

# Modeling and Recognition of Software Applications using FSM

Kavita Pandey<sup>1</sup>, Dhruv Singh<sup>1</sup>, Sumit Bansal<sup>1</sup> and Vrishti Gahlaut<sup>1</sup>

<sup>1</sup> Department of Computer Science and Engineering, Jaypee Institute of Information Technology, Sector 62 Noida, Uttar Pradesh, India

E-mail: kavita.pandey@jiit.ac.in, {0910dhruv, sumitbansal291994, vrishti13}@gmail.com

**Abstract**—Finite State Machines (FSMs) are mathematical abstractions which have a rich use in computer applications. If we build a FSM for software, then it becomes much easier and simpler to understand, debug and modify. The insights developed in FSMs have had great influence on various domains. FSM is as important as any other computer science tool. This paper minimizes the gap which currently exists between software development and the formal method of theoretical computer science. So importance of FSMs has been explored over various such domains in this article. These domains include *spoken web technology* that enables a user to access a massive network of voice sites through speech. Information transfer protocol for *Vehicular Computing* is another domain, where users can get on road support services using vehicular sensors and global position system. Adding on to the domain list, we have modeled FSMs for *Ankle Monitor* and *Sensor Dust* which are applications of mobile ad-hoc networks. Ankle monitor is primarily used to track movements of an individual through wireless communication and *Sensor Dusts* are tiny sensors which monitor the environment in which they are deployed. We have also modeled *Palm Operating System*, which is a mobile OS that runs on Linux kernel. This paper presents the FSM for the booting sequence and the user interface of Palm OS. To conclude with we have taken up the domain of satellite simulation and have presented a FSM for the steps involved in *Satellite launching* and *Image Handling* within a satellite. Therefore, by modeling a range of applications using FSM we attempt to add-on to the significance of this concept and at the same time provide a single document comprising numerous FSM models. This concept can be used by software developers for easy modeling of their designs and can further use it to verify and debug the model whenever required.

**Index Terms**—Finite State Machines, Spoken Web, Vehicular Computing, Ankle Monitor, Sensor Dust, Palm OS, Disaster Management.

## I. INTRODUCTION

The theoretical Finite State Machines is one of the longest established areas in Computer Science and has a non-exhaustive list of applications. The application of

FSM can be observed in many domains of the technical world which perform a sequence of actions depending on a sequence of inputs with which they are controlled. This paper describes some of these typical applications that are modeled specifically with FSMs with an aim to provide a single source of scattered application of this theoretical concept.

First of the applications that we are going to model is *spoken web* that was founded by Eyal Shalom in 2006. It is a massive web portal in which users can upload vocal sites and access them using unique phone numbers analogous to URL in World Wide Web. It is a concept primarily for underdeveloped areas lacking net connectivity and visually impaired people. In this paper we have presented a FSM for Spoken Web which can be used for tourism management where all inputs will be voice signals.

Vehicular Information Transfer Protocol is another application which is modeled using FSM. It provides assistance to the drivers based upon the information collected by vehicular sensors and GPS. Suppose a driver wants to find out information about a particular route, then it will send a request for traffic information along with its own location coordinates. The driver receives suggestions based upon the processing within the internal data structure of roads and vehicles embedded in the system. There are a number of applications where this protocol is needed, but we have specifically modeled some of events of accident reporting and GPS navigation.

Next domain which can borrow from the concept of FSM is networking. Sensor Dust is an application of mobile ad-hoc networks where sensors are deployed in an environment which is to be analyzed and controlled. Ankle monitor is also a mobile ad-hoc application which is extensively used by law enforcement to track individuals or is used to monitor pets. The monitor sends radio frequency which contains location and other information and this data is received after specified intervals of time. This paper presents the working of sensor dust and the application of Ankle Monitor for a parolee using FSM.

Another application where we have applied FSM is operating systems. Of the several operating systems, we have taken up Palm OS. Palm webOS is Palm's proprietary mobile operating system running on the Linux kernel. WebOS's graphical user interface is designed for use on devices with touchscreens. It includes a suite of applications for personal information management and makes

use of a number of web technologies such as HTML5, JavaScript, and CSS. Under this vast domain we have specifically modeled FSMs for the booting sequence and the user interface of Palm OS.

The last domain we have touched upon is Satellite simulation. Satellite launching is an integral part of satellite simulation which describes the various steps involved in launching a satellite. Also, image handling is a key requirement for any satellite to achieve its objectives. Both these functionalities have been depicted using FSM in this paper.

