Міністерство освіти і науки, молоді та спорту України

Національний Технічний Університет України

«Київський Політехнічний Інститут»

Навчально-науковий комплекс

«Інститут прикладного системного аналізу»

Кафедра системного проектування

«ПРОЕКТУВАННЯ ІНФОРМАЦІЙНИХ СИСТЕМ»

Лабораторна робота № 2

Скласти опис передпроектної документації (Software Requirement Specifications).

Виконала:

студент 4 курсу,

Група ДА-62

Василевич Б.С.

м. Київ

2019

1. **Введение**
   1. **Назначение**

Этот документ предоставляет подробное описание системы контроля физической нагрузки. Он объясняет назначение и особенности системы, интерфейсы системы, задачи системы, ограничения, при которых система должна работать, и как система реагирует на внешние раздражители. Этот документ предназначен для команды, которая будет реализовывать и проверять корректность работы системы

* 1. **Область действий**

Эта система будет спроектирована таким образом, чтобы максимально улучшить качество жизни пользователя системы, предоставляя инструменты для автоматизации контроля процесса физической активности. Благодаря максимизации эффективности использования времени и продуктивности пользователя, система будет отвечать потребностям пользователя, при этом оставаясь легкой в понимании и использовании.

В частности, эта система предназначена для того, чтобы пользователь мог следить за своим здоровьем, путем получения информации о потраченных калориях после ввода данных о физической активности. Система содержит реляционную базу данных со списком физических упражнений и калорий.

* 1. **Определения, акронимы и сокращения**

Программный продукт (ПП) - это совокупность отдельных программных средств, их документации, гарантий качества, рекламных материалов, мер по обучению пользователей, распространению и сопровождению готового программного обеспечения.

Пользователь - лицо или организация, которое использует действующую систему для выполнения конкретной функции.

* 1. **Публикации**
  2. **Краткий обзор**

Следующий раздел “Полное описание” этого документа дает обзор функциональных возможностей продукта. Этот раздел не устанавливает конкретные требования. Вместо этого, он обеспечивает предварительные сведения о тех требованиях, которые подробно определяются в третьем разделе.

Третий раздел “Специфические требования” этого документа, написан в первую очередь для разработчиков и в технических терминах описывает детали функциональности продукта.

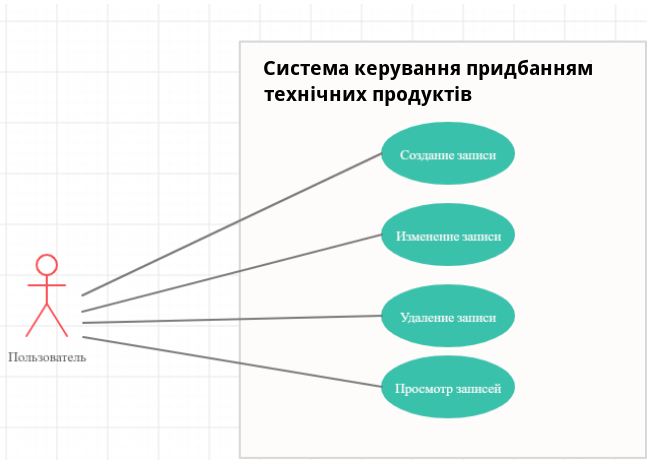
Оба раздела документа описывают один и тот же ПП в полном объеме, но предназначения для разных аудиторий и поэтому используют разные языки.

1. **Полное описание**
   1. **Перспектива изделия**

Пользователь общается с системой напрямую, вводит информацию и получает ответ на выходе.

* 1. **Функции изделия**

В этом разделе описаны варианты использования системы пользователем.



* 1. **Характеристики пользователя**

Пользователь - это покупатель или продавец продукта.

* 1. **Ограничения**

Весь программный код должен соответствовать версии JavaScript(ES6+).

Cистема должна использовать последнюю версию базы данных mongodb.

1. **Специфические требования**
   1. **Внешние интерфейсы**

Единственная ссылка на внешнюю систему - это ссылка на базу данных DB для предоставления информации о пользователях (таблица users) и записи технических продуктов (таблица products).

Поля таблицы users включают в себя имя пользователя, его тип(покупатель/продавец) и его номер (ID).

Поля таблицы products включают в себя дату, когда был создан продукт, номер технического продукта(ID), его название, количество и его продавец.

* 1. **Функции**

3.2.1

|  |  |
| --- | --- |
| Use Case Name | Добавить новую запись |
| Trigger | Пользователь выбирает добавить новую запись. |
| Path | 1.Пользователь нажимает на кнопку “Добавить”.  2.Система предоставляет пустую форму для ввода информации о записи: выпадающее меню с названиями упражнений, поле для ввода количества повторение  3. Пользователь вводить необходимую информацию и отправляет форму  4.Система проверяет правильность введенной информации и обновляет базу данных.  5.Система делает расчёт калорий в функции по полученной информации и отображает созданную запись. |

