

**Document code: DN-DESIGN-v1.1**

**Project code: DN**

**Dekiru Nihongo**

Software Architectural Design

**Ha Noi, 01/10/2016**

Table of Contents

[1. Discussion of chosen Design 3](#_Toc463200597)

[2. Description of System Interface 4](#_Toc463200598)

[3. Component diagram 5](#_Toc463200599)

[3.1 Server 5](#_Toc463200600)

[3.2 Client 6](#_Toc463200601)

[4. Details Description of components 7](#_Toc463200602)

[4.1 Client 7](#_Toc463200603)

[4.1.1 UI 7](#_Toc463200604)

[4.1.1.1 Brief Description 7](#_Toc463200605)

[4.1.1.2 Class diagram 7](#_Toc463200606)

[4.1.1.3 Class diagram explanation 9](#_Toc463200607)

[4.1.2 DbAcess 17](#_Toc463200608)

[4.1.2.1 Brief Description 17](#_Toc463200609)

[4.1.2.2 Class diagram 17](#_Toc463200610)

[4.1.2.3 Class diagram explanation 17](#_Toc463200611)

[4.2. Server 19](#_Toc463200612)

[5. Sequence Diagram 19](#_Toc463200613)

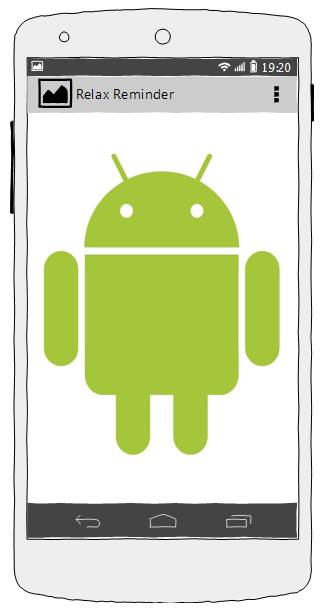
[5.1 Update newest data. 19](#_Toc463200614)

[5.2 Enter Lesson Content to Learn. 20](#_Toc463200615)

[5.3 Enter Learn Vocabulary 21](#_Toc463200616)

[5.4 Enter Quiz 22](#_Toc463200617)

# Discussion of chosen Design



DB

**SERVER**

Web services

SOURCE

**Client**

*Figure 1: System Overview*

* 1. **Server**
* The main controller receives the HTTP request
* Server has web services which allow clients to transfer data through HTTP request and JSON format
  1. **Client**
* Application is based on the MVC pattern

# Description of System Interface

Client 1

Client 2

Internet

1. Web Service

**CORE**

Operating System

Validation

Data management

….

*Figure 2: System Interface*

# Component diagram

## 3.1 Server

**Business**

Services

Model

**Data Access layer**

DBAccess

**Content**

**Database**



**Content Manager**

Figure 3.1: Server components

## 3.2 Client

Data Access Components

Content Synchronize Utilities

UI Components

**Presentation**

**Business**

**Data Access**

Content Database

**Local Database**

*Figure 3.2: Client Component*

Each layer has its own responsibility:

* **Presentation Layer:**
  + Presentation Layer has responsibility for displaying application’s GUI and in which the users interact with the Java Application.
  + Presentation Layer will receive input data from users, respond against these data and give off the result to users. This layer contents Java class used to design applications’ GUI, catch Events while users interacting with application.
* **Business Layer:**
  + Business Layer is a mediate layer; it lies between Data Access Layer and Presentation Layer. It has responsibility as a bridge, when users interact with the application 🡪Presentation Layer will receive input data and pass them to Business Layer 🡪 Business Layer processes these data and requests before they were transferred down to Data Access Layer.
  + Content Synchronize Utilities will get data from the server 🡪 process raw data then pass them to Data Access layer.
* **Data Access Layer:**
  + Data Access Layer is responsible for interacting with Database Management System (SQLite). Main activities of this layer are retrieving data, inserting, updating or deleting data, then response to users.
* **Database Layer:**
  + Database Layer is a place where store the data of application; it includes relational tables. We include database into the software.
* **Java Shared Preferences:** 
  + This layer contain all global setting for application
* **Java platform:** Java SDK 8

# Details Description of components

In order to understand logically the system, this part will describe the details of each component by separates function as SRS document. Each function will follow the general component design as above.

