**ООО «Системы МАСП»**

**Проект MapBulApp**

**ОПИСАНИЕ СТРУКТУРЫ**

Исполнитель: Копытин М. В.

2016 г.

Оглавление

[Общее описание 3](#_Toc459153329)

[Проект Map\_Bul\_App 4](#_Toc459153330)

[Пространство имен Models 5](#_Toc459153331)

[Модели базы данных 5](#_Toc459153332)

[Пространство имен ViewModel 5](#_Toc459153333)

[BaseViewModel 5](#_Toc459153334)

[MenuViewModel 6](#_Toc459153335)

[TKMapViewModel 6](#_Toc459153336)

[PinDetailsViewModel 7](#_Toc459153337)

[FilterViewModel 8](#_Toc459153338)

[LoginViewModel 9](#_Toc459153339)

[RegistrationViewModel 9](#_Toc459153340)

[AddNewPinViewModel 9](#_Toc459153341)

[ArticlesViewModel 11](#_Toc459153342)

[ArticleDetailsViewModel 12](#_Toc459153343)

[FavoritesViewModel 12](#_Toc459153344)

[Пространство имен ViewModel.Design 13](#_Toc459153345)

[Пространство имен Views 13](#_Toc459153346)

[Пространство имен Settings 14](#_Toc459153347)

# Общее описание

Приложение использует 2 решения:

* *Map\_Bul\_App.sln* – решение самого приложения X-island.
* *TK.CustomMap.sln* – решение кастомизации карты, позволяет использовать привязки MVVM, настраивать отображения превью меток, перегружать стандартные действия с картой.

# Проект Map\_Bul\_App

В данном проекте описана основная логика приложения. Целевой файл сборки проекта – Map\_Bul\_App.dll, подключаемый в проекты *Android* и *iOS*.

Сборка состоит из следующих пространств имен:

* *Models.* В данном пространстве имен находятся описания классов моделей для приложения.
  + *Tables.* В данном пространстве имен находятся модели для таблиц базв данных.
* *Converters.* В данном пространстве имен находятся конвертеры значений для привязок MVVM.
* *Design.* В данном пространстве имен находятся классы, описывающие *UI* – элементы приложения.
* *ResX*. Здесь находятся файлы с локализованными текстами приложения.
* *Settings*. В данном пространстве имен описаны служебные классы приложения.
* *ViewModel*.В данном пространстве имен описаны вьюмодели представлений приложения.
* *Views*.В данном пространстве имен описаны страницы приложения.

## Пространство имен Models

* ArticleEventItem – класс, описывающий модель статьи или метки.
* LoadedPins – класс, описывающий набор загруженных меток и набор регионов, для которых метки уже загружены.
* PinCategory – класс, описывающий модель категории метки/статьи.
* PinDescriptor – класс, описывающий модель метки.
* UserPinDescriptor – расширенная модель метки.

### Модели базы данных

* User – класс, описывающий модель пользователя приложения.
* Session – класс, описывающий модель сессии приложения, нужен для синхронизации загруженных меток с сервером и экономии трафика.
* Place – класс, описывающий модель метки, добавленной в Избранное.
* ArticleEvent – класс, описывающий модель статьи или события, добавленной в Избранное.
* FilterSettings – класс, описывающий последнее состояние фильтра меток.

## Пространство имен ViewModel

BaseViewModel – базовая вьюмодель реализует интерфейс INotifyPropertyChanged, от нее наследуются все остальные вьюмодели

Поля:

* + public Color BackgroundColorOfActivityIndicator – цвет фона загрузки.
  + public bool IsActivityIndicatorVisible – флаг видимости индикатора загрузки.
  + public bool IsActivityIndicatorRuning – флаг активности индикатора загрузки.
  + public bool IsInitialize – флаг, показывающий, что вьюмодель инициализирована.

Методы:

* + public Color BackgroundColorOfActivityIndicator – цвет фона загрузки.
  + public void Initialize – метод инициализации вьюмодели.
  + public abstract void InitilizeFunc – метод инициализации вьюмодели для переопределения в наследниках.
  + public void StartLoading() – метод запуска индикатора загрузки.
  + public void StoptLoading() – метод остановки индикатора загрузки.

Команды:

* + public ICommand MenuShowCloseClickCommand – команда нажатия на иконку меню, открывает боковое меню.
  + public ICommand BackClickCommand – команда нажатия на иконку «назад», возвращает пользователя на предыдущую страницу.

