ISSN 2277-2685
IJESR/April-June. 2025/ Vol-15/Issue-2s/942-949
D Sathwika *et. al.*, / International Journal of Engineering & Science Research

Student Abnormal Behavior Recognition

AVS Radhika, D Sathwika, K Sphoorti, K Srujana, A Yamini
[1] Assistant Professor, Department of CSE, Bhoj Reddy Engineering College for Women, India.
[2,3,4,5]B.Tech Students, Department of CSE, Bhoj Reddy Engineering College for Women, India.

ABSTRACT
An intelligent campus surveillance system enhances school safety by utilizing abnormal behavior recognition, a key aspect of action recognition in computer vision. While Convolutional Neural Networks (CNNs) are commonly used for action recognition, capturing comprehensive motion sequence features from videos remains challenging. This work addresses these challenges in video-based abnormal behavior recognition on campuses. It introduces a novel framework combining long-range temporal video modeling and a global sparse uniform sampling strategy, dividing videos into three equal segments for uniform snippet sampling. The method leverages a consensus of three temporal segment transformers (TST), which connect patches globally and compute self-attention using joint spatiotemporal factorization. The model is developed on the CABR50 dataset, featuring 50 abnormal action classes with over 700 clips per class.

INTRODUCTION

Ensuring campus safety is an essential responsibility of educational institutions worldwide. Campuses are dynamic environments where students engage in a variety of activities, making the detection and prevention of abnormal behaviors a complex yet critical challenge. Over the years, incidents such as fights, accidents, falling, and suicides have occurred with alarming frequency, creating widespread concern among educators, parents, and administrators. These incidents highlight the pressing need for innovative solutions that can proactively address safety challenges in real-time.

Traditional methods of monitoring student behavior rely heavily on manual observation and retrospective analysis of incident reports. However, these methods are often reactive and fail to provide timely interventions. To address these limitations, automated systems leveraging advanced technologies have emerged as a promising solution. Such systems are capable of efficiently and accurately detecting abnormal student behaviors, offering real-time warnings and enabling swift corrective actions. This proactive approach not only enhances safety management but also fosters a secure and conducive learning environment for students.

This project aims to design and implement a system that utilizes advanced datasets and cutting-edge machine learning algorithms to identify improper behaviors among students. By employing Linear Regression, Support Vector Machines (SVM), and Naive Bayes algorithms, the system can analyze and interpret complex datasets, including attributes such as student personal data, time spent on studies, and grades. These insights are used to predict and classify behaviors as proper or improper, ensuring a robust and reliable safety mechanism. The system's ability to provide high accuracy and real-time detection makes it an invaluable tool for campus safety management.

Existing System

Recent advances in historical data analysis have enabled the identification of abnormal behaviour in various domains, including campuses and public spaces. Existing systems rely on datasets containing incident reports, activity logs, grades, and disciplinary records to identify patterns and behaviors. Data collection primarily involves aggregating manual records and extracting features relevant to student behavior. Predictive models, such as Classification, Regression, and Time Series Forecasting, are used to detect patterns in these datasets. Visualization methods are based on historical campus incident data and manual analysis, providing limited real-time insights.

Proposed System

The proposed system employs anomaly detection techniques tailored for identifying abnormal behavior. By leveraging curated datasets containing both normal and abnormal instances, the system detects deviations from normal patterns indicative of anomalies. Extensive experimentation and comparative analysis validate its efficacy in accurately identifying and categorizing abnormal behaviors. Performance validation across datasets ensures robustness and generalization.

