Contents lists available at Science-Gate

International Journal of Advanced and Applied Sciences

Journal homepage: http://www.science-gate.com/IJAAS.html

SMAS: An efficient automated student movement authentication system

Abdullah J. Alzahrani *

College of Computer Science and Engineering, University of Ha'il, Ha'il, Saudi Arabia

ARTICLE INFO

Article history:
Received 24 April 2020
Received in revised form
19 July 2020
Accepted 30 July 2020

Keywords:
Authentication
Runtime validation
RFID
Tracking system

ABSTRACT

Current tracking technologies can produce substantial amounts of personal data that improve users' lifestyles. This data can be utilized to enhance and support the safety of student transportation and monitor outcomes. Many vendors with profitable purposes are collecting and processing this data. One of the main tracking technologies currently in use is the Radio Frequency Identification (RFID) system that can be enabled as self-tracking, but it can also become more sophisticated with the resulting lack of privacy. RFID features and constraints could result in a variety of privacy and authentication issues. In this paper, an efficient automated student movement authentication system is proposed. This system will be useful for fleet operators in monitoring student movement and for parents to watch over their children's safety. It uses two-factor authentication, i.e., RFID and biometric technologies, to guarantee automatic student movement identification. The proposed system is divided into four main components: a student movements collector, the enforcement of policies, the movement runtime validation, and the decision components that are based on rule-based and policy enforcement approaches. In addition, this approach evaluates with a simplified scenario by presenting the whole process through different phases, namely the student enrollment phase, the policy enforcement phase, and the validation algorithm phase.

1. Introduction

Every day, students around the world attend school, and parents worry about their safety. Many accidents can occur, such as a student falling asleep on the school bus and the bus driver forgetting to check the school bus after dropping the students at school, which on a number of occasions, has resulted in the loss of a child's life. The security of school bus transportation is a crucial issue in terms of ensuring that the students ride in the correct school bus and that they are safely transferred from home to school and vice versa at the right time. With the new era of technology, these issues become easy to solve by using various types of approaches. To automate public transportation and student safety during the academic year, a real-time system using authentication techniques needs to be made available. The system described in this paper does exactly that and solves the problems outlined above.

Authentication is the process of identifying a user's legitimacy and verifying the user, device, or other entity in a computer system attempting to access a resource (Cranor and Garfinkel, 2005). Various authentication techniques can be used to provide reliable and safe school transportation environments for children. The three main characteristics of authentication approaches are knowledge factors, something known to the individual (e.g., a password, or passphrase); possession factors, something the authorized individual has in their possession (e.g., radio frequency identification (RFID), phone, or smartcard); and inhered factors, metrics intrinsically owned by the authorized individual (e.g., biometric features, such as fingerprint, eye pattern, or voice pattern) (Syed Idrus et al., 2013).

In today's advanced technological world, many systems employ automatic identification methods to detect and track the movement of students and school buses. RFID and wireless sensor networks (WSNs) are well known in the area of automatic tracking systems. RFID depends on wirelessly communicating between an applied tag and its reader (Wamba et al., 2013). It acts as a base for automated data collection, identification, monitoring, tracking, and reporting. RFID systems consist of two

* Corresponding Author.
Email Address: aj.alzahrani@uoh.edu.sa
https://doi.org/10.21833/ijaas.2020.12.009
Corresponding author's ORCID profile:
https://orcid.org/0000-0001-5297-3494

elements: A reader (or interrogator) and a transponder (or tag). The reader contains a transmitter, an antenna, a receiver, and a power supply, with an interface card to connect to a host or server. The RFID tag is a simple and small device, similar to a sticker that can be attached, and has a transmitter, an antenna, and a uniquely identified integrated circuit (IC) with memory (Karantzalis, 2015).

There are two types of RFID tags, active and passive, both containing an internal power supply. Active tags dynamically produce a radio frequency signal, while passive tags produce a radio frequency signal only in response to a query from a transporter (Want, 2006). Active tags are highly reliable and support transmission power, allowing them to obtain data from distances of several tens of meters. Also, a sensor integrated into active tags contains a larger memory and has the ability to store the history of sensor data in much the same manner as data loggers. On the other hand, passive tags have a longer lifetime and a significantly lower cost. The generated signal of the RFID tag contains information that has a unique identifier linked to a database, or it could be data programmed into the tags and then broadcast by signal to the RFID reader (Finkenzeller and Muller, 2010). RFID readers send a signal to the tag and read its response. These responses are generally transferred to a controller on a server that is running RFID software. The sensor decodes the information on the IC tag, and the data is transferred to the server for processing (Finkenzeller and Muller, 2010). RFID and the sensor together enforce access control policy.

Using RFID and biometrics as an authentication technique in a tracking system helps to guarantee student safety and security while traveling to and from school. In Saudi Arabia, currently, no school tracking system exits. Students leave their home and wait for a school bus to arrive. A school bus picks up a group of students who have gathered together, and the bus drivers do not keep track of students who get on the bus. The bus driver drops the students at their school and parks the bus without keeping track of students who left the bus. Upon students arriving at school, the school administrators usually track the students manually and do not recognize the absent students easily. Due to a large number of students at school, this method becomes inappropriate. Keeping track of students' movements at every step by counting the number of students and calling out their names is time-consuming and less accurate. There is a big chance of losing information and having illegal people attend a certain school. For example, any person can board a school bus and get into the school undetected. Also, a student can request a friend to mark them present at school, even if the student is not there. It is also challenging for a few teachers and administrative staff to supervise all the students and track their movements to and from school precisely and efficiently. Many of the parents are working and do not have time to drop off and pick up their children from school. Parents are also worried about the safety of their children during the trip to school and back home again. Recently, two children were forgotten on a school bus, sadly resulting in the loss of their lives.

To overcome the issues surrounding tracking student movements between home and school, a real-time tracking and monitoring system needs to be utilized. This system can investigate all events related to the movement of students via the historical behavior of each student by putting movement policies in place. It obtains data on student movements through particular movement activities from the beginning of the school day all the way through the system until the student returns home. The authentication model improves assessment of the movement behavior of students as these students will be observed corresponding to a policy at runtime. This paper presents an efficient and reliable student movement authentication system using a variety of authentication techniques. It utilizes RFID and biometric devices that need to be presented per school buses and school gates. The proposed system observes and records the movements of the students from the time that the student leaves their house in the morning until they return home again. It also monitors school bus transportation and provides safe school environments. The system gathers valuable information that is useful for students, parents, school administrators, school bus drivers, and managers.