## 4.1 Client

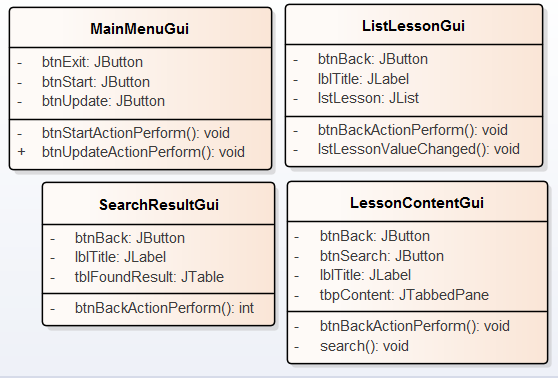
|  |  |  |  |
| --- | --- | --- | --- |
| No. | Package name | Description | Naming convention |
| 1 | UI | Contain all classes about UI | xxxGui |
| 2 | DbAccess | Contain DB interaction classes | xxxSql |

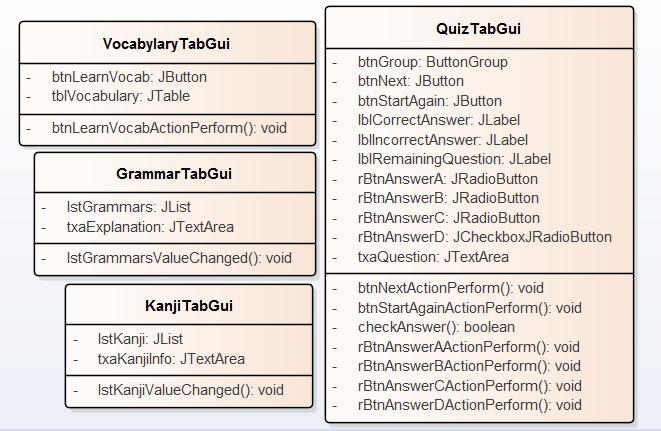
### UI

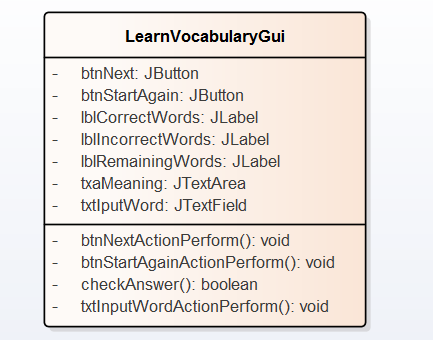
#### Brief Description

This package contain all classes provide user interface for user.

#### Class diagram







#### Class diagram explanation

##### 4.1.1.3.1 MainMenuGui.java

Attributes

| **Attribute** | **Type** | **Description** |
| --- | --- | --- |
| **btnStart**  Private | JButton | Get Button “Start” in MainMenuGui |
| **btnExit**  Private | JButton | Get Button “Exit” in MainMenuGui |
| **btnUpdate**  private | JButton | Get button “Update” in MainMenuGui |

Operations

| **Method** | **Purpose** | **Parameters** |
| --- | --- | --- |
| **btnStartActionPerform()** void  Private | Start Learn with DN app. |  |
| **btnUpdateActionPerform()** void  public | Update latest data from server. |  |

##### 4.1.1.3.3 ListLessonGui.java

Attributes

| **Attribute** | **Type** | **Description** |
| --- | --- | --- |
| **lblTitle**  Private | JLabel | Get “Title” in ListLessonGui. Display the name of DN book |
| **lstLesson** | JList | Get “List of Lessons” in ListLessonGui. Display list of lessons in DN book |
| **btnBack**  **Private-** | JButton | Get Button “Back” in ListLessonGui. Back to main menu  This button will be implemented to back to previous screen |

Operations

| **Method** | **Purpose** | **Parameters** |
| --- | --- | --- |
| **btnBackActionPerform()** void  Private | Back to main menu |  |
| **lstLessonValueChanged ()** void  Private | Access to a lesson when user click on an item. |  |

##### 4.1.1.3.3 SearchResultGui.java

Attributes

| **Attribute** | **Type** | **Description** |
| --- | --- | --- |
| **btnBack**  Private | JButton | Get Button "Back" in SearchResultGui  This button will be implemented to back to previous screen |
| **lblTitle**  Private | JButtom | Get “Title” in SearchResultGui  This label will be implemented to display name of DN Book |
| **tblFoundResult**  Private | JTable | Get Table “Result” in SearchResultGui |

