Варианты проекта:

* Картотека электронных книг. *Элементы данных: книга – название, ISBN, авторы, количество страниц, тег(и), год издания, издательство; автор – имя, фото, дата рождения, список книг; издательство – название, город, список книг.*
* Каталог фильмов пункта проката. *Элементы данных: фильм – название, постер, жанр(ы),возрастное ограничение, актеры, режиссер, год выпуска; актер – имя, дата рождения, краткая биография, список фильмов; режиссер – имя, список фильмов.*
* Система тестирования знаний (с редактором тестов). *Элементы данных: тест – название, предмет, количество времени, режим измерения времени (на весь тест, на каждый вопрос, …), список вопросов; вопрос – текст вопроса, ценность в баллах, тест, к которому он относится; вопрос первого типа – дополнительно варианты ответа, некоторые из которых правильные; вопрос второго типа – дополнительно левый и правый список, между элементами которых нужно установить правильное соответствие; вопрос третьего типа – без вариантов ответа, предусматривающий текстовый ответ.*
* Система управления группами и студентами на факультете. *Элементы данных: студент – фамилия, имя, отчество, фото, год первого поступления, оценки, группа, группы, в которых он обучался раньше, и периоды обучения в других группах; группа – номер, староста (студент), имена преподавателей в каждом семестре по каждому предмету, расформирована ли.*
* Складской учёт компьютеров и компьютерных комплектующих. *Элементы данных: компьютер – инвентарный номер, материнская плата, блок питания, винчестеры, приводы оптических дисков; материнская плата – серийный номер, производитель, модель, тип сокета процессора, установленный процессор, количество и типы слотов ОЗУ, установленные ОЗУ, установленная видеокарта, список внешних разъемов; блок питания – серийный номер, производитель, номинальная мощность; винчестер – серийный номер, производитель, емкость, тип SATA; привод оптических дисков – серийный номер, производитель, тип (CD-R, DVD-RW, …); процессор – серийный номер, производитель, модель, тип сокета, тактовая частота; ОЗУ – серийный номер, производитель, тип, объем, частота; видеокарта – серийный номер, производитель, модель; дополнительно все комплектующие – инвентарный номер компьютера, в котором они установлены.*
* Программа для управления персоналом предприятия. *Элементы данных: работник – фамилия, имя, отчество, фото, должность, дата приема на работу на предприятие и дата увольнения, предыдущие должности и время, проведенное на этих должностях, отдел, начальник работника и его подчиненные из других отделов; отдел – название, год образования, описание деятельности, начальник.*
* Каталог данных о звёздах и планетах. *Элементы данных: созвездие – название, изображение, положение на звездной карте, звезды; звезда – название, радиус, масса, светимость, тип, планеты-спутники; планета – название, радиус, масса, период обращения вокруг своей оси, тело, вокруг которого планета вращается, период ее обращения вокруг этого тела, радиус орбиты.*
* Генеалогическое дерево (редактирование и просмотр). *Элементы данных: человек – фамилия, имя, отчество, пол, дата рождения, фото, муж/жена, мать, отец, дети, братья, сестры, двоюродные братья, двоюродные сестры (последние четыре – только для чтения).*

**Лабораторная работа 1. Создание модели и классов для коллекций**

**Постановка задачи**. Необходимо выполнить следующее:

1) Разработать функциональные требования к выбранному проекту – определить список возможностей, которые будут реализованы программой.

2) Создать классы для представления элементов данных и коллекций, с которыми работает приложение. Реализовать стандартные операции для коллекций (добавление, удаление, поиск элемента), а также операции, специфичные для предметной области. Реализовать интерфейс IEnumerable<T> для коллекций. Рассмотреть возможность использования нестандартного базового класса для создания коллекции (KeyedCollection, OrderedDictionary и т. п.).

**Лабораторная работа 2. Создание приложения с помощью Xamarin.Forms**

**Постановка задачи**. Необходимо выполнить следующее:

1) Разработать общую архитектуру проекта, в том числе определить необходимый набор страниц и моделей представлений для приложения Xamarin.Forms.

2) Создать главную страницу приложения, а также хотя бы одну дополнительную. Главная страница должна обладать базовыми возможностями по отображению данных коллекции. Страницы, не связанные с главной, можно демонстрировать с помощью Xamarin Forms Previewer.

**Лабораторная работа 3. Компоновка и связывание с данными**

**Постановка задачи**. Необходимо создать и реализовать на базовом уровне все страницы приложения. На страницах должна быть реализована компоновка представлений, а передача им данных должна осуществляться с помощью привязки данных. По желанию можно продумать и реализовать компоновку, учитывающую оба различных положения мобильного устройства.

**Лабораторная работа 4. Навигация между страницами**

**Постановка задачи**. Необходимо выполнить следующее:

1) Заменить на некоторых страницах ContentPage на другой подходящий тип страницы.

2) Реализовать модель навигации, не нарушающую принципа разделения кода между представлением и моделью представления (или сводящую такие нарушения к минимуму). На основе созданной модели необходимо реализовать в страницах приложения функциональность перехода от одной страницы к другой.

**Лабораторная работа 5. Отображение коллекций и таблиц**

**Постановка задачи**. Необходимо воспользоваться на страницах приложения представлениями ListView и TableView для отображения коллекций и форм ввода. В TableView необходимо воспользоваться ViewCell для оформления хотя бы одной ячейки таблицы. По желанию можно использовать в приложении дополнительные возможности этих представлений, такие, как контекстные меню.

**Лабораторная работа 6. Использование технологии LINQ to Objects**

**Постановка задачи**. Необходимо реализовать с помощью LINQ to Objects методы для выборки, упорядочения и группировки данных в классах модели, вычисления агрегатных функций. Воспользоваться LINQ-запросами для отображения в приложении необходимой части данных. По желанию сделать свою реализацию оператора упорядочения, группировки, объединения или работы с множествами, применяя концепцию отложенных вычислений там, где это возможно.

**Лабораторная работа 7. Работа с файловой системой**

**Постановка задачи**. Необходимо реализовать методы для сохранения и чтения данных приложения с помощью текстовых и двоичных файлов. Применить при чтении/за­писи потоки, поддерживающие сжатие информации (по желанию – дополнительно шифровать информацию). По желанию можно реализовать в приложении функцию «Открыть в…» (переслать приложению файл, находящийся в распоряжении другого приложения).

**Лабораторная работа 8. Работа с ресурсами и конфигурацией**

**Постановка задачи**. Необходимо разместить в ресурсах и использовать в приложении экран заставки, а также необходимый набор иконок. Необходимо выделить несколько настроек приложения и сохранить их в файл (формат файла – XML или JSON). При запуске приложение должно считывать свои настройки из файла и применять их, при закрытии – записывать текущие настройки в файл. Одной из этих настроек должен быть язык приложения (приложение должно поддерживать хотя бы два языка).

**Лабораторная работа 9. Графика и анимация**

**Постановка задачи**. Необходимо создать визуальный стиль для некоторых (минимум двух) элементов управления. Создать анимации для элементов управления. Реализовать элементы 2D-графики / трансформаций / графических эффектов (можно представить график или диаграмму, построенную по неким данным приложения; нарисовать логотип или векторный рисунок). По желанию можно дополнительно:

* реализовать в приложении поддержку жестов, стандартных или собственных;
* реализовать работу с аппаратным обеспечением устройства, таким, как камера, геолокация и т. д.