The rest of this paper is organized as follows: In Section 2, the related work is presented; in Section 3, the Automated Student Movement Authentication System (SMAS) is explained; in Section 4, the Automated SMAS design and implementation is illustrated; in Section 5, the conclusion and future work are summarized.

2. Related work

The concept of adopting automated tracking systems in the public transportation system is a challenging area of research that involves smart technology. Currently, several RFID technologies have been embedded and developed for a smart school system, and research is still ongoing.

An RFID system for school bus transportation requires a set of RFID readers organized with structure and functionality. Based on the work by Shaaban et al. (2013), RFID technology is utilized to track children during their journey to and from school on school buses. Their system consists of four components: the on-board/in-school RFID tracking system, the on-board/in-school smart gateway, the back-end server, and the end-user applications. Vishaka and Godse (2015) proposed a system to monitor the pick-up/drop-off of students at school to improve the safety of children while they are transported to and from school. Their system consists of a bus unit that is used to detect when a child boards or leaves the bus, a parent unit that is used to provide information to the parents, and a

school unit that monitors which of the children did or did not ride or leave the bus.

Bhor et al. (2017) introduced a monitoring and tracking system that focuses on tracking a school bus, discovering its location, and observing the children. Their approach employs a module kit consisting of RFID, global positioning (GPS), and mobile communication (GSM) to obtain accurate location and time of specific students, and monitors the bus speed and reports on it. Their system analyzes the position of the school bus and information about drivers and children, and whether the bus follows the right track. Minu and Adithya (2018) presented an application that can assist college students to get to bus waiting points at the right time so as not to miss the bus. They used IoT and Arduino to implement this product. Abhilash et al. (2017) presented a tracking school bus system that offers collaboration more competently and successfully, resulting in better consistency and security. Their proposed system uses a straightforward yet clean GUI and utilizes an Android application. Also, their approach improves and preserves student information, parent/authority contacts, and emergency notifications.

Various researchers have studied the school bus tracking system, but SMAS is focused more on the authentication of student movements using various approaches. The proposed system has the ability to prevent the students from getting on the wrong bus, to monitor absent students, to check if a student has left the bus at the wrong station, and to observe if a student has been left behind at school or on the bus. This system employs RFID technology, which adapts to student movement authentication with its inherent resource limitations.

3. Proposed system

Focusing on tracking student movements to and from school, the proposed system designed for school bus passengers uses authentication approaches to validate student movements. The architecture of the Student Movement Authentication System is shown in Fig. 1. The architecture illustrates a conceptual model that defines the structure and the authentication approach of student movements. In this research, a passive RFID tag is used, which is very cheap, small, and it does not need a battery. Also, passive tags can serve for a longer time than active tags. On the other hand, an RC522 RFID reader is used, which needs 5 VDC power, has a read range of 12 cm, and sends signals to passive tags. In return, each tag transmits a unique identification number to the RFID reader. In addition, an R305 fingerprint scanner is utilized that handles fingerprint enrollment and fingerprint matching. The system will create a pattern of the finger and evaluate it with its pattern collection. The spaced frequency transformation algorithm is used (Mil'Shtein et al., 2008).

A student holds a unique RFID tag that is implanted on the student's smartcard. The card issued to each student has an RFID chip with a unique identification number, which is placed alongside the RFID reader. It sends the data to the movements collector and then checks against the authentication policy by using the checker, along with the other collected data sent by the sensor reader. A student's movements through the RFID and the sensor are stored and updated in the database. The result is then reported and displayed on the school bus screen for the drivers, linked to the parent's application, and shown on the school office dashboard. As depicted in Fig. 1, SMAS is divided into four areas: A student movements collector component, a policy enforcement component, a student movement runtime validation component, and a decision component.

3.1. Student movements collector component

This component is responsible for collecting, combining, storing, and retrieving student movement data in order to perform more robust authentications and validations. The main purpose of the collection component is that it is continually active and receiving data from RFID readers and sensors. Upon receiving information from the RFID reader, the movements collector timestamps and extracts the required features from the received data and then sends the student movement information to the policy enforcement component. The content of the student movements collector will be updated periodically with relevant features, such as student ID, school bus ID, school gate ID, student location, and movement status. Additionally, the collector deals with the features provided by all types of RFID sensors and employs mechanisms that are located on the bus, at the student's home, and at the school to send data to this component. This has been one of the most important challenges for smart transportation systems. Sensor deployment within a transportation network requires school buses, students' houses, and each school to be equipped with new services such as RFID. Sensors collect environmental information in real-time, which is then handled and investigated to enhance the students' transportation experience and make it resilient.

3.2. Policies enforcement component

Authentication is the key feature of any measurement system in order to recognize and verify the movements of students and their safety continuously in a tracking system environment. This system focuses on possession and biometrics techniques combined with behavioral level authentication in the proposed approaches, established by making use of student cards that permit students to use school transportation. In addition, the system checks student movements by issuing a challenge randomly to request a student to use a fingerprint in addition to their RFID card. This challenge guarantees student authentication

according to student profiles and behavior within school trip activities. The policy enforcement component operates on two levels as follows.

