

# **DUYTAN UNIVERSITY INTERNATIONAL SCHOOL**



## **PROJECT BABYFUN APPLICATION ON ANDROID SDK 2.2**



## **DATABASE DESIGN DOCUMENT**

**Group ID: CMU\_DTU**

**Code: BBF.DBD.1.1**

**Date: 2012/03/28**

**Version: 1.2**

**Create by: Tran Quoc Dai**

**Da Nang, 5/2012**

Revision	Date	Description	Author
1.0	2012/03/09	Create Document	Tran Quoc Dai
1.1	2012/03/28	Format document	Le Thi Le Thuy

Document Approvals: The following signatures are required for approval of this document.

Nguyen Tan Thuan Mentor 1	Date
Le Van Chung Mentor 2	Date
Nguyen Tan Huy FPT SoftWare	Date
Thi Le Le Thuy Project Manager	Date
Phan Thi Bich Phuong Tester & Configuration management	Date
Dinh Van Dinh Developer	Date
Tran Quoc Dai Developer	Date

## Table of Contents

OVERVIEW .....	5
1.1. Introduction .....	5
1.2. Scope.....	5
1.3. Related Documents .....	5
1.4. Sub application/Application Overview .....	5
1.5. Hardware and Software Architecture .....	6
2. Introduce about SQLite database management system.....	6
2.1. Introduce .....	6
2.2. Constraint .....	7
2.3. Version .....	7
3. Database Design Decisions .....	7
3.1. Assumptions.....	7
3.2. Mapping rules .....	7
3.3. Table Relationship Diagrams .....	8
3.4. Description of Table.....	8
3.5. Key mappings .....	9

## OVERVIEW

### 1.1. Introduction

This Database Design provides the basis for the Babyfun Application Database Design. It defines the database that will support the Babyfun Application Data Model. It describes both logical and physical definition, non-functional issues, and the database interfaces; storage aspects are defined in the physical database design sections. The design is created with expected data volumes, functional and non-functional usage of the tables, and performance considerations and requirements in mind.

The following topics are covered in this document:

- Assumptions and decisions on database design
- Entity-mapping
- Table, column definitions
- Primary, unique and foreign key definitions
- Column and row level validation rules (check constraints)
- Rules for populating specific columns (sequences, derivations, renormalized (columns))
- Interfaces and dependencies with other components
- Data access description

### 1.2. Scope.

The Database Design for the Babyfun Application is composed of definitions for database objects derived by mapping entities to tables, attributes to columns, unique identifiers to unique keys and relationships to foreign keys.

During design, these initial definitions are enhanced to support the functionality described in the functional specification / use cases and defined in the primary and supporting modules of the application high level design.

### 1.3. Related Documents

This specification refers to the following documents:

- Babyfun Application Project Plan
- Babyfun Application Requirements Document

### 1.4. Sub application/Application Overview

Baby Fun is an application to play & study on the mobile that parent can manage question, select language that they want English or Vietnamese, select theme for interface, control volume...

Baby can study English, Vietnamese, Math on application simultaneous to they listen music while they are studying. Study and play coordinating will create baby comfortably...

### 1.5. Hardware and Software Architecture

This section provides an overview of the software and hardware architecture. The following describes the technology components of the Babyfun Application.

The technology component of the Babyfun Application

Attribute	Description
Design	Stand-alone configuration
Platform	Android
Database	SQLite is used to stored data fields.
Software	The Babyfun application was developed stand-alone configuration. The proposed programming language for the development of the main Babyfun application is Java.
Hardware	Mobile phone has Android OS.

## 2. Introduce about SQLite database management system

### 2.1. Introduce

SQLite is a software database management (DBMS) like MySQL, PostgreSQL ... Features of SQLite is a compact, simple. The program includes a mere single file less than 500KB, no installation, no configuration or boot that can be used immediately.

The data are stored in a database file only. No concept of user, password or powers in the SQLite database.

