Claims:

1. A method for mapping inputs to a software application, comprising implementing on one or more processors, the steps of:

receiving a keymap that correlates a first set of input events capable of being generated through an input device, with a second set of input events that are recognized by the software application, wherein the second set of input events includes one or more input events that are not included within the first set of input events;

receiving from an input device within the first set of input devices, a first data signal associated with a first input event within the first set of input events;

identifying based on the received keymap, a second input event within the second set of input events, wherein the second input event is mapped to the first input event by the received keymap;

generating a second data signal, wherein the second data signal is recognizable by the software application as corresponding to the second input event; and

implementing a state change in the software application based on the second data signal;

wherein the method further comprises:

generating on a display controlled by the software application:

a first set of graphic objects defined by or selected based on a state of the software application; and

a second set of graphic objects overlaid upon one or more graphic objects within the first set of graphic objects, wherein a specific graphic object within the

second set of graphic objects comprises a visual representation of an input control that is enabled to generate the first input event within the first set of input events;

and wherein:

the second input event within the second set of input events is a touchpress second input event initiated through a touchscreen interface; and

the specific graphic object within the second set of graphic objects is overlaid on the touchscreen display, at a defined position relative to a region of the display screen that has been defined by the software application for the purposes of initiating the touchpress second input event within the second set of input events, and which touchpress second input event is correlated by the received keymap with the first input event within the first set of input events.

2. The method as claimed in claim 1, wherein the state change implemented in the software application instantiates a corresponding state change in an output device controlled by the software application.

3. The method as claimed in claim 2, wherein the output device is any of a display, speaker, headphone, visual feedback device, audio feedback device, illumination device, tactile feedback device, haptic feedback device, or sensory feedback device.

4. The method as claimed in claim 1, wherein (i) the first set of input events are not recognized by the software application, or (ii) ~~the~~ software response that is associated with the first set of input events by the software application is different from ~~the~~ software response that is associated with the second set of input events by the software application.

5. The method as claimed in claim 1, wherein:

the software application is executed within an instance of a guest operating system that is implemented within a host operating system, wherein the guest operating system and the host operating system are different from each other;

the first set of input events are recognizable by a hardware device driver compatible with and executed within the host operating system; and

the second set of input events are (i) recognizable by the software application being executed within the instance of the guest operating system or (ii) recognized by a hardware device driver within an operating system kernel of the guest operating system; or (iii) recognized by the guest operating system.

6. The method as claimed in claim 5, wherein the second data signal is transmitted to the software application within the guest operating system from a host operating system application software that instantiates and controls the instance of the guest operating system that is being implemented within the host operating system.

7. The method as claimed in claim 1, ~~further comprising:~~

~~generating on a display controlled by the software application:~~

~~a first set of graphic objects defined by or selected based on a state of the software application; and~~

~~a second set of graphic objects overlaid upon one or more graphic objects within the first set of graphic objects,~~ wherein one or more graphic objects within the second set of graphic objects represent one or more input events within the first set of input events.

8. The method as claimed in claim 7, wherein the second set of graphic objects comprises:

a visual representation of one or more data elements retrieved from the received keymap; or

a visual representation of one or more input controls that are enabled to generate input events within the first set of input events.

9. The method as claimed in claim 8, wherein:

~~one or more~~ the input events within the second set of input events are touchpress events initiated through ~~a~~ the touchscreen interface~~; and~~

~~a graphic object within the second set of graphic objects is overlaid on the touchscreen display, at a defined position relative to a region of the display screen that has been defined by the software application for the purposes of initiating a touchpress event within the second set of input events, and which touchpress event is correlated by the received keymap with an input event within the first set of input events that corresponds to said graphic object~~.

10. The method as claimed in claim 1, wherein:

the received keymap is selected from among a plurality of keymaps;

wherein the selection of the received keymap is responsive to detection of one or more predefined state parameters, and

the one or more predefined state parameters are hardware or software state parameters.