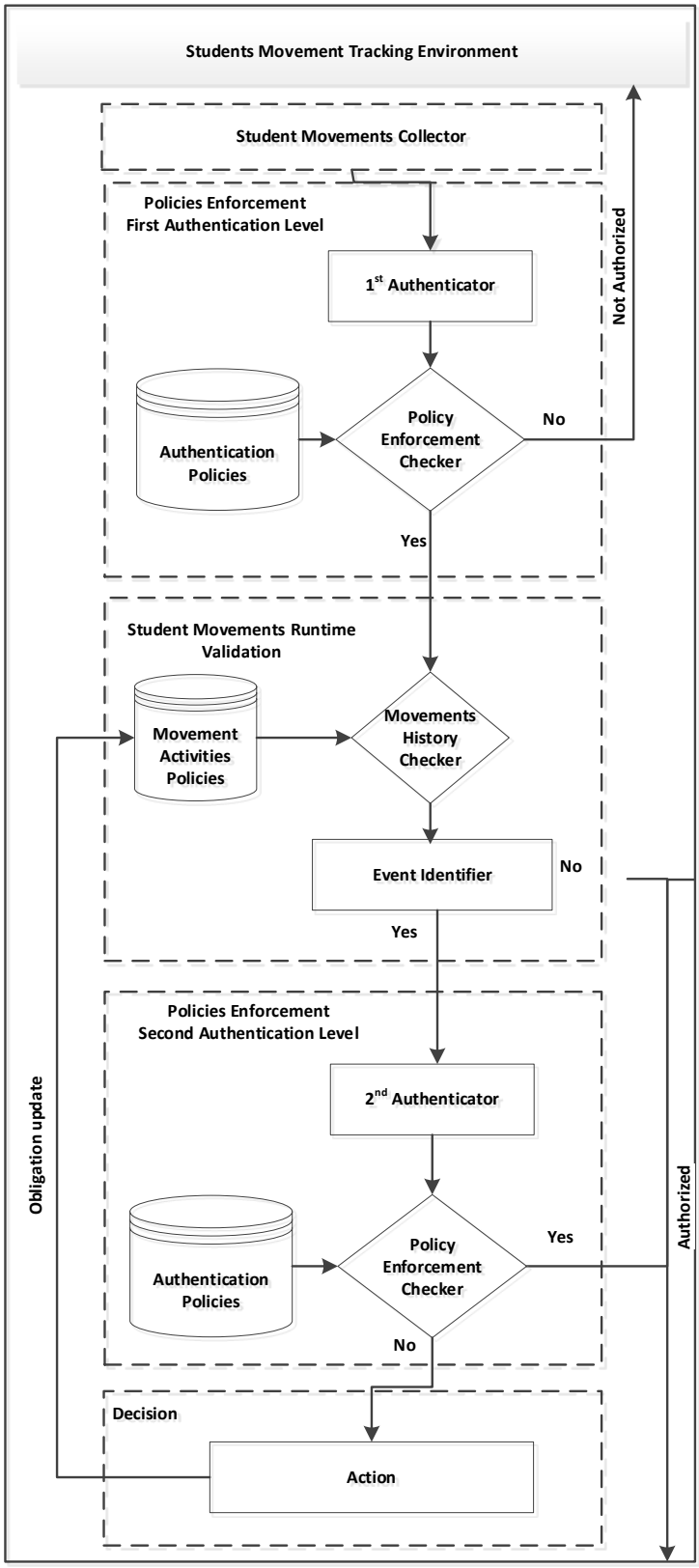

Fig. 1: The students' movement authentication system architecture

3.2.1. First authentication level

This level has three elements, namely the 1st authenticator, a policy enforcement checker, and an authentication policy repository. To authenticate a student's identity with their movements, the RFID card is scanned by the RFID reader as the first authenticator. This then passes the information to the policy enforcement checker, which imposes the authorization policies to confirm the student's movements. If a student is verified by the checker, the system can classify the student's previous

movements in the student movement runtime validation component to determine if a student is exhibiting abnormal behaviors, such as getting on the wrong school bus or off at the wrong bus stop. Also, it is used to verify whether the student has left or stayed on the school bus, the student has been left behind at the school, or the student is absent.

3.2.2. Second authentication level

In the case of abnormal behavior of a student's movements, the second-factor authentication level uses a fingerprint biometric reader (2nd Authenticator), which is capable of verifying certain students or can be set to randomly select a number of students to be verified. Subsequently, the policy enforcement checker informs the movement policies database of any obligation rule, and the result is displayed on the screen of the school bus, and parents are sent notifications.

3.3. Student movements runtime validation component

The students' movement behavior is validated against the active policy at runtime to ensure that the given policies are being followed and violations are reported. This technique is very important in effectively inspecting the movement behavior of students on their trip from home to school and vice versa. There are a number of constraints associated with student movements in a policy. Additionally, this technique is accomplished using a runtime validation method that operates in accordance with the AnaTempura validation toolkit (CSI, 2019). At runtime, safety and timing properties are endorsed through this tool. As illustrated in Fig. 1, this component has two elements, and these are described in the following subsections.

3.3.1. Movements history checker

The movements history checker involves the actions in the previous student movement states and the movements repository. The judgment of whether the constraints of the policies are being met by the movement behavior of the student on their trips to and from school is made by the movements history checker. Determining the baseline values of a new student who has no movement history is frequently a first step in making the system homogeneous. It is important that the movement history is produced as identical as possible to real movement history. The baseline value of new students with no movement history will be entered information by the school administrator at the enrolment phase. To examine the movement behavior of the students, SMAS utilizes history-based events from the movement repository. These events are implemented by the movements history checker, which discovers whether students have had undesirable movements. Students' movement behavior will be updated continually according to movement history checker outputs.

3.3.2. Event identifier

Each school bus drives on a certain route, and the direction of the bus route depends on the student's house and school locations. The student's journeys to and from school go through many steps. As an example, a student who is getting on the school bus is called an event. These steps basically can be called a set of events. The event identifier gets an understanding of the time and location of an event by tracking the daily movements of students (trips to/from school). As a dialogue system research, the event concepts are more widespread than "trip" related tasks. The event identifier is carrying out the tasks of the workflow whereby a student's movements are required to be authenticated again by the second authentication factor, based on the output of the movements history checker. The results of the movements history checker are considered for the classification of an event, verifying whether the movement is similar to the past behavior of that particular student. Subsequently, the event identifier element is able to spot abnormal events: if a student is exhibiting normal behavior (good history record) and is not selected for random re-authentication by the 2nd Authenticator, the student is considered authorized; otherwise, the event identifier demands that the student must be re-verified by the 2nd Authenticator.

3.4. Decision component

There is a correlation between the main components, namely the first authentication level, the student movements runtime validation component, the second authentication level, and the decision component. In this last component, when the policy enforcement checker has discovered a policy breach based on a student's movements at the second authentication level, the decision component takes action and issues notifications. It then analyzes the results sent by the policy enforcement checker to prevent possible abnormalities in the student's behavior that could occur and affect the safety of the student and shows the actions that are being taken for that particular student. The action performed is based on the present state of the student's movements, and the second authentication level updates the movements repository only if the student is not authorized.

