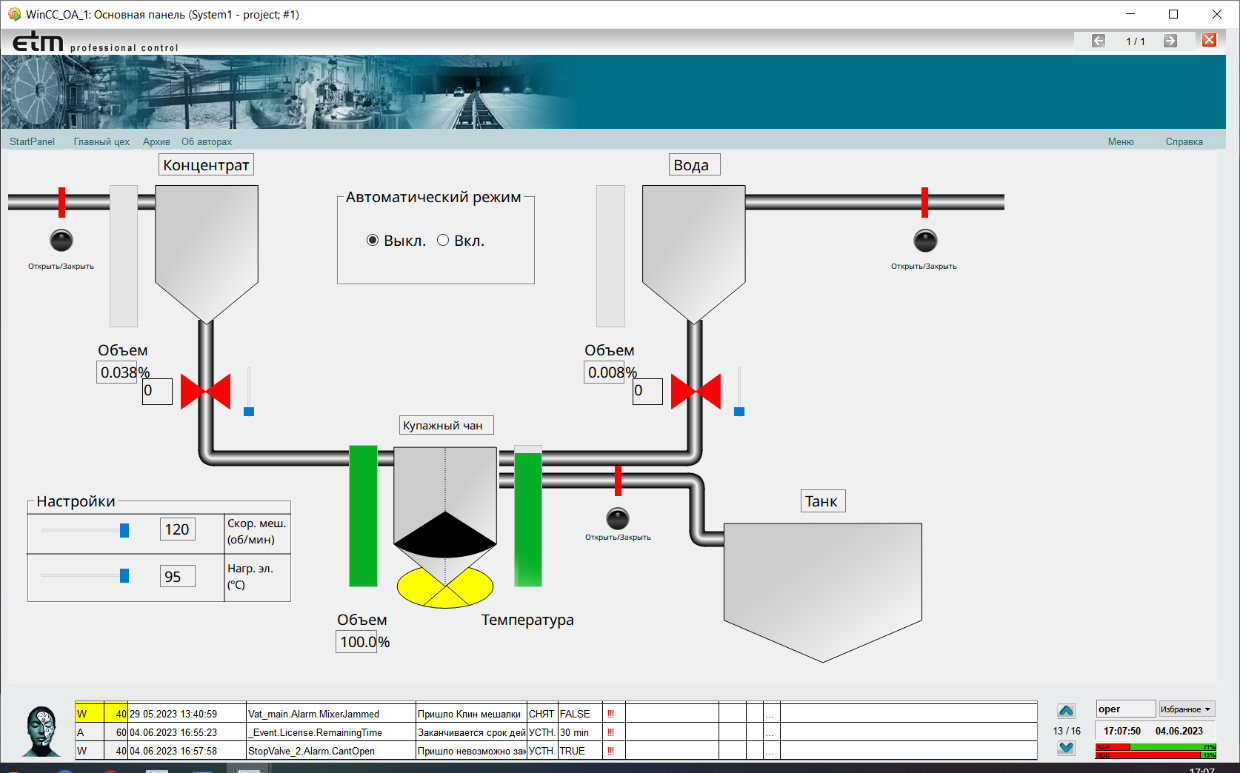
На рисунке 20 приведен процесс предварительного перемешивания сока. Для запуска этого процесса оператору необходимо установить скорость мешалки.