11. The method as claimed in claim 10, wherein the detected one or more predefined state parameters comprise any of a visual display state parameter, an audio state parameter, a state parameter corresponding to data stored in memory, a state parameter corresponding to data received over a network, a state parameter corresponding to input received from a hardware device, or a state parameter corresponding to any computing system or computing system environment associated with execution of the software application.

12. The method as claimed in claim 10, wherein the software application is executed within an instance of a guest operating system that is being implemented within a host operating system, wherein:

the guest operating system and the host operating system are different from each other;

the first set of input events are recognizable by a hardware device driver compatible with and executed within the host operating system;

the second set of input events are (i) recognizable by the software application being executed within the instance of the guest operating system or (ii) recognized by a hardware device driver within an operating system kernel of the guest operating system; or (iii) recognized by the guest operating system; and

one or more of the detected state parameters are detected based on data, a request or an instruction received at, or from, any of:

an application framework layer within the instance of the guest operating system;

a library layer within the instance of the guest operating system;

a hardware abstraction layer within the instance of the guest operating system;

an application program executed within an application program layer in the guest operating system;

the host operating system through a data channel established between the instance of the guest operating system and a host operating system application software that instantiates and controls the instance of the guest operating system that is being implemented with the host operating system; or

the host operating system application software from a host operating system device driver.

13. A system for mapping inputs to a software application, the system comprising a processor implemented computing system configured for:

receiving a keymap that correlates a first set of input events capable of being generated through an input device, with a second set of input events that are recognized by the software application, wherein the second set of input events includes one or more input events that are not included within the first set of input events;

receiving from an input device within the first set of input devices, a first data signal associated with a first input event within the first set of input events;

identifying based on the received keymap, a second input event within the second set of input events, wherein the second input event is mapped to the first input event by the received keymap;

generating a second data signal, wherein the second data signal is recognizable by the software application as corresponding to the second input event; and

implementing a state change in the software application based on the second data signal

wherein the processor implemented computing system is further configured for:

generating on a display controlled by the software application:

a first set of graphic objects defined by or selected based on a state of the software application; and

a second set of graphic objects overlaid upon one or more graphic objects within the first set of graphic objects, wherein a specific graphic object within the

second set of graphic objects comprises a visual representation of an input control that is enabled to generate the first input event within the first set of input events;

and wherein:

the second input event within the second set of input events is a touchpress second input event initiated through a touchscreen interface; and

the specific graphic object within the second set of graphic objects is overlaid on the touchscreen display, at a defined position relative to a region of the display screen that has been defined by the software application for the purposes of initiating the touchpress second input event within the second set of input events, and which touchpress second input event is correlated by the received keymap with the first input event within the first set of input events.

14. The system as claimed in claim ~~12~~13, wherein the state change implemented in the software application instantiates a corresponding state change in an output device controlled by the software application.

15. The system as claimed in claim 14, wherein the output device is any of a display, speaker, headphone, visual feedback device, audio feedback device, illumination device, tactile feedback device, haptic feedback device, or sensory feedback device.

16. The system as claimed in claim 13, wherein (i) the first set of input events are not recognized by the software application, or (ii) ~~the~~ software response that is associated with the first set of input events by the software application is different from ~~the~~ software response that is associated with the second set of input events by the software application.

17. The system as claimed in claim 13, wherein:

the software application is executed within an instance of a guest operating system that is implemented within a host operating system, wherein the guest operating system and the host operating system are different from each other;

the first set of input events are recognizable by a hardware device driver compatible with and executed within the host operating system; and

the second set of input events are (i) recognizable by the software application being executed within the instance of the guest operating system or (ii) recognized by a hardware device driver within an operating system kernel of the guest operating system; or (iii) recognized by the guest operating system.

18. The system as claimed in claim 17, wherein the second data signal is transmitted to the software application within the guest operating system from a host operating system application software that instantiates and controls the instance of the guest operating system that is being implemented within the host operating system.