The student is authorized and is reports sent under the following circumstances: When a student leaves or returns to their house, the parents are notified; when a student gets on or off a bus, the bus driver is notified; when a student arrives at or leaves school, a school administrator receives a notification. The student will not be given authorization, and an alert will be raised under the following circumstances: If a student arrived at or left school at the incorrect time, a red flag notification will be sent

to their parents, the bus driver, and the school administrators; if a student got on the wrong school bus, the bus driver and the parents would be notified; if a student did not leave the school bus at the right stop, the bus driver and the parents would be notified; if a student did not arrive at the house or did not leave the house at the right time, the parents will be notified.

3.5. Movement authorization policies

Previously, several policy requirements were discussed. To discover the abnormal movement behavior of students, the students' movement objectives adhere to certain rules with an authentication approach to express these rules, such as discretionary and obligation rules, in many policy languages. To discuss policy coverage criteria in general, this paper models access requests and policies as follows.

3.5.1. Authorization policy

Many access control policy languages are used in various systems. One approach is called discretionary policies, which depend on the identity of the requestors to impose access on the system and determine explicit access rules. These indicate under what circumstances a subject is authorized or not authorized to access certain objects in a system and allows who can or cannot perform which action on the system (Samarati and de Vimercati, 2000).

To build the rule structure, let S, O, and A denote the set of subjects, objects, and actions in the proposed system, respectively. Each subject, object, or action is associated with a set of elements that may be used for authorization decisions. A student is represented by subjects, a student's movements are represented by objects, and actions are performed by a subject on an object. Based on the system workflow, objects and actions are granted privileges when they meet the defined rules by providing a context where access is allowed. Otherwise, the system can provide different privileges. All of the events are checked and verified by the defined rules in the system. The status denotes $Done\ (S, O, A)$, where action A has been performed by subject S on object O. However, an access control decision is denoted by $auth\ (S, O, A)$ in the policy language utilized. In this research, the semantics of the rule is not the focus of this study, which is used mainly for security access control with policy-based management systems (Samarati and de Vimercati, 2000), and this research focuses on controlling the student movements.

Authorization rules assert the status of the access control, whether to allow or deny access and will authorize the access (PERMIT) if the request passes the rules; otherwise, the authorization rule will reject (DENY) when the result of an access request is negative. The conditions of whether to permit or deny an access request can be perceived based on the positive or negative rules. Thus, the condition should utilize system-based and behavioral-based policies that can be determined by the historical behavior of a student's movements, and then the authorization rules apply the environmental policies. Every school day, the policies are imposed and verified to check when there is a risk of abnormal behavior of the student movement and unauthorized access, such as when a student rides the wrong school bus or uses a different student ID to get into the school. The system will ask the student to use the second authentication factor, such as a fingerprint, which defines behavioral policy interventions designed to convey individuals' behavior toward optimal choices.

The scenario of a student's trip from home to school and back is discussed in the following cases:

1. Get on the School Bus Policy: To guarantee a student rides the right school bus from home or from school, the Get on the School Bus policy allows access if there is a corresponding positive authorization and denies it otherwise. In other words, students scan their ID cards on the school bus reader when boarding, and if it matches, it will permit the student to ride on the bus. Otherwise, it will deny the student access.

a. Let a student ID (SID2019) be the S and the school bus ID (SchBusID01) be the O; the action needed to be taken is permitted:

$$\boxed{Ture \rightarrow auth(SID2019; SchBusID01; GetOnBus)}$$

b. Where the location of the student movement is 47.662528, 21.710028 by taking the latitude and longitude of the current position:

$$\boxed{position(47.662528,21.710028) \rightarrow auth(SID2019; SchBusID01; GetOnBus)}$$

c. Where access at this point has not previously been denied for the student:

$$\boxed{\oplus\ Done(SID2019; SchBusID01; GetInBus) \rightarrow auth(SID2019; SchBusID01; GetOnBus)}$$

d. Let a student ID (SID2019) be the S and the school bus ID (SchBusID01) be the O; the action needing to be taken is denied. The system will ask the student to use the second authentication factor:

$$\boxed{\ominus\ denied(SID2019; SchBusID01; GetInBus) \rightarrow auth(SID2019; SchBusID01; GetOnBus)}$$

2. Get off the School Bus Policy: To verify the correct location of a student's school and house, the Get off the School Bus policy denies access if there is a corresponding negative authorization and allows it otherwise. In other words, upon arrival of the school bus at the school, it will get through the school gate, which will prove the bus can drop the students off at a certain school, and then the school bus door will be

opened. Otherwise, notify the bus driver that they are at the wrong school. The same will apply when the school day ends, and the students are taken to their homes:

a. Let a school bus ID (SchBusID01) be the S and the school ID (SchID001) be the O; the action needing to be taken Open Bus Door:

$$Ture \rightarrow auth(SchBusID01; SchID001; GetOffBus)$$

b. Where the location of a student's home or school is 47.473676, 21.644764 by taking the latitude and longitude of the current position:

$$position(47.473676,21.644764) \rightarrow auth(SchBusID01; SchID001; GetOffBus)$$

c. Where access at this point has not previously been denied for the student:

$$\oplus\ Done(SchBusID01; SchID001; GetOffBus) \rightarrow auth(SchBusID01; SchID001; GetOffBus)$$

d. Let a school bus ID (SchBusID01) be the S and the school ID (SchID001) be the O; the action needing to be taken to deny opening the bus door. The system will ask the student to use a fingerprint scanner as the second authentication factor:

$$\ominus\ denied(SchBusID01; SchID001; GetOffBus) \rightarrow auth(SchBusID01; SchID001; GetOffBus)$$

