МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

УО «БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ»

Индивидуальное практическое задание №2

по дисциплине «Проектирование программного обеспечения»

Вариант 1

«Управление базами данных в Android»

Студент

ФИНО, ИИТП, 793551, 3 курс И. В. Галкин

Преподаватель Ю. И. Алексеев

МИНСК 2019

**Содержание**

[1 Постановка задачи 3](#_Toc29147145)

[2 Структура и основные классы ПС 4](#_Toc29147146)

[3 Скриншоты результатов работы программы 6](#_Toc29147147)

# **1 Постановка задачи**

В соответствии с индивидуальным заданием создать приложение, которое позволяет просматривать и редактировать списки различных объектов (товаров, сотрудников и т.п.). Данные для списков хранить в базе данных SQLite

Список товаров (название товара, тип товара, цена). Создание, редактирование, удаление.

# **2 Структура и основные классы ПС**

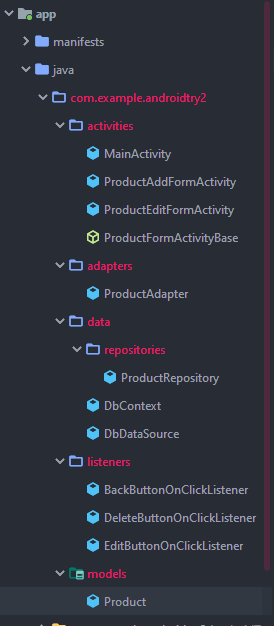


Рисунок 1 Структура проекта ч.1

* **MainActivity**– класс активности со списком продуктов. Содержит методы по инициализации компонентов и загрузки данных из базы данных.
* **ProductFormActivityBase** – абстрактный класс активности, описывающий логику работы с полями ввода на формах создания и редактирования продукта.
* **ProductAddFormActivity** – активность на которой создается продукт. Наследуется от ProductFormActivityBase. Внутри описывается логика работы кнопок формы: кнопки вернуться назад и кнопки подтвердить создание.
* **ProductEditFormActivity** – активность на которой изменяется продукт. Наследуется от ProductFormActivityBase. Внутри описывается логика работы кнопок формы: кнопки вернуться назад и кнопки подтвердить изменение.
* **ProductAdapter** – адаптер списка продуктов. Описывает логику соотношения данных из базы данных к записям в таблице продуктов. Также описывает добавление кнопок действий к этим же записям.
* **DbContext** – класс-посредник между приложением и базой данных. Содержит методы создания структуры базы данных и обработку версионности. А также содержит названия таблиц и их столбцов в базе данных.
* **DbDataSource** – класс-посредник между бизнес-логикой приложения и методами доступа к базе данных.
* **ProductRepository –** класс, содержащий методы доступа к таблице Product. Реализует методы получения, добавления, удаления и изменения записей в этой таблице.
* **BackButtonOnClickListener –** обработчик нажатия на кнопку «назад» в форме создание/изменения продукта.
* **DeleteButtonOnClickListener –** обработчик нажатия на кнопку «удалить» в списке продуктов.
* **EditButtonOnClickListener-** обработчик нажатия на кнопку «изменить» в списке продуктов.
* **Product –** класс, описывающий все характеристики продукта и содержащий методы доступа к ним.

