Child Education Software

Arindam Ghosh

# Index

Contents

[Index 2](#_Toc351476512)

[Introduction: 4](#_Toc351476513)

[**Figure:** Overview of Child Education Software 5](#_Toc351476514)

[Objective: 5](#_Toc351476515)

[System Analysis 6](#_Toc351476516)

[Identification of Need: 6](#_Toc351476517)

[Preliminary Investigation: 6](#_Toc351476518)

[Feasibility Study: 6](#_Toc351476519)

[Project Planning & Scheduling: 6](#_Toc351476520)

[Gantt chart 6](#_Toc351476521)

[Tracking Gantt 7](#_Toc351476522)

[Pert Chart 7](#_Toc351476523)

[Software requirement specifications (SRS): 8](#_Toc351476524)

[Functional Requirement 9](#_Toc351476525)

[Non-functional Requirements 11](#_Toc351476526)

[Software Engineering Paradigm applied 12](#_Toc351476527)

[Data models 12](#_Toc351476528)

[Context Diagram 12](#_Toc351476529)

[0-Level DFD 13](#_Toc351476530)

[1-Level DFD 14](#_Toc351476531)

[2-Level DFD 15](#_Toc351476532)

[Sequence diagrams 15](#_Toc351476533)

[Entity Relationship Model, 18](#_Toc351476534)

[Class Diagrams 19](#_Toc351476535)

[Activity Diagrams 20](#_Toc351476536)

[System Design 23](#_Toc351476537)

[Modularisation details 23](#_Toc351476538)

[DNBSN Engine: 23](#_Toc351476539)

[DNBSN GUI: 23](#_Toc351476540)

[DNBSN Storage: 23](#_Toc351476541)

[Google Calendar: 23](#_Toc351476542)

[Facebook/Twitter/LinkedIn API: 24](#_Toc351476543)

[Data integrity and constraints 24](#_Toc351476544)

[Entity integrity 24](#_Toc351476545)

[Referential Integrity 24](#_Toc351476546)

[Domain Integrity 24](#_Toc351476547)

[User Defined Integrity 24](#_Toc351476548)

[Database design 24](#_Toc351476549)

[User Interface Design 28](#_Toc351476550)

[Test Cases (Unit Test Cases and System Test Cases) 28](#_Toc351476551)

[Coding 28](#_Toc351476552)

[Complete Project Coding 28](#_Toc351476553)

[Comments and Description of Coding segments 28](#_Toc351476554)

[Standardization of the coding 28](#_Toc351476555)

[Code Efficiency 28](#_Toc351476556)

[Error handling 29](#_Toc351476557)

[Parameters calling/passing 29](#_Toc351476558)

[Validation checks 29](#_Toc351476559)

[Testing 29](#_Toc351476560)

[Testing techniques and Testing strategies used 29](#_Toc351476561)

[Testing Plan used 29](#_Toc351476562)

[Test reports for Unit Test Cases and System Test Cases 29](#_Toc351476563)

[Debugging and Code improvement: 29](#_Toc351476564)

[System Security measures: 29](#_Toc351476565)

[Database/data security: 29](#_Toc351476566)

[Creation of User profiles and access rights 29](#_Toc351476567)

[Cost Estimation of the Project along with Cost Estimation Model 29](#_Toc351476568)

[Reports (sample layouts should be placed) 30](#_Toc351476569)

[Future scope and further enhancement of the Project 31](#_Toc351476570)

[Bibliography 31](#_Toc351476571)

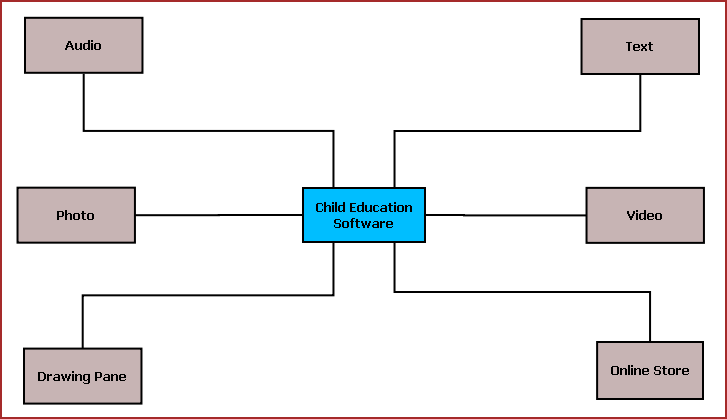
[Appendices (if any) 31](#_Toc351476572)

[Glossary. 31](#_Toc351476573)

# Introduction:

Child Education Software will enable a new aspect of teaching children with the help of modern technology. It will incorporate audio visual teaching method with user friendly interface. Almost every family is having a computer now. We are used to computerized system in every walk of our life. So kids also will be attracted towards computer. We can use computer as a mean of teaching new things. Because of rich multi-media content kids will be happy to use computer for learning new topics.

In Child Education Software users can use the existing tutorial as well as they can download new tutorial from online store. It will be an eco-friendly option. Children wastes paper a lot and they get bored as the colour of the fades gradually. The child education software will make learning more efficient, attractive and tech-savvy for the children.



**Figure 1:** Overview of Child Education Software

The main features of this software are listed below:

* + Textual display of teaching material.
  + Video display for video tutorial.
  + Sound system to play audio clips.
  + Drawing Pane or Handwriting practice area.
  + Photo viewer.
  + Easy navigation.
  + Interactive UI.
  + Printing of selected content.
  + Online store for buying new tutorials and upgrades.