3. Open the School Gate Policy: This policy allows access if there is a corresponding position authorization and denies it otherwise. In other words, students are not allowed in or out of the school without their information being verified, and if a student is authenticated, the access would be granted, and the gate opened. Otherwise, access will be denied, and the system will send an alert to the school administrator:

a. Let a student ID (SID2019) be the S and the school ID (SchID001) be the O; the action needing to be taken Open School Gate:

$$Ture \rightarrow auth(SID2019; SchID001; OpenSchoolGate)$$

b. Where the location of the school is 47.473441, 21.645864 by taking the latitude and longitude of the current position:

$$position(47.473441,21.645864) \rightarrow auth(SID2019; SchID001; OpenSchoolGate)$$

c. Where access at this point has not previously been denied for the student:

$$\oplus\ Done(SID2019; SchID001; OpenSchoolGate) \rightarrow auth(SID2019; SchID001; OpenSchoolGate)$$

d. Let a student ID (SID2019) be the S and the school ID (SchID001) be the O; the action needing to be taken to deny the opening of the school gate:

$$\ominus\ denied(SID2019; SchID001; OpenSchoolGate) \rightarrow auth(SID2019; SchID001; OpenSchoolGate)$$

3.5.2. Obligation policy

The subject performs an action on a particular object that is specified by obligation rules (Samarati and de Vimercati, 2000). An obligation action is taken when one of the policy enforcements fails, such as when a student is permitted access by RFID as the first authentication factor and is not authorized by a fingerprint as the second authentication factor. The system processes this kind of behavioral policy by creating obligation rules.

Let a student ID (SID2019) be the S and the school ID (SchID001) be the O; the action needing to be taken to permit the opening of the school gate. Consequently, the action that can occur would be to notify the school administrator (SchAdmin001).

$$\oplus\ Done(SID2019; SchID001RFID; begun)\Lambda \ominus\ denied(SID2019; SchID001FINGRPRINT; begun) \rightarrow Oblige(sys, SchAdmin001;$$

3.5.3. Technology: Runtime validation toolkit (AnaTempura)

Using a logical representation of time is required for a system that can employ appropriate interpretation of the collected contents. The obtained content involves extrapolation of the time and location of events that draw from the sequence of the event and includes the previous and next events. Propositional Interval Temporal Logic (PITL) denoted a flexible means of presenting propositional and first-order reasoning and was designed as a tool for the specification and verification of systems depending on the periods present in descriptions of systems. Also, a variety of research used events calculus as alternative formulas. Tempura technique employs a runtime validation of functional and real-time properties, which were applied previously in many studies where timing is vital (Zhou et al., 2005). Developing knowledge-based policies are fundamentally safety properties; thus, they can be implemented in Tempura.

AnaTempura is a runtime validation tool used to verify the system using PITL. The runtime validation technique is used to check whether a system satisfies timing, locations, and safety properties defined in the PITL (Janicke et al., 2005). In the SMAS architecture, the event identifier component designates the assertion by receiving a series of events that include movement states. The runtime validation evaluates whether the series of events obtainable by the event identifier component belongs to the predefined rules related to the safety and timeliness properties presented by the student

movement policy. Therefore, this is the reason for using AnaTempura to check the runtime events of the movement policy.

4. System design and implementation

To illustrate the use of the Automated Student Movement Authentication System, an explanation of the proposed system architecture and design is presented within a small simplified scenario that demonstrates the use of the movement policies. SMAS introduces a system that provides the student enrollment phase, the policy enforcement phase, and the validation algorithm phase as follows. Also, Fig. 2 and Fig. 3 shows a sequence diagram for student enrollment and sequence diagram for the authentication phase, respectively.

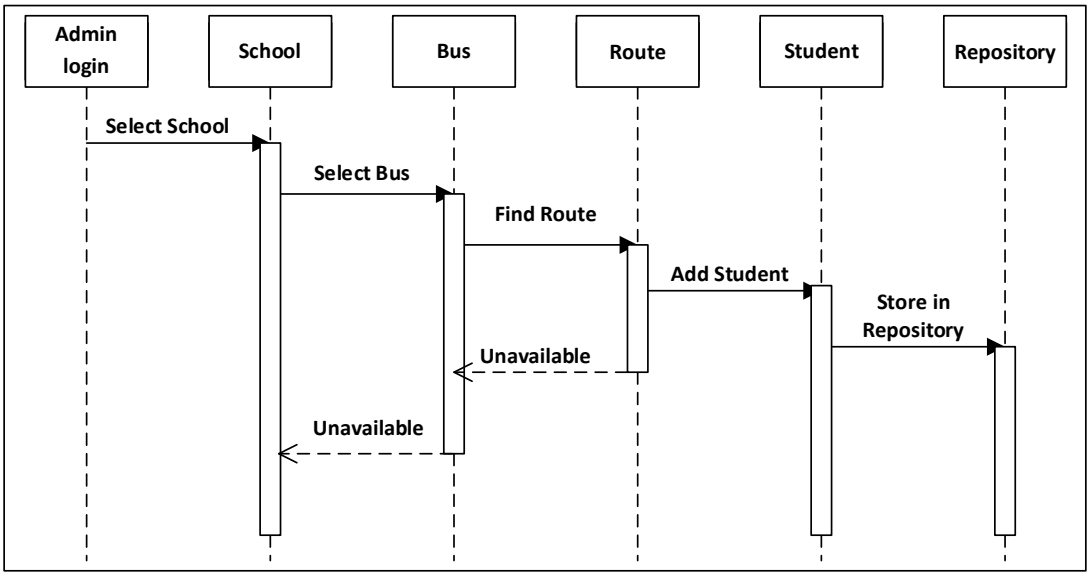

Fig. 2: Sequence diagram for student enrollment

4.1. Student enrolment phase

To allow the students to obtain the benefits of riding the school bus safely, they are required to sign up through the administration office. A sequence diagram gives an external view of the student enrollment phase, depicted in Fig. 2, which shows the process of adding schools, adding school buses, checking student house locations (route), checking bus schedules, adding new students, and adding data to the system. The school administrator has to set up a school location by entering latitude and longitude values and has to select a school bus and check available seats. The administrator has to find the shortest route to the student house locations in latitude and longitude and can add new students. The students' information is taken, including full name, student ID, and parents' phone numbers, which are stored in the database.

4.2. Policy enforcement phase

Once students have enrolled to use the school transportation system, the system keeps a check on the student's movements and reports abnormal behavior. A simplified scenario is used below to demonstrate how SMAS uses history-based policies for student movements. Scenarios help to understand student movements and show how every student's movements in each school day are recorded. Students journey from their house to school and back through a number of steps. In the first step, a student gets on the school bus. In the second step, the school bus arrives at the school, and the student gets off the bus. In the third step, the school gate opens, and the student goes into the school. These steps are illustrated in Fig. 3.

