# **Varys – The Game of Automata!**

**Objective:**

Varys, The game of Automata, in which we created an application that can navigate an entire application autonomously and provide the complete application flow along with various combination of data and property information of every screen labels. With the help of this tool, we can minimize manual efforts of doing same thing repeatedly.

**Scope**:

Automata theory is a branch of computer instructions that deals with designing abstract self-propelled computing devices that follow a predetermined sequence of operations automatically.

* Any manual task in system (Web or Desktop) is a set of instruction that works sequentially. Capturing all this series of manual task and making it in a reusable, in the form of automata.
* Any manual task needs human intervention, to validate any operation, i.e., to visualize or to make next decision. Adding this functionality to automata can remove human intervention as well as added valuable validations.

**Design Approach:**

We have developed an application that can capture events of each task which is performed either on web or on desktop as well as captures images which can be used to validate the process while execution.

Application consist of three major functions –

* Record and Validate.
* Modify and Create automata.
* Run automata.

1. **Record and Validate**: This two functions is used to capture events and validation images of task.
   1. **Record:** It will record all the clicks and key press event of the task.
   2. **Validate:** In the middle of your task, if you need anything to validate to go to next step of task we use this function. It will take the snap of your selected screen. This snap can be used later for validation.
2. **Modify and Create automata:** After recording “events” and taking validation snaps, we need to create automata. Here, automata is a combination of events and validation snaps. Recorded events and snaps than combined to form final automata.
3. **Run automata:** Once the events and validation images are combined and final automata is get created, we can run this automata.

**Engine work process:**

* + **Events**
    - **Click (Left/Right)**
    - **Key press (Single/Multiple/Hotkeys)**
    - **Validate**
  + **PSEUDOCODE:**

BEGIN

FOR Event in EventsList:

INPUT Events

IF Event == Click

PERFORM **Click\_Event()**

ELSE IF Event == Key Press

PERFORM **Key\_Press\_Event()**

ELSE IF Event == Validate

WAIT UNTIL TIMEOUT:

REPEAT

PERFORM **Validate\_Image()**

IF Validate\_Image == TRUE

Continue

* + **Click\_Event():** Performs recorded click events.
  + **Key\_Press\_Event():** Performs recorded Key events
  + **Validate\_Image():** This function will check the snap taken for the validation in the current screen. It will wait until that image appear. If image appears in the screen than it will resume the task, and go to next event to get executed. If image doesn’t appear than, it will wait to get timeout.

Record

Validate

Merge & Create Automata

Play

**Fig: Work process flow chart**

**Input and output:**

We are using below inputs/outputs while designing and developing the Automata:

**Input**: Mouse click, keyboard buttons, and Image Snaps.

**Output**: Automated sequence of events.

**Benefits:**

Below are the benefits of this Automata:

1. Manual intervention is less.
2. It enables record and playback for web applications/desktop application task and can run multiple scripts across various browsers.
3. Reusability - When you have your events prepared using automata application, they are saved for the future requirements. So, you can utilize as many times as you want especially for automating repeated task.
4. Automata uses less hardware resources.
5. It’s a portable application.
6. It offers a user-friendly interface that helps and execute application easily and effectively.
7. Less configuration than other available tools.
8. Easy usability for anyone.

**Novelty of Design:**

1. Easy configuration, not like other tools which require prior knowledge of coding and languages.
2. Validation based on visual images, making more effective for decision making.
3. Simple and easy to use.
4. Same automata can be used for other systems.
5. For this automata, its design pattern is quite different from other applications like you don’t need to write any test scripts for testing something. You just need to record all the events by clicking Record tab that you want to perform over and over. That will get recorded in Automata and when you need to do you can simply play around with that using Play tab.

**Extensibility of Design:** As events are essential part of any computer related task, it can be easily extended to desktop as well as web application or the combination of both.

**Future Scope:**

1. With the current design we can automate simple task like login to page and raising tickets to ITSM tools.
2. With the help of this tool, we can also automate major task like installation of ignio without letting any human intervention.
3. We can also configure it for the testing and assurance.
4. We can have better and effective validation logic using more research on image validation.
5. We can include better tracking of events and managing automata, with the help of UI. Creating and modifying automata can be done using UI.
6. Right now, we have developed application based on windows task, in future we can accommodate for other OS as well.

**Conclusion:**

This would definitely help you to improve your processes with the help of the leading automated application.