Оглавление

[1. Специальная часть 3](#_Toc357852736)

[1.1. Введение 3](#_Toc357852737)

[1.2. Обзор программного обеспечения 10](#_Toc357852738)

[1.3. Постановка задачи 18](#_Toc357852739)

[1.4. Разработка технического задания 20](#_Toc357852740)

[1.5. Метод анализа подписи 23](#_Toc357852741)

[2. Конструкторско-технологическая часть 45](#_Toc357852742)

[2.1. Выбор жизненного цикла информационной системы 45](#_Toc357852743)

[2.2. Инструментальные средства разработки и обзор СУБД 52](#_Toc357852744)

[2.3. Реализация на ЭВМ 59](#_Toc357852745)

[2.3.1. Структура классов 59](#_Toc357852746)

[2.3.2. Структура данных 62](#_Toc357852747)

[2.3.3 Пример функционирования 64](#_Toc357852748)

[3. Охрана труда 71](#_Toc357852749)

[3.1 Опасные факторы при работе с ЭВМ 73](#_Toc357852750)

[3.2. Требования к освещению на рабочих местах, оборудованных ЭВМ 74](#_Toc357852751)

[3.3 Требования к качеству восприятия информации, отображаемой на дисплеях 76](#_Toc357852752)

[3.4. Эргономические требования к цветовым параметрам 78](#_Toc357852753)

[3.5. Требования безопасности к визуальным параметрам 79](#_Toc357852754)

[3.6. Искажения изображения по рабочему полю 79](#_Toc357852755)

[3.7. Требования безопасности к параметрам создаваемых полей 80](#_Toc357852756)

[3.8. Требования к конструкции 81](#_Toc357852757)

[3.9. Выводы 82](#_Toc357852758)

[4. Экологическая часть 83](#_Toc357852759)

[4.1. Влияние освещения на организм человека 83](#_Toc357852760)

[4.2. Влияние электромагнитного излучения на организм человека 85](#_Toc357852761)

[4.3. Выводы 87](#_Toc357852762)

[Заключение 89](#_Toc357852763)

[Список используемой литературы: 90](#_Toc357852764)

# 1. Специальная часть

## 1.1. Введение

История графологии чуть более трех сот лет, несмотря на то, что в древних документах можно найти подтверждение тому, что интерес к ней проявляли даже такие известные личности как Нерон и Конфуций.

Первый известный трактат по графологии был написан в в 1622 году. Автор, профессор медицины Болонского университета Камило Бальдо назвал ее «Как оценить природу и характер человека по его почерку». Согласно его утверждению, совершенно очевидно, что каждый пишет по-особому, при этом получаются характерные формы букв, которые не могут быть точно скопированы другим.

Другой итальянец, современник Бальдо, профессор анатомии и хирургии из Неаполя **Марк Аврелий Северин**пишет в 1650 году труд, названный «Пророк, или предсказание по почерку».

Цюрихский проповедник **Йоган Каспар Лафатер**(1741-1801) становится известен благодаря своей основательной четырехтомной монографии «Элементы физиогномики». В третьем томе он касается анализа почерка.

Евангелистский теолог **Адольф Хенце**в 1862 году выпускает труд под названием «Chirogrammatomantie». Хенце обладал богатейшим собранием образцов почерка. Говорили, что он проанализировал до 70 000 документов.

**Жан-Ипполит Мишон.**До 1875 года графология, которая тогда еще даже не имела этого названия, оставалась настолько экзотичной, что даже в кругах интеллектуалов того времени о ней не знали. Именно этот год становится переломным. Аббат Жан-Ипполит Мишон (1806-1881) издает труд «Система графологии».

Чем замечательна эта работа? Во-первых, Мишон ввел теперь уже привычный термин «графология», образованный от греческих слов «графо» – писать и «логос» – наука, знание. Во-вторых, его работа написана просто и доступно и относится к жанру научно-популярной литературы.

Принцип Мишона прост: один признак – одна психологическая черта. Если этот признак присутствует, то человек обладает данной чертой характера, а если отсутствует – обладает противоположной чертой. Это явное упрощение. Сегодня такой подход в чистом виде назвали бы дилетантством и ересью. Тем не менее многие из выявленных таким образом закономерностей общеприняты и по сей день.

**Жюль Крепье-Жамен.**Один из учеников Мишона, француз Жюль Крепье-Жамен (1858-1940), посвятил жизнь дальнейшему развитию результатов работы своего учителя. Он не во всем согласен с Мишоном, особенно с жесткой привязкой одного признака к одной черте характера. Крепье-Жамен вводит понятие гармонии или дисгармонии как некой обобщенной характеристики. Он формулирует набор групп признаков или категорий, который до сегодняшнего дня принят во французской школе графологии: форма, размер, нажим, скорость, наклон, связность и расположение. При этом, отступая от принципа Мишона, он утверждает, что любой признак в отдельности еще не может указывать на определенную черту характера. Интерпретация конкретного признака почерка зависит от параллельно присутствующих других его характеристик.

Крепье-Жамен переиздает переработанный труд Мишона. Это новое издание становится (и остается до сих пор) одним из основополагающих учебников по графологии. Он основывает Французское общество графологии. Таким образом, Крепье-Жамен фактически является основоположником современной французской графологии. Он становится первым профессиональным графологом.

Профессор университета в Йене **Вильгельм Прайер** (1841-1897), автор «Психологии письма» (Zur Psychologie des Schreibens), вышедшей в свет в 1895 году, интерпретирует письмо как результат импульсов, поступающих из головного мозга. Он исследует, в частности, почерк людей, потерявших руки. При этом он пытается доказать, что основные признаки почерка сохраняются, если человек начинает писать левой рукой, ногой или даже ртом.

**Георг Майер.** Майер (1869-1917) соотносит написанное с эмоциями писавших. Он исследует маниакально-депрессивных пациентов и приходит к выводу, что характеристики почерка являются производными психики. Наблюдения и выводы он публикует под названием «Научные основы графологии» (Die wissenschaftlichen Grundlagen der Graphologie).

**Людвиг Клагес.**Доктор Людвиг Клагес (1872-1956) является для немецкой школы тем же, кем был Крепье-Жамен для французской. Он становится основателем научной графологии в Германии. Его фундаментальная работа «Почерк и характер», которая вышла в 1916 году, переиздавалась до 1989 года.

В 1896 году Клагес и Майер основывают в Мюнхене Немецкое графологическое общество.

**Роберт Саудек.**Роберт Саудек (1880-1935), чех по происхождению, писатель, живший в Англии, вводит в графологию понятие скорости. Он проводит много экспериментов, изучая двигательные аспекты письма. При этом Саудек использует даже киносъемку. Он устанавливает интересные закономерности: например, замечает, что длинные штрихи выполняются быстрее коротких, что запятая ставится быстрее точки, что движение при письме замедляется при изменении направления. Свои результаты он изложил, в частности, в книге «Эксперименты с почерками» (Experiments with Handwriting), вышедшей в 1929 году.

**Макс Пульвер.** В предисловии к своей выдающейся монографии «Символизм почерка» (Symbolik der Handschrift), вышедшей в 1931 году (за ней последуют еще две монографии – в 1934-м и 1949-м), Пульвер (1889-1952) пишет о своей цели исследовать графологию с позиций феноменологии и психоанализа.

Пульвер обращается к психоаналитической концепции подсознания и символа как средства выражения. Для него «сознательное письмо есть подсознательное рисование». Он вводит в графологию понятие символизма пространства и методически применяет его при анализе различных характеристик почерка.

Почерк у Пульвера имеет трехмерное измерение. Помимо высоты и ширины он анализирует глубину почерка. Пульвер исследует также направления движений (координаты) при письме сверху вниз и справа налево, при этом левая часть связывается с прошлым человека, а правая – с будущим.

С 1924 года Пульвер преподает в Цюрихском университете. В 1950-м он основывает Швейцарское графологическое общество, председателем которого является до своей смерти в 1952 году.

**Анна Тайллард.** В 20-е годы прошлого века Тайллард (1889-1978) учится под руководством Клагеса и занимается графологическими исследованиями, поддерживая связи с Крепье-Жаменом, Пульвером, Саудеком. Определяющим моментом в ее работе становится обучение психоанализу в Цюрихе непосредственно у Юнга.

Более чем 20-летняя совместная работа с Юнгом определяет интересы Тайллард в графологии. Она начинает разрабатывать новый графологический язык, основанный на теории психологической глубины.

В ноябре 1945 года в одной из своих лекций Тайллард предлагает использовать психологические типы Юнга в графологии. Наиболее известная ее работа «Толкование почерка» (Handschriftdeutung), вышедшая в Мюнхене в 1952 году, относится доктором Хельмутом Плугом, председателем Объединения профессиональных графологов Германии, к десяти самых важных книг по графологии. Тайллард внесла большой вклад в преодоление различий между отдельными европейскими школами и интеграцию различных методов графологии.

**Рода Визер.**Рода Визер (1894-1986) была последовательницей Людвига Клагеса, введшей понятие основного ритма (Grundrhytmus). Любопытно, что работала она в Венском институте криминалистики, тем самым как бы связывая оба направления использования почерка.

**Рудольф Попал.**Рудольф Попал (1893-1966), профессор Гамбургского университета, пришел в графологию из медицины. Развивая нейрологическое направление исследования почерка, заложенное в свое время Вильгельмом Прайером, он исследовал, как различные моторные центры головного мозга влияют на движения пишущего и определяют его почерк. Попал выделял четыре нейропсихологических типа и, соответственно, четыре вида или стиля почерка. Хотя современная нейрология исходит из других принципов, результаты работ Попала продолжают иметь большое значение для исследования почерка.

**Роберт Хайс.**Роберт Хайс (1903-1974) был директором института психологии при университете Фрайбурга в Германии. В психологии он развивал направление, рассматривающее человека не как статическую структуру или совокупность качеств, а как процесс, постоянное становление. В графологии Хайс ввел систематику, рассматривающую три основных аспекта почерка: движение, форму и пространство. Такое представление по сей день доминирует в немецкой школе графологии и почти в каждой книге ее приверженцев признаки почерка группируются по этим трем категориям.

**Клара Роман.** Так же, как и Рудольф Попал, Клара Роман исходила из нейрологических основ почерка. Клара Роман тщательнейшим образом занималась изучением нажима, используя даже специальный прибор для его непосредственного измерения во время письма. Кроме того, она первой стала использовать в графологии психодрамы, то есть графическое представление психологических характеристик.[1]

Последние десятилетия, в сущности, характеризуются отсутствием новых идей и направлений. Зато практическая графология распространяется довольно активно. Директоры и президенты серьезных фирм и корпораций считают, что почерк может дать им ключи к информации о сильных и слабых личностных качествах кандидатов на работу, конкурентов, партнеров, рассказать о проблемах их здоровья. Ведущие психологи и графологи считают, что в наше время почерк хорошо изучен. Он является подсознательным актом, которым управляет сознание. Анализ почерка основан на идеомоторной теории, идее о том, что мысль, накапливающаяся в сознательной области, создает напряженность в мышцах те­ла, в том числе и тех, что задействованы в процессе письма. Даже мельчайшие движения передаются ручке, поэтому они обнаруживаются в почерке. Часто дипломированных специалистов высшего класса в США спрашивают, не являются ли они экстрасенсами или телепатами, поскольку результаты анализа почерка дают очень точные результаты.[2]

Валидность графологии по результатам эксперимента Мари Анн Ноэр на основе сравнения результатов Центра оценки персонала и результатов графологического анализа почерка составляет 90%.[29]

В 1979 году графологию включили в Десятичную классификацию Дьюи (По этой системе расставляют книги на библиотечных полках в более чем 200 000 библиотеках в 135 странах мира). Книги по графологии стали относить к таким разделам как «Диагностическая графология», «Экспертиза документа» и «Подбор персонала».[28]

На Западе графология получила широкое распространение. Кроме Франции, графология развита во многих странах и особенно в Германии, Голландии, Израиле и США. В Голландии имеются две профессорские кафедры (в университетах Лейдена и Утрехта) и два графологических общества (в Амстердаме и Гааге). В США создан Институт почерка (Нью-Йорк), а также Американское графологическое общество.

Во многих странах по всему миру действуют различные институты, общества, ассоциации, созданные для изучения и популяризации графологии.

В Австралии: Австралийский Институт Графологии основан в 2005 году.

В Аргентине: Католический Университет Аргентины Санта-Мария (Буэнос-Айрос); Эмерсон Колледж (Университет) (Буэнос-Айрос).

В Бельгии: Бельгийская Ассоциация Анализа почерка основана в 1972 году.

В Великобритании: Британская Графологическая Академия (Лондон) основана в 1985 году; Колледж Графологии (Лондон) основан в 1985 году.

В Венгрии: Институт графологии (Будапешт) основан в 1991 году.

В Германии: Профессиональная Ассоциация сертифицированных графологов/психологов (Мюнхен).

В Евросоюзе: Европейская Деонтологическая Ассоциация Графологов (Объединяет европейские графологические ассоциации с целью контроля качества их работы, соблюдения принципов деонтологии и соответствия высокому научному уровню). Основана в 1994 году.

В Израиле: Израильское Общество Научной Графологии основано в 1977 году доктором Арье Нафтали, графологом с мировым именем, преподававшем в Институте Криминалистики Иерусалимского Университета, автором книги «Графология и Здоровье». Также он основал Институт Нафтали в 1969 году.

В Испании: Институт Научной Графологии; Испанская Графологическая Ассоциация (Мадрид) основана в 1979 году.

В Италии: Итальянская Графологическая Ассоциация (Анкона) основана в 1961 году; Итало-французская Графологическая Ассоциация (Рим) основана в 1981 году; Итальянская Исследовательская Графологическая Ассоциация (Рим) основана в 1981 году.

В Китае: Китайская Графология (Management Consulting, Ltd ) (Пекин) 2007 год.

В Нидерландах: Голландская Графологическая Ассациация (Роттердам) основана в 1929.

В США: Американское Общество Профессиональных Графологов (Нью-Йорк).

Во Франции: Союз и профессиональная организация подготовки графологов (Париж) основан в 1946 году; Французское Графологоческое Общество (Париж) основано в 1871 году Жаном-Ипполитом Мишоном и признано общественностью в 1971 году; Итало-французская Графологическая Ассоциация (Рим); Синдикат квалифицированных специалистов из Французского Графологического Общества (Марсель) основан в 1990 году.

В Чехии: Чешская Графологическая Палата (Прага) основана в 1992 году.

В Швейцарии: Щвейцарское Графологическое Общество (Цюрих) основано в 1950 году Максом Пулвером.

В ЮАР: Институт Профессиональной Графологии.[29]

По оценкам некоторых психологов отдельные разработки в графологии по сравнению с рядом других методов изучения личности дают больше информации.

На сегодня услугами графологов в Соединенных Штатах пользуются такие известные кампании как Ford, General Electric, Mutual of Omaha, H&R Block, Firestone, USX Corp., Olsten, Honeywell Inc. и другие. ЦРУ и ФБР в обязательном порядке использует графологическое тестирование при подборе кандидатов на службу. Там считается, что благодаря графологическому анализу можно диагностировать не только отдельные черты психики человека, тип его характера, но и установить его хорошие и плохие привычки, умение обращаться с деньгами, склонность к алкоголизму или наркомании. Одна из техасских компаний, занимающихся сельским хозяйством, при помощи графологических тестов выявляет невезучих водителей.[2]

Существует много областей применения психологии почерка.[3]

1. Прием на работу. Служит для оценки и сравнения нескольких кандидатов на определенную должность. Они еще не работают в самой фирме, и фирма не имеет о них представления, кроме отзывов с предыдущего места работы. Должности, на которые претендуют кандидаты, могут быть совершенно разными: от менеджеров до продавцов мороженого, – и к любому виду работы предъявляются свои требования, а претендент должен им удовлетворять.
2. Аттестация кадров. Напоминает приемом на работу, но в отличии от неё оцениваются люди, которые уже работают в фирме. Требуется их характеристика для выдвижение на новые должности или оценка их адекватность уже занимаемым позициям. Обычно в таких случаях рассматриваются управляющие должности.
3. Профессиональное ориентирование. Охватывает работу с молодыми людьми или социальными случаями. Такое направление графологии может потребоваться для решения вопроса об оптимальной интеграции в новой стране. Для этого подбираются виды деятельности, которые наиболее подходят молодым людям или приезжим в другую страну. Заказчиками в таких случаях могут являться социальные службы, службы, занимающиеся трудоустройством и безработными, организации, отвечающие за интеграцию иностранцев.
4. Персональная консультация. Проводится в тех случаях, когда сам человек, а не фирма или организация, приходит к психологу, чтобы лучше разобраться в себе. Обычно это может быть связано с его работой, социальным положением, личными отношениями.
5. Проверка третьих лиц. Проходит по просьбе заинтересованных в этом людей, которые хотят больше узнать о своем партнере по бизнесу, о няне ребенка, о друге или родственнике.
6. Брачные консультации. Интересуют в основном различные бюро по знакомству. Иногда и сами молодожены или люди еще только вступающие в брак хотят узнать, насколько они подходят друг другу.
7. Исторические исследования. Занимаются личностями, как известными (цари, полководцы, деятели искусства), так и не очень. Об их характере или сложно найти какую-либо информацию, или имеющиеся воспоминания противоречивы.
8. Криминалистика. Здесь графологию иногда используют для создания психологического портрета преступника при его определении и поимке. Иногда графология наоборот помогает при вынесении оправдательного приговора и становятся дополнительным признаком невиновности подозреваемых.
9. В клинической психологии и медицине. Род-Айлендский психолог Марк Сеифер выделил признаки, по которым можно предположить наличие у человека шизофрении и эпилепсии. Нью-Йоркский графолог Патрисия Сигель своими исследованиями подтвердила результаты Марка Сеифера в отношении эпилепсии.

## 1.2. Обзор программного обеспечения

Анализ почерка – это трудоемкий и сложный процесс. Может потребоваться несколько часов или даже дней, чтобы проанализировать страницу рукописного текста, в зависимости от детализации и полноты требуемого отчета. Для каждого почерка необходимо провести серию различных оценок, которые включают в себя такие факторы как размер, наклон, размашистость, стиль и нажим. Обычно несколько строчек текста недостаточно для полноценного анализа. Будучи заключенными во временные рамки, утомленные повторением анализа, сложностью процесса и необходимостью предоставления отчета, графологи осознали необходимость компьютеризации анализа почерка. Создание систем анализа почерка даже увеличивает спрос на этот вид психологического тестирование и на навыки графолога. Некоторые графологи сделали платными разработанные ими анализаторы почерка и выпустили их на рынок. Так же существует множество различных онлайн систем для анализа почерка, разработанных профессиональными графологами.

Ниже будут описаны шесть систем анализа почерка, такие как Jerral Sapienza’s Self-Analysis, Andy Hunt’s Graphonomizer, Handwriting University’s Handwriting Wizard, Sheila Lowe’s Handwriting Analyzer и Garth Michaels’ Handwriting Analyst, HSDetect.

**Jerral Sapienza’s Self-Analysis**

Система была спроектирована, написана и поддерживается Джерралом Сапиензой, графологом и учителем информатики с более чем 30-летним опытом анализа почерка (2004 г.).

Работа системы, заключается в том, что она задает пользователю вопросы о различных аспектах его почерка. Пользователю предлагается выбрать один ответ из списка возможных. На основании этих ответов система составляет отчет.

Системное руководство помогает использовать систему более эффективно. Пользователь может просмотреть готовые примеры и подготовить свой образец почерка. Руководство также бывает полезным, если пользователь не уверен насчет некоторых графологических терминов. Пользователю предлагается оценить пользовательский интерфейс и правильность составленного системой отчета. [4][7]



Рис. 1. Jerral Sapienza’s Self-Analysis

**Andy Hunt’s Graphonomizer**

Графономайзер был создан Энди Хантом в 1992 году и адаптирован для работы с ним в интернете в 1996 году. Он разрабатывался с использованием хорошо обоснованных научных процедур, объективных доказательств, полученных из статистического анализа, и изучений работы многих научных исследователей Америки и Европы.

Перед тем, как пользователь начнет анализ, ему предлагается подготовить образец своего почерка в соответствии с приведенной детальной инструкцией. Затем пользователю предстоит пройти через ряд вопросов. Когда ответы будут получены, результаты будут автоматически отправлены компьютерной программе, которая использует баллы для составления графика и создания персонального профиля. График покажет баллы для восьми черт характера объекта, таких как независимость, уверенность, подчинение, перфекционизм, амбициозность, агрессия, экстраверсия и суетность, оцененные от 0 до 10.

По завершении пользователю необходимо кликнуть по кнопке “Submit” и отчет появиться на экране компьютера. Отчет – это полноценный анализ, который состоит из порядка 500-1200 слов и зависит от графиков с баллами по восьми главным чертам характера. Оценка пользователями полноты отчета, предоставляемого этой системой, оценивается в 76,2%.[4][9]

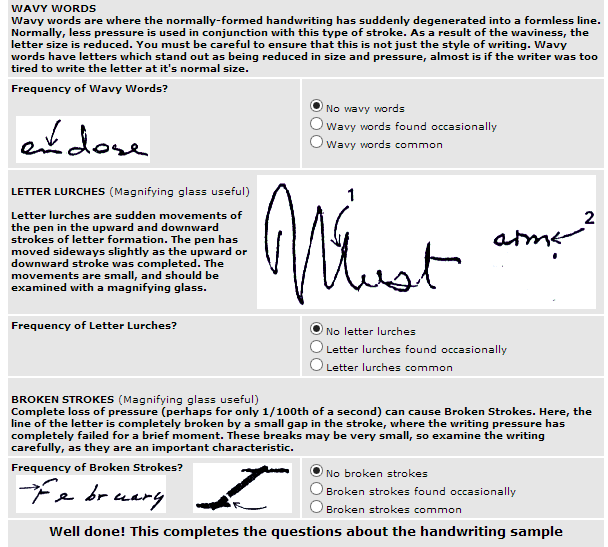


Рис. 2. Andy Hunt’s Graphonomizer

**Handwriting University’s Handwriting Wizard**

Мастер Почерка – это бесплатная онлайн система самостоятельного тестирования почерка, которая принадлежит и спонсируется Handwriting University (2004 г.). Пользователю предлагается подготовить рукописный образец почерка и нелинованную бумагу. Анализ начинается с ответов на серию вопросов и сравнения подготовленных образцов с показываемыми примерами. Отчет будет отправлен пользователю посредствам e-mail.