Operations

| **Method** | **Purpose** | **Parameters** |
| --- | --- | --- |
| **btnBackActionPerform ()** void  Private | Back to previous screen |  |

##### 4.1.1.3.4 LessonContentGui.java

Attributes

| **Attribute** | **Type** | **Description** |
| --- | --- | --- |
| **btnBack**  Private | JButton | Get button "Back" in LessonContentGui.  This button will be implemented to save data to SharedPreferences. |
| **lblTitle**  Private | JLabel | Get “Title” in LessonContentGui.  This label will be implemented to display name of DN Book |
| **tbpContent**  Private | JTabbedPane | Get the TabPane contains all Tabs of a lesson. |
| **btnSearch**  Private | JButton | Get Button “Search” in LessonContentGui  This button will be implemented to search a word in all vocabulary |

Operations

| **Method** | **Purpose** | **Parameters** |
| --- | --- | --- |
| **btnBackActionPerform()** void  Private | Back to previous screen |  |
| **search()** void  Private | Search a word in all Vocabulary | - String word |

##### 4.1.1.3.5 VocabularyTab.java

Attributes

| **Attribute** | **Type** | **Description** |
| --- | --- | --- |
| **tblVocabulary**  Private | JTable | Get Table contains Vocabulary in VocabularyTabGui.  This table will be implemented to display all words and meanings of current lesson. |
| **btnLearnVocab**  Private | JButton | Get button "Learn Vocabulary" in VocabularyTabGui.  This button will be implemented to switch to LearnVocabularyGui |

Operations

| **Method** | **Purpose** | **Parameters** |
| --- | --- | --- |
| **btnLearnVocabActionPerform()** void  Private | Switch to Learn Vocabulary interface. |  |

##### 4.1.1.3.6 GrammarTab.java

Attributes

| **Attribute** | **Type** | **Description** |
| --- | --- | --- |
| **lstGrammar**  Private | JList | Get List contains Grammar in GrammarTabGui.  This table will be implemented to display all grammar structure of current lesson. |
| **txaExplanation**  Private | JTextArea | Get Text Area "Explanation" in GrammarTabGui.  This text area will be implemented to display grammar explanation. |

Operations

| **Method** | **Purpose** | **Parameters** |
| --- | --- | --- |
| **lstGrammarValueChanged()** void  Private | Allow user click on an item and the explanation of that grammar will be appear |  |

##### 4.1.1.3.7 KanjiTab.java

Attributes

| **Attribute** | **Type** | **Description** |
| --- | --- | --- |
| **lstKanji**  Private | JList | Get List contains Kanji in KanjiTabGui.  This table will be implemented to display all Kanji of current lesson. |
| **txaKanjiInfo**  Private | JTextArea | Get Text Area "Information" in KanjiTabGui.  This text area will be implemented to display Kanji information. |

Operations

| **Method** | **Purpose** | **Parameters** |
| --- | --- | --- |
| **lstKanjiValueChanged()** void  Private | Allow user click on an item and information of that Kanj will be appear |  |

##### 4.1.1.3.8 QuizTab.java

Attributes

| **Attribute** | **Type** | **Description** |
| --- | --- | --- |
| **btnGroup**  Private | ButtonGroup | Get Button Group in QuizTabGui.  This button will be implemented to make sure that user only pick one answer. |
| **btnNext**  Private | JButton | Get Button “Next” in QuizTabGui.  This button will be implemented to go to next question. |
| **btnStartAgain**  Private | JButton | Get button “Start Again” in QuizTabGui.  This button will be implemented to Start Quiz again from the beginning. |
| **lblRemainingQuestion**  Private | JLabel | Get label “Remaining” in QuizTabGui  This label will be implemented to display number of remaining questions. |
| **lblCorrectAnswer**  Private | JLabel | Get Label “Correct” in QuizTabGui  This label will be implemented to display number of correct answer. |
| **lblInCorrectAnswer**  Private | JLabel | Get Label “Incorrect” in QuizTabGui  This label will be implemented to display number of incorrect answer. |
| **txaQuestion**  Private | JTextArea | Get Text Area “Question” in QuizTabGui  This text area will be implemented to display the question. |
| **rBtnAnswerA**  Private | JRadioButton | Get button “A” in QuizTabGui  This button will be implemented to display Answer A |
| **rBtnAnswerB**  Private | JRadioButton | Get button “B” in QuizTabGui  This button will be implemented to display Answer B |
| **rBtnAnswerC**  Private | JRadioButton | Get button “C” in QuizTabGui  This button will be implemented to display Answer C |
| **rBtnAnswerD**  Private | JRadioButton | Get button “D” in QuizTabGui  This button will be implemented to display Answer D |