3.2.2

|  |  |
| --- | --- |
| Use Case Name | Изменить существующую запись |
| Trigger | Пользователь выбирает изменить запись. |
| Path | 1.Система предоставляет список всех имеющихся записей в порядке добавления.  2.Пользователь выбирает запись.  3.В открывшемся окне информации о записи, пользователь нажимает кнопку “Изменить”.  4.Система предоставляет форму с данными, которые были получены по данной записи с базы данных.  5. Пользователь изменяет необходимые ему поля и подтверждает изменение.  6.Система проверяет правильность введенной информации и обновляет базу данных.  7. Система делает перерасчет калорий по функции и отображает модифицированную запись. |

3.2.3

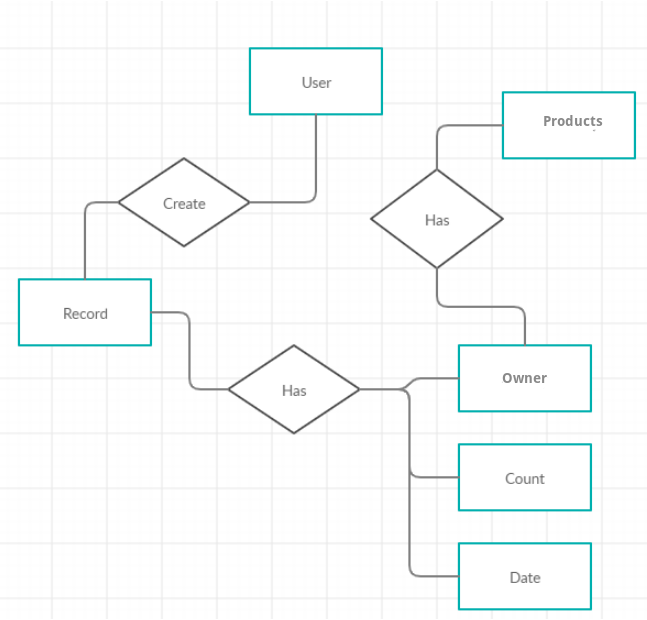
|  |  |
| --- | --- |
| Use Case Name | Удалить существующую запись |
| Trigger | Пользователь выбирает удаление запись. |
| Path | 1.Система предоставляет список всех имеющихся записей в порядке добавления.  2.Пользователь выбирает запись.  3.В открывшемся окне информации о записи, пользователь нажимает кнопку “Удалить”.  4.Система предоставляет информацию про запись и просит подтверждение удаления.  5. Пользователь подтверждает удаление. |

3.2.4

|  |  |
| --- | --- |
| Use Case Name | Продажа продуктов |
| Trigger | Покупка товара на сайте |
| Path | 1.Система предоставляет список всех имеющихся записей в порядке добавления.  2.Пользователь выбирает запись с продуктом.  3.В открывшемся окне информации о записи, пользователь нажимает кнопку “Приобрести товар”. |

* 1. **Логические требования к базе данных**

Логическая структура данных, хранимых в базе данных представлена ниже:



**Описание таблиц из базы данных DB:**

Таблица users:

|  |  |  |
| --- | --- | --- |
| Data Item | Type | Description |
| username | text | имя пользователя |
| type | text | тип пользователя (покупатель/продавец) |
| ID | integer | ID пользователя |

Таблица products:

|  |  |  |
| --- | --- | --- |
| Data Item | Type | Description |
| Data | date | дата добавления записи |
| Count | integer | количество товара |
| productName | text | название продукта |

* 1. **Безопасность**

Сервер, на котором будет находится продукт, будет иметь собственную защиту от несанкционированного доступа к записи / удалению.

**Приемочные тесты**

Тест 1: “Добавление записи”

Цель тестирования:

* проверить корректность работы кнопки добавления записи
* проверить корректность вводимых данных для добавления
* проверить корректность добавления записи

Ожидаемый результат:

* кнопка работает корректно
* все данные вводятся корректно
* запись успешно добавляется

Тест 2: “Изменение записи”

Цель тестирования:

* проверить корректность работы кнопки изменения записи
* проверить корректность ввода изменяемых данных
* проверить корректность изменения записи

Ожидаемый результат:

* кнопка работает корректно
* все данные изменяются корректно
* запись успешно изменяется

Тест 3: “Удаление записи”

Цель тестирования:

* проверить корректность работы кнопки удаления записи
* проверить корректность подтверждения удаления записи
* проверить корректность удаления записи

Ожидаемый результат:

* кнопка работает корректно
* подтверждение удаления работает корректно
* запись успешно удаляется

Тест 4: “Просмотр данных”

Цель тестирования:

* проверить корректность вывода всех записей в списке
* проверить корректность сортировки данных по последней добавленной записи

Ожидаемый результат:

* записи выводятся корректно
* записи находятся в порядке последнего добавлени

Тест 5: “Продажа товара”

Цель тестирования:

* проверить корректность работы вывода доступного кол-ва товара
* проверить корректность обновление количества товара после покупки

Ожидаемый результат:

* кол-во товаров выводится корректно
* подсчёт нового кол-ва товаров и его продажа осуществляется успешно

**Вывод**: в процессе лабораторной работы были рассмотрены требования к документации, разработано описание документации согласно стандарту IEEE830 SRS та написаны собственные приемочные тесты.