

# Final Report

---

*Project Enlightenment*

*Computer Training For Visually Impaired Automation Tool  
(CTVIAT)*

**Prepared By:** Project Enlightenment

**Date:** January 14, 2011

## Table of Contents

1.	INTRODUCTION .....	3
1.1	PURPOSE OF THE SYSTEM .....	3
1.2	SCOPE OF THE SYSTEM .....	3
1.3	ASSUMPTIONS AND CONSTRAINTS .....	4
1.4	GOALS AND TRADE-OFFS.....	4
	User-Friendliness.....	5
	Ease of Use .....	5
	Reliability .....	5
	High Performance.....	5
	Minimum Number of Errors .....	5
	Security.....	5
	Completeness of Functionality .....	5
2.	MAJOR TOPICS.....	6
2.1	DEVELOPMENT PROCESS.....	6
2.2	REQUIREMENTS: PROBLEM STATEMENT .....	7
2.3	REQUIREMENTS SPECIFICATION .....	8
	Functional Requirements Specification.....	8
	Nonfunctional Requirements Specification.....	9
2.4	ARCHITECTURAL DESIGN .....	9
	System Decomposition.....	9
	Layers and Partitions .....	10
	System Topology .....	10
2.5	DESIGN RATIONALE .....	10
2.6	DATABASE DESIGN .....	11
2.7	IMPLEMENTATION .....	13
2.7	OBJECT DESIGN.....	13
3.	OUTCOMES and LESSONS LEARNED.....	15
4.	RESOURCES.....	15

## **1. INTRODUCTION**

### **1.1 PURPOSE OF THE SYSTEM**

This paper describes the details of Computer Training for Visually Impaired (CTVI) Automation Tool. The CTVI enables visually impaired students to take some teaching courses followed some training exercises on the computers to test their knowledge at Enable India. Since process of correcting the exercises is currently done manually by trainers, it is an exhaustive and time consuming task. The Automation Tool we present here automates this process. Its main purpose is to make the training program more productive, efficient and accurate.

By automating the training process, the tool also aims to make evaluation process take no more than a minute. Also, since the automation is used, the evaluations are supposed to be 100% true. Moreover, tests are not needed to evaluate by instructors but by the automation, this automation decreases the efforts of instructors. From a student's perspective, since each student will log in to the system, choose a test and with no help rather than automation tool finish the test by herself and at the end of test will see her score, this automation also speeds up feedbacks to students and that helps students improve. Besides, at any time, the student has a chance to repeat a test. Also, no manual intervention is needed to correct the exercises. All the controls, checks and hints will be provided by the Automation Tool. Furthermore, the tool presents a 100% secure platform, because each student can log in to the system with their own accounts. This clearly prevents a third person to reach student's private information.

From an admin or a trainer's perspective, the tool purposes to reduce the burden of controlling, correcting and pursuing the testing process. Also, the tool allows creating tests in different modules of instructions (such as editing, word, excel) and in different types (e.g. objective, descriptive or action based), adding questions to question bank, designing specific question papers to perform examination of students. Besides, it monitors actions of students during the examination by both giving question specific assistance to help finish their tests and correcting the exercises when the student fails to handle a question. This alleviates the need for manual intervention of a trainer. Furthermore, the tool evaluates all examinations and creates a student specific report for the trainers.

The CTVI Automation tool also aims to have a friendly interface for the students to use tool efficiently.

### **1.2 SCOPE OF THE SYSTEM**

The CTVI Automation Tool is implemented for two kinds of users: vision-impaired students and CTVI trainers (or admins). Those students who are capable of using a computer with the help of Braille-

based tools are the primary users of the automation tool. These users require no other additional computer experience.

The CTVI administrators are the main beneficiaries of the Automation Tool. The tool performs all the tasks the trainers perform manually today. What admins are required to do is just to login to the system and use the tool and benefit from the facilities of the tool. The tool will be designed such that all the user related properties will be accessible by any user and (s)he will require minimum knowledge when using the tool. Through this, users can receive more personalized services and obtain relevant information with relatively less effort and time.

