Идея проекта

# Название программного продукта

Система обучения использованию библиотек работы с сетевым трафиком

# Концепция программного продукта

Программный продукт представляет собой систему, предназначенную для самостоятельного освоения работы с сетевым трафиком и получения сопутствующих навыков. Пользователи заходят на сайт системы, после чего получают возможность

1. Иметь постоянный доступ к справочным системам и систематизированным каталогам ссылок на источники.
2. Обсуждать на форуме вопросы на темы, по которым ведется обучение.
3. Задавать вопросы в службу информационной поддержки.
4. Просматривать видео-уроки и примеры кода.
5. Систематизировано хранить свои наработки.
6. Выкладывать свои учебные материалы и делиться опытом.

# Сроки и бюджет

Проект должен быть реализован в течение 11 месяцев.

Бюджет проекта: 1,5 млн. руб.

# Требования

Понятность и удобство использования, возможность быстрого самостоятельного освоения.

# ОРГАНИЗАЦИЯ ВЫПОЛНЕНИЯ

# Исполнители и подбор персонала

Для выполнения проекта будет сформирована команда, включающая

* Руководителя проекта.
* Разработчика интерфейса программы (дизайнера).
* 3-х ведущих разработчиков.
* 2-менеджеров по поиску и подготовке контента.
* Тестировщика.

Подбор персонала производится руководителем проекта из имеющихся трудовых ресурсов с учетом возможности привлечения временных сотрудников.

# АНАЛИЗ РИСКОВ

Возможные риски:

* Несоответствие продукта представлениям заказчика.

Вероятность риска: Низкая (10-25%) – 1 балл

Степень влияния: Высокая – 3 балла

Ранг риска: 3 – Средний

Возможные последствия: отказ заказчика от продукта, необходимость переделывать работу.

Стратегия реагирования: уклонение от риска

План действий: подробное обсуждение с заказчиком требований к продукту, фиксация требований и согласование с заказчиком в письменном виде, демонстрация заказчику прототипов и промежуточных версий для своевременной корректировки.

В случае реализации риска: корректировка продукта, если это возможно.

Ответственные: руководитель проекта, все разработчики

* Непредвиденные обстоятельства, связанные с сотрудниками (заболевание, увольнение)

Вероятность риска: Средняя (25-50%) – 2 балла

Степень влияния: Катастрофическая – 3 балла

Ранг риска: 6 – Высокий

Возможные последствия: невыполнение работы в срок, ухудшение качества продукта

Стратегия реагирования: снижение вероятности риска, принятие

План действий: горячий чай с медом, лимоном и печеньками.

В случае реализации риска: привлечение свободных разработчиков с других проектов либо привлечение сторонних разработчиков.

Ответственные: руководитель проекта, все разработчики.

* Изменение заказчиком требований к продукту или сроков.

Вероятность риска: Низкая (10-25%) – 1 балл

Степень влияния: Высокая – 3 балла

Ранг риска: 3 – Средний

Возможные последствия: невыполнение работы в срок, ухудшение качества продукта, невозможность удовлетворить требования заказчика.

Стратегия реагирования: снижение вероятности риска, передача риска

План действий: разъяснение заказчику, что изменение им требований приведет к снижению качества продукта, увеличению стоимости и сроков выполнения, заключение соглашения о выплате неустойки в таком случае.

В случае реализации риска: привлечение свободных разработчиков с других проектов либо привлечение сторонних разработчиков, вложение дополнительных средств, пересмотр условий договора с заказчиком.

Ответственные: руководитель проекта.

* Отказ заказчика от проекта.

Вероятность риска: Низкая (10-25%) – 1 балл

Степень влияния: Катастрофическая – 3 балла

Ранг риска: 3 – Средний

Возможные последствия: провал проекта, .

Стратегия реагирования: передача риска, принятие

План действий: заключение соглашения с заказчиком о выплате неустойки в случае отказа от проекта.

В случае реализации риска: изучение рынка, поиск потенциального заказчика на данного рода продукт.

Ответственные: руководитель проекта.

* Несоответствие выбранных для разработки продуктов поставленным задачам.

Вероятность риска: Низкая (10-25%) – 1 балл

Степень влияния: Катастрофическая – 3 балла

Ранг риска: 3 – Средний

Возможные последствия: невыполнение работы в срок, невозможность удовлетворить требования заказчика, необходимость начинать разработку сначала.

Стратегия реагирования: снижение вероятности риска

План действий: подробное изучения средств, которые будут использоваться; консультации с более узкими специалистами для подбора наиболее подходящих средств.

