Child Education Software

*Arindam Ghosh* **Table of Contents**

Introduction & Objective of the Project: 4

Introduction 4

Objective 5

Project Category: 5

Project Structure: 6

Hardware and Software Specification 6

Hardware Requirement 6

Software Requirement 6

Tools/Platforms used (Hardware/Software): 6

Hardware: 6

Software: 6

REQUIREMENTS AND ANALYSIS 7

Problem Definition 7

Requirements Specification 7

Functional Requirement 7

Register User 7

Login User 7

Update Note or Event 8

Load Google Map 8

Share event in Google calendar 8

Search Event 9

Add Contact 9

Add To Do 9

Technical Specification 10

Planning and Scheduling 10

Gantt chart 11

Tracking Gantt 11

Pert Chart 12

Scope of the Solution 13

Analysis 13

Context Diagram 13

0-Level DFD 14

1-Level DFD 15

2-Level DFD 16

E-R Diagram 16

Class Diagram 18

Database & Table Details 19

Complete Structure 19

Module Description 19

Estimation 21

Data Structure 22

Implementation Methodology 25

List of Reports 25

Implementation of Security Mechanism at Various Levels 25

Future Scope and Further enhancement of the project: 26

Bibliography 26

# 

# Introduction & Objective of the Project:

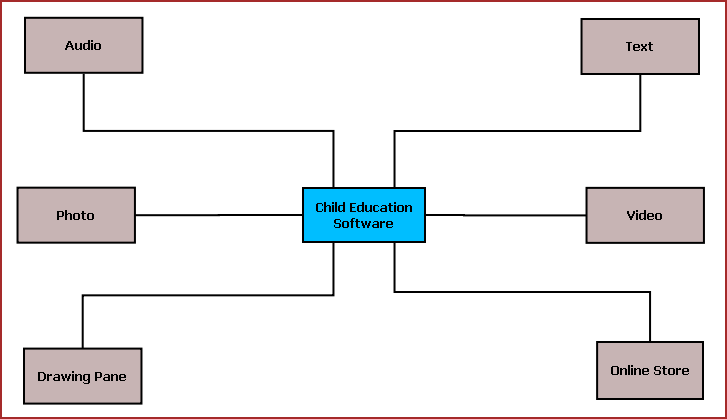
## Introduction

|  |
| --- |
| * Facebook integration * Twitter integration * Google Calendar integration * Linkedin integration * Google plus integration * Google Maps integration |

* Maintaining To Do/Task list

Reminder/ alarm for scheduled task 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.

# Project Category:

This software will follow Object Oriented Programming Paradigm and use below mentioned areas.

**Front End/ GUI Tools:** Visual Studio 2010, .NET 4, C#, WPF

**Backend:** MySQL 5.5.15

**Internet Technologies:** HTML, JavaScript, PHP, Google App Framework, facebook,SDK,twitterSDK,LinkedIn SDK.

**Operating Systems**: Windows XP, Windows 7

**Applications**: PC Utility application

# Project Structure:

## Hardware and Software Specification

### Hardware Requirement

* **Disc capacity :** 10 MB of available hard disk space
* **RAM :** 1 GB (32 Bit) or 2 GB (64 Bit)
* **Processor :** 1.6GHz or faster
* DVD-ROM Drive / USB **Port**

### Software Requirement

* Windows XP (x86) with Service Pack 3 / Windows Vista (x86 & x64) with

Service Pack 2 / Windows 7 (x86 & x64)

* Microsoft .NET 4.0

## Tools/Platforms used (Hardware/Software):

### Hardware:

* Laptop with 2GHZ processor
* 2 GB RAM
* 320 GB hard disk (NTFS File System)
* 8 GB USB Pen Drive (FAT File System)

### Software:

* Windows XP (x86) with Service Pack 3 / Windows Vista (x86) with Service Pack 2 / Windows 7 (x86)
* Microsoft .NET 4.0
* Windows Presentation Framework(WPF)
* Windows Communication Framework(WCF)
* Visual Studio 2010 Express Edition (IDE)
* MySQL database
* Dia for Diagram Drawing & Modeling.
* Google App Framework
* Facebook Php Sdk & FBML
* Twitter Sdk
* LinkedIn Sdk
* RSS Feed Burner Sdk

# REQUIREMENTS AND ANALYSIS

## Problem Definition

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.