Operations

| **Method** | **Purpose** | **Parameters** |
| --- | --- | --- |
| **btnNextActionPerform()** void  Private | Go to next question. |  |
| **btnStartAgainActionPerform() void**  Private | Start quiz again from beginning. |  |
| **checkAnswer()**  Boolean | Check the answer wright or wrong. | - int questionNum  - String answer |
| **rBtnAnswerAActionPerform()**  void | User choose A. Call the checkAnswer() method. |  |
| **rBtnAnswerBActionPerform()**  void | User choose **B**. Call the checkAnswer() method. |  |
| **rBtnAnswerCActionPerform()**  void | User choose C. Call the checkAnswer() method. |  |
| **rBtnAnswerDActionPerform()**  void | User choose D. Call the checkAnswer() method. |  |

##### 4.1.1.3.9 LearnVocabularyTab.java

Attributes

| **Attribute** | **Type** | **Description** |
| --- | --- | --- |
| **btnGroup**  Private | ButtonGroup | Get Button Group in LearnVocabularyTab.  This button will be implemented to make sure that user only pick one answer. |
| **btnNext**  Private | JButton | Get Button “Next” in LearnVocabularyTab.  This button will be implemented to go to next question. |
| **btnStartAgain**  Private | JButton | Get button “Start Again” in LearnVocabularyTab.  This button will be implemented to Start Quiz again from the beginning. |
| **lblRemainingQuestion**  Private | JLabel | Get label “Remaining” in LearnVocabularyTab  This label will be implemented to display number of remaining questions. |
| **lblCorrectAnswer**  Private | JLabel | Get Label “Correct” in LearnVocabularyTab  This label will be implemented to display number of correct answer. |
| **lblInCorrectAnswer**  Private | JLabel | Get Label “Incorrect” in LearnVocabularyTab  This label will be implemented to display number of incorrect answer. |
| **txaQuestion**  Private | JTextArea | Get Text Area “Question” in LearnVocabularyTab  This text area will be implemented to display the question. |
| **txtInputWord**  Private | JTextField | Get Text field “Input” in LearnVocabularyTab |

Operations

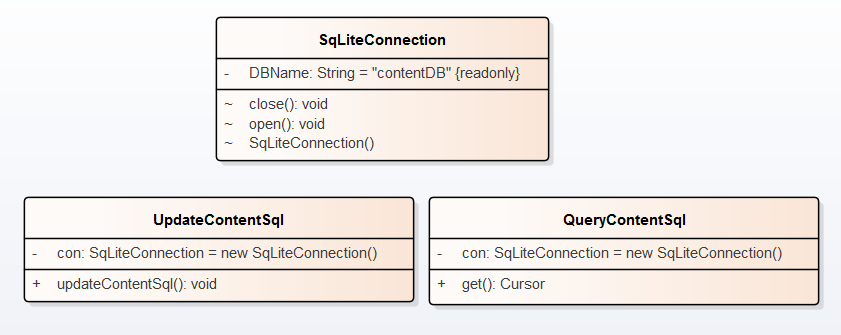
| **Method** | **Purpose** | **Parameters** |
| --- | --- | --- |
| **btnNextActionPerform()** void  Private | Go to next question. |  |
| **btnStartAgainActionPerform() void**  Private | Start Learn Vocabulary again from beginning. |  |
| **checkAnswer()**  Boolean | Check the answer wright or wrong. | - int questionNum  - String answer |
| **txtInputWordActionPerform()**  void | Get the text in inputted text field and call checkAnswer() method to confirm it. |  |

### DbAcess

#### Brief Description

This package contain all classes and function that involve interacting with database.