## Objective

The objectives of the projects are given below:

* To provide a new kind of medium for educating children
* To cop up with technological revolutions happening around us and to involve the kids with new technology
* To make education process more interesting to the kids
* To utilize our electronic gadgets to do more meaningful work like teaching children
* To save paper waste and to take more green initiatives
* By accomplishing this project I could learn new technologies like .NET, C#, XML and I am able to be involved with the complete software development lifecycle.

# System Analysis

## Identification of Need:

Now we are used to computerized system in every walk of our life and kids also will be attracted towards computer. We can use computer as a mean of teaching new things. Because of rich multi-media content kids will be happy to use computer for learning new topics. Since we are also dedicated to the education we are always searching for ways to create educational material which starts from "simple" alphabets to complex grammar exercises and study material for other subjects such as science, mathematics, history etc. Our primary focus is on schoolchildren aged 3 to 18, and the specialized user interface needs of young users. However, we will also have programs to aid teachers in planning lessons, and others that are of interest to university students and anyone else with a desire to learn.

CES will allow participants to have lessons from offline resources as well as online study materials. They can also do a hands-on with the exercises and evaluate their progress. Additional study material and tool can be availed by purchasing modules from various online resources.

## Preliminary Investigation:

From everywhere I tried to spoke with the parents and guardian about this modern system of education. All they agree with may think about this application. I tried to collect opinion from software experts and took their opinion to develop the child education system.

## Feasibility Study:

Feasibility study is an essential requirement of any proposed system. It proposes one or more conceptual solutions to the problem set for the project. It is the review of the findings analyzed so far. Before proceeding directly into the design phase I had to check whether this project is worth doing. For this I carried out feasibility study of the proposed system in 3 different ways

* To check whether the project is technically feasible that is whether I have the necessary skills and know-how to complete the project.
* To check whether the proposed system is easy to use and that it satisfies the user objectives (operational feasibility) and can be fitted into current system operation.
* To determine whether the project’s goal can be achieved within the resource limits allocated that is to find out whether the project is economically feasible.

### Technical Feasibility

It has been already mentioned that ‘**Child Education System’** is purely a desktop based project with some mobile application and related web-based project. A lots of such types of projects have already been made or been running to make every day. So it is not even technically impossible rather difficult to build such a software. The technical software knowledge that is required for desktop is C#, Mysql, WPF for mobile apps is nokia web tool and some anyone who has a basic knowledge of computer software and has a good sense of design can almost be an expert in handling these software. As the project will proceed one can easily make himself informative about ASP to do the project. So not very much technical capability is required by the project. We see that the project is technically feasible and worth to do because of the reasons specified above.

### Operational Feasibility

To know whether the proposed system is acceptable to the end user, they were subjected to a mini prototype. The users were asked to respond as to how they feel about the system. To a large extent the users were satisfied with the prototype, which I have shown to them, minor modifications were also done to closely match with the users’ requirement.

To implement the user requirements in a software system is the true goal of a systems analyst. I rigorously followed the requirements of the user, what they want from the system and how it should help them. Prototyping results show that the proposed system is acceptable to the user.

### Economic Feasibility

If it is seen from real life view then MY PROJECT IS ECONOMICALLY FEASIBLE as this sort of software is going to get the market. As these types of engineering colleges emerge they all will try to communicate with at most people they can. When this communication is required portal will be the most wanted communication medium that they rely for. So this type of web-based project will always have a commercial aspect. So I am not wasting my time in a non-economic feasible project.

Technology is growing up day by day. The parents can easily afford to buy various software for their children’s education. If we keep the cost in comfortable label with the basic feature, which help to maintain the child education system, we can sell the application. And according to their need we can modify the applications and customize it for their help.

## Project Planning & Scheduling:

### Gantt chart



### Tracking Gantt



### 

### Pert Chart



## Software requirement specifications (SRS):

### Functional Requirement

#### Audio visual display of study materials:

**Introduction**:

User should be able to watch animated video lessons as well as audio lessons.

**Input**:

Click on show lesion button.

**Processing**:

System will find and fetch a predefined lesion and display that.

**Output**:

Kids can watch them and learn new things.

#### Exercise for students:

**Introduction**:

There should be relevant and user friendly exercise topics.

**Input**:

User will click on start practice button.

**Processing**:

System will open relevant practice set for kids from database.

**Output**:

Kids can interact with the graphical practice sets and solve them.

#### Drawing pane for practice

**Introduction**:

There should be a drawing pane for kids with an attractive GUI.

**Input**:

User will click on start drawing button and kids will start interacting with the GUI.

**Processing**:

Drawn shapes will be saved temporarily in memory and could be permanently kept in database.

**Output**:

Stored drawings will be shown to the user whenever is asked.

#### Download new lesions:

**Introduction**:

New lesions could be downloaded from the web if user wants.

**Input**:

User clicks on download new lesions button.

**Processing**:

System searches for relevant data in the previously stored online database and copies them in the local system.

**Output**:

System shows the newly downloaded lesions to the user.

#### Learn about various topics like nature, economy etc.

**Introduction**:

There should be various GK tutorials with relevant images and examples.

**Input**:

User clicks on the learn button.

**Processing**:

Various options provided to the user and as per his choice a new image and description gets opened.

**Output**:

Students get to see topics with images and examples that help them remember the topics better.

#### Online payment

**Introduction**:

For purchasing lesions, user should be able to pay online using debit card, internet banking, credit card etc.

**Input**:

User opts for online payment option in the application GUI.

**Processing**:

After getting required details, system takes user to his banker’s website and deducts the required amount. Generates a online bill as well.

**Output**:

User gets the digital bill’s copy in his email inbox and gets the new lesion in his system as well.