  
Рисунок 20 – Процесс предварительного перемешивания сока

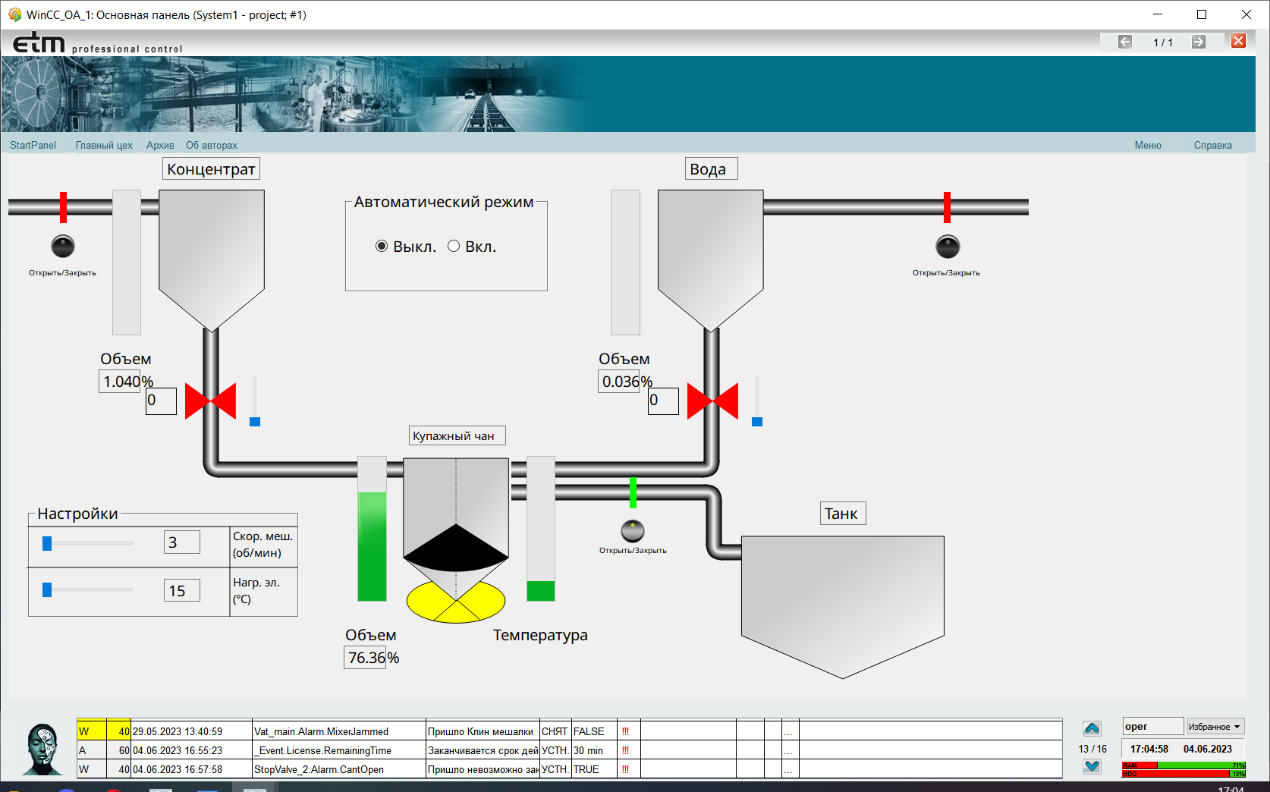
На рисунке 21 приведен процесс нагрева сока. Для запуска этого процесса оператору необходимо установить температуру нагрева.

  
Рисунок 21 – Процесс нагрева сока

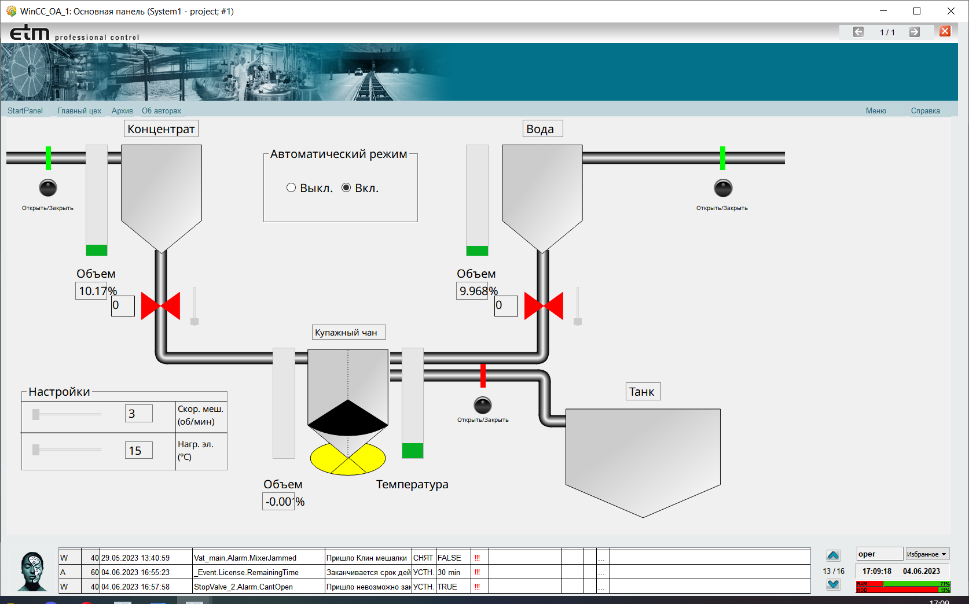
На рисунке 22 приведен процесс интенсивного перемешивания сока. Для запуска этого процесса оператору необходимо установить скорость мешалки на максимум.

  
Рисунок 22 – Процесс интенсивного перемешивания сока

На рисунке 23 приведен процесс слива готового сока. Для начала этого процесса оператору необходимо открыть запорный клапан, после чего продукт из купажного чана начнёт переливаться в танк.