MenuViewModel – вьюмодель бокового меню

Поля:

* + public bool IsFavoritesVisible – флаг видимости пункта «Избранное».

Методы:

* + private static void OnMenuItemClicked – метод перехода на выбранный экран.
  + private void CurrentUser\_PropertyChanged – метод изменения контента меню после измемнения данных текущего пользователя.
  + public abstract void InitilizeFunc – метод инициализации вьюмодели для переопределения в наследниках.
  + public void StartLoading() – метод запуска индикатора загрузки.
  + public void StoptLoading() – метод остановки индикатора загрузки.

Команды:

* + public ICommand LoginLogoutCommand – команда переход на экран авторизации
  + public ICommand MenuItemClickedCommand – команда нажатия на пункт меню

TKMapViewModel – вьюмодель основного экрана приложения (карта)

Поля:

* + public FilterPage Filter – текущая страница фильтра меток.
  + public MapSpan MapRegion – текущая область карты, отображаемая на экране.
  + public ObservableCollection<TKCustomMapPin> Pins – текущая коллекция меток на карте.
  + public Position MapCenter – текущий центр карты.
  + public TKCustomMapPin SelectedPin – выбранная метка.
  + public bool IsPreviewVisible – флаг открытого превью метки.
  + public string FindText – текст строки поиска.

Команды:

* + public ICommand OpenFilterCommand – команда открытия экрана фильтра меток.
  + public ICommand GoToUserLocation – команда перехода карты к местоположению пользователя (iOS).
  + public ICommand FindByTagCommand – команда поиска меток по строке *FindText.*
  + public ICommand LoadMarkersCommand – команда загрузки меток в области *MapRegion.*
  + public ICommand MapLongPressCommand – команда , срабатывающая по долгому нажатию на карте (добавление новой метки).
  + public Command PinSelectedCommand – команда, срабатывающая по нажатию на метку (переход центра карты в местоположение метки).
  + public Command<Position> MapClickedCommand – команда нажатие на чистую область карты (снятие фокуса с выделенной метки).
  + public Command CalloutClickedCommand – команда нажатия на превью метки (открытие экрана метки).
  + public Command ClearMapCommand – команда удаления всех меток с карты (не используется).
  + public ICommand DragEndCommand – команда окончания перетаскивания метки (не используется).

Методы:

* + private void OpenFilter() – функция открытия фильтра меток.
  + private void ShowFavorites() – функция показа на карте избранных меток.

### PinDetailsViewModel

Вьюмодель экрана метки.

Поля:

* + private DeserializeGetMarkerDetailsData PinFromServer – vмодель метки, полученной с сервера.
  + public UserPinDescriptor Pin – метка на карте.
  + public ImageSource Icon – иконка категории.
  + private TimeSpan? OpenTime – время открытия.
  + private TimeSpan? CloseTime – время закрытия.
  + public string WorkTime – строковое представление часов работы.
  + public bool IsFavorite – флаг метки из избранного.
  + public Position MapCenter – центр карты.
  + public ObservableCollection<TKCustomMapPin> Pins – метки на карте.

Команды:

* + public CommandAddToFavoritesCommand – команда добавления/удаления метки в избранное.

События:

* + public event EventHandler PinDeleted – событие удаления метки из избранного.

FilterViewModel – вьюмодель страницы фильтра меток.

Поля:

* + public MapSpan MapRegion – текущая область карты, отображаемая на экране.
  + public List<PinDescriptor> Pins – текущая коллекция меток в заданной области *MapRegion.*
  + public List<PinDescriptor> SelectedPins – коллекция меток , подходящих под условия фильтра.
  + public int SelectedPinsCnt – размер коллекции *SelectedPins*.
  + public List<PinCategory> PinCategories – коллекция категорий меток.
  + public List<PinCategory> PinSubCategories – коллекция подкатегорий меток.
  + public List<int> SelectedCategories – коллекция выбранных категорий меток.
  + public List<int> SelectedSubCategories – коллекция выбранных подкатегорий меток.
  + public bool WifiSelected – флаг фильтра по наличию *Wifi*.
  + public bool NowOpenSelected – флаг фильтра по времени работы.
  + public bool CategoriesVisible – флаг видимости списка категорий.
  + public bool SubCategoriesVisible – флаг видимости списка подкатегорий.
  + public bool MyMarkerSelected – флаг видимости личных меток пользователя.

