**Методические указания по выполнению итогового проекта**

по дисциплине «Технологии разработки программных приложений»

**Часть 1. Система контроля версий и оформление кода (Оценивается в 2 балла).**

1. Выберите тему проекта: программное приложение, которое вы будете разрабатывать в семестре. Возможные направления (звездочкой отмечены направления, отсутствующие в учебном плане до 4 семестра):
   1. WEB-разработка
   2. Мобильная разработка
   3. Разработка под 1С
   4. Разработка базы данных
   5. Desktop приложение
   6. \*Интернет вещей (IoT)
   7. \*GameDev
   8. \*Разработка бот-сервиса

Тема проекта может пересекаться с темами практических работ или проектов по другим дисциплинам этого семестра. Проект должен быть достаточно простым (разработка не должна занять больше 10-15 часов на каждого члена команды).

Проекты должны быть ориентированы под Linux-среды (во избежание дальнейших сложностей с интеграцией программного приложения).

1. Определитесь с составом команды. В команде может быть от 2 до 3 человек в зависимости от сложности темы проекта. Каждый член команды будет выступать в роли разработчика.
2. Сформулируйте функциональные требования к своему проекту. Требования должны быть достаточными для описания программного продукта, который вы будете разрабатывать. Рекомендуется оформить требования в виде User Story.

Структура User Story: *Как, <роль/персонаж юзера>, я <что-то хочу получить>, <с такой-то целью>*

Пример: *Мне как потребителю нужен крупнейший в мире магазин книг, где я могу купить любую книгу в любое время.*

Фрагментация истории (необходимо сделать ее конкретной и функциональной):

*Как потребителю мне удобно искать книги по жанрам, чтобы быстро найти те, которые я люблю читать.*

*Как потребитель я, отбирая книги для покупки, хочу класть сразу каждую в корзину.*

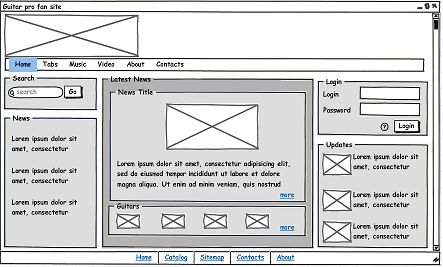
*Как управляющий по выпуску новой продукции я хочу иметь возможность отслеживать покупки наших клиентов, чтобы быть в курсе, какие книги им можно предлагать.*

Оформите User Story в виде карточек, например:

|  |  |  |
| --- | --- | --- |
| **Заголовок** | | Контактная информация о статье |
| **Заказчик (actor)** | Как | журналист |
| **Примечание** | Я хочу | увидеть контактную информацию, относящуюся к статье, которую я читаю |
| **Цель** | Чтобы | получить немедленный доступ к пресс-офису по вопросам, связанным со статьей |

Требования могут изменяться на протяжении работы над проектом.

1. Придумайте интерфейс разрабатываемого продукта, отразите его в виде макета , например:



1. Создайте репозиторий для своего проекта и обеспечьте к нему удаленный доступ для всех членов команды.
2. Начинайте разработку проекта. Снабжайте программный код разработанных классов, методов, функций и т.д. комментариями. Код должен быть легко читаем. В процессе разработки программного продукта необходимо пользоваться выбранной системой контроля версий.

**Литература для изучения:**

1. Как писать User Story:

<https://medium.com/@alexandertvar/%D0%BA%D0%B0%D0%BA-%D0%BF%D0%B8%D1%81%D0%B0%D1%82%D1%8C-user-story-2410093b23c2> .

1. Как составлять хорошие User Story: <https://cmsmagazine.ru/journal/items-how-to-make-a-good-user/>
2. Работаем с User stories: Руководство:

<https://habr.com/ru/company/friifond/blog/284032/>

1. Инструменты для создания макета проекта <https://cameralabs.org/2519-15-udobnich-prilozheniy-dlya-sozdaniya-maketa-proekta>
2. Документирование кода в Python. PEP 257

<https://pythonworld.ru/osnovy/dokumentirovanie-koda-v-python-pep-257.html>

1. Комментарии, без которых можно обойтись

<https://habr.com/ru/company/enterra/blog/182574/>

1. Советы Google по кодированию на языке Python <https://habr.com/ru/post/180509/>
2. О красоте кода <https://habr.com/ru/company/yandex_praktikum/blog/528070/>
3. Руководство по оформлению HTML/CSS кода от Google <https://habr.com/ru/post/143452/>