Министерство науки и высшего образования Российской Федерации

ФГБОУ ВО «Кубанский государственный технологический университет»

Кафедра информационных систем и программирования

**КОЛЛЕКТИВНАЯ РАЗРАБОТКА УПРАВЛЕНИЕ ПРОГРАМНЫМ ПРОДУКТОМ**

**Отчет по лабораторной работе №3**

Выполнил:

Студент 4 курса

группы 19-КБ-ПИ1

Корендюк А.Ю.

Краснодар

2022

**Задание для самостоятельной работы**

Построить схему проектного управления для конкретного программного проекта. Сделать краткое описание работ и взаимодействий по ролям членов команды.

**Выполнение задания:**

Компания для рассмотрения в лабораторной работе: Автосервис.

Общие работы при создании программного продукта:

1. Создание функциональных требований.

2. Создание системных требований.

3. Формирование ТЗ на основе требований.

4. Разработка ПО для Автосервиса.

5. Тестирование ПО для Автосервиса.

6. Создание пользовательской документации для пользования продуктом.

7. Сборка проекта.

8. Организация приемки-передачи и ввод в эксплуатацию.

10. Сопровождение ПО после ввода в эксплуатацию.

Типы ролевых кластеров по модели команды MSF

|  |  |
| --- | --- |
| Тип | Решаемые кластером задачи |
| 1. Кластер управления созданием программного обеспечения для Автосервиса | Основные задачи:  1. Создание функциональных требований.  2. Создание системных требований.  3. Формирование ТЗ на основе требований. |
| 2. Кластер управления сроками выполнения проекта | Обеспечение управленческих функций:  1. Проверка выполнения каждого этапа календарного плана в срок.  2. Контроль за использованием ресурсов проекта: денежные, техническое обеспечение, человеческие ресурсы |
| 3. Разработка программного обеспечения Автосервиса | Написания программного кода для ПО |
| 4. Тестирование программного обеспечения Автосервиса | Проверка программного обеспечения на корректность работы, выявление ошибок и их исправления |
| 5. Создание пользовательской документации и сборка проекта | Сборка проекта и написания пользовательской документации для использования |
| 6. Кластер организация приемки программного обеспечения, ввод в эксплуатацию и сопровождение | Основные задачи:  1. Организация приемки созданного программного обеспечения  2. Ввод ПО в эксплуатацию  3. Дальнейшее сопровождение ПО. |

Роли членов команды по модели MSF

|  |  |  |
| --- | --- | --- |
| Формулировка цели | Роль, выполняемая членом команды | Краткое описание работ |
| Создание программного обеспечения, соответствующего требованиям заказчика | Аналитик (связь между заказчиком и технической стороной) | 1. Общение с заказчиком и составление бизнес-процессов Автосервиса.  2. Выявление основных требований к ПО от руководителя отдела.  3. Создание функциональных требований и (уточняются у рукодителей отделов, например: какие данные, получаются от клиентов) системных требований (требования уточняются у системных администраторов: строение БД).  4. Создание ТЗ с руководителем проекта и руководителем Автосервиса. |
| Управление сроками проекта | Руководитель проекта | 1. Создание ТЗ с аналитиком и руководителем автосервиса.  2. Создание календарного плана.  3. Управление выполнением каждого этапа проекта.  4. Внедрение проекта в Автосервис. |
| Создание программного кода для ПО | Разработчик | 1. Создание архитектуры ПО автосервиса отдельно для технического отдела и отдела администрирования.  2. Написание основных библиотек(требования функциональные и системные).  3. Написание программного кода. |
| Тестирование программного кода | Тестировщик | 1. Разработка тестовов  2. Функциональное тестирование.  3. Системное тестирование.  4. Отчет о проведенном тестировании. |
| Приемка и внедрение ПО | Специалист по внедрению | 1. Создание эксплуатационной документации отдельно для отдела администрирования (работа с базами данных) и технического отдела (работа с интерфейсом, работа с базой неисправностей).  2. Внедрение проекта в Автосервисе.  3.Проверка в течение месясца |

Сочетание ролей членов команды в модели MSF:

1. Аналитик и руководитель проекта (составляют ТЗ).