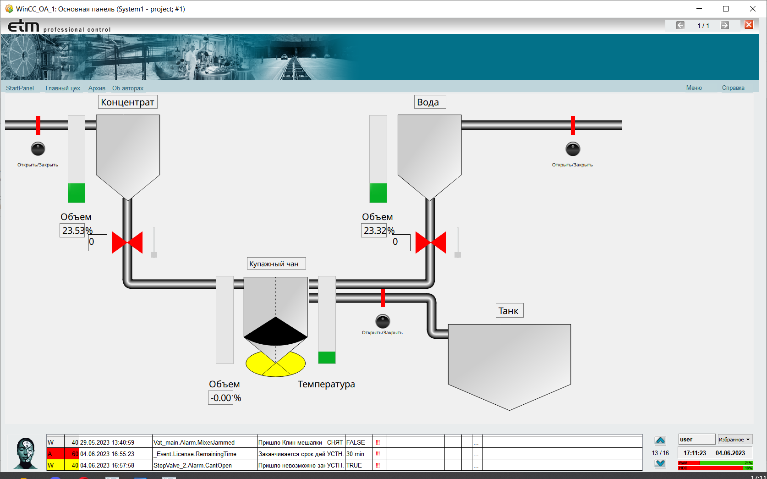
  
Рисунок 23 – Процесс слива готового сока

На рисунке 24 приведен процесс приготовления сока в автоматическом режиме. Для запуска автоматического режима оператору необходимо переключить автоматический режим на «Вкл.».

  
Рисунок 24 – Процесс приготовления сока (автоматический режим)

* + 1. Режим гостя

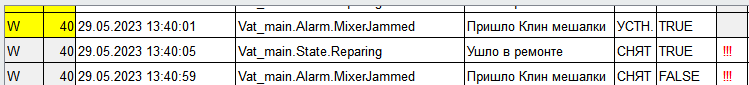
На рисунке 25 система приведена в начальном состоянии. Пользователь не имеет доступа к работе с оборудованием. Однако он может видеть весь происходящий процесс.

  
Рисунок 25 – Процесс приготовления сока

* + 1. **Обработка внештатных ситуаций**

**В системе предусмотрено возникновение внештатных ситуаций (alarm). Внештатная ситуация может быть связана с отказом оборудования (мешалки, клапанов).**

**Аварийные сообщения о возникновении внештатной ситуации отображаются в нижней части рабочего окна. В списке выводятся время возникновения критической ситуации и описание проблемы (рисунок 26).**

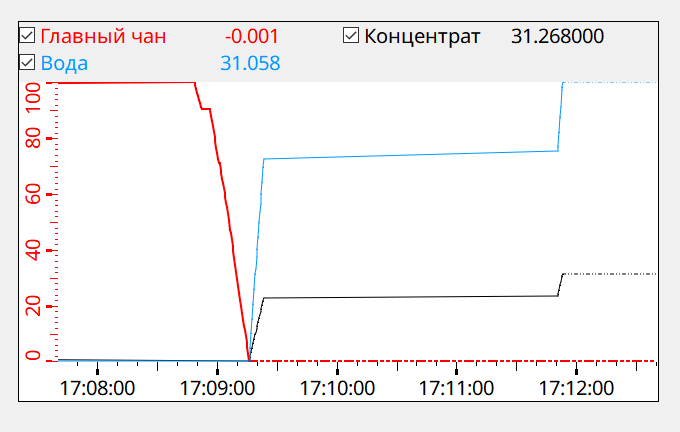
 **Рисунок 26 –** Сообщение о внештатной ситуации   
«Оборудование неисправно»

Для устранения отказа мешалки необходимо нажать на кнопку «Ремонт», которая появляется после возникновения ошибки. Для устранения отказа клапана требуется включить его заново.

* + 1. **Просмотр параметров оборудования**

**Для отображения архива значений параметров перейти на мнемосхему «Архив». В этой вкладке отобразится тренд – визуализация** зависимости выбранной величины от времени. Таким образом можно отследить изменение значений параметра в определенный период времени.

**Пример тренда объёма чанов приведен на рисунке 27.**

 **Рисунок 27 – Тренд объёма чанов**

**Заключение**

**В ходе выполнения лабораторного практикума были изучены возможности SCADA-системы WinCC OA 3.17, с ее помощью была разработана система восстановления и пастеризации сока.**