REQUIREMENTS ANALYSIS

Functional Requirements

Service Provider

The Service Provider module is designed for users with administrative access to manage and oversee the campus behavior recognition system. After logging in with valid credentials, the service provider can perform various operations, including:

1. **Train & Test Campus Data Sets:** Initiate the training and testing of the campus datasets to develop the model for behavior recognition.
2. **View Trained and Tested Campus Data Sets Accuracy in Bar Chart:** Visualize the accuracy of the trained and tested datasets using bar charts.
3. **View Campus Data Sets Trained and Tested Accuracy Results:** Review the detailed accuracy results of the trained and tested datasets

Non-Functional Requirements

Accuracy

The system should achieve a high degree of accuracy in detecting abnormal behaviors. This includes minimizing false positives, where normal behavior is incorrectly flagged as abnormal, and false negatives, where actual abnormal behavior goes undetected. To ensure reliability, the system must be thoroughly tested on diverse datasets, encompassing various scenarios and behaviors, to refine its predictive capabilities.

Scalability

The system must be designed to handle large volumes of data from multiple sources, such as activity logs, academic records, and sensor data, without compromising performance. It should be able to seamlessly scale as the size of the student population and the complexity of the data increase. This involves using efficient algorithms and robust infrastructure capable of processing and analyzing high-dimensional data in real-time.

3-DESIGN

Architectures

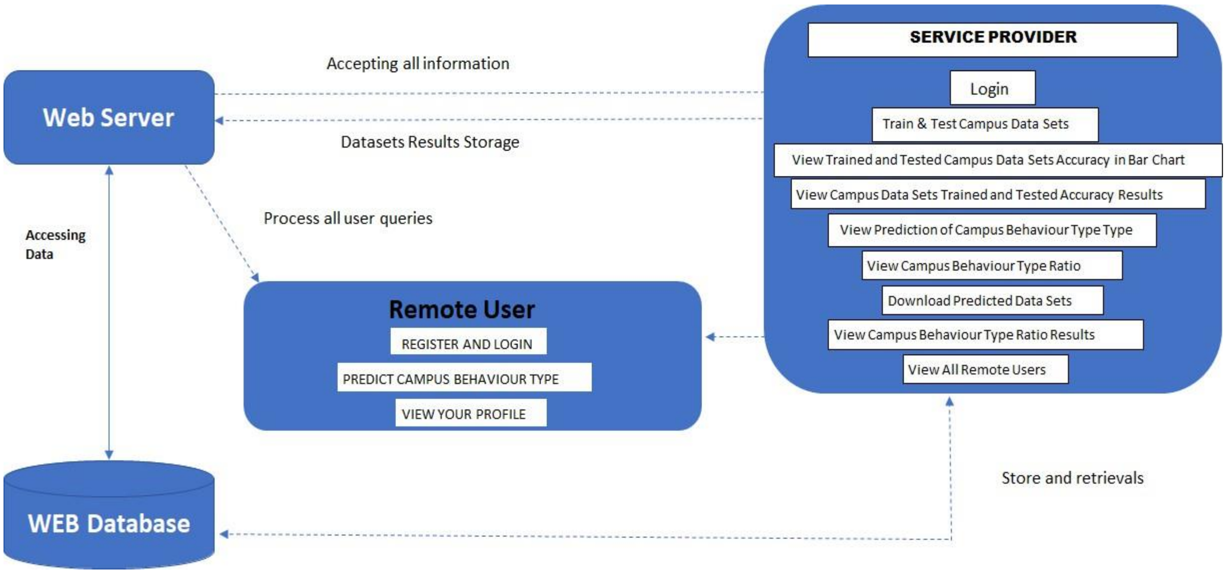
Project architecture represents number of components we are using as a part of our project and the flow of request processing i.e. what components in processing the request and in which order. An architecture description is a formal description and representation of a system organized in a way that supports reasoning about the structure of the system. Architecture is of two types.

