アンドロイドアプリ「ListManager」仕様書 ver1.1

　　　　　チーム「胡椒博士」

　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　堀江友也

　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　伊藤陽太

１．概要

２．クラス図

３．状態遷移図

４．関数定義

５．Xml部品定義

１．概要

　　バージョン：２．３．１

　　開発環境　：eclipse

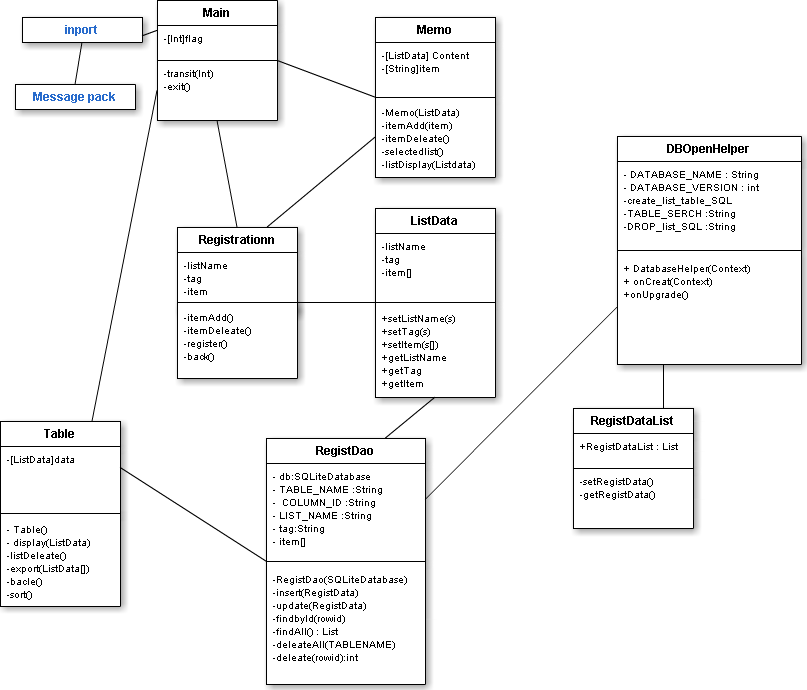
　　開発言語　：Java

　　リポジトリ：Github、Dropbox

　　現段階でJSONなどのデータ記述言語やMessagePackといったライブラリなどの調査に致命的な遅れが出ているため、Import及びExportの機能実装は保留とする