В случае реализации риска: анализ текущего состояния, поиск новых решений.

Ответственные: руководитель проекта, разработчики.

# НЕОБХОДИМЫЕ РЕСУРСЫ

Аппаратные ресурсы:

10 компьютеров (по одному для каждого разработчика + 2 для проверки и демонстрации работы).  
Принтер и презентационное оборудование.

Программные ресурсы:

* Среды разработки на ЯП С++ (VS, Eclipse, codeblocks) – 10 шт.
* Библиотеки для работы с сетевым трафиком (libpcap, winpcap, netfilter и т.д.)
* Операционные системы (Windows, Linux) – 10 шт, в том числе серверные версии – 2 шт.
* …

# ОСНОВНЫЕ ЭТАПЫ И ВЕХИ

Этапы:

* Этап обсуждения и согласования с заказчиком
* Этап планирования и распределения работ
* Этап проектирования
* Этап разработки
* Этап тестирования и корректировки
* Сдача проекта

Вехи:

* Получены все спецификации от заказчика, согласованы сроки и бюджет – конец первого месяца.
* Составлен план работ – первая неделя второго месяца.
* Распределены обязанности работников – вторая неделя второго месяца.
* Проведена оценка рисков – третья неделя второго месяца.
* Написан каркас серверной части программы – середина четвертого месяца.
* Написан каркас клиентской части программы - середина четвертого месяца.
* //Ряд вех, связанных с промежуточными частями кода.
* Написан код программного продукта – конец девятого месяца.
* Протестирован код программного продукта – первая неделя десятого месяца.
* Программный продукт проходит все тесты – середина одиннадцатого месяца.
* Продукт принят клиентом и начал работать – конец одиннадцатого месяца.

ГРАФИК РАБОТ   
Возможно наложение этапов

* Этап обсуждения и согласования с заказчиком – 1 месяц (руководитель проекта, а также остальные разработчики на части собраний)
* Этап планирования и распределения работ – 1,5 месяца (руководитель проекта и частично остальные разработчики для оценки объема работ)
* Этап проектирования – 1 месяц (все, кроме тестировщика)
* Этап разработки - 6 месяцев (все человеческие ресурсы)
* Этап тестирования и корректировки – 3,5 месяца (все человеческие ресурсы)
* Сдача проекта – 1 неделя

# План-график работ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Наименование** | **Продолжительность** | **Начало** | **Окончание** | **Предшествующие** |
| 1 | Этап обсуждения и согласования с заказчиком | 21 дней? | 10.12.2012 8:00 | 07.01.2013 17:00 |  |
| 2 | обсуждение с заказчиком | 3 дней? | 10.12.2012 8:00 | 12.12.2012 17:00 |  |
| 3 | формулировка целей | 2 дней? | 13.12.2012 8:00 | 14.12.2012 17:00 | 2 |
| 4 | сбор требований | 5 дней? | 13.12.2012 8:00 | 19.12.2012 17:00 | 2 |
| 5 | предварительное планирование и разработка структуры проекта | 7 дней? | 20.12.2012 8:00 | 28.12.2012 17:00 | 4;3 |
| 6 | согласование плана, сроков, бюджета с заказчиком | 5 дней? | 31.12.2012 8:00 | 04.01.2013 17:00 | 5;9;10 |
| 7 | подготовка договора | 5 дней? | 01.01.2013 8:00 | 07.01.2013 17:00 | 5 |
| 8 | Этап планирования и распределения работ | 29 дней? | 20.12.2012 8:00 | 29.01.2013 17:00 |  |
| 9 | обсуждение в команде | 7 дней? | 20.12.2012 8:00 | 28.12.2012 17:00 | 4;3 |
| 10 | выбор средств и ресурсов | 7 дней? | 20.12.2012 8:00 | 28.12.2012 17:00 | 3;4 |
| 11 | планирование работ | 9 дней? | 08.01.2013 8:00 | 18.01.2013 17:00 | 6;7 |
| 12 | распределение задач | 7 дней? | 10.01.2013 8:00 | 18.01.2013 17:00 | 7;6 |
| 13 | работа с рисками | 7 дней? | 21.01.2013 8:00 | 29.01.2013 17:00 | 12;11 |
| 14 | Этап проектирования | 22 дней? | 30.01.2013 8:00 | 28.02.2013 17:00 |  |
| 15 | построение архитектуры проекта | 6 дней? | 30.01.2013 8:00 | 06.02.2013 17:00 | 13 |
| 16 | разработка прототипа и его тестирование | 10 дней? | 07.02.2013 8:00 | 20.02.2013 17:00 | 15 |
| 17 | анализ результатов и корректировка архитектуры | 6 дней? | 21.02.2013 8:00 | 28.02.2013 17:00 | 16 |
| 18 | документация проекта | 10 дней? | 11.02.2013 8:00 | 22.02.2013 17:00 | 15 |
| 19 | Этап разработки | 135 дней? | 25.02.2013 8:00 | 30.08.2013 17:00 |  |
| 20 | написание кода отдельных модулей | 43 дней? | 01.03.2013 8:00 | 30.04.2013 17:00 | 17;18 |
| 21 | реализация взаимодействия модулей | 43 дней? | 01.05.2013 8:00 | 28.06.2013 17:00 | 20 |
| 22 | реализация интерфейса | 43 дней? | 01.06.2013 8:00 | 31.07.2013 17:00 | 20 |
| 23 | анализ и корректировка | 45 дней? | 01.07.2013 8:00 | 30.08.2013 17:00 | 20 |
| 24 | написание пользовательской документации | 114 дней? | 25.02.2013 8:00 | 01.08.2013 17:00 | 18 |
| 25 | Этап тестирования и корректировки | 84 дней? | 01.07.2013 8:00 | 24.10.2013 17:00 |  |
| 26 | написание тестов | 34 дней? | 01.07.2013 8:00 | 15.08.2013 17:00 | 21 |
| 27 | тестирование | 32 дней? | 02.09.2013 8:00 | 15.10.2013 17:00 | 26;23;24;22 |
| 28 | выявление и исправление ошибок | 33 дней? | 02.09.2013 8:00 | 16.10.2013 17:00 | 26 |
| 29 | демонстрация заказчику | 4 дней? | 21.10.2013 8:00 | 24.10.2013 17:00 | 27;28 |
| 30 | Сдача проекта | 5 дней? | 25.10.2013 8:00 | 31.10.2013 17:00 | 29 |