They are

(1) Software Architecture

(2) Technical Architecture

Software Architecture

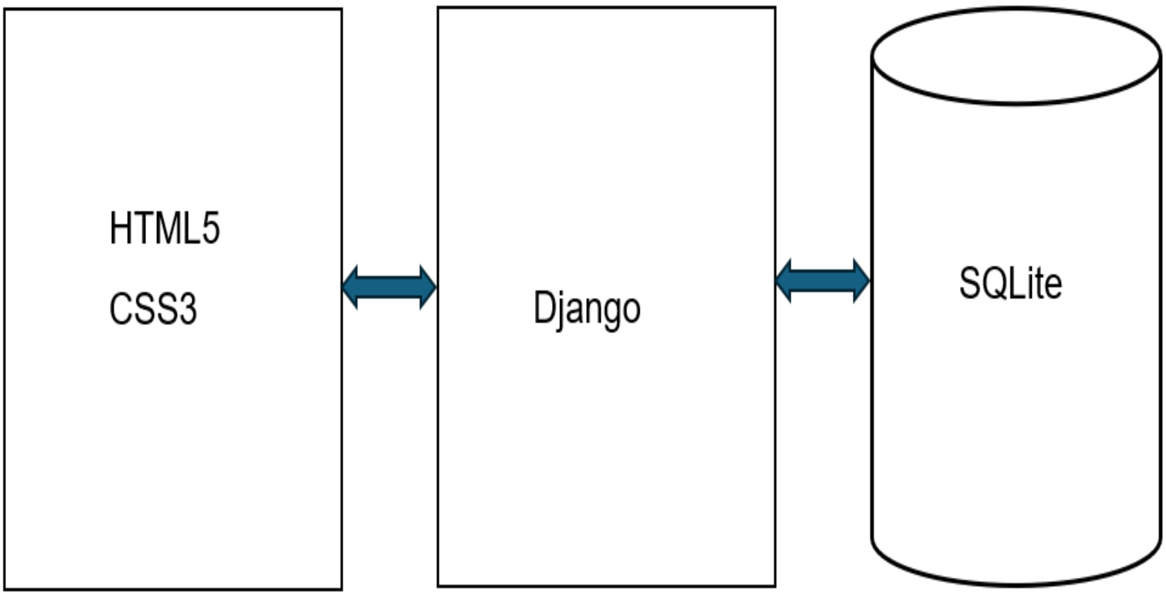


3.1.1 Software Architecture

Technical Architecture

The technical architecture represents a simple web application architecture using **HTML5** and **CSS3** for the frontend, **Django** as the backend framework, and **SQLite** as the database. The frontend handles user interactions and visual presentation, Django processes requests and implements business logic, and SQLite stores and retrieves application data. Communication flows between the frontend, Django, and SQLite to ensure seamless operation.

Technical Architecture



4- IMPLEMENTATION

Technologies

Here's a detailed breakdown of the technologies used in the architecture:

Frontend Development

The frontend is responsible for the visual interface of the application, ensuring users can interact with the platform easily.

- **HTML5:**
- Acts as the backbone for creating the structure and content of the web pages.
- Supports semantic tags (like <header>, <footer>, <article>) for better readability.
- Ensures cross-browser compatibility for smooth rendering across different web browsers.
- **CSS3:**
- Used to design the layout, colors, fonts, and animations, ensuring a visually appealing user interface.
- Enables responsive design with features like Flexbox, Grid, and media queries to adapt the interface across devices (mobile, tablet, desktop).
- Supports advanced styling like transitions, gradients, and shadows to enhance user experience.

Pseudo Code

Pseudocode is a detailed yet readable description of what a computer program or algorithm must do, expressed in a formally-styled natural language rather than in a programming language. It allows designers to express the design in great detail and provides programmers a detailed template for the next step of writing code in a specific programming language. Because pseudocode is detailed yet readable, it can be inspected by the team of designers and programmers as a way to ensure that actual programming is likely to match design specifications. Catching errors at the pseudocode stage is less costly than catching them later in the development process. Once the pseudocode is accepted, it is rewritten using the vocabulary and syntax of a programming language.