Step 1: Students make use of their RFID cards to get onto a school bus. For the students' movement validation, the checker obtains the student identity in each movement forwarded by the RFID reader. If the authorization is granted, the current movement is logged by the system in the database. In Policy 1, students (STD) and their movement ($mvRFID$) belong to getting onto the school bus. $mvRFID$ indicates the movement which must be conveyed by RFID technology. However, if the Time Unit in the last movement of the interval is below the time of the previous movement, implying the number of sessions for a particular student movement and always over the states, the previous movement is approved via the RFID or fingerprint sensors by using the time of the current location of the school bus. In the following state, the student is allowed to use the RFID reader for the movement $mvRFID_i$. If the authorization is denied, it means that a student is found to have an untrustworthy movement history, the history of the abnormal movement as determined by the system. Policy 2 is then enforced, where if the previous session has been unauthorized by the RFID technology then in the subsequent movement the student must use the fingerprint

technology $mvFINGERPRINT_i$. Due to the abnormal behavior of a particular student, this policy is enforced to prevent the student from conducting authentication through RFID solely all the time.

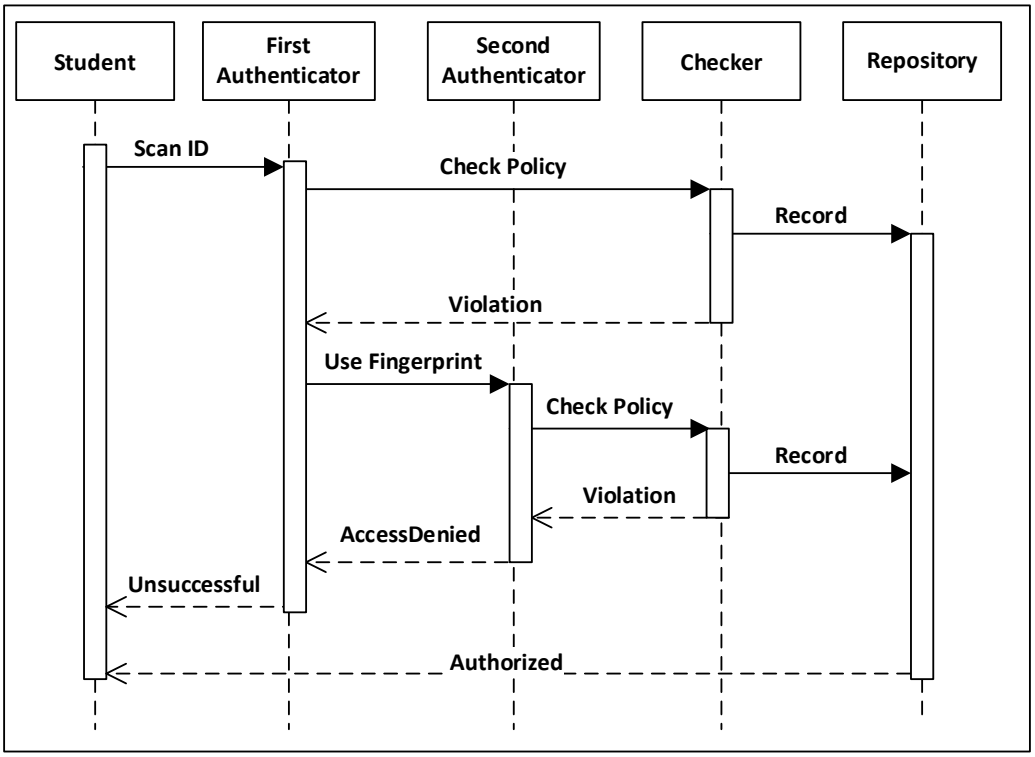

Fig. 3: Sequence diagram for the authentication phase

$Policy1 =$

$$\begin{pmatrix} fin(TimeUnit < CurrLocationTime) \wedge \\ Student(STD, SchoolBus) \wedge \\ movement(mvRFID, SchoolBus) \wedge_{i=0}^{n-1} \\ \blacksquare\ done(STD, mvRFID, apply) \end{pmatrix}$$
$$\rightarrow Auth^{+}(STD, mvRFID_n, GetInBus)$$

$Policy2 =$

$$\begin{pmatrix} fin(TimeUnit < CurrLocationTime) \wedge \\ Student(STD, SchoolBus) \wedge \\ movement(mvRFID, SchoolBus) \wedge_{i=0}^{n-1} \\ \blacksquare\ done(STD, mvRFID, apply) \end{pmatrix}$$
$$Auth^{-}(STD, movRFID_n, GetInBus)$$
$$\rightarrow \wedge$$
$$Auth^{+}(STD, movFINGERPRINT_n, GetInBus)$$

Step 2: Students use RFID cards to get off the school bus for their movement authentication, and the checker receives the information sent by the RFID reader. Policy 3 is employed if the authorization is obtained to permit the students to get off the school bus, where students' STD and their movement (mvRFID) belong to getting off the school bus. $mvRFID$ indicates the movement which must be conveyed by RFID technology. Policy 4 is applied if the access is not authenticated, meaning a student is found to have an untrustworthy movement history such as a wrong house or school location. It is imperative to employ different identification techniques, such as the fingerprint technology for the student movement $mvFINGERPRINT_i$. This policy is enforced for all students who have abnormal behavior preventing them from using RFID technology.