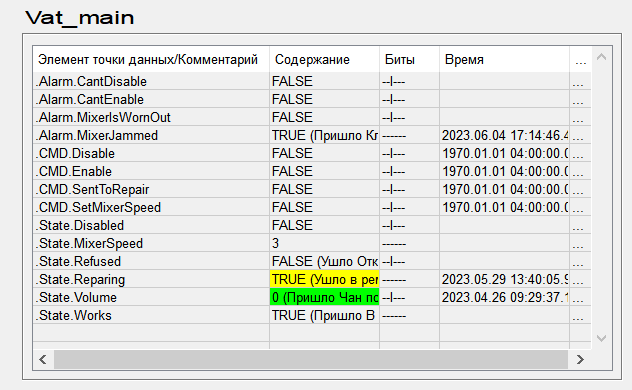
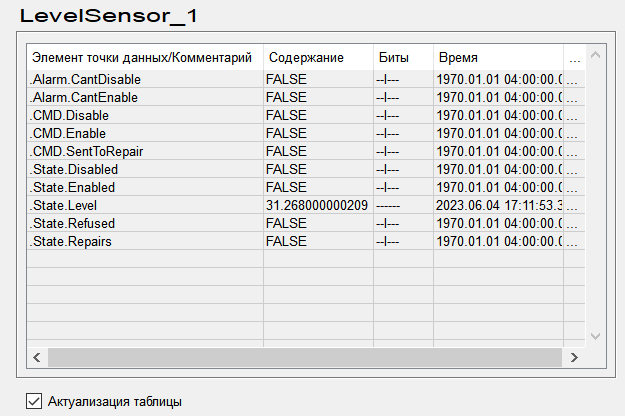
**В первой главе приведены основные понятия предметной области, рассмотрены принципы работы процесса восстановления и пастеризации сока. На основании данной информации были определены основные функции разрабатываемой системы и требования к ней.**

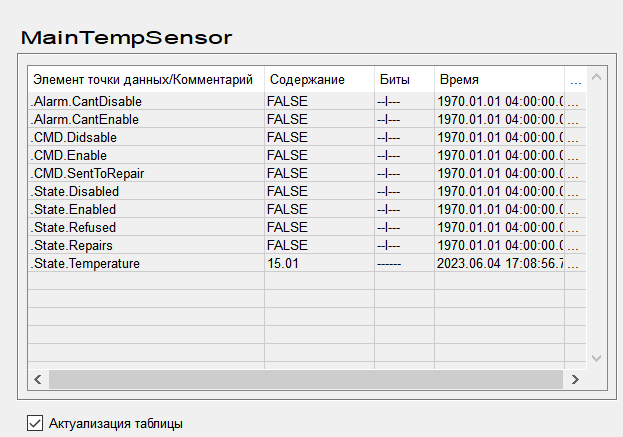
**Во второй главе приведены результаты проектирования системы: разработана информационная модель системы, определены основные точки данных для оборудования,** описан прототип интерфейса системы.

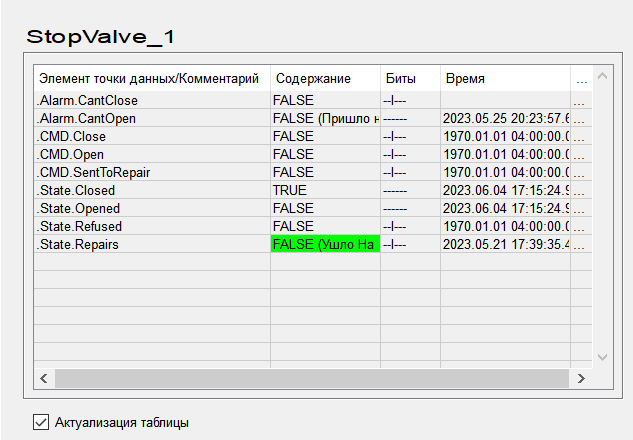
**В третьей главе описаны основные возможности разработанной системы: интерфейс пользователя с соответствием с мнемосхемой, созданы точки данных, написаны скрипты для реализации работы системы (см. приложение Б) и ее взаимодействия с пользователем, при нарушении допустимых норм в журнал сообщений приходит предупреждение.**