Basically, the aim of choosing such a diverse list of applications is twofold. First, is to show the applicability of FSM concept over various domains and the next is to provide a single source document where users can find and understand various applications using the concept of FSM. Now, let's briefly introduce the finite state machine concept with a small example. A particular FSM is defined by the following elements

- Set of states which include a start state, one or more finish states and other intermediate states.
- Set of Inputs.
- A transition set stating the transitions from one state to another based on an input.

Finite state machines can be represented either as a state diagram or a state transition table. A state diagram is a very basic representation where states are represented by circles; arrows denote transition from the base state to the state pointed by the arrowhead along with input required for that transition. *Fig.1.* shows a simple example for state diagram of a two floor elevator. If you are present at the ground floor then the down button keeps you at the ground floor and the up button takes you to the first floor. Similarly at the first floor, the up button keeps you at the first floor and the down button takes you to ground floor.

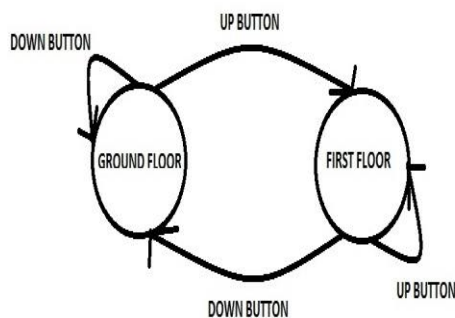


Fig.1. State Diagram of Two Floor Elevator

This can also be represented in the form of state transition table where rows represent the various states and the columns contain transitions to the next state corresponding to an input. The *table 1* shows the same example of two floor elevator using state transition table. After a brief explanation of FSM, the next section describes the work where researchers applied the FSM concept on various applications.

Table 1. State Transition Table

State \ Input	Next State	
	Up	Down
Ground Floor	First Floor	Ground Floor
First Floor	First Floor	Ground Floor

The remaining paper is organized as follows, as we have mentioned above, section II describes the work in this direction. The following section III presents the FSM for various applications like spoken web, information transfer protocol, sensor dust and ankle monitor etc. At the last conclusion is given in section IV.

## II RELATED WORK

Various researches have been carried out in order to demonstrate the applications of Finite State Machines over multiple domains. Some of these researches are: Qureshi et al. [1] proposed a verilog model of adaptable traffic control system using mealy state machine in which the design of traffic control system is carried out for an intersection point consisting of five roads and each road is divided into main road (for straight movement) and cross road (for crossing) with the aim to prevent traffic jams. Also, Sohaib Qureshi et al. [2] summarized the design of an arcade game using Automata theory tool. They primarily used Deterministic finite state automaton and nondeterministic finite state automaton in various levels of designing the game. Another research that used the concept of FSM was conducted by Hong et al. [3]. They proposed an approach for gesture learning and recognition in which they built a FSM recognizer for each gesture. The computational efficiency of these FSM recognizers allowed them to reach real time on-line performances. Further implying to the never ending applications of FSM, Monga and Singh [4] described the designing of multi select Vending machine with auto-billing features using mealy machine to demonstrate the four basic functionalities of a vending machine. These functionalities are user selection, Waiting for money insertion, product delivery and servicing. Furthermore, Drumea and Popescu [5] discussed the application of FSM in software for industrial control. They implemented FSMs for multiple control systems like modem control, microcontrollers etc. Also, Saxena and Kumar [6] used FSM as a tool to validate UML class models. They used FSM to show the dynamic behavior of the states of an object oriented system and to depict the whole life of an object.

All the above researches were focused on one application and have concentrated their efforts in gaining appreciation for FSM in one domain. However in this paper, Finite State Machines lend themselves for various applications over scattered domains. Here, we attempt to provide usability of this concept in more than one application. Also the reader can strengthen his concept of FSM using this single paper which provides examples from a range of domains. The next section lists various such domains and describes their modeling using FSM.

III. FSMs FOR VARIOUS APPLICATIONS

To explain the importance of FSMs, a number of diversified applications have been taken. This section is divided into various sub sections, where each sub section describes the working of an application using FSM.

3.1 Spoken Web

In the first sub section we have tried to explain the working of a spoken web voice site. Finite State Machines can be effectively used to describe a Spoken Web Voice Site where the user can access a network of voice sites through speech. We have taken the example of tourism Voice Site which is modeled using the FSM shown in Fig. 2. This Voice Site has many features like, a user can get to know about India and its states according to which the user can plan a trip. Users can avail the services of route planner and trip planner, available in both hindi and english languages just by dialing the required numbers.