### **1.3 ASSUMPTIONS AND CONSTRAINTS**

Constraints include the following:

- The deadline must be met because Enable India is planning to upgrade to the automation according to the deadline.
- The budget constraint must be met because the budget is a must to build each part of the project. If the cost was greater than the budget, the project would be closed.
- The product must be reliable and secure because private data may be revealed and the tool may affect the further study of the candidate.
- The architecture must be open so that additional functionality may be added later because the technology is evolving rapidly which necessitates new programs to be learnt and these programs may be added to the curriculum.
- The product must be user-friendly because the main users of the tool are visually impaired.

Assumptions include the following:

- The project is designed for and will be tested on general purpose keyboard. However, since the main user group is visually impaired, Braille-based keyboard will be required. Adaptation from general purpose keyboard to Braille-based keyboard should be easy.
- Although communication of the tool with the database is guaranteed by the tool itself, physical and digital security of the database should be guaranteed by Enable India.

### **1.4 GOALS AND TRADE-OFFS**

CTVIAT module will interact with two possible user groups: the candidates and the admins. Since the candidate users are assumed to have no previous experience in using the system and to be visually impaired, the system should provide some features.

## **User-Friendliness**

The system should be user-friendly to attract the candidates with proper help mechanisms to do more practice such as audio outputs and shortcuts which can be used instead of the mouse clicks. Since the candidates are impaired, CTVIAT should be an awesome guide, especially in action-based questions because this question type requires the candidates to do some advanced operations which can get them to wrong places. Moreover, the number of errors the customer may face cause of the tool itself, should be minimized as much as possible so that the candidates do not have the concerns about the system and focus their attention to the practicing.

## **Ease of Use**

This feature would make the users more enthusiastic while using the tool. There are some tasks, such as the registration of the candidates or preparing of the test, which may be boring for the users. With well-designed interfaces and easy-to-use components, these tedious processes may be converted into a joyful activity. Moreover, some exciting features will be implemented to entertain the candidates such as the ability of changing the voice of audio helper.

## **Reliability**

One of the most important features of the system is reliability. There are two aspects of the reliability. First, the information gathered from the customers (private information) must not be revealed under any circumstances. Moreover, the database in which this valuable information is kept should be secure enough.

## **High Performance**

Since the system talks the database server frequently and keeps the tracks of what the user is doing right now, the tool must respond to the user in a short time. Keeping tracks problem can be solved by intelligent algorithms and the computer that has a primary memory that is large enough. Database connection limitation will be overcome by a proper database design which is in BCNF, execution of queries in batches and prepared batch queries which makes query optimization one times which is in preparation.

## **Minimum Number of Errors**

Like any system, the CTVIAT module should be error-free. The problems that the user may face (in solving for candidate and in preparing for admin) and the corner points and the exceptional cases defined in the RAD should be properly handled. Every necessary test and its correction in code must be done.

## **Security**

Security is an important issue in this system. The security of the information kept in the database is vital. No matter how costly, the database must be protected because it is critical for the system to work properly.

## **Completeness of Functionality**

The completeness of functionality is also important. Every function is necessary of make the CTVIAT module efficient, practical and useful. They should be carefully designed and implemented.

## 2. MAJOR TOPICS

### 2.1 DEVELOPMENT PROCESS

Firstly, software development life cycle is determined. Extreme programming is selected as methodology; general principles are explained below table and causes as the following: CTVIAT is not hardware originated project and even purely software originated except Braille tools. This eliminates waterfall methodologies. CTVIAT is not large and expensive project, so not spiral life cycle. Cowboy coding is not formal enough for this software competition. Agile and XP are very similar indeed. However, XP is chosen because:

- XP is more relax,
- Group has two developers who can easily program in pair,
- Group has one designer and one reviewer(documenter) who can review progress,
- 4)Group has one ambitious tester who can do unit testing of all code,
- Group has one responsive advisor whom group can easily communicate for progress, code, design and changing requirements,
- Program domain is not familiar to staff (Braille for developers), 7)We will code for Word automation first and delay Windows Explorer, Excel, etc. In the beginning, we will put empty functions for these functionalities, after Word automation is completely functional, we will go on them.