# Структурная декомпозиция работ по этапам

**Этап обсуждения и согласования с заказчиком**

обсуждение с заказчиком

формулировка целей

сбор требований

предварительное планирование и разработка структуры проекта

согласование плана, сроков, бюджета с заказчиком

подготовка договора

**Этап планирования и распределения работ**

обсуждение в команде

выбор средств и ресурсов

планирование работ

распределение задач

работа с рисками

**Этап проектирования**

построение архитектуры проекта

разработка прототипа и его тестирование

анализ результатов и корректоровка архитектуры

документация проекта

**Этап разработки**

написание кода отдельных модулей

реализация взаимодействия модулей

реализация интерфейса

анализ и корректировка

написание пользовательской документации

**Этап тестирования и корректировки**

написание тестов

тестирование

выявление и исправление ошибок

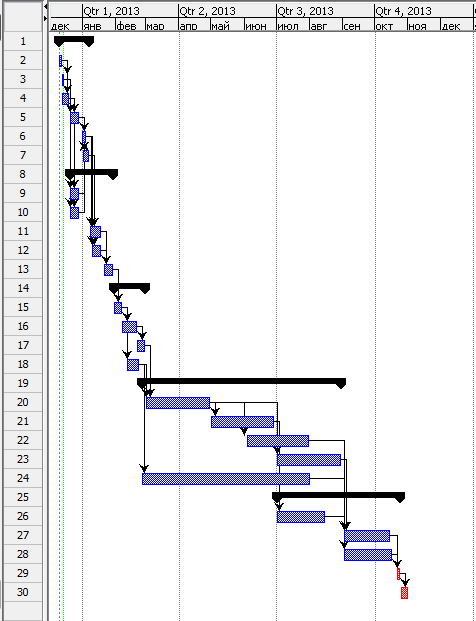
демонстрация заказчику

**Сдача проекта**

# Структурная декомпозиция работ по функциональным блокам

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **ПРОЕКТ** | | | | | | | | | | | | | | |
|  | | | | | |  | | | |  | | |  | |
| **Серверная часть** | | | | | | **Клиентская часть (web)** | | | | **Хранилище данных (БД)** | | | **Коммуниукации** | |
| 1. | Проектирование | | | | | 1. | Проектирование | | | 1. | Проектирование | | 1. | Связь клиент-сервер |
| 2. | Разработка | | | | | 2. | Разработка | | | 2. | Разработка | | 2. | Связь сервер-БД |
|  | 1. | | Функционал | | |  | 1. | Дизайн | |  | 1. | данные о пользователях |  |  |
|  |  | | 1. | | возможность авторизации пользователя |  | 2. | Функционал | |  | 2. | проекты пользователей |  |  |
|  |  | | 2. | | форум |  |  | 1. | авторизация |  | 3. | видео-уроки |  |  |
|  |  | | 3. | | компиляция пользовательского кода |  |  | 2. | форум | 3. | Документация | |  |  |
|  |  | | 4. | | запуск и тестирование пользовательского кода |  |  | 3. | работа с кодом |  |  |  |  |  |
|  |  | | 5. | | работа с пользовательскими проектами |  |  | 4. | работа с проектами |  |  |  |  |  |
|  |  | | 6. | | работа с видео-уроками |  |  | 5. | просмотр видео-уроков |  |  |  |  |  |
| 3. | Документация | | | | | 3. | Документация | | |  |  |  |  |  |
|  |  |  | |  | |  | 1. | пользовательская | |  |  |  |  |  |
|  |  |  | |  | |  | 2. | разработчику | |  |  |  |  |  |

