

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение высшего
образования
**«Сибирский государственный университет науки и технологий имени
академика М.Ф. Решетнева»**
Институт инженерной экономики
Кафедра информационных экономических систем

ОБЛАЧНЫЕ ТЕХНОЛОГИИ В ЭКОНОМИКЕ

Руководитель проекта: студент группы БПЭ 20-02


Булавко В.Р.

Выполнил: студент группы БПЭ20-02

Бут Я.С.

Проверил: преподаватель Баляков Д.Ф.

Красноярск, 2022 г.

Инв. № подл.	Подпись и дата	Взам. инв. №	Инв. № дубл.	Подпись и дата					
Изм.	Лист	№ докум.	Подпись	Дата					
Разраб.									
Пров.									
Н.конт									
Утв.									
					Лит.		Лист	Листов	
							2	9	
									

1 Введение

1.1. Основанием для выпуска является техническое задание.

1.2 Надежность сайта – эксплуатационное качество сайта, характеризующееся вероятностью безотказной работы сайта в определенный промежуток времени при сохранении параметров, заданных в техническом задании на сайт. Расчёт надежности является одним из основных характеристик качества сайта.

1.3 Основные задачи

Основная задача Расчета надёжности состоит в определении значения показателей надежности объекта по известным значениям ее элементов при заданных условиях эксплуатации.

2 Объект расчёта

2.1 Описание объекта

«Сыт по горло» предназначен для исполнения своих функций в удаленном режиме. Основными функциями приложения являются:

- удаленный список готовой пищевой продукции к продаже;
- удаленный заказ пищевой продукции;
- оплата готовой пищевой продукции;

2.2 Основные допущения при расчётах.

Для расчёта использовалась методика, предложенная Н.В. Василенко и В.А. Макаровым. Предлагается определять надежность ПО на основе двух оценок: оценки процесса его разработки и оценки результатов тестирования.

Выделены следующие наиболее существенные факторы, влияющие на надежность ПО в процессе разработки:

- совершенство процесса управления разработкой;
- квалификация участников разработки программного продукта;
- сложность архитектуры системы;
- язык программирования;
- трудоемкость разработки;

Инов. № подл.	Подпись и дата	Взам. инв. №	Инов. № дубл.	Подпись и дата						Лист
										3
Изм.	Лист	№ докум.	Подпись	Дата						

- опыт в разработке подобных систем;
- взаимодействие в команде;
- полнота и качество документации.

3 Расчётная часть

3.1 Управление

Предлагается оценивать в соответствии с моделью зрелости процесса разработки ПО - CMM (Capability Maturity Model). В модели CMM определено пять уровней зрелости организаций. В результате аттестации компании присваивается определенный уровень, который в дальнейшем может повышаться или (теоретически) понижаться.

Так как наша организация достаточно простая, то наш уровень можно считать начальным. Оценка – 4.

3.2 Квалификация

$$Q = 0,4 \cdot 5 \cdot \frac{N_c}{N} + 0,6 \cdot \frac{\sum_{i=1}^N V_i \cdot Z_i}{N}$$

где Q — оценка квалификации команды; N — количество человек в команде разработки; N_c — число сертифицированных специалистов в команде; V_i — статусная оценка i -го участника команды (от 1 до 5); Z_i — загрузка i -го участника команды в проекте;

0,4 и 0,6 — весовые коэффициенты, установленные эмпирически;

5 — коэффициент приведения к единой шкале.

$$Q = 0.4 \cdot 5 \cdot \frac{3}{10} + 0.6 \cdot \frac{(5 \cdot 3) + (4 \cdot 5) + (4 \cdot 3) + (4 \cdot 4) + (3 \cdot 4) + (2 \cdot 2) + (3 \cdot 2) + (2 \cdot 4) + (1 \cdot 1) + (1 \cdot 1)}{13} = 6.48.$$

Оценка – 4.

3.3 Архитектура

Различают следующие основные классы архитектур программных средств: цельная программа, комплекс автономно выполняемых программ, слоистая программная система, коллектив параллельно выполняемых программ.

Инов. № подл.	Подпись и дата	Взам. инв. №	Инов. № дубл.	Подпись и дата						Лист
										4
Изм.	Лист	№ докум.	Подпись	Дата						

Оценка – 4.

3.4 Язык

Язык программирования формально можно оценить как количество операторов данного языка на один функциональный указатель. К функциональным указателям относят внешний ввод, внешний вывод, внутренний логический файл, внешний интерфейсный файл, внешние запросы.

Так как мы используем готовую конфигурацию, которую нужно доработать, то нашей оценкой языка программирования будет – 3.

3.5 Трудоекость

Трудоемкость разработки оценивается по факту после ее окончания.

Оценка – 3.

3.6 Опыт разработки

Опыт в разработке подобных систем оцениваются экспертно.

Оценка – 2.

3.7 Командное взаимодействие

Взаимодействие в команде оцениваются экспертно.

Оценка – 3.

3.8 Документация

Полнота и качество документации оцениваются экспертно.

Оценка – 4.

3.9 Подсчёт результатов

Обобщенную оценку надежности как результата организации процесса разработки R_1 определим как взвешенную сумму оценок факторов, влияющих на надежность:

$$R_1 = \frac{\sum_{i=1}^8 V_i \cdot \lambda_i}{8}$$

где V_i — оценка i -го фактора по пятибалльной шкале; λ_i — вес i -го фактора по пятибалльной шкале. Теоретически $0 \leq R_1 \leq 25$.

$$R_1 = \frac{5 \cdot 4 + 5 \cdot 4 + 4 \cdot 4 + 3 \cdot 4 + 3 \cdot 3 + 2 \cdot 2 + 3 \cdot 3 + 3 \cdot 4}{8} = 12.75$$

Изн. № подл.	Подпись и дата	Взам. инв. №	Изн. № дубл.	Подпись и дата						Лист
										5
Изм.	Лист	№ докум.	Подпись	Дата						

Перечислим требования, которым должна удовлетворять корректная модель.

