1. **Реализовать интерфейс взаимодействия с API bin’ов банковских карт** (bin – первые 6 цифр банковской карты, однозначно определяют принадлежность карты стране и банку).

Сценарий использования:

* 1. Пользователь вводит в форме ввода БИН-карты 6 знаков номера банковской карты для проверки принадлежности к банку и нажимает кнопку "ПРОВЕРИТЬ"
  2. Приложение делает запрос в собственную базу данных и, если не находит в ней соответствия данных к текущему BIN, то делает запрос к API binlist.net, получает ответ, который отображает клиенту и записывает в собственную БД.
  3. Ответ на текущий запрос показывается под формой HTML в читаемой таблице.
  4. Справа от формы ввода показывать 10 последних проверенных БИНов (10 последних строк из БД)

1. **Моделирование работы автобусного парка**.   
   Сведения о каждом автобусе содержат: номер автобуса, фамилию и инициалы водителя, номер маршрута. Программа должна обеспечивать выбор с помощью меню и выполнение одной из следующих функций:
   1. Начальное формирование данных о всех автобусах в парке в виде списка (из файла);
   2. Имитация выезда автобуса из парка: вводится номер автобуса; программа удаляет данные об этом автобусе из списка автобусов, находящихся в парке, и записывает эти данные в список автобусов, находящихся на маршруте;
   3. Имитация въезда автобуса в парк: вводится номер автобуса; программа удаляет данные об этом автобусе из списка автобусов, находящихся на маршруте, и записывает эти данные в список автобусов, находящихся в парке;
   4. Вывод сведений об автобусах, находящихся в парке, и об автобусах, находящихся на маршруте, упорядоченных по номерам автобусов;
   5. Вывод сведений об автобусах, находящихся в парке, и об автобусах, находящихся на маршруте, упорядоченных по номерам 10 маршрутов.
2. **Парсер музыкальных файлов.**Пользователь загружает в определенный каталог файлы формата «.mp3», после чего система должна их «подхватить» – получить из каждого файла информацию о названии, альбоме, жанре и т. п. с дальнейшей загрузкой информации в базу данных. Дополнительно необходимо реализовать запрос для получения списка последних 10 загруженных файлов.
3. **Программа учета заявок на авиабилеты.**  
   Каждая заявка содержит – пункт назначения, номер рейса, фамилию и инициалы пассажира, желаемую дату вылета. Программа должна обеспечивать выбор с помощью меню и выполнение одной из следующих функций:
   1. Добавление заявок в список;
   2. Удаление заявок;
   3. Вывод заявок по заданному номеру рейса и дате вылета;
   4. Вывод всех заявок, упорядоченных по пунктам назначения;
   5. Вывод всех заявок, упорядоченных по датам вылета.
4. **База данных студентов.**Система должна иметь следующие функции:
   1. Отображение списка групп;
   2. Добавление новой группы, редактирование и удаление существующей;
   3. Отображение списка студентов;
   4. Фильтрация списка студентов по фамилии и по номеру группы;
   5. Добавление нового студента, редактирование и удаление существующего.
5. **HTTP-сервис для сокращения URL** (наподобие Bitly и других сервисов)

UI не нужен, достаточно сделать JSON API сервис. Должна быть возможность:

* 1. Сохранить короткое представление заданного URL;
  2. Перейти по сохраненному ранее короткому представлению и получить redirect на соответствующий исходный URL.

1. **Калькулятор валют**. Калькулятор должен уметь работать с валютами — доллар, евро, юани и рубль — и позволять выполнять операции сложения и вычитания.  
   Складывать и вычитать можно только значения в одной валюте, должна быть реализована операция конвертации из одной валюты в другую по курсу, который задается во внешнем файле конфигурации.  
   Необходимо реализовать поддержку не только целых, но и дробных значений. Так, значение в долларах обозначается символом $, расположенным перед числом (например, $57,75), значение в рублях — символом “р”, расположенным после числа (например, 57,75р). Пример операции конвертации долларов в рубли — toRubles ($57,75), рублей в доллары — toDollars (57,75р). Разделитель целой и дробной части может быть любым на выбор разработчика. Пример консольного ввода: toDollars (737р + toRubles ($85,4)). Для данного ввода вывод должен быть (при курсе 60 рублей за доллар, например): 5861р.
2. **MyTop100Movies**.   
   Приложение позволяет пользователю составить топ из 10 любимых фильмов. Спарсить фильмы нужно либо из The Movie Database (http://tmdb.org), либо с сайта сервиса Кинопоиск (https://www.kinopoisk.ru/). Должны быть реализованы следующие функции:
   1. Добавление фильма;
   2. Топ-10 любимых фильмов;
   3. Выставление оценки фильму.
3. **Online checker.**   
   Приложение, отображающее число пользователей, находящихся в данный момент онлайн. Для этого воспользуйтесь одним из key-value хранилищ, например, Memcached или Redis. Считается, что пользователь находится онлайн, если он запрашивал одну из страниц сайта в течение последних пяти минут. Страниц у приложения должно быть несколько, при этом должна обязательно быть реализована страница **/logout**, при переходе на которую пользователь выходит из онлайна.