#### Easily sharable downloaded lesions:

**Introduction**:

People should be able to share the lesions to other computers.

**Input**:

User clicks on share the lesions button

**Processing**:

System takes the tutorials from database and sends them to the portable device as per user choice for sharing.

**Output**:

User gets the data and could easily copy them to the other portable device.

#### Generate grade card

**Introduction**:

After completing exercises, a grade card should be generated.

**Input**:

Kids complete the exercises within the given time.

**Processing:**

After matching the answers with predefined answers, system generates a report card and saves them in database.

**Output**:

User can see the report card after the exam and see them in future as well.

## Non-functional Requirements

* The application will be self-dependent and no dependency on other parties required.
* There will be a digital backup and restore system.
* There will be more opportunity to extend the application in various type of device in future.
* The response time will be low and the system will response fast.
* It will be very user friendly and usable by any person with minimal computer knowledge.
* In terms of security unauthorized access will be denied and register user will be able to change as necessary.
* It will be efficient as it reduces manual labor and searching.
* DNBSN will have user manual and help documents.
* It is designed such a way that it can be maintained with minimal effort.

## Data models

### Context Diagram

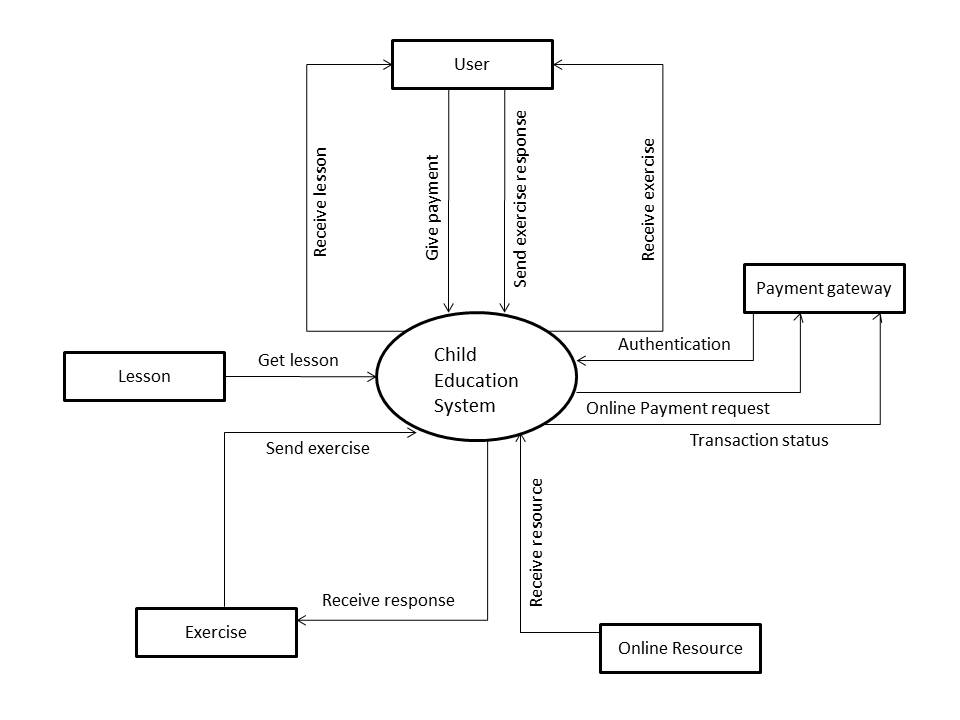


Figure 1: Context Diagram of CES

### 0-Level DFD

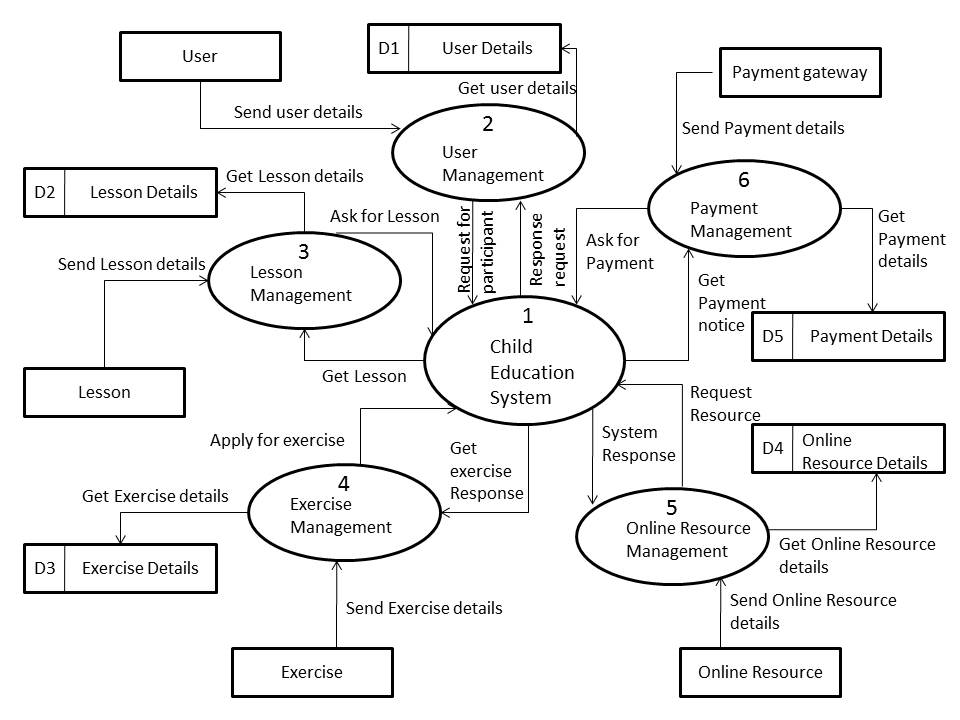


Figure 2: DFD level 0 of CES

### 1-Level DFD

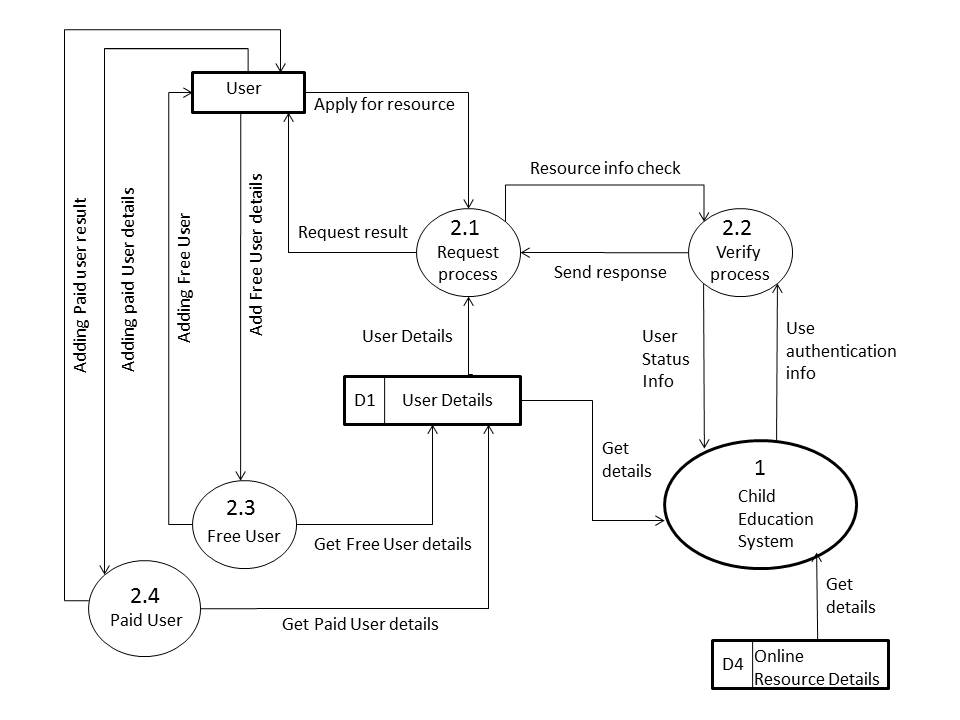