Мастер Почерка состоит из девяти графологических характеристик. Каждая характеристика представляет определенный элемент почерка и его уникальность. Все они приведены в таблице 1.

Таблица 1. Девять характеристик Мастера почерка.

|  |  |
| --- | --- |
| **Характеристика** | **Элемент почерка** |
| 1 | Наклон |
| 2 | Размер |
| 3 | Ножка прописных букв “d” и “t” |
| 4 | Прописная “o” |
| 5 | Случайные штрихи |
| 6 | Округлости букв “M” и “N” |
| 7 | Насколько высока черта буквы “t”, пересекающая ножку? |
| 8 | Фома петель и хвостов букв “y” и “g” |
| 9 | Границы и отступы |

Каждая характеристика состоит из одного или нескольких вопросов. В помощь пользователю для каждого из них приведены образцы. Есть вопросы, в которых можно выбрать более одного варианта ответа, и такие, которые могут быть пропущены. Для более полного отчета требуется, чтобы пользователь следовал инструкции и ответил на как можно большее число вопросов. [4][6]



Рис. 3. Handwriting University’s Handwriting Wizard

**Sheila Lowe’s Handwriting Analyser**

Анализатор почерка Шейлы Лоу применяет германскую теорию анализа почерка, называемую гештальт или единой графологической концепцией. В ней присутствуют 65 важных и поддающихся определению черт характера. Для каждой черты есть список характеристик почерка для сравнения. Программное обеспечение может описать 65 личных черт характера и использует для этого около 5000 различных подписей.

Для формирования отчета должны быть заполнены, по крайней мере, 10 категорий. Пользователь может не отвечать на все вопросы, а выбрать столько, сколько захочет. Однако чем больше он выбрал, тем полнее будет отчет.

Анализатор почерка Шейлы Лоу состоит из описания 14 должностей, хотя в реальной жизни компания имеет свои собственные потребности. «Мастер должностей» позволяет пользователю создавать свои собственные варианты должности, основанные на описаниях работы, что позволяет более точно сравнить кандидата с требованиями должности. Для того чтобы создать новый вариант должности, «Мастер должностей» предоставляет пользователю список из порядка 40 различных утверждений о работе. Пользователь оценивает каждое утверждение по пятибалльной шкале по тому, насколько важно это утверждение для этой работы. Компьютер использует полученные баллы для того, чтобы создать новый вариант должности.

После того, как вариант должности был выбран, компьютер оценивает баллы, полученные в результате проведения анализа, с баллами, необходимыми для данной работы. Результирующий граф показывает, насколько хорошо кандидат подходит на претендуемую должность. [4][10]



Рис. 4. Sheila Lowe’s Handwriting Analyser

**Garth Michaels’ Handwriting Analyst**

Графологический аналитик – это программа для анализа почерка, разработанная Гартом Мичелом, Дороти Ходос и Мэрилин Мейз (2004 г.). Последняя версия программы - 5.0. Для проведения анализа пользователю требуется предоставить свое имя, пишущую руку и пол. Графологический аналитик состоит из 60 вопросов. Каждый вопрос предоставляет пользователю возможность выбора нескольких вариантов ответа. Каждый вариант ответа иллюстрируется экранным образцом. Оцениваются такие характеристики почерка, как наклон основной линии, ее постоянство, наклон почерка, величина отступов между словами, размах, связность и разорванность прописных букв, связь заглавных букв, подчеркивания, давление и т. д.

Присутствуют два типа вопросов: «зависимые элементы» и «независимые элементы». В вопросах, касающихся зависимых элементов, можно выбрать только один вариант ответа. В вопросах же, касающихся независимых элементов, предоставляется возможность выбора нескольких вариантов.

Также предоставляются два типа отчетов: быстрые и полные. Быстрые отчеты содержат много информации, которую можно найти в полных отчетах. Быстрые отчеты предоставляют краткое описание личности пользователя. Разница заключается в том, что быстрые отчеты содержат короткие, сжатые предложения, в то время как полные отчеты состоят из детализированных описаний каждой характеристики.[4][8]



Рис. 5. Garth Michaels’ Handwriting Analyst

**HSDetect**

HSDetect создана российским графологом Ю. Г. Черновым и представляет собой автономное программное обеспечение (2008 г.). Это комплекс баз данных, программ и моделей. Система предназначена для решения многих задач, связанных с исследование почерка, включая графологический анализ, сравнение почерков, статистические исследование, обучение и т. д. Она включает обширный банк, который построен на данных из множества источников, книг, работ, выполненных специалистами из разных стран и графологических школ.

Для широкого круга пользователей разработана упрощенная версия программы HSDetect – HSD-BHV. Технически HSD-BHV была переведена в MS Excel.

Примерный алгоритм анализа следующий. Пользователь вводит свое имя и, оценивая свой почерк, по каждому признаку выбирает из раскрывающегося списка его значение и уровень. Уровень может быть «Высокий», «Значительный» и «Достаточный». Если пользователь не может определить признак почерка или сомневается, то он вправе оставить это поле пустым. После нажатия кнопки «Оценить черты личности» стартует режим оценки. Результат расчета записывается на странице «Y». Для каждой черты личности, вошедшей в список, возможны два уровня ее наличия у пользователя: «Явно» и «Возможно». Не вошедшие черты личности оценены для автора почерка как незначительные.[5]



Рис. 6. HSDetected

**Резюме по шести автоматизированным системам анализа почерка.**

Нет лучшей системы анализа почерка. Все зависит от типа ее использования и области применения. Правильная система та, которая отвечает большему количеству требований. Каждая из описанных выше компьютерных систем уникальна. Чем больше пользователей понимает графологию, тем более полным и всесторонним будет анализ.[4]

Таблица 2. Резюме по шести автоматизированным системам анализа почерка.

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **Название системы** | **Описание** | **Область применения** |
| 1 | Jerral Sapienza’s  Self-Analysis | Компьютерная система анализа почерка, основанная на интернет-технологиях. Система задает пользователю вопросы о различных аспектах его почерка. Не зависит от платформы. Система является бесплатной. Текстовый отчет. | Персональная оценка |
| 2 | Andy Hunt’s  Graphonomizer | Компьютерная система анализа почерка, основанная на интернет-технологиях. Пользователь проходит через ряд вопросов. Когда ответы получены, результаты автоматически отправляются компьютерной программе, которая использует баллы для составления графика и создания персонального профиля. Не зависит от платформы. Система является бесплатной. Текстовый отчет, график. | Персональная оценка |
| 3 | Handwriting  University’s  Handwriting  Wizare | Компьютерная система анализа почерка, основанная на интернет-технологиях. Система состоит из 9 характеристик почерка. По каждой характеристике задается один или несколько вопросов. Не зависит от платформы. Система является бесплатной. Текстовый отчет. | Персональная оценка |
| 4 | Sheila Lowe’s  Handwriting  Analyzer | Применяется гештальт или единая графологическая концепция. Характеристики почерка, такие как отступ между линиями, границы, наклон, давление разбиты на категории. Для составления отчета необходимо заполнить, по крайней мере, 10 категорий. Автономное приложение. Операционная система: Windows. Текстовый отчет, графы, графики. | Персональная оценка и трудоустройство. |
| 5 | Garth Michaels’  Handwriting  Analyst | Система состоит из 60 вопросов с множественным выбором вариантов ответов. Каждый вопрос соответствует одной характеристике почерка. Автономное приложение. Операционная система: Windows. Текстовый отчет. | Персональная оценка |
| 6 | HSDetect | Пользователь оценивает уровень характеристик своего почерка. Две версии программы: профессиональная (HSDetect) и для широкого круга пользователей (HSD-BHV). Автономное приложение. Операционная система: Windows. В качестве отчета оценивается уровень характеристик, присущих пользователю. | Персональная оценка |

## 1.3. Постановка задачи

Общей деталью всех перечисленных выше программ является то, что ни одна из них не работает с оригиналом почерка. Системы получают информацию о почерке пробанда с помощью тестирования: опрос пользователя о различных его аспектах. Здесь могут возникать две проблемы: субъективная оценка пользователя и его неподготовленность в вопросе анализа почерка. Пробанд не всегда может точно оценить уровень того или иного признака. Если определить наклон, типы которого довольно четко разграничены, достаточно легко, то с такой характеристикой, как округлость, для определения которой необходимо подсчитать количество дугообразных и угловатых элементов и вычислить их долевое соотношение, могут возникнуть трудности. Даже графологи порой могут не сойтись во мнении относительного оценки какой-нибудь характеристики почерка.

Так же следует отметить то, что в обзоре приведены программы, созданные в период с 1992 по 2008 гг. И так как они используют общий принцип получения необходимой информации о почерке пробанда, то можно говорить о том, что в области разработки компьютерных систем анализа почерка нет прогресса. Кроме тестирования характеристик почерка пока нет других решений этой задачи.

В данный момент на рынке нет программного обеспечения, способного работать с оригиналом подписи, не смотря на то, что технический прогресс предоставляет такую возможность. К примеру, есть возможность сканирования экземпляров почерка и последующая их обработка и приведение к виду, удобному для анализа. Еще один вариант – это использование устройств с сенсорным вводом. Такой подход предоставляет возможность получения дополнительных сведений о почерке, которые нельзя получить при сканировании, например, скорость ввода, нажим, направление движения пера.

Таким образом, тема разработки системы анализа почерка является актуальной.

Одной из основных особенностей будущей системы является то, что она будет работать с оригиналом почерка. Но из этого вытекает следующая проблема. Для полноценного анализа почерка необходимо не менее одного листа рукописного текста. Такой объем сложно представить к виду, необходимому для анализа. В связи с этим необходимо найти какой-либо объект, относящийся к почерку, содержащий как можно больше информации о пробанде и более подходящий для анализа, чем несколько листов текса.

**В современном мире всё меньше и меньше рукописного материала.** Повсеместное применение компьютерной техники при формировании документов привело к значительному повышению роли подписи как единственного удостоверительного знака**. Поэтому она, зачастую, становится единственным информативным материалом. Профессиональные графологи утверждают, что по одной подписи нельзя сделать полного анализа личности. И они правы. Тем не менее, подпись несет хоть и частичную, но вполне адекватную информацию о человеке.**

**Как было сказано выше, можно воспользоваться двумя способами ввода подписи: сканирование и использование сенсорных устройств. Можно выделить несколько причин, по которым использование сенсорных устройств выгоднее, чем сканирование:**

1. **Сканирование существенно повышает сложность задачи. Появляется необходимость использования дополнительных алгоритмов, таких как: устранение помех, лишних деталей и прочих недостатков отсканированного образца, векторизация подписи и т. д.**
2. **При использовании сенсорных устройств можно получить дополнительную информацию о характеристиках подписи, таких как: направление движения пера, скорость ввода, контрольные точки подписи и т. п.**
3. Хотя на первый взгляд точность информации при вводе подписи с помощью сенсорного экрана снижается, но психологические характеристики человека не зависят от его умения писать. По этому поводу высказался профессор сравнительной патологии Кильского университета Г. Шнейдемиль. Он указывает, что процессы высшей нервной деятельности человека проявляются внешне. Это происходит путем известных волевых актов, конкретизирующихся через движения. «Психические процессы мы не можем наблюдать непосредственно и познаем их только через органические движения, – говорит он далее, – если же выражения желания надлежит рассматривать как результат рефлекторного последствия постоянно разыгрывающихся процессов мышления или чувствований, то и через них возможно судить о характерных особенностях человека. Следовательно, движения при ходьбе, выражение лица при разговоре и, наконец, упражнения в письме могут быть использованы для изучения внутренних процессов организма». Таким образом, практически любые движения человека можно рассматривать как своеобразный сейсмограф психической деятельности человека. Но в отличии от других движений человека, почерк оставляет после себя конкретный и уникальный для каждого след, который хорошо пригоден для исследования, независимо от того, пишет ли человек рукой или применяет другой орган тела, удобно ли ему использовать сенсорный кран или нет.[11]

Таким образом, целью данной работы является разработка системы автоматизированного анализа подписи.

## 1.4. Разработка технического задания

1. Назначение и цели создания (развития) системы.

Назначением разрабатываемой системы является автоматизация процесса графологического анализа подписи.

Целью создания данной системы является снижение трудоемкости графологического анализа подписи.

2. Характеристика объекта автоматизации

Объектом автоматизации является графологический анализ подписи. Он представляет собой определение психологических качеств личности на основе характерных особенностей подписи. Характерные особенности подписи выделены в список характеристик подписи, подлежащих оценке (например, длина подписи, скорость написания, амплитуда, ровность, наклон и т. д.).

3. Требования к системе:

3.1. Требования к системе в целом.

Система должна состоять из следующих модулей:

- модуль ввода подписи;

- модуль обработки характеристик подписи;

- модуль оценивания результатов обработки характеристик подписи;

3.2. Требования к функциям (задачам), выполняемым системой.

1. Модуль ввода подписи:

- разрабатываемая система должна уметь работать с оригиналом подписи;

- должно быть обеспечено получение дополнительной информации о подписи, такой, как: направление движения траектории, скорость, нажим и т. п.;

2. Модуль обработки характеристик подписи:

- каждая характеристика подписи должна обрабатываться независимо от остальных характеристик;

3. Модуль оценивания результатов обработки характеристик подписи:

- каждой характеристике подписи должно соответствовать одно единственное значение характеристики личности;

3.3. Требования к видам обеспечения.

1. Математическое обеспечение.

Разрабатываемое программное обеспечение должно поддерживать методы дифференциальной геометрии, численного дифференцирования, интерполяции и аппроксимации заданного набора значений.

2. Информационное обеспечение.

- разрабатываемая система должна уметь работать с оригиналом подписи;

- должно быть обеспечено получение дополнительной информации о подписи, такой, как: направление движения траектории, скорость, нажим и т. п.;

- каждой характеристике подписи должно соответствовать одно единственное значение характеристики личности;

3. Лингвистическое обеспечение.

- для организации диалога с пользователем будет использован режим меню и ввод подписи с помощью рисования;

- для реализации системы был выбран язык высокого уровня Java;

4. Программное обеспечение.

Для реализации системы будут использованы следующие программные средства:

- IDE Eclipse Classic 4.2.2;

- JDK 6;

- Android Development Tools;

- Android SDK;

5. Техническое обеспечение.

Основной особенностью разрабатываемого программного обеспечения является работа с оригиналом подписи, поэтому используемое оборудование должно поддерживать сенсорный ввод.

4. Состав и содержание работ по созданию системы

|  |  |  |  |
| --- | --- | --- | --- |
| **№ этапа** | **Содержание этапа работы** | **Сроки выполнения этапа** | **Результаты, полученные на данном этапе** |
| 1 | Ознакомление с методами анализа подписи | 29.01.13 – 10.02.13 | Отчет |
| 2 | Составления перечня характеристик, пригодных для компьютерной реализации | 11.02.13 – 24.02.13 | Отчет |
| 3 | Разработка алгоритмов для анализа характеристик подписи | 25.02.13 – 10.03.13 | Отчет |
| 4 | Разработка модуля ввода | 11.03.13 – 24.03.13 | Структура программного обеспечения |
| 5 | Разработка модуля обработки характеристик подписи | 25.03.13 – 04.07.13 | Структура программного обеспечения |
| 6 | Разработка модуля оценки результатов обработки характеристик подписи | 08.04.13 – 21.04.13 | Структура программного обеспечения |
| 7 | Реализация системы | 22.04.13 – 05.05.13 | Исходные коды |
| 8 | Тестирование и отладка | 06.05.13 – 19.05.13 | Исходные коды |
| 9 | Сдача студентом окончательного варианта выпускной квалификационной работы научному руководителю | 20.05.13 | Полный отчет |

## 1.5. Метод анализа подписи

**Характеристики подписи**

Были рассмотрены два подхода к анализу подписи: Джеймса Раскина и [Ойбека Наджимов](http://lib.ru/URIKOVA/GRAFOLOG/aybek/aybek.txt)а. Оба подхода основаны на анализе подписи по перечню характеристик.

Ойбек Наджимов предлагает следующий перечень характеристик:[12]

1. Направление надписи;
2. Длина подписи;
3. Начало и конец подписи;
4. Величина букв;
5. Закругленность и острота букв;
6. Связность и разорванность букв;
7. Уверенность в написании букв;
8. Украшения подписи;
9. Размашистость в написании букв;
10. Расстояние между буквами;
11. Сила давления при написании букв;
12. Подчеркивания, хвостики, зачеркивания;
13. Вертикальные лини в подписи;
14. Ровность и неровность подписи;
15. Различные петли в подписи;
16. Точка в подписи;
17. Нагруженность в подписи;
18. Одинаковые элементы в подписи;
19. Ясность;
20. Иностранные буквы в подписи;
21. Графологичность в подписи;
22. Простота в подписи;
23. Скорость написания;
24. Наклон букв в подписи;
25. Степень гармоничность подписи;
26. Различные виды подписи у одного и того же человека;
27. Изменение подписи с течением времени;
28. Возвраты в подписи;
29. Аномальные признаки в подписи;
30. Волнистость в подписи;
31. Геометрическая выдержанность в подписи;
32. Удлиненное начало в подписи;
33. Амплитуда подписи;
34. Комбинаторика в подписи;
35. Признаки симметрии;
36. Цифровые добавления в подписи;

Перечень признаков по Джеймсу Раскину имеет вид: [13]

1. Размер подписи;
2. Длина подписи;
3. Тип подписи;
4. Расстояние между буквами;
5. Присутствие в подписи различных элементов;
6. Наклон подписи;
7. Направление заключительного штриха;
8. Длина «хвоста» подписи;
9. Подчеркивание подписи;
10. Признаки симметрии;
11. Сложность и простота;
12. Разборчивость;
13. Нажим;

Некоторые характеристики, такие как: уверенность в написании букв, нагруженность, графологичность, степень гармоничности, различные виды подписи у одного и того же человека, изменение подписи с течением времени, аномальные признаки, геометрическая выдержанность, комбинаторика - представляют большую сложность для реализации и автоматической оценки, и поэтому были исключены из рассмотрения. Из двух перечней были выбранных характеристики, наиболее подходящие для программной реализации.

1. Направление букв;
2. Длина подписи;
3. Количество заглавных букв в начале;
4. Количество заглавных букв в конце;
5. Величина первой буквы;
6. Закругленность и острота;
7. Связность и разорванность;
8. Размашистость подписи;
9. Расстояние между буквами;
10. Наличие подчеркиваний;
11. Наличие вертикальных линий;
12. Ровность и неровность;
13. Наличие петель;
14. Наличие точек;
15. Нагруженность и простота в подписи;
16. Ясность;
17. Наличие иностранных букв;
18. Скорость написания;
19. Наклон букв;
20. Наличие возвратов;
21. Волнистость;
22. Амплитуда;
23. Наличие цифровых добавлений;

Подпись будет представлена отдельными траекториями (под траекторией будем понимать жесты, введенные пользователем с момента касания пером экрана до момента отрыва пера от экрана), каждая из которых состоит из точек, имеющих координаты (x,y).

**Скелетизация**

Под скелетизацией будем понимать разбиение рукописного текста на штрихи. Так как подпись состоит из точек, имеющих координаты (x,y), то ее можно рассматривать в качестве функции. Тогда разбиение будет происходит в точках локальных максимумов и минимумов этой функции. Штрихи будут иметь вертикальную направленность. При необходимости получения горизонтальных штрихов можно поменять ролями ось абсцисс и ординат.



Рис 1. Пример работы алгоритма скелетизации рукописного текста

**Нахождение средней линии и определение угла её наклона.**

В некоторых алгоритмах требуется нахождение средней линии набора точек. Для этого применяется аппроксимация линейной зависимостью y=ax+b с помощью метода наименьших квадратов. Прямая однозначно определяется парой параметров (a,b), где a – тангенс угла наклона прямой, b – расстояние до прямой по оси ординат.

Функция двух переменных *а* и *b F(a,b)=* должна принимать наименьшее значение.

Составляется и решается система из двух уравнений с двумя неизвестными. Находим частные производные функции *F(a,b)=* по переменным a и b, приравниваем эти производные к нулю.

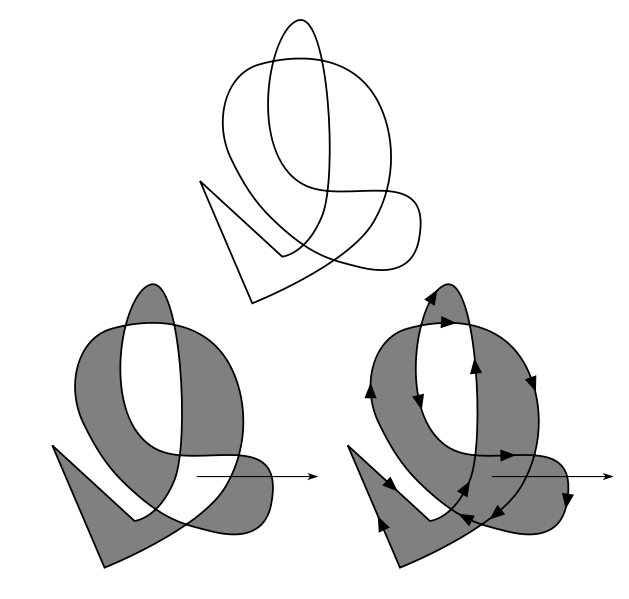
После нахождения производных по переменным a и b и решения системы методом подстановки формулы для нахождения коэффициентов выглядят следующим образом[15]:

**Задача о принадлежности точки многоугольнику**

Также в некоторых алгоритмах требуется определить, принадлежит ли точка произвольному простому многоугольнику.