Команды:

* + public ICommand ChangeWifiEnabledCommand – команда изменения состояния *WifiSelected*.
  + public ICommand ChangeNowOpenEnabledCommand – команда изменения состояния *NowOpenSelected*.
  + public ICommand OpenCloseCategories - команда открытия/закрытия списка категорий.
  + public ICommand OpenCloseSubCategories – команда открытия/закрытия списка подкатегорий.
  + public ICommand ClearFilterCommand – команда сброса фильтра
  + public ICommand ShowOnMapCommand – команда перехода к карте.

LoginViewModel – вьюмодель страницы входа в приложение.

Поля:

* + public string Login – логин пользователя.
  + public string Password – пароль пользователя.

Команды:

* + public ICommand LoginCommand – команда входа в приложение (вызывает метод private void LoginToApp()).
  + public ICommand GuestLoginCommand – команда гостевого входа в приложение (вызывает метод private void GuestLogin()).
  + public ICommand RecoverPasswordCommand – команда перехода на экран восстановления пароля (вызывает метод private void RecoverPassword ()).
  + public ICommand RegistrationCommand– команда перехода на экран регистрации пользователя (вызывает метод private void GotoRegistration ()).

RegistrationViewModel – вьюмодель страницы регистрации пользователя

Свойства:

* + public string Login – логин пользователя.
  + public string FirstName, MiddleName, LastName – ФИО пользователя .
  + public string Phone – телефон пользователя.
  + public int SelectedSexIndex – признак пола пользователя.
  + public DateTime BirthDate – дата рождения пользователя.
  + private bool AllFieldsIsFilled – флаг заполнения всех полей, по нему активируется кнопка «Регистрация».

Команды:

* + public ICommand RegisterCommand – команда регистрации нового пользователя.

AddNewPinViewModel– вьюмодель страницы добавления новой метки

Свойства:

* + private List<DeserializeCitiesData> ValidCities – список городов, в которые можно добавлять метки для текущего пользователя (запрашивается с сервера).
  + private string City – название города, куда добавляется метка.
  + private int CytiId – *ID* города, куда добавляется метка (*Google*).
  + private string Street – название улицы, куда добавляется метка.
  + private string House – название дома, куда добавляется метка.
  + private string Building – название здания, куда добавляется метка.
  + private bool BaseCategorySelected – флаг, показывающий, что выбрана базовая категория.
  + public List<PinCategory> PinCategories – категории меток.
  + public List<PinCategory> PinSubCategories – подкатегории меток.
  + public string BaseCategoryString – название выбранной базовой категории.
  + public Color BaseCategoryColor – цвет выбранной базовой категории.
  + public ImageSource BaseCategoryImage – иконка выбранной базовой категории.
  + public int SelectedDiscountIndex – выбранное значение скидки
  + public ObservableCollection<string> OldPhotos – набор старых фотографий метки
  + public ObservableCollection<string> Photos – набор выбранных фотографий.
  + public bool IsValidLocation – флаг правильности локации, куда добавляется метка.
  + public bool IsAllFieldsValid – флаг заполнения всех обязательных полей.
  + public IEnumerable<PinCategory> VisibleTags – коллекция видимых тегов на экране выбора тегов.
  + public IEnumerable<PinCategory> SelectedTags – коллекция выбранных тегов.
  + public List<int> SelectedSubCategories – коллекция выбранных категорий.
  + public string Tags – строка тегов в формате «#тег1 #тег2».
  + public List<WorkTimeViewModel> WorkTimeViewModels – коллекция заполненных ячеек «время работы».
  + public List<PhoneViewModel> PhoneViewModels – коллекция заполненных ячеек «телефон».
  + public WeekViewModel WeekViewModel – коллекция дней недели.
  + public UserPinDescriptor Pin – добавляемая метка на карте.
  + public Position MapCenter – центр карты.
  + public ObservableCollection<TKCustomMapPin> Pins – коллекция меток на карте.
  + public bool CategoriesPopupVisible – флаг открытого окна выбора категории.
  + public bool TagsPopupVisible – флаг открытого окна выбора тегов.
  + public bool DaysPopupVisible – флаг открытого окна выбора дней недели.
  + public bool IsPersonalMarker – флаг личной метки.