## Requirements Specification

### Functional Requirement

### Technical Specification

**Front End/ GUI Tools:** Windows Presentation Framework (WPF)

**IDE:** Visual Studio 2010

**Framework:** Microsoft .NET 4.0

**Database:** MySQL

**Database Tool:** MySQL workbench CE

**Operating Systems**: Windows XP, Windows 7

**Others**: MySQL .NET connector, Facebook/ Twitte/ LinkedIn SDK, Google Map

**Cloud Technology**: Google calendar, Google Drive

## Planning and Scheduling

### Gantt chart



### Tracking Gantt



### Pert Chart



# Scope of the Solution

Daily Notebook allows user to manage his entire updates and events of all the leading daily social networking sites. It allows us to note down our thoughts inside the application and update them whenever we want to single/ multiple social networking sites at once. They can keep track of the status updates and the response of his social networking website buddies. Not only that, it allows us to add remainder of various events and synchronize them with Google calendar later so that we could access them from anywhere.

# Analysis

## Context Diagram



## 0-Level DFD



## 1-Level DFD



## 2-Level DFD



## E-R Diagram

We will design a RDBMS for Daily notebook & Social Networking Updater. The entities and their attributes are listed below. Attributes in Bold letter is the unique key.

|  |  |
| --- | --- |
| **Entities** | **Attributes** |
| Note | **Note Id**, Content, Time, Size, Web Service id, user |
| RSS Feed | **RSS Feed Id** , Web Service Id, Content, time, size |
| Daily notebook & Social Networking Updater | **Sw Id**, Web Services Supported, Users, Size |
| User | **User Id**, Name, Social Network Data, Preferences. |
| Web Service | **Web Service Id,** Authentication Data, Feed Data, Preferences. |

**Relationship between Entities:**

* Daily notebook & Social Networking Updater has User 🡪 1 : N
* Users post Notes 🡪 1 : N
* Web Service generates Feeds 🡪 M : N



E-R Diagram of Daily notebook & Social Networking Updater

## Class Diagram



# Database & Table Details

The database used for this software is called **Dnbdb**. 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** |
| Note | **Note Id**, Content, Time, Size, Web Service id, user |
| RSS Feed | **RSS Feed Id** , Web Service Id, Content, time, size |
| Daily notebook & Social Networking Updater | **Sw Id**, Web Services Supported, Users, Size |
| User | **User Id**, Name, Social Network Data, Preferences. |
| Web Service | **Web Service Id,** Authentication Data, Feed Data, Preferences. |

Screenshots of table structures:

Table: user



Table: contact

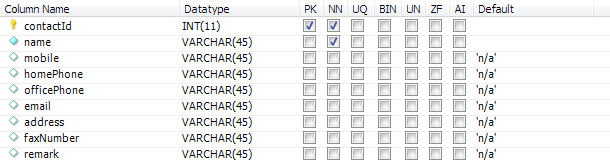


Table: Note



Table: password

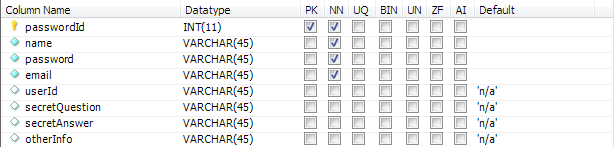
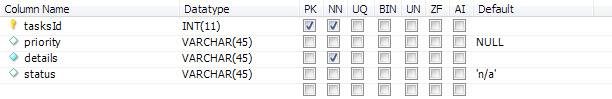


Table: tasks

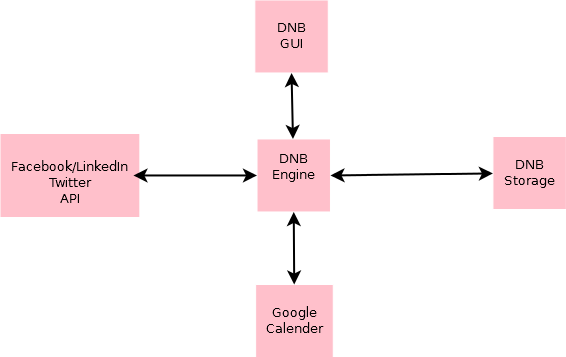


# Complete Structure

## Module Description

The above picture represents various modules of Daily Notebook software. Their detailed description is written below:

DNB Engine: This module handles all the logical parts of DNB. It takes data from user through DNB GUI module and stores them to database using DNB Storage module. It sends the data to the user’s social networking account using corresponding site’s API. The data stored is taken from the DNB Storage and sent to the API. It saves the events details in the Google calendar.



DNB GUI: This part is the place through which user interacts. This module contains all the designs which are visible and intractable by users. User provides input through it and gets the output through it. It is generally created by various tools like buttons and listviews.