Comparison of the software development life cycles are given in the table below:

Waterfall	Spiral	Agile	XP	Cowboy
-Big Design Up Front -Sequential manner -Used for hardware originated and stable problems -Changes costly -Bug fix in requirements -Easily markable milestones	-Relaxed waterfall -Initial phase BDUF -Second phase prototype, review and improve on incremental(design-code in small motivations) -Usually used in large, expensive and complicated projects -Chosen in game development -Agile is chosen in smaller projects	-Relaxed spiral -No long-term planning and minimal planning in short -Lack of planning, requires small increments -Each increment involves full sdlc -Changes are welcome -Customer satisfaction by rapid development	-Some is good, extreme is better(best) -Introduce new principles on top of agile -Programming in pairs -Extensive code review -Unit testing of all code - Implement when needed -Simple and clear code -Expect changes	-No external management -Lack of formal methodologies -Decrease burden of bureaucracy -Free working -Student level(inexperienced developers and experimental projects) -Chosen as hobby by talented developers -Quick and dirty, code and fix implementation

		-Continuous delivery(Live code) -Live code is measure of progress -Coop business and dev -Face-to-face communication -Self organizing Team	in requirements(via customer and developer) -Frenzy communication	(unreadable source and conflicts in semantics)
--	--	--	--	--

The two team members will work separately on their assigned artifacts and three developers will work together. Eray Saltik's assigned role will be to monitor the daily progress of the group, oversee implementation, be responsible for overall quality, interact with the client and help implementation. Team members will meet after three days and discuss problems and progress. Formal meetings with client will be held at the end of each week to report progress and determine if any changes need to be made and meetings with instructor will be held at scheduled timeline by the syllabus. Eray Saltik will ensure that schedule and budget requirements are met. Risk management will also be Eray Saltik's responsibility. Eray Saltik also has overall responsibility for all documentation and has to ensure that it is up to date. Actually, Eray Saltik is the manager. Design is Ozge Inan's responsibility and testing is Emmar Kardeslik's and implementation is controlled by Osman Sokuoglu and Ferhat Elmas. Maximizing user-friendliness and functionalities is Ozge Inan's and minimizing faults Emmar Kardeslik's top priorities.

## 2.2 REQUIREMENTS: PROBLEM STATEMENT

Here is a list of problem statements of Computer Training for Visually Impaired Automation Tool:

1. The exercise topics should have the ability to be separated into modules of instructions ( e.g., editing, dialog, windows explorer, jaws help, word, excel etc )
2. The tool should also support the ability to add mp3 files as instructions/answer options.
3. The tool should have the ability to create specific question types ( Objective / Descriptive and Action-based )
4. The time taken by the candidate to complete the exercises should be captured by the tool.
5. The marks scored should be captured in a format that can be sent / evaluated at a later stage. The program should interpret the responses and the results of the tests.
6. In case the user desires to complete the exercise in installments, the tool should support the option for the user to continue from the point where he left.
7. In case the user desires, he/she should be able to repeat an exercise. The tool should record and report all exercises that have been repeated.

8. Students should be able to re-read the instructions/questions word by word or line by line or letter by letter. If the user cannot understand the word/ sentence, he/she can use their arrow keys.
9. The candidate should be able to choose multiple choice answers by the use of Numeric keys (instead of having to deal with radio buttons or combo boxes). The intended user may not have the required expertise to work with these advanced windows controls.
10. The instructions/questions and the answer choices should be editable by the ADMIN at any stage of the exercise/question bank preparation.
11. The user (candidate) should have the ability to choose the desired topic/module that he wants to be quizzed on (e.g., editing, dialog, windows explorer, jaws help, word, excel etc.)
12. The admin should be able to generate individual reports based on each individual user's performance.
13. The instructions/questions should be stored in Unicode format such that it is compatible and supports other languages.
14. The candidates should have individual logins to the application so that they can review their results, check their answers vis-à-vis that of the correct ones.

## 2.3 REQUIREMENTS SPECIFICATION

### Functional Requirements Specification

The functional requirements of the CTVIAT can be grouped into three main parts: opening menu, administrator menu and candidate menu.