5. TESTING

Testing is a process, to evaluate the functionality of a software application with an intent to find whether the developed software met the specified requirements or not and to identify the defects to ensure that the product is defect free in order to produce the quality product.

As per the current trend, due to constant change and development in digitization, our lives are improving in all areas. The way we work is also changed. We access our bank online, we do shop online; we order food online and many more. We rely on software's and systems. What if these systems turnout to be defective? We all know that one small bug shows huge impact on business in terms of financial loss and goodwill. To deliver a quality product, we need to have Software Testing in the Software Development Process.

Some of the reasons why software testing becomes very significant and integral part in the field of information technology are as follows.

1. Cost effectiveness
2. Customer Satisfaction
3. Security
4. Product Quality

Dimensions of Testing

There are many different dimensions to consider:

Layers of the application (database, APIs, UI)

Scale of testing (unit, module, integration, scenario)

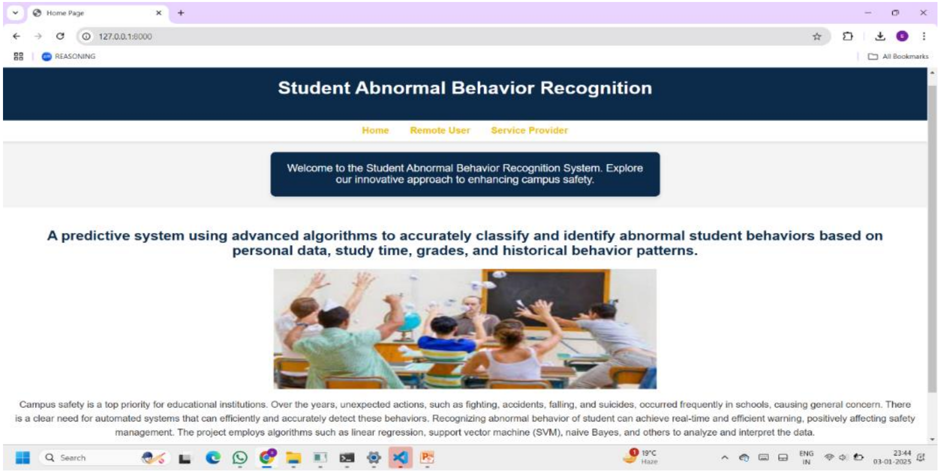
Type of testing (functional, performance, security, etc.)

Methodology (exploratory, scripted manual, automated)

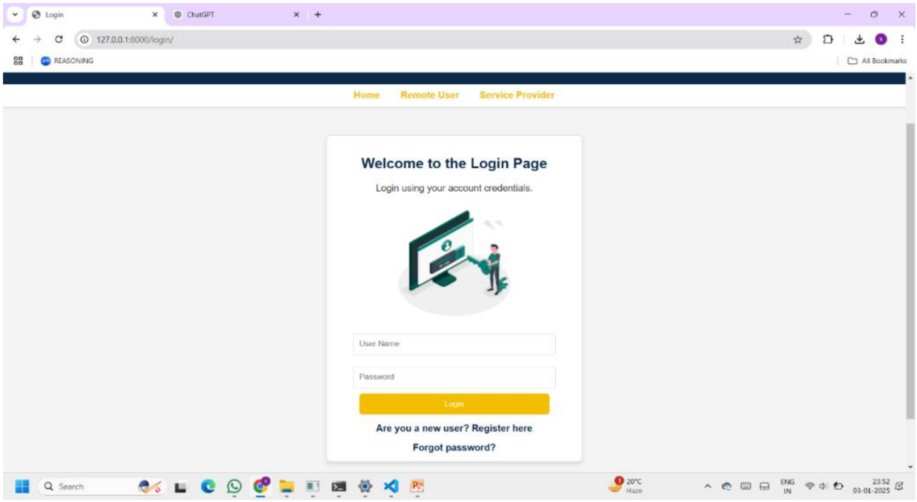
System Testing

System testing is the first level in which the complete application is tested as a whole. The goal at this level is to evaluate whether the system has complied with all of the outlined requirements and to see that it meets Quality Standards. System testing is undertaken by independent testers who haven't played a role in developing the program. This testing is performed in an environment that closely mirrors production. System Testing is very important because it verifies that the application meets the technical, functional, and business requirements that were set by the customer.