SQLite is not suitable for larger systems/application but in the medium range promote

SQLite is powerful and not weak in terms of functionality or speed. With the

characteristics of SQLite is used more in the development, testing and appropriate choice for the beginner database

## 2.2. Constraint

- Database relational form (RDBMS) support SQL-92 standard
- Using the library as a dip, do not run at server independent manner
- Supports popular languages: C + +,Java ...
- There is no need to specify the data type (SQLite is type less)
- Support UTF8 code.
- Support the command line.
- Support the transaction.
- Support for display.
- Support for C extensions.
- There have tools for graphically.
- Download & use for free.

## 2.3. Version

There currently are 2 different branches SQLite is version 2.x and version 3.x, the internal data structure different to the command name and library name of each different branch. SQLite command is of version 2.x, sqlite3 command of version 3.x. [Http://sqlite.org](http://sqlite.org) home address.

## 3. Database Design Decisions

This section contains the decisions that were made when designing the database for the Babyfun. Problems, alternative solutions and motivated choices are listed below.

The section also lists any design assumptions that had to be made. In case the assumptions are results of ambiguities or lack of details, they will need verifying by the analyst team

### 3.1. Assumptions

Some assumptions made due to lack of information, will be update in next version of database design document

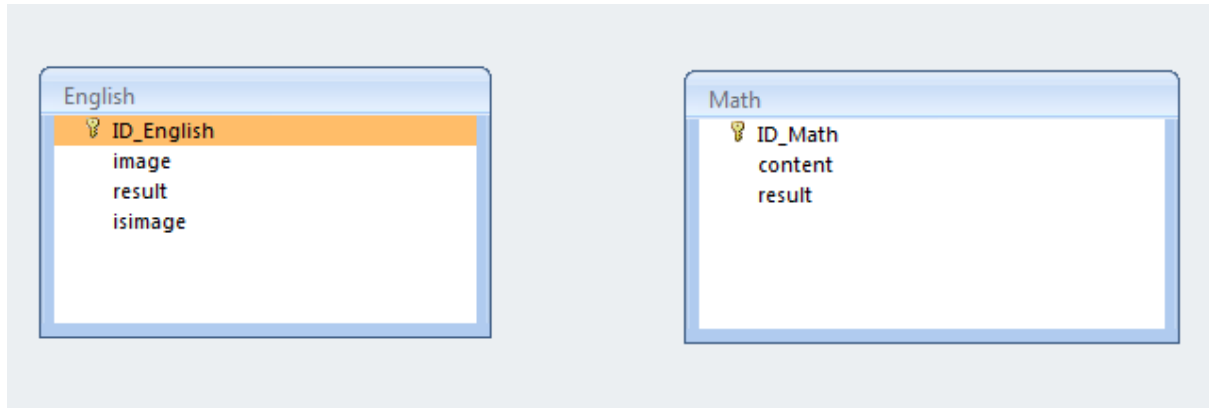
### 3.2. Mapping rules

When mapping entities to tables, the following rules were applied:

- Entities are mapped into tables in a one to one manner.
- Attributes are mapped to columns in a one to one manner.
- One-to-many relationships are mapped to foreign keys.
- Many-to-many relationship are implemented using two one-to-many

relationships with intersection table (if have)

### 3.3. Table Relationship Diagrams



### 3.4. Description of Table

#### a) Description of English table

No.	Field	Type	Length	Null/ Not Null	Key	Default	Description
1	<u>ID_English</u>	Character		Not Null	Primary key		Index of question English
2	image	Varchar	255	Not Null			Picture of question
3	result	Varchar	255				Result Correct of question
4	isimage	Integer					Check exist image of question



## b) Description of Math table

No.	Field	Type	Length	Null/ Not Null	Key	Default	Description
1	<u>ID_Math</u>	Character		Not Null	Primary key		Index of question Math
2	content	varchar	255	Not Null			Question of Math
3	result	varchar	255				Result correct of question

## 3.5. Key mappings

The following tables have primary keys created from sequences:

Table	Primary key column	Sequence
English	ID_English	Character
Math	ID_Math	Character