# **3 Скриншоты результатов работы программы**

1. **Экран при запуске приложения**

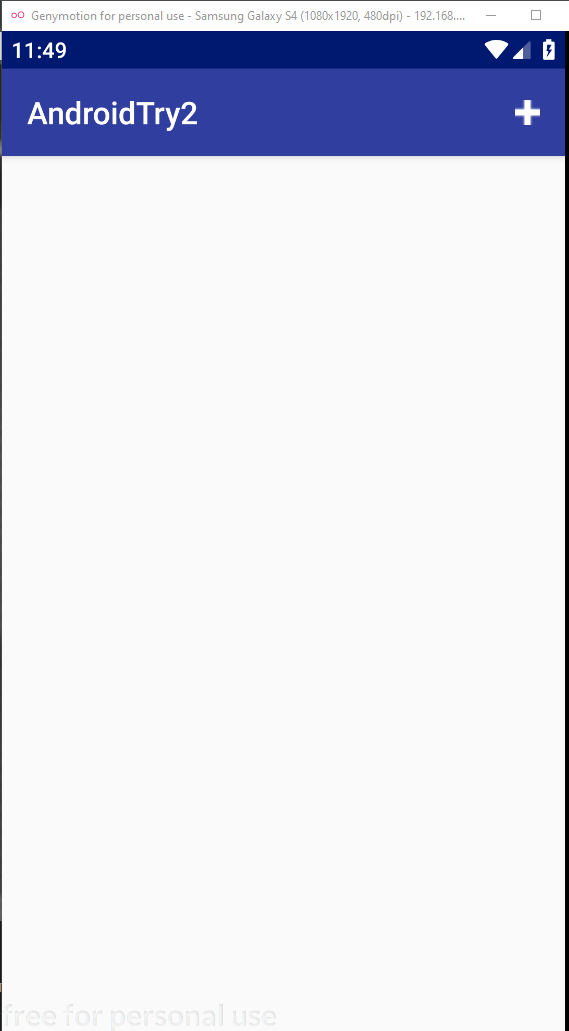


Рисунок 2 Экран при запуске приложения

1. **Форма создания нового продукта**

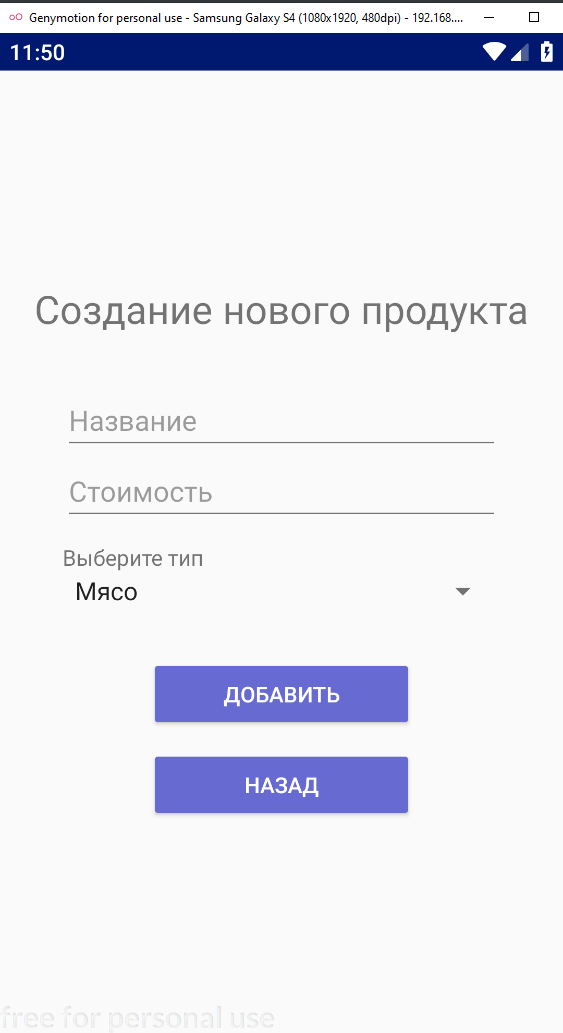