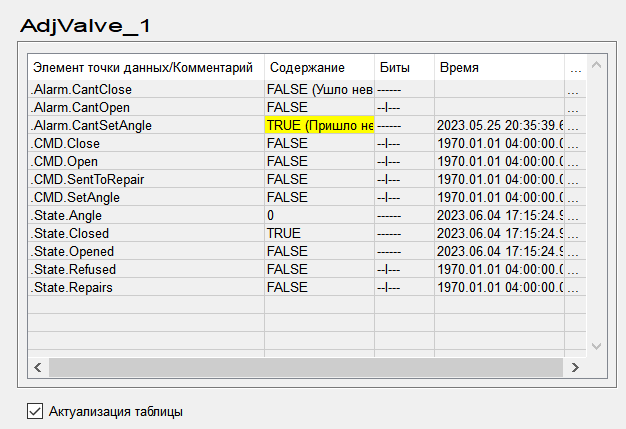
**Список использованных источников**

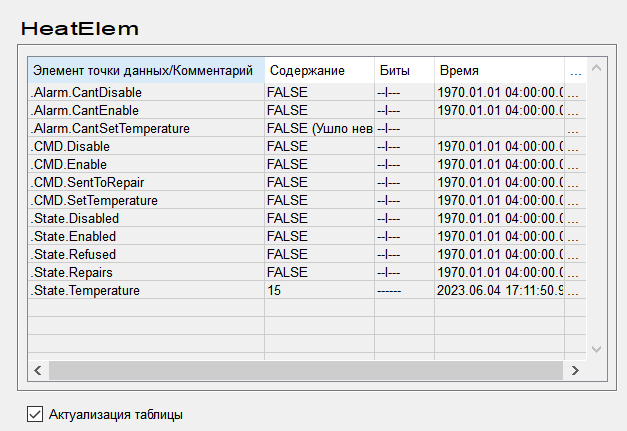
1. Зеленко Л.С. Презентация «SCADA-системы» [Электронный ресурс] / Л.С. Зеленко// Систем. требования: Power Point 2010/ Самарский университет, 2023 (дата обращения: 01.03.2023).
2. Производство соков и сокового концентрата: оборудование и технология [Электронный ресурс]. URL: https://prcs.ru/oborudovanie-i-tehnologii/proizvodstvo-sokov-i-sokovogo-koncentrata-oborudovanie-i-tehnologiy (дата обращения: 01.03.2023).
3. Как делают сок [Электронный ресурс]. URL: https://ochakovo.ru/media/kak-delayut-sok/ (дата обращения: 01.03.2023).
4. Промышленное оборудование [Электронный ресурс]. URL: https://ochakovo.ru/media/kak-delayut-sok/ (дата обращения: 03.03.2023).
5. Купажные ёмкости [Электронный ресурс]. URL: https://supersteel.ru/catalog/emkostnoe\_oborudovanie/nestandartnyie\_emkosti/kupajnyie\_emkosti (дата обращения: 03.03.2023).
6. Классификации ТЭН и нагревательных элементов [Электронный ресурс]. URL: https://elektroteni.ru/blog/nagrevateli/klassifikatsii-elektronagrevateley/ (дата обращения: 10.03.2023).
7. Нагревательные элементы – определение и типы [Электронный ресурс]. URL: https://ten24.com.ua/blog/nagrevatelnye-elementy-opredelenie-i-tipy/ (дата обращения: 10.03.2023).
8. ТЭНы для нагрева воды и воздуха [Электронный ресурс]. URL: https://ten24.com.ua/catalog/eletronagrevateli-teny/teny-dlya-vody-i-vozdukha/ (дата обращения: 01.06.2023).
9. Типы клапанов [Электронный ресурс]. URL: https://proarma.ru/tipy-klapanov-kakiye-klapany-byvayut (дата обращения: 19.03.2023).
10. Запорно-регулирующие клапаны: виды, применение [Электронный ресурс]. URL: https://armstroy-nn.ru/useful\_info/article/reguliruyushchie-klapany-vidy-primenenie/ (дата обращения: 19.03.2023).
11. Преимущества и недостатки UML нотации в бизнес моделировании [Электронный ресурс] // Блог о бизнес-процессах и BPMN. URL: https://bpmn.pro/process/uml-notatsiya (дата обращения: 25.03.2023).
12. Зеленко Л.С. Презентация WinCC OA [Электронный ресурс] Л.С. Зеленко// Систем. требования: Power Point 2010// Самарский университет, 2023 (дата обращения: 25.03.2023).
13. Графический пользовательский интерфейс [Электронный ресурс] // Википедия: электрон. энциклопедия. 2001-2017. URL: https://en.wikipedia.org/wiki/Graphical\_user\_interface (дата обращения: 02.06.2023).ПРИЛОЖЕНИЕ А  
    Описание настроек точек данных

   
Рисунок А.1 – Точка данных «Купажный чан»    
Рисунок А.2 – Точка данных «Датчик уровня»

   
Рисунок А.3 – Точка данных «Датчик температуры»

   
Рисунок А.4 – Точка данных «Запорный клапан»

   
Рисунок А.5 – Точка данных «Регулируемый клапан»

   
Рисунок А.6 – Точка данных «Нагревательный эелемент»

ПРИЛОЖЕНИЕ Б  
Сценарии (скрипты управления)

Ручной режим управления

//Поворотный клапан

Slide(int value)

{

bool state = rand() % 100 == 0;

if (state) {

if (value != 0) {

dpSet("System1:AdjValve" + $valueNumber + ".Alarm.CantSetAngle", state);

} else {

dpSet("System1:AdjValve" + $valueNumber + ".Alarm.CantClose", state);

}

return;

}

dpSet("System1:AdjValve" + $valueNumber + ".State.Opened", value != 0);

dpSet("System1:AdjValve" + $valueNumber + ".State.Closed", value == 0);

dpSetWait("System1:AdjValve" + $valueNumber + ".State.Angle", value);

if (value != 0) {

startThread("run", value);