Figure 3: DFD level 1 of CES

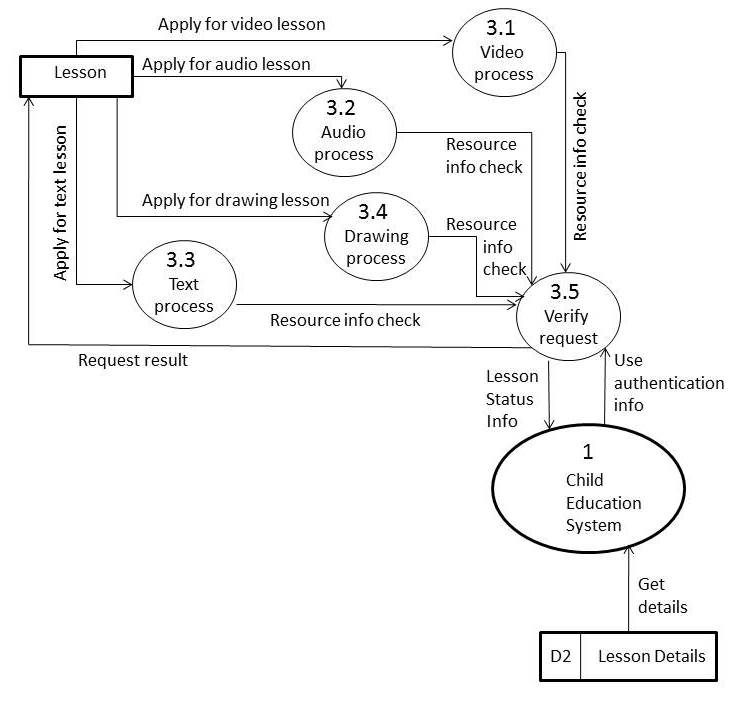


Figure 4: DFD level 1 of CES

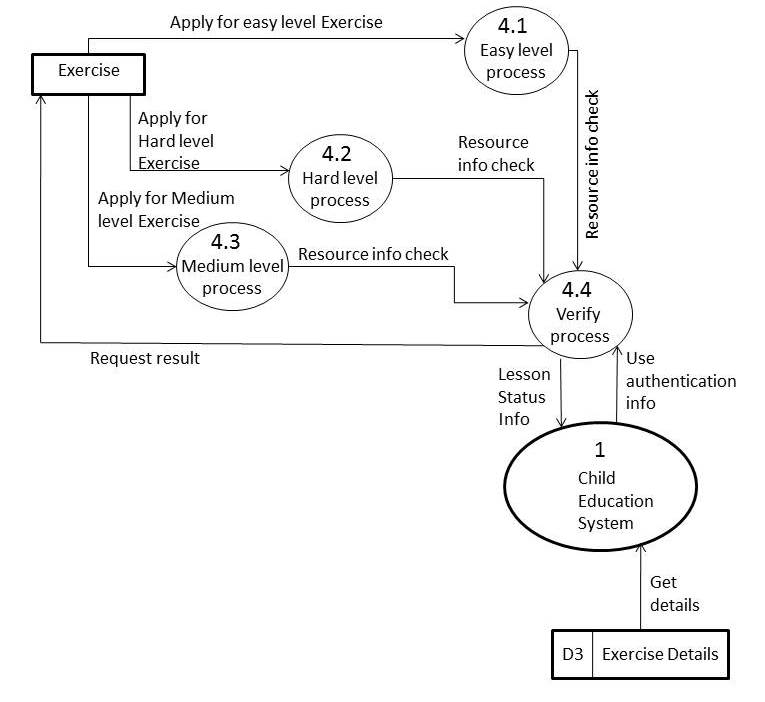


Figure 5: DFD level 1 of CES

### 2-Level DFD

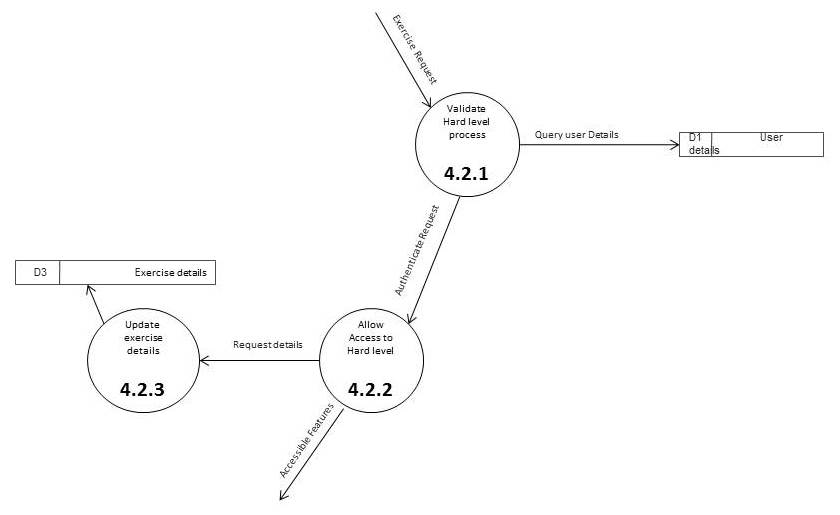


Figure 6: DFD level 2 of CES

## Sequence diagrams

## Entity Relationship Model,

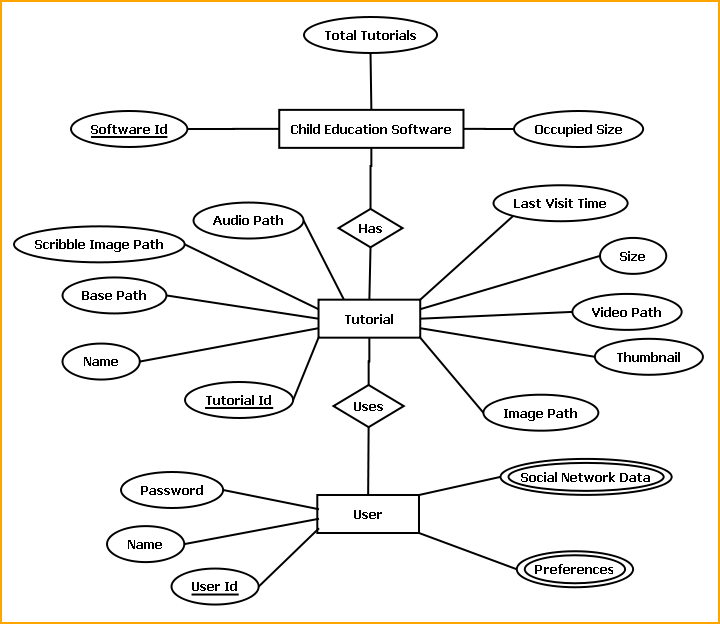
We will design a RDBMS for Child Education Software. The entities and their attributes are listed below. Attributes in Bold letter is the unique key.

|  |  |
| --- | --- |
| **Entities** | **Attributes** |
| Tutorial | **Tutorial Id**, Name, Base Path, Last Visit Time, Size, Thumbnail, video path, image path, audio path, scribble image path |
| Child Education Software | **Software Id**, Total tutorials , Occupied Size |
| User | **User Id**, Name, Social Network Data, password, Preferences. |