Рисунок 3 Форма создания нового продукта

1. **Список после добавления продукта**

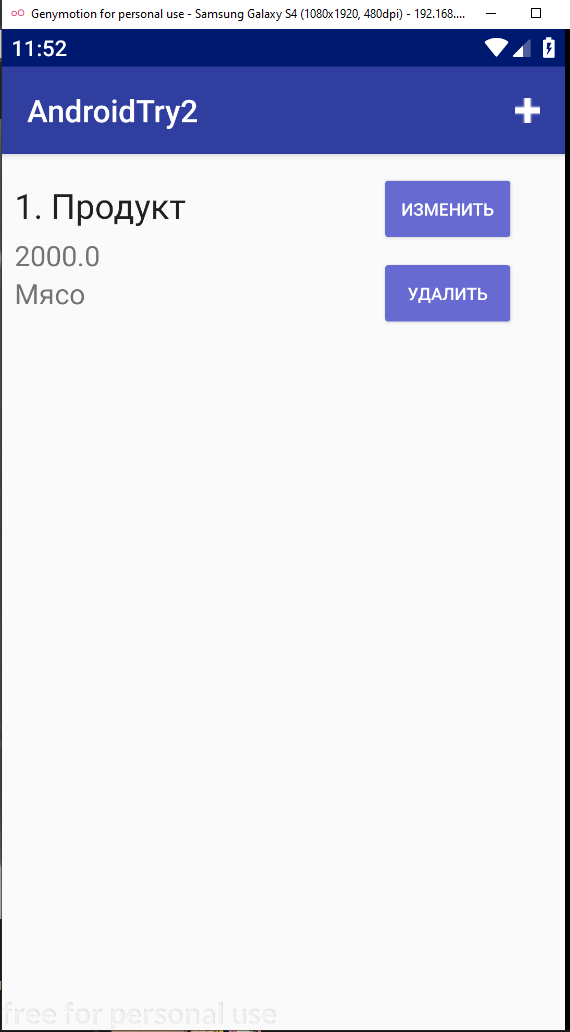


Рисунок 4 Список после добавления продукта

1. **Форма изменения продукта**

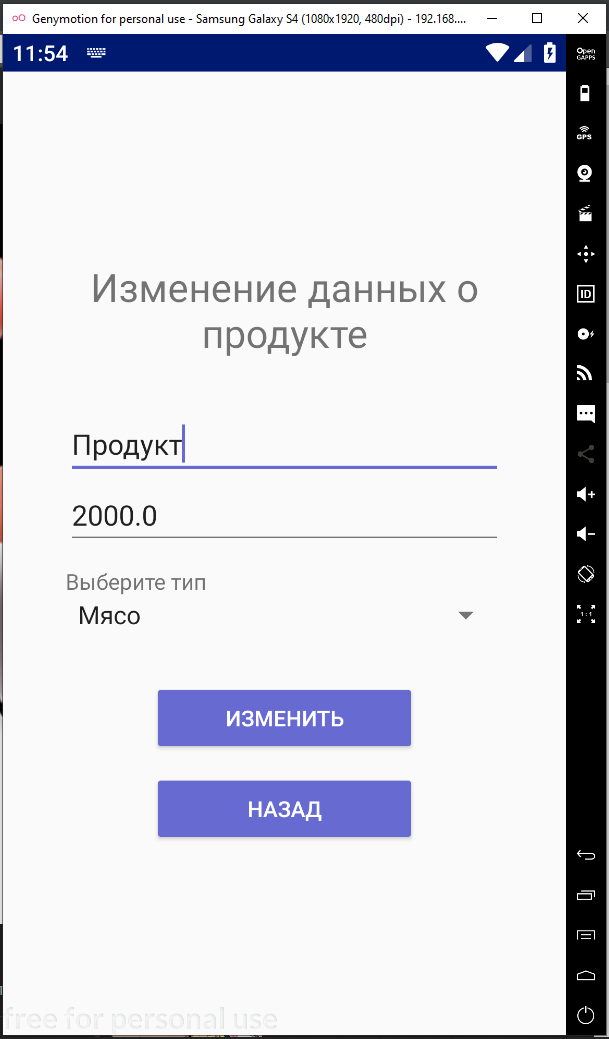


Рисунок 5 Форма изменения продукта