Для решения этой задачи применяется следующий метод. Из точки выпускается луч в произвольном направлении (например, в направлении начала координат). Далее подсчитывается количество пересечений ребер многоугольника. Для этого запускается цикл по всем ребрам многоугольника, каждое из которых представляет собой прямую, ограниченную двумя точками. Для каждой прямой определяется, пересекает ли она прямую, заключенную между точкой начала координат и точкой, для которой выполняется алгоритм. Если число пересечений нечётно, то точка лежит внутри многоугольника. В противном случае — то снаружи. Дело в том, что с каждым пересечением границы точка оказывается то внутри, то снаружи многоугольника. Данный алгоритм носит название *crossing number (count) algorithm* или *even-odd rule*.

Проблема возникает, когда луч пересекает вершину многоугольника. Решение заключается в том, чтобы считать, что такие вершины лежат на бесконечно малую величину выше или ниже луча. В таком случае пересечения нет. Пересечение выполняется только в том случае, если один из концов ребра лежит ниже луча, а другой конец — выше или лежит на луче.[16]

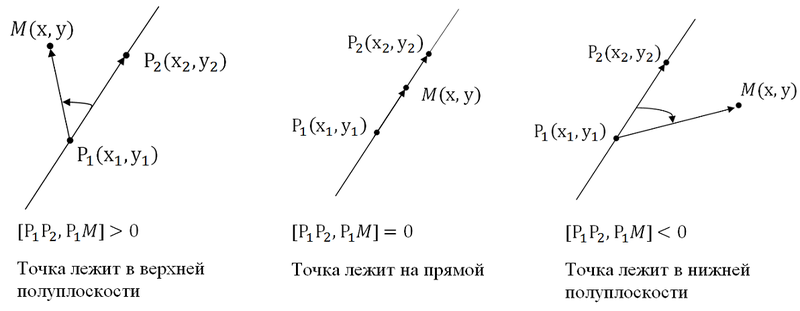


**Определение пересечения двух отрезков**

В некоторых алгоритмах встречается задача определения пересечения двух отрезков, концы которых заданы координатами. Отрезки пересекаются тогда, когда, концы каждого отрезка лежат по разные стороны от другого отрезка.

Под косым произведением векторов p1 и p2 с декартовыми координатами (x1, y1) и (x2, y2) можно понимать ориентированную площадь параллелограмма, образованного точками (0,0), (x1, y1), (x2, y2), (x1 + x2, y1 + y2), которая равна p1´p2 = –p2´p1= x1y2 – x2y1.

Косое произведение двух векторов положительно, если поворот от первого вектора ко второму идет против часовой стрелки, равно нулю, если векторы коллинеарные, и отрицательно, если поворот идет по часовой стрелки.



Для определения пересечения двух отрезков необходимо проверить расположение ограничивающих точек первого отрезка относительно второго отрезка и наоборот. Если произведение косых произведений в обоих случаях будет меньше нуля, то отрезки пересекаются.



Пусть даны два отрезка A(ax1, ay1, ax2, ay2) и B(bx1, by1, bx2, by2).

v1=(bx2-bx1)\*(ay1-by1)-(by2-by1)\*(ax1-bx1);

v2=(bx2-bx1)\*(ay2-by1)-(by2-by1)\*(ax2-bx1);

v3=(ax2-ax1)\*(by1-ay1)-(ay2-ay1)\*(bx1-ax1);

v4=(ax2-ax1)\*(by2-ay1)-(ay2-ay1)\*(bx2-ax1);

Отрезки пересекаются в том случае, если выполняется условие: (v1\*v2<0) и (v3\*v4<0).[17]

**Нахождение точки пересечения двух отрезков**

Если два отрезка заданы точками (а1х; а1y), (a2x; a2y), (b1x; b1y), (b2x; b2y), то задача нахождения точки пересечения двух отрезков сводится к решению системы уравнений:

;

;

\*;

((

(\*(

1. **Направление букв**

Пользователь выделяет каждую букву подписи. В результате получается некоторый набор точек, ограниченный областью выделения. Определяется нижняя точка каждой буквы. Для этого перебором находится точка, имеющая наименьшую координату по оси абсцисс. Получается набор точек. Составляется функция f(t), где t – номер буквы, к которой относится точка. Функция аппроксимируется линейной зависимостью y=ax+b с помощью метода наименьших квадратов.

Необходимо определить, возрастает или убывает функции. Для этого достаточно определить угол наклона прямой относительно оси координат. Если угол между прямой и осью координат лежит в пределах от 10° до -10°, то подпись можно считать ровной. Если угол больше заданного предела, то подпись можно считать возрастающей. В противном случае подпись убывает.

1. **Длина подписи**

Подпись считается короткой, если в ней не больше трех букв. В противном случае подпись считается длинной

1. **Количество заглавных букв в начале**

Проверка этой характеристики происходит в том случае, если количество букв подписи больше одной. Количество букв делится пополам. Для каждой из букв, входящих в первую половину, вычисляется среднее арифметическое амплитуд составляющих их штрихов. Вычисляется среднее арифметическое среднего арифметического амплитуд букв, входящих в первую половину A1. Вычисляется среднее арифметическое среднего арифметического амплитуд штрихов остальных букв A2. Вычисляется отношение A2/А1=B. Если B > B1 (B1 – величина, полученная экспериментально), то считается, что величина заглавных букв значительно отличается от остальных.

1. **Количество заглавных букв в конце**

Проверка этой характеристики происходит в том случае, если количество букв подписи больше одной. Количество букв делится пополам. Для каждой из букв, входящих в вторую половину, вычисляется среднее арифметическое амплитуд составляющих их штрихов. Вычисляется среднее арифметическое среднего арифметического амплитуд букв, входящих в вторую половину A2. Вычисляется среднее арифметическое среднего арифметического амплитуд штрихов остальных букв A1. Вычисляется отношение A1/А2=B. Если B > B1 (B1 – величина, полученная экспериментально), то считается, что величина букв, входящих во вторую половину, значительно отличается от остальных.

1. **Величина первой буквы**

Отношение заглавной буквы к остальным вычисляется по формуле AS/А1=B, где A1 – среднее арифметическое амплитуд штрихов заглавной буквы, АS – среднее арифметическое средних арифметических амплитуд штрихов остальных букв. Если B > B1 (B1 – величина, полученная экспериментально), то считается, что величина заглавной буквы значительно отличается от остальных. В противном случае отличие незначительное.

1. **Закругленность и острота**

Для определения округлости или остроты подписи необходимо вычислить долевое соотношение округлых и угловатых элементов.

Подпись скелетизируется на штрихи в вертикальном и горизонтальном (т. е. от крайней левой до крайней правой точки и наоборот) направлении.

Для определения количества округлых и угловатых элементов рассматриваются отдельные штрихи и углы между ними.

Угол между двумя штрихами считается острым, если он меньше 90°. В противном случае угол считается тупым.

Если округлых элементов более 70% процентов от общего числа, то подпись считается округлой. Если же угловатых элементов больше 70% процентов от общего числа, то подпись считается острой. В противном случае отношение закругленности и остроты в подписи считается смешанным.[18]

1. **Связность и разорванность**

Для того, чтобы определить являются ли буквы связными, нужно найти последнюю точку в первой букве и проследить траекторию движения пера до вхождения ее в следующую букву. Пользователь выделяет каждую букву подписи. В результате получается некоторый набор точек, ограниченный областью выделения. Так как точки последовательно связаны друг с другом в том порядке, в каком их вводил пользователь, то можно проследить траекторию движения и определить, какая точка является последней до выхода траектории из области выделения.

Если точки соединены последовательно и отсутствуют разрывы, то буквы считаются связанными. В противном случае буквы считаются разорванными.

Другими словами, если есть хотя бы одна траектория, принадлежащая обеим буквам, то они считаются связанными. В противном случае буквы считаются разорванными.

1. **Размашистость**

Под размашистостью будем понимать отношение площади подписи к площади листа (S/SSIGN=B). Если B>B1 (B1 – величина, полученная экспериментально), то подпись считается размашистой. Ssign вычисляется как asign\*bsign (asign = (xr – xl), bsign = (xb – xt), где xr - крайняя правая координата подписи, xl – крайняя левая координата подписи, xt – крайняя верхняя координата подписи, xb – крайняя нижняя координата подписи). Все необходимые координаты находятся перебором.

1. **Расстояние между буквами**

Каждая буква является набором точек. Первым действием нужно убедиться, что ни одна из точек одной буквы не попадает в область другой буквы, в противном случае расстояние считается нулевым.

*L1={x1, x2,…, xn}*

*L2={x1, x2,…, xm}*

*,i=1..n*

Затем находится две ближайшие точки, каждая из которых принадлежит противоположным буквам . Вычисляется расстояние между ними *l=|x1-x2|*. Вычислив расстояние между всеми буквами, вычисляется среднее арифметическое расстояние . В качестве критерия расстояния выбирается ширина самой узкой буквы *b*. Пользователь выделяет каждую букву подписи. В результате получается некоторый набор точек, ограниченный областью выделения. Перебором находятся крайняя левая точка буквы *xl* и крайняя правая точка буквы *xr*. В результате ширина вычисляется как *b=*|*xr –xl*|. Выбирается минимальная ширина. Если среднее расстояние не превышает ее (), то расстояние между буквами считается маленьким. В противном случае расстояние считается большим.

1. **Наличие подчеркиваний**

Рассматривается отдельная траектория. Траектория скелетизируется на штрихи в горизонтальном направлении, т. е. от крайней левой до крайней правой точки и наоборот. Каждый штрих будет представлен набором точек. С помощью метода наименьших квадратов будет производиться аппроксимация набора точек прямой линейной зависимостью y=ax+b.

Перебором находится максимальное отклонение штриха от прямой:

*i=1…n*

где n – кол-во точек в штрихе.

Также находим длину штриха.

Штрих является прямой, если максимальное отклонение *dmax* не превышает предельного значения (23 пикселя) и угол наклона аппроксимирующей прямой лежит в пределах от 80° до 100° и длина штриха больше 75% от высоты всей подписи.

1. **Наличие вертикальных линий**

Рассматривается отдельная траектория. Траектория скелетизируется на штрихи. Каждый штрих будет представлен набором точек. С помощью метода наименьших квадратов будет производиться аппроксимация набора точек прямой линейной зависимостью y=ax+b.

Перебором находится максимальное отклонение штриха от прямой:

*i=1…n*

где n – кол-во точек в штрихе.

Также находим длину штриха.

Штрих является прямой, если максимальное отклонение *dmax* не превышает предельного значения (23 пикселя) и угол наклона аппроксимирующей прямой лежит в пределах от 35° до -35° и длина штриха больше 75% от высоты всей подписи.

1. **Ровность и неровность**

Пользователь выделяет каждую букву подписи. В результате получается некоторый набор точек, ограниченный областью выделения. Для каждой буквы находится ее нижняя точка. Для этого перебором находится точка, имеющая наименьшую координату по оси абсцисс. Полученный набор точек аппроксимируем прямой линией с помощью метода наименьших квадратов зависимостью y=ax+b и находим тангенс угла наклона a и координату пересечения с осью ординат b. Для каждой точки находим отклонение от аппроксимирующей прямой:

*i=1…n*

где n – кол-во букв.

Если максимальное отклонение *dmax* больше предельного значения (23 пикселя), то это считается отклонением. Если количество отклонений меньше половины количества букв, то подпись считается ровной. В противном случае нет.

1. **Наличие петель**

Каждая траектория подписи состоит из точек, соединенных прямыми линиями. Для нахождения петли необходимо проверить пересечение этих линий друг с другом. Наличие такого пересечения означает, что траектория пересекает саму себя и в этом месте присутствует петля.

Далее необходимо вычислить площадь этой петли. Петля разбивается на верхнюю и нижнюю части. Для каждой части вычисляется площадь под ней, из большего вычитается меньшее и полученный результат будет является площадью петли.

S =

Если площадь петли больше 20% от площади подписи, то петля считается большой. В противно случае – маленькой.

В случае если количество больших петель в подписи больше, чем количество маленьких, то петли в подписи считаются большими. В противном случае маленькими.

1. **Наличие точек**

Траектория считается точкой, если количество составляющих ее контрольных точек из общего числа контрольных точек подписи составляет менее одно процента (это число должно быть получено экспериментально).

1. **Нагруженность и простота**

Для определения нагруженности подписи необходимо произвести подсчет траекторий, не вошедших ни в одну букву. Если их количество превышает число X (X – величина, полученная экспериментально), то подпись считается нагруженной. В противном случае подпись считается простой.

1. **Скорость написания**

Скорость написания будет определяться с помощью средств используемого языка программирования. Будет запускаться таймер, который определит скорость написания в миллисекундах. Если время написания превышает число Х (Х – величина, полученная экспериментально), то скорость написания считается быстрой. В противном случае скорость написания считается медленной.

1. **Наклон букв**

Каждая буква представлена набором точек. С помощью метода наименьших квадратов будет производиться аппроксимация набора точек прямой линейной зависимостью y=ax+b.

Угол между осью координат и аппроксимирующей прямой и будет являться углом наклона. После вычисления углов наклона каждой буквы будет вычисляться среднее арифметическое угла наклона . Если лежит в пределах от 80° до 100°, то наклон подписи считается прямым. Если же значение меньше этого предела, то наклон подписи считается правым. В противном случае наклон считается левым.

1. **Наличие возвратов**

Если траектория, выходя из последней буквы, последовательно и без разрывов пересекает середину подписи, или после ввода последней буквы были введены еще траектории, точки которых находятся на противоположной половине подписи, то это означает, что был совершен возврат. Середина подписи вычисляется по формуле:

где xr – крайняя правая координата подписи по оси абсцисс, а xl – крайняя левая координата.

1. **Амплитуда**

В подписи амплитудой является длина штриха. Производится скелетизация подписи. Вычисляются длины всех штрихов. Составляется функция f(t), где t – номер штриха. Функция аппроксимируется линейной зависимостью y=ax+b с помощью метода наименьших квадратов.

Необходимо определить, возрастает или убывает функция. Для этого достаточно определить угол наклона прямой относительно оси координат. Если угол между прямой и осью координат лежит в пределах от 10° до -10°, то амплитуду можно считать ровной. Если угол больше заданного предела, то амплитуду можно считать возрастающей. В противном случае амплитуда убывает.

1. **Ясность**

Для определения ясности вначале пользователю предлагается ввести своё имя, и введённая строка запоминается. После того, как введена подпись, составляющие её буквы выделяются. Для определения ясности анализируется каждая буква отдельно.

Находятся крайние правая, верхняя, нижняя и левая координаты буквы, таким образом, определяются границы квадрата, в который заключена буква. Каждый пиксел квадрата проверяется на наличие чёрного цвета. Составляется матрица размером, соответствующим высоте и ширине квадрата, в которой единице равны те элементы, соответствующие пикселы которых имели чёрный цвет. Остальные элементы матрицы равны нулю.

Производится свёртка матрицы к размеру 16х16. Исходная матрица разбивается на 256 квадратов по 16 квадратов в высоту и 16 квадратов в ширину. Для каждого квадрата определяется коэффициент его заполнения единицами. Если он равен или превышает общий коэффициент заполнения единицами исходной матрицы, то соответствующему элементу новой матрицы присваивается единица. В противном случае – ноль. В результате образуется новая матрица размером 16х16.

Полученная матрица сравнивается с эталонами, находящимися в базе данных. Для каждого эталона находится расстояние до него. Каждая матрица представляется точкой в 256-мерном пространстве, каждая координата которой может принимать значение ноль или единица.

*,;*

Эталон, для которого вычисленное расстояние является наименьшим, будет считаться введенной буквой.

После идентификации буквы определяется её наличие в имени пользователя. Если больше половины букв подписи входят в имя пользователя, то подпись считается ясной. В противном случае нет.[19]

1. **Наличие иностранных букв**

Для анализа подписи требуется, чтобы пользователь ввел свое имя. Наличие иностранных букв определяется по наличию иностранных букв в веденном имени.

1. **Наличие цифры**

Для анализа подписи требуется, чтобы пользователь ввел свое имя. Наличие цифр определяется по наличию цифр в веденном имени.

1. **Волнистость**

Для определения волнистости выполняется процедуры поиска «волн» и серпантина.

Под волной понимается траектория или ее часть, абсцисса каждой следующей точки которой больше абсциссы предыдущей точки и в которой как минимум четыре раз подряд чередуются локальный максимум и минимум.

Под серпантином понимается траектория или ее часть, в которой друг за другом следуют несколько петель. Для определения серпантина рассматривается отдельная траектория. Находятся ее точки самопересечения. Они аппроксимируются прямой линией с помощью метода наименьших квадратов, и вычисляется ее угол наклона. Если он лежит в пределах от -35 до 35 градусов, то в данной траектории присутствует серпантин.

1. **Определение темперамента**

Определение темперамента выполняется с помощью пятнадцати бальной шкалы. Темперамент, набравший наибольшее кол-во баллов, является доминирующим. Баллы начисляются за счет наличия тех или иных характеристик в подписи пробанда.

**Холерик**

|  |  |
| --- | --- |
| Обидчивый | Подчеркивания |
| Неспокойный | Острота |
| Агрессивный | Острота |
| Возбудимый | Быстрая скорость |
| Переменчивый | Неровность |
| Импульсивный | Неровность, разорванность |
| Оптимистичный | Направление вверх |
| Активный | Направление вверх, отсутствие вертикальных линий |

Амплитуда уменьшается: 4

Подчеркивания: 1

Острота: 7

Быстрая скорость: 3

Неровность: 2

Разорванность: 1

Направление вверх: 2

Отсутствие вертикальных линий: 1

**Сангвиник**

|  |  |
| --- | --- |
| Общительный | Округлость |
| Открытый | Ясность |
| Разговорчивый | Волнистость, ясность |
| Доступный | Ясность |
| Беспечный | Направление вверх, округлость |
| Живой | Направление вверх |
| Беззаботный | Направление вверх, округлость |
| Лидирующий | Заглавные буквы в начале |

Амплитуда уменьшается: 4

Округлость: 8

Волнистость: 2

Ясность: 3

Направление вверх: 3

Заглавные буквы в начале: 1

**Флегматик**

|  |  |
| --- | --- |
| Пассивный | Округлость |
| Осторожный | Ровность |
| Рассудительный | Длинная подписи, ровность |
| Мирный | Округлость |
| Сдержанный | Ровность |
| Надежный | Амплитуда ровная |
| Ровный | Ровность |
| Спокойный | Округлость |

Амплитуда ровная: 5

Округлость: 3

Ровность: 9

Длинная подпись: 4

**Меланхолик**

|  |  |
| --- | --- |
| Угрюмый | Направление вниз |
| Тревожный | Неровность |
| Ригидный | Связность |
| Трезвый | Длинная подпись |
| Пессимистичный | Направление вниз |
| Замкнутый | Большие буквы |
| Необщительный | Большие буквы |
| Тихий | Округлость |

Амплитуда увеличивается: 4

Направление вниз: 7

Неровность: 1

Связность: 2

Длинная подпись: 3

Большие буквы: 3

Округлость: 1

1. **Определение подходящей профессии**

Определение подходящей профессии выполняется с помощью сорок бальной шкалы. Профессия, набравшая наибольшее кол-во баллов, является наиболее подходящей. Баллы начисляются за счет наличия тех или иных характеристик в подписи пробанда.[20]

**Человек - природа**

|  |  |
| --- | --- |
| Развитое воображение | Нагруженность |
| Наглядно-образное мышление | Разорванность, отсутствие вертикальных линий |
| Хорошая зрительная память | Левый наклон, длинная подпись |
| Наблюдательность | Левый наклон, длинная подпись |
| Способность предвидеть и оценивать изменчивые природные факторы | Острота |
| Терпение | Длина подписи, малая величина букв, компактность, ровная подпись, точки, возвраты, амплитуда возрастает или ровная |
| Настойчивость | Длина подписи, малая величина букв, амплитуда возрастает или ровная |
| Готовность работать вне коллектива | Заглавные буквы в конце, отсутствие подчеркиваний, точки, возвраты, волнистость, петли |
| Готовность работать в трудных погодных условиях | Заглавные буквы в конце, отсутствие подчеркиваний, точки, возвраты , волнистость, петли |

Длинная подпись: 4

Заглавные буквы в конце: 5

Малая величина букв: 2

Острота: 1

Разорванность: 1

Компактность: 3

Левый наклон: 3

Отсутствие подчеркиваний: 3

Отсутствие вертикальных линий: 1

Ровность: 1

Наличие петель: 2

Наличие точек: 5

Нагруженность: 1

Наличие возвратов: 4

Волнистость: 2

Амплитуда возрастает или ровная: 2

**Человек - Техника**

|  |  |
| --- | --- |
| Хорошая координация движений | Ровная подпись |
| Точное зрительное, слуховое, вибрационное и кинестетическое восприятие | Левый наклон, длинная подпись |
| Развитое техническое и творческое мышление и воображение | Нагруженность, разорванность, отсутствие вертикальных линий, иностранные буквы, цифровые добавления |
| Умение переключать и концентрировать внимание | Острота, скорость, возвраты |
| Наблюдательность | Левый наклон, длинная подпись |
| Склонность к практической деятельности | Заглавные буквы в конце, компактность |

Длинная подпись: 2

Заглавные буквы в конце: 7

Острота: 6

Разорванность: 1

Компактность: 7

Левый наклон: 6

Отсутствие вертикальных линий: 1

Ровность и неровность подписи: 6

Нагруженность: 1

Наличие иностранных букв: 1

Скорость написания: 1

Наличие возвратов: 1

Наличие цифровых добавлений: 1

**Человек – Знаковая система**

|  |  |
| --- | --- |
| Хорошая оперативная и механическая память | Точки |
| Способность к длительной концентрации внимания на отвлеченном материале | Длина подписи, малая величина букв, возвраты |
| Хорошее распределение и переключение внимания | Малая величина букв, скорость |
| Точность восприятия | Точки |
| Умение видеть то, что стоит за условными знаками | Острота, отсутствие вертикальных линий |
| Усидчивость терпение | Длина подписи, малая величина букв, ровная подписи, точки, возвраты, амплитуда ровная |
| Логическое мышление | Большое кол-во заглавных букв в начале, связность, отсутствие вертикальных линий, прямой наклон, амплитуда ровная, цифровые добавления, размашистость |