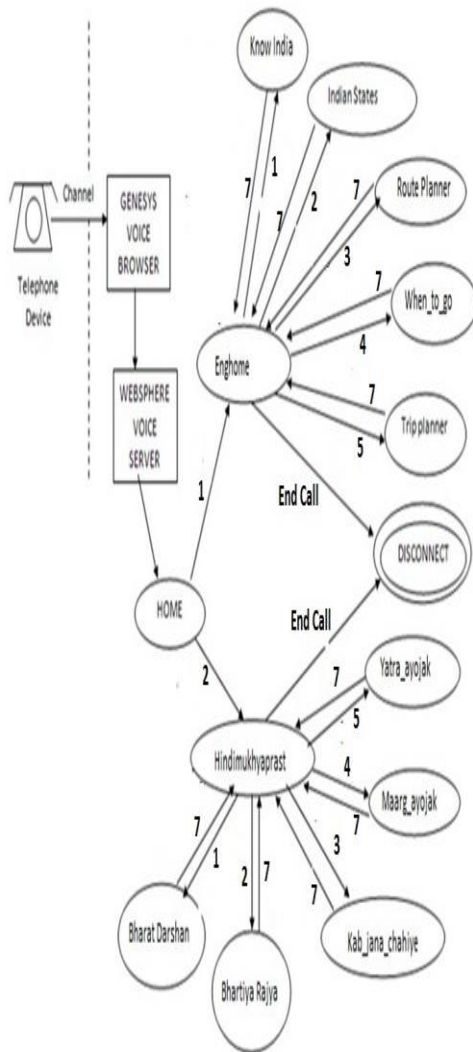


Fig.2. FSM for Voice Site

When the number for this voice site is dialed, it takes the user to the home page, which is the start state for the FSM. Pressing '1' or '2' at the home page makes transi-

tions to either engnhome or hindimukhyaprast respectively which are menu pages in english and hindi languages respectively. From here the user can dial numbers '1' to '5' to avail the services provided by the site, and can return back to the menu page by pressing '7'. The user can hit the end call button to leave the voice site.

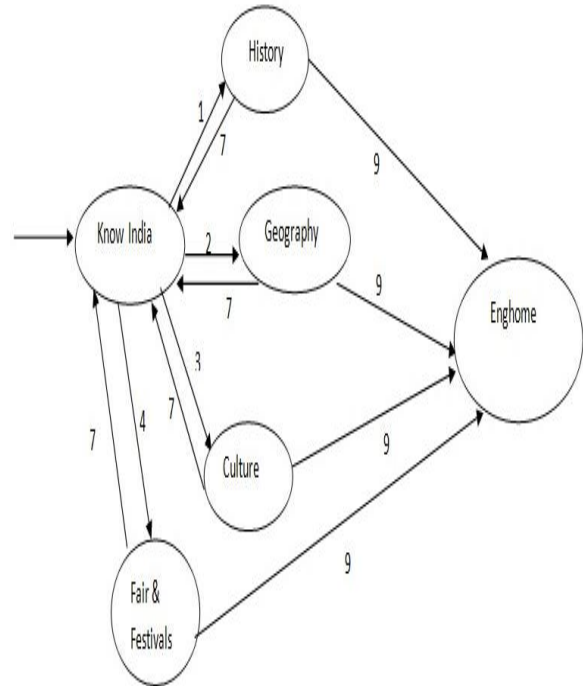


Fig.3. FSM for Know India

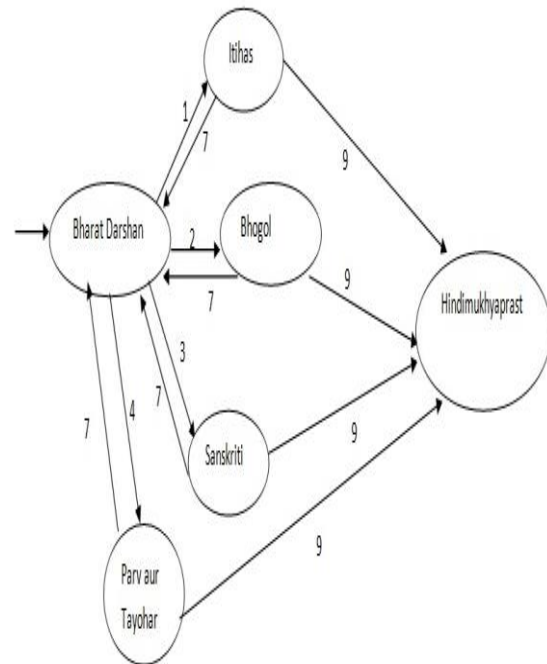


Fig.4. FSM for Bharat Darshan

Further, the detailed working of the services 'Know India' and 'Bharat Darshan' have been modeled using FSM. Fig.3. shows the FSM for 'Know India' and Fig. 4. shows the FSM for 'Bharat Darshan'. Once the user

chooses to avail the service of ‘Know India’, dialing numbers ‘1’ to ‘4’ makes transition to the states, ‘history’, ‘geography’, ‘culture’ and ‘fair & festival’. From these states the user can again go back to the state ‘Know India’ by dialing ‘7’ or can go the english menu page by dialing ‘9’. Similar working is for the hindi counterpart.