# Диаграмма Ганта



# Метод функциональных точек

Шаг 1. Типы оценки:

● Проект разработки ПП.

Шаг 2. Область оценки и граница продукта.

● Для проекта разработки: все разрабатываемые

функции.

Опишем для примера пять функций

1. Авторизация пользователя

Шаги 2-4.

логические файлы: логин и пароль пользователя - ILF;

тип транзакции: EI;

DET = 5: логин;

пароль;

кнопка «Войти»;

сообщение об авторизации;

сообщение об ошибке авторизации;

FTR=1;

Сложность транзакции: Low;

Количество (невыровненных) ФТ: +=3;

RET=5;

Сложность лог.файла:Low;

Количество (невыровненных) ФТ: +=7;

1. Отправка сообщения на форум

Шаги 2-4.

логические файлы: логин пользователя - ILF;

сообщение – ILF;

тип транзакции: EI;

DET = 5: логин;

текст сообщения;

кнопка «Отправить»;

кнопка «Отменить»

сообщение об ошибке;

FTR=2;

Сложность транзакции: Average;

Количество (невыровненных) ФТ: +=4;

RET=5;

Сложность лог.файла:Low;

Количество (невыровненных) ФТ: +=7;

1. Просмотр видео-урока

Шаги 2-4.

логические файлы: видеоурок - ILF;

тип транзакции: EQ;

DET = 5: видео;

кнопка «Смотреть»;

кнопка «Прервать»;

сообщение об ошибке плеера;

FTR=1;

Сложность транзакции: Low;

Количество (невыровненных) ФТ: +=3;

RET=4;

Сложность лог.файла:Low;

Количество (невыровненных) ФТ: +=7;

1. Отправка кода серверу для компиляции и запуска

Шаги 2-4.

логические файлы: код пользователя - ILF;

результат компиляции –ILF;

результат запуска – ILF;

подключаемые библиотеки – ILF;

тип транзакции: EI&EO;

DET = 7: код программы;

кнопка «Компилировать»;

кнопка «Запустить»;

сообщение об удачной компиляции;

сообщение об ошибке компиляции;

результат запуска;

сообщение об ошибке времени выполнения;

FTR=4;

Сложность транзакции: High;

Количество (невыровненных) ФТ: +=7;

RET=7;

Сложность лог.файла:Average;

Количество (невыровненных) ФТ: +=10;

1. Сохранение текущего состояния проекта на сервере

Шаги 2-4.

логические файлы: логин пользователя - ILF;

файл с проектом – ILF;

тип транзакции: EI;

DET = 4: логин;

кнопка «Сохранить»;

сообщение о сохранении;

сообщение об ошибке сохранения;

FTR=2;

Сложность транзакции: Low;

Количество (невыровненных) ФТ: +=3;

RET=4;

Сложность лог.файла:Low;

Количество (невыровненных) ФТ: +=7;

Шаг 5. Суммарный UFP:=58

Шаг 6. Коэффициент выравнивания.

