

ROBOTPEN 鲁伯特

iOS RobotSqlManager

参考手册

目录

简介：	3
一、 笔记相关	5
1、 笔记数据库操作.....	5
2、 分页数据库操作.....	7
1、 页码新建	7
2、 页码删除	8
3、 页码变更	8
4、 页码查询	9
3、 笔迹数据库操作。	11
1、 笔迹保存	11
2、 笔迹删除	11
3、 笔记修改	12
4、 笔迹查询	13
4、 笔记列表相关操作	15
5、 白板截图相关操作	17
6、 临时笔记相关操作。	18
二、 录课相关	20
1、 视频数据库操作.....	20
1、 创建更新录课	20
2、 获取录课信息	22
2、 视频列表操作	23
1、 视频列表获取	23
2、 视频列表删除	24
三、 页码检测设备专用.....	24
四、 其他	26

简介：

iOS RobotSqlManager 包含在 SDK 中无需单独导入。

数据库服务封装了 SDK 中所有的数据库操作。

数据库服务主要功：

一、 笔记相关

- 1、 笔记数据库操作。
- 2、 分页数据库操作。
- 3、 笔迹数据库操作。
- 4、 白板截图相关操作。
- 5、 临时笔记相关操作。
- 6、 笔记列表相关操作。

二、 录课相关

- 1、 视频数据库操作。
- 2、 视频列表操作

三、页码检测设备专用

四、其他

基本概念：

笔记：多组分页集合，相当于纸质笔记本。

分页（页码）：多组笔迹数据集合，相当于一页纸质笔记纸。

笔迹：多组点数据集合，相当于笔记纸上的笔划轨迹。

点：一条笔迹中的一个点。

一本笔记（Note）有 0-N 个分页（Block），一个分页（Block）有 0-N 个笔迹点集合（Trails），一条笔迹中含有 1-N 个点。

所有的点操作都是基于笔迹，笔迹操作基于分页，分页操作基于笔记。

笔记操作都是基于笔记唯一标识（NoteKey），根据 NoteKey 进行增删改查等操作。

分页操作都是基于分页唯一标识（BlockKey），根据 BlockKey 进行相关操作。

离线笔记：存储在电磁板设备上的笔记，可通过同步笔记功能同步到应用中。

临时笔记：自定义的一个唯一标识固定的笔记（@"TempNote"），可作为暂存笔记。

一、 笔记相关

1、 笔记数据库操作

1、 创建一本完整的笔记

创建一本笔记（Note），笔记默认带有一个分页（Block）。

```
+ (void)BuildNote:(RobotNote *)NoteObj Success:(void (^)(id responseObject))Success Failure:(void (^)(NSError *error))Failure;
```

2、 创建笔记头（空笔记）

创建一本笔记头，不含有分页，需手动添加分页（非必要情况下不建议使用本方法）

```
+ (void)BuildNoteHeader:(RobotNote *)NoteObj Success:(void (^)(id responseObject))Success Failure:(void (^)(NSError *error))Failure;
```

3、 查询笔记是否存在。

根据笔记唯一标识（NoteKey）查询笔记是否存在。

```
+ (BOOL)checkNoteWithNoteKey:(NSString *)NoteKey;
```

4、获取笔记头信息。

根据笔记唯一标识（NoteKey）获取笔记头信息

```
+ (void)GetNoteListInfoWithNoteKey:(NSString *)NoteKey  
Success:(void (^)(id responseObject))Success Failure:(void  
(^)(NSError *error))Failure;
```

5、修改笔记名称。

根据笔记的唯一标识（NoteKey）和新的笔记名称，修改笔记名称。

```
+ (void)ChangeNameWithNoteKey:(NSString *)NoteKey  
TagertTitle:(NSString *)TagertTitle Success:(void (^)(id  
responseObject))Success Failure:(void (^)(NSError  
*error))Failure;
```

注：创建一本完整的笔记，即可进行相关笔记、页码操作，添加页码调用“页码创建”中的 1、2、3 方法即可。

2、分页数据库操作

1、页码新建

1、查询新建下一页页码信息

按笔记页码当前顺序向下新建一页，并返回新页码唯一标识

BlockKey

```
+ (NSString *)getNextWithNoteKey:(NSString *)NoteKey
```

```
BlockKey:(NSString *)ActivityBlockKey;
```