Длинная подпись: 2

Большое кол-во заглавных букв в начале: 5

Малая величина букв: 3

Острота: 1

Связность: 5

Компактность: 4

Отсутствие вертикальных линий: 6

Ровность: 1

Наличие точек: 3

Скорость написания: 1

Наклон букв в подписи: 1

Наличие возвратов: 2

Ровная амплитуда: 2

Наличие цифровых добавлений: 4

**Человек – Художественный образ**

|  |  |
| --- | --- |
| Художественные способности | Нагруженность, направление вверх |
| Развитое зрительное восприятие | Левый наклон, длинная подпись |
| Наблюдательность, зрительная память | Левый наклон, длинная подпись |
| Наглядно-образное мышление | Разорванность, отсутствие вертикальных линий |
| Творческое воображение | Нагруженность, иностранные буквы |
| Знание психологических законов эмоционального воздействия на людей | Размашистость, ясность, большое расстояние между буквами |
| Склонность к умственной деятельности | Большое кол-во заглавные буквы в начале подписи |

Направление вверх: 8

Длинная подпись: 2

Большое кол-во заглавные буквы в начале подписи: 6

Разорванность букв: 1

Размашистость в написании: 1

Большое расстояние между буквами: 7

Отсутствие вертикальных линий: 1

Нагруженность: 9

Ясность: 1

Наличие иностранных букв: 2

Левый наклон: 2

**Человек – Человек**

|  |  |
| --- | --- |
| Умение легко вступать в контакт с незнакомыми людьми | Ясность, простота, скорость, волнистость |
| Устойчивое хорошее самочувствие при работе с людьми | Округлость |
| Доброжелательность, отзывчивость | Направление ровное, отсутствие подчеркиваний |
| Выдержка | Округлость, ровная подписи, точки |
| Умение сдерживать эмоции | Малая величина букв, отсутствие подчеркиваний, волнистость |
| Способность анализировать поведение окружающих и своё собственное, понимать намерения и настроение других людей, способность разбираться во взаимоотношениях людей, уметь улаживать разногласия между ними, организовывать их взаимодействие | Малое кол-во заглавных букв в начале, размашистость |
| Способность мысленно ставить себя на место другого человека, умение слушать, учитывать мнение другого человека | Размашистость, отсутствие подчеркиваний, правый наклон |
| Способность владеть речью, жестами, мимикой | Связность, ровная подписи |
| Развитая речь, способность находить общий язык с разными людьми | ясность, скорость, правый наклон, волнистость |
| Умение убеждать людей | Малое кол-во заглавных букв в начале, острота, размашистость, ясность |
| Аккуратность, пунктуальность, собранность | Малая величина букв |
| Знание психологии людей | Размашистость |

Направление ровное: 1

Малое кол-во заглавных букв в начале: 6

Величина первой буквы: 2

Закругленность: 5

Связность: 1

Размашистость: 5

Отсутствие подчеркиваний: 3

Ровность: 2

Наличие точек: 1

Простота: 1

Ясность: 4

Скорость написания: 2

Правый наклон: 2

Волнистость: 5

# 2. Конструкторско-технологическая часть

## 2.1. Выбор жизненного цикла информационной системы

Жизненный цикл программного обеспечения (ПО) — период времени, длящийся от момента принятия решения о необходимости создания программного продукта и до момента его полного изъятия из эксплуатации.

Стандарт [ГОСТ](http://ru.wikipedia.org/wiki/%D0%93%D0%9E%D0%A1%D0%A2) 34.601-90 насчитывает следующие стадии и этапы создания автоматизированной системы[22]:

1. Формирование требований к автоматизированной системе

1. Обследование объекта и обоснование необходимости создания автоматизированной системы;

2. Формирование требований пользователя к автоматизированной системе;

3. Оформление отчета о выполнении работ и заявки на разработку автоматизированной системы;

2. Разработка концепции автоматизированной системы

1. Изучение объекта

2. Проведение необходимых научно-исследовательских работ

3. Разработка вариантов концепции автоматизированной системы и выбор варианта концепции автоматизированной системы, удовлетворяющего требованиям пользователей

4. Оформление отчета о проделанной работе

3. Техническое задание

1. Разработка и утверждение технического задания на создание автоматизированной системы;

4. Эскизный проект

1. Разработка предварительных проектных решений по системе и частям автоматизированной системы;

2. Разработка документации на автоматизированную систему и ее части;

5. Технический проект

1. Разработка проектных решений по системе и частям автоматизированной системы

2. Разработка документации на автоматизированную систему и ее части;

3. Разработка и оформление документации на поставку комплектующих изделий

4. Разработка заданий на проектирование в смежных частях проекта;

6. Рабочая документация

1. Разработка рабочей документации на автоматизированную систему и ее части

2. Разработка и адаптация программ;

7. Ввод в действие

1. Подготовка объекта автоматизации;

2. Подготовка персонала;

3. Комплектация автоматизированной системы поставляемыми изделиями (программными и техническими средствами, программно-техническими комплексами, информационными изделиями);

4. Строительно-монтажные работы;

5. Пусконаладочные работы;

6. Проведение предварительных испытаний;

7. Проведение опытной эксплуатации;

8. Проведение приемочных испытаний;

8. Сопровождение автоматизированной системы

1. Выполнение работ в соответствии с гарантийными обязательствами;

2. Послегарантийное обслуживание;

Стадия «Эскизный проект» и некоторые этапы работ на любых стадиях могут быть исключены. Также разрешается объединять стадии «Технический проект» и «Рабочая документация» в «Технорабочий проект». Допускается параллельное выполнение различных этапов и работ, а также включение дополнительных.

1. Водопадная (каскадная, последовательная) модель**[21]**

Водопадная модель жизненного цикла программного обеспечения ([Уинстон Ройс](http://ru.wikipedia.org/w/index.php?title=%D0%A3%D0%B8%D0%BD%D1%81%D1%82%D0%BE%D0%BD_%D0%A0%D0%BE%D0%B9%D1%81&action=edit&redlink=1), 1970 г.) подразумевает последовательное выполнение различных этапов деятельности (анализ требований, проектирование, кодирование и тестирование отдельных модулей, тестирование сборок, интегрированное тестирование всего конечного продукта). При этом требуется четкое разграничение этапов, на которых набор документов, выработанный на предыдущей этапе, будет передан в качестве входных данных на следующий этап. Каждый вид деятельности исполняется на одной фазе жизненного цикла программного обеспечения. Движение в обратную сторону по этой цепочке невозможно.

Этапы проекта в соответствии с каскадной моделью:

1. Формирование требований;
2. Проектирование;
3. Реализация;
4. Тестирование;
5. Внедрение;
6. Эксплуатация и сопровождение.

Преимущества:

* Полная и согласованная документация на каждом этапе;
* Легко определить сроки и затраты на проект.

Недостатки:

* Накопление различных ошибок, которые были допущены на ранних стадиях проекта. Если только к концу проекта, становится очевидно, что были допущены ошибки - любой возврат к предыдущим стадиям с целью исправления ошибок становится крайне сложным. Каскадный метод не позволяет выявлять и нивелировать последствия подобных рисков.
* Неоправданное увеличение времени реализации, превышение бюджета и риск полного срыва проекта из-за накопления ошибок от этапа к этапу.
* Ключевые решения принимаются, когда у аналитиков и разработчиков нет полного понимания системы. В такой ситуации сложно уложить реальный процесс создания программного обеспечения в такую жесткую схему. В результате постоянно возникает необходимость возврата к предыдущим этапам с целью уточнения и пересмотра принятых решений. Перед началом проекта должны полностью определить требования к системе. Для ее решения необходимо тщательно и всесторонне обсудить с заказчиками и исследовать бизнес-процессы. Они должны согласиться со всем, что выясняется в ходе такого обследования. Однако, они могут и не ознакомиться до конца с его результатами. Таким образом, на стадии анализа удается собрать около 80% требований к системе. При проектировании могут возникнуть новые проблемы. Их необходимо опять обсуждать с заказчиками. Это приведет к появлению новых требований к системе. В процессе реализации и тестирования часто выясняется, что некоторые решения невозможно осуществить или, что требования были плохо детализированы и их реализация теперь некорректна. Тогда необходимо возвращаться на этап анализа и пересматривать требования.
* Метод водопада не дает возможности быстрой адаптации к изменениям, особенно на поздних стадиях жизненного цикла программного обеспечения.
* Конечный продукт может оказаться невостребованным из-за неточной формулировки требований или их изменения за длительное время создания ПО.

2. Итерационная модель[21]

Это каскадная модель с промежуточным контролем. Ее также называют эволюционной, итеративной или инкрементальной моделью. Ошибки или недоработки предыдущих стадий, обнаруженные на последующих стадиях, устраняются путем возврата к предыдущим стадиям, т.е. итерационным путем.

В отличии от каскадной модели, в итерационной модели происходит разбиение жизненного цикла проекта на последовательность итераций. Каждая итерация представляет собой «мини-проект». Она включает все процессы разработки, но только в отношении меньших компонентов функциональности. В конце каждой итерации происходит получение рабочей версии программной системы. Такая версия включает в себя функциональность, которая определена интегрированным содержанием всех предыдущих и текущей итерации. Результатом финальной итерации является продукт, который содержит всю требуемую функциональность. С завершением каждой итерации он получает приращение к своим возможностям, развивающимся эволюционно.

Итерационный подход имеет и свои недостатки, которые являются обратной стороной достоинств.

* Продолжительное время нет полного понимания какими возможностями должен обладать проект и какие должны быть наложены ограничения.
* На каждой итерации происходит отбрасывание части сделанной на предыдущем этапе работы.
* От стадии к стадии разработчики начинают выполнять свою работы все менее добросовестно. У них возникает ощущение, что «всё равно всё можно будет переделать и улучшить позже».

3. Спиральная модель[21]

[Спиральная модель](http://ru.wikipedia.org/wiki/%D0%A1%D0%BF%D0%B8%D1%80%D0%B0%D0%BB%D1%8C%D0%BD%D0%B0%D1%8F_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C) (Барри Боэм, 1980 г.) основана на классическом [цикле Деминга](http://ru.wikipedia.org/wiki/%D0%A6%D0%B8%D0%BA%D0%BB_%D0%94%D0%B5%D0%BC%D0%B8%D0%BD%D0%B3%D0%B0) PDCA (plan-do-check-act). При применении этой модели программное обеспечение разрабатывается в несколько [итераций](http://ru.wikipedia.org/wiki/%D0%98%D1%82%D0%B5%D1%80%D0%B0%D1%86%D0%B8%D1%8F) (витков спирали) [методом прототипирования](http://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D1%82%D0%BE%D1%82%D0%B8%D0%BF%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%BE%D0%B3%D0%BE_%D0%BE%D0%B1%D0%B5%D1%81%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%B8%D1%8F).

Каждый виток спирали – это создание очередной версии продукта, на которой требования к проекту уточняются, определяется его качество, планируются работы следующего витка. Особое внимание уделяется начальным стадиям разработки — анализу и проектированию, на которых проверяется и обосновывается реализуемость тех или иных технических решений посредством макетирования (создания прототипов).

Происходит углубление и последовательно конкретизируются детали проекта, в результате чего осуществляется выбор обоснованного варианта. Он будет удовлетворять действительным требованиям заказчика и доводится до реализации.

При использовании спиральной модели разрешается переходить на следующий этап, не заканчивая работы предыдущего этапа. При этом достигается решение главной задачи — как можно быстрее продемонстрировать заказчикам системы работоспособный продукт. Это позволяет ускорить процесс уточнения и дополнения требований.

На основе спиральной модели реализована технология быстрой разработки приложений (RAD-технология).

Основным недостатком спиральной модели является сложность в определении момента перехода на следующую стадию. Для того, чтобы решить эту проблему применяются временные ограничения на каждой из стадий. В результате переход выполняется в соответствии с планом, даже если не вся работа предыдущего этапа закончена. Планирование реализуется на основе статистических данных, полученных в предыдущих проектах, а также с использованием личного опыта программистов.

**Этапы проекта в соответствии с каскадной моделью:**

1. Формирование требований:

Были рассмотрены варианты реализации систем автоматизированного анализа почерка. На основании проведенного обзора были учтены недостатки и сделаны выводы о необходимости устранения самостоятельной оценки пользователем характеристик своего почерка и реализации в системе возможности работы с его оригиналом. Так же учтен факт того, что образцы почерка, необходимы для проведения анализа, обычно имеют большой объем и необходимо найти какой-либо объект, относящийся к почерку, содержащий как можно больше информации о пробанде и более подходящий для анализа, чем несколько листов текса. В качестве такого объекта была выбрана подпись пользователя.

Так же выбран способ ввода и представления подписи в системе. Подпись будет вводиться посредствам сенсорного экрана.

Таким образом, основными требованиями к системе являются:

* Отказ от самостоятельной оценки пользователем характеристик почерка;
* Возможность системы работать с оригиналом почерка;
* Возможность ввода подписи посредствам сенсорного экрана;

1. Проектирование:

Рассмотрен ряд методов анализа подписи. Составлен перечень характеристик, которые будет анализировать система.

Система разбита на три модуля:

- модуль ввода подписи;

- модуль обработки характеристик подписи;

- модуль оценивания результатов обработки характеристик подписи;

Модуль ввода подписи. Основной особенностью разрабатываемого программного обеспечения является работа с оригиналом подписи, поэтому используемое оборудование должно поддерживать сенсорный ввод.

С программной точки зрения алгоритм ввода подписи реализован следующим образом. Устройство обрабатывает события касания экрана, движения пера и его отрывания от экрана. При обработке первого и последнего события запоминаются координаты касания и отрывания пера от экрана соответственно. При обработке движения пера устройство определяет через определенный промежуток времени его координаты, которые соединяются прямыми линиями. Таким образом, подпись будет представлять собой набор точек.

Также от пользователя требуется выделить буквы подписи. Для этого требуется обвесит буквы, входящие в подпись.

Модуль обработки характеристик подписи. Для обработки характеристик подписи используются алгоритмы, разработанные на предыдущем этапе.

Модуль оценки результатов обработки характеристик подписи. Как было сказано выше, обработка каждой характеристики может иметь несколько результатов. Необходимо связать каждый результат с соответствующей ему характеристикой личности. Для этого будет использована база данных. Пример нескольких кортежей базы данных, содержащих информацию о результатах работы алгоритма и их интерпретациях, выглядит следующим образом:

|  |  |  |
| --- | --- | --- |
| Характеристика подписи | Результат анализа | Характеристика личности |
| Амплитуда | Увеличивается | Человек постепенно развивает активность от начала к концу деятельности. |
| Амплитуда | Уменьшается | Работоспособность, энергия, интерес и другие качества человека снижаются от начала к концу деятельности. У таких лиц может иметься склонность к раздражительности, утомляемости. |
| Амплитуда | Ровная | Работоспособность сохраняется на стабильном уровне от начала до конца деятельности. |

Первичным ключом базы данных будет являться составной ключ «Характеристика подписи» и «Результат анализа».

1. Реализация:

Для реализации системы были выбраны:

* Операционная система Android версии 4.0 и выше;
* IDE Eclipse, JDK 6, Android SDK и SQLite;
* Язык высокого уровня Java;

Программа разбита на несколько классов:

* MainActivity – основной класс, отображающий содержимое экрана и управляющий работой остальных классов;
* SgnAnlsView – класс, реализующий возможность ввод подписи;
* Data – класс, содержащий информацию, введенную пользователем, а также информацию, полученную в результате работы алгоритмов оценки характеристик подписи;
* Characteristics – класс, реализующий базу данных, связывающую оценку характеристик подписи, полученные в результате работы алгоритмов системы, с психологическими свойствами пробанда;
* ResultActivity – класс, отображающий результаты работы системы и выводящий на экран психологическую оценку личности;
* Analysis – класс, производящий непосредственную оценку характеристик подписи;

1. Тестирование

Для тестирования компонентов системы использовались юнит-тесты на основе JUnit4. Тестирование проводилось для:

* основных и вспомогательных функций ввода подписи;
* функций, выполняющих оценку характеристик подписи;
* функций, выполняющих передачу данных между классами;
* функций, занимающихся обработкой результатов;

Также проводилось тестирование всей системы с помощью тридцати двух подписей различных пользователей. Помимо проверки функциональности и работоспособности программы подписи использовались для сбора статистики для калибровки оценки характеристик подписи, для которых в литературе не указано точное значение.

## 2.2. Инструментальные средства разработки и обзор СУБД

1. Компилятор:

Компилятор - это программа перевода с языка программирования высокого уровня в машинные коды.

Виды компиляторов:

* Векторизующий компилятор переводит [исходный код](http://ru.wikipedia.org/wiki/%D0%98%D1%81%D1%85%D0%BE%D0%B4%D0%BD%D1%8B%D0%B9_%D0%BA%D0%BE%D0%B4) программы в [машинный код](http://ru.wikipedia.org/wiki/%D0%9C%D0%B0%D1%88%D0%B8%D0%BD%D0%BD%D1%8B%D0%B9_%D0%BA%D0%BE%D0%B4) компьютеров, оборудованных [векторным процессором](http://ru.wikipedia.org/wiki/%D0%92%D0%B5%D0%BA%D1%82%D0%BE%D1%80%D0%BD%D1%8B%D0%B9_%D0%BF%D1%80%D0%BE%D1%86%D0%B5%D1%81%D1%81%D0%BE%D1%80).
* Гибкий компилятор сконструирован по [модульному](http://ru.wikipedia.org/wiki/%D0%9C%D0%BE%D0%B4%D1%83%D0%BB%D1%8C%D0%BD%D0%BE%D1%81%D1%82%D1%8C_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)) принципу и управляется таблицами, а также и запрограммирован на [языке высокого уровня](http://ru.wikipedia.org/wiki/%D0%92%D1%8B%D1%81%D0%BE%D0%BA%D0%BE%D1%83%D1%80%D0%BE%D0%B2%D0%BD%D0%B5%D0%B2%D1%8B%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F) или реализован с помощью компилятора компиляторов.
* Инкрементальный компилятор повторно переводит фрагменты программы и дополнения к ней без перекомпиляции всей программы.
* Интерпретирующий компилятор последовательно выполняет независимую компиляцию каждого отдельной команды исходной программы.
* [Компилятор компиляторов](http://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%D0%B8%D0%BB%D1%8F%D1%82%D0%BE%D1%80_%D0%BA%D0%BE%D0%BC%D0%BF%D0%B8%D0%BB%D1%8F%D1%82%D0%BE%D1%80%D0%BE%D0%B2) – транслятор, работающий с формальным описанием [языка программирования](http://ru.wikipedia.org/wiki/%D0%AF%D0%B7%D1%8B%D0%BA_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F) и генерирующий компилятор для этого языка.
* Отладочный компилятор устраняет отдельные виды [синтаксических ошибок](http://ru.wikipedia.org/w/index.php?title=%D0%A1%D0%B8%D0%BD%D1%82%D0%B0%D0%BA%D1%81%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B0%D1%8F_%D0%BE%D1%88%D0%B8%D0%B1%D0%BA%D0%B0&action=edit&redlink=1).
* Резидентный компилятор постоянно находится в оперативной памяти и может применяться для повторного использования многими задачами.
* Самокомпилируемый компилятор написан на том же языке программирования, с которого происходит трансляция.
* Универсальный компилятор основан на формальном описании синтаксиса и семантики входного языка. Составными частями такого компилятора являются ядро, [семантический](http://ru.wikipedia.org/wiki/%D0%A1%D0%B5%D0%BC%D0%B0%D0%BD%D1%82%D0%B8%D0%BA%D0%B0_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)) и [синтаксический](http://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D0%BD%D1%82%D0%B0%D0%BA%D1%81%D0%B8%D1%81_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)) загрузчики.

Генерация кода:

1. Генерация машинного кода

Обычно компиляторы транслируют программу с некоторого [языка высокого уровня](http://ru.wikipedia.org/wiki/%D0%92%D1%8B%D1%81%D0%BE%D0%BA%D0%BE%D1%83%D1%80%D0%BE%D0%B2%D0%BD%D0%B5%D0%B2%D1%8B%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F) в [машинный код](http://ru.wikipedia.org/wiki/%D0%9C%D0%B0%D1%88%D0%B8%D0%BD%D0%BD%D1%8B%D0%B9_%D0%BA%D0%BE%D0%B4), который непосредственно выполняется [процессором](http://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D1%86%D0%B5%D1%81%D1%81%D0%BE%D1%80). Этот код выполняется в среде той [операционной системы](http://ru.wikipedia.org/wiki/%D0%9E%D0%BF%D0%B5%D1%80%D0%B0%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0), для которой был написан. Он использует ее возможности (системные вызовы, библиотеки функций). Под целевой машиной подразумевают архитектуру, для которой выполняется компиляция.

Результатом компиляции будет являться исполнимый модуль. Он будет иметь максимально возможную производительность. Недостатком будет являться то, что он привязан к определённой операционной системе и процессору.

Написание компилятора требуется для каждой целевой машины и каждой операционной системы или семейства операционных систем, работающих на целевой машине.

Компиляторы, которые позволяют на одной машине и в среде одной операционной системы создавать код, подходящий для выполнения на другой целевой машине и в среде другой операционной системе, называются кросс-компиляторами. Такие компиляторы могут выполнять оптимизацию кода для разных моделей процессоров из одного семейства.

Существуют компиляторы, которые транслируют программу с высокоуровневого языка программирования не напрямую в машинный код, а сначала на язык ассемблера. Это позволяет упростить те части компилятора, которые отвечают за генерацию кода и повышение его переносимости. Также это помогает корректировать результаты компиляции.