6-SCREENSHOTS

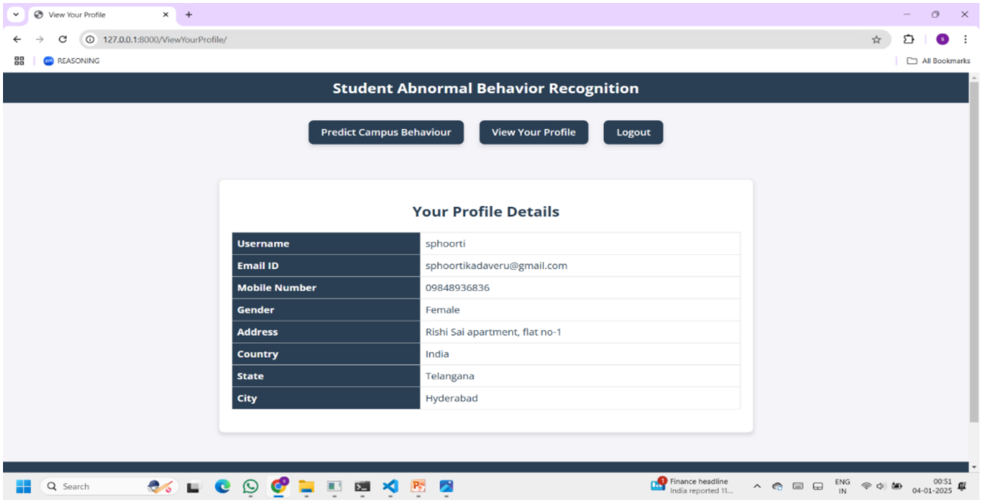


6.1 Home page

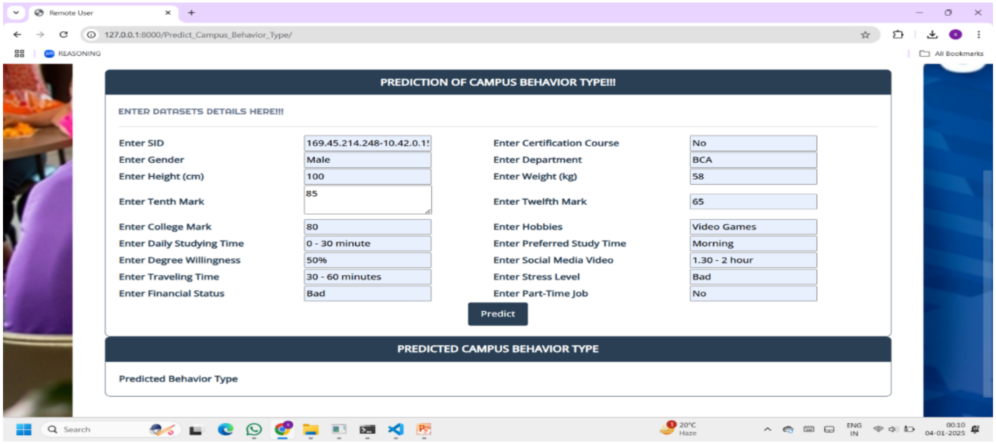


6.2 User login

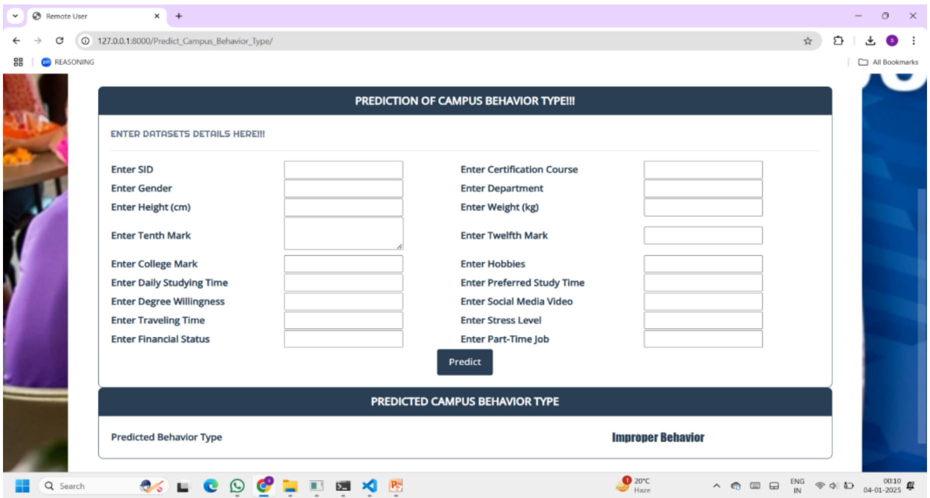
ISSN 2277-2685
IJESR/April-June. 2025/ Vol-15/Issue-2s/942-949
D Sathwika *et. al.*, / International Journal of Engineering & Science Research