クラス図

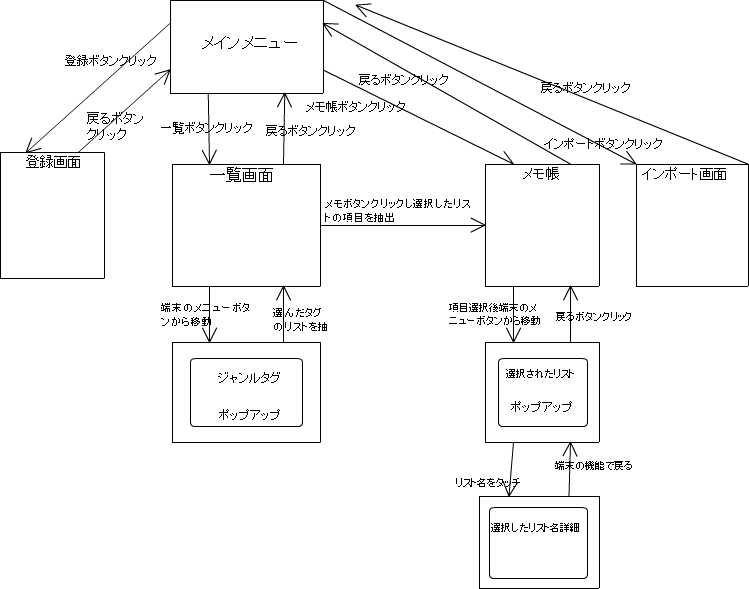
※青文字はデータ転送用のクラスであり、ver1.1で追記する予定



Registaddtion

状態遷移図

※画面内のコンポーネントについては５．Xml部品定義に記述



メンバ変数及び関数定義

Mainクラス

-[Int]flag　　　 //transitに渡す変数で遷移する画面ごとに変数を設定されている

-transit(flag) //flagを受け取ってflag通りの画面に遷移させるメソッド

-exit() //アプリを終了させるメソッド

＊画面遷移する際にDBからデータを抽出しIntentに付加する

Registrationクラス

-listName　 //リスト名を格納する変数

-tag //ジャンルタグを格納する変数

-item //アイテムを格納する変数

-itemAdd() //itemの入力欄を増やすメソッド

-register() //SQLを発行するメソッド

　-back() //メイン画面に戻るメソッド

Tableクラス

-[ListData]data[] //Tableメソッドから呼び出した情報を格納する

配列

-[list]Tag\_List[]<ListData> //createListメソッドで作成した配列

　 -Tag\_createList() 　　　 //抽出するタグのリストを格納する配列を作成す

るメソッド

-Table() 　　//ListDataクラスから情報を呼び出すメソッド

-display(data) 　　//配列dataの情報をリスト画面に反映させるメソ

ッド

-list\_Deleate() 　　//選択したリストを削除するメソッド

-export(ListData[]) 　　//選択したリストをエキスポートするメソッド

-back() 　　//メイン画面に戻るメソッド

-transit\_To\_Memo() 　　//メモ帳画面に遷移と選択したリストの情報を

メモ帳画画面に送るメソッド

-sort() 　　 //選択したジャンルタグで抽出するメソッド、抽

出するジャンルタグの選択はポップアップ画面に

て行うためポップアップの呼び出しも行う

Memoクラス

-[list]Memo\_List[]<ListData> //Memo\_createListメソッドで作成したリスト

-[ListData]Content[] 　　 //Memoメソッドで呼び出した情報を呼び出す配列

-[String]item //新しくデータを入力する際に格納される変数

　　-Memo\_createList() //リストを格納する配列を作成するメソッド

　　-Memo() //ListDataクラスから情報を呼び出すメソッド

-itemAdd(item) //item変数の内容を画面に反映させるメソッド

-itemDeleate() //リスト画面から選択した情報を削除するSQLは関係

なし。

-itemAllDeleate() //リスト画面から情報を全削除するSQLは関係なし

　　　-selectedlist() //端末の機能でポップアップしたリスト一覧画面でリ

　　　　　　　　　　　　　　　　　　　　ストを選択するメソッド

-listDisplay() //リスト詳細画面ポップアップするメソッド