Opening menu of the CTVIAT will be a main window of the program. When a user opens the program, through opening menu users will see two options which are candidate menu and administrator menu. In addition, she can choose a language from available languages. Then, another window will open according to the user type and the language.

Administrator menu of the CTVIAT will be available for only administrators. To log in as an admin, one should have admin account of the CTVIAT. Administrator can log in by her username and password to the CTVIAT, in order to create questions for the question-bank, select the questions for to create and edit the questions that were previously set.

Candidate menu of the CTVIAT will be available for one who wants to take tests. The user logs in to CTVIAT with her individual account.

When an administrator logs in, at admin menu, there will be two options: either she will access the question form or test creates form. At question form part, administrator will be able to create new questions, edit questions and delete questions. On the other hand, at the test create form part, admin will be able to create new tests, edit tests and delete tests.



## Nonfunctional Requirements Specification

There will basically be two groups of users who will interact with the tool. The first group involves administrator and the second one consists of candidates. The admin has the privilege to create the questions for the question-bank. Also, she is supposed to select the questions to create and edit the questions that were previously set.

On the other hand, the candidate may be considered as the user who is administered with the test. The candidate logs into the program with her individual logins. She has the privilege to take tests which are created by admin. She is allowed to choose test about any subject. She is also informed about their test results, retakes tests.

Although the user interface will be designed as simple as possible, User Manual and Install Manual will be provided.

The system database is installed to each user since the system has not server-client architecture. The tool is designed to operate in standard computers which Windows (XP, Vista or Windows 7) are used and have enough memory to hold databases, mp3 files, or any data that can be used by the tool.

Any unforeseen errors detected on the test phase are adjusted separately by the developers of the program in order to ensure system integrity. In CTVI Automation Tool, error handling and extreme conditions should be handled by the developers of the program in order to ensure system integrity.

The program is supposed to be upgraded to add new features to the tool if needed. According to the feedback from the users, new versions of the tool should be launched.

## 2.4 ARCHITECTURAL DESIGN

### System Decomposition

CTVIAT project is composed of following subsystems: LOGIN, USER, AUTOMATION and DAO.

The principal subsystem of CTVIAT is the Login since services are served according to the result of this subsystem. Login subsystem is the entrance of the tool so it is the bodyguard. It directly talks to DAO subsystem to authenticate users.

Another subsystem within the CTVIAT module is the User. Purpose of the subsystem is to serve functionality of the tool to the user with a user-friendly interface. This subsystem is composed of two subsystems, namely Candidate and Admin, since there are two types of user. If Login system authenticates a user, one of the user systems will be loaded.

The subsystem is used by User systems is Automation system. Automation system is the core of the tool and its functionality can change for the user. For admin, system can be used to add new users, to prepare new questions and new test, to query reports for candidates' progress. For candidate, system can be used to load completely new test or old test, to ask for help in questions and to guide the user what she should practice. This subsystem uses Test and Report subsystems in itself to do these varying

tasks. For instance, while the candidate is solving the test, the subsystem shows the questions and saves the answers and also collects the other information such as time for reporting.

DAO means database access object, so DAO system includes database and adapter classes to access it. Each of these classes is mapped into database tables. This level is required because DAO will be an abstraction of the database and provide query optimization and batch mode execution of queries.

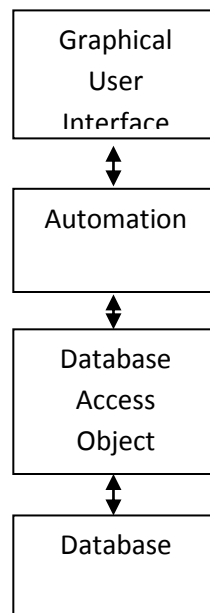
All subsystems require use of the database so does DAO. Main functionality is served by automation but it does not mean anything without user-friendly service.

### Layers and Partitions

The CTVIAT module offers a layered architecture with three different layers. The subsystems which include graphical user interfaces, namely Login and User systems, are in the first level. Automation uses three subsystems to serve main functionality so it is the heart of the project and it stays at the mid-level, -second level. Third level is filled by Dao and it is an encapsulation of the database.