1. Вся совокупность тестов и возможных комбинаций входных данных (все тесты, принадлежащие программе функциональных испытаний) есть одно испытание на надежность.

2. Целесообразно, чтобы модель учитывала процесс доработки сайта по результатам испытаний

3. Оценка надежности сайта должна производиться тогда, когда после испытания данной версии сайта не было обнаружено ни одной ошибки.

4. В модели не должны учитываться временные интервалы между обнаруженными ошибками, так как они отражают в основном очередность отдельных проявляющих ошибки условий или очередность смены наборов входных данных.

5. Модель не должна требовать априорных данных или данных предшествующего периода функционирования однотипных программных средств

Для описания процесса проявления и исправления ошибок в сайте можно воспользоваться моделью Пуассона, удовлетворяющей всем пяти условиям.

Пусть $N(t_r)$ — число проявившихся ошибок за время t тестирования версии r сайта есть неоднородный процесс Пуассона с интенсивностью $\lambda(t)$, $t \geq 0$.

Обозначим справедливо

$$P\{N(t_r, t_{r-1}) = k\} = \frac{[m(t_r) - m(t_{r-1})]^k}{k!} \cdot e^{-(m(t_r) - m(t_{r-1}))}, m(t_r) = \int_0^t \lambda(u) du,$$

где $k = 0, 1, 2, \dots$ — число ошибок; $r = 0, 1, 2, \dots$ — ряд последовательно испытываемых версий ПО; $m(t_r)$ — среднее число проявлений ошибок за время t в версии r . Отсюда определяется вероятность того, что на интервале (t, t_s) ошибки не проявятся.

Изн. № подл.	Подпись и дата				
	Изн. № дубл.				
	Взам. инв. №				
	Подпись и дата				
Изн. № подл.					
Изм.	Лист	№ докум.	Подпись	Дата	Лист
					6

В результате мы получим некоторое положительное число ≤ 100 . Полученное значение может трактоваться как вероятность того, что в последней версии в процессе эксплуатации не проявится ошибка. Обозначим эту оценку как R_2 .

$$\begin{aligned}
 R_2 &= \frac{\left[\frac{15}{20000} - \frac{1}{20000}\right]^{14}}{14!} \cdot e^{-\left(\frac{15}{20000} - \frac{1}{20000}\right)} \cdot 100\% \\
 &= \frac{[0,00075 - 0,00005]^{14}}{14!} \cdot e^{-(0,00075 - 0,00005)} \cdot 100\% \\
 &= \frac{[0,0007]^{14}}{14!} \cdot e^{-(0,0007)} \cdot 100\% = \frac{7 \cdot 10^{10}}{8 \cdot 10^{10}} \cdot 0,99 \cdot 100\% \\
 &= 0,875 \cdot 0,99 \cdot 100\% = 86\%
 \end{aligned}$$

Таким образом, мы получили две оценки надежности ПО: R_1 как результат оценки процесса разработки и R_2 как результат тестирования готового программного продукта.

На основании этих оценок предлагается новая модель оценки надежности ПО.

В рамках этой модели введем два новых понятия: класс надежности и матрица надежности. Под классом надежности будем понимать совокупную оценку надежности программного продукта, которая базируется на оценке процесса разработки и результатов испытания ПО и позволяет сравнивать программные продукты между собой по надежности их функционирования. Таблица иллюстрирует понятие матрицы надежности.

Таблица – 1

		В процентах			
$R_2 \backslash R_1$	0-50 %	51-75%	76-90%	91-98%	99-100%
0-10	Класс М	Класс К	Класс Н	Класс F	—
11-20	Класс L	Класс J	Класс G	Класс D	Класс В
21-25	—	Класс I	Класс E	Класс C	Класс А

Будем откладывать по горизонтали интервалы оценки надежности R_2 , а по вертикали — интервалы оценки надежности R_1 . На пересечении этих интервалов получим класс надежности для данного продукта.

Инов. № подл.	Подпись и дата	Взам. инв. №	Инв. № дубл.	Подпись и дата					
Изм.	Лист	№ докум.	Подпись	Дата					
					Лист				
					7				

Два пересечения не определяют класса надежности: когда одна из оценок низшая, а вторая — наивысшая. В таких случаях будем считать, что либо процесс тестирования организован некорректно, либо дана не отражающая реальности оценка процесса разработки. Остальные классы могут быть ранжированы по степени увеличения надежности в соответствии с их положением в матрице надежности. Класс А мы представляем как класс высоконадежного ПО, показавшего максимальную вероятность дальнейшей безотказной работы и процесс разработки которого оценен максимально. Остальные классы представляют ПО меньшей надежности, однако в некоторых случаях этой надежности будет достаточно для нормального функционирования конкретного ПО.

По матрице надежности наш класс - Класс G.

4 Выводы

Определены факторы, влияющие на надежность, по которым возможно производить оценку процесса разработки ПО, сформулированы принципы оценки этих факторов по пятибалльной шкале.

Формализованы требования к модели оценки надежности по результатам его тестирования. Адаптирована модель оценки надежности, основанная на неоднородных Пуассоновских процессах, которая максимально ориентирована на оценку надежность программного средства по результатам его тестирования в производстве.

5 Заключение

Исходя из полученного результата расчета надежности, разрабатываемое ПО имеет средний уровень надежности.

Инв. № подл.	Подпись и дата					Лист
	Инв. № дубл.					
	Взам. инв. №					
	Подпись и дата					
Изм.	Лист	№ докум.	Подпись	Дата		
					8	

