**План разработки программного обеспечения**

Лист изменений

|  |  |  |  |
| --- | --- | --- | --- |
| **Номер версии** | **Дата изменений** | **Изменения** | **Автор изменений** |
| Версия 1.0. | 01.03.2016 | Документ был создан | Акимутин Е.Е, Асеев М.Д. |
| Версия 1.0 | 02.03.2016 | Документ был одобрен | Сорокин С.В. |

1. Введение
   1. Назначение

Данный документ предназначен для сбора всей информации, необходимой для управления проектом. Он описывает подход к разработке программного обеспечения и является планом верхнего уровня, создаваемым и применяемым руководителями для направления усилий по разработке продукта.

* 1. Область применения

Документ применяется исключительно для демонстрации мероприятий, которые могут использоваться при достижении целей процессов, определённых рамками учебного проекта по созданию системы развития алгоритмического мышления.

* 1. Ссылки
  2. Термины, определения и соглашения

Документация — набор документов, включающий в себя спецификацию требований и сопутствующие ей документы.

Закрепленная роль — роль, которая выполняется участником проекта от начала до конца разработки.

Роль — активность, которую участник проекта выполняет в конкретный промежуток времени.

Участник — человек, входящий в команду разработчиков проекта.

Инспекция — способ верификации документов, основанный на экспертной оценке их правильности, выполняемый одним или несколькими инспекторами.

Описание проекта ПО — совокупность документов, включающая в себя use-case диаграммы, модели базы данных и структуру веб-интерфейса.

1. Организация жизненного цикла ПО
   1. Ответственность

В ходе организации работы над проектом выделяются следующие роли:

* Технический писатель

В обязанности технического писателя входит составление требований верхнего уровня.

* Проектировщик структуры web-интерфейса

В обязанности проектировщика web-интерфейса входит составление документа, описывающего содержание каждой страницы системы, а также их взаимосвязей.

* Дизайнер макетов страниц

В обязанности дизайнера входит проектирование внешнего вида системы и создание макетов страниц.

* Аналитик

В обязанности аналитика входит составление UML-, IDEF0-диаграмм и блок-схем, описывающих различные процессы, происходящие внутри системы, а также процессы разработки ПО.

* Проектировщик базы данных

В обязанности проектировщика базы данных входит создание концептуальной, логической и физической моделей базы данных.

* Архитектор системы

В обязанности архитектора системы входит проектирование архитектуры ПО, т.е. принятие ключевых проектных решений относительно внутреннего устройства программной системы и её технических интерфейсов.

* Front-end разработчик

В обязанности front-end разработчика входит разработка интерфейса взаимодействия между пользователем и основной программно-аппаратной частью системы без обращения к серверу.

* Back-end разработчик

В обязанности back-end разработчика входит написание содержательной части программы, т.е. логики её действий, а также реализация модели.

* Тестировщик

В обязанности тестировщика входят разработка тест-планов и тест-кейсов, проверка работоспособности системы и представление результатов проверки.

* Контент-менеджер

В обязанности контент-менеджера входит наполнение базы задач, составление названий разделов, условий задач, классификация задач по разделам, а также разработка тестов, покрывающих все ограничения, указанные в условиях задач.

* Менеджер проекта

В обязанности менеджера проекта входит управление процессом разработки: составление календарного плана, распределение ролей и задач, а также контроль за выполнением этих задач и оценивание качества работа каждого участника проекта.

* Главный программист

В обязанности главного программиста входит принятие ключевых решений, касающихся процесса проектирования и кодирования. Также главный программист контролирует выполнение задач, связанных с проектированием и кодированием ПО.

Распределение ролей участников проекта не является жестким, то есть в ходе процесса разработки каждый участник может попробовать себя в нескольких ролях. Единственными закрепленными ролями являются роли менеджера и главного программиста.

* 1. Взаимодействие процесса разработки с другими процессами жизненного цикла ПО

Все процессы разработки ПО взаимодействуют посредством процесса управления конфигурацией ПО, таким образом, все данные, создающиеся или модифицирующиеся в ходе выполнения процессов разработки ПО, являются входными данными для процесса управления конфигурацией ПО, все процессы разработки ПО получают входные данные из процесса управления конфигурацией ПО.