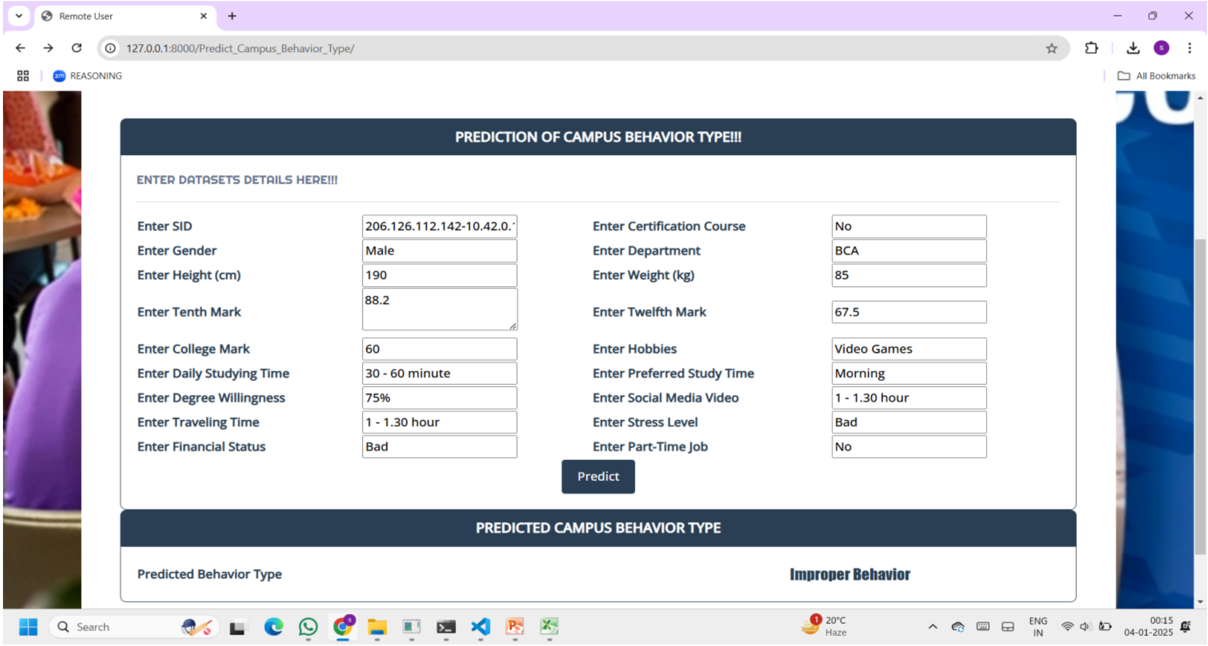
6.3 User Registration



6.4 User Profile Details



6.5 Prediction of Campus Behaviour Type



6.6 Result to prediction

7. CONCLUSION AND FUTURE SCOPE

Conclusion

In conclusion, this study proposes effective anomaly detection methods for intelligent campus surveillance, emphasizing the identification of abnormal student behaviors. By utilizing machine learning algorithms such as Linear Regression, SVM, and Naive Bayes, trained on datasets containing both normal and abnormal behavior instances, the system demonstrates its potential to accurately classify and predict behavioral anomalies. These techniques pave the way for real-time monitoring and intervention, ensuring a safer campus environment. However, achieving this requires addressing several critical challenges. Ensuring high-quality datasets with diverse and accurate labeling is fundamental to model reliability, while strict privacy measures are essential to safeguard student data and maintain ethical standards. Additionally, seamless integration with existing campus surveillance infrastructure is crucial for real-world implementation, requiring compatibility with hardware and scalability for dynamic environments. By tackling these challenges, this study establishes a foundation for developing robust and ethical anomaly detection systems that can enhance campus safety and support student well-being effectively.

Future Scope

The future scope of this study offers significant opportunities for advancements in intelligent surveillance systems. Key areas include:

- **Convolutional Neural Networks (CNNs):** CNNs can analyze video frames from surveillance cameras, enabling the detection of spatial patterns related to student behaviours.
- **Real-Time Detection:** Real-time anomaly detection systems enable immediate identification of abnormal behaviours, facilitating timely interventions. Such systems could utilize edge computing to process data locally, reducing latency and ensuring swift responses to potential incidents.