1. Обмен данными. 5

2. Распределённая обработка. 0

3. Производительность. 4

4. Загруженность аппаратных ресурсов. 5

5. Количество транзакций в единицу времени. 5

6. Оперативная обработка данных. 4

7. Эргономичность. 5

8. Интенсивность изменения ILF в транзакциях. 4

9. Сложная логическая или математическая обработка. 1

10. Повторное использование кода/компонент. 4

11. Лёгкость установки и миграции. 4

12. Лёгкость администрирования. 4

13. Работа во многих местах и организациях. 4

14. Удобство модификации. 3

Х:=52;

VAF:= 0.65 + 0.01\*X = 0.65 + 0.52 = 1.17

Шаг 7. Выровненные ФТ.

AFP = UFP + CFP\*VAF = 58 +CFP\*1.17

Вывод: при учете полного функционала проекта AFP окажется не более 400 – 500 функциональных точек, что является вполне посильным объемом проекта при выделенных нами ресурсах.

# COCOMO

Для языка С++: LOC/UPF =55 => LOC ?=? 25000 условных строк кода;

a=2.4, b=1.05, c=2.5, d=0.38;

Трудоемкость = a\*(KLOC)b = 2.4\*30=70,5[человеко-месяцев]

Срок = c \* (Трудоемкость)d= 2,5\*5=12,6[месяцев]

Число разработчиков = Трудоемкость/ Срок разработки = 5,6[человек]

В нашем случае запланировано незначительно меньшее время, но при большем количеству разработчиков;

# Метод Pert

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **Наименование** | **В** | **П** | **О** | **Tperti** | **σi** |
| 1 | обсуждение с заказчиком | 3 | 5 | 2 | 3,166667 | 0,5 |
| 2 | формулировка целей | 2 | 4 | 1 | 2,166667 | 0,5 |
| 3 | сбор требований | 5 | 8 | 3 | 5,166667 | 0,833333 |
| 4 | предварительное планирование и разработка структуры проекта | 7 | 10 | 6 | 7,333333 | 0,666667 |
| 5 | согласование плана, сроков, бюджета с заказчиком | 5 | 7 | 4 | 5,166667 | 0,5 |
| 6 | подготовка договора | 5 | 8 | 2 | 5 | 1 |
| 7 | обсуждение в команде | 7 | 9 | 4 | 6,833333 | 0,833333 |
| 8 | выбор средств и ресурсов | 7 | 8 | 5 | 6,833333 | 0,5 |
| 9 | планирование работ | 9 | 13 | 5 | 9 | 1,333333 |
| 10 | распределение задач | 7 | 9 | 3 | 6,666667 | 1 |
| 11 | работа с рисками | 7 | 15 | 6 | 8,166667 | 1,5 |
| 12 | построение архитектуры проекта | 6 | 20 | 5 | 8,166667 | 2,5 |
| 13 | разработка прототипа и его тестирование | 10 | 15 | 7 | 10,33333 | 1,333333 |
| 14 | анализ результатов и корректировка архитектуры | 6 | 9 | 2 | 5,833333 | 1,166667 |
| 15 | документация проекта | 10 | 14 | 8 | 10,33333 | 1 |
| 16 | написание кода отдельных модулей | 43 | 60 | 30 | 43,66667 | 5 |
| 17 | реализация взаимодействия модулей | 43 | 55 | 35 | 43,66667 | 3,333333 |
| 18 | реализация интерфейса | 43 | 55 | 30 | 42,83333 | 4,166667 |
| 19 | анализ и корректировка | 45 | 60 | 20 | 43,33333 | 6,666667 |
| 20 | написание пользовательской документации | 114 | 120 | 50 | 104,3333 | 11,66667 |
| 21 | написание тестов | 34 | 45 | 25 | 34,33333 | 3,333333 |
| 22 | тестирование | 32 | 40 | 27 | 32,5 | 2,166667 |
| 23 | выявление и исправление ошибок | 33 | 50 | 20 | 33,66667 | 5 |
| 24 | демонстрация заказчику | 4 | 7 | 2 | 4,166667 | 0,833333 |
| 25 | Сдача проекта | 5 | 7 | 3 | 5 | 0,666667 |
|  | Итого |  |  |  | 482,3333 | 57 |

T = Tpert + Z\*σ;

Z = (T – Tpert)/σ;

Какова вероятность, что затраты не превысит значения T = 500?

Z = (500 – 482.3)/57 = 0.31, а значит вероятность равна 0.62.

# МЕХАНИЗМЫ МОНИТОРИНГА И КОНТРОЛЯ

* Еженедельная отчетность о ходе работ руководителю в виде электронных отчетов
* Ежедневное устное согласование с руководителем
* Ежемесячная отчетность перед заказчиком посредством электронной переписки, корректировка хода работ