2. Разработчик и тестировщик (исправление ошибок в работе программного обеспечения).

3. Аналитик и специалист по внедрению (обеспечение организации приемки ПО).

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Роль | Аналитик | Руководитель проекта | Разработчик | Тестировщик | Специалист по внедрению |
| Аналитик |  | + | - | - | + |
| Руководитель проекта | + |  | +/- | +/- | - |
| Разработчик | - | +/- |  | + | - |
| Тестировщик | - | +/- | + |  | - |
| Специалист по внедрению | + | - | - | - |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

4. Руководитель проекта (контроль за работой разработчика и тестироващика).

Состав команды: Аналитик, руководитель проекта, разработчик, тестировщик, специалист по внедрению.

**Расчет временной продолжительности**

Программные продукты могут быть отнесены по степени новизны к одной из 4-х групп:

группа А — разработка принципиально новых задач;

группа Б — разработка оригинальных программ;

группа В — разработка программ с использованием типовых решений;

группа Г — разовая типовая задача.

Для данной задачи степень новизны — Б , так как программа Автосервиса является разработкой оригинальной программы.

По степени сложности программные продукты могут быть отнесены к одной из 3-х групп:

1) 1 — алгоритмы оптимизации и моделирования систем;

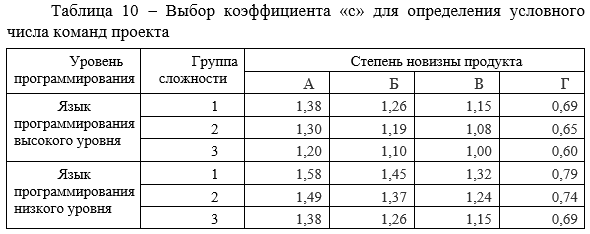
2) 2 — задачи учета, отчетности и статистики;

3) 3 — стандартные алгоритмы.

Данная задача может быть отнесена к 1 группе сложности.

Кроме того такие программы пишуться на языках высокого уровня, для разработки этой программы был выбран язык C#. Он является языком программирования высокого уровня.

На основании всех этих параметров можно выделить коэффициент «с» который определяется из таблицы 3 на пересечении групп сложности и степени новизны.



В нашем случае коэффициент с = 1.26

Теперь, исходя из формулы можно определить условное число команд

Q = q \* c = 1500 \* 1.26 = 1890

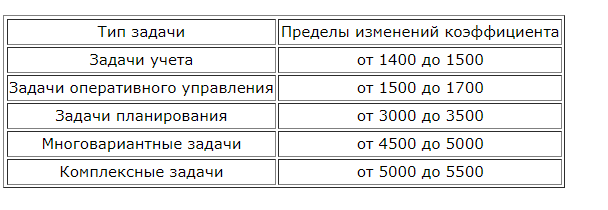
где q - коэффициент, учитывающий условное число команд в зависимости от типа задачи;

Поскольку в данной работе была разработана система по работе Автосервиса. Данная программа решает задачи учета и задачи оперативного управления, будь то работа руководителя с программой или работа с программой сервисных работников. В связи с этим был принят коэффициент = 1500.

Важнейшие задачи учета: сбор, обработка и отображение первичных данных о хозяйственной деятельности систематизация данных с целью получения и обобщения итоговой информации о хозяйственной деятельности

создание исходной информационной базы для планирования и осуществления контроля за выполнением планов.

Задача оперативного управления — обеспечить быстрое, своевременное и эффективное выполнение всех функций внутри процесса.