**Relationship between Entities:**

* Child Education Software has Tutorials 🡪 1 : N
* User uses tutorials 🡪 M : N

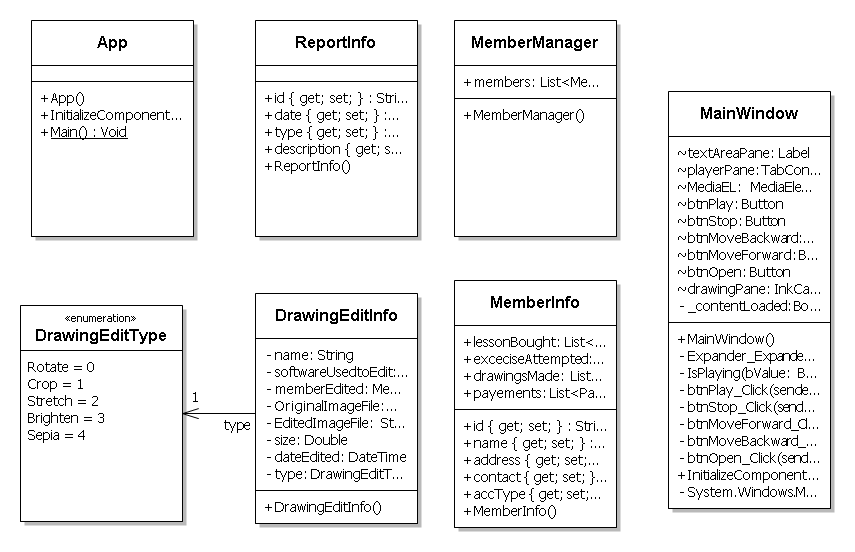
## E-R Diagram

****

E-R Diagram of Child Education Software

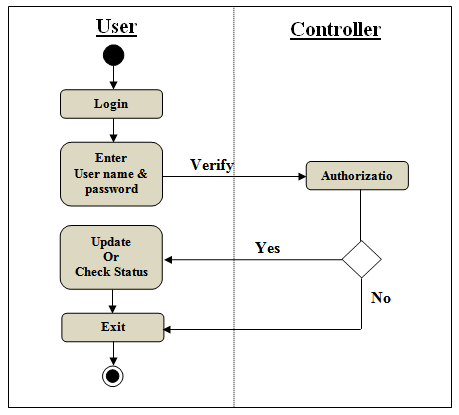
## Class Diagrams

****

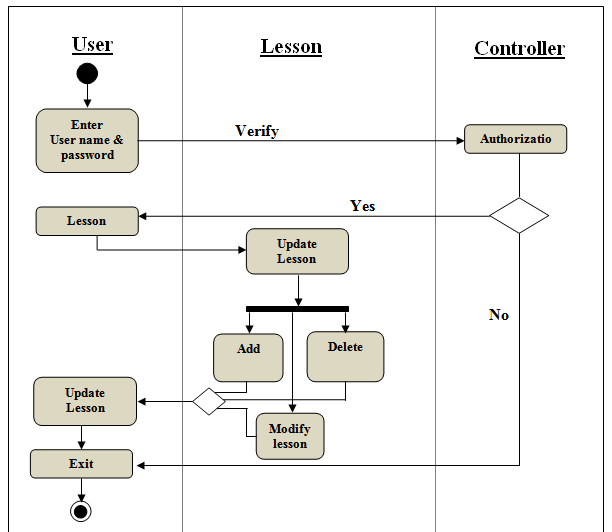


## Activity Diagrams

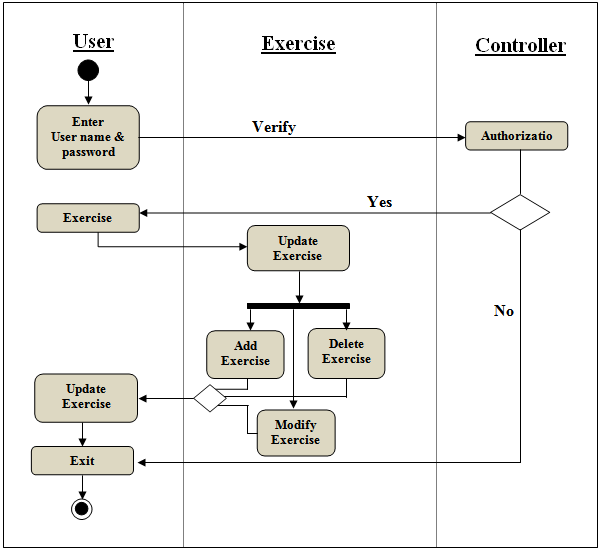
### User Login



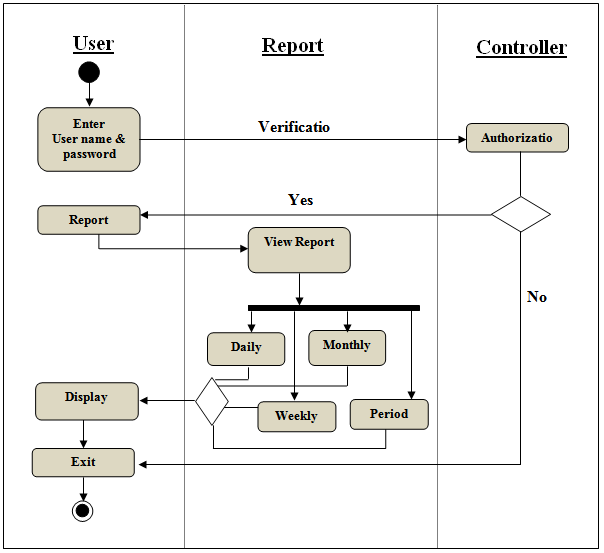
### Lesson Management



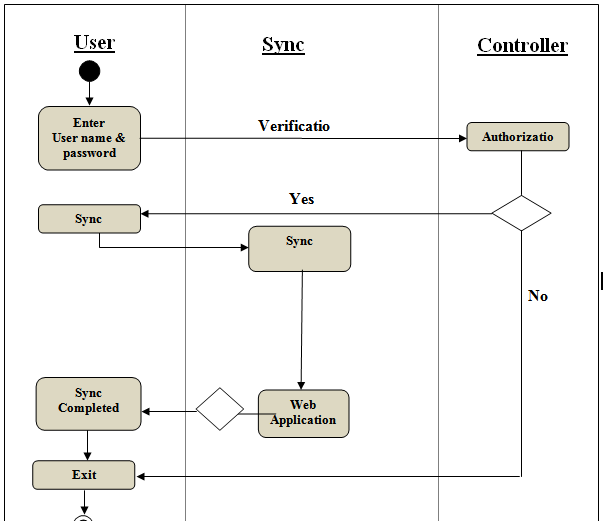
### Exercise Management



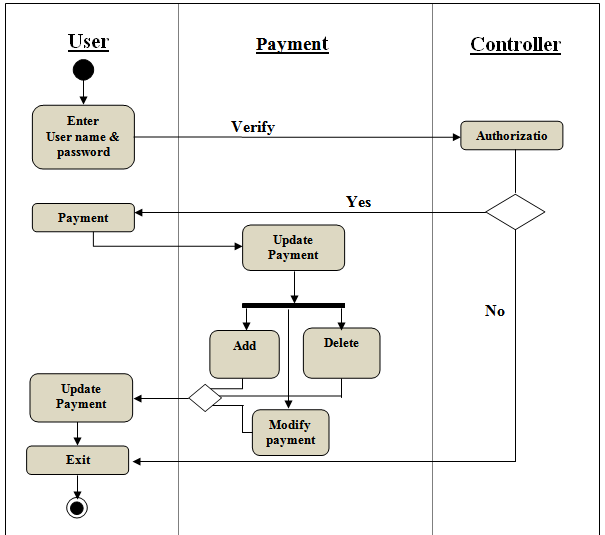
### View Report



### Sync



### Payment Management



# System Design

## Modularisation details

# Database & Table Details

The database used for this software is called CESdb. Database tables and corresponding keys are shown in tabular form. It shows the tables and its columns. A key in Bold is the primary key.

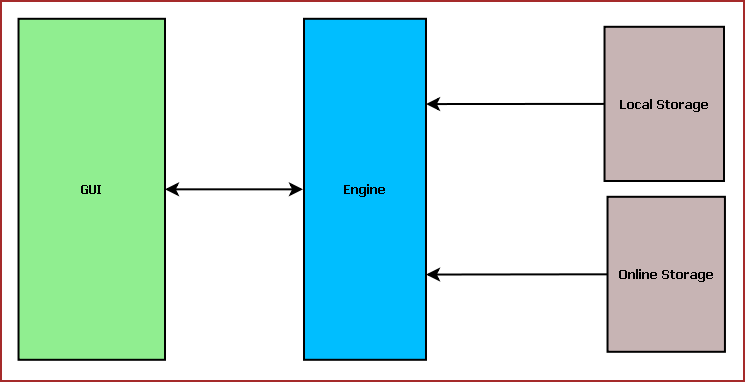
|  |  |
| --- | --- |
| **Tables** | **Keys** |
| Tutorial | **Tutorial Id**, Name, Base Path, Last Visit Time, Size, Thumbnail, video path, image path, audio path, scribble image path |
| Child Education Software | **Software Id**, Total tutorials , Occupied Size |
| User | **User Id**, Name, Social Network Data, password, Preferences. |

# Complete Structure

## Module Description

Child Education Software is divided into three main modules.

* GUI
* Engine
* Local Storage & Online Storage



**Figure:** Child Education System Components

### Child Education Software GUI:

Child Education Software GUI will display tutorial, navigator, online store interface. The main components of GUI are:

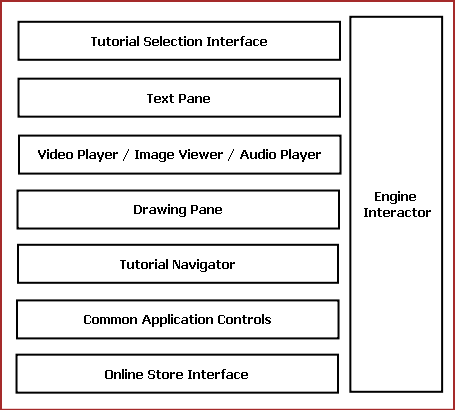
* Tutorial Selection Interface
* Text Pane