19. The system as claimed in claim 13, ~~configured for:~~

~~generating on a display controlled by the software application:~~

~~a first set of graphic objects defined by or selected based on a state of the software application; and~~

~~a second set of graphic objects overlaid upon one or more graphic objects within the first set of graphic objects,~~ wherein one or more graphic objects within the second set of graphic objects represent one or more input events within the first set of input events.

20. The system as claimed in claim 19, wherein the second set of graphic objects comprises:

a visual representation of one or more data elements retrieved from the received keymap; or

a visual representation of one or more input controls that are enabled to generate input events within the first set of input events.

21. The system as claimed in claim 20, wherein:

~~one or more~~ the input events within the second set of input events are touchpress events initiated through ~~a~~ the touchscreen interface~~; and~~

~~a graphic object within the second set of graphic objects is overlaid on the touchscreen display, at a defined position relative to a region of the display screen that has been defined by the software application for the purposes of initiating a touchpress event within the second set of input events, and which touchpress event is correlated by the received keymap with an input event within the first set of input events that corresponds to said graphic object~~.

22. The system as claimed in claim ~~12~~13, wherein:

the received keymap is selected from among a plurality of keymaps;

wherein the selection of the received keymap is responsive to detection of one or more predefined state parameters, and

the one or more predefined state parameters are hardware or software state parameters.

23. The system as claimed in claim 22, wherein the detected one or more predefined state parameters comprise any of a visual display state parameter, an audio state parameter, a state parameter corresponding to data stored in memory, a state parameter corresponding to data received over a network, a state parameter corresponding to input received from a hardware device, or a state parameter corresponding to any computing system or computing system environment associated with execution of the software application.

24. The system as claimed in claim 22, wherein the software application is executed within an instance of a guest operating system that is being implemented within a host operating system, wherein:

the guest operating system and the host operating system are different from each other;

the first set of input events are recognizable by a hardware device driver compatible with and executed within the host operating system;

the second set of input events are (i) recognizable by the software application being executed within the instance of the guest operating system or (ii) recognized by a hardware device driver within an operating system kernel of the guest operating system; or (iii) recognized by the guest operating system; and

one or more of the detected state parameters are detected based on data, a request or an instruction received at, or from, any of:

an application framework layer within the instance of the guest operating system;

a library layer within the instance of the guest operating system;

a hardware abstraction layer within the instance of the guest operating system;

an application program executed within an application program layer in the guest operating system;

the host operating system through a data channel established between the instance of the guest operating system and a host operating system application software that instantiates and controls the instance of the guest operating system that is being implemented with the host operating system; or

the host operating system application software from a host operating system device driver.

25. A computer program product for mapping inputs to a software application, the computer program product comprising a non-transitory computer usable medium having computer readable program code embodied therein, the computer readable program code comprising instructions for implementing at a processor, the steps of:

receiving a keymap that correlates a first set of input events capable of being generated through an input device, with a second set of input events that are recognized by the software application, wherein the second set of input events includes one or more input events that are not included within the first set of input events;

receiving from an input device within the first set of input devices, a first data signal associated with a first input event within the first set of input events;

identifying based on the received keymap, a second input event within the second set of input events, wherein the second input event is mapped to the first input event by the received keymap;

generating a second data signal, wherein the second data signal is recognizable by the software application as corresponding to the second input event; and

implementing a state change in the software application based on the second data signal;

generating on a display controlled by the software application:

a first set of graphic objects defined by or selected based on a state of the software application; and

a second set of graphic objects overlaid upon one or more graphic objects within the first set of graphic objects, wherein a specific graphic object within the

second set of graphic objects comprises a visual representation of an input control that is enabled to generate the first input event within the first set of input events;

and wherein:

the second input event within the second set of input events is a touchpress second input event initiated through a touchscreen interface; and

the specific graphic object within the second set of graphic objects is overlaid on the touchscreen display, at a defined position relative to a region of the display screen that has been defined by the software application for the purposes of initiating the touchpress second input event within the second set of input events, and which touchpress second input event is correlated by the received keymap with the first input event within the first set of input events.