Генерация байт-кода

Некоторые компиляторы генерируют программы на проблемно-ориентированном [языке](http://ru.wikipedia.org/wiki/%D0%9D%D0%B8%D0%B7%D0%BA%D0%BE%D1%83%D1%80%D0%BE%D0%B2%D0%BD%D0%B5%D0%B2%D1%8B%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F) низкого уровня. Такие программы интерпретируются виртуальной машиной, а подобный язык носит название псевдокод или [байт-код](http://ru.wikipedia.org/wiki/%D0%91%D0%B0%D0%B9%D1%82-%D0%BA%D0%BE%D0%B4). Он не является машинным кодом конкретного процессора. Программы на нём могут исполняться на различных архитектурах, имеющих соответствующую виртуальную машину. Для некоторых случаев специально разрабатываются аппаратные платформы, которые напрямую поддерживают псевдокод какого-либо языка. В качестве примера можно привесит псевдокод языка Java. Он называется [байт-кодом Java](http://ru.wikipedia.org/wiki/%D0%91%D0%B0%D0%B9%D1%82-%D0%BA%D0%BE%D0%B4_Java) и исполняется в [Java Virtual Machine](http://ru.wikipedia.org/wiki/Java_Virtual_Machine). Для его прямого выполнения была создана спецификация процессора [picoJava](http://ru.wikipedia.org/wiki/PicoJava).

Некоторые реализации интерпретируемых высокоуровневых языков применяют байт-код для оптимизации выполнения программы. Дорогостоящие этапы синтаксического анализа и перевод текста программы в байт-код производятся один раз при загрузке. Далее выполненный код может применяться напрямую.

1. Динамическаякомпиляция

Байт-код требует интерпретации. В результате, он исполняется значительно медленнее аналогичного машинного кода. Но в отличие от него, байт-код гораздо легче переносится на другие операционные системы или модели процессора. Динамическая компиляция применяется для увеличения скорости исполнения байт-кода. Перед его первым исполнением виртуальная машина переводит псевдокод в машинный код. При последующих обращениях к коду будет выполняться уже скомпилированный вариант.[23]

Так как основным языком разработки для платформы Android является язык высокого уровня Java, то выбор языка программирования был сделан в пользу него. Компилятором языка Java является Javac.

Javac — [компилятор](http://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%D0%B8%D0%BB%D1%8F%D1%82%D0%BE%D1%80) языка [java](http://ru.wikipedia.org/wiki/Java), включенный в [Java Development Kit](http://ru.wikipedia.org/wiki/Java_Development_Kit) (JDK) от [Oracle](http://ru.wikipedia.org/wiki/Oracle).

Компилятор принимает [исходные коды](http://ru.wikipedia.org/wiki/%D0%98%D1%81%D1%85%D0%BE%D0%B4%D0%BD%D1%8B%D0%B9_%D0%BA%D0%BE%D0%B4), сооветствующий спецификации [Java language specification](http://ru.wikipedia.org/wiki/Java) (JLS), и возвращает [байт-код](http://ru.wikipedia.org/wiki/%D0%91%D0%B0%D0%B9%D1%82-%D0%BA%D0%BE%D0%B4), соответствующий спецификации [Java Virtual Machine](http://ru.wikipedia.org/wiki/Java_Virtual_Machine) Specification (JVMS).

Javac сам написан на [Java](http://ru.wikipedia.org/wiki/Java)[25].

1. Eclipse:

Интегрированная среда разработки — система программных средств, применяемая [программистами](http://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%81%D1%82) для разработки [программного обеспечения](http://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%BE%D0%B5_%D0%BE%D0%B1%D0%B5%D1%81%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%B8%D0%B5) (ПО).

Обычно среда разработки включает в себя:

* [текстовый редактор](http://ru.wikipedia.org/wiki/%D0%A2%D0%B5%D0%BA%D1%81%D1%82%D0%BE%D0%B2%D1%8B%D0%B9_%D1%80%D0%B5%D0%B4%D0%B0%D0%BA%D1%82%D0%BE%D1%80)
* [компилятор](http://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%D0%B8%D0%BB%D1%8F%D1%82%D0%BE%D1%80) и/или [интерпретатор](http://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%82%D0%B5%D1%80%D0%BF%D1%80%D0%B5%D1%82%D0%B0%D1%82%D0%BE%D1%80)
* средства автоматизации сборки
* [отладчик](http://ru.wikipedia.org/wiki/%D0%9E%D1%82%D0%BB%D0%B0%D0%B4%D1%87%D0%B8%D0%BA).

Иногда имеет также средства для интеграции с [системами управления версиями](http://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D1%83%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D1%8F_%D0%B2%D0%B5%D1%80%D1%81%D0%B8%D1%8F%D0%BC%D0%B8) и разнообразные инструменты для упрощения создания [графического интерфейса пользователя](http://ru.wikipedia.org/wiki/%D0%93%D1%80%D0%B0%D1%84%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B8%D0%B9_%D0%B8%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81_%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8F). Многие современные среды разработки имеют помимо прочего еще и браузер классов, инспектор объектов и диаграмму [иерархии классов](http://ru.wikipedia.org/wiki/%D0%98%D0%B5%D1%80%D0%B0%D1%80%D1%85%D0%B8%D1%8F_%D0%BA%D0%BB%D0%B0%D1%81%D1%81%D0%BE%D0%B2) — для использования при [объектно-ориентированной разработке](http://ru.wikipedia.org/wiki/%D0%9E%D0%B1%D1%8A%D0%B5%D0%BA%D1%82%D0%BD%D0%BE-%D0%BE%D1%80%D0%B8%D0%B5%D0%BD%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B5%D0%BA%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5) программного обеспечения. Существуют интегрированные среды разработки, применяемые для разработки программного обеспечения на нескольких [языках программирования](http://ru.wikipedia.org/wiki/%D0%AF%D0%B7%D1%8B%D0%BA_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F): [Eclipse](http://ru.wikipedia.org/wiki/Eclipse_(%D1%81%D1%80%D0%B5%D0%B4%D0%B0_%D1%80%D0%B0%D0%B7%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B8)), [NetBeans](http://ru.wikipedia.org/wiki/NetBeans_IDE), [Embarcadero RAD Studio](http://ru.wikipedia.org/wiki/Embarcadero_RAD_Studio), [Qt Creator](http://ru.wikipedia.org/wiki/Qt_Creator) и [Microsoft Visual Studio](http://ru.wikipedia.org/wiki/Microsoft_Visual_Studio). Однако, обычно интегрированная среда разработки предназначается только для определённого языка программирования: [Visual Basic](http://ru.wikipedia.org/wiki/Visual_Basic), [Delphi](http://ru.wikipedia.org/wiki/Delphi_(%D1%81%D1%80%D0%B5%D0%B4%D0%B0_%D1%80%D0%B0%D0%B7%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B8)), [Dev-C++](http://ru.wikipedia.org/wiki/Dev-C%2B%2B).

Eclipse — [интегрированная среда разработки](http://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%82%D0%B5%D0%B3%D1%80%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%BD%D0%B0%D1%8F_%D1%81%D1%80%D0%B5%D0%B4%D0%B0_%D1%80%D0%B0%D0%B7%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B8) для [модульных](http://ru.wikipedia.org/wiki/Plug-in_Development_Environment) кроссплатформенных приложений.

Различные «Eclipse IDE» являются наиболее распространенными приложениями, которые разработаны на основе [Eclipse Platform](http://ru.wikipedia.org/wiki/Eclipse_Platform). Они применяются для разработки программного обеспечения на множестве языков программирования. Наиболее известной является «[Java](http://ru.wikipedia.org/wiki/Java" \o "Java) IDE». Она поддерживается изначально и не полагается на какие-либо закрытые расширения, а также применяет стандартный открытый [API](http://ru.wikipedia.org/wiki/API) для доступа к Eclipse Platform.

Основой Eclipse является платформа расширенного клиента. Эту платформу составляют следующие компоненты:

* Ядро платформы (загрузка Eclipse, запуск модулей);
* [OSGi](http://ru.wikipedia.org/wiki/OSGi) (стандартная среда поставки комплектов);
* [SWT](http://ru.wikipedia.org/wiki/Standard_Widget_Toolkit) (портируемый инструментарий [виджетов](http://ru.wikipedia.org/wiki/%D0%92%D0%B8%D0%B4%D0%B6%D0%B5%D1%82" \o "Виджет));
* [JFace](http://ru.wikipedia.org/wiki/JFace) ([файловые](http://ru.wikipedia.org/wiki/%D0%A4%D0%B0%D0%B9%D0%BB) буферы, работа с [текстом](http://ru.wikipedia.org/wiki/%D0%A2%D0%B5%D0%BA%D1%81%D1%82), [текстовые редакторы](http://ru.wikipedia.org/wiki/%D0%A2%D0%B5%D0%BA%D1%81%D1%82%D0%BE%D0%B2%D1%8B%D0%B9_%D1%80%D0%B5%D0%B4%D0%B0%D0%BA%D1%82%D0%BE%D1%80));
* Рабочая среда Eclipse (панели, редакторы, проекции, мастеры).

[Графический](http://ru.wikipedia.org/wiki/%D0%93%D1%80%D0%B0%D1%84%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B8%D0%B9_%D0%B8%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81_%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8F) интерфейс в Eclipse написан с применением Standart Widget Toolkit. SWT самостоятельно эмулирует графические элементы управления. Он применяет графические компоненты той операционной системы, для которой разрабатывается приложение. Графический интерфейс Eclipse также зависит от промежуточного слоя GUI (JFace). Он упрощает создание пользовательского интерфейса, базирующегося на Standart Widget Toolkit.

Подключаемые модули обеспечивают гибкость. В результате имеется возможность разрабатывать приложения не только на Java, но и на других языках: [C](http://ru.wikipedia.org/wiki/%D0%A1%D0%B8_(%D1%8F%D0%B7%D1%8B%D0%BA_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F))/[C++](http://ru.wikipedia.org/wiki/C%2B%2B), [Perl](http://ru.wikipedia.org/wiki/Perl), [Groovy](http://ru.wikipedia.org/wiki/Groovy), [Ruby](http://ru.wikipedia.org/wiki/Ruby),[Python](http://ru.wikipedia.org/wiki/Python), [PHP](http://ru.wikipedia.org/wiki/PHP), [Erlang](http://ru.wikipedia.org/wiki/Erlang), [Zonnon](http://ru.wikipedia.org/wiki/Zonnon) и прочих.[26]

1. Android SDK:

Android Software Development Kit – набор инструментов и библиотек для разработки программного обеспечения для платформы Android.[27]

1. СУБД:

Система управления базами данных (СУБД) — совокупность программных и лингвистических средств общего или специального назначения, которые позволяют управлять созданием и использованием [баз данных](http://ru.wikipedia.org/wiki/%D0%91%D0%B0%D0%B7%D0%B0_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85).

Основные функции СУБД

* управление данными во [внешней памяти](http://ru.wikipedia.org/wiki/%D0%92%D0%BD%D0%B5%D1%88%D0%BD%D1%8F%D1%8F_%D0%BF%D0%B0%D0%BC%D1%8F%D1%82%D1%8C);
* управление данными в [оперативной памяти](http://ru.wikipedia.org/wiki/%D0%9E%D0%BF%D0%B5%D1%80%D0%B0%D1%82%D0%B8%D0%B2%D0%BD%D0%B0%D1%8F_%D0%BF%D0%B0%D0%BC%D1%8F%D1%82%D1%8C) с применением [дискового кэша](http://ru.wikipedia.org/wiki/%D0%94%D0%B8%D1%81%D0%BA%D0%BE%D0%B2%D1%8B%D0%B9_%D0%BA%D1%8D%D1%88);
* [журнализация изменений](http://ru.wikipedia.org/wiki/%D0%96%D1%83%D1%80%D0%BD%D0%B0%D0%BB%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D1%8F_%D0%B8%D0%B7%D0%BC%D0%B5%D0%BD%D0%B5%D0%BD%D0%B8%D0%B9), [резервное копирование](http://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D0%B7%D0%B5%D1%80%D0%B2%D0%BD%D0%BE%D0%B5_%D0%BA%D0%BE%D0%BF%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5) и [восстановление базы данных](http://ru.wikipedia.org/wiki/%D0%92%D0%BE%D1%81%D1%81%D1%82%D0%B0%D0%BD%D0%BE%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5_%D0%B1%D0%B0%D0%B7%D1%8B_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85) после сбоев;
* поддержка языков БД.

Обычно СУБД имеет следующие компоненты:

* Ядро. Отвечает за контроль данных во внешней и оперативной памяти и [журнализацию](http://ru.wikipedia.org/wiki/%D0%96%D1%83%D1%80%D0%BD%D0%B0%D0%BB%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D1%8F_%D0%B8%D0%B7%D0%BC%D0%B5%D0%BD%D0%B5%D0%BD%D0%B8%D0%B9);
* Процессор языка базы данных. Позволяет оптимизировать запросы на извлечение и изменение данных и создание машинно-независимого исполняемого внутреннего кода;
* Подсистема поддержки времени исполнения. Интерпретирует программы управления данными, которые создают пользовательский интерфейс с СУБД
* Сервисные программы. Имеют ряд дополнительных возможностей по обслуживанию информационной системы.

Классификации СУБД

По [модели данных](http://ru.wikipedia.org/wiki/%D0%9C%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85)

Примеры:

* [Иерархические](http://ru.wikipedia.org/wiki/%D0%98%D0%B5%D1%80%D0%B0%D1%80%D1%85%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B0%D1%8F_%D0%A1%D0%A3%D0%91%D0%94)
* [Сетевые](http://ru.wikipedia.org/wiki/%D0%A1%D0%B5%D1%82%D0%B5%D0%B2%D0%B0%D1%8F_%D0%A1%D0%A3%D0%91%D0%94)
* [Реляционные](http://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D0%BB%D1%8F%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D0%A1%D0%A3%D0%91%D0%94)
* [Объектно-ориентированные](http://ru.wikipedia.org/wiki/%D0%9E%D0%B1%D1%8A%D0%B5%D0%BA%D1%82%D0%BD%D0%BE-%D0%BE%D1%80%D0%B8%D0%B5%D0%BD%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%BD%D0%B0%D1%8F_%D0%A1%D0%A3%D0%91%D0%94)
* [Объектно-реляционные](http://ru.wikipedia.org/wiki/%D0%9E%D0%B1%D1%8A%D0%B5%D0%BA%D1%82%D0%BD%D0%BE-%D1%80%D0%B5%D0%BB%D1%8F%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D0%A1%D0%A3%D0%91%D0%94)

По степени распределенности

* Локальные СУБД (все части локальной СУБД размещаются на одном компьютере)
* Распределённые СУБД (части СУБД могут размещаться на двух и более компьютерах).

По способу доступа к БД

* [Файл-серверные](http://ru.wikipedia.org/wiki/%D0%A4%D0%B0%D0%B9%D0%BB-%D1%81%D0%B5%D1%80%D0%B2%D0%B5%D1%80)

Файлы данных в таких СУБД расположены централизованно на [файл-сервере](http://ru.wikipedia.org/wiki/%D0%A4%D0%B0%D0%B9%D0%BB-%D1%81%D0%B5%D1%80%D0%B2%D0%B5%D1%80), а сама СУБД находится на всех клиентских компьютерах. Доступ СУБД к данным выполняется посредствам [локальной](http://ru.wikipedia.org/wiki/%D0%9B%D0%BE%D0%BA%D0%B0%D0%BB%D1%8C%D0%BD%D0%B0%D1%8F_%D0%B2%D1%8B%D1%87%D0%B8%D1%81%D0%BB%D0%B8%D1%82%D0%B5%D0%BB%D1%8C%D0%BD%D0%B0%D1%8F_%D1%81%D0%B5%D1%82%D1%8C) сети, а синхронизация чтений и обновлений выполняется через файловые блокировки. Преимущество такой архитектуры заключается в низкой нагрузке на процессор файлового сервера, а недостатками являются: потенциально высокая загрузка локальной сети; затруднённость или невозможность [централизованного](http://ru.wikipedia.org/wiki/%D0%A6%D0%B5%D0%BD%D1%82%D1%80%D0%B0%D0%BB%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D1%8F) [управления](http://ru.wikipedia.org/wiki/%D0%A3%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5), затруднённость или невозможность обеспечения таких важных характеристик как высокая [надёжность](http://ru.wikipedia.org/wiki/%D0%9D%D0%B0%D0%B4%D1%91%D0%B6%D0%BD%D0%BE%D1%81%D1%82%D1%8C), [высокая доступность](http://ru.wikipedia.org/wiki/%D0%92%D1%8B%D1%81%D0%BE%D0%BA%D0%B0%D1%8F_%D0%B4%D0%BE%D1%81%D1%82%D1%83%D0%BF%D0%BD%D0%BE%D1%81%D1%82%D1%8C) и высокая [безопасность](http://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D0%B1%D0%B5%D0%B7%D0%BE%D0%BF%D0%B0%D1%81%D0%BD%D0%BE%D1%81%D1%82%D1%8C). Файл-серверные СУБД применяются чаще всего в локальных приложениях, использующих функции управления базой данных, в системах с низкой интенсивностью обработки данных и низкими пиковыми нагрузками на базу данных.

Сегодня файл-серверная технология является устаревшей. Применение же ее в крупных информационных системах считается недостатком.

Примеры: [Microsoft Access](http://ru.wikipedia.org/wiki/Microsoft_Access), [Paradox](http://ru.wikipedia.org/wiki/Paradox), [dBase](http://ru.wikipedia.org/wiki/DBase), [FoxPro](http://ru.wikipedia.org/wiki/FoxPro), [Visual FoxPro](http://ru.wikipedia.org/wiki/Visual_FoxPro).

* [Клиент-серверные](http://ru.wikipedia.org/wiki/%D0%9A%D0%BB%D0%B8%D0%B5%D0%BD%D1%82-%D1%81%D0%B5%D1%80%D0%B2%D0%B5%D1%80%D0%BD%D0%B0%D1%8F_%D0%A1%D0%A3%D0%91%D0%94)

Клиент-серверная СУБД находится на сервере вместе с базой данных. Она осуществляет доступ к ней непосредственно в монопольном режиме. Все клиентские запросы на обработку данных выполняет клиент-серверная СУБД. Клиент-серверные СУБД отличают повышенные требования к серверу, что является недостатком. К достоинствам можно отнести: потенциально более низкая загрузка локальной сети; удобство [централизованного](http://ru.wikipedia.org/wiki/%D0%A6%D0%B5%D0%BD%D1%82%D1%80%D0%B0%D0%BB%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D1%8F) [управления](http://ru.wikipedia.org/wiki/%D0%A3%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5); удобство обеспечения таких важных характеристик как высокая [надёжность](http://ru.wikipedia.org/wiki/%D0%9D%D0%B0%D0%B4%D1%91%D0%B6%D0%BD%D0%BE%D1%81%D1%82%D1%8C), [высокая доступность](http://ru.wikipedia.org/wiki/%D0%92%D1%8B%D1%81%D0%BE%D0%BA%D0%B0%D1%8F_%D0%B4%D0%BE%D1%81%D1%82%D1%83%D0%BF%D0%BD%D0%BE%D1%81%D1%82%D1%8C) и высокая [безопасность](http://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D0%B1%D0%B5%D0%B7%D0%BE%D0%BF%D0%B0%D1%81%D0%BD%D0%BE%D1%81%D1%82%D1%8C).

Примеры: [Oracle](http://ru.wikipedia.org/wiki/Oracle_(%D0%A1%D0%A3%D0%91%D0%94)), [Firebird](http://ru.wikipedia.org/wiki/Firebird), [Interbase](http://ru.wikipedia.org/wiki/Interbase), [IBM DB2](http://ru.wikipedia.org/wiki/IBM_DB2), [Informix](http://ru.wikipedia.org/wiki/Informix), [MS SQL Server](http://ru.wikipedia.org/wiki/MS_SQL_Server), [Sybase Adaptive Server Enterprise](http://ru.wikipedia.org/wiki/Sybase#.D0.9F.D1.80.D0.BE.D0.B4.D1.83.D0.BA.D1.86.D0.B8.D1.8F), [PostgreSQL](http://ru.wikipedia.org/wiki/PostgreSQL), [MySQL](http://ru.wikipedia.org/wiki/MySQL), [Caché](http://ru.wikipedia.org/wiki/Cach%C3%A9), [ЛИНТЕР](http://ru.wikipedia.org/wiki/%D0%9B%D0%98%D0%9D%D0%A2%D0%95%D0%A0).

* [Встраиваемые](http://ru.wikipedia.org/wiki/%D0%92%D1%81%D1%82%D1%80%D0%B0%D0%B8%D0%B2%D0%B0%D0%B5%D0%BC%D0%B0%D1%8F_%D0%A1%D0%A3%D0%91%D0%94)

Встраиваемая СУБД поставляется как составная часть программного продукта. Она не требует процедуры самостоятельной [установки](http://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%81%D1%82%D0%B0%D0%BB%D0%BB%D1%8F%D1%86%D0%B8%D1%8F_(%D0%9F%D0%9E)). Такая СУБД применяется для локального хранения данных своего приложения, но не рассчитана на коллективное использование в сети. Встраиваемая СУБД представлена в виде [подключаемой библиотеки](http://ru.wikipedia.org/wiki/%D0%91%D0%B8%D0%B1%D0%BB%D0%B8%D0%BE%D1%82%D0%B5%D0%BA%D0%B0_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)). Доступ к данным со стороны приложения может выполнятся через [SQL](http://ru.wikipedia.org/wiki/SQL) либо через специальные [программные интерфейсы](http://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D1%8B%D0%B9_%D0%B8%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81).

