WHAT IS CLAIMED IS:

1. A computer system comprising:

a memory resource storing instructions; and

one or more processors using the instructions stored in the memory resource to perform operations including:

providing a design interface to render at least a portion of a graphic design; and

providing a code interface to render a code representation for at least a portion of the graphic design, wherein providing the code interface includes executing one or more plugins to generate the code representation, the one or more plugins including at least a first plugin that is selectable by a user to configure the generation of the code representation in accordance with one or more preferences.

2. The computer system of claim 1, wherein the first plugin supports a plurality of programming languages for the code representation.

3. The computer system of claim 2, wherein the one or more preferences include a selection of a first programming language from the plurality of programming languages.

4. The computer system of claim 3, wherein the first plugin converts an internal representation of at least some of the graphic design into the first programming language, the first program language being different from the internal representation.

5. The computer system of claim 1, wherein the first plugin retrieves data from one or more external repositories and integrates the data into the code representation.

6. The computer system of claim 1, wherein the graphic design is structured to define a plurality of layers, wherein each layer of the plurality of layers corresponds to one of an object, a group of objects, or a type of object; and

wherein executing the one or more plugins is performed in response to the user selecting a layer of the plurality of layers using one of the design interface or the code interface.

7. The computer system of claim 1, wherein executing the one or more plugins includes causing the code interface to identify an asset as part of the code representation.

8. The computer system of claim 4, wherein executing the one or more plugins includes inferring a layout rule amongst one or more layers of the graphic design.

9. The computer system of claim 8, wherein the layout rule specifies a spatial relationship amongst multiple layers of the graphic design.

10. The computer system of claim 8, wherein the layout rule specifies a spacing between multiple layers of the graphic design.

11. The computer system of claim 1, wherein the first plugin is a third-party plugin.

12. The computer system of claim 1, wherein the first plugin has read-only access to the graphic design.

13. A method being implemented by one or more processors and comprising:

providing a design interface to render at least a portion of a graphic design; and

providing a code interface to render a code representation for at least a portion of the graphic design, wherein providing the code interface includes executing one or more plugins to generate the code representation, the one or more plugins including at least a first plugin that is selectable by a user to configure the generation of the code representation in accordance with one or more preferences.

14. The method of claim 13, wherein the first plugin supports a plurality of programming languages for the code representation.

15. The method of claim 14, wherein the one or more preferences include a selection of a first programming language from the plurality of programming languages.

16. The method of claim 15, wherein the first plugin converts an internal representation of at least some of the graphic design into the first programming language, the first program language being different from the internal representation.

17. The method of claim 13, wherein the first plugin retrieves data from one or more external repositories and integrates the data into the code representation.

18. The method of claim 13, wherein the graphic design is structured to define a plurality of layers, wherein each layer of the plurality of layers corresponds to one of an object, a group of objects, or a type of object; and

wherein executing the one or more plugins is performed in response to the user selecting a layer of the plurality of layers using one of the design interface or the code interface.

19. The method of claim 13, wherein executing the one or more plugins includes causing the code interface to identify an asset as part of the code representation.

20. A non-transitory computer-readable medium that stores instructions, executable by one or more processors, to cause the one or more processors to perform operations including:

providing a design interface to render at least a portion of a graphic design; and

providing a code interface to render a code representation for at least a portion of the graphic design, wherein providing the code interface includes executing one or more plugins to generate the code representation, the one

or more plugins including at least a first plugin that is selectable by a user to configure the generation of the code representation in accordance with one or more preferences.