2、查询新建上一页页码信息

按笔记页码当前顺序向上新建一页，并返回新页码唯一标识

BlockKey

```
+ (NSString *)getFrontWithNoteKey:(NSString *)NoteKey
```

```
BlockKey:(NSString *)ActivityBlockKey;
```

3、插页

在当前页和后一页之间插入新页并返回 Blockkey

```
+ (NSString *)GetNewNextWithNoteKey:(NSString *)NoteKey
```

```
BlockKey:(NSString *)ActivityBlockKey;
```

2、页码删除

根据页码唯一标识 (BlockKey) 删除指定页

```
+ (void)DeleteBlockKey:(NSString *)ActivityBlockKey  
  
Success:(void (^)(id responseObject))Success  
  
Failure:(void (^)(NSError *error))Failure;
```

3、页码变更

1、设置当前页的下一页页码

设置当前页的下一页的 Block, 修改 block 指向用于更换页码顺序

```
+ (void)SetNextBlockKey:(NSString *)nextBlockKey  
  
BlockKey:(NSString *)blockKey : (NSString *)Notekey;
```

2、设置当前页的上一页页码

设置当前页的上一页的 Block, 修改 block 指向用于更换页码顺序

```
+ (void)SetBlockKey:(NSString *)blockKey  
  
NextBlockKey:(NSString *)nextBlockKey : (NSString *)Notekey;
```


4、页码查询

1、获取当前笔记的页码信息

根据笔记的唯一标识（NoteKey），获取当前笔记页码信息，

总页码 当前页 活动页

```
+ (void)GetPageInfosWithNoteKey:(NSString *)NoteKey  
Success:(void (^)(id responseObject))Success Failure:(void  
(^)(NSError *error))Failure;
```

2、获取上一页的页码唯一标识

根据笔记的唯一标识（NoteKey），获取上一页的 BlockKey

```
+ (NSString *)GetFront:(NSString *)NoteKey  
BlockKey:(NSString *)ActivityBlockKey;
```

3、获取下一页的页码唯一标识

根据笔记的唯一标识（NoteKey），获取下一页的 BlockKey

```
+ (NSString *)getNext:(NSString *)NoteKey  
BlockKey:(NSString *)ActivityBlockKey;
```

4、获取当前页是否存在

根据笔记的唯一标识（NoteKey），获取当前页是否存在

```
+ (BOOL)IsExist:(NSString *)NoteKey BlockKey:(NSString *)ActivityBlockKey;
```

5、获取当前笔记的所有 Block

根据笔记的唯一标识（NoteKey），获取当前笔记的所有

Block

```
+ (void)GetAllBlockWithNoteKey:(NSString *)NoteKey  
Success:(void (^)(id responseObject))Success Failure:(void  
(^)(NSError *error))Failure;
```

3、笔迹数据库操作。

1、笔迹保存

1、保存笔迹数据

根据笔迹模型，保存笔迹数据

```
+ (void)SaveTrails:(RobotTrails *)TrailsObj Success:(void (^)(id responseObject))Success Failure:(void (^)(NSError *error))Failure;
```

2、笔迹删除

1、删除指定笔迹数据

删除指定笔迹数据

```
+ (void>DeleteTrails:(RobotTrails *)TrailsObj Success:(void (^)(id responseObject))Success Failure:(void (^)(NSError *error))Failure;
```

2、删除指定页码中的所有笔迹数据

页码唯一标识 (BlockKey)，删除指定页笔迹数据

```
+ (void)DelNoteTrailsWithBlockKey:(NSString *)ActivityBlockKey Success:(void (^)(id responseObject))Success Failure:(void (^)(NSError *error))Failure;
```

3、删除指定页的图片数据

页码唯一标识 (BlockKey) ， 删除指定页图片数据

```
+ (void)DelNoteImgsWithBlockKey:(NSString *)ActivityBlockKey Success:(void (^)(id responseObject))Success Failure:(void (^)(NSError *error))Failure;
```