Web Technology and networks go hand in hand, therefore further we have discussed about the application of FSM in modern vehicular ad-hoc networks in the following sub section.

### 3.2 Information Transfer Protocol

Information transfer protocol is an application of vehicular ad-hoc networks (VANETs) used to assist the drivers based upon the information collected by the various sensors embedded in the car exteriors. This domain constitutes various sub domains like automated speed control system, anti-theft security system etc. In this paper we have modeled FSMs for two sub domains namely accident reporting and GPS navigation.

#### Accident Reporting

Accident reporting is a vital component of VANET whose functioning is depicted using FSM shown in Fig.5. Initially at the start state, the car is at rest. When the car is on the move and an accident occurs, the FSM reaches a state from where two transitions are possible. If there is no other car within a particular range, say 100 meters for instance, then the system will not report the accident because no other car will be affected by this accident. In the case of presence of a car within that specified range which is 100 meter here, the system will transmit a warning report. If such a report of an accident is received by any system then the driver of that car is warned, so that traffic jam at the site of accident can be avoided.

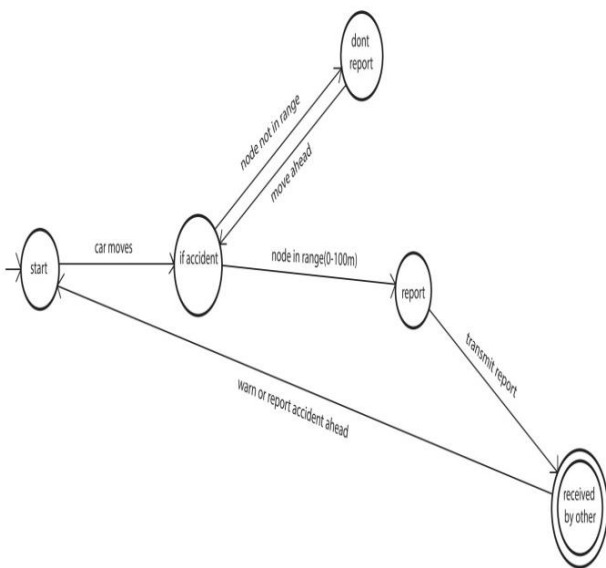


Fig.5. FSM for Accident Reporting

#### GPS Navigation

Another most widely used application of VANET is GPS. Beginning from the start state, when the driver of a

car inputs the destination location, the system accesses the satellite connection to reach a state in the FSM which suggests a number of routes from which the car chooses an appropriate route. Then the car navigates through that route and reaches the destination which is the final state. If the car wants to navigate to a new destination then there will be a transition back to the start state and same process will again commence. The FSM for this whole procedure is shown in Fig.6.

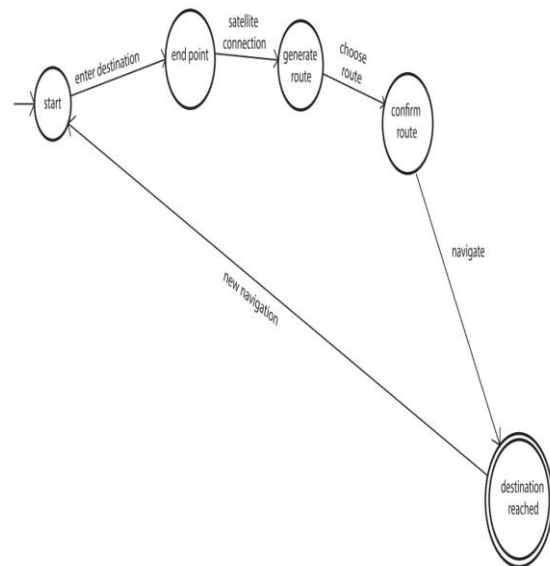


Fig.6. FSM for GPS Navigation

Continuing with the application of FSM in networking, further we present the use of FSMs in mobile ad-hoc networks.