$Policy3 =$

$$\begin{pmatrix} fin(TimeUnit < CurrLocationTime) \wedge \\ Student(STD, SchoolBus) \wedge \\ movement(maRFID, SchoolBus) \wedge_{i=0}^{n-1} \\ \blacksquare\ done(STD, maRFID, send) \end{pmatrix}$$
$$\rightarrow Auth^{+}(STD, maRFID_n, GetOffBus)$$

$Policy4 =$

$$\begin{pmatrix} fin(TimeUnit < CurrLocationTime) \wedge \\ Student(STD, SchoolBus) \wedge \\ movement(maRFID, SchoolBus) \wedge_{i=0}^{n-1} \\ \blacksquare\ done(STD, maRFID, send) \end{pmatrix}$$
$$Auth^{-}(STD, maRFID_n, GetOffBus)$$
$$\rightarrow \wedge$$
$$Auth^{+}(STD, mvFINGERPRINT_n, GetOffBus)$$

Step 3: To Open the School Gate, students use an RFID tag to get inside their school, and the checker receives student information sent by the RFID reader. When the authorization is granted, the current state is recorded by the system in the database. Policy 5 is enforced when the authorization is not granted, meaning a student has been determined to have an abnormal movement history, such as trying to get into the wrong school. It is necessary to use different authentication techniques, such as fingerprint technology. The checker authenticated the student's fingerprint for the student movement $mvFINGERPRINTi$ as in Policy 6. Students who have a bad history are required to use this policy to perform authentication by using fingerprint technology.

$$Policy5 = \begin{pmatrix} fin(Location(SchGate))\Lambda \\ Student(STD, schoolGatet)\Lambda \\ movement(maRFID, schoolGatet)\Lambda_{i=0}^{n-1} \\ \blacksquare\ done(STD, maRFID, opengate) \end{pmatrix} \rightarrow Auth^{+}(STD, maRFID_n, OpenSchoolGate)$$

$$Policy6 = \begin{pmatrix} fin(Location(SchoolGate))\Lambda \\ Student(STD, schoolGatet)\Lambda \\ movement(maRFID, schoolGatet)\Lambda_{i=0}^{n-1} \\ \blacksquare\ done(STD, maRFID, opengate) \end{pmatrix} Auth^{-}(STD, movRFID_n, OpenSchoolGate) \rightarrow \Lambda\ Auth^{+}(STD, movFINGERPRINT_n, OpenSchoolGate)$$

The abnormal movements are recognized by the observing system on the basis of conducting movement authentication by using both RFID and fingerprint technologies instantaneously. Policy 7 is an obligation policy utilized to alert the student movements database that the movements of certain students are required to be carefully observed due to previous inappropriate behaviors. The movements of these students will be recorded instantly, and they will be required to employ a second means of authentication, such as a fingerprint, for every movement thereafter, or until they display good behavior for a time. Additionally, the system selects random individual students to authenticate their movements through the fingerprint reader, especially when getting in and out of school. The movement checker permits a student to be added to a database as a present student, but if denied, the student is logged as absent, and their movement is labeled as abnormal behavior in the database.

$$Policy7 = \begin{pmatrix} \blacksquare\ done(STD, mvRFID, update)\Lambda \\ \Diamond\ done(STD, mvFINGERPRINT, update) \end{pmatrix} \rightarrow Obligation(System, maDATABASE, report)$$

4.3. Validation algorithm phase

To monitor student movement behavior and to ensure that the policies are implemented correctly in the system, the student movement runtime validation component will verify the enforcement of the policies, as illustrated in the below algorithms. The state transition diagram illustrates the policy enforcement algorithm, shown in Algorithm 1. This algorithm examines the student movement events, starting with the students leaving their house and ending when they return home. In other words, the algorithm investigates student movements from the beginning session until the student movements complete. Also, the algorithm executes the defined policies on each existing state of the student movements.

The monitor student movement algorithm shifts from one state to other states, as shown in the illustration of Algorithm 1. This algorithm describes the global variables utilized in the whole process by determining the active movement recognized as the main state of a process. In this algorithm, the current state of the student movement refers to the variable state. There are next movements (*submovement*) of current movement *current*= (PickMov, StartMov, EndMov, OngoMov, AbortMov, AbnormalMov), and the final movement is an inactive state. Algorithm 1 describes the procedure to monitor student movement behavior. It combines tracking the student movements with the execution of policies, where the input parameter is the current location of a student.

Algorithm 1: Monitor Student Movement

```
Require
  Position[n], CurrMov, ChosenMov
Results
  Mov(Allow Access, Deny Access, Abnormal Movement )
Procedure
1: while CurrMovement.MoveToNext do
2:    (PickMov, StartMov, EndMov, OngoMov, AbortMov, AbnormalMov)
3:    while CurrMov = PickMov do
4:       (GainingAccess, Permitted, Rejected)
5:       GainingAccess is the Initial Substate!
6:    ChosenMov = readCurrMovement()
7:          if i = ChosenMov then
8:     CurrMov = Active Movement
9:     GettingAuthorization is the Initial Substate!
10:    else if CurrMovtime < TimUnit then
11:       return Report Error
12:    else
13:       CurrMov = CurrMovementPosition()
14:       for each CurrMovement do
15:       CheckClearance(Position[n])
16:       CheckEndMov(Position[n])
17:       EndorsementCheck(Position[n])
18:       CurrMov = Inactive // Final State in this process!
```

A student demands access, and the system checks the student's movements, again, either allowing or denying access based on the results of the applicable policy. The procedure uses access function *CheckClearance* () as illustrated in Algorithm 2. This algorithm takes the relevant policy from predefined policies and the movement events related to the policy parameter as input. Consequently, the predefined policies are required to check whether the student can be granted access based on the existing movement and complete the movement, or the student is not permitted to gain access, which results in ending the monitoring procedure by reporting the abort status to the parents and the system administrators.

Algorithm 2: The Procedure of CheckClearance() Function

```
Require
  Check Movement policy for clearance
Procedure
1: if CurrMov[position[n]] = AbnornmalMov ∧ EndMov ∧ AbortMov then
2:    CurrMov[position[n]] = PickMov
3:    CurrMovClearance[position[n]]=GainingAccess
4:    GetMovementPolicy(position [n])
5:    return PolicyX
6:    GetMovEvents(position[n])
7:    return SequenceOfEvents!
8:    CheckPolicyGainstEvents
9: else if Policy is Fulfilled then then
10:   MovClearance[position[n]] ← Allowed
11:   CurrMov[position [n]] ← StartMov
12: else
13:    Movementpermission[position[n]]← Denied
14:    CurrMovement[position[n]]← Aborted
```

The procedure of verifying the authentication state of the existing movements to allow or deny access to enter or leave school is illustrated in Algorithm 3. The *EndorsementCheck* () function

enforces the predefined policies despite the consequences whether the current movement is conducted or not. In the case where the system determines an abnormal movement, it requires the student to use biometric technology using the fingerprint reader to bypass abnormal movement behavior.