### System Topology

The CTVIAT Module system can be represented by the following topology:



## 2.5 DESIGN RATIONALE

There are mainly two important design issues to discuss about the CTVIAT module and subsystems of the module.

The first one is the visual part of the CTVIAT consisting of windows by which the users, candidates and admins, will interact with the system. As the CTVIAT is a desktop application and excessively talks to a data server which runs at Lan, it is decided with the accord of the project team that

the proposed WPF programming methodology is the best solution among alternatives such as WinForms.

Since platform is Windows, integration with Windows is the most important criteria that brings two options for desktop applications, namely WPF and WinForms. However, Microsoft advertised that there will be no support for WinForms in upcoming version of Windows. Moreover, WPF is the Microsoft's gun to compete with Flash so Microsoft provides good documentation. Finally, implementation in WPF is simple and project members are familiar with this framework.

The second part of the design needs a management tool for the database purposes. Since the CTVIAT module will work on Windows platform, for better integration data server should also work on Windows. Therefore, there are mainly two alternatives on the subject. The first one the usage of MS Access and the other one the implementation on SQL Server. The arguments for MS Access were the availability and the familiarity of the software, but to ensure a secure multi-connection handling, the best solution was the choice of the SQL server although its usage will require the installation of the software on the project members' PC's for development purposes. At this point, it will also be worthwhile to mention about some disagreements on database tables and fields. Some module teams have argued that a "username & password" entry in database would cause an extra burden for the designers and a dissatisfaction for the customers. This issue has also been resolved by a positive decision on the problem saying that those entries were crucial to build a security barrier since the system contains private data.

The main tool to achieve a proper software design and development is the Rational Rose case tool whose usage has been proposed by the Project Manager and been accepted by all members due to its visual representation capabilities with UML features.

The system seems to be quite responsive to technological improvements for the moment. WPF is used which supports hardware acceleration. Multi-threading is used to utilize of modern cpus of nowadays. Main performance issues are the efficient handling of database operations and required io operations by questions. Database operations are handled via proper design and indexing, while io operations via multithreading.