ListDataクラス

-[String]listName //リスト名を格納する変数

-[String]tag //タグを格納する変数

-[String]item[] //SQLから引き出した情報を格納する配列

+[String]setListName() //変数を格納する（セッター）

+[String]setTag() //変数を格納する（セッター）

+[String]setItem() //変数を格納する（セッター）

+[String]getListName() //変数を呼び出す（ゲッター）

+[String]getTag() //変数を呼び出す（ゲッター）

+[String]getItem[] //変数を呼び出す（ゲッター）

DBOpenHelperクラス

-DATABASE\_NAME //データベース名の変数

-DATABASE\_VERSION //データベースバージョンの変数

　　　　-CREATE\_LIST\_TABLE\_SQL //SQLを発行するための構文を格納

する変数

-TABLE\_SEARCH //テーブルを検索するための構文を格納するための変数

-DROP\_LIST\_SQL //テーブルを削除する構文を格納するための変数

　+DatabaseHelper(Context) //データベースに接続するメソッド

+onCreate(Context) //初期化処理テーブルを発行するた

めのクエリーを実行するメソッド

　+onUpgrade() //更新処理を行うクエリーを発行す

　るメソッド

RegistDaoクラス

-[SQLiteDatabase]db //SQLに接続するための変数

-[String]TABLE\_NAME //レコードの挿入、削除、検索、更新、全取得

に使用する変数

-[String]COLUMN\_ID //item配列に格納する変数item配列に格納

もする

-[String]LIST\_NAME //レコードの挿入、更新などに使用する変数

-[String]tag

-[String]item[] //レコード検索などで使用する変数を格納

する配列

-RegistDao(SQLitedatabase)//SQLに接続するメソッド

-insert(RegistData) //データの挿入処理を行うメソッド

-update(RegistData) //データの更新処理を行うメソッド

-findbyid(rowid) //行番号のレコードの検索を行うメソッ

　　　　　　　　　　　　　　　　　　　　　　　ド

-[List]findAll() //全レコードを取得するメソッド

-deleateAll(TABLE\_NAME) //全レコードを削除するメソッド

-[int]deleate(rowid) 　　//行番号のレコードを削除するメソッド

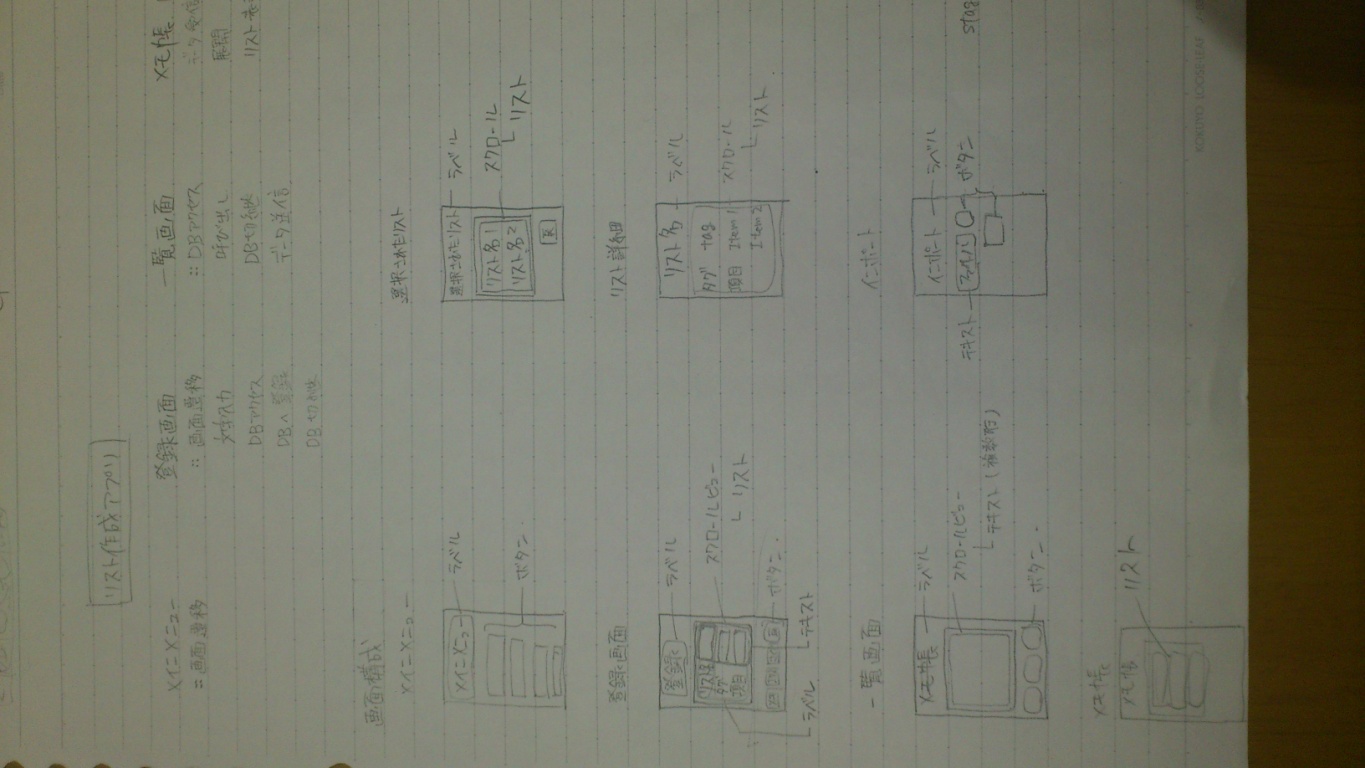
RegistDataListクラス

+[List]RegistDataList //全レコードを格納する配列

-setRegistData() //レコードを格納するメソッド

-getRegistData()//レコードを返すメソッド

Xml部品定義

+

IDについての規則

　例：btnMain\_1

　　メイン画面のボタンの番号

　　番号は上から順その次は左から順に番号をふる