### 3.3 Mobile Ad-Hoc Networks

A mobile ad hoc network (MANET) is a self-configuring infrastructureless network of mobile devices connected by wireless means. There are a number of applications mobile ad-hoc networks, such as air pollution monitoring, telephony services etc. Here we used FSM to model two applications of MANETs i.e. Sensor dust and ankle monitor.

#### Sensor Dust

To be able to analyze - for example - a possible atomic reactor leak, it may be a pleasant solution to place tiny sensors throughout a huge area, probably in regions where no information infrastructure is present or destroyed, instead of risking human lives by sending them to hazardous areas. Fig.7. depicts the FSM for the functioning of such Sensors.

Within the nuclear reactor, there is no transition from the start state unless there is a gas leak. In case of a gas leak, the sensors determine the particle level. If the particle level is low, then no alarm is raised, but in if the particle level is high, the state is changed and the sensors trigger a signal which initializes automatic shower and at the same time raises the alarm to make a transition to the “control room server” state passing on the further functioning to the control room sensors which transmit emer

gency signals to the security team, fire brigade and the technician. Any time during the event, the failure of the sensors to perform its task automatically triggers the backup sensors. This complicated procedure is much easier to understand with the help of FSM.

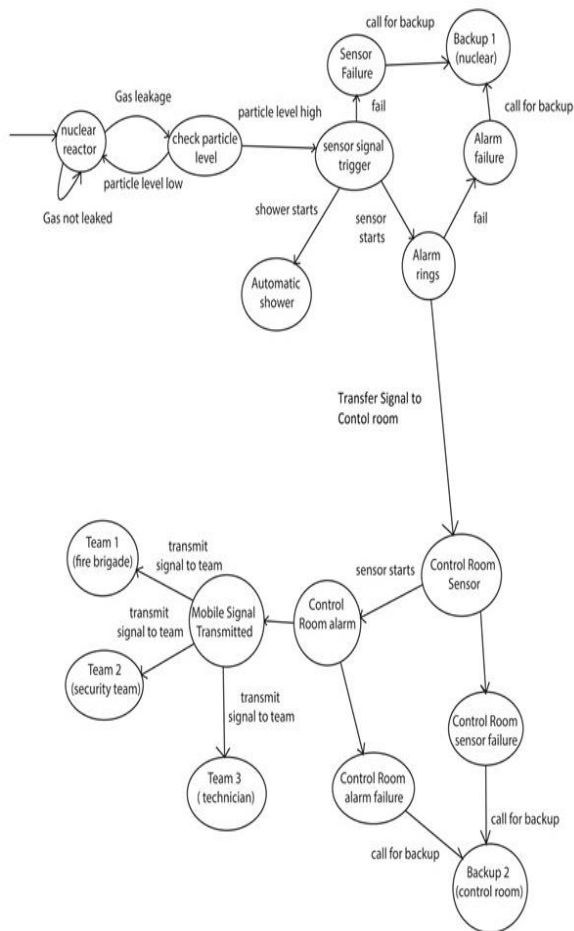


Fig.7. FSM for Sensor Dust

**Ankle Monitor**

As the name suggests, ankle monitor is mainly used for the purpose locating and searching an individual. This setup involves parole officer, parolee, server room and local police as the states for the FSM. A radius is set for the parolee to be able to move in. Any movement outside the radius or he tries to break the ankle monitor circuit, triggers a transition from state parolee to the state server room and it sets off an alarm in the server room. Server room first informs the parolee to back out and also alerts his parole officer. In case of non-cooperation of the parolee local police is called in by the server room and the parolee is arrested. Also, if the parolee is found anywhere near a crime scene or taking part in some illegal actions, etc. he gets arrested. Then the court decides based on the severity of his nuisance whether to leave him on parole or to get him back to prison thus determining the transitions from the state “court”. In the end, keeping parolee’s good conduct (if any) or at the time when his supposed parole time is coming to halt the court assesses the parolee. The court can either release him or can make him continue the

parole time both of which are transitions to a final state. The FSM for this implementation is shown in Fig.8.