1. Процесс разработки ПО
   1. Разработка требований к ПО



Рисунок 1. Процесс разработки программных требований

* + 1. Входные данные процесса разработки требований к ПО

Исходные материалы для разработки программных требований включают в себя:

* Последняя базовая версия спецификации требований;
* Описание проектных решений на уровне системы (например, описание архитектуры системы);
* Запрос на изменение;
* Список несоответствий в программных требованиях (при повторных входах в процесс по результатам инспекции).
  + 1. Условия входа в процесс разработки требований к ПО

Вход в процесс разработки требований к ПО осуществляется в случае выполнения любого из приведённых ниже условий:

1. Запрос на изменение требований к ПО назначен ответственному разработчику менеджером проекта, при этом установлены базовые версии всех входных данных процесса разработки требований к ПО.
2. В результате инспекции запроса на изменение требований к ПО обнаружены несоответствия, и запрос на изменение требований к ПО отправлен на исправление.
   * 1. Описание процесса разработки требований к ПО

При входе в процесс разработки требований к ПО ответственный разработчик должен перевести запрос на изменение в состояние выполнения (оранжевая метка), тем самым извещая руководителя проекта о начале работы по запросу на изменение, после чего он должен выполнять действия, заданные в запросе на изменение.

На основании анализа описания запроса на изменение (при входе в процесс по условию A), записи об инспекции (при входе в процесс по условию B) и последней базовой версии спецификации требований ответственный разработчик определяет те документы требований к ПО, которые необходимо изменить, создать или удалить.

Если новая версия спецификации требований готова, ответственный разработчик вносит изменения в указанные в запросе на изменение данные: изменяет состояние запроса на изменение в состояние готовности к инспекции (жёлтая метка).

Если в процессе разработки требований обнаружены несоответствия во входных данных процесса, то ответственный разработчик должен создать сообщение о проблеме.

Для всех требований к ПО, не являющихся производными, необходимо установить трассировочные связи с соответствующей им спецификацией требований.

После завершения работы над требованиями ответственный разработчик должен известить об этом менеджера проекта. Для этого он должен выполнить следующие действия:

Если вход в процесс был выполнен по условию A, то завершив работу по запросу на изменение, менеджер проекта проводит инспекцию запроса на изменение и меняет его состояние в зависимости от результатов проведённой инспекции:

1. В случае, если инспекция не выявила несоответствий в последней версии спецификации требований, то менеджер проекта меняет состояние запроса на изменение на состояние готовности (зелёная метка);
2. В случае, если инспекция выявила несоответствия в последней версии спецификации требований, то менеджер проекта формирует список несоответствий, отправляет его ответственному разработчику и меняет состояние запроса на изменение на состояние доработки (красная метка).
   * 1. Условия выхода из процесса разработки требований к ПО

Выход из процесса разработки требований к ПО происходит при выполнении всех следующих условий:

1. Все найденные во входных данных несоответствия зарегистрированы в сообщениях о проблемах;
2. Работы, начатые по запросу на изменение или по результатам инспекции (3.1.2.), завершены;
3. Менеджер проекта извещён о завершении работ в соответствие с 3.1.3 и провёл верификацию изменений.
   * 1. Выходные данные процесса разработки требований к ПО

* Спецификация требований (изменённая версия);
* Сообщения о проблемах.
  1. Проектирование ПО

Целью процесса проектирования ПО является определение архитектуры системы развития алгоритмического мышления.

* + 1. Входные данные процесса проектирования ПО

• План разработки ПО (настоящий документ);

• Спецификация требований;

• Запрос на изменение;

• Запись об инспекции описания проекта ПО (при повторных входах в процесс по результатам инспекции).

* + 1. Условия входа в процесс проектирования ПО

1. Вход в процесс проектирования ПО осуществляется в случае выполнения любого из приведённых ниже условий:
2. Запрос на изменение описания проекта ПО назначен ответственному разработчику менеджером проекта, при этом установлены базовые версии всех входных данных процесса проектирования ПО.
3. В результате инспекции запроса на изменение описания проекта ПО обнаружены несоответствия, и запрос на изменение описания проекта ПО отправлен на исправление.
   * 1. Описание процесса проектирования ПО

С процедурной точки зрения процесс изменений в файлы описания проекта ПО аналогичен процессу внесения изменений в требования к ПО, описанному в разделе 3.1.3.

Основным методом при разработке описания проекта ПО будет являться метод структурного проектирования, представляющий собой процесс последовательного разбиения (декомпозиции) ПО на компоненты, а компонента – на подкомпоненты и функции.

