

Министерство образования и науки Российской Федерации

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИМЕНИ Н. Г. ЧЕРНЫШЕВСКОГО»

Кафедра математической
кибернетики и компьютерных наук

**РАЗРАБОТКА ИНФОРМАЦИОННОЙ СИСТЕМЫ ДЛЯ
ПРОГНОЗИРОВАНИЯ АВАРИЙ НА НПЗ**

АВТОРЕФЕРАТ МАГИСТЕРСКОЙ РАБОТЫ

студента 2 курса 271 группы
направления 09.04.01 — Информатика и вычислительная техника
факультета КНиИТ
Белякова Михаила Алексеевича

Научный руководитель
профессор, д. т. н.

В. А. Кушников

Заведующий кафедрой
к. ф.-м. н.

С. В. Миронов

Саратов 2018

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	3
1 Метод структурного анализа ФТА	5
1.1 Применение метода ФТА	5
2 Выбор инструментальных средств для разработки приложения	6
3 Описание приложения	7
3.1 Реализация базы данных	7
3.2 Структура файлов	9
3.3 Рассмотрение самого приложения	12
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	14

ВВЕДЕНИЕ

Процесс эксплуатации технических средств, нуждается в обеспечении безопасности, которую можно повысить благодаря увеличению отказоустойчивости технической системы. Чтобы понять какой элемент системы может представлять потенциальную опасность, проводятся процедуры исследования опасности методом структурного анализа FTA – анализ дерева отказов.

Анализ дерева отказов (АДО) или в английской терминологии FTA метод анализа отказов в сложных системах, в котором отрицательные состояния или отказы системы анализируются с помощью булевой алгебры. Происходит объединение последовательности нижестоящих событий (отказов низшего уровня), которые приводят к отказу всей системы.

Анализ дерева отказов активно используется в различных отраслях, например, машиностроении, для того чтобы понять, как система может перейти в состояние отказа, выявить способы уменьшения рисков или определения частоты системного отказа.

Удаленный сбор данных и прогнозирование особенно необходимы на инфраструктурных объектах и системах просто в силу их больших географических размеров, а значит и большей аварийности. Примером такого объекта является нефтеперерабатывающий завод. Аварии на таких заводах способны нанести ущерб не только нефтяной компании, но и превратить регион в зону экономического бедствия. Пожароопасность процессов, происходящих на НПЗ, требует практически мгновенной реакции на изменение параметров нагрузок, состояния оборудования. Наличие опасных факторов НПЗ, сложность производственных процессов, рост мощностей и размеров применяемого оборудования – все это требует предоставления управляющему персоналу специализированных систем для сбора контрольных параметров, их визуализации, приоритизации и первичного анализа с целью прогнозирования и дальнейшего предотвращения возможных аварийных ситуаций. Как только произошла коммерческая реализация ЭВМ, они стали применяться так или иначе в технических процессах, а также в качестве системы контроля, что позволяет применить информационную систему для прогнозирования аварий и катастроф на НПЗ.

Цель работы. В связи с вышеизложенным целью работы является разработка информационной системы для прогнозирования аварий и катастроф

на НПЗ. Для достижения цели необходимо выполнить следующие задачи:

- осуществить системный анализ процесса функционирования нефтеперерабатывающего предприятия;
- обосновать выбор инструментальных средств, используемых для разработки, отладки и тестирования информационной системы;
- разработать графический интерфейс пользователя информационной системы;
- выполнить проектирование базы данных;
- разработать, отладить и протестировать информационную систему, используя тестовые примеры, предоставляемые специалистами ИПТМиУ РАН.

1 Метод структурного анализа ФТА

Этот метод интересен тем что в нем осуществлен дедуктивный метод (причина - следствие), что делает его подходящим для поиска корневых источников событий для статичных систем, вследствие того, что дает подробную и наглядную схему корреляций элементов событий и инфраструктуры, влияющих на их надежность. В нем применяются логические знаки И/ИЛИ, которые соединяют события в соответствии с их причинными взаимосвязями. Логический символ может иметь один или несколько входов, но только одно выходное событие, или один выход.