Методы:

* + private List<DeserializeCitiesData> ValidCities – список городов, в которые можно добавлять метки для текущего пользователя (запрашивается с сервера).
  + public void CheckFields() – проверяет заполнение обязательных полей.
  + private void UpdateSubcatgoriesVisible – обновляет коллекцию видимых тегов после выбора категории.
  + private async Task<KeyValuePair<bool, int>> GetRegion() – определяет параметры местоположения (адрес), куда добавляется метка.
  + private async Task CheckRegion() – проверяет верность местоположения, куда добавляется метка.
  + private void Category\_ItemTapped – вызывается по выбору категории, обновляет коллекцию видимых тегов.
  + private void Tag\_ItemTapped – вызывается по выбору тега, обновляет коллекцию выбранных тегов.
  + public void WoktimeVm\_CalendarClicked – вызывается по нажатию на иконку календаря, открывает окно выбора дней недели.
  + public void Phone\_PlusMinusClicked – вызывается по нажатию на иконку добавления/удаления телефона.
  + public void WoktimeVm\_PlusMinusClicked – вызывается по нажатию на иконку добавления/удаления ячейки времени работы.

Команды:

* + public ICommand AddPinCommand – команда добавления метки.
  + public Command<TKCustomMapPin> DragEndCommand – команда окончания перетаскивания метки на карте.
  + public ICommand SelectPhotoCommand – команда выбора фотографии.
  + public ICommand OpenCloseCategoriesCommand – команда открытия/закрытия окна выбора категории.
  + public ICommand OpenCloseTagsCommand – команда открытия/закрытия окна выбора тегов.
  + public ICommand CloseDaysCommand – команда закрытия окна выбора дней недели.

ArticlesViewModel – вьюмодель страниц «Календарь» и «Статьи»

Свойства:

* + private bool IsFilterEnable – флаг, показывающий, что применен фильтр.
  + public bool IsRefreshing – флаг, показывающий, что обновляется список статей/событий.
  + public DateTime StartDate – начальная дата фильтра.
  + public DateTime StopDate – конечная дата фильтра.
  + public List<ArticleEventItem> Articles – коллекция загруженных статей/событий.
  + public IEnumerable<ArticleEventItem> VisibleArticles – коллекция видимых статей/событий.
  + public ArticleEventItem SelectedArticleEventItem – выбранная статья/событие.
  + private string City – название города, куда добавляется метка.
  + public bool IsFavorite – флаг нахождения статьи в избранном
  + public int SelectedFilter – id выбранного фильтра: будущие, прошедшие

Команды:

* + public ICommand SearchCommand – команда применения фильтра событий.
  + public ICommand RefreshCommand – команда обновления списка событий/статей.
  + public Command LazyLoadCommand – команда подгрузки событий/статей с сервера.
  + public ICommand SearchFutureCommand – команда поиска событий в будущем от заданных дат
  + public ICommand SearchPastCommand – команда поиска событий в прошлом от заданных дат
  + public ICommand SearchFilterCommand – команда поиска по заданным датам

ArticleDetailsViewModel – вьюмодель экрана статьи/события

Свойства:

* + public string Title – заголовок экрана.
  + public ArticleEventItem ArticleEventItem – статья/событие
  + public bool IsFavorite – флаг статьи/события из избранного

Команды:

* + public Command AddToFavoritesCommand – команда добавления/удаления статьи/метки в избранное.

FavoritesViewModel – вьюмодель окна «Избранное»

Свойства:

* + public int SelectedFrame – номер текущего фрейма.
  + public int SelectedTitle – номер текущей вкладки.
  + public bool IsScrolling – флаг переключения между вкладками.
  + public bool IsRefreshing – флаг обновления списков.
  + public ObservableCollection<ArticleEventItem> Articles – коллекция избранных статей.
  + public ObservableCollection<ArticleEventItem> Articles – коллекция избранных событий.
  + public ObservableCollection<UserPinDescriptor> Pins – коллекция избранных меток.
  + public ArticleEventItem SelectedArticleItem – выбранная статья/событие.
  + public UserPinDescriptor SelectedPin – выбранная метка.

Команды:

* + public ICommand SelectFrameCommand – команда переключения на выбранную вкладку.
  + public ICommand DeleteArticle – команда удаления статьи/события из избранного.
  + public ICommand DeletePin – команда удаления метки из избранного.

Методы:

* + private void Pin\_DeleteFromFavorites - метод удаления метки из избранного.
  + private void Result\_DeleteItem – метод удаления статьи/события из избранного.
  + private void DetailsViewModel\_PinDeleted – команда удаления меткти из избранного.

## Пространство имен ViewModel.Design

В данном пространстве имен содержатся вьюмодели для *UI*-элементов пространства *Design*.