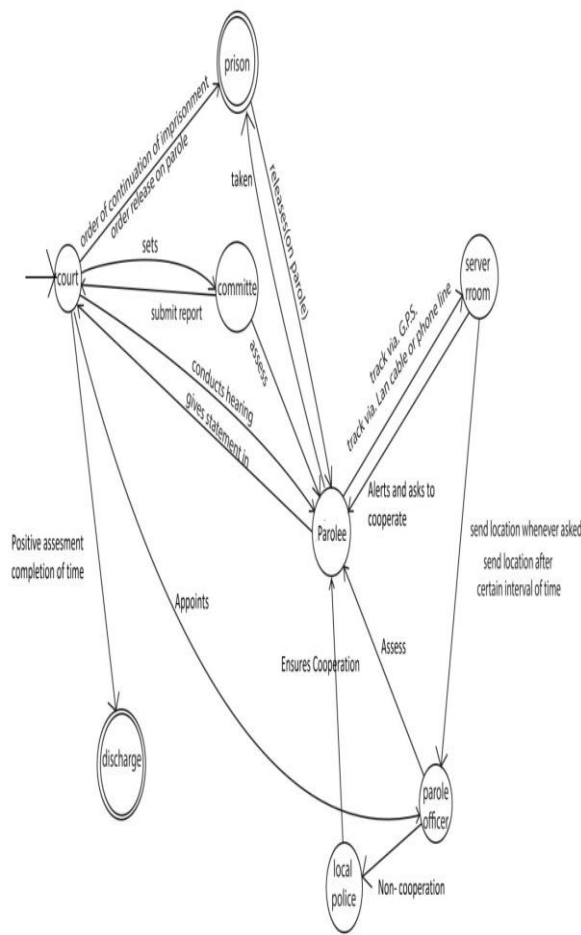


Fig.8. FSM for Ankle Monitor

After a discussion about application of FSM in the field of networking, in the following sub section the implementation and working of palm operating system using FSMs, has been presented.

**3.4 Palm Operating System**

The Palm webOS is based on the Linux 2.6 kernel whose graphical user interface is designed for use on the devices with touchscreens. In this article, two keys components of Palm OS namely, the booting sequence and the user interface have been modeled using FSM.

**Booting sequence of Palm OS**

The booting sequence is a crucial component of palm OS because it initiates the installation of the kernel onto the main memory. The whole sequence of steps involved in the booting of palm OS is depicted using FSM in Fig. 9.

The first step in the booting sequence is loading the kernel which is marked by the transition from the start state to the state “Linux kernel loaded”. After the kernel is loaded, then the nova installer is loaded into the memory, then finally the initial ramdisk is loaded into the temporary memory to finish the booting sequence. In case

the initial ramdisk is not found or the kernel is not loaded properly then the sequence enters into a trap state which would require re-booting the OS from the start.

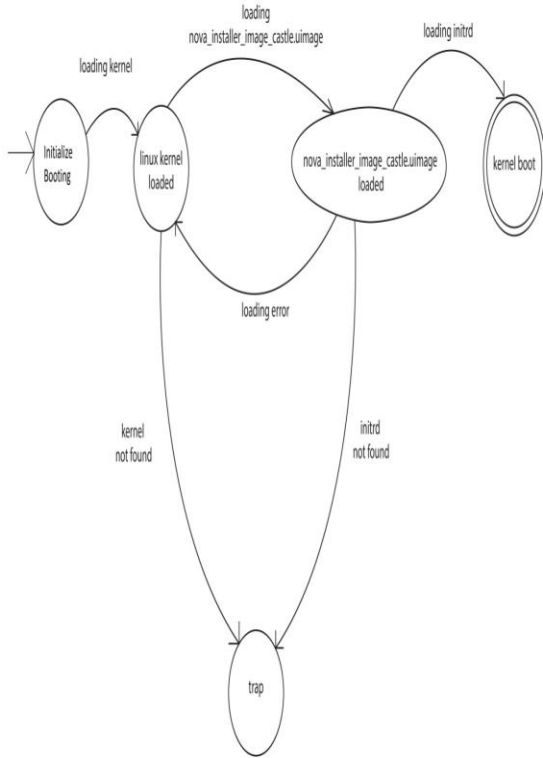


Fig.9. FSM for Booting Sequence of Palm

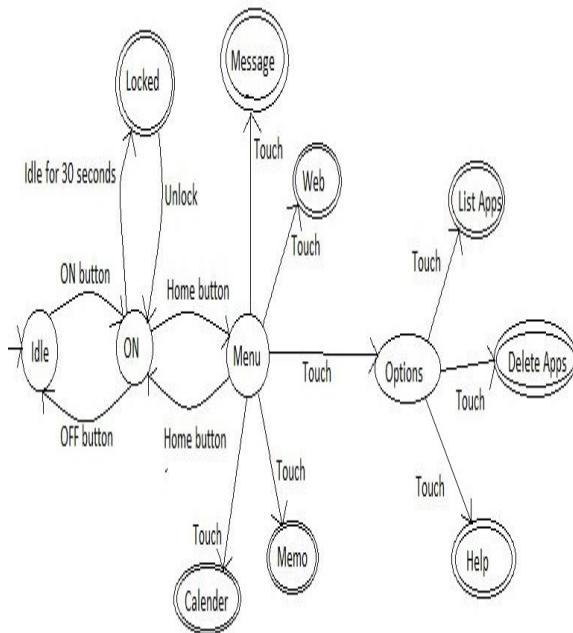


Fig.10. FSM for Palm OS user Interface

User Interface of Palm OS

The user never interacts with the core OS, instead the user only interacts with the application and the user interface provide by the OS. Fig.10. depicts the FSM for the various options available in this interface.