}

}

void run(int angle) {

float currentLevel;

dpGet("System1:LevelSensor" + $levelSensorId + ".State.Level", currentLevel);

float mainLevel;

dpGet("System1:LevelSensor" + $mainLevelSensorId + ".State.Level", mainLevel);

while (angle != 0 && currentLevel > 0 && mainLevel < 100){

dpGet("System1:AdjValve" + $valueNumber + ".State.Angle", angle);

delay(0.001);

currentLevel-= 0.001 \* 2 angle;

mainLevel+= 0.001 angle;

dpSetWait("System1:LevelSensor" + $levelSensorId + ".State.Level", currentLevel);

dpSetWait("System1:LevelSensor" + $mainLevelSensorId + ".State.Level", mainLevel);

}

}

//Запорный клапан

toggled(bool on)

{

bool state = rand() % 100 == 0;

if (state) {

if (on) {

dpSet("System1:StopValve" + $stopValveId + ".Alarm.CantOpen", state);

} else {

dpSet("System1:StopValve" + $stopValveId + ".Alarm.CantClose", state);

}

return;

}

dpSet("System1:StopValve\_" + $stopValveId + ".State.Closed:\_original..value", !on);

dpSet("System1:StopValve" + $stopValveId + ".State.Opened:\_original..value", on);

if (on) {

startThread("run", on);

}

}

void run(bool on) {

float currentLevel;

dpGet("System1:LevelSensor" + $levelSensorId + ".State.Level", currentLevel);

while (on && currentLevel < 100){

dpGet("System1:StopValve" + $stopValveId + ".State.Opened", on);

delay(0.001);

currentLevel+=0.01;

dpSetWait("System1:LevelSensor" + $levelSensorId + ".State.Level:\_original..\_value", currentLevel);

}

}

//Мешалка

// SimpleCtrlScriptStart {invalid}

main()

{

EP\_setEnabled();

if (!getUserPermission(6)) {

setValue("", "enabled", FALSE);

}

float currValue;

dpGet("System1:Vat\_main.State.MixerSpeed", currValue);

setValue("", "value", currValue);

}

void EP\_setEnabled()

{

dpConnect("EP\_setEnabledCB",

"System1:Vat\_main.Alarm.MixerJammed:\_online..\_value");

}

void EP\_setEnabledCB(string dp, bool boNewValue)

{

bool isAuto;

dpGet("System1:automode.State.isAuto", isAuto);

if (isAuto) {

setValue("", "enabled", FALSE);

return;

}

if (!getUserPermission(6)) {

setValue("", "enabled", FALSE);

return;

}

setValue("", "enabled", !boNewValue);

}

// SimpleCtrlScript {EP\_setEnabled}

// DP {System1:Vat\_main.Alarm.MixerJammed}

// DPConfig {:\_online..\_value}

// DPType {bool}

// Active {0}

// Operator {less}

// Value {TRUE}

// SimpleCtrlScriptEnd {EP\_setEnabled}

//Нагревательный элемент

// SimpleCtrlScriptStart {invalid}

main()

{

EP\_setEnabled();

if (!getUserPermission(6)) {

setValue("", "enabled", FALSE);

}

float currentValue;

dpGet("System1:MainTempSensor.State.Temperature", currentValue);

setValue("", "value", currentValue);

}

void EP\_setEnabled()

{

dpConnect("EP\_setEnabledCB",

"System1:Vat\_main.Alarm.MixerJammed:\_online..\_value");

}

void EP\_setEnabledCB(string dp, bool boNewValue)

{

bool isAuto;

dpGet("System1:automode.State.isAuto", isAuto);

if (isAuto) {

setValue("", "enabled", FALSE);

return;

}

if (!getUserPermission(6)) {

setValue("", "enabled", FALSE);

return;

}

setValue("", "enabled", !boNewValue);

}

// SimpleCtrlScript {EP\_setEnabled}

// DP {System1:Vat\_main.Alarm.MixerJammed}

// DPConfig {:\_online..\_value}

// DPType {bool}

// Active {0}

// Operator {less}

// Value {TRUE}

// SimpleCtrlScriptEnd {EP\_setEnabled}

Автоматический режим управления

void run(string dp, bool value) {

if (value) {

bool isRun;

dpGet("System1:automode.State.isRun", isRun);

delay(1);

if (!isRun) {

dpSet("System1:automode.State.isRun", TRUE);

dpSet("System1:automode.State.isFilled1", FALSE);

dpSet("System1:automode.State.isFilled2", FALSE);

dpSet("System1:automode.State.isMainFilled", FALSE);

dpSet("System1:automode.State.isFried", FALSE);

dpSet("System1:automode.State.currentCycle", 0);

startThread("autorun");