* Video Player / Image Viewer / Audio Player
* Drawing Pane
* Tutorial Navigator
* Common Application Control
* Online Store Interface
* Engine Interactor

#### Tutorial Selection Interface:

This interface will display the available tutorials from local storage and the tutorials bought from Online Store. From here the user will select the tutorial to be viewed. The tutorials will be displayed according to different category such as: Age, Subject, culture etc.

#### Text Pane:

This area will display the text portion of the tutorial. This will contain the description about the subject if different colours. Users can select text, copy and paste the text.



**Fig:** Components of Child Education Software GUI

#### Common Controls:

Common controls include several controls of the application. Such as: Menu Bar, Toolbar, status bar, Context Menu and progress bar.

#### Engine Interaction:

This module handles the interaction between GUI and Engine. This module defines all callback for the Engine events.

#### Video Player / Image Viewer / Audio Player:

This Pane displays the image, video & audio artwork associated with the tutorial content. This will have two components such as: Display Area & Control Pane. Display Area will display the content. Control Pane contains the control like Play, Pause, Previous, Next for controlling the displayed content.

#### Drawing Pane:

Drawing Pane allows user to draw, sketch & scribble. This will help the kids to draw and practice while learning. User can save the drawing as a Image or load a image to scribble.

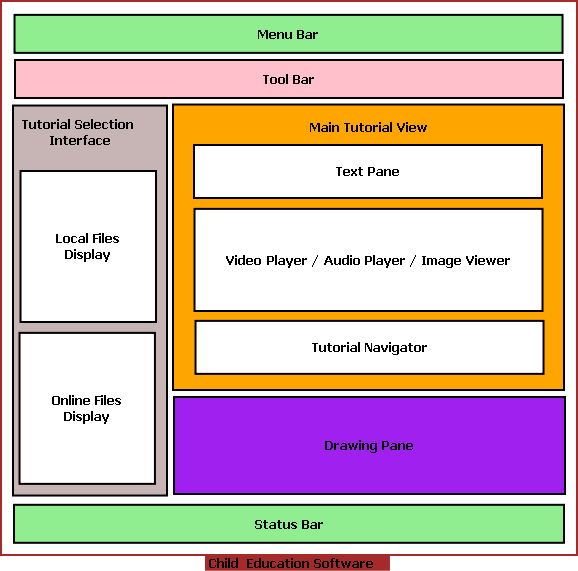
#### Tutorial Navigator:

This pane contains the controls for navigating within the tutorial. It will have “Next”, “Previous” and “Go to Home” options.