1. **Список после изменения**

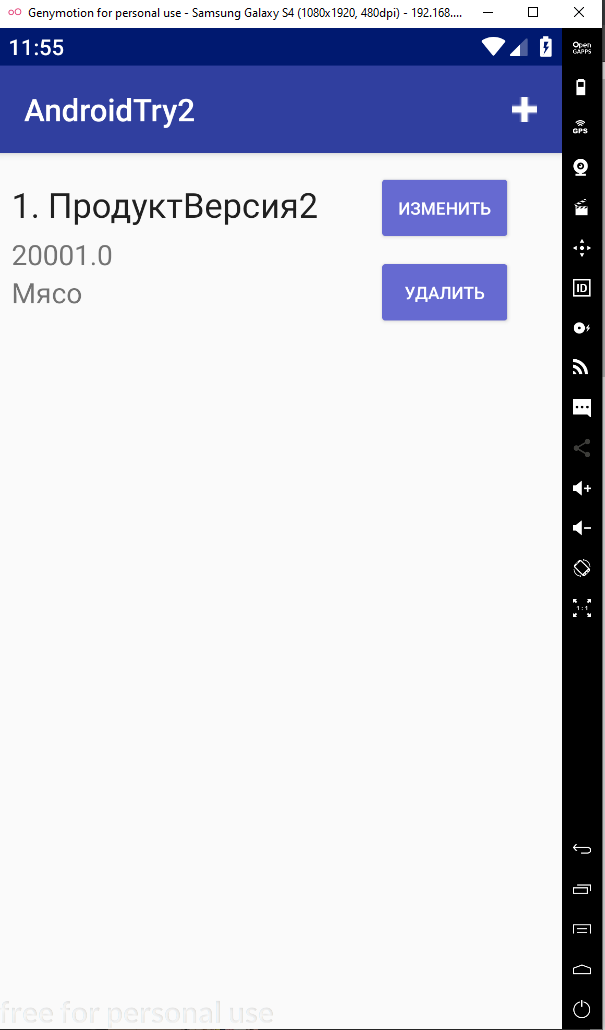


Рисунок 6 Список после изменения

1. **Список до удаления продукта**

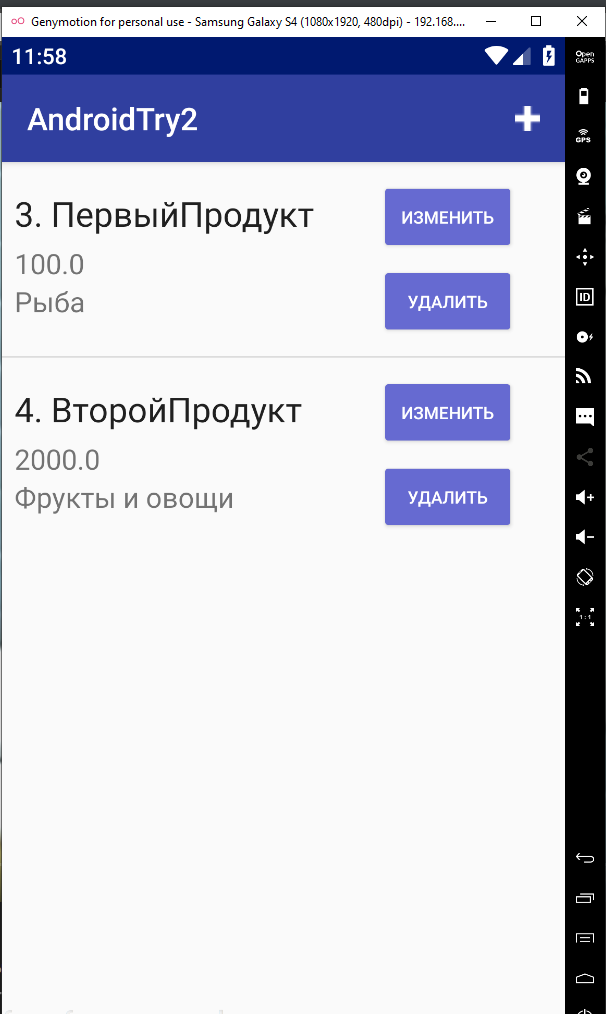


Рисунок 7 Список до удаления продукта

1. **Список после удаления продукта**