## Пространство имен Views

В данном пространстве имен содержатся UI – представления экранов

* TKMapView – класс главного экрана карты.
* AddPinView – класс экрана добавления метки.
* PinDetailsView – класс экрана метки.
* ArticlesView – класс экрана статей.
* EventsView – класс экрана событий.
* ArticleDetailsView – класс экрана события/статьи.
* FavoritesAndroidView – класс экрана избранного для *Android*.
* FavoritesAppleView – класс экрана избранное для *iOS*.
* LoginView – класс экрана входа в приложение.
* RecoverPasswordView – класс экрана восстановления пароля.
* RegistrationView – экрана регистрации.

## Пространство имен Settings

В данном пространстве имен содержатся системные классы для внутренней работы приложения.

* ApplicationSettings – статический класс, через который предоставляется доступ к БД, загруженным меткам, категориям и методам работы с сервером.

События:

* + event EventHandler CategoriesLoaded – оповещает приложение о том, что категории загружены с сервера.

Поля:

* + App MainApp – текущее приложение.
  + bool LoadingMarkers – флаг загрузки меток с сервера.
  + MapBulService Service – обеспечивает взаимодействие с сервером.
  + UserInformation CurrentUser – информация о текущем пользователе.
  + LoadedPins LoadedPins – набор загруженных меток и регионов, для которых метки загружены.
  + MapBulDataBaseRepository DataBase – база данных.
  + NavPage MainPage – навигация в приложении.
  + IDevice ThisDevice – текущее устройство.
  + IToastNotificator Notificator – обеспечивает показ уведомлений.
  + string MapsApiKey – ключ к *Google Maps Api*.
  + bool IsConnectToInternet – флаг наличия соединения с интернетом.
  + List<PinCategory> PinCategories – загруженные категории меток.
  + List<PinCategory> PinSubCategories – загруженные подкатегории меток.

Методы:

* + List<PinCategory> GetAllCAtegories () – получение всех категорий, включая подкатегории.
  + void GoToPage(Page page) – переход на страницу.
  + async void GoToAuth() – переход на страницу входа в приложение
  + int SaveNewUser(User user) – сохранение профиля пользователя в БД.
  + static void LoadCategories() – загрузка категорий с сервера.
  + void SetCurrentSession()– установка новой сессии с сервером.
  + void ClearServerSession()– разрыв всех предыдущих сессий.
  + void CreateToast(ToastNotificationType toastType, string text) – показ уведмления.
  + void OnCategoriesLoaded() – вызов события CategoriesLoaded.
* Файл *Deserialize.cs* – содержит классы для парсинга ответов от сервера.
* Файл *Enums.cs* – содержит перечислители (типы пользователей, типы страниц и т.д.).
* GoogleJsonResponse – класс для парсинга ответа от *Google Geocoder*.
* IFileSystemWork – интерфейс для считывания файлов из файловой системы устройства, он реализуется в проектах *iOS* , *Android*.
* MapBulService – класс, реализующий вызов методов веб-службы сервера.
* MyMasterDetailPage – класс, обеспечивающий показ бокового меню приложения.
* CustomNavPage – класс для навигации в приложении, метод async void GoToPage(Page page) открывает новое окно в приложении.
* Файл *SqlDB.cs* – содержит интерфейс ISqLite, реализуемый в проектах *iOS* , *Android*, с помощью него создается подключение к базе данных. Сама база данных описана классом MapBulDataBaseRepository, он содержит компоненты *CRUD* для работы с БД.
* Statics – класс, содержащий методы расширения для различных классов:
  + bool Contains(this MapSpan firstRegion, MapSpan secondRegion) – метод, определяющий, лежит ли secondRegion внутри firstRegion.
  + bool Contains(this MapSpan firstRegion, Position position) – метод, определяющий, лежит лит position внутри firstRegion.
  + string ToTagsString(this IEnumerable<string> self, bool enableHashTag = true) – метод, преобразующий коллекцию строк в одну строку тегов.
  + int ToRussian(this DayOfWeek item) – метод, преобразующий английские номера дней недели в российские.
  + DayOfWeek ToDayOfWeek(this int item) – метод, преобразующий номер дня недели в экземпляр встроенного перечислители дней недели.
  + class ComparePins – класс-компаратор, который позволяет проверять равенство меток.

Работа приложения начинается с конструктора класса *App*.

Запускается загрузка категорий меток в фоновом режиме, устанавливается новая сессия работы с сервером, проверяется последний вошедший пользователь, если он находится – осуществляется вход в приложение из-под его профиля, если нет – открывается экран входа в приложение.

Как только были загружены категории меток, срабатывает событие ApplicationSettings.CategoriesLoaded, можно начинать работу с картой, мети подгружаются динамически при изменении области видимости карты на экране.