ФТА включает 5 шагов:

- Определенное нежелательно событие.
- Глубокое понимание причин.
- Построение дерева отказов на основе изученных причин.
- Оценка дерева отказов.
- Контроль определения опасности.

1.1 Применение метода ФТА

Взяв на вооружение метод ФТА, было построено дерево событий (отказов) НПЗ, которое имеет в качестве конечного событий возникновение пожара, охватывающего значительную часть завода. Но на данном этапе научно-исследовательской работы, были использованы только символы логических знаков: конъюнктор, дизъюнктор.

2 Выбор инструментальных средств для разработки приложения

Для создания информационной системы были использованы следующие инструменты:

- Java – используется для создания бек-енда веб приложения;
- JSP – используется для обработки страниц, позволяет отделить динамическую часть страниц от статического HTML;
- Bootstrap – этот CSS-фреймворк используется для верстки страниц;
- SQL – язык структурированных запросов используется для взаимодействия с БД;
- JDBC – драйвер соединения информационной системы с БД;
- Treant.js – это специальная библиотека может строить деревья любого вида, основываясь на данных полученных из любого источника.

3 Описание приложения

Приложение написано на языке Java версии 8.131 и использует технологию JSP. Для верстки страницы применялся фреймворк bootstrap.

3.1 Реализация базы данных

Применялась реляционная БД. Реляционная база данных — это совокупность взаимосвязанных таблиц, каждая из которых содержит информацию об объектах определенного типа. Строка таблицы содержит данные об одном объекте (например, товаре, клиенте), а столбцы таблицы описывают различные характеристики этих объектов — атрибутов (например, наименование, код товара, сведения о клиенте).

Приложение использует для взаимодействия с БД систему управления базами данных MySQL.

База данных состоит из 11 таблиц:

- positions - для должностей
 - уникальный идентификатор
 - наименование должности
- departments - таблица отделов
 - уникальный идентификатор
 - наименование отдела
- employeers - таблица сотрудников
 - уникальный идентификатор
 - ФИО сотрудника
 - идентификатор должности
 - идентификатор подразделения
- help - таблица со справкой для работы с системой
 - уникальный идентификатор
 - заголовок статьи
 - содержимое
- tests - таблица тестов
 - уникальный идентификатор
 - дата
 - результат
 - идентификатор элемента системы

- characteristics - таблица характеристик
 - уникальный идентификатор
 - наименование
 - значение
 - идентификатор элемента системы
- reports - таблица отчетов
 - уникальный идентификатор
 - дата создания
 - тип
 - идентификатор пользователя
- archive - таблица архивирования
 - уникальный идентификатор
 - дата
 - идентификатор пользователя
- logs - таблица пользовательских действий
 - уникальный идентификатор
 - действие
 - имя пользователя
 - элемент
 - дата и время добавления
- users - таблица пользователей
 - уникальный идентификатор
 - логин
 - зашифрованный пароль
 - соль
 - идентификатор сотрудника
 - права доступа
- data - таблица элементов
 - уникальный идентификатор
 - наименование
 - предыдущий элемент
 - операция
 - идентификатор сотрудника, ответственного за данный элемент
 - описание

- изображение
- дата добавления

3.2 Структура файлов

Структура классов имеет следующий вид:

- controller
 - ArchiveService.java - сервис для получения, добавления и редактирования записей о создании архива
 - CharacteristicService.java - сервис для получения, добавления и редактирования записей о характеристиках элемента
 - DepartmentService.java - сервис для получения, добавления и редактирования записей о подразделении
 - ElementService.java - сервис для получения, добавления и редактирования записей о элементах системы.
 - EmployeeService.java - сервис для получения, добавления и редактирования записей о сотрудниках
 - HelpService.java - сервис для получения, добавления и редактирования записей в справке
 - LogsService.java - сервис для записи и чтения логов
 - MySQL.java - сервис для работы с MySQL.
 - PositionService.java - сервис для получения, добавления и редактирования записей о должностях
 - ReportService.java - сервис для получения, добавления и редактирования записей о создании отчётов
 - TestService.java - сервис для получения, добавления и редактирования записей о тестах
 - Tree.java - сервис для отображение дерева поломок
 - UserService.java - сервис для получения, добавления и редактирования записей о пользователях
- model
 - Archive.java - модель для таблицы архивов
 - Characteristic.java - модель для таблицы характеристика
 - Department.java - модель для таблицы отделов
 - Element.java - модель для таблицы элементов системы
 - Employee.java - модель для таблицы сотрудников

- Help.java - модель для таблицы справки
- Logs.java - модель для таблицы логов
- Position.java - модель для таблицы должностей
- Report.java - модель для таблицы отчётов
- Test.java - модель для таблицы тестов
- User.java - модель для таблицы пользователей

Структура страниц приложения выглядит следующим образом:

- WEB-INF
 - web.xml - конфигурационный файл
- actions - директория с действиями
 - add_department.jsp - страница добавления подразделения
 - add_element.jsp - страница добавления элемента
 - add_employee.jsp - страница добавления сотрудника
 - add_help.jsp - страница добавления справки
 - add_position.jsp - страница добавления должности
 - delete_help.jsp - страница удаления справки.
 - edit_department.jsp - страница редактирования подразделения
 - edit_element.jsp - страница редактирования элемента
 - edit_employee.jsp - страница редактирования сотрудника
 - edit_help.jsp - страница редактирования справки
 - edit_position.jsp - страница редактирования должности
 - edit_user.jsp - страница редактирования пользователя
- archive - директория для работы с архивом
 - index.jsp - страница списка архивирования
 - load.jsp - страница загрузки данных
 - save.jsp - страница сохранения данных
- auth.jsp - страница авторизации
- core - директория с ключевыми файлами
 - footer.jsp - нижняя часть страницы
 - head.jsp - верхняя часть страницы
 - menu.jsp - элемент меню
- css - файлы стилей
 - Treant.css
 - bootstrap-material-design.min.css

- bootstrap-theme.min.css
- bootstrap.min.css
- ripples.min.css
- template.css
- departments - работа с подразделениями
 - add.jsp - страница создания
 - edit.jsp - страница редактирования
 - index.jsp - страница просмотра подразделений
- elements - работа с элементами системы
 - add.jsp - страница добавления
 - edit.jsp - страница редактирования
 - index.jsp - страница просмотра списка элементов системы
- employees - работа с сотрудниками
 - add.jsp - страница добавления
 - edit.jsp - страница редактирования
 - index.jsp - страница просмотра списка сотрудников
- fonts - директория для шрифтов
 - glyphsicons-halflings-regular.eot
 - glyphsicons-halflings-regular.svg
 - glyphsicons-halflings-regular.ttf
 - glyphsicons-halflings-regular.woff
 - glyphsicons-halflings-regular.woff2
- help - директория для работы со справкой
 - add.jsp - страница добавления
 - edit.jsp - страница редактирования.
 - index.jsp - страница просмотра справки.
- index.jsp - страница авторизации
- js - директория со скриптами
 - Treant.js
 - bootstrap.min.js
 - jquery.maskedinput.min.js
 - material.min.js
 - raphael.js
 - ripples.min.js