- **Temporal Segmentation:** Incorporating temporal segmentation techniques adds contextual depth to the

analysis by examining sequences of actions over time. This approach helps distinguish between isolated events and sustained patterns of behaviour, improving detection precision.

REFERENCES

[1]. Ali, O., Ali, S. and Koray, C., “GPS And GSM Application For Interactive Location And Speed Tracking”, IJCI roceedings of International Conference on Signal Processing, ISSN,
pp. 1304-2386, and Vol. 1, No. 2, September 2003.

[2]. Al-Jazzar, S., Caffery, J. and You, H.R., “A Scattering Model Based Apprach to NLOS Mitigation in TOA Location Systems”, IEEE, 2002.

[3]. Anamika, D., “Advanced public transportation system”, Location A Bi-Monthly Magazine on Satellite Positioning and Navigation, Vol. 02, Issue 02, pp. 46-48, March-April 2007.

[4]. Ananda, M. “The NAVSTAR GPS System”, AGARD Lecture series, 1988.

[5]. Ananth, R., “From dial tone to location tone: Location enabled handsets are giving rise to new and diverse mobile workforce management applications”, Proceedings of Location 2006, 2nd International conference on Positioning, Navigation and Timing, Bangalore, 7-9 June 2006.

Web References

[1]. Athanasia, A., Spyridon, P., Nikos, H. and Alexandros, K., “Issues for the provision of Location-dependent services over 3G networks”, Athens, 2003.

[2]. Ayala, K.J., “The 8051 Micro controller Architecture, Programming and Applications”, Penram International, 2001.

[3]. Bahl, P., Padmanabhan, V. and Balachandran, A., “A software system for locating mobile users: Design, evaluation and lessons”, MSR Technical Report, Feb., 2000.

[4]. Bancroft, S., “An Algebraic Solution of the GPS Estimation”, IEEE Trans. on Aerospace and Electronic Systems, vol. AES-21, pp. 56-59, January 1985.

[5]. Bansal A., “3G Rollout in India”, Electronics for you, Vol. 37, No. 11, Nov., 2005.