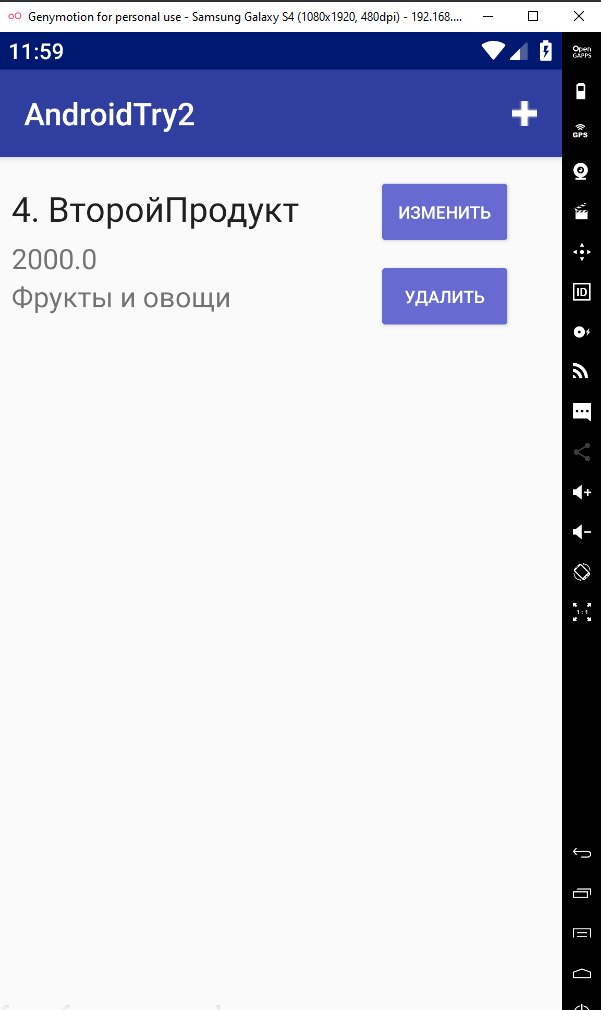


Рисунок 8 Список после удаления продукта

1. **Поддержка ландшафтного режима**

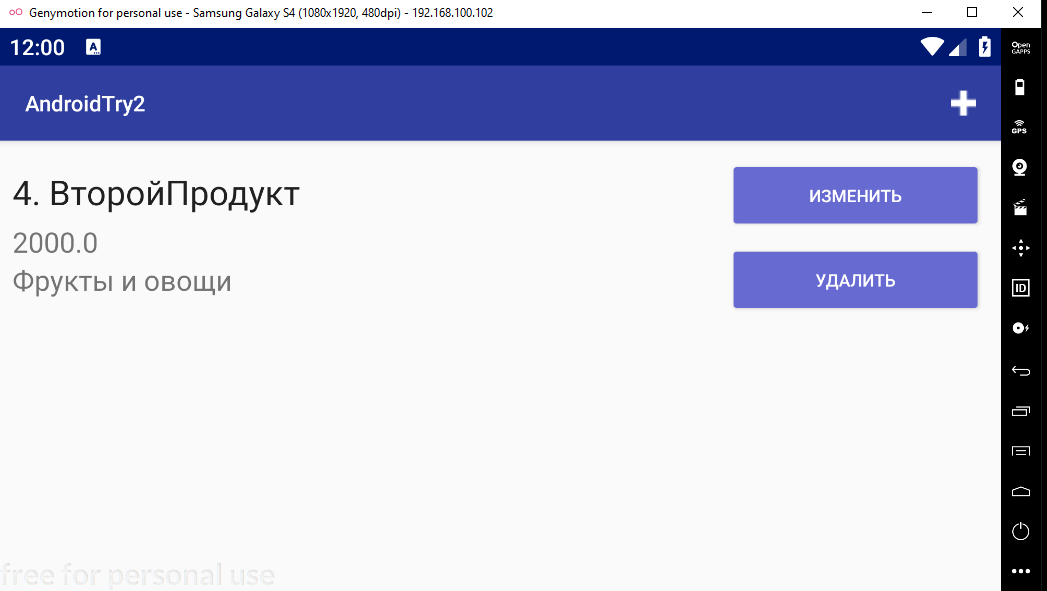


Рисунок 9 Поддержка ландшафтного режима на странице со списком

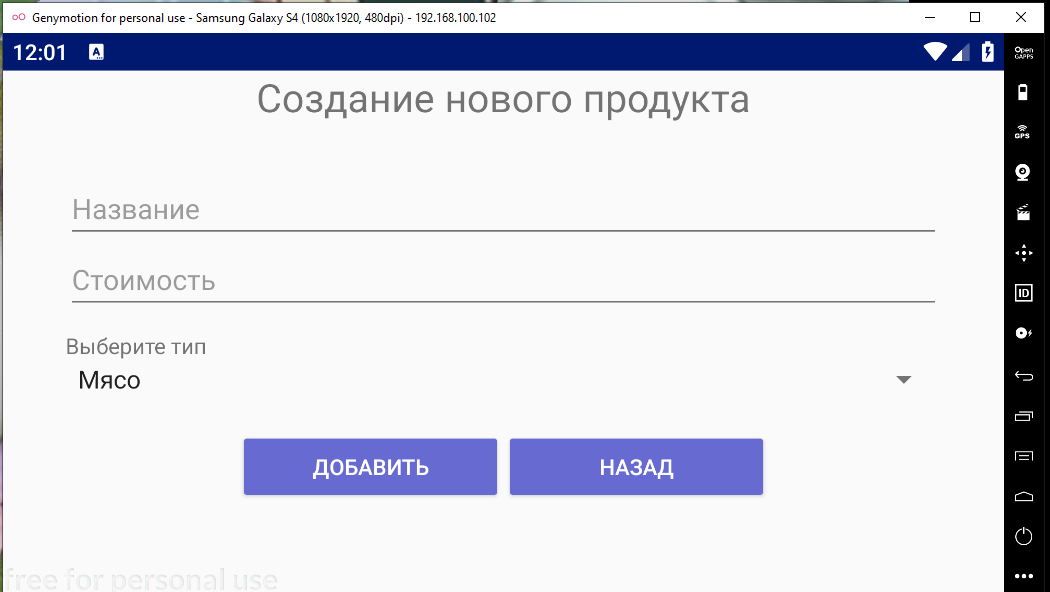