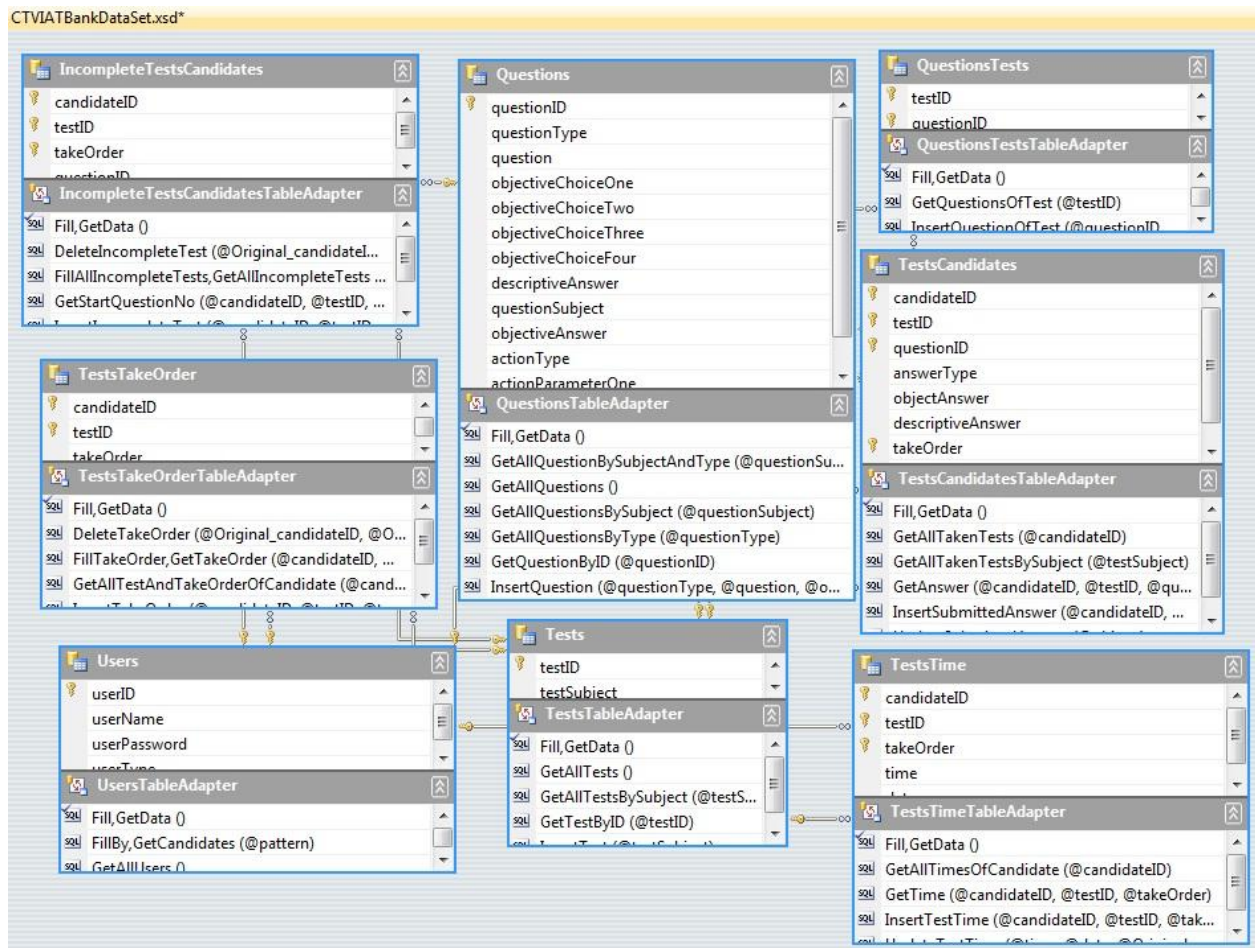
## 2.6 DATABASE DESIGN

CTVIAT project is a desktop application that will be installed on client computers which are connected to a database where all data about the application will be stored. The database is designed in a secure, efficient, reliable, scalable and compact.

In this project, we used MS-SQL Server 2008 in order to implement tables, stored procedures primary keys and foreign keys. Microsoft SQL Server has the following features such as transaction, selection, insertion, update, stored procedures, triggers, views, referential integrity and foreign keys. The integration with the other Microsoft solutions we will use such as Visual Studio IDE and Windows Presentation Foundation is one of the key factors why we choose Microsoft SQL Server. We preferred SQL Server 2008 since this new version provides a secure, easy and fast way to store and manage data. The main disadvantage of the SQL Server is that it is not an open source system.

In our design, we have used Requirements Analysis Document and we have contacted with the other group representatives in order to determine the requirements of the database. We updated

or changed our design according to the changing requirements. We have tried to update our design efficiently. We have tried to minimize the usage of join operations in order to improve performance. We preferred not to use stored procedures. We used Visual Studio 2010's database connection and dataset tools which are really fast for implementation and create a reliable database environment. We have also wanted to small tables instead of tables with many columns. Maintenance of the system has been done by monthly patch controls and system upgrades when needed. Microsoft SQL Server sites will be followed for possible patches and/or advisories on Microsoft SQL Server.



The database consists of eight tables which have a generic and relationships with the classes of the project. *Users Table* represents users of the program according to their mode, *admin* or *candidate* through their *userID*. *TestCandidates Table* represents every question of every test that a unique user solved and it stores the answer of the user, too. *Tests Table* represents tests that are created by admins and consists of questions that are created by admins. *Questions Table* represents questions created by admins. *QuestionsTests Table* represents which questions are used in a test by an admin. *IncompleteTestsCandidates Table* represents incompletest tests of a candidate. *TestsTime Table* represents the total time elapsed for a test of a candidate. *TestsTakeOrder* represents the take order status for a test of a candidate.

