**21st century software solutions here mentioned, Detecting Anomalous Insiders in Collaborative Information Systems, project in detail as an abstract, which includes what is actually involved in the entire mini project to be done by A.S.S.BHARADWAZA and his team members under guidance of Raghava Rao (Associate Developer) in the company premises**

**We are taking responsibility to train them in all basic requirements in general use to develop the project**

**The project proposal is as follows**

Initially we will record (EHR) system in a large medical center. The results illustrate our models exhibit significant performance gains over state-of-the-art competitors. When the number of Collaborative information systems (CISs) are deployed within a diverse array of environments that manage sensitive information. Current security mechanisms detect insider threats, but they are ill-suited to monitor systems in which users function in dynamic teams. In this paper, we introduce the community anomaly detection system (CADS), an unsupervised learning framework to detect insider threats based on the access logs of collaborative environments. The framework is based on the observation that typical CIS users tend to form community structures based on the subjects accessed (e.g., patients’ records viewed by healthcare providers).

CADS consists of two components: 1) relational pattern extraction, which derives community structures and 2) anomaly prediction, which leverages a statistical model to determine when users have sufficiently deviated from communities? We further extend CADS into MetaCADS to account for the semantics of subjects (e.g., patients’ diagnoses). To empirically evaluate the framework, we perform an assessment with three months of access logs from a real electronic health illicit users is low, MetaCADS is the best model, but as the number grows, commonly accessed semantics lead to hiding in a crowd, such that CADS is more prudent.

**Existing System:**

It can be seen that the performance of the supervised classification models is significantly worse than the unsupervised models. The supervised models consistently have a lower true positive rate at all operating points. Second, unlike the previous experiment, HVU achieves comparable results to the supervised classification models. This is due to the fact that this model is correctly characterizing the intruders that access a larger number of records. Third, with respect to AUC, we observe the same trend as earlier regarding the dominance of the unsupervised models as a function of the mix rate. Specifically,

**How our proposal overcomes the limitations of existing**

MetaCADS dominates when the mix rate is low, but CADS dominates when the mix rate is high. Notably the disparity between MetaCADS and CADS is more pronounced at the low mix rate (0.91 versus 0.88) in this setting than in the previous setting. However, at lower false positive operating points, CADS appears to dominate MetaCADS.

**Proposed system**

Several notable approaches have been proposed to address this type of intruder. The first is nearest neighbor anomaly detection techniques, which are designed to measure the distances between instances by assessing their relationship to “close”

Instances. If the instance is not sufficiently close, then it may be classified as an anomaly. However, social structures in a CIS are not explicitly defined and need to be inferred from the utilization of system resources. If distance measurement procedures are not tuned to the way in which social structures have been constructed, the distances will not represent the structures well. Our experimental results confirm this notion.

**Modules:**

* Pattern Extraction
* Anomaly Detection
* Detection Performance Metrics
* Varying Number of Accessed Subjects

**Software Requirements:**

* OS : Windows XP with SP2
* Database : MS-SQL server 2008
* Language : C#.NET
* IDE : Visual Studio .Net 2010
* Browser : IE

**Hardware requirements**

* Processor :Pentium
* HDD :20 GB Min 40 GB Recommended
* RAM :1 GB Min 2 GB Recommended

**21st century software solutions here mentioned ,an Oracle database maintenance project in detail as an abstract, which includes what is actually involved in the entire mini project to be done by A.S.S.BHARADWAZA and his team members(T.NAVEEN KUMAR & P.SAI ARUN KUMAR) under guidance of Raghava Rao (oracle developer) in the company premises**

**We are taking responsibility to train them in all basic requirements in general use to develop the project**

**The project proposal is as follows**

Each student must complete an independent Project in which a relational database is described and diagramed according to the concepts learned during project. The student will then create tables and a variety of queries using Oracle, and document all work. Database must be done independently, must be efficient in structure, and must be delivered on time.

This is an official assessment project for the intern candidates with in **21st Century Software Solutions**; the core requirements listed below cannot be changed. The instructor may add additional items. The due dates of each phase and the grading system are determined by the instructor.

**Phase I: Proposal and Design**

Each student must identify an application or select a topic for his or her database project by the appointed due date. The chosen application should be different from any of the three case studies provided by the textbook. The following are minimum items that must be included in Phase I. The instructor and student should confer with each other regarding the feasibility of the student’s plan so that corrections can be made before proceeding.

A typed, one-page report that describes the business or personal environment, the specific subject matter of the database, the purpose of each table in the database, the useful queries that can be produced from the database, and the advantages that the database has for the owner or operator. Focus on overall information requirements.

An E-R diagram that appropriately documents the design of the database. The diagram must contain the following items.

* An appropriate number of related entities/tables (four or more recommended) in third normal form (3NF)
* Typical attributes of each entity
* Properly defined primary keys and foreign keys
* Appropriate data types and lengths
* Relationships between tables – include both a one-to-many relationship and a many-to-many relationship

**Phase II: Basic Database Development and Construction**

Create the required tables and insert records into the database using Oracle. The following must be included:

* Create and print the script Oracle code for each table.
* Enter a minimum of ten rows/records into each individual table and print a listing of those rows/records. Remember that there should be variety in the content of the sample data in each table to insure that all required queries and tasks can be demonstrated. Good sample data is important in testing the integrity of the overall database design.
* Table design must exhibit functional dependency and must adhere to the rules of third normal form (3NF).
* Relationships between tables must be accurately defined and established.

**Phase III: Query Development and Results**

Each student will create a variety of queries, using both individual and multiple tables to demonstrate each of the following:

* Three queries should demonstrate the ability to use operators such as “And”, “Or”, and “Not”.
* Two queries should demonstrate the use of computed columns.
* Three queries should demonstrate the use of aggregate functions.
* Four of the queries should demonstrate the joining of two tables.
* Demonstrate the ability to sort data.
* Demonstrate the ability to change table structure
* Demonstrate the ability to delete, append, and update data.

**Phase IV: Professional Techniques for Documentation and Completion**

When finished, the overall project should exhibit the following features:

* Efficiency - database is useful and reusable.
* Independent Work. This is not a team project. Collaboration with the instructor and other students is allowed so long as the concepts from the textbook and class lectures are independently integrated into the project. Direct copying of materials from another person or resource such as the textbook would be considered a violation of the honor code.
* Each aspect of every project phase should be professionally documented including descriptions and explanations of the work that was completed.
* Incorporate the use of professional tools and techniques:
* Define and utilize views to improve user access
* Include indexes to improve retrieval efficiency
* Assign foreign keys to help maintain referential integrity
* Utilize both the EXPLAIN command and the PROCEDURE ANALYSE() FUNCTION to evaluate query performance
* Initiate integrity support features to establish and enforce integrity constraints
* Professional Organization and Presentation: all documentation should be collected and neatly organized into a professional looking folder or binder.
* Prompt Delivery: each phase of the project should be delivered on time.