Рисунок 10 Поддержка ландшафтного режима на форме создания

1. **Поддержка нескольких локализаций**

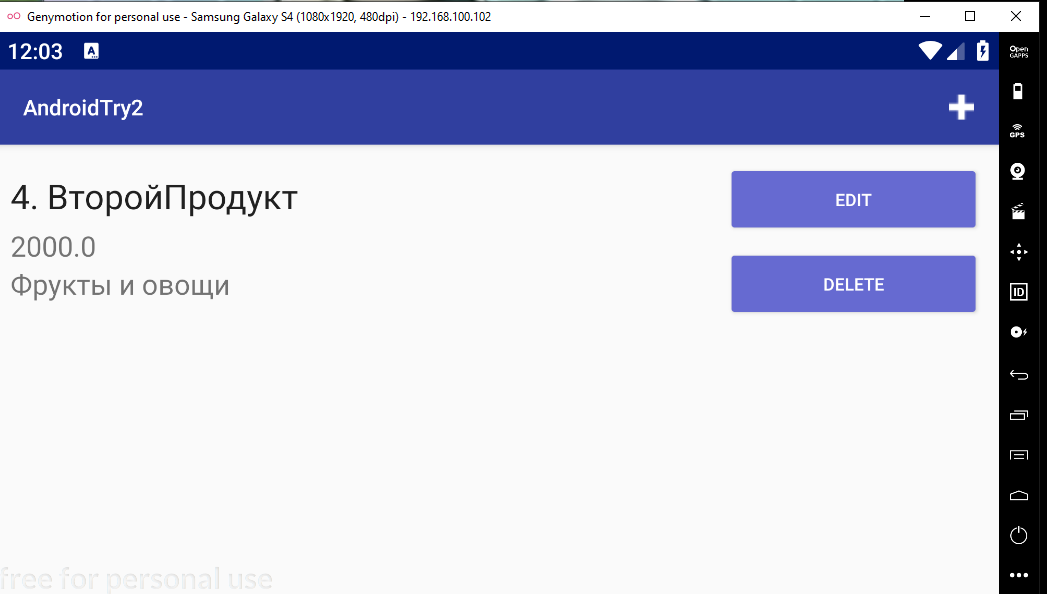


Рисунок 11 Поддержка английской локализации

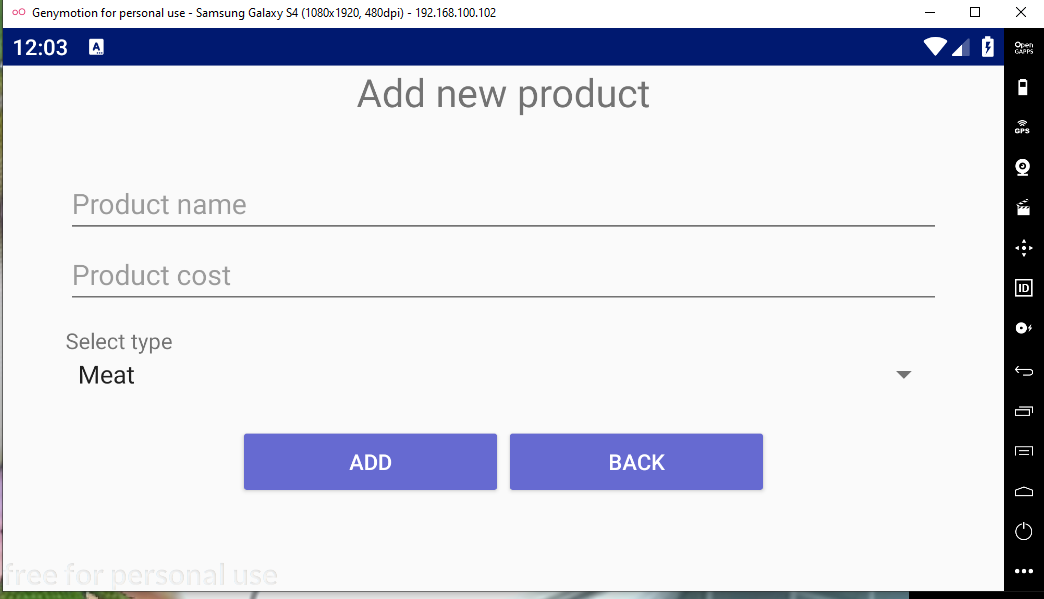


Рисунок 12 Поддержка английской локализации