#### Class diagram



#### Class diagram explanation

##### 4.1.3.3.1 SqLiteConnection.java

Attributes

| **Attribute** | **Type** | **Description** |
| --- | --- | --- |
| **DBNAME**  Private | String | Name of Database in use. |

Operations

| **Method** | **Purpose** | **Parameters** |
| --- | --- | --- |
| **Close()** void  Package | - Close DB  - Display message on error |  |
| **open()** void  Package | - Open DB  - Display message on error |  |
| **SqLiteConnection()**  Package | - Constructor SqLiteConnection object, set DB source.  - Display message on error |  |

##### 4.1.3.3.2 queryContentSql.java

Attributes

| **Attribute** | **Type** | **Description** |
| --- | --- | --- |
| **con**  Private | SqLiteConnection | Instance of SqLiteConnection object |

Operations

| **Method** | **Purpose** | **Parameters** |
| --- | --- | --- |
| **get()** Cursor  Public | Get content data and pass to Main Menu.  Display message on error |  |

##### 4.1.3.3.3 UpdateContentSql.java

Attributes

| **Attribute** | **Type** | **Description** |
| --- | --- | --- |
| **con**  Private | SqLiteConnection | Instance of SqLiteConnection object |

Operations

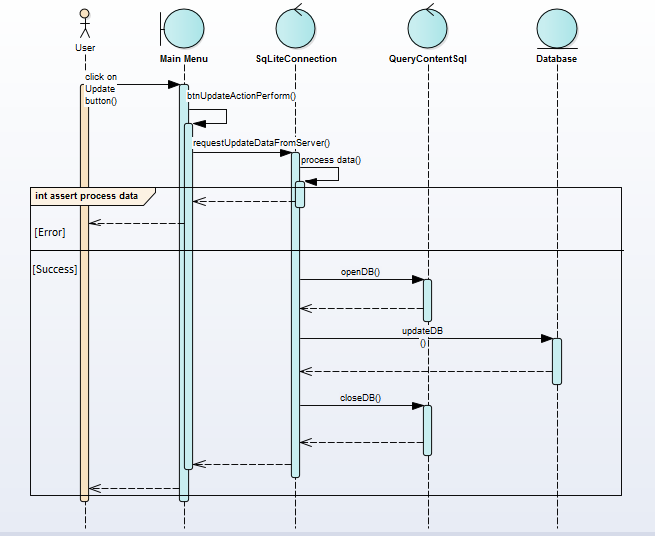
| **Method** | **Purpose** | **Parameters** |
| --- | --- | --- |
| **updateContentSql()**  Public | Update newest database from server |  |

## 4.2. Server

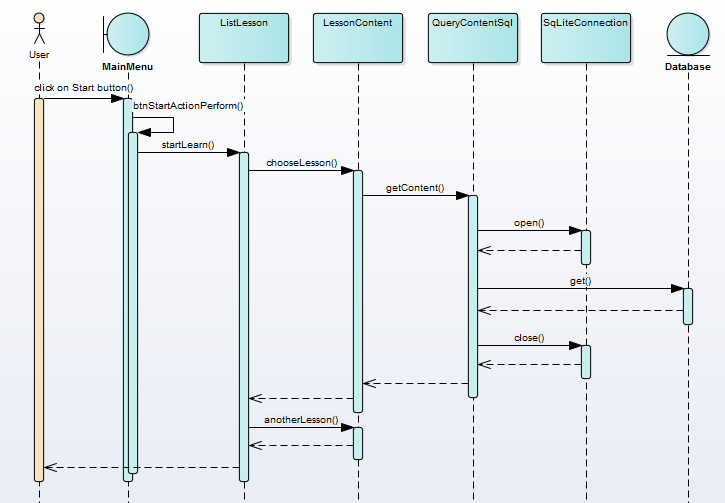
This Part is not completed because it’s out range of our knowledge. We will fill in this part later.

# Sequence Diagram

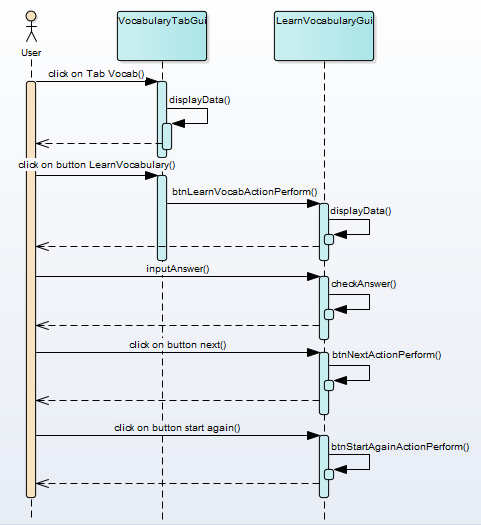
## 5.1 Update newest data.



## 5.2 Enter Lesson Content to Learn.



## 5.3 Enter Learn Vocabulary



## 5.4 Enter Quiz