#### Online Store Interface:

This pane allows user to login into Online Store. Then it will display the available tutorial online for download. User may need to purchase or download it for free.

The GUI layout of Child Education Software is shown below:



### Child Education Software Engine:

Child Education Software Engine provides multimedia framework for displaying Video/Image, tutorial controlling logic. It is mainly divided into following components:

* Tutorial Controller
* Image Controller
* Archive Manager
* Video /Audio Controller
* Online Store Controller
* GUI Interactor
* Database Manager

#### Tutorial Controller:

This module defines the tutorial format. It controls tutorial flow, content to be displayed. Every tutorial is internally a XML file mentioning the tutorial contents.

Tutorial controller has a XML parser to read the XML and load tutorial contents accordingly. Tutorial Controller controls the other engine components to coordinate tutorial display.

Different components of Child Education Software Engine and their interaction with other modules are displayed in the following diagram.

## User Interface Design

Screen shot

## Test Cases (Unit Test Cases and System Test Cases)

# Coding

## Complete Project Coding

Code

## Comments and Description of Coding segments

Various types of comments and description we use in our coding section. Some of them are:

//open the connection

This comment is use at the data interaction section where the code to open the MySql connection.

//define the command reference

To define a command reference in MySql.

//define the connection used by the command object

To define the connection, which is used by the comment object.

//always close the connection

It is indicating to close connection after code is executed.

Manu Unused code in our project we did comment them also like :  
<!--<Condition Property="Password" Value="c" />-->

## Standardization of the coding

12

## Code Efficiency

We started working on the project keeping in mind that we must develop it in a way that it not only provides a very easy to use GUI but also provide a fast and flexible service to the users. We know that a particular work can be done in more than one ways. We have tried all the options and then chose the one which provides the fastest and most secure performance. First of all, we have used the latest technologies of Microsoft like visual studio 2010 as IDE and WPF as GUI to keep our application’s performance few steps ahead. We have studies all the rules of software development life cycle and applied them to keep our application flexible. We have given special attention to the storage related codes. We have avoided all the unnecessary database codes and kept them as short as possible without harming our purpose so that insertion, updating, deletion and fetching of data take place flexibly. You can see the result as a user; our application does all the works very smoothly.

## Error handling

## Parameters calling/passing

## Validation checks

# Testing

## Testing techniques and Testing strategies used

## Testing Plan used

## Test reports for Unit Test Cases and System Test Cases

## Debugging and Code improvement:

# System Security measures:

## Database/data security:

It encrypts the data stored in the database so that even if someone succeeds to hack the database still not much harm could be done.

The application will use Google open-id authentication for web interface.

## Creation of User profiles and access rights

The software requires a predefined username and password to login.

It allows a guest login as well which lets a guest user user this application with very limited access to the user data.

# Cost Estimation of the Project along with Cost Estimation Model

We used the basic COCOMO model, which gives an approximate estimate of our **CES** project parameters. The basic COCOMO estimation model is given by the following expressions:

Effort = a1 \* (KLOC)a2 PM

Tdev = b1 \* (Effort)b2 months

Where

KLOC is the estimated size of the software product expressed in Kilo Lines of Code a1, a2, b1, b2 are constants for each category of software products.

Tdev is the estimated time to develop the software, expressed in months.

Effort is the total effort required to develop the software product, expressed in person-month (PM).

Our project is semidetached type, because the development team consists of a mixture of experienced and inexperienced staff like my guide and me. Team members may have limited experience on related system but may be unfamiliar with aspects of the system being developed.

## Estimation of development effort

For our Semi-detached class software product **CES**, the formula for estimating the effort based on the code size is shown below:

Semi-detached **CES**: Tdev = 3.0\*(KLOC)1.12 PM

## Estimation of development time

For our Semi-detached class software product **CES**, the formula for estimating the development time based on the effort is given below:

Semi-detached **CES**: Tdev = 2.5\*(Effort)0.35 months

Assume that the size of a Semi-detached CES product has been estimated to be 4,000 lines of source code. Assume that the average salary of software engineer(me) is Rs. 15,000 per month.

Assume that the size of our

The basic COCOMO estimation formula for CES semidetached software:

Our Effort = 3.0 \* (4)1.12 PM

= 14 PM

Normal Development time = 2.5 \* (14)0.35 months

= 6 months

Cost required to develop the product = Rs. 6 \* 15,000

= Rs. 90,000

# Reports (sample layouts should be placed)

* List of Lesion updates could be downloading.
* Hand writing could be generated.
* A list of events could be generated.
* Typing test could be generated.
* List of google plus update could be generated.

# Future scope and further enhancement of the Project

* Now it will display the text based RSS feeds and link of the multimedia contents. We will display the Multimedia contents like Video, Audio & Image in future.
* To support UNIX / Linux Based Operating systems.
* To Support Mobile Operating systems for Symbian, Meego & Android.

# Bibliography

* <http://en.wikipedia.org>
* <http://msdn.microsoft.com/en-us/>
* <http://www.microsoft.com/en-us/default.aspx>
* <http://www.codeplex.com/>
* <http://stackoverflow.com/>
* <http://www.codeguru.com/>
* <http://www.w3schools.com>
* <http://blogs.technicise.com/>
* [www.mysql.org](http://www.mysql.org)

# Appendices (if any)

# Glossary.