## 2.7 IMPLEMENTATION

The system contains private data so login is required for everybody. After a successful login, according to user type different flows start. There are two type users, namely candidate who is visually impaired and admin who prepares the question and tests and also can query the history of the candidates for a report. Firstly, there is a complete audio support for candidate all over the execution. Candidate can get audio help whenever she presses the Ctrl + H (elp). Therefore, if logged on user is a candidate, user manual is started to be read. Then, candidate continues to next windows to select the test properties. After she selects the properties, tests are filtered according to these properties. Candidate chooses one of them and starts to solve. System reads questions and gives necessary instructions. Candidate can repeat the question by Ctrl + R (repeat). We utilize of a lot of keys since candidates are visually impaired. Complete list of the keys will be provided in user manual. Candidate can travel in questions and at the end she can finish the test or discontinue the test to continue later. If she finishes the test, she can see the answers or she can skip the answers, if she skips, she can practice a new test or quit. If she discontinues the test, she can't see the answers. She can see the answers only if she finishes the test. When she opened this test, test will be opened from where she was and at the time how much time she used. Therefore, time is accumulated but answers present the final version. For example, if she had answered question one correctly and she changes the answer to wrong one and wrong answer will be saved. Moreover, candidate can solve the tests which she has solved before to check her understanding. All operations of the candidate will be followed and will be reported in the report for admin.

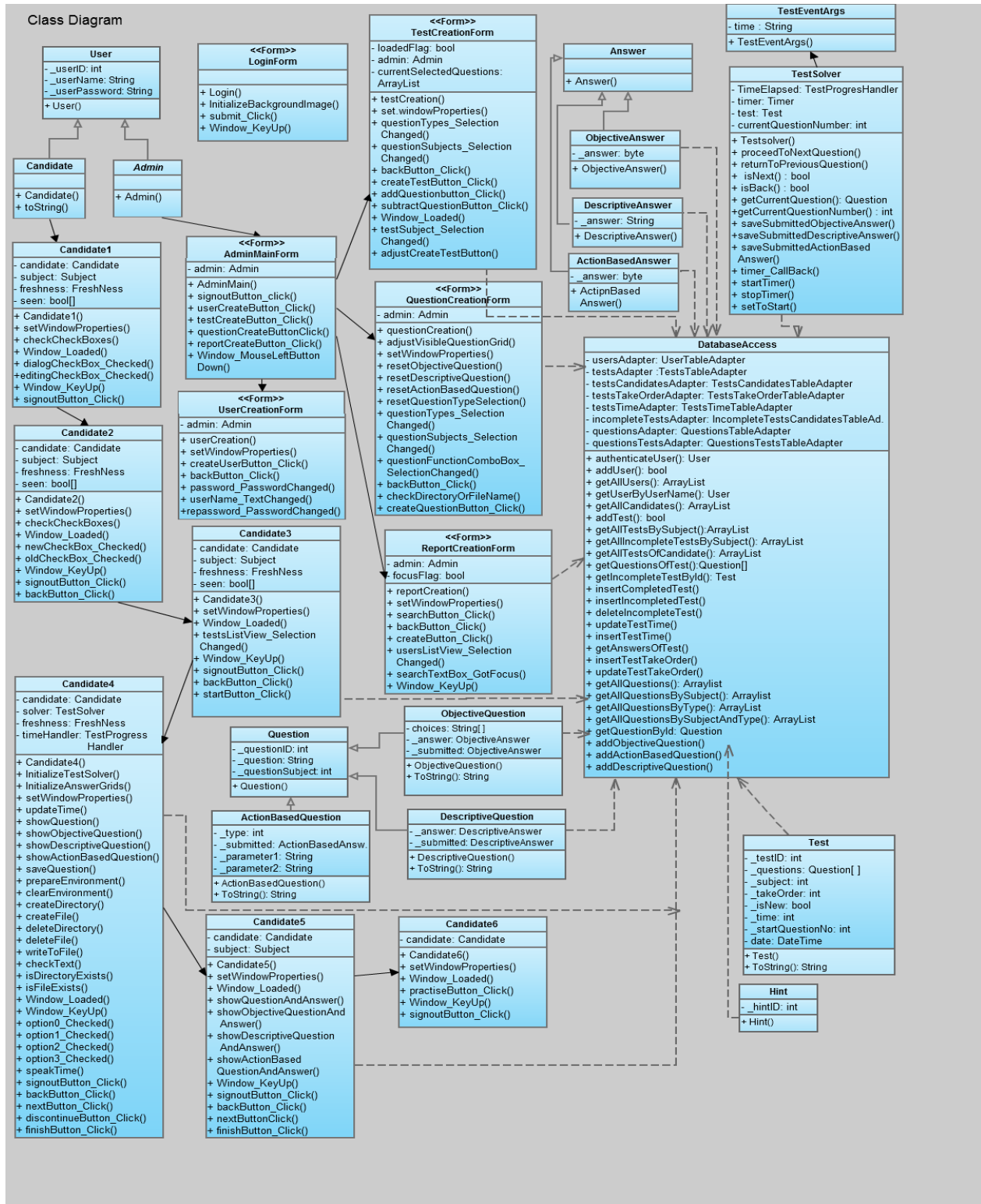
If logged on user is an admin, admin main window loads and admin can do four tasks, namely, create user, create question, create test and create report. In user creation, admin provides a username and password where username isn't used by anyone. In question creation, admin chooses the question type and system helps the admin for necessary parts according to the type. In test creation, admin filters the questions according to the question type and subject and then chooses the questions that he wants to add to the test. Moreover, admin must provide test subject to create the test. In report part, admin searches the candidates via some keywords and chooses a directory where report is saved to and then report is created.