* + 1. Условия выхода из процесса проектирования ПО

Выход из процесса проектирования ПО происходит при выполнении всех следующих условий:

1. Все найденные во входных данных несоответствия зарегистрированы в сообщениях о проблемах;
2. Работы, начатые по запросу на изменение или по результатам инспекции (3.2.2.), завершены;
3. Менеджер проекта извещён о завершении работ в соответствие с 3.2.3 и провёл верификацию изменений.
   * 1. Выходные данные процесса проектирования ПО

• Описание проекта ПО;

• Сообщения о проблемах.

* 1. Кодирование ПО

Цель процесса кодирования ПО – разработать исходный код, который являлся бы трассируемым, верифицируемым, непротиворечивым и правильно реализующим требования к ПО.

* + 1. Входные данные процесса кодирования ПО

• План разработки ПО (настоящий документ);

• Описание проекта ПО;

• Требования к ПО;

• Запрос на изменение;

• Запись об инспекции исходного кода ПО (при повторных входах в процесс по результатам инспекции);

• Стандарт на кодирование ПО (Code style).

* + 1. Условия входа в процесс кодирования ПО

Вход в процесс кодирования ПО осуществляется в случае выполнения любого из приведённых ниже условий:

1. Запрос на изменение исходного кода ПО назначен ответственному разработчику менеджером проекта, при этом установлены базовые версии всех входных данных процесса кодирования ПО.
2. В результате инспекции запроса на изменение исходного кода ПО обнаружены несоответствия, и запрос на изменение исходного кода ПО отправлен на исправление.
   * 1. Описание процесса кодирования ПО

С процедурной точки зрения процесс внесения изменений в файлы исходного кода аналогичен процессу внесения изменений в требования к ПО, описанному в разделе 3.1.3.

Отличия заключаются в следующем:

1. Разработчики вносят изменения в модули исходного кода, руководствуясь стандартом на кодирование ПО (Code style), и используют в качестве входных данных описание проекта ПО и последнюю версию спецификации требований.

2. При выполнении мероприятий ответственный разработчик использует инструментарий процесса кодирования (см раздел 4.1).

* + 1. Условия выхода из процесса проектирования ПО

Выход из процесса проектирования ПО происходит при выполнении всех следующих условий:

1. Все найденные во входных данных несоответствия зарегистрированы в сообщениях о проблемах;
2. Работы, начатые по запросу на изменение или по результатам инспекции (3.3.2.), завершены;
3. Менеджер проекта извещён о завершении работ, т.е. выполнены действия, аналогичные описанным в 3.1.3.
   * 1. Выходные данные процесса проектирования ПО

• Исходный код ПО;

• Сообщения о проблемах.

1. Среда разработки ПО
   1. Инструменты разработки ПО

На компьютерах разработчиков ПО устанавливаются следующие инструменты:

*Таблица 1. Программные средства, устанавливаемые на рабочих местах разработчиков ПО.*

|  |  |  |
| --- | --- | --- |
| № п/п | Case-средство | Назначение |
| 1 | Microsoft Word 2013 | Составление и редактирование документации |
| 2 | Microsoft Power Point 2013 | Создание презентации для отчетного мероприятия |
| 3 | NinjaMock | Прототипирование интерфейсов |
| 4 | Silverrun | Проектирование концептуальных моделей базы данных |
| 5 | ERwin | Проектирование логической и физической моделей базы данных |
| 6 | Microsoft Visio 2013 | Создание схем и диаграмм |
| 7 | StarUML | Проектирование UML-диаграмм |
| 8 | PyCharm | IDE для реализации |
| 9 | Google Chrome | Тестирование работы системы, заполнение базы данных |
| 10 | Microsoft Excel 2013 | Создание отчетов о проделанной работе, хранение оценок участников |
| 11 | Skype | Коммуникация между участниками |
| 12 | Git | Система контроля версий |
| 15 | TeamViewer | Совместное выполнение задач |

* 1. Языки программирования

Основным языком программирования при разработке исходного кода должен являться язык Python версии 3.5.1 и фреймворк данного языка — Django. Помимо языка Python использовались:

• прототипно-ориентированный сценарный язык программирования JavaScript;

• стандартный язык разметки документов в Интернете HTML5;

• формальный язык описания внешнего вида документа, написанного с использованием языка разметки, CSS3;

• формальный непроцедурный язык программирования для работы с базами данных SQLite версии 3.