4、删除指定页的所有数据（笔迹和图片数据）

页码唯一标识 (BlockKey) ， 删除指定页所有的数据

```
+ (void)DelNoteWithBlockKey:(NSString *)ActivityBlockKey Success:(void (^)(id responseObject))Success Failure:(void (^)(NSError *error))Failure;
```

3、笔记修改

1、向前合并笔迹

根据目标页码唯一标识 (BlockKey) 、笔迹时间节点和当前页码唯一标识，合并笔迹到前一个页上。

```
+ (void)MergeTrailsWithBlockKey:(NSString *)Blockkey WithEndTime:(long long)endTime WithActivityBlockKey:(NSString *)ActivityBlockKey
```

```
Success:(void (^)(id responseObject))Success Failure:(void  
(^)(NSError *error))Failure;
```

2、向后分离笔迹

根据目标页码唯一标识（BlockKey）、笔迹时间节点和当前页码唯一标识，分离笔迹到下一个页上。

```
+ (void)SeparateTrailsWithBlockKey:(NSString *)Blockkey  
WithEndTime:(long long)endTime  
WithActivityBlockKey:(NSString *)ActivityBlockKey  
Success:(void (^)(id responseObject))Success  
Failure:(void (^)(NSError *error))Failure;
```

3、更新分页的更新时间

根据笔记的唯一标识（NoteKey）和分页唯一标识（BlockKey）更新笔记分页的 UpdateTime

```
+ (void)UpdateNoteWithNoteKey:(NSString *)NoteKey  
BlockKey:(NSString *)BlockKey Success:(void (^)(id  
responseObject))Success Failure:(void (^)(NSError  
*error))Failure;
```

4、笔迹查询

1、获取当前页的所有笔迹

根据当前页码唯一标识（BlockKey），获取当前页的笔迹数据

```
+ (void)GetNoteTrailsWithBlockKey:(NSString  
*)ActivityBlockKey SuccessTrails:(void (^)(id  
responseObject))SuccessTrails SuccessImages:(void (^)(id
```

```
responseObject))SuccessImages Failure:(void (^)(NSError
*error))Failure;
```

2、获取指定页的所有笔迹

根据笔记的唯一标识（NoteKey）和指定页页码唯一标识（BlockKey），获取指定页的笔迹数据

```
+ (void)GetAllTrailsWithNoteKey:(NSString *)NoteKey
WithBlockKey:(NSString *)BlockKey Success:(void (^)(id
responseObject))Success Failure:(void (^)(NSError
*error))Failure;
```

3、获取指定页的所有图片（目前只有一张）

根据笔记的唯一标识（NoteKey）和指定页页码唯一标识（BlockKey），获取指定页的所有图片

```
+ (void)GetAllImageOfTrailsWithNoteKey:(NSString *)NoteKey
WithBlockKey:(NSString *)BlockKey Success:(void (^)(id
responseObject))Success Failure:(void (^)(NSError
*error))Failure;
```

4、获取指定页的最后五张图片

根据当前页页码唯一标识（BlockKey）查询画板中的最后五张图片

```
+ (void)GetFiveImageWith:(NSString *)ActivityBlockKey  
  
Success:(void (^)(id responseObject))Success  
  
Failure:(void (^)(NSError *error))Failure;
```

4、笔记列表相关操作

笔记列表的排序是根据笔记的最后更新时间降序排列的。

笔记列表的获取，默认每页拉取 20 条，可查看“其他”方法自定义设置。

页码参数 Page >= 0;

1、获取指定页的笔记列表

根据传入的页码值获取该页笔记列表，返回该页的笔记列表数据。

```
+ (void)GetAllNoteListWithPage:(int)Page  
  
Success:(void (^)(id responseObject))Success  
  
Failure:(void (^)(NSError *error))Failure;
```

2、获取指定页之前的所有笔记列表

根据传入的页码值获取 Page 页之前所有笔记列表，返回 0

– Page 页的所有笔记列表数据

```
+ (void)GetAllNoteListBeforePage:(int)Page  
Success:(void (^)(id responseObject))Success  
Failure:(void (^)(NSError *error))Failure;
```