Общая формула для расчета времени продолжительности работ по проектированию и разработки ПО Автосервиса.

https://studfile.net/html/2706/363/html_1obn8N1IGE.2GHp/htmlconvd-uygMES_html_76286ab846f7bec7.gif

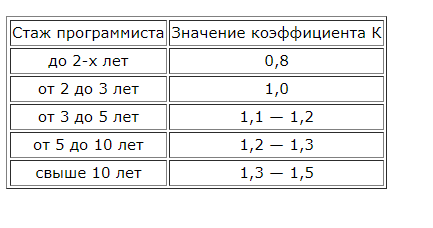
Далее коэффициент с – выбран 1,26 т.к язык разработки по сложности 3 группы и степень новизны группы Б и использующая типовые решения.

Условное число команд:

Q = q \* c = 1500 \* 1.26 = 1890

B – Коэффициент в зависимости от сложности задачи и числа изменений выбирается в интервале от 1,2 до 1,5.

Поэтому примем коэффициент B = 1.4, так как проект не является самым сложным, но и не является легким.



K – коэффициент, учитывающий квалификацию программиста.

В нашем проекте будут работать люди которые имеют стаж от 3 лет и до 5 лет. Для такого проекта, мы не можем взять разработчиков имеющих небольшой опыт работы. В связи со сложностью проекта и выбранным языком программирования было принято решение взять в команду разработки программистов со стажем работы от 3 до 5 лет, так как эти специалисты смогут решить как и легкие задачи, так и справиться со сложными задачами.

Для данного проекта примем К = 1.2.

Определяем время, затраченное на каждый этап создания программного продукта:

По формуле

https://studfile.net/html/2706/363/html_1obn8N1IGE.2GHp/htmlconvd-uygMES_html_76286ab846f7bec7.gif

где *Тразр* – общее время на создание программного продукта;

*Тпо*–подготовка описания задачи;

*То* – описание задачи;

*Та* – разработка алгоритма;

*Тбс* – разработка блок-схемы алгоритма;

*Тн*– написание программы;

*Тп* – набор программы на ПК;

*Тот*– отладка и тестирование программы;

*Тд* – оформление документации, инструкции пользователю, пояснительной записки и др.

 Тпо (время на подготовку описания задачи), берется по факту и составляет (от 3-х до 5-ти дней по 8 часов):

Тпо = 40 чел / час.

То = Q \* B / (50 \*K),

To = 1890 \* 1,4 / (50 \* 1,2) = 44.1 чел/час  
  
Ta (время на разработку алгоритма) рассчитываем по формуле:

Ta = Q / 50k

Ta = 1890 / 50 \* 1,2 = 31.5 чел/час  
  
Tбс время на разработку блок - схемы определяется аналогично по той же формуле, по которой высчитывается время на разработку алгоритма.  
Tбс =1890 / 50 \* 1,2 = 31.5 чел/час  
  
Тн (время написания программы на языке программирования) определяется по формуле:  
Tн = Q \* 1,5 / 50K  
Тн = 1890 \* 1.5 / 50 \* 1.2 = 47.25 чел/час

Тn - (время набора программы) определяется по формуле:

Тn = Q / 50 = 1890/50 = 37.8 чел/час  
  
Тот (время отладки и тестирования программы) определяется по формуле:  
Тот = Q \* 4.2 / 50k = 1890 \* 4.2 / 50 \* 1.2 = 132.3 чел/час  
  
Тд - время на оформление документации, инструкции пользователю, пояснительной записки определяется по формуле  
Тд = Q / 24k = 1890 / 24 \* 1.2 = 65.6 чел/час

https://studfile.net/html/2706/363/html_1obn8N1IGE.2GHp/htmlconvd-uygMES_html_76286ab846f7bec7.gif

Тразр = 40+44.1 +31.5+31.5+47.2+37.8+132.3+65.6 = 430 чел/час

Тдн = Т/8 = 430 / 8 = 54 дня на разработку командой этого проекта.

Вывод: для разработки данного проекта требуется 54 дня на все этапы проектирования, разработки и тестирования. Для таких проектов это считается средним показателем, расчет производился согласно ГОСТ 19.102-77 «Стадии разработки» и формуле Тразр.