Примеры: [OpenEdge](http://ru.wikipedia.org/w/index.php?title=OpenEdge&action=edit&redlink=1), [SQLite](http://ru.wikipedia.org/wiki/SQLite), [BerkeleyDB](http://ru.wikipedia.org/wiki/BerkeleyDB), [Firebird](http://ru.wikipedia.org/wiki/Firebird) Embedded, [Microsoft SQL Server Compact](http://ru.wikipedia.org/wiki/Microsoft_SQL_Server_Compact), [ЛИНТЕР](http://ru.wikipedia.org/wiki/%D0%9B%D0%98%D0%9D%D0%A2%D0%95%D0%A0).[24]

SQLite является встроенной библиотекой для работы с СУБД на платформ Android.

SQLite — легковесная [встраиваемая](http://ru.wikipedia.org/wiki/%D0%92%D1%81%D1%82%D1%80%D0%B0%D0%B8%D0%B2%D0%B0%D0%B5%D0%BC%D0%B0%D1%8F_%D0%A1%D0%A3%D0%91%D0%94) [реляционная база данных](http://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D0%BB%D1%8F%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D0%B1%D0%B0%D0%B7%D0%B0_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85).

SQLite не использует парадигму [клиент-сервер](http://ru.wikipedia.org/wiki/%D0%9A%D0%BB%D0%B8%D0%B5%D0%BD%D1%82-%D1%81%D0%B5%D1%80%D0%B2%D0%B5%D1%80). Ее [движок](http://ru.wikipedia.org/wiki/%D0%94%D0%B2%D0%B8%D0%B6%D0%BE%D0%BA) для не является отдельно работающим процессом для программы, с которой он взаимодействует. SQLite предоставляет [библиотеку](http://ru.wikipedia.org/wiki/%D0%91%D0%B8%D0%B1%D0%BB%D0%B8%D0%BE%D1%82%D0%B5%D0%BA%D0%B0_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)). Программа компонуется вместе с ней, в результате чего движок превращается в составную частью программы.

В качестве протокола обмена используются вызовы функций библиотеки SQLite. Следствием такого подхода является уменьшение накладных расходов, времени отклика и упрощение программы. SQLite хранит всю базу данных, в том числе определения, таблицы, индексы и данные, в одном файле на том компьютере, на котором выполняется программа. Перед началом выполнения транзакции записи происходит блокировка файла, в котором храниться база данных. В результате чего достигается простота реализации. [ACID](http://ru.wikipedia.org/wiki/ACID)-функции достигаются посредствам создания файла журнала.

Нескольким процессам или потокам разрешается одновременно считывать данные из одной базы. Только в том случае, если никаких других запросов в данный момент нет, в базу может быть осуществлена запись. Иначе запись оканчивается неудачей, и в программу возвращается код ошибки.

Также в течение заданного интервала времени может выполняться автоматическое повторение попыток записи.

В комплекте поставки имеется также функциональная клиентская часть в виде исполняемого файла sqlite3. Она демонстрирует реализацию функций основной библиотеки. Клиентская часть работает из командной строки, позволяет обращаться к файлу БД на основе типовых функций ОС.

Возможность использовать SQLite как на [встраиваемых системах](http://ru.wikipedia.org/wiki/%D0%92%D1%81%D1%82%D1%80%D0%B0%D0%B8%D0%B2%D0%B0%D0%B5%D0%BC%D0%B0%D1%8F_%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0), так и на выделенных машинах с гигабайтными массивами данных, достигается за счет архитектуры движка.

## 2.3. Реализация на ЭВМ

### 2.3.1. Структура классов

MainActivity – основной класс, контролирующий работу всех остальных.

OnCreate – выполняет инициализацию объектов всех необходимых классов и переменных;

OnCreateOptionMenu – создает меню;

OnOptionItemSelected – определяет, какой из пунктов меню был выбран, и выполняет соответствующие действия;

SgnAnlsView – класс, занимающийся работой с графикой и управляющий вводом подписи.

OnDraw – выполняет отрисовку содержимого экрана;

OnTouchEvent – определяет момент касания экрана и выполняет соответствующие действия;

touchStarted – обрабатывает касание экрана;

touchMoved – обрабатывает движение по экрану;

touchEnd – обрабатывает отрывание от экрана;

isContourCrossSign – определяет пересечение области выделения с подписью;

isSignGetsIntoContour – определяет попадание подписи в область выделения;

isInputedCotourClosed – определяет замкнутость контура области выделения

Characteristics – класс, управляющий работой с базой данных, содержащий информацию о характеристиках подписи и личности.

OnCreate – создает базу данных;

getPersonCharacteristic – возвращает характеристику личности по первичному ключу (характеристика подписи и результат анализа);

fillDatabase – заполняет базу данных при инициализации;

ResultActivity – класс, управляющий отображением результатов анализа

onCreate – создает элементы окна;

HelpActivity – класс, выполняющий отображения справки

onCreate – создает элементы окна;

Symbol\_db – класс, управляющий работой с базой данных, содержащей эталоны букв и цифр.

onCreate - создает базу данных;

getSymbolFromDatabase – возвращает матрицу символа и сам символ по номеру строки;

fillRussianSymbolDatabase – заполняет таблицу русских символов при инициализации;

fillEnglishSymbolDatabase – заполняет таблицу иностранных символов при инициализации;

fillDecimalSymbolDatabase – заполняет таблицу цифр при инициализации;

Analysis – класс, выполняющий анализ подписи.

AnalyzeSignature – выполняет анализ подписи (вызывает ниже перечисленные функции);

defineLenghtOfSign – определяет длину подписи;

defineSpeedOfInputing – определяет скорость ввода подписи;

findVerticalLinesOrUnderlines – определяет наличии вертикальных линий и подчеркиваний в подписи;

defineSignRoundnessOrSharpness – определяет закругленность и остроту букв;

defineSignAmplitude – определяет возрастание или убывание амплитуды;

defineSignDirection – определяет направление подписи;

defineSignSweep – определяет размашистость подписи;

findPointsInSign – определяет наличие точки в подписи;

defineSignPitch – определяет наклон подписи;

findReturningInSign – определяет возвраты в подписи;

defineLettersConnectedness – определяет связанность и разорванность букв;

defineSignStressLoading – определяет нагруженность подписи;

defineFirstLetterSize – определяет величину первой буквы подписи;

defineSignSlickness – определяет ровность подписи;

defineLettersDistance – определяет расстояние между буквами;

analyzeSignBegins – анализирует начало подписи;

findSignLoops – определяет наличие петель в подписи;

defineSignUndulation – определяет волнистость подписи;

defineLegibility – определяет ясность подписи, а также наличие цифр и иностранных букв;

defineForeignLettersInSign – определяет наличие иностранных букв в подписи;

defineFiguresInSign – определяет наличие цифр в подписи;

isLinesCrossed – определяет пересечение двух отрезков, концы которых заданы координатами;

isPointBelongToLetterMarquee – определяет принадлежность точки некоторой замкнутой области;

findMinMaxLabel – находит локальные максимумы и минимумы в подписию.

### 2.3.2. Структура данных

На входе программы пользователь должен ввести свою подпись. Ввод будет осуществляться следующим образом:

* при касании экрана программа запоминает точку касания;
* при движении программа через определенные промежутки времени определяет место нахождения пера, запоминает точку и чертит прямую линий от предыдущей точки к текущей точке;
* при отрывании пера от экрана программа запоминает точку отрыва и чертит прямую линию от предыдущей точки к текущей точке;

В результате подпись оказывается представленной в виду упорядоченного набора контрольных точек. Такое представление удобно тем, что появляется возможность проследить направление движения пера, понять очередность ввода траекторий, составляющих подпись, количество точек в каждой траектории, обнаружить места разрывов; более просто определять точки локальных максимумов и минимумов.

Помимо прочего программа отслеживает время в момент касания и отрывания пера от экрана и суммирует их, что позволяет определить время ввода подписи, что также является одной из психологических характеристик личности.

Кроме того, программе требуется информация о буквах, составляющих подпись. От пользователя требуется самостоятельно выделить каждую букву, в результате чего программа получает упорядоченный набор областей выделения, что дает информацию о количестве букв, какие траектории не являются частью букв, позволяет рассчитать наклон букв, их направление и ровность.

Получив входные данные, система обрабатывает их с помощь соответствующих алгоритмов. На выходе каждый алгоритм дает оценку каждой из двадцати трех характеристик подписи. Все оценки записываются в таблицу напротив названия оцениваемой характеристики.

В системе для оценки ясности подписи используется база данных на основе SQLite. Она состоит из одной таблицы, которая в свою очередь состоит из трех столбцов, содержащих номер строки, матрицу символа и сам символ. Пример одного кортежа выглядит следующим образом:

|  |  |  |
| --- | --- | --- |
| Номер строки | Матрица символа | Символ |
| 1 | 0000000000011000000000000011100000000000001110000000000001111000000000000101100000000000110110000000000010010000000000011001000001111111111111110000000100010000000000110001000010000010000100001000011000010000110001000001000001001100000110000011100000010000 | «А» |

Матрица символа имеет целочисленный тип и размер 16х16, но для удобства она представлена в виде строки. При извлечении информации из базы данных, она преобразовывается обратно в матрицу целочисленного типа и приобретает размер 16х16.

К системе подключена база данных результатов также на основе SQLite. Она состоит из трех столбцов: характеристика подписи, результат анализа, характеристика личности. Пример нескольких кортежей базы данных, содержащих информацию о результатах работы алгоритма и их интерпретациях, выглядит следующим образом:

|  |  |  |
| --- | --- | --- |
| Характеристика подписи | Результат анализа | Характеристика личности |
| Амплитуда | Увеличивается | Человек постепенно развивает активность от начала к концу деятельности. |
| Амплитуда | Уменьшается | Работоспособность, энергия, интерес и другие качества человека снижаются от начала к концу деятельности. У таких лиц может иметься склонность к раздражительности, утомляемости. |
| Амплитуда | Ровная | Работоспособность сохраняется на стабильном уровне от начала до конца деятельности. |

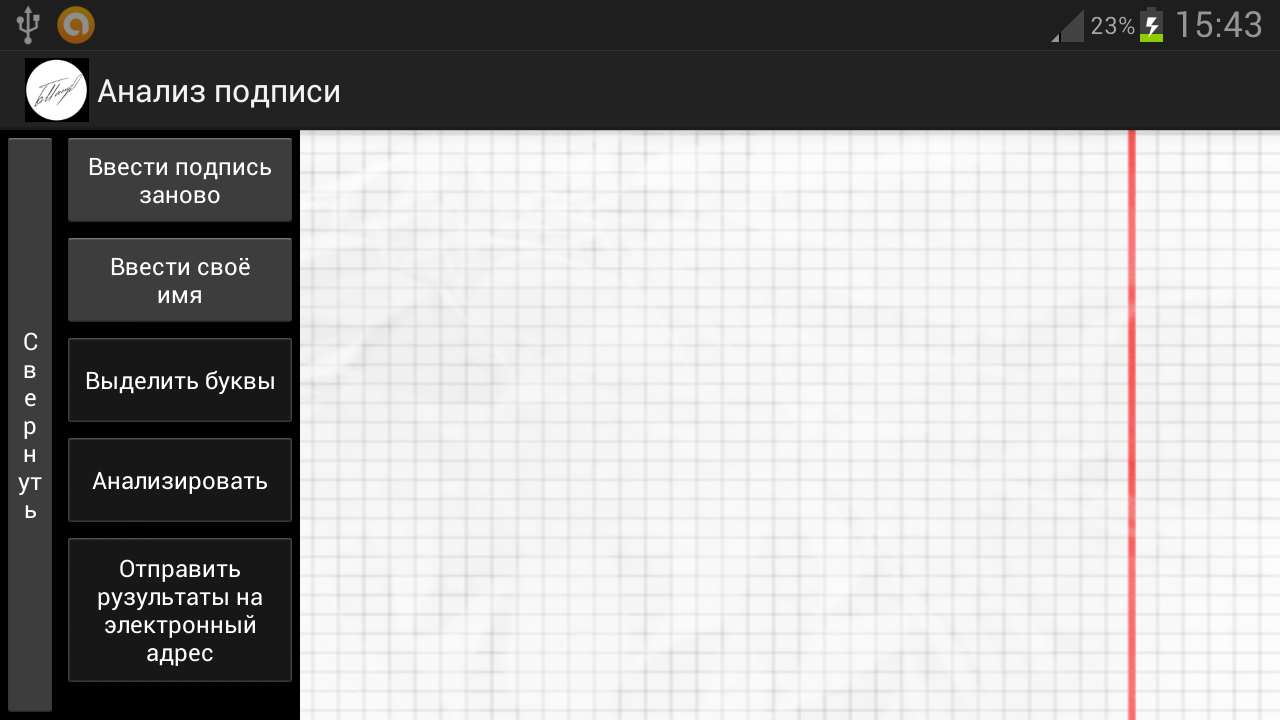
Первичным ключом базы данных будет являться составной ключ «Характеристика подписи» и «Результат анализа».

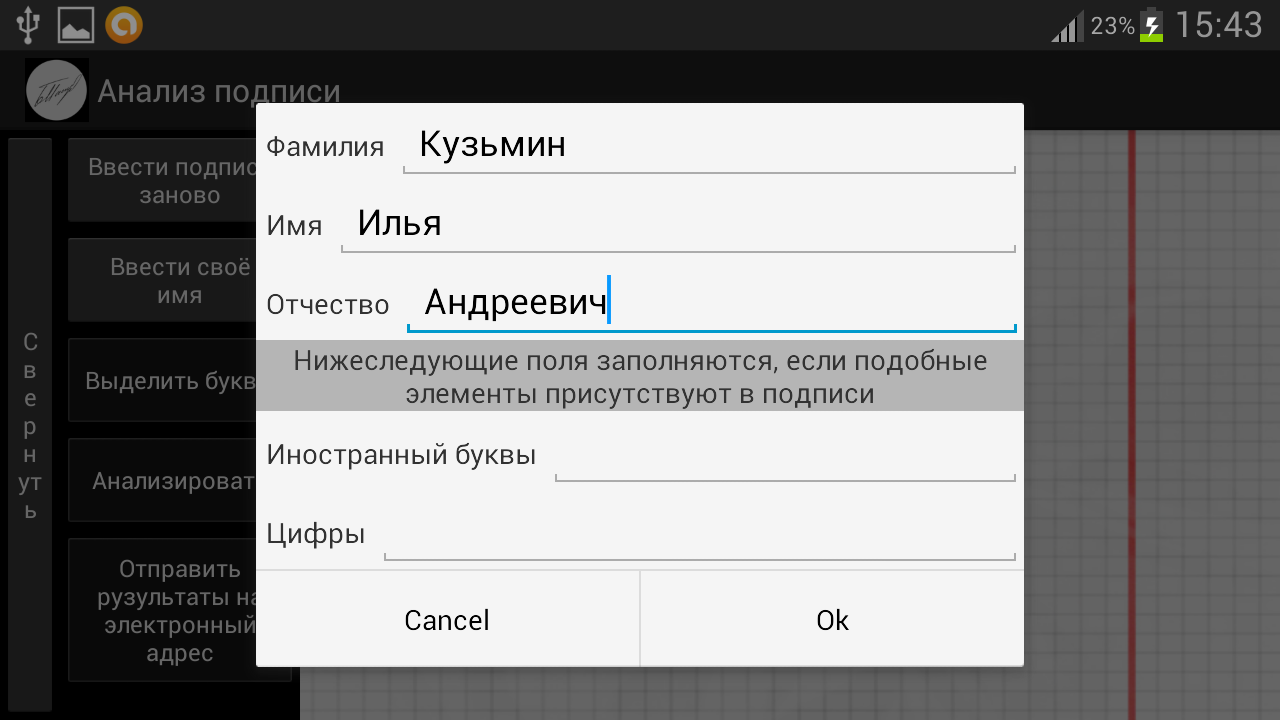
Таким образом, таблица результатов, полученная при работе системы, содержит в себе первичные ключи базы данных и используется для получения информации о психологических характеристиках пользователя.

В качестве результата работы системы пользователю предоставляется информация об анализируемых характеристиках, их оценках и соответствующих характеристиках личности.

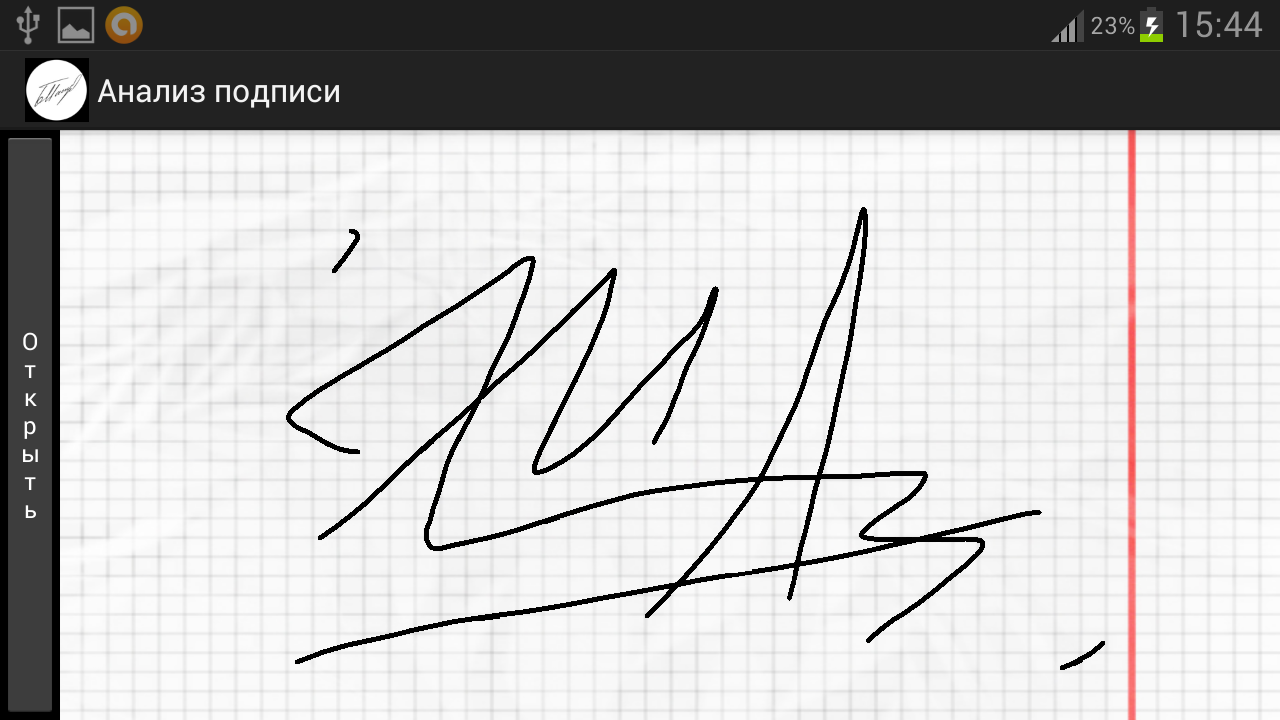
### 2.3.3 Пример функционирования

1) Запуск программы. Основной интерфейс пользователя.



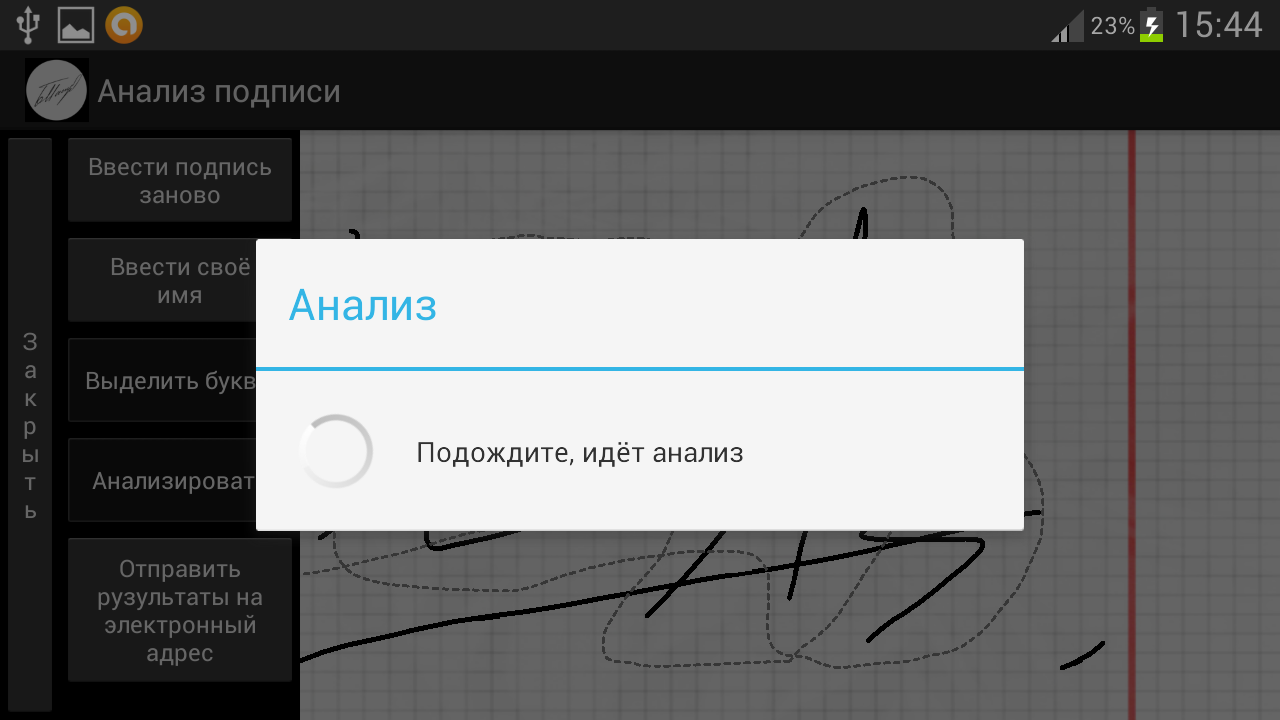
2) Ввод имени пользователя.