3、删除指定笔记

根据笔记的唯一标识（NoteKey），删除该笔记

```
+ (void>DeleteNoteWithNoteKey:(NSString *)NoteKey  
Success:(void (^)(id responseObject))Success Failure:(void  
(^)(NSError *error))Failure;
```

4、删除所有笔记

删除笔记列表全部笔记。

```
+ (void>DeleteAllNoteSuccess:(void (^)(id  
responseObject))Success Failure:(void (^)(NSError  
*error))Failure;
```


5、白板截图相关操作

1、获取缩略图保存的完整路径

根据笔记的唯一标识（NoteKey），获取笔记缩略图路径（用于笔记缩略图）

```
+ (NSString *)GetNoteImagePathWithKey:(NSString *)Key;
```

2、获取笔记指定页的截图保存的完整路径

根据笔记的唯一标识（NoteKey）和页码唯一标识（BlockKey），获取笔记指定页的截图

```
+ (NSString *)GetNoteShareImagePathWithKey:(NSString *)Key andBlockKey:(NSString *)blockkey;
```

3、获取新的截图保存完整路径

获取新的截图保存完整路径，后可将图片写入该路径

```
+ (void)SaveImageKeySuccess:(void (^)(id responseObject))Success  
Failure:(void (^)(NSError *error))Failure;
```

4、获取指定页保存截图列表的相对路径

根据指定页码信息，获取 Page 页的截图列表数据

```
+ (void)GetImagesArrayWithPage:(int)Page Success:(void
(^)(id responseObject))Success
Failure:(void (^)(NSError
*error))Failure;
```

5、删除指定截图图片

根据传入的图片保存相对路径 Key 数组，删除图片

```
+ (void>DeleteImageWithKeyArray:(NSArray *)KeyArray
Success:(void (^)(id responseObject))Success Failure:(void
(^)(NSError *error))Failure;
```

6、获取截图图片总数

获取截图图片总数

```
+ (long)GetImageNumber;
```

7、获取截图存储完整路径

根据截图的相对路径获取截图图片存储完整路径

```
+ (NSString *)GetImagePathWithKey:(NSString *)Key;
```

6、临时笔记相关操作。

1、保存临时笔记

保存临时笔记

```
+ (void)SaveTempNote:(NSString *)NoteKey Success:(void (^)(id responseObject))Success Failure:(void (^)(NSError *error))Failure;
```

2、设置临时笔记的设备类型

设置临时笔记的设备类型

```
+ (void)ChangeTempNoteDeviceType:(int)DeviceType;
```

3、设置临时笔记的画板方向

设置临时笔记的画板方向

```
+ (void)ChangeTempNoteIsHorizontal:(BOOL)mark;
```

二、 录课相关

1、 视频数据库操作

1、 创建更新录课

1、 创建录课文件

根据视频唯一标识 (NameKey) 创建视频文件

```
+ (void)BuildVideoWithNameKey:(NSString *)NameKey
```

```
Success:(void (^)(id responseObject))Success
```

```
Failure:(void (^)(NSError
```

```
*error))Failure;
```

2、 更新视频信息

更新视频信息

```
+ (void)PerfectVideoInfosWithVideo:(RobotVideo *)Video
```

```
Success:(void (^)(id responseObject))Success
```

```
Failure:(void
```

```
(^)(NSError *error))Failure;
```

3、 更新视频名称

根据视频唯一标识 (NameKey) 和新视频名称更改视频名称

```
+ (void)ChangeVideoWithNameKey:(NSString *)NameKey
Alias:(NSString *)Alias Success:(void (^)(id
responseObject))Success Failure:(void (^)(NSError
*error))Failure;
```

4、更新视频标签状态

根据视频唯一标识（NameKey）和新视频标签，更新视频标签（可作为状态判断）

```
+ (void)SetOnlineWithNameKey:(NSString *)NameKey
IsOnline:(int)IsOnline Success:(void (^)(id
responseObject))Success Failure:(void (^)(NSError
*error))Failure;
```

5、更新视频 MD5 值

根据视频唯一标识（NameKey）和新视频 MD5 值，更新 MD5 信息

```
+ (void)SaveMd5StrWithNameKey:(NSString *)NameKey
Md5Str:(NSString *)Md5Str Success:(void (^)(id
responseObject))Success Failure:(void (^)(NSError
*error))Failure;
```