Algorithm 3: The Procedure of EndorsementCheck (Position[n]) Function

```
Require:
   Check Movement policy for the Endorsement status
Procedure
1: if CurrMove[Position[n]] = CompeletState then
2:        GetMovPolicy(Position[n]) //Find two-factor authentication (FRID and Fingerprint) Policy from the repository!
3:     return Policy X
4: else
5:     GetCurrEvents(Position[n])
6:     return Sequance of Events!
7:           for Check Policy against Events do
8:           if Policy is Satisfied then
9:            CurrMove[Position[n]] ←Completed
10:          else if Policy is Satisfied then
11:            CurrMove[Position[n]] ←Abnormal
12:          else
13:           CurrMove[Position[n]] ← Idle
```

4.4. Implementation

The proposed system is implemented in Python, and Raspberry Pi devices are used. RFID reader and Fingerprint Sensors with Raspberry Pi are simple with using Python Library for RFID (MFRC522 library), Fingerprint, LCD, and database. To checks whether the RFID reader and fingerprint are connected, the connection parameters are defined for the LCD screen. To obtain scanned information, new values are defined to determine the current location and timestamp of the scanned RFID cards. The card number is stored in STD and establishes a connection with the database. The proposed system checks the current location to see whether it is the same as or different from the previous location for the student movement and whether it is the last location that the student is supposed to arrive at. If the generated random number is equal to 1 or the first authentication fails, the student must use a fingerprint, and if it is matched to the record, the student can gain access. Otherwise, access is denied. The student information is shown on the screen. Similarly, the automated system records the student card information and fingerprints in the database.

5. Conclusions and future work

Supervising and tracking the students at any school manually, for example, in a school transportation system, is extremely complicated as it requires much time and involves many workers, and additionally leaves many holes. It is also difficult for school administrators and bus drivers to keep monitoring all the students at the school and for parents to be able to keep track of their children's safety. This paper describes SMAS, an effective and consistent student movement authentication system employing smart technology and a range of authentication techniques. The proposed approach exploits RFID and biometric technologies to observe students' journeys and monitors bus drivers to make sure they have not driven off with the children into the wrong stop, as well as offering safe school environments. In addition, the system provides valuable information that is useful for students, parents, and school administrators.

In summary, the proposed system consists of four components: The student movements collector, the policy enforcement, the student movement runtime validation, and the decision components that employ rule-based and history-based policy enforcement approaches. Also, the system has been evaluated with a simplified scenario by presenting the whole process through different phases: Student enrollment, policy enforcement, and the validation algorithm phases. The SMAS has been developed and can be used in a school to enhance student safety using this tracking system. It is much easier in terms of accuracy and time than the old-fashioned supervision approach. It should be mentioned that this research is completed as the first phase of the project. The future scope includes routing and scheduling techniques with an information management system that will provide more features to the operators and can also include the concept of bus-to-bus communication.

Compliance with ethical standards

Conflict of interest

The authors declare that they have no conflict of interest.

References

Abhilash R, Mahima R, Monisha S, and Nagashri R (2017). Smart tracking system for school buses. International Research Journal of Engineering and Technology, 4(4): 2519-2525.

Bhor M, Kadam N, Shinde D, and Mane P (2017). Children safety and school bus tracking solution. International Journal of Electrical, Electronics and Computer Systems, 5(1): 19-22.

Cranor LF and Garfinkel S (2005). Security and usability: Designing secure systems that people can use. O'Reilly Media, Inc., Newton, USA.

CSI (2019). Original RFID smart gate reader: RFID walk through smart gate reader. Childs Safety India, Mumbai, India.

Finkenzeller K and Muller D (2010). Security of RFID systems. In: Finkenzeller K and Muller D (Eds.), RFID handbook: Fundamentals and applications in contactless smart cards, radio frequency identification and near-field communication: 213–232. John Wiley and Sons, Hoboken, USA. https://doi.org/10.1002/9780470665121.ch8

Janicke H, Siewe F, Jones K, Cau A, and Zedan H (2005). Analysis and run-time verification of dynamic security policies. In the International Workshop on Defence Applications of Multi-Agent Systems, Springer, Utrecht, Netherlands: 92-103. https://doi.org/10.1007/11683704_8

Karantzalis P (2015). Baseband circuits for an RFID receiver. In: Dobkin B and Williams J (Eds.), Analog circuit design: 1075-1076. Volume 3, Elsevier, Amsterdam, Netherlands. https://doi.org/10.1016/B978-0-12-800001-4.00499-3

Mil'Shtein S, Pillai A, Shendye A, Liessner C, and Baier M (2008). Fingerprint recognition algorithms for partial and full fingerprints. In the IEEE Conference on Technologies for Homeland Security, IEEE, Waltham, USA: 449-452. https://doi.org/10.1109/THS.2008.4534494

Minu MS and Adithya KD (2018). Real time college bus monitoring and notification system. International Journal of Advance Research in Science and Engineering, 7(4): 14-16.

Samarati P and de Vimercati SC (2000). Access control: Policies, models, and mechanisms. In: Focardi R and Gorrieri R (Eds.), International school on foundations of security analysis and design: 137-196. Springer, Berlin, Germany. https://doi.org/10.1007/3-540-45608-2_3

Shaaban K, Bekkali A, Hamida EB, and Kadri A (2013). Smart tracking system for school buses using passive RFID technology to enhance child safety. Journal of Traffic and Logistics Engineering, 1(2): 191-196. https://doi.org/10.12720/jtle.1.2.191-196

Syed Idrus SZ, Cherrier E, Rosenberger C, and Schwartzmann JJ (2013). A review on authentication methods. Australian Journal of Basic and Applied Sciences, 7(5): 95-107.

Vishaka A and Godse SP (2015). A survey on enhance security and tracking system for school bus and children. International Journal of Trend in Research and Development, 2(6): 272-276.

Wamba SF, Anand A, and Carter L (2013). RFID applications, issues, methods and theory: A review of the AIS basket of TOP journals. Procedia Technology, 9(1): 421-430. https://doi.org/10.1016/j.protcy.2013.12.047

Want R (2006). An introduction to RFID technology. IEEE Pervasive Computing, 5(1): 25-33. https://doi.org/10.1109/MPRV.2006.2

Zhou S, Zedan H, and Cau A (2005). Run-time analysis of time-critical systems. Journal of Systems Architecture, 51(5): 331-345. https://doi.org/10.1016/j.sysarc.2004.12.003