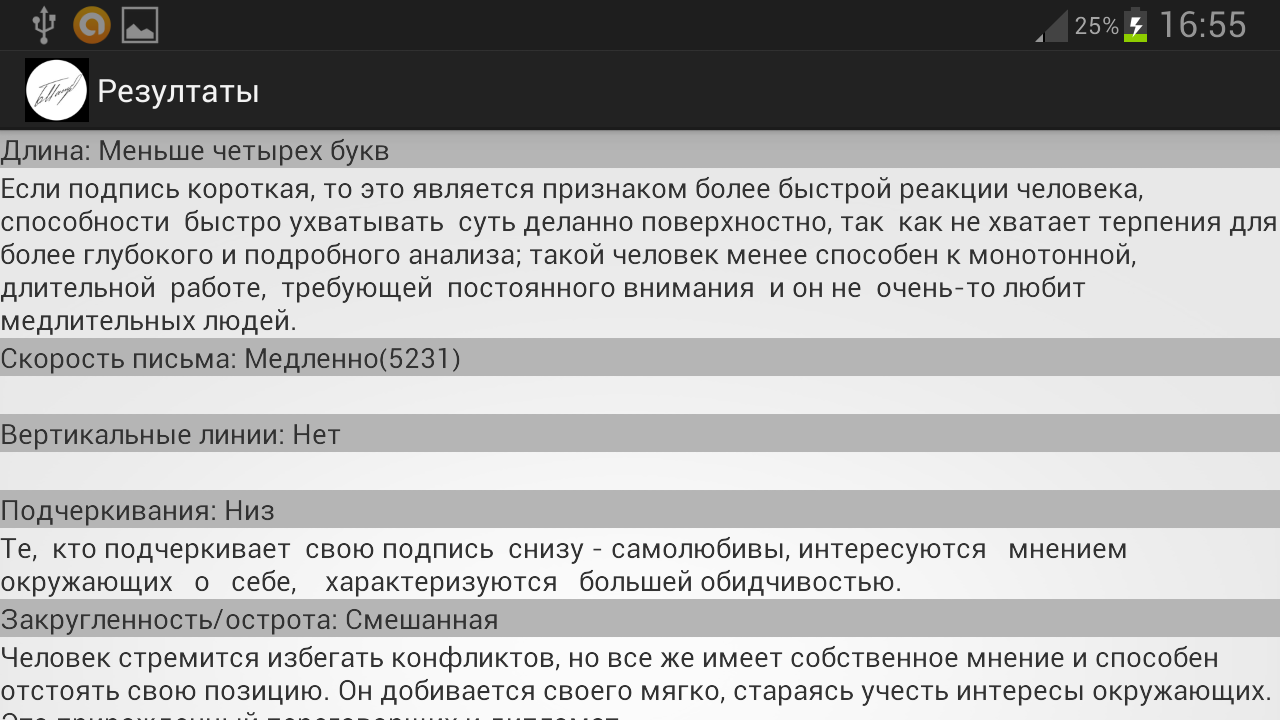
3) Ввод подписи пользователя.



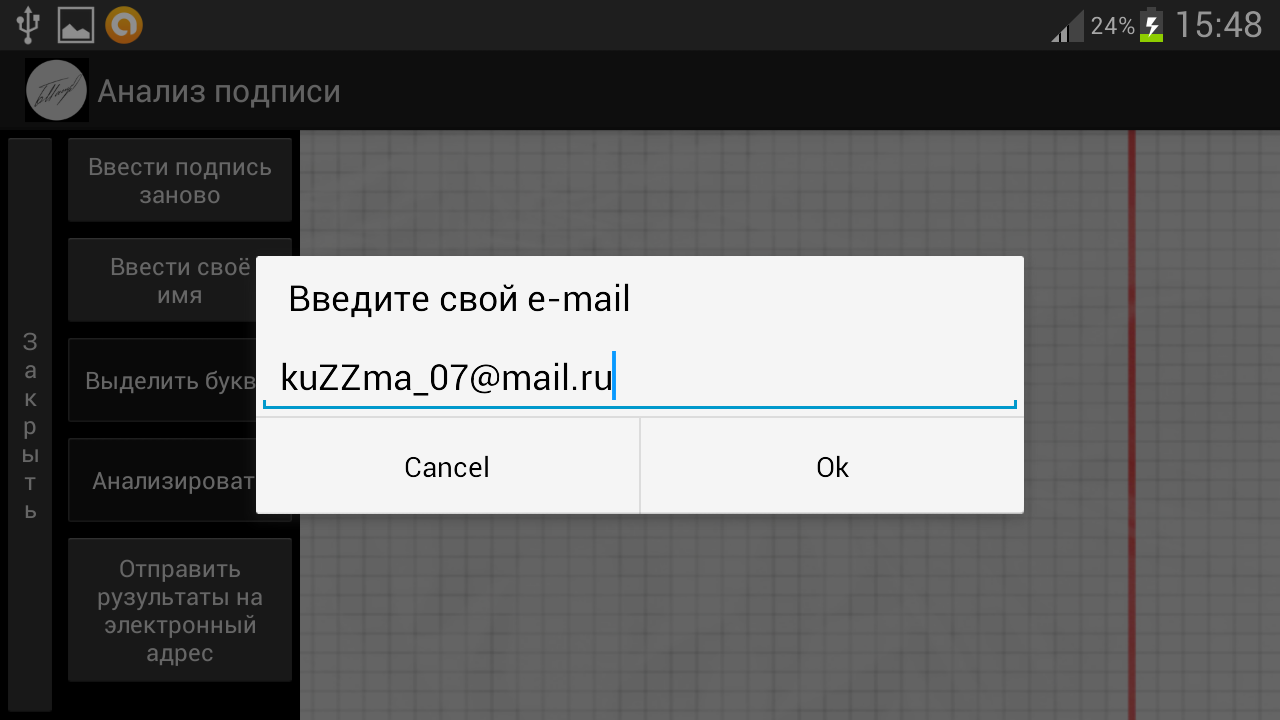
4) Выделение пользователем букв подписи.

5) Информирование о проведении процедуры анализа.

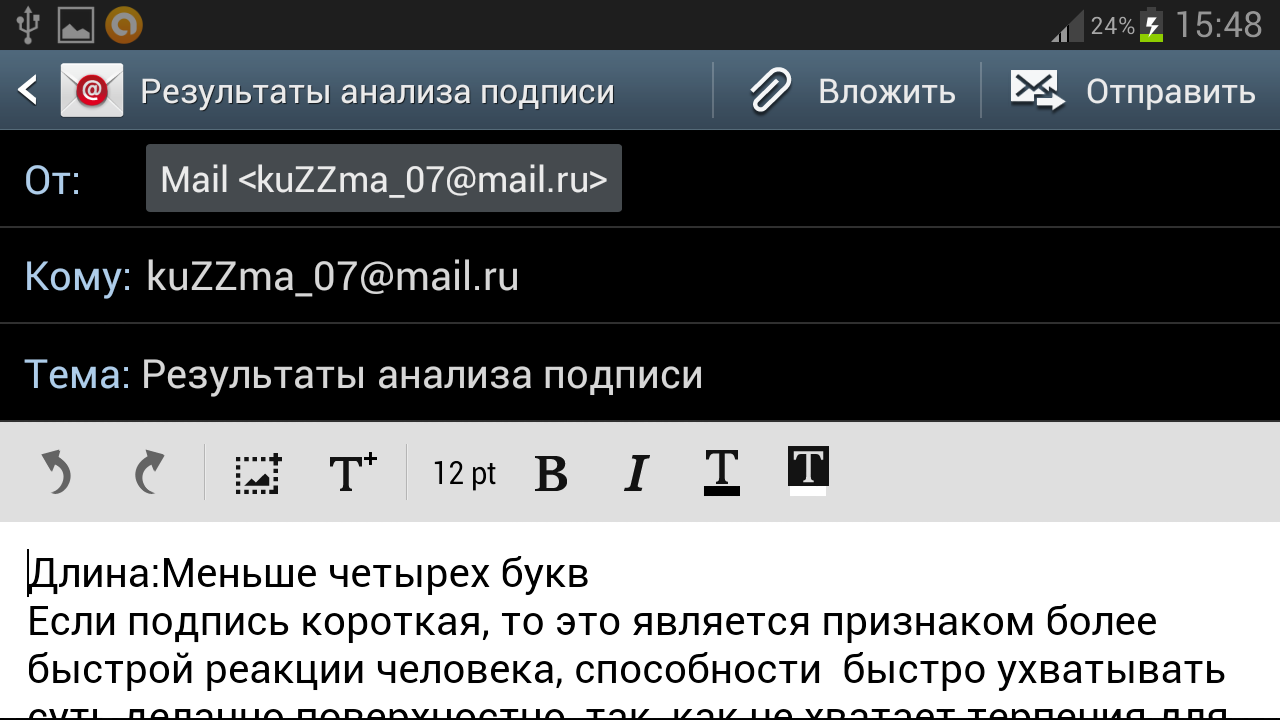


6) Отображение результатов анализа.

7) Ввод электронного адреса пользователя для отправки результатов.



8) Запуск почтового клиента для отправки результатов на электронный адрес пользователя.



Результаты анализа подписи:

**Длина: Меньше четырех букв;**

Если подпись короткая, то это является признаком более быстрой реакции человека, способности быстро ухватывать суть деланно поверхностно, так как не хватает терпения для более глубокого и подробного анализа; такой человек менее способен к монотонной, длительной работе, требующей постоянного внимания, и он не очень-то любит медлительных людей.

**Скорость письма: Медленно;**

**Вертикальные линии: Нет;**

**Подчеркивания: Низ;**

Те, кто подчеркивает свою подпись снизу - самолюбивы, интересуются мнением окружающих о себе, характеризуются большей обидчивостью.

**Закругленность/острота: Смешанная;**

Человек стремится избегать конфликтов, но все же имеет собственное мнение и способен отстоять свою позицию. Он добивается своего мягко, стараясь учесть интересы окружающих. Это прирожденный переговорщик и дипломат.

**Амплитуда: Увеличивается;**

Человек постепенно развивает активность от начала к концу деятельности.

**Направление: Прямо;**

Если конец подписи направлен прямо - то это свидетельствует о сбалансированности проявлений оптимизма и пессимизма . Немаловажную роль при этом играет влияние окружающей среды.

**Размашистость/компактность: Компактность;**

Подпись компактную, мелкую обычно имеют "тактики", то есть те, чей ум занят конкретными, какими видами деятельности.

**Точки: Есть;**

Точка является положительным признаком. Ее наличие в подписи свидетельствует о дисциплинированности и склонности к завершению намеченного.

**Наклон: Правый;**

Часто встречается правонаклонная подпись, что говорит о сбалансированности черт характера человека, о способности к пониманию, компромиссам. При этом, конечно, следует учитывать и другие факторы.

**Возвраты: Нет;**

**Связанность: Излишнее число разрывов**

Излишнее количество разрывов свидетельствует об образно-конкретном мышлении, непредсказуемости поступков для окружающих, мечтательности, стремлении обратить на себя внимание, произвести впечатление на окружающих.

**Нагруженность: Нет;**

Чем проще подпись - тем с меньшими проблемами живет человек - он стремится упростить жизнь.

**Величина первой буквы: Не отличается;**

Если заглавная буква по амплитуде незначительно отличается от малых букв, - то владелец подписи скромен, без особых претензий к жизни.

**Ровность: Есть;**

Если подпись написана ровно, как бы по линейке, то в этом случае это свидетельствует о сдержанности, самоконтроле, о рассудочном типе человека.

**Расстояние между буквами: Маленькое;**

Если же буквы в подписи налезают друг на друга, то это указывает на экономность и даже скупость.

**Количество заглавных букв в начале: Маленькое;**

Умеренная нагрузка первой части /минимум заглавных букв, их невысокая амплитуда говорит о склонности к деятельности прикладного характера. Часто люди, в подписи которых имеются указанные характеристики, по профессии являются служащими и общественными деятелями.

**Количество заглавных букв в конце: Маленькое;**

При наличии крупных образований во второй части подписи, привлекательной становится практическая деятельность.

**Петли: Маленькие;**

Человек скрытен и независим по характеру. Степень выраженности зависит от величины петель.

**Волнистость: Нет;**

**Иностранные буквы: Нет;**

**Цифры: Нет;**

**Темперамент: Меланхолик;**

Легко ранимый, склонный к постоянному переживанию различных событий, он остро реагирует на внешние факторы. Свои астенические переживания он зачастую не может сдерживать усилием воли, он повышено впечатлителен, эмоционально раним.

**Подходящая профессия: Человек – Человек;**

Предметом интереса, распознавания, обслуживания, преобразования здесь являются социальные системы, сообщества, группы населения, люди разного возраста. Примеры: продавец продовольственных товаров, парикмахер, инженер-организатор производства, врач, учитель.

# 3. Охрана труда

В процессе соей жизнедеятельности человек может попадать под воздействие различных опасных факторов. К опасным факторам относятся явления, процессы, объекты, которые в определенных условиях могут нанести ущерб здоровью человека непосредственно или косвенно.

Во время своей трудовой деятельности человек также подвергается воздействию опасностей. Пространство, в котором осуществляется трудовая деятельность, называется производственной средой.  В этой среде на человека в действуют техногенные опасности (связанные с техникой). Их принято называть опасными и вредными производственными факторами.[30]

Из трудового кодекса РФ от 30.12.2001 N 197-ФЗ, статья 209:

Охрана труда - система сохранения жизни и здоровья работников в процессе трудовой деятельности, включающая в себя правовые, социально-экономические, организационно-технические, санитарно-гигиенические, лечебно-профилактические, реабилитационные и иные мероприятия.

Условия труда - совокупность факторов производственной среды и трудового процесса, оказывающих влияние на работоспособность и здоровье работника.

Вредный производственный фактор - производственный фактор, воздействие которого на работника может привести к его заболеванию.

Опасный производственный фактор - производственный фактор, воздействие которого на работника может привести к его травме.

Безопасные условия труда - условия труда, при которых воздействие на работающих вредных и (или) опасных производственных факторов исключено либо уровни их воздействия не превышают установленных нормативов.

Средства индивидуальной и коллективной защиты работников - технические средства, используемые для предотвращения или уменьшения воздействия на работников вредных и (или) опасных производственных факторов, а также для защиты от загрязнения.

Производственная деятельность - совокупность действий работников с применением средств труда, необходимых для превращения ресурсов в готовую продукцию, включающих в себя производство и переработку различных видов сырья, строительство, оказание различных видов услуг.

Требования охраны труда - государственные нормативные требования охраны труда, в том числе стандарты безопасности труда, а также требования охраны труда, установленные правилами и инструкциями по охране труда.[29]

Можно выделить четыре класса вредности условий труда, исходя из степени отклонения фактических уровней факторов рабочей среды и трудового процесса от гигиенических нормативов условия труда по степени вредности и опасности[30]

1 класс: оптимальные условия труда

В таких условиях сохраняется здоровье работника, а также создаются предпосылки для поддержания высокого уровня работоспособности. Нормативы устанавливаются на трудовую нагрузку и микроклиматические параметры.

2 класс: допустимые условия труда

Для таких условий труда факторы среды и трудового процесса не должны превышать установленных гигиенических нормативов для рабочих мест. Возможные изменения функционального состояния организма восстанавливаются во время отдыха. К началу следующей смены они не оказывают неблагоприятного действия в ближайшем и отдаленном периоде на состояние здоровья работников и их потомство. Такие условия труда относят к безопасным.

3 класс: вредные условия труда

Такие условия характеризуются наличием вредных факторов. Уровни могут превышать гигиенические нормативы, а также оказывать неблагоприятное действие на организм и потомство человека.

Исходя из степени превышения гигиенических нормативов, для вредных условий труда можно выделить четыре степени вредности:

1  степень 3 класса. В таких условиях труда отклонения уровней вредных факторов от нормативов вызывают функциональные изменения. Восстановление осуществляется более длительным прерыванием контакта с вредными факторами. Также в таких условиях увеличивается риск повреждения здоровья;

2  степень 3 класса. В подобных условиях труда уровни вредных факторов, которые вызывают стойкие функциональные изменения, приводят к увеличению профессиональных заболеваемости, как в легкой форме, так и с потерей трудоспособности;

3  степень 3 класса. В таких условиях труда уровни факторов приводят к развитию профессиональных болезней с потерей трудоспособности, а также к росту хронической патологии;

4  степень 3 класса. В подобных условиях труда могут возникать тяжелые формы профессиональных заболеваний с потерей общей трудоспособности. Отмечается значительный рост числа хронических заболеваний и высокие уровни заболеваемости с временной утратой трудоспособности.

4 класс: опасные (экстремальные) условия труда

Такие условия труда отличаются уровнями факторов рабочей среды, создающих угрозу для жизни. Также они несут высокий риск развития острых профессиональных поражений.

Есть 2 группы нормативов безопасности:

* Предельно допустимы концентрации. Содержание вредных веществ биологической и химической природы в воздухе безопасно для организма человека.
* Предельно допустимые уровни. Они характеризуют воздействие различных опасных и вредных производственных факторов физической природы.

Психофизиологические производственные факторы могут характе­ризоваться параметрами трудовых нагрузок и/или по­казателями воздействия этих нагрузок на человека.

Для определения оценки условий труда производится анализ отдельных факторов производственной среды, которые включают:

* Параметры микроклимата;
* Содержание вредных веществ в воздухе;
* Уровень шума, вибраций, инфра- и ультразвука, освещенности.

## 3.1 Опасные факторы при работе с ЭВМ

При выполнении дипломной работы использовалась портативная ПВЭМ с дисплеем [SuperAMOLED](http://ru.wikipedia.org/wiki/Super_Active_Matrix_Organic_Light-Emitting_Diode).

Во время работы с ПЭВМ на оператора возможно воздействие следующих опасных и вредных факторов:

1. физических:

* низкочастотные электрические и магнитные поля;
* статическое электричество;
* лазерное и ультрафиолетовое излучение;
* повышенная температура;
* ионизация воздуха;
* опасное напряжение в электрической сети;

1. химических:

* пыль;
* вредные химические вещества, выделяемые при работе принтеров и копировальной техники;

1. психофизиологических:

* напряжение зрения и внимания;
* интеллектуальные и эмоциональные нагрузки;
* длительные статические нагрузки и монотонность труда.

**Освещенность**

Следует очень четко соблюдать требования по санитарным нормативам освещенности в административных, учебных и иных учреждения. Наше зрение напрямую зависит от количества света в помещении и на рабочем месте. От освещенности также зависит здоровье всего организма, сопротивляемость стрессам, усталости, физическим и умственным нагрузкам.

Вопрос освещенности рабочих мест, оборудованных персональными компьютерами (ПЭВМ) коротко, но очень четко изложен в СанПиН 2.2.2/2.4.1340-03 "Гигиенические требования к персональным электронно-вычислительным машинам и организации работы".

Данный документ содержит много интересных разделов, которые должен знать каждый офисный работник, поскольку санитарно-гигиеническое состояние рабочего места касается каждого из нас.

## 3.2. Требования к освещению на рабочих местах, оборудованных ЭВМ[31]

1. Необходимо, что естественный свет падал преимущественно слева. Для этого рабочие столы следует размещать так, чтобы видеодисплейные терминалы были направлены боковой стороной к световым проемам.

2. Система общего равномерного освещения должна создавать в помещениях для эксплуатации ЭВМ искусственное освещение. В производственных и административно-общественных помещениях следует применять системы комбинированного.

3. Освещенность на поверхности стола должна быть от 300 до 500 лк; создавать бликов на поверхности экрана. Освещенность поверхности экрана должна быть не менее 300 лк.

4. Необходимо ограничивать прямую блесткость, которая создается источниками освещения. Яркость светящихся поверхностей (окна, светильники и др.), находящихся в поле зрения, должна быть не более 200 кд/м2.

5. Блесткость на рабочих поверхностях должна ограничиваться правильным выбором светильников, а также их расположением рабочих мест по отношению к источникам естественного и искусственного освещения. Яркость бликов на экране ПЭВМ не должна превышать 40 кд/м2 и яркость потолка не должна превышать 200 кд/м2.

6. Показатель ослепленности для источников общего искусственного освещения в производственных помещениях должен быть менее 20.

Показатель дискомфорта в административно-общественных помещениях не более 40.

Показатель дискомфорта в дошкольных и учебных помещениях должен быть не более 15.

7. Яркость светильников в зонах, где угол излучения более 50 ˚ и менее 90˚, должна быть менее 200 кд/м. Защитный угол светильников должен быть более 40˚.

8. Светильники местного освещения должны оснащаться не просвечивающими отражателями. Защитный угол должен быть более 40 ˚.

9. Неравномерность распределения яркости в поле зрения пользователя должна быть ограничена. Соотношение яркости между рабочими поверхностями не должно превышать 3:1 - 5:1. Между рабочими поверхностями и поверхностями стен и оборудования 10:1.

10. Люминесцентные лампы типа ЛБ применяются для создания искусственного освещения. В этих же целях применяются компактные люминесцентные лампы. При устройстве отраженного освещения разрешается применение металлогалогенных ламп. Разрешается применение ламп накаливания, а также галогенных ламп, в светильниках местного освещения.

11. Лампы с зеркальными параболическими решетками, укомплектованные электронными пуско-регулирующими аппаратами (ЭПРА), применяются для освещения помещений с ЭВМ. Разрешается использование многоламповых светильников с электромагнитными пуско-регулирующими аппаратами (ЭПРА), состоящими из равного числа опережающих и отстающих ветвей.

12. Общее освещение при использовании люминесцентных ламп необходимо выполнять в виде сплошных или прерывистых линий ламп, которые должны быть расположены сбоку от рабочих мест, параллельно линии зрения пользователя при рядном расположении видеодисплейных терминалов. При периметральном расположении компьютеров линии ламп необходимо располагать локализовано над столом ближе к его переднему краю, обращенному к пользователю.

13. Коэффициент запаса для осветительных установок общего освещения следует принимать равным 1,4.

14. Коэффициент пульсации должен быть не более 5%.

15. Необходимо проводить чистку стекол оконных рам и светильников не реже двух раз в год и проводить своевременную замену перегоревших ламп.