2、获取录课信息

1、根据 NameKey 获取视频信息

根据视频唯一标识 (NameKey)，获取视频信息

```
+ (void)GetVideoInfoWithNameKey:(NSString *)NameKey  
Success:(void (^)(id responseObject))Success  
Failure:(void (^)(NSError *error))Failure;
```

2、根据 MD5 获取视频信息

根据 MD5 值获取视频信息

```
+ (void)GetNoteInfoWithMd5Str:(NSString *)Md5Str  
Success:(void (^)(id responseObject))Success  
Failure:(void (^)(NSError *error))Failure;
```

3、根据 NameKey 获取视频路径

根据 NameKey 获取视频路径

```
+ (NSString *)GetVideoPathWithNameKey:(NSString  
*)NameKey;
```

4、根据 Md5 获取视频路径

根据 Md5 获取视频路径

```
+ (void)GetVideoPathWithMd5Str:(NSString *)Md5Str  
Success:(void (^)(id responseObject))Success  
Failure:(void (^)(NSError *error))Failure;
```

2、视频列表操作

1、视频列表获取

1、获取指定页视频列表中的所有视频

根据传入的页码值获取 Page 页的所有视频列表，返回当前视频列表 age 页的所有数据

```
+ (void)GetVideoListWithPage:(int)Page Success:(void (^)(id responseObject))Success Failure:(void (^)(NSError *error))Failure;
```

2、获取视频列表中指定页之前的所有视频

根据传入的页码值获取 Page 页之前所有视频列表，返回 0 - Page 页的所有笔记列表数据

```
+ (void)GetAllVideoListBeforePage:(int)Page  
Success:(void (^)(id responseObject))Success  
Failure:(void (^)(NSError *error))Failure;
```

3、获取视频列表中的视频总数

获取视频列表中的视频总数

```
+ (long)GetVideoTotalNum;
```

2、视频列表删除

1、删除指定视频

根据视频唯一标识（NameKey），删除指定视频

```
+ (void)DeleteVideoWithNameKey:(NSString *)NameKey  
  
Success:(void (^)(id responseObject))Success  
Failure:(void (^)(NSError *error))Failure;
```

2、删除视频列表所有视频

删除视频列表所有视频

```
+ (void)DeleteAllVideoSuccess:(void (^)(id  
responseObject))Success Failure:(void (^)(NSError  
*error))Failure;
```

三、 页码检测设备专用

1、获取或新建页码笔记本中的指定页的唯一标识。

根据笔记的唯一标识（NoteKey）和页码（Page）获取该分页的 Blockkey，为空则新建一个分页

```
+ (NSString *)getOrBuildBlockKeyWithNoteKey:(NSString  
*)Notekey PageNumber:(int)Page;
```


2、获取页码笔记本中的指定页的唯一标识

根据笔记的唯一标识（NoteKey）和页码（Page）获取该分页的 Blockkey，不会新建

```
+ (NSString *)getBlockKeyWithNoteKey:(NSString *)Notekey  
PageNumber:(int)Page;
```

3、获取页码笔记的页码编号

根据笔记的唯一标识（NoteKey）和页码唯一标识（BlockKey）获取该页的页码编号（PageNumber）

```
+ (int)getPageNumberWithNoteKey:(NSString *)NoteKey  
BlockKey:(NSString *)BlockKey;
```

4、页码识别设备专用离线笔记保存方法

根据笔记头信息 (RobotNote) 和页码信息（Page）保存页码识别离线笔记

```
+ (void)SaveTANote:(RobotNote *)NoteObj  
andPage:(int)Page Success:(void (^)(id  
responseObject))Success Failure:(void (^)(NSError  
*error))Failure;
```

四、 其他

1、获取当前时间戳

根据系统时间获取当前时间戳

```
+ (NSTimeInterval)GetTimeInterval;
```

2、获取一个新的 NoteKey

根据时间戳创建一个新的 NoteKey

```
+ (NSString *)GetNewKey;
```

3、设置列表默认拉取拉取条数

设置列表默认拉取拉取条数

```
+ (void)SetNumberOfPageWith: (int)page;
```