DNB Storage: In this module all the data are stored. DNB Engine stores data in this module and fetches them for output through this module.

Google Calendar: This place gets input from the DNB engine. All the relevant data sent by user to Google calendar is stored here which could be accessed by user globally.

Facebook/Twitter/LinkedIn API: These modules provide news feed to user using the DNB engine through DNB GUI. It further gets the input, i.e. status update to the user’s account.

## Estimation



## Data Structure

|  |
| --- |
| LinkedInDetails |
| public class LinkedInDetails  {  public string Appid { get; set; }  public string username { get; set; }  public string password { get; set; }  public List<string> updates;  public DateTime lastLoginTime;  } |

|  |
| --- |
| GoogleCalenderDetails |
| public class GoogleCalenderDetails  {  public string Appid { get; set; }  public string username { get; set; }  public string password { get; set; }  public List<string> updates;  public DateTime lastLoginTime;  } |

|  |
| --- |
| TwitterDetails |
| public class TwitterDetails  {  public string Appid { get; set; }  public string username { get; set; }  public string password { get; set; }  public List<string> updates;  public DateTime lastLoginTime;  } |

|  |
| --- |
| FacebookDetails |
| public class FacebookDetails  {  public string Appid { get; set; }  public string username { get; set; }  public string password { get; set; }  public List<string> updates;  public DateTime lastLoginTime;  } |

|  |
| --- |
| GoogleMapDetails |
| public class GoogleMapDetails  {  public string Appid { get; set; }  public string username { get; set; }  public string password { get; set; }  public List<string> updates;  public DateTime lastLoginTime;  } |

|  |
| --- |
| NoteDetails |
| public class NoteDetails  {  public string Appid { get; set; }  public string username { get; set; }  public string password { get; set; }  public List<string> updates;  public DateTime lastLoginTime;  public bool isFbSynced;  public bool isTwitterSynced;  public bool isLinkedInSynced;  public bool isGoogleCalenderSynced;  } |

|  |
| --- |
| FacebookController |
| public class FacebookController  {  public List<FacebookDetails> fbs;  } |

|  |
| --- |
| TwitterController |
| public class TwitterController  {  public List<TwitterDetails> twits;  } |

|  |
| --- |
| LinkedInController |
| public class LinkedInController  {  public List<LinkedInDetails> linkedin;  } |

|  |
| --- |
| GoogleMapController |
| public class GoogleMapController  {  public List<GoogleMapDetails> map;  } |

|  |
| --- |
| GoogleCalenderController |
| public class GoogleCalenderController  {  public List<GoogleCalenderDetails> calender;  } |

|  |
| --- |
| ReportType |
| public enum ReportType  {  Single,  Daily,  Weekly,  Monthly,  Quarterly,  Yearly  } |

|  |
| --- |
| ReportInfo |
| public class ReportInfo  {  public string id { get; set; }  public DateTime date { get; set; }  public ReportType type { get; set; }  public string description { get; set; }  } |

|  |
| --- |
| DaliyNoteBookController |
| public class DaliyNoteBookController  {  public List<NoteDetails> notes;  public GoogleCalenderController GoogleCalenderController;  public GoogleMapController GoogleMapController;  public LinkedInController LinkedInController;  public FacebookController FacebookController;  public TwitterController TwitterController;  public MemberManager MemberManager;  } |

|  |
| --- |
| MemberInfo |
| public class MemberInfo  {  public string id { get; set; }  public string name { get; set; }  public string address { get; set; }  public string contact { get; set; }  public List<PreferenceInfo> prefs;  } |

|  |
| --- |
| MemberManager |
| public class MemberManager  {  public List<MemberInfo> members;  } |

|  |
| --- |
| PreferenceInfo |
| public class PreferenceInfo  {  string name;  string Description;  DateTime date;  } |

## Implementation Methodology

* Facebook, LinkedIn, Twitter sdk has been used to develop the application.
* Object Oriented Programming methodology will be adopted
* User interface development will be done in MVC architecture using Windows Presentation Framework.
* Relational DBMS MySQL will be used to implement & execute SQL query to database.
* Agile Software Development model will be used while developing this software.

## List of Reports

* List of Facebook updates could be generated.
* List of twitter update could be generated.
* A list of events could be generated.
* List of LinkedIn update could be generated.
* List of Google plus update could be generated.

# Implementation of Security Mechanism at Various Levels

The software requires a predefined username and password to login.

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.

# 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>
* [www.mysql.org](http://www.mysql.org)

----------------------------- Thank You-------------------------