**Слайд 1.** Здравствуйте уважаемые члены аттестационной комиссии. Я предлагаю вашему вниманию свой дипломный проект на тему **«Разработка автоматизированной системы хранения программного кода и документации для станков с ЧПУ»**.

**Целью моего дипломного проекта является: Автоматизация процесса наладки управляющей части станка с ЧПУ**.

**Слайд 2.** Сейчас я для лучшего понимания вкратце расскажу вам о станках, с которыми я работаю.

Используются немецкие станки «Monforts». Станок можно разделить на 2 основные части: управляющую и рабочую. Рабочая часть станка включает в себя: инструменты, которые крепятся револьверную головку, которая, по сути, является держателем инструментов, зажим для деталей, и т.д.

Процесс наладки также можно разбить на 2 части и, прежде всего, происходит наладка управляющей части. Первым этапом наладки управляющей части является поиск технологического процесса и программы, необходимой для обработки деталей.

**Слайд 3.** Изображён процесс наладки рабочей части.

**Слайд 4.** Управляющая часть – это просто компьютер с ОС Windows XP, в которую интегрирована среда Simens Sinumerik 840D. Siemens Sinumerik – полностью цифровая система для практически всех типов применений. Во всем мире Sinumerik применяется для: токарной обработки, сверления, фрезерования, шлифования, лазерной обработки, порезки, перфорации, изготовления оснастки и инструмента, как система управления прессами, для высокоскоростного раскроя материалов, обработки древесины и стекла, транспортировки, складских задач.

В управляющей части есть сетевое гнездо, которая даёт возможность образовывать из станков компьютерную сеть.

**Слайд 5.** Так выглядит среда Simens Sinumerik 840D, которая предназначена для разработки программ и управления станком.

**Слайд 6.** В ходе дипломного проектирования были сформулированы и, впоследствии, решены следующие основные задачи (читать со слайда). Мы хотим перейти от ручного процесса наладки управляющей части к автоматизированному.

**Слайд 7.** В настоящее время поиск конкретного технологического процесса и программы является неупорядоченным и бессистемным. Что влечет за собой потери времени, повышает вероятность возникновения ошибок в процессе наладки, которые могут нанести серьёзный ущерб, как оборудованию, так и здоровью человека. Разработанная централизованная система, объединяет в себе программы и технологические процессы, доступ к которым является легким и быстрым. Доступ осуществляется из любого места, где есть подключение к интернету (например, с самого станка). Эти свойства системы приводят к сокращению затрат времени и сил, снижают вероятность возникновения ошибок и укрепляет производственную дисциплину в процессе наладки станочного оборудования.

**Слайд 8.** Для реализации системы была выбрана клиент-серверная архитектура.

**Серверная** часть исполняется на сервере, выполняет хранение данных, все операции с данными, проверку прав и т.д.

**Клиентская** часть – это **пользовательский интерфейс** и она выполняется в **браузере** на компьютере каждого пользователя. С точки зрения пользователя это обычный сайт.

**Слайд 9.** На этапе выбора, каким именно должно быть приложение, рассматривались 2 основных типа приложений: обычное настольное и приложение с вэб-интерфейсом.

Что касается обычного настольного приложения, то при его использовании возникают следующие трудности

* Для доступа к файлам, каждый раз необходимо будет устанавливать приложение туда, где оно будет использоваться (в данной работе этот недостаток является неприемлемым). Проблематично установить на станок.
* Сложность в реализации нужного уровня защиты информации.

Для выполнения данного проекта, была выбрана реализация в виде Web-приложения. Основными предпосылками к этому стали:

* возможность использования такого приложения из любого места, где есть компьютер и подключение к интернету, т.к. зачастую то или иное производство разделено территориально (находится в разных цехах, производственных участках), но, при этом, имеет одни и те же задачи;
* web-приложение не требует установки;
* наличие централизованного сервера – это дает возможность объединить в единой базе данных необходимые для работы файлы с программами и документацией, структурировать их и осуществлять быстрый поиск, что позволяет экономить массу времени и предотвратить возникновение ошибок, которые являются следствием “неразберихи” в процессе наладки оборудования.

Также, такие качества как надежность, низкие требования к компьютерным ресурсам, возможность модернизации, наличие централизованного обновления, высокая степень защиты и постоянное развитие в программировании Web-приложений, делают данный способ реализации актуальным и востребованным.

**Слайд 10.** Для разработки были использованы:

Denwer – Джентельменский набор веб-разработчика, включающий в себя сервер Apache, язык программирования PHP, базу данных MySQL.

PHPStorm – удобная среда разработки для PHP/HTML/CSS/JavaScript. C CSS практически не работал, используется стандартное оформление для Yii Framework.

DBForge Studio for MySQL – для разработки структуры базы данных. Этот инструмент позволяет визуально редактировать ER-диаграммы и сразу получать готовую MySQL базу данных.

**Слайд 11.** База данных хранит все необходимые сведения. В ней хранятся сведения о пользователях системы (в таблице user \*показать на слайде\*), хранятся тексты программы в таблице programma, версии этих текстов (если программа имеет несколько версий) в таблице text\_programmy, технологические процессы в таблице chertezh, станки в таблице stakon и типы станков в таблице: vid\_stanka. Подробное описание полей есть в пояснительной записке.

**Слайд 12.** Всё понятно. \*рассказывать по слайду\*

**Слайд 13.** На слайде приведена одна из центральных форм пользовательского интерфейса. Таблица программ. В этой форме можно просматривать список программ. Осуществлять поиск по значениям отдельных полей. Например, можно ввести название (или часть названия) программы и искать этому полю. Можно отсортировать программы по названиям, автору, типу станка, дате создания программы. Для сортировки нужно нажать на заголовок таблицы. При нажатии на название программы происходит переход на страницу этой программы, где указано название, тип станка, автор, дата, а также привязанные к этой программе чертежи и версии программы.

**Слайд 14.** Страница пользователя системы.

**Слайд 15.** Страница определённого типа станка. Внизу страницы показываются программы, которые предназначены для данного типа станка и все станки данного типа. По ссылкам можно перейти на страницы программ и на страницы станков.

**Слайд 16.** В результате была \*читаем со слайдов\*..

Доклад закончил, готов ответить на ваши вопросы.