}

} else {

dpSet("System1:automode.State.isRun", FALSE);

dpSet("System1:StopValve\_1.State.Closed:\_original..\_value", TRUE);

dpSet("System1:StopValve\_1.State.Opened:\_original..\_value", FALSE);

dpSet("System1:StopValve\_2.State.Closed:\_original..\_value", TRUE);

dpSet("System1:StopValve\_2.State.Opened:\_original..\_value", FALSE);

dpSet("System1:AdjValve\_1.State.Opened", FALSE);

dpSet("System1:AdjValve\_1.State.Closed", TRUE);

dpSet("System1:AdjValve\_1.State.Angle", 0);

dpSet("System1:AdjValve\_2.State.Opened", FALSE);

dpSet("System1:AdjValve\_2.State.Closed", TRUE);

dpSet("System1:AdjValve\_2.State.Angle", 0);

}

}

void autorun() {

bool isRun;

dpGet("System1:automode.State.isRun", isRun);

clearMainVat();

dpSet("System1:StopValve\_1.State.Closed:\_original..\_value", FALSE);

dpSet("System1:StopValve\_1.State.Opened:\_original..\_value", TRUE);

startThread("fill1");

dpSet("System1:StopValve\_2.State.Closed:\_original..\_value", FALSE);

dpSet("System1:StopValve\_2.State.Opened:\_original..\_value", TRUE);

startThread("fill2");

bool isFilled1;

dpGet("System1:automode.State.isFilled1", isFilled1);

bool isFilled2;

dpGet("System1:automode.State.isFilled2", isFilled2);

while(!isFilled1 || !isFilled2) {

dpGet("System1:automode.State.isFilled1", isFilled1);

dpGet("System1:automode.State.isFilled2", isFilled2);

dpGet("System1:automode.State.isRun", isRun);

if (!isRun) {

return;

}

}

startThread("fillAdj");

bool isMainFilled;

dpGet("System1:automode.State.isMainFilled", isMainFilled);

while (!isMainFilled) {

dpGet("System1:automode.State.isMainFilled", isMainFilled);

dpGet("System1:automode.State.isRun", isRun);

if (!isRun) {

return;

}

}

startThread("startCooking");

bool isFried;

dpGet("System1:automode.State.isFried", isFried);

while (!isFried) {

dpGet("System1:automode.State.isFried", isFried);

dpGet("System1:automode.State.isRun", isRun);

if (!isRun) {

return;

}

}

clearMainVat();

}

void clearMainVat() {

dpSet("System1:HeatElem.State.Temperature", 15);

float mainVatLevel;

dpGet("System1:LevelSensor\_3.State.Level", mainVatLevel);

bool isRun;

dpGet("System1:automode.State.isRun", isRun);

dpSet("System1:StopValve\_3.State.Closed:\_original..\_value", FALSE);

dpSet("System1:StopValve\_3.State.Opened:\_original..\_value", TRUE);

while(isRun && mainVatLevel > 0){

delay(0.0005);

mainVatLevel-=0.01;

dpSetWait("System1:LevelSensor\_3.State.Level:\_original..\_value", mainVatLevel);

}

dpSet("System1:StopValve\_3.State.Closed:\_original..\_value", TRUE);

dpSet("System1:StopValve\_3.State.Opened:\_original..\_value", FALSE);

bool isFried;

dpGet("System1:automode.State.isFried", isFried);

int currentCycle;

dpGet("System1:automode.State.currentCycle", currentCycle);

int maxCycle;

dpGet("System1:automode.State.cycle", maxCycle);

if (isFried) {

dpSet("System1:automode.State.isFilled1", FALSE);

dpSet("System1:automode.State.isFilled2", FALSE);

dpSet("System1:automode.State.isMainFilled", FALSE);

dpSet("System1:automode.State.isFried", FALSE);

int currentCycle;

dpGet("System1:automode.State.currentCycle", currentCycle);

currentCycle = currentCycle + 600;

dpSet("System1:automode.State.currentCycle", currentCycle);

int maxCycle;

dpGet("System1:automode.State.cycle", maxCycle);

if (currentCycle < maxCycle) {

startThread("autorun");

} else {

dpSet("System1:automode.State.isAuto", FALSE);

dpSet("System1:automode.State.isRun", FALSE);

}

}

}

void fill1() {

float currentLevel1;

dpGet("System1:LevelSensor\_1.State.Level", currentLevel1);

bool isRun;

dpGet("System1:automode.State.isRun", isRun);

while (isRun && currentLevel1 < 100){

dpGet("System1:automode.State.isRun", isRun);

delay(0.0005);

currentLevel1+=0.01;

dpSetWait("System1:LevelSensor\_1.State.Level:\_original..\_value", currentLevel1);

}

dpSet("System1:StopValve\_1.State.Closed:\_original..\_value", TRUE);

dpSet("System1:StopValve\_1.State.Opened:\_original..\_value", FALSE);

dpSet("System1:automode.State.isFilled1", TRUE);