## 2.7 OBJECT DESIGN

In the design of CTVIAT module, there is a trade-off between performance and complexity. Since, CTVI Automation Tool will provide many functionalities to users, there should be a short respond time when carrying out those functionalities to satisfy user expectations. Because, during the usage of the application, visually impaired people will be alone and must not wait too much. In another word, the performance of the application is the main issue in the design.

In the project, mainly, there are three packages are defined: Interface, Classes and Database. The Classes package containing the main classes used in the implementation of the CTVIAT application and in general each class consists of many subclasses. In Interface package, there are classes for login screen, admin screen and candidate screen and all classes responsible to handle interface operations. In

Database package, there is a database access object class which is responsible to manage database connection and enable other classes to reach database through this class. This class includes User, Test, Question, Answer and Candidate-Test Methods to handle Method related database operations. The class diagram of the project is as follows:





The class diagram of the CTVIAT represents the classes, their attributes and methods, and the interactions between.

### **3. OUTCOMES and LESSONS LEARNED**

Firstly, this project is our first group project so we couldn't plan the time and schedule the tasks very well. As a result of this situation, some members worked harder to compensate the other members because our group is composed by the teacher for software engineering course and group members are very different from each other and so expectations are. By their extra work, we could complete the project on time but we learned how cooperation can be difficult to accomplish and how it is important for excellent product.

Secondly, this project is our first project that continues for two periods. We tried to come together weekly and review what is completed. Moreover, we prepared progress presentations bi weekly since this project is an important part of our course. We tried to proceed according to software development methodologies. At the start, we learned about each development methodology and write down their advantages and disadvantages. According to our capabilities, contest requirements and course suggestions we compared them and chose the most appropriate one and tried to conform. Extreme programming is really applied at extremes. To sum up, we have experienced long-running project.

Third, we have designed a program for visually impaired people. This was challenging since all of us are unaware of the difficulties that they have experienced. We were inexperienced about designing user interferences and we were asked to design an application for visually impaired people. We can say that we joined the application design world from the most difficult point. However we succeeded our project and we created friendly user interference although we do not have any hardware which is special to visual impaired people. Besides we enriched our design with Text – Speech features which is not a requirement for the project.

In conclusion, we want to emphasize that we are really happy to be a part of SCORE contest, to make an extraordinary tool for the visually impaired people and to build a hard-working and professional team environment for ourselves. We hope that our project will help Enable India company with admirable work of making disabled people's lives easier.

### **4. RESOURCES**

We have already created all software design documents such as database design document, software design document and an extensive project plan for a two month implementation process. If you request these documents, graphs or any schemas, we can send them to you.