Освещенность измеряется в Лк – люксах; 1 люмен/м. кв. (люмен Lm – единица величины светового потока).

## 3.3 Требования к качеству восприятия информации, отображаемой на дисплеях[32]

Должны быть созданы комфортные условия для восприятия информации и ее считывания с экрана. Для этого необходимо подобрать такие сочетания значений яркости и контраста изображения, внешней освещенности экрана, углового размера знака и угла наблюдения экрана, которые попадают в оптимальные или предельно допустимые диапазоны.

Неравномерность яркости рабочего поля экрана должна быть не более 20 %.

Расчет неравномерности яркости рабочего поля экрана:

Lцентр =600 кд/м2

Lлево-верх = 598 кд/м2

Lлево-низ = 599 кд/м2

Lправо-верх = 597 кд/м2

Lправо-низ = 598 кд/м2

= 598,4 кд/м2;

Неравномерность яркости рабочего поля экрана - ;

Выводы по расчету: Неравномерность яркости рабочего поля экрана равна и лежит в пределах нормы (не более 20%).

Неравномерность яркости элементов знака должна быть не более 20 %.

Расчет неравномерности яркости элементов знаков плоских дискретных экранов:

L’1 = 588 кд/м2

L’2 = 597 кд/м2

L’3 = 598 кд/м2

L’4 = 600 кд/м2

L’5 = 599 кд/м2

L’6 = 597 кд/м2

L’7 = 599 кд/м2

= 598,285 кд/м2;

Неравномерность яркости элементов знаков плоских дискретных экранов - ;

Выводы по расчету: Неравномерность яркости элементов знаков плоских дискретных экранов равна и лежит в пределах нормы (не более 20%).

Угол наблюдения экрана оператором относительно горизонтальной линии взгляда должен быть не более 60˚.

Внешняя освещенность экрана должна лежать в диапазоне от 100 лк до 250 лк.

Угловой размер знака должен лежать в диапазоне от 16 угл.мин до 60 угл.мин.

## 3.4. Эргономические требования к цветовым параметрам[32]

1. В случае необходимости распознавания или идентификации цветовых параметров программа должна предлагать устанавливаемый по умолчанию набор цветов, в котором не используются синий и красный цвет спектра на темном фоне и красный цвет спектра на синем фоне. Так как пользователь может изменять цвета во время пользования программой, то она должно уметь восстанавливать заданный по умолчанию набор цветов.

2. Для идентификации знаков, а также в полях ввода данных высота символа не должна быть менее 20¢ (при проектном расстоянии наблюдения).

3. Для точной идентификации цвета изображения угловой размер изображения не должен быть менее 30¢ (при проектном расстоянии наблюдения).

4. Для обеспечения читабельности текстов, буквенно-цифровых знаков и символов изображения запрещается применять синий цвет спектра на красном фоне.

5. Запрещается применение крайних цветов видимого спектра для изображений, требующих непрерывного просмотра или чтения, так как они могут приводить к нежелательным эффектам глубины изображаемого пространства.

6. Распознавание и идентификация цветов требуют применения цветных изображений переднего плана на ахроматическом фоне или ахроматических изображений переднего плана на цветном фоне.

7. В каждый заданный по умолчанию набор цветов должны входить не более одиннадцати цветов. Это способствует более точной идентификации цвета.

8. Быстрый поиск, связанный с распознаванием цветов требует применения максимум шесть различных цветов.

9. Вызов параметров цветов из памяти ЭВМ требует применения максимум шести различных цветов.

## 3.5. Требования безопасности к визуальным параметрам[32]

1. Для плоских дискретных экранов необходимо установить:

* Значение яркости знака более 20 кд/м2;
* Значение яркостного контраста изображения при угле наблюдения в диапазоне от минус 40° до плюс 40° более 3:1;
* Значение яркостного контраста знака более 3:1;
* Значение обновления частоты изображения более 60 Гц;

2. Ширина контура знака должна не менее 0,25 и не более 0,5 мм.

3. Амплитуда смещения изображения должна быть менее 2×10-4l, где l - проектное расстояние наблюдения, мм.

## 3.6. Искажения изображения по рабочему полю[32]

1. Высота однотипных знаков должна лежать в диапазоне от минус 5 до плюс 5%.

2. Для определения норматива по максимальной разности длин строк текста необходимо вычислить среднюю длину. Затем определить максимальную разность. Она должна быть менее 2 % средней длины.

3. Для определения норматива по максимальной разности длин столбцов текста необходимо вычислить среднюю длину. Затем определить максимальную разность. Она должна быть менее 2 % средней длины.

4. Отклонение формы рабочего поля от прямоугольника определяется по следующим формулам:

по вертикали

по горизонтали

по диагонали

где Н1, H2, B1, B2, D1 и D2 - значения длин крайнего левого и крайнего правого столбца, верхней, нижней строки и диагоналей на рабочем поле соответственно, мм.

Крайний левый столбец (H1) – 1080 мм;

Крайний правый столбец (H2) – 1080 мм;

Верхняя строка (B1) – 620 мм;

Нижняя строка (B2) – 620 мм;

Диагонали (D1, D2) – 1240 мм;

По вертикали:

По горизонтали:

По диагонали:

;

Выводы по расчету: отклонения формы рабочего поля от прямоугольника не наблюдается.

## 3.7. Требования безопасности к параметрам создаваемых полей[32]

1. Напряженность электрической составляющей переменного электромагнитного поля дисплея должна быть не более:

25 В/м - в диапазоне частот от 5 Гц до 2 кГц. Для дисплеев портативных компьютеров должно достигаться в точке, расположенной по нормали к центру экрана на расстоянии 0,4 м от центра клавиатуры портативного компьютера;

2,5 В/м - в диапазоне частот от 2 до 400 кГц. Для дисплеев портативных компьютеров это значение должно достигаться в точках с координама 0°, 90°, 180°, 270° на расстоянии r = a/2 + 0,4 м, где а - габаритный размер дисплея, измеряемый по нормали к центру экрана.

2. Плотность магнитного потока должна быть не более:

250 нТл - в диапазоне частот от 5 Гц до 2 кГц;

25 нТл - в диапазоне частот от 2 до 400 кГц.

Плотность магнитного потока переменного ЭМП дисплея портативного компьютера устанавливаются для обоих диапазонов частот в 48 точках через 22°30¢ от нормали к центру экрана, на расстоянии r = a/2 + 0,4 м, где а - габаритный размер дисплея по нормали к центру экрана.. Часть из них находятся в горизонтальной плоскости, проходящей через центр экрана. Другая часть находится в горизонтальных плоскостях, которые расположены на 0,3 м выше и ниже указанной плоскости.

## 3.8. Требования к конструкции

1. Должно обеспечиваться возможность фронтального наблюдения изображения на экране. Для это корпус дисплея должен вращаться вокруг вертикальной оси на ±30°. Также он должен вращаться вокруг горизонтальной оси в пределах от плюс 30° до минус 15° с фиксированием дисплея в заданном положении.

2. Цвет корпуса дисплея должен иметь мягкие тона, а также матовую поверхностью и не содержать блестящих деталей, которые способны создавать блики.

3. Органы управления не должны быть расположены на лицевой стороне корпуса. В случае расположения органов управления на лицевой панели, их следует закрыть крышкой или утопить в корпусе.

4. Необходимо, чтобы конструкция дисплея предоставляла возможность регулирования яркости и контраста.

5. Необходимо, чтобы конструкция дисплея предоставляла возможность снижения уровней электростатического и электромагнитного полей.

## 3.9. Выводы

В главе перечислены опасные и вредные факторы, которые могут возникнуть при выполнении дипломной работы, причины их возникновения и меры по предотвращению или снижению их негативного воздействия на работника. Произведен расчет неравномерности яркости рабочего поля экрана и неравномерности яркости элементов знаков плоских дискретных экранов. Показания расчетов не выходят за пределы установленных санитарных норм. Также произведена оценка отклонения формы рабочего поля от прямоугольника. Отклонения не выявлено.

# 4. Экологическая часть

## 4.1. Влияние освещения на организм человека

Свет помогает обеспечивать нормальную жизнедеятельность человека. За счет света определяются его жизненный тонус и биоритмы. Длины волны, интенсивности и количества излучения являются определяющими факторам силы его воздействия. Видимая часть спектра обеспечивает нормальную работу зрительного анализатора, а также регулирует биоритмы человека. Длительное световое голодание может привести к ослаблению иммунобиологической реактивности организма, а также обернуться функциональными нарушениями нервной системы. Свет осуществляет воздействие на психику и эмоциональное состояние человека. Неблагоприятные условия освещения ведут к снижению работоспособности и возникновению заболеваний органов зрения.

Условия труда в большой степени зависят от освещенности на рабочих местах и в помещениях. Неправильное освещение может привести к травматизму. Плохо освещенные опасные зоны, слепящие лампы, резкие тени могут ухудшать или вызывать полную потерю зрения.

Неправильная эксплуатация осветительных установок в пожароопасных цехах может привести к взрыву, пожару и несчастным случаям.

Обычно пользуются естественными, искусственным и совмещенным освещением.

Естественное освещение более предпочтительным. Солнечный свет наиболее благоприятен для человека. Спектр солнечного излучения разделяется на видимую и невидимую части. К невидимой относятся ультрафиолетовое и инфракрасное излучение. Ультрафиолетовое излучение оказывает положительное воздействие на организм человека, а также является источником эритемного эффекта (загара). Однако, при большой интенсивности оно может вызывать ожоги кожи, ожоги сетчатки глаз. Ультрафиолетовое излучение возникает при работе кварцевых ламп, электрической дуги, лазерных установок, электро- и газовой сварки, эритемных ламп.

Защитой от УФ может являться ткань одежды, очки.

Инфракрасное излучение - это тепловое излучение.

Видимое излучение при больших яркостях может приводить к ослепленности и снижению остроты зрения.

Согласно санитарным нормам все помещения с постоянным пребыванием людей должны иметь естественное освещение.

Виды естественного освещения:

* боковое (световые проемы в наружных стенах);
* верхнее (световые проемы (фонари) в покрытиях, проемы в стенах в местах перепада высот зданий);
* комбинированное (сочетание верхнего и бокового).

Искусственное освещение осуществляется в темное время суток. Источниками искусственного освещения являются два типа ламп: лампы накаливания и люминесцентные лампы. Последние имеют ряд преимуществ перед лампами накаливания:

* Их спектр близок к естественному. Это создает оптимальные условия для зрительной работы;
* Они обладают меньшей яркостью и не дают резких теней;
* Они не повышают температуру в помещении;
* Они более экономичны при равном уровне освещенности.

Однако, люминесцентным лампам присущи два недостатка – это высокая, до 35-65% глубины пульсации (глубина пульсации ламп накаливания - 5-15%), которая создает эффект стробоскопа, и шумовой эффект.

Эффект стробоскопа связан с незаметными для глаза пульсациями. Он проявляется при рассматривании движущегося предмета, когда возникает искажение зрительного восприятия в виде множест­венности контуров объекта наблюдения. Кажется, что объект изменил направление и скорость движения. Поэтому люминесцентные лампы не должны быть установлены в местах, где требуется следить за быстро перемещающимся предметом (игровые и спортивные залы, теннисные корты, площадки для спортивных игр и т. д.). Пульсации вызывают заметное зрительное утомление и ухудшают функциональное состояние центральной нервной системы.

Для предотвращения эффекта стробоскопа люминесцентные необходимо включать лампы в разные фазы и применять схему с искусственным сдвигом фаз.

Шумовой эффект люминесцентных ламп оказывает негативное воздействие на центральную нервную систему. Он вызывает повышенное возбуждение нервных клеток и торможение. Для его устранения используются специальные бесшумные пускорегулирующие агрегаты (ПРА).

Недостатки люминесцентных ламп могут быть устранены правильным монтажом.

**Общее** освещение - это освещение для создания минимально необходимой освещенности в производственном помещении.

Оно может быть как равномерным (при симметричном расположении светильников), так и усиленным на отдельных участках производственного помещения за счет локализованного расположения светильников Общее Просвещение итлення применяют в помещениях, где условия работы требуют освещенности не более 50лк и где применение местного освещения связано с техническими трудностями.

**Комбинированное** освещение применяется для создания достаточно высоких уровней освещенности на рабочих поверхностях благодаря одновременному использованию систем общего и местного освещения

**Местное** освещение подразделяется на стационарное и переносное Использование только местного освещения при условии промышленных предприятий не допускается вследствие того, что большая разница в освещенности рабочего места ц и окружающей среды приводит к возникновению несчастных случаев и снижению производительности трудаі.

**Переносное местное** освещения разрешается только при проведении разовых и периодических работ

## 4.2. Влияние электромагнитного излучения на организм человека

Источники электромагнитных излучений, к которым относятся воздушные линии электропередачи высокого и сверхвысокого напряжения, технические средства радиовещания, телевидения, радиорелейной и спутниковой связи, радиолокационные и навигационные системы, лазерные маяки и др., существенно повлияли на естественный электромагнитный фон. На значительных территориях, особенно вблизи прохождения воздушных линий электропередач высокого и сверхвысокого напряжения, радио- и телецентров, радиолокационных установок, напряженность электрических и магнитных нолей возросла от двух до пяти порядков, создавая реальную опасность для людей, животного и растительного мира.

Целенаправленное использование электромагнитной (ЭМ) энергии в самых разнообразных областях человеческой деятельности привело к тому, что к существующему электрическому и магнитному полям Земли, атмосферному электричеству, радиоизлучению Солнца и Галактики добавилось электромагнитное поле искусственного происхождения. Его уровень значительно [превышает уровень естественного ЭМ фона](http://airestech.ru/media). Энергоресурс мира удваивается каждые десять лет, а удельный вес переменных электромагнитного поля (ЭМП) в электроэнергетике за это время возрастает еще в три раза.

[Биологически значимыми](http://airestech.ru/technologies/printsip-deystviya/kak-proverit-effekt) являются техногенные электрические поля частотой 50 Гц, создаваемые воздушными линиями и подстанциями, На­пряженность магнитных полей промышленной частоты в местах размещения воздушных линий и подстанций сверхвысокого напряжения на 1-3 порядка превышает естественные уровни магнитного поля Земли. Высокие уровни электромагнитных излучений (ЭМИ) наблюдаются на тер­риториях, а нередко и за пределами разме­щения передающих радиоцентров низкой, средней и высокой частоты.

ЭМП ультранизкой (0-10 Гц) и низ­кой частоты (10-1000 Гц) создаются в процессе эксплуатации электрифицированного го­родского и железнодорожного транспорта, линиями электропередач, подстанциями и кабельными трассами.

[Широко распространенными источниками ЭМИ](http://airestech.ru/media) в населенных местах являются радиотелевизионные передающие центры, излуча­ющие в окружающую среду ультракороткие волны особо высокочастотных и ультравысо­кочастотных диапазонов. Причем наибольшие уровни облучения людей и воздействия на окружающую среду наблюдаются в районе размещения радио- и телепередающих центров «старой постройки» с высотой антенной опоры не более 180 м. Наибольший вклад в суммарную интенсивность воздействия вносят «уголковые» трех- и шестиэтажные антенны очень высокой частоты частотно-модулированного (ЧМ, а в международной терминологии FM) вещания.

Установлено также влияние ЭМИ на организм человека от бытовых электроприборов, которое может быть достаточно высоким. Например, на расстоянии 3 см магнитная индукция при работе фена равна 2 тыс. мкТл, электробритвы — 1,5 тыс. мкТл (сравните: естественный геомагнитный фон составляет 30-61 мкТл). При массовом распространении радиотелефон, прикладываемый время от времени к виску, как излучатель волн дециметрового диапазона с большой проникающей способностью, представляет опасность не только для индивидуальных пользователей, но и для всех окружающих. С 1970-х гг. производится несколько миллионов микроволновых печей, в которых используется энергия сверхвысокочастотного (СВЧ) электромагнитного излучения.

ЭМИ сильно влияет на иммунную, нервную, эндокринную и половую систему.

В результате иммунная система может уменьшать выброс в кровь специальных ферментов, которые выполняют защитную функцию. Ослабляется система клеточного иммунитета.

Эндокринная система осуществляет выброс в кровь большого количества адреналина. Происходит возрастание нагрузки на сердечно-сосудистую систему организма. Кровь сгущается, что приводит к тому, что клеткам не хватает кислорода.

Если человек долгое время подергается воздействию электромагнитного излучения, то у него может уменьшаться сексуальное влечение (это может быть вызвано изменениями в деятельности эндокринной системы), падает потенция.

Как уже отмечалось выше, признаками расстройства являются раздражительность, быстрая утомляемость, ослабление памяти, нарушение сна, общая напряженность, люди становятся суетливыми.

Таковы последствия воздействия электромагнитного излучения.

## 4.3. Выводы

В главе проанализировано воздействие освещения и электромагнитного излучения на организм человека, а также опасные и вредные факторы, которые могут оказать негативное влияние.

**Рациональное освещение производственных помещений и** рабочих мест способствует лучшему выполнению работающим своих обязанностей обеспечению комфортных условий труда. В нормальных документах по ОТ сформулированы основные требования к производственному освещению: достаточная освещенность рабочих поверхностей, равномерное распределение яркости, отсутствие резких теней, спектр светового потока должен быть близок к естественному, постоянство освещенности во времени.

В качестве защитных мер от негативного воздействия электромагнитного излучения можно назвать:

1. регулярные прогулки на свежем воздухе;
2. проветривание помещения, занятия спортом;
3. соблюдение элементарных правил работы;

работа с хорошей техникой, которая удовлетворяет всем стандартам безопасности и санитарным нормам.

# Заключение

В данной работе была разработана система автоматизированного графологического анализа подписи. Данная система позволяет снизить трудоемкость процесса графологического анализа подписи.

Система использует возможности сенсорного экрана, в результате чего работает непосредственно с оригиналом подписи.

Все характеристики подписи анализируются независимо друг от друга, и каждой характеристике соответствует одно единственное значение характеристики личности.

Данная система может успешно применяться в различных направлениях диагностики и психологического анализа личности.

# Список используемой литературы:

[1] Щеглов И. В., Чернов Ю. Г. «Графология XXI века», СПб.: Питер, 2008.

[2] Фармагей А. И., «Графология как наука и не только», Киев: Ника-Центр, 2007

[3] Чернов Ю. Г. «Психологический анализ почерка. Системный подход и компьютерная реализация в психологии», М.: Генезис, 2011

[4] Siew Hock Ow, Kean Siang Teh « An Overview of the Handwriting Analysis Systems in the Market»

[5] Чернов Ю. Г. «Анализ почерка в работе с кадрами»

[6] Handwriting University. 2004. Handwriting Wizard. Free Handwriting Self-Test. [Online]. Available: http://www.handwritingwizard.com/index.html [20 June 2004].

[7] HWA.ORG. 2004. Handwriting Analysis. The Original Web Handwriting Site. [Online]. Available: http://www.hwa.org [20 June 2004].

[8] Michaels, G., M. Maze, and D. Hodos. 2004. Handwriting Analyst. Graphology Software for Personality Profiling. [Online]. Available: http://www.garthmichaels.com/ [22 June 2004].

[9] Quantum Enterprises. 2004. Handwriting-The Inner Secrets Revealed. [Online]. Available: http://www.quantumenterprises.co.uk/handwriting/index.htm [20 June 2004].

[10] RI Software Inc. 2004. Sheila Lowe & Associates. Handwriting Analysis Software for Windows. [Online]. Available: http://www.writinganalysis.com [21 June 2004].

[11] Кравченко В. И. «Графология: характер по почерку», уч-метод. Пособие, 2006.

[12] Ойбек Наджимов «Как узнать характер человека по его подписи», http://lib.ru/URIKOVA/GRAFOLOG/aybek/aybek.txt

[13] Джеймс Раскин «Графологический анализ подписи»

[14] Линник Ю. В. «Метод наименьших квадратов и основы математико-статистической теории обработки наблюдений», 2-е изд. — М., 1962.

[15] Eric Haines. Point in Polygon Strategies, http://erich.realtimerendering.com/ptinpoly/

[16] Прасолов В. В. «Задачи по планиметрии», 4-е изд., дополненное — М.: Изд-во Московского центра непрерывного математического образования, 2001.

[17] Тараненко В. «Почерк, портрет, характер. Скрытая психодиагностика в практическом изложении. – Киев: Ника-Центр, 2008

[18] Юрий Кисляков, Реализация простейшего алгоритма распознавания графических образов, http://www.delphikingdom.com/asp/viewitem.asp?catalogid=1203

[19] Григорьева М.В. «Психология труда», М.: Высшее образование, 2006.

[20] Орлик С., «Модели жизненного цикла», http://swebok.sorlik.ru/software\_lifecycle\_models.html

[21] ГОСТ 34601-90, «Автоматизированные системы. Стадии создания»

[22] Альфред В. Ахо, Моника С. Лам, Рави Сети, Джеффри Д. Ульман. «Компиляторы: принципы, технологии и инструментарий», М.: Вильямс, 2010

[23] Кузнецов С. Д. «Основы баз данных», 2-е изд. — М.: Интернет-университет информационных технологий; БИНОМ. Лаборатория знаний, 2007

[24] http://www.oracle.com/us/sun/index.htm

[25] http://eclipse.org/

[26] <http://developer.android.com/tools/sdk/eclipse-adt.html>

[27] Гольдберг И., «Психология почерка», Екатеринбург: У\_ФАКТОРИЯ; М.: АСТ, 2008.

[28] <http://grafologia.by/informaciya/sekreti-grafologii/effektivnost.html>

[29] Трудовой кодекс РФ (ТК РФ) от 30.12.2001 N 197-ФЗ.

[30] Бобкова О. В. «Охрана труда и техника безопасности. Обеспечение прав работника», М.: Омега-Л, 2008

[31] СанПиН 2.2.2/2.4.1340-03 «Гигиенические требования к персональным электронно-вычислительным машинам и организации труда».

[32] ГОСТ 50948 «Средства отображения информации индивидуального пользования».