}

void fill2() {

float currentLevel2;

dpGet("System1:LevelSensor\_2.State.Level", currentLevel2);

bool isRun;

dpGet("System1:automode.State.isRun", isRun);

while (isRun && currentLevel2 < 100){

dpGet("System1:automode.State.isRun", isRun);

delay(0.0005);

currentLevel2+=0.01;

dpSetWait("System1:LevelSensor\_2.State.Level:\_original..\_value", currentLevel2);

}

dpSet("System1:StopValve\_2.State.Closed:\_original..\_value", TRUE);

dpSet("System1:StopValve\_2.State.Opened:\_original..\_value", FALSE);

dpSet("System1:automode.State.isFilled2", TRUE);

}

void fillAdj() {

bool isRun;

dpGet("System1:automode.State.isRun", isRun);

dpSet("System1:AdjValve\_1.State.Opened", TRUE);

dpSet("System1:AdjValve\_1.State.Closed", FALSE);

dpSet("System1:AdjValve\_1.State.Angle", 45);

float currentLevel;

dpGet("System1:LevelSensor\_1.State.Level", currentLevel);

float mainLevel;

dpGet("System1:LevelSensor\_3.State.Level", mainLevel);

while (isRun && currentLevel > 0 && mainLevel < 100){

dpGet("System1:automode.State.isRun", isRun);

delay(0.0005);

currentLevel-= 0.001 \* 2 \* 45;

mainLevel+= 0.001 \* 45;

dpSetWait("System1:LevelSensor\_1.State.Level", currentLevel);

dpSetWait("System1:LevelSensor\_3.State.Level", mainLevel);

}

dpSet("System1:AdjValve\_1.State.Opened", FALSE);

dpSet("System1:AdjValve\_1.State.Closed", TRUE);

dpSet("System1:AdjValve\_1.State.Angle", 0);

dpSet("System1:AdjValve\_2.State.Opened", TRUE);

dpSet("System1:AdjValve\_2.State.Closed", FALSE);

dpSet("System1:AdjValve\_2.State.Angle", 45);

dpGet("System1:LevelSensor\_2.State.Level", currentLevel);

while (isRun && currentLevel > 0 && mainLevel < 100){

dpGet("System1:automode.State.isRun", isRun);

delay(0.0005);

currentLevel-= 0.001 \* 2 \* 45;

mainLevel+= 0.001 \* 45;

dpSetWait("System1:LevelSensor\_2.State.Level", currentLevel);

dpSetWait("System1:LevelSensor\_3.State.Level", mainLevel);

}

dpSet("System1:AdjValve\_2.State.Opened", FALSE);

dpSet("System1:AdjValve\_2.State.Closed", TRUE);

dpSet("System1:AdjValve\_2.State.Angle", 0);

dpSet("System1:automode.State.isMainFilled", TRUE);

}

void startCooking() {

bool isRun;

dpGet("System1:automode.State.isRun", isRun);

dpSet("System1:Vat\_main.State.MixerSpeed", 60);

int mixerTimer;

dpGet("System1:automode.State.mixerTimer", mixerTimer);

for (int i = 0; i < mixerTimer && isRun; ++i) {

delay(1);

}

dpSet("System1:HeatElem.State.Temperature", 50);

float currentTemp;

dpGet("System1:MainTempSensor.State.Temperature", currentTemp);

while(isRun && currentTemp > 50) {

dpGet("System1:automode.State.isRun", isRun);

delay(0.001);

currentTemp-=0.01;

dpSet("System1:MainTempSensor.State.Temperature", currentTemp);

}

while(isRun && currentTemp < 50) {

dpGet("System1:automode.State.isRun", isRun);

delay(0.0005);

currentTemp+=0.01;

dpSet("System1:MainTempSensor.State.Temperature", currentTemp);

}

dpSet("System1:Vat\_main.State.MixerSpeed", 100);

dpSet("System1:HeatElem.State.Temperature", 90);

while(isRun && currentTemp < 90) {

dpGet("System1:automode.State.isRun", isRun);

delay(0.0005);

currentTemp+=0.01;

dpSet("System1:MainTempSensor.State.Temperature", currentTemp);

}

int heatTimer;

dpGet("System1:automode.State.heatTimer", heatTimer);

for (int i = 0; i < heatTimer && isRun; ++i) {

delay(1);

}

while(isRun && currentTemp > 15) {

dpGet("System1:automode.State.isRun", isRun);

delay(0.0005);

currentTemp-=0.01;

dpSet("System1:MainTempSensor.State.Temperature", currentTemp);

}

dpSet("System1:Vat\_main.State.MixerSpeed", 15);

dpSet("System1:automode.State.isFried", TRUE);

}