- template.js
- logout.jsp - страница выхода из системы
- logs
 - index.jsp - страница логов
- positions - работа с должностями
 - add.jsp - страница добавления
 - edit.jsp - страница редактирования
 - index.jsp - страница просмотра списка должностей
- reg.jsp - страница регистрации
- register.jsp - страница с формной регистрации
- reports - работа с отчётами
 - breaks.jsp - отчёт по авариями
 - employees.jsp - отчёт по сотрудникам
 - index.jsp - список полученных отчётов
 - logs.jsp - отчёт по логам
- tree
 - index.jsp - страница дерева
- users - работа с пользователями
 - edit.jsp - страница редактирования
 - index.jsp - страница просмотра списка пользователей

3.3 Рассмотрение самого приложения

Работа с приложением осуществляется посредством пользовательского интерфейса. Пользовательский интерфейс (user interface - UI) – интерфейс, обеспечивающий передачу информации между пользователем-человеком и программно-аппаратными компонентами компьютерной системы. Интерфейс пользователя позволяет добавлять, удалять и изменять элементы системы, просматривать дерево, осуществлять управление пользователями системы, сохранять данные во внешний файл и загружать из него информацию. Все данные хранятся в базе данных.

Когда входит в программу то он должен осуществить авторизацию, после чего перед ним предстает меню программы.

Также у каждого элемента имеются характеристики, которые можно вносить через интерфейс.

При редактировании элемента указывается предыдущий элемент – бла-

годаря чему можно построить дерево событий, которое демонстрируется после нажатия кнопки «Дерево отказов». Построение производится с помощью библиотеки Treant.js.

На странице помощи можно просмотреть, добавить, отредактировать и удалить справку. Страница разделена на статьи, для повышение удобства пользователя.

На странице отчётов можно просмотреть список созданных отчётов и создать:

- Отчёт по поломкам
- Отчёт по сотрудникам
- Отчёт по действиям

Страница персонал позволяет увидеть список работников. Имеются возможности добавления и редактирования записей о сотрудниках.

На нефтеперерабатывающем заводе ответственность распределяется между отделами, поэтому в информационной системе имеется соответствующая страница.

Каждый работник НПЗ имеет определенную должность, которая привязывает его к определенному отделу, что отмечено на странице должностей, где также имеются интерфейс для их редактирования. На странице логов показан список действий всех пользователей. На странице архивирования можно посмотреть список пользователей, которые делали копию данных и время архивирования, а также можно архивировать данные и загрузить из копии.

Страница пользователей нужна для редактирования прав пользователя

* привязки сотрудника к пользователю и доступна только администраторам системы.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1 Сейран Артюнович Фарамазов ЭКСПЛУАТАЦИЯ ОБОРУДОВАНИЯ НЕФТЕПЕРЕРАБАТЫВАЮЩИХ ЗАВОДОВ, Издательство “Химия”, М., 1969 г. 304 с.
- 2 Константин Яковлевич Иванец, Анатолий Никанорович Лейбо, Оборудование нефтеперерабатывающих заводов и его эксплуатация, Издательство “Химия”, М., 1966 г. 344 с.
- 3 Ликвидация аварийных ситуаций [Электронный ресурс] <http://sait-spasatel.ru/larn/vidy-planirovaniya-larn.html/> (Дата обращения 31.04.2018). Загл. с экр. Яз. Рус.
- 4 Крупнейшие аварии и катастрофы на предприятиях РФ за 1991 -2015 гг. [Электронный ресурс] http://www.propylen-glycol.ru/avarii_rf.php/ (Дата обращения 31.04.2018). Загл. с экр. Яз. Рус.
- 5 Архив информационных бюллетеней [Электронный ресурс] <http://ib.safety.ru/> (Дата обращения 31.04.2018). Загл. с экр. Яз. Рус.
- 6 Аварии на предприятии нефтепереработки и их последствия [Электронный ресурс] <https://www.bibliofond.ru/view.aspx?id=649300> (Дата обращения 31.04.2018). Загл. с экр. Яз. Рус.
- 7 Пожары на НПЗ [Электронный ресурс] <http://www.oilru.com/news/400868/> (Дата обращения 31.04.2018). Загл. с экр. Яз. Рус.