Beginning from the start state, the user first switches on the device on reach the ‘ON’ state. Hitting the home button on this state opens the menu for the user. The user can now touch on the desired option to open the application.

After applying FSMs in the domains of Operating Systems, networking and Web Design, in the following sub section the final application of FSM modeling for our paper, satellite launching and image handling procedures have been described which are highly critical components of satellite simulation.

3.5 Satellite Simulation

Satellite simulation is the integration of procedures ranging from construction to deployment and working of a satellite. Satellite launching and image handling within a satellite are two key components of satellite simulation which have been depicted with the help of FSM in this article.

Satellite Launching

Satellite launching is an important procedure for any satellite simulation, thus it needs to be accurate and easy to test. Therefore we have used FSM to model the steps involved in satellite launching. Fig.11. depicts the various stages, testing and calculations involved in satellite launching.

To begin with the procedure, first the fuel tank is filled, and then the ignition is turned on to reach the state where the satellite ready to take off. In case if of a technical problem or bad weather, there is a transition to the “vehicle waiting” state where the satellite is kept waiting until the issue is resolved. The launching sequence ends when the satellite procures the speed of 11.3 km/s and enters into the geo orbit. The satellite is then launched denoting the final state for the procedure.

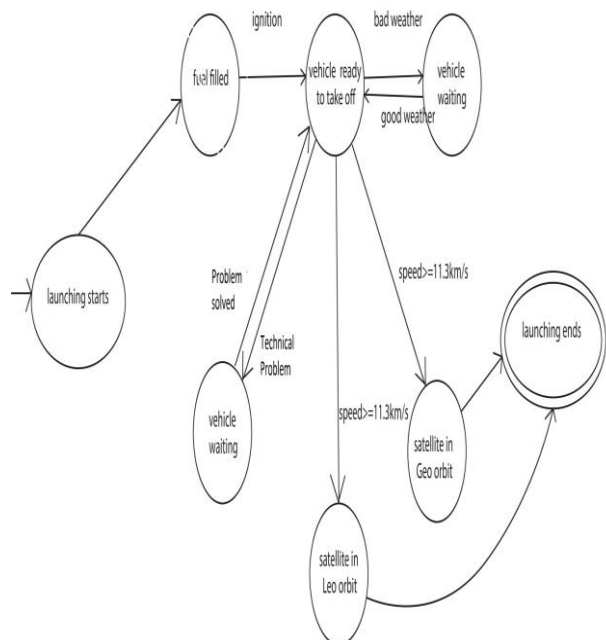


Fig.11. FSM for Satellite Launching

### Image Handling

One of the primary objectives of any satellite is to capture images and send them to the ground station for further processing. Thus the procedure for image handling is vital. Fig.12. demonstrates the various steps involved in image handling by a satellite.

Initially at the start state, the satellite is ready to capture an image. Once an image is clicked, depending on the quality of the image, there is a transition to any one of the several possible states. The image is processed and queued for sending. Transition to the final state is made when the image is sent to the ground station.

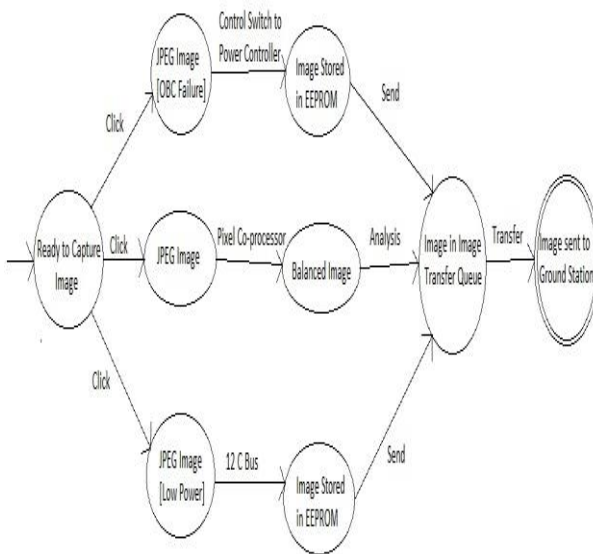


Fig.12. FSM for Image Handling

### IV. CONCLUSION

In this article, Finite state machine has been used to model various applications over multiple domains. FSMs have the power to make an application easier to understand, implement, debug, verify and modify. FSM can be programmed in any languages starting from low level to current generation high level languages, therefore its application cannot be limited to computational task, but can be applied to medical sciences, social media network. As from this article, it can be seen that FSM can be used to effectively model a tourism voice site where users can access a network via speech signals, also FSM can be utilized to model a robust information transfer to assist drivers with GPS navigation and accident reporting. In addition to this FSMs are also efficient to depict the functionalities of sensor dust and ankle monitor. It is also evident that FSM can effectively describe the booting sequence as well as the user interface of palm webOS and can be instrumental in depicting the steps involved in satellite launching and image handling by a satellite. In all, we have successfully complied a single source document pertaining with the application of finite state machines in scattered areas of science and management.

### REFERENCES

- [1] M. Ali Qureshi, Abdul Aziz, and S. Hammad Raza, "A Verilog Model of Adaptable Traffic Control System Using Mealy State Machines", International Journal of Computer and Electrical Engineering, Vol. 4, No. 3, June 2012.
- [2] Noman Sohaib Qureshi, Hassan Mushtaq, Muhammad Shehzad Aslam, Muhammad Ahsan, Mohsin Ali and Muhammad Aqib Atta, "Computing Game Design with Automata Theory", International Journal of Multidisciplinary Sciences and Engineering, Vol. 3, No. 5, May 2012.
- [3] Pengyu Hong, Matthew Turk, Thomas S. Huang, "Gesture Modeling and Recognition using Finite State Machine", Microsoft Research.
- [4] Ana Monga, Balwinder Singh, "Finite State Machine based Vending Machine Controller with Auto-Billing Features", International Journal of VLSI design & Communication Systems (VLSICS) Vol.3, No.2, April 2012.
- [5] Andrei Drumea, Camelia Popescu, "Finite State Machines and their applications in software for industrial control", 27th International Spring Seminar on Electronics Technology.
- [6] Vipin saxena, Santosh Kumar, "Validation of UML Class Model through Finite-State Machine", International Journal of Computer Applications (0975 – 8887) Volume 41– No.19, March 2012.
- [7] Eric Gribko, UC Davis, "Applications of Deterministic Finite Automata", ECS 120, spring 2013.
- [8] Deepak Chenthati, Supriya Vaddi, Hrushikesh Mohanty, Avula Damodaram, "Verification of web services modeled as finite state machines", Fourth Asia International Conference on Mathematical/Analytical Modelling and Computer Simulation (AMS), 2010.
- [9] Maheswari S, Justus Selwyn, "Assessing the Behaviour of Web Services using Finite States", International Journal of Information Engineering and Electronic Business, Vol.7, No.4, July 2015.
- [10] G.Jose Moses, P.Suresh Varma, N.Supriya, G.Naga Satish, "Security Aspects and Challenges in Mobile Adhoc Networks", International Journal of Computer Network and Information Security (IJCNIS), Vol.4, No.6, June 2012.
- [11] Qingqiang Yang, Wenxiong Kang, "General Research on Image Segmentation Algorithms", International Journal of Image, Graphics and Signal Processing (IJIGSP), Vol.1, No.1, Oct. 2009.

### Author's Profiles



**Ms. Kavita Pandey** did B.Tech in Computer Science from M.D. University in year 2002 and M.Tech. in Computer Science from Banasthali Vidyapith University in year 2003. She is pursuing PhD from IIIT, Noida. She is working as an Assistant Professor in IIIT, Noida. She is currently pursuing Ph.D. from IIIT, Noida. Her current research interests include Mobile Adhoc Networks, Vehicular Adhoc Networks, Optimization Techniques and Network Security.



**Mr. Dhruv** is 4<sup>th</sup> year student of Computer science and Engineering at Jaypee Institute of Information Technology, Noida. His research interests are Machine Learning, Theory of computation.



**Mr. Sumit** is 4<sup>th</sup> year student of Computer science and Engineering at Jaypee Institute of Information Technology, Noida. His research interests are Machine Learning, Theory of computation.



**Ms. Vrishti** is 4<sup>th</sup> year student of Computer science and Engineering at Jaypee Institute of Information Technology, Noida. Her research interests